STA 570 — Spring 2012 — Dr. Charnigo: Computing

Introduction. This document, available from my web page as either {SAS570S12.ps} or {SAS570S12.pdf}, will primarily focus on those aspects of SAS required to carry out the analyses described in the lectures or necessary for completion of your written assignments.

Descriptive statistics

Please refer to {SASDescriptive.txt}, a text file in which I provide the SAS code that produced the numerical and graphical output in {TABLE213.pdf}. I will describe each segment of code below. You will notice that, in this document (but not in the text file), I *italicize* certain parts of the code. These are parts of the code that you will have to modify when you want to obtain analogous output for other data sets. If you like, you can first try running the code unmodified, for practice, except that you will have to replace
C:\Documents and Settings\richc\My Documents\STA570S12
by the name of an appropriate path. Note that you can copy and paste material from a text file into SAS by highlighting the material in the text file with the mouse, pressing Ctrl-C on the keyboard, going to the Editor box in SAS, and then pressing Ctrl-V. In SAS, choosing Run and then Submit at the top of the screen will execute whatever code is in the Editor box; if you want only a portion of the code executed, highlight that portion with the mouse before choosing Run and Submit.

```
PROC IMPORT DATAFILE = 'C:\Documents and Settings\richc\
   My Documents \STA570S12\TABLE213.xls'
   OUT = Serum  DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

The segment of code above reads data from the Excel file {TABLE213.xls} into SAS. In general, you will need to change
C:\Documents and Settings\richc\My Documents\STA570S12
to the name of an appropriate path; change *TABLE213.xls* to the name of the Excel file from which you want to read data; change *Serum* to whatever you want to call the data inside SAS; and, change *Sheet1* to the name of the sheet in the Excel file containing the data. [Make sure that the Excel file is closed before you execute such code in SAS.]

```
PROC PRINT data = Serum;
RUN;
```

The segment above asks SAS to print the data just imported. Although an error message in the Log box will alert you if the attempt to read data has failed, issuing a PROC PRINT allows you to see whether SAS has changed any of the variable names. [SAS will sometimes change variable names if they contain unusual characters.]

ODS PDF FILE = '*C:\Documents and Settings\richc\ My Documents \STA570S12\TABLE213.pdf*';

The line above invokes the Output Display System, telling SAS that the numerical and graphical output produced by subsequent commands is to be saved to a PDF file in the indicated path. [If you wish to produce an RTF document that can be opened in Microsoft Word instead of a PDF document, replace all instances of "PDF"/"pdf" by "RTF"/"rtf".]

title '*Serum cholesterol changes*';
PROC UNIVARIATE data = *Serum* plots;
var *Difference*;
Histogram / cfill = ywh *midpoints = -20 to 50 by 10*;
RUN;

title '*Serum cholesterol changes*';
PROC BOXPLOT data = *Serum*;
plot *Difference**Sample /
    boxstyle = schematic
    nohlabel
    cframe = vligb
    cboxes = dagr
    cboxfill = ywh;
RUN;

The title commands tell SAS how to label the graphical output. The first segment of code produces the output on pages 1 through 3 of {TABLE213.pdf}. In general, change *Serum* to the name of the data set inside SAS; change *Difference* to the name of the variable in which you are interested; and, change *midpoints = -20 to 50 by 10* to identify your desired bin widths for the histogram (or remove this altogether to accept SAS defaults). The boxplot on page 2 of {TABLE213.pdf} is pretty revolting; the second segment of code produces the more attractive boxplot on page 4 of {TABLE213.pdf}. Since PROC BOXPLOT is designed to produce side-by-side boxplots for different groups, you have to "trick" SAS into producing a single boxplot. You can do this by defining a variable called "Sample" in the Excel spreadsheet, assigning the value "Sample" to all subjects, and then using SAS code like that above.

```
title 'Serum cholesterol changes by gender';
PROC UNIVARIATE data = Serum plots;
by Gender;
var Difference;
Histogram / cfill = ywh midpoints = -20 to 50 by 10;
RUN;
```

```
title 'Serum cholesterol changes by gender';
PROC BOXPLOT data = Serum;
plot Difference*Gender /
   boxstyle = schematic
   cframe = vligb
   cboxes = dagr
   cboxfill = ywh;
RUN;
```

The segments of code above allow us to look at the two genders separately. In general, change *Gender* to the name of the variable defining different groups that you want to look at. Note that the data must be sorted by this variable. Hence, if the first twelve subjects had been male (which begins with “m”) and the last twelve female (which begins with “f”), I could have sorted the data in Excel prior to reading the data into SAS. The results of running the segments above are found on pages 5 through 12 of {TABLE213.pdf}.

