

Introduction to SAS

STAT 3505

Week 1 (January 18, 2024)

Gunes Fleming Ph.D.



Please Introduce yourself

- Your name
- Your college and program
 - When do you expect to complete your current program?
- Your work experience (if any)
 - Your software technical skills (programming, databases)
 - Have you ever used SAS before?
- Your expectations from this course (WHY ??)
- Optional
 - Anything that you would like us to know about yourself (or)
 - Fun fact about you



The Statistics Software Landscape

- SAS – Large, highly used in corporate world and university research settings, has several interfaces
- SPSS – IBM bought them recently (2009) – widely used in social sciences (and now more in business) – Quick to learn in menu mode
- JMP – a SAS product that is highly visual and menu driven
- R – A user supported programming language, free and expansive, and commonly used in academia, but a larger learning curve
- EXCEL – has some statistical functions and procedures
- WINKS – Simple, low-cost general use statistics program, with a special version for Time Series
- MINITAB – Used in a number of intro stat courses



SAS

SAS (“Statistical Analysis System”) is a software suite developed by SAS Institute for advanced analytics, multivariate analyses, business intelligence, data management, and predictive analytics.

https://www.sas.com/en_us/home.html



History

- Developed in 1976
- Many packages to satisfy the statistical analysis requirements
- Many related software and platforms, e.g. JMP, SAS studio
- This class uses SAS OnDemand or SAS university edition



Getting Acclimated with SAS Environment

Let's run some code snippets and see

Hello World Example

```
/* SAS Hello World Program */  
/*Create Hello World Data Set */  
data HelloWorld;  
    msg = "Hello, World!";  
run;  
/*Print Hello World*/  
proc print data = HelloWorld;  
run;
```



Hello World Example (Cont).



