Stat-342 – Term Test 2 – 2016 Spring Term

Part 1 - Multiple Choice

Enter your answers to the multiple choice questions on the provided bubble sheets. Each of the multiple choice question is worth 1 mark – there is no correction for guessing. Be sure your student name and number are completed on the bubble sheets.

1. Consider the following *SAS* code:

```
data select;
   set cereal;
   if ranuni(3333) < .5;
   if ranuni(4444) < .25 then fat = fat + 10*rannor(10);
   calories = calories + rannor(40);
run;
```

Which of the following is correct?

(a) This code adds a random number of calories (with a mean of 40) to the existing number of calories.

(b) This code selects exactly 1/3 of the cereals;

(c) This code will chose about about 50% of the cereals.

(d) This code will select those cereals with less than 40 calories/serving.

(e) This code will subset about 25% of observations based on the number of grams of fat.

Solution: (c)
Option A - 27% chose 2016. The *rannor()* function generates random normal values with a mean of 0 and a SD of 1.
Option B - 0% chose 2016. The seed (3333) has NOTHING to do with how many observations are selected.
Option C - 29% chose 2016.
Option D - 5% chose 2016. The *rannor()* function will generate random normal values, but this statement does NOT select observations.
Option E - 40% chose 2016. There is only one statement that selects observations.

2. Consider the following *SAS* code:

```
data glm;
   class shelf;
   model sugars = shelf;
   lsmeans shelf / diff cl lines adjust=tukey;
   ods output lsmeans=mylsmeans;
run;
```

Which of the following is correct?

(a) This code tests if the mean shelf is equal among sugar levels.

(b) This code estimates the mean sugar amount/serving among the different shelves.

(c) This code creates a *SAS* dataset that contains the estimated differences in mean sugars/serving among the shelves.

(d) This code compare the proportion of cereals that have high sugar content among the shelves.

(e) This code performs a two-sample t-test to compare the mean sugar amount over the three shelves.

Solution: (b)
Option A - 2% chose 2016. The model statement indicates a comparison of mean sugar levels among shelves is to be performed.
Option B - 48% chose 2016.
Option C - 48% chose 2016. The *lsmeans* table has the estimated means but not the estimated differences.
Option D - 2% chose 2016. *Proc GLM* compares MEANS and not proportions.
Option E - 2% chose 2016. A two-sample t-test is used to compare means between TWO groups, not three groups.

3. Consider the following *SAS* code that inputs temperature readings of human bodies. Normal body temperature is often taken to be 37°C.

```
data blah;
   infile datalines missover;
   length name  cbodytemp $10. typereading $16.;
   input name cbodytemp;
   whereless  = index(cbodytemp, "<");
   wheregreat = index(cbodytemp ,">");
   if whereless = 0 and wheregreat = 0 then do;
      bodytemp = input(cbodytemp, 20.0);
   end;
   if whereless > 0  then do;
```

```
      typereading = '< normal';
      bodytemp = input( substr( cbodytemp,1, whereless-1), 20.0);
   end;
   if wheregreat > 0  then do;
      typereading = '> normal';
      bodytemp = input( substr( cbodytemp,1, wheregreat-1), 20.0);
   end;
   datalines;
carl  38.4
lois 25.5<normal
matthew 39.6>fever
37.3    marianne
david 26>normal
;;;;
```

Which of the following is correct?

(a) The value of *bodytemp* for *Carl* is 38.

(b) The value of *wheregreat* for *Lois* is 0.

(c) The value of *whereless* for *Matthew* is 5.

(d) The value of *cbodytemp* for *Marianne* is "37.3".

(e) The value of *typereading* for *David* is "<normal";

Solution: (b)
Option A - 48x% chose 2016. Any decimal places override the format in the *input()* function. This is an INformat, and not an OUTformat.
Option B - 40% chose 2016. The value of *whereless* for *Lois* is 5.
Option C - 6% chose 2016. The value of *whereless* for *Matthew* is 0.
Option D - 3% chose 2016. Notice that the data values are reversed for Marianne, so that the variable *cbodytemp* will have the value "Marianne".
Option E - 2% chose 2016. The data for *David* is incorrect (26 is less than normal), but the logic would give a value of ">normal".

4. Which of the following is correct about bootstrapping.

(a) This is used to estimate the mean standard deviation in the data.

(b) This is used to estimate the standard deviation of the data values.

(c) This is used to estimate the standard error of the sample size.

(d) This is used to estimate the standard deviation of an estimator.

(e) This is used to estimate the standard error of the population mean.

Solution: (d)
Option A - 5% chose 2016 Bootstrapping says nothing about the data values.

Option B - 3% chose 2016. Bootstrapping says nothing about the data values.
Option C - 5% chose 2016. This makes no sense – the sample size is fixed.
Option D - 30% chose 2016.This is correct and is the definition of a standard error of an estimator.
Option E - 57% chose 2016. The population mean is fixed and has no standard error.

5. Consider the following *SAS* code. The original *cereal* dataset has 50 observations and 25 variables.

```
%let nboot=100;

proc surveyselect data=cereal out=bootcereal
        reps=&nboot  samprate=1 outhits method=urs;
run;
```

Which of the following is correct?

(a) The resulting dataset has 5,000 observations and 2,500 variables.

(b) The code samples WITHOUT replacement from the cereal dataset.

(c) The *outhits* option ensures that the bootstrap samples don't duplicate any records.

(d) The code creates 100 bootstrap samples.

(e) The code *samprate=1* option ensures that sampling is done with replacement.

Solution: (d)
Option A - 2% chose 2016.
Option B - 3% chose 2016.
Option C - 14% chose 2016.
Option D - 78% chose 2016.
Option E - 3% chose 2016..

6. Consider the following *SAS* code:

```
data blah;
   infile datalines missover;
   input a b c d e;
   if a > 10;
   if b < 20 then delete;
   drop a b;
   keep c d;
   datalines;
```

```
9 8 7 6 5
19 18 17 16 15
29 28 27 26 25 24
39 38 37 36 35 34
49 48 47 46 45 44
;;;;
```

Which of the following is correct?

(a) The dataset has 5 observations and 3 variables.

(b) The dataset has 4 observations and 4 variables.

(c) The dataset has 3 observations and 2 variables.

(d) The dataset has 2 observations and 4 variables.

(e) The dataset has 1 observation and 5 variables.

Solution: (c)
The subsetting *removes the first observation.* The second observation is removed by the *b<20* statement. The *keep* statement limits the variables in the dataset to *c* and *d*. Option A - 21% chose 2016.
Option B - 3% chose 2016.
Option C - 73% chose 2016.
Option D - 0% chose 2016.
Option E - 3% chose 2016.

7. Here are two data sets that are created using appropriate *SAS* commands.

```
Dataset DS1:
StudentID     Visit  Weight
012             1    150
123             1       .
456            1     190
789             1 155

Dataset DS2:
StudentID      Visit Weight
012              2  50
175              2 85
456              2 90
899              2  55
```

Consider the created dataset from :

```
data all;
   set ds1 ds2;
   by studentid;
run;
```

Which statement is correct about the *all*? dataset?