```
ODS PDF CLOSE;
RUN;
```

These two lines tell SAS that you no longer want output written to a PDF file.

Evaluating binomial probabilities

Please refer to {SASbinomial.txt}. The first piece of code evaluates $P(X = x)$, where X is a binomial random variable based on n independent trials with probability of success p. The rows below the word “cards” contain values of x, p, and n for which you want to find $P(X = x)$. Prior to any changes that you may make, the code will find $P(X = 3)$ when $p = 0.55$ and $n = 5$ as well as $P(X = 6)$ when $p = 0.20$ and $n = 15$.

```
data binomial;
input x p n;
prob = PMF('Binomial',x,p,n);
cards;
3 0.55 5
6 0.20 15
run;

proc print;
run;
```

The second piece of code evaluates $P(a < X \leq b)$, where X is a binomial random variable based on n independent trials with probability of success p. If you take $a = -1$, then you have $P(X \leq b) = F(b)$, where $F(x)$ denotes the cumulative distribution function of X. The rows below the word "cards" contain values of a, b, p, and n for which you want to find $P(a < X \leq b)$. Right now the code will find $P(2 < X \leq 3) = P(X = 3)$ when $p = 0.55$ and $n = 5$, $P(5 < X \leq 6) = P(X = 6)$ when $p = 0.20$ and $n = 15$, and $P(-1 < X \leq 6) = P(X \leq 6)$ when $p = 0.20$ and $n = 15$.

```
data binomial2;
input a b p n;
prob = CDF('Binomial',b,p,n)-CDF('Binomial',a,p,n);
cards;
2 3 0.55 5
5 6 0.20 15
-1 6 0.20 15
run;
```

```
proc print;
run;
```

Evaluating normal probabilities

Please refer to {SASnormal.txt}. The first piece of code evaluates $P(X \leq b)$, where X is a normal random variable with mean μ and standard deviation σ. The rows below the word "cards" contain values of b, μ, and σ for which you want to find $P(X \leq b)$.

```
data normal;
input b mu sigma;
prob = CDF('Normal',b,mu,sigma);
cards;
8 6 1
4 6 1
run;

proc print;
run;
```

The second piece of code evaluates $P(a < X \leq b)$, where X is a normal random variable with mean μ and standard deviation σ. The rows below the word "cards" contain values of a, b, μ, and σ for which you want to find $P(a < X \leq b)$.

```
data normal2;
input a b mu sigma;
prob = CDF('Normal',b,mu,sigma)-CDF('Normal',a,mu,sigma);
cards;
4 8 6 1
4 8 -2 8
run;
proc print;
run;
```

Finding quantiles

Please refer to {SASquantiles.txt}. The probit command returns quantiles of the standard normal distribution; for example, *probit(0.975)* supplies $z_{0.975}$. The cinv command returns quantiles of chi-square distributions; for example, *cinv(0.95,1)* supplies $\chi^2_{1,0.95}$. The tinv command returns quantiles of T distributions; for example, *tinv(0.975,80)* supplies $t_{80,0.975}$. The finv command returns quantiles of F distributions; for example, *finv(0.95,1,40)* supplies $f_{1,40,0.95}$.

```
data quantiles;
q1 = probit(0.975);
q2 = cinv(0.95,1);
q3 = tinv(0.975,80);
q4 = finv(0.95,1,40);
run;
proc print;
run;
```

One-sample problems

Please refer to {SASOneSample.txt}, which you can find (along with {TABLE612.xls} and {TABLE612.pdf}) on my STA 570 page.

First I read in {TABLE612.xls}, which provides latency times in years ("Latency") as well as an indicator of latency times less than or equal to five years ("Within5", equals 1 if latency time is less than or equal to five years, equals 0 if latency time is greater than five years).