Run the code

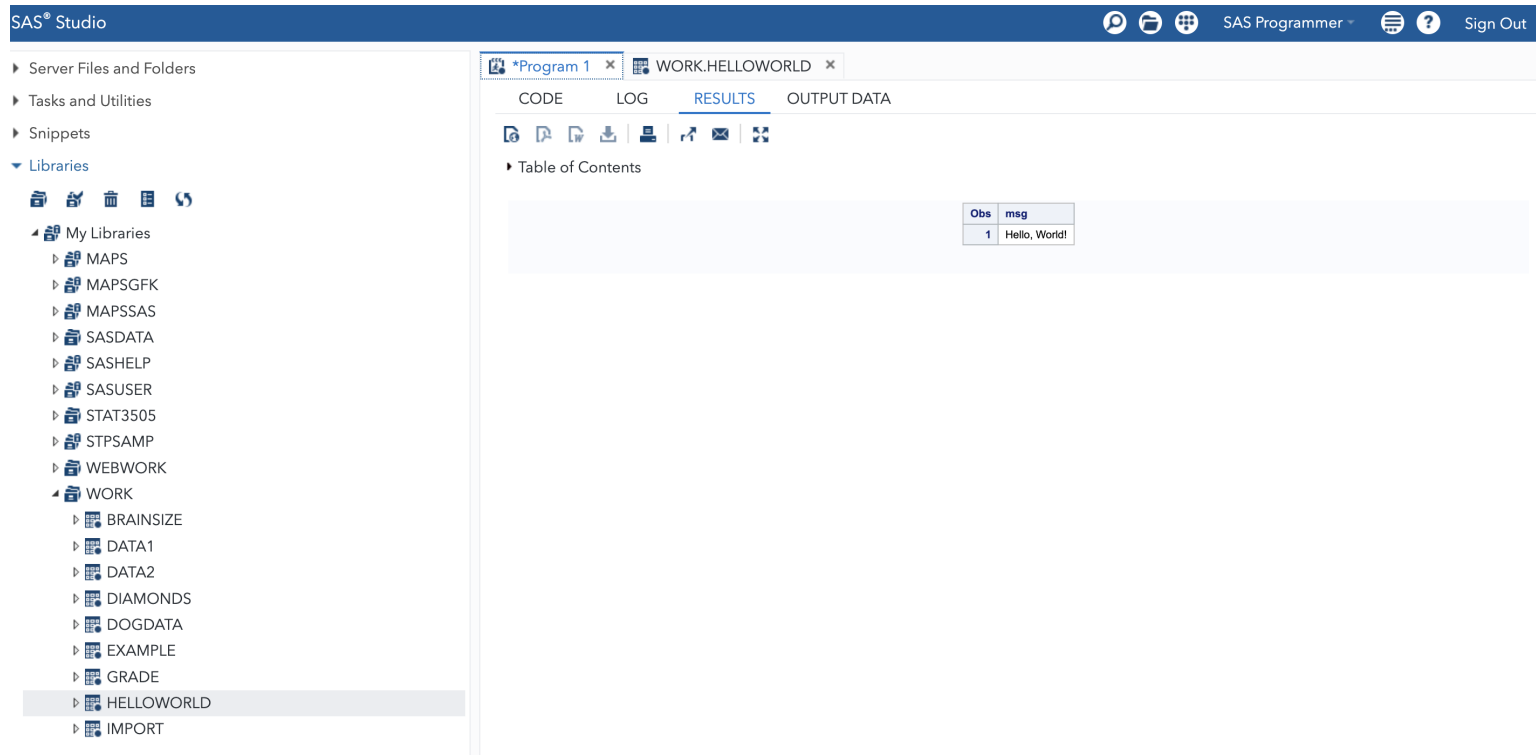


The screenshot shows the SAS Studio interface. At the top, there is a dark blue header bar with icons for location, files, and a grid, followed by the text "SAS Programmer" and a "Sign Out" link. Below the header, a tab labeled "*Program 1" is active. The main area has four tabs: "CODE", "LOG", "RESULTS", and "OUTPUT DATA". The "CODE" tab is selected, showing a SAS program. An orange arrow points to the "Run" button (a person running icon) in the toolbar. The toolbar also includes icons for undo, redo, save, print, and other editing functions, along with a "Line #" input field and execution controls like play, stop, and refresh. The SAS program code is as follows:

```
17  
20  
21 /* Example 1 */  
22 data HelloWorld;  
23     msg = "Hello, World!";  
24 run;  
25  
26 /*Print Hello World*/  
27 proc print data = HelloWorld;  
28 run;  
29
```



Hello World Example (Cont).



The screenshot displays the SAS Studio web interface. The top navigation bar includes the 'SAS® Studio' logo, user profile icons, the text 'SAS Programmer', and a 'Sign Out' link. The left sidebar contains a tree view with categories: 'Server Files and Folders', 'Tasks and Utilities', 'Snippets', and 'Libraries'. Under 'Libraries', 'My Libraries' is expanded, showing a list of libraries including MAPS, MAPSGFK, MAPSSAS, SASDATA, SASHELP, SASUSER, STAT3505, STPSAMP, WEBWORK, and WORK. The 'WORK' library is selected, and its contents are listed: BRAINSIZE, DATA1, DATA2, DIAMONDS, DOGDATA, EXAMPLE, GRADE, HELLOWORLD (highlighted), and IMPORT. The main workspace area has tabs for '*Program 1' and 'WORK.HELLOWORLD'. The 'RESULTS' tab is active, showing a 'Table of Contents' section with a table of results.


Obs	msg
1	Hello, World!

Define a Dataset and Display Data – Another Example









```
data grade;  
  input subject sex $ exam1 exam2 hwgrade $;  
  datalines;  
  10 M 80 84 A  
  7 . 85 89 A  
  4 F 90 . B  
  20 M 82 85 B  
  25 F 94 94 A  
  14 F 88 84 C  
  ;  
proc print data=grade;  
  title 'Grade Data';  
  var subject sex exam2; * print student ID, exam2 and sex;  
run;
```



Another Example (Cont.)

