Translation Syntax (SPSS, Stata, SAS and R)

The Basics

Calling in a data set

SPSS	**GET FILE=‘P:\QAC\qac201\Studies**\study name\filename.sav
Stata	**use “P:\QAC\qac201\Studies**\study name\filename”

SAS	**LIBNAME in “P:\QAC\QAC201\**study name;
	**DATA new; set in.**filename;

R	**>** newdata **<- read.delim(file = “**filename.txt**”, sep = “\t”, header=T)**

Selecting variables you want to examine

SPSS	**/KEEP** VAR1 VAR2 VAR3 VAR4 VAR5 VAR6 VAR7 VAR8. (Must follow the **SAVE OUTFILE**='dataname' command)
Stata	**keep** var1 var2 var3 var4 var5 var6 var7 var8
SAS	**KEEP** VAR1 VAR2 VAR3 VAR4 VAR5 VAR6 VAR7 VAR8;
R	**> var.keep <- c(“**VAR1**”, “**VAR2**”, “**VAR3**”, “**VAR4**”, “**VAR5**”, “**VAR6**”, “**VAR7**”, “**VAR8**”)** **>** title_of_new_data_set **<-** new.data**[,**var.keep**]**

Outputting your abbreviated data set

SPSS	**SAVE OUTFILE= 'P:\QAC\qac201\Studies**\study name\title_of_new_data_set’
Stata	**save** filename
SAS	**Data libname**.title_of_new_data_set; **set** *dataname*; **by** *unique_id*;
R	**> write.table**(title_of_data_set**, file=**”filename.txt”**, sep=”\t”, row.names=F)**

Sorting the data

SPSS	**SORT CASES BY** UNIQUE_ID.
Stata	**sort** unique_id
SAS	**proc sort; by** unique_id;
R	**>** title_of_data_set **<-** title_of_data_set**[order(**title_of_data_set**$**unique_id**,decreasing=F),]**

Displaying frequency tables

SPSS	**FREQUENCIES VARIABLES**=var1 var2 var3 **/ORDER=ANALYSIS.**
Stata	**tab1** var1 var2 var3
SAS	**PROC FREQ; tables** var1 var2 var3;
R	**> library(descr)** **> freq(as.ordered(**title_of_data_set**$**VAR1**))** **> freq(as.ordered(**title_of_data_set**$**VAR2**))** **> freq(as.ordered(**title_of_data_set**$**VAR3**))**

Data management

Basic Operations:

SPSS	EQ or =	>= or GE	<= or LE	> or GT	< or LT	NE
STATA	==	>=	<=	>	<	!=
SAS	EQ or =	>= or GE	<= or LE	> or GT	< or LT	NE
R	==	>=	<=	>	<	!=

***Examples*:**

1. *Need to identify missing data*

SPSS	**RECODE** var1 (9=**SYSMIS**)
Stata	**replace** var1=. **if** var1==9
SAS	**if** VAR1=9 **then** VAR1=.;
R	**>** title_of_data_set**$**VAR1**[**title_of_data_set**$**VAR1==9] <- **NA**

2. *Need to recode responses to “no” based on skip patterns*

SPSS	**RECODE** var1 (**SYSMIS**=7).
Stata	**replace** var1=7 **if** var1==.
SAS	**if** VAR1=. **then** VAR1=7;
R	**>** title_of_data_set**$**VAR1**[is.na(**title_of_data_set**$**VAR1)] <- 7

3. *Recoding string variables into numeric*

SPSS	**RECODE** TREE (‘Maple’=1) (‘Oak’=2) INTO TREE_N.
Stata	**generate** TREE_N=. **replace** TREE_N=1 if TREE=="Maple" **replace** TREE_N=2 if TREE=="Oak" OR by using the encode command encode TREE, gen(TREE_N)

SAS	**IF** TREE=‘Maple’ **then** TREE_N=1;
	else if TREE= ‘Oak’ **then** TREE_N=2;

R	(Not necessary in R)

4. *Need to collapse response categories*

SPSS

```
COMPUTE new_region=2.
IF (region=1| region=2|region=3| region=5|region=6) new_region=1.
```