```
PROC IMPORT DATAFILE = 'C:\Documents and Settings\richc
   \My Documents\STA570S12\TABLE612.xls'
   OUT = AIDS  DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

Next I tell SAS to start writing the results to a PDF file.

```
ODS PDF FILE = 'C:\Documents and Settings\richc
   \My Documents\STA570S12\TABLE612.pdf';
```

The following segment of code carries out inferential tasks related to μ and σ, the mean and standard deviation of the latency times in the population of which the sample in Table 6.12 is representative.

The Chapter 6 tasks include the construction of 95% confidence intervals for μ and σ, assuming normality. [The lower and upper limits of the latter confidence interval are just the square roots of the lower and upper limits of the confidence interval for σ^2.]

The Chapter 7 task is testing $H_0 : \mu = \mu_0$ against $H_1 : \mu \neq \mu_0$, again assuming normality. [The normality assumption is implicit because SAS uses a T distribution as the reference instead of the standard normal distribution, even when the sample size is large.] In this example, I have taken $\mu_0 = 5$.

To adapt this code to a new data set, replace *5* by your value of μ_0 and *0.05* by the α for which you want $100(1-\alpha)\%$ confidence intervals. Finally, change *AIDS* and *Latency* to the name of the data set and the name of the variable in which you are interested.

```
proc ttest data = AIDS  h0=5  alpha = 0.05;
var Latency;
run;
```

The following segment of code carries out inferential tasks related to p, the proportion of patients with latency times greater than five years in the population of which the sample in Table 6.12 is representative.

The Chapter 6 tasks include the construction of approximate and exact 95% confidence intervals for p. The approximate confidence interval employs the Central Limit Theorem and is adequate provided that $n\hat{p}(1-\hat{p}) \geq 5$ (≥ 10 is even better). The exact confidence interval does not appeal to the Central Limit Theorem and may be used with any sample size; there is no simple pencil-and-paper formula for this exact confidence interval, so I will not ask you to produce that on an examination.

The Chapter 7 task is testing $H_0 : p = p_0$ against $H_1 : p \neq p_0$. In this example, I have taken $p_0 = 0.40$, so the null hypothesis asserts that 40% of patients have latency times greater than five years (and 60% of patients have latency times less than or equal to five years).

To adapt this code to a new data set, change *.40* to your value of p_0 and

0.05 to the α for which you want $100(1-\alpha)\%$ confidence intervals. Finally, change *AIDS* and *Within5* to the name of the data set and the name of the variable in which you are interested.

```
proc freq data=AIDS;
tables Within5 / binomial(p=.40)  alpha=0.05 ;
run;
```

Once finished, I tell SAS to stop recording the results.

```
ODS PDF CLOSE;
RUN;
```

Two-sample problems

Please refer to {SASTwoSample.txt}, which you can find (along with {FEV.xls} and {FEV.pdf}) on my STA 570 page.

First I read in {FEV.xls}, which provides forced expiratory volumes ("FEV"), an indicator of forced expiratory volumes greater than or equal to three ("FEVge3", equals 1 if forced expiratory volume is greater than or equal to three, equals 0 if forced expiratory volume is less than three), an indicator of smoking status ("Smoke", equals 1 for smokers, equals 0 for nonsmokers), and some other variables with which we are not immediately concerned.

```
PROC IMPORT DATAFILE = 'C:\Documents and Settings\richc
   \My Documents\STA570S12\FEV.xls'
   OUT = TwoSamp  DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

Next I tell SAS to start writing the results to a PDF file.

```
ODS PDF FILE = 'C:\Documents and Settings\richc
   \My Documents\STA570S12\FEV.pdf';
```

Let μ_1 and μ_2 denote the mean FEV scores for nonsmokers and smokers, respectively. Let σ_1^2 and σ_2^2 denote the corresponding variances. We can test $H_0 : \mu_1 - \mu_2 = 0$ against $H_1 : \mu_1 - \mu_2 \neq 0$ assuming normality. [The normality assumption is implicit because SAS uses a T distribution as the reference instead of the standard normal distribution.] There are actually two variants of the test, one in which we assume $\sigma_1^2 = \sigma_2^2$ and one in which we do not. Also, we can test $H_0 : \sigma_1^2 = \sigma_2^2$ against $H_1 : \sigma_1^2 \neq \sigma_2^2$, again assuming normality. In the code below, change *smoke* to the variable defining the two groups that you want to compare and *FEV* to the variable of interest.