 *Program 1 x

CODE LOG **RESULTS**

► Table of Contents

Grade Data

Obs	subject	sex	exam2
1	10	M	84
2	7		89
3	4	F	.
4	20	M	85
5	25	F	94
6	14	F	84



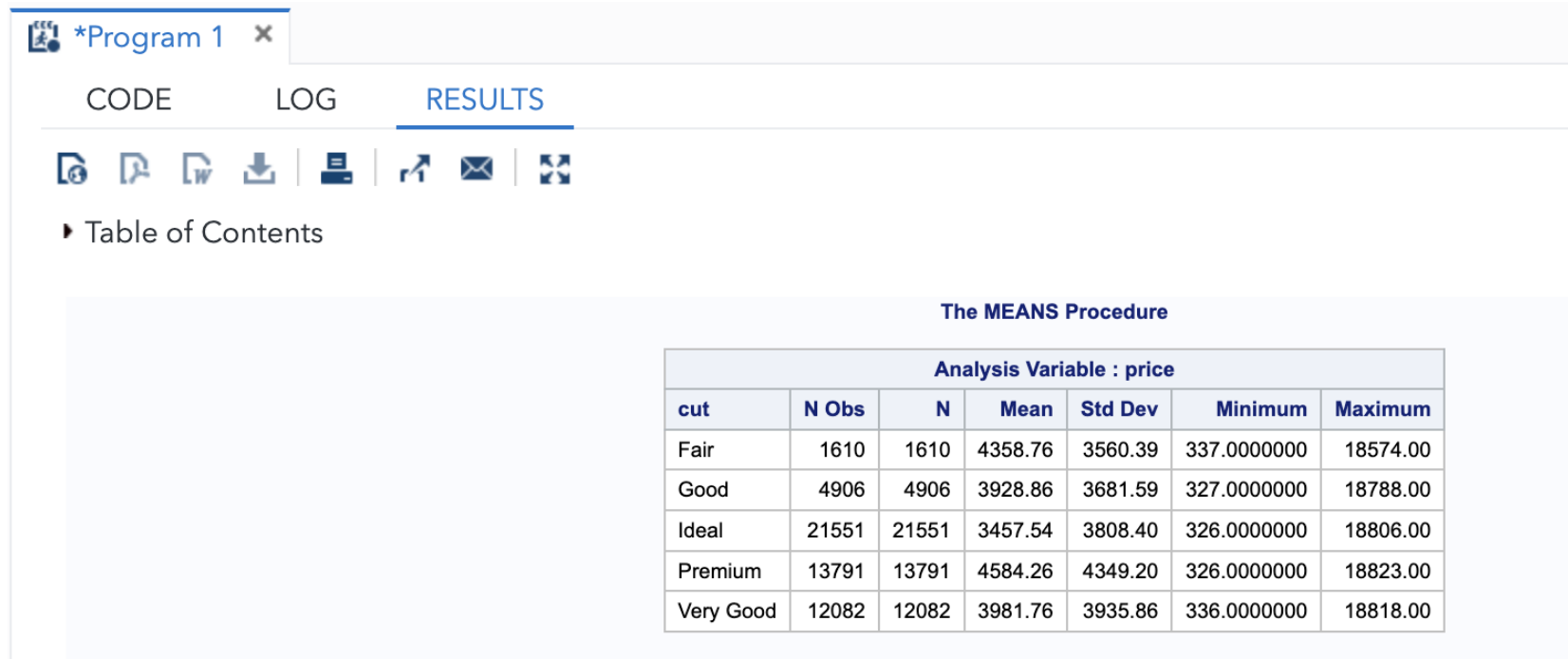
Your First SAS Analysis Example

```
/* Import data into the SAS session from our class folder (local machine) */  
proc import file = '~/my_shared_file_links/u63742093/diamonds.csv'  
  out=work.diamonds  
  dbms=csv;  
  getnames=YES;  
run;  
  
proc means data=diamonds;  
  class cut;  
  var price;  
run;
```

The second section of the code requests an analysis. In this case, the requested analysis is called “MEANS” (which indicates simple statistics) and the CLASS (classification or grouping) of the data is request by CUT and PRICE is summarized.



Your First SAS Analysis Example (Cont.)



The screenshot displays the SAS Studio interface. At the top, there is a tab labeled '*Program 1' with a close button. Below the tab are three sub-tabs: 'CODE', 'LOG', and 'RESULTS', with 'RESULTS' being the active tab. Under the 'RESULTS' tab, there is a row of icons for various actions: a magnifying glass, a document, a download, a print, a share, an email, and a refresh. Below these icons is a link labeled 'Table of Contents'. The main content area shows the output of 'The MEANS Procedure'. The title 'The MEANS Procedure' is centered above a table. The table has a header row 'Analysis Variable : price' and a second row with column names: 'cut', 'N Obs', 'N', 'Mean', 'Std Dev', 'Minimum', and 'Maximum'. The table contains five data rows for different 'cut' categories: Fair, Good, Ideal, Premium, and Very Good, each with corresponding statistics.