Stata

```
generate new_region =2
replace new_region=1 if region==1| region==2|region==3| region==5|region==6
OR by using the recode command
recode region (1/3 5 6=2) gen(new_region)
```

SAS

```
if region=1 or region=2 or region=3 or region=5 or region=6 then new_region=1;
else if region=4 or region=7 or region=8 or region=9 then new_region=2;
```

R

```
> new_region <- rep(NA, # of observations)
> new_region[title_of_data_set$region == 1 | title_of_data_set$region == 2 |
title_of_data_set$region == 3 | title_of_data_set$region == 5 | title_of_data_set$region
== 6] <- 1
> new_region[title_of_data_set$region == 4 | title_of_data_set$region == 7 |
title_of_data_set$region == 8 | title_of_data_set$region == 9] <- 2
```

5. *Need to aggregate variables*

SPSS

```
IF (socphob=1|gad=1|specphob=1| panic=1|agora=1|ocd=1) anxiety=1.
RECODE anxiety (SYSMIS=0).
```

Stata

```
gen anxiety=1 if socphob==1|gad==1|specphob==1| panic==1|agora==1|ocd==1
replace anxiety=0 if anxiety==.
```

SAS

```
if socphob=1 or gad=1 or specphob=1 or panic=1 or agora=1 or ocd=1 then anxiety=1;
else anxiety=0;
```

R

```
> anxiety <- rep(0, # of observations)
> anxiety[title_of_data_set$socphob == 1 | title_of_data_set$gad==1 |
title_of_data_set$panic == 1 | title_of_data_set$agora==1 | title_of_data_set$ocd == 1]
<- 1
```

6. *Need to create continuous variables*

SPSS

```
COMPUTE nd_sum=sum(nd_symptom1 nd_symptom2 nd_symptom3
nd_symptom4).
```

Stata

```
egen nd_sum=rsum(nd_symptom1 nd_symptom2 nd_symptom3 nd_symptom4)
```

SAS

```
nd_sum=sum (of nd_symptom1 nd_symptom2 nd_symptom3 nd_symptom4);
```

R

```
> nd_sum <- title_of_data_set$nd_symptom1 +  title_of_data_set$nd_symptom2 +
        title_of_data_set$nd_symptom3 +  title_of_data_set$nd_symptom4
> title_of_data_set$nd_sum <- nd_sum
```

7. *Renaming variables*

SPSS	**COMPUTE** newvarname=var1
Stata	**rename** var1 newvarname
SAS	**RENAME** var1=newvarname;
R	**> names(**title_of_data_set)**[**names(title_of_data_set)=="VAR1"**] <-** "newvarname"

8. ---

9. *Labeling variable responses/values*

SPSS	**VALUE LABELS** variable 0 ‘value’ 1 ‘value’ 2 ‘value’ 3 ‘value’
Stata	**label define** name1 0 “value” 1 “value” 2 “value” 3 “value” **label values** variable name1
SAS	**proc format;** variable 0=“value” 1=“value” 2=“value” 3=“value”;
R	**> levels(**title_of_data_set**$**VARIABLE**) <- c("**value**", “**value**")**

10. *Need to further subset the sample*

SPSS	**/SELECT**=diabetes2 EQ 1 (must be added as a command option)
Stata	**if** diabetes2==1 (put this after the command)
SAS	**if** diabetes2=1; (put in the data step before sorting the data)
R	**>** title_of_subsetted_data **<-** title_of_data_set**[**“diabetes2”==1**,]**

Graphing and Data Visualization

1. Univariate

Code for Univariate Output (Categorical):

SPSS

```
FREQUENCIES VARIABLES=var1 var2 var3
/ORDER=ANALYSIS.
```

Stata

```
tab1 var1 var2 var3
```

SAS

```
PROC FREQ; tables var1 var2 var3;
```

R

```
> library(descr)
> freq(as.ordered(title_of_data_set$var1))
> freq(as.ordered(title_of_data_set$var2))
> freq(as.ordered(title_of_data_set$var3))
```

Code for Univariate Output (Quantitative):

