ספר לימוד

SAS

Statistical Analysis System

גיא הוכמן ואלדד יחיעם

הוצאת
הוד-עמי
ספרי מחשבים

שמות מסחריים

שמות המוצרים והשירותים המוזכרים בספר הינם שמות מסחריים רשומים של החברות שלהם. הוצאת הוד-עמי והמחברים עשו כמיטב יכולתם למסור מידע אודות השמות המסחריים המוזכרים בספר זה ולציין את שמות החברות, המוצרים והשירותים. שמות מסחריים רשומים (registered trademarks) המוזכרים בספר צוינו בהתאמה.

הודעה חשובה

ספר זה מיועד לתת מידע אודות מוצרים שונים. נעשו מאמצים רבים לגרום לכך שהספר יהיה שלם ואמין ככל שניתן, אך אין משתמעת מכך אחריות כלשהי.

תוכן הספר וההפניות לספרים, לתוכנות, לאתרים ולמקורות מידע המוזכרים בו מסופקים "כמו שהם (as is)". השימוש בכל אלה הוא על אחריותו הבלעדית של המשתמש. הוצאת הוד-עמי והמחברים אינם אחראים כלפי יחיד או ארגון עבור כל אובדן או נזק ישיר או עקיף, אשר ייגרם, אם ייגרם, מהשימוש בספר ו/או בתוכנות ו/או באתרים ו/או כל מקור מידע או תוכנה המוזכרים בספר, ובכלל זה (רשימה חלקית): הפרעה במתן שירות, אובדן מידע, אובדן זמן, אובדן רווח וכד'.

המשתמש רשאי להשתמש בתוכנות המוזכרות בספר ו/או לפנות לאתרים ו/או למקורות מידע אחרים על אחריותו. כל אלה הם בבעלות ובאחריות החברות המייצרות, משווקות ומציגות אותם. הוד-עמי והמחברים אינם גובים תשלום עבור השימוש בתוכנות ובמידע ממקורות אחרים המוזכרים בספר. הוד-עמי והמחברים אינם מספקים תמיכה בהתקנה ו/או ההפעלה של התוכנות ו/או בגישה לאתרים ומידע אחר. מחלקת התמיכה בהוצאת הוד-עמי תגיש עזרה רק עבור מקרים של אי בהירות של הסבר בספר או שיבוש דפוס. כל שאלה לגבי תוכנה ו/או אתר ו/או מקור מידע כלשהם יש להפנות אל מפתחי/יוצרי/משווקי התוכנה ו/או אל בעלי האתרים ו/או מקורות המידע.

הוצאת הוד-עמי והמחברים עשו כל מאמץ שתוכן הספר יהיה אמין ושלם. עם זאת, ההוצאה והמחברים אינם טוענים לאמינות ולשלמות של התכנים המוצגים בספר זה, ובמיוחד דוחים כל אחריות, ובכלל זה טענה להתאמה של הנאמר בספר למקרה ספציפי כלשהו. לא ניתן ליצור או להרחיב אחריות על ידי מידע שיווקי ו/או פרסומי כלשהו. ייתכן שההצעות ו/או ההמלצות הניתנות בספר לא יתאימו לכל מצב ומקרה. הספר משווק ונמכר תוך הבנה שההוצאה והמחברים אינם מספקים שירותים שונים הכרוכים בשימוש בספר, אלא לשם הבנת הכתוב ותיקון שיבושי לשון. לקבלת שירות מקצועי יש לפנות אל בעלי המקצוע בתחום. הן ההוצאה והן המחברים אינם אחראים לכל אובדן או נזק ישיר או עקיף, אשר ייגרם, אם ייגרם, מהשימוש בספר ו/או בתוכנות ו/או באתרים ו/או כל מקור מידע או תוכנה המוזכרים בספר. אין בכוונת ההוצאה ו/או המחברים להמליץ או להעדיף תוכנה ו/או אתר ו/או מקור מידע כלשהם. רק המשתמש הוא שיחליט כיצד לנהוג על פי המוצג בספר. המשתמש צריך להיות ער לעובדה שאתרי האינטרנט הינם דינמיים ועלולים להיסגר, לשנות את התכנים שלהם וכד'. ההוצאה והמחברים אינם אחראים לשינויים אשר עלולים לחול באתרים המוזכרים בספר, ועל כן להיות שונים ממה שהוצג בספר.

אין לעשות שימוש מסחרי ו/או להעתיק, לשכפל, לצלם, לתרגם, להקליט, לשדר, לקלוט ו/או לאחסן במאגר מידע בכל דרך ו/או אמצעי מכני, דיגיטלי, אופטי, מגנטי ו/או אחר - בחלק כלשהו מן המידע ו/או התמונות ו/או האיורים ו/או כל תוכן אחר הכלולים ו/או שצורפו לספר זה, בין אם לשימוש פנימי או לשימוש מסחרי. כל שימוש החורג מציטוט קטעים קצרים במסגרת של ביקורת ספרותית אסור בהחלט, אלא ברשות מפורשת בכתב מהמוציא לאור.

עריכה ועיצוב: **גיא הוכמן**
עיצוב עטיפה: **גיא הוכמן**
צילום עטיפה: **יובל טבול**

תודה לאריאל תלפז על הערות והארות

לשם שטף הקריאה כתוב ספר זה בלשון זכר בלבד. ספר זה מיועד לגברים ונשים כאחד ואין בכוונתנו להפלות או לפגוע בציבור המשתמשים/ות.

(C)
כל הזכויות שמורות

הוצאת הוד-עמי בע"מ
ת.ד. 6108 הרצליה 46160
טלפון: 09-9564716 פקס: 09-9571582
info@hod-ami.co.il
www.hod-ami.co.il

הודפס בישראל נובמבר 2010

All Rights Reserved
HOD-AMI Ltd.
P.O.B. 6108, Herzliya
ISRAEL, 2010

מסת"ב 978-965-361-408-6 ISBN

קבצי קוד המקור

את קבצי קוד המקור (התוכניות) ניתן להוריד מאתר האינטרנט של הוצאת הוד עמי:
www.hod-ami.co.il

מצא את הספר באתר ואת הלינק "**קוד מקור**" להורדת הקבצים. לחץ עליו ועקוב אחר ההוראות.
אם לא תגדיר אחרת, יועתקו הקבצים אוטומטית לדיסק שלך, לתיקיה זו:

C:\HodAmiBooks\59444\

תוכל לבחור בעת ההתקנה בכל תיקיה אחרת.

בדרך כלל הפעולות שיש לבצע לאחר הלחיצה על הלינק: לחיצה על הפעל, לחיצה על הפעל, לחיצה על UnZip, לחיצה על OK ו-Close.

כדי להשלים את כל התרגילים בספר תצטרך את הדברים הבאים:

חשוב: הקובץ **אינו מכיל את התוכנה SAS או כל תוכנה אחרת.** את התוכנה יש לרכוש ולהתקין **לפני** תחילת השימוש בספר זה.

לאחר שתוריד את קבצי קוד המקור מאתר הוד-עמי הם יימצאו (אם לא שינית) בתיקיה 59444 שנמצאת תחת HodAmiBooks שנמצאת בכונן הראשי C.

תוכן עניינים

רשימת איורים

רשימת טבלאות

פרק 1
מבוא

אודות הספר

ספר ההדרכה הנוכחי מהווה מבוא מקיף לתכנות בסיסי ב-SAS (קליטת נתונים, יצירת נתונים חדשים) ולביצוע ניתוחים סטטיסטיים. הספר אינו מצריך ידע או ניסיון קודם בתכנות ב-SAS, והוא מאפשר למתכנתים חדשים ללמוד על יסודות התכנות ועל יישומים בצורה אינטראקטיבית ונוחה. במקביל, הספר מאפשר למתכנתים מנוסים לנצל מאפיינים מתקדמים של SAS ולשפר את היכולת שלהם ליצור פלטי SAS מותאמים אישית. הספר מחולק לנושאים וכולל תרגילים ותוכניות לדוגמא. עם זאת, הספר כן מצריך ידע סטטיסטי, אם כי ניתן ליישם את החומר הנלמד בספר זה גם בלי ידע מוקדם זה.

כיצד להתמצא בספר

הספר הנוכחי כולל ארבעה נושאים:

א. מבוא – כולל סקירה כללית של SAS
ב. DATA STEP – כולל כיצד להכניס נתונים ל-SAS ולתפעל אותם
ג. PROC STEP – כולל פרוצדורות נפוצות לניתוח סטטיסטיים ותפעול משתנים
ד. פונקציות מתקדמות – כולל פעולות כלליות של SAS שנועדו להגביר יעילות

כל נושא כולל מספר פרקים המהווים כל אחד בפני עצמו מרכיב תפקודי עצמאי (component – חלק קוד SAS עצמאי המהווה תוכנית ברת הרצה) של SAS. כל פרק מתחיל בהצגה כללית של המרכיב, הכוללת את המבנה הכללי של הקוד להפעלת מרכיב זה. במקרה שזה רלוונטי, הפרק כולל גם את הפלט הבסיסי ש-SAS מפיקה מהרצת הקוד של המרכיב. בהמשך, הפרק כולל הוראות ואופציות מתקדמות יותר של כל מרכיב, שנועדו לאפשר למשתמשים להתאים את הפעולות של המרכיב לצרכים הספציפיים שלהם.

באופן כללי, הספר מתחיל עם רמת פירוט גבוהה הכוללת הסברים מפורטים, תוך מתן דוגמאות ספציפיות, לכל מרכיב תפקודי. כדי למנוע עומס, ככל שאנחנו מתקדמים בספר רמת הפירוט יורדת ומרכיבים מסוימים (במיוחד כאלה שיש להם מבנה דומה למבנים אחרים שנידונו בתת פרקים קודמים) מוצגים ברמה הכללית, ללא דוגמאות ספציפיות. במקרה בו יש חפיפה מלאה באופן הכתיבה או היישום של מרכיבים מסוימים, ניתנת בגוף הטקסט הפנייה לדוגמאות או הסברים רלוונטיים, שנידונו יותר בהרחבה בחלקים קודמים. לכן, מומלץ כי הקורא המתחיל יעבור על הספר לפי הסדר, במטרה לצבור את הידע והניסיון המתאימים כדי להתמודד עם מרכיבים תפקודיים מורכבים יותר המוצגים בחלקים מתקדמים יותר של הספר.

לדוגמא, למעט במקרים בהם מוגדר אחרת, כאשר רוצים להגדיר מספר משתנים עליהם יישום מרכיב תפקודי, או כאשר רוצים להגדיר להוראה מסוימת של המרכיב מספר אפשרויות, יש לכתוב את המשתנים השונים אחד לאחר השני, כאשר המשתנים מופרדים על ידי רווחים. בתחילת הספר עובדה זו מפורטת, אולם בהמשך, אנחנו פשוט כותבים "רשימת משתנים" מתוך כוונה שהקורא כבר למד כי הדרך להכניס משתנים שונים (או אופציות שונות) היא על ידי כך שהמשתנים ברשימה יהיו מופרדים על ידי רווחים.

בנוסף, לאורך כל הספר, אנחנו משתמשים בסימנים מוסכמים המייצגים מונח או מושג מילולי. כדי להפיק את המירב מהספר, מומלץ ללמוד היטב סימנים אלה בטרם תתחילו לקרוא. הרשימה המלאה של הסימנים, כמו גם הפירוש שלהם, מוצגת בטבלה 1.

המלצות לקריאה יעילה

מאחר וחלק מההוראות והפקודות הזמינות ב-SAS עשויות להיות מאוד טכניות ומורכבות, למידתן והבנתן עשויות להיות משימה לא קלה בכלל, בעיקר למתכנתים חדשים, שרק מתחילים את דרכם בנבכי התוכנה. לכן, אנו ממליצים, בעיקר למשתמשים חדשים, לקרוא ספר הדרכה זה בשני שלבים, כפי שיפורט להלן.

בשלב הראשון, מומלץ להכיר את ההוראות הבסיסיות והפשוטות ביותר של SAS לקליטה של נתונים, המוצגות בהרחבה בפרקים הדנים ביצירת קבצי נתונים ב-SAS. בשלב זה לא מומלץ להיכנס לאפשרויות מתקדמות יותר של התוכנה, אלא להתמקד בעיקר בפעולות ההכרחיות והבסיסיות ביותר להתחלת העבודה. אותו הדין קיים גם לתפעול של קבצי נתונים, ולניתוח סטטיסטי של נתונים, שם מומלץ להתמקד במבנה הכללי של ההוראות, ולא להיכנס לכל האופציות הרבות והמורכבות הזמינות לכל סוג ניתוח. כדי להקל על משימה זו, הפרקים הדנים בפרוצדורות הסטטיסטיות של SAS מתחילים בהסבר כללי של הפרוצדורה ובדוגמא קונקרטית של קוד בסיסי ליישום הפרוצדורה, כמו גם את הפלט הבסיסי שנוצר כתוצאה מהרצה של קוד בסיסי זה. בשלב זה מומלץ גם להשתמש בתרגילים לתרגול עצמי הזמינים לקורא, בעיקר באלה שלא דורשים ידע מעמיק בתכנות, לשם יישום של החומר הנלמד.

בשלב הבא, ולאחר שנרכש כבר הידע הבסיסי הנדרש כדי לקלוט ולעבד קבצי נתונים, כמו גם לעבוד על קבצים אלה כדי להפיק נתונים סטטיסטיים בסיסיים שונים, מומלץ להתחיל להיכנס יותר אל הפרטים הטכניים שנועדו לספק ידע מקיף ונרחב יותר של האפשרויות הרבות הגלומות ב-SAS. בשלב זה מומלץ גם להתחיל להיכנס לתוך האופציות הרבות והמגוונות הזמינות למרבית הפרוצדורות הקיימות ב-SAS.

אנו מאמינים כי אופן עבודה זה יסייע לקורא "לשחות" בים הרחב שהוא עולם התכנות העשיר ש-SAS מציעה, בצורה הנוחה והיעילה ביותר, ויקל משמעותית את תהליך הלמידה של התוכנה.

כדי לסייע לקורא במשימה זו, כללנו בספר שני סוגים של טיפים:

1. טיפים לקריאה – טיפים אלה מסומנים על ידי הסימן  , והם נועדו לכוון את הקריאה לפרקים בסיסיים יותר, ולהרחיק את הקורא הלא מנוסה מפרקים טכניים ומורכבים שעשויים לפגוע בתהליך הלמידה הראשוני.
2. טיפים ממומחה – טיפים אלה מסומנים על ידי הסימן  , והם נועדו להפנות את תשומת הלב של הקורא להוראות או פקודות חשובות במיוחד, כמו גם להפנות את תשומת לבו לטעויות אופייניות בתכנות ב-SAS, בעיקר למתכנתים הנמצאים בתחילת דרכם.

קוראים שינהגו לפי טיפים אלה יוכלו למצוא בנקל את הדרך הנכונה והטובה ביותר לחלק את הקריאה של ספר זה לשני שלבים, ובכך יבטיחו למידה יעילה וטובה יותר של תכנות ב-SAS.

הסימן המוסכם	משמעות
<>	כל ביטוי המוכנס לתוך סוגריים משולשים מבטא מרכיב אופציונאלי בקוד, שאינו חובה, וניתן להשמיט אותו ללא פגיעה בתפקוד הבסיסי של המרכיב התפקודי
| (קו אנכי)	סימן זה מייצג את המילה "או" והוא מפריד בין ביטויים הניתנים להחלפה בתוך הקוד
אותיות גדולות באנגלית	כל צעד (PROC) או הוראה (STATMENT) בקוד SAS מיוצגת באותיות גדולות (להבדיל ממשתנים או אופציות של צעדים והוראות).

טבלה 1 – סימנים מוסכמים ומשמעותם

אודות התוכנה

SAS (Statistical Analysis System) היא מערכת משולבת של תוכניות מחשב שפותחה על ידי SAS Institute Inc.
SAS מאפשרת למשתמש לבצע, בין היתר:

א. ניהול נתונים:
   1. קליטה וארגון של נתונים חדשים
   2. טרנספורמציה של נתונים קיימים על מנת ליצור קבצי נתונים חדשים
   3. שליפה של נתונים
   4. יצירת דוחות
ב. ניתוח סטטיסטי:
   1. הצגת נתונים בטבלאות שכיחות
   2. הצגה גרפית
   3. חישוב מתאמים
   4. מבחנים סטטיסטיים
   5. בדיקת השערות
   6. משוואת מבניות (SEM)
ג. אחסון נתונים
ד. כריית מידע
   1. מיצוי ידע מתוך קבצי נתונים גדולים
   2. מידול
ה. OLAP (On Line Analytical Processing) – ניתוחים על קבצי נתונים רב ממדיים.
ו. משחקים!!! למרות ש-SAS היא לא תוכנה גרפית, היא כוללת מספר משחקים מובנים. המשחקים כוללים סולייטר, בלאק ג'ק, פוקר, מכונת מזל, איקס-עיגול וסידור אותיות. כדי להגיע למשחקים ב-SAS, יש לבחור Solutions→ Games → Accessories.

קובץ הנתונים

SAS עובדת עם קובץ נתונים ASCII, הבנוי ממשתנים ותצפיות. כברירת מחדל תצפיות בקובץ נתונים של SAS רשומות בשורות ומשתנים בעמודות. ניתן להפריד בין המשתנים על ידי רווחים, פסיקים, נקודות, וניתן גם לרשום משתנים (ותצפיות) ללא רווחים כלל (אך במצב כזה יש להגדיר לתוכנה באילו עמודות מופיע כל משתנה, כפי שיפורט בהמשך). לשם הנוחיות, ניתן לייבא קבצי נתונים שונים ל-SAS, כגון קובץ txt, excel, dat וכדומה).

בקובץ נתונים של SAS קיימים שני סוגים של משתנים:

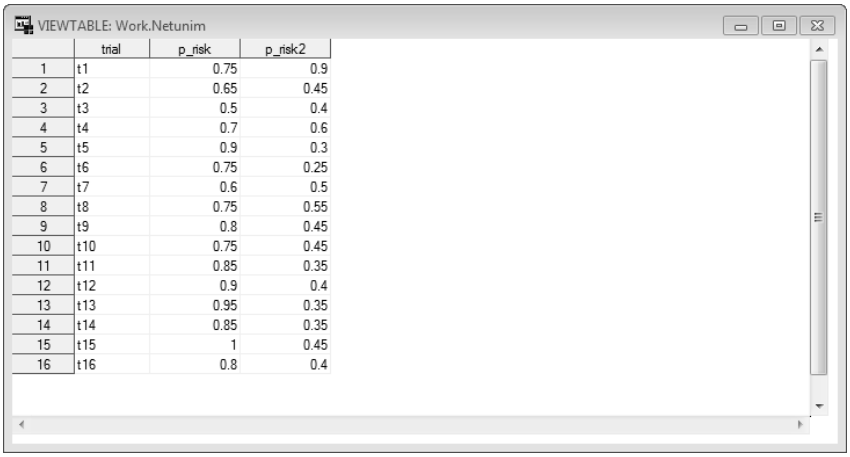
א. משתנים נומריים – משתנים בעלי ערכים מספריים.
ב. מחרוזות (משתנים אלפאנומריים) – משתנים היכולים להכיל גם מספרים וגם אותיות (אך מטופלים כמשתנים שמיים בלבד).

איור 1 מציג קובץ נתונים בסיסי (המוצג ב-notepad) המכיל משתנה מחרוזת אחד ושני משתנים נומריים.

עבודה ב - SAS

העבודה ב-SAS נעשית בשתי דרכים:

<u>תכנות</u>: בניגוד לתוכנות רבות אחרות, ב - SAS יש אפשרות לעבוד ללא שימוש בתפריטים, אלא לכתוב קוד עבור כל פעולה. תכונה זו של התוכנה מאפשרת גמישות רבה יותר ולפעמים אף מקצרת תהליכים בהשוואה לתוכנות מבוססות תפריטים, בהם כל הפרוצדורות קבועות מראש עפ"י מבנה התפריטים. ספר זה ייתמקד בעבודה ב-SAS באמצעות תכנות בלבד.

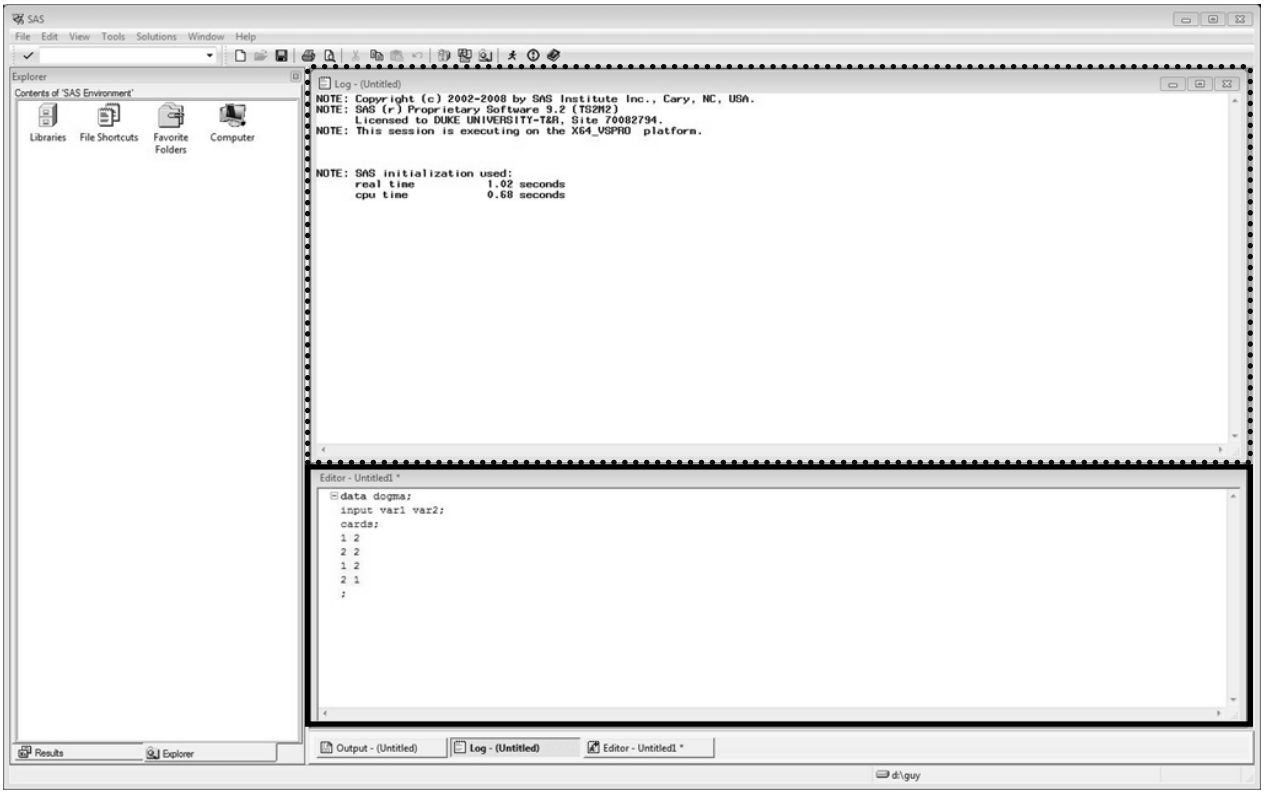

VIEWTABLE: Work.Netunim

	trial	p_risk	p_risk2
1	t1	0.75	0.9
2	t2	0.65	0.45
3	t3	0.5	0.4
4	t4	0.7	0.6
5	t5	0.9	0.3
6	t6	0.75	0.25
7	t7	0.6	0.5
8	t8	0.75	0.55
9	t9	0.8	0.45
10	t10	0.75	0.45
11	t11	0.85	0.35
12	t12	0.9	0.4
13	t13	0.95	0.35
14	t14	0.85	0.35
15	t15	1	0.45
16	t16	0.8	0.4

איור 1 – דוגמא לקובץ נתונים המכיל משתנה מחרוזת ושני משתנים נומריים

תוכניות SAS בדרך כלל נכתבות, נבחנות ומורצות מתוך SAS Display Manager. מנהל התצוגה של SAS מורכב מחלונות. שלושת החלונות העיקריים הם:

1. החלון Editor (ראה מסגרת שחורה מלאה באיור 2)
2. החלון LOG (ראה מסגרת שחורה מקווקוות באיור 2)
3. החלון OUTPUT



איור 2 – מנהל התצוגה של SAS

כל אחד משלושת החלונות הללו נפתח כאשר מפעילים את SAS. כדי לעבור מחלון לחלון, יש ללחוץ על שם החלון מתוך שורת המשימות של התוכנה (ראה חלק תחתון באיור 2). לחילופין, ניתן לבחור בתפריט View או Window ולבחור את החלון הרצוי, או ללחוץ F5 בשביל החלון Editor, F6 בשביל החלון Log ו-F7 בשביל החלון Output.

<u>עבודה עם תפריטים</u>: SAS מאפשרת גם עבודה יותר אוטומטית עם תפריטים באמצעות SAS/ASSIST (על ידי בחירה בתפריט ASSIST→Solution). עם זאת, במסגרת ספר זה העבודה תתמקד בעבודת תכנות בלבד.

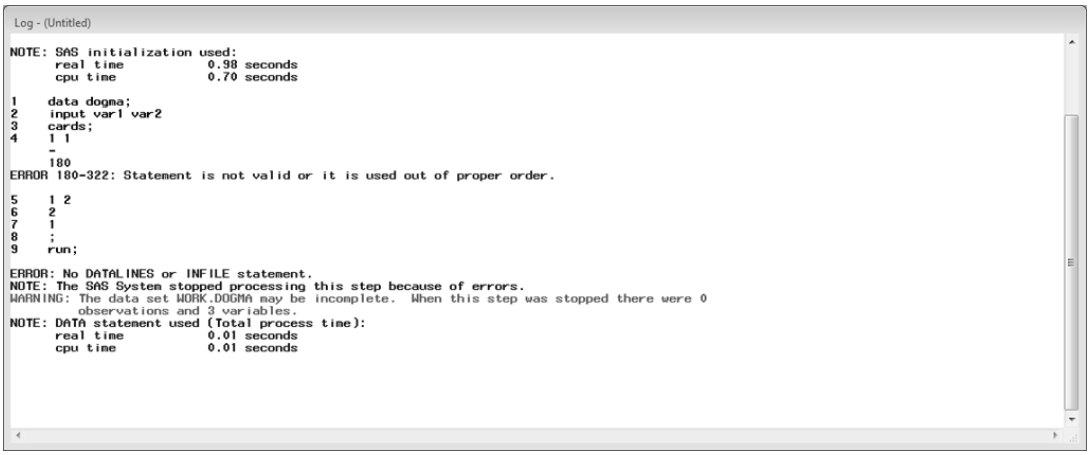
החלון EDITOR

במהלך עבודה ב-SAS, שורות קוד נכתבות לחלון Editor. כפי שיפורט בהמשך, בחלון Editor ניתן ליצור ולערוך קבצי נתונים, ולבצע פעולות (סטטיסטיות ואחרות) על נתונים באמצעות פרוצדורות מובנות ב-SAS, או באמצעות תכנות חופשי.

כדי לשמור את התוכנית הכתובה בחלון Editor, יש לוודא כי חלון זה הוא החלון הפעיל (לחיצה על F5). לאחר מכן, יש לבחור את התפריט File→Save As, ולתת שם לתוכנית. בדומה, כדי "לנקות" את התוכן של החלון Editor, יש לבחור את התפריט Edit→Clear All. לבסוף, כאשר רוצים לפתוח חלון Editor נוסף, יש לבחור את התפריט File→New Program.

החלון LOG

כאשר מריצים קוד ב-SAS, רשימה של הפעולות שנעשות על ידי התוכנה מוצגת בחלון Log. בנוסף, החלון Log מציג הערות לגבי קובץ הנתונים (כגון כמות המשתנים בהם נעשה שימוש בעיבוד וכמות התצפיות, משתנים בהם לא נעשה כלל שימוש וכדומה) ומידע כללי על תוכנת SAS (כגון גרסא נוכחית, פרטי הרישיון, זמן עיבוד הנתונים וכדומה). לבסוף, החלון Log מאפשר לעשות Debugging, שכן הוא מציג את השגיאות הקיימות בקוד.



איור 3 – החלון Log

הערות בקובץ Log מופיעות בטקסט כחול, אזהרות בטקסט ירוק ושגיאות בטקסט אדום. הקוד, הכולל את הפעולות שנעשו במהלך ההרצה, מופיע בטקסט שחור. איור 3 מציג דוגמא לסוגי הרשומות השונים המופיעים בחלון Log, ואת הצבעים האופייניים לכל סוג.

המידע המוצג בחלון Log הוא מצטבר. דהיינו, כל הרצה מוסיפה למידע שמוצג בחלון Log את המידע הרלוונטי להרצה הנוכחית, ולא מוחקת את המידע שהוצג על הרצות קודמות.

כדי לשמור את הרשומות המוצגות בקובץ Log, יש לוודא ראשית כי חלון זה הוא החלון הפעיל (F6). לאחר מכן, יש ללחוץ על התפריט File→Save As, ולתת שם לקובץ. את המוצג בחלון Log ניתן לשמור כקובץ Log ייעודי, קובץ RTF או קובץ DATA. כדי "לנקות" את תוכן החלון Log יש לבחור את התפריט Edit→Clear All או את התפריט File→New.

החלון OUTPUT

הפלט מההרצה של הקוד של SAS מוצג בתור קובץ נתונים בעל פורמאט ייחודי בחלון Output. אלא אם יישמר, הפלט המוצג בחלון Output נשמר כל עוד SAS מופעלת, אך הוא נמחק כאשר סוגרים את התוכנה.

בדומה לחלון Log, גם החלון Output מציג פלט מצטבר. כדי לשמור את הפלט המוצג בקובץ Output, יש לוודא ראשית כי חלון זה הוא החלון הפעיל (F7). לאחר מכן, יש ללחוץ על התפריט File→Save As, ולתת שם לקובץ. את המוצג בחלון Output ניתן לשמור כקובץ Output ייעודי, קובץ RTF או קובץ DATA. כדי "לנקות" את תוכן החלון Output יש לבחור את התפריט Edit→Clear All או את התפריט File→New. כמו כן, ניתן להדפיס את הפלט ישירות מהתוכנה, באמצעות בחירה בתפריט File→Print.

כתיבת תוכניות ב-SAS

כתיבת תוכניות ב-SAS נעשית באמצעות שפת תכנות פשוטה (SAS command language) מבוססת הוראות (Statements). כל הוראה אומרת למערכת SAS לבצע פעולה מסוימת, או מספקת מידע כלשהו.

<u>מבנה הכתיבה</u> (כמה כללי אצבע):

- ניתן לכתוב ב SAS תוך שימוש באותיות קטנות או גדולות (SAS איננה case sensitive).
- ניתן להתחיל לכתוב הוראות (שורות קוד) בכל מקום של השורה.
- הוראה יכולה להימשך אל מעבר לשורה אחת.
- ניתן לכתוב מספר שורות קוד באותה שורה, כאשר שורות הקוד השונות מופרדות על ידי נקודה פסיק.
- כל הוראה (שורת קוד) ב-SAS חייבת להסתיים בסימן ; (נקודה פסיק). אחרת, SAS מתייחסת לשורה כאל תחילתה של השורה הבאה.
- כדי להפוך שורות קוד ללא פעילות (למשל כדי להכניס הערות), יש לנקוט באחת משתי הפעולות הבאות:
  - להכניס את הסימן * (כוכבית) בתחילת השורה. במצב כזה, כל שורת הקוד (על לסימן ה - ; הקרוב) יצבע בצבע ירוק, והתוכנה תתעלם משורה זו.
  - להכניס את הסימן /* בתחילת קטע, ואת הסימן */ בסוף קטע. במצב כזה כל שורות הקוד בין שני הסימנים יצבעו בירוק, והתוכנה תתעלם מקטע זה.
- שם של משתנה חייב להתחיל באות.
- שם משתנה יכול לכלול אותיות, ספרות ואת הסימן _ (קו תחתון). שם המשנה יכול להכיל את תו הרווח, בתנאי שמגדירים זאת (כפי שיפורט בהמשך).
- נקודות מסמנות ערכים חסרים במשתנים נומריים, ורווחים מסמנים ערכים חסרים במשתנים אלפאנומריים. SAS הופכת אוטומטית ערכים חסרים (רווחים) במשתנים נומריים לנקודות.

<u>מבנה התוכנית</u>:

- תוכנית SAS היא סדרה של שורות קוד המורות למערכת לבצע משימות שונות או עיבודים שונים. ההוראות השונות מופיעות בשני סוגים של צעדים (steps): DATA STEP ו-PROC STEP (כפי שיפורט בהמשך). בסוף כל צעד צריכה להופיע הפקודה ;RUN. עם זאת, ניתן לכלול כל צעד מספר פעמים בתוכנית SAS אחת.

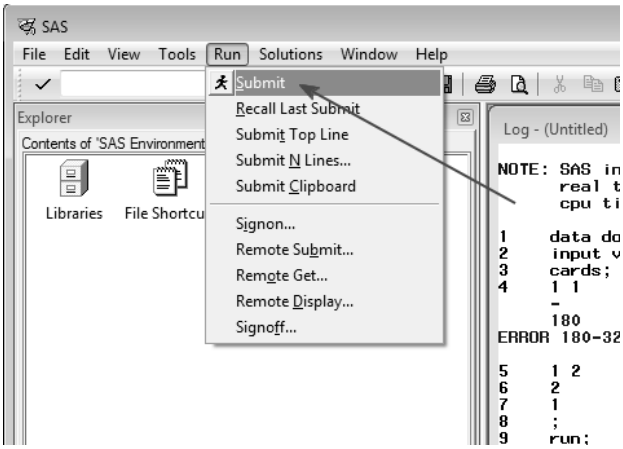
טיפ ממומחה: ב-SAS, למרכיבים שונים של הקוד יש צבעים שונים: הגדרת צעד (PROC או DATA) מופיע בקוד בצבע כחול כהה, הוראות ואופציות מופיעות בצבע כחול בהיר, מחרוזות מופיעות בצבע סגול, ומספרים מופיעים בצבע ירוק. לכן, בחינה של הקוד לפני ההרצה (לוודא שמה שאמור להיות בצבע כחול הוא כחול, ומה שאמור להיות ירוק הוא ירוק וכדומה) יכולה לעזור ולמצוא טעויות בהקלדה, ומהווה סימן טוב לבדיקה ראשונית שלו.

<u>הרצת התוכנית</u>:

כאשר מסיימים לכתוב תוכנית ב-SAS, יש להריץ אותה. ניתן להריץ תוכנית ב-SAS או מתוך חלון התוכנה הראשי או מתוך החלון Program editor, על פי הפירוט הבא:

מתוך חלון התוכנה:

- לחיצה על "Run" מתוך תפריט הפקודות ובחירה בפקודה "Submit" (ראה איור 4).
- לחיצה על המקש של האיש הרץ (ראה איור 5).
- לחיצה על המקש F8 (יעבוד רק כאשר החלון של SAS הוא החלון הפעיל).

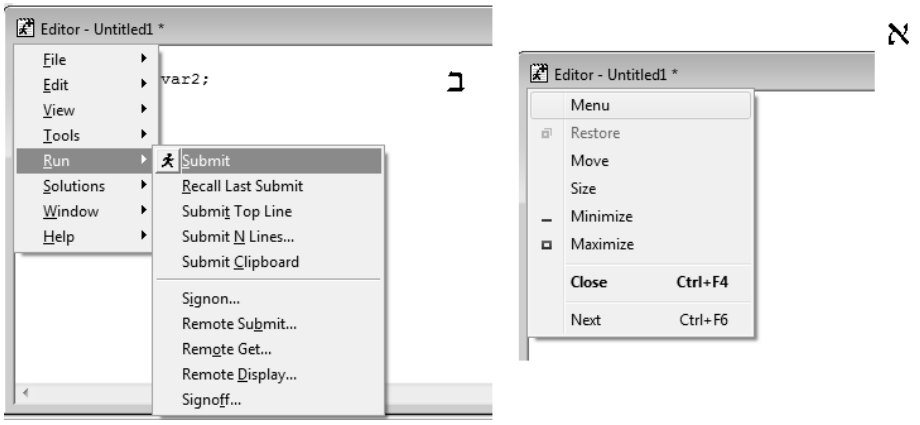


איור 4 – הרצת קוד SAS דרך התפריט הראשי

איור 5 – הרצת קוד SAS דרך סרגל הכלים

מתוך החלון Program editor:

- לחיצה על הפינה השמאלית העליונה של החלון Editor (על סימן ה -)، ובחירה ב-Submit→Run→Menu (ראה איור 6).
- כאשר רוצים להריץ חלקים מסוימים בלבד של התוכנית: סימון החלקים הרלוונטיים עם העכבר ("צובע" את שורות הקוד בשחור), וחזרה על אחד מהשלבים להרצת תוכנית שתוארו לעיל.



איור 6 – הרצת קוד SAS מתוך החלון Editor

פרק 2
יצירת קבצי נתונים

ה-DATA STEP

הצעד DATA (DATA STEP) מכיל את כל ההוראות הדרושות ליצירת קובץ נתונים, טיפול בנתונים וכדומה. צעד זה כולל הוראה המציינת את שם קובץ הנתונים ומיקומו, את שם המשתנים ומיקומם בקובץ, את סוג המשתנים, כמו גם מאפיינים נוספים הכוללים ציון דרך הטיפול בנתונים חסרים, ביצוע שינויים (או טרנספורמציות) במשתנים קיימים, הגדרת משתנים חדשים, סינון תצפיות, בחירת משתנים וכדומה.

ה-DATA STEP יוצר מהנתונים הקיימים קובץ בפורמט SAS שהתוכנה יודעת לקרוא.
ה-DATA STEP מתחיל בהוראה DATA ומסתיים בהוראה RUN:

```
DATA שם קובץ הנתונים;
.................
.................
run;
```

ישנם שני סוגים של קבצי נתונים (dataset) ב-SAS:

1. קובץ נתונים זמני – קובץ נתונים הנשמר בספריה work וקיים כל עוד תוכנת SAS רצה.
2. קובץ נתונים קבוע – קובץ נתונים הנשמר בספרייה מוגדרת על ידי המשתמש ונשמר גם אחרי ש-SAS איננה רצה.

ההוראה DATA

הוראת DATA יוצרת את קובץ הנתונים ב-SAS ונותנת לו שם המוגדר על ידי המשתמש.
אופן הכתיבה:

DATA שם כלשהו;

דוגמא:

```
data dogma;
```

קריאת קובץ נתונים חיצוני

ההוראה INFILE

ההוראה INFILE מציינת את שם קובץ הנתונים החיצוני ואת מיקומו במחשב, והיא מאפשרת ל-SAS לקרוא את הנתונים הנמצאים בקובץ ולטעון אותם לזיכרון העבודה של התוכנה. ההוראה INFILE חייבת להופיע אחרי ההוראה DATA, והיא

חייבת לכלול את שם הקובץ ונתיבו (בתוך גרשיים). בנוסף, ההוראה INFILE יכולה לכלול גם אופציות שונות, כפי שיפורט בהמשך.
אופן הכתיבה:

INFILE 'שם הקובץ ונתיבו' <אופציות שונות>;

דוגמא:

```
data dogma; infile 'c:\course\data.txt';
```

אופציות של ההוראה INFILE

האופציות של ההוראה INFILE מאפשרות לציין מאפיינים שונים של קובץ הנתונים החיצוני או להגדיר פעולות שהוראת DATA תבצע עליו. כל האופציות נכתבות אחת אחרי השנייה לאחר שם הקובץ ומיקומו, כאשר האופציות מופרדות על ידי רווחים.

1. האופציה delimiter – אופציה זו מציינת את סוג התו המשמש כמפריד בין המשתנים (נקודה, פסיק, רווח וכדומה). כאשר אופציה זו לא מוגדרת, SAS מניח כי המשתנים מופרדים על ידי רווחים.
אופן הכתיבה:

dlm = 'סוג התו המפריד'

דוגמא לקריאת קובץ המופרד על ידי טאבים:

```
data class1; infile 'c:\course\data.txt' dlm = '09'x;
```

דוגמא לקריאת קובץ המופרד על ידי נקודה פסיק:

```
data class1; infile 'c:\course\data.txt' dlm = ';';
```

טיפ ממומחה: הטעות הנפוצה ביותר של מתחילים ב SAS היא כתיבת פקודות ללא ; (נקודה פסיק) בסיומן

2. האופציה dsd – לאופציה זו יש שלושה תפקידים כאשר SAS קוראת קובץ חיצוני בעל משתנים מופרדים. התפקיד הראשון הוא "להפשיט" מרכאות המקיפות את ערכי המשתנים בתוך הקובץ (דבר האופייני מאוד לקבצים המופקים לדוגמא מתוכנת ויז'ואל בייסיק). התפקיד השני קשור לערכים חסרים. כאשר SAS נתקלת בתוך הקובץ החיצוני במפרידים עוקבים בתוך קובץ (לדוגמא ;; בקובץ המופרד על ידי נקודה פסיק), ברירת המחדל של התוכנה תהיה להתייחס למפרידים אלה כאל יחידה אחת. אולם, בדרך כלל מפרידים עוקבים מציינים ערכים חסרים למשתנים.
האופציה DSD אומרת ל-SAS להתייחס למפרידים עוקבים בנפרד, כך שערך החסר בין שני מפרידים עוקבים יוגדר ככזה על ידי התוכנה. התפקיד השלישי מניח שהמפריד בין ערכי המשתנים הוא פסיק. לכן, אם נשתמש באופציה DSD במצב בו התו המפריד הוא אכן ",", אין צורך להשתמש גם באופציה DLM. לעומת זאת, אם המפריד הוא לא התו ",", יש צורך להשתמש גם באופציה DLM במקביל.
אופן הכתיבה:

dsd

דוגמא לקריאת קובץ המופרד על ידי נקודה פסיק, כאשר המשתנים נמצאים בתוך מרכאות:

```
data class1; infile 'c:\course\data.txt' dlm = ';' dsd;
```

3. האופציה lrecl – אופציה זו מציינת את מספר התווים בשורה של הקובץ החיצוני (רלוונטי רק כאשר מדובר על קובץ ASCII). יש להשתמש באופציה זו רק כאשר מספר התווים בשורה עולה על 256 (ברירת המחדל), שכן במצב כזה, כל ערך שנמצא מעבר לטווח 256 התווים יקוצץ. ערכים תקפים של אופציה זו נעים בין 1 ל-65,535.
אופן הכתיבה:

lrecl = מספר

דוגמא:

```
data class1; infile 'c:\course\data.txt' dlm = ';' dsd lrecl =
     1000;
```

טיפ ממומחה: מומלץ להשתמש באופציה lrecl ולציין מספר תוים גבוה (למשל 1000).
בצורה זו מוודאים כי אם במקרה קובץ הקלט הוא ארוך מאד, השורות ייקראו כמו שצריך.

4. האופציה missover – אופציה זו מגדירה ל-SAS איך לטפל בערכים חסרים בסוף השורה. כברירת מחדל, אם SAS מגיעה לסוף השורה בקובץ, ולא נקראו כל המשתנים המוגדרים (למשל במצב של נתונים חסרים), היא ממשיכה לקרוא נתונים השייכים לשורה זו מהשורה הבאה. פעולה זו נקראת flowover. האופציה missover מבטלת פעולה זו, וגורמת למשתנה שלא נקרא (או שנקרא בחלקו) להירשם בקובץ הנתונים של SAS כערך חסר.
אופן הכתיבה:

missover

דוגמא:

```
data class1; infile 'c:\course\data.txt' dlm = ';' missover;
```

טיפ ממומחה: מומלץ מאוד להשתמש באופציה missover. אחרת, אם במקרה יש משתנה בעל ערך חסר, ערכו ייקבע כערך המשתנה הבא אחריו, וזה ייצור "אפקט דומינו" שבו כל המשתנים בהמשך השורה לא ייקראו כמו שצריך.
האופציות הנפוצות ביותר לשימוש בהוראה INFILE הן לפיכך lrecl ו missover

5. האופציה obs – אופציה זו מציינת את מספר התצפיות שהמשתמש רוצה לקרוא מתוך הקובץ החיצוני.
אופן הכתיבה:

obs = n

(כאשר n מייצג את מספר התצפיות הראשונות מתוך הקובץ).

דוגמא לקריאת 50 התצפיות הראשונות מהקובץ:

```
data class1; infile 'c:\course\data.txt' dlm = ';' obs = 50;
```

6. האופציה firstobs – אופציה זו מציינת את התצפית הראשונה ממנה רוצים להתחיל לקרוא את הקובץ החיצוני.
אופן הכתיבה:

firstobs = m

(SAS תתחיל לקרוא את הקובץ מהתצפית ה-m ית).

דוגמא לקריאת קובץ המכיל שורת כותרת, המוגדרת לקרוא החל מהשורה השנייה (שהיא התצפית הראשונה) עד לתצפית ה - 50:

```
data class1; infile 'c:\course\data.txt' dlm = ';' firstobs = 2
     obs = 51;
```

ההוראה INPUT

ההוראה INPUT מגדירה את שמות המשתנים בקובץ, את מיקומם ואת סוגם. כפי שצוין קודם, שם משתנה ב-SAS חייב להתחיל באות, והוא יכול לכלול אותיות, מספרים וקו תחתון (_). תוספת של הסימן דולר ($) בסוף שם המשתנה מגדירה ל-SAS כי מדובר במשתנה מחרוזת (אלפאנומרי). סימן ה - $ יכול להופיע בצמוד לשם או כאשר רווח מפריד בין שם המשתנה לסימן. לשם הנוחיות, מומלץ לקרוא למשתנים בשמות שיאפיינו את המהות שלהם (לדוגמא לקרוא למשתנה שמציין גיל age).
אופן הכתיבה:

INPUT שמות המשתנים (לפי הסוג ולפי מיקומם בקובץ הנתונים)

דוגמאות:

כאשר הנתונים של כל תצפית בקובץ מופרדים על ידי רווחים בהתאם לסדר המשתנים:

1 24 male safe
2 26 male risky
3 32 female risky
4 22 male safe
5 27 female safe

או על ידי תווים (לדוגמא, פסיק):

1,24,male,safe
2,26,male,risky
3,32,female,risky
4,22,male,safe
5,27,female,safe

אין צורך לציין את מיקום המשתנים אלא רק את שמותיהם לפי סדר הופעתם בקובץ (משמאל לימין):

```
input sub_number age gender$ choice$;
```

לעומת זאת, אם נתוני התצפיות אינם מופרדים כלל:

124malesafe
226malerisk
332femarisk
422malesafe
527femasafe

יש צורך, בנוסף להגדרת שמם של המשתנים ומיקומם, גם לציין את מיקומם בעמודות (ואז חייבים לוודא שכל משתנה מכיל את אותו מספר ערכים):

```
input sub_number 1 age 2-3 gender$ 4-7 choice$ 8-11;
```

עם זאת, יש לציין כי במקרה כזה אין צורך לרשום את שמות המשתנים לפי סדר הופעתם בקובץ הנתונים, שכן המשתמש מצהיר במדויק על מיקומם בתוך הקובץ. כמו כן, המשתמש לא חייב לקרוא לכל המשתנים הקיימים בקובץ, אלא רק לאלה הנחוצים לו לעיבוד או ניתוח.

לעיתים, יש מצבים בהם נתונים של אותה תצפית נמצאים בשורות נפרדות:

01 Male
23 Risky
02 Female
24 Safe
03 Female
23 Risky

ניתן לקרוא קבצי נתונים כאלה, כאשר מציינים עבור כל משתנה, בנוסף להגדרות שתוארו לעיל, באיזה שורה הוא מופיע.

אופן הכתיבה:

מספר השורה #

דוגמא:

```
input #1 sub_number 1-2 #2 age 3-4 #1 gender$ 4-9 #2 choice$ 7-
       11;
```

כאשר רוצים לקרוא כמה רשומות של אותם משתנים מאותה שורה:

01 32 167 02 24 170 03 25 189
04 26 175 05 27 180

יש להשתמש בסימן @@, אשר אומר ל-SAS להתייחס לרשומה שנמצאת לאחר סוג המשתנה האחרון שהוגדר ב-input כאל המשתנה הראשון שהוגדר וכך הלאה, עד שיגמרו הרשומות בשורה.

אופן הכתיבה:

INPUT שמות המשתנים (לפי הסוג ולפי מיקומם בקובץ הנתונים) @@

דוגמא:

```
input sub age height @@;
```

לבסוף, כאשר רוצים לכלול רווחים במשתנים אלפאנומריים, לדוגמא:

Eli Ronen 25 175
Nir Yemini 27 180
Avi Israeli 31 179

יש להגדיר ל-SAS כי מדובר במשתנה אלפאנומרי (באמצעות הסימן $) וכי הוא מכיל *n* תווים, על ידי כתיבת מספר התווים ולאחריהם נקודה לאחר הגדרת שם המשתנה.

אופן הכתיבה:

שם המשתנה INPUT $ מספר.

דוגמא:

```
input sub $ 13. age height;
```

	A	B	C	D	E
1	1	24	male	safe	
2	2	26	male	risky	
3	3	32	female	risky	
4	4	22	male	safe	
5	5	27	female	safe	
6					
7					
8					
9					

איור 7 – מבנה קובץ אקסל ש-SAS יכולה לקרוא מתוך ה-DATA STEP

טיפ ממומחה: למרות שניתן לכתוב ב-SAS בכל מקום בשורה, מומלץ מטעמי נוחיות ליישר את הקוד בצורה כזאת שהגדרת הצעד (PROC או DATA) וההוראה RUN יהיו מיושרים לשמאל, וכל הוראה הקשורה לתוכנית זאת תורחק מעט מהשוליים (ראה דוגמא לעיל). בצורה זו יהיה קל יותר לעקוב אחרי הקוד ולעשות בו שינויים במקרה הצורך.

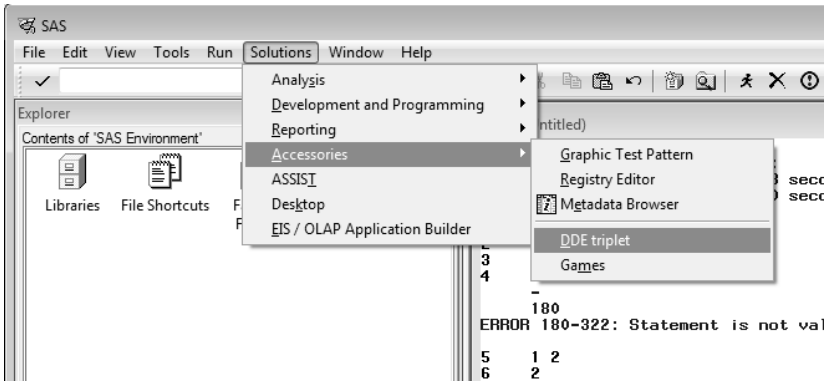
קריאת נתונים מקובץ EXCEL

ישנן שתי דרכים עיקריות לקרוא נתונים מקובץ EXCEL. דרך אחת היינה להשתמש בפרוצדורת ייבוא (PROC IMPORT). אולם, מאחר וה-PROC STEP לא נידון בחלק זה של הספר, הדרך לייבוא נתונים מקובץ EXCEL באמצעות פקודה זו תידון בפרק העוסק ב-PROC STEP. הדרך השנייה (והקצת יותר מסובכת) היא להשתמש באופציית DDE (Dynamic Data Exchange) של SAS. כאשר משתמשים באופציה זו, יש לוודא כי גיליון האקסל ממנו רוצים

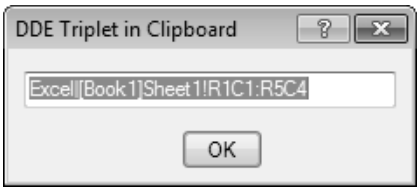
לקרוא נתונים פתוח, וכי הוא אינו כולל שורה של שמות משתנים. כדי לקרוא נתונים מקובץ אקסל (ראה לדוגמא איור 7), יש להשתמש בקוד הבא:

```
filename example dde 'Excel|C:\[auto.xls]Sheet1!R1C1:R5C4';
 data bean;
 infile example;
 input sub_number age gender$ choice$;
run;
```

המחרוזת שמופיעה בהצהרה של שם הקובץ לאחר הפקודה dde קרויה DDE triplet, והיא מגדירה את גיליון העבודה של EXCEL ממנו אנו רוצים לקרוא את הנתונים ואת השורות והעמודות בהן יש נתונים. כמובן, הגדרות אלה משתנות מקובץ לקובץ, ולעיתים קשה מאוד לקבוע אילו ערכים בדיוק אמורים להיות כלולים במחרוזת זו. עם זאת, יש דרך פשוטה וקלה מאוד להשיג מידע זה. בתוך הקובץ EXCEL, יש לסמן את כל השורות והעמודות שאתם רוצים לייבא ל-SAS ולהעתיק אותם ל-Clipboard (על ידי לחיצה על כפתור ימני בעכבר ובחירה בהעתק, או על ידי Ctrl+C). לאחר מכן יש לגשת ל-SAS, ולבחור את התפריט Solutions→Accessories→DDE triplet (ראה איור 8). פעולה זו תקפיץ את החלון DDE triplet in clipboard, אשר כולל את ה-DDE triplet הרלוונטי לגיליון העבודה איתו תרצו לעבוד (ראה איור 9).



איור 8 – כיצד להשיג את המחרוזת DDE triplet לקריאת קבצי אקסל



איור 9 - מחרוזת DDE triplet

כתיבת קובץ הנתונים ישירות בקוד

ההוראה CARDS

אם רוצים להקליד את הנתונים ישירות לתוכנת SAS, ולא לקרוא אותם מקובץ חיצוני, יש צורך להשתמש בהוראה CARDS, אשר מציינת ל-SAS כי בשורות הבאות כתובים ערכים של נתונים, ולסיים בסימן נקודה פסיק (;), אשר מציין ל-SAS כי אין יותר נתונים לקריאה. סימן הנקודה פסיק חייב לבוא בשורה נפרדת, לאחר השורה האחרונה של הנתונים.
כאשר משתמשים בהוראה CARDS, אין צורך להשתמש בהוראה INFILE. השימוש בהוראה INPUT, לעומת זאת, נעשה בדיוק באותה צורה כמו בקריאת נתונים מקובץ חיצוני, למעט העובדה שהגדרת שמות הנתונים וסדר הופעתם מתרחשת לפני

ההוראה CARDS. לרוב, אופציה זו נמצאת בשימוש כאשר רוצים ליצור סט נתונים קטן, או כאשר רוצים לעשות ניסוים או בדיקות על חלק קטן מקובץ הנתונים הקיים.

אופן הכתיבה:

CARDS;
שורות של נתונים
;

דוגמא:

```
data example;
input subject_number age gender$ choice$;
cards;
1 24 male safe
2 26 male risky
3 32 female risky
4 22 male safe
5 27 female safe
;
```

הערה: כאשר משתמשים בהוראה CARDS, אפשר להפריד את הנתונים על ידי רווחים, או לרשום את הנתונים ללא רווחים בלבד.

תרגול עצמי – יצירת קובץ נתונים

תרגיל 1

כתוב תוכנית SAS היוצרת קובץ נתונים על בסיס קובץ נתונים חיצוני המופרד על ידי פסיקים (,). הקובץ החיצוני כולל 5 תצפיות ו - 5 משתנים לכל תצפית: מספר סידורי של הנבדק, מין, גיל, משקל וגובה. צור את קובץ הנתונים החיצוני בפורמט .txt

תרגיל 2

כתוב תוכנית SAS בה כתיבת הנתונים מתבצעת בתוך ה-DATA STEP, כאשר קובץ הנתונים הינו ללא רווחים כלל. הנתונים כוללים 5 תצפיות ו-5 משתנים לכל תצפית: מספר סידורי של הנבדק, מין, גיל, משקל וגובה.

פרק 3
קריאת קובץ נתונים הקיים ב-SAS

לאחר שיצרנו קובץ נתונים ב-SAS, ייתכן כי נרצה לעשות שינויים מסוימים בנתונים, להחסיר או להוסיף נתונים וכדומה. SAS מאפשרת לקרוא קובץ נתונים קיים ולעשות עליו מניפולציות שונות באמצעות ההוראה SET.

טיפ קריאה: למתחילים, מומלץ בשלב זה להסתפק בדרכי הקלט שהופיעו למעלה. ההוראה SET מיועדת לעבודה מתקדמת ב SAS. למתחילים מומלץ לעבור בשלב זה לפרק 4, הדן בתפעול קבצי נתונים.

ההוראה SET

ההוראה SET מופיעה ב-DATA STEP, והיא יוצרת קובץ נתונים חדש מתוך קובץ נתונים קיים. ניתן ליצור באמצעות הוראה זו קובץ נתונים חדש לגמרי, או לשכתב את קובץ הנתונים הקיים. לקובץ הנתונים החדש שיוצרים ניתן להעביר חלק מהנתונים הקיימים בקובץ הנתונים המקורי, ליצור משתנים חדשים או לבצע עיבודים שונים על המשתנים הקיימים. בנוסף, ההוראה SET יכולה לאחד בין מספר קבצי נתונים. עם זאת, אפשרות זאת תידון בהרחבה בפרק הבא.

אופן הכתיבה:

DATA שם של קובץ נתונים חדש; SET שם של קובץ נתונים קיים;

דוגמא:

```
data targil2; set targil;
run;
```

במצב זה קובץ הנתונים הקיים (targil) מועתק בשלמותו לקובץ הנתונים חדש (targil2). לדוגמא, אם קובץ הנתונים targil הכיל את הרשומות הבאות:

```
height age
175 23
180 26
167 22
```

אזי הקובץ החדש targil2 יכיל בדיוק את אותן רשומות:

```
height age
175 23
180 26
167 22
```

הערה: בנוסף להעתקת קובץ נתונים (בחלקו או בשלמותו) לקובץ נתונים חדש, ניתן לשכתב קובץ נתונים קיים. במקרה זה, קובץ הנתונים החדש שנוצר "ידרוס" את קובץ הנתונים הקיים.

דוגמא:

```
data targil2; set targil2;
run;
```

במצב זה, כל פעולה שתתבצע על קובץ הנתונים הישן תדרוס את המידע הקיים ותכתב במקומו.

אופציות של ההוראות DATA ו - SET

כל האופציות של ההוראות DATA ו-SET נכתבות בתוך סוגריים. בחלק מהאופציות אין חשיבות לשאלה האם האופציה נכתבת לאחר ההוראה DATA או לאחר ההוראה SET. לעומת זאת, בחלק מהאופציות יש לכך חשיבות מכרעת, כפי שיפורט בהמשך.

1. האופציה rename – אופציה זו מאפשרת לשנות את שמות המשתנים בקובץ.
אופן הכתיבה:

(Rename = (שם ישן = שם חדש שם ישן2 = שם חדש2))

דוגמא:

```
data targil2 (rename =(age = gil height = gova)); set targil;
```

או:

```
data targil2 ; set targil (rename =(age = gil height = gova));
```

2. האופציה obs – אופציה זו מציינת את מספר התצפיות שרוצים לקרוא מהקובץ המקורי. אופציה זו תעבוד רק אם היא תכתב לאחר ההוראה SET.
אופן הכתיבה:

obs = (מספר שלם (קטן או שווה למסספר התצפיות בקובץ

דוגמא:

```
data targil2; set targil (obs = 2);
```

3. האופציה firstobs – אופציה זו מציינת את התצפית הראשונה ממנה רוצים להתחיל לקרוא את הקובץ הקיים. גם אופציה זו תעבוד רק אם היא תכתב לאחר ההוראה SET. כמו כן, ניתן להשתמש באופציה זו בשילוב עם האופציה obs או בלעדיה.
אופן הכתיבה:

firstobs = (מספר שלם (גדול מ-1 וקטן ממספר התצפיות בקובץ

דוגמא:

```
data targil2; set targil (firstobs = 2 obs = 3);
```

4. האופציה drop – אופציה זו מאפשרת להשמיט חלק מהמשתנים המקוריים בקובץ הנתונים החדש. עם זאת, יש לשים לב להבדל כאשר כותבים אופציה זו לאחר ההוראה DATA ולאחר ההוראה SET.
אופן הכתיבה:

drop = שמות של משתנים המופרדים על ידי רווחים

לדוגמא:

```
data targil2 (drop = age); set targil;
```

במצב זה (בו האופציה כתובה לאחר ההוראה DATA), SAS תקרא את הקובץ targil במלואו ואז תשמור כקובץ נתונים את targil2 בלי המשתנה המוגדר (גיל בדוגמא הנוכחית). לכן, במצב זה ניתן יהיה להשתמש במשתנה גיל במהלך העבודה ב-data step הנוכחי.
לעומת זאת, במצב של:

```
data targil2; set targil(drop = age);
```

SAS תקרא את הקובץ targil ללא המשתנה "גיל", כך שלא יהיה ניתן להשתמש בו במהלך העבודה ב-data step הנוכחי.

5. האופציה keep – אופציה זו אומרת ל-SAS איזה משתנים להשאיר במעבר מקובץ הנתונים הישן לקובץ הנתונים החדש.
אופן הכתיבה:

keep = שמות של משתנים מופרדים על ידי רווחים

דוגמא:

```
data targil2 (keep = age); set targil;
```

או:

```
data targil2; set targil (keep = age);
```

יצירת קבצי נתונים קבועים

ברירת המחדל בעבודה עם SAS היא ליצור קבצי נתונים זמניים, הקיימים כל עוד התוכנה עובדת. קבצי נתונים זמניים אלה נמצאים בספרייה work, והם נמחקים כאשר המשתמש סוגר את התוכנה. עם זאת, ניתן להגדיר ל-SAS ספרייה חדשה ולשמור בה קבצי נתונים קבועים שישמרו במחשב גם אחרי שהתוכנה תיסגר, ויהיו זמינים לעבודות נוספות.

ההוראה LIBNAME

ההוראה LIBNAME יוצרת ספרייה פנימית חדשה של SAS שבה ישמר קובץ הנתונים שיוגדר על ידי המשתמש. הוראה זו מגדירה את שם הספרייה ואת הנתיב שלה (המיקום הפיזי שלה במחשב). הוראה זו מופיעה לפי ה-DATA STEP של התוכנה, ויש להגדיר אותה מחדש בכל פעם שנכנסים לתוכנת SAS.

אופן הכתיבה:

LIBNAME שם של הספרייה החדשה 'הנתיב המלא של הספרייה/כונן';

דוגמא:

```
libname sascodes 'C:\My documents\SAS';
```

בדוגמא הנוכחית, SAS תיצור ספרייה בשם sascodes הנמצאת בתוך התיקייה SAS שנמצאת בתיקיית המסמכים שלי בכונן C.

הערה: שם הספרייה החדשה (sascodes במקרה הנוכחי) חייב להיות שונה מ-work, שכן שם זה הוא ברירת המחדל של SAS לספרייה בה נשמרים קבצי נתונים זמניים.

טיפ ממומחה: כאשר יוצרים משתנים זמניים ב-DATA STEP (למשל כאשר יוצרים לולאות), ורוצים למחוק אותם כדי שלא יופיע בקובץ הנתונים הסופי, כדאי לתת לכל המשתנים הזמניים הללו שם שמתחיל בקו תחתון (הסימן _). לאחר מכן, ניתן להגדיר באמצעות האופציה drop את המחיקה של כל המשתנים הללו מקובץ הנתונים (באמצעות הגדרת האופציה "drop = _:").

ההוראה DATA

כדי ליצור קובץ נתונים קבוע יש לציין את שם הספרייה ולאחר מכן את שם קובץ הנתונים, כאשר שני השמות מופרדים על ידי נקודה.

אופן הכתיבה:

שם הקובץ.שם הספרייה DATA

דוגמא:

```
libname sascodes 'C:\My documents\SAS';
data sascodes.dogma; set dogma;
run;
```

בדוגמא זו המשתמש קורא לקובץ נתונים זמני בשם dogma הקיים בספרייה work, ויוצר קובץ נתונים חדש באותו שם, שישמר כקובץ קבוע בספרייה פנימית של SAS. מיקומו הפיזי של קובץ הנתונים יהיה בתיקייה SAS הנמצאת בתיקייה My documents שנמצאת בכונן C.

דוגמא:

```
libname sascodes 'C:\My documents\SAS';
data sascodes.dogma;
 infile 'C:\My documents\SAS\dogma.txt' dlm = ',';
 input sub gil gender$;
run;
```

בדוגמא זו אנו קוראים קובץ נתונים חיצוני בשם dogma הנמצא בכונן C (תיקייה SAS הנמצאת בתיקייה My documents) ושומרים אותו כקובץ קבוע בשם dogma בתיקייה sascodes. קובץ הנתונים כולל את המשתנים מספר נבדק, גיל ומין.

הערה: כדי לקרוא את הקובץ החיצוני וליצור ממנו קובץ נתונים זמני, יש לכתוב את השורה:

```
data dogma;
```

במקום השורה

```
data sascodes.dogma;
```

דוגמא:

```
libname sascodes 'C:\My documents\SAS';
data sascodes.dogma;
input sub gil gender$;
cards;
1 32 M
2 25 F
3 22 M
4 27 F
;
```

בדוגמא זו אנו כותבים את הנתונים ישירות ב-data step ושומרים אותם בקובץ קבוע בשם dogma שישמר בספרייה sascodes.

תרגול עצמי – קריאת קובץ נתונים

תרגיל 3

כתוב קוד SAS שיצור קובץ נתונים חדש מתוך קובץ נתונים קיים:

א. קובץ הנתונים החדש צריך להכיל את כל המשתנים של קובץ הנתונים הישן, אך שם קובץ הנתונים החדש צריך להיות שונה (דהיינו, לא לדרוס את קובץ הנתונים הקיים).
ב. קובץ הנתונים החדש צריך "לדרוס" את קובץ הנתונים הישן.

תרגיל 4

קח את קובץ הנתונים הבא, וצור קובץ נתונים חדש שאינו כולל את המספר הסידורי של הנבדק.

```
data targil4;
input subject_number age gender$ choice$;
cards;
1 24 male safe
2 26 male risky
3 32 female risky
```

```
4 22 male safe
;
```

תרגיל 5

קח את קובץ הנתונים המקורי מתרגיל 4, ושנה את שמות המשתנים age ו-$gender ל-gil ו-$sex בהתאמה.

פרק 4
תפעול קבצי נתונים

בתוך ה-DATA STEP ניתן להגדיר משתנים חדשים ו/או לשנות ערכים של משתנים קיימים. פעולות תפעול המשתנים נעשות על שורות (התצפיות עצמן) ולא על עמודות (המשתנים).

יצירת משתנים חדשים ושינוי ערכי משתנים קיימים

SAS היא מה שנקרא weak typing software, דהיינו תוכנה שלא מצריכה הגדרה של משתנים (הן מבחינת שם המשתנה והן מבחינת סוג המשתנה) מראש (בעת כתיבת התוכנית). לכן, כדי ליצור משתנה חדש ב-SAS צריך פשוט לתת למשתנה שם ולהכניס לו ערך. לדוגמא, כדי ליצור משתנה בשם condition ולתת לו את הערך 1, צריך לכלול בקוד את השורה הבאה:

```
condition = 1;
```

כאשר נריץ את הקוד SAS הכולל שורה זו, ניצור למעשה משתנה חדש בשם condition, אשר יכיל עבור כל תצפית את הערך 1.

הערה: ניתן בצורה זאת גם לשנות ערכים של משתנים קיימים.

בנוסף, ניתן ליצור משתנים חדשים, או לשנות ערכים של משתנים קיימים, תוך שימוש בטרנספורמציה מסוימת על הנתונים (משתנים) הקיימים, באמצעות פעולות חשבוניות בסיסיות כגון חיבור (+), חיסור (-), כפל (*), חילוק (/), העלה בחזקה (**) וכדומה, או באמצעות פונקציות קיימות ב-SAS, כגון שורש ריבועי (sqrt), ממוצע (mean) וכדומה על פי האופן הבא:

טרנספורמציה על משתנים = שם משתנה (ישן או חדש)

דוגמאות:

הצבת סכום המשתנים x, y, z במשתנה secume:

```
secume = sum (x,y,z);
```

הצבת המשתנה x בריבוע במשתנה square:

```
square = x**2;
```

הכפלת המשתנה x ב-2 (פעולה זו דורסת את המשתנה x הישן):

```
x = x * 2;
```

מציאת הערך הנמוך ביותר מתוך המשתנים x, y, z והצבתו במשתנה minimum:

```
minimum = min(x, y, z);
```

ל-SAS פונקציות רבות מסוגים שונים לתפעול משתנים. להלן רשימה חלקית ביותר של הפונקציות העיקריות בהן נעשה שימוש בספר הדרכה זה:

1. הוצאת הערך של הפונקציה המעריכית (אקספוננציאלית):

```
expon = exp (x**2 + y);
```

2. הוצאת מעריך בסיס החזקה (log):

```
logaritem = log(2x+y);
```

3. הוצאת השורש הריבועי של משתנה (או ביטוי מתמטי):

```
root = sqrt(x);
```

4. הוצאת ערך מקסימום מתוך קבוצה של תצפיות או ביטויים מתמטיים (שים לב כי על הביטויים או המשתנים להיות מופרדים על ידי פסיקים):

```
maximum = max(x, y, z, t, r+w, q);
```

5. הוצאת ערך מינימום מתוך קבוצה של תצפיות או ביטויים מתמטיים:

```
minimum = min(x, y, z, r*q);
```

הערה: SAS מתייחסת לערך חסר כערך הקטן ביותר.

6. הוצאת הסכום של מספר תצפיות או ביטויים מתמטיים:

```
secume = sum(z, y, r, t);
```

7. הוצאת הערך האבסולוטי של תצפית או של ביטוי מתמטי:

```
absolute = abs(x-3*b);
```

8. הוצאת הממוצע של מספר תצפיות או ביטויים מתמטיים:

```
memotza = mean(x/2, y, z, r, q);
```

9. הוצאת השונות של מספר תצפיות או ביטויים מתמטיים:

```
shonot = var(x, y, z, t, r, f, g);
```

10. הוצאת סטיית התקן של מספר תצפיות או ביטויים מתמטיים:

```
teken = std(x, y, z, t, r, f, g);
```

11. הוצאת טעות התקן של מספר תצפיות או ביטויים מתמטיים:

```
err = stderr(x, y, z, t, r, f, g);
```

12. הורדת החלק העשרוני של מספר:

```
x = int(y.z);
```

בדוגמא זו, הערך של x יהיה שווה ל-y. כך, למשל, אם המספר בסוגריים היה 1.3, אזי הערך של x יהיה שווה ל-1.

הפונקציה RUNUNI

פונקציה זו מחוללת מספר אקראי מתוך התפלגות אחידה. ניתן להגדיר את טווח ההתפלגות (גבול תחתון ועליון). כאשר טווח ההתפלגות לא מוגדר, כברירת מחדל נדגם אקראית מספר מהטווח 0 ל-1.

אופן הכתיבה:

ranuni (seed);

seed הוא מספר שלם כלשהו, הקובע את נקודת ההתחלה של התהליך הבחירה האקראית. כאשר ה-seed שווה ל-0, SAS משתמשת בשעון של המחשב כ-seed. לכן, אם רוצים שכל פעם יידגם רצף שונה של מספרים, מומלץ להשתמש ב-seed = 0 (שכן הזמן משתנה כל הזמן).

דוגמא:

```
number = x*ranuni(0) + y;
```

בדוגמא זו אנחנו מחוללים מספר אקראי מהתפלגות אחידה הנעה בין y ל-x+y.

הפונקציה NORMAL

פונקציה זו מחוללת מספר אקראי מתוך התפלגות נורמלית בעלת תוחלת 0 וסטיית תקן 1.

אופן הכתיבה:

normal(seed);

הפונקציה UNIFORM

פונקציה זו זהה לפונקציה rununi וגם היא מחוללת מספר אקראי מתוך התפלגות אחידה הנעה בין 0 ל-1.

אופן הכתיבה:

uniform(seed);

ההוראות PUT ו-OUTPUT

ב-DATA STEP, ההוראה PUT אומרת ל-SAS להדפיס את הערך הנוכחי של משתנה המוגדר על ידי ההוראה בחלון Log, וההוראה OUTPUT אומרת ל-SAS לכתוב את הערך הנוכחי של משתנה לתוך קובץ הנתונים.

אופן הכתיבה:

PUT x; | OUTPUT;

דוגמא (להוראה PUT):

```
x = 5;
```

```
put x;
```

בדוגמא זו ,תופיע הספרה 5 (הערך של המשתנה x) בחלון Log. למעשה, ניתן לכתוב מחרוזת אחרי ההוראה PUT, וגם היא תופיע בחלון Log.

דוגמא:

```
put x = ;
```

בדוגמא זו, יופיע x = 5 (ולא רק הספרה 5) בחלון Log.

דוגמא (להוראה OUTPUT):

```
x = 5;
output;
```

בדוגמא זו המשתנה x יתווסף לקובץ הנתונים, כאשר עבור כל תצפית ערכו של x יהיה שווה 5.