The MEANS Procedure

Analysis Variable : price						
cut	N Obs	N	Mean	Std Dev	Minimum	Maximum
Fair	1610	1610	4358.76	3560.39	337.0000000	18574.00
Good	4906	4906	3928.86	3681.59	327.0000000	18788.00
Ideal	21551	21551	3457.54	3808.40	326.0000000	18806.00
Premium	13791	13791	4584.26	4349.20	326.0000000	18823.00
Very Good	12082	12082	3981.76	3935.86	336.0000000	18818.00



Data Management Facility

SAS dataset: Data organized into a rectangular form with a known number of rows (observations) and columns (variables)

Name	Sex	Age	Height	Weight
Steve	M	41	74	170
Rocky	M	42	68	166
Kurt	M	39	72	167
Deborah	F	30	66	124
Jacqueline	F	33	66	115

← observation

data value

↑
variable



TIPS AND TRICKS FOR RUNNING SAS

- Within a SAS program, each statement begins with an identifying keyword (DATA, PROC, INPUT, DATALINES, RUN, etc.) and ends with a semicolon “;”. For example:
- **DATA** TEMP;
- **PROC** PRINT DATA=TEMP;
- **RUN**;



TIPS AND TRICKS FOR RUNNING SAS

- Statements can begin and end anywhere
- Statements can continue over several lines, **ends with semi-colon**
- Several statements may be on the same line
- **Blanks**, as many as you want but at least one, separating the components (words) in a SAS program statement.
- **Case**, (lower and upper) **doesn't matter** in most SAS statements.
- **Case does make a difference in data and quoted information** . (such as M or m for “MALE” or “male”).



TIPS AND TRICKS FOR RUNNING SAS

- The most common error in SAS programming is a **misplaced (or missing) semicolon** .
- A second common error is a **missing RUN;** statement.
- A third common error in a SAS program is the presence of **unbalanced quotation marks**.
- Look for errors in a program log from the top down.
- (Base SAS) If program errors cause problems that result in SAS “freezing up” or not completing the steps in your program, a way to stop SAS from continuing to run is to press CTRL-Break and to select the “Cancel Submitted Statements” option.
- If you cannot resolve a problem within SAS, save your files, exit the SAS program, and restart.
- Make the structure of your SAS programs easy to read.



Data Management Facility

To build a SAS dataset with base SAS software you use the **DATA** step:

KEYWORDS

DATA example;

INPUT Name \$ 1-10 Sex \$ 12 Age Height WEIGHT;

bmi = (weight*.045)/((Height*.0254)2);**

DATALINES;

STEVE M 41 74 170

ROCKY M 42 68 166

KURT M 39 72 167

DEBORAH F 30 66 124

JACQUELINE F 33 66 115

;

RUN;



Fox School of Business

TEMPLE UNIVERSITY®

File Extensions Referenced

- **SAS code file - (filename.sas)**
- **SAS log file - (filename.log)**
- **SAS listing file - (filename.lst)**
- **SAS data file - (filename.sasb7dat)**
- **Raw data files - (filename.dat or filename.txt or filename.csv)**
- **Excel file - (filename.xls or filename.xlsx)**



SAS Program Code (SAS OnDemand)

Enhanced Editor (Base SAS)

- **Green** - Comments appear in green.
- **Dark Blue** - Major SAS commands (also called “step-boundaries”) begin with the keyword in dark blue.
- **Blue** - Key words that have special meaning as SAS commands appear in blue.
- **Yellow highlight** - Data are highlighted in yellow. (Yellow/Brown for OnDemand)
- **Boundary Line** - A section boundary line separates each step.



Programming Language

Rules for SAS Statements:

- **SAS statement MUST end with a semi-colon (;)!!!**
- SAS statements are not case-sensitive
- SAS statements may begin anywhere on a line and can be continued on more than one line
- You can write several SAS statements on a single line
- Words in SAS statements are separated by blanks or special characters



Comparison of Softwares (SAS vs R)

SAS

Certified by SAS Inc

“Closed platform”

Not Free

More popular in Industries

R

Certified by individuals

Open platform

Free

More popular in Research



SAS Certified Exams

SAS Global Certification programs:

https://www.sas.com/en_us/certification.html



Fox School of Business
TEMPLE UNIVERSITY®

Before the start of Next Class on 01/26

- Textbook Reading
 - Required: Chapters 1 and 2
 - Recommended: Chapters 12 and 13
- That's already lot of work for the first class
 - No specific assignment / submission before next class