SPSS

```
DESCRIPTIVES VARIABLES=var1 var2 var3
/STATISTICS=MEAN STDDEV
```

Stata

```
summarize var1 var2 var3
```

SAS

```
proc means; var var1 var2 var3;
```

R

```
> library(descr)
> freq(as.ordered(title_of_data_set$var1))
> freq(as.ordered(title_of_data_set$var2))
> freq(as.ordered(title_of_data_set$var3))
(Or for mean and sd: )
> summary(title_of_data_set$var1)
```

2. *Bivariate*

Code for Bivariate Output (Categorical IV and Quantitative DV):

SPSS

```
MEANS TABLES=IV by DV
/CELLS MEAN COUNT STDDEV.
```

Stata

```
bys IV: su DV
```

SAS

```
proc sort; by IV;
proc means; var DV; by IV;
```

R

```
> by(title_of_data_set$DV, title_of_data_set$IV, mean)           # for table
> barplot(by(title_of_data_set$DV, title_of_data_set$IV, mean))  # for plots
```

Code for Bivariate Output (Categorical IV and Categorical DV):

SPSS

```
CROSSTABS
/TABLES=DV by IV.
/CELLS=COUNT ROW COLUMN TOTAL.
```

Stata

```
tab DV IV, row column cell
```

SAS

```
Proc freq; tables DV*IV;
```

R

```
> table(title_of_data_set$DV, title_of_data_set$IV) # for table
> prop.table(table(title_of_data_set$DV, title_of_data_set$IV)) # for cell %ages
> prop.table(table(title_of_data_set$DV, title_of_data_set$IV),1) # for row %ages
> prop.table(table(title_of_data_set$DV, title_of_data_set$IV),2) # for column %age

> barplot(prop.table(table(title_of_data_set$DV, title_of_data_set$IV),2)[rows,])) #
for plots of column percentages
```

Note: If your IV is continuous, for graphing purposes, create meaningful categories and then use the code above.

3. *Multivariate*

Code for Multivariate Output (Categorical IV, Quantitative DV, Categorical 3rd VAR):

SPSS

```
MEANS TABLES=DV BY IV BY THIRD_VAR
/CELLS MEAN COUNT STDDEV.
```

Stata

```
bys IV THIRD_VAR: su DV
```

SAS

```
proc sort; by IV THIRD_VAR;

proc means; var DV; by IV THIRD_VAR;
```

R

```
> ftable(by(title_of_data_set$DV, list(title_of_data_set$IV,
title_of_data_set$THIRD_VAR), mean))                # to get table
> barplot(by(title_of_data_set$DV, list(title_of_data_set$IV,
title_of_data_set$THIRD_VAR), mean), beside=T)     # to get plot
```

Code for Multivariate Output (Categorical IV and Categorical DV, Categorical 3rd VAR):

SPSS

```
CROSSTABS
/TABLES=DV BY IV BY THIRD_VAR.
```

Stata

```
bys IV THIRD_VAR: tab DV
```

SAS

```
proc sort; by THIRD_VAR;
proc freq; tables DV*IV; by THIRD_VAR;
```

R

```
> ftable(title_of_data_set$DV, title_of_data_set$IV, title_of_data_set$THIRD_VAR)
                                                              # to get table
> prop.table(ftable(title_of_data_set$DV, title_of_data_set$IV, title_of_data_set$THIRD_VAR))
                                                              # for cell %ages
> prop.table(ftable(title_of_data_set$DV, title_of_data_set$IV,
title_of_data_set$THIRD_VAR),1)                               # for row %ages
> prop.table(ftable(title_of_data_set$DV, title_of_data_set$IV,
title_of_data_set$THIRD_VAR),2)                               # for column %age

> barplot(prop.table(table(title_of_data_set$DV, title_of_data_set$IV,
title_of_data_set$THIRDVAR),2)[rows,]))                       # for plots of column percentage
```

Note: If your 3rd variable is continuous, for graphing purposes, create meaningful categories and then use the code above.