(a) The value of *weight* in the first observation has the value of 50.

(b) The value of *weight* in the third observation has the value of missing.

(c) There are 6 observations and 4 variables in the final dataset.

(d) There are 12 observations and 3 variables in the final dataset.

(e) There final dataset cannot be created because not every student is in both datasets.

Solution: (b)
Option A - 8% chose 2016.
Option B - 57% chose 2016.
Option C - 25% chose 2016.
Option D - 0% chose 2016.
Option E - 10% chose 2016..

8. Consider the following statements about generating random numbers in *SAS*. Which of the following is correct?

(a) The *seed* is used to ensure that the generated random numbers are the same across runs of the program.

(b) The *ranuni(1234)* generates a random uniform between 0 and 1234.

(c) The code *10*rannor(234)* generates a random normal random variable with a mean of 10.

(d) The code *rand("lognormal")* generates a random normal random variable with a mean of 0.

(e) The code's *seed* values sets the mean of the random number generator.

Solution: (a)
Option A - 79% chose 2016.
Option B - 6% chose 2016.
Option C -10% chose 2016.
Option D - 3% chose 2016.
Option E - 2% chose 2016.

9. Consider the following SAS code.

```
data 123data;
   infile c:\mydirectory\myfile  dlm=x  dsd skipmissing;
   length name $20 temp $ 10.
   input name temp;
   datalines;
carl x  37
```

```
Lois x 38
38   x Jane
Fred x 35
;;;;
```

Which statement is FALSE ?

(a) The code is in error because the dataset name is not a valid *SAS* name.

(b) The code is in error because the filename is not enclosed in quotes.

(c) The code is in error because the data value delimiter is not enclosed by quotes.

(d) The code is in error because the temperature (a numeric value) appears in the length statement and is declared as a character.

(e) The DATA appears to be in error because the value of name and the temperature appear to be reversed in the 3rd data value.

Solution: (d)
Option A - 6% chose 2016.
Option B - 10% chose 2016.
Option C -17% chose 2016.
Option D - 43% chose 2016.
Option E - 24% chose 2016. Notice that both *name* and *temp* are CHARACTER variables. Consequently, the 4th data value is perfectly "good".

10. Consider the following *SAS* code when applied to the data on birds collecting cigarette butts.

```
proc tabulate data=butts missing;
   class species NestContent;
   var ButtWeight NumberMites;
   table Species, NestContent*ButtWeight*sd*f=7.2;
   table Species, NestContent*NumberMites*mean*f=8.3;
```

Which of the following is correct?

(a) The first table statement will generate a table of standard deviations of the weight of cigarette butts for each species, ignoring nest content and display it with 2 decimal places.

(b) The second table statement will generate a table of the mean number of mites for each combination of nest content and species and display it with 8 decimal places.

(c) The missing option on the procedure statement will exclude data with missing values of species or nest content or butt weight or number of mites.

(d) The two table statements will make one big table showing both the standard deviation of the butt weights and the mean number of mites.

(e) The *var* statement identifies the numerical variables for which statistics will be computed.

Solution: (e)
Option A - 3% chose 2016. A table of SD for EACH combination of *species* and *nest content* will be generated.
Option B - 2% chose 2016. The **f=8.3* will print 3 decimal places but reserve a maximum of 8 characters for the display of the mean.
Option C - 8% chose 2016. The *missing* option INCLUDES observations with missing values in the classification statement.
Option D - 2% chose 2016. The two table statements will generate two tables, not one big table.
Option E - 86% chose 2016.

Part II - Long Answer

Stat-342 - 2016 Spring Term - Term Test 2

Name

Student Number:

Put your name and student number on the upper right of each of the following pages as well in case the pages get separated.

Answer the following questions in the space provided. Be sure that your answers are legible.

The marks given to these questions are 5, 8, and 7 respectively.

1. **Interpretation** - 5 Marks:

   Systolic blood pressure (mm Hg) tends to increase as people age. It is also believed that smokers tend, on average, to have higher blood pressure than non-smokers for a given age. A doctor measured the blood pressure (mm Hg), age (years) and smoking status for a sample of faculty members selected at SFU. The output from an analysis is attached to the end of this test. Write a SHORT paragraph here summarizing the results.

   Solution:
   Your answer should touch on the following points:

   - What is the objective of the experiment or survey?
   - The raw data. How many people were measured? How many smokers and non-smokers? What is the age range? Any outliers? Is there a big difference in age between smokers and non-smokers? This comes from Table 1.
   - Is there evidence of a relationship between blood pressure and age? Refer to Figure 1. The omnibus test information comes from the REG output.
   - How big is the slope? How precisely is it estimated?
   - Concerns - problems with the study?

   Here is model solution:

   > A sample of 32 patients (17 smokers; 13 non-smoker; 2 unknown status) were measured for their age (years) and blood pressure (mm Hg) to investigate if blood pressure tends to increase with age. The mean age and blood pressure of the smokers and non-smokers groups were comparable (Table 1); the subjects with an unknown smoking status tended to be younger but no formal analysis was done. Figure 1 shows a generally increasing trend in mean blood pressure with age but there is no clear separation between smokers and non-smokers. There were no obvious outliers or unusual points.
   >
   > There is strong evidence ($p < .0001$) that mean blood pressure tends to increase with age with an estimated increase of 1.6 (SE .23; 95% ci from 1.1 to 2.1) mm Hg/year. Residual and other diagnostic plots showed no evidence of problems with the fit.
   >
   > The sample was selected from the faculty at SFU. This may not be representative of the population at large for a number of reasons such as .....

   Common problems in solutions from students include:

   - Don't just give the table values as "facts" – add some interpretation to the information in the table.

- Conclusions must be about MEAN blood pressure. It makes no sense to way that older people have higher blood pressures than younger people because not all older subjects have higher blood pressure than younger subjects. However, it is sensible to say that the MEAN blood pressure tends to increase with age (that is exactly what is shown by the regression line).
- Conclusions are about POPULATION values, not sample values. So it makes no sense to say that "... there was evidence that the estimated slope was different from 0." There is NOT uncertainty about the sample values actually seen - the uncertainty lies on how well they reflect the underlying POPULATION values. Similarly, the confidence intervals are about the POPULATION slope – so it makes no sense to say ".. the 95% confidence interval for the sample slope....".
- No jargon. "Rejecting the null hypothesis" etc. have NO place in a report.
- Always couch your conclusions in terms of "evidence". So don't say "... the small p-value shows that the slope is different than zero". Rather, "There is evidence that the slope was different from zero ($p < .0001$)". Think of a trial – just because you find them "guilty" (evidence of an effect) does not imply that they actually did the crime. We are all aware of wrongful convictions. Conversely, failing to find evidence of an effect, does not imply NO effect.
- Some students said that a larger sample size was needed to make the study more representative. Sample size DOES NOT IMPLY representation. Representation is strictly controlled by randomization. Sample size only affects precision. A very large, poorly chosen sample is NOT representative of the population.
- Some students said that the p-value was very small and that the ci for the slope did not include zero. The p-value and ci will always be consistent (they are both measuring the same thing). So it is not necessary to mention both things. It is not wrong to say both statements, but it is redundant.
- Table 1 has some raw means, but no confidence intervals or *se*s so you can't make any formal conclusions about differences in age or blood pressure among smoking status.
- Some students report 6 decimal places for the slope. This is far too many (look at the se!). Rarely is more than 2 significant figures needed.
- Some students just said that the small p-value indicated that there was evidence of a relationship, but failed to give the estimated slope and to interpret the slope.
- Some students said the points that fell outside the ci are outliers. Confidence intervals say NOTHING about individual data points and so you can't make these types of statements.