ההוראה OUTPUT שימושית כאשר יוצרים משתנים חדשים או מתפעלים משתנים קיימים, ורוצים לאחר ביצוע הפעולה להוסיף את המשתנה המעודכן או החדש לקובץ הנתונים.

האופרטור הבוליאני	המשמעות	מילת מפתח
הסימן >	גדול מ-	Gt
הסימן <	קטן מ-	Lt
הסימן =	שווה ל-	Eq
הסימן =<	גדול שווה ל-	Ge
הסימן >=	קטן שווה ל-	Le
הסימן =^	לא שווה ל-	Ne

טבלה 2 – אופרטורים בוליאניים ומשמעותם

ההוראות IF THEN ELSE

הוראות אלה מאפשרות לתפעל משתנים המקיימים תנאי מסוים (או תנאים מסוימים).

אופן הכתיבה:

IF תנאי כלשהו או ביטוי לוגי THEN פעולה;
ELSE פעולה;

ב-SAS, תנאים או ביטויים לוגיים הינם אופרטורים בוליאניים. בנוסף, לכל ביטוי לוגי (או אופרטור בוליאני) יש מילת מפתח מקבילה. בקוד SAS ניתן להשתמש או באופרטור הבוליאני או במילת המפתח. טבלה 2 מציגה כמה מהאופרטורים הנפוצים ביותר ב-SAS, את המשמעות שלהם, ואת מילת המפתח המקבילה שלהם.

טיפ ממומחה: טעות נפוצה ב-SAS היא לסמן אי שוויון באמצעות האופרטור <> (האופייני לשפות תכנות רבות), שכן SAS לא מכירה בסימון זה, ומתייחסת אליו כאל הביטוי MAX.

הערה: ניתן להשתמש בפונקציות על משתנים הן בתנאי והן בפעולה.

דוגמאות:

1. תנאי הבודק תצפיות גיל ומציב במשתנה child את הערך 1 לכל תצפית שגילה שווה או מתחת ל-15 ואת הערך 0 לכל תצפית שגילה מעל הערך 15:

```
if gil <= 15 then child = 1;
else child = 0;
```

או

```
if gil <= 15 then child = 1;
if gil > 15 then child = 0;
```

2. תנאי הבודק האם ממוצע התצפיות x, y, z גדול מ-20. אם כן, המשתנה good מקבל את הערך המקסימאלי מבין שלושת המשתנים, ואם לא, הוא מקבל את הערך המינימאלי מבין שלושת המשתנים:

```
if mean(x, y, z) > 20 then good = max(x, y, z);
else good = min(x, y, z);
```

3. תנאי הלוקח משתנה בעל 6 ערכים (education) והופך אותו למשתנה חדש (educaion2) בעל 3 ערכים בלבד:

```
if education=1 then education2=1;
if 1 < education < 4 then education2=2;
if education > 3 or then education2=3;
```

4. תנאי היוצר משתנה חדש y, המקבל את הערך 1 כאשר המשתנה x קטן מ-10, ואת הערך 0 אחרת:

```
if x < 10 then y = 1;
else y = 0;
```

הערה: מאחר ו-SAS מתייחסת לערך חסר כאל הערך הכי קטן, בכל מצב בו יש תצפית חסרה במשתנה x המשתנה החדש y יקבל את הערך 1.

5. תנאי היוצר משתנה חדש y, המקבל את הערך 1 כאשר המשתנה x קטן מ-10:

```
if x < 10 then y = 1;
```

במצב כזה, כל פעם שהמשתנה x גדול מ-10, המשתנה y יקבל ערך חסר.

לעומת זאת, אם נכתוב:

```
if x < 10 then x = 1;
```

אזי כל פעם שהמשתנה x קטן מ-10, הוא יקבל את הערך 1, וכאשר הוא גדול מ-10, ערכו יישאר כפי שהיה.

6. תנאי היוצר משתנה חדש כאשר מתקיים תנאי של משתנה אלפאנומרי. שים לב כי במצב כזה יש לשים את ערכי המשתנה בתוך גרשיים:

```
if gender = 'M' then gender1 = 1;
if gender = 'F' then gender1 = 2;
```

כאשר מתפעלים משתנים אלפאנומריים, יש לשים לב כי SAS תלוית רישיות (דהיינו, יש הבדל בין אותיות קטנות לגדולות), וכן כי רווח נחשב אף הוא לתו (ולכן יש לוודא שאין רווחים בין הגרשיים). בנוסף, שים לב כי בהגדרת משתנים אלפאנומריים יש לכתוב את הסימן $, אך כאשר עובדים עליהם בתוכנה עצמה, לא כוללים סימן זה.

7. תנאי היוצר משתנה אלפאנומרי חדש מתוך תצפיות (בערכים מספריים) של ציוני מבחן:

```
if grade >= 90 then tzion = 'A';
if 90 > grade >= 80 then tzion = 'B';
if 80 > grade >= 70 then tzion = 'C';
if grade < 70 then tzion = 'F';
```

הערה: אם רוצים ליצור משתנה חדש בינארי, המקבל רק שני ערכים (1 או 0), לא חייבים להשתמש בהוראת if then. במקום זאת, ניתן לכתוב את הקוד הבא:

```
child = (age < 15);
```

במצב כזה, המשתנה child יקבל את הערך 1 כאשר המשתנה age קטן מ-15, ואת הערך 0 כאשר המשתנה גדול מ-15.

ביטויים לוגיים מורכבים AND,OR

ביטויים לוגיים מסוג AND ו-OR מאפשרים להשוות מספר תנאים תחת פקודה אחת (תנאים מורכבים). תנאי המורכב משני תנאים או יותר הקשורים ביניהם על ידי הביטוי הלוגי AND יתקיים רק בתנאי שכל התנאים המרכיבים אותו יהיו נכונים. לעומת זאת, כאשר התנאים קשורים ביניהם על ידי הביטוי OR, מספיק שאחד התנאים יהיה נכון כדי שהתנאי המורכב יהיה נכון.

דוגמאות:

1. תנאי מורכב הכולל את התנאי שהנבדק הוא זכר, וגילו מעל 15:

```
if (gender = 'M' and gil > 15) then mchild = 0;
```

2. כאשר התנאי מורכב ממספר ערכים של המשתנה, ניתן לכתוב אותו בשתי דרכים:

```
if (grade = 90 or grade = 80 or grade = 70) then over = 1;
```

או:

```
if grade in (90, 80, 70) then over = 1;
```

3. כאשר התנאי המורכב כולל ערכים של המשתנה שלא אמורים לקיים את התנאי, ניתן גם כאן לכתוב זאת בשני דרכים:

```
if (grade ^= 90 or grade ^= 80 or grade ^= 70) then over = 0;
```

או:

```
if grade notin (90, 80, 70) then over = 0;
```

הוראת IF מקוננת

הוראת IF מקוננת (nested if) נועדה לטפל במצבים בהם רוצים לתפעל משתנים בהתאם לקבוצות ערכים. כאשר למדת על הוראות IF THEN, ראית כי ניתן להגדיר הוראת IF THEN שונה לכל קבוצה. במצב כזה, SAS צריכה לעבור על כל ההוראות, ללא קשר לשאלה האם התקיים כבר תנאי אחד. לעומת זאת, במצב של הוראת IF מקוננת, התוכנה יוצאת מהלולאה וממשיכה לשלב הבא מיד כאשר אחד התנאים מתקיים. בנוסף, ניתן להניח בכל שלב בלולאה כי אם הגענו לשלב הנוכחי, אזי כל השלבים הקודמים לא מתקיימים. לכן, כדי ליצור קודים יעילים יותר (שגם יוצרים פחות עומס על המערכת), כדאי לכתוב בתחילת ההוראה את התנאים שהכי סביר שיתקיימו.

דוגמא:

אם נרצה לכתוב קוד שהופך ערכי ציון מספריים לערכים אלפאנומריים, נוכל לכתוב זאת כך:

```
if grade ge 90 then tzion = 'A';
if 90 > grade ge 80 then tzion = 'B';
if 80 > grade ge 70 then tzion = 'C';
if 70 > grade ge 60 then tzion = 'D';
if grade < 60 then tzion = 'F';
```

ואולם, בעזרת הוראת IF מקוננת, ניתן לכתוב קוד זה בצורה יותר אלגנטית, שגם מעמיסה פחות על התוכנה:

```
if grade ge 90 then tzion = 'A';
else if grade ge 80 then tzion = 'B';
else if grade ge 70 then tzion = 'C';
else if grade ge 60 then tzion = 'D';
else if grade ne . then tzion = 'F';
```

הערה: שים לב כי כאשר משתמשים בהוראת IF מקוננת, ניתן גם להתייחס בצורה קלה יותר לערכים חסרים. בקוד שלעיל אנחנו יודעים כי אם התוכנה הגיעה ללולאת ה ELSE IF האחרונה, הווה אומר שהציון הוא קטן מ-60. לכן, שורה זו אומרת כי אם התצפית לא מהווה ערך חסר, אזי ניתן לקודד אותה כ-F.

טיפ קריאה: למתחילים, מומלץ בשלב זה לעבור לסוף פרק זה, לחלק הדן בסינון תצפיות מקובץ.

ההוראה DO

הוראה זו תוחמת סדרה של הוראות ו/או פקודות שיתבצעו רק אם התנאי המוגדר על ידיה מתקיים. ניתן לשלב את ההוראה ELSE DO כדי לתחום סדרה נוספת של הוראות ו/או פקודות שיתבצעו אם התנאי אינו מתקיים. כמו בהוראת IF מקוננת, גם כאן ניתן להשתמש בכמה הוראות IF THEN DO שונות, או לשלב אותן עם ELSE DO, כדי ליצור קוד אלגנטי יותר שיעמיס פחות על התוכנה.

אופן הכתיבה:

IF תנאי כלשהו THEN DO;
סדרת ההוראות או הפקודות
END;
ELSE DO;
סדרה נוספת של הוראות או פקודות
END;

דוגמא (ללא ELSE DO):

```
if gil < 15 then do;
  if gender = 'M' then type = 'Boy';
  else type = 'Girl';
end;

if gil >= 15 then do;
 if gender = 'M' then type = 'Man';
  else type = 'Women';
end;
```

דוגמא (עם ELSE DO):

```
if gil < 15 then do;
  if gender = 'M' then type = 'Boy';
  else type = 'Girl';
end;

else do;
 if gender = 'M' then type = 'Man';
  else type = 'Women';
end;
```

הוראת DO מקוננת

בדומה להוראת IF, גם הוראת DO ניתנת להיות מקוננת, באמצעות הוראות IF מקוננות.

דוגמא:

```
if sub = 1 then do;
 tzion = 'B';
 grade = 8;
end;
  else if sub = 2 then do;
   tzion = 'A';
   grade = 9;
```

```
  end;
    else if sub = 3 then do;
      tzion = 'F';
      grade = 5;
    end;
     else do;
       tzion = 'C';
       grade = 7;
end;
```

לולאות DO

לולאות DO הן חזרות על קטע קוד כלשהו (הקרוי גוף הלולאה) מספר פעמים. כמות הפעמים שקטע הקוד יחזור על עצמו יכולה להיות מוגדרת על ידי אינדקס (הקובע מראש כמה פעמים קטע הקוד יחזור על עצמו) או על ידי תנאי (המגדיר את התנאים שכל עוד הם מתקיימים קטע הקוד יחזור על עצמו). ללא קשר לסוג, כל לולאה כוללת את ההוראה DO, והיא חייבת להסתיים בהוראה END (המציינת את סיומה).

אופן הכתיבה (צורה כללית):

DO …;
גוף הלולאה;
END;

<u>לולאה מוגדרת על ידי אינדקס</u>

אופן הכתיבה:

DO משתנה האינדקס = גבול אינדקס תחתון TO גבול אינדקס עליון <BY פקטור הגדלה>;
גוף הלולאה;
END;

דוגמא:

```
do i = 1 to 10;
 x = x + i;
end;
```

בדוגמא זו יצרנו לולאה שרצה מ-1 עד 10. בכל פעם שהלולאה רצה, המשתנה x מוסיף לערכו את הערך של משתנה האינדקס (i).

הערה: כברירת מחדל, אינדקס הלולאה מתקדם כל פעם ביחידת מונה אחת (מה שמכונה פקטור ההגדלה. למשל מ-1 עד 10 בצעדים של 1). עם זאת, ניתן באמצעות הפקודה BY לשנות את ברירת מחדל זאת.

דוגמאות:

```
do i = 1 to 10 by 2;
```

```
do i = 1 to 3 by 0.5;

do i = 10 to 1 by -3;
```

<u>לולאה מוגדרת על ידי תנאי עצירה</u>

אופן הכתיבה:

DO WHILE | UNTIL (תנאי כלשהו);
גוף הלולאה;
END;

ניתן להגדיר תנאי לולאה על ידי ההוראה WHILE או על ידי ההוראה UNTIL. כאשר משתמשים בהוראה WHILE, קטע הקוד מתבצע כל עוד התנאי מתקיים. לעומת זאת, כאשר משתמשים בהוראה UNTIL, קטע הקוד מתבצע עד שהתנאי יתקיים (ושהוא מתקיים, הלולאה מסתיימת). בנוסף, יש הבדל חשוב נוסף בין ההוראות WHILE ו-UNTIL. תנאי המוגדר על ידי ההוראה WHILE נבדק בטרם מתבצע קטע הקוד המוגדר בלולאה. לכן, אם התנאי לא מתקיים, קטע הקוד לא יתבצע אפילו פעם אחת. לעומת זאת, תנאי המוגדר על ידי ההוראה UNTIL נבדק לאחר ביצוע קטע הקוד. לכן, קטע הקוד המוגדר בלולאה יתבצע לפחות פעם אחת, אפילו אם התנאי המוגדר על ידי ההוראה כלל לא מתבצע.

דוגמא (עם DO WHILE):

```
data loops;
do while (x < 10);
 x + 1;
 output;
end;
```

דוגמא (עם DO UNTIL):

```
data loops;
do until (x > 9);
 x + 1;
 output;
end;
```

בשתי הדוגמאות שלהלן אנחנו יוצרים משתנה חדש (x), ונותנים לו את הערכים 1 עד 10. עם זאת, בגלל השוני בין שתי סוגי הלולאות, הלולאה DO WHILE מוגדרת לבצע את הקוד כל עוד x קטן מ-10, בעוד שהלולאה DO UNTIL מוגדרת עד ש-x יהיה גדול מ-9 (שכן לולאה זו קודם מבצעת את קטע הקוד ורק אחרי זה היא בודקת את התנאי).

הערה: בדומה להוראות IF THEN, ניתן ליצור גם לולאות DO מקוננות.

טיפ ממומחה: כאשר עובדים עם לולאות, יש להזהר לא ליצור לולאות אין-סופיות (מה שיגרום ל-SAS לעבוד ללא הפסקה).

מערכים

מערכים ב-SAS שימושיים כאשר אנו מעוניינים לבצע פעולה זהה על קבוצה של משתנים. המערך הוא קיבוץ זמני של משתנים המאורגנים בסדר מסוים. המשתנים האלה מהווים את האלמנטים של המערך, והם מזוהים על ידי שם המערך ומיקומם (מבחינת הסדר) בתוכו. כל המשתנים במערך מקבלים את אותו שם, והם מובחנים רק על ידי מיקומם (האינדקס) בתוכו (לדוגמא, VAR1, VAR2,…,VAR*n*). עם זאת, יש לציין כי איברי המערך עצמם אינן משתנים. הערכים של איברי המערך מוצבים (בערכם המקורי בתוך המערך) לתוך משתנים, כאשר שמות משתנים אלה מוגדרים אוטומטית על ידי SAS (בהתאם לשם המערך) או על ידי המשתמש, כפי שיפורט בהמשך.

מערך הוא זמני שכן הוא קיים (וזמין למשתמש) רק בתוך ה-DATA STEP בו הוא מוגדר. כדי להגדיר מערך ב-SAS, יש להשתמש בהוראה ARRAY.

אופן הכתיבה:

ARRAY שם המערך {מספר האיברים במערך} <$> <שמות המשתנים במערך>
<(ערכים ראשוניים של איברי המערך)>

דוגמא:

```
array x{8} var1-var8 (1 2 3 4 5 6 7 8);
```

בדוגמא זו יצרנו מערך בשם x, המקושר למשתנים var1 עד var8. דהיינו, כל ערך שיוגדר בתא i של המערך x יעודכן אוטומטית בערך של המשתנה var*i*, ולהפך, כל ערך שיוגדר למשנתה var*i* יעודכן אוטומטית גם כערך בתא i של המערך x. בנוסף, בהגדרת המערך, הגדרנו כי הערך הראשוני של המשתנים var1 עד var8 יהיה 1 עד 8 (כך שהמשתנה var5 יקבל את הערך 5, המשתנה var6 יקבל את הערך 6 וכן הלאה). לכן, מבחינת SAS התא ה-i במערך x והמשתנה var*i* הם היינו הך.

למרות שבדוגמא ציינו את שמות המשתנים במערך, אין זה הכרחי. אם שמות המשתנים לא יוגדרו , SAS תיצור אותם באופן אוטמטי.

לדוגמא:

```
array x x1-x8;
```

או

```
array x{8};
```

בשתי הדוגמאות SAS תיצור 8 משתנים חדשים (x1 עד x8), ותקשר אותם למערך x.

מערכים ב-SAS יכולים להיות מוגדרים כוקטורים (דהיינו מערכים בעלי ממד אחד) או כמטריצות (דו או רב ממדיות). כדי להגדיר מטריצה רב ממדית, יש להגדיר את מספר האיברים בכל ממד בתוך הסוגריים המסולסלים, כאשר כל מספר מופרד על ידי פסיק. עם זאת, מבחינת SAS, אין משמעות למספר הממדים במערך. לדוגמא:

שלוש ההגדרות הבאות יוצרות קבוצה של 8 משתנים – המשתנה x1 עד המשתנה x8:

מערך חד ממדי בעל 8 איברים:

```
array y{8};
```

מערך דו ממדי בעל 8 איברים (מטריצה של 4X2):

```
array y{2, 4};
```

מערך תלת ממדי בעל 8 איברים (2 מטריצות של 2X2):

```
array y{2, 2, 2};
```

הגדרת ערך של איבר במערך נעשית בדיוק באותה צורה בה אנחנו מגדירים ערכי משתנים ב-SAS. עם זאת, ההבדל הוא שבהגדרת איבר במערך צריך להגדיר ל-SAS לא רק את שם המערך (בדומה להגדרת שם המשתנה) אלא גם את האינדקס שלו (את מספר התא), על ידי ציון מספר התא בתוך סוגריים מסולסלים אחרי שם המערך.

אופן הכתיבה:

שם המערך {מספר התא} = ערך כלשהו

דוגמא:

```
x{5} = 3;
```

בדוגמא זו אנחנו מגדירים ל-SAS כי האיבר ה-5 במערך x יקבל את הערך המספרי 3. עם זאת יש לציין כי הפנייה למערך כלשהו (למשל לקביעת ערך של משתנה) יכולה להיעשות רק ב-DATA STEP שבו מוגדר המערך.

השימוש במערכים יכול להיות מאוד שימושי כדי לבצע את אותה פעולה על מספר משתנים, והוא יכול לחסוך הן בזמן הקלדה והן בזמן עיבוד. לדוגמא, הנח כי קיים קובץ נתונים בשם arrays, הכולל את המשתנים x1 עד x10. אתה רוצה לכתוב קוד SAS שיחליף את סימני הנקודה (המציינים ערכים חסרים) בקובץ הנתונים בספרה 0. ללא שימוש במערכים, הקוד יהיה:

```
data arrays; set arrays;
 if x1 = . then x1 = 0; if x2 = . then x2 = 0;
 if x3 = . then x3 = 0; if x4 = . then x4 = 0;
 if x5 = . then x5 = 0; if x6 = . then x6 = 0;
 if x7 = . then x7 = 0; if x8 = . then x8 = 0;
 if x9 = . then x9 = 0; if x10 = . then x10 = 0;
run;
```

לעומת זאת, תוך שימוש במערך, ניתן לבצע את אותה פעולה בצורה הרבה יותר קלה וחסכונית:

```
data arrays2; set arrays;
 array x {10};
 do i = 1 to 10;
  if x{i} = . then x{i} = 0;
 end;
run;
```

סינון תצפיות מקובץ

ניתן לבחור תצפיות מסוימות או למחוק תצפיות באמצעות ההוראות IF או WHERE. הסינון נעשה באמצעות תנאים, אותם מגדירים באמצעות האופרטורים הבוליאניים, כפי שלמדנו בפרק הקודם. בדומה, ניתן להשתמש בתנאים גם בפונקציות של משתנים.

ההוראה IF

בהוראה זו SAS קודם כל כותבת את קובץ הנתונים. לאחר מכן, הסינון מתבצע בהתאם למיקום של ההוראה ב- DATA STEP. לכן, את ההוראה IF ניתן לבצע גם על משתנים שהוגדרו במהלך ה-DATA STEP, בתנאי שההוראה מופיעה בקוד לאחר הגדרת המשתנים. עם זאת, את ההוראה IF ניתן להחיל בשלב ה-DATA STEP בלבד.

אופן הכתיבה:

IF תנאי או מספר תנאים;

דוגמאות:

1. שמירת תצפיות של משתנה שאינן ערכים חסרים או תצפיות שערכן אפס:

```
if grade;
```

בדוגמא זו SAS תשמור את כל התצפיות של המשתנה grade שערכן שונה מאפס ושהן לא ריקות (ערכים חסרים).

2. מחיקת תצפיות מסוימות מהקובץ:

```
if gender = 'M' then delete;
```

בדוגמא זו SAS תמחק את כל התצפיות (של כל המשתנים) של כל מי שהמין שלו הוא זכר (מוגדר כ-M בקובץ).

3. יצירת קובץ חדש מקובץ נתונים קיים, תוך שמירה על משתנה חדש המוגדרת ב-data step:

```
data sascodes2; set sascodes;
 mean_g = mean(grade1,grade2,grade3);
 if mean_g > 50;
run;
```

בדוגמא זו SAS קוראת את הקובץ sascodes במלואו (קובץ הכולל נתונים על מין וגיל הנבדק, ועל 3 ציונים של מבחנים), מגדירה את המשתנה mean_g (ממוצע שלושת הציונים), ושומרת בקובץ sascodes2 רק את התצפיות להן ממוצע ציונים גבוה מ-50.

ההוראה WHERE

בהוראה זו SAS קודם כל מבצעת את הסינון, ללא קשר למיקום של ההוראה. רק לאחר מכן מתבצע הרישום של קובץ הנתונים או הפרוצדורה. לכן, בניגוד להוראה IF, ניתן להשתמש בהוראה WHERE גם ב-PROC STEP. עם זאת, את ההוראה WHERE לא ניתן לבצע על משתנים שהוגדרו במהלך ה-DATA STEP, אלא רק על משתנים המופיעים בקובץ

input. בנוסף, לא ניתן להשתמש בהוראה WHERE כאשר קוראים נתונים מקובץ חיצוני או כאשר כותבים את הנתונים בתוך ה-DATA STEP, אלא רק כאשר קוראים נתונים מתוך קובץ של SAS (קבוע או זמני).

אופן הכתיבה:

WHERE תנאי או מספר תנאים

דוגמא:

```
where gil > 25 and gender = 'M';
```

דוגמא זאת אומרת ל-SAS לכלול רק את התצפיות בהן הגיל הוא מעל 25 והמין הוא זכר.

בנוסף, ניתן להשתמש בהוראה WHERE גם כאופציה של ההוראה DATA (הן בהגדרת הקובץ החדש והן בהגדרת הקובץ הישן). עם זאת, לא ניתן להשתמש ב-WHERE ביחד עם האופציות obs ו-firstobs של ההוראה DATA.

כאשר משתמשים בהוראה WHERE בהגדרה של קובץ הנתונים הישן, השימוש בה זהה לשימוש בהוראה WHERE בתוך ה-DATA STEP.

דוגמא:

```
data sascodes2; set sascodes (where gil > 25);
```

או,

```
data sascodes2; set sascodes;
where gil > 25;
```

בשני המקרים SAS תקרא מהקובץ הישן sascodes רק את התצפיות בהן הגיל גדול מ-25, תבצע את כל הפעולות שיוגדרו על ידי המשתמש ב-DATA STEP, ולאחר מכן תשמור כקובץ נתונים sascodes2 את כל התצפיות שנקראו.

לעומת זאת, כאשר משתמשים בהוראה WHERE בהגדרה של קובץ הנתונים החדש, SAS תקרא את קובץ הנתונים הישן במלואו, תבצע את כל הפעולות שיוגדרו על ידי המשתמש ב-DATA STEP, ואז תשמור בקובץ החדש sascodes2 את התצפיות שעבורן הערך של משתנה הגיל גדול מ-25.

דוגמא:

```
data sascodes2; set sascodes (where gil > 25);
where gil > 25;
```

נעשה שימוש בהוראה WHERE בהגדרה של קובץ הנתונים החדש כאשר נרצה לסנן תצפיות ממשתנים שנוצרו ב- DATA STEP.

תרגול עצמי – תפעול משתנים

תרגיל 6

כתוב תוכנית SAS בה כתיבת הנתונים מתבצעת בתוך ה-DATA STEP, כאשר הנתונים הקיימים מופרדים על ידי רווחים. הנותנים כוללים 4 תצפיות ו-3 משתנים לכל תצפית: מספר סידורי של הנבדק, ציון מבחן ראשון וציון מבחן שני. בהגדרת קובץ הנתונים, צור משתנה חדש המכיל את ממוצע ציוני שני המבחנים.

תרגיל 7

נתון קובץ הנתונים הבא, הכולל משקל של ארבעה נבדקים (בק"ג) בארבע תקופות שונות:

```
Sub weight1 weight2 weight3 weight4
1   65   70   72   62
2   80   65   71   70
3   76   66   80   79
4   81   80   83   80
```

כתוב קוד SAS היוצר משתנה חדש בשם min_weight. משתנה זה יכיל עבור כל נבדק את המשקל המינימאלי שלו מתוך ארבעת המדידות הקיימות.

תרגיל 8

לאחר איסוף נתונים על מחקר הבודק הבדלים בין נשים לגברים בביצוע מטלה כלשהי, נמצא כי בקידוד תוצאות המבחן של הגברים נעשתה טעות, וכי הציון של כל הנבדקים הגברים למעשה גבוה יותר ב-3 נקודות מהציון שקודד. ידוע כי המשתנה המכיל את ציוני הנבדקים הוא המשתנה test, וכי המשתנה המכיל את נתוני המגדר הוא המשתנה gender. כמו כן, ידוע כי הערך "גבר" במשתנה gender מיוצג על ידי הספרה 0.

כתוב תוכנית SAS שתוסיף 3 נקודות למשתנה test רק עבור הנבדקים הגברים.

תרגיל 9

נתון קובץ הנתונים הבא, הכולל את התשובות שנתנו סטודנטים במבחן סיום באחד הקורסים שלהם (כאשר 1 מסמן תשובה נכונה ו-0 תשובה שגויה):

```
Sub q1 q2 q3 q4 q5 q6 q7 q8 q9 q10
1 1 0 1 1 1 0 0 1 0 1
2 0 1 0 0 1 0 0 1 0 0
3 1 1 1 0 1 0 1 1 1 1
4 1 0 0 1 1 1 1 1 1 0
5 0 1 1 1 1 1 1 1 1 1
6 1 0 1 0 1 0 1 1 1 1
```

המבחן כלל 10 שאלות, כך שכל שאלה נכונה מזכה את הסטודנט ב-10 נקודות.

א. כתוב תוכנית שתחשב את הציון של כל סטודנט במבחן.

ב. כתוב תוכנית שתייחס לכל סטודנט משתנה אלפאנומרי שמציין האם הוא עבר או לא עבר את המבחן, כאשר ציון עובר הוא כל ציון הגבוה מ-65.

ג. ידוע כי כל הסטודנטים בקורס היו גברים. כתוב תוכנית SAS שתאחד בין הקובץ הנתון לקובץ נתונים המכיל משתנה אחד: מין הסטודנט.

תרגיל 10

נתון הקוד הבא המקודד נבדקים לקבוצות בהתאם לגיל שלהם:

```
If age <= 10 then group = "child       ";
If age => 11 and age <= 19 then group = "teenager   ";
If age => 20 and age <= 29 then group = "young adult";
If age => 30 and age <= 45 then group = "adult      ";
If age => 46 and age <= 59 then group = "middle age ";
If age => 60 then group = "senior      ";
```

כתוב את הקוד בצורה יעילה יותר, תוך שימוש בהוראת IF THEN מקוננות.

תרגיל 11

כתוב תוכנית SAS שתיצור קובץ נתונים המכיל שני משתנים: x ו-y. המשתנה x יקבל ערכים אקראיים בין 1 ל-10, והמשתנה y יקבל ערכים אקראיים בין 0 ל- 10. על כל משתנה לכלול 10 תצפיות.

תרגיל 12

כתוב תוכנית SAS שתיצור קובץ נתונים בעל 40 תצפיות. המשתנים הכלולים בקובץ הנתונים הם sub, המציין את מספר התצפית, grade1, grade2 ו-grade3, המכילים כל אחד מהם מספר אקראי שלם בין 0 ל-100, ו-ave_grade, המכיל את הממוצע של grade1 עד grade3. יש ליצור את הקוד תוך שימוש במערך.

פרק 5
צירוף קבצים

ישנן שתי הוראות שונות לצירוף או מיזוג קבצי נתונים ב-SAS. הוראות אלה מופיעות ב-DATA STEP והן מאפשרות ליצור קובץ נתונים חדש ממספר קבצי נתונים קיימים. קובץ הנתונים החדש שנוצר יכול להכיל את כל הנתונים הקיימים בקבצי הנתונים המאוחדים, רק חלק מהמשתנים, משתנים שעברו טיפול, ומשתנים חדשים.

טיפ קריאה: צירוף קבצים היא פעולה מתקדמת, שלמתחילים מומלץ לדלג בשלב זה. קודם כדאי לבדוק שניתן להפעיל את התוכנה ולרכוש בה מיומנות עם קובץ קלט אחד...

ההוראה SET

בפרקים קודמים למדנו שההוראה SET מאפשרת ליצור קובץ נתונים חדש מקובץ נתונים קיים. עם זאת, ניתן להשתמש בהוראה SET גם כדי לצרף קבצי נתונים קיימים, כאשר אופן הצירוף על פי הוראה זו הוא קובץ נתונים אחד מתחת לשני.

אופן הכתיבה:

DATA שם של קובץ נתונים חדש; SET שמות של מספר קבצי נתונים קיימים;

דוגמא:

```
data targil; set targil1 targil2;
```

במצב זה, קובץ הנתונים targil הוא צירוף של שני קבצי הנתונים הקיימים targil1 ו-targil2. לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

age height
23 175
26 180
22 167

וקובץ הנתונים targil2 הכיל את הרשומות:

age height
28 168
32 170
25 189

אזי הקובץ החדש targil יכיל את כל הרשומות הללו, בסדר שהן מופיעות בהוראה:

age height
23 175
26 180
22 167
28 168

32 170
25 189

במצב שבו יש בקבצי הנתונים משתנים אחרים, SAS תיתן למשתנים שאינם קיימים ערכים חסרים. לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

age height
23 175
26 180
22 167

וקובץ הנתונים targil2 הכיל את הרשומות:

age weight
28 75
32 80
25 92

אזי הקובץ החדש targil יכיל את כל הרשומות הללו:

age height weight
23 175 .
26 180 .
22 167 .
28 . 75
32 . 80
25 . 92

האופציה IN

אופציה זו יוצרת משתנה בינארי חדש, המציין מאיזה קובץ הגיעה התצפית. משתנה זה מקבל את הערך 1 כאשר התצפית הגיעה מקובץ הנתונים הרלוונטי, ו-0 כאשר התצפית הגיעה מקובץ אחר.

אופן הכתיבה:

DATA שם של קובץ נתונים חדש;
SET שם של קובץ נתונים קיים (in = שם המשתנה הבינארי) שם קיים (in = שם משתנה בינארי נוסף);

דוגמא:

```
data targil; set targil1 (in = in1) targil2 (in = in2);
```

אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

age height
23 175
26 180
22 167

וקובץ הנתונים targil2 הכיל את הרשומות:

age height
28 168
32 170
25 189

אזי הקובץ החדש targil יכיל את כל הרשומות הללו:

age height in1 in2
23 175 1 0
26 180 1 0
22 167 1 0
28 168 0 1
32 170 0 1
25 189 0 1

הערה: למרות שהמשתנים in1 ו-in2 קיימים בקובץ הנתונים החדש (targil), הם לא יופיעו בפלט הקובץ. לכן, אם רוצים שהם יופיעו, יש להגדיר אותם ב-data step כקבצים חדשים (בעלי שמות שונים מאלה שהוגדרו באופציה IN).

דוגמא:

```
data targil; set targil1 (in = in1) targil2 (in = in2);
ina = in1; inb = in2;
```

ההוראה BY

הוראה זו מאפשרת לצרף קבצי נתונים לפי סדר (עולה או יורד) של משתנה או מספר משתנים. עם זאת, כדי להשתמש בהוראה זו, יש לוודא כי הקבצים מסודרים לפי הסדר (עולה או יורד, תלוי במה שרוצים) לפני צירופם.

אופן הכתיבה:

DATA שם חדש; SET שם של מספר קבצי נותנים קיימים;
BY שם של משתנה;

דוגמא:

```
data targil; set targil1 targil2;
by age;
```

בקובץ הנתונים החדש שייווצר (targil), קבצי הנתונים הקיימים יופיעו לפי סדר הערכים של משתנה הגיל:

age height
22 167
23 175
25 189
26 180
28 168
32 170

הערה: ברירת המחדל של SAS היא לסדר את התצפיות בסדר עולה. לכן, אם רוצים לסדר את התצפיות בסדר יורד, יש להוסיף את האופציה descending לפני שם המשתנה.

דוגמא:

```
data targil; set targil1 targil2;
by descending age;
```

מיזוג קובץ עם תצפית אחת עם קובץ עם מספר תצפיות

כאשר רוצים לצרף תצפית אחת בודדת לכל סדרת תצפיות בקובץ נתונים גדול יותר, לא ניתן להשתמש בהוראה SET באופן הרגיל.

דוגמא:

נתונים שני קבצי הנתונים הבאים:

Targil1:

Sub grade
1 80
2 75
3 90
4 67
5 80

Targil2:

gender
M

אם נריץ את ההוראה SET על שני קבצי נתונים אלה (באמצעות הקוד):

```
data targil; set targil1 targil2;
```

נקבל את הקובץ המאוחד הבא:

1 80 .
2 75 .
3 90 .
4 67 .
5 80 .
. . M

כדי לצרף את המשתנה "מין" לכל אחת מהתצפיות, ניתן להשתמש בפקודה של SAS הסופרת את מספר התצפיות הקיימות בקובץ נתונים, ולקבוע כי כל פעם שהספירה מגיעה ל-1 (מספר התצפיות בקובץ הנתונים השני), SAS צריכה להכניס את המשתנה מקובץ זה לקובץ המאוחד.

אופן הכתיבה:

IF _n_ = מספר כלשהו THEN SET שם של קובץ נתונים קיים

דוגמא:

```
data targil; set targil1;
 if _n_ = 1 then set targil2;
run;
```

במקרה הזה, קובץ הנתונים החדש targil יראה כך:

sub grade gender
1 80 M
2 75 M
3 90 M
4 67 M
5 80 M

ההוראה MERGE

ההוראה MERGE מצרפת קבצי נתונים קיימים, כאשר אופן הצרוף על פי הוראה זו הוא קובץ נתונים אחד ליד השני.

אופן הכתיבה:

DATA שם של קובץ נתונים חדש; MERGE שם של מספר קבצי נתונים קיימים;

דוגמא:

```
data targil; merge targil1 targil2;
```

במצב כזה, קובץ הנתונים targil הוא מיזוג של שני קבצי הנתונים הקיימים targil1 ו-targil2, לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

age height
23 175
26 180
22 167

וקובץ הנתונים targil2 הכיל את הרשומות:

weight gender
80 M
95 M
50 F

אזי הקובץ החדש targil יכיל את כל הרשומות הללו, אחת ליד השנייה, בסדר שהן מופיעות בהוראה:

age height weight gender
23 175 80 M
26 180 95 M
22 167 50 F

כאשר בשני קבצי הנתונים שאנו רוצים למזג אין את אותו מספר התצפיות, SAS תיצור את הקובץ החדש לפי הקובץ עם מספר התצפיות המקסימאלי, והמשתנה החסר בקובץ הנתונים הקטן יותר יקבל ערך חסר. לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

```
age height
23 175
26 180
22 167
27 182
```

וקובץ הנתונים targil2 הכיל את הרשומות:

```
weight gender
80 M
95 M
50 F
```

אזי הקובץ החדש targil יכיל את כל הרשומות הללו, אחת ליד השנייה, בסדר שהן מופיעות בהוראה:

```
age height weight gender
23 175 80 M
26 180 95 M
22 167 50 F
27 182 . .
```

במקרה בו קבצי הנתונים שאנו רוצים למזג כוללים משתנה בעל אותו שם, SAS תדרוס את התצפיות של משתנה זה מהקובץ הראשון, והקובץ החדש יכיל את הערכים של המשתנה מהקובץ השני. לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

```
age height
23 175
26 180
22 167
```

וקובץ הנתונים targil2 הכיל את הרשומות:

```
age weight
28 75
32 80
25 92
```

אזי הקובץ החדש targil יכיל את כל הרשומות הללו:

```
age height weight
28 175 75
32 180 80
25 167 92
```

האופציה IN

כמו בהוראה SET, גם כאן האופציה in יוצרת משתנה בינארי חדש המציין מאיזה קובץ הגיעה התצפית.

אופן הכתיבה:

DATA שם של קובץ נתונים חדש;
MERGE שם של קובץ נתונים קיים (שם המשתנה הבינארי = in) שם קיים (שם משתנה בינארי נוסף = in);

דוגמא:

```
data targil; merge targil1 (in = in1) targil2 (in = in2);
```

גם כאן, המשתנים הבינאריים קיימים בקובץ, אבל הם לא נראים בפלט, כך שאם רוצים לראות אותם יש להגדיר אותם ב-DATA STEP כמשתנים חדשים (בעלי שמות שונים).

ההוראה BY

גם באמצעות ההוראה BY, בדומה להוראה SET, ניתן לצרף קבצי נתונים על פי סדר של משתנה או מספר משתנים.
התנאים החלים על ההוראה SET בהקשר זה זהים גם להוראה MERGE.

אופן הכתיבה:

DATA שם חדש; MERGE שם של מספר קבצי נותנים קיימים;
BY שם של משתנה;

דוגמא:

```
data targil; merge targil1 targil2;
by sub;
```

לדוגמא, אם קובץ הנתונים targil1 הכיל את הרשומות הבאות:

sub age height
1 23 175
2 26 180
3 22 167

וקובץ הנתונים targil2 הכיל את הרשומות:

sub weight
1 78
2 80
3 50

אזי הקובץ החדש targil יכיל את כל הרשומות הללו:

sub age height weight
1 23 175 78
2 26 180 80
3 22 167 50

תרגול עצמי – מיזוג קבצים

תרגיל 13

נתונים שני קבצי הנתונים הבאים:

Sub var1 var2
1 1 10
2 2 20
3 3 30

Sub var1 var2
4 4 40
5 5 50
6 6 60

מה הצורה הפשוטה ביותר לאחד בין שני קבצים אלה?

תרגיל 14

נתונים שני קבצי הנתונים הבאים:

Targil14a:

height weight
168 56
174 85
180 75
180 80
190 96

Targil14b:

height gender$
169 F
170 F
180 F
181 M
190 M

א. צור קובץ נתונים חדש, שיהווה מיזוג של שני קבצי הנתונים הקיימים. את המיזוג יש לבצע על פי המשתנה "גובה".
ב. צור קובץ נתונים חדש שיכלול רק את הנתונים הקיימים בשני הקבצים.

פרק 6
פרוצדורות שירות I: מיון והפקת פלט

ה-PROC STEP

ניתן לבצע ניתוחים סטטיסטיים ותפעול נתונים ב-SAS או באמצעות ה-DATA STEP (באמצעות כתיבת תוכנה מותאמת אישית על ידי המשתמש) או באמצעות פרוצדורות SAS מובנות אשר נועדו לבצע חישובים וניתוחים, כמו גם להציג ולהדפיס את התוצאות של חישובים וניתוחים אלה. השימוש בפרוצדורות SAS המובנות נעשה באמצעות ה-PROC STEP.

הצעד PROC (PROC STEP) מכיל את כל ההוראות הדרושות לביצוע ניתוחים או עיבודים על קובץ הנתונים שנוצר בשלב ה- DATA STEP. בנוסף, ה-PROC STEP כולל פרוצדורות לביצוע ניתוחים סטטיסטיים ועיבודים על קבצי נתונים, אופציות לפרוצדורות אלה, הגדרת המשתנים המשתתפים בעיבוד ובניתוח, וכן גם את אופי הפלט.

ה-PROC STEP מתחיל בהוראה PROC ומסתיים בהוראה RUN:

```
PROC שם הפרוצדורה;
.................
.................
run;
```

PROC SORT

הפרוצדורה SORT ממיינת את התצפיות הקיימות בקובץ נתונים, על פי משתנה אחד או יותר המוגדרים על ידי המשתמש, או בסדר עולה (ברירת מחדל) או בסדר יורד. את המיון ניתן לבצע ישירות על קובץ הנתונים הקיים או ליצור קובץ נתונים חדש המכיל את התצפיות הממוינות.

אופן הכתיבה:

PROC SORT <אופציות שונות>;
BY רשימת המשתנים על פיהם יתבצע המיון;
run;

דוגמא:

```
proc sort;
 by age;
run;
```

ההוראה BY

מיון התצפיות נעשה על פי המשתנה/משתנים המוגדרים בהוראה BY. כדי לכתוב מספר משתנים על פיהם ימוין קובץ הנתונים, יש לכתוב אותם לאחר ההוראה BY, כאשר השמות מופרדים על ידי רווחים. ברירת המחדל של SAS היא למיין תצפיות על פי סדר עולה. לכן, אם רוצים למיין לפי סדר יורד, צריך להוסיף את הפקודה descending לפני שם המשתנה הרצוי. SAS תמיין את התצפיות בקובץ לפי כל המשתנים המופיעים לאחר ההוראה BY, כאשר המשתנה השמאלי ביותר יהיה משתנה המיון הראשון וכך הלאה לפי הסדר משמאל לימין.

דוגמא:

```
proc sort;
 by age descending weight;
run;
```

בדוגמא זו קובץ הנתונים ימוין ראשית לפי משתנה הגיל (מהקטן לגדול) ולאחר מכן לפי משקל (מהגדול לקטן). לכן, אם יש לנו קובץ המכיל תצפיות גיל ומשקל כדלקמן:

age weight
23 75
32 89
23 80
27 65
26 90
26 75
22 49
30 82
27 69

לאחר המיון הקובץ ייראה כך:

age weight
22 49
23 80
23 75
26 90
26 75
27 69
27 65
30 82
32 89

טיפ קריאה: הפרוצדורות השונות של SAS לא מחייבות הגדרה של אופציות לשם הרצתן. לכן, מומלץ לדלג בשלב הראשון על תת-הפרקים העוסקים באופציות של הפרוצדורות השונות.

1. האופציה data – אופציה זו מגדירה ל-SAS על איזה קובץ נתונים יש לבצע את המיון. אם לא מגדירים קובץ נתונים מסוים, SAS תעשה מיון על קובץ הנתונים האחרון עליו נעשתה עבודה.

הערה: האופציה data זהה מבחינת הגדרה ומבחינה פונקציונאלית לרוב הפרוצדורות, ולכן בפרקים הבאים, בהם נדון בפרוצדורות נוספות, נציין אופציות ייחודיות לכל פרוצדורה.

אופן הכתיבה:

PROC SORT data = שם של קובץ נתונים;

דוגמא:

```
proc sort data = dogma;
```

2. האופציה out – אופציה זו יוצרת קובץ נתונים חדש המכיל את המשתנים הממוינים לפי פרוצדורת המיון.

הערה: האופציה out זהה מבחינת הגדרה ומבחינה פונקציונאלית לכל הפרוצדורות.

אופן הכתיבה:

PROC SORT data = שם של קובץ נתונים out = שם חדש;

דוגמא:

```
proc sort data = dogma out = dogma2;
```

PROC PRINT

רוב הפרוצדורות ב-SAS, למעט פרוצדורות המטפלות בקבצי נתונים (כגון SORT, או פרוצדורות אחרות שנלמד עליהן בהמשך, כגון IMPORT, TRANSPOSE) מפיקות באופן אוטומטי פלט. הפרוצדורה PRINT אומרת ל-SAS להדפיס אל החלון Output תצפיות מקובץ נתונים של SAS, תוך שימוש בחלק או בכל המשתנים, על פי הגדרת המשתמש (ראה לדוגמא איור 10).

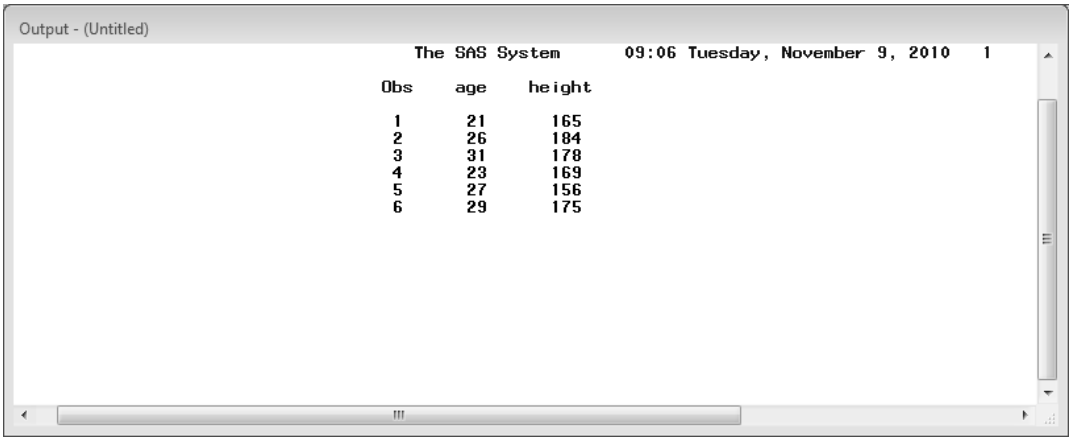
טיפ ממומחה: מומלץ להשתמש בפרוצדורה PROC PRINT מיד לאחר קליטת קובץ נתונים. כך ניתן לוודא שכל הנתונים נקלטו בשלמותם, ושלא חל בלבול בין שדות שונים.

אופן הכתיבה:

PROC PRINT<אופציות שונות>;
BY <descending> שם/שמות משתנים מופרדים על ידי רווחים < notsorted>;
PAGEBY by -משתני ה;
ID שם/שמות משתנים מופרדים על ידי רווחים;
SUM שמות משתנים מופרדים על ידי רווחים;
VAR שמות משתנים מופרדים על ידי רווחים;
RUN;

דוגמא:

```
proc print;
 var age height;
run;
```



איור 10 – דוגמא לפלט בסיסי של הפרוצדורה PRINT

אופציות של PROC PRINT

1. האופציה double – אופציה זו מוסיפה שורה ריקה בין כל שורת תצפיות בקובץ הפלט.
אופן הכתיבה:

PROC PRINT double;

דוגמא:

```
proc print double;
```

2. האופציה n – אופציה זו אומרת ל-SAS להוסיף לקובץ הפלט את מספר התצפיות (גודל המדגם) הכללי בקובץ הנתונים, וכן להוסיף את מספר התצפיות של כל תת קבוצה (כאשר תת קבוצה מוגדרת על ידי ההוראה BY, כפי שיפורט בהמשך). בנוסף, האופציה n מאפשרת גם להגדיר מחרוזת טקסט שתוצמד למספר התצפיות (הן למספר התצפיות הכללי והן למספר של כל תת קבוצה).
אופן הכתיבה:

PROC PRINT n = "מחרוזת1" "מחרוזת2";

דוגמא:

```
proc print double n = "sub = " "Overall = ";
```

הערה: מחרוזת1 (המחרוזת הראשונה המופיעה לאחר האופציה n) מתארת את גודל המדגם של המשתנים הספציפיים המוגדרים על ידי ההוראה by, ומחרוזת2 (המחרוזת השנייה) מתארת את גדול המדגם הספציפי (ראה דוגמא באיור 13). לא ניתן להגדיר יותר משתי מחרוזות על ידי האופציה n.

3. האופציה noobs – כאשר מפיקים פלט באמצעות PROC PRINT, SAS מוסיפה באופן אוטומטי עמודה של תצפיות (obs), הממוספרת מ-1 (שורת התצפיות הראשונה) עד *n* (שורת התצפיות האחרונה), כדי לשייך מספר לכל

שורת תצפית (ראה לדוגמא את העמודה השמאלית באיור 10). ההוראה noobs אומרת ל-SAS לא להוסיף עמודה זו.
אופן הכתיבה:

PROC PRINT noobs;

דוגמא:

```
proc print noobs;
```

4. האופציה obs – אופציה זו מגדירה ל-SAS כיצד לקרוא לעמודה obs המתווספת אוטומטית לפלט (ראה לדוגמא איור 13).
אופן כתיבה:

PROC PRINT obs = "שם העמודה";

דוגמא:

```
proc print obs = "subject";
```

5. האופציה heading – אופציה זו מגדירה את הכיוון של כותרות העמודות של קובץ הפלט. כיוון הכותרות יכול להיות אופקי (horizontal) או אנכי (vertical, ראה דוגמא באיור 11). כותרות העמודות יכולות להיות כולן אופקיות או כולן אנכיות. כברירת מחדל, SAS קובעת את הכיוון של הכותרות בהתאם לאורך שלהם (כותרות קצרות הן אופקיות).
אופן הכתיבה:

PROC PRINT heading = הכיוון הרצוי

דוגמא:

```
proc print data=dogma heading = vertical n;
 var age height sex;
run;
```

6. האופציה round – אופציה זו אומרת ל-SAS לעגל את המספר העשרוני המופיע בפלט לעד 2 מספרים אחרי הנקודה.
אופן הכתיבה:

PROC PRINT round;

דוגמא:

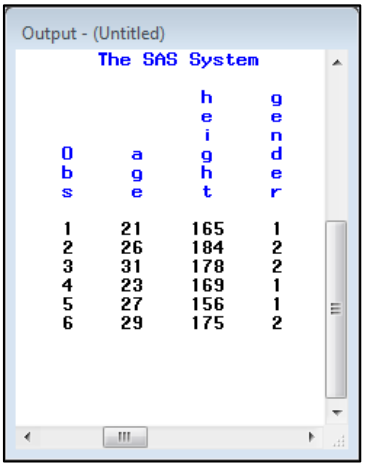
```
proc print round;
```

7. האופציה split – אופציה זו מגדירה את התו המציין את סוף השורה ומעבר לשורה חדשה בכותרות של עמודות. תו ה-split נועד רק לציון סוף השורה, והוא אינו כלול בכותרת עצמה. יש לציין כי האופציה obs מצייתת לאופציה split.
אופן הכתיבה:

PROC PRINT split = 'התו המפריד';

דוגמא:

```
proc print data=dogma split = '*';
```



איור 11 – כותרות אנכיות ב-PROC PRINT

ההוראה BY

הוראה זו אומרת ל-SAS להפיק חלק נפרד של הפלט לכל קבוצת ערכים של המשתנה המוגדר על ידי ההוראה (לדוגמא, באיור 12 ב' הפלט מחולק לפי מין וגיל הנבדק). כאשר משתמשים בהוראה BY יש לוודא שקובץ הנתונים ממוין על פי המשתנה המוגדר על ידי ההוראה. לחילופין, במקרה שזה לא קורה, ניתן להשתמש באופציה notsorted. עם זאת, יש לציין כי כאשר משתמשים באופציה זו, SAS יוצרת חלק נפרד בכל פעם שהמשתנה המוגדר על ידי ההוראה by משנה את ערכו (דהיינו, בכל פעם שערך המשתנה בתצפית t שונה מערך המשתנה בתצפית $t+1$).

אופן הכתיבה:

BY <descending> משתנה1…<descending> *n* משתנה <notsorted>;

דוגמא:

```
proc print noobs;
 by sex;
run;
```

טיפ ממומחה: טעות נפוצה היא לא למיין את הנתונים לפי משתנה המוגדר בהוראה BY, לפני שמגדירים אותה בפרוצדורה.

ההוראה VAR

הוראה זו מגדירה ל-SAS אילו משתנים יש להציג בקובץ הפלט, ובאיזה סדר. כאשר הוראה זו מושמטת מהקוד, SAS תדפיס את כל המשתנים בקובץ.

אופן הכתיבה:

VAR שמות המשתנים לפי הסדר הרצוי;

דוגמא:

```
proc print;
 var age height;
run;
```

ההוראה ID

הוראה זו מזהה תצפיות על ידי שימוש בערכים של המשתנה המוגדר על ידי ההוראה ולא על ידי מספר התצפית (העמודה obs). אם משתנה הנמצא בהוראה ID מופיע גם בהוראה VAR, קובץ הפלט יכלול שתי עמודות של אותו המשתנה. אם המשתנה או משתנים המוגדרים על ידי ההוראה ID מופיעים בהוראה BY ובאותו סדר, SAS יוצרת מערך מיוחד של פלט. לדוגמא, איור 12 א' מציג את הפלט של PROC PRINT כאשר ההוראה ID כוללת את המשתנים של ההוראה BY באותו הסדר, ואילו איור 12 ב' מציג פלט ללא ההוראה ID.

אופן הכתיבה:

ID שמות המשתנים;

דוגמא:

```
proc print;
 by sex age;
 var height;
 id sex age;
run;
```

ההוראה PAGEBY

הוראה זו מגדירה ל-SAS מתי להוסיף לפלט מעבר עמוד (מתי לעבור לעמוד הבא גם אם העמוד הקודם לא התמלא בפלט). הוראה זו לא יכולה להופיע ללא ההוראה BY, והיא חייבת לכלול את המשתנה (או משתנים) שהוגדר על ידי ההוראה BY. למעשה, הוראה זו אומרת ל-SAS להתחיל עמוד פלט חדש בכל פעם שהערך של המשתנה המוגדר על ידי ההוראה BY משתנה.

אופן הכתיבה:

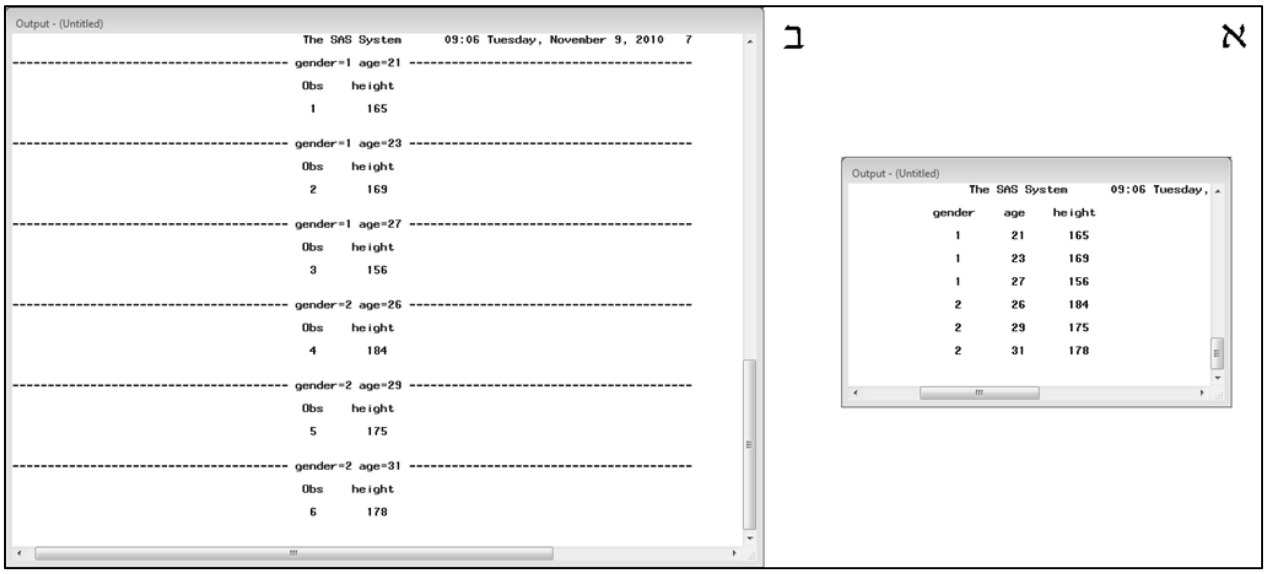
PAGEBY by המשתנה המוגדר על ידי ההוראה

דוגמא:

```
proc print;
 by sex;
 var height age;
```

```
 pageby sex;
run;
```

בדוגמא זו, SAS ייצור דף פלט חדש בכל פעם שהמשתנה sex ישנה את ערכו.



איור 12
(א) פלט של PROC PRINT עם ההוראה BY, הכולל את ההוראה ID.
(ב) פלט של PROC PRINT עם ההוראה BY, אך ללא ההוראה ID

ההוראה SUM

הוראה זו אומרת ל-SAS לעשות סיכום של הערכים הנומריים של המשתנים המוגדרים על ידי ההוראה. כאשר משתמשים בהוראה זו במקביל עם ההוראה BY, הפרוצדורה תחשב סכום לכל קבוצת ערכים אשר כוללת יותר מתצפית אחת, וגם תחזיר את הסכום הכללי מעבר לכל הקבוצות.

אופן הכתיבה:

SUM שמות משתנים;

דוגמא:

```
proc print double data=dogma n obs = 'subject';
 var age height;
 by sex;
 sum age;
run;
```

דוגמא זו תפיק את הפלט המוצג באיור 13.

ההוראה LABELS

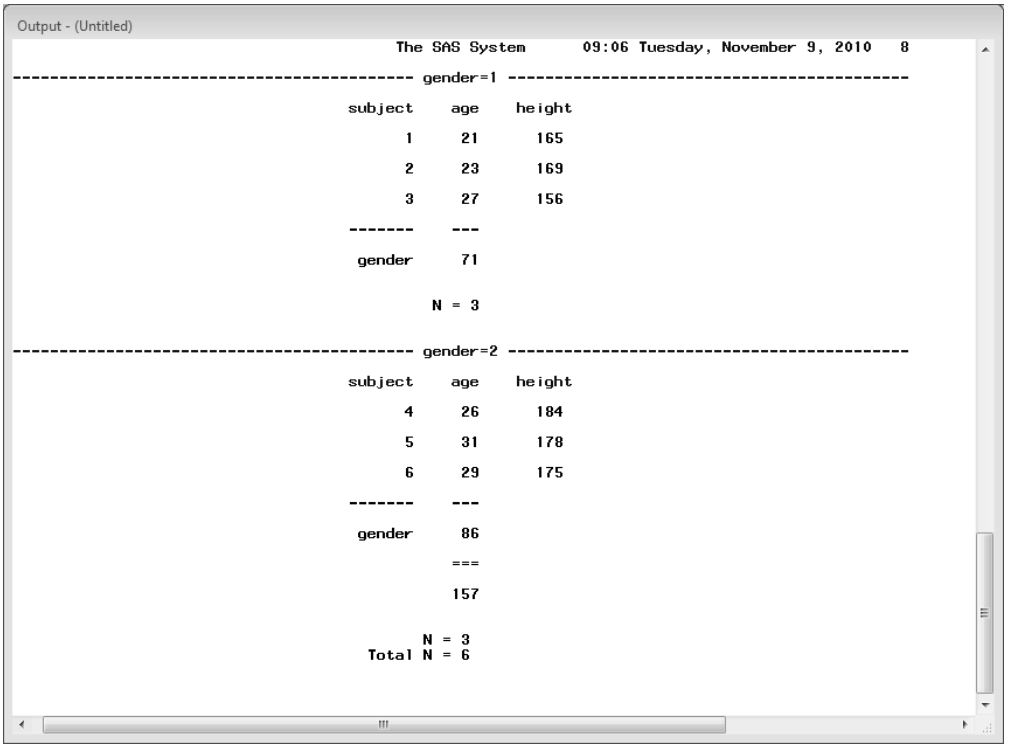
הוראה זו מגדירה את שמות העמודות (המשתנים) בקובץ הנתונים. כדי שהוראה זו תעבוד, האופציה split חייבת להיות מוגדרת.

אופן הכתיבה:

LABEL שם משתנה = 'התווית של המשתנה";

דוגמא:

```
proc print data=dogma split = '*' obs='Sub number';
 var age height sex;
 label age='Gil'
      height='Gova'
      sex='Min';
run;
```



איור 13 – פלט של PROC PRINT הכולל את ההוראות BY ו-SUM, ואת האופציות double, obs ו - n

ההוראה TITLE

הוראה זו מגדירה כותרת כללית לקובץ הפלט.

אופן הכתיבה:

TITLE 'מחרוזת כלשהי';

דוגמא:

```
proc print data=dogma noobs;
 title 'dogma of title';
run;
```

תרגול עצמי – מיון והפקת פלט

תרגיל 15

מיין את קובץ הנתונים הבא על פי מין בסדר עולה, ועל פי ציונים בסדר יורד:

Sub age gender grade

1 23 0 75
2 31 0 88
3 29 1 90
4 25 0 92
5 27 1 95
6 21 0 80
7 30 1 79

תרגיל 16

נתון הפלט הבא:

write a SAS code to replicate this output

observation Number	age	height	sex
1	21	168	2
2	26	173	1
3	24	181	1
4	27	158	2
5	30	185	1
6	24	173	1

כתוב קוד SAS שיפיק קובץ פלט זהה (כולל הכותרת של הפלט).

תרגיל 17

נתון קובץ הנתונים הבא:

Sub	age	gender	choice
1	31	1	0.85
2	24	0	0.5
3	26	1	0.6
4	23	1	0.4
5	27	0	0.62
6	26	1	0.43

כתוב תוכנית SAS שתדפיס את מספר הנבדק ואת המשתנה choice רק עבור שלושת התצפיות הראשונות של הערך "1" של המשתנה gender. על הפלט לא לכלול את העמודה obs.

פרק 7

פרוצדורות שירות II:
הגדרת משתנים וטיפול בתצפיות

PROC FORMAT

טיפ קריאה: PROC FORMAT היינה פרוצדורה מתקדמת, והיא מומלצת לקריאה למתכנתי SAS מתקדמים.

הפרוצדורה FORMAT מאפשרת למשתמש להגדיר את הפורמאט וה-informat של המשתנים, להגדיר תיאורים של המשתנים וכדומה. כל המידע הנוצר בפרוצדורה FORMAT מאוכסן בקובץ קטלוג (זמני או קבוע), שהוא קובץ נתונים מיוחד של SAS, המאפשר להכיל סוגים שונים של רשומות.

ה-informats קובעים כיצד ייקראו ויאוכסנו הערכים של נתונים גולמיים, בעוד שהפורמאטים קובעים כיצד ערכי המשתנים יופקו בפלט. לכן, ניתן לומר באופן כללי כי ה-informats אחראים על המרה (שינוי ערכי המשתנים) והפורמאטים אחראים על הדפסה (אופן ההצגה של משתנים). הן הפורמאטים והן ה-informats אומרים ל-SAS את סוג המשתנים (נומריים או אלפאנומריים), והן את אופן הטיפול בהם (למשל בערכים חסרים). עם זאת, לכל אחד מהם יש תפקוד ספציפי.

עם informats ניתן:

- להמיר מספרים למחרוזות (לדוגמא, ניתן להפוך 1 ל "זכר" ו-2 ל "נקבה")
- להמיר מחרוזות מסוימות למחרוזות אחרות (לדוגמא להפוך כל Yes ל-Correct)
- להמיר מחרוזות למספרים
- להמיר מספרים מסוימים למספרים אחרים (לדוגמא להפוך כל מספר בין 1 ל-9 ל-1, כל מספר בין 10 ל-19 ל-2 וכדומה)

עם פורמאטים ניתן:

- להדפיס ערכים מספריים כמחרוזות
- להדפיס מחרוזות מסוימות כמחרוזות אחרות
- להדפיס ערכים מספריים תוך שימוש בתבניות (לדוגמא, להדפיס את המשתנה 7855222 כ-222-5-785)

כמו כן, ניתן להשתמש בפרוצדורה כדי לקבץ ערכים, לייחס תווית לכל משתנה וכדומה. הפרוצדורה יוצרת את הפורמאט ראשית ללא קשר למשתנים עצמם, ורק אחרי זה מקשרת בינו לבין המשתנה (עם הוראת FORMAT מיוחדת שלא כלולה בפרוצדורה, כפי שיפורט בהמשך).

אופן הכתיבה:

```
PROC FORMAT<אופציות שונות>;
INVALUE שם <informats של אופציות >
  ערך 1 = טווח ערכים1
   .                .
```

```
  .              .
  .              .
 <nטווח ערכים = n ערך>;
VALUE ($) שם <אופציות של פורמאטים>
  1טווח ערכים = 1 ערך
  .              .
  .              .
 <nטווח ערכים = n ערך>;
PICTURE שם <formats אופציות של>
  1טווח ערכים = 1 תבנית תצוגה
  .              .
  .              .
 <nטווח ערכים = n תבנית תצוגה>;
RUN;
```

דוגמא:

```
proc format;
 value sex
  0 = 'Male'
  1 = 'Female'
  other = .;
 value gil
  0  - 10 = 'kid'
  10 - 20 = 'teen'
  20 - 50 = 'adult'
  50 - 100 = 'senior';
 value lect
    1='Eldad''s class'
    2='Ido''s class';
run;
```

הערה: ניתן לכתוב מספר בלתי מוגבל של פורמאטים ו-informats בפרוצדורת FORMAT אחת.

אופציות של PROC FORMAT

1. האופציה library – אופציה זו מגדירה קטלוג שבו יאוכסנו הפורמאטים וה-informats המוגדרים ב- PROC FORMAT הנוכחית. הפרוצדורה מאכסנת את הנתונים בקטלוג לשימוש חוזר.
אופן הכתיבה:

```
PROC FORMAT library = שם קטלוג.שם תיקייה;
```

דוגמא:

```
proc format library = sascodes.varfor;
```

הערה: ללא הגדרת האופציה library, SAS תיצור קטלוג בשם formats בתיקייה הזמנית work (או בתיקיית ברירת המחדל שהוגדרה על ידי המשתמש). קטלוג זה יהיה זמין כל עוד SAS עובדת, אך ימחק מהזיכרון בסיום העבודה עם התוכנה. אם מגדירים באופציה רק שם תיקייה, SAS יוצרת קטלוג בשם formats בתיקייה זו. קטלוג זה יהיה זמין גם בהפעלות עתידיות של התוכנה.

2. האופציה noreplace – כברירת מחדל, SAS משכתבת פורמאטים או informats בעלי אותו שם הקיימים בקטלוג. אופציה זו אומרת ל-SAS לא לכתוב מידע חדש על פורמאטים או informats קיימים.
אופן הכתיבה:

```
PROC FORMAT noreplace;
```

הערה: בניגוד לפרוצדורות אחרות שנלמדו או יילמדו בספר זה, האופציה data איננה רלוונטית ל-PROC FORMAT, שכן פורמאטים נוצרים באופן כללי, ורק לאחר מכן משוייכים לקובץ נתונים מסוים.

ההוראה INVALUE

ההוראה INVALUE נועדה להגדרת informats, דהיינו להמרת ערכים של תצפיות. ניתן להמיר משתנים נומריים לאלפאנומריים ולהפך, או להמיר משתנים נומריים לנומריים ואלפאנומריים לאלפאנומריים חדשים. אם ה-informat שנוצר מכיל משתנים אלפאנומריים, השם שלו צריך להתחיל בתחילית $.

אופן הכתיבה:

```
INVALUE ($) informat שם ה-   <אופציות שונות>
        ערך מומר1 = ערך או טווח1
        <ערך מומר2 = ערך או טווח2>
          .           .
          .           .
        <ערך מומרn = ערך או טווחn>
```

המרה יכולה להתבצע על ערך בודד של משתנה, או על טווח ערכים. לדוגמא:

- 1 – 25: הגדרה זו תגיד ל-SAS לבצע informat על כל התצפיות שערכן נע בין 1 ל-25.
- 'A'-'Z': הגדרה זו תגיד ל-SAS לבצע informat על כל התצפיות שמתחילות באותיות A עד Z.
במקרה זה יש לשים לב כי כל מחרוזת צריכה להיות מוכנסת למרכאות בנפרד.

בנוסף, ניתן להשתמש במילות מפתח כדי לבצע את ההמרה, הן מבחינת הגדרת טווח הערכים להמרה והן מבחינת הגדרת הערכים המומרים.

מילות מפתח להגדרת טווח:

- low - < 10: מגדיר טווח של כל ערך הקטן מ-10 (לא כולל 10)
- low – 10: מגדיר טווח של כל ערך הקטן מ-10 (כולל 10). מדוגמא זו עולה כי השימוש בסימן < נועד כדי להשמיט ערכים מהטווח.
- High - > 50: מגדיר טווח של כל ערך הגדול מ-50 (לא כולל 50)
- Other – מגדיר כל ערך שלא נכלל בטווח הערכים שהוגדרו.

הערה: כאשר שני טווחים שונים כוללים ערך זהה (או ערכים זהים), SAS תייחס את הערך המומר לערך שנמצא בטווח הראשון. כמו כן, אם לא מגדירים ערך או משתנה כלשהו, SAS תשאיר אותו כפי שהוא מופיע בקובץ הנתונים המקורי.

הגדרת ערכים מומרים:

- 'מחרוזת' : מגדיר את הערך של informat אלפאנומרי. המחרוזת חייבת להופיע במרכאות
- מספר : מגדיר את הערך של informat נומרי
- _ERROR_ : מגדיר שכל הערכים בטווח הם ערכים שגויים. במקרה זה SAS מייחסת לכל תצפית כזאת ערך חסר.
- _SAME_ : מגדיר שכל הערכים בטווח צריכים לשמור על הערך שלהם לפני ההמרה.

דוגמא:

```
invalue $gil
  low - < 10 = 'kid'
  10 - 20 = 'teen'
  20 - 50 = 'adult'
  50 < - high = 'senior';
```

ההוראה VALUE

ההוראה VALUE מאפשרת ליצור פורמאטים של תצוגה מוגדרים על ידי המשתמש. פורמאטים אלה קובעים כיצד הנתונים יוצגו בקובץ פלט. מבחינה מעשית, הגדרת פורמאטים דומה מאוד להגדרת informats, וברוב המקרים היא גם משתמשת באותן אופציות. עם זאת, בעוד ש-informats יוצרים נתונים חדשים, הפורמאטים מטפלים אך ורק בדרך התצוגה.

כאשר הפורמאטים מגדירים משתנים אלפאנומריים, יש להשתמש בתחילית $ (ללא רווח) כדי להגדיר את שם הפורמאט. לעומת זאת, כאשר מגדירים תצוגת מחרוזת למשתנה נומרי, אין צורך להשתמש בתחילית.

אופן הכתיבה:

VALUE ($) אופציות שונות שם הפורמאט

ערך מומר1 = ערך או טווח1

<ערך מומר2 = ערך או טווח2>

. .

. .

<ערך מומר*n* = ערך או טווח*n*>;

בדומה ל-informats, ניתן להגדיר פורמאטים על ערך בודד או על טווח ערכים. עם זאת, מילות המפתח לערכים מומרים _ERROR_ ו - _SAME_ ישימות ל-informats בלבד. שאר מילות המפתח ישימות הן לפורמאטים והן ל-informats.

בנוסף, כאשר רוצים להגדיר פורמאט מסוג מחרוזת, עם גרש (הסימן ') בתוך המחרוזת, יש לכתוב את זה כשני גרשיים נפרדים.

דוגמאות:

1. דוגמא ליצירת פורמט להצגת משתנה מין המקודד כ-1 או 0 כ-Male או Female

```
value sex (multilabel)
  0 = 'Male'
  1 = 'Female'
  other = .;
```

2. דוגמא ליצירת פורמאט להצגת משתנה של גיל בצורה מילולית בהתאם לטווח גילאים:

```
value gil
  low - < 10 = 'kid'
  10 - 20 = 'teen'
  20 - 50 = 'adult'
  50 < - high = 'senior';
```

3. דוגמא ליצירת פורמאט מחרוזת עם גרש במחרוזת:

```
value lect
    1='Eldad''s class'
    2='Ido''s class';
```

ההוראה PICTURE

ההוראה PICTURE משמשת כדי ליצור תבניות תצוגה לערכי משתנים בקבצי פלט. לכן, ניתן באמצעותה להציג את המשתנים בדיוק בצורה הרצויה (לדוגמא, ערכים כספיים ניתן להציג עם הסימן $). הגדרה של תבניות picture זהה לצד השמאלי של ההגדרה של פורמאטים ו-informats (דהיינו, ההגדרה של הערכים והטווחים), אך היא שונה בצד הימני שלה (ההגדרה של דרך התצוגה של הערכים או הטווחים).