```
proc ttest data = TwoSamp  h0=0  alpha = 0.05;
class smoke;
var FEV;
run;
```

Let p_1 and p_2 denote the proportions of nonsmokers and smokers for whom FEV is less than 3 (i.e., for whom FEVge3 = 0). We can test $H_0 : p_1 = p_2$

against $H_1 : p_1 \neq p_2$ either with an appeal to the Central Limit Theorem or without such an appeal. The former (large-sample) test is a special case of the chi-square test for association, while the latter (small-sample) test is referred to as Fisher's exact test. We can also obtain a 95% confidence interval for $p_1 - p_2$ if we are willing to invoke the Central Limit Theorem.

```
proc freq data=TwoSamp;
   tables smoke*FEVge3 / chisq  riskdiff  alpha=0.05 ;
run;
```

Finished, I tell SAS to stop writing output to the PDF file.

```
ODS PDF CLOSE;
RUN;
```

Nonparametric methods

Please refer to {SASNonparametric.txt}, which you can find (along with {Ointment.xls}, {Acuity.xls}, and {Rabbits.xls}) on my STA 570 page.

The code below reads in the ointment data set used to illustrate the sign test and the signed-rank test.

```
PROC IMPORT DATAFILE = 'U:\STA570S12\Ointment.xls'
   OUT = Ointment   DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

To get results for the sign test and the signed-rank test, simply apply PROC UNIVARIATE to the difference measurements.

```
PROC UNIVARIATE data = Ointment;
var Difference;
RUN;
```

The code below reads in the visual acuity data set used to illustrate the rank-sum test.

```
PROC IMPORT DATAFILE = 'U:\STA570S12\Acuity.xls'
   OUT = Acuity   DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

To get results for the rank-sum test, use PROC NPAR1WAY (nonparametric one-way). Change *Dominant* to the name of the variable identifying the sample to which each subject belongs. Change *Rating* to the name of the variable containing the measurements. [Specifying wilcoxon in the first line is not really necessary, but it prevents SAS from spewing out a ton of material that you are not interested in.]

```
PROC NPAR1WAY data = Acuity  wilcoxon;
class Dominant;
var Rating;
RUN;
```

The code below reads in the rabbit anti-inflammatory data set used to illustrate the Kruskal-Wallis test.

```
PROC IMPORT DATAFILE = 'U:\STA570S12\Rabbits.xls'
   OUT = Rabbits  DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

To get results for the Kruskal-Wallis test, again use PROC NPAR1WAY. Change *Treatment* to the name of the variable identifying the sample to which each subject belongs. Change *Score* to the name of the variable containing the measurements. [Again, specifying wilcoxon in the first line is not really necessary, but it prevents SAS from spewing out a ton of material that you are not interested in.]

```
PROC NPAR1WAY data = Rabbits  wilcoxon;
class Treatment;
var Score;
RUN;
```

Analysis of variance

Please refer to {SASanova.txt}, which you can find (along with {ANOVAExamples.pdf}, {BWTsmoke.xls}, and {lead.xls}) on my STA 570 page.

First I tell SAS to write the results to a PDF file.

ODS PDF FILE = *'U:\STA570S12\ANOVAExamples.pdf'*;

Now I read in the birthweight data set that I will use to illustrate the one-way analysis of variance. This data set is described in "Obstetrics" on page 620.

PROC IMPORT DATAFILE = *'U:\STA570S12\BWTsmoke.xls'*
OUT = *BW* DBMS = EXCEL REPLACE;
SHEET = *Sheet1*;
GETNAMES = YES;
RUN;

I use the class statement to tell SAS that *Group* is not really a numeric variable but rather a "classification" variable (i.e., a variable whose purpose is to identify to which sample a subject belongs). For instance, *Group* equals 1 for women who never smoked and equals 4 for women who averaged at least one pack per day during pregnancy. The model statement tells SAS that the response variable is called *BWT* and that we want to make inferences about population means based on independent samples identified by *Group*. That is, we want to test