- Some students were worried about points that fell outside the 95% prediction interval. By definition, about 5% of points must fall outside the prediction interval.

2. **Creating the blood pressure output** - 8 Marks:
Write *SAS* code that would generate the output for the blood pressure study. The data is available in a file called *bloodpressure.csv* and has the variable names in the first row. The variables are:

- Subject ID (numeric) up to 9 digits
- Blood pressure (numeric) up to 3 digits measured in mm of Hg.
- Age (numeric) - integers in years.
- Smk (numeric) - value of 0 = non-smoker; value of 1 = smoker; value of *missing* = unknown.

You will need to recode the *smk* variable into a new variable, *smoke_status* which takes the values *yes* or *no* or *unknown*. Don't forget to check the recoding.

Pay careful attention to the formatting of the output in terms of labels of variables and number of decimal places.

The table name in the output that is extracted is called *ParameterEstimates*.

Put your code here and overleaf if necessary.
Solution: Here is a sample code.

```
title 'Effects of age on systolic blood pressure';
data bp;
   infile 'bloodpressure.csv' dlm=',' dsd missover firstobs=2;
   input person bp quet age smk;
   length smoke_status $10.;
   if smk=1 then smoke_status = 'yes';
   if smk=0 then smoke_status = 'no';
   if smk=. then smoke_status = 'unknown';
   attrib smoke_status label='Smoking Status';
run;

proc print data=bp(obs=10) label split=' ' noobs;
   title2 'first few observations';
   var age smk smoke_status;
run;

proc tabulate data=bp missing;
  title2 'Check the recode';
  class smk smoke_status;
  table smk, smoke_status*n*f=5.0;
run;

proc tabulate data=bp;
  title2 'Table 1. Preliminary statistics';
```

```
  class smoke_status;
  var bp age;
  table smoke_status, age*n=' '*f=5.0
                      age='Age'*mean*f=7.1
                      bp='Blood Presssure'*mean*f=7.1;
run;

proc sgplot data=bp;
   title2 'Figure 1. Preliminary plot';
   scatter x=age y=bp / group=smoke_status jitter;
   reg     x=age y=bp;   /* no group= -- see comments below */
   xaxis label='Age (years)';
   yaxis label='Systolic blood pressure (mm Hg)';
run;

proc reg data=bp;
   title2 'regression analysis';
   model bp = age / clb;
   ods output parameterestimates=myest;
run;

proc print data=myest;
   title2 'table created by lsmeans';
run;

proc print data=myest noobs label split=' ';
   title2 'Table 2. Summary table';
   where variable = 'age';
   var variable estimate stderr lowercl uppercl probt;
   attrib estimate label="Estimated Slope" format=7.2;
   attrib stderr   label='SE Est Slope'       format=7.2;
   attrib lowerCL  label='LCL Slope'      format=7.2;
   attrib upperCL  label="UCL Slope"      format=7.2;
   attrib probt    label='p-value testing slope=0'  format=7.4;
run;
```

Comments about student responses:

- You could use *Proc Import* as well to read in the data using

  ```
  proc import file='bloodpressure.csv'  out=bp dbms=csv replace;
  run;
  ```

  You then need a second data step to do the recode.
- Many students forgot the *dlm*, *dsd*, *firstobs* options on the *infile* statement.

- Many students used code such as

```
data bp;
   infile ....
   length bp $10 age $10 smk $1 smoke_status $10;
   input person $ bp $ age $ smk $  smoke_status $;
   ...
```

  Both age and blood pressure are NUMERIC variables and so should NOT be declared as character in the *length* statement.
  The variable *smoking_ status* is being CREATED in the dataset and so should NOT appear on an *input* statement.

- It is very common to use *Proc Tabulate* to check recoding. We did many examples of this in the assignments and in class. Because you want to treat both *smk* and *smoking_ status* as categorical variables, both must appear in the *class statement*. Just because the *smk* variable is numeric, doesn't change the fact that it is a categorical variable.

- You also check the recode using *Proc Freq*

```
proc freq data=bp;
   table smk * smoke_status / nocol norow nopercent;
```

  As you just checking that all coding was done correctly, you don't need (nor is it sensible) to compute ANY percentages or a chi-square statistic.

- You could also compute the summary statistics using a combination of *Proc Means* and *Proc Print*.

- Notice that if you want to compute statistics on multiple variables in *Proc Tabulate* there are several possible syntaxes as shown below:

```
table smoke_status, age*'n=' '*f=5.0
                    age='Age'*mean*f=7.1
                    bp='Blood Presssure'*mean*f=7.1;
table smoke_status, age*n*f=5.0
                   (age='Age'   bp='Blood Pressure')*mean*f=7.1;
```

  but NOT

```
/* following does NOT work */
table smoke_status, age='n'*n=' '*f=5.0
                    age='Age'*bp='blood pressure*mean*f=7.1;
```

- Many students forgot to use the *jitter* option in the first plot. The *group=smoke_ status* option should NOT appear on the *reg* statement in *Proc SGplot*, otherwise you will get a separate line for smokers and non-smokers.

- For the final table, use a *where* statement to select only the slope from the table of estimates and the *var* statement to specify which variables to list. Alternatively you could have done something like:

```
data myest2;
   set myest;
   if variable ="Age";
run;
proc print data=myest2;
   ...
run;
```

  to only select the estimate of the slope and then print it in the usual fashion.

- Many students used *Proc Reg* and then also used *Proc GLM* to generate the parameter estimates. *Proc Reg* does it all.
- Many students "forgot" to pretty-up the output from *Proc Print* using labels and format to give proper column titles and the appropriate number of decimal places.

(

3. **Estimating the SE for a ratio of MEDIANs estimator** - 7 Marks:

We saw in class two common estimators for a ratio (the means of ratios and ratio of means). Similar estimators can be computed using medians. In particular,

$$\widehat{R}_{ratio\ of\ medians} = \frac{MEDIAN\ of\ numerator\ variable}{MEDIAN\ of\ denominator\ variable}$$

There is NO simple formula available for the SE of this estimator, so you will need to compute it using bootstrapping.