ההגדרה של דרך התצוגה עשויה לכלול את הערכים הבאים:

א. הגדרת מספר הספרות של המשתנה שיופיעו לפני ואחרי הנקודה העשרונית. לדוגמא, ההגדרה *9999.9999* אומרת ל-SAS לקחת 4 ספרות לפני ו-4 ספרות אחרי הנקודה. כל ערך שיחרוג מהגדרה זו יקוצץ. לכן, תצפית שערכה 12345.00015 תוצג כ-2345.0001.

ב. הגדרת תו או מחרוזת שיופיע אחרי כל ערך מוגדר. לדוגמא, ההגדרה 999 years אומרת ל-SAS להוסיף את המחרוזת years אחרי ערך של כל תצפית.

אופן הכתיבה:

PICTURE ($) שם אופציות שונות;
תבנית תצוגה1 = ערך או טווח1
<תבנית תצוגה2 = ערך או טווח2>
. .
. .
<תבנית תצוגה*n* = ערך או טווח*n*>;

תבניות תצוגה בהוראה PICTURE מוגדרות לרוב על ידי בוררי ספרה (ספרות). בהקשר זה יש לציין 2 דברים:

א. שימוש בבוררי סיפרה שהם 0 (כאשר בדרך כלל נהוג להשתמש בספרה *9*) "מכריח" את הערכים של המשתנה לתפוס את כל התבנית של המספר (וכאשר המספר לא ארוך מספיק, SAS תוסיף לו במצב זה אפסים מובילים).

ב. שימוש בבורר סיפרה 0 יגרום למספר לתפוס בתבנית רק את כמות התווים שהוא כולל (ללא אפסים מובילים).

דוגמא:

```
proc format;
 picture dogma
  Low-high = '999.99';
run;
```

טבלה 3 שלהלן מדגימה כיצד ההוראה PICTURE שבדוגמא משפיעה על דרך התצוגה של ערכים קיימים. יש לשים לב כי בדוגמא הנוכחית, בורר הסיפרה היה 9, דבר המורה ל-SAS להוסיף אפסים מובילים. העמודה השמאלית בטבלה 3 כוללת דוגמא לתצוגה של ערכי משתנים שעברו picture זהה לדוגמא, מלבד העובדה שבורר הסיפרה הוא 0.

משתנה קיים	משתנה שעבר פרמוט (בורר ספרה = 9)	משתנה שעבר פרמוט (בורר ספרה = 0)
1234.567	234.56	234.56
123.456	123.45	123.45
123.450	123.45	123.45
123.400	123.40	123.40
123.000	123.00	123.00
12.345	012.34	12.34
12.340	012.34	12.34
12.300	012.30	12.30
12.000	012.00	12.00
1.234	001.23	1.23
1.230	001.23	1.23
1.200	001.20	1.20
1.000	001.00	1.00

טבלה 3 – הקשר בין ההוראה PICTURE ב-PROC FORMAT להגדרת בורר ספרה

אופציות של ההוראה PICTURE

1. האופציה fill – לכל תבנית תצוגה יש אורך מסוים. כאשר יש ערכים מסוימים במשתנה שאורכם קטן יותר מהאורך המקסימאלי, ניתן באמצעות האופציה fill לקבוע את התו שימלא את המקומות החסרים.
   הערה: הפרוצדורה תשתמש בתו המוגדר באופציה fill רק כאשר בורר הסיפרה מוגדר כ-0. במצב בו בורר הספרה מוגדר אחרת (לדוגמא 9), SAS משתמשת באפסים כדי למלא את התווים החסרים, כך שבמקרה זה לאופציה fill לא תהיה כל השפעה.
   אופן הכתיבה:

   (fill = 'תו')

   דוגמא:

```
proc format;
 picture test
    Low-high = '000.00' (fill='*');
run;
```

2. האופציה multiplier – אופציה זו מגדירה מספר שערכי המשתנה יוכפלו בו לפני שהם עוברים את ההמרה לאופן התצוגה המוגדר בהוראה.
   אופן הכתיבה:

(multiplier = מספר)

דוגמא:

```
proc format;
 picture million
  low-high='00.0M' (multiplier=.00001);
run;
```

בדוגמא זו אומרים ל-SAS לקחת משתנים מספריים בטווחי הערכים של מיליון ולהציג אותם כדלקמן: לקחת לדוגמא את הערך 3,500,000 ולהציג אותו כ-3.5M.

3. האופציה noedit – אופציה זו אומרת ל-SAS שספרות המופיעות בהגדרת דרך התצוגה (תבנית תצוגה) הן מחרוזות ולא בוררי סיפרה. לכן, אופציה זו אומרת ל-SAS להציג את הספרות האלה כפי שהן מופיעות בתבנית התצוגה ללא קשר לערך התצפית הרלוונטית.
אופן הכתיבה:

(noedit)

דוגמא:

```
proc format;
 picture threemil 1000000-3000000='00.0M' (multiplier=0.00001)
        3000000<-high='>3.0M'(noedit);
run;
```

דוגמא זו אומרת ל-SAS להציג כל ערך בין 1000000 ל-3000000 על פי התבנית x.xM, וכל ערך מעל 3000000 על פי התבנית 3.0M > (גדול מ-3 מליון).

4. האופציה prefix – אופציה זו אומרת ל-SAS להוסיף תחילית לפני הספרה הממשית הראשונה של כל תצפית. יש לציין כי אופציה זו לא תעבוד אם לא משתמשים בבורר סיפרה מסוג 0. כמו כן, כאשר משתמשים באופציה fill, SAS קודם מציגה את התחילית ורק לאחר מכן מתחילה להציב את תווי המילוי.
אופן הכתיבה:

(prefix = 'תו או מחרוזת')

דוגמא:

```
proc format;
 picture million
   low-high='00.0M' (prefix = '$' multiplier=.00001);
run;
```

בדוגמא זו SAS תוסיף לכל ערך את התחילית $. לדוגמא, הערך 3,500,000 יוצג כ - $3.5M.

הפקודות INPUT ו-PUT

כפי שצוין קודם, הפרוצדורה FORMAT יוצרת informats ו-formats ללא קשר למשתנה או אפילו לקובץ נתונים ספציפי. כדי לעשות את הקישור בין informat או format לבין משתנה בקובץ נתונים, יש להשתמש בפקודה INPUT או בפקודה PUT, תלוי בסוג המשתנה ובשאלה האם מדובר על format או על informat, כפי שיפורט בהמשך. עם זאת, יש לציין כי הפקודות input ו-put נכתבות ב-DATA STEP או ב-PROC STEP, אבל לא בתוך ה-PROC FORMAT עצמו.

כדי לעשות קישור (או בעצם להמיר) מחרוזות לערכים מספריים, יש להשתמש בפקודה INPUT. כדי לעשות קישור בין ערכים מספריים למחרוזות, יש להשתמש בפקודה PUT.

אופן הכתיבה:

שם משתנה חדש = input (שם משתנה ישן, informat - שם ה.);

שם משתנה חדש = put (שם משתנה ישן, format - שם ה.);

עם זאת, יש לשים לב כי הפקודה INPUT עובדת עם informats, בעוד שהפקודה PUT עובדת עם formats.

דוגמא:

```
data dogma2; set dogma;
 gil = put(age, gil.);
run;
```

בדוגמא זו, נוצר קובץ נתונים חדש (dogma2). קובץ זה מכיל את כל המשתנים שהיו כלולים בקובץ הנתונים dogma, כולל משתנה חדש gil, הכולל את הערכים המומרים של המשתנה age כפי שהוגדרו ב-informat בשם $gil (שכן מדובר כאן על informat מערכים נומריים של גיל לערכים אלפאנומריים).

הערה: באופן כללי, עדיף להשתמש בפורמאטים ולא ב-informat כדי להפוך נתונים נומריים לאלפאנומריים. כאשר משתמשים ב - informat למטרה זו, תתקבל בחלון Log ההערה הבאה:

```
NOTE: 10 is a numeric field and a character format is defined.
```

עם זאת, חרף ההודעה המתקבלת בחלון Log, הפעולה תתבצע.

הפקודה FORMAT

כדי לעשות קישור בין משתנה לפורמאט, או בין משתנה ל-PICTURE מסוים (דהיינו לשנות את דרך התצוגה של המשתנה) משתמשים בהוראה FORMAT. ניתן, תחת אותה הוראת FORMAT, לקשר בין כמה משתנים לפורמאטים (או PICTURES), או בין משתנה אחד לפורמאט. ההוראה FORMAT יכולה להופיע ב-DATA STEP או ב- PROC STEP, בכל אחת מהפרוצדורות של SAS.

אופן הכתיבה:

FORMAT משתנה\רשימת-משתנים שם הפורמאט.
;שם הפורמאט משתנה\רשימת-משתנים
FORMAT משתנה\רשימת-משתנים picture - שם ה.;

דוגמא לקישור בין משתנה לפורמאט:

```
proc print data = dogma noobs;
 format gender sex.;
run;
```

דוגמא לקישור בין משתנה ל-picture:

```
proc print data = dogma1;
 format var1 million.;
run;
```

PROC TRANSPOSE

הפרוצדורה TRANSPOSE אומרת ל - SAS לבצע פעולת החלפה בין השורות והעמודות של קובץ הנתונים, דהיינו להפוך את השורות (תצפיות) לעמודות (משתנים) ואת העמודות לשורות (לבצע שחלוף). אופציה זו שימושית מאוד כאשר רוצים, לדוגמא, לשנות את דרך הצגת הנתונים או את רמת הניתוח, והיא חוסכת בזמן הקלדה יקר.

אופן הכתיבה:

PROC TRANSPOSE<אופציות שונות> ;
BY <descending> משתנה 1 … <descending> משתנה n <notsorted>;
VAR שמות משתנים;
ID שם משתנה;
RUN;

דוגמא:

```
proc transpose data = dogma out = dogma2;
run;
```

נניח שיש לנו קובץ נתונים בשם dogma, המכיל את התצפיות הבאות:

age weight
23 75
32 89
23 80
27 65
26 90
26 75
22 49
30 82
27 69

לאחר ביצוע שחלוף באמצעות PROC TRANSPOSE, נקבל את הקובץ dogma2 בעל הצורה הבאה:

age 23 32 23 27 26 26 22 30 27
weight 75 89 80 65 90 75 49 82 69

ההוראה BY

ההוראה BY ב-PROC TRANSPOSE לוקחת את קובץ הנתונים ויוצרת תצפית אחת לכל ערך ספציפי של המשתנה המוגדר על ידי ההוראה ומשתנה המכיל את הערך של המשתנים המוגדרים בפרוצדורת השחלוף לכל רשומה של המשתנה המוגדר על ידי ההוראה. כדי שהוראה זו תעבוד, יש למיין ראשית את קובץ הנתונים לפי המשתנה המוגדר על ידי ההוראה.

אופן הכתיבה:

BY <descending> משתנה 1 … <descending> משתנה n <notsorted>;

דוגמא:

```
proc transpose data = dogma out = trans;
 by age;
run;
```

בדוגמא זו PROC TRANSPOSE לוקחת את קובץ הנתונים הזה:

age weight height
23 75 170
32 89 185
23 80 172
27 65 170
26 90 186
26 75 167
22 49 157
30 82 180
27 69 175

ויוצרת שחלוף על פי המשתנה "גיל" בקובץ נתונים חדש, trans (בתנאי כמובן שקובץ הנתונים המקורי יהיה ממויין על פי המשתנה המוגדר על ידי ההוראה BY, דהיינו המשתנה age):

age	_NAME_	COL1	COL2
22	weight	49	.
22	height	157	.
23	weight	75	80
23	height	170	172
26	weight	90	75
26	height	186	167
27	weight	65	69
27	height	170	175
30	weight	82	.
30	height	180	.
32	weight	89	.
32	height	185	.

1. האופציה descending – אופציה זו מגדירה ל-SAS לכתוב בקובץ השחלוף את המשתנה לפיו יש לבצע את השחלוף מהגדול לקטן (ברירת המחדל היא מהקטן לגדול).
אופן הכתיבה:

שם של משתנה BY descending

דוגמא:

```
proc transpose data = dogma out = trans;
 by descending age;
run;
```

הערה: ניתן להשתמש באופציה descending רק כאשר קובץ הנתונים ממויין על פי המשתנה המוגדר על ידי ההוראה BY (באמצעות PROC SORT המופיעה בפרק 7) בסדר יורד (מהערך הגדול לערך הקטן).

2. האופציה notsorted – אופציה זו אומרת ל-SAS כי המשתנה על פיו יש לבצע את השחלוף לא ממוין בסדר עולה או יורד. אופציה זו חשובה כאשר יש חשיבות לסידור המשתנים בסדר מסוים לא כרונולוגי או נומרי (למשל כאשר עוסקים במדידות חוזרות בזמנים שונים).
אופן הכתיבה:

שם של משתנה BY notsorted

דוגמא:

```
proc transpose data = dogma out = trans;
 by notsorted age;
run;
```

ההוראה VAR

הוראה זו מגדירה ל-SAS על אילו משתנים בקובץ הנתונים יש לבצע את השחלוף.

אופן הכתיבה:

PROC TRANSPOSE<אופציות שונות>;
VAR שם של משתנה/משתנים;
RUN;

דוגמא:

```
proc transpose data = dogma out = trans;
 var height;
run;
```

בדוגמא זו הפרוצדורה עושה שחלוף רק למשתנה הגובה.

ההוראה ID

הוראה זו מגדירה ל-SAS מעבר לאיזה משתנה יש לעשות את השחלוף על המשתנה המוגדר על ידי ההוראה BY. כאשר הוראה זו לא נכתבת, SAS בוחרת משתנה כמשתנה ID באופן אוטומטי. לכן, כאשר יש מספר רב של משתנים, או כאשר אין התאמה מלאה בין מספר קטגוריות הערכים של המשתנה המוגדר על ידי ההוראה BY למשתנים אחרים, אי כתיבת ההוראה ID בקוד עשויה לגרום לקידוד לא נכון של הנתונים.

אופן הכתיבה:

PROC TRANSPOSE;
ID שם של משתנה;

דוגמא:

נתון קובץ הנתונים הבא, הכולל 5 מדידות חוזרות של קצב לב (hr – heart rate) בקרב 5 נבדקים שונים:

trial sub hr
1 1 75
2 1 80
3 1 79
4 1 69
5 1 75
1 2 69
2 2 78
3 2 75
4 2 73
5 2 70
1 3 101
2 3 93
3 3 100
4 3 100
5 3 95
1 4 67
2 4 52
3 4 50
4 4 61
5 4 60
1 5 79
2 5 83
3 5 71
4 5 85
5 5 80

הקוד הבא משחלף את המשתנה "קצב לב" לפי המשתנה "נבדק" מעבר לסיבובים:

```
proc transpose data = dogma out = dogma2;
 by sub;
 var hr;
```

```
 id trial;
run;
```

לאחר הרצת הקוד, קובץ השחלוף שנוצר יקבל את הצורה:

sub	_NAME_	_1	_2	_3	_4	_5
1	hr	75	80	79	69	75
2	hr	69	78	75	73	70
3	hr	101	93	100	100	95
4	hr	67	52	50	61	60
5	hr	79	83	71	85	80

אופציות של PROC TRANSPOSE

1. האופציה name – אופציה זו מגדירה את השם של עמודת המשתנים בקובץ הנתונים שנוצר בפרוצדורת השחלוף (העמודה המכילה את שמות המשתנים. כברירת מחדל, SAS קוראת לעמודה זו _NAME_.
אופן הכתיבה:

PROC TRANSPOSE name = שם העמודה;

דוגמא:

```
proc transpose data = dogma out = trans name = var;
run;
```

2. האופציה prefix – אופציה זו מגדירה תחילית לשימוש בבניית שמות למשתנים שעברו שחלוף בקובץ הנתונים החדש שנוצר. SAS משתמשת בתחילית זו ומוסיפה לה מספר בהתאם לסדר הופעת המשתנים בקובץ. לדוגמא, אם התחילית שנקבעה היא var, SAS תתחיל למספר את העמודות של המשתנים שעברו שחלוף, var*1*, var*2*,…,var*n*.
אופן הכתיבה:

PROC TRANSPOSE prefix = שם התחילית הרצויה;

דוגמא:

```
proc transpose data = dogma out = trans prefix = sub;
run;
```

דוגמא זו מניחה כי פונקציית השחלוף בוצעה מעבר לנבדקים. לכן, כל עמודה (המציינת נתונים של נבדק שונה) תיקרא sub*n*, כאשר n מייצגת את מספר השורה.

הערה: כאשר משתמשים בהוראה ID ולא מגדירים את האופציה prefix, העמודות יקבלו את הערכים של משתנה ה-ID. מאחר וב-SAS שמות משתנים לא יכולים להתחיל במספר, SAS תתחיל את שם העמודה בקו תחתון (_) ואחריו תשים את הערך של המשתנה. לעומת זאת, כאשר כן מגדירים את האופציה prefix שמות העמודות נקבעות לפי הגדרה זו.

3. האופציה let – כאשר ערך מסוים של המשתנה שמעבר אליו עושים את השחלוף חוזר כמה פעמים תחת אותו ערך של המשתנה על פיו עושים את השחלוף (לדוגמא, כאשר רוצים לעשות שחלוף על פי המשתנה "סוג טיפול" מעבר למשתנה "מין", ויש כמה תצפיות מאותו מין את אותו סוג הטיפול), SAS מודיעה על שגיאה. האופציה let אומרת ל-SAS להתעלם מהשגיאה ולבצע את השחלוף. השחלוף מתבצע רק על התצפית המכילה את ההתרחשות האחרונה בערך של המשתנה ID בתוך קובץ הנתונים.
אופן הכתיבה:

PROC TRANSPOSE let;

דוגמא:

```
proc transpose let data = dogma out = dogma2;
 by trt;
 var hr;
 id sex;
run;
```

בדוגמא זו קיים קובץ הנתונים dogma, המורכב מהתצפיות הבאות:

treat gender hr
1 1 75
1 2 80
1 2 79
2 2 69
2 1 78
2 1 70

אם לא היינו משתמשים באופציה let, SAS הייתה מודיעה על שגיאה בחלון Log:

WARNING: The ID value "_2" occurs twice in the same BY group.

ולא הייתה משלימה את הפעולה של הפרוצדורה. אולם, מאחר שהשתמשנו באופציה, קובץ הנתונים שנוצר כתוצאה מפרוצדורת השחלוף יהיה בעל הצורה הבאה:

treat _NAME_ _1 _2
1 hr 75 79
2 hr 70 69

תרגול עצמי – הגדרת משתנים וטיפול בתצפיות

תרגיל 18

להלן קובץ נתונים הכולל תצפיות של המשתנים מספר נבדק, גיל, מין, כמות נקודות נצברת ואחוז בחירות:

sub	age	gender	points	prop
1	24	1	125	0.45
2	31	0	100	0.56
3	27	0	150	0.53
4	22	1	130	0.60
5	26	1	115	0.43

6	29	0	110	0.50
7	30	1	105	1.20

כתוב פורמאט כדי לקודד את המשתנה gender כמחרוזת (כאשר 0 = male 1 = female). בנוסף, המשתנה prop צריך לקבל את הערך 'risk seeker' כאשר ערך המשתנה גדול מ-0.5, 'risk aversive' כאשר הערך קטן מ-0.5, ו-'indifferent' כאשר הערך שווה ל-0.5.

תרגיל 19

נתון קובץ הנתונים הבא:

Treat	sex	HR
1	1	63
1	2	75
2	1	101
2	2	93
3	2	49
3	1	52
4	2	79
4	1	80

כתוב קוד שיבצע שחלוף לקובץ, בו המשתנה HR יוצג בשורות. את השחלוף יש לבצע מעבר לסוג הטיפול (treat) ומעבר למין (sex). תן לעמודות בקובץ השחלוף שנוצר שם שישקף את שני המשתנים שמעבר אליהם עושים את השחלוף (לדוגמא: זכר_טיפול 1).

תרגיל 20

הערה: שאלה זו מבוססת על הנתונים בשאלה 18

א. 1. כתוב פורמאט picture שיציג את המשתנה prop כאחוז בחירות ולא כמספר עשרוני (לדוגמא, במקום 0.45 יופיע 45) ויוסיף את הסימן % לאחר כל ערך. במקרה בו ערך המשתנה קטן מ-0 או גדול מ-100, התצפית תקבל את המחרוזת 'Out of bounds'.

2. הגדר ב-picture של המשתנה prop שבנוסף למחרוזת 'Out of bounds' יוצג גם הערך המקורי של המשתנה.

ב. כתוב picture להמרת הנקודות במשתנה points לכסף, על פי שער ההמרה של 10 נקודות = 1 ₪, והוסף את התחילית NIS לכל ערך במשתנה זה.

ג. הדפס את קובץ הנתונים הכולל את כל ה-pictures שיצרת בסעיף א' 1 ובסעיף ב'.

פרק 8
פרוצדורות שירות III: טיפול בקבצי נתונים

PROC IMPORT

הפרוצדורה IMPORT אומרת ל-SAS לקרוא נתונים מקובץ חיצוני ולכתוב אותם לקובץ נתונים של התוכנה. ניתן לייבא ל-SAS קבצי אקסל, access, dBASE, Lotus, CSV, ו-txt.

טיפ קריאה: הפרוצדורה IMPORT קוראת נתונים מקובץ חיצוני, בדומה להוראה INFILE שנלמדה ב-DATA STEP. מאחר ופרוצדורה זו מורכבת יותר, למתחילים מומלץ להשתמש בתחילה בפקודה INFILE כדי לייבא נתונים מקובץ חיצוני אל תוך SAS.

אופן הכתיבה:

PROC IMPORT datafile = שם קובץ out = שם קובץ נתונים <אופציות שונות>;
<הוראות שונות>;
RUN;

דוגמא:

```
proc import DATAFILE= "C:\dogma.xls" OUT= dogma SHEET="sheet1";
run;
```

אופציות של PROC IMPORT

ל-PROC IMPORT יש אופציות חובה, שאי הגדרתם תגרום לטעות בהרצת הקוד, ואופציות רשות.

<u>אופציות חובה</u>:

1. האופציה datafile – אופציה זו מגדירה את הנתיב המלא ואת השם של הקובץ החיצוני.
אופן הכתיבה:

Datafile = "נתיב הקובץ ושמו, כולל סיומת"

דוגמא:

```
DATAFILE= "C:\dogma.xls"
```

הערה: אם השם לא כולל סימנים מיוחדים (כגון קו אלכסוני), אותיות קטנות או רווחים, ניתן להשמיט את המרכאות הכפולות.

2. האופציה table – אופציה זו לא רלוונטית לכל סוגי הקבצים, והיא מגדירה ל-SAS את שם הטבלה של הקובץ החיצוני.
אופן הכתיבה:

table = "שם טבלה"

דוגמא:

```
table= "table1"
```

הערה: אם השם לא כולל סימנים מיוחדים (כגון קו אלכסוני), אותיות קטנות או רווחים, ניתן להשמיט את המרכאות הכפולות. כמו כן, יש לשים לב כי שם טבלה הוא תלוי רישיות.

3. האופציה out – אופציה זו מגדירה ל-SAS תחת איזה שם לשמור את קובץ הנתונים המכיל את הנתונים המיובאים מהקובץ החיצוני.
אופן הכתיבה:

out = שם של קובץ נתונים

דוגמא:

```
OUT= dogma
```

<u>אופציות רשות</u>:

1. האופציה dbms – אופציה זו מגדירה את סוג הקובץ אותו מייבאים ל-SAS. כאשר הסיומת של שם הקובץ כפי שהיא מוגדרת באופציה datafile הינה סיומת מוכרת כך שהפרוצדורה יכולה לזהות אותה (לדוגמא xls, txt, csv), אין צורך להגדיר אופציה זו. עם זאת, אם רוצים לייבא טבלה, חייבים להגדיר את האופציה תוך שימוש בשם של יישום קבצי נתונים תקף. טבלה 4 מסכמת את סוג היישומים ואת סיומת הקובץ שלהם.
אופן הכתיבה:

dbms = שם מזהה של היישום

דוגמא:

```
table = "table1" dbms = access;
```

דוגמא זו מגדירה ל-SAS לייבא את הנתונים מטבלה1 בקובץ של Access.

מזהה	יישום	סיומת
ACCESS	Microsoft Access database	.mdb
DBF	dBASE file	.dbf
WK1	Lotus 1 spreadsheet	.wk1
WK3	Lotus 3 spreadsheet	.wk3
WK4	Lotus 4 spreadsheet	.wk4
EXCEL	Microsoft Excel spreadsheet	.xls
DLM	Delimited file	.*
CSV	Delimited file (comma-separated)	.csv
TAB	Delimited file (tab-separated)	.txt

טבלה 4 – סוגי קבצים הניתנים לייבוא על ידי PROC IMPORT

הערה: סוג הקבצים אותם ניתן לייבא בפועל תלוי ברישיון SAS/ACCESS שיש לתוכנת SAS המותקנת במחשב. אם אין רישיון ל- SAS/ACCESS כלל, ניתן לייבא קבצים של DLM, CSV ו-txt בלבד.

2. האופציה replace – אופציה זו אומרת ל-SAS לכתוב על קובץ נתונים קיים. במקרה שמגדירים קובץ נתונים קיים (באמצעות האופציה out) ללא האופציה replace, הפרוצדורה import לא תכתוב על קובץ הנתונים הקיים.
אופן הכתיבה:

replace

דוגמא:

```
proc import datafile = "c:\dogma.txt" out = dogma replace;
```

הוראות לקבצי נתונים חיצוניים

ל-PROC IMPORT יש מספר הוראות ייחודיות שמגדירות ל-SAS כיצד לטפל בקובץ הנתונים המיובא. הוראות מסוימות רלוונטיות לסוגי קבצים מסוימים, כפי שמפורט בטבלה 5.

ההוראה GETNAMES – הוראה זו קובעת האם לייצר שמות משתנים מתוך שמות העמודות בקובץ החיצוני (השורה הראשונה של כל עמודה). אופציה זו מוגדרת באמצעות הפקודה Yes או No. אם תבחר ב-No, או אם שמות המשתנים אינם שמות SAS חוקיים, SAS תיתן למשתנים את השמות var*0*, var*1*,…,var*n*. כברירת מחדל, הערך של הוראה זו הוא Yes.

אופן הכתיבה:

GETNAMES = yes | no

דוגמא:

```
proc import datafile= "C:\dogma.xls" out= dogma;
 getnames=no;
run;
```

סוג הקובץ	GETNAMES	DATAROW	DELIMITER	RANGE	SHEET	MIXED
WK1	V			V	V	V
WK3	V			V	V	V
WK4	V			V	V	V
EXCEL	V			V	V	V
DLM	V	V	V			
CSV	V	V				
TAB	V	V				

טבלה 5- הוראות של PROC IMPORT

ההוראה DATAROW – הוראה זו אומרת ל-SAS להתחיל לקרוא את הקובץ החיצוני מהשורה ה- *n* ית. כברירת מחדל, SAS תתחיל לקרוא את הקובץ מהשורה הראשונה (דהיינו *n* = 1) כאשר GETNAMES = no, ומהשורה השנייה (*n* = 2) כאשר GETNAMES = yes. לכן, כאשר GETNAMES = yes, n חייב להיות גדול או שווה ל-2, וכאשר GETNAMES = no, גדול או שווה ל-1.

אופן הכתיבה:

DATAROW = מספר שורה

דוגמא:

```
proc import datafile= "C:\dogma.txt" out= dogma;
 datarow=5;
run;
```

ההוראה DELIMITER – הוראה זו מגדירה ל-SAS איזה תו משמש כתו מפריד בין התצפיות של המשתנים השונים (מה מפריד בין העמודות). אם לא מגדירים הוראה זו, SAS מניחה כברירת מחדל כי התו המפריד הוא רווח.

אופן הכתיבה:

DELIMETER = התו המפריד

דוגמא:

```
proc import datafile= "C:\dogma.txt" out= dogma;
 delimiter=',';
run;
```

ההוראה RANGE – הוראה זו אומרת ל-SAS לייבא רק חלק מהנתונים הנמצאים בגיליון האלקטרוני שמייבאים, באמצעות הגדרת התחום הרלוונטי בו נמצאים הנתונים הרצויים. הטווח המוגדר הוא המלבן בתוך הגיליון שבתוכו נמצאים הנתונים. את המלבן מגדירים על ידי התא הנמצא בפינה העליונה השמאלית שלו והפינה התחתונה הימנית, כאשר שני התאים מופרדים על ידי נקודותיים (:). SAS מניחה כי השורה הראשונה של הטווח מכילה שמות של משתנים. אם לא מגדירים טווח, SAS מייבאת את כל הנתונים הנמצאים בגיליון האלקטרוני.

אופן הכתיבה:

RANGE = תא שמאלי תחתון:תא ימיני עליון

דוגמא:

```
proc import datafile= "C:\dogma.xls" out= dogma;
 range=C9:F12;
run;
```

ההוראה SHEET – הוראה זו מזהה עבור SAS את הגיליון הספציפי מתוך הקובץ החיצוני ממנו יש לייבא את הנתונים. יש להשתמש בהוראה זו רק כאשר היישום ממנו מייבאים תומך בגיליונות מרובים (כגון אקסל) בתוך קובץ אחד. אם הוראה זו לא מוגדרת, SAS מייבאת כברירת מחדל את הנתונים מהגיליון הראשון.

אופן הכתיבה:

SHEET=”שם הגיליון”

דוגמא:

```
proc import datafile= "C:\dogma.xls" out= dogma;
 sheet=”sheet3”;
run;
```

ההוראה MIXED – באופן כללי SAS משתמשת בשמונה השורות הראשונות של הנתונים בכדי לקבוע האם המשתנים הם נומריים או אלפאנומריים. ברירת המחדל, המגדירה את ההוראה הזו כ-No, מניחה כי המשתנה כולל רק ערכים נומריים או רק ערכים אלפאנומריים. לעומת זאת, אם הנתונים כוללים משתנה (או משתנים) שיש לו גם ערכים נומריים וגם ערכים אלפאנומריים, או אם יש משתנים עם ערכים חסרים, צריך להגדיר הוראה זאת כ-Yes, כדי להבטיח ש-SAS תקרא את הקובץ החיצוני כראוי.

אופן הכתיבה:

MIXED = Yes/No

דוגמא:

```
proc import datafile= "C:\dogma.xls" out= dogma;
 mixed=yes;
run;
```

PROC DATASETS

הפרוצדורה DATASETS היא פרוצדורת שירות שמאפשרת לנהל ולתפעל קבצי SAS. בעזרת PROC DATASETS ניתן, בין היתר:

1. להעתיק קבצי SAS מספריית SAS אחת לספריית SAS אחרת
2. לשנות שמות של קבצי SAS
3. למחוק קבצי SAS
4. להפיק רשימת קבצי SAS הנמצאים בספרייה נתונה
5. לאחד בין קבצי נתונים שונים

אופן הכתיבה:

PROC DATASETS<אופציות שונות>;
APPEND base = <שם ספרייה> שם קובץ נתונים <אופציות שונות>;
CHANGE שם ישן 1 = שם חדש 1…<*n* שם ישן = *n* שם חדש></אופציות שונות>;
CONTENTS <אופציות שונות>;
COPY out = שם ספרייה 1 <אופציות שונות>;
EXCLUDE שמות קבצים</memtype = mtype>;
SELECT שמות קבצים</אופציות שונות>;
DELETE שמות קבצים<אופציות שונות>;
EXCHANGE שם 1 = שם אחר 1…<שם n = שם אחר n></אופציות>;
MODIFY שמות קבצים<אופציות של הקבצים>;
FORMAT משתנה 1 <פורמאט 1.>
. .
. .
<*n* משתנה <*n* פורמאט.>>;
INFORMAT משתנה 1 <informat.>
. .
. .

<*n* משתנה < informat*n*.>>;
LABELS 1 משתנה = <'1 תווית'>
. .
. .
<*n* משתנה <'n תווית'>>;
RENAME 1 שם ישן = 1 שם חדש...<*n* שם ישן = *n* שם חדש>;
REPAIR SAS שם קבצי<memtype = סוג קובץ>;
SAVE SAS קבצי</memtype = סוג הקובץ>;
RUN;

דוגמא:

```
proc datasets;
run;
```

כברירת מחדל, PROC DATASETS לא מפיקה פלט לחלון Output, אלא מציגה נתונים בחלון Log. הנתונים הבסיסיים כוללים את שם הספרייה, מנוע העבודה של SAS, נתיב הספרייה, ועבור כל קובץ נתונים שם, סוג הקובץ, גודלו, והתאריך האחרון בו הוא שונה (או נוצר):

```
                                Directory

Libref         WORK
Engine         V9
Physical Name  C:\DOCUME~1\GUYHOC~1\LOCALS~1\Temp\SAS Temporary Files\_TD2936
File Name      C:\DOCUME~1\GUYHOC~1\LOCALS~1\Temp\SAS Temporary Files\_TD2936


                  Member     File
#  Name           Type       Size  Last Modified

1  FORMATS  CATALOG   17408  12:05:46 09 2008 דצמבר
2  HUMP     DATA       5120  12:05:46 09 2008 דצמבר
3  HUMP2    DATA       5120  12:05:46 09 2008 דצמבר
4  HUMP3    DATA       5120  12:05:46 09 2008 דצמבר
5  HUMP4    DATA       5120  12:05:46 09 2008 דצמבר
```

הערה: המונח member המוצג בפלט של PROC DATASETS מקביל למונח קובץ SAS.

אופציות של PROC DATASETS

טיפ קריאה: אופציה זו היא למשתמשים מתקדמים.

1. האופציה library – אופציה זו מגדירה את הספרייה ממנה הפרוצדורה לוקחת את קבצי ה-input. כברירת מחדל, ספריית העבודה היא הספרייה Work (או ספרייה אחרת שהוגדרה כספריית ברירת המחדל על ידי המשתמש).
אופן הכתיבה:

library = נתיב הספרייה

דוגמא:

2. האופציה detail – אופציה זו אומרת ל-PROC DATASETS להוסיף לפלט הבסיסי את מספר התצפיות, מספר המשתנים ותוויות לכל קובץ בספרייה.
אופן הכתיבה:

details

לדוגמא, הפלט המתקבל בחלון Log לאחר הגדרת אופציה זו יהיה (בהשוואה לפלט הבסיסי שהוצג בדף הקודם):

#	Name	Member Type	Obs, Entries or Indexes	Vars	Label	File Size	Last Modified
1	FORMATS	CATALOG	1			17408	12:05:46 09 2008 דצמבר
2	HUMP	DATA	40	4		5120	12:05:46 09 2008 דצמבר
3	HUMP2	DATA	80	4		5120	12:05:46 09 2008 דצמבר
4	HUMP3	DATA	80	4		5120	12:05:46 09 2008 דצמבר
5	HUMP4	DATA	80	4		5120	12:05:46 09 2008 דצמבר

הערה: תוויות (labels) יוצגו רק לקבצים מסוג DATA (במקרה ותוויות מוגדרות לקובץ).

3. האופציה kill – אופציה זו מוחקת את כל קבצי ה-SAS הנמצאים בספרייה המוגדרת הזמינים בזיכרון התוכנה לעיבוד.
אופן הכתיבה:

kill

4. האופציה memtype – אופציה זו מגדירה את סוג הקבצים שיוצגו על ידי PROC DATASETS בחלון Log.
כאשר אופציה זו מוגדרת ביחד עם האופציה kill, היא קובעת אילו סוג של קבצים יימחקו על ידי הפרוצדורה.
אופן הכתיבה:

memtypes = סוג הקובץ/קבצים מופרדים על ידי רווחים

ישנם מספר סוגי קבצים ב-SAS. להלן סוגי קבצים נפוצים ביותר:
data – קובץ נתונים
catalog – קובץ קטלוג
program – קובץ קוד (תוכנית SAS)
fdb – קובץ נתונים פיננסי

5. האופציה nolist – אופציה זו מבטלת את הפלט ש-PROC DATASETS מפיקה לחלון Log.
אופן הכתיבה:

nolist

ההוראה APPEND

ההוראה APPEND מצרפת את התצפיות של קובץ SAS אחד לסוף של קובץ SAS אחר.

אופן הכתיבה:

APPEND <שם ספרייה> base = קובץ אליו רוצים להוסיף <אופציות שונות>;

דוגמא:

```
proc datasets;
 append base = hump2 data = hump3;
run;
```

בדוגמא זו, התצפיות בקובץ hump3 יסופחו לקובץ hump2. בחלון Log תופיע ההודעה:

NOTE: Appending WORK.HUMP3 to WORK.HUMP2.

כפי שניתן לראות, בהוראה APPEND אפשר להגדיר גם קובץ שלא נמצא בספריה עליה עובדת PROC DATASETS (על ידי הגדרת שם הספרייה ואחריה נקודה, לפני שם הקובץ אליו רוצים להוסיף).

הערה: במקום ההוראה APPEND, ניתן להשתמש בפרוצדורה PROC APPEND (שלא תידון בספר זה). עם זאת, ראוי לציין כי ההוראה APPEND ו-PROC APPEND זהות לגמרי מבחינת התפקוד (וההגדרה) שלהם, למעט ספריית ברירת המחדל עליה עובדת הפרוצדורה.

אופציות של ההוראה APPEND

האופציה data – אופציה זו מגדירה קובץ אותו יש לספח לקובץ המוגדר על ידי הפקודה base. במקרה שמשמיטים אופציה זו (דהיינו, משתמשים רק בפקודה base), PROC DATASETS תצרף לקובץ המוגדר על ידי base את עצמו (תכפיל אותו).

אופן הכתיבה:

APPEND <ספרייה> base = קובץ אלינו רוצים להוסיף<ספרייה> = data קובץ אותנו רוצים להוסיף;

ההוראה CHANGE

ההוראה CHANGE משנה שמות של קבצי נתונים בספריית העבודה.

אופן הכתיבה:

CHANGE שם ישן 1 = שם חדש 1…<שם ישן n = שם חדש n><memtype = סוג קובץ>;

דוגמא:

```
proc datasets;
   change hump = dogma1 hump2 = dogma2 hump3 = dogma3
   hump4 = dogma4;
run;
```

במקרה הזה, SAS תשנה את שמות קבצי הנתונים hump-hump4 ל-dogma1-dogma4. החלון Log יציג את ההודעה:

```
NOTE: Changing the name WORK.HUMP to WORK.DOGMA1 (memtype=DATA).
NOTE: Changing the name WORK.HUMP2 to WORK.DOGMA2 (memtype=DATA).
NOTE: Changing the name WORK.HUMP3 to WORK.DOGMA3 (memtype=DATA).
NOTE: Changing the name WORK.HUMP4 to WORK.DOGMA4 (memtype=DATA).
```

אופציות של ההוראה CHANGE

האופציה memtype זמינה להוראה CHANGE. מאחר וההגדרה כמו גם הפונקציה של memtype נידונו כבר בחלק הדן באופציות של PROC DATASETS, לא נחזור על דיון זה כאן.

ההוראה CONTENTS

ההוראה CONTENTS מציגה מידע על קבצי הנתונים הקיימים בספריית העבודה של SAS. כברירת מחדל, ההוראה CONTENTS מציגה מידע רק על קובץ הנתונים האחרון שנמצא בזיכרון של SAS.

המידע על קובץ הנתונים מופק לחלון Output, והוא כולל את תאריך היצירה שלו, את מספר התצפיות הקיימות בו, מספר המשתנים ושמותיהם, סוג המשתנים, אורכם, מיקומם בקובץ, התווית שלהם והפורמאט שלהם (אם הוגדר).

לדוגמא:

```
                    Extinction Phase - Full vs. Partial_2

                        The DATASETS Procedure

Data Set Name        WORK.HUMP4                      Observations          80
Member Type          DATA                            Variables             4
Engine               V9                              Indexes               0
Created              13:30:56 2006 ינואר 15 יום׳ראשון  Observation Length    32
Last Modified        13:30:56 2006 ינואר 15 יום׳ראשון  Deleted Observations  0
Protection                                           Compressed            NO
Data Set Type                                        Sorted                NO
Label
Data Representation  WINDOWS_32
Encoding             whebrew  Hebrew (Windows)


                     Engine/Host Dependent Information

Data Set Page Size          4096
Number of Data Set Pages    1
First Data Page             1
Max Obs per Page            126
Obs in First Data Page      80
Number of Data Set Repairs  0
File Name                   C:\DOCUME~1\GUYHOC~1\LOCALS~1\Temp\SAS
Temporary Files\_TD2936\hump4.sas7bdat
Release Created             9.0101M3
Host Created                XP_PRO


                Alphabetic List of Variables and Attributes

                #    Variable    Type    Len

                1    b           Num       8
                4    block       Num       8
                2    cond        Num       8
                3    p_risk      Num       8
```

אופן הכתיבה:

CONTENTS<אופציות שונות>;

הערה: במקום ההוראה CONTENTS, ניתן להשתמש בפרוצדורה PROC CONTENTS (שלא תידון בספר זה). עם זאת, ראוי לציין כי ההוראה CONTENTS ו-PROC CONTENTS זהות לגמרי מבחינת התפקוד (וההגדרה) שלהם, למעט ספריית ברירת המחדל עליה עובדת הפרוצדורה.

ההוראה COPY

ההוראה COPY מעתיקה קבצי נתונים מספרייה אחת לספרייה אחרת.

אופן הכתיבה:

COPY out = ספרייה 1 in = ספרייה 2;

ספרייה 1, המוגדרת על ידי הפקודה out, מציינת את הספרייה אליה רוצים להעתיק את קובצי הנתונים המוגדרים על ידי ההוראה. ספרייה 2, המוגדרת על ידי הפקודה in, מציינת את הספרייה ממנה רוצים להעתיק את קובצי הנתונים המוגדרים על ידי ההוראה.

ההוראה EXCLUDE

ההוראה EXCLUDE מגדירה קבצי SAS שלא יועתקו לספרייה חדשה כאשר מגדירים את ההוראה COPY. לכן, הוראה זו חייבות לבוא ביחד עם ההוראה COPY.

אופן הכתיבה:

COPY out = ספרייה 1 in = ספרייה 2;
EXCLUDE שמות קבצי SAS</memtype = סוג קובץ>;

דוגמא:

```
copy out = myworks in = work;
exclude dogma1 dogma2/memtype = data;
```

הערה: לא ניתן להשתמש בהוראה EXCLUDE כאשר מגדירים את ההוראה SELECT (שתידון להלן).

אופציות של ההוראה EXCLUDE

האופציה memtype – אופציה זו פועלת בדיוק כמו בהוראות הקודמות שנידונו, אך היא מוגדרת מעט אחרת.

אופן הכתיבה:

/memtype = סוג הקובץ

ההוראה SELECT

ההוראה SELECT מגדירה קבצי SAS ספציפיים שיועתקו לספרייה חדשה כאשר מגדירים את ההוראה COPY. לכן, הוראה זו חייבות לבוא ביחד עם ההוראה COPY.

אופן הכתיבה:

```
COPY out = שם הספרייה אליה רוצים להעתיק<אופציות שונות>;
SELECT   SAS שמות קבצי</memtype = סוג קובץ>;
```

הערה: האופציה memtype הזמינה להוראה SELECT מוגדרת ופועלת באותה צורה בה היא מוגדרת ופועלת בהוראות הקודמות של PROC DATASETS שנידונו לעיל.

ההוראה DELETE

ההוראה DELETE מגדירה קבצי SAS אותם יש למחוק מספריית העבודה.

אופן הכתיבה:

```
DELETE   SAS קבצי <memtype>;
```

הערה: האופציה memtype הזמינה להוראה DELETE מוגדרת ופועלת באותה צורה בה היא מוגדרת ופועלת בהוראות הקודמות של PROC DATASETS שנידונו לעיל.

ההוראה MODIFY

הוראה זו משנה את המאפיינים של קבצי SAS (למשל את התוויות של הקובץ) ושל המשתנים הכלולים בהם.

אופן הכתיבה:

```
MODIFY   SAS קובץ <(אופציות של הקובץ)><memtype = סוג קובץ>;
```

אופציות של ההוראה MODIFY

האופציה label – אופציה זו יוצרת (או משנה או מוחקת במקרה בו מוגדרת לקובץ תווית) את התווית של קובץ SAS המוגדר בהוראה.

אופן הכתיבה:

```
(label = 'מחרוזת בין 1 ל-40 תווים');
```

הערה: כאשר רוצים למחוק תווית, יש להשתמש בקוד:

```
(label = ) |  (label = '')
```

ההוראה FORMATS

ההוראה FORMATS מגדירה, משנה או מסירה פורמאטים למשתנים המוגדרים בהוראה MODIFY. לכן, ההוראה FORMATS חייבת להופיע ביחד עם ההוראה MODIFY.

אופן הכתיבה:

MODIFY SAS שם קובץ;
FORMAT משתנה 1 <פורמאט 1>.
. .
<משתנה *n* <פורמאט *n*>.>;

דוגמא:

```
proc datasets;
 modify hump4;
 format cond reinf.;
run;
```

הערה: כדי להוריד פורמאט ממשתנה, יש להשמיט מהקוד את שם הפורמט. לדוגמא:

```
proc datasets;
 modify hump4;
 format cond;
run;
```

ההוראה INFORMATS

ההוראה INFORMATS מגדירה, משנה או מסירה informats למשתנים המוגדרים בהוראה MODIFY. לכן, ההוראה INFORMATS חייבת להופיע ביחד עם ההוראה MODIFY.

אופן הכתיבה:

MODIFY SAS שם קובץ;
INFORMAT משתנה 1 <informat.>
. .
. .
<משתנה *n* < informat*n*.>>;

דוגמא:

```
proc datasets;
 modify hump4;
 informat block sivuv.;
run;
```

הערה: כדי להוריד informat ממשתנה, יש להשמיט מהקוד את שם ה-informat.

ההוראה LABELS

ההוראה LABELS מגדירה, משנה או מסירה תוויות למשתנים המוגדרים בהוראה MODIFY. לכן, ההוראה זו חייבת להופיע ביחד עם ההוראה MODIFY.

אופן הכתיבה:

MODIFY SAS שם קובץ;
LABELS משתנה 1 = <'תווית 1'>
. .
<*n* משתנה <'תווית n'>>;

דוגמא:

```
proc datasets;
 modify hump4;
  label cond = 'Tnai';
run;
```

הערה: הורדת תוויות ממשתנה נעשית באותה צורה כמו הורדת תוויות מקובץ SAS (ראה האופציה label של ההוראה MODIFY).

ההוראה RENAME

ההוראה RENAME משנה את השם של משתנים בתוך קובץ SAS המוגדר בהוראה MODIFY. לכן הוראה זו חייבת להופיע ביחד עם ההוראה MODIFY.

אופן הכתיבה:

MODIFY SAS שם קובץ;
RENAME שם ישן 1 = שם חדש 1…<שם ישן *n* = שם חדש *n*>;

הערה: שם חדש שמוגדר למשתנה לא יכול להיות שם שקיים כבר בקובץ.

ההוראה REPAIR

ההוראה REPAIR מנסה לשחזר קובץ SAS פגוע במטרה להחזיר אותו למצב שמיש. לרוב, קובץ SAS פגום הוא תוצאה של:

א. קריסת מערכת בזמן עבודה על קובץ
ב. דיסק פגום
ג. טעות I/O בזמן עבודה על הקובץ
ד. הדיסק התמלא במהלך הניסיון לשמור קובץ מעודכן

אופן הכתיבה:

REPAIR SAS שם קבצי<memtype = סוג קובץ>;

ההוראה SAVE

ההוראה SAVE מוחקת את כל קבצי SAS הנמצאים בספריית העבודה למעט אלה המוגדרים על ידי ההוראה (שומרת על קבצים המוגדרים על ידה ממחיקה).

אופן הכתיבה:

SAVE SAS קבצי</memtype = סוג הקובץ>;

תרגול עצמי – טיפול בקבצי נתונים

תרגיל 21

כתוב קוד SAS לייבוא קובץ Excel. קרא לקובץ האקסל targil21.xls, ולקובץ הנתונים של targil21 SAS. על הקובץ לכלול בשורה הראשונה של כל עמודה את שם המשתנה. הקוד לייבוא צריך להעביר לקובץ הנתונים של SAS את כל המשתנים המופיעים בו.

תרגיל 22

כתוב קוד SAS לייבוא קובץ txt. קרא לקובץ ה - targil22.txt txt, ולקובץ הנתונים של targil22 SAS. הקובץ txt לא יכלול שורת שמות משתנים. הקוד לייבוא צריך להעביר לקובץ הנתונים של SAS את כל המשתנים (ואת התצפיות שלהם) החל מהשורה השלישית.

תרגיל 23

הנח כי עבדת על מספר קבצי נתונים ב-SAS בספריית ברירת המחדל (work). לאחר שסיימת לעבוד, אתה מעוניין להעביר את כל הקבצים שיצרת לתיקיית העבודה שלך (לדוגמא הספרייה myworks) שנמצאת בתיקייה MyDocuments שבכונן D.

כתוב קוד SAS להעביר את כל קבצי הנתונים שיצרת לתיקיית העבודה שלך.

תרגיל 24

לאחר שעבדת על מספר קבצי נתונים ב-SAS, קיימים בספריית העבודה שלך קבצי הנתונים הבאים: exp1, exp2, raw_1, raw_2, sub_data, means_dat, results. כתוב תוכנית SAS למחוק את קבצי הנתונים raw_1, raw_2 ו-exp1, ולהשאיר את כל שאר קבצי הנתונים.

פרק 9
פרוצדורות סטטיסטיות I:
סטטיסטיקה תיאורית

PROC MEANS

הפרוצדורה MEANS מחשבת סטטיסטים תיאוריים (ממוצע, סטית תקן, רבעונים, טווח, רווח סמך וכדומה) לכל משתנה נומרי בקובץ הנתונים (דהיינו לכל עמודה). בנוסף, החל מגרסא 8 של SAS, הפרוצדורה יכולה גם להחזיר ערכי t לבדיקת השערות (כאשר השערת האפס היא שהתוחלת שווה לאפס).

אם לא הוגדרו בקוד סטטיסטים מסוימים, כברירת מחדל הפרוצדורה MEANS מחשבת את כל הסטטיסטים התיאוריים הכלולים בה (רשימה מפורטת ראה טבלה 6), אך מפיקה לקובץ קלט רק את הסטטיסטים מספר תצפיות (n), ממוצע (Mean), סטיית תקן (Std Dev), ערך מינימאלי (min), וערך מקסימאלי (max). אלא אם מוגדר אחרת (כפי שיפורט בהמשך), PROC MEANS מפיקה אוטומטית קובץ פלט.

לדוגמא, הרצה בסיסית של PROC MEANS לקובץ נתונים בעל 3 משתנים (גיל, מין ותשלום) תפיק את הפלט שלהלן בחלון Output:

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
age	9	27.0000000	3.0822070	23.0000000	32.0000000
gen	9	0.5555556	0.5270463	0	1.0000000
pay	9	99.4444444	6.8211273	90.0000000	110.0000000

אופן הכתיבה:

PROC MEANS <אופציות שונות> <מילות קוד לסטטיסטים>;
BY <descending> רשימת משתנים <notsorted>;
CLASS משתנים</אופציות שונות>;
OUTPUT out = שם </אופציות שונות> הגדרת סטטיסטים להפקה לקובץ;
TYPES בקשות;
VAR משתנים;
RUN;

דוגמא:

```
proc means data = dogma;
 by sex;
 var choice pay;
run;
```

1. האופציה alpha – אופציה זו קובעת את הטווח של רווח הסמך המופק על ידי הפרוצדורה. הטווח מוגדר על ידי הנוסחא 100*(1- alpha). כברירת מחדל, טווח רווח הסמך הוא 95 (alpha = 0.05).
אופן הכתיבה:

ערך מספרי (בין 0 ל – 1) = alpha

מילת קוד	הסטטיסטי
N	מספר התצפיות בקובץ (לא כולל ערכים חסרים)
Nmiss	מספר התצפיות בקובץ (כולל ערכים חסרים)
MIN	הערך המינימאלי
MAX	הערך המקסימאלי
RANGE	טווח התצפיות
SUM	סכום התצפיות
MEAN	ממוצע התצפיות
VAR	שונות התצפיות
STD	סטיית התקן של התצפיות
CV	מקדם השונות
STDERR	סטיית התקן של הממוצע
CLM	רווח סמך לתוחלת
UCLM	גבול העליון של רווח סמך לתוחלת
LCLN	גבול תחתון של רווח הסמך לתוחלת
T	ערך הסטטיסטי t לבדיקת ההשערה $H_0 = 0$
PRT	הגדרת הערך p (רמת המובהקות) לבדיקת השערת האפס
SKEWNESS	בדיקת הסימטריות של ההתפלגות
KURTOSIS	בדיקת מידת השטיחות של ההתפלגות והזנבות ביחס להתפלגות נורמאלית
USS	סכום הריבועים של כל התצפיות
CSS	סכום הפרשי הריבועים מהממוצע
P*n*	האחוזון ה-*n* י (n = 99, 95, 90, 10, 5, 1)
Q1	הרבעון הראשון
Q3	הרבעון השלישי
MEDIAN	החציון (רבעון שני)

טבלה 6 – סטטיסטים תיאוריים הזמינים ב-PROC MEANS

2. האופציה maxdec – אופציה זו מגדירה את מספר הספרות אחרי הנקודה שיופיעו בסטטיסטים בקובץ הפלט. כברירת מחדל, SAS מציגה 7 ספרות אחרי הנקודה.
אופן הכתיבה:

ערך מספרי (בין 0 ל – 7) = maxdec

3. האופציה nonobs – אופציה זו אומרת ל-SAS להשמיט את העמודה המציינת את מספר התצפית לכל משתנה או סטטיסטי, במצבים בהם SAS מוסיפה עמודה זו (העמודה N obs). כברירת מחדל, SAS מוסיפה עמודה זו כאשר מגדירים משתנה/י CLASS.

אופן הכתיבה:

nonobs

4. האופציה noprint – כפי שצוין, ברירת המחדל של PROC MEANS היא להפיק קובץ output. האופציה noprint אומרת ל-SAS לא להציג את הניתוח הסטטיסטי שבוצע בפרוצדורה. נהוג להשתמש באופציה כאשר רוצים להשתמש ב-PROC MEANS כדי להפיק קובץ נתונים חדש.
אופן הכתיבה:

noprint

5. האופציה vardef – אופציה זו מגדירה ל-SAS את המחלק בו יש להשתמש בחישוב הסטטיסטים שונות וסטיית תקן. כברירת מחדל המחלק לחישוב הסטטיסטים הוא דרגות החופש (דהיינו n-1). אולם, ניתן להגדיר ל-SAS גם מחלקים אחרים.

הערכים האפשריים של המחלקים הם:
א. DF – דרגות החופש (n-1)
ב. N – מספר התצפיות
ג. WDF – סכום המשקולות פחות 1
ד. WGT – סכום המשקולות

אופן הכתיבה:

vardef = מחלק

ההוראה BY

הוראה זו אומרת ל-SAS לחשב את הסטטיסטים באופן נפרד לכל קבוצה של המשתנה המוגדר על ידי ההוראה. אלא אם מוגדר אחרת, קובץ הנתונים עליו מורצת PROC MEANS חייב להיות מסודר (על ידי PROC SORT) לפי המשתנה/משתנים המוגדרים בהוראה BY.

אופן הכתיבה:

BY <descending> משתנה 1 <descending> משתנה n <notsorted>;

דוגמא:

```
proc means data = dogma maxdec = 4;
  by gender;
run;
```

כתוצאה מהרצת קוד זה, יופיע בחלון Output הפלט:

```
------------------------------------- gender=0 -------------------------------------

                          The MEANS Procedure

Variable     N            Mean         Std Dev         Minimum         Maximum
_____________________________________________________________________________
age          5         25.2000          2.3875         23.0000         29.0000
pay          5         98.0000          7.5829         90.0000        110.0000
_____________________________________________________________________________
```

```
---------------------------------------- gender=1 ----------------------------------------

Variable     N            Mean         Std Dev         Minimum         Maximum
_____________________________________________________________________________________
age          3         28.6667          2.8868         27.0000         32.0000
pay          3         86.0000          5.2915         80.0000         90.0000
_____________________________________________________________________________________
```

ההוראה CLASS

הוראה זו אומרת ל-SAS לחשב את הסטטיסטים באופן נפרד לכל קבוצה של המשתנה המוגדר על ידי ההוראה. בניגוד להוראה BY, הקובץ עליו נעשה עיבוד לא חייב להיות מסודר על פי המשתנה/משתנים המוגדרים על ידי ההוראה.

אופן הכתיבה:

CLASS רשימת משתנים </אופציות שונות>;

דוגמא:

```
proc means data = dogma maxdec = 4;
   class gender;
run;
```

כאמור, ההוראה CLASS מבצעת פעולה זהה לזו שמבצעת ההוראה BY. עם זאת, דרך התצוגה של ההוראות שונה. במצב בו משתמשים בהוראה CLASS, יתקבל קובץ הפלט הבא:

```
                                The MEANS Procedure

gender     Obs     Variable     N        Mean         Std Dev        Minimum     Maximum
_____________________________________________________________________________________
0          5       age          5        25.2000      2.3875         23.0000      29.0000
                   pay          5        98.0000      7.3875         90.0000     110.0000

1          3       age          3        28.6667      2.8868          27.0000     32.0000
                   pay          3        86.0000      5.2915          80.0000     90.0000
_____________________________________________________________________________________
```

אופציות של ההוראה CLASS

1. האופציה ascending – אופציה זו אומרת ל-SAS לסדר את הקובץ OUTPUT לפי משתני ה-CLASS בסדר עולה.
   אופן הכתיבה:

/ascending

2. האופציה descending – אופציה זו אומרת ל-SAS לסדר את הקובץ output לפי משתני ה-CLASS בסדר יורד.
   אופן הכתיבה:

/descending

3. האופציה missing – אופציה זו מגדירה ל-PROC MEANS להתייחס לערכים חסרים כאל קבוצת CLASS נפרדת. כברירת מחדל, אם לא משתמשים באופציה זו, SAS תשמיט מהניתוח הסטטיסטי את כל התצפיות בהן יש ערכים חסרים למשתנה ה-CLASS.
אופן הכתיבה:

/missing

ההוראה OUTPUT

הוראה זו אומרת ל-SAS ליצור מהסטטיסטים המחושבים ב-PROC MEANS קובץ נתונים חדש.

אופן הכתיבה:

OUTPUT out = שם של קובץ הנתונים החדש
שם לסטטיסטי עבור כל משתנה (על פי סדר הופעתם) = מילת קוד לסטטיסטי
</אופציות שונות>;

דוגמא:

```
proc means noprint;
 output out=dogma2 mean = meanage meanchoice meanpay;
run;
```

קובץ הנתונים החדש dogma2 שנוצר יכיל תצפית אחת ו-6 משתנים (שלושת המשתנים שיצרנו, העמודה של מספר התצפית, ואת המשתנים _TYPE_ ו - _FREQ_, שיידונו בהמשך):

Obs	_TYPE_	_FREQ_	meanage	meanchoice	meanpay
1	0	12	27.6667	0.44444	95.8333

אופציות של ההוראה OUTPUT

1. האופציה out – אופציה זו מגדירה את שם קובץ הנתונים שנוצר על ידי PROC MEANS ושמכיל את הסטטיסטים שהוגדרו על ידי ההוראה.
אופן הכתיבה:

OUTPUT out = שם קובץ נתונים חדש

דוגמא:

```
output out=dogma2;
```

אם לא מגדירים את שם קובץ הנתונים החדש, SAS תתן לו אוטומאטית את השם data*n*, כאשר *n* הוא המספר הקטן ביותר שהופך את השם לייחודי (n = 1 במצב בו לא קיימים קבצים בשם data בזיכרון של התוכנה).

הגדרת סטטיסטים שיאוכסנו בקובץ הנתונים שנוצר על ידי הפרוצדורה נעשית כדלקמן:

שמות (התואמים למשתנים, על פי סדר הופעתם בקובץ) = (משתנה/ים) שם קוד הסטטיסטי

כך, ניתן להגדיר סטטיסטיים ספציפיים להפקה לקובץ הנתונים, גם אם סטטיסטים אלה לא הופיעו בפלט של הפרוצדורה. במצב בו לא מגדירים סטטיסטים ספציפיים לאכסון, , SAS תיצור קובץ נתונים המכיל 5 תצפיות, אחת לכל סטטיסטי המופק לקובץ הפלט כברירת מחדל על ידי PROC MEANS (לכל משתנה בנפרד).

שמות הקוד של הסטטיסטים להפקה לקובץ הנתונים זהים לשמות הקוד לסטטיסטים להפקה לקובץ output (ראה טבלה 6). ניתן להפיק סטטיסטי לכל המשתנים הקיימים בקובץ הנתונים (על ידי השמטת שמות המשתנים הרלוונטיים המופיעים בסוגריים בשורת הקוד – ראה דוגמא להלן), או למשתנה/משתנים מסוימים (על ידי הכנסת שמותיהם, מופרדים על ידי רווחים) לתוך סוגריים אחרי שם קוד הסטטיסטי.

דוגמא:

```
output out=dogma2 mean(age pay) = m_age m_pay max(choice) =
       max_c;
```

2. האופציה autolabel – אופציה זו מגדירה ל-PROC MEANS לצרף את שם הסטטיסטי לתווית של המשתנה (בסוף). אם אין למשתנה תווית, SAS תיצור אוטומטית תווית המכילה את שם המשתנה ולאחריו את שם הסטטיסטי.
אופן הכתיבה:

/autolabel

3. האופציה autoname – אופציה זו אומרת ל-SAS לתת שמות למשתנים הנוצרים על ידי PROC MEANS לקובץ הנתונים החדש באופן אוטומטי. במצב כזה ניתן לחסוך בשורות קוד, שכן אופציה זו חוסכת את הצורך להגדיר שם לכל משתנה שנוצר בקובץ החדש. בנוסף, באמצעות אופציה זו ניתן ליצור מספר עמודות המכילות את אותו המשתנה, שכן SAS מייחסת לכל עמודה שם ייחודי, אפילו אם העמודה מכילה את אותו סטטיסטי לאותו משתנה.

כאשר משתמשים באופציה זו, יש להשמיט את שמות המשתנים החדשים, אך להשאיר את הסימן = (שווה) אחרי הגדרת הסטטיסטים להפקה.
אופן הכתיבה:

/autoname

דוגמא:

```
output out=dogma2 mean(age pay) = max(choice) = max(choice) =
       /autoname;
```

כתוצאה מהרצת קוד זה, יתקבל הקובץ output:

Obs	_TYPE_	_FREQ_	age_Mean	pay_Mean	choice_Max	choice_Max2
1	0	12	27.6667	95.8333	0.65	0.65

ההוראה TYPES

הוראה זו מגדירה את הצירופים של המשתנים המוגדרים על ידי ההוראה CLASS שיופקו על ידי הפרוצדורה. כדי להשתמש בהוראה זו, חובה להגדיר משתני CLASS.

אופן הכתיבה:

TYPES בקשות;

בקשות:

הבקשות מגדירות אילו מבין 2^k הקומבינציות האפשריות של משתני ה-CLASS ישמשו כדי ליצור את ה-types. במקרה דנן, k מציין את מספר המשתנים המוגדרים על ידי ההוראה CLASS.

כל בקשה מורכבת משם של משתנה CLASS אחד, מספר שמות של משתני CLASS מופרדים על ידי כוכבית (*) או סוגריים. אופן הכתיבה של בקשות מקביל לסדר הפעולות של מכפלת מספרים בסוגריים. לכן, ניתן לכתוב syntax לבקשות באופן מקוצר, כפי שמודגם להלן:

- כדי לבקש את השילוב של משתנה A ומשתנה B ואת השילוב של משתנה A עם משתנה C:

A*(B C) או A*B A*C

- כדי לבקש את השילוב משתנה A עם משתנה C, משתנה A עם משתנה D, משתנה B עם משתנה C ומשתנה B עם משתנה D:

(A B) (C D) או A*C A*D B*C B*D

- כדי לבקש את השילוב של משתנה D עם A, משתנה D עם B ומשתנה D עם C:

(A B C)*D או A*D B*D C*D

הערה: כדי לבקש שילוב של כל משתני ה-CLASS האפשריים, יש להשתמש בבקשה ( ).

דוגמא:

```
proc means mean std n nonobs;
 class cond run_n gender;
 types cond*run_n run_n*gender;
run;
```

בדוגמא זו הוגדרו המשתנים cond (המציין את תנאי הניסוי), run_n (המציין את בלוק הניסוי), ו-gender (מין) כמשתני CLASS, וההוראה TYPES מגדירה ל-PROC MEANS להפיק את שילוב משתני ה-CLASS cond ו-run_n ואת השילוב run_n ו-gender. כפי שניתן לראות, הקוד בדוגמא מגדיר ל-PROC MEANS לחשב את הסטטיסטים של הממוצע, סטיית התקן וכמות התצפיות.

כתוצאה מהרצת הקוד, יתקבל הפלט:

The MEANS Procedure

run_n	gender	Variable	Mean	Std Dev	N
1	0	pay	91.7500000	9.7898370	12
		choice	0.4441667	0.0862827	12
	1	pay	92.8333333	10.7552997	18
		choice	0.5216667	0.1413485	18
2	0	pay	89.7500000	14.1107631	12
		choice	0.4208333	0.0652907	12
	1	pay	92.8888889	9.0741319	18
		choice	0.5027778	0.1089417	18

cond	run_n	Variable	Mean	Std Dev	N
1	1	pay	95.0000000	8.4983659	10
		choice	0.4700000	0.1197219	10
	2	pay	89.8000000	11.1634324	10
		choice	0.4700000	0.1197219	10
2	1	pay	92.1000000	10.7852780	10
		choice	0.5010000	0.1352734	10
	2	pay	92.2000000	13.6772317	10
		choice	0.4700000	0.0966092	10
3	1	pay	90.1000000	11.5993295	10
		choice	0.5010000	0.1352734	10
	2	pay	92.9000000	9.4451634	10
		choice	0.4700000	0.0966092	10

לעומת זאת, במצב בו נריץ את אותו הקוד ללא ההוראה TYPES, יתקבל הפלט הבא:

The MEANS Procedure

cond	run_n	gender	Variable	Mean	Std Dev	N
1	1	0	pay	97.5000000	8.6602540	4
			choice	0.4125000	0.0997914	4
		1	pay	93.3333333	8.7559504	6
			choice	0.5083333	0.1241639	6
	2	0	pay	81.0000000	8.6023253	4
			choice	0.4125000	0.0997914	4
		1	pay	95.6666667	8.7559504	6
			choice	0.5083333	0.1241639	6
2	1	0	pay	89.0000000	2.0000000	4
			choice	0.4600000	0.0875595	4
		1	pay	94.1666667	13.9343700	6
			choice	0.5283333	0.1615446	6
	2	0	pay	96.7500000	16.9975488	4
			choice	0.4250000	0.0525991	4
		1	pay	89.1666667	11.6518954	6
			choice	0.5000000	0.1115347	6
3	1	0	pay	88.7500000	14.3614066	4
			choice	0.4600000	0.0875595	4
		1	pay	91.0000000	10.7703296	6
			choice	0.5283333	0.1615446	6
	2	0	pay	91.5000000	13.9880902	4
			choice	0.4250000	0.0525991	4
		1	pay	93.8333333	6.3691967	6
			choice	0.5000000	0.1115347	6

ההוראה VAR

הוראה זו מגדירה לפרוצדורה על אילו משתנים לעשות את חישוב הסטטיסטים, כמו גם את סדר הופעתם בקובץ output.
אם לא מגדירים הוראה זו, הפרוצדורה תחשב סטטיסטיים לכל המשתנים בקובץ, למעט משתנים המוגדרים בהוראות אחרות של הפרוצדורה.

הערה: כאשר כל המשתנים בקובץ הנתונים הם אלפאנומריים, הפרוצדורה תפיק רק את הסטטיסטי N, המציין את מספר התצפיות לכל משתנה.

אופן הכתיבה:

VAR רשימת משתנים;

דוגמא:

```
proc means mean std n nonobs;
 var payoff choice;
run;
```

PROC SUMMARY

הפרוצדורה SUMMARY מחשבת סטטיסטים תיאוריים למשתנים מעבר לכל התצפיות או בתוך קבוצה של תצפיות. כברירת מחדל, PROC SUMMARY לא מפיקה קובץ output. פרוצדורה זו דומה מאוד ל-PROC PRINT, ולמעט מספר הבדלים (שיידונו להלן), כל המידע המופיע בתת-הפרק הקודם (הפרק על PROC MEANS) רלוונטי ל- PROC SUMMARY.

אופן הכתיבה:

PROC SUMMARY <אופציות שונות> <מילות קוד לסטטיסטים>;
BY <descending> רשימת משתנים <notsorted>;
CLASS משתנים</אופציות שונות>;
OUTPUT out = שם הגדרת סטטיסטים להפקה לקובץ </אופציות שונות>;
TYPES בקשות;
VAR משתנים;
RUN;

הבדלים בין PROC SUMMARY ל-PROC MEANS

בעוד שברירת המחדל של PROC MEANS הינה להפיק קובץ output, ברירת המחדל של PROC SUMMARY הינה לא להפיק קובץ זה. לכן, אם רוצים ש-PROC SUMMARY תפיק קובץ output, יש להשתמש באופציה print (הזהה באופן הכתיבה שלה ובהגדרה שלה לאופציה noprint של PROC MEANS).

בנוסף, כאשר משמיטים את ההוראה VAR מהקוד של PROC SUMMARY, הפרוצדורה תפיק ספירה פשוטה של התצפיות בלבד, לעומת PROC MEANS, שתחשב את הסטטיסטים לכל המשתנים הנומריים בקובץ שאינם מוגדרים על ידי הוראה אחרת.

מלבד הבדלים אלה, PROC SUMMARY ו-PROC MEANS זהות.

דוגמא:

```
proc summary print;
 var pay choice;
run;
```

הקוד הבסיסי של PROC SUMMARY המופיע בדוגמא יפיק בחלון Output את הפלט:

The SUMMARY Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
pay	60	92.0166667	10.6556131	70.0000000	110.0000000
choice	60	0.4803333	0.1141062	0.2700000	0.7000000

PROC UNIVARIATE

הפרוצדורה UNIVARIATE מחשבת סטטיסטים עבור משתנים נומריים, מידע על התפלגות המשתנים, טבלת שכיחויות, ומספקת תצוגות גראפיות.

כברירת מחדל, PROC UNIVARIATE מחשבת את כל הסטטיסטים הכלולים בפרוצדורה (ראה פירוט בטבלה 7), ולא ניתן לחשב רק חלק מהם. בנוסף, PROC UNIVARIATE מפיקה אוטומטית קובץ פלט.

לדוגמא, הרצה בסיסית של PROC UNIVARIATE לקובץ נתונים המכיל את המשתנים מספר התלמיד וציון מבחן (כאשר הפרוצדורה מוגדרת לספק סטטיסטים למשתנה ציון המבחן בלבד) תפיק את הפלט שלהלן בחלון Output:

The UNIVARIATE Procedure
Variable: g1

Moments

N	10	Sum Weights	10
Mean	80.7	Sum Observations	807
Std Deviation	14.7651993	Variance	218.011111
Skewness	-0.858228	Kurtosis	0.78581848
Uncorrected SS	67087	Corrected SS	1962.1
Coeff Variation	18.2964056	Std Error Mean	4.669166

Basic Statistical Measures

Location		Variability	
Mean	80.70000	Std Deviation	14.76520
Median	82.50000	Variance	218.01111
Mode	90.00000	Range	50.00000
		Interquartile Range	18.00000

Tests for Location: Mu0=0

Test	-Statistic-		-----p Value------	
Student's t	t	17.2836	Pr > |t|	<.0001
Sign	M	5	Pr >= |M|	0.0020
Signed Rank	S	27.5	Pr >= |S|	0.0020

Quantiles (Definition 5)

Quantile	Estimate
100% Max	100.0
99%	100.0
95%	100.0
90%	97.5
75% Q3	90.0

```
50% Median        82.5
25% Q1            72.0
10%               59.0
5%                50.0
1%                50.0
0% Min            50.0

       Extreme Observations
----Lowest----        ----Highest---
Value      Obs        Value      Obs
   50        9           85        1
   68        8           90        2
   72        3           90        7
   77       10           95        5
   80        4          100        6
```

אופן הכתיבה:

PROC UNIVARIATE אפציות שונות;
BY <descending> 1 משתנה <descending> *n* משתנה <notsorted>;
ID רשימת משתנים;
CLASS 1 משתנה <(אופציות של המשתנה)> 2 משתנה <(אופציות של המשתנה)>>;
FREQ משתנים;
HISTOGRAM רשימת משתנים </ אופציות שונות>><(תת אופציות שונות)>;
PROBPLOT משתנים </ אופציות שונות> <(תת אופציות שונות)>;
QQPLOT משתנים </ אופציות שונות><(תת אופציות שונות)>;
INSET מילות מפתח DATA=שם קובץ נתונים</ אופציות שונות>;
OUTPUT out = שם קובץ נתונים הגדרת סטטיסטים = שמות לסטטיסטים <הגדרת אחוזונים>;
VAR רשימת משתנים;
RUN;

דוגמא:

```
proc univariate;
 var grade;
run;
```

טיפ קריאה: זכור, אופציות של הפרוצדורות לא הכרחיות להבנת הפרוצדורה או לתפקודה. לכן, מומלץ כי קוראים מתחילים ידלגו בשלב הראשון על החלק הדן באופציות ויתמקדו בהוראות העיקריות של כל פרוצדורה.