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$$

against its logical complement, where μ_1 through μ_4 denote the population means. The *Group 0.80 0.20 -0.40 -0.60* in the estimate statement asks SAS to test

$$H_0 : 0.80\mu_1 + 0.20\mu_2 - 0.40\mu_3 - 0.60\mu_4 = 0$$

against

$$H_1 : 0.80\mu_1 + 0.20\mu_2 - 0.40\mu_3 - 0.60\mu_4 \neq 0.$$

The *'Nonsmokers versus smokers'* ensures that the test just mentioned is appropriately labeled in the SAS output. The lsmeans statement followed by adjust=t requests pairwise comparisons (i.e., tests of $H_0 : \mu_i - \mu_j = 0$ against $H_1 : \mu_i - \mu_j \neq 0$ for $i \neq j$) with no adjustment to control the overall Type I error probability across all such comparisons. The lsmeans statement followed by adjust=bon requests pairwise comparisons with the Bonferroni adjustment.

```
proc glm data = BW;
class Group;
model BWT = Group / ss3;
estimate 'Nonsmokers versus smokers' Group 0.80 0.20 -0.40 -0.60;
lsmeans Group / pdiff=all adjust=t;
lsmeans Group / pdiff=all adjust=bon;
run;
```

Now I read in the lead exposure data set that I will use to illustrate the two-way analysis of variance. As you may have noticed, Rosner uses this data set for case studies throughout the textbook.

```
PROC IMPORT DATAFILE = 'U:\STA570S12\lead.xls'
   OUT = LEAD   DBMS = EXCEL REPLACE;
   SHEET = lead;
   GETNAMES = YES;
RUN;
```

I use the class statement to tell SAS that both *Area* and *Sex* are classification variables. The model statement tells SAS that the response variable is *Iqf*, that the two factors of interest are represented by *Area* and *Sex*, and that we are allowing for the possibility of *Area*Sex* interactions.

```
proc glm data = LEAD;
class Area  Sex;
model Iqf = Area  Sex  Area*Sex / ss3;
run;
```

Finished, I tell SAS to stop writing output to the PDF file.

```
ODS PDF CLOSE;
RUN;
```

Simple linear regression

Refer to {SASregression.txt}, which contains three segments of code. The first segment of code, printed below, shows how to read in data from an Excel file. The parts in italics are what you need to modify. Replace *'U:\STA570S12\Hypertension.xls'* by your filename and *Hyper* by whatever you want to call your data set inside SAS. In addition, replace *Sheet1* by the name of the sheet in the Excel file that contains the data. Make sure that the Excel file is closed before you execute the code.

```
PROC IMPORT DATAFILE = 'U:\STA570S12\Hypertension.xls'
   OUT = Hyper  DBMS = EXCEL REPLACE;
   SHEET = Sheet1;
   GETNAMES = YES;
RUN;
```

The second segment of code allows you to verify that you have successfully read in your data. Just replace *Hyper* by whatever you decided to call your data set inside SAS. If the variable names in Excel have characters that cannot be used in SAS variable names, SAS will change these characters to underscores; you will want to note if this happens, as you will need to refer to variables by their SAS names in what follows.

PROC PRINT DATA = *Hyper*;
RUN;

The third segment of code fits a simple linear regression model. Change *Hyper* to whatever you decided to call your data set inside SAS. Change *SBP* to the name of the response variable. Replace *AGE* by the name of the explanatory variable. Replace *0.05* with the value of α for which you desire $100(1-\alpha)\%$ confidence intervals for mean responses and $100(1-\alpha)\%$ prediction intervals. In the PLOT statement, replace *SBP* and *AGE* by the name of the response variable and the name of the explanatory variable. For your information: clm and cli request confidence intervals for mean responses and prediction intervals.

PROC REG DATA = *Hyper*;
MODEL *SBP* = *AGE* / clm cli ALPHA = *0.05*;
PLOT *SBP* * *AGE*;
RUN;

Relative risks and odds ratios

Refer to {SASoddsratio.txt}. The first piece of code is shown below. Change the numbers *33, 1667, 27, 2273* to the values a, b, c, d that you have, where a, b, c, d are as displayed in Table 13.1; my example is based on Table 13.3.

```
data Measures;
   input Exposure Response Count;
   datalines;
0 0 33
0 1 1667
1 0 27
1 1 2273
;
```

The second piece of code produces output like that shown in {Lungcancer.pdf}. All you need to do is change the title.

```
proc freq data=Measures;
   weight count;
   tables exposure*response / relrisk riskdiff;
   title 'Lung Cancer versus Heavy Drinking';
run;
```