Consider the cereal dataset and interest lies in estimating the ratio of the number of calories to the number of grams of fat. A **.csv* file is available with variables *calories* and *fat*. Write code to do the following.

- Read the **.csv* file. The variable names are in the first row. You can assume that there are NO missing values.
- Compute the $\widehat{R}_{ratio\ of\ medians}$. BE CAREFUL! Do NOT compute the median of ratios.
- Creates 1000 bootstrap samples.
- Compute the $\widehat{R}_{ratio\ of\ medians}$ for each bootstrap sample.
- Estimates the SE of the $\widehat{R}_{ratio\ of\ medians}$ from the bootstrap estimates.
- Draws a histogram of the bootstrap sampling distribution of $\widehat{R}_{ratio\ of\ medians}$.

Put your code here and overleaf if necessary.

Solution: Sample code:

```
title 'Ratio of median estimator and SE via bootstrapping';

proc import file='cereal.csv' dbms=csv out=cereal replace;
   guessingrows = 9999;
run;

proc print data=cereal(obs=10);
   title2 'part of the raw data';
run;

/* compute the ratio of medians by first finding the median of calories and fat */
proc means data=cereal n mean std median;
   var calories fat;
   output out=medians median(calories)=med_calories median(fat)=med_fat;
run;

data medians;
```

```
   set medians;
   ratio_med = med_calories / med_fat;
run;

proc print data=medians;
   title2 'estimated ratio of medians of calories : fat';
run;


/****** now to do the bootstrap sampling */
%let nboot=1000;

proc surveyselect data=cereal out=bootcereal
       method=urs samprate=1 outhits
   reps=&nboot;
run;

proc sort data=bootcereal; by replicate; run;
proc means data=bootcereal noprint;
   by replicate;
   var calories fat;
   output out=bootmedians median(calories)=med_calories
          median(fat)=med_fat;
run;

data bootmedians;
   set bootmedians;
   ratio_med = med_calories / med_fat;
run;

proc means data=bootmedians n std;
   title2 'SE of ratio of medians of calories:fat';
   var ratio_med;
   output out=se_ratio_med    stddev(ratio_med) = se_ratio_med;
run;

proc sgplot data=bootmedians;
   title2 'Histogram of sampling distribution';
   histogram ratio_med;
run;
```

Comments about student responses:

- You could use a data step as well to read in the data.

- Notice that you need to compute the medians over the entire dataset first, before computing the ratio.
- Many students wanted to compute the SE using

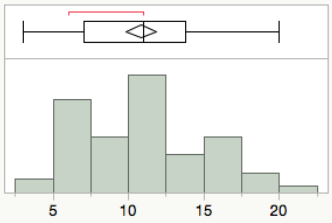
```
proc means data=bootmedians n std;
   title2 'SE of ratio of medians of calories:fat';
   var ratio_med;
   output out=se_ratio_med    STDERR(ratio_med) = se_ratio_med;
run;
```

  The definition of a SE is the STANDARD DEVIATION of the ESTIMATES over repeated samples so computing a STDERR of repeated estimates is no sensible.

Statistics about the term test:

Distributions - Stat 342 - SAS - 2016 Spring Term

Test 2 LA - out of 20



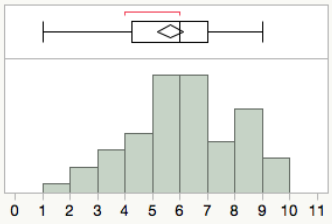
Quantiles

100.0%	maximum	20
99.5%		20
97.5%		19.375
90.0%		17
75.0%	quartile	13.75
50.0%	median	11
25.0%	quartile	7
10.0%		6
2.5%		3.625
0.5%		3
0.0%	minimum	3

Summary Statistics

Mean	10.84375
Std Dev	4.0560312
Std Err Mean	0.5070039
Upper 95% Mean	11.856916
Lower 95% Mean	9.8305836
N	64

Test 2 MC - out of 10



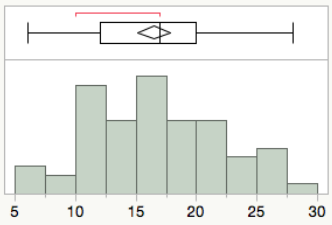
Quantiles

100.0%	maximum	9
99.5%		9
97.5%		9
90.0%		8
75.0%	quartile	7
50.0%	median	6
25.0%	quartile	4.25
10.0%		3
2.5%		1.625
0.5%		1
0.0%	minimum	1

Summary Statistics

Mean	5.65625
Std Dev	1.929049
Std Err Mean	0.2411311
Upper 95% Mean	6.1381121
Lower 95% Mean	5.1743879
N	64