סטטיסטים המופיעים בפלט תחת הכותרת **Moments**	
הסטטיסטי	**השם המופיע בפלט**
מספר התצפיות (לא כולל ערכים חסרים)	N
ממוצע התצפיות	Mean
סטיית התקן	Std Deviation
מידת וכיוון האסימטריות של התפלגות התצפיות	Skewness
סכום ריבועי התצפיות	Uncorrected SS

Coeff variation	מקדם השונות
Sum weights	סכום המשקולות (כברירת מחדל המשקולת לכל תצפית שווה ל-1)
Sum observations	סכום התצפיות
Variance	שונות
Kurtosis	חוזק הזנב של ההתפלגות – מדד לבדיקת מידת הנורמאליות של ההתפלגות (ערכו 0 כאשר ההתפלגות נורמאלית).
Corrected SS	סכום ריבועי ההפרשים מהממוצע
Std Error Mean	סטיית התקן של הממוצע
סטטיסטי המופיעים בפלט תחת הכותרת Basic statistical Measures	
השם המופיע בפלט	**הסטטיסטי**
Mean	ממוצע
Median	חציון
Mode	שכיח
Std Deviation	סטיית תקן
Variance	שונות
Range	טווח התצפיות
Interquartile Range	הטווח הבין-רבעוני
סטטיסטים המופיעים בפלט תחת הכותרת Tests for Location Mu0 = 0	
השם המופיע בפלט	**הסטטיסטי**
Student's t	ערך t לבדיקת ההשערה שהתוחלת שווה ל-Mu0
Sign	ערך סטטיסטי לבדיקת ההשערה שהחציון שווה ל-Mu0
Signer Rank	ערך סטטיסטי למבחן Wilcoxon לבחינת ההשערה שהחציון שווה ל-Mu0
סטטיסטים המופיעים בפלט תחת הכותרת Quantiles	
השם המופיע בפלט	**הסטטיסטי**
n%	האחוזון ה-*n* י
סטטיסטים המופיעים בפלט תחת הכותרת Extreme Observations	
השם המופיע בפלט	**הסטטיסטי**
Lowest	חמשת התצפיות בעלות הערכים הנמוכים ביותר בהתפלגות (מציין את מספר התצפית בקובץ המקורי ואת הערך שלה)
Highest	חמשת התצפיות בעלות הערכים הגבוהים ביותר בהתפלגות (מציין את מספר התצפית בקובץ המקורי ואת הערך שלה)
סטטיסטים המופיעים בפלט תחת הכותרת Missing Values (מופיע בפלט רק כאשר יש ערכים חסרים בקובץ)	
השם המופיע בפלט	**הסטטיסטי**
Missing Value	התו המציין ערכים חסרים (יכול להיות יותר מאחד)
Count	מספר התצפיות בעלות ערכים חסרים בקובץ
Percent Of All obs	אחוז התצפיות בעלות ערכים חסרים מתוך כלל התצפיות בקובץ
Percent Of Missing obs	אחוז התצפיות בעלות תו ספציפי המציין ערכים חסרים מתוך כלל מספר התצפיות בעלות ערכים חסרים בקובץ

טבלה 7 – הסטטיסטים התיאוריים ב-PROC UNIVARIATE

<u>אופציות כלליות</u>:

1. האופציה data – אופציה זו מגדירה את קובץ הנתונים עליו תעבוד הפרוצדורה. אם אופציה זו לא מוגדרת, הפרוצדורה תעבוד על קובץ הנתונים האחרון שנוצר.
אופן הכתיבה:

data = שם קובץ נתונים

<u>אופציות הקשורות לניתוחים הסטטיסטיים</u>:

1. האופציה cibasic – אופציה זו אומרת ל-SAS להוסיף לניתוח רווח סמך לממוצע, לסטיית התקן ולשונות, תוך הנחה שהנתונים מתפלגים נורמאלית.
אופן הכתיבה:

cibasic (type = מילת מפתח alpha = ערך בין 0 ל-1)

דוגמא:

```
proc univariate cibasic;
 var fg;
run;
```

הפקודה type מגדירה את סוג הגבול של רווח הסמך, כאשר מילות המפתח האפשריות הן lower לגבול תחתון בלבד, upper לגבול עליון בלבד, ו-twosided לשני הכיוונים. מאחר וברירת המחדל היא רווח סמך דו צדדי, אם לא רוצים גבול אחד, אין צורך להגדיר את ה-Type. באופן דומה, הפקודה alpha מגדירה את רמת הביטחון של רווח הסמך. ברירת המחדל של התוכנה היא 95%, כך שבמצב בו לא רוצים רמת ביטחון אחרת, ניתן להשמיט פקודה זו.

אופציה זו מוסיפה לפלט את הנתונים הבאים:

Basic Confidence Limits Assuming Normality

Parameter	Estimate	95% Confidence Limits	
Mean	75.27273	64.14585	86.39960
Std Deviation	16.56255	11.57254	29.06618
Variance	274.31818	133.92365	844.84287

2. האופציה cipctldf – אופציה זו אומרת ל-SAS להוסיף לניתוח רווח סמך לאחוזונים בהסתמך על שיטה שלא מניחה שהנתונים מתפלגים נורמאלית.
אופן הכתיבה:

cipctldf (type = מילת מפתח alpha = ערך בין 0 ל-1)

דוגמא:

```
proc univariate cipctldf (type = upper alpha = 0.1);
 var fg;
run;
```

אופציה זו מוסיפה לפלט את הנתונים הבאים:

Quantiles (Definition 5)

Quantile	Estimate	90% Upper Confidence Bound Distribution Free	--Order Statistics-- UCL Rank	Coverage
100% Max	95.5			
99%	95.5	95.5	11	10.47
95%	95.5	95.5	11	43.12
90%	95.5	95.5	11	68.62
75% Q3	84.5	95.5	11	95.78
50% Median	78.0	84.5	9	96.73
25% Q1	69.5	78.0	6	96.57
10%	60.5	69.5	3	91.04
5%	37.0	69.5	3	98.48
1%	37.0	60.5	2	99.48
0% Min	37.0			

3. האופציה alpha – אופציה זו מגדירה את רמת הביטחון לחישוב גבולות רווח הסמך. רמת הביטחון מחושבת על פי הנוסחא:

100 x (1 – הערך המוגדר על ידי האופציה)

אופן הכתיבה:

alpha = ערך בין 0 ל- 1

4. האופציה mu0 – אופציה זו מגדירה את הערך של הממוצע או פרמטר המיקום (לדוגמא החציון) בהשערת האפס. אם מגדירים ערך אחד, PROC UNIVARIATE משתמשת באותה השערת אפס לכל המשתנים הכלולים בניתוח. לעומת זאת, אם רוצים להגדיר ערך שונה לכל משתנה, יש להגדיר מספר ערכים באופציה, כאשר הערכים מופרדים על ידי רווחים. במצב זה הפרוצדורה משתמשת בהשערת אפס שונה לכל משתנה המוגדר על ידי ההוראה VAR (שתוסבר בהמשך), לפי הסדר בו הערכים והמשתנים מוגדרים (בהתאמה). כברירת מחדל, הערך של השערת האפס הוא 0.
אופן הכתיבה:

Mu0 = ערכים (מופרדים על ידי רווחים)

5. האופציה nextrval – אופציה זו מגדירה את מספר הערכים הקיצוניים ש-PROC UNIVARIATE מציגה בטבלת הערכים הקיצוניים. כברירת מחדל, טבלה זו לא מוצגת (שכן הערך ברירת המחדל הוא 0).
אופן הכתיבה:

nextrval = מספר שלם בין 0 למחצית ממספר התצפיות

אופציה זו תוסיף לקובץ הפלט את טבלת הערכים הקיצוניים (3 בדוגמא הנוכחית):

Extreme Values

Order	Lowest Value	Lowest Freq	Order	Highest Value	Highest Freq
1	37.0	1	8	82.5	1
2	60.5	1	9	84.5	1
3	69.5	1	10	95.5	2

6. האופציה normal – אופציה זו מבקשת מבחן לבדיקת נורמאליות הכוללת את מבחן Shapiro-Wilk וסדרת מבחני goodness-of-fit המבוססים על פונקצית התפלגות אמפירית.
אופן הכתיבה:

normal

אופציה זו מוסיפה לפלט את הנתונים הבאים:

Tests for Normality

Test	--Statistic---		-----p Value------	
Shapiro-Wilk	W	0.911097	Pr < W	0.2513
Kolmogorov-Smirnov	D	0.181899	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.054735	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq	0.388198	Pr > A-Sq	>0.2500

7. האופציה vardef – אופציה זו מגדירה ל-SAS את המחלק בו יש להשתמש בחישוב הסטטיסטים שונות וסטיית תקן. כברירת מחדל המחלק לחישוב הסטטיסטים הוא דרגות החופש (דהיינו n-1). אולם, ניתן להגדיר ל-SAS גם מחלקים אחרים.

הערכים האפשריים של המחלקים הם:

א. DF – דרגות החופש (n-1)

ב. N – מספר התצפיות

ג. WDF – סכום המשקולות פחות 1

ד. WGT – סכום המשקולות

אופן הכתיבה:

vardef = מחלק

<u>אופציות הקשורות לקובץ הפלט</u>:

1. האופציה noprint – אופציה זו אומרת לפרוצדורה לא להפיק פלט לחלון Output. אופציה זו חייבת לבוא ביחד עם ההוראה OUTPUT (שתידון בהרחבה בהמשך).
אופן הכתיבה:

noprint

2. האופציה freq – אופציה זו אומרת לפרוצדורה להוסיף לפלט טבלת שכיחויות הכוללת את ערכי המשתנים, השכיחות של כל ערך, כמה אחוזים מהווה כל שכיחות, ונתוני אחוזים מצטברים.
אופן הכתיבה:

freq

אופציה זו מוסיפה לפלט את הנתונים הבאים:

Frequency Counts

		Percents				Percents				Percents	
Value	Count	Cell	Cum	Value	Count	Cell	Cum	Value	Count	Cell	Cum
37.0	1	9.1	9.1	72.0	1	9.1	45.5	82.5	1	9.1	72.7
60.5	1	9.1	18.2	78.0	1	9.1	54.5	84.5	1	9.1	81.8
69.5	1	9.1	27.3	81.5	1	9.1	63.6	95.5	2	18.2	100.0
71.5	1	9.1	36.4								

3. האופציה modes – אופציה זו אומרת לפרוצדורה להוסיף לפלט טבלה של כל השכיחים. כברירת מחדל, כאשר הנתונים כוללים מספר ערכים שכיחים, PROC UNIVARIATE מציגה את השכיח הקטן ביותר בטבלה של המדדים הסטטיסטים הבסיסיים. במצב בו כל תצפית היא ייחודית, הפרוצדורה לא תפיק את טבלת השכיחים, אפילו אם היא מוגדרת.

אופן הכתיבה:

modes

אופציה זו מוסיפה לפלט את הנתונים הבאים:

Modes

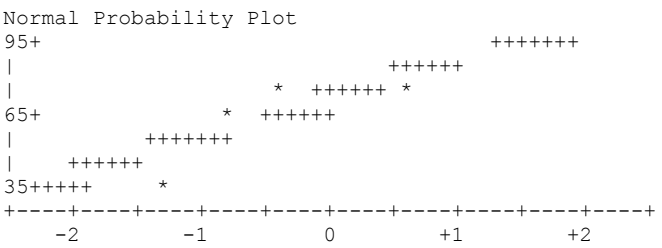
Mode	Count
78.0	2
95.5	2

4. האופציה plot – אופציה זו מפיקה stem-and-leaf plot, box plot ו-normal probability plot.
אופן הכתיבה:

plots

אופציה זו מוסיפה לפלט את התרשימים הבאים:

```
Stem Leaf                #             Boxplot
9 66                     2                |
8 2                      1             +--+--+
7 022889                 6             +--+--+
6 0                      1                |
5
4
3 7                      1                0
----+----+----+----+
Multiply Stem.Leaf by 10**+1
```



ההוראה BY

הוראה זו אומרת לפרוצדורה לחשב סטטיסטים בנפרד לכל קבוצת BY. הוראה זו מפיקה פלט נפרד לכל קבוצה המוגדרת על ידי משתנה ה-BY.

אופן הכתיבה:

By <decending> משתנה 1 <descending> משתנה*n* <notsorted>

דוגמא:

```
proc univariate;
 by gender;
 var fg;
run;
```

ההוראה ID

הוראה זו מזהה את התצפיות החריגות המוצגות בטבלת התצפיות החריגות, דהיינו מציינת מה הערך של התצפית במשתנה (או משתנים) המוגדר על ידי ההוראה, המתאים לכל תצפית חריגה. ערכי התצפיות של המשתנים המוגדרים על ידי ההוראה ID נוספים לטבלת התצפיות החריגות כעמודה בין העמודה המציינת את ערכי התצפיות החריגות במשתנה עליו עובדת הפרוצדורה לבין העמודה המציינת את השכיחות של כל תצפית חריגה.

אופן הכתיבה:

ID משתנה/ים;

ההוראה CLASS

הוראה זו מגדירה משתנה אחד או שניים (מקסימום) שהפרוצדורה תעשה בהם שימוש כדי לקבץ את הניתוחים לקבוצות שונות, בהתאם לערכי המשתנה/משתנים. משתני CLASS (משתנים קטגוריאליים) יכולים להיות נומריים או אלפאנומריים, והם יכולים להיות בדידים או רציפים. בניגוד להוראה BY, לא חייבים למיין את הנתונים לפי המשתנים המוגדרים על ידי ההוראה. PROC UNIVARIATE משתמשת בערכי הפורמאט של משתני ה-CLASS כדי לקבוע את רמות המיון השונות של המשתנה.

כאשר מגדירים את ההוראות HISTOGRAM, PROBPLOT ו-QQPLOT (הוראות שיידונו בהמשך) ביחד עם ההוראה CLASS, SAS תיצור תרשים השוואתי לכל קבוצה. כאשר מגדירים רק משתנה CLASS אחד, PROC UNIVARIATE תציג מערך של תרשימים, אחד לכל ערך של המשתנה. כאשר מגדירים שני משתני CLASS, הפרוצדורה תיצור מטריצת תרשימים, כאשר כל תא מייצג שילוב בין הערכים של משתני ה-CLASS.

אופן הכתיבה:

CLASS משתנה1 <(אופציות שונות)> משתנה2 <(אופציות שונות)>;

דוגמא:

```
proc univariate noprint;
 class age (missing) gender;
 var fg;
run;
```

אופציות של ההוראה CLASS

1. האופציה missing – אופציה זו אומרת ל-PROC UNIVARIATE להתייחס לערכים חסרים במשתנה/י ה-CLASS כערכים תקפים לקיבוץ. אם נעשה שימוש בתווים מיוחדים לציון ערכים חסרים (האותיות A עד Z וקו תחתון _), הקיבוץ ייעשה לכל תו בנפרד.
   אם משמיטים אופציה זו, PROC UNIVARIATE לא תכלול את התצפיות עם הערכים החסרים בניתוח.
   אופן הכתיבה:

(missing)

2. האופציה order – אופציה זו מגדירה את הסדר שבו הערכים של משתני ה-CLASS יופיעו בקובץ output.

סדר ההצגה יכול להיות מוגדר בהתאם ל –

א. DATA – סדר הערכים בקובץ הפלט יהיה בהתאם לסדר בו הם מופיעים בקובץ הנתונים.
ב. FORMATED – סדר הערכים בקובץ הפלט יהיה לפי ערכי הפורמאט בו מוגדרים משתני ה-CLASS (בסדר עולה).
ג. FREQ – סדר הערכים בקובץ הפלט יהיה בהתאם לשכיחות התצפיות של כל ערך. הסידור יהיה כך שקבוצה עם הכי הרבה תצפיות תופיע בקובץ הפלט ראשונה וכך הלאה עד הקבוצה עם הכי פחות תצפיות.
ד. INTERNAL – סדר הערכים בקובץ הפלט יהיה על פי הערך שלהם באופן הזהה לסדר הנעשה על פי PROC SORT. כברירת מחדל (אם משמיטים אופציה זו), PROC UNIVARIATE תשתמש ב-INTERNAL.

אופן הכתיבה:

(order = data | formatted | freq | internal)

ההוראה HISTOGRAM

הוראה זו אומרת ל-PROC UNIVARIATE ליצור היסטוגרמות באיכות גבוהה (תרשים שנוצר בתוכנה SAS/GRAPH ולא בחלון Output של SAS) למשתני הניתוח. בנוסף, הוראה זו יכולה להוסיף לתרשימי ההיסטוגרמה גם עקומות התפלגות פרמטריות ואפרמטריות לנתונים.

אופן הכתיבה:

HISTOGRAM משתנים </אופציות שונות>;

הערה: כאשר לא מגדירים משתנים בהוראה, PROC UNIVARIATE תיצור תרשים לכל משתנה בהוראה VAR. כאשר גם ההוראה VAR לא מוגדרת, הפרוצדורה תיצור תרשים לכל משתנה נומרי בקובץ הנתונים.

דוגמא:

```
proc univariate noprint;
 histogram fg;
run;
```

דוגמא זו תפיק את ההיסטוגרמה המוצגת באיור 14.

אופציות של ההוראה HISTOGRAM

טיפ קריאה: להוראה HISTORGAM יש המון אופציות, כולל אופציות רבות לעיצוב מותאם אישית של התרשים, אשר אינן חיוניות להרצת הקוד. לכן, מומלץ בשלב הראשון לא להתמקד באופציות אלה.

אופציות הקשורות לאופי התצוגה:

1. האופציה barwidth – אופציה זו מגדירה את רוחב העמודות בהיסטוגרמה (ביחידות של אחוזי מסך). כברירת מחדל, רוחב העמודות מוגדר ל-20 (ראה איור 14 לדוגמא).

אופן הכתיבה:

/barwidth = מספר

2. האופציה forcehist – כברירת מחדל, PROC UNIVARIATE לא תיצור היסטוגרמה כאשר סטיית התקן של הנתונים שווה לאפס (למשל כאשר יש רק תצפית אחת). אופציה זו "כופה" על הפרוצדורה ליצור היסטוגרמה במצב בו ברירת המחדל מגדירה לה לא ליצור.
אופן הכתיבה:

/forcehist

3. האופציה grid – אופציה זו אומרת לפרוצדורה להוסיף קווים אופקיים לשנתות של ציר x בהיסטוגרמה.
אופן הכתיבה:

/grid

4. האופציה hoffset – אופציה זו מגדירה את המרווח הריק (ביחידות אחוז מסך) משני צידי הציר האופקי של התרשים. כברירת מחדל, ערך זה מוגדר ל-1 (ראה איור 14 לדוגמא). אם רוצים לבטל את המרווח לגמרי, יש לקבוע את ה- offset לאפס.
אופן הכתיבה:

/hoffset = מספר

5. האופציה href – אופציה זו מוסיפה קווים מקווקווים אופקיים לתרשים בנקודה/נקודות המוגדרות על ידי האופציה (קווי ייחוס).
אופן הכתיבה:

/href = ערכים בטווח המשתנים (מופרדים על ידי רווחים)

6. האופציה hreflabels – אופציה זו מגדירה תוויות לקווי הייחוס האופקיים שהוגדרו באופציה href.
אופן הכתיבה:

/hreflabels = 'תווית 1' ... 'תווית *n*'

7. האופציה hreflabpos – אופציה זו מגדירה היכן למקם על ציר קו הייחוס את התווית שלו.
ניתן למקם את התווית על פי ההגדרות שלהלן:

א. למקם את התווית מצד ימין של קו הייחוס, בקצה העליון של ההיסטוגרמה. מיקום זה הוא ברירת המחדל של הפרוצדורה. הקוד של מיקום זה הוא 1.

ב. למקם את התווית של קו הייחוס הראשון בקצה העליון (מצד ימין של הקו), ולמקם כל תווית נוספת קצת יותר נמוך מהתווית הקודמת. הקוד של מיקום זה הוא 2.

ג. למקם את התווית מצד ימין של קו הייחוס, בקצה התחתון של ההיסטוגרמה. הקוד של מיקום זה הוא 3.

אופן הכתיבה:

/hreflabpos = קוד

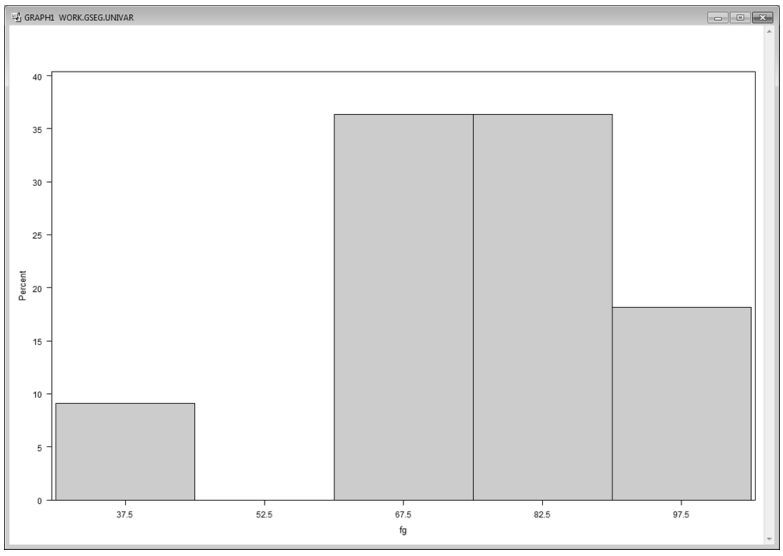
8. האופציה vref – אופציה זו מוסיפה קווי ייחוס אנכיים לתרשים בנקודה/נקודות המוגדרות על ידי האופציה.
אופן הכתיבה:

/vref = ערכים בטווח המשתנים (מופרדים על ידי רווחים)

9. האופציה vreflabels – אופציה זו מגדירה תוויות לקווי הייחוס האנכיים שהוגדרו באופציה vref.

אופן הכתיבה:

/vreflabels = 'תווית 1' ... 'תווית *n*'



איור 14 – פלט בסיסי של ההוראה HISTOGRAM ב-PROC UNIVARIATE

10. האופציה vreflabpos – אופציה זו מגדירה היכן למקם על ציר קו הייחוס את התווית שלו.
ניתן למקם את התווית על פי ההגדרות שלהלן:
   א. למקם את התווית מצד שמאל של ההיסטוגרמה. הקוד של מיקום זה הוא 1 (מיקום ברירת המחדל).
   ב. למקם את התווית מצד ימין של ההיסטוגרמה. הקוד של מיקום זה הוא 2.

אופן הכתיבה:

/hreflabpos = קוד

11. האופציה lgrid– אופציה זו מגדירה את סגנון הקו של הקווים האופקיים המוגדרים על ידי האופציה grid. אם משתמשים באופציה זו, היא מגדירה אוטומטית את האופציה grid. כברירת מחדל, סגנון הקו מוגדר ל-1 (המייצג קו רציף).
סגנון קו 2 מייצג קו מקווקו. שאר הסגנונות מייצגים קווים מקווקווים באורכים שונים, שילובים בין קווים ארוכים לקצרים, או בין קווים לנקודות.
אופן הכתיבה:

/lgrid = סגנון הקו (מספר בין 1 ל - 46)

12. האופציה lhref – אופציה זו מגדירה את סגנון הקו של קווי הייחוס האופקיים המוגדרים על ידי האופציה href.
כברירת מחדל, סגנון הקו מוגדר ל-2 (קו מקווקו).
אופן הכתיבה:

/lhref = סגנון הקו

13. האופציה lvref – אופציה זו מגדירה את סגנון הקו של קווי הייחוס האנכיים המוגדרים על ידי האופציה vref.
כברירת מחדל, סגנון הקו מוגדר ל-2 (קו מקווקו).
אופן הכתיבה:

/lvref = סגנון הקו

14. האופציה midpoints – אופציה זו מגדירה איך לקבוע את נקודות האמצע של כל עמודה בהיסטוגרמה.

אופן הכתיבה:

/midpoints = ערך מינימום to ערך מקסימום by אינטרוול

כאשר:

אינטרוול-רוחב כל עמודה (על ציר x).
ערך מינימום –

(2/אינטרוול) – הערך של התצפית המינימאלית בקובץ הנתונים

ערך מקסימום –

(2/אינטרוול) – הערך של התצפית המקסימאלית בקובץ הנתונים

הערה: כאשר ערך המינימום המוגדר גדול מהערך הממשי שלו, וערך המקסימום קטן מהערך הממשי שלו, SAS תתאים ערכים אלה באופן אוטומטי, ותכתוב את ההודעה הבאה בחלון Log:

```
WARNING: The MIDPOINTS= list was extended to accommodate the data.
```

15. האופציה nobars – אופציה זו משמיטה את עמודות ההיסטוגרמה מהתרשים. יש להשתמש באופציה זו כאשר רוצים להציג את עקומת ההתפלגות בלבד.
אופן הכתיבה:

/nobars

16. האופציה noframe – אופציה זו משמיטה את המסגרת סביב ההיסטוגרמה.
אופן הכתיבה:

/noframe

17. האופציה nohlabel – אופציה זו משמיטה את התווית של ציר x.
אופן הכתיבה:

/nohlabel

18. האופציה novlabel – אופציה זו משמיטה את התווית של ציר y.
אופן הכתיבה:

/novlabel

19. האופציה noplot – אופציה זו מבטלת את היצירה של התרשים. כאשר משתמשים באופציה זו, יש להשתמש באופציה Outhistogram (ראה פירוט בהמשך). משתמשים באופציה זו כאשר רוצים ליצור קובץ נתונים המכיל נתונים על ההיסטוגרמה, או כאשר רוצים להפיק סטטיסטים על התפלגות הדגימה בלבד.
אופן הכתיבה:

/noplot

20. האופציה noprint – אופציה זו מבטלת את הפקת טבלת הסטטיסטים ש-PROC UNIVARIATE מפיקה כאשר מבקשים להוסיף לתרשים עקומת התפלגות. סטטיסטים אלה מלמדים האם עקומת ההתפלגות המחושבת מתאימה לנתונים.

אופן הכתיבה:

/noprint

21. האופציה novtick – אופציה זו משמיטה את השנתות ואת תוויות השנתות מהציר y.
אופן הכתיבה:

/novtick

22. האופציה vaxis – אופציה זו מגדירה את ערכי השנתות בציר y. כדי להגדיר אופציה זו, יש להשתמש בערכים בעלי אינטרוולים שווים (לדוגמא, 10, 20, 30) ולהציג אותם בסדר עולה. בנוסף, הערך ההתחלתי חייב להיות 0, והערך האחרון חייב להיות שווה או גדול יותר מהערך של העמודה הגבוהה ביותר (התצפית הכי גדולה). לבסוף, יחידות המדידה צריכות להתאים למשתנה אותו מציגים בהיסטוגרמה.
אופן הכתיבה:

/vaxis = ערכים בסדר עולה (מופרדים על ידי רווחים)

23. האופציה voffset – אופציה זו מגדירה את המרווח בין הקצה העליון של ציר y לשנתה האחרונה על ציר זה, ביחידות של אחוזי מסך. כברירת מחדל, אופציה זו מוגדרת ל-0.5 (כאשר הערך 0 אומר שהשנתה האחרונה תופיע בדיוק בקצה של ציר y.
אופן הכתיבה:

/voffset = ערך

24. האופציה waxis – אופציה זו מגדירה את עובי הקווים (בפיקסלים) של הצירים, המסגרת וקווי הייחוס בתרשים. כברירת מחדל, עובי הקווים מוגדר ל-1.
אופן הכתיבה:

/waxis = ערך מספרי

25. האופציה wbarline – אופציה זו מגדירה את העובי של הקווים המקיפים את עמודות ההיסטוגרמה. כברירת מחדל, עובי הקווים מוגדר ל-1.
אופן הכתיבה:

/wbarline = ערך מספרי

26. האופציה wgrid – אופציה זו מגדירה את העובי של קווי הרשת האופקיים המוגדרים על ידי האופציה grid. כברירת מחדל, עובי הקווים מוגדר ל-1.
אופן הכתיבה:

/wgrid = ערך מספרי

27. האופציה vscale – אופציה זו מגדירה את הסקאלה של ציר y. הסקאלה יכולה להיות אחת מהאופציות
   א. Count – מספר התצפיות לכל יחידת נתונים
   ב. Percent – אחוז התצפיות בכל יחידת נתונים. זוהי סקאלת ברירת המחדל
   ג. Proportion – הפרופורציה של התצפיות בכל יחידת נתונים
אופן הכתיבה:

/vscale = סקאלה

דוגמא:

```
histogram fg /vscale = proportion;
```

28. אופציה vaxislable – אופציה זו מגדירה תווית לציר y. תוויות אלה יכולות להיות בעלות 40 תווים מקסימום.
אופן הכתיבה:

/vaxislabel = 'תווית'

29. האופציה caxis – אופציה זו מגדירה את הצבע של קווי הצירים.
אופן הכתיבה:

/caxis = שם הצבע הרצוי (באנגלית)

דוגמא:

```
histogram fg /caxis = red
```

בדוגמא זו, ציר x וציר y שבתרשים יוצגו בצבע אדום.

30. האופציה cbarline – אופציה זו מגדירה את הצבע של הקווים המקיפים את עמודות ההיסטוגרמה.
אופן הכתיבה:

/cbarline = שם הצבע הרצוי (באנגלית)

31. האופציה cfill – אופציה זו מגדירה את הצבע של עמודות ההיסטוגרמה.
אופן הכתיבה:

/cfill = שם הצבע הרצוי (באנגלית)

32. האופציה cframe – אופציה זו מגדירה את הצבע של הרקע של המסגרת סביב עמודות ההיסטוגרמה. כאשר האופציה noframe מוגדרת, לאופציה זו אין השפעה על התרשים. כברירת מחדל SAS משאירה את המסגרת סביב ההיסטוגרמה ללא צבע.
אופן הכתיבה:

/cframe = שם הצבע הרצוי (באנגלית)

33. האופציה cvref – אופציה זו מגדירה את הצבע של קווי הייחוס האנכיים.
אופן הכתיבה:

/cvref = שם הצבע הרצוי (באנגלית)

34. האופציה chref – אופציה זו מגדירה את הצבע של קווי הייחוס האופקיים.
אופן הכתיבה:

/chref = שם הצבע הרצוי (באנגלית)

35. האופציה ctext – אופציה זו מגדירה את הצבע של תווי הטקסט הנמצאים מחוץ לתרשים (ערכי הצירים והתוויות).
אופן הכתיבה:

/ctext = שם הצבע הרצוי (באנגלית)

36. האופציה cgrid – אופציה זו מגדירה את הצבע של קווי הרשת האופקיים.
אופן הכתיבה:

/cgrid = שם הצבע הרצוי (באנגלית)

37. האופציה font – אופציה זו מגדירה את סוג הפונט של תווי הטקסט הנמצאים מחוץ לתרשים.
אופן הכתיבה:

/font = שם הפונט הרצוי

דוגמא:

```
histogram fg/font = david
```

38. האופציה infont – אופציה זו מגדירה את סוג הפונט של תווי הטקסט הנמצאים בתוך התרשים.
אופן הכתיבה:

/infont = שם הפונט הרצוי

39. האופציה height – אופציה זו מגדירה את גודל הפונטים של תווי הטקסט הנמצאים מחוץ לתרשים. כברירת מחדל, גודל הפונטים מוגדר ל - 1.5.
אופן הכתיבה:

/height = ערך מספרי

40. האופציה inheight – אופציה זו מגדירה את גודל הפונטים של תווי הטקסט הנמצאים בתוך התרשים. כברירת מחדל, גודל הפונטים מוגדר ל - 1.5.
אופן הכתיבה:

/inheight = ערך מספרי

41. האופציה pfill – אופציה זו מגדירה את תבנית המילוי של עמודות ההיסטוגרמה. כברירת מחדל, SAS יוצרת עמודות לא מלאות.
להלן מספר תבניות קישוט הזמינות ב-SAS:
   א. M3N0 – קווים אופקיים
   ב. M3X0 – קווי רשת ישרים
   ג. M3N90 – קווים אנכיים
   ד. M3N45 – קווים אלכסוניים (משמאל לימין)
   ה. M3X45 – קווי רשת אלכסוניים
   ו. MSN135 – קווים אלכסוניים (מימין לשמאל)

אופן הכתיבה:

/pfill = תבנית קישוט

דוגמא:

```
histogram fg/pfill = M3X0;
```

42. האופציה hminor – אופציה זו מגדירה את מספר השנתות הקטנות שיופיעו בין השנתות הראשיות של הציר האופקי. SAS לא מוסיפה תוויות לשנתות קטנות.
אופן הכתיבה:

/hminor = ערך מספרי

43. האופציה vminor – אופציה זו מגדירה את מספר השנתות הקטנות שיופיעו בין השנתות הראשיות של הציר האנכי. SAS לא מוסיפה תוויות לשנתות קטנות.
אופן הכתיבה:

/vminor = ערך מספרי

אופציות הקשורות לתרשימים השוואתיים (הנוצרים כאשר מגדירים את ההוראה CLASS):

1. האופציה cframeside – אופציה זו מגדירה את הצבע למילוי המסגרת של תוויות השורות בתרשים השוואתי (כאשר כל שורה מציינת ערך ייחודי של משתנה ה-CLASS). כברירת מחדל, מסגרות אלה לא צבועות.
אופן הכתיבה:

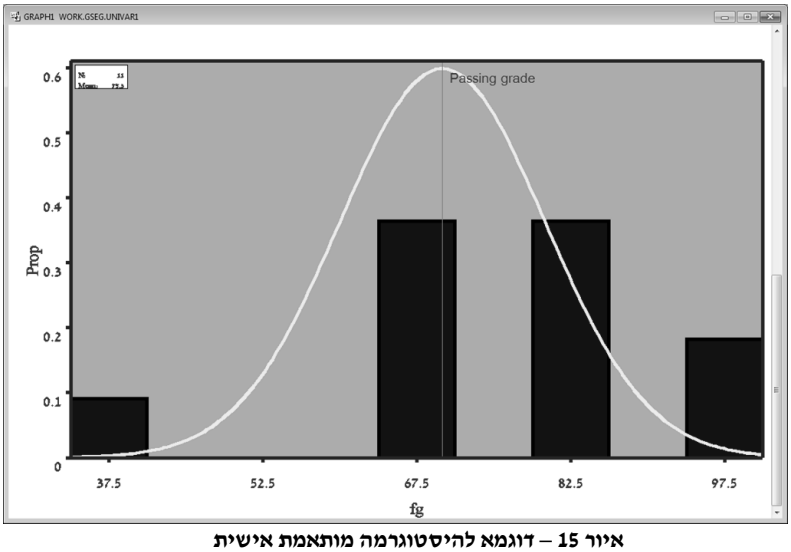
/cframeside = שם הצבע הרצוי (באנגלית)

2. האופציה cframetop – אופציה זו מגדירה את הצבע למילוי המסגרת של תוויות העמודות בתרשים השוואתי (רלוונטי כאשר מוגדרים שני משתני CLASS). כברירת מחדל, מסגרות אלה לא צבועות.
אופן הכתיבה:

/cframetop = שם הצבע הרצוי (באנגלית)

3. האופציה intertile – אופציה זו מגדירה את המרווח האופקי בין התרשימים ההשוואתיים (באחוזי מסך). כברירת מחדל, אופציה זו מוגדרת ל-0.75.
אופן הכתיבה:

/intertile = ערך מספרי



איור 15 – דוגמא להיסטוגרמה מותאמת אישית

4. האופציה nrows – אופציה זו מגדירה את מספר השורות בתרשים השוואתי. כברירת מחדל, מספר השורות מוגדר ל-2.

אופן הכתיבה:

/nrows = ערך מספרי

5. האופציה ncols – אופציה זו מגדירה את מספר העמודות בתרשים השוואתי (רלוונטי רק כאשר מגדירים 2 משתני CLASS). כברירת מחדל, מספר העמודות מוגדר ל-2.
אופן הכתיבה:

/ncols = ערך מספרי

אופציות הקשורות להוספת עקומת התפלגות:

1. האופציה normal – אופציה זו מציגה עקומת התפלגות נורמלית על ההיסטוגרמה. כברירת מחדל, ממוצע המדגם (μ) משמש כממוצע ההתפלגות, וסטיית התקן של המדגם (σ) משמשת כסטיית התקן של ההתפלגות.
אופן הכתיבה:

/normal <(אופציות שונות)>;

הערה: בדומה להוספת עקומת התפלגות נורמאלית, ניתן להציג על ההיסטוגרמה גם עקומת התפלגות פרופורציית beta (באמצעות האופציה beta), עקומה אקספוננציאלית (באמצעות האופציה exponential, עקומת התפלגות גמא (באמצעות האופציה gamma), עקומת kernel (באמצעות האופציה kernel), עקומת התפלגות לוג-נורמלית (באמצעות האופציה lognormal), ועקומת Weibull (באמצעות האופציה weibull). מאחר ואופן הכתיבה של כל האופציות הללו זהה (לדוגמא, כדי להוסיף תרשים התפלגות weibull יש לרשום את האופציה weibull), לא נפרט כאן על כל סוגי העקומות (ראה פירוט על אופציות אלה בחלק הדן בהוראה PROBPLOT). עם זאת יש לציין כי ניתן להציג כמה עקומות התפלגות על אותה היסטוגרמה.

בנוסף, קריאה להוספת עקומת התפלגות מוסיפה לפלט של PROC UNIVARIATE גם את הסטטיסטים של אותה ההתפלגות (האמדים, אחוזונים, ופרמטרים למידת טיב ההתאמה בין העקומה לנתונים). לדוגמא, קריאה לקו התפלגות נורמלית יוסיף לפלט את הנתונים:

The SAS System
The UNIVARIATE Procedure
Fitted Distribution for fg

Parameters for Normal Distribution

Parameter	Symbol	Estimate
Mean	Mu	70
Std Dev	Sigma	10

Goodness-of-Fit Tests for Normal Distribution

Test	---Statistic----		-----p Value-----	
Kolmogorov-Smirnov	D	0.33359915	Pr > D	0.139
Cramer-von Mises	W-Sq	0.41321403	Pr > W-Sq	0.070
Anderson-Darling	A-Sq	3.25753991	Pr > A-Sq	0.022

Quantiles for Normal Distribution

	------Quantile------	
Percent	Observed	Estimated
1.0	37.0000	46.7365
5.0	37.0000	53.5515
10.0	60.5000	57.1845
25.0	69.5000	63.2551
50.0	78.0000	70.0000
75.0	84.5000	76.7449
90.0	95.5000	82.8155
95.0	95.5000	86.4485
99.0	95.5000	93.2635

תת אופציות של ההתפלגויות (חייבות להופיע בסוגריים אחרי הגדרת ההסתברות):

1. האופציה mu – אופציה זו מגדירה את ערך הפרמטר μ (ממוצע האוכלוסייה) ליצירת עקומת ההתפלגות הנורמלית. כברירת מחדל, הערך של μ שווה לממוצע המדגם עליו מבוססת ההיסטוגרמה.
אופן הכתיבה:

/normal (mu = ערך מספרי);

2. האופציה sigma – אופציה זו מגדירה את ערך הפרמטר σ (סטיית התקן של האוכלוסייה) ליצירת עקומת ההתפלגות הנורמלית. כברירת מחדל, הערך של σ שווה לסטיית התקן של המדגם עליו מבוססת ההיסטוגרמה.
אופן הכתיבה:

/normal (sigma = ערך מספרי);

הערה: בדומה, ניתן להשתמש באופציות zeta להגדיר פרמטר מידה לעקומת התפלגות לוג-נורמלית (הפרמטר ζ), באופציה c להגדיר את פרמטר הצורה (C) לעקומת Weibull, באופציות alpha ו-beta להגדיר פרמטרים של צורה (α ו - β) לעקומת beta, באופציה theta כדי להגדיר פרמטר סף (θ) לעקומות התפלגות אקספוננציאליות, wiebull, גמא, בטא ולוג-נורמלי, ובאופציה sigma כדי להגדיר פרמטר מידה להתפלגויות בטא, אקספוננציאליות, גמא ו-wiebull.

3. האופציה color – אופציה זו מגדירה את צבע הקו של עקומת ההתפלגות המוגדרת להוספה להיסטוגרמה.
אופן הכתיבה:

/normal (color = שם הצבע הרצוי);

4. האופציה L – אופציה זו מגדירה את סגנון הקו היוצר את התרשים של עקומת ההתפלגות. כברירת מחדל, סגנון הקו מוגדר ל-1 (קו רצוף).
אופן הכתיבה:

/normal (L = ערך מספרי);

5. האופציה w – אופציה זו מגדירה את עובי הקו היוצר את תרשים עקומת ההתפלגות המוגדרת (בפיקסלים). כברירת מחדל, עובי הקו מוגדר ל-1.
אופן הכתיבה:

/normal (w = ערך מספרי);

6. האופציה fill – אופציה זו מגדירה שהאזור בהיסטוגרמה מתחת לעקומת ההתפלגות (שטח העקומה) יהיה צבוע בצבע המוגדר על ידי ההוראה cfill, ובדפוס המוגדר על ידי ההוראה pfill (או, אם לא מוגדרת הוראה זו, ללא דפוס).
אופן הכתיבה:

/normal (fill)

הערה: בניגוד לאופציות תצוגה אחרות של קווי המגמה, ניתן להגדיר את האופציה fill רק באופציה של עקומת התפלגות אחת.

לשם המחשה, איור 15 מציג כמה מהאופציות הגרפיות העיקריות של ההוראה HISTOGRAM. הקוד שיצר היסטוגרמה זו מוצג להלן:

```
proc univariate noprint;
 histogram fg /normal (color = yellow w = 5 mu = 70 sigma = 10 )
 cframe = orange ctext = red font = david infont = miriam height
 = 3 hminor=5 cfill=blue cbarline = black caxis = purple barwidth
 = 10 hoffset=0 href=70 wbarline = 5 waxis = 5 vscale =
 proportion vaxislabel='Prop' hreflabels='Passing grade';
 inset N='N:' (10.0) MEAN = 'Mean:' (4.1)/CFILL=white ;
run;
```

ההוראה PROBPLOT

הוראה זו יוצרת probability plots תוך שימוש בתרשימים באיכות גבוהה המשמשים להערכה האם האחוזונים של הנתונים תואמים לאחוזונים של התפלגות תיאורטית מסוימת (לדוגמא התפלגות weibull).

כברירת מחדל, הוראה זו משווה את הנתונים להתפלגות נורמאלית.

אופן הכתיבה:

PROBPLOT משתנה/משתנים </ אופציות שונות>(תת אופציות שונות)>>;

דוגמא:

```
proc univariate noprint;
 probplot risk;
run;
```

דוגמא זו תפיק את הפלט הבסיסי של ההוראה, כפי שמוצג באיור 16.

אופציות של ההוראה PROBPLOT

הגדרת התפלגות:

1. האופציה beta – אופציה זו מבקשת התפלגות הסתברות בטא.
   אופן הכתיבה:

/beta <(אופציות שונות)>;

הערה: כאשר מבקשים התפלגות בטא, חייבים להגדיר את האופציות alpha ו-beta (שיורחבו בהמשך).

טיפ קריאה: בדומה להוראה HISTORGAM, גם להוראה PROBPLOT יש המון אופציות אשר אינן חיוניות להרצת הקוד. לכן, למתכנתים חדשים לא מומלץ בשלב הראשון להתמקד באופציות אלה.

2. האופציה exponential – אופציה זו מבקשת התפלגות אקספוננציאלית.
   אופן הכתיבה:

/exponential <(אופציות שונות)>;

3. האופציה gamma – אופציה זו מבקשת התפלגות הסתברות גמא.
אופן הכתיבה:

/gamma <(אופציות שונות)>;

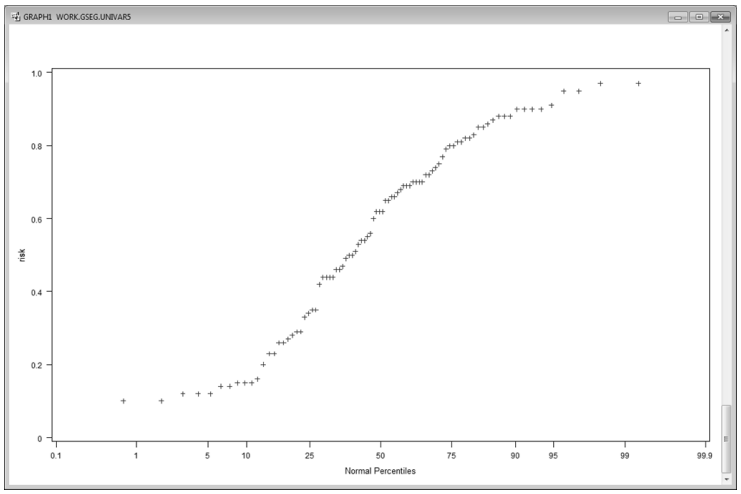
הערה: כאשר מבקשים התפלגות גמא, יש להגדיר את האופציה alpha.

4. האופציה lognormal – אופציה זו מבקשת התפלגות לוג-נורמאלית.
אופן הכתיבה:

/lognormal <(אופציות שונות)>;

5. האופציה normal – אופציה זו מבקשת התפלגות נורמאלית.
אופן הכתיבה:

/normal <(אופציות שונות)>;



איור 16 – פלט בסיסי של ההוראה PROBPLOT ב-PROC UNIVARIATE

6. האופציה weibull – אופציה זו מבקשת התפלגות הסתברות weibull, בעלת 3 פרמטרים.
אופן הכתיבה:

/weibull <(אופציות שונות)>;

הערה: כאשר מבקשים התפלגות weibull, יש להגדיר את האופציה c, שתוגדר להלן.

7. האופציה weibull2 – אופציה זו מבקשת התפלגות הסתברות weibull, בעלת 2 פרמטרים.
אופן הכתיבה:

/weibull2 <(אופציות שונות)>;

תת אופציות של ההתפלגויות (חייבות להופיע בסוגריים אחרי הגדרת ההסתברות):

1. האופציה alpha – אופציה זו מגדירה את פרמטר הצורה α של ההתפלגויות בטא או גמא. אם רוצים ש- PROC UNIVARIATE תחשב אמד לפרמטר, יש לכתוב באופציה את הפקודה est.

אופן הכתיבה:

/gamma|beta (alpha = ערך מספרי גדול מאפס | est);

הערה: ניתן לכתוב יותר מערך אחד אחרי הסימן שווה (כאשר הערכים מופרדים על ידי רווחים). PROC UNIVARIATE תיצור עקומה עבור כל ערך שהוגדר. הערה זו נכונה גם לאופציות beta ו-c שיוגדרו להלן.

2. האופציה beta – אופציה זו מגדירה את פרמטר הצורה β של ההתפלגות בטא.
אופן הכתיבה:

/beta (beta = ערך מספרי גדול מאפס | est);

3. האופציה c – אופציה זו מגדירה את פרמטר הצורה c של ההתפלגות weibull, או את קו הייחוס של ההתפלגות weibull2.
אופן הכתיבה:

/weibull|weibull2 (c = ערך מספרי גדול מאפס | est);

4. האופציה mu – אופציה זו מגדירה את הממוצע μ של ההתפלגות הנורמאלית. אם אופציה זו לא מוגדרת, היא מקבלת כברירת מחדל את הערך של ממוצע המדגם.
אופן הכתיבה:

/normal (mu = ערך מספרי);

5. האופציה sigma – אופציה זו מגדירה את הפרמטר σ, שמשמעותו משתנה בהתאם להתפלגות שהוגדרה. בנוסף, עם שילוב עם אופציות אחרות, האופציה sigma משמשת כדי לבקש קו ייחוס להסתברות.
   א. שילוב של האופציות sigma ו-Theta: הוספת קו ייחוס להתפלגויות beta, exponential, gamma, weibull
   ב. האופציה sigma בלבד: בהתפלגות נורמלית – סטיית התקן. בהתפלגות לוג-נורמלית – פרמטר הצורה
   ג. שילוב של האופציות sigma ו-mu: הוספת קו ייחוס להתפלגות הנורמלית
   ד. שילוב של sigma ו-c: הוספת קו ייחוס להתפלגות weibull2
אופן הכתיבה:

/התפלגות (sigma = ערך מספרי גדול מאפס | est);

6. האופציה slope – אופציה זו מגדירה השיפוע של קו הייחוס, כאשר מבקשים התפלגות לוג-נורמלית או התפלגות weibull2.
אופן הכתיבה:

/lognormal|weibull2 (slope = ערך מספרי | est);

7. האופציה theta – אופציה זו מגדירה את הסף התחתון של הפרמטר θ לכל העקומות, מלבד העקומה הנורמלית. כאשר משתמשים באופציה זו עם תת אופציה נוספת של ההתפלגות (אלפא, בטא, זטה וכדומה), theta משמשת להערכת θ_0 לקו הייחוס של ההתפלגות.
אופן הכתיבה:

/התפלגות (למעט נורמלית) (theta = ערך מספרי | est);

8. האופציה zeta – אופציה זו מגדירה ערך לפרמטר ζ להתפלגות לוג-נורמלית.

אופן הכתיבה:

/lognormal (zeta = ערך מספרי | est);

אופציות הקשורות לקו הייחוס להתפלגות (חייבות להופיע בסוגריים אחרי הגדרת ההסתברות):

1. האופציה color – אופציה זו מגדירה את הצבע של קו הייחוס האלכסוני להתפלגות.
אופן הכתיבה:

(color = צבע);

2. האופציה L – אופציה זו מגדירה את סגנון הקו של קו הייחוס האלכסוני להתפלגות. כברירת מחדל, סגנון הקו מוגדר ל-1 (קו רציף).
אופן הכתיבה:

(L = סגנון קו);

3. האופציה w – אופציה זו מגדירה את עובי הקו (בפיקסלים) של קו הייחוס האלכסוני להתפלגות. כברירת מחדל, עובי הקו מוגדר ל-1.
אופן הכתיבה:

(w = ערך מספרי);

אופציות הקשורות לתצוגה:

1. האופציה grid – אופציה זו אומרת לפרוצדורה להוסיף קווים אנכיים לציר x (ציר האחוזונים), החופפים לשנתות המרכזיות.
אופן הכתיבה:

/grid

2. האופציה lgrid– אופציה זו מגדירה את סגנון הקו של הקווים האופקיים המוגדרים על ידי האופציה grid. אם משתמשים באופציה זו, היא מגדירה אוטומטית את האופציה grid. כברירת מחדל, סגנון הקו מוגדר ל-1 (המייצג קו רציף).
אופן הכתיבה:

/lgrid = סגנון הקו

3. האופציה href – אופציה זו מוסיפה קווים מקווקווים אופקיים לתרשים בנקודה/נקודות המוגדרות על ידי האופציה (קווי ייחוס).
אופן הכתיבה:

/href = ערכים בטווח המשתנים (מופרדים על ידי רווחים)

4. האופציה hreflabels – אופציה זו מגדירה תוויות לקווי הייחוס האופקיים שהוגדרו באופציה href.
אופן הכתיבה:

/hreflabels = 'תווית 1' 'תווית *n*' ...

5. האופציה lhref – אופציה זו מגדירה את סגנון הקו של קווי הייחוס האופקיים המוגדרים על ידי האופציה href. כברירת מחדל, סגנון הקו מוגדר ל-2 (קו מקווקו). להזכירך, סגנון קו 1 הוא קו רציף.

אופן הכתיבה:

/lhref = סגנון הקו

6. האופציה vref – אופציה זו מוסיפה קווי ייחוס אנכיים לתרשים בנקודה/נקודות המוגדרות על ידי האופציה.
אופן הכתיבה:

/vref = ערכים בטווח המשתנים (מופרדים על ידי רווחים)

7. האופציה vreflabels – אופציה זו מגדירה תוויות לקווי הייחוס האנכיים שהוגדרו באופציה vref.
אופן הכתיבה:

/vreflabels = 'תווית 1' ... 'תווית *n*'

8. האופציה lvref – אופציה זו מגדירה את סגנון הקו של קווי הייחוס האנכיים המוגדרים על ידי האופציה vref.
כברירת מחד, סגנון הקו מוגדר ל-2 (קו מקווקו).
אופן הכתיבה:

/lvref = סגנון הקו

9. האופציה noframe – אופציה זו משמיטה את המסגרת סביב התרשים.
אופן הכתיבה:

/noframe

10. האופציה caxis – אופציה זו מגדירה את הצבע של קווי הצירים.
אופן הכתיבה:

/caxis = שם הצבע הרצוי (באנגלית)

11. האופציה cframe – אופציה זו מגדירה את הצבע של הרקע של המסגרת.
אופן הכתיבה:

/cframe = שם הצבע הרצוי (באנגלית)

12. האופציה cvref – אופציה זו מגדירה את הצבע של קווי הייחוס האנכיים.
אופן הכתיבה:

/cvref = שם הצבע הרצוי (באנגלית)

13. האופציה chref – אופציה זו מגדירה את הצבע של קווי הייחוס האופקיים.
אופן הכתיבה:

/chref = שם הצבע הרצוי (באנגלית)

14. האופציה ctext – אופציה זו מגדירה את הצבע של תווי הטקסט הנמצאים מחוץ לתרשים (ערכי הצירים והתוויות).
אופן הכתיבה:

/ctext = שם הצבע הרצוי (באנגלית)

15. האופציה font – אופציה זו מגדירה את סוג הפונט של תווי הטקסט הנמצאים מחוץ לתרשים.

אופן הכתיבה:

/font = שם הפונט הרצוי

דוגמא:

```
probplot fg/font = david
```

16. האופציה hminor – אופציה זו מגדירה את מספר השנתות הקטנות שיופיעו בין השנתות הראשיות של הציר האופקי. SAS לא מוסיפה תוויות לשנתות קטנות.
אופן הכתיבה:

/hminor = ערך מספרי

17. האופציה vminor – אופציה זו מגדירה את מספר השנתות הקטנות שיופיעו בין השנתות הראשיות של הציר האנכי. SAS לא מוסיפה תוויות לשנתות קטנות.
אופן הכתיבה:

/vminor = ערך מספרי

18. האופציה square – אופציה זו מציגה את התרשים במסגרת מרובעת. כברירת מחדל, התרשים מוצג במסגרת מלבנית.
אופן הכתיבה:

/square

19. האופציה pctlminor – אופציה זו מוסיפה שנתות משניות לציר x (ציר האחוזונים).
אופן הכתיבה:

/pctlminor

20. האופציה pctlorder – אופציה זו מגדירה את התוויות של השנתות העיקריות בציר x. כברירת מחדל, SAS מציגה את הערכים 1, 5, 10, 25, 50, 75, 90,95 ו-99.
אופן הכתיבה:

/pctlorder = ערכים בסדר עולה (בין 0 ל-100, כולל)

אופציות הקשורות לתרשימים השוואתיים (הנוצרים כאשר מגדירים את ההוראה CLASS):

1. האופציה cframeside – אופציה זו מגדירה את הצבע למילוי המסגרת של תוויות השורות בתרשים השוואתי (כאשר כל שורה מציינת ערך ייחודי של משתנה ה-CLASS). כברירת מחדל, מסגרות אלה לא צבועות.
אופן הכתיבה:

/cframeside = שם הצבע הרצוי

2. האופציה cframetop – אופציה זו מגדירה את הצבע למילוי המסגרת של תוויות העמודות בתרשים השוואתי (רלוונטי כאשר מוגדרים שני משתני CLASS). כברירת מחדל, מסגרות אלה לא צבועות.
אופן הכתיבה:

/cframetop = שם הצבע הרצוי

3. האופציה intertile – אופציה זו מגדירה את המרווח האופקי בין התרשימים ההשוואתיים (באחוזי מסך). כברירת מחדל, אופציה זו מוגדרת ל-0.75.
אופן הכתיבה:

/intertile = ערך מספרי

4. האופציה nrows – מגדיר את מספר השורות בתרשים השוואתי. כברירת מחדל, מוגדר ל-2.
אופן הכתיבה:

/nrows = ערך מספרי

5. האופציה ncols – מגדיר את מספר העמודות בתרשים השוואתי (רלוונטי רק כאשר מגדירים 2 משתני CLASS). כברירת מחדל, מוגדר ל-2.
אופן הכתיבה:

/ncols = ערך מספרי

ההוראה QQPLOT

הוראה זו יוצרת quantile-quantile plots (תרשימי QQ) תוך שימוש בתרשימים באיכות גבוהה, אשר משווים את הנתונים עם רבעונים ספציפיים של התפלגות תיאורטית מסוימת. למעשה, QQPLOT היא טכניקה דומה מאוד ל-probability plots, שבמקרים רבים (מלבד מדגמים קטנים) תספק תוצאות מאוד דומות.

אופן הכתיבה:

QQPLOT משתנה/משתנים / אופציות שונות <(תת אופציות שונות)>;

אופציות של ההוראה QQPLOT

כל האופציות של ההוראה QQPLOT זהות לאופציות של ההוראה PROBPLOT, הן מבחינת הגדרה, והן מבחינת שימוש. לכן לא נחזור על הדיון בהן כאן.

ההוראה INSET

הוראה זו מוסיפה לתרשימים באיכות גבוהה הנוצרים על ידי PROC UNIVARIATE (דהיינו, היסטוגרמה, probability plots ו-qq-plots) מקרא-תרשים (legend). לכן, הוראה זו לא יכול להופיע ללא ההוראות HISTOGRAM, PROBPLOT או QQPLOT. מקרא-התרשים יכול לכלול נתונים סטטיסטים של הנתונים (כגון ממוצע, סטיית תקן), או נתונים על עקומות ההתפלגות.

אופן הכתיבה:

HISTOGRAM|PROBPLOT|QQPLOT/ אופציות שונות <(תת אופציות שונות)>;
INSET מילות מפתח /<אופציות שונות>;

דוגמא:

```
histogram fg /normal (color = yellow);
inset N='N:' (10.0) MEAN = 'Mean:' (4.1)/CFILL=white ;
```

דוגמא זו תפיק את מקרא התרשים המוצג באיור 15.

מילות מפתח

מילות המפתח מגדירות את המידע שיוצג במקרא-התרשים. המידע יוצג בסדר בו הוא מופיע בהוראה. ניתן להגדיר מילות מפתח סטטיסטיות (כגון Mean, Range, Sum), מילות מפתח ראשיות (כגון Exponential, Normal, Weibull, המייצגות עקומת הסתברות מסוימת), ומילות מפתח משניות (כגון Alpha, Sigma, Theta, המייצגות פרמטרים שונים של עקומת ההסתברות המוגדרת).

מילות מפתח ראשוניות מאפשרות להגדיר מילות מפתח משניות (המופיעות בסוגריים מיד לאחר מילת המפתח הראשונית). כאשר מגדירים מילות מפתח ראשוניות ללא מילות מפתח משניות, המקרא-תרשים מציג קו צבעוני ואת שם ההתפלגות כמפתח לקו ההתפלגות המופיע בתרשים (בצבע התואם לקו).

כברירת מחדל, PROC UNIVARIATE מגדירה את הסטטיסטים המוגדרים בהוראה INSET עם תוויות מתאימות. כדי לשנות את התוויות, יש להגדיר את הסטטיסטי המוגדר באמצעות הסימן =, ולאחריו השם המבוקש בתוך גרשיים. בנוסף, כדי לשנות את אופן התצוגה של הערך הסטטיסטי (מספר התווים ומספר הנקודות העשרוניות), יש להגדיר ערך זה בסוגריים אחרי ההגדרה של התווית (כאשר שתי ההגדרות של המספרים מופרדות על ידי נקודה).

לדוגמא:

```
inset N = 'sample Size' STD = 'standard deviation' (5.2);
```

בדוגמא זו, התרשים המוגדר ב-PROC UNIVARIATE יכלול מקרא-התרשים את גודל המדגם וסטיית התקן (עם התווית sample size ו-standard deviation, בהתאמה), כאשר הסטטיסטי סטיית התקן יוצג בפורמט של 5 תווים ו - 2 נקודות עשרוניות אחרי הנקודה.

הערה: סטטיסטים מסויימים של ההוראה INSET לא יהיו זמינים אלא אם הם יוגדרו לחישוב על ידי אחת ההוראות של התרשימים (סטטיסטים אשר לא מחושבים על ידי PROC UNIVARIATE כברירת מחדל).

אופציות של ההוראה INSET

1. האופציה cfill – אופציה זו מגדירה את צבע הרקע של מקרא-התרשים. כברירת מחדל, לא מוגדר רקע למקרא-התרשים.
   אופן הכתיבה:

/cfill = צבע

2. האופציה cfillh – אופציה זו מגדירה את צבע הרקע של הכותרת של מקרא-התרשים. אם משמיטים אופציה זו, צבע הרקע של הכותרת יהיה באותו צבע המוגדר על ידי cfill.
   אופן הכתיבה:

/cfillh = צבע

3. האופציה cframe – אופציה זו מגדירה את צבע המסגרת של מקרא-התרשים.

אופן הכתיבה:

/cframe = צבע

4. האופציה cheader – אופציה זו מגדירה את צבע הטקסט של הכותרת של מקרא-התרשים.
אופן הכתיבה:

/cheader = צבע

5. האופציה ctext – אופציה זו מגדירה את צבע הטקסט של מקרא-התרשים.
אופן הכתיבה:

/ctext = צבע

6. האופציה cshadow – אופציה זו מגדירה את צבע הצללית הנופלת של מקרא-התרשים. כברירת מחדל, מקרא-התרשים לא כולל צללית נופלת.
אופן הכתיבה:

/cshadow = צבע

7. האופציה font – אופציה זו מגדירה את סוג הפונט לתוויות של מקרא-התרשים.
אופן הכתיבה:

/font = סוג הפונט

8. האופציה header – אופציה זו מגדירה את הכותרת של מקרא-התרשים. כברירת מחדל, מקרא-התרשים מוצג ללא כותרת.
אופן הכתיבה:

/header = 'מחרוזת כלשהי'

הערה: הכותרת לא יכולה להיות יותר מ - 40 תוויות.

9. האופציה height – אופציה זו מגדירה את גודל הפונט במקרא-התרשים. כברירת המחדל, גודל הפונט הוא 2.
אופן הכתיבה:

/height = ערך מספרי

10. האופציה noframe – אופציה זו מורידה את המסגרת של מקרא-התרשים.
אופן הכתיבה:

/noframe

11. האופציה position – אופציה זו מגדירה את המיקום על התרשים בו יופיע מקרא-התרשים. כאשר מגדירים את המיקום בתוך התרשים, ניתן לעשות זאת באמצעות כיווני המצפן (צפון, דרום, מזרח מערב), או באמצעות זוג קואורדינאטות על המסך (x, y). כאשר רוצים להגדיר את המיקום מחוץ לתרשים (למשל כאשר מקרא-התרשים מכיל הרבה נתונים), עושים זאת באמצעות מילות מפתח של השוליים).

מיקום תוך שימוש בנקודות ציון של המצפן:
כדי למקם את מקרא-התרשים תוך שימוש בנקודות ציון של המצפן, יש להשתמש האחת ממילות המפתח N, E, S, W (המציינות צפון, מזרח, דרום, מערב, בהתאמה) או שילוב של שני כיוונים כדי לבחור מיקומי ביניים (כגון, NE,

המציין צפון מזרח). מיקומי המצפן על התרשים הם כך שהאמצע העליון של התרשים הוא הצפון, והאמצע התחתון הוא הדרום. כברירת מחדל, מיקום מקרא-התרשים הוא NW (צפון מערב) – הפינה העליונה השמאלית של התרשים.
אופן הכתיבה:

/position = n | ne | e | se | s | sw | w | nw ;

מיקום תוך שימוש בזוג קואורדינאטות:
כדי למקם את מקרא-התרשים באמצעות זוג קואורדינאטות, יש להגדיר שתי נקודות (x, y).
נקודות אלה יכולות לציין נקודות נתונים (בסקאלה של קובץ הנתונים עליו הפרוצדורה עובדת, כך שהפינה השמאלית של מקרא-התרשים תופיע על נקודת ה-x וה-y של הגרף, בהתאם לערכים המוגדרים באופציה), או בנקודות מסך כאשר הקואורדינאטות (0,0) מציינות את הפינה השמאלית התחתונה של התרשים, והקואורדינאטות (100,100) מציינות את הפינה העליונה הימנית של המסך. לכן, כאשר משתמשים בקביעת מיקום באמצעות קואורדינאטות מסך, יש להשתמש במספרים בין 0 ל-100.
אופן הכתיבה:

/position = (x, y) <data>

הערה: האופציה data הכרחית כאשר משתמשים בנתונים בתור קואורדינאטות, והיא תידון בסעיף הבא.

מיקום בשוליים:
כדי למקם את מקרא-התרשים באחד מארבעת מיקומי השוליים המקיפים את אזור התרשים, יש להשתמש במילות המפתח LM, RM, TM, BM (כאשר L מציין left, R מציין right, T מציין top, ו-B מציין bottom).
אופן הכתיבה:

/position = lm | rm | tm | bm ;

12. האופציה data – אופציה זו מגדירה האם קואורדינאטות המיקום מהוות נקודות מסך או נקודות נתונים. כברירת מחדל, הקואורדינאטות מהוות נקודות מסך. לכן, יש להגדיר אופציה זו אם רוצים להשתמש בנקודות נתונים.
אופן הכתיבה:

/position = (x, y) data

13. האופציה refpoint – אופציה זו מגדירה את איזו פינה במסגרת של מקרא-התרשים תמוקם על הקואורדינאטות המוגדרות על ידי ההוראה position = (x, y). כדי להגדיר זאת יש להשתמש במילות המפתח BL (bottom left), BR (bottom right), TL (top left) או TR (top right). כברירת מחדל, הקואורדינאטות מציינות את המיקום של הפינה השמאלית התחתונה של מסגרת מקרא-התרשים (bl).
אופן הכתיבה:

/position = (x, y) refpoint = bl | br | tl | tr

ההוראה OUTPUT

הוראה זו שומרת נתונים סטטיסטיים ומשתנים המוגדרים על ידי ההוראה BY בקובץ נתונים חיצוני. ניתן להגדיר מספר הוראות OUTPUT תחת אותה פרוצדורה בכדי ליצור מספר קבצי נתונים חדשים במקביל.

אופן הכתיבה:

OUTPUT <out = שם קובץ נתונים> שם(ות) = מילת מפתח סטטיסטית1
<שם(ות) = מילת מפתח סטטיסטיתn>
<הגדרות אחוזונים>;

דוגמא:

```
proc univariate noprint;
 var test1 test2;
 output out=unidata min = min_p max = max_p;
run;
```

דוגמא זו יוצרת קובץ נתונים חדש בשם unidata, הכולל שני משתנים: המשתנה min_p, המכיל את הערך המינימאלי בקרב המשתנים test1 ו-test2, והמשתנה max_p, המכיל את הערך המקסימאלי בקרב המשתנים test1 ו-test2.

סטטיסטים להפקה לקובץ פלט ב-PROC UNIVARIATE	
סטטיסטיקה תיאורית	
מילת מפתח	**תיאור**
CSS	סכום הפרשי הריבועים
CV	מקדם השונות
KURTOSIS	מידת השטיחות של ההתפלגות
MAX	הערך המקסימאלי
MIN	הערך המינימאלי
MODE	השכיח
RANGE	הטווח
NMISS	מספר הערכים החסרים
NOBS	מספר התצפיות
STDMEAN	סטית התקן של הממוצע
SKEWNESS	מידת הסימטריות של ההתפלגות
STD	סטיית התקן
USS	סכום הריבועים
SUM	סכום התצפיות
SUMWGT	סכום משוקלל
VAR	שונות
אחוזונים	
מילת מפתח	**תיאור**
MEDIAN	חציון
P1, P5, P10, P90, P95, P99	האחוזון ה-*n* י
Q1, Q3	הרבעון הראשון והשלישי, בהתאמה
QRANGE	הטווח הבין רבעוני
סטטיסטיקה רובסטית	
מילת מפתח	**תיאור**
GINI	מדד GINI
MAD	הבדל אבסולוטי בין החציונים

QN	השתנות (סטייה) מרמה שנייה של ההבדל בין החציונים
SN	השתנות (סטייה) מרמה ראשונה של ההבדל בין החציונים
STD_GINI	סטיית התקן של GINI
STD_MAD	סטיית תקן של ההבדל האבסולוטי בין החציונים
STD_QN	סטיית התקן של QN
STD_QRANGE	סטיית התקן של הטווח הבין רבעוני
STD_SN	סטיית התקן של SN
בדיקת השערות	
מילת מפתח	**תיאור**
NORMAL	בדיקת הנחת הנורמליות
PROBN	ההסתברות שהנתונים הגיעו מהתפלגות נורמלית
MSIGN	מבחן הסימן (Sign test)
PROBM	הסתברות לערך אבסולטי גבוה יותר במבחן הסימן
SIGNRANK	מבחן signed rank
PROBS	הסתברות לערך אבסולטי גבוה יותר במבחן signed rank
T	מבחן T
PROBT	רמת מובהקות דו זנבית למבחן t

טבלה 8– רשימת סטטיסטים הניתנים להפקה לקובץ פלט על ידי ההוראה OUTPUT ב-PROC UNIVARIATE

אופציות של ההוראה OUTPUT

1. האופציה out – אופציה זו מגדירה את שם קובץ הנתונים שנוצר על ידי PROC MEANS ושמכיל את הסטטיסטים שהוגדרו על ידי ההוראה.
אופן הכתיבה:

שם קובץ נתונים חדש = OUTPUT out

דוגמא:

```
output out=dogma;
```

אם לא מגדירים את שם קובץ הנתונים החדש, SAS תתן לו אוטומאטית את השם data*n*, כאשר n הוא המספר הקטן ביותר שהופך את השם לייחודי (n = 1 במצב בו לא קיימים קבצים בשם data בזיכרון של התוכנה).

2. מילות מפתח סטטיסטיות – באמצעות מילות מפתח סטטיסטיות (המפורטות בטבלה 8), ניתן לבקש מ-SAS להפיק לקובץ הפלט ערכים סטטיסטיים של משתנים, כגון ערך מינימאלי, ערך מקסימאלי, ממוצע וכדומה.
אופן הכתיבה:

שם משתנה = מילת מפתח סטטיסטית

דוגמא:

```
output out=unidata mean = memotza
```

בדוגמא זו, SAS תפיק לקובץ הנתונים החדש את המשתנה memotza, הכולל את ממוצע הערכים של המשתנים המוגדרים בפרוצדורה. ניתן להגדיר מילות מפתח סטטיסטיות נוספות באותה הוראה, על ידי הפרדתן באמצעות רווחים.

דוגמא:

```
output out=unidata mean = memotza range = range_p mode = mode_p;
```

הגדרת אחוזונים

ניתן להגדיר בהוראה OUTPUT אחוזון אחד או יותר לאחסון בקובץ הנתונים המופק על ידי ההוראה, וכן להגדיר את השם של המשתנה (משתנים) המכיל את האחוזון.

אופן הכתיבה:

PCTLPTS = אחוזון או אחוזונים הרצויים PCTLPRE = תחילית של שם(ות) PCTLNAME = סיומת של שם(ות);

ההוראה PCTLPTS – הוראה זו מגדירה אחוזון אחד או יותר לחישוב. ניתן להגדיר מספר אחוזונים על ידי כתיבת האחוזונים הרצויים, מופרדים על ידי רווחים, או על ידי הגדרת *ערך התחלתי* עד *לערך סופי בצעדים של*, כפי שמודגם להלן:

מספר התחלתי (גדול מ - 0) TO *מספר סופי (קטן מ - 100)* BY *מספר המציין את גודל הצעד (גודל ההתקדמות)*

דוגמא:

```
output pctlpts = 50, 85 to 90 by 2.5;
```

דוגמא זו מבקשת לחשב את האחוזון ה-50, ואת האחוזונים ה-85, 87.5 ו-90.

ההוראה PCTLPRE – הוראה זו מגדירה את התחילית ליצירת שם המשתנים המכילים את האחוזונים המוגדרים על ידי ההוראה PCTLPTS. כדי לשמור אחוזונים ליותר ממשתנה אחד, יש ליצור מספר תחיליות, בהתאם לסדר המשתנים המוגדרים בהוראה VAR (שתידון להלן).

לדוגמא:

```
output pctlpre =P_ pctlpts = 50, 85 to 90 by 2.5;
```

בדוגמא זו, האחוזון ה-50 יקבל את השם P_50, ושאר האחוזונים המוגדרים יקבלו את השם P_85, P_87.5, P_90, בהתאמה.