Bivariate Analyses

ANOVA

SPSS

```
ONEWAY QUAN_DV BY CAT_IV
/STATISTICS DESCRIPTIVES.
```

Stata

```
oneway quan_DV cat_IV, tabulate
```

SAS

```
proc anova;
class CAT_IV;
model QUAN_DV = CAT_IV;
means CAT_IV;
```

R

```
> summary(aov(DV ~ IV, data=title_of_data_set))
```

Pearson correlation

SPSS

```
CORRELATIONS
/VARIABLES= QUANIV QUANDV
/STATISTICS DESCRIPTIVES.
```

Stata

```
corr quan_IV quan_DV

OR

pwcorr quant_IV quant_DV, sig
```

SAS

```
Proc corr; var QUAN_IV QUAN_DV;
```

R

```
> cor.test(title_of_data_set$DV, title_of_data_set$IV)
```

Chi-square test

SPSS

```
CROSSTABS
/TABLES= CAT_DV by CAT_IV
/STATISTICS=CHISQ.
```

Stata

```
tab cat_dv cat_iv, chi2 row col
```

SAS

```
Proc freq; tables CAT_DV*CAT_IV/ chisq;
```

R

```
> chisq.test(title_of_data_set$DV, title_of_data_set$IV)
```

POST HOC TESTS WITHIN ANOVA

SPSS

```
UNIANOVA QUAN_DV BY CAT_IV
/POSTHOC=CAT_IV (TUKEY)
/PRINT=ETASQ DESCRIPTIVE.
```

Stata

```
oneway quan_DV cat_IV, sidak
```

SAS

```
Proc anova; class CAT_IV; model QUAN_DV=CAT_IV;
means CAT_IV /duncan;
```

R

```
> TukeyHSD(aov(DV ~ IV, data=title_of_data_set))
```

POST HOC TESTS FOR CHI SQUARE (must subset data in order to conduct 2X2 comparisons)

SPSS

```
TEMPORARY.
SELECT IF CATIV=X OR CAT_IV=Y.
CROSSTABS
/TABLES= CAT_DV CAT_IV
/STATISTICS=CHISQ.
```

Stata

```
keep if cat_IV==1 | cat_IV==3
tab cat_IV cat_DV, chi2
```

SAS

```
IF (CAT_IV = 1) AND (CAT_IV = 3); (in data step)
Proc freq; tables CAT_DV*CAT_IV / chisq;
```

R

```
>  chisq.test(title_of_data_set$DV, title_of_data_set$IV)$observed
         # for actual cell counts
>  chisq.test(title_of_data_set$DV, title_of_data_set$IV)$expected
         # for cell counts expected by chance
>  chisq.test(title_of_data_set$DV, title_of_data_set$IV)$residuals
         # for Pearson residuals (z scores)
```

For 2x2 comparisons:
> chisq.test(title_of_data_set**$**DV[subset]**,** title_of_data_set$IV[subset]**)**

Multivariate Regression: Testing for Confounding

MULTIPLE REGRESSION

SPSS	**REGRESSION** **/DEPENDENT** QUAN_DV **/METHOD ENTER** IV THIRDVAR1 THIRDVAR2
Stata	**reg** quan_DV IV THIRDVAR1 THIRDVAR2
SAS	**Proc reg; model** QUAN_DV=IV THIRDVAR1 THIRDVAR2;
R	my.lm **<- lm(**DV **~** IV **+** THIRDVAR1 **+** THIRDVAR2**, data=**title_of_data_set**)** **> summary(**my.lm**)**

LOGISTIC REGRESSION

SPSS	**LOGISTIC REGRESSION** BINARY_DV with IV THIRDVAR1.
Stata	**logistic** binary_DV IV thirdvar1 thirdvar2 or **logit** binary_DV IV thirdvar1 thirdvar2
SAS	**Proc logistic; class** IV THIRDVAR (when these variables are categorical); **model** BINARY_DV=IV THIRDVAR1 THIRDVAR2;
R	> my.logreg **<- glm(**DV ~ IV + THIRDVAR1 + THIRDVAR2**, data=**title_of_data_set**, family="binomial")** > **summary(**my.logreg**)** # for p-values > **exp(**my.logreg**$coefficients)** # for odds ratios