Test2 Total - out of 30



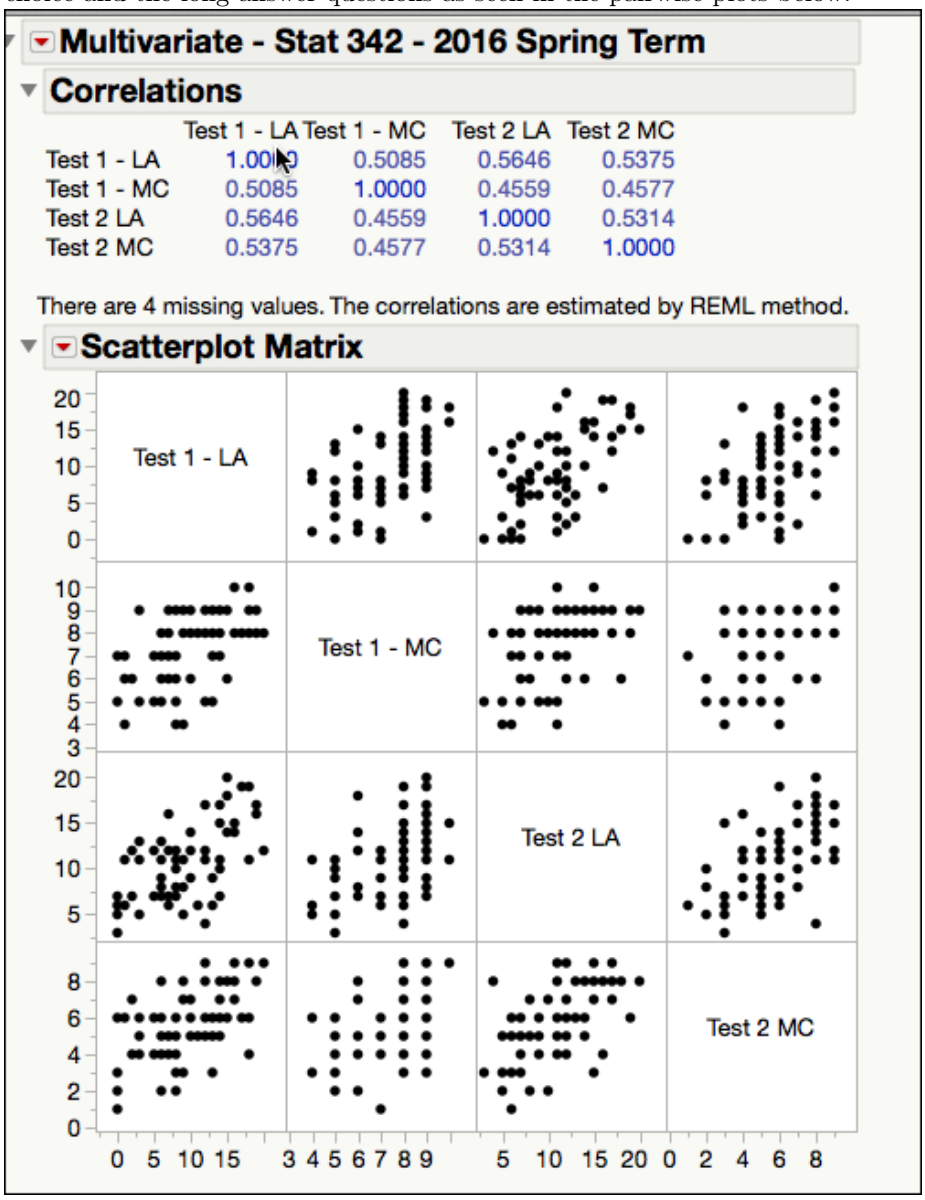
Quantiles

100.0%	maximum	28
99.5%		28
97.5%		26.75
90.0%		24.5
75.0%	quartile	20
50.0%	median	17
25.0%	quartile	12
10.0%		10
2.5%		6.625
0.5%		6
0.0%	minimum	6

Summary Statistics

Mean	16.5
Std Dev	5.3303563
Std Err Mean	0.6662945
Upper 95% Mean	17.831483
Lower 95% Mean	15.168517
N	64

There is some evidence that of relationship between grades on the multiple choice and the long-answer questions as seen in the pairwise plots below.



Effects of age on systolic blood pressure
first few observations

age	smk	Smoking Status
45	.	unknown
41	.	unknown
49	0	no
52	0	no
54	1	yes
47	1	yes
60	1	yes
48	1	yes
44	1	yes
64	1	yes

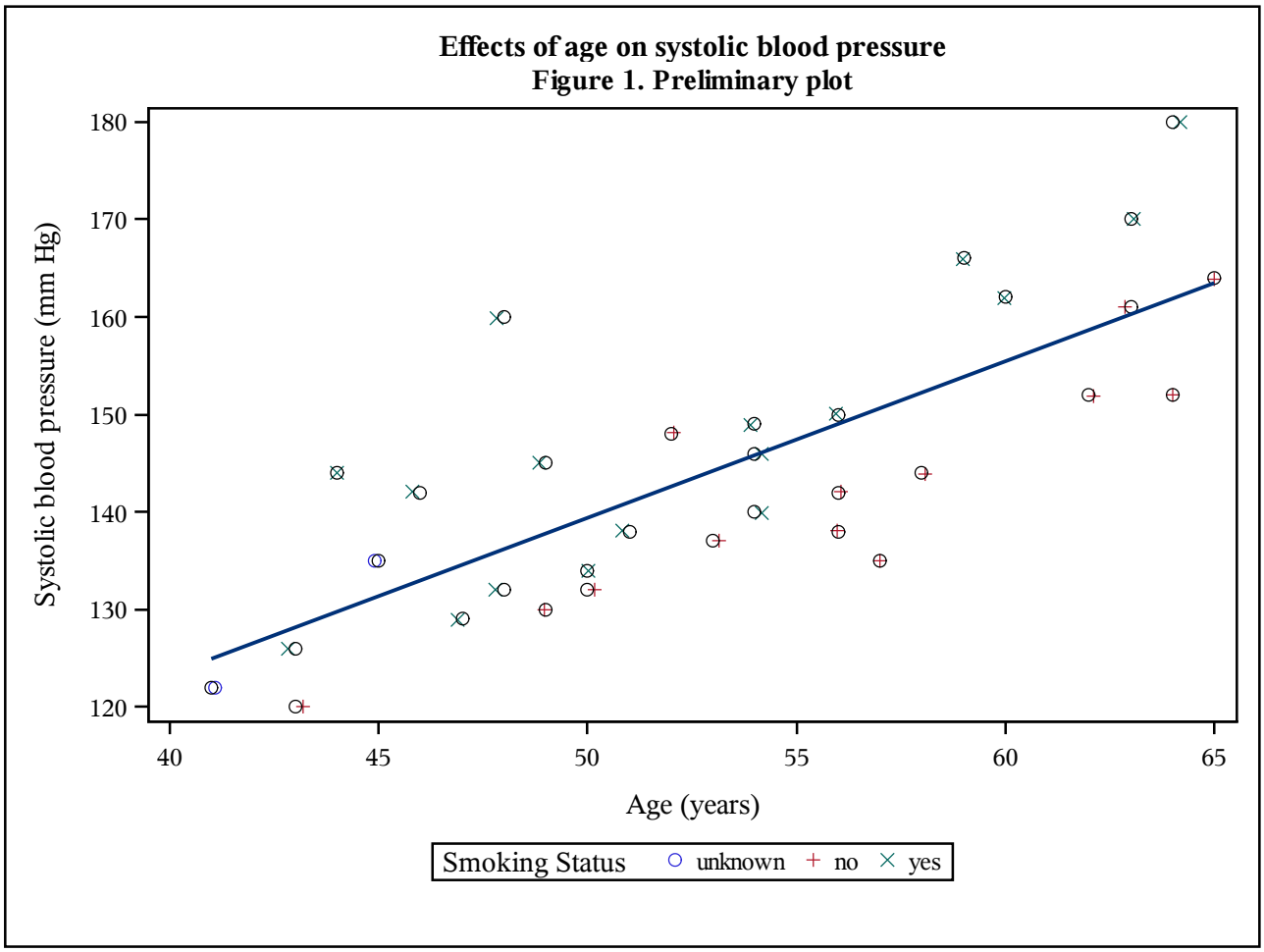
Effects of age on systolic blood pressure
Check the recode

	Smoking Status		
	no	**unknown**	**yes**
	N	**N**	**N**
smk			
.	.	2	.
0	13	.	.
1	.	.	17

Effects of age on systolic blood pressure

Table 1. Preliminary statistics

	n	Age	Blood Presssure
		Mean	Mean
Smoking Status			
no	13	56.0	142.7
unknown	2	43.0	128.5
yes	17	52.4	147.8


Effects of age on systolic blood pressure
Figure 1. Preliminary plot
Systolic blood pressure (mm Hg)
180
170
160
150
140
130
120
40
45
50
55
60
65
Age (years)
Smoking Status
unknown
no
yes