ההוראה PCTLNAME – הוראה זו מגדירה סיומת לשמות המשתנים המכילים את האחוזונים המוגדרים על ידי PCTLPTS. בניגוד להוראה PCTLPRE, בהוראה PCTLNAME יש להגדיר סיומות כמספר האחוזונים המוגדרים על ידי ההוראה PCTLPTS, כאשר הסיומות מופרדות על ידי רווחים. חוץ מהבדל זה, מגדירים את הסיומות בדיוק כמו שמגדירים את תחילית השם.

לדוגמא:

```
output out=unidata pctlpts=10 50 90 pctlpre=Var1_ Var2_
       pctlname=P10 P50 P90;
```

בדוגמא זו, האחוזון ה-10 של Var1 יקבל את השם Var1_P10 ושל Var2 יקבל את השם Var2_10. האחוזון ה-50 של Var1 יקבל את השם Var1_P50 ושל Var2 יקבל את השם Var2_50. לבסוף, האחוזון ה-90 של Var1 יקבל את השם Var1_P90 ושל Var2 יקבל את השם Var2_90.

ההוראה VAR

הוראה זו מגדירה את המשתנים עליהם תבצע PROC UNIVARIATE את הניתוחים, כמו גם את הסדר בו התוצאות של כל משתנה מוגדר יופיעו בקובץ הפלט. אם משמיטים הוראה זו, PROC UNIVARIATE תנתח את כל המשתנים הנומריים שבקובץ הנתונים, למעט אלה המופיעים בהוראות אחרות של הפונקציה.

אופן הכתיבה:

```
VAR רשימת משתנים;
```

PROC FREQ

הפרוצדורה FREQ היא גם פרוצדורה לסטטיסטיקה תיאורית, וגם לבחינת השערות. ספציפית, הפרוצדורה מחשבת cross tabs (המסכמות את הנתונים לשני משתנים או יותר, על ידי הצגת מספר התצפיות לכל קומבינציה של ערכי המשתנים) וטבלאות שכיחויות n ממדיות לנתונים, כמו גם מבחנים סטטיסטים א-פרמטריים (כגון χ^2) ומדדים לקשר בין משתנים.

כברירת מחדל, PROC FREQ מחשבת לכל משתנה בקובץ הנתונים טבלת שכיחויות. טבלה זו מכילה שכיחות, אחוזים, שכיחות מצטברת ואחוזים מצטברים. כמו כן, PROC FREQ מפיקה קובץ פלט באופן אוטומטי.

לדוגמא, הרצה בסיסית של PROC FREQ לקובץ נתונים המכיל את המשתנים גיל ומין תפיק את הפלט שלהלן בחלון Output:

The SAS System
The FREQ Procedure

age	Frequency	Percent	Cumulative Frequency	Cumulative Percent
21	1	4.55	1	4.55
22	1	4.55	2	9.09
23	2	9.09	4	18.18
24	2	9.09	6	27.27
25	4	18.18	10	45.45
26	3	13.64	13	59.09
27	3	13.64	16	72.73
28	1	4.55	17	77.27
29	2	9.09	19	86.36
30	1	4.55	20	90.91

gender	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	12	54.55	12	54.55
1	10	45.45	22	100.00

אופן הכתיבה:

```
PROC FREQ <אופציות שונות>;
BY <descending> 1 משתנה < descending> n משתנה <notsorted>;
WEIGHT רשימת משתנים;
OUTPUT מילות מפתח סטטיסטיות <OUT = שם קובץ נתונים>;
TABLES הגדרת טבלאות/<אופציות שונות>;
RUN;
```

דוגמא:

```
proc freq;
run;
```

אופציות של PROC FREQ

1. האופציה noprint – אופציה זו אומרת ל-PROC FREQ לא להציג את הטבלאות המחושבות (או כל סטטיסטי אחר שהוגדר) בקובץ הפלט.
אופן הכתיבה:

noprint

4. האופציה formchar – אופציה זו מגדירה את התווים בהם PROC FREQ משתמשת להרכבת קווי המתאר והחוצצים של התאים בטבלאות crosstabs. כברירת מחדל, התו "**f**" מרכיב את הקווים אופקיים של התא, התו "," מרכיב את התווים האנכיים של התא, והתו "^" מרכיב את החוצצים של התאים. עם זאת, ברירת המחדל עשויה להשתנות מגירסת SAS אחת לשנייה.
אופן הכתיבה:

formchar <(מיקום/ים – מופרדים על ידי פסיקים)> = 'התווים הרצויים';

מיקומים הם מספרים המציינים את הקו אותו התוו מגדיר, כאשר:
1 – קווים אנכיים
2 – קווים אופקיים
7 - חוצצים

כאשר משמיטים את הגדרת המיקום, SAS תייחס את התו הראשון ל-1, את התו השני ל-2, ואת התו השלישי ל-7. לעומת זאת, כאשר מגדירים רק חלק מהמיקומים, SAS תשתמש בתווי ברירת המחדל למיקומים שלא הוגדרו.

את התווים הרצויים כותבים בתוך גרשיים לאחר סימן השווה, ללא רווח, שכן רווח מהווה תו גם. לכן, אם רוצים טבלה ללא גבולות בכלל, יש לשים בתוך הגרשיים שלושה סימני רווח (' ').

דוגמא:

```
proc freq formchar (1,2,7) = '|-+';
```

דוגמא זו תיצור טבלה שבה התווים של הקווים האנכיים הם |, התווים של הקווים האופקיים הם -, והוחצצים הם +.

הערה: אופציה זו רלוונטית רק כאשר מגדירים את ההוראה TABLES.

ההוראה BY

הוראה זו אומרת לפרוצדורה לחשב ולהציג טבלאות וסטטיסטים בנפרד לכל קבוצת BY. הוראה זו מפיקה פלט נפרד לכל קבוצה המוגדרת על ידי משתנה ה-BY.

אופן הכתיבה:

BY <decending> משתנה 1 <descending> משתנה n <notsorted>;

ההוראה WEIGHT

הוראה זו אומרת ל-PROC FREQ כי ערכי המשתנים המוגדרים בהוראה מייצגים שכיחויות של משתנים ולא ערכים אבסולטיים. לכן, כאשר מגדירים משתנים על ידי ההוראה WEIGHT, PROC FREQ מתייחסת לערכי משתנה זה כאל השכיחויות של התצפיות השונות של המשתנים הקיימים בקובץ הנתונים.

אופן הכתיבה:

WEIGHT רשימת משתנים;

דוגמא:

```
proc freq;
 weight var1;
 table var2*var3;
run;
```

לכן, בדוגמא הנוכחית PROC FREQ תתייחסת לכל תצפית במשתנים var2 ו-var3 כאילו היא מופיעה X פעמים, בהנחה ש-X מייצג את ערך התצפית של המשתנה var1.

ההוראה TABLES

ההוראה TABLES מאפשרת להפיק טבלאות שכיחויות ו-crosstabs חד ממדיות או n ממדיות, ומחשבת את הסטטיסטים לטבלאות אלה. כברירת מחדל (כאשר הוראה זו לא מוגדרת), PROC FREQ מחשבת טבלת שכיחויות חד ממדית לכל המשתנים בקובץ הנתונים, שאינם מוגדרים בהוראות אחרות.

אופן הכתיבה:

TABLES בקשות לטבלאות/<אופציות שונות>;

בקשות לטבלאות מגדירות את הטבלאות crosstabs וטבלאות השכיחויות ש-PROC FREQ תפיק.

בקשה מורכבת משם אחד (לבקשת טבלה חד ממדית) או ממספר שמות (לבקשת טבלה n ממדית) של משתנים, מופרדים על ידי כוכביות (*). הערך הייחודי של כל משתנה מוגדר יוצר את העמודות, השורות והשכבות של הטבלה (בהתאם למספר הממדים והערכים), לפי הפירוט הבא:

שורות – הערכים של המשתנה הלפני אחרון
עמודות – הערכים של המשתנה האחרון
שכבות – הערכים של המשתנים המוגדרים מהמשתנה הראשון עד אחד לפני אחרון

קוד	שווה ערך ל-
TABLES a*(b c);	הטבלאות a * b ו-a * c
TABLES (a b) * (c d);	הטבלאות a * c, a * d, b * c, b * d
TABLES a--c;	הטבלאות a, b, c

טבלה 9 – קודי קיבוץ להפקת טבלאות ב-PROC FREQ

לכל שכבה, PROC FREQ יוצרת טבלה נפרדת.

ניתן להגדיר מספר הוראות TABLES בצעד PROC FREQ אחד, או להגדיר מספר טבלאות (חד או n ממדיות) בהוראת TABLES אחת. בקשת מספר טבלאות בהוראה אחת נעשית באמצעות קוד קיבוץ, כפי שמודגם בטבלה 9.

כברירת מחדל, אם מגדירים טבלה חד ממדית בהוראה TABLES, יתקבל פלט זהה לפלט הבסיסי של PROC FREQ ללא הגדרת ההוראה. לעומת זאת, כאשר מגדירים טבלה n ממדית בסיסית, הפלט המתקבל נראה כדלקמן:

```
                The FREQ Procedure
              Table of age by gender

age       gender

Frequency|
Percent  |
Row Pct  |
Col Pct  |       0|       1|  Total
---------+--------+--------+
      26 |      3 |      0 |      3
         |  13.64 |   0.00 |  13.64
         | 100.00 |   0.00 |
         |  25.00 |   0.00 |
---------+--------+--------+
      27 |      3 |      2 |      5
         |  13.64 |   9.09 |  22.73
         |  60.00 |  40.00 |
         |  25.00 |  20.00 |
---------+--------+--------+
      29 |      1 |      0 |      1
         |   4.55 |   0.00 |   4.55
         | 100.00 |   0.00 |
         |   8.33 |   0.00 |
---------+--------+--------+
      32 |      1 |      0 |      1
         |   4.55 |   0.00 |   4.55
         | 100.00 |   0.00 |
         |   8.33 |   0.00 |
---------+--------+--------+
Total          12       10       22
              54.55    45.45   100.00
```

אופציות של ההוראה TABLES

1. האופציה alpha – אופציה זו מגדירה את רמת הביטחון (confidence level) לרווח הסמך ולמבחני המובהקות הסטטיסטיים. רמת הביטחון מוגדרת כ-100 * (1 – alpha), כך שאם למשל נגדיר alpha = 0.05, נקבל רווח סמך של 95%. כברירת מחדל, הערך של alpha נקבע ל-0.05.
אופן הכתיבה:

/alpha = ערך מספרי בין 0 ל-1

2. האופציה cl – אופציה זו מבקשת מ-PROC FREQ להפיק רווח סמך לסטטיסטיים שמוגדרים על ידי האופציה measures (ראה תת-סעיף 7 להלן). כברירת מחדל, האופציה cl מגדירה אוטומטית את ההוראה measures, גם אם הוראה זו לא הוגדרה על ידי המשתמש. כמו כן, גבולות רווח הסמך מוגדרות על ידי ההוראה alpha (או 95% כברירת מחדל).
אופן הכתיבה:

/cl

3. האופציה comcol – אופציה זו אומרת ל-PROC FREQ להציג עמודה של אחוז מצטבר בטבלה המופקת באמצעות ההוראה TABLES.

אופן הכתיבה:

/cumcol

4. האופציה deviation – אופציה זו אומרת ל-PROC FREQ להציג את הסטיות של שכיחויות התאים מהשכיחות המצופה (מה היינו מצפים ששכיחות המשתנים תיהיה בהתאם להשערה האפס) של תאי הטבלה המופקת באמצעות ההוראה TABLES.
אופן הכתיבה:

/deviation

5. האופציה expected – אופציה זו אומרת ל-PROC FREQ להציג את שכיחויות התאים המצופות תחת הנחת אי תלות (תחת השערת אפס שאין קשר בין המשתנים) בטבלה המופקת באמצעות ההוראה TABLES.
אופן הכתיבה:

/expected

6. האופציה list – אופציה זו אומרת ל-PROC FREQ להציג טבלה סטנדרטית במקום הטבלה crosstabulation (crosstabs) המופקת כברירת מחדל באמצעות ההוראה TABLES.
אופן הכתיבה:

/list

כאשר מגדירים את האופציה list, PROC FREQ תפיק את הפלט הבא:

The FREQ Procedure

Var1	Var2	Frequency	Percent	Cumulative Frequency	Cumulative Percent
51	65	1	8.33	1	8.33
63	75	1	8.33	2	16.67
71	77	1	8.33	3	25.00
75	70	1	8.33	4	33.33
75	78	1	8.33	5	41.67
79	76	1	8.33	6	50.00
80	75	1	8.33	7	58.33
87	73	1	8.33	8	66.67
89	82	1	8.33	9	75.00
92	55	1	8.33	10	83.33
94	91	1	8.33	11	91.67
95	97	1	8.33	12	100.00

7. האופציה measures – אופציה זו אומרת ל-PROC FREQ להציג בנוסף לטבלה גם את הסטטיסטיים gamma, Kendall's tau-b, Stuart's tau-c, Somers' D, Pearson and Spearman correlation coefficients, lambda, uncertainty coefficient, ובמקרה של טבלה 2 X 2, גם את ה-odd ratio, relative risk ורווחי סמך של הסטטיסטיים. בנוסף, האופציה מפיקה לכל סטטיסטי גם את ה-asymptotic standard error (ASE) שלו.
אופן הכתיבה:

/measures

כאשר מגדירים את האופציה measures, PROC FREQ מוסיפה לפלט גם את הטבלה הבאה:

The FREQ Procedure
Statistics for Table of Var1 by Var2

Statistic	Value	ASE
Gamma	0.3438	0.2829
Kendall's Tau-b	0.3385	0.2791
Stuart's Tau-c	0.3361	0.2778
Somers' D C|R	0.3385	0.2795
Somers' D R|C	0.3385	0.2789
Pearson Correlation	0.4086	0.3068
Spearman Correlation	0.4316	0.3234
Lambda Asymmetric C|R	0.9000	0.0949
Lambda Asymmetric R|C	0.9000	0.0949
Lambda Symmetric	0.9000	0.0700
Uncertainty Coefficient C|R	0.9512	0.0312
Uncertainty Coefficient R|C	0.9512	0.0312
Uncertainty Coefficient Symmetric	0.9512	0.0209

Sample Size = 12

8. האופציה missing – אופציה זו אומרת ל-PROC FREQ להתייחס לערכים חסרים כערכים לא חסרים, ולכלול אותם בחישוב של השכיחויות ושאר הסטטיסטיים.
אופן הכתיבה:

/missing

9. האופציה misprint – אופציה זו אומרת ל-PROC FREQ להציג את השכיחות של הערכים החסרים של כל תא בטבלה, אבל לא להתחשב בהם בחישוב של הסטטיסטיים.
אופן הכתיבה:

/missprint

10. האופציה noprint – אופציה זו אומרת ל-PROC FREQ לא להפיק טבלת crosstabs או טבלת שכיחויות, אך כן להציג את הסטטיסטיים שהוגדרו על ידי המשתמש.
אופן הכתיבה:

/noprint

11. האופציה chisq – אופציה זו מחשבת מבחן X^2 (חי בריבוע) להומוגניות או אי תלות עבור טבלת 2 X 2. עבור טבלת one-way, האופציה מחשבת מדד של טיב התאמה לשיוויון שכיחויות.
אופן הכתיבה:

/chisq

תוצאות מבחן ה - X^2 מופקות לפלט כדלקמן:

Statistics for Table of var1 by var2

Statistic	DF	Value	Prob
Chi-Square	1	9.9429	0.0016
Likelihood Ratio Chi-Square	1	10.2501	0.0014
Continuity Adj. Chi-Square	1	8.3548	0.0038
Mantel-Haenszel Chi-Square	1	9.7714	0.0018
Phi Coefficient		-0.4140	
Contingency Coefficient		0.3825	
Cramer's V		-0.4140	

```
        Fisher's Exact Test
_____________________________________
Cell (1,1) Frequency (F)          8
Left-sided Pr <= F           0.0017
Right-sided Pr >= F          0.9997

Table Probability (P)        0.0015
Two-sided Pr <= P            0.0035

         Sample Size = 58
```

12. האופציה testf – במצב שבו מעוניינים לבחון את קיומן של שכיחויות ספציפיות אופציה זו מאפשרת להגדיר את השכיחויות להשערת האפס עבור מבחן חי-בריבוע חד-כיווני,. שכיחויות אלה למעשה מגדירות את השכיחות המצופה של כל תא בטבלה (כל ערך של המשתנה), ואשר אליהן מושוות השכיחות הנצפית במבחן חי-בריבוע.
אופן הכתיבה:

/testf = (ערכים)

דוגמא:

```
/testf = (10 48);
```

סכום הערכים של השכיחויות השונות חייב להיות שווה לשכיחות הכללית של האיברים בכל השורה (דהיינו למספר התצפיות). אחרת, תתקבל בחלון Log הודעת השגיאה הבאה והאופציה לא תתבצע:

ERROR: The sum of the TESTF frequencies must equal the total frequency, for the table of var1.

בנוסף, מספר הערכים השונים שיוגדר חייב להיות שווה למספר הקטגוריות של הערכים הקיימים למשתנה. סדר הערכים צריך להיות תואם לסדר הופעתם בטבלה. באם לא יוגדרו ערכים כמספר הקטגוריות, תתקבל בחלון Log הודעת השגיאה הבאה, והאופציה לא תתבצע:

ERROR: The number of TESTF values must equal the number of table cells, for the table of var1.

הערה: את הערכים השונים ניתן לכתוב בתוך הסוגריים מופרדים על ידי רווחים או על ידי פסיקים.

13. האופציה testp – במצב שבו מעוניינים לבחון פרופורציות ספציפיות, אופציה זו מגדירה את הפרופורציות להשערת האפס עבור מבחן one-way X^2. פרופורציות אלה מגדירות למעשה את הפרופורציה המצופה של כל תא בטבלה, ואשר אליה מושוות הפרופורציה הנצפית.
אופן הכתיבה:

/testp = (ערכים)

דוגמא:

```
/testp = (0.1 0.2 0.7);
```

הערכים של הפרופורציות חייבים לנוע בין 0 ל-1, והסכום שלהם חייב להיות שווה ל-1. אחרת, תתקבל בחלון Log הודעת השגיאה הבאה והאופציה לא תתבצע:

ERROR: The sum of the TESTP probabilities must equal one.

בנוסף, מספר הערכים השונים שיוגדר חייב להיות שווה למספר הקטגוריות של הערכים הקיימים למשתנה. סדר הערכים צריך להיות תואם לסדר הופעתם בטבלה. באם לא יוגדרו ערכים כמספר הקטגוריות, תתקבל בחלון Log הודעת השגיאה הבאה, והאופציה לא תתבצע:

```
ERROR: The number of TESTP values must equal the number of table cells, for the
table of var1.
```

הערה: את הערכים השונים ניתן לכתוב בתוך הסוגריים מופרדים על ידי רווחים או על ידי פסיקים.

תרגול עצמי – פרוצדורות סטטיסטיות I – סטטיסטיקה תיאורית

תרגיל 25

נתון קובץ נתונים המכיל בחירות של 14 נבדקים, הכולל נתונים על מין (gen), תנאי ניסויי (cond), רצף הבחירה הגדול ביותר (max) וכמות בחירות בכל אחת מ-4 חפיסות של קלפים (A עד D):

sub	gen	cond	max	A	B	C	D
1	M	1	10	27	32	16	25
2	F	1	30	39	14	7	40
3	F	1	5	16	40	15	29
4	M	1	11	16	40	17	27
5	M	1	15	33	40	23	4
6	M	1	21	15	40	13	32
7	F	1	26	40	40	12	8
8	M	2	7	26	21	19	34
9	M	2	7	31	30	16	23
10	M	2	3	20	23	28	29
11	M	2	17	16	27	17	40
12	M	2	12	25	40	20	15
13	F	2	26	10	40	10	40
14	F	2	9	9	47	10	34

א. ידוע כי כמות הבחירות מכל אחת מהחפיסות יכול לנוע בין 0 בחירות ל-50 בחירות. כיצד ניתן באמצעות PROC MEANS לבדוק אם יש טעויות קידוד הנתונים של הבחירות בכל אחת מארבעת החפיסות (כיצד ניתן לבדוק אם נעשתה טעות בהקלדת התוצאות)?

ב. כתוב קוד SAS להפקת פלט של ממוצע וסטיית תקן עבור הבחירות בכל אחת מארבעת החפיסות. הגדר את הפלט כך שיהיו רק שתי ספרות אחרי הנקודה.

ג. כתוב קוד SAS להפקת פלט דומה לזה שעשית בסעיף ב', אך בנפרד עבור כל אחת משתי קבוצות הניסוי.

תרגיל 26

נתון קובץ נתונים, המכיל את הבחירות שעשו 10 נבדקים בין שתי אלטרנטיבות ב-10 סיבובים:

```
1 0 1 0 0 0 1 0 1 1
1 0 1 0 0 1 1 1 1 1
0 1 0 1 1 0 0 1 1 1
1 0 0 0 1 1 0 1 1 0
0 1 1 0 1 1 1 1 1 1
0 0 1 1 0 1 0 1 1 1
```

1 0 1 1 0 0 1 1 0 1
1 1 0 1 0 0 1 0 1 1
0 1 0 1 0 1 1 1 0 1
1 1 1 0 0 1 0 1 1 1

כתוב קוד SAS שיחשב את אחוז הבחירות באלטרנטיבה המיוצגת על ידי התצפית "1" מעבר לכל הנבדקים. הקוד לא יפיק קובץ Output, אלא ישמור את נתוני הממוצעים בקובץ נתונים חדש. בסופו של תהליך, קובץ הנתונים החדש צריך להכיל עמודה של מספר סיבוב ועמודה של פרופורציית בחירות בלבד.

תרגיל 27

נתון קובץ נתונים המכיל ציונים של 20 סטודנטים בקורס פסיכולוגיה חינוכית בשני בחנים ומבחן סיום, ואת מין הסטודנט:

gender quiz1 quiz2 test
0 75 82 67
0 80 95 72
1 95 92 90
0 65 70 62
1 90 88 70
1 82 75 92
1 90 92 95
0 69 79 83
1 72 80 67
0 76 92 83
0 77 80 96
1 90 89 95
0 100 95 100
1 95 100 100
0 88 76 83
0 67 83 88
0 90 83 77
1 90 80 77
1 100 85 89
0 66 70 88

צור קובץ SAS המכיל את המשתנים ציון סופי (שהוא ממוצע של כל ציוני שני הבחנים ומבחן הסיום) ומין הסטודנט, והצג את הציון הסופי בטבלת שכיחויות לפי מין (מין בשורות וציון סופי בעמודות). גבולות הטבלה צריכים להיות קווים ישרים (אנכים ואופקיים), ולא הגבולות המופיעים בטבלה כברירת מחדל.

תרגיל 28

השתמש בקובץ הנתונים שנוצר בתרגיל 27, והצג את הציון סופי בקורס בהיסטוגרמה. צור היסטוגרמה שונה לנשים וגברים (היסטוגרמה השוואתית), והוסף לכל היסטוגרמה קו עקומת התפלגות נורמלית. כדי שאופן התצוגה יהיה ברור, צור פורמאט שיגדיר את הערך 0 כ-male ואת הערך 1 כ-female, וקשר אותו למשתנה gender. כמו כן, צור את עמודות ההיסטוגרמה בצבע כחול, את הקו המקיף אותן בצבע צהוב (עובי 3 פיקסלים). את קו העקומה הנורמלית צבע בצבע אדום (עובי 5 פיקסלים). לבסוף, צור להיסטוגרמות מקרא-תרשים שיכלול ממוצע וסטיית תקן.

פרק 10
פרוצדורות סטטיסטיות II:
קשר בין משתנים

PROC CORR

הפרוצדורה CORR הינה פרוצדורה סטטיסטית לחישוב מתאמים בין משתנים ואת רמת המובהקות של מתאמים אלה.
מקדמי המתאם המחושבים על ידי הפרוצדורה הם:

1. מקדם המתאם של פירסון
2. מקדם המתאם של ספירמן
3. Kendall's tau-b
4. מדד Hoeffding לחישוב תלות (D)
5. מתאמים חלקיים לפירסון, ספירמן וקנדל
6. מקדם המתאם אלפא של קרונבאך (לחישוב מהימנות)

אופן הכתיבה:

PROC CORR אופציות שונות;
BY <descending> משתנה 1 … <descending> משתנה n <notsorted>;
VAR רשימת משתנים;
PARTIAL רשימת משתנים;
WITH רשימת משתנים;
RUN;

דוגמא:

```
proc corr;
 var con1 con2 con2;
run;
```

כברירת מחדל, הפרוצדורה מחזירה חישוב של סטטיסטים תיאוריים (ממוצע, סטיית תקן, סכום, מינימום ומקסימום) ומטריצת מתאמים (כולל רמת מובהקות):

```
                                The SAS System
                              The CORR Procedure

                    3  Variables:    con1     con2     con3

                                Simple Statistics

Variable            N          Mean       Std Dev           Sum       Minimum       Maximum

con1                9      20.66667       5.76628     186.00000      13.00000      25.00000
con2                9      30.00000       2.59808     270.00000      27.00000      33.00000
con3                9      15.66667       4.92443     141.00000      11.00000      22.00000
```

```
Pearson Correlation Coefficients, N = 9
      Prob > |r| under H0: Rho=0
           con1          con2          con3

con1      1.00000       0.82603       0.18049
                        0.0061        0.6422

con2      0.82603       1.00000       0.70345
           0.0061                     0.0345

con3      0.18049       0.70345       1.00000
           0.6422        0.0345
```

אופציות של PROC CORR

אופציות כלליות:

1. האופציה cov – אופציה זו אומרת ל-PROC CORR לחשב שונויות משותפות למשתנים ולהציג אותן.
אופן הכתיבה:

cov

2. האופציה vardef – אופציה זו מגדירה את המכנה בו יש להשתמש כדי לחשב את השונות המשותפת, השונות וסטיית התקן. המכנה יכול להיות:
   א. DF – מספר דרגות החופש (מוגדר כ-$n - k - 1$, כאשר k מספר המשתנים המוגדרים על ידי ההוראה PARTIAL).
   ב. N – מספר התצפיות.
   ג. WDF – סכום המשקולות פחות 1.
   ד. סכום המשקולות.

   כברירת מחדל, PROC CORR משתמשת ב-DF.
   אופן הכתיבה:

vardef = מכנה

אופציות הקשורות להגדרת סטטיסטיים לחישוב:

1. האופציה hoeffding – אופציה זו אומרת ל-PROC CORR לחשב ולהציג את המדד Hoeffding D.
אופן הכתיבה:

hoeffding

הערה: כאשר מגדירים את ההוראות WEIGHT ו-PARTIAL, אופציה זו אינה תקפה.

2. האופציה kendall – אופציה זו אומרת ל-PROC CORR לחשב ולהציג את מקדם המתאם Kendall tau-b בהתבסס על מספר הזוגות של התצפיות התואמות והלא-תואמות.
אופן הכתיבה:

kendall

הערה: כאשר משתמשים בהוראה WEIGHT אופציה זו אינה תקפה.

3. האופציה pearson – אופציה זו אומרת ל-PROC CORR לחשב ולהציג גם את מקדם המתאם של פירסון כאשר מגדירים את האופציות hoeffding, kendall, spearman.
אופן הכתיבה:

pearson

4. האופציה spearman – אופציה זו אומרת ל-PROC CORR לחשב ולהציג את מקדם המתאם של ספירמן.
אופן הכתיבה:

spearman

הערה: כאשר משתמשים בהוראה WEIGHT אופציה זו אינה תקפה.

5. האופציה alpha – אופציה זו אומרת לפרוצדורה לחשב את מקדם המתאם אלפא של קרונבאך לחישוב מהימנות. עבור כל משתנה המוגדר על ידי ההוראה VAR, הפרוצדורה תחשב את המתאם בין המשתנה וסך כל המשתנים הנותרים.
אופן הכתיבה:

alpha

הערה: כאשר משתמשים בהוראה WITH אופציה זו אינה תקפה. בנוסף, כאשר מגדירים אופציה זו, האופציה pearson מוגדרת אוטומטית.

ההוראה BY

הוראה זו אומרת ל-PROC CORR לחשב מתאמים לכל קבוצה המוגדרת על ידי ההוראה באופן נפרד. יש למיין את קובץ הנתונים על פי המשתנה או המשתנים המוגדרים על ידי ההוראה BY (אלא אם משתמשים באופציה notsorted).

אופן הכתיבה:

BY <descending> משתנה 1 … <descending> משתנה n <notsorted>;

דוגמא:

```
proc corr;
 by gender;
run;
```

ההוראה VAR

הוראה זו מגדירה ל-PROC CORR את המשתנים עליהם יש לבצע את הניתוחים הסטטיסטים. אם לא כוללים הוראה זו, הפרוצדורה תחשב מקדמי מתאם לכל המשתנים הנומריים הקיימים בקובץ הנתונים שלא מוגדרים באף אחת מההוראות.

אופן הכתיבה:

VAR שמות משתנים;

ההוראה PARTIAL

הוראה זו אומרת ל-PROC CORR לחשב מקדם מתאם חלקי לסטטיסטיים פירסון, ספירמן או Kendall tau-b.

אופן הכתיבה:

PARTIAL שמות של משתנים;

דוגמא:

```
proc corr;
 var test_grade final_grade;
 partial gender;
run;
```

ההוראה WITH

הוראה זו קובעת את המשתנים שביניהם לבין המשתנים המוגדרים על ידי ההוראה VAR תחשב PROC CORR מתאמים. במצב כזה מוודאים שהפרוצדורה תחשב מתאמים רק לקומבינציות ספציפיות של משתנים.

אופן הכתיבה:

WITH רשימת משתנים;

דוגמא:

```
proc corr;
 var con1 con2 con3;
 with con4 con5 con6;
run;
```

בדוגמא זו, PROC CORR תחשב המתאמים לצרופים הבאים של המשתנים:

con4 X con1 con4 X con2 con4 X con3
con5 X con1 con5 X con2 con5 X con3
con6 X con1 con6 X con2 con6 X con3

כאשר משתמשים בהוראה WITH, המשתנים המוגדרים על ידי ההוראה מייצגים את השורות של מטריצת המתאמים, בעוד שהמשתנים המוגדרים על ידי ההוראה VAR (או כל שאר המשתנים בקובץ הנתונים, במקרה בו לא משתנים בהוראה VAR) מייצגים את העמודות של המטריצה:

```
      Pearson Correlation Coefficients, N = 9
            Prob > |r| under H0: Rho=0

                 con1          con2          con3

con4          -0.56362       0.00000       0.71074
               0.1140        1.0000        0.0319

con5           0.36149       0.82411       0.98231
               0.3391        0.0063        <.0001

con6           0.78046       0.29231      -0.47408
               0.0131        0.4453        0.1973
```

PROC FACTOR

הפרוצדורה FACTOR מאפשרת לבצע ניתוח גורמים על סטים של נתונים, במטרה לאתר מבנים או קשרים בין משתנים בתוך סט נתונים רב-ממדי. ההנחה של ניתוח גורמים היא שניתן לקבץ משתנים שונים מתוך סט הנתונים על סמך הקשר (מתאם) ביניהם, ואז כל קבוצה מייצגת גורם (מבנה כלשהו).

אופן הכתיבה:

```
PROC FACTOR <אופציות שונות>;
VAR רשימת משתנים;
PRIORS communalities;
PARTIAL רשימת משתנים;
FREQ רשימת משתנים;
WEIGHT רשימת משתנים;
BY רשימת משתנים;
RUN;
```

עם זאת, מספיק להגדיר רק את ההוראה VAR כדי להריץ ניתוח באמצעות PROC FACTOR.

דוגמא:

```
proc factor;
 var q1-q15;
run;
```

הפלט הבסיסי המתקבל מהרצת קוד זה כולל את ה-eigenvalues של הפקטורים, את ה-loading של כל משתנה על כל פקטור (דפוס הפקטורים), את החלק של השונות המוסבר על ידי כל פקטור, ואת אמדי ה-communality (החלק במשתנה שמוסבר על ידי הפקטורים השונים) של כל משתנה:

The FACTOR Procedure
Initial Factor Method: Principal Components

Prior Communality Estimates: ONE

Eigenvalues of the Correlation Matrix: Total = 15 Average = 1

	Eigenvalue	Difference	Proportion	Cumulative
1	3.99560641	1.89459436	0.2664	0.2664
2	2.10101205	0.63162983	0.1401	0.4064
3	1.46938221	0.18071979	0.0980	0.5044
4	1.28866243	0.27740321	0.0859	0.5903
5	1.01125922	0.14195666	0.0674	0.6577
6	0.86930256	0.05448125	0.0580	0.7157
7	0.81482131	0.08871129	0.0543	0.7700
8	0.72611001	0.09814156	0.0484	0.8184
9	0.62796845	0.05497442	0.0419	0.8603
10	0.57299404	0.06865169	0.0382	0.8985
11	0.50434235	0.16241286	0.0336	0.9321
12	0.34192949	0.03669103	0.0228	0.9549
13	0.30523846	0.09186659	0.0203	0.9752
14	0.21337187	0.05537273	0.0142	0.9895
15	0.15799914		0.0105	1.0000

5 factors will be retained by the MINEIGEN criterion.

Factor Pattern

	Factor1	Factor2	Factor3	Factor4	Factor5
q1	-0.28132	0.60970	-0.25007	-0.16640	0.23331
q2	-0.53986	0.49674	0.06939	-0.29253	-0.03282
q3	-0.47880	0.52007	-0.20536	-0.00996	-0.10086
q4	-0.25652	0.07344	0.64490	0.16889	0.20960
q5	-0.49193	0.59945	0.06396	-0.28923	-0.12010
q6	-0.16714	0.05739	0.82393	0.09781	-0.05696
q7	0.14116	0.41696	0.03973	0.55584	0.43641
q8	-0.17716	0.39479	-0.16388	0.59599	0.15019
q9	-0.45111	0.21727	0.07992	0.04003	-0.46608
q10	0.78859	0.26682	-0.12543	0.04971	-0.13559
q11	0.81669	0.27791	-0.19956	0.07008	-0.11598
q12	0.60614	0.28264	0.23804	0.03677	-0.33132
q13	0.73367	0.37045	0.11714	0.02223	-0.15993
q14	0.68396	0.30924	0.30990	-0.24702	0.08729
q15	0.37800	0.07704	0.06488	-0.56351	0.52815

The FACTOR Procedure
Initial Factor Method: Principal Components

Variance Explained by Each Factor

Factor1	Factor2	Factor3	Factor4	Factor5
3.9956064	2.1010120	1.4693822	1.2886624	1.0112592

Final Communality Estimates: Total = 9.865922

q1	q2	q3	q4	q5	q6	q7	q8
0.59552124	0.62966711	0.55216629	0.55954926	0.70351384	0.72290737	0.69477177	0.59185701

q9	q10	q11	q12	q13	q14	q15
0.47592433	0.72965390	0.80240849	0.61507306	0.71529374	0.72810371	0.74951118

אופציות של PROC FACTOR

1. האופציה corr – אופציה זו מציגה את מטריצת המתאמים או מטריצת המתאמים החלקיים של כל המשתנים.
אופן הכתיבה:

corr

2. האופציה cov – אופציה זו מבקשת מ-PROC FACTOR לחשב את הפקטורים על סמך מטריצת ה-covariance במקום על סמך מטריצת המתאמים.
אופן הכתיבה:

cov

הערה: אופציה זו ניתנת להגדרה רק כאשר מוגדרת האופציה method (שתפורט להלן).

3. האופציה data – אופציה זו מגדירה את קובץ הנתונים עליו PROC FACTOR תבצע את ניתוח הגורמים.
אופן הכתיבה:

data = שם קובץ נתונים

קובץ הנתונים יכול להיות קובץ SAS הקיים בזיכרון, או קובץ נתונים מיוחד (לדוגמא מטריצת מתאמים) אשר הופק מפרוצדורה אחרת (לדוגמא מ-PROC CORR) או שהוקלד על ידי המשתמש.

דוגמא:

```
proc corr out = corr_mat noprint;
proc factor data = corr_mat;
run;
```

בדוגמא זו הפקנו באמצעות PROC CORR מטריצת מתאמים בין המשתנים הבלתי תלויים לקובץ נתונים של SAS, והשתמשו במטריצת מתאמים זו כדי לבצע ניתוח גורמים. התוצאות של ניתוח זה יהיו זהות לתוצאה של הניתוח בו נגדיר את המשתנים ב-PROC FACTOR (באמצעות ההוראה VAR).

הערה: כאשר משתמשים במטריצת מתאמים, אין צורך להגדיר את ההוראה VAR בפרוצדורה.

טיפ קריאה: הדוגמא הבאה קשה להבנה, ומיועדת למשתמשי SAS מתקדמים. מומלץ למשתמשי SAS מתחילים לדלג על דוגמא זו בשלב הראשון, ולעבור לאופציה הבאה.

דוגמא נוספת:

```
data corr_mat (type = corr);
_TYPE_ = 'CORR';
input _var_ $ q1-q4;
cards;
q1 1.0 . . .
q2 0.29541 1.0 . .
q3 0.36128 0.47993 1.0 .
q4 0.04800 0.13008 -0.00368 1.0
;
proc factor data = corr_mat;
run;
```

בדוגמא זו הקלדנו ישירות את מטריצת המתאמים כקובץ נתונים, ואמרנו ל-PROC FACTOR להשתמש בקובץ נתונים זה לניתוח הגורמים (מטעמי פשטות השתמשנו בדוגמא זו רק ב-4 משתנים).

הערה: בנוסף למטריצת המתאמים, אפשר להשתמש גם בסוגים שונים של קבצי נתונים, כגון ה-loading של הגורמים (פקטורים) השונים (באמצעות הגדרת type = factor). _TYPE_ הוא סוג מיוחד של סט נתונים של SAS המכיל סוגים שונים של נתונים סטטיסטיים, כגון ממוצעים ('MEAN' = _TYPE_), סטיות תקן ('STD' = _TYPE_), דפוסי פקטורים ('FACTOR' = _TYPE_ - loading), מתאמים (= _TYPE_ 'CORR') ועוד. לפרוצדורות שונות יש סוגי משתני TYPE שונים שהן יכולות לקבל כקלט או להפיק כפלט (באמצעות ההוראה או האופציה OUT).

4. האופציה hey – אופציה זו קובעת את הערך 1 לכל communality הגדולה מ-1. אופציה זו נועדה לאפשר במצב בו יש Heywood case (שערכי ה-communality עולים על 1, דבר המעיד על בעיה במודל) להמשיך להריץ את ניתוח הגורמים ולאפשר עוד איטרציות על הנתונים.
אופן הכתיבה:

hey

5. האופציה maxiter – אופציה זו מגדירה את מספר האיטרציות המקסימלי שיתבצע בניתוח הגורמים. כברירת מחדל, המספר מוגדר ל-30 איטרציות.
אופן הכתיבה:

maxiter

6. האופציה method – אופציה זו מגדירה את השיטה לחישוב גורמים. כברירת מחדל, השיטה לחישוב הגורמים היא principal (פירוק לגורמים ראשוניים).
אופן הכתיבה:

method = alpha | harris | image | ml | pattern | principal | prinit | score | uls

7. האופציה mineigen – אופציה זו מגדירה את ערך ה-eigenvalue המינימאלי שעבורו ניתן לקבוע על קיום של גורם.
אופן הכתיבה:

mineigen = ערך מספרי מעל 0

8. האופציה msa – אופציה זו מחשבת את המתאמים החלקיים בין כל שילוב אפשרי של שני משתנים.
אופן הכתיבה:

msa

9. האופציה nfactors – אופציה זו מגדירה את מספר הפקטורים המקסימאלי שיש להוציא מהניתוח. כברירת מחדל, ערך זה שווה למספר המשתנים בקובץ הנתונים. רצוי להשתמש באופציה זו כאשר יש השערה מראש על כמות הגורמים שאמורים להסביר את הנתונים. אופציה זו מקצרת את זמן העיבוד ומקילה על עומס העבודה של המעבד.
אופן הכתיבה:

nfactors = מספר שלם הנע בין 1 למספר המשתנים בקובץ הנתונים

10. האופציה nobs – אופציה זו מגדירה את מספר התצפיות מקובץ הנתונים עליהם יש לבצע את ניתוח הגורמים. כברירת מחדל מספר זה שווה לכל מספר התצפיות בקובץ.
אופן הכתיבה:

nobs = מספר שלם הנע בין 1 למספר התצפיות בקובץ

11. האופציה noprint – אופציה זו מונעת מ-PROC FACTOR להפיק את קובץ הפלט.
אופן הכתיבה:

noprint

12. האופציה out – אופציה זו אומרת ל-PROC FACTOR ליצור קובץ נתונים המכיל את כל הנתונים מניתוח הגורמים, כולל משתנים הקרויים factor*1* עד factor*n*, המכילים את אמדי ציוני הפקטורים.
אופן הכתיבה:

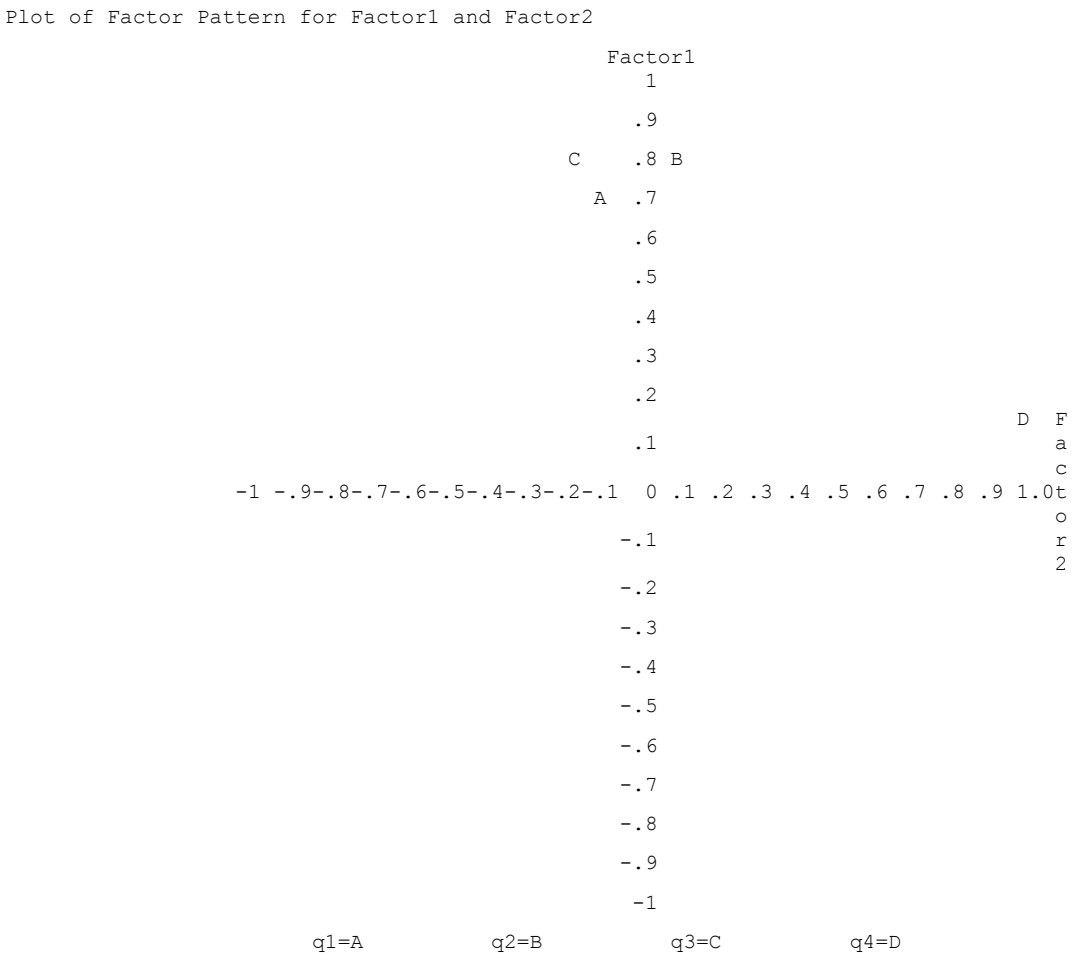
out = שם קובץ נתונים

13. האופציה plot – אופציה זו אומרת ל-PROC FACTOR להפיק תרשים של דפוס הפקטורים לאחר הרוטציה.

אופן הכתיבה:

plot

לדוגמא, שימוש באופציה זו יפיק את התרשים הבא:



14. האופציה rotate – אופציה זו מגדירה את סוג הרוטציה בניתוח הגורמים. כברירת מחדל, לא מוגדרת שום רוטציה על הנתונים.
אופן הכתיבה:

rotate = equamax | hk | none | orthomax | parsimax | procrustes | promax | quartimax | varima

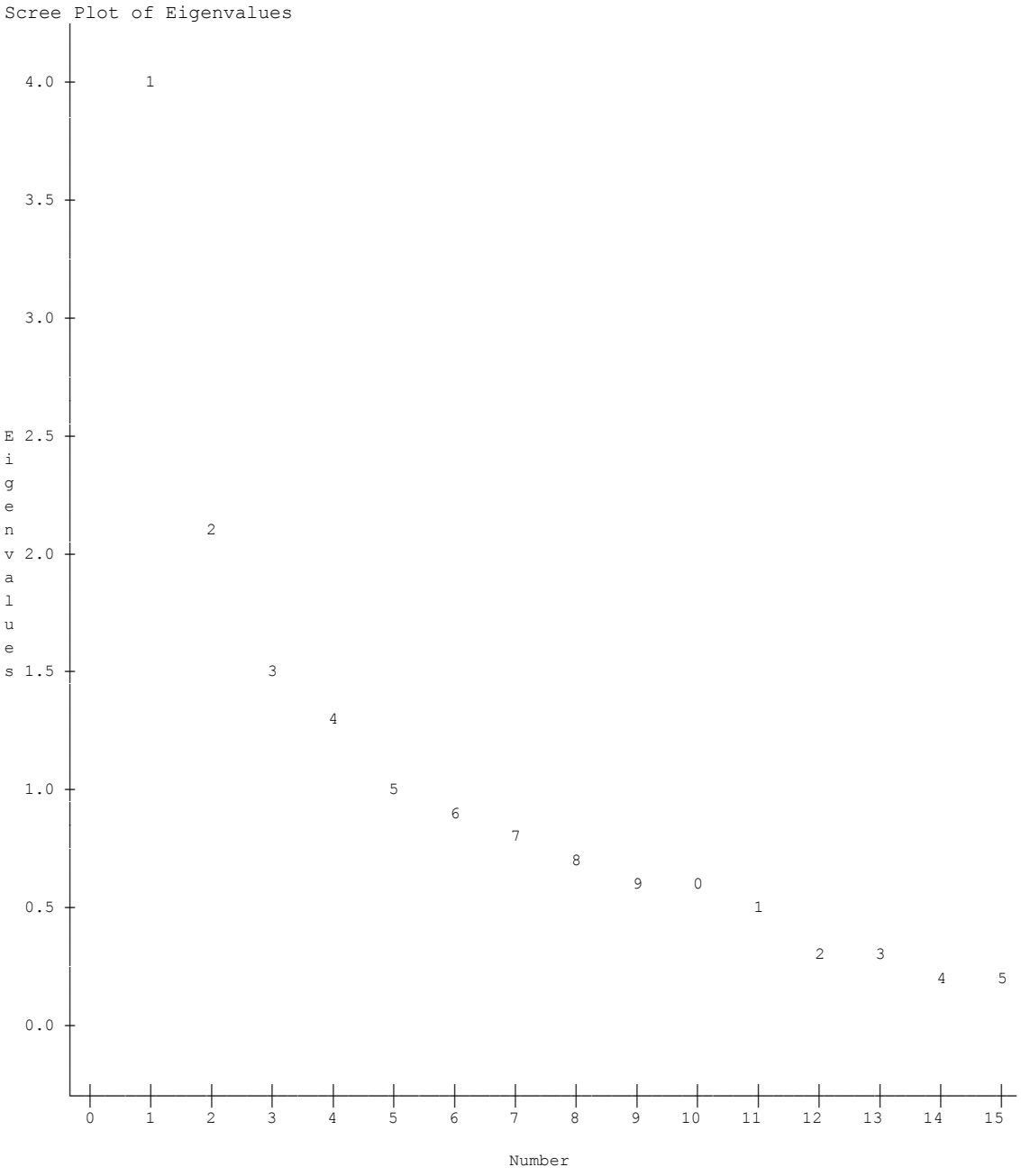
15. האופציה simple – אופציה זו תפיק סטטיסטיקה תיאורית (ממוצעים, סטיות תקן ומספר תצפיות) בנוסף לניתוח הגורמים.
אופן הכתיבה:

simple

16. האופציה scree – אופציה זו מציגה את ה-scree plot של ה-eigenvalues.
אופן הכתיבה:

scree

שימוש באופציה זו יפיק תרשים כגון:



17. האופציה priors – אופציה זו מגדירה ל-PROC FACTOR את השיטה לחישוב הערכות אמדי communalities שמוגדרים מראש על ידי המשתמש (באמצעות ההוראה PRIORS, כפי שיוגדר להלן).
אופן הכתיבה:

priors = ASMC | input | MAX | one | random | SMC

כאשר:

- ASMC – מגדיר את האמדים ל-communalities באופן פרופורציונאלי למתאם המותאם (adjusted) המרובה הריבועי של המשתנים, כך שהסכום שלהם יהיה שווה לערך המתאם המקסימאלי (בערך מוחלט)
- input – קורא את האמדים ל-communalities מקובץ חיצוני (המוגדר כ-TYPE = FACTOR)

- MAX – מגדיר את האמדים ל-communalities עבור כל משתנה כמתאם המקסימאלי (בערך אבסולוטי) עם כל משתנה אחר
- one – מגדיר את כל האמדים ל-communalities כ-1.0
- random – מגדיר את כל האמדים ל-communalities למספר פסאודו-אקראי הנדגם מהתפלגות אחידה בין 0 ל-1
- SMC – מגדיר את האמדים ל-communalities באופן פרופורציונאלי למתאם המרובה הריבועי של המשתנים

ברירת המחדל של האופציה priors תלויה בסוג הרוטציה המוגדרת ב-PROC FACTOR (בדרך כלל one או SMC).

ההוראה PRIORS

ההוראה PRIORS מגדירה את הערכים המספריים של האמדים ל-communalities (שמוגדרים על ידי המשתמש). כאשר מגדירים מראש את ה - communalities באמצעות ההוראה PRIOIRS, יש להגדיר ל-PROC FACTORS את שיטת הניתוח לחישוב הערכות אמדי communalities באמצעות האופציה priors ב-PROC FACTOR.

אופן הכתיבה:

PRIORS communalities - ערכי ה

יש לכתוב ערך לכל משתנה המופיע בהוראה VAR, כך שהערך הראשון יתייחס למשתנה הראשון המופיע בהוראה, הערך השני יתייחס למשתנה השני המופיע בהוראה, וכך הלאה.

דוגמא:

```
proc factor priors = max;
 var q1 q2 q3;
 priors .6 .5 .7;
run;
```

ההוראה PARTIAL

ההוראה PARTIAL מגדירה ל-PROC FACTOR לבסס את ניתוח הגורמים על מתאמים חלקיים או מטריצת covariance בין המשתנים.

אופן הכתיבה:

PARTIAL רשימת משתנים;

רשימת המשתנים המוגדרת בהוראה ישמשו לחישוב המתאמים החלקיים.

ההוראה FREQ

ההוראה FREQ באה להגדיר משתנים אשר ערכם מייצג את השכיחות של ערך כלשהו בתצפיות (דהיינו, פרופורציה של התרחשות של ערך כלשהו, ולא את הערך עצמו). כתוצאה מכך, PROC FACTOR תתייחס לערך זה כאילו הוא מופיע x פעמים בקובץ הנתונים (כאשר x שווה לשכיחות הערך, או הערך של המשתנה המוגדר בהוראה FREQ).

אופן הכתיבה:

```
שמות משתנים FREQ;
```

הערה: ההוראה FREQ מתפקדת בדיוק כמו ההוראה WEIGHT ב-PROC FREQ.

ההוראה WEIGHT

ההוראה WEIGHT דומה מאוד להוראה FREQ, רק שבמקום להגדיר שכיחות, מגדירים את המשקולות הסובייקטיבים שרוצים לתת לכל משתנה.

אופן הכתיבה:

```
רשימת משתנים WEIGHT;
```

ההוראה BY

ההוראה BY גורמת ל-PROC FACTOR לבצע ניתוח גורמים בנפרד לכל קבוצה של משתנים המוגדרים על ידי ההוראה BY.

אופן הכתיבה:

```
רשימת משתנים BY;
```

תרגול עצמי – פרוצדורות סטטיסטיות II – קשר בין משתנים

תרגיל 29

נתון קובץ נתונים הכולל את המשתנים שעות לימוד (hour), רמת חרדה (anx – משתנה בינארי, המקבל את הערך 1 לרמת חרדה גבוהה ואת הערך 0 לרמת חרדה נמוכה) וציון מבחן (grade).

hour	anx	grade
9	0	90
4	1	75
5	1	80
5	0	65
7	1	65
5	0	90
6	0	95
3	1	67
4	1	57

5	0	85
8	0	90
7	1	70
9	0	90
4	1	61
5	1	80
5	0	90
7	1	65
8	0	95
6	0	95
3	1	59
4	1	57
3	0	70
8	0	90
7	1	60

א. בדוק את הקשר בין מספר שעות הלימוד לציון המבחן. מה ניתן להסיק מקשר זה?

ב. המרצה בקורס טוען כי הקשר בין שעות הלימוד לציון המבחן תלוי ברמת החרדה. בדוק האם טענתו של המרצה נכונה.

תרגיל 30

חוקר א' פיתח שאלון לבדיקת רמת חרדה הכולל 8 שאלות. להלן התוצאות של 20 נבדקים שענו על השאלון:

2	2	2	2	2	1	1	2
1	3	2	2	2	3	1	1
3	2	2	1	2	1	4	4
1	3	3	2	2	2	2	2
2	1	3	1	1	1	3	2
1	2	2	3	2	2	2	2
1	1	1	1	1	1	2	2
1	2	2	1	1	1	3	2
1	2	1	3	1	4	2	3
1	1	1	1	1	1	2	2
2	3	2	3	1	2	4	3
1	1	1	1	1	1	1	1
3	3	3	2	3	1	3	3
2	2	2	3	2	3	2	2
2	2	2	2	2	1	3	3
1	2	1	1	1	1	3	3
2	2	2	2	1	1	2	2
2	3	2	1	1	2	2	2
2	1	2	4	1	2	3	2
2	2	2	1	1	1	2	2

חוקר ב' טען כי הוא בדק ומצא שיש בעיה עם השאלון, שכן העקיבות הפנימית בין הפריטים לא מספיק גבוהה. בדוק האם הטענה של חוקר ב' מוצדקת.

תרגיל 31

הערה: תרגיל זה לא מתייחס לקובץ נתונים ספציפי

כתוב קוד SAS לבדיקת קשר בין קבוצת המשתנים Max, Avg, P ו-Recencey, לבין קבוצת המשתנים Ant, Mid ו-Pos. הקוד צריך להיות מוגדר כך שהפלט ייתן את המתאמים רק בין משתנים מקבוצות שונות, אך לא בין משתנים מאותה קבוצה (למשל, הפלט צריך לכלול את הקשר בין Recencey או Max לבין המשתנים Ant, Mid ו-Pos, אך לא בין Recencey לבין משתנים אחרים מהקבוצת המשתנים הראשונה כגון P או Max).

תרגיל 32

להלן קובץ נתונים המכיל תשובות של 28 נבדקים על שאלון בן 10 שאלות:

3	3	3	3	3	2	2	2	2	2
3	3	4	3	4	1	3	2	2	2
2	2	1	5	3	3	2	2	1	2
2	3	1	5	5	1	3	3	2	2
3	3	4	4	2	2	1	3	1	1
5	3	2	5	4	1	2	2	3	2
1	2	2	2	3	1	1	1	1	1
1	2	1	3	3	1	2	2	1	1
3	3	3	4	4	1	2	1	3	1
3	2	2	5	5	1	1	1	1	1
2	4	3	4	5	2	3	2	3	1
3	2	2	4	3	1	1	1	1	1
2	2	2	4	3	3	3	3	2	3
3	2	1	5	3	2	2	2	3	2
1	4	1	5	3	2	2	2	2	2
3	2	2	5	4	1	2	1	1	1
4	3	3	4	5	2	2	2	2	1
1	1	1	3	4	2	3	2	1	1
4	2	2	4	4	2	1	2	4	1
3	2	1	3	4	2	2	2	1	1
3	3	2	4	4	2	3	2	4	3
4	3	2	4	4	1	2	2	2	2
2	3	3	3	2	3	3	3	2	2
3	3	4	4	5	1	3	3	4	1
3	3	3	4	3	2	1	1	1	1
4	2	3	4	4	3	2	2	1	2
4	4	4	4	3	2	2	2	2	2
5	4	4	4	4	1	2	1	1	1

כתוב קוד SAS לביצוע ניתוח גורמים על נתונים אלה. ניתוח הגורמים צריך להתבצע על מטריצת הקורלציות של המשתנים (כאשר כל שאלה מוגדרת כמשתנה), ולא על המשתנים הגולמיים עצמם (לא על התשובות לשאלות). הפרוצדורה לניתוח הגורמים צריכה לכלול רוטציה מסוג varimax ולהפיק scree plot.

פרק 11
פרוצדורות סטטיסטיות III: מודלים ליניארים

PROC TTEST

הפרוצדורה TTEST מחשבת מבחני t (מבחני השוואת ממוצעים) ל-

1. מדגם יחיד – השוואת ממוצע המדגם לערך כלשהו (מוגדר על ידי המשתמש)
2. מדגמים בלתי תלויים – השוואת ממוצע התצפיות של קבוצה א' לממוצע התצפיות של קבוצה ב'
3. מדגמים מזווגים – השוואת ממוצע הפרשי התצפיות בין שתי קבוצות תלויות (אותם נבדקים תחת טיפולים או תנאי ניסוי שונים, בני זוג, הורים וילדים וכדומה) לערך כלשהו

כברירת מחדל PROC TTEST משתמשת בהשערת אפס לפיה ממוצע המדגם או ממוצע ההפרשים בין ממוצעי שתי הקבוצות (או שני המדגמים) שווה לאפס (בכל שלושת המקרים).PROC TTEST פועלת תחת ההנחה שכל התצפיות בקובץ הנתונים מהווים דגימה מקרית מאוכלוסיה המתפלגת נורמאלית. ניתן לבחון הנחה זאת על ידי שימוש ב- PROC UNIVARIATE (ראה פרק 9). במקרה בו הנחת הנורמאליות לא מתקיימת, יש להשתמש ב-PROC NPAR1WAY (ראה פרק 12).

בנוסף, הפרוצדורה מניחה שהשונות של שתי הקבוצות שוות. עם זאת, במצב בו לא ניתן להניח זאת, PROC TTEST מחשבת הערכה של ערך הסטטיסטי t במקרה בו השונויות אינן שוות.

אופן הכתיבה:

```
PROC TTEST אופציות שונות;
BY <descending> משתנה 1 <descending> … משתנה n <notsorted>;
CLASS משתנה;
PAIRED רשימת משתנים מזווגים;
VAR רשימת משתנים;
RUN;
```

דוגמא (מבחן t למדגם יחיד):

```
proc ttest h0 = 10 data=dogma;
 var con1;
run;
```

בדוגמא זו, אנחנו בודקים האם ממוצע המשתנה con1 שונה באופן מובהק מהערך 10.

הפלט הבסיסי המתקבל מהרצת הפרוצדורה למבחן t למדגם יחיד הוא:

```
                                The SAS System
                              The TTEST Procedure

                                  Statistics

                    Lower CL          Upper CL  Lower CL           Upper CL
Variable  N    Mean    Mean      Mean    Std Dev  Std Dev   Std Dev  Std Err  Minimum  Maximum
con1     24  16.207    18.5    20.793    4.2198   5.4294    7.6161   1.1083       12       25

                                  T-Tests

                          Variable      DF    t Value    Pr > |t|

                          con1          23       7.67      <.0001
```

כברירת מחדל, פלט זה כולל ממוצע, גבול סמך עליון ותחתון לממוצע, סטיית תקן, גבול סמך עליון ותחתון לסטיית התקן, טעות התקן, וערך מינימאלי ומקסימאלי. בנוסף הפלט כולל את הסטטיסטי t של המבחן, דרגות החופש ואת רמת המובהקות. כפי שניתן לראות מהתוצאות, ממוצע המשתנה con1 שווה ל-18.5. ערך זה שונה במובהק (ברמת מובהקות של 0.0001) מ-10.

דוגמא (מבחן t למדגמים מזווגים):

```
proc ttest data=dogma;
 paired con1*con2;
run;
```

בדוגמא זו אנחנו בודקים האם הממוצע היה שונה תחת תנאי 1 לעומת תנאי 2, בהנחה שהמדגם שלנו עבר גם את התנאי הראשון וגם את התנאי השני.

הפלט הבסיסי המתקבל מהרצת הפרוצדורה למבחן t למדגמים מזווגים הוא:

```
                              The TTEST Procedure

                                  Statistics

                    Lower CL          Upper CL  Lower CL           Upper CL
Difference  N   Mean    Mean      Mean    Std Dev  Std Dev   Std Dev  Std Err  Minimum  Maximum
con1 - con2 24 -12.2  -9.333    -6.465    5.2796    6.793     9.529   1.3866      -19        2

                                  T-Tests

                          Difference       DF    t Value    Pr > |t|

                          con1 - con2      23      -6.73      <.0001
```

למעשה, פלט זה זהה לפלט המתקבל ממבחן t למדגם יחיד, למעט העובדה שהסטטיסטים המחושבים הם על הפרש הממוצעים ולא על הממוצע. כפי שניתן לראות, הפרש הממוצעים בין שני התנאים הוא 9.33- (מה שאומר שבתנאי הראשון הממוצע נמוך יותר מאשר הממוצע בתנאי השני ב-9.33 יחידות). כפי שהתוצאות מראות, הפרש זה מובהק ברמת מובהקות 0.0001.

דוגמא (מבחן t למדגמים בלתי תלויים):

```
proc ttest data=dogma;
 class x;
 var y;
run;
```

דוגמא זו בודקת האם הממוצע שלך קבוצה א' שונה במובהק מהממוצע של קבוצה ב' בערך של המשתנה y, כאשר ההבחנה בין הקבוצות השונות נעשית על ידי המשתנה הקטגוריאלי המוגדר על ידי ההוראה CLASS (המשתנה x בדוגמא זו).

הפלט הבסיסי המתקבל מהרצת הפרוצדורה למבחן t למדגמים בלתי תלויים כולל את הסטטיסטים לכל קבוצה בנפרד, ולממוצע ההפרשים בין הקבוצות. כמו כן, הפלט כולל את הסטטיסטי t של המבחן, דרגות החופש ואת רמת המובהקות, הן במקרה בו ניתן להניח שוויון בשונויות, והן במקרה בו לא ניתן להניח זאת. לבסוף, הפלט כולל גם את תוצאות המבחן הסטטיסטי לבדיקת הנחת השוויון בשונויות:

The SAS System

The TTEST Procedure

Statistics

Variable	x	N	Lower CL Mean	Mean	Upper CL Mean	Lower CL Std Dev	Std Dev	Upper CL Std Dev	Upper CL Std Err
y	0	24	16.207	18.5	20.793	4.2198	5.4294	7.6161	1.1083
y	1	24	23.791	27.833	31.875	7.4395	9.572	13.427	1.9539
y	Diff (1-2)		-13.85	-9.333	-4.812	6.4662	7.7814	9.7734	2.2463

T-Tests

Variable	Method	Variances	DF	t Value	Pr > |t|
y	Pooled	Equal	46	-4.15	0.0001
y	Satterthwaite	Unequal	36.4	-4.15	0.0002

Equality of Variances

Variable	Method	Num DF	Den DF	F Value	Pr > F
y	Folded F	23	23	3.11	0.0087

כאשר מריצים מבחן t למדגמים בלתי תלויים, ראשית יש לבחון בפלט את הנתונים המוצגים תחת הכותרת "Equality of Variance". הנתונים הנמצאים תחת כותרת זו נותנים לנו את התוצאות של המבחן אשר בדק את ההנחה שהשונויות של שתי הקבוצות שוות. השערת האפס של מבחן זה מניחה שוויון בין השונויות. לכן, אם המבחן יוצא מובהק (כאשר הערך p נמוך מספיק), לא ניתן להניח שוויון בין השונויות, ולהפך. בהתאם לתוצאות של מבחן זה, יש לבחון את תוצאות הפלט של המבחן t.

ספציפית:

א. במקרה בו ניתן להניח שוויון בין השונויות הפרוצדורה מחשבת את ערך הסטטיסטי t בהתאם לשונות המשותפת של שתי הקבוצות. לכן, במצב זה נבחן את התוצאות בשורה הראשונה המוצגת בפלט תחת הכותרת "T-Tests".

ב. במקרה בו לא ניתן להניח שוויון בין השונויות הפרוצדורה מחשבת את ערך הסטטיסטי t בצורה שונה, ולא על סמך השונות המשותפת. במצב זה נבחן את התוצאות בשורה השנייה המוצגת בפלט תחת הכותרת "T-Tests".

בדוגמא הנוכחית, המבחן להנחת שוויון בין השונויות יוצא מובהק ($p = 0.0087$). לכן לא נוכל להניח שוויון בין השונויות במקרה זה, ונצטרך לבחון את תוצאות המבחן t המוצגות בשורה השנייה. כפי שניתן לראות, ההבדל בין הקבוצות יצא מובהק ברמת מובהקות של 0.0002 (לעומת 0.0001 במצב בו השונויות היו שוות).

הערה: יש לשים לב כי כאשר לא ניתן להניח שוויון בין השוניות, דרגות החופש של המבחן שונות (36.4 בדוגמא שלנו), והן לא $n_1 + n_2 - 2$ (דהיינו גודל שני המדגמים פחות 2).

אופציות של PROC TTEST

1. האופציה data – אופציה זו מגדירה את קובץ הנתונים עליו PROC TTEST תעבוד. אם לא מגדירים אופציה זו, הפרוצדורה תעבוד על קובץ הנתונים האחרון שנמצא בזיכרון התוכנה.
אופן הכתיבה:

data = שם של קובץ נתונים

הערה: קובץ הנתונים יכול להכיל סיכום של הסטטיסטים של המשתנים (גודל המדגם, ממוצע, וסטיית תקן) ולא חייב להכיל את כל הנתונים הגולמיים (התצפיות).

2. האופציה alpha – אופציה זו מגדירה רמת הביטחון לרווח הסמך לממוצע התצפיות או לממוצע הפרשי התצפיות המופק על ידי הפרוצדורה. רווח הסמך מוגדר על ידי הנוסחא $100(1-p)$, כאשר p הוא הערך המוגדר על ידי האופציה ($0 < p < 1$). כברירת מחדל, PROC TTEST משתמשת בערך $p = 0.05$.
אופן הכתיבה:

alpha = ערךמספרי בין 0 ל- 1

3. האופציה h0 – אופציה זו מבקשת לבחון את נתוני המדגם אל מול השערת אפס שונה מ-0. אופציה זו רלוונטית לגבי שלושת סוגי המבחנים.
אופן הכתיבה:

h0 = ערך מספרי

ההוראה BY

הוראה זו אומרת ל-PROC TTEST לחשב מתאמים לכל קבוצה המוגדרת על ידי ההוראה באופן נפרד. לפני השימוש בהוראה יש למיין את קובץ הנתונים על פי המשתנה או המשתנים המוגדרים על ידי ההוראה BY (אלא אם משתמשים באופציה notsorted).

אופן הכתיבה:

BY <descending> משתנה 1 … <descending> משתנה n <notsorted>;

דוגמא:

```
proc ttest;
 by gender notsorted;
run;
```

ההוראה CLASS

הוראה זו מגדירה ל-PROC TTEST את המשתנה הקטגוריאלי אשר קובע את החלוקה לשתי הקבוצות השונות עליהן יש לבצע מבחן t למדגמים בלתי תלויים. לכן, הוראה זו חייבת לבוא אך ורק במקרים בהם משתמשים במבחן t למדגמים בלתי תלויים, והיא לא יכולה להופיע כאשר משתמשים במבחן t למדגמים מזווגים או מבחן t למדגם יחיד. משתנה ה-CLASS יכול להיות נומרי או אלפאנומרי, אך הוא חייב לכלול שני ערכים בלבד.

כאשר לא מגדירים משתנים ספציפיים לניתוח (על ידי ההוראה VAR), כל המשתנים הנומריים בקובץ הנתונים (חוץ מאלה המוגדרים על ידי ההוראות BY ו-CLASS) נכללים בניתוח.

אופן הכתיבה:

CLASS משתנה;

דוגמא:

```
proc ttest;
 class group;
run;
```

ההוראה PAIRED

הוראה זו מגדירה את המשתנים עליהם יש לבצע את המבחן t למדגמים מזווגים. כאשר מגדירים את ההוראה PAIRED, לא ניתן להשתמש בהוראה CLASS או בהוראה VAR.

אופן הכתיבה:

PARIED רשימת משתנים מזווגים;

ניתן לכתוב רשימה אחת של משתנים או מספר רשימות של משתנים, כאשר משתנים בתוך כל רשימה מופרדים על ידי כוכבית (*) או נקודתיים (:). הכוכבית מבקשת השוואה בין כל המשתנים המופיעים ברשימה. לעומת זאת, נקודתיים מבקשת השוואות בין המשתנה הראשון משמאל לנקודתיים למשתנה הראשון מימין לנקודתיים, המשתנה השני משמאל למשתנה השני מימין וכך הלאה. לכן, כאשר משתמשים בנקודתיים, חייבים לוודא שמספר המשתנים הרשומים משמאל שווה למספר המשתנים הרשומים מימין.

להלן מספר דוגמאות לשימוש בכוכביות ונקודתיים כדי להגדיר השוואות במבחן t למדגמים מזווגים:

1.

```
paired x * y;
```

דוגמא זו אומרת לפרוצדורה לבצע מבחן t להבדל בין המשתנה x למשתנה y.

2.

```
paired x * y  z * w;
```

דוגמא זו אומרת לפרוצדורה לבצע מבחן t להבדל בין x ל-y ומבחן t להבדל בין z ל-w.

3.

```
paired (x  y) * (z  w);
```

או

```
paired (x-y) * (z-w);
```

דוגמאות אלה אומרות לפרוצדורה לבצע מבחן t להבדל בין x ל-z, בין x ל-w, בין y ל-z, ובין y ל-w.

4.

```
paired (x-y) : (z-w);
```

דוגמא זו אומרת לפרוצדורה לבצע מבחן t להבדל בין x ל-z ובין y ל-w.

ההוראה VAR

הוראה זו מגדירה ל-PROC TTEST את המשתנים עליהם יש לבצע את הניתוחים הסטטיסטיים. הוראה זו יכולה לבוא כאשר עושים מבחן t למדגם יחיד, או למדגמים בלתי תלויים (בליווי ההוראה CLASS בלבד). לעומת זאת, לא ניתן להגדיר את ההוראה כאשר מבצעים מבחן t למדגמים מזווגים.

כאשר לא מגדירים הוראה זו, PROC TTEST תבצע את הניתוחים הסטטיסטיים על כל המשתנים הנומריים בקובץ הנתונים, למעט משתנים המוגדרים על ידי הוראות אחרות.

אופן הכתיבה:

VAR רשימת משתנים;

PROC REG

הפרוצדורה REG הינה פרוצדורה כללית לחישוב מודלים ליניאריים רגילים ופולינומים של רגרסיה. בנוסף להפקת אומדנים למודל (כולל סטיות תקן ומבחן wald למובהקות), הפרוצדורה מחשבת לכל תצפית בקובץ את טעות האמידה, ואת הערך המנובא של המשתנה התלוי על פי המודל. כמו כן, הפרוצדורה מפיקה תרשימים (דיאגרמות פיזור) של משתני המודל, ומבצעת בדיקת השערות ליניאריות על הנתונים.

אופן הכתיבה:

PROC REG אופציות שונות;
MODEL משתנה תלוי = משתנים בלתי תלויים / אופציות שונות;
BY רשימת משתנים;
OUTPUT <OUT = שם של קובץ נתונים> מילות מפתח = שמות;
PLOT <y משתנה לציר * x משתנה לציר> <= סמל> / אופציות שונות;
RUN;

למרות של-PROC REG יש הרבה הוראות, כאשר מריצים מודל רגרסיה ניתן להשתמש רק במעט מהן.