The REG Procedure
Model: MODEL1
Dependent Variable: bp

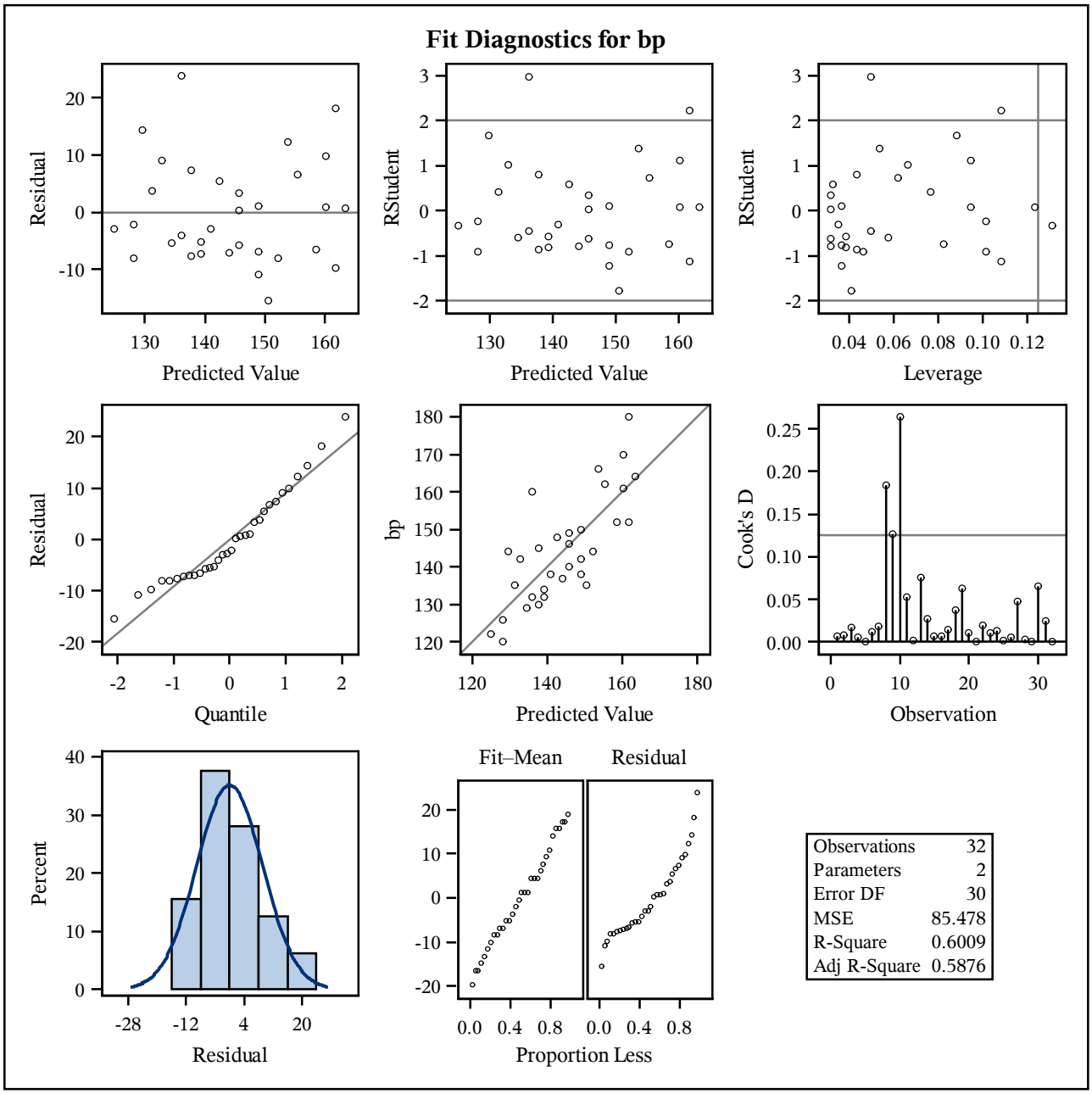
Number of Observations Read	32
Number of Observations Used	32

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	1	3861.63038	3861.63038	45.18	<.0001
Error	30	2564.33838	85.47795		
Corrected Total	31	6425.96875			

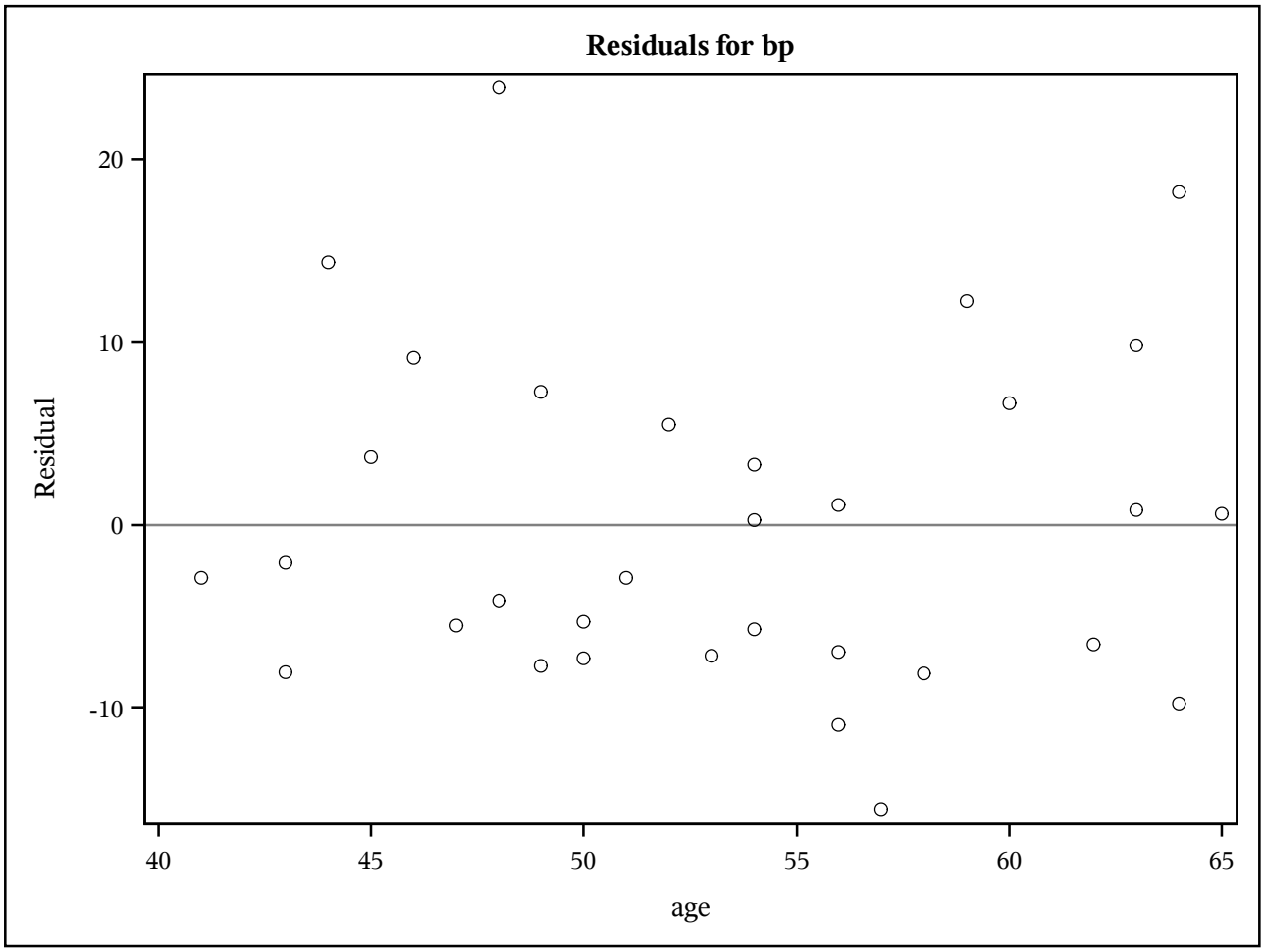
Root MSE	9.24543	**R-Square**	0.6009
Dependent Mean	144.53125	**Adj R-Sq**	0.5876
Coeff Var	6.39684		