דוגמא:

```
proc reg;
 model m_con = con1 con2;
run;
```

הפלט הבסיסי המתקבל בעקבות הרצה זו כולל את מספר התצפיות, את דרגות החופש, סכום הריבועים, ערך הסטטיסטי F ורמת המובהקות של המודל, את ממוצע המשתנה התלוי, R2, את השורש של סכום הטעויות, את שונות המקדמים, ואת הסטטיסטים של מקדמי המודל:

```
                    The REG Procedure

                      Model: MODEL1
                Dependent Variable: m_con

          Number of Observations Read          24
          Number of Observations Used          24
```

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	2	1127.33333	563.66667	Infty	<.0001
Error	21	0	0		
Corrected Total	23	1127.33333			

Root MSE	0	R-Square	1.0000
Dependent Mean	23.16667	Adj R-Sq	1.0000
Coeff Var	0		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	2.58613E-14	0	Infty	<.0001
con1	1	0.50000	0	Infty	<.0001
con2	1	0.50000	0	Infty	<.0001

אופציות של PROC REG

אופציות הקשורות לקבצי נתונים:

1. האופציה alpha – אופציה זו מגדירה רמת הביטחון לרווח הסמך.
אופן הכתיבה:

alpha = מספר בין 0 ל-1

אופציות הקשורות לקובץ הפלט:

1. האופציה corr – אופציה זו מוסיפה לקובץ הפלט מטריצת מתאמים בין כל המשתנים המוגדרים על ידי ההוראה MODEL או VAR.
אופן הכתיבה:

corr

2. האופציה simple – אופציה זו מוסיפה לקובץ הפלט סטטיסטים תיאוריים (סכום, ממוצע, שונות, סטיית תקן וסכום ריבועי ההפרש) לכל משתנה המוגדר על ידי PROC REG.
אופן הכתיבה:

simple

3. האופציה noprint – אופציה זו מבטלת את התצוגה הרגילה שמופקת על ידי PROC REG.
אופן הכתיבה:

noprint

4. האופציה lineprinter – אופציה זו אומרת ל-SAS להפיק את הגרפים המוגדרים על ידי הפרוצדורה בקובץ הפלט, ולא כתרשים ברזולוציה גבוהה בחלון SAS/GRAPH (ברירת המחדל ליצירת תרשימים ב-PROC REG). יש להגדיר אופציה זו במצב בו SAS/GRAPH לא מותקנת במסגרת חבילת SAS.
אופן הכתיבה:

lineprinter

ההוראה BY

הוראה זו אומרת ל-PROC REG לבצע ניתוח רגרסיה לכל קבוצה המוגדרת על ידה באופן נפרד. לפני השימוש בהוראה יש למיין את קובץ הנתונים על פי המשתנה או משתנים המוגדרים על ידי ההוראה BY (אלא אם משתמשים באופציה notsorted).

אופן הכתיבה:

BY משתנים;

ההוראה MODEL

הוראה זו מגדירה את המשתנים התלויים והבלתי תלויים של המודל אותו PROC REG צריכה לאמוד. לאחר ההוראה MODEL יש לכתוב את המשתנים התלויים, ולאחר מכן את הסימן שווה (=) ואת המשתנים המסבירים.משתנים המופיעים בהוראה MODEL חייבים להיות משתנים נומריים.

ניתן להגדיר תווית למודל באמצעות כתיבת התווית (ללא רווחים) לפני ההוראה, ולהפריד בין התווית להוראה באמצעות הסימן נקודתיים (:).

אופן הכתיבה:

תווית:>MODEL משתנה תלוי = משתנים בלתי תלויים /אופציות שונות;

אופציות של ההוראה MODEL

אופציות כלליות:

1. האופציה noprint – אופציה זו מבטלת את הפלט הרגיל של תוצאות ניתוח הרגרסיה.
   אופן הכתיבה:

\noprint

2. האופציה noint – אופציה זו מבטלת את הפקת החותך של מודל הרגרסיה, המופק באופן רגיל במודל הרגרסיה בצורה אוטומטית.
   אופן הכתיבה:

\noint

אופציות להפקת חישובים למודל הרגרסיה:

1. האופציה SS1 – אופציה זו אומרת ל-SAS להוסיף לפלט גם את סכום הריבועיים הפחותים מסוג I (Type I SS), ביחד עם כל אמדי הפרמטרים של המודל
   אופן הכתיבה:

\ss1

2. האופציה ss2 – אופציה זו אומרת ל-SAS להוסיף לפלט גם את סכום הריבועיים הפחותים מסוג II (Type II SS), ביחד עם כל אמדי הפרמטרים של המודל.

אופן הכתיבה:

\ss2

3. האופציה stb – אופציה זו אומרת ל-SAS להפיק מקדמי רגרסיה מתוקננים.
אופן הכתיבה:

\stb

4. האופציה tol – אופציה זו מדפיסה את ערכי ה-tolerance עבור האמדים.
אופן הכתיבה:

\tol

5. האופציה vif – אופציה זו מדפיסה את ה-variance inflation factors ביחד עם אמדי הפרמטרים במודל. vif הוא למעשה ההופכי של ה-tolerance.
אופן הכתיבה:

\vif

הערה: האופציות tol ו-vif משמשות כמדד למולטיקולינאריות (מדד המציין האם יש מתאם גבוהה בין המשתנים הבלתי תלויים במודל)

6. האופציה collin – אופציה זו מפיקה ניתוח מפורט של קולינאריות (collinearity) בין הנבאים בניתוח הרגרסיה.
אופן הכתיבה:

\collin

ההוראה PLOT

הוראה זו אומרת ל-PROC REG להפיק עקומת פיזור של הנתונים או של סטטיסטיים של המודל. ניתן להפיק עקומה ברזולוציה גבוהה (ב-SAS/GRAPH) או עקומה ברזולוציה נמוכה בחלון Output (על ידי הגדרת האופציה lineprinter ב-PROC REG). הפקה של עקומת פיזור טובה כדי לבחון האם הקשר בין המשתנים הוא בעל דפוס לינארי (ההכרחי לביצוע ניתוחי רגרסיה) או האם יש נקודות חריגות (outliers) בנתונים.

ניתן להגדיר מספר הוראות PLOT לכל הוראת MODEL, או להגדיר מספר עקומות תחת הוראת PLOT אחת.

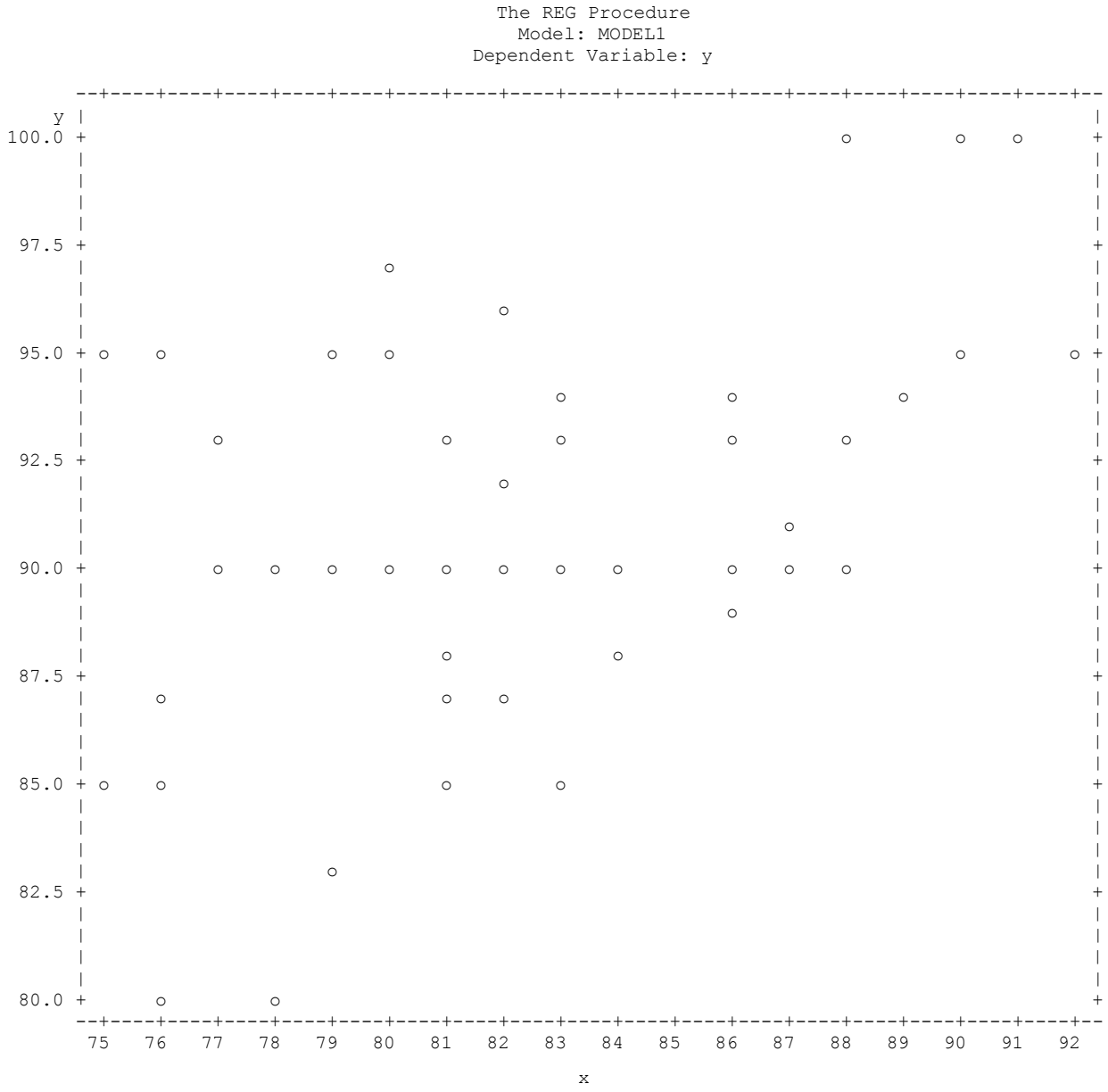
אופן הכתיבה:

PLOT <y משתנה ציר * x משתנה ציר> < = סמל> < y משתנה ציר * x משתנה ציר><= סמל><אופציות שונות />

דוגמא:

```
proc reg lineprinter;
 model y = x;
 plot y * x = 'o';
run;
```

קוד זה יפיק בחלון Output את עקומת הפיזור הבאה:



לחילופין, אם לא נשתמש בהוראה lineprinter, תופק עקומת פיזור בחלון SAS/GRAPH (ראה לדוגמא איור 17).

הערה: הגדרה של סמל (הגדרה של התו המייצג את נקודות המפגש של התצפיות בתוך העקומה) רלוונטית רק כאשר מפיקים עקומה בחלון Output (כאשר מגידרים את האופציה lineprinter). כאשר מפיקים עקומה לחלון SAS/GRAPH, להגדרת הסמל אין כל השפעה על יצירת התרשים.

הגדרה של משתנים או סטטיסטים ליצירת עקומת פיזור

כדי להפיק עקומת פיזור באמצעות ההוראה PLOT, יש להגדיר את המשתנה של ציר x ואת המשתנה של ציר y. משתנים אלה יכולים להיות:

1. כל משתנה מקובץ הנתונים המופיע במודל (המוגדר בהוראה MODEL)

אופן הכתיבה:

Plot y משתנה ציר * x משתנה ציר;

2. מילת מפתח (שבסיומה מופיעה נקודה), המסמלת סטטיסטי כלשהו של המודל. לרשימה מלאה של הסטטיסטים ראה טבלה 10.
אופן הכתיבה:

plot מילת מפתח1. * מילת מפתח2.;

דוגמא:

```
plot predicted. * stdr.;
```

בדוגמא זו הפקנו עקומת פיזור של הערכים המנובאים של המודל בהתאם לטעות התקן של טעות הניבוי (standard error of the residual).

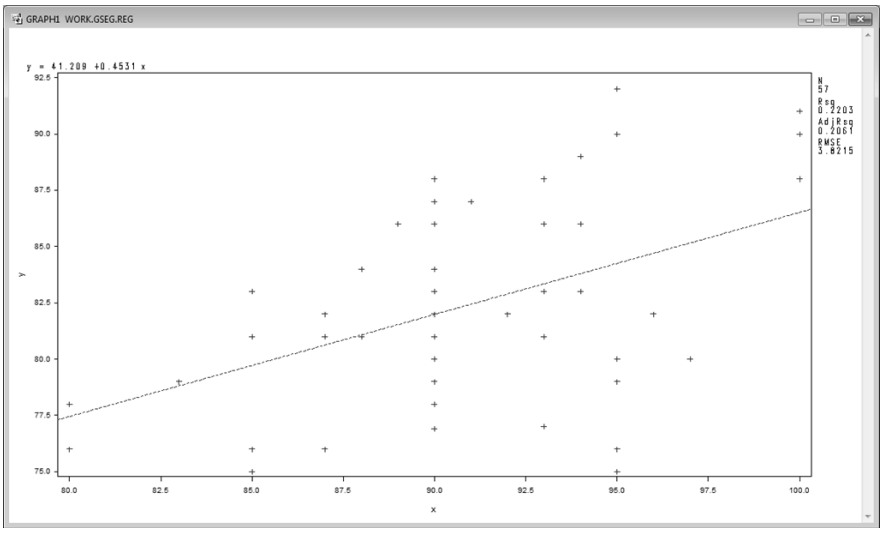
3. מילת המפתח obs. (מספר התצפיות). ניתן להפיק עקומת פיזור של מספר התצפיות כנגד כל אחד ממשתני המודל.
אופן הכתיבה:

plot obs. * שם משתנה;

4. מילות המפתח npp. (pp-plot – התפלגות נורמלית מצטברת) או nqq. (qq-plot – כלי דיאגנוסטי להערכת טיב התאמה של מודל פרמטרי). ניתן להגדיר מילות מפתח אלה עם כל אחד ממשתני המודל כדי ליצור pp-plot או qq-plot, בהתאמה.
אופן הכתיבה:

plot npp.|nqq. * שם משתנה;

הערה: השימוש במילות המפתח npp. ו-nqq. אפשרי רק כאשר מפיקים עקומת פיזור ברזולוציה גבוהה (ללא הגדרת האופציה lineprinter).



איור 17 – עקומת פיזור המופקת על ידי ההוראה PLOT ב-PROC REG, לקשר בין המשתנה התלוי למשתנה בלתי תלוי

בכל אחד מהמקרים, ניתן להגדיר יותר מזוג אחד של משתני x ו-y בכל הוראת PLOT (וכך בעצם להגדיר מספר עקומות פיזור תחת הוראה אחת). כדי לעשות זאת, ניתן להגדיר כל עקומת פיזור בנפרד, או להשתמש בקיצורי דרך. לדוגמא, כדי להפיק עקומות פיזור בין מספר התצפיות (obs.) לבין כל אחד משני משתני המודל (המשתנים x ו - y), ניתן לכתוב:

```
plot obs. * x  obs. * y;
```

או

```
plot (obs.) * (x  y);
```

מילת המפתח	תיאור
סטטיסטים דיאגנוסטים	
cookd.	הסטטיסטי Cook's D
covratio.	ההשפעה הסטנדרטית של התצפיות על השונות המשותפת של אמדי הפרמטרים
dffits.	ההשפעה הסטנדרטית של התצפיות על הערכים המנובאים
h.	$h_i = x_i\,(\mathbf{X}'\mathbf{X})^{-1}\,x_i'$
lcl.	הגבול התחתון של רווח הסמך לניבוי ספציפי.
lclm.	הגבול התחתון של רווח הסמך לערך הממוצע של הניבויים
predicted.	הערכים המנובאים על ידי המודל
press.	טעות הניבוי מהתאמה מחדש של המודל כאשר מוחקים תצפיות מסוימות
residual.	טעות הניבוי
rstudent.	טעות ניבוי מתקוננת מהתאמה מחדש של המודל כאשר מוחקים תצפיות מסוימות
stdi.	טעות תקן של ערך מנובא ספציפי
stdp.	טעות תקן של הערך המנובא הממוצע
stdr.	טעות התקן של טעות הניבוי
student.	טעות הניבוי חלקי טעות התקן
ucl.	הגבול העליון של רווח הסמך לניבוי ספציפי.
uclm.	הגבול העליון של רווח הסמך לערך הממוצע של הניבויים
סטטיסטים של טיב המודל	
adjrsq.	השונות המוסברת המתוקננת (adjusted R^2)
aic.	קריטריון המידע של Akaike
bic.	קריטריון המידע הבייסיאני של Sawa
cp.	הסטטיסטי C_P של Mallows
edf.	דרגות החופש של הטעות
gmsep.	תוחלת השגיאה הריבועית (MSE) של הניבויים
in.	מספר המנבאים במשוואת הרגרסיה, לא כולל החותך
jp.	טעות הניבוי הסופית
mse.	תוחלת השגיאה הריבועית
np.	מספר הפרמטרים במודל, כולל החותך
pc.	קריטריון הניבוי של Amemiya
rmse.	השורש הריבועי של MSE
rsq.	השונות המוסברת (R^2)
sbc.	הסטטיסטי SBC
sp.	הסטטיסטי SP
sse.	סכום הריבועים של הטעות

טבלה 10 - מילות מפתח למשתנה x ולמשתנה y בהוראה PLOT ב-PROC REG

אופציות של ההוראה PLOT

אופציות לתרשימים ברזולוציה גבוהה (המופקים בחלון SAS/GRAPH):

1. האופציה aic – אופציה זו מוסיפה את הערך של קריטריון האינפורמציה של Akaike לשוליים של התרשים.

אופן הכתיבה:

/aic

2. האופציה bic – אופציה זו מוסיפה את הערך של קריטריון האינפורמציה הבייסיאני של Sawa לשוליים של התרשים.
אופן הכתיבה:

/bic

3. האופציה cp – אופציה זו מוסיפה את הערך של הסטטיסטי C_P של Mallows לשוליים של התרשים.
אופן הכתיבה:

/cp

4. האופציה edf – אופציה זו מוסיפה את מספר דרגות החופש של הטעות לשוליים של התרשים.
אופן הכתיבה:

/edf

5. האופציה gmsep – אופציה זו מוסיפה את תוחלת השגיאה הריבועית של הניבויים לשוליים של התרשים.
אופן הכתיבה:

/gmsep

6. האופציה in – אופציה זו מוסיפה את מספר הנבאים במודל (לא כולל החותך) לשוליים של התרשים.
אופן הכתיבה:

/in

7. האופציה jp – אופציה זו מוסיפה את את הסטטיסטי J_P לשוליים של התרשים.
אופן הכתיבה:

/jp

8. האופציה mse – אופציה זו מוסיפה את תוחלת השגיאה הריבועית לשוליים של התרשים.
אופן הכתיבה:

/mse

9. האופציה np – אופציה זו מוסיפה את מספר הפרמטרים במודל (כולל החותך) לשוליים של התרשים.
אופן הכתיבה:

/np

10. האופציה pc – אופציה זו מוסיפה את הסטטיסטי PC לשוליים של התרשים.
אופן הכתיבה:

/pc

11. האופציה sbc – אופציה זו מוסיפה את הסטטיסטי SBC לשוליים של התרשים.

אופן הכתיבה:

/sbc

12. האופציה sp – אופציה זו מוסיפה את הסטטיסטי S_P לשוליים של התרשים.
אופן הכתיבה:

/sp

13. האופציה sse – אופציה זו מוסיפה את סכום הריבועים של הטעות לשוליים של התרשים.
אופן הכתיבה:

/sse

14. האופציה noline – אופציה זו מבטלת את ההצגה של קווים המוצגים כברירת מחדל בתרשים. קווי ברירת המחדל הם קו הרגרסיה, המוצג כאשר מגדירים תרשים של המשתנה התלוי כנגד המשתנה הבלתי תלוי, וקו ייחוס המוצג כאשר מגדירים תרשים הכולל את טעות הניבוי (residuals).
אופן הכתיבה:

/noline

15. האופציה nomodel – אופציה זו מבטלת את ההצגה של משוואת הרגרסיה מופיעה בראשית התרשים כברירת מחדל.
אופן הכתיבה:

/nomodel

16. האופציה nostat – כברירת מחדל, PROC REG מציגה בשוליים של התרשים את הסטטיסטים N (מספר התצפיות), rsq (R^2 – השונות המוסברת), AdjRsq (Adjusted R^2), ו-RMSE (השורש הריבועי של MSE).
האופציה nostat מבטלת את ההצגה של סטטיסטים אלה בשוליים. עם זאת, האופציה לא מבטלת את ההצגה של סטטיסטים שהוגדרו כאופציות בהוראה PLOT.
אופן הכתיבה:

/nostat

17. האופציה href – אופציה זו מציגה קו ייחוס אנכי לציר y בערכים המוגדרים על ידה.
אופן הכתיבה:

/href = ערכים על ציר x (מופרדים על ידי רווחים)

18. האופציה vref – אופציה זו מציגה קו ייחוס אנכי לציר x בערכים המוגדרים על ידה.
אופן הכתיבה:

/vref = ערכים על ציר y (מופרדים על ידי רווחים)

19. האופציה lhref – אופציה זו מגדירה את סוג הקו המוצג בהגדרת ההוראה href. כברירת מחדל, הקו המוצג הוא קו מקווקו (סוג קו 2). להזכירך, סוג קו 1 הוא קו רציף.
אופן הכתיבה:

/lhref = מספר (סוג הקו)

20. האופציה lvref – אופציה זו מגדירה את סוג הקו המוצג בהגדרת ההוראה vref. כברירת מחדל, הקו המוצג הוא קו מקווקו (סוג קו 2).
אופן הכתיבה:

/lvref = מספר

21. האופציה lline – אופציה זו מגדירה את סוג הקו של הקווים המוצגים בתרשים כברירת מחדל. סוג הקו המוגדר כברירת מחדל לקו זה הוא קו מקווקו (סוג קו 2).
אופן הכתיבה:

/lline = מספר

22. האופציה caxis – אופציה זו מגדירה את הצבע של הצירים, המסגרת, וקווי השנתות של התרשים.
אופן הכתיבה:

/caxis = צבע (באנגלית)

23. האופציה cframe – אופציה זו מגדירה את צבע המילוי של איזור התרשים (האזור התחום על ידי הצירים והמסגרת).
אופן הכתיבה:

/cframe = צבע (באנגלית)

24. האופציה chref – אופציה זו מגדירה את צבע הקו המוצג על ידי האופציה href.
אופן הכתיבה:

/chref = צבע (באנגלית)

25. האופציה ctext – אופציה זו מגדירה את צבע הטקסט שיוצג בתרשים. טקסט זה כולל את התוויות של שנתות הצירים, כותרות הצירים, כותרת המודל ומשוואת המודל, הסטטיסטים המוצגים בשולי התרשים, ומקרא התרשים.
אופן הכתיבה:

/ctext = צבע (באנגלית)

26. האופציה cvref – אופציה זו מגדירה את צבע הקו המוצג על ידי האופציה vref.
אופן הכתיבה:

/cvref = צבע (באנגלית)

27. האופציה modellab – אופציה זו מגדירה את התווית שתוצג עם משוואת המודל. כברירת מחדל, שום תווית לא מוצגת. כמו כן, אם התווית המוגדרת ארוכה מידי (לא נכנסת בשורה אחת), היא לא תוצג.
אופן הכתיבה:

/modellab = 'תווית רצויה'

28. האופציה modelfont – אופציה זו מגדירה את סוג הפונט של תווית המודל ושל משוואת המודל.
אופן הכתיבה:

/modelfont = שם הפונט (באנגלית)

29. האופציה modelht – אופציה זו מגדירה את הגובה של הפונט (הגודל שלו) של תווית המודל ושל משוואת המודל. כברירת מחדל, גובה הפונט מוגדר ל-2. הגדרת גובה פונט ל-0 מבטלת את ההצגה של משוואת המודל (או של תווית המודל במקרה בו היא מוגדרת להצגה).
אופן הכתיבה:

/modelht = מספר שלם (המייצג את גודל הפונט)

30. האופציה statfont – אופציה זו מגדירה את סוג הפונט של הסטטיסטים המוצגים בשולי התרשים.
אופן הכתיבה:

/statfont = שם הפונט (באנגלית)

31. האופציה statht – אופציה זו מגדירה את הגובה של הפונט של הסטטיסטים המוצגים בשולי התרשים. כברירת מחדל, גובה זה מוגדר ל-2.
אופן הכתיבה:

/statht = מספר

32. האופציה name – אופציה זו מגדירה את השם ש-SAS נותנת לתרשים. כברירת מחדל, שם התרשים הוא REG. לכן, בכל פעם שמפיקים יותר מתרשים אחד, SAS נותנת לו את השם REG*n*, ורושמת בחלון Log את ההודעה:

```
NOTE: Graph's name, REG, changed to REG1. REG is already used or not a valid
SAS name.
```

כדי להימנע מהודעה זו, אפשר להגדיר את האופציה name (כאשר זוכרים לתת לכל תרשים שם ייחודי).
אופן הכתיבה:

/name = 'שם התרשים'

33. האופציה overlay – אופציה זו מציגה את כל התרשימים המוגדרים להצגה (תחת אותו מודל) על אותה מערכת צירים. תוויות הצירים נקבעות על פי התרשים הראשון המוגדר להצגה. כאשר מגדירים אופציה זו, קווי ברירת המחדל (כדוגמת משוואת הרגרסיה) לא מוצגים.
אופן הכתיבה:

/overlay

הערה: אופציה זו זמינה גם לתרשימים ברזולוציה נמוכה.

אופציות לתרשימים ברזולוציה נמוכה (המופקים בחלון Ouput):

1. האופציה hplots – אופציה זו מגדירה את מספר עקומות הפיזור שיכולות להיות מוצגות לרוחב הדף. כברירת מחדל, מספר זה מוגדר ל-1.
אופן הכתיבה:

/hplots = מספר שלם גדול מ - 0

2. האופציה vplots – אופציה זו מגדירה את מספר עקומות הפיזור שיכולות להיות מוצגות לאורך הדף. כברירת מחדל, מספר זה מוגדר ל-1.
אופן הכתיבה:

/vplots = מספר שלם גדול מ-0

דוגמא:

```
plot x1 * y1 x2 * y2 x3 * y3 x4 * y4/vplots = 2 hplots = 2;
```

בדוגמא זו הגדרנו ש-SAS תציג 4 עקומות פיזור באותו עמוד: 2 עקומות לאורך, ו-2 עקומות לרוחב.

3. האופציה symbol – אופציה זו מגדירה את התווית של הנקודות המייצגות תצפיות בתוך התרשים. כאשר לא מגדירים אופציה זו, ברירת המחדל היא להציג '1' עבור נקודות המייצגות מיקום עם תצפית אחת, '2' עבור נקודות המייצגות מיקום עם 2 תצפיות וכך הלאה. כאשר יש 10 תצפיות או יותר, ברירת המחדל היא להציג '*'.
אופן הכתיבה:

/symbol = 'תו'

דוגמא:

```
plot x * y/symbol = '2';
```

הערה: כאשר מגדירים סמל לעקומה בתוך ההוראה PLOT, הגדרה זו "דורסת" את האופציה symbol.

דוגמא:

```
plot x * y z * t = 'f'/symbol = '2';
```

בדוגמא זו, בעקומת הפיזור של x ו-y כל מיקום בתרשים ייוצג בספרה 2, בעוד שבעקומת הפיזור של z ו-t, כל מיקום בתרשים ייוצג על ידי האות f.

ההוראה OUTPUT

הוראה זו יוצרת קובץ נתונים חדש השומר את המדדים שחושבו על ידי PROC REG בהרצת המודל. כדי להפיק קובץ נתונים, יש להגדיר לפחות מדד אחד לשמירה לקובץ. קובץ הנתונים החדש יכיל את כל הנתונים המקוריים (בהם נעשה שימוש לבחינת המודל), וכן את כל המדדים שהוגדרו לשמירה בהוראה OUTPUT.

אופן הכתיבה:

OUTPUT <out = שם קובץ נתונים> שם משתנה = מילת מפתח <...שם משתנה = מילת מפתח>;

החלק העליון של טבלה 10 מציג את מילות המפתח הסטטיסטיות הזמינות ל-PROC REG

דוגמא:

```
output out = dogma lcl = lower residual = ped_error;
```

אופציות של ההוראה OUTPUT

1. האופציה out – אופציה זו מגדירה את שם קובץ הנתונים שנוצר על ידי PROC REG.
אופן הכתיבה:

OUTPUT out = שם קובץ נתונים חדש

2. מילות מפתח סטטיסטיות – באמצעות מילות מפתח סטטיסטיות (המפורטות בטבלה 10).
אופן הכתיבה:

שם משתנה = מילת מפתח סטטיסטית

בהוראה OUTPUT של PROC REG, חייבים לכלול לפחות מילת מפתח אחת.

PROC ANOVA

הפרוצדורה ANOVA מבצעת ניתוחי שונות (ANOVA – analysis of variance), לנתונים מאוזנים בלבד (נתונים עם מספר שווה של תצפיות לכל שילוב של תנאי הניסוי).

באופן עקרוני, PROC GLM (שתדון בהמשך הפרק) עדיפה על PROC ANOVA בכל ניתוח (לרבות ניתוחי שונות). אולם, PROC GLM דורשת יותר משאבי עיבוד. לכן, כאשר מדובר על נתונים מאוזנים, עדיף להשתמש ב-PROC ANOVA.

אופן הכתיבה:

PROC ANOVA;
BY <descending> משתנה1…<descending> *n* משתנה <notsorted>;
CLASS משתנים;
MODEL משתנים תלויים = אפקטים </אופציות שונות>;
ABSORB משתנים;
MANOVA <אפשרויות מבחן> </אופציות שונות>;
MEANS אפקטים</אופציות שונות>;
REPEATED הפקטור של ספציפיקציות</אופציות שונות>;
RUN;

עם זאת, רק ההוראות CLASS ו-MODEL הן הוראות חובה ב-PROC ANOVA (למעט מקרים בהם לא מגדירים משתנים בלתי תלויים, ואז מספיקה ההוראה MODEL).

דוגמא:

```
proc anova;
 class  phase;
 model s1 = phase;
run;
```

הפלט הבסיסי המתקבל מהרצת מודל ANOVA כולל נתונים על משתנה ה-CLASS, טבלת ANOVA, R^2, ונתונים על המשתנה התלוי:

```
                          The SAS System
                        The ANOVA Procedure

                   Class Level Information

          Class         Levels    Values

          phase              2       1 2

        Number of Observations Read          40
        Number of Observations Used          40
```

Dependent Variable: s1

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	7.52125007	7.52125007	303.92	<.0001
Error	38	0.94038882	0.02474707		
Corrected Total	39	8.46163889			

R-Square	Coeff Var	Root MSE	s1 Mean
0.888864	32.88753	0.157312	0.478333

Source	DF	Anova SS	Mean Square	F Value	Pr > F
phase	1	7.52125007	7.52125007	303.92	<.0001

ההוראה BY

הוראה BY אומרת ל-PROC ANOVA לבצע ניתוח שונות נפרד לכל קבוצת ערכים זהים של המשתנה המוגדר על ידי ההוראה.

אופן הכתיבה:

BY <descending> משתנה1 <descending>…משתנה *n* <notsorted>;

ההוראה CLASS

ההוראה CLASS מגדירה את המשתנה או המשתנים הקטגוריאליים (לדוגמא, תנאי, מין, גזע) שיוגדרו כאפקטים (משתנים בלתי תלויים) במודל. ההוראה CLASS היא הוראת חובה ב-PROC ANOVA, והיא חייבת להופיע לפני ההוראה MODEL (שתידון להלן).

אופן הכתיבה:

CLASS שמות משתנים;

ההוראה MODEL

ההוראה MODEL מגדירה את המשתנה התלוי והאפקטים (המשתנים) הבלתי תלויים של מודל ה - ANOVA. כל המשתנים המוגדרים כאפקטים חייבים להופיע בהוראה CLASS, שכן PROC ANOVA לא תומכת בניתוח אפקטים למשתנים רציפים.

כדי להגדיר אינטראקציה בין אפקטים, יש לרשום את הסימן כוכבית (*) בין שני המשתנים הרצויים. במקרה בו לא מגדירים אף אפקט, הניתוח מחשב רק אמד לחותך. יש להשתמש באפשרות זאת כאשר רוצים לבדוק את ההשערה שהממוצע של המשתנה התלוי שונה במובהק מאפס.

אופן הכתיבה:

MODEL משתנים תלויים = אפקטים</אופציות שונות>;

1. האופציה intercept – אופציה זו מציגה את תוצאות ניתוח ההשערה שהחותך הוא אפקט בפני עצמו במודל.
כברירת מחדל, PROC ANOVA כוללת את החותך בניתוח המודל, אבל לא מציגה את הניתוחים הקשורים לכך.
אופן הכתיבה:

/intercept

הערה: כאשר מגדירים את ההוראה ABSORB, האופציה intercept מתבטלת (למעט העובדה שסכום הריבועים הלא מתוקנן של החותך מוצג).

2. האופציה nouni – אופציה זו אומרת ל-PROC ANOVA לא להפיק את הפלט של הניתוח ANOVA. אופציה זו שימושית מאוד כאשר משתמשים בהוראות MULTIVATIATE או REPEATED, בהן אנו לרוב לא נהיה מעוניינים בפלט הסטנדרטי.
אופן הכתיבה:

/nouni

טיפ קריאה: ההוראה ABSORBED לא ממולצת לקריאה לקוראים ללא ידע נרחב בסטטיסטיקה.

ההוראה ABSORB

ההוראה ABSORB מגדירה משתנה או משתנים שינותחו במודל בטכניקה הקרויה Absorption. טכניקה זו חוסכת בזמן וזיכרון עיבוד, והיא נועדה לטפל בעיקר במשתנים קטגוריאליים בעלי מספר גדול של רמות, או במשתנים שלא מניחים שהם מקיימים אינטראקציות עם משתנים אחרים (משתנים שכן מקיימים אינטראקציות). במצב בו מגדירים משתני ABSORB, PROC ANOVA לא תחשב למשתנים אלה את הסכומים הריבועיים מסוג II (הירידה בטעות הניבוי כתוצאה מהוספת המשתנה למודל, בהנחה ששאר המשתנים בפנים), III (התרומה היחסית של המשתנה לשונות המוסברת), ו-IV (דומה לסוג III, רק מחושב במצב בו הנתונים כוללים ערכים חסרים). בנוסף, משתנים המוגדרים על ידי ההוראה ABSORB מנותחים בדומה לניתוח מקונן (היררכי).

כאשר מגדירים משתני ABSORB, אין להגדיר אותם בהוראה CLASS, ולא בהוראה MODEL, בתור מתשנים בלתי תלויים (אפקטים).

אופן הכתיבה:

ABSORB שמות משתנים;

הערה: כאשר מגדירים משתנה ABSORB, יש למיין את קובץ הנתונים על פי משתנה זה. במקרה שההוראה BY מוגדרת גם, יש למיין את הקובץ תחילה על פי משתנה ה-BY ואחר כך לפי משתנה ה-ABSORB.

ההוראה MANOVA

כאשר מגדירים בהוראה MODEL יותר ממשתנה תלוי אחד, ניתן לבצע ניתוח רב משתני (Multivariate analysis) באמצעות ההוראה MANOVA.

אופן הכתיבה:

MANOVA <אופציות מבחן></אופציות שונות>;

אופציות מבחן

אופציות מבחן בהוראה MANOVA מגדירות את האפקטים אותם יש לבחון.

1. האופציה h – אופציה זו מגדירה את האפקטים בהם יש להשתמש במטריצות ההשערות (מטריצות סכום הריבועים והמכפלות הוקטוריות – SSCP matrices).
אופן הכתיבה:

h = אפקטים | intercept | _ALL_

הפקודה intercept נועדה להמציא מבחן לחותך, והפקודה _ALL_ נועדה להגדיר את כל האפקטים המוגדרים בהוראה MODEL

2. האופציה e – אופציה זו מגדירה את אפקט הטעות. במקרה שאפקט זה לא מוגדר, PROC ANOVA משתמשת במטריצת הטעות (residual SSCP) שהופקה מתוך הניתוח.
אופן הכתיבה:

e = אפקט

דוגמא:

```
e=B(A)
```

כאשר A ו-B הם משתני CLASS.

אופציות של ההוראה MANOVA

האופציות של ההוראה MANOVA נקראות detail-options והן מגדירות כיצד לבצע את הניתוחי MANOVA ואיזה תוצאות יש להציג.

1. האופציה canonical – אופציה זו מגדירה ל-PROC ANOVA להפיק ניתוח קאנוני (canonical) למטריצות של H ו-E.
אופן הכתיבה:

/canonical

2. האופציה printe – אופציה זו אומרת ל-PROC ANOVA להפיק פלט של מטריצת הטעויות E.
אופן הכתיבה:

/printe

3. האופציה printh – אופציה זו אומרת ל-PROC ANOVA להפיק פלט של מטריצת ההשערות H, הקשורה לכל אפקט המוגדר על ידי אופציית המבחן H.
אופן הכתיבה:

/printh

4. האופציה summary – אופציה זו מפיקה טבלת ניתוח שונויות (analysis-of-variance) לכל אחד מהמשתנים התלויים.
אופן הכתיבה:

/summary

ההוראה MEANS

ההוראה MEANS מחשבת ממוצעים למשתנים התלויים המוגדרים על ידה עבור כל אפקט (משתנה בלתי תלוי) המוגדר בהוראה MODEL.

אופן הכתיבה:

MEANS אפקטים</אופציות שונות>;

דוגמא:

```
proc anova;
 class phase1 phase2;
 model s1 = phase1 phase2 phase1 * phase2;
 means phase1;
 means phase1 * phase2;
run;
```

ניתן לכלול מספר בלתי מוגבל של הוראות MEANS, בתנאי שהוראות אלה יהיו כתובות לאחר ההוראה MODEL.
בדוגמא שלהלן, הקוד אומר ל-PROC ANOVA להפיק טבלת ממוצעים של המשתנה התלוי (s1) בנפרד לכל ערך של המשתנה phase1 (בהוראה MEANS הראשונה), ולהפיק טבלת ממוצעים של המשתנה התלוי בנפרד עבור כל שילוב אפשרי של המשתנה phase1 והמשתנה phase2. עם זאת, ניתן גם להגדיר את אותן פעולות באמצעות הוראת MEANS אחת:

```
means phase1 phase1 * phase2;
```

הפלט שיופק בעקבות ההוראות MEANS הללו הוא:

```
                    The ANOVA Procedure

Level of                 ---------------s1--------------
phase1          N              Mean           Std Dev

1               6        65.1666667        5.63619257
2               6        90.6666667        6.97614985

                    The ANOVA Procedure

Level of     Level of              ---------------s1--------------
phase1       phase2          N              Mean           Std Dev

1            1               3        68.0000000        7.00000000
1            2               3        62.3333333        2.51661148
2            1               3        92.3333333        6.11010093
2            2               3        89.0000000        8.71779789
```

השימוש בהוראה MEANS מאפשר לעשות בחינה לא רק של המודל עצמו, אלא גם להשוות בין קבוצות שונות הכלולות בו.

אופציות של ההוראה MEANS מאפשרות לבצע השוואות מרובות בין ערכים שונים ו/או שילובים שונים של הערכים של המשתנים הבלתי תלויים. השוואות מרובות ב-PROC ANOVA ניתנות לביצוע רק על אפקטים עיקריים במודל (ולא על אינטראקציות). השוואות מרובות על אינטראקציות ניתנות לביצוע ב-PROC GLM (שתידון בתת הפרק הבא).

1. האופציה alpha – אופציה זו מגדירה את רמת המובהקות של ההשוואות בין הממוצעים. כברירת מחדל, alpha = 0.05.
   אופן הכתיבה:

/alpha = ערך מספרי כלשהו בין 0 ל- 1

2. האופציה hovtest – אופציה זו מבקשת מ-PROC ANOVA לבצע מבחן שוויון שונויות (מבחן לוין) עבור הקבוצות השונות המוגדרות על ידי ההוראה MEANS.
   אופן הכתיבה:

/hovtest

3. האופציה bon – אופציה זו מבצעת מבחני t של Bonferroni להבדלים בין הממוצעים עבור כל האפקטים העיקריים המופיעים בהוראה MEANS.
   אופן הכתיבה:

/bon

4. האופציה Duncan – אופציה זו מבצעת מבחן Duncan עבור כל האפקטים העיקריים המופיעים בהוראה MEANS.
   אופן הכתיבה:

/duncan

5. האופציה dunnett – אופציה זו מבצעת מבחן Dunnett (מבחן t דו-זנבי), כדי לבחון האם טיפול כלשהו (ערך של משתנה בלתי תלוי) שונה מקבוצת ביקורת בודדת. מבחן זה נעשה עבור כל אפקט עיקרי במודל המוגדר באמצעות ההוראה MEANS.
   אופן הכתיבה:

/dunnett

6. האופציה scheffe – אופציה זו מבצעת את פרוצדורת scheffe להשוואות מרובות על כל האפקטים העיקריים המופיעים בהוראה MEANS.
   אופן הכתיבה:

/scheffe

7. האופציה lsd – אופציה זו מבצעת מבחני t מזווגים (מקביל למבחן פישר) לכל האפקטים העיקריים המוגדרים על ידי ההוראה MEANS.
   אופן הכתיבה:

/lsd

8. האופציה tukey – אופציה זו מבצעת מבחן Tukey עבור כל האפקטים העיקריים המופיעים בהוראה MEANS.
אופן הכתיבה:

/Tukey

9. האופציה duncan – אופציה זו מבצעת מבחן Duncan עבור כל האפקטים העיקריים המופיעים בהוראה MEANS.
אופן הכתיבה:

/duncan

ההוראה REPEATED

כאשר הערכים של המשתנה התלוי בהוראה MODEL מייצגים מדידות חוזרות באותו מערך ניסוי (למשל כאשר מודדים ביצוע של משימה ספציפית תחת תנאים שונים או כאשר מבצעים מדידה של אותו משתנה בזמנים שונים), ההוראה REPEATED מאפשרת לבחון השערות לגבי פקטורים (אפקטים) תוך-נבדקיים, וכן אינטראקציות של פקטורים תוך נבדקיים עם משתנים בין נבדקיים.

ההוראה REPEATED דורשת שלכל ערך של המשתנה התלוי יהיה ערך בכל המשתנים הבלתי תלויים המופיעים במודל. אחרת, ההוראה תתייחס לכל תצפית בה אין ערך לכל אחד מהרמות של המשתנים הבלתי תלויים כתצפית חסרה.

אופן הכתיבה:

REPEATED הגדרת פקטורים </אופציות שונות>;

דוגמא:

```
proc anova;
 model RT1-RT4 = /nouni;
 repeated Time 4 (1 2 3 4);
run;
```

בדוגמא זו PROC ANOVA תפיק ניתוח מדדים חוזרים לפקטור בשם Time (המייצג מדידת זמן תגובה תחת 4 תנאים שונים) בעל 4 רמות (1, 2, 3, ו-4).

הגדרת פקטורים

בהגדרת הפקטורים בהוראה REPEATED ניתן לרשום מספר הגדרות של פקטורים, מופרדות על ידי פסיקים, כאשר הגדרת הפקטורים יכולה לכלול את הנתונים הבאים:

<טרנספורמציות> < (ערכי-רמות)> מספר-רמות שם-הפקטור

<u>שם-הפקטור</u> הוא שם הניתן על ידי המשתמש כדי לקשר אותו למשתנים התלויים (ולכן השם צריך לייצג את מה שרוצים לבדוק באמצעות הניתוח). השם יכול להיות כל שם חוקי למשתנים בתוכנת SAS, אך הוא לא יכול להיות שם של משתנה שכבר קיים בקובץ הנתונים.

<u>מספר-רמות</u> מציין את מספר הרמות הקשורות לפקטור המוגדר בשם-הפקטור. במצב בו קיים רק פקטור תוך-נבדקי אחד, מספר הרמות יהיה שווה למספר המשתנים התלויים, ולכן מצב כזה אין צורך להגדיר את מספר הרמות. עם זאת, בשאר המצבים חובה להגדיר מספר-רמות, ומספר הרמות הכללי (של כל הפקטורים המוגדרים), חייב להיות שווה למספר המשתנים התלויים המוגדרים בהוראה MODEL.

(ערכי-רמות) מגדיר ערכים התואמים לרמות של הפקטור הנבחן בניתוח של המדדים החוזרים (Repeated). מספר הרמות המוגדרות חייב להיות תואם למספר הרמות של הפקטור המוגדר בהוראה REPEATED. את ערכי הרמות יש לכתוב בתוך סוגריים אחרי מספר הרמות. עם זאת, הגדרה של ערכי רמות היא לא חובה.

טרנספורמציות מגדירות קונטרסטים (בעלי דרגת חופש אחת) עבור הפקטורים המוגדרים בהוראה REPEATED.
הטרנספורמציות הזמינות ב-PROC ANOVA הן:

1. Contrast – טרנספורמציה המייצרת קונטרסטים בין רמות שונות של הפקטור.
אופן הכתיבה:

contrast <(רמת-פקטור-להשוואה)>

דוגמא:

```
contrast(1)
```

בדוגמא זו, PROC ANOVA תפיק קונטרסט בין הרמה הראשונה של הפקטור לבין שאר הרמות שלו.

2. Mean – טרנספורמציה המייצרת קונטרסטים בין רמות של הפקטור לבין הממוצע של כל שאר הרמות של הפקטור.
אופן הכתיבה:

mean <(רמת-פקטור-להשוואה)>

3. Profile – טרנספורמציה המייצרת קונטרסטים בין רמות סמוכות של הפקטור.
אופן הכתיבה:

profile

4. Helmert – טרנספורמציה המייצרת קונטרסטים בין כל רמה של הפקטור לממוצע של הרמות העוקבות.
אופן הכתיבה:

helmert

דוגמא:

```
proc anova;
 model disrt1 simrt1 = / nouni;
 repeated similarity 2 contrast(1);
run;
```

בדוגמא זו PROC ANOVA תפיק ניתוח מדדים חוזרים לפקטור בשם similarity בעל 3 רמות, ותבדוק את ההבדל בין הרמה הראשונה של הפקטור לבין 2 הרמות הנותרות של הפקטור.

הפלט המופק כתוצאה מהרצת דוגמא זו הוא:

```
                    The ANOVA Procedure
          Repeated Measures Analysis of Variance

            Repeated Measures Level Information

          Dependent Variable      disrt1   simrt1

         Level of similarity           1        2
```

MANOVA Test Criteria and Exact F Statistics for the Hypothesis of no similarity Effect
H = Anova SSCP Matrix for similarity
E = Error SSCP Matrix

S=1 M=-0.5 N=16

Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.87176663	5.00	1	34	0.0320
Pillai's Trace	0.12823337	5.00	1	34	0.0320
Hotelling-Lawley Trace	0.14709599	5.00	1	34	0.0320
Roy's Greatest Root	0.14709599	5.00	1	34	0.0320

The ANOVA Procedure
Repeated Measures Analysis of Variance
Univariate Tests of Hypotheses for Within Subject Effects

Source	DF	Anova SS	Mean Square	F Value	Pr > F
similarity	1	147.530424	147.530424	5.00	0.0320
Error(similarity)	34	1002.953429	29.498630		

אופציות של ההוראה REPEATED

1. האופציה canonical – אופציה זו מגדירה ל-PROC ANOVA להפיק ניתוח קאנוני (canonical) למטריציות של H ו-E.
אופן הכתיבה:

/canonical

2. האופציה nom – אופציה זו מציגה רק את התוצאות של הניתוח univariate.
אופן הכתיבה:

/nom

3. האופציה nou – אופציה זו מציגה רק את התוצאות של הניתוח multivariate.
אופן הכתיבה:

/nou

4. האופציה summary – אופציה זו אומרת ל-PROC ANOVA להפיק טבלת ניתוח שונות לכל קונטרסט המוגדר על ידי הפקטור הבין-נבדקי.
אופן הכתיבה:

/summary

האופציה summary תפיק, בנוסף לפלט הסטנדרטי של ההוראה REPEATED, גם את הפלט הבא:

similarity_N represents the contrast between the nth level of similarity and the last

Contrast Variable: similarity_1

Source	DF	Anova SS	Mean Square	F Value	Pr > F
Mean	1	99.4856835	99.4856835	4.37	0.0442
Error	34	774.3811113	22.7759150		

Contrast Variable: similarity_2

Source	DF	Anova SS	Mean Square	F Value	Pr > F
Mean	1	51.884423	51.884423	0.95	0.3359
Error	34	1851.145267	54.445449		

Contrast Variable: similarity_3

Source	DF	Anova SS	Mean Square	F Value	Pr > F
Mean	1	147.017251	147.017251	4.91	0.0335
Error	34	1017.988597	29.940841		

בדוגמא זו קיימות 3 רמות לפקטור.

PROC GLM

הפרוצדורה GLM עושה שימוש בשיטת הריבועים הפחותים כדי להתאים מודלים ליניאריים כלליים (General linear models), הבאים להסביר את הקשר בין משתנה תלוי רציף (או משתנים) לבין משתנה בלתי תלוי (או משתנים).

מאחר ו-GLM מתאים לניתוח של משתנים קטגוריאליים ומשתנים רציפים, PROC GLM מסוגלת לבצע מגוון רחב של ניתוחים, כגון:

1. מגוון ניתוחי רגרסיה, כגון רגרסיה פשוטה, רגרסיה מרובה, רגרסיה פולינומית
2. ניתוח שונות (ANOVA) – כולל ניתוח לנתונים לא מאוזנים (ללא מספר שווה של תצפיות בכל קבוצה)
3. ANCOVA
4. MANOVA
5. Repeated measures ANOVA
6. מתאמים חלקיים

אופן הכתיבה:

```
PROC GLM <אופציות שונות>;
CLASS משתנים;
MODEL משתנים תלויים = משתנים בלתי תלויים </ אופציות שונות>;
ABSORB משתנים;
BY משתנים;
ID משתנים;
LSMEANS אפקטים </אופציות>;
MANOVA <test אופציות> </detail אופציות>;
MEANS אפקטים </אופציות>;
OUTPUT <OUT = שם קובץ נתונים> שם = מילת מפתח...> שם = מילת מפתח <שם </אופציות>;
RANDOM אפקטים </אופציות>;
REPEATED הגדרת פקטורים </אופציות>;
RUN;
```

עם זאת, רק ההוראה MODEL היא הוראת חובה ב-PROC GLM.

דוגמא:

```
proc glm;
 model s1 = phase1;
run;
```

הפלט הבסיסי המתקבל נתונים על המודל, טבלת ANOVA, R^2, ונתונים על המשתנה התלוי והבלתי תלוי:

```
                                The GLM Procedure
Dependent Variable: s1

                                           Sum of
Source                      DF         Squares     Mean Square    F Value    Pr > F
Model                        1     1950.750000     1950.750000      48.51    <.0001
Error                       10      402.166667       40.216667
Corrected Total             11     2352.916667

                  R-Square     Coeff Var      Root MSE       s1 Mean
                  0.829077      8.139030      6.341661      77.91667

Source                      DF       Type I SS     Mean Square    F Value    Pr > F
phase1                       1     1950.750000     1950.750000      48.51    <.0001

Source                      DF     Type III SS     Mean Square    F Value    Pr > F
phase1                       1     1950.750000     1950.750000      48.51    <.0001

                                          Standard
          Parameter         Estimate           Error    t Value    Pr > |t|
          Intercept      39.66666667      5.78911814       6.85      <.0001
          phase1         25.50000000      3.66135980       6.96      <.0001
```

אופציות של PROC GLM

1. אופציה alpha – אופציה זו מגדירה את רמת המובהקות p של המודל. ערך זה משמש גם להגדרת רווח הסמך בהוראות LSMEANS, MEANS, MODLE ו-OUTPUT.
   אופן הכתיבה:

/alpha = ערך מספרי בין 0 ל-1

הערה: ניתן להגדיר גם ערך alpha בנפרד עבור כל הוראה.

2. האופציה data – אופציה זו מגדירה את קובץ הנתונים עליו PROC GLM תבצע את הניתוח.
   אופן הכתיבה:

/data = שם קובץ נתונים

3. האופציה namelen – אופציה זו מגדירה את האורך של שמות האפקטים בטבלאות ובקובץ הפלט. כברירת מחדל, אורך זה מוגדר ל-20 תווים.
   אופן הכתיבה:

/namelen = מספר שלם בין 20 ל - 200

4. האופציה noprint – אופציה זו מבטלת את הפלט הבסיסי המופק על ידי PROC GLM. אופציה זו שימושית במקרה בו רוצים להפיק את תוצאות הניתוח לקובץ נתונים.
   אופן הכתיבה:

/noprint

5. האופציה outstat – אופציה זו יוצרת קובץ נתונים של SAS המכיל את הסכומים הריבועיים, דרגות החופש, הסטטיסטי F, ורמות המבוהקות של כל האפקטים המוגדרים במודל.
   אופן הכתיבה:

/outstat = שם של קובץ נתונים

ההוראה MODEL

ההוראה MODEL מגדירה את המשתנה התלוי ואת האפקטים הבלתי תלויים של ה-GLM. ניתן להגדיר רק הוראת MODEL אחת בכל פרוצדורה.

אופן הכתיבה:

MODEL משתנים תלויים = משתנים בלתי תלויים </אופציות שונות>;

אופציות של ההוראה MODEL

1. האופציה intercept – אופציה זו מציגה את תוצאות ניתוח ההשערה שהחותך הוא אפקט בפני עצמו במודל. כברירת מחדל, PROC GLM כוללת את החותך בניתוח המודל, אבל לא מציגה את הניתוחים הקשורים לכך.
   אופן הכתיבה:

/intercept

2. האופציה noint – אופציה זו משמיטה את החותך מהמודל.
   אופן הכתיבה:

/noint

3. האופציה nouni – אופציה זו אומרת ל-PROC ANOVA לא להפיק את הפלט של הניתוח ANOVA. אופציה זו שימושית מאוד כאשר משתמשים בהוראות MULTIVATIATE או REPEATED, בהן אנו לרוב לא ניהיה מעוניינים בפלט הסטנדרטי.
   אופן הכתיבה:

/nouni

4. האופציה clm – אופציה זו מפיקה רווח סמך לממוצע של הערכים המנובאים של המודל.
   אופן הכתיבה:

/clm

5. האופציה ss1 – אופציה זו מוסיפה לקובץ הפלט את הסכומים הריבועיים מסוג I למודל.
   אופן הכתיבה:

/ss1

6. האופציה ss2 – אופציה זו מוסיפה לקובץ הפלט את הסוכמים הריבועיים מסוג II למודל.
   אופן הכתיבה:

/ss2

7. האופציה ss3 – אופציה זו מוסיפה לקובץ הפלט את הסוכמים הריבועיים מסוג III למודל.
   אופן הכתיבה:

/ss3

8. האופציה ss4 – אופציה זו מוסיפה לקובץ הפלט את הסוכמים הריבועיים מסוג IV למודל.
אופן הכתיבה:

/ss4

ההוראה LSMEANS

ההוראה LSMEANS מחשבת ממוצע ריבועים פחותים (LS means) לכל אפקט במודל המוגדר על ידי ההוראה. בהוראה זו ניתן להגדיר רק משתני CLASS (משתנים המוגדרים בהוראה CLASS). בנוסף, הוראה זו מבצעת, בנוסף להשוואות של אפקטים עיקריים, גם השוואות בין אינטראקציות (כאשר אינטראקציות מסומנות על ידי כוכבית בין שמות המשתנים).

אופן הכתיבה:

LSMEANS אפקטים </אופציות שונות>;

דוגמא:

```
proc glm data=anv;
 class phase1 phase2;
 model s1 = phase1 phase2 phase1 * phase2 /nouni;
 lsmeans  phase1 phase2 phase1 * phase2;
run;
```

הרצת הקוד תגרום להפקת הפלט הבא:

```
                  Least Squares Means
                 phase1       s1 LSMEAN
                 1           65.1666667
                 2           90.6666667
                 phase2       s1 LSMEAN
                 1           80.1666667
                 2           75.6666667
phase1    phase2       s1 LSMEAN
1         1           68.0000000
1         2           62.3333333
2         1           92.3333333
2         2           89.0000000
```

ההוראה OUTPUT

ההוראה OUTPUT יוצרת קובץ נתונים חדש השומר את המדדים שחושבו על ידי PROC GLM בהרצת המודל. כדי להפיק קובץ נתונים, יש להגדיר לפחות מדד אחד לשמירה לקובץ. קובץ הנתונים החדש יכיל את כל הנתונים המקוריים (בהם נעשה שימוש לבחינת המודל), וכן את כל המדדים שהוגדרו לשמירה בהוראה OUTPUT.

אופן הכתיבה:

OUTPOT <out = שם קובץ נתונים> מילת מפתח = שם משתנה <...מילת מפתח = שם משתנה </אופציות שונות>;

טבלה 11 מציגה את מילות המפתח הסטטיסטיות הזמינות ל-PROC GLM.

סטטיסטים להפקה לקובץ פלט ב-PROC GLM	
מילת מפתח	**תיאור**
coockd	הסטטיסטי Cook's D
covratio	ההשפעה הסטנדרטית של התצפיות על השונות המשותפת של אמדי הפרמטרים
dffits	ההשפעה הסטנדרטית של התצפיות על הערכים המנובאים
h	$h_i = x_i\,(\mathbf{X}'\mathbf{X})^{-1}\,x_i'$
lcl	הגבול התחתון של רווח הסמך לניבוי ספציפי.
lclm	הגבול התחתון של רווח הסמך לערך הממוצע של הניבויים
predicted	הערכים המנובאים על ידי המודל
press	טעות הניבוי מהתאמה מחדש של המודל כאשר מוחקים תצפיות מסוימות
residual	טעות הניבוי
rstudent	טעות ניבוי מתקוננת מהתאמה מחדש של המודל כאשר מוחקים תצפיות מסוימות
stdi	טעות תקן של ערך מנובא ספציפי
stdp	טעות תקן של הערך המנובא הממוצע
stdr	טעות התקן של טעות הניבוי
student	טעות הניבוי חלקי טעות התקן
ucl	הגבול העליון של רווח הסמך לניבוי ספציפי.
uclm	הגבול העליון של רווח הסמך לערך הממוצע של הניבויים

טבלה 11 - רשימת סטטיסטים הניתנים להפקה לקובץ פלט על ידי ההוראה OUTPUT ב-PROC GLM

אופציות של ההוראה OUTPUT

1. האופציה out – אופציה זו מגדירה את שם קובץ הנתונים שנוצר על ידי PROC GLM.
אופן הכתיבה:

OUTPUT out = שם קובץ נתונים חדש

2. מילות מפתח סטטיסטיות – באמצעות מילות מפתח סטטיסטיות (המפורטות בטבלה 11).
אופן הכתיבה:

שם משתנה = מילת מפתח סטטיסטית

בהוראה OUTPUT של PROC GLM, חייבים לכלול לפחות מילת מפתח אחת.

3. האופציה alpha – אופציה זו מגדירה את רמת המובהקות של רווחי הסמך שיופקו לקובץ הנתונים.
אופן הכתיבה:

alpha = מספר בין 0 ל- 1

ההוראה RANDOM

ההוראה RANDOM מאפשרת להגדיר אפקט אקראי (דהיינו, אפקט הנדגם אקראית ממדגם של אפקטים המתפלג נורמאלית), ולא אפקט מוגדר מראש. ההוראה RANDOM חייבת להופיע לאחר ההוראה MODEL. ניתן להגדיר מספר בלתי מוגבל של הוראות RANDOM בפרוצדורה אחת. עם זאת, ניתן לכלול בהוראה RANDOM רק משתני CLASS.

אופן הכתיבה:

RANDOM אפקטים </אופציות שונות>;

דוגמא:

```
proc glm data=anv;
  class phase1 phase2;
  model s1 = phase1 phase2 phase1 * phase2;
  random phase1 phase2;
run;
```

ההוראות ABSORBED, BY, ID, MANOVA, MEANS, REPEATED

הגדרת הוראות אלה ב-PROC GLM, כמו גם התפקוד והמטרה שלהן זהים להגדרה, למטרה ולתפקוד של הוראות אלה ב-PROC ANOVA. לכן לא נרחיב את הדיבור על הוראות אלה בתת פרק זה. קורא המעוניין לרענן את הזיכרון מוזמן לפנות לתת הפרק הדן ב-PROC ANOVA.

תרגול עצמי – מודלים ליניארים

תרגיל 33

עשרה נבדקים נתבקשו לבחור בכל יום, למשך תקופה של 10 ימים, האם להשקיע במנייה של חברה מסויימת או לא. להלן אחוז הימים בהם כל נבדק בחר להשקיע במנייה:

Sub	choice
1	0.6
2	0.4
3	0.7
4	0.8
5	0.5
6	0.3
7	0.6
8	0.5
9	0.8
10	0.9

בדוק האם ניתן לומר על סמך הנתונים כי לנבדקים הייתה העדפה להשקיע במנייה (כך שבאופן מובהק, הם בחרו להשקיע ביותר מ-50% מהמקרים).

תרגיל 34

דיקן הפקולטה למשפטים רצה לבחון את ההשפעה של שתי שיטות לימוד שונות על ציוני הסיום של התלמידים. קובץ הנתונים שלהלן כולל את ציוני הסיום של שתי כיתות שונות, כאשר כל כיתה עברה תהליך למידה בשיטה שונה (כל עמודה מכילה ציונים של כיתה אחת):

75 89
80 67
94 80
77 85
90 82
78 83
90 68

80 92
66 100
56 87
64 78
69 95
50 86

בדוק האם יש הבדל בציוני הכיתות השונות. בהתבסס על תוצאות בדיקה זו, מה תמליץ לדיקן הפקולטה?

תרגיל 35

נתון קובץ נתונים המכיל מידע על סטודנטים בפקולטה לסוציולגיה. הקובץ כולל נתונים על מין הסטודנט (gender, זכר = 0, נקבה = 1), גיל הסטודנט (age), מסלול הלימוד (maslul), תואר הלימוד (toar, תואר ראשון = 1, תואר שני = 2), ממוצע הציונים בתואר (memutza), ציון הפסיכומטרי (psycho), ציון בקורס מבוא לסוציולוגיה (tzion) וציון בסולם מיקום שליטה (locus):

1	27	14	2	92	727	95	11
1	27	14	2	.	.	94	9
0	26	4	1	79	598	83	9
0	25	6	1	88	700	100	13
0	27	4	1	75	644	85	5
0	25	.	1	87	680	91	0
1	25	3	1	79	687	90	10
1	24	4	1	83	664	90	9
0	25	14	2	90	700	95	15
0	25	1	1	86	685	94	11
1	25	3	1	76.9	712	90	1
1	25	4	1	76	721	87	8
1	21	3	1	80	647	95	7
1	21	3	1	78	632	90	14
0	26	1	1	87	747	90	12
1	21	3	1	83	725	90	10
0	25	10	1	84	650	88	16
0	25	4	1	81	660	88	8
1	25	1	1	82	706	90	9
1	25	10	1	81	639	85	12
0	29	8	1	76	647	85	8
0	28	1	1	79	698	90	9
1	23	10	1	82	669	90	6
0	25	4	1	83	658	94	9
0	27	3	1	81	721	93	14
0	24	4	1	82	694	90	16
1	25	1	1	86	718	93	4
1	28	1	2	86	620	90	7
0	29	1	1	81	686	90	12

0	27	4	1	80	632	97	8
1	24	3	1	84	750	90	4
0	25	3	1	90	626	100	12
0	26	3	1	91	710	100	8
0	22	4	1	82	716	87	8
1	22	4	1	76	640	95	7
1	34	14	2	75	697	95	6
1	32	3	2	83	670	85	9
1	21	3	1	82	642	96	4
1	25	11	1	79	680	95	11
0	23	1	1	78	657	80	9
0	24	1	1	76	590	80	13
0	22	1	1	78	692	80	10
0	22	4	1	83	624	90	11
1	21	4	1	77	625	93	11
1	24	1	1	84	676	90	12
0	27	3	1	88	754	90	9
1	27	3	1	83	734	90	11
0	29	4	1	82	.	90	13
0	27	1	1	86	821	89	2
1	21	4	1	86	720	90	9
0	26	3	1	88	692	93	10
1	21	3	1	89	620	94	12
1	24	4	1	82	648	92	10
1	25	4	1	82	699	90	8
0	25	1	1	77	642	93	11
0	25	3	1	83	691	93	15
1	23	4	1	80	701	90	12
1	25	4	1	81	661	87	7

א. בדוק האם ניתן לנבא את את הציון הממוצע על סמך תואר הלימוד של הסטודנט.

ב. האם הוספת המשתנה "ציון בקורס מבוא לסוציולוגיה" תשפר את הניבוי של המודל?

ג. בדוק האם יש אינטראקציה בין המשתנה "מין הסטודנט" למשתנה "תואר הלימוד" (דהיינו, האם למשתנה תואר הלימוד יש השפעה שונה לסטודנטים נשים וגברים)

ד. בדוק אילו משתנים, מכל המשתנים הקיימים בקובץ הנתונים, מספקים את המודל הטוב ביותר לניבוי הציון הממוצע. כמה נבאים קיימים במודל? מהו אחוז השונות בממוצע אותו ניתן להסביר באמצעות המודל? כדי לענות על שאלה זאת, יש להשתמש ב-stepwise regression.

תרגיל 36

23 אצנים השתתפו במחקר הבודק שיטות אימון שונות לשיפור זמן הריצה. כל אצן עבר 4 סדרות אימונים שונות, ובתום כל סדרה מדדו החוקרים את הזמן שלקח לו לסיים את מסלול הריצה. להלן זמני הריצה של האצנים תחת ארבעת שיטות האימון השונות:

8.86	5.75	5.22	3.83
9.36	5.71	3.46	4.80

15.76	10.86	5.82	5.73
11.43	9.27	8.23	7.14
10.99	11.04	10.84	8.35
13.44	7.70	8.73	8.76
19.08	21.38	18.35	12.02
12.28	14.51	15.52	9.27
16.93	11.63	12.41	8.53
18.05	15.83	11.06	6.73
18.47	19.02	16.16	7.33
13.20	17.82	10.40	8.66
11.23	14.55	8.52	6.93
47.69	10.37	8.58	9.28
19.28	12.31	15.76	8.51
9.81	12.95	7.83	7.58
12.78	13.47	9.49	6.53
17.64	12.47	11.30	5.77
15.37	12.59	10.48	7.75
16.30	15.13	12.42	7.77
7.19	14.18	9.07	5.47
12.22	12.64	10.94	8.41
12.76	12.41	10.88	6.18

בדוק האם לשיטת האימון יש השפעה על זמן הריצה. אם כן, מהי שיטת האימון (שיטה 1 – שיטה 4) היעילה ביותר?

רמז: להזכירך, מדובר כאן על מדידות חוזרות של אותם נבדקים.

תרגיל 37

במחקר שבחן התנהגות כלכלית של אנשים, נבדקים נתבקשו לבחור בין שני הימורים. הניסוי כלל שני תנאים שהורצו במערך תוך-נבדקי: סכום ההימורים (גבוה או נמוך), וההפרש בין ההימורים (גבוה או נמוך). להלן זמן ההחלטה הממוצע שלקח לכל נבדק לבחור הימור, בכל אחד מארבעת תנאי הניסוי (LD_HP LD_LP HD_HP HD_LP, כאשר H מייצג high, L מייצג Low, P מייצג pay ו-D מייצג difference. לדוגמא: LD_HP מייצג את התנאי בו ההבדל בין ההימורים היה נמוך אבל סכום ההימורים היה גבוה):

1	12.57	7.19	6.52	7.29
2	8.20	7.14	4.32	8.49
3	19.71	13.58	7.27	9.67
4	14.29	11.59	10.29	11.43
5	13.71	13.80	13.55	16.69
6	16.80	9.63	10.91	13.45
7	23.85	26.72	22.94	17.52
8	15.34	18.13	28.14	27.84
9	21.16	14.53	11.76	13.15
10	10.06	19.78	13.82	10.91
11	10.59	11.28	10.20	11.66
12	16.49	22.27	15.49	13.32
13	14.03	18.19	14.40	11.16
14	59.61	12.97	15.72	14.10

15	24.10	15.39	19.70	13.13
16	12.26	16.19	13.54	11.98
17	18.47	16.84	14.37	10.66
18	9.55	3.08	11.63	9.71
19	6.72	6.99	8.10	9.68
20	11.63	11.41	10.52	9.71
21	8.99	17.72	11.34	9.34
22	15.27	15.80	11.17	13.02
23	15.96	15.52	12.35	10.23

חוקר א׳ טען כי מה שאמור להשפיע על זמני התגובה הוא גובה ההימורים. לעומתו, חוקר ב׳ טען כי דווקא ההבדל בין ההימורים אמור להשפיע. על סמך התוצאות שלהלן, מי מהחוקרים צודק?

תרגיל 38

ניסוי למידה קלאסי כולל שני שלבים: שלב למידה ושלב הכחדה. ניסוי למידה שנערך לאחרונה כלל 3 תנאים. כל אחד מהתנאים כלל 100 סיבובים מכל שלב (ובסך הכל 200 סיבובים), בהם נתבקשו הנבדקים לבחור בין האופציה הרצויה (אופציה א׳) לבין אופציה אלטרנטיבית (אופציה ב׳). 63 נבדקים השתתפו בניסוי (21 נבדקים בכל תנאי). כאשר ניתחו את הממצאים, עלו התוצאות הבאות: בשלב הלמידה של תנאי 1 נמצא כי ב-93.54% מהמקרים בממוצע הנבדקים בחרו באופציה הרצויה, בעוד שבשלב ההכחדה אופציה זו נבחרה רק ב-2.13% מהמקרים. לעומת זאת, בשלב הלמידה של תנאי 2 נמצא כי רק ב-56.54% מהמקרים בממוצע הנבדקים בחרו באופציה הרצויה, בעוד שבשלב ההכחדה אופציה זו נבחרה ב-5.63% מהמקרים. לבסוף, בשלב הלמידה של תנאי 3 האופציה הרצויה נבחרה ב-56.42% מהמקרים בממוצע, וב-4.12% מהמקרים בשלב ההכחדה.

חוקרים רצו לבדוק האם אחוז המקרים בהם נבחרה האופציה הרצויה בשלב ההכחדה שונה במובהק בין התנאים 2 ו-3. מאחר ובשלב הלמידה של שני תנאים אלה אחוז הפעמים בהם נבחרה האופציה הרצויה היה כמעט זהה, הם החליטו להשתמש באחוז הבחירות באופציה הרצויה מ-10 הסיבובים האחרונים של שלב הלמידה של תנאים אלה כ-covariate בניתוח. כתוב קוד SAS לביצוע ניתוח זה.

פרק 12
פרוצדורות סטטיסטיות IV: מבחנים א-פרמטריים

PROC NPAR1WAY

הפרוצדורה NPAR1WAY מבצעת מבחנים א-פרמטריים להבדלים בין קבוצות. מבחנים אלה רלוונטיים כאשר לא ניתן להניח התפלגות נורמאלית של התצפיות (הנתונים מוטים, יש מעט תצפיות וכדומה) והם מתמקדים בעיקר בסימן (פלוס או מינוס) או בדירוג של התצפיות, ולא בערכן הממשי.

בין היתר, PROC NPAR1WAY כוללת את הניתוחים הבאים:

ניתוחים להשוואת מספר מדגמים בלתי תלויים:

1. מבחן Kruskal-Wallis (מבחן הזמין גם ב-PROC FREQ)
2. מבחן Van der Waerden
3. מבחן החציון
4. מבחן Savage

ניתוחים ל-scale differences:

1. מבחן Siegal-Tukey
2. מבחן Ansari-Bradley
3. מבחן Klotz
4. מבחן Mood

מבחני השערות:

1. מבחן Mann-Whitney
2. מבחן Wilcoxon

בחינת התפלגויות – האם ההתפלגות של משתנה זהה מעבר לקבוצות שונות (empirical distribution function - EDF):

1. Kolmogorov-Smirnov statistic
2. Cramer-von Mises statistic
3. Kuiper statistic (במקרה של שתי קבוצות בלבד)

אופן הכתיבה:

```
PROC NPAR1WAY <אופציות שונות>;
VAR משתנים;
BY משתנים;
CLASS משתנה;
```

EXACT אופציות סטטיסטיות</אופציות חישוביות>;
FREQ משתנה;
OUTPUT <out = קובץ נתונים><אופציות שונות>;
RUN;

דוגמא:

```
proc npar1way;
 class condition;
run;
```

כברירת מחדל, PROCNPAR1WAY מפיקה פלט של המבחנים הבאים (לפי סדר הופעתם): anova, Wilcoxon, Kruskal-Wallis, median, Van der Waerden, Savage, Kolmogorov-Smirnov, Cramer-von Mises, ו-Kuiper. להלן הפלט הבסיסי של PROCNPAR1WAY:

The SAS System

The NPAR1WAY Procedure

Analysis of Variance for Variable q1
Classified by Variable condition

condition	N	Mean
1	7	2.00
0	5	2.20

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Among	1	0.116667	0.116667	0.2431	0.6327
Within	10	4.800000	0.480000		

Average scores were used for ties.

Wilcoxon Scores (Rank Sums) for Variable q1
Classified by Variable condition

condition	Sum of N	Expected Scores	Std Dev Under H0	Mean Under H0	Score
1	7	42.50	45.50	5.461643	6.071429
0	5	35.50	32.50	5.461643	7.100000

Average scores were used for ties.

Wilcoxon Two-Sample Test

Statistic	35.5000

Normal Approximation

Z	0.4577
One-Sided Pr > Z	0.3236
Two-Sided Pr > |Z|	0.6471

t Approximation

One-Sided Pr > Z	0.3280
Two-Sided Pr > |Z|	0.6561

Z includes a continuity correction of 0.5.