Parameter Estimates							
Variable	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**95% Confidence Limits**	
Intercept	1	59.09163	12.81626	4.61	<.0001	32.91733	85.26592
age	1	1.60450	0.23872	6.72	<.0001	1.11698	2.09202

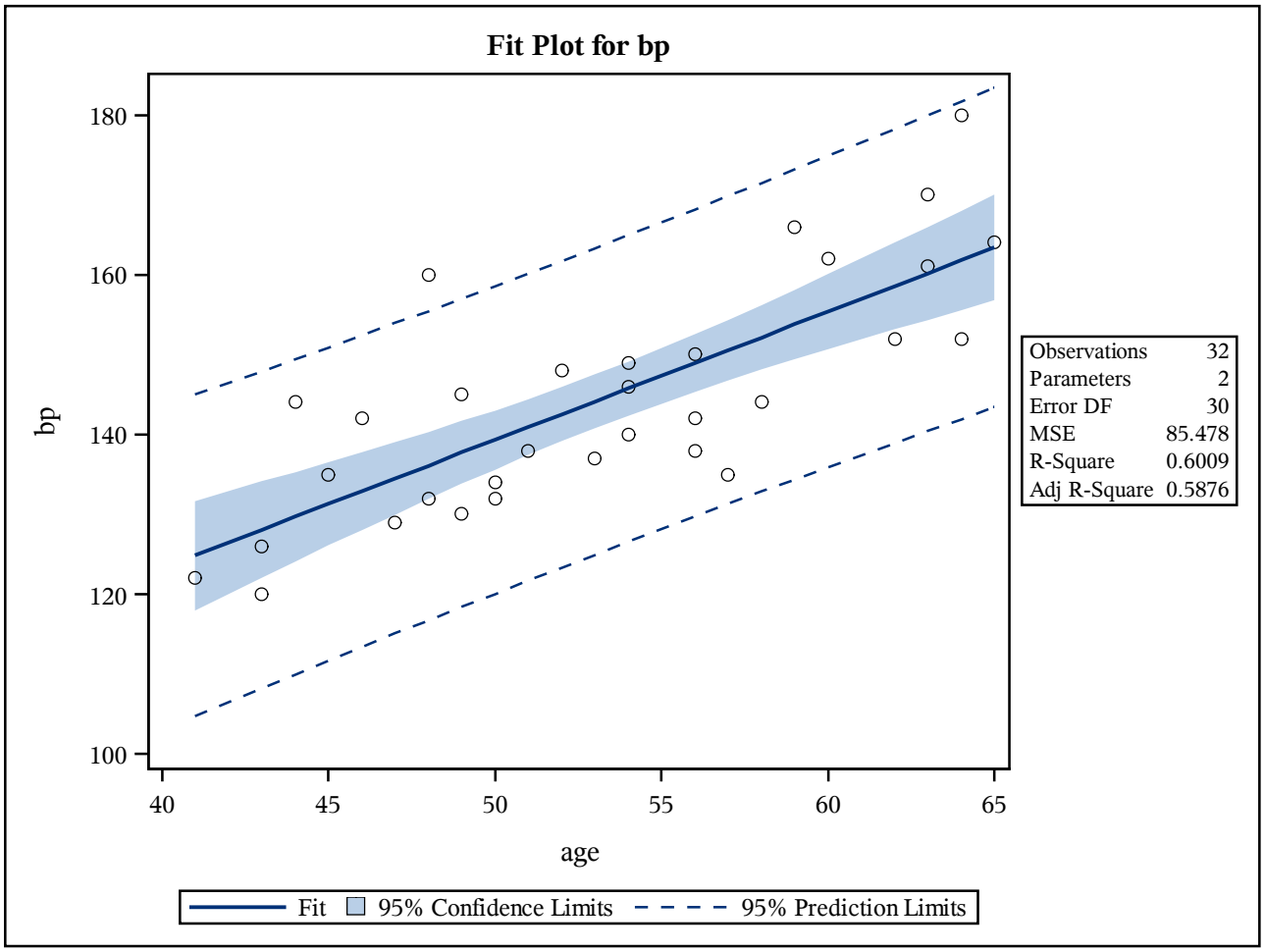
The REG Procedure
Model: MODEL1
Dependent Variable: bp



The REG Procedure
Model: MODEL1
Dependent Variable: bp



The REG Procedure
Model: MODEL1
Dependent Variable: bp



Effects of age on systolic blood pressure
table extracted from REG output

Obs	Model	Dependent	Variable	DF	Estimate	StdErr	tValue	Probt	LowerCL	UpperCL
1	MODEL1	bp	Intercept	1	59.09163	12.81626	4.61	<.0001	32.91733	85.26592
2	MODEL1	bp	age	1	1.60450	0.23872	6.72	<.0001	1.11698	2.09202

Effects of age on systolic blood pressure
Table 2. Summary table

Variable	Estimated Slope	SE Est Slope	LCL Slope	UCL Slope	p-value testing slope=0
age	1.60	0.24	1.12	2.09	0.0000

SAS Cheat Sheet for Stat-342

Carl James Schwarz

April 9, 2016

1 Reading data files into *SAS* dataset

```
DATA dsname1(dsoptions)
     dsname2(dsoptions) ....;
   INFILE  filelocation infileoptions;
   LENGTH cvar1 $length1
          cvar2 $length2 ....;
   INPUT  var1 cvar2 var3 cvar4 ....;
   ATTRIB var1 LABEL=   FORMAT=   ;
   ... processing statements ....
run;
```

1.1 Important INFILE options

- MISSOVER
- DLM= DSD
- FIRSTOBS=

1.2 Common INFORMATS

Use the : format modifier with these formats.

- `COMMAw.d` numeric values with commas
- `$` character data, but don't forget the LENGTH
- `ANYDATEw` accommodates a variety of dates but look at documentation!

1.3 Common out FORMATS

- `w.d` standard numeric
- `$n` character
- `COMMAw.d` numeric values with commas
- `DATEw.d`,`YYMMDDw.` - use the ISO standard with 4 digit years!