Kruskal-Wallis Test

Chi-Square	0.3017
DF	1
Pr > Chi-Square	0.5828

Median Scores (Number of Points Above Median) for Variable q1
Classified by Variable condition

condition	Sum of N	Expected Scores	Std Dev Under H0	Mean Under H0	Score
1	7	3.142857	3.50	0.583874	0.448980
0	5	2.857143	2.50	0.583874	0.571429

Average scores were used for ties.

Median Two-Sample Test

Statistic	2.8571
Z	0.6117
One-Sided Pr > Z	0.2704
Two-Sided Pr > |Z|	0.5408

Median One-Way Analysis

Chi-Square	0.3741
DF	1
Pr > Chi-Square	0.5408

Van der Waerden Scores (Normal) for Variable q1
Classified by Variable condition

condition	Sum of N	Expected Scores	Std Dev Under H0	Mean Under H0	Score
1	7	-0.688193	0.0	1.307282	-0.098313
0	5	0.688193	0.0	1.307282	0.137639

Average scores were used for ties.

Van der Waerden Two-Sample Test

Statistic	0.6882
Z	0.5264
One-Sided Pr > Z	0.2993
Two-Sided Pr > |Z|	0.5986

Van der Waerden One-Way Analysis

Chi-Square	0.2771
DF	1
Pr > Chi-Square	0.5986

Savage Scores (Exponential) for Variable q1
Classified by Variable condition

condition	Sum of N	Expected Scores	Std Dev Under H0	Mean Under H0	Score
1	7	-1.077912	0.0	1.359086	-0.153987
0	5	1.077912	0.0	1.359086	0.215582

Average scores were used for ties.

Savage Two-Sample Test

Statistic	1.0779
Z	0.7931
One-Sided Pr > Z	0.2139
Two-Sided Pr > |Z|	0.4277

Savage One-Way Analysis

Chi-Square	0.6290
DF	1
Pr > Chi-Square	0.4277

Kolmogorov-Smirnov Test for Variable q1
Classified by Variable condition

condition	N	EDF at Maximum	Deviation from Mean at Maximum
1	7	0.857143	0.283473
0	5	0.600000	-0.335410
Total	12	0.750000	

Maximum Deviation Occurred at Observation 6
Value of q1 at Maximum = 2.0

Kolmogorov-Smirnov Two-Sample Test (Asymptotic)

KS	0.126773	D	0.257143
KSa	0.439155	Pr > KSa	0.9905

Cramer-von Mises Test for Variable q1
Classified by Variable condition

```
                Summed Deviation
  condition       N        from Mean
 ------------------------------------
  1               7         0.047536
  0               5         0.066551

Cramer-von Mises Statistics (Asymptotic)
     CM  0.009507    CMa  0.114087

     Kuiper Test for Variable q1
   Classified by Variable condition

                Deviation
  condition       N      from Mean
 ------------------------------------
  1               7       0.257143
  0               5       0.057143

   Kuiper Two-Sample Test (Asymptotic)
K  0.314286    Ka  0.536745    Pr > Ka  1.0000
```

לכן, אם מעוניינים רק בפלט של מבחן ספציפי, יש להגדיר מבחן זה באופציות של PROC NPAR1WAY (כפי שיפורט להלן). במקרה כזה, פלט הפרוצדורה יכלול רק את התוצאות של המבחן המוגדר באופציה.

אופציות של PROC NPAR1WAY

1. האופציה anova – אופציה זו מבקשת מ-PROC NPAR1WAY לבצע ניתוח שונות סטנדרטי (ANOVA).
אופן הכתיבה:

anova

2. האופציה data – אופציה זו מגדירה את קובץ הנתונים עליו תבצע PROC NPAR1WAY את הניתוח.
אופן הכתיבה:

data = שם קובץ נתונים

3. האופציה EDF – אופציה זו מבקשת מ-PROC NPAR1WAY סטטיסטיים המבוססים על ניתוח CDF אמפירי (empirical cumulative distribution function). סטטיסטיים אלה כוללים את הסטטיסטים Kolmogotov-Smirniv, Cramer-von Mises, ו-Kuiper.
אופן הכתיבה:

edf

4. האופציה klotz – אופציה זו אומרת ל-PROC NPAR1WAY לבצע ניתוח תוך שימוש בציוני Klotz.
אופן הכתיבה:

klotz

5. האופציה median – אופציה זו מבקשת מ-PROC NPAR1WAY לבצע את הניתוח תוך שימוש בחציונים.
אופן הכתיבה:

median

6. האופציה missing – אופציה זו אומרת ל-PROC NPAR1WAY להתייחס לערכים חסרים של משתנים קטגוריאליים (משתני CLASS) כאל ערכים תקפים (דהיינו כאחת הרמות של המשתנה עם הערכים החסרים המוגדר בהוראה CLASS).

אופן הכתיבה:

missing

7. האופציה mood – אופציה זו מבקשת מ-PROC NPAR1WAY לבצע את הניתוח תוך שימוש בשכיחים.
אופן הכתיבה:

mood

8. האופציה noprint – אופציה זו מבטלת את ההפקה של קובץ הפלט של PROC NPAR1WAY.
אופן הכתיבה:

noprint

9. האופציה st – אופציה זו אומרת ל-PROC NPAR1WAY לבצע ניתוח תוך שימוש בציוני Siegel-Tukey.
אופן הכתיבה:

st

10. האופציה vw – אופציה זו אומרת ל-PROC NPAR1WAY לבצע ניתוח תוך שימוש בציוני Van der Waerden.
אופן הכתיבה:

vw

11. האופציה wilcoxon – אופציה זו אומרת ל-PROC NPAR1WAY לבצע ניתוח תוך שימוש בציוני Wilcoxon.
אופן הכתיבה:

wilcoxon

ההוראה VAR

הוראה זו מגדירה ל-PROC NPAR1WAY על איזה משתנים יש לבצע את הניתוח הסטטיסטי. אם לא כוללים הוראה זו, הפרוצדורה תחשב מקדמי מתאם לכל המשתנים הנומריים הקיימים בקובץ הנתונים, ואשר לא מוגדרים באף אחת מההוראות האחרות.

אופן הכתיבה:

VAR שמות משתנים;

ההוראה BY

הוראה זו אומרת ל-SAS לבצע את הניתוח באופן נפרד לכל קבוצה של המשתנה המוגדר על ידי ההוראה.

אופן הכתיבה:

BY <descending> משתנה 1 <descending> משתנה n <notsorted>;

דוגמא:

```
proc npar1way;
 class condition;
 by gender;
run;
```

ההוראה CLASS

ההוראה CLASS היא הוראת חובה ב-PROC NPAR1WAY. הוראה זו מזהה קבוצות במשתנה, ו- PROC NPAR1WAY מבצעת ניתוח למציאת הבדלים בין קבוצות אלה. ניתן לכתוב רק משתנה אחד בהוראת CLASS.

אופן הכתיבה:

CLASS משתנה;

דוגמא:

```
proc npar1way;
 class condition;
run;
```

ההוראה EXACT

ההוראה EXACT מבקשת ש-PROC NPAR1WAY תבצע exact tests (מבחנים סטטיסטיים בהם כל ההנחות בנוגע לסטטיסטים של המבחן מתקיימות) לסטטיסטים שהוגדרו לניתוח על ידי ההוראה.

אופן הכתיבה:

EXACT הגדרת סטטיסטיים </אופציות חישוביות>;

הגדרת סטטיסטים:

הגדרת הסטטיסטים בהוראה EXACT קובעת לאיזה סטטיסטים יש לספק בניתוח exact tests. הסטטיסטים לגביהן ניתן לבקש ב-PROC NPAR1WAY ניתוח exact test הם AB (מבחן Ansari-Bradley), klotz (מבחן klotz), median (מבחן חציונים), mode (מבחן שכיחים), savage (מבחן Savage), st (מבחן Siegel-Tukey), wilcoxon (מבחן Wilcoxon ומבחן Kruskal-Wallis), ו-vw (מבחן Van der Waerden).

אופן הכתיבה:

EXACT שם הסטטיסטי;

דוגמא:

```
exact klotz st;
```

הערה: כאשר לא מגדירים אף סטטיסטי בהוראה EXACT, PROC NPAR1WAY תפיק exact tests לכל הסטטיסטים המוגדרים בפרוצדורה.

האופציות החישוביות של ההוראה EXACT מגדירות אופציות לחישוב של ה - exact statistics.

1. האופציה alpha – אופציה זו מגדירה את רמת המובהקות ורווח הסמך של האומדנים. כברירת מחדל, ערך זה נקבע ל-0.01.
אופן הכתיבה:

/alpha = מספר בין 0.0001 ל-0.9999

2. האופציה maxtime – אופציה זו מגדירה ל-PROC NPAR1WAY את זמן העיבוד המקסימאלי (בשניות) בה היא יכולה לחשב exact tests. באם החישוב לא יגמר עד הזמן המוקצב, הפעולה תיפסק ללא הפקת פלט. מאחר ובמקרים מסוימים (כגון סט נתונים גדול) חישוב של exact tests עשוי להיות ארוך מאוד (דבר שמעמיס מאוד על זיכרון המערכת), אופציה זו חשובה מאוד.
אופן הכתיבה:

/maxtime = ערך חיובי

3. האופציה mc – אופציה זו מגדירה ל-PROC NPAR1WAY שה-exact test יתבצעו בשיטת Monte Carlo (שיטה לניתוח סטטיסטי באמצעות מספרים אקראיים). ניתן להגדיר ניתוח Monte Carlo עבור כל אחד מהסטטיסטים בהוראה EXACT.
אופן הכתיבה:

/mc

4. האופציה n – אופציה זו מגדירה את מספר הדגימות בהן ייעשה שימוש בניתוח Monte Carlo. ברירת המחדל היא 10000 דגימות. ככל שמספר זה גדול יותר, תוצאות הניתוח מדויקות יותר. עם זאת, ככל שמספר הדגימות גדול יותר, העומס על זיכרון המערכת עולה.
אופן הכתיבה:

/n = מספר גדול מ - 0

הערה: כאשר מגדירים את האופציה n, היא מגדירה אוטומטית גם את האופציה mc.

5. האופציה seed – אופציה זו מגדירה את ה-initial seed (ערך התחלתי המוזן לאלגוריתם של מחולל המספרים האקראי) ליצירת המספרים האקראיים בשיטת Monte Carlo. באם אופציה זו לא מוגדרת, PROC NPAR1WAY תשתמש בזמן המערכת כ-initial seed.
אופן הכתיבה:

/seed = מספר גדול מ - 0

הערה: כאשר מגדירים את האופציה seed, היא מגדירה אוטומטית גם את האופציה mc.

האופציה FREQ

האופציה FREQ מגדירה משתנה אשר הערך שלו מציין שכיחויות של ערכי המשתנים בקובץ הנתונים. לכן, כאשר PROC NPAR1WAY מבצעת את הניתוח, היא משתמשת בכל ערך של המשתנים x פעמים, כאשר x מוגדר כערך של המשתנה המוגדר באופציה FREQ.

אופן הכתיבה:

FREQ משתנה;

האופציה OUTPUT

הוראה זו אומרת ל-SAS ליצור מהסטטיסטים המחושבים ב-PROC NPAR1WAY קובץ נתונים חדש.

אופן הכתיבה:

OUTPUT out = שם של קובץ הנתונים החדש <אופציות שונות>;

דוגמא:

```
proc npar1way;
 class condition;
 output out=npar_data median;
run;
```

האופציות שניתן להגדיר בהוראה OUTPUT הן כל הסטטיסטיים שניתנים להגדרה כאופציות ב-PROC NPAR1WAY. הפלט שנוצר מכיל משתנים כמספר הסטטיסטים שהוגדרו על ידי ההוראה (עבור כל אחד מהמשתנים המוגדרים בהוראה VAR), והוא כולל תצפית אחד לכל משתנה.

PROC FREQ

כפי שתואר בפרק *9*, PROC FREQ היינה במהותה פרוצדורה להפקת סטטיסטיקה תיאורית. עם זאת, הפרוצדורה גם מאפשרת ניתוח של מבחנים א-פרמטריים, כגון מבחן חי בריבוע. מאחר ו-PROC FREQ נידונה בהרחבה בפרק *9* (כולל כיצד לבצע ניתוחים א-פרמטריים כגון חי בריבוע), לא נחזור על הדיון כאן.

תרגול עצמי – מבחנים א-פרמטריים

תרגיל 39

בקפיטריה של אחת האוניברסיטאות בארץ התעורר ויכוח מי משלם יותר כסף בחודש על ספרי לימוד: סטודנטים לפיסיקה או סטודנטים למתמטיקה. אחד הפרופסורים ששמע את הויכוח, החליט לבדוק את העניין. הוא שאל 8 סטודנטים מכל מחלקה (פיסיקה ומתמטיקה) כמה כסף כל אחד מהם הוציא על ספרי לימוד במהלך החודש האחרון. להלן הנתונים שאסף הפרופסור:

student	math	physics
1	205	250
2	450	240
3	300	250
4	279	725
5	470	380
6	90	370

7	340	150
8	220	620

כתוב קוד SAS שיעזור לפרופסור לפתור את הויכוח.

תרגיל 40

כתוב קוד SAS להרצת סימולציה של זריקת קוביה. על התוכנית לזרוק קוביה 10000 פעם, ולשמור את התוצאה של כל זריקה. בדוק את אחוז הפעמים שכל תוצאת זריקה התרחשה (כמה פעמים תוצאת הזריקה הייתה 1, כמה פעמים תוצאת הזריקה הייתה 2 וכדומה. באמצעות מבחן חי בריבוע, בדוק האם תוצאות הסימולציה דימו תוצאות של קובייה הוגנת (כך שכל תוצאה יצאה 16% מהפעמים).

תרגיל 41

נתון כי בשנת למודים כלשהי נרשמו לפקולטה א׳ 60 נשים ו-40 גברים, וכי לפקולטה ב׳ נרשמו 60 גברים ו-40 נשים. האם ניתן לטעון כי נשים מעדיפות את פקולטה א׳ וגברים מעדיפים את פקולטה ב׳?

פרק 13
פרוצדורות גראפיות: תרשימים וגרפים

PROC PLOT

הפרוצדורה PLOT מציירת דיאגרמות פיזור לשני משתנים. ניתן ליצור מספר דיאגרמות חופפות, או להציג כל דיאגרמה בתרשים נפרד. כמו כן, SAS מאפשרת לערוך את הסגנון של התרשים, כפי שיודגם להלן.

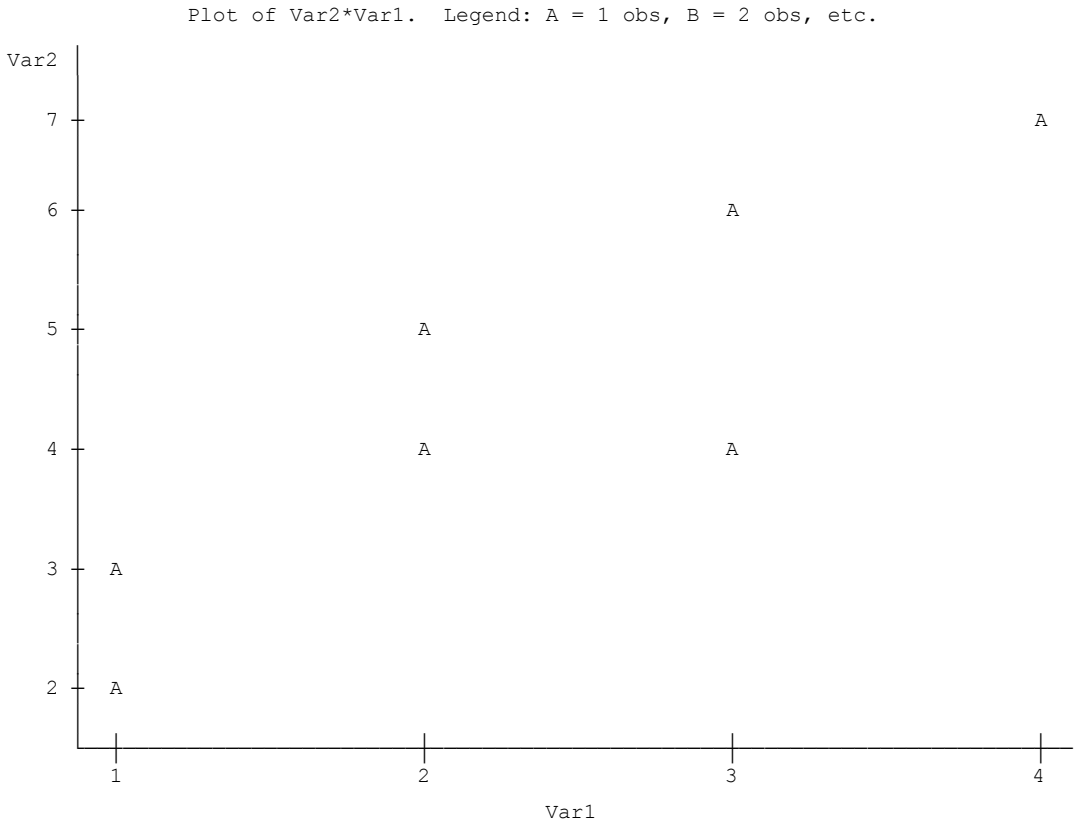
אופן הכתיבה:

PROC PLOT אופציות שונות;
BY <decending> משתנה 1 <descending> n משתנה <notsorted>;
PLOT אופציות שונות/>בקשת-דיאגרמה/ות<;
RUN;

דוגמא:

```
proc plot;
 plot s1*block;
run;
```

הפלט הבסיסי של PROC PLOT הוא:



1. האופציה formchar – אופציה זו מגדירה את התווים בהם PROC PLOT תעשה שימוש כדי ליצור את הגבולות של התרשים.
אופן הכתיבה:

formchar <(מיקומים)> = 'התווים הרצויים'

מיקומים:

המיקומים מגדירים אילו תווים ירכיבו אילו חלקים מגבולות התרשים. יש לכתוב את המיקומים הרצויים מופרדים על ידי פסיקים, על פי הפירוט הבא:

מיקום	תו ברירת המחדל	גבול
1	|	ציר אנכי
2	-	ציר אופקי
3, 5, 9, 11	-	פינתי 3 – פינה שמאלית עליונה 5 – פינה ימנית עליונה 9- פינה שמאלית תחתונה 11 – פינה ימנית תחתונה
7	+	שנתות הגרף (מפגש בין אופקי ואנכי)

תווים רצויים:

את התווים הרצויים עבור כל מיקום יש לכתוב לפי הסדר בו רשומים המיקומים, ללא רווחים ביניהם. כאשר לא יוגדר באופציה מיקום כלשהו, SAS תשתמש למיקום זה בתו ברירת המחדל.

דוגמאות:

```
formchar (9, 7) = '&$';
```

דוגמא זו אומרת ל-SAS להשתמש בסימן & לפינה השמאלית התחתונה, ובסימן $ למפגש בין קווים אופקיים ואנכיים (שנתות הגרף).

```
formchar (1, 2, 7, 3, 5, 9, 11) = '       '
```

דוגמא זו תצייר טבלה ללא גבולות. יש לוודא כי מספר הרווחים המופיעים בגרשיים שווה למספר המיקומים המוגדרים.

2. האופציה nolegend – אופציה זו אומרת ל-SAS להפיק תרשים ללא מקרא-תרשים.
אופן הכתיבה:

nolegend

3. האופציה uniform – אופציה זו אומרת ל-SAS ליצור יחס צירים זהה לכל תרשים במצב בו PROC PLOT יוצרת תרשימים נפרדים לכל קבוצת משתנים המוגדרת באמצעות ההוראה BY. אופציה זו מאוד שימושית כאשר מעוניינים להשוות בין תרשימים.
אופן הכתיבה:

uniform

4. האופציה hpercent – אופציה זו קובעת את האחוז מהשטח האופקי הזמין לעמוד בדף הפלט שיוקצה לכל תרשים. כך, ניתן לבקש מ-PROC PLOT להוציא בפלט כמה תרשימים באותו עמוד, או להגדיר את הגודל המקסימילי משטח העמוד שיתפוס כל תרשים בודד.

הערה: ניתן להגדיר באופציה יותר מאחוז אחד, ובכך לקבוע כמה אחוז משטח הדף ייקחו תרשימים שונים, או להגדיר מעבר עמוד בין תרשים לתרשים (באמצעות הגדרת האחוז ל-0).
אופן הכתיבה:

hpercent = (מופרדים על ידי רווחים) אחוז/ים

דוגמאות:

```
hpercent = 40 20 40
```

בדוגמא זו, התרשים הראשון והאחרון יתפסו 40% מרוחב העמוד, והתרשים השני 20%.

```
hpercent = 50 0
```

בדוגמא זו, התרשימים יתפסו 50% מרוחב העמוד, וכל תרשים יופיע בעמוד נפרד.

הערה: ניתן לקבוע גם אחוז גדול יותר מ-100, מה שיגרום לתרשים להיות ברוחב של יותר מעמוד. לדוגמא, הגדרת האחוז 200 תגרום ליצירת תרשים ברוחב של שני עמודים.

5. האופציה vpercent – אופציה זו זהה לאופציה hpercent, למעט העובדה ש-vpercent קובעת את האחוז מהשטח האנכי הזמין לעמוד בדף הפלט שיוקצה לכל תרשים.
אופן הכתיבה:

hpercent = (מופרדים על ידי רווחים) אחוז/ים

ההוראה BY

הוראה זו יוצרת תרשים נפרד לכל ערך של המשתנה המוגדר על ידי ההוראה BY. כל תרשים נוצר בדף נפרד.

אופן הכתיבה:

BY <decending> משתנה 1 <descending> n משתנה <notsorted>;

ההוראה PLOT

ההוראה PLOT מגדירה את דיאגרמות הפיזור שיופקו על ידי PROC PLOT. ניתן להשתמש במספר הוראות PLOT בצעד PROC אחד.

אופן הכתיבה:

PLOT בקשת דיאגרמה/ות</אופציות שונות>;

הערה: ללא ההוראה PLOT, הפרוצדורה PROC PLOT לא תצייר שום דיאגרמה.

בקשת דיאגרמה מתבצעת באמצעות הגדרת המשתנה לציר y כפול (*) המשתנה לציר x. בנוסף, ניתן להגדיר את הסימן שיציין את הנקודות על הדיאגרמה. כפי שיודגם להלן, הסימן המציין את הנקודה על הדיאגרמה יכול להיות תו, ערך של משתנה או אות מה-A B C (ברירת המחדל).

דוגמא להגדרת תו:

```
plot age * gender = '*';
```

דוגמא זו מבקשת לצייר דיאגרמה של גיל (על ציר y) מול מין (על ציר x), כאשר הסימן שמציין כל נקודה על הדיאגרמה מוגדר להיות כוכבית (*).

דוגמא להגדרת ערך של משתנה:

```
plot y * x = gender;
```

דוגמא זו מבקשת לצייר דיאגרמה של המשתנה y מול המשתנה x, כאשר כל נקודה על הדיאגרמה מציינת את הערך של המשתנה gender המתאים לנקודה זו.

דוגמא לשימוש בברירת המחדל (אותיות ה-A B C):

```
plot y * x;
```

בדוגמא זו, כל נקודה על הדיאגרמה שיש תצפית המתאימה לה תסומן באות A, כל נקודה שיש זוג תצפיות המתאים לה תסומן באות B וכך הלאה.

כדי להגדיר זוגות משתנים לציור, ניתן לכתוב את זוג המשתנים, כאשר הסימן * מפריד ביניהם, או להשתמש בקודי קיבוץ, כפי שמודגם בטבלה 12.

קוד	שווה ערך ל-
PLOT (a – d) או PLOT (a - - d)	a * b, a * c, a * d, b * c, b * d, c * d
PLOT (_numeric_)	כל הקומבינציות של המשתנים הנומריים בקובץ הנתונים
PLOT x(y1-y4)	x * y1, x * y2, x * y3, x * y4
PLOT (x1-x2) : (y1-y2)	x1 * y1, x2 * y2
PLOT (y1-y2) * (x1-x2)	y1 * x1, y1 * x2, y2 * x1, y2 * x2

טבלה 12 – קודי קיבוץ להפקת דיאגרמות פיזור על ידי PROC PLOT ו-PROC GPLOT

אופציות של ההוראה PLOT

1. האופציה box – אופציה זו מציירת מסגרת מסביב לכל התרשים (ולא רק מסגרת של ציר x ו-y).
   אופן הכתיבה:

/box

2. האופציה href – אופציה זו מוסיפה קווים אופקיים לתרשים בנקודה/נקודות המוגדרות על ידי האופציה (קווי ייחוס).
   אופן הכתיבה:

/href =(מופרדים על ידי רווחים) ערכים בטווח המשתנים

3. האופציה hrefchar – אופציה זו מגדירה את התווים בהם יש להשתמש כדי ליצור את הקו האופקי המוגדר באופציה href. כברירת מחדל, התו המוגדר לקו זה הוא קו אופקי (הסימן |).
אופן הכתיבה:

/hrefchar = 'תו כלשהו'

4. האופציה hreverse – אופציה זו הופכת את הסדר של הערכים על הציר האופקי (מהגדול לקטן במקום ברירת המחדל שהיא מהקטן לגדול).
אופן הכתיבה:

/hreverse

5. האופציה haxis – אופציה זו מגדירה את ערכי השנתות וטווח השנתות של הציר האופקי.
אופן הכתיבה:

/haxis = ערכים

עבור ערכים מספריים של המשתנה, ניתן לכתוב או ערכים ספציפיים מופרדים על ידי פסיקים (כגון 1, 2, 3, 4), או הגדרת טווח ורווחים על פי הדוגמא הבאה:

```
/haxis = 1 to 10 by 3
```

עבור משתנים אלפאנומריים, יש לכתוב את כל הערכים הרצויים מופרדים על ידי פסיקים. כל הערכים הרצויים צריכים להופיע במרכאות, לפי הדוגמא הבאה:

```
/haxis = 'male' 'female' 'unknown'
```

6. האופציה vref – אופציה זו זהה לאופציה href למעט העובדה שהיא מוסיפה קווים אנכיים לתרשים.
אופן הכתיבה:

/vref =(מופרדים על ידי רווחים) ערכים בטווח המשתנים

7. האופציה vrefchar – אופציה זו זהה לאופציה hrefchar למעט העובדה שהיא מגדירה את התווים בהם יש להשתמש כדי ליצור את הקו האנכי.
אופן הכתיבה:

/vrefchar = 'תו כלשהו'

8. האופציה vreverse – אופציה זו הופכת את הסדר של הערכים על הציר האנכי.
אופן הכתיבה:

/vreverse

9. האופציה vaxis – אופציה זו מגדירה את ערכי השנתות ואת טווח השנתות של הציר האנכי. אופן ההגדרה של האופציה זהה להגדרה של האופציה haxis.
אופן הכתיבה:

/vaxis = ערכים

10. האופציה overlay – אופציה זו מציגה את כל התרשימים המוגדרים בהוראה PLOT במערכת צירים אחת. כאשר מגדירים אופציה זו, הערכים של משתני התרשים הראשון (או התוויות שלהן אם הן מוגדרות) יישמשו כערכי השנתות במערכת הצירים.

אופן הכתיבה:

/overlay

PROC GPLOT

הפרוצדורה GPLOT, בדומה ל-PROC PLOT, מציירת דיאגרמות פיזור לשני משתנים. עם זאת, היא נבדלת מ- PROC PLOT בכמה דברים:

1. תרשימים של PROC PLOT מופקים בחלון Output של SAS, בעוד ש-GPLOT מפיקה תרשימים באיכות גבוהה לחלון SAS/GRAPH.
2. בנוסף לדיאגרמת פיזור רגילה, GPLOT יכולה גם להוסיף קו אנכי נוסף לתרשים, ולצייר bubble plots ו- logarithmic plots.

אופן הכתיבה:

PROC GPLOT <אופציות שונות>;
BUBBLE בקשת תרשים/ים <אופציות שונות>;
BUBBLE2 בקשת תרשים/ים <אופציות שונות>;
PLOT בקשת תרשים/ים <אופציות שונות>;
PLOT2 בקשת תרשים/ים <אופציות שונות>;
RUN;

הערה: PROC GPLOT חייבת לכלול לפחות הוראת PLOT או הוראת BUBBLE אחת.

דוגמא (לציור דיאגרמה):

```
proc gplot;
 plot s1*block;
run;
```

דוגמא זו מציירת דיאגרמה לקשר בין המשתנה s1 למשתנה block (ראה איור 18 א).

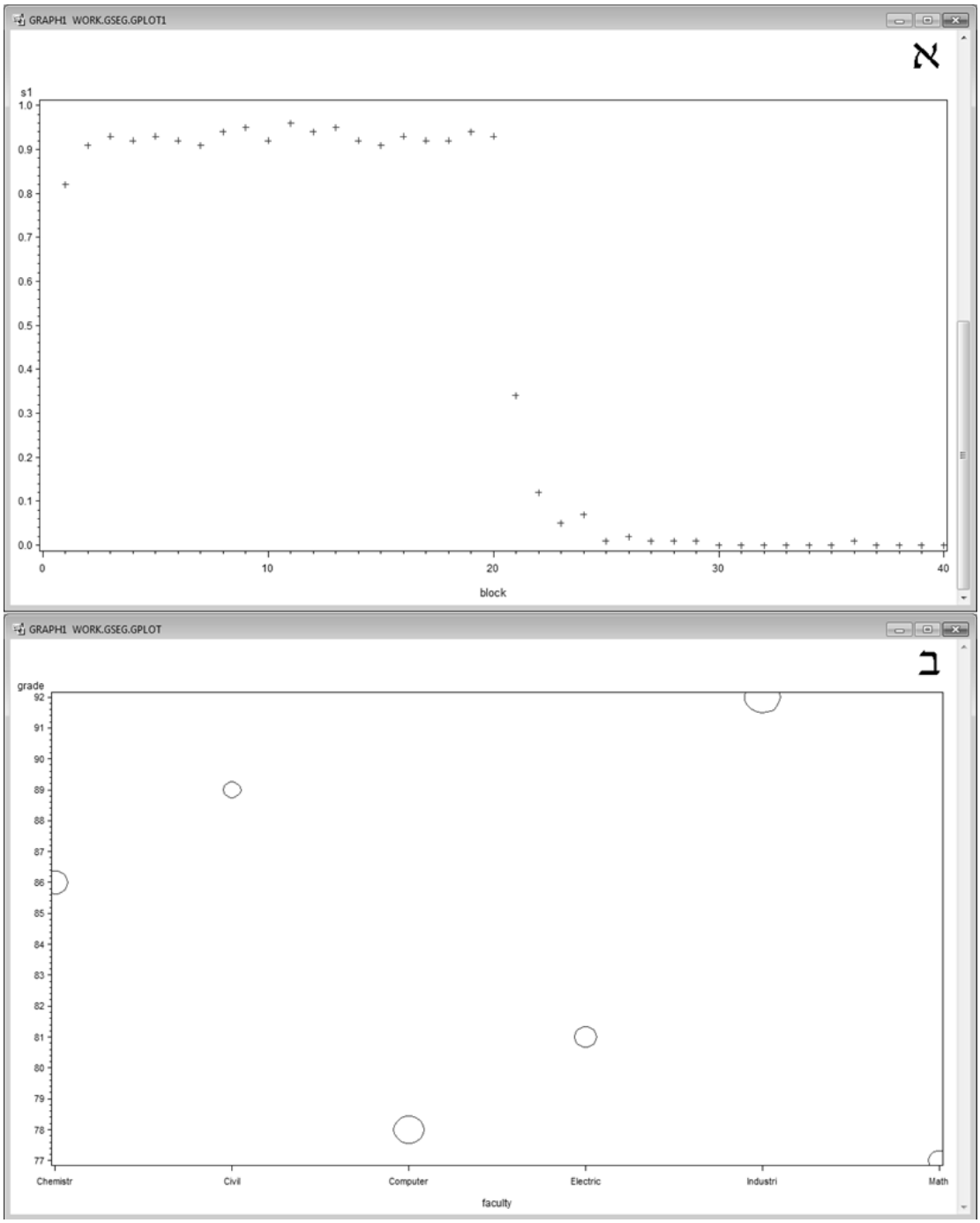
דוגמא (לציור Bubble plot):

```
proc gplot;
 bubble grade*faculty = prop;
run;
```

דוגמא זו יוצרת bubble plot למשתנה "ציון" (grade) כפונקציה של הפקולטה (faculty). המשתנה "גודל הקבוצה" (prop) קובע את גודל הבועה (כאשר ככל שהערך של prop גבוה יותר, הבועה גדולה יותר). כפי שניתן לראות, כברירת מחדל הבועות בקצוות "נחתכות" על ידי מסגרת התרשים (ראה איור 18 ב). בהמשך נלמד כיצד ניתן להימנע מכך.

1. האופציה uniform – כאשר מגדירים אופציה זו, טווח הסקאלות של כל התרשימים שיופקו על ידי PROC GPLOT יהיו שווים. בנוסף, אם מגדירים מקרא-תרשים, אופציה זו שומרת שמקרא התרשים יהיה זהה לכל התרשימים המופקים על ידי הפרוצדורה.
אופן הכתיבה:

uniform



איור 11
(א) דיאגרמת פיזור בסיסית של ROC GPLOT (ב) דיאגרמת BUBBLE בסיסית של PROC GPLOT

ההוראה PLOT

ההוראה PLOT אומרת ל-SAS/GRAPH לצייר דיאגרמה (אחת או יותר) לשני משתנים. על ציר x (הציר האופקי) יופיע המשתנה הבלתי תלוי, ועל ציר y (הציר האנכי השמאלי) יופיע המשתנה התלוי.

אופן הכתיבה:

PLOT בקשת דיאגרמה/ות <אופציות שונות/>;

הערה: להוראות של PROC GPLOT יש אינספור אופציות, המאפשרות לעצב את התרשימים ולהוסיף להם מאפיינים רבים ומגוונים. עם זאת, מאחר ו-SAS אינה תונה גרפית, לא נרחיב עליהן את הדיבור בספר זה.

בקשת דיאגרמה מתבצעת באמצעות הגדרת המשתנה לציר y כפול (*) המשתנה לציר x, או באמצעות שימוש במילות קיבוץ (ראה, לדוגמא, טבלה 12). בנוסף, ניתן להגדיר את הסימן שיציין את הנקודות על הדיאגרמה. הסימן המציין את הנקודה על הדיאגרמה יכול להיות תו (כולל אותיות גדולות – capital letters – ומספרים) או ערך של משתנה. כברירת מחדל, התו המוצג הוא הסימן +.

דוגמא להגדרת תו:

```
plot age * gender = 'A';
```

דוגמא זו מבקשת לצייר דיאגרמה של מין (על ציר x) מול גיל (על ציר y), כאשר הסימן שמציין כל נקודה על הדיאגרמה מוגדר להיות האות A.

הערה: כאשר מגדירים בהוראה PLOT תווים (לדוגמא הסימן *, SAS לא מציג את התו עצמו, אלא סימן גרפי הקשור לתו זה - הסימן ♀ במקרה הנוכחי).

דוגמא להגדרת ערך של משתנה:

```
plot y * x = gender;
```

דוגמא זו מבקשת לצייר דיאגרמה של המשתנה y מול המשתנה x, כאשר כל נקודה על הדיאגרמה מציינת את הערך של המשתנה gender המתאים לנקודה זו.

דוגמא לשימוש בברירת המחדל (+):

```
plot y * x;
```

הערה: במקרה בו משתמשים בסימן ברירת המחדל, ניתן להגדיר את צבע הסימן באמצעות כתיבת מספר קוד אחרי הסימן שווה, כפי שמודגם להלן:

```
plot y * x = 2;
```

במקרה הזה, סימני הפלוס בדיאגרמה יופיעו בצבע אדום (ברירת המחדל היא 1 – שחור).

ההוראה PLOT2

ההוראה PLOT2 מציירת דיאגרמה (אחת או יותר) עם ציר אנכי נוסף מצדו הימני של התרשים, שביחס אליו ניתן להציג משתנה תלוי נוסף.

לא ניתן להגדיר את ההוראה PLOT2 ללא ההוראה PLOT. כמו כן, המשתנה הבלתי תלוי (ציר x) חייב להיות זהה בעבור ההוראה PLOT וההוראה PLOT2. אחרת, תתקבל הודעת השגיאה הבאה בחלון Log:

```
ERROR: Horizontal axis variables on PLOT and PLOT2 statements must be the same.
Variables do not match in plot pair.
```

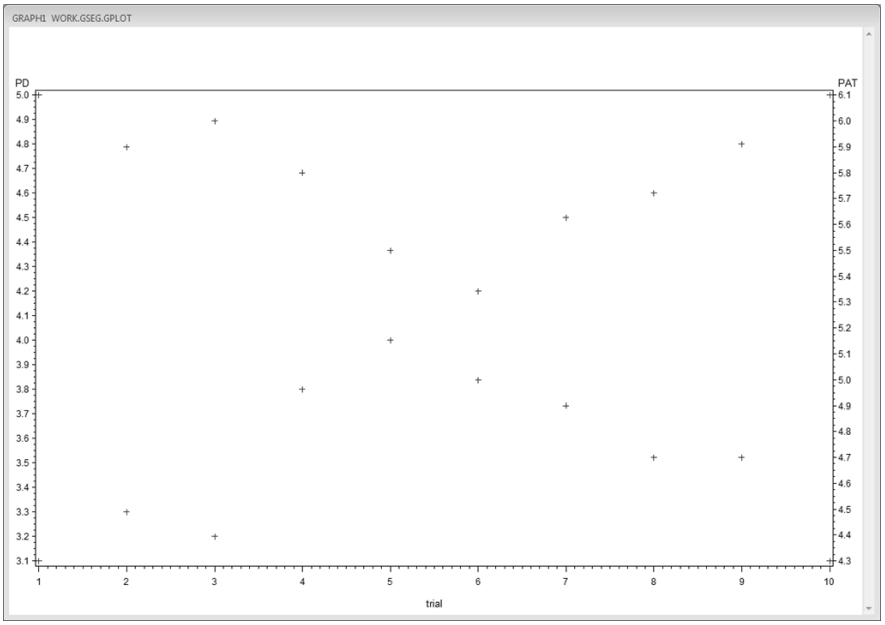
אופן הכתיבה:

PLOT בקשת דיאגרמה/ות </אופציות שונות>;
PLOT2 בקשת דיאגרמה/ות </אופציות שונות>;

דוגמא:

```
proc gplot;
 plot PD * trial = 2;
 plot2 PAT * trial = 4;
run;
```

דוגמא זו מבקשת לצייר דיאגרמה של הקשר בין גודל האישון (המשתנה PD) לסיבוב הניסויי (המשתנה trial), לעומת הקשר בין רוחב כלי הדם ההיקפיים (המשתנה PAT) לסיבוב הניסויי. הפלט של דוגמא זו מוצג באיור 19.



איור 19 – דיאגרמת פיזור ב-PROC GPLOT הכוללת שני צירי Y

בקשת הדיאגראמות והגדרת הסימנים לציון הנקודות על הדיאגראמות נעשה באותו אופן כמו בהוראה PLOT.

ההוראה BUBBLE

ההוראה BUBBLE מגדירה ל-SAS/GRAPH לצייר דיאגרמת Bubble (אחת או יותר), שבה משתנה שלישי מוצג אל מול שני משתנים המיוצגים על ידי שני הצירים. הערך של משתנה שלישי זה קובע את גודלה של הבועה בכל נקודה בתרשים.

אופן הכתיבה:

BUBBLE בקשת דיאגרמה/ות <אופציות שונות/>;

בקשת דיאגרמת Bubble נעשית באופן הבא:

y משתנה ציר *x משתנה ציר =(גודל הבועה) משתנה שלישי;

דוגמא:

```
proc gplot;
 bubble PD * trial = num;
run;
```

בדוגמא זו, המשתנה num מגדיר את גודל המדגם לכל זוג תצפיות מהמשתנים trial ו-PD.

ההוראה BUBBLE2

ההוראה BUBBLE2 יוצרת ציר אנכי שני בצדו הימני של התרשים שביחס אליו ניתן להציג משתנה תלוי נוסף. לא ניתן להגדיר את ההוראה BUBBLE2 ללא ההוראה PLOT או ההוראה BUBBLE. כמו כן, המשתנה הבלתי תלוי (ציר x) חייב להיות זהה בעבור ההוראה PLOT/BUBBLE וההוראה BUBBLE2.

הערה: בדומה לכך שניתן להגדיר על אותו ציר גם דיאגרמת פיזור וגם דיאגרמת בועות (באמצעות שילוב ההוראה PLOT וההוראה BUBBLE2), ניתן גם להגדיר זאת באמצעות שילוב ההוראה BUBBLE וההוראה PLOT2 (תלוי מה רוצים שיהיה בציר y השמאלי ומה רוצים שיהיה בציר y הימני).

אופן הכתיבה:

```
proc gplot;
 bubble PD * trial = num;
 bubble2 PAT * trial = num;
run;
```

בקשת הדיאגראמות בהוראה BUBBLE2 נעשית באותו האופן כמו בהוראה BUBBLE.

PROC CHART

הפרוצדורה CHART מציירת דיאגרמת מקלות (היסטוגרמה דו ממדית) אנכית או אופקית, דיאגרמת בלוקים (היסטוגרמה תלת ממדית), דיאגרמת פאי, ודיאגרמת כוכב, שנועדו להציג גרפית ערכים של משתנים (נומריים או אלפאנומריים) או סטטיסטיים הקשורים לערכים אלה.

אופן הכתיבה:

PROC CHART <אופציות שונות>;
BY <decending> 1 משתנה <descending> n משתנה <notsorted>;
BLOCK רשימת משתנים</אופציות שונות>;

VBAR רשימת משתנים</אופציות שונות>;
HBAR רשימת משתנים</אופציות שונות>;
PIE רשימת משתנים</אופציות שונות>;
STAR רשימת משתנים</אופציות שונות>;
RUN;

דוגמא:

```
proc chart;
 block Var1; Vbar Var1;
 hbar Var1; pie Var1; star Var1;
run;
```

הפלט הבסיסי המתקבל לכל סוג של איור ב-PROC CHART מוצג באיור 20.

אופציות של PROC CHART

1. האופציה formchar – אופציה זו מגדירה את התווים בהם PROC CHART תעשה שימוש כדי ליצור את הגרפים (כולל גבולות התרשים).
אופן הכתיבה:

formchar <(מיקומים)> = 'התווים הרצויים'

<u>מיקומים</u>:
המיקומים מגדירים אילו תווים ירכיבו אילו חלקים מהתרשים. יש לכתוב את המיקומים הרצויים מופרדים על ידי פסיקים, על פי הפירוט הבא:

מיקום	תו ברירת המחדל	תיאור
1	|	צירים אנכיים בדיאגרמות קווים, הקווים האנכיים המפרידים בין בלוקים בהיסטוגרמה, וקווי ייחוס בדיאגרמת קווים אופקית.
2	-	צירים אופקיים בדיאגרמות קווים, גבולות הבלוקים בהיסטוגרמה, וקווי ייחוס בדיאגרמת קווים אנכית.
7	+	שנתות בדיאגרמות קווים וקווי האמצע בדיאגרמות פאי וכוכב
9	-	צומת (מפגש בין אופקי ואנכי) בדיאגרמות קווים
16	/	סיומת של בלוקים וקווי האלכסון המפרידים בין בלוקים בהיסטוגרמה
20	*	עיגולים בדיאגרמות פאי וכוכב

<u>תווים רצויים</u>:
את התווים הרצויים עבור כל מיקום יש לכתוב לפי הסדר בו רשומים המיקומים, ללא רווחים ביניהם. כאשר לא יוגדר באופציה מיקום כלשהו, SAS תשתמש למיקום זה בתו ברירת המחדל.

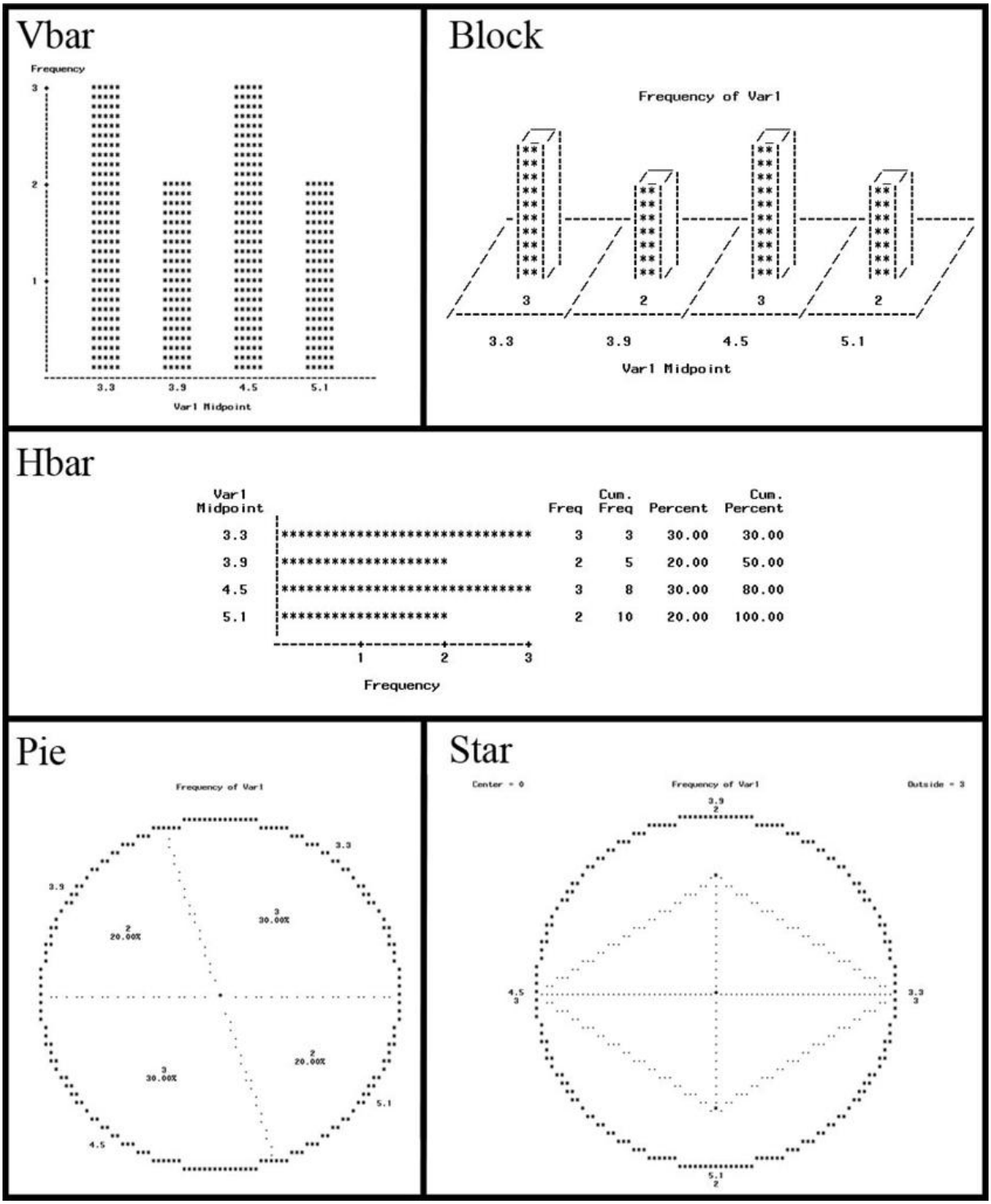
2. האופציה lpi – אופציה זו מגדירה את הפרופורציות של תרשימי פאי וכוכב.
אופן הכתיבה:

lpi = ערך כלשהו

הערך של היחס נקבע על פי הנוסחא:

10 * (עמודות באינץ' / קווים באינץ')

ברירת המחדל של היחס היא 6 (מה שמתאים בקירוב ליחס של 8 קווים בינאץ ו-12 עמודות באינץ').



איור 20 – הצורה הבסיסית של כל התרשימים (דיאגרמת קווים אופקית ואנכית, דיאגרמת בלוקים, דיאגרמת פאי ודיאגרמת כוכב) המופקים על ידי PROC CHART

ההוראה BY

הוראה זו יוצרת תרשים נפרד לכל ערך של המשתנה המוגדר על ידי ההוראה BY. כל תרשים נוצר בדף נפרד.

אופן הכתיבה:

BY <decending> משתנה 1 <descending> משתנה n <notsorted>;

ההוראה BLOCK

ההוראה BLOCK מציירת היסטוגרמה אנכית תלת ממדית. כאשר המשתנה המוגדר על ידי ההוראה בדיד, כל עמודה מייצגת ערך של המשתנה. כאשר המשתנה רציף, SAS מחלקת את הערכים למקטעים, וכל עמודה מייצגת את הערך האמצעי של המקטע הרלוונטי.

ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אופן הכתיבה:

BLOCK רשימת משתנים</אופציות שונות>;

ההוראה VBAR

ההוראה VBAR מציירת היסטוגרמה דו-ממדית (דיאגרמת מקלות) אנכית. כאשר המשתנה המוגדר על ידי ההוראה בדיד, כל עמודה מייצגת ערך של המשתנה. כאשר המשתנה רציף, SAS מחלקת את הערכים למקטעים, וכל עמודה מייצגת את הערך האמצעי של המקטע הרלוונטי.

ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אופן הכתיבה:

VBAR רשימת משתנים</אופציות שונות>;

ההוראה HBAR

ההוראה HBAR יוצרת היסטוגרמה דו-ממדית אופקית. כאשר המשתנה המוגדר על ידי ההוראה בדיד, כל עמודה מייצגת ערך של המשתנה. כאשר המשתנה רציף, SAS מחלקת את הערכים למקטעים, וכל עמודה מייצגת את הערך האמצעי של המקטע הרלוונטי. בנוסף, ההוראה מפיקה במקביל לתרשים גם טבלה המכילה את השכיחות, האחוזים, השכיחות המצטברת והאחוזים המצטברים של הנתונים.

ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אופן הכתיבה:

HBAR רשימת משתנים</אופציות שונות>;

ההוראה PIE

ההוראה PIE יוצרת דיאגרמת פאי. כאשר המשתנה המוגדר על ידי ההוראה בדיד, כל פלח ב-PIE מייצג ערך של המשתנה. כאשר המשתנה רציף, SAS מחלקת את הערכים למקטעים, וכל פלח מייצג את הערך האמצעי של המקטע הרלוונטי.

ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אופן הכתיבה:

PIE רשימת משתנים</אופציות שונות>;

ההוראה STAR

ההוראה STAR יוצרת דיאגרמת כוכב. ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אם כל הערכים של המשתנה המוגדר על ידי ההוראה STAR חיוביים, מרכז הכוכב מייצג את הערך 0 והשוליים של המעגל מייצגים את הערכים המקסימאליים. אם הנתונים כוללים גם ערכים שליליים, המרכז מייצג את הערך המינימאלי.

אופן הכתיבה:

STAR רשימת משתנים</אופציות שונות>;

הערה: כאשר מנסים לצייר דיאגרמת כוכב למשתנה בעל יותר מ-24 רמות, PROC CHART תפיק במקום זאת היסטוגרמה אופקית.

התאמה אישית של דיאגרמות

אופציות של ההוראות BLOCK, VBAR, HBAR, PIE, STAR

אופציות כלליות:

1. האופציה type – אופציה זו מגדירה מה העמודות בהיסטוגרמה או השטחים בתרשמי הפאי או הכוכב מייצגים מבחינה סטטיסטית.
אופן הכתיבה:

/type = סטטיסטי כלשהו

הסטטיסטים הזמנים לאופציה type הם:

א. cfreq – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצגים את השכיחות המצטברת.
ב. cprecent – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצגים אחוז מצטבר
ג. freq – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצגים שכיחות
ד. mean – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצגים את הממוצע של המשתנה המוגדר באופציה sumvar (שתידון להלן) מעבר לכל התצפיות השייכות לאותה עמודה (או מקטע).
ה. percent – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצגים את אחוז התצפיות שיש להן ערך נתון או הנופלות בטווח של כל הערך של המשתנה השייך לאותה עמודה (או מקטע)
ו. sum – סטטיסטי המגדיר כי כל עמודה, בלוק או מקטע מייצג את הסכום של המשתנה המוגדר באופציה sumvar מעבר לכל התצפיות השייכות לאותה עמודה (או מקטע).

2. האופציה sumvar – אופציה זו מגדירה את המשתנה שעבורו PROC CHART תציג את הערכים או הממוצעים (תלוי בהגדרה של האופציה type).

אופן הכתיבה:

/sumvar = שם משתנה

3. האופציה discrete – אופציה זו מגדירה כי הערכים של המשתנה המוצג בתרשים הם בדידים ולא רציפים. כברירת מחדל, PROC CHART מניחה כי כל משתנה נומרי הוא רציף, וכתוצאה מכך מחשבת טווחים להצגת העמודות או המקטעים. לכן, אופציה זו חשובה אם רוצים להציג ערכים מדויקים של משתנה בדיד, ולא טווחים של משתנה רציף.
אופן הכתיבה:

/discrete

4. האופציה subgroup – אופציה זו מחלקת את העמודות או הבלוקים לתווים שונים, כאשר כל תו מייצג ערך שונה של המשתנה המוגדר על ידי האופציה. אופציה זו יעילה אם רוצים למשל לבדוק את התרומה היחסית של כל תת קבוצה (ערכים של המשתנה) לכל עמודה ספציפית. לדוגמא, כאשר יוצרים תרשים עמודות של ציונים, ניתן להשתמש באופציה כדי להגדיר תת קבוצות (למשל פקולטות שונות), ולראות מה החלק היחסי של כל פקולטה בכל אחד מהציונים (כמה סטודנטים מכל פקולטה השיגו את כל אחד מהציונים בתרשים).
התווים השונים נקבעים על ידי התו הראשון של הערך של המשתנה (אלא אם יש כמה קבוצות ערכים המתחילים באותה אות, ובמקרה זה PROC CHART פשוט תתן לתו הראשון את הערך A, לתו השני את הערך B וכך הלאה). התווים בהם PROC CHART השתמשה, כמו גם הערכים שהם מייצגים יופיעו כמקרא תרשים בתחתית העמוד.
אופציה זו זמינה רק להוראות BLOCK, HBAR ו-VBAR.
אופן הכתיבה:

subgroup = שם משתנה

אופציות לעיצוב התרשים:

1. האופציה ascending – אופציה זו אומרת ל-PROC CHART להציג את העמודות בהיסטוגרמה בסדר עולה (מהקטן לגדול). אופציה זו זמינה רק להוראות HBAR ו-VBAR.
אופן הכתיבה:

/ascending

2. האופציה descending – אופציה זו אומרת ל-PROC CHART להציג את העמודות בהיסטוגרמה בסדר יורד (מהגדול לקטן). אופציה זו זמינה רק להוראות HBAR ו-VBAR.
אופן הכתיבה:

/descending

3. האופציה space – אופציה זו קובעת את הרווח בין עמודה לעמודה בהיסטורגרמה. אופציה זו זמינה רק להוראות HBAR ו-VBAR.
אופן הכתיבה:

/space = מספר שלם

4. האופציה nozeros – אופציה זו משמיטה מהתרשים ערכים של המשתנה עם שכיחות של אפס (ערכים שלא קיימים מתוך טווח הערכים). אופציה זו זמינה רק להוראות HBAR ו-VBAR.

אופן הכתיבה:

/nozeros

5. האופציה noheader – אופציה זו משמיטה מהתרשים את הכותרת העליונה, המוצגת כברירת מחדל של ידי PROC CHART (כותרת זו אומרת "שכיחות של (שם המשתנה)". אופציה זו זמינה רק להוראות BLOCK, PIE ו-STAR.

אופן הכתיבה:

/noheader

6. האופציה ref – אופציה זו מציגה קו ייחוס אופקי (או קווים) בערך המוגדר מראש על ידי המשתמש (או ערכים). אופציה זו זמינה רק להוראות VBAR ו-HBAR.

אופן הכתיבה:

/ref = ערך(ים – מופרדים על ידי רווחים)

7. האופציה symbol – אופציה זו מגדירה את התו שירכיב את העמודות בהיסטוגרמה או היסטוגרמה תלת-ממדית. ברירת המחדל של תו זה היא סימן הכוכבית (*). אופציה זו זמינה רק להוראות VBAR, HBAR ו-BLOCK.

אופן הכתיבה:

/symbol = 'תו כלשהו'

דוגמא:

```
vbar Var2 /symbol = 'A';
```

8. האופציה width – אופציה זו מגדירה את הרוחב של העמודות בהיסטוגרמה. אופציה זו זמינה רק להוראות VBAR ו-HBAR.

אופן הכתיבה:

/width = מספר שלם

הערה: רוחב העמודה (המספר) קובע למעשה את מספר התווים מהם תורכב כל שורה של כל עמודה.

PROC GCHART

הפרוצדורה GCHART מציירת דיאגרמת קווים, דיאגרמת בלוקים, דיאגרמת פאי ודיאגרמת כוכב בדומה ל- PROC CHART. עם זאת, בדומה ל-GPLOT, היא נבדלת מ-PROC CHART בכמה דברים:

1. PROC GCHART מפיקה תרשימים באיכות גבוהה לחלון של SAS/GRAPH
2. PROC GCHART כוללת דיאגרמות קווים ודיאגרמת פאי תלת ממדיות
3. PROC GCHART כוללת דיאגרמת "סופגניה"

אופן הכתיבה:

PROC GCHART <אופציות שונות>;
BLOCK משתנים</אופציות שונות>;

HBAR משתנים</אופציות שונות>;
VBAR משתנים</אופציות שונות>;
HBAR3D משתנים</אופציות שונות>;
VBAR3D משתנים</אופציות שונות>;
PIE משתנים</אופציות שונות>;
PIE3D משתנים</אופציות שונות>;
DONUT משתנים</אופציות שונות>;
STAR משתנים</אופציות שונות>;
RUN;

דוגמא:

```
proc gchart;
 block PD;
 hbar PD;
 hbar3d PD;
 vbar PD;
 vbar3d PD;
 pie PD;
 pie3d PD;
 donut PD;
 star PD;
run;
```

הפלט הבסיסי המתקבל לכל סוג של איור ב-PROC GCHART מוצג באיור 21.

ההוראה BLOCK

ההוראה BLOCK מציירת היסטוגרמה אנכית תלת ממדית. ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים. במקרה בו מוגדרים מספר משתנים, SAS תיצור כל איור בדף חדש.

אופן הכתיבה:

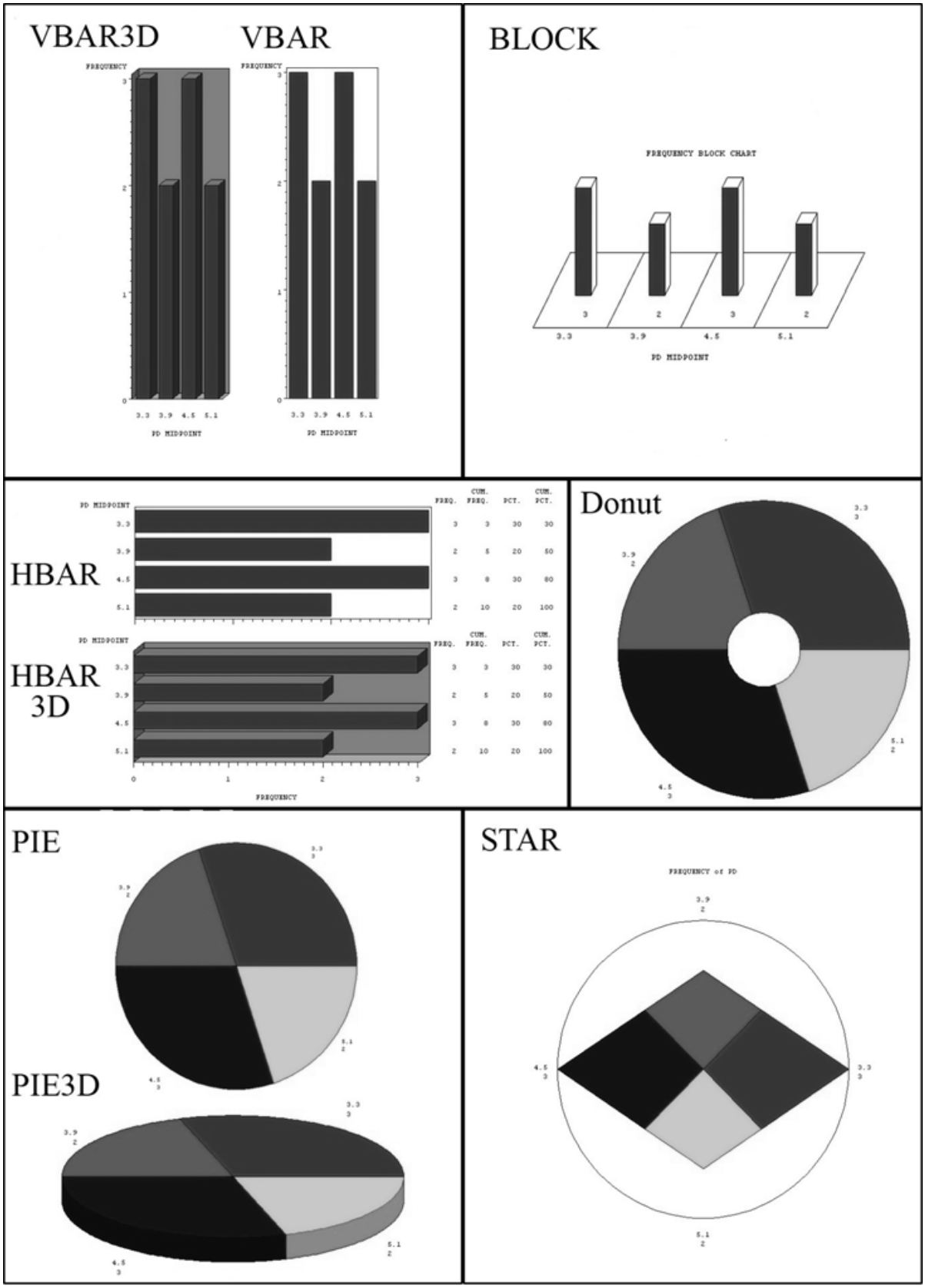
BLOCK רשימת משתנים</אופציות שונות>;

ההוראות HBAR, HBAR3D, VBAR ו-VBAR3D

הוראות אלה מציירות היסטוגרמה אנכית או אופקית, דו או תלת ממדית. ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים.

אופן הכתיבה:

VBAR | VBAR3D | HBAR | HBAR3D רשימת משתנים</אופציות שונות>;



איור 21 – הצורה הבסיסית של כל התרשימים המופקים על ידי PROC GPLOT

ההוראה PIE ו-PIE3D

ההוראות אלה יוצרות דיאגרמת פאי דו או תלת ממדית (בהתאמה).

אופן הכתיבה:

PIE | PIE3D רשימת משתנים</אופציות שונות>;

ההוראה DONUT

ההוראה DONUT יוצרת דיאגרמת "סופגניה", בה הגודל של כל חתיכה מייצג את הערך היחסי של כל קבוצת תצפיות ביחס לשאר הערכים.

אופן הכתיבה:

DONUT רשימת משתנים</אופציות שונות>;

ההוראה STAR

ההוראה STAR יוצרת דיאגרמת כוכב. ניתן להגדיר משתנה אחד, או מספר משתנים מופרדים על ידי רווחים.

אופן הכתיבה:

STAR רשימת משתנים</אופציות שונות>;

הערה: כמו ב-PROC GPLOT, גם להוראות השונות ב-PROC GCHART יש אופציות רבות ומגוונות, המאפשרות התאמה אישית של התרשימים. עם זאת, כפי שכבר ציינו, SAS איננה תוכנה גראפית במהותה, ולכן לא נפרט בספר זה על אופציות אלה.

תרגול עצמי – תרשימים וגרפים

תרגיל 42

מתוך ההנחה ש"תמונה שווה אלף מילים" חוקר רצה לבחון בצורה גרפית את הקשר בין אחוז הפגיעה בגפיים עליונות ותחתונות (המשתנים upper ו - lower) לבין יכולת ניידות (המשתנה mobile). להלן הנתונים על אחוז פגיעה בגפיים עליונות ותחתונות, וכן על יכולת הניידות של 14 פגועי גפיים:

upper	lower	mobile
78.6	37.3	1.01
67.5	48.4	1.01
9.8	35.2	1.01
43.0	45.0	1.01
67.2	51.8	0.39
80.0	43.2	0.31
17.5	22.8	0.30
9.5	12.1	0.10
83.9	12.0	0.07

13. 3 14.8 0.05
0 0 0.05
0 16.7 0.02
13.8 37.6 0.01
0 0. 3 0.01

כתוב תוכנית SAS להפקת התרשים הרצוי.

תרגיל 43

להלן נתונים על הוצאה חודשית (בדולרים) של 3 פרופסורים בפקולטה לאווירונאוטיקה במהלך השנה האחרונה:

month prof1 prof2 prof3
1 10258 9000 8000
2 9100 8500 9500
3 13000 7900 7200
4 8000 11000 10000
5 9500 9980 9500
6 7900 7900 8790
7 11000 9600 8600
8 8923 12000 6000
9 7500 9999 7890
10 9870 8560 9870
11 9000 7960 6999
12 8523 8700 8000

א. אחד הפרופסורים (prof1) מעוניין להציג את ההוצאות החודשיות שלו על גרף, כדי לבדוק באילו חודשים הוא חרג מהתקציב החודשי שלו (העומד על 10000$ בחודש). כתוב קוד SAS להצגת ההוצאה (בציר y) לפי החודש (בציר x). הוסף קו ייחוס אופקי ב-10000, כדי לסייע לפרופסור לראות מתי הוא חורג מהתקציב.

ב. הדיקן של הפקולטה רצה לראות מי מבין 3 הפרופסורים נוטה יותר לחרוג מהתקציב החודשי שלו. כדי לעשות זאת הוא החליט להציג את כל הנתונים על מערכת צירים אחת (על גרף אחד). כתוב קוד SAS לסייע לדיקן להציג את כל הנתונים על אותה מערכת צירים. הוסף למערכת את קו הייחוס מהסעיף הקודם.

תרגיל 44

להלן נתוני מכירות של סוכנות הונדה מסוימת במהלך התקופה האחרונה. הנתונים כוללים את שם סוכן המכירות (המשתנה Sales_man), סוג המודל שנמכר (המשתנה Brand), וכמות היחידות שנמכרה מאותו מודל על ידי אותו סוכן מכירות (המשתנה Cars_sold):

Sales_man Brand Cars_sold
Joe Civic 134
Joe Civic 238
Joe Odyssey 98
Joe Odyssey 88
Joe Element 200
Joe Element 105
Joe Accord 35
Joe Accord 128
John Civic 239
John Civic 201
John Odyssey 204

John Odyssey 197
John Element 187
John Element 200
John Accord 64
John Accord 152
Ben Civic 155
Ben Civic 219
Ben Odyssey 163
Ben Odyssey 155
Ben Element 89
Ben Element 287
Ben Accord 143
Ben Accord 133

מנהל הסוכנות רצה לבדוק את היקף המכירות של כל מודל, וכן לבדוק כמה מכוניות מכל מודל מכר כל אחד מסוכני המכירות שלו. כתוב קוד SAS להפיק תרשים כזה, אשר יציג בו זמנית את היקף המכירות של כל מודל, וכן את התרומה היחסית של כל איש מכירות עבור כל אחד מהמודלים.

תרגיל 45

להלן ציונים (באותיות) של 42 סטודנטים במבחן סיום של אחד הקורסים שלהם:

A B C B B C C B D B C B D A B A F B C C D
D A B C C B C D C B C F C D B D F C A B B

א. כתוב קוד SAS להצגת התפלגות ציוני הסטודנטים בתרשים עוגה.
ב. כתוב קוד SAS להצגת התפלגות ציוני הסטודנטים כאחוזים ולא כשכיחויות.

פרק 14
פונקציות SAS מתקדמות

עבודה עם מאקרו

מאקרו ב-SAS הם משתנים או קטעי קוד שאינם קשורים לקובץ נתונים או לפרוצדורה ספציפית. למעשה, ביסודו של דבר מאקרו ב-SAS הם כלים לתפעול והחלפה של מחרוזות טקסט. כאשר מגדרים משתנה מאקרו או קוד מאקרו, יוצרים איזכור (reference). כאשר קוראים לאיזכור מאקרו זה בקטע קוד SAS כלשהו, מעבד המאקרו של SAS פשוט מחליף את הטקסט הקיים במאקרו (האיזכור) עם הטקסט המופיע בקוד התוכנית. כך, לדוגמא, ניתן באמצעות מאקרו להגדיר משתנים (ולצקת אליהם תוכן בעת ההגדרה או מאוחר יותר בתוך הקוד), לכתוב קטעי קוד ארוכים ומורכבים שיורצו מספר פעמים במהלך תוכנית ללא הצורך להקליד אותם שוב ושוב, וכדומה.

עבודה עם מאקרו ב-SAS מתבצעת בשני אופנים:

1. משתני מאקרו
2. קוד מאקרו

הערה: העבודה עם מאקרו ב-SAS היא מורכבת וכוללת הוראות ופונקציות רבות ומגוונות. תת הפרק הנוכחי נועד לתת מבוא כללי לעבודה עם מאקרו ב-SAS, אך הוא אינו מהווה מדריך מקיף למאקרו.

משתני מאקרו

משתנה מאקרו הוא משתנה SAS שלא שייך לקובץ נתונים ספציפי. כל משתנה מאקרו יכול לקבל רק ערך אחד, כאשר ערך זה יכול להיות מספר או מחרוזת כלשהי (המייצגת ערך של משתנה, שם משתנה, כותרת וכדומה).

הגדרת משתני מאקרו נעשית או מחוץ לקטעי קוד (משתנים גלובאלים) או בתוך קטע קוד מאקרו (משתנים לוקאלים). ההבדל בין שני סוגי משתני מאקרו אלה הוא שניתן להשתמש במשתנים גלובאלים בכל מקום בעוד שניתן להשתמש במשתנים לוקאלים רק בתוך קוד המאקרו בו הם מוגדרים. כדי להגדיר משתנה מאקרו, יש להשתמש בהוראה LET (הוראת מאקרו).

אופן הכתיבה:

%LET שם משתנה המאקרו = ערך משתנה המאקרו;

דוגמא:

```
%let macvar = 3;
```

בדוגמא זו יצרנו משתנה מאקרו בשם macvar, המכיל את הערך המספרי 3.

הערה: מאחר ומשתני מאקרו הם בהגדרה משתני מחרוזת (ולא משתנים מספריים – למרות שניתן להציב בהם מספרים), אין צורך להכניס את המשתנים האלה למרכאות (גם אם הם מכילים מחרוזות). בנוסף, כדי ליצור את משתנה המאקרו הזה (או כל משתנה מאקרו אחר), יש לוודא כי שורת הקוד שבדוגמא לא מופיעה בתוך שום DATA STEP או PROC STEP.

כאשר רוצים לקרוא למשתנה מאקרו בתוך קטע קוד (לדוגמא ב- DATA STEP), יש לכתוב את שם המשתנה עם סימן חיבור (&) בתחילתו. בדומה, אם רוצים לקרוא למשתנה מאקרו מחוץ לקטע קוד SAS, יש לכתוב את שם המשתנה עם סימן החיבור (&) בתחילתו, ללא הסימן נקודה פסיק (;) בסוף שורת הקוד.

דוגמא לקריאה למשתנה מאקרו בתוך תוכנית SAS:

```
data macros; set macros;
var1 = var1 * &macvar;
```

דוגמא לקריאה למשתנה מאקרו מחוץ לתוכנית SAS (כאשר, למשל רוצים לקרוא למשתנה מתוך קוד מאקרו):

```
&macvar
```

משתני מאקרו אוטומאטים

כאשר פותחים את SAS, היא יוצרת באופן אוטומטי מספר משתני מאקרו הזמינים למשתמש. טבלה 13 מציגה רשימה חלקית משתני המאקרו האוטומטיים הזמינים ב-SAS:

שם משתנה	תוכן	דוגמא
Sysdate	התאריך של היום בו מורצת התוכנית, בפורמט של DDMMMYY	15JUL89
Sysday	היום בשבוע שבו מורצת התוכנית	Wedensday
systime	הזמן שבו SAS נפתח לראשונה באותו יום	11:30
sasver	מספר הגרסא של התוכנה SAS המותקנת על המחשב	9.2

טבלה 13 – משתני מאקרו אוטומטיים

ניתן להשתמש בכל אחד ממשתנים אלה (כמו גם במשתני מאקרו מוגדרים על ידי המשתמש) בכל מקום ב-SAS. ניתן גם לשלב משתנים מאקרו אלה (כמו גם משתני מאקרו אחרים) אחד עם השני, או בתוך משפט.

דוגמא:

```
%let First_name = first;
%let Last_Name = last;
%let Full_Name = &First_name &Last_Name;
```

בדוגמא זו יצרנו 3 משתני מאקרו: המשתנה הראשון מכיל שם פרטי, המשתנה השני מכיל שם משפחה, והמשתנה השלישי מאחד את שני המשתנים הראשונים שיצרנו, ומכיל את השם המלא (פרטי + משפחה).

דוגמא:

```
title "This code was run in &sysday, &sysdate";
```

בדוגמא זו, כתבנו כותרת לכל פלט שיצא מהקוד SAS המלא שניצור, הכולל את היום והתאריך שבו נריץ את התוכנית. כתוצאה מדוגמא זו, לפני הפלט תופיע שורת הכותרת הבאה:

This code was run in Wednesday, 18AUG10

ההוראה PUT

ההוראה PUT מאפשרת להציג את ערכם של משתני המאקרו. עם זאת, בניגוד למשתנים רגילים (שמוצגים בחלון Output) ההוראה PUT מציגה את ערכי משתני המאקרו בחלון Log.

אופן הכתיבה:

%PUT משתנה מאקרו

למעשה, כל מה שייכתב לאחר ההוראה PUT יודפס בחלון Log, ולא רק ערכו של משתנה המאקרו.

דוגמא:

```
%put ====== &Full_name ======;
```

בעקבות קוד זה, החלון Log יציג (בין היתר) את:

```
2577  %put ====== &Full_name ====== ;
====== first last ======
```

הערה: כדי להציג את כל משתני המאקרו המוגדרים בקוד, ניתן לכתוב את ההוראה %PUT _ALL_. הוראה זו תציג בחלון Log הן את משתני המאקרו שהוגדרו על ידי המשתמש (תחת GLOBAL) והן כל כל משתני המאקרו האוטומאטיים הזמינים ב-SAS (תחת AUTOMATIC).

ההוראה STR

ההוראה STR מאפשרת לכלול בתוך משתנה מאקרו תווים מיוחדים בתוך סוגריים, כגון נקודה פסיק (;). כך, ניתן להגדיר קטעי קוד SAS בתוך משתנה מאקרו.

אופן הכתיבה:

%LET שם משתנה מאקרו = %STR (קטעי קוד רצויים, מופרדים באמצעות נקוה פסיק);

דוגמא:

```
%let prnt = %str (proc print noobs; var z; run;);
```

בדוגמא זו, יצרנו משתנה מאקרו בשם prnt. כאשר נקרא למשתנה מאקרו זה, בעצם נבקש מ-SAS להדפיס את המשתנה z (באמצעות הרצת PROC PRINT).

קוד מאקרו

קוד מאקרו הוא קטע קוד שלא קשור לשום DATA STEP או PROC STEP, ואשר יכול להכיל משפטים וטיעונים לוגיים מורכבים (כולל צעדי DATA ו-PROC שלמים). לקוד המאקרו שפה משלו (SAS macro language), הכוללת הוראות ייחודיות שיפורטו להלן.

אופן הכתיבה:

```
%MACRO שם מאקרו;
קטעי קוד המאקרו
%MEND <שם מאקרו>;
```

ההוראה MACRO מגדירה ל-SAS את תחילת קטע קוד המאקרו, וההוראה MEND מודיעה ל-SAS על סיום קטע הקוד. כל מה שמופיע בין ההוראה MACRO להוראה MEND נכלל בקוד המאקרו.

טיפ ממומחה: למרות שלא חובה לכתוב את שם המאקרו אחרי ההוראה MEND, כדאי לעשות זאת למען למען הסדר הטוב. כך יהיה הרבה יותר קל לבחון את הקוד לאחר הכתיבה שלו. עם זאת, כאשר כותבים את שם המאקרו אחרי ההוראה MEND, יש לוודא כי הוא תואם לשם המאקרו בהוראה MACRO.

דוגמא:

```
%macro prnt;
  proc print noobs;
   var z;
  run;
%mend prnt;
```

בדוגמא זו כתבנו קוד מאקרו שמבצע את אותה הפעולה של משתנה המאקרו שהגדרנו בתת הפרק הדן במשתני מאקרו – מריץ את PROC PRINT כדי להדפיס את המשתנה z.

הערה: סימני הנקודה פסיק המופיעים בסוף כל שורה בקוד המאקרו הם לא חלק מקוד המאקרו, אלא חלק מהקוד SAS שקוד המאקרו בא להחליף. דהיינו, כאשר כותבים קוד מאקרו, אין חובה לסיים כל שורה בנקודה פסיק, אלא אם רוצים שקוד המאקרו "יחקה" קוד SAS.

לדוגמא:

```
%macro nosemi;
 example
%mend nosemi;
```

קוד מאקרו זה מכיל רק את המחרוזת "example". כאשר נקרא למאקרו nosemi, למעשה נבקש לקרוא למחרוזת example (למשל כדי להכניס ערך זה לתוך משתנה מחרוזת, להוסיף ערך זהלכותרת של תרשים וכדומה).

בדומה למשתני מאקרו, כאשר רוצים לקרוא לקוד מאקרו יש לכתוב את שם קוד המאקרו עם הסימן % בתחילתו. כאשר הקריאה לקטע הקוד נעשית מחוץ לקטע קוד SAS מסוים (דהיינו מחוץ ל-DATA STEP או PROC STEP ספציפיים), בדומה לקריאה של משתני מאקרו, אין צורך לסיים את המשפט עם סימן הנקודה-פסיק (;).

דוגמא:

```
%prnt
```

דוגמא זו תקרא למאקרו prnt.

ניתן להגדיר משתני מאקרו בתוך סוגריים בצמוד להוראה מאקרו. משתנים אלה משמשים כפרמטרים של מאקרו אשר מאפשרים להעביר מידע מקוד SAS אל המאקרו. לדוגמא, ניתן להגדיר קוד מאקרו עם שמות משתני מאקרו כלליים, ואז באמצעות קוד SAS לצקת לתוך הפרמטרים האלה שמות שונים של משתני SAS, כדי להריץ את אותו קוד מאקרו על משתני SAS שונים.

אופן הכתיבה:

% MACRO שם מאקרו (שם משתנה1 = , שם משתנה*n* =);

דוגמא:

```
%macro prnt (par = );
 proc print noobs;
  var &par;
  run;
%mend prnt;
```

בדוגמא זו אנחנו מגדירים קוד מאקרו ל-PROC PRINT, ומבקשים להדפיס את המשתנה ששמו תואם לתוכן של משתנה המאקרו par.

כאשר נרצה מאוחר יותר להריץ קוד מאקרו זה, נצטרך להגדיר לקוד המאקרו את התוכן של משתנה המאקרו.

אופן הכתיבה:

%שם המאקרו (שם משתנה המאקרו = מחרוזת)

דוגמא:

```
%prnt (par = x z)
```

בדוגמא זו הגדרנו למאקרו prnt שתוכנו של המשתנה (הפרמטר) par יהיה x ו-z. לכן, כאשר נריץ קוד מאקרו זה, בעצם נבקש מ-SAS (באמצעות PROC PRINT) להדפיס את המשתנים x ו-z.

הערה: כתבנו את x ו-z מופרדים על ידי רווחים, מאחר וההוראה VAR ב-PROC PRINT דורשת שהמשתנים הנדרשים להדפסה יופיעו מופרדים באמצעות רווחים. באם היינו רוצים לכלול בקוד המאקרו הוראת SAS הדורשת שהמשתנים יופרדו באמצעות פסיקים, היינו צריכים להכניס למשתנה par את שמות המשתנים הרצויים כולל פסיקים.

השימוש בפרמטרים יכול להיות יעיל מאוד ולקצר מאוד את קוד התוכנית במקרה בו רוצים, למשל, להריץ פרוצדורות או קטעי קוד זהים על כמה משתנים.

דוגמא:

```
%macro regres (par = , par2 = );
 proc reg data = &par2;
  model &par;
  run;
%mend regres;
```

```
%regres (par = x = z, par2 = data1)
%regres (par = y = t, par2 = data2)
%regres (par = w = q, par2 = data3)
```

בקוד מאקרו זה הגדרנו הרצה של מודל רגרסיה (תוך שימוש ב-PROC REG), וכל פעם אנחנו מריצים קוד זה על קובץ נתונים שונה וסט נתונים שונה. הרצה של קוד מאקרו זה תיהיה זהה לכתיבה של 3 צעדי REG שונים:

```
proc reg data = data1;
 model x = z;
run;
proc reg data = data2;
 model y = t;
run;
proc reg data = data3;
 model w = q;
run;
```

הוראות תנאי

ניתן לכלול גם הוראות תנאי כגון IF…THEN…ELSE או DO…WHILE… בקוד מאקרו. בעזרת הוראות אלה ניתן ליצור תנאים לוגיים בהם יתקיימו קטעי הקוד או לא, או ליצור תנאים בהם יפעל קטע קוד SAS אחד ובהם יפעל קטע קוד SAS אחר.

הוראת תנאי IF…THEN…ELSE

אופן הכתיבה:

```
%MACRO שם קוד מאקרו;
%IF תנאי כלשהו %THEN %DO;
  הגדרת פעולות לתנאי
  %END;
<%ELSE ;<%IF תנאי כלשהו %THEN> %DO;
  הגדרת פעולות לתנאי
  %END;>
%MEND שם קוד מאקרו;
```

דוגמא:

```
%macro condi(dataset =, var =);
%if  &dataset = data %then %do;
  proc reg data = &dataset;
    model &var;
  run;
 %end;
%else %do;
  proc corr spearman;
```

```
   var &var;
  run;
 %end;
%mend condi;
```

בדוגמא זו, הגדרנו קוד מאקרו שיריץ ניתוח רגרסיה על סט מסוים של נתונים, ויריץ ניתוח מתאמים על סט אחר. לכן, כאשר נקרא לקוד המאקרו באמצעות ההגדרה שלהלן:

```
%condi (dataset = data, var = x = z)
```

נבקש להריץ מודל רגרסיה לניבוי x באמצעות z. לעומת זאת, כאשר נקרא לקוד באמצעות ההגדרה שלהלן:

```
%condi (dataset = data2, var = q s)
```

נבקש לחשב מתאם בין q ל-s.

הוראת תנאי DO…WHILE

אופן הכתיבה:

```
%MACRO שם קוד מאקרו;
 %DO %WHILE;
   הגדרת פעולות לתנאי
 %END;
 <%ELSE % DO;>
   <הגדרת פעולות לתנאי>
 <%END>;
%MEND שם קוד מאקרו;
```

הערה: בדומה להוראת תנאי DO…WHILE, ניתן גם להגדיר הוראת תנאי מאקרו DO…UNTIL.

פונקציות מאקרו

פונקציות מאקרו הם פונקציות המעבדות הוראה (אחת או יותר) ומחזירות תוצאה כלשהי. להלן מספר מצומצם של פונקציות מאקרו מתוך מגוון פונקציות המאקרו הקיימות ב-SAS:

1. הפונקציה index – פונקציה זו בודקת את המיקום של תו מסוים בתוך מחרוזת (ההופעה הראשונה של התו המוגדר במקרה בו הוא מופיע כמה פעמים באותה מחרוזת), ומחזירה אותו כערך מספרי.
   אופן הכתיבה:

```
%index (מקור,תו);
```

   המקור הוא משתנה המחרוזת והתו הוא תו המחרוזת הרצוי.

   דוגמא:

```
%let source = an example sentence;
%let result = %index(&source, x);
%put X appears at the &result.'th position.;
```

דוגמא זו תפיק לחלון Log את המשפט הבא:

```
X appears at the 5'th position.
```

2. הפונקציה length – פונקציה זו בודקת ומחזירה את אורך התווים של מחרוזת (כולל רווחים).
אופן הכתיבה:

%length (שם משתנה|מחרוזת כלשהי);

דוגמא תוך שימוש בשם משתנה:

```
%let source = an example sentence;
%let result = %length(&source);
%put ***** The sentence has &result words *****;
```

דוגמא תוך שימוש במחרוזת כלשהי:

```
%let result = %length(an example sentence);
%put ***** The sentence has &result characters *****;
```

בשני המקרים, נקבל את אותה תוצאה. יש לציין כי ניתן לכתוב גם מספר משתנים או שילוב בין מחרוזות ומשתנים.
הפונקציה length תחזיר את מספר התווים (כולל רווחים) הכולל של כל המשתנים ו/או התווים המופיעים בהגדרת הפונקציה.

3. הפונקציה str – פונקציה זו חוסמת (ממסכת) תווים מיוחדים (לדוגמא נקודה-פסיק). התפקוד של פונקציה זו, כמו גם אופן הכתיבה שלה נידון בתת הפרק הדן במשתני מאקרו, ולכן לא נחזור כאן על הדיון בפונקציה זו.

SAS SYSTEM OPTIONS

SAS SYSTEM OPTIONS הן סט הוראות כלליות, השולטות על הדרך שבה SAS מבצעת פעולות, מפיקה פלט, ומעבדת את הנתונים.

ההבדל בין SYSTEM OPTIONS ל-DATA STEP OPTIONS הוא שברגע שמגדירים SYSTEM OPTIONS, הגדרות אלה נשארות קבועות מעבר לכל צעדי ה-DATA וה-PROC הקיימים בהרצה ספציפית של SAS, בעוד ש- DATA STEP OPTIONS רלוונטיות רק לצעד בו הן מופיעות.

את ה-SYSTEM OPTIONS ניתן להגדיר או בכמה אופנים, כגון באמצעות הפרוצדורה OPLOAD, ההוראה OPTIONS, הקובץ autoexec ועוד. בספר זה, למעט חריגים בודדים, נתמקד בהגדרת אופציות באמצעות ההוראה OPTIONS, אותה יש לכתוב בקוד נפרד (מחוץ לצעד DATA או PROC מסוימים).

ל-SAS יש מגוון רחב מאוד של SYSTEM OPTIONS. בפרק זה של הספר נסקור רק האופציות השכיחות והשימושיות ביותר שכדאי למשתמשי SAS להכיר כדי לבצע התאמה אישית ל-SAS.

ההוראה OPTIONS

ההוראה OPTIONS נועדה להגדיר אופציות מערכת של SAS.

אופן הכתיבה:

OPTIONS אופציות שונות;

אופציות של ההוראה OPTIONS

אופציות הקשורות לקובץ הפלט:

1. האופציה nocenter/center – אופציה זו מגדירה האם הפלט בקובץ הפלט (בחלון Output) ימורכז לאמצע העמוד או לא. כברירת מחדל, SAS מציגה את הפלט מיושר לאמצע העמוד. באמצעות האופציה nocenter ניתן לבטל ברירת מחדל זו. במקרה כזה, הטקסט בקובץ הפלט יישור לשמאל.
אופן הכתיבה:

nocenter | center

2. האופציה nodate/date – אופציה זו מגדירה האם יופיעו התאריך והשעה בקובץ הפלט או לא. כברירת מחדל, מוגדרת האופציה date, והתאריך והשעה מוצגים. ביטול ברירת המחדל נעשה באמצעות הגדרת האופציה nodate.
אופן הכתיבה:

nodate | date

3. האופציה nonumber/number – אופציה זו מגדירה האם יופיעו מספרי עמוד בקובץ הפלט. כברירת מחדל SAS מציגה מספרי עמודים. כדי לבטל ברירת מחדל זו, יש להגדיר את האופציה nonumber.
אופן הכתיבה:

nonumber | number

4. האופציה linesize – אופציה זו מגדירה את רוחב העמוד (בתווים) של החלון Editor ושל החלון Log.
אופן הכתיבה:

linsize = מספר שלם בין 64 ל-256 | MAX | MIN

הערה: הגדרת ה-linesize כ-MAX מגדירה בעצם את רוחב העמוד ל-256 תווים, והגדרה כ-MIN ל-64 תווים.

5. האופציה pagesize – אופציה זו מגדירה את מספר השורות שיכולות להיות מודפסות בכל עמוד בחלון Output.
אופן הכתיבה:

pagesize = מספר שלם בין 15 ל-32767 | MAX | MIN

6. האופציה formchar – אופציה זו מגדירה את התווים שיצרו את גבולות הטבלאות ו/או צירי הגרפים המופקים על ידי הפרוצדורות של SAS (כגון PROC FREQ,PROC PLOT ועוד). ניתן להגדיר באמצעות האופציה formchar עד 64 תווים. כאשר מגדירים פחות מ-64 תווים, SAS משלימה את יתרת התווים החסרים ברווחים.
אופן הכתיבה:

formchar = 'רשימת התווים הרצויים'

כאשר מגדירים את האופציה formchar, SAS משייכת כל תו לרכיב אחר בטבלה/גרף, בהתאם למיקום שלו במחרוזת המגדירה את רשימת התווים הרצויים. להלן מיפוי של מיקומים נבחרים במחרוזת לרכיב בטבלה/גרף:

התו הראשון (משמאל) – מגדיר גבול או ציר אנכי
התו השני – מגדיר את הגבול או הציר האופקי
התו השלישי – מגדיר את הפינה השמאלית העליונה של הגבול (בטבלה או בגרף)
התו הרביעי – מגדיר את הגבול האמצעי אנכי של השורה העליונה בטבלה
התו החמישי – מגדיר את הפינה הימנית העליונה של הגבול
התו השישי – מגדיר את הגבול הפינתי התחתון השמאלי האמצעי של טבלה
התו השביעי – מגדיר את השנתות של הגרף או את הגבול האמצעי אנכי של טבלה
התו השמיני - מגדיר את הגבול הפינתי התחתון הימני האמצעי של טבלה
התו התשיעי – את הפינה השמאלית התחתונה של הגבול
התו העשירי – מגדיר את הגבול האמצעי אנכי של השורה התחתונה בטבלה
התו האחד עשרה – מגדיר את הפינה הימנית התחתונה של הגבול

דוגמא:

```
formchar = '|-<|>|+|<|>'
```

הדוגמא הנוכחית תספק את עיצוב הטבלה הבאה:

```
<------------|------------>
|      x     |      y     |
|     Sum    |     Sum    |
|     20.1   |     30.2   |
<------------|------------>
```

טיפ ממומחה: ברירת המחדל של גבולות הטבלה ו/או הגרף ב-SAS עשויים להשתנות ממערכת הפעלה אחת לשנייה או מהגדרות מערכת אלו או אחרות. כדי ליצור פורמט אחיד של תצוגה לטבלאות וגרפים ב-SAS, שיתאימו לכל מערכת הפעלה ולכל הגדרות מערכת, מומלץ להשתמש בקטע הקוד הבא:

```
formchar="|----|"|---+=|-/\<>*";      "
```

אופציות הקשורות לקובץ הנתונים:

1. האופציה firstobs – אופציה זו מגדירה ל-SAS מאיזו תצפית בקובץ הנתונים יש להתחיל לקרוא את הקובץ.
אופן הכתיבה:

firstobs = מספר שלם בין 1 למספר התצפיות בקובץ

2. האופציה obs – אופציה זו מגדירה ל-SAS מהי התצפית האחרונה אותה יש לקרוא מתוך קובץ הנתונים.\
אופן הכתיבה:

obs = מספר שלם בין 1 למספר התצפיות בקובץ

אופציות שקשורות לטיפול בשגיאות:

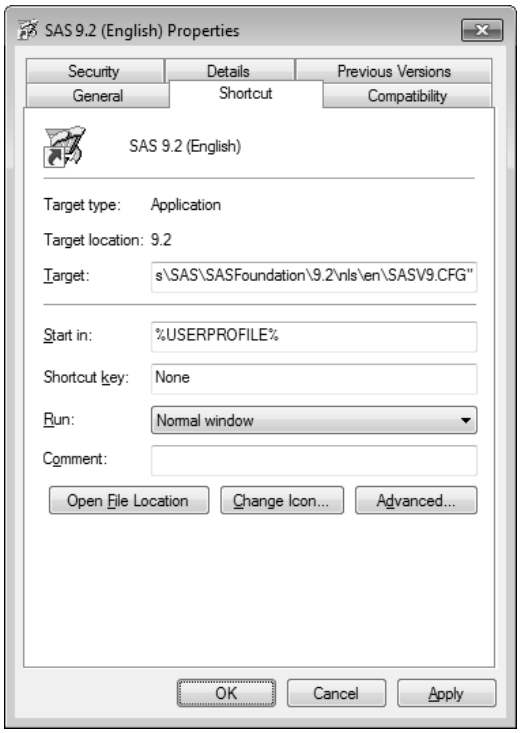
1. האופציה dkrocond – אופציה זו קובעת את הרמה של איתור שגיאות לקבצי נתונים במהלך עיבוד של האופציות drop, keep ו-rename. כברירת מחדל, SAS מתריעה אם קובץ הנתונים לא כולל משתנה המוגדר באחת מהאופציות הללו. באמצעות האופציה dkrocond ניתן לבטל את ההתרעה של שגיאה זו, או להגביל את ההתראה לכתיבה של הודעה בחלון Log בלבד.
אופן הכתיבה:

dkrocond = ERROR | WARNING | NOWORNING

כדי לבטל את ברירת המחדל error (התרעה + הודעת שגיאה), יש להגדיר את האופציה ל-nowarning. הגדרה ל-warning תפיק רק הודעת שגיאה לחלון Log.

2. האופציה nofmterr/fmterr – אופציה זו מגדירה האם SAS תפיק הודעת שגיאה כאשר היא לא מצליחה למצוא פורמאטים של משתנים שהוגדרו. כברירת מחדל, SAS מפיקה במקרה זה הודעת שגיאה ומספיקה את התהליך של קריאת הקובץ. כדי לבטל ברירת מחדל זו יש להגדיר את האופציה nofmterr. הגדרה זו תגרום ל-SAS להחליף את הפורמאטים שלא נמצאו בפורמאטים ברירת מחדל, ולהמשיך את תהליך קריאת הקובץ.
אופן הכתיבה:

nofmterr | fmterr



איור 22 – חלון המאפיינים של קיצור הדרך לתוכנת SAS

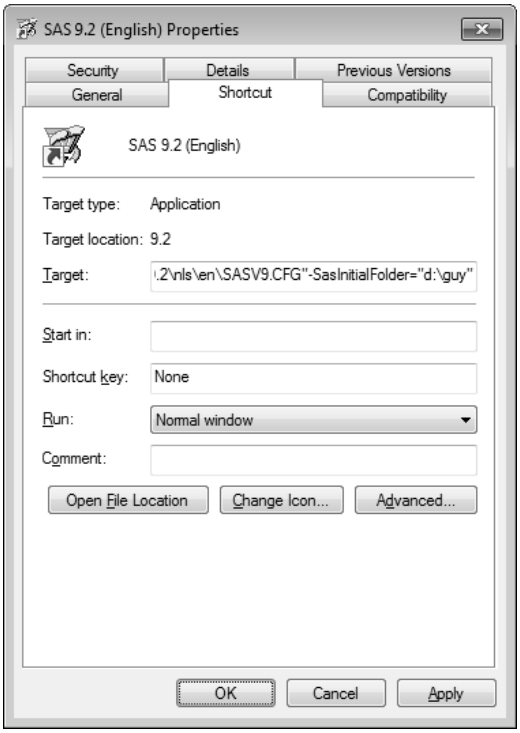
אופציות מערכת:

1. האופציה sasinitialfolder – אופציה זו מגדירה ל-SAS את תיקיית ברירת המחדל, שהתוכנה פותחת כדי לקרוא ממנה או לשמור לתוכה קבצים בחלון הדיאלוג שנפתח לקריאה או לשמירה של קובץ.

בניגוד לאופציות הקודמות שנסקרו כאן, אופציה זו לא מוגדרת בהוראה OPTION. כדי להגדיר את תיקיית ברירת המחדל יש לנקוט את הפעולות הבאות:

1. עמוד עם העכבר על קיצור הדרך לתוכנת SAS (בעודה סגורה) ולחץ על הכפתור הימני של העכבר.
2. בחר בתפריט "מאפיינים"
3. בחלון המאפיינים שנפתח (ראה איור 22) הוסף את השורה "C:/MyProject/SAS"=sasInitialFolde– לתיבת הדיאלוג Target (בסוף המחרוזת שכתובה שם – ראה דוגמא באיור 22).
4. מחק את הכתוב בתיבת הדיאלוג Start in (ראה איור 23).
5. לחץ על אישור.

כעת, כל פעם שתרצה לפתוח או לשמור קובץ, SAS תפנה ישר לתיקייה SAS שנמצאת בתוך התיקייה MyProjects שבכונן C (כמובן, שהקורא יצטרך להתאים נתיב זה לצרכים האישיים שלו).



איור 23 – חלון המאפיינים של קיצור הדרך לתוכנת SAS לאחר הגדרת sasInitialFolder

אינטראקציה בין SYSTEM OPTIONS לבין אופציות SAS אחרות

ישנה חפיפה רבה בין חלק מה-SYSTEM OPTIONS לבין אופציות אחרות ב-DATA STEP או ב-PROC STEP, כגון האופציות obs, formchar, וכדומה. למרות ש- SYSTEM OPTIONS משפיעים על כל ההרצה ולא רק על קטע קוד זה או אחר (בניגוד ל-DATA STEP או PROC STEP), אופציות SAS אחרות "דורסות" את ה-SYSTEM OPTIONS באותו הצעד בו הן מוגדרות.

לדוגמא, אם מוגדר ב-SYSTEM OPTIONS להתחיל לקרוא את התצפיות מהתצפית הרביעית (באמצעות האופציה firstobs), בעוד שב-DATA STEP מסוים מוגדר לקרוא את התצפיות מהתצפית הראשונה, אזי בכל אחד מהצעדים הכלולים בתוכנית SAS זאת התצפית הראשונה שתקרא תיהיה התצפית הרביעית, למעט אותו צעד DATA ספציפי, בו התצפית הראשונה שתקרא תהיה התצפית הראשונה.

תרגול עצמי – פונקציות SAS מתקדמות

תרגיל 46

כתוב מאקרו ל-SAS. על המאקרו לקרוא קובץ טקסט בשם targil46.txt המצוי בכונן C של המחשב. קובץ זה כולל שני משתנים: Day (היום בשבוע) ו-exp (הוצעה כספית בש"ח). הנח כי קובץ זה מתעדכן מידי יום, כאשר כל יום מעדכנים אותו בהוצאות כספיות שהתבצעו. לאחר קריאת קובץ הנתונים, המאקרו צריך לבדוק אם היום הוא יום שישי או לא. אם היום הוא לא יום שישי, הוא צריך להציג את ההוצאות של אותו יום בלבד, תחת הכותרת "Expenses from *day_x*" (ש-day_x מייצג את היום בשבוע בו מורץ המאקרו). אם היום הוא יום שישי, הוא צריך להציג הוצאה ממוצעת (כולל הוצאה מינימאלית והוצאה מקסימאלית) עבור כל יום בנפרד, תחת הכותרת "Averaged expense per day".

פתרון תרגילים

תרגיל 1

ראשית, ניצור קובץ txt בעל השם targil1, ונשים אותו בכונן C. קובץ זה יכלול, לדוגמא, את הנתונים הבאים:

```
01,M,27,83,185
02,F,23,52,169
03,M,24,75,178
04,F,26,57,170
05,F,25,50,167
```

כעת, נכתוב בחלון Editor של SAS את הקוד הבא:

```
Data targil1;
 infile 'c:\targil1.txt' dlm =',';
 input sub gender$ age weight height;
run;
```

הערה: מאחר והאופציה dsd הופכת את הפסיק (,) לברירת המחדל של התו המפריד בין תצפיות במשתנים השונים, ניתן היה בקוד זה גם להגדיר את האופציה dsd במקום להגדיר את האופציה dlm.

תרגיל 2

כדי לקרוא נתונים אלה מתוך ה-DATA STEP, נכתוב את הקוד הבא:

```
Data targil2;
input sub 1-2 gender$ 3 age 4-5 weight 6-7 height 8-10;
cards;
01M2783185
02F2352169
03M2475178
04F2657170
05F2550167
;
```

תרגיל 3

ראשית, נניח כי קובץ הנתונים הקיים עליו נעבוד בתרגיל זה נקרא targil3.

א. כדי ליצור קובץ נתונים חדש שיהיה העתק של קובץ הנתונים הישן, צריך פשוט לכתוב את הקוד הבא:

```
data targil3_new; set targil3;
```

ב. כדי "לדרוס" קובץ נתונים קיים, יש פשוט לכתוב את שורת הקוד הבאה:

```
data targil3; set targil3;
```

תרגיל 4

תחילה יש להכניס את הנתונים לזיכרון של SAS

```
data targil4;
input subject_number age gender$ choice$;
cards;
1 24 male safe
2 26 male risky
3 32 female risky
4 22 male safe
5 27 female safe
;
```

לאחר מכן, נכתוב את הקוד הבא כדי להשמיט את המשתנה של המספר הסידורי של הנבדק:

```
data targil4_new (drop = subject_number); set targil4;
```

תרגיל 5

כדי לשנות את שמות המשתנים, נכתוב את הקוד הבא (לאחר שהכנסנו לזיכרון התוכנה את קובץ הנתונים, כמו בשאלה הקודמת):

```
data targil5_new (rename =(age = gil gender$ = sex$)); set targil4;
```

תרגיל 6

```
data grades;
input subject$ grade1 grade2;
grade_mean=(grade2 + grade1)/2;
cards;
1 65 88
2 78 91
3 82 83
4 59 95
;
```

תרגיל 7

כדי להפיק את הציון המינימאלי בכל תצפית, נשתמש בפונקציה min:

```
data targil7; set targil7;
 min_w = min(weight1, weight2, weight3, weight4);
run;
```

תרגיל 8

```
data targil8; set targil8;
 if gender = 0;
 test = test + 3;
run;
```

תרגיל 9

ראשית, נעתיק את הנתונים לקובץ טקסט ונקרא אותם ב-SAS:

```
Data targil9;
 infile 'c:\targil9.txt';
```

```
 input sub q1-q10;
run;
```

א. כדי לחשב את הציון של כל סטודנט, נחשב את מספר התשובות הנכונות, ונכפיל אותו ב-10, על פי הקוד הבא:

```
data targil9; set targil9;
 garde = (10 * sum(q1, q2, q3, q4, q5, q6, q7, q8, q9, q10));
run;
```

ב. כדי ליצור את המשתנה החדש, נוסיף את שורת הקוד הבאה מתחת לשורה המגדירה את המשתנה grade:

```
if garde > 65 then over = 'True'; else over = 'False';
```

ג. כדי לייחס לכל תצפית את הערך M (המייצג גבר) של המשתנה "מין", ניצור בתחילה קובץ נתונים נוסף המכיל רק את הסימן M, ונקרא לו targil9b.txt. לאחר שנכניס אותו ל-SAS (באמצעות ההוראה infile), נכתוב את הקוד הבא:

```
data targil9c; set targil9;
 if _n_ = 1 then set targil9b;
run;
```

תרגיל 10

```
If age <= 10 then group = "child      ";
 else if 11 <= age <= 19 then group = "teenager   ";
 else if 20 <= age <= 29 then group = "young edult";
 else if 30 <= age <= 45 then group = "adult      ";
 else if 46 <= age <= 59 then group = "middleage  ";
 else group = "senior      ";
```

תרגיל 11

כדי לענות על שאלה זו, נשתמש בפונקציות ranuni ו-uniform:

```
data targil11 (drop = i);
 do i = 1 to 10;
  x=int(uniform(0)*10)+1;
  y=int(10*ranuni(0)+0);
  output;
 end;
run;
```

תרגיל 12

כדי לענות על שאלה זו, נגדיר מערך (בשם wb), ונקשר אותו לחמשת המשתנים. בנוסף, נגדיר את הערכים הראשונים של כל המשתנים כ - 0:

```
data targil12 (drop = x i j);
 array wb{5} sub grade1-grade3 ave_grade (0 0 0 0 0);
  do i = 1 to 40;
   do j = 1 to 3;
    x = int(100*ranuni(0) + 0);
    wb(j+1) = x;
    wb{5} = wb{5} + x;
```

```
   end;
   wb{5} = wb{5} / 3;
   wb{1} = i;
   output;
  wb{5} = 0;
  end;
 run;
```

הקוד הנוכחי מריץ 2 לולאות DO: לולאה אחת הרצה מ-1 עד 40, ולולאה שנייה הרצה מ-1 עד 3. הלולאה השנייה רצה בתוך הלולאה הראשונה, כך שעבור כל תצפית (מ-1 עד 40) אנחנו יוצרים 3 מספרים אקראיים בין 0 ל-100, ומכניסים אותם לתאים 2, 3 ו-4 של המערך, ובכך מקשרים אותם למשתנים grade1 עד grade3. במקביל, אנחנו מקשרים את מונה הלולאה i, הרץ מ-1 עד 40 לתא הראשון במערך, והוא מייצג לנו את מספר התצפית (המשתנה sub). לבסוף, אנחנו מכניסים את הסכום של המשתנים grade1 עד garde3 לתא החמישי של המערך, ובכל פעם לפני שמונה הלולאה הראשונה (i) מתקדם, אנחנו מחלקים סכום זה ב-3, כדי לחשב את הממוצע של שלושת המשתנים. שים לב כי בכתיבת הקוד הגדרו ל-SAS להוריד את משתני הלולאה (i ו-j), וכן את המשתנה הזמני x מקובץ הנתונים הסופי.

תרגיל 13

מאחר ואין חפיפה בין ערכים שונים של המשתנה Sub, אין צורך לאחד את קבצי הנתונים באמצעות ההוראה MERGE, וניתן להשתמש פשוט בהוראת SET:

```
data targil13; set targil13a targil13b;
```

תרגיל 14

ראשית, ניצור קבצי טקסט המכילים את קבצי הנתונים, ונטען אותם לתוך SAS:

```
Data targil14a;
 infile 'c:\targil14a.txt';
 input height weight;
run;
```

```
Data targil14b;
 infile 'c:\targil14b.txt';
 input height gender$;
run;
```

א. כדי למזג את שני הקבצים יש ראשית לוודא כי הם מסודרים על פי סדר עולה במשתנה הרלוונטי (גובה). לאחר מכן, נכתוב את הקוד הבא:

```
data targil14; merge targil14a targil14b;
 by height;
run;
```

במצב כזה, יתקבל קובץ הנתונים הבא:

height	weight	gender
168	56	
169	.	F
170	.	F

174	85	
180	75	M
180	80	M
180	.	M
190	96	M

הערה: בקובץ הנתונים הנתון, הקבצים מסודרים בסדר עולה על פי המשתנה הרלוונטי. בפרק הבא נלמד כיצד למיין משתנים לפי סדר עולה, במקרה בו המשתנה הרלוונטי לא מסודר לפי סדר עולה.

ב. כדי לכלול רק את הנתונים המגיעים משני הקבצים, נכתוב את הקוד הבא:

```
data targil14; merge targil14a (in = in14a) targil14b (in = in14b);
 by height;
 if in14a and in14b;
run;
```

במצב כזה, קובץ הנתונים שיתקבל יהיה:

height	weight	gender
180	75	M
180	80	M
190	96	M

תרגיל 15

```
proc sort data = targil15;
 by gender descending grade;
run;
```

תרגיל 16

```
data targil16;
input age height sex;
cards;
21 168 2
26 173 1
24 181 1
27 158 2
30 185 1
24 173 1
;

proc print data=targil16 split ='*' obs =
                                 'observation*Number*==========';
  var age height sex;
  label age='age**====='
        height='height**====='
        sex='sex**========';
  title 'write a SAS code to replicate this output';
run;
```

תרגיל 17

```
proc print data = targil17 (obs = 3) noobs;
 where gender = 1;
 var sub choice;
run;
```

תרגיל 18

קוד זה מניח כי יצרת קובץ נתונים בשם targil18 המכיל את נתוני התרגיל.

```
proc format;
value sex
  0 = 'Male'
  1 = 'Female';
value risk
  low - < 0.5 = 'Risk seeker'
  0.5 < - high = 'Risk aversive'
  0.5 = 'Indifferent';
run;
```

כפי שציינו, הפורמאטים שהוגדרו כללים, ולא ספציפיים למשתנים של קובץ הנתונים. כדי לעשות את הקישור, נוכל למשל לכלול את ה-PROC PRINT שלהלן:

```
proc print data=targil18;
 format gender sex. prop risk.;
run;
```

תרגיל 19

ראשית, ניצור קבצי טקסט המכילים את קבצי הנתונים, ונטען אותם לתוך SAS:

```
Data targil19;
 infile 'c:\targil19.txt';
 input treat sex HR;
run;
```

הבעיה בתרגיל זה היא שלא ניתן להגדיר יותר ממשתנה ID (המשתנה שמעבר עליו יש לבצע את השחלוף) אחד. לכן, נצטרך ליצור פרוצדורת שחלוף לכל משתנה מין בנפרד, ולאחר מכן לאחד בין הקבצים:

```
proc transpose data = targil19 out = targil19_male prefix =
                                                    male_treat;
 where sex = 1;
 var hr;
 id treat;
run;

proc transpose data = targil19 out = targil19_female prefix =
        female_treat;
 where sex = 2;
 var hr;
```

```
 id treat;
run;

data targil19_final; merge targil19_male targil19_female; run;
```

במצב כזה, בו משתנה ה-ID השני הוא רק בעל שתי רמות (זכר או נקבה), הקוד הוא די קצר ופשוט. ואולם, אם היה לנו משתנה עם מספר גדול של ערכים, הקוד היה ארוך ומורכב מאוד. כדי להימנע מליצור שורות רבות של קוד, ניתן להשתמש בשיטה הבאה:

ראשית, ניצור קובץ נתונים חדש, המכיל משתנה חדש המהווה כל צירוף אפשרי בין שני משתני ה-ID:

```
data targil19; set targil19_final;
 sextreat = compress(treat)||compress(sex);
run;
```

כעת, יש לנו רק משתנה אחד המוגדר כ-ID, כך שניתן לעשות פרוצדורת שחלוף רגילה:

```
proc transpose data = targil19_final out = final;
 var hr;
 id sextreat;
run;
```

הערה: קוד זה יעבוד ללא קשר לכמות הערכים שיש לכל משתנה.

תרגיל 20

א. 1.

```
proc format;
 picture per
   0-1='000%' (multiplier=100)
   other='Out of bounds';
run;
```

2. כדי לכלול גם את הערך המקורי של התצפית החריגה, נחליף את השורה בקוד המתחילה במילה other עם השורה שלעיל:

```
 other='Value of 0.0 is Out of bounds';
```

ב. נוסיף לקוד שלעיל את הוראת ה-Picture הבאה:

```
picture money
  low-high='000' (prefix='NIS ' multiplier=0.1);
```

ג.

```
proc print data=targil20;
 format points money. prop per.;
run;
```

תרגיל 21

```
proc import datafile='c:\targil21.xls' out = targil21;
```

```
  sheet="sheet1";
run;
```

תרגיל 22

```
proc import datafile='c:\targil22.txt' out = targil22;
 getnames = no;
 DELIMITER = ',';
 datarow=3;
run;
```

תרגיל 23

כדי להעביר את קבצי הנתונים הקיימים בספריית ברירת המחדל של SAS לספרייה אחרת המוגדרת על ידי המשתמש, ניתן להשתמש בהוראה COPY ב-PROC DATASETS:

```
proc datasets;
 libname myworks 'D:\MyDocuments';
 copy out = myworks in = work;
run;
```

תרגיל 24

כדי למחוק קבצי נתונים מסויימים ולשמור קבצי נתונים אחרים, ניתן להשתמש או בהוראה DELETE, והלגדיר את קבצי הנתונים שרוצים למחוק:

```
proc datasets;
 delete raw_1 raw_2 exp1;
run;
```

או בהוראה SAVE, ולהגדיר את קבצי הנתונים שרוצים להשאיר:

```
proc datasets;
 save exp2 sub_data means_dat results;
run;
```

תרגיל 25

א. אחת הדרכים הביסיות והקלות ביותר לבחון טעויות קידוד היא להפיק לפלט את הערכים המינימאליים והמקסימאליים של המשתנים, ובדיקה האם ערכים אלה נופלים בטווח הערכים האפשריים של משתנים אלו. יתרה מזו, באמצעות הפקת פלט של ערכי המינימום והמקסימום ניתן לאתר באמצעות הקוד ערכים חריגים בנתונים:

```
proc means min max;
 var A B C D;
run;
```

ב.

```
proc means mean std maxdec=2;
 var A B C D;
run;
```

ג.

```
proc means mean std maxdec=2;
```

```
 var A B C D;
 by cond;
run;
```

תרגיל 26

ראשית, ניצור קובץ טקסט בשם targil26 המכיל את קובץ הנתונים, ונכניס אותו לזיכרון של SAS.

```
Data targil26;
 infile 'c:\targil26.txt';
 input t1-t10;
run;
```

כעת, נשתמש ב-PROC MEANS כדי לחשב את הממוצע של כל סיבוב מעבר לנבדקים (לא לשכוח להשתמש באופציה noprint, שכן אנחנו לא מעוניינים בפלט בחלון ה -outpu):

```
proc means mean noprint;
 output out = targil26b mean = t1-t10;
run;
```

כרגע הבעיה היא שנתוני הממוצעים מופיעים בשורה אחת, וכל ממוצע של סיבוב מעבר לנבדקים מהווה משתנים. כדי להפוך משתנים אלה לעמודות, נשתמש ב-PROC TRANSPOSE. אולם, בטרם נשתמש בפרוצדורה זו, נמחק משתנים לא רלוונטיים שהוספו לנו כברירת מחדל על ידי PROC MEANS:

```
data targil26b (drop = _TYPE_ _FREQ_); set targil26b;
```

כעת, נוכל ליצור את הקובץ הסופי:

```
proc transpose data = targil26b out = targil26b name = sivuv
               prefix = prop;
run;
```

תרגיל 27

ראשית, ניצור קובץ טקסט בשם targil27, נכניס אותו לזיכרון של SAS, וניצור משתנה חדש final_grade הכולל את ממוצע כל הציונים. מאחר ולא נצטרך בהמשך להשתמש בנתוני הציונים של הבחנים והמבחן המסכם, נשמיט אותם מקובץ הנתונים בשלב זה:

```
data targil27 (drop = quiz1 quiz2 test);
 infile 'c:\targil27.txt';
 input gender quiz1 quiz2 test;
 final_grade = mean (quiz1,quiz2,test);
run;
```

כעת, נשתמש ב-PROC FREQ כדי ליצור את טבלת השכיחויות. לא נשכח בדרך להגדיר את הגבולות של הטבלה כקווים ישרים:

```
proc freq formchar (1,2,7) = '|-+';
 table gender * final_grade;
run;
```

תרגיל 28

ראשית, נגדיר פורמאט למשתנה מין באמצעות PROC FORMAT:

```
proc format;
 value sex
    0 = 'male'
    1 = 'female';
run;
```

כדי לבנות את ההיסטוגרמות ההשוואתיות, נשתמש ב-PROC UNIVARIATE, כאשר קובץ הנתונים עליו הפרוצדורה תעבוד הוא targil27 שיצרנו בתרגיל הקודם. לא נשכח לקשר בין הפורמאט שייצרנו למשתנה gender, ולא נשכח להגדיר את כל האופציות של ההיסטוגרמה:

```
proc univariate noprint;
 class gender;
 histogram final_grade/cfill = blue cbarline = yellow wbarline = 3
                       normal (color = red w = 5);
 inset MEAN std;
 format gender sex.;
run;
```

תרגיל 29

ראשית, ניצור את הקובץ targil29, המכיל את קובץ הנתונים ונכניס אותו לזיכרון של SAS:

```
data targil29;
 infile 'c:\targil29.txt';
 input hour anx grade;
run;
```

א. כדי לבדוק את הקשר בין המשתנים, נשתמש ב PROC CORR:

```
proc corr;
 var hour grade;
run;
```

תוצאות הפרוצדורה מלמדות על קשר חיובי חזק יחסית בין המשתנים (0.52). עוד עולה מהתוצאות כי קשר זה מובהק ($p = 0.009$). לכן, נוכל להסיק כי ככל שמשקיעים יותר שעות בלימודים, ציון המבחן עולה.

ב. כדי לבדוק אם המרצה צודק, נחשב מתאם חלקי בין המשתנים באמצעות ההוראה PARTIAL של PROC CORR:

```
proc corr;
 var hour grade;
 partial anx;
run;
```

תוצאות הפרוצדורה מלמדות כי כאשר מחזיקים את המשתנה חרדה קבוע הקשר בין המשתנים יורד (ועכשיו הוא 0.38). עוד נמצא כי רמת המובהקות של הקשר שולית (0.075). לכן, ניתן לומר שיש אמת בטענת המרצה, שכן הקשר בין כמות השעות לציון תלוי ברמת החרדה.

תרגיל 30

כדי לבדוק את טענתו של חוקר ב', נחשב את מקדם המתאם אלפא של קורנבאך לתוצאות השאלון. ראשית, ניצור קובץ טקסט בשם targil30, המכיל את הנתונים של השאלון, ונכניס אותו לזיכרון של SAS:

```
data targil30;
 infile 'c:\targil30.txt';
 input q1-q8;
run;
```

כעת, נשתמש ב-PROC CORR כדי לחשב את מקדם המתאם אלפא של קורנבאך:

```
proc corr data = targil30 alpha;
 var q1-q8;
run;
```

תוצאות הניתוח מלמדות על מקדם מתאם של $\alpha = 0.7$. מאחר וכלל האצבע אומר ש-0.7 הוא הגבול התחתון להנחת עקיבות פנימית, ניתן לומר כי חוקר ב' טועה, במיוחד אם נתייחס לכך שגודל המדגם קטן יחסית.

תרגיל 31

כדי ליצור קוד זה, נשתמש בהוראה WITH של PROC CORR:

```
proc corr;
 var  Max Avg P Recency;
 with Ant Mid Pos;
run;
```

תרגיל 32

ראשית, נכתוב את הקוד לחישוב מטריצת הקורלציות בין המשתנים השונים. הקוד למטריצת הקורלציות מניח כי יצרתם קובץ נתונים בשם targil32a, המכיל את הנתונים הגולמיים של התשובות של 28 הנבדקים לכל אחת מ-10 השאלות, כאשר המשתנים של קובץ הנתונים מכונים q1 עד q10 (שאלה 1 עד שאלה 10). כדי לחשב את מטריצת הקורלציות, נשתמש ב-PROC CORR:

```
proc corr;
 var q1-q10;
 with q1-q10;
run;
```

כעת, נוכל ליצור קובץ נתונים מסוג מתאמים, ועליו נבצע ניתוח גורמים באמצעות PROC FACTOR. לא נשכח לקבוע ב-PROC FACTOR את הרוטציה ל-varimax, ולהגדיר הפקה של scree plot:

```
data targil32b (type = corr);
_TYPE_ = 'CORR';input _var_ $ q1-q10;
cards;
q1  1 . . . . . . . . .
q2  0.28936 1 . . . . . . . .
q3  0.48376 0.55172 1 . . . . . . .
q4  0.28795 0.16204 -0.20435 1 . . . . . .
q5  0.25898 0.06152 0.00866 0.25332 1 . . . . .
q6 -0.14678 -0.10573 -0.05595 -0.01873 -0.39573 1 . . . .
q7 -0.21904 0.1769 0.01391 -0.06056 0.27815 0.18653 1 . . .
q8 -0.16504 0.0982 0.0376 -0.00504 -0.1064 0.42571 0.55724 1 . .
```

```
q9  0.22003 0.30011 0.12758 0.18072 0.31086 -0.02924 0.36355 0.32228 1 .
q10 0.00746 0.13706 -0.12905 0.18593 -0.214 0.431 0.481 0.44478 0.29018 1
;

proc factor data = targil32 scree rotate = varimax;
run;
```

תרגיל 33

כדי לענות על שאלה זו, נבצע מבחן t למבחן יחיד, ובו נשווה את אחוז הבחירה הממוצע ל-50%. כדי לעשות זאת, נשתמש ב-PROC TTEST:

```
proc ttest h0 = 0.5;
 var choice;
run;
```

מתוצאות הניתוח, עולה כי למרות שבממוצע, הנבדקים בחרו להשקיע במנייה ב-61% אחוז מהמקרים, אחוז זה לא שונה במובהק מ-50% ($p = 0.1$). לכן, לא ניתן לקבוע כי לנבדקים הייתה העדפה להשקיע במנייה.

תרגיל 34

ראשית, ניצור קובץ טקסט בשם targil34, המכיל את ציוני הכיתות השונות:

```
data targil34;
 infile 'c:\targil34.txt';
 input grade_cita1 grade_cita2;
run;
```

כעת, נבחן את ההבדל בממוצעי הציונים בין שתי הכיתות באמצעות PROC TTEST. עם זאת, מדובר כאן על מבחן t למדגמים בלתי תלויים. במצב כזה PROC TTEST לא מאפשרת בחינה של הבדלי ממוצעים של שני משתנים שונים. לכן, נצטרך קודם ליצור קובץ נתונים חדש המכיל שני משתנים: המשתנה הראשון יהיה משתנה CLASS (בשם condition) שיקבל את הערך 0 עבור התצפיות של כיתה 1 ואת הערך 1 עבור התצפיות של כיתה 2. המשתנה השני יכיל את ציוני הסיום (בשם grade):

```
data targil34 (drop = grade_cita1 grade_cita2); set targil34;
 condition = 0;
 grade = grade_cita1;
 output;
 condition = 1;
 grade = grade_cita2;
output;
```

שים לב כי השמטנו מהקובץ את הנתונים המקוריים, שכן הם כבר לא רלוונטיים.

כעת, נוכל להריץ את הקוד לבחינת ההבדלים בממוצעי שתי הכיתות:

```
proc ttest data = targil34;
 class condition;
 var grade;
run;
```

מאחר והמבחן לשוויון בין שונויות יוצא לא מובהק ($t(12) = 1.94, p = 0.27$), נוכל להניח שוויון בין שונויות. לכן, כדי לבדוק את תוצאות המבחן נסתכל על השורה הראשונה של קטע הפלט הכולל מידע על המבחן t. תוצאות המבחן מלמדות כי

ממוצע ציוני הסיום של כיתה 1 הוא 74.5, בעוד שממוצע ציוני הסיום של כיתה 2 הוא 84 (ההפרש בין הממוצעים הוא 9.5-).
על פי תוצאות המבחן t, הפרש זה מובהק ($t(24) = -2.10, p < 0.05$). לכן, ניתן לומר כי באופן מובהק הציונים המתקבלים מלימוד בשיטה בה למדה כיתה 2 גבוהים יותר מהציונים המתקבלים בשיטה בה למדה כיתה 1. אי לכך, נוכל להמליץ לדיקן להשתמש בשיטה שנלמדה בכיתה 2.

תרגיל 35

ראשית, ניצור קובץ txt המכיל את הנתונים של הסטודנטים. נקרא לקובץ הנתונים data, ונמקם אותו בכונן C. כעת, נוכל לקרוא את קובץ הנתונים לתוך SAS:

```
data targil35;
 infile 'c:\data.txt' dlm='09'x;
 input gender age maslul toar memutza psycho tzion locus;
```

הערה: הקוד הנוכחי מניח שהמשתנים בקובץ מופרדים על ידי טאבים. באם הקובץ שיצרת מופרד בדרך אחרת (רווחים, פסיקים וכדומה), יש לשנות את הקוד בהתאם.

א. כעת, לאחר שהכנסנו את הנתונים ל-SAS, נוכל לבדוק את הסעיף הראשון של השאלה. אולם, בטרם נעשה זאת, יש לעדכן את קובץ הנתונים. ספציפית, לפני שנבצע את הניתוח הסטטיסטי, יש להפוך את המשתנה "תואר הלימוד" למשתנה בינארי (1 או 0). הסיבה לכך היא העובדה שתואר לימוד הוא משתנה שמי. כדי להפוך את המשתנה "תואר הלימוד" למשתנה בינארי, נכתוב את הקוד הבא:

```
data targil35; set targil35;
  if toar = 1 then toar = 0;
   else if toar = 2 then toar = 1;
 output;
run;
```

כעת, תואר ראשון מיוצג כ-0 ותואר שני מיוצג כ-1 (כמובן, היה ניתן לקודד את המשתנה להפך).

עכשיו, נוכל להשתמש ב-PROC REG כדי לבחון האם ניתן לנבא את הציון הממוצע על סמך תואר הלימוד:

```
proc reg;
 model memutza = toar;
run;
```

מאחר ופלט התוכנית מלמד אותנו כי מודל הניבוי שבחנו אינו מובהק ($p = 0.1$), ניתן וומר כי לא ניתן לנבא את הציון הממוצע על סמך תואר הלימוד.

ב. כדי לענות על סעיף זה, נוסיף למודל הניבוי שלנו את המשתנה tzion כמשתנה מנבא נוסף:

```
proc reg;
 model memutza = toar tzion;
run;
```

כעת, נראה כי מודל הניבוי שלנו מובהק ($p < 0.0004$). לכן, ניתן לומר כי הוספת השתנה tzion אכן תשפר את יכולת הניבוי של המודל שלנו.

ג. כדי לבחון אינטראקציה בין שני המשתנים, ניצור משתנה חדש בשם inter, שיהווה את משתנה האינטראקציה שלנו:

```
inter = toar*gender;
```

את שורת קוד זו יש להוסיף ל-DATA STEP הראשון שיצרנו (בו הגדרנו את קובץ הנתונים שלנו), ממש מתחת להוראה INPUT (שמגדירה את כל המשתנים שלנו).

כעת, נוכל לבדוק מודל ניבוי הכולל גם את משתנה האינטרקציה:

```
proc reg;
 model memutza = gender toar inter;
run;
```

מתוצאות הניתוח, נראה כי המודל אינו מובהק ($p = 0.159$). יתרה מזו, נראה גם כי משתנה האינטרקאציה שלנו אינו מובהק ($p = 0.3264$). לכן, ניתן לומר כי אין אינטראקציה בין מין הסטודנט לתואר הלימוד, דהיינו לתואר הלימוד אין השפעה שונה לגברים ולנשים.

ד. כדי לבחון את כל המשתנים, נשתמש במודל הרגרסיה שלנו באופציה selection, המאפשרת לנו לבצע ניתוח רגרסיה בצעדים:

```
proc reg;
 model memutza = gender age maslul toar psycho tzion
                 locus/selection = stepwise;
run;
```

מתוצאות הניתוח עולה כי המודל הטוב יותר מכיל 5 משתנים (gender, age, toar, psycho, tzion). מודל זה מסביר כ-44% מהשונות בציון הממוצע.

תרגיל 36

כדי לבחון את ההשערה האם יש או אין הבדל בין ממוצע של אותה קבוצת נבדקים מעבר לשיטות טיפול (או אימון) שונות, יש להשתמש בניתוח שונות (ANOVA). לכן, הדרך הטובה ביותר לענות על התרגיל הנוכחי היא להשתמש ב- PROC ANOVA. עם זאת, ב- PROC ANOVA יש להגדיר משתני CLASS (משתנים המגדירים את הקבוצות השונות). לכן, יש לשנות את קובץ הנתונים הקיים שיתאים לפורמט הנדרש על ידי PROC ANOVA:

```
data targil36;
sub+1;
 do training = 1 to 4;
   input RT @@;
   output;
end;
cards;
8.86 5.75 5.22 3.83
9.36 5.71 3.46 4.80
15.76 10.86 5.82 5.73
11.43 9.27 8.23 7.14
10.99 11.04 10.84 8.35
13.44 7.70 8.73 8.76
19.08 21.38 18.35 12.02
12.28 14.51 15.52 9.27
16.93 11.63 12.41 8.53
18.05 15.83 11.06 6.73
18.47 19.02 16.16 7.33
13.20 17.82 10.40 8.66
11.23 14.55 8.52 6.93
```

```
47.69 10.37 8.58 9.28
19.28 12.31 15.76 8.51
9.81 12.95 7.83 7.58
12.78 13.47 9.49 6.53
17.64 12.47 11.30 5.77
15.37 12.59 10.48 7.75
16.30 15.13 12.42 7.77
7.19 14.18 9.07 5.47
12.22 12.64 10.94 8.41
12.76 12.41 10.88 6.18
;
```

באמצעות קוד זה אנחנו בעצם יוצרים 3 משתנים: המשתנה sub, המייצג את 23 האצנים השונים (לכן, ערכי המשתנה נעים בין 1 ל-23), המשתנה training, המייצג את שיטת האימון (ולכן ערכי המשתנה נעים בין 1 ל-4), והמשתנה RT (המייצג את זמן הריצה של כל אצן בכל אחת משיטות האימון השונות). לתשומת לבך, הקובץ שנוצר מכיל בעצם 4 תצפיות שונות לכל אצן, ולכן גודל קובץ הנתונים הוא כמספר האצנים כפול 4.

כעת, נוכל להשתמש ב-PROC ANOVA כדי לבחון האם לשיטת האימון יש השפעה על זמן הריצה. בנוסף, כדי להשוות בין שיטות האימון השונות, נגדיר את ההוראה MEANS, כדי להשוות בין הממוצעים השונים:

```
proc anova;
 class sub training;
 model RT = sub training;
 means training/tukey;
run;
```

מהרצת הקוד נראה כי אכן שיטת האימון משפיעה באופן מובהק על זמן הריצה ($p < 0.0001$). בנוסף, נראה כי שיטת האימון הרביעית היא הטובה ביותר (זמן ריצה ממוצע של 7.45).

תרגיל 37

כדי לענות על שאלה זו, יש לבצע ניתוח שונות ולבחון את ההבדל בזמן התגובה בין ארבעת הקבוצות השונות. עם זאת, יש לשים לב כי מערך הניסוי הוא פקטוריאלי, וכי זמן התגובה של כל נבדק נמדד בנפרד תחת ארבעת תנאי הניסוי. לכן, אחת הדרכים לבצע ניתוח כזה היא באמצעות PROC GLM, ולהגדיר את זמני התגובה כמדדים חוזרים. אולם, ראשית יש להכניס את הנתונים ל-SAS:

```
data targil37;
input sub LD_LP LD_HP HD_LP HD_HP;
cards;
1 12.57 7.19 6.52 7.29
2 8.20 7.14 4.32 8.49
3 19.71 13.58 7.27 9.67
4 14.29 11.59 10.29 11.43
5 13.71 13.80 13.55 16.69
6 16.80 9.63 10.91 13.45
7 23.85 26.72 22.94 17.52
8 15.34 18.13 28.14 27.84
9 21.16 14.53 11.76 13.15
10 10.06 19.78 13.82 10.91
11 10.59 11.28 10.20 11.66
12 16.49 22.27 15.49 13.32
```

```
13 14.03 18.19 14.40 11.16
14 59.61 12.97 15.72 14.10
15 24.10 15.39 19.70 13.13
16 12.26 16.19 13.54 11.98
17 18.47 16.84 14.37 10.66
18 9.55 3.08 11.63 9.71
19 6.72 6.99 8.10 9.68
20 11.63 11.41 10.52 9.71
21 8.99 17.72 11.34 9.34
22 15.27 15.80 11.17 13.02
23 15.96 15.52 12.35 10.23
;
```

לאחר מכן, נכתוב את הקוד לניתוח:

```
proc glm;
 model LD_LP LD_HP HD_LP HD_HP = /nouni;
 repeated diff 2, pay 2;
run;
```

שים לב כי הגדרו את האופציה nouni מאחר ואנחנו מתעניינים באפקטים בין התנאים השונים. מתוצאות הניתוח עולה כי חוקר ב' צודק. בעוד שההפרש בין ההימורים נמצא כאפקט המשפיע באופן מובהק על זמן התגובה ($P = 0.034$), נראה כי סכום ההימורים לא ($p = 0.2182$). יתרה מזו, לא נמצאה גם שום אינטראקציה מובהקת בין שני התנאים ($p = 0.7419$).

תרגיל 38

מאחר וקובץ הנתונים לתרגיל זה גדול מידי, לא כללנו אותו בספר. עם זאת, נוכל לכתוב תוכנת סימולציה שתיצור קובץ נתונים "מלאכותי", בהתאם למה שידוע על ממצאי הניסוי. כדי לעשות זאת, נריץ שתי לולאות: לולאה אחת שרצה מ-1 עד 63 (המדמה את מספר הנבדקים), ולולאה שנייה שרצה מ-1 עד 200 (המדמה את 200 הסיבובים של הניסוי). עבור כל "נבדק", נקבע בכל "סיבוב" האם הוא בחר את האופציה הרצויה או לא. את זאת נעשה באמצעות הגדרת המשתנה "בחירה" (המשתנה t_n), המגדיר את הבחירה בכל אחד מהסיבובים) או כ-1 או כ-0, בהתאם לפרופורציית הבחירות האמיתית באופציה זו בכל אחד מהתנאים (באמצעות שלושת מבני IF…THEN שהגדרנו, המגדירים בעצם את שלושת תנאי הניסוי). בכל אחד ממבנה התנאי, אנחנו יוצרים מספר אקראי, ואם הוא גדול יותר מפרופורציות הבחירה באופציה הרצויה בהתאם לתנאי ולשלב שאנחנו מדמים, אזי הבחירה תיהיה 0, ולהפך:

```
data targil38 (drop = i j);
 array trial{200} t1-t200;
  do i = 1 to 63;
   do j = 1 to 200;
    if i < 22 then do;
        condition = 1;
       if j < 101 then do;
         trial{j} = 1;
         if ranuni(0) > 0.9354 then trial{j} = 0;
       end;
       else if j > 100 then do;
         trial{j} = 1;
         if ranuni(0) > 0.0213 then trial{j} = 0;
       end;
      end;
      else if 21 < i < 43 then do;
        condition = 2;
```

```
      if j < 101 then do;
        trial{j} = 1;
        if ranuni(0) > 0.5654 then trial{j} = 0;
      end;
      else if j > 100 then do;
        trial{j} = 1;
        if ranuni(0) > 0.0563 then trial{j} = 0;
       end;
    end;
      else if  i > 42 then do;
       condition = 3;
      if j < 101 then do;
        trial{j} = 1;
        if ranuni(0) > 0.5642 then trial{j} = 0;
      end;
      else if j > 100 then do;
        trial{j} = 1;
        if ranuni(0) > 0.0412 then trial{j} = 0;
       end;
    end;
   end;
   output;
end;
run;
```

לאחר שיצרנו את קובץ הנתונים עבור התרגיל, נוכל לבצע את הניתוח הסטטיסטי. מאחר ואנחנו מעוניינים בהבדלים בין הממוצעים (פרופורציות הבחירה באופציה הרצויה בתנאי 2 מול תנאי 3), ומאחר והגדרנו להשתמש בממוצע הבחירות ב-10 הסיבובים האחרונים של שלב הלמידה כ - covar–ate, נחשב תחילה את המשתנים הרצויים (ממוצע הבחירות באופציה הרצויה בשלב ההכדחה – המשתנה ext, וה-covariate – ממוצע הבחירות באופציה הרצויה ב-10 הסיבובים האחרונים של שלב הלמידה – המשתנה prac). שים לב כי אנו עושים זאת רק לתצפיות בהן ערכו של המשתנה "תנאי" גדול מ-1 (שכן רק ההבדל בין תנאי 2 ל-תנאי 3 מעניין אותנו):

```
data targil38b; set targil38;
 prac=mean(of t81-t100);
 ext=mean(of t102-t200);
 cond=condition;
 if cond>1;
```

כעת, נשתמש ב-PROC GLM כדי לבחון את השאלה האם יש הבדל באחוז הבחירה באופציה הרצויה בשלב ההכחדה של תנאי 2 ו-3. המודל שנגדיר הוא אחוז הבחירה באופציה הרצויה (ext) כמשתנה תלוי, ומשתנה ה-covariate (prac) ותנאי הניסוי כמשתנים מסבירים (ולא נשכח להגדיר כי המשתנה cond – תנאי הניסוי – הוא משתנה ה-CLASS שלנו – המשתנה המגדיר את קבוצות הניסוי):

```
proc glm data=targil38b;
 class cond;
 model ext=prac cond;
run;
```

הערה: מאחר ואנחנו עובדים על נתונים שהופקו באמצעות סימולציה, ולא על הממצאים האמיתיים מהניסוי, הנתונים הספציפיים, כמו גם ממצאי הניתוח הסטטיסטי, עשויים להיות שונים מהרצה להרצה.

תרגיל 39

כדי לעזור לפרופסור לפתור את הויכוח, נבדוק האם יש הבדל בין ממוצע העלות של קניית ספרי מתמטיקה לממוצע העלות של קניית ספרי מתמטיקה. מאחר ואין לנו מספיק נתונים להשתמש במודל לינארי, נשתמש ב PROC NPAR1WAY.
ראשית, נכניס את הנתונים ל-SAS:

```
data targil39;
input student math physics;
cards;
1 205 250
2 450 240
3 300 520
4 279 725
5 470 380
6 90 370
7 340 150
8 220 620
;
```

כעת, מאחר ו-PROC NPAR1WAY דורשת הגדרה של משתנה class, נשנה את מבנה קובץ הנתונים, שיתאים לניתוח הרצוי:

```
data targil39 (drop = student math physics); set targil39;
 price = math;
 cond = 1;
 output;
 price = physics;
 cond = 2;
 output;
proc sort; by cond;
```

עכשיו נוכל להשתמש ב-PROC NPAR1WAY כדי לבצע את הניתוח הרצוי (באמצעות מבחן wilcoxon):

```
proc npar1way wilcoxon;
 class cond;
 var price;
 exact wilcoxon;
run;
```

מאחר ותוצאת הניתוח לא הייתה מובהקת, נראה כי הויכוח מיותר. ספציפית, על פי הנתונים לא נמצא כל הבדל בין ההוצאה הממוצעת על ספרי לימוד במחלקה למתמטיקה לבין ההוצאה הממוצעת על ספרי לימוד במחלקה לפיסיקה.

תרגיל 40

ראשית, נכתוב את הקוד להרצת הסימולציה של הטלת הקוביה. כדי לעשות זאת נריץ לולאה מ-1 עד 10000 (המייצגת את 10000 הטלות הקוביה), ובכל פעם נדגום מספר אקראי מ-1 עד 6 (ונשמור את התוצאה במשתנה outcome, באמצעות ההוראה OUTPUT), שייצג את התוצאה של כל אחת מהטלות הקוביה:

```
data targil40(keep=outcome);
 do roll = 1 TO 10000;
  outcome = 1+int(6*ranuni(0));
  output;
 end;
```

כעת, נוכל לבחון את התפלגות התוצאות באמצעות PROC FREQ, וכן לבחון באמצעות מבחן חי בריבוע האם יש הבדל בין התפלגויות התוצאות. כדי לוודא שהקוביה ש"יצרנו" אכן הייתה הוגנת, נצפה למצוא חוסר הבדל בין ההתפלגויות של התוצאות השונות:

```
proc freq data=targil40;
 table outcome/chisq;
run;
```

תרגיל 41

כדי לענות על השאלה, יש לבצע מבחן חי בריבוע לאי תלות (הבודק למשל, האם יש תלות בין מין הסטודנט לפקולטה). אולם, לפני שנוכל לבצע את המבחן, יש ליצור את קובץ הנתונים ב-SAS.

דרך אחת ליצור את קובץ הנתונים תיהיה ליצור קובץ נתונים עם שני משתנים: מין הסטודנט והפקולטה אליה נרשם. אולם, דרך מהירה וחסכונית יותר תיהיה ליצור קובץ נתונים הכולל את הפרופורציות של גברים ונשים שנרשמו לכל אחת משתי הפקולטות. את זאת נוכל לעשות באמצעות הקוד הבא:

```
data targil41;
 do Faculty = 1 to 2;
  do Gender = 1 to 2;
    input prop @@;
    output;
   end;
 end;
cards;
60 40
40 60
;
```

באמצעות קוד זה אנחנו יוצרים בעצם את הערכים השונים של המשתנים מין ופקולטה (2 ערכים לכל משתנה), ומגדירים את השכיחות של כל אחת מהתצפיות (באמצעות המשתנה prop).

כעת, שיש בידינו את קובץ הנתונים הרלוונטי, נוכל לבצע את המבחן הסטטיסטי הרצוי, תוך שימוש ב-PROC FREQ. לא לשכוח להגדיר את המשתנה prop באמצעות ההוראה WEIGHT, כדי להגדיר ל-PROC FREQ את השכיחות של כל תצפית:

```
proc freq;
 weight prop;
 table Faculty*Gender/chisq;
run;
```

מאחר ותוצאת המבחן הייתה מובהקת ($p < 0.005$), ניתן להניח כי אכן קיימת תלות בין המשתנים. לכן, ניתן לומר כי בשנה הנוכחית, אכן נשים מעדיפות את פקולטה א' וגברים אכן מעדיפים את פקולטה ב'.

תרגיל 42

ראשית, נכניס את הנתונים ל-SAS:

```
data targil42;
input upper lower mobile;
cards;
78.6 37.3 1.01
67.5 48.4 1.01
9.8 35.2 1.01
```

```
43.0 45.0 1.01
67.2 51.8 0.39
80.0 43.2 0.31
17.5 22.8 0.30
9.5 12.1 0.10
83.9 12.0 0.07
13.3 14.8 0.05
0 0 0.05
0 16.7 0.02
13.8 3701
0 0.3 0.01
;
```

כעת, נוכל להשתמש בהוראה BUBBLE של PROC GPLOT כדי ליצור תרשים המאפשר לנו להציג משתנה אחד כפונקציה של שני משתנים אחרים. בתרשים זה, אחוז הפגיעה בגפיים מיוצג על ראשי הצירים (x ו-y) של התרשים, בעוד שגודל הבועה מייצג את יכולת הניידות (המשתנה mobile):

```
proc gplot;
 bubble lower * upper = mobile;
run;
```

תרגיל 43

ראשית, נכניס את הנתונים ל - SAS

```
data targil43;
input month prof1 prof2 prof3;
cards;
1 10258 9000 8000
2 9100 8500 9500
3 13000 7900 7200
4 8000 11000 10000
5 9500 9980 9500
6 7900 7900 8790
7 11000 9600 8600
8 8923 12000 6000
9 7500 9999 7890
10 9870 8560 9870
11 9000 7960 6999
12 8523 8700 8000
;
```

א. כדי להציג את הנתונים בגרף, נשתמש ב-PROC PLOT כדי להציג את הנתונים על הגרף, ונגדיר את האופציה vref כדי להוסיף לגרף את קו הייחוס:

```
proc plot;
 plot prof1 * month = '+' / vref = 10000;
run;
```

ב. כדי להציג את הנתונים של שלושת הפרופסורים על אותה מערכת צירים, נשתמש באופציה overlay של PROC PLOT. לא נשכח להגדיר לכל פרופסור סימן מזהה שונה, כדי שנוכל להבחין בתרשים בנתונים של שלושת הפרופסורים השונים:

```
proc plot;
 plot prof1 * month = 'A' prof2 * month = 'B' prof3 * month = 'C' /
      overlay vref = 10000;
run;
```

תרגיל 44

ראשית, נכניס את הנתונים ל-SAS:

```
data targil44;
input Sales_man$ Brand$ Cars_Sold;
cards;
Joe  Civic 134
Joe  Civic 238
Joe  Odyssey 98
Joe  Odyssey 88
Joe  Element 200
Joe  Element 105
Joe  Accord  35
Joe  Accord  128
John Civic   239
John Civic   201
John Odyssey 204
John Odyssey 197
John Element 187
John Element 200
John Accord  64
John Accord  152
Ben  Civic   155
Ben  Civic   219
Ben  Odyssey 163
Ben  Odyssey 155
Ben  Element 89
Ben  Element 287
Ben  Accord  143
Ben  Accord  133
;
```

כעת, כדי להציג את היקף המכירות לפי הדגם נשתמש בהוראה VBAR של PROC CHART. בנוסף, כדי להציג במקביל גם את התרומה היחסית של כל איש מכירות להיקף מכירות זה, נשתמש באופציה subgroup של ההוראה, וכן נגדיר את המשתנה Cars_sold כמשתנה אותו אנחנו רוצים להציג (על ידי ההוראה sumvar):

```
proc chart;
 vbar Brand / subgroup=Sales_man sumvar=Cars_Sold;
run;
```

תרגיל 45

ראשית נכניס את הנתונים ל-SAS:

```
data targil45;
input grades$ @@;
cards;
A B C B B C C B D B C B D A B A F B C C D
D A B C C B C D C B C F C D B D F C A B B
;
```

א. כדי להציג את התפלגות הציונים בתרשים עוגה, נוכל להשתמש בהוראה PIE3D של PROC GPLOT:

```
proc gchart;
 pie3d grades;
run;
```

ב. כדי להציג את התפלגות הציונים כאחוזים ולא כשכיחויות, נשתמש באופציה type של ההוראה PIE3D:

```
proc gchart;
 pie3d grades/ type = percent;
run;
```

תרגיל 46

```
%MACRO data;
  data targil46;
   infile 'c:\targil46.txt';
   input Day$ exp;
  run;
%IF &SYSDAY ne Friday %THEN %DO;
   PROC PRINT data = targil46;
      where Day = "&sysday";
      TITLE  'Expenses from' &sysday;
     run;
 %END;
  %ELSE %IF &SYSDAY = Friday %THEN %DO;
      PROC MEANS MEAN MIN MAX data = targil46;
        CLASS Day;
        VAR exp;
        TITLE 'Averaged Expense per day';
      %END;
%MEND data;

%data
```