1.4 Format modifiers

Used typically with list input to modify the format attached to a variable.

- (colon) : typically used for numeric variables for dates/times/commas etc.
- (ampersand) & used for character values with embedded blanks

2 Importing data from database systems

```
Proc IMPORT FILE=filename
        OUT=dsname DBMS=dbms REPLACE;
   GUESSINGROWS=nnnn;
   GETNAMES=yes;
run;
```

Common dbms are `csv`.

3 Modifying existing *SAS* datasets

3.1 Subsetting observations

```
DATA dsname;
   SET dsname;
   IF condition ;
   IF condition then DELETE;
   .... statements ....
run;
```

3.2 Selecting vriables

```
DATA dsname;
   SET dsname;
   ...
   KEEP var1 var2 ...;
   DROP var1 var2 ....;
run;
```

3.3 Merging datasets

```
Proc SORT data=DS1; by bvar1 bvar2 ...; run;
Proc SORT data=DS2; by bvar1 bvar2 ...; run;
DATA both;
   MERGE ds1 ds2 ....;
   BY bvar1 bvar2 ....;
run;
```

What happens if records are in one dataset but not the other? Refer to manuals for use of IN= variables to keep track of which dataset is active in the merge.

3.4 Stacking datasets

```
Proc SORT data=DS1; by bvar1 bvar2 ...; run;
Proc SORT data=DS2; by bvar1 bvar2 ...; run;
DATA both;
   SET ds1 ds2 ....;
   BY bvar1 bvar2 ....;
run;
```

3.5 Derived variables and functions

SAS has an extensive list of function. See the help file for details. Useful functions are:

- day(date), month(date), year(date), weekday(date), etc.
- index(text, string)
- max(var1, var2, ...), min(), sum(), mean() etc.
- round(var)

- substr(test, begin, length) - differs from C
- upcase(), lowcase() - change case of text
- word(string, n)

4 Graphical Procedures

```
Proc SGPLOT data=dsnamel;
   SCATTER X=  Y= / GROUP=;
   HIGHLOW X= HIGH= LOW= ;
   XAXIS label=
     order=
     offset=(left, right);
```

Check manual for many other options.

5 Reporting Procedures

5.1 PRINT

```
Proc PRINT DATA=dsname(OBS=nnn)
     LABEL SPLIT NOOBS;
   VAR var1 var2 ...;
   ATTRIB var1 LABEL=   FORMAT= ;
   PAGEBY var;
run;
```

5.2 TABULATE

```
Proc TABULATE DATA=dsname MISSING;
   CLASS  pagevar1 rowvar2 rowvar3 colvar4 ...;
   VAR analvar ...;
   TABLE pagevar1,
       rowvar2*rowvar3,
       colvar4*analvar*(N*F=w.d MEAN*F=w.d...);BY bvar1 bvar2 ...;
run;
```

6 Analysis Procedures

6.1 FREQ

```
Proc Freq data=     ;
    table v1 * v2 / chisq nocol nopercent;
run;
```

6.2 GENMOD

```
Prog GENMOD data=...;
    class group;
    model y = group / dist=binomial
              link=logit type3;
    lsmeans group / cl diff
          adjust=tukey ilink;
    ods output lsmeans=.....;
```

See also LOGISTIC for logistic regression models;

6.3 GLIMMIX

6.4 GLM

Avoid using GLM for any linear mixed models. Use MIXED.

```
proc GLM data= ...;
   class  ...
   model y = x;
   lsmeans x / lines cl pdiff adjust=tukey;
   ods output lsmeans=....;
 run;
```

6.5 MEANS

```
Proc SORT data=dsname;
   BY bvar1 bvar2 ...; run;
Proc MEANS DATA=dsname NOPRINT;
   VAR var1 var2 ...;
   OUTPUT OUT=
      statistic(var)=name ....;
run;
```

Popular statistics are *n*, *mean*, *stddev*, *stderr*, *lclm*, *uclm*. See also SUMMARY.

6.6 MIXED

6.7 REG

```
Proc REG data=....l
   model y= x1 x2 x3 / clb;
run;
```

6.8 SURVEYSELECT

```
Proc surveyselect data=population
            out=sample
            method=
            sampsize= samprate=
            seed=
            outhits
            reps=;
 run;
```

Common methods are *srs* and *urs*. Specify the *sampsize*= or *samprate*= but not both. Many other methods and ways to specify sampling (e.g. if clustering exists) can be specified. Many other similar procedures for analysis of survey data.

6.9 TTEST

```
Proc TTEST data=....l
   class group;
   var  var1; /* independent sample */
run;
```

Always use the Welch version of the t-test. *Paired* statement used for paired t-test or use *Proc Univariate* on difference.

6.10 UNIVARIATE

```
Proc UNIVARIATE data=....  cibasic robustscale;
   var  var1 var2 ....;
   output out=   statistic(variable)=   ;
run;
```

Generate lots of output! Can also generate histograms etc, but I prefer *SGplot*.

7 Split-Apply-Combine

The `BY` statement can be used with any procedure along with `ODS tablename=dsname` to send selected output to a *SAS* dataset.

```
Proc SORT data=dsname; by ....;
Proc BLAH data=dsname;
   BY bvar1 bvar2;
   ... statements...
   ODS tablename=newds;
run;
```

8 Wide-Long and Long-Wide

```
data long;
   set wide;
   length test $10.;
   grade = t1; test="Test1'; output;
   grade = t2; test="Test2'; output;
   ....
   drop t1 t2 ....;
run;

proc transpose data=long out=wide;
   by studentnumber;
   var grade;
   id  test;
run;
```

9 Sending *SAS* output to other destinations

Refer to manual for extensive help on using `ODS`. Common usage to create PDF or MSWord document.

```
ODS PDF FILE='filename.pdf' style=.....;
   .... procedures that generate output ...
ODS PDF CLOSE;
ODS RTF FILE='filename.rtf' style=.....;
   .... procedures that generate output ...
ODS RTF CLOSE;
```

10 Generating random numbers

SAS has a complete set of function to generate pseudo-random numbers. Some examples are:

- ranuni(seed); rand('uniform'); a U[0,1] with E=.5 and SD=$\sqrt{1/12}$.
- rannor(seed); rand("normal"); a N(0,1) with E=0, and SD=1;
- rand("lognormal") with E=$e^{.5}$, and SD=$\sqrt{(e-1)e}$.

11 Bootstrapping

General procedures for CRD/SRS and statistics related to the mean.

- Resample K times with replacement from the original dataset using *Proc SurveySelect* and the *method=urs*, *sampfrac=1* , *outhits* and *reps=* options.
- Compute estimate for each bootstrap sample.
- Look at bootstrap sampling distribution to compute relevant quantities of interest.

12 Macro Variables

%LET mvar= ...; /* sets the macro variable */
Use &mvar where you want to replace the macro variable by the assigned text. Careful of quotes, i.e. "&mvar" vs '&mvar'.