EDUC 643 Lab: Applied Statistics in Education and Human Services II

Lab 7: 2/18 and 2/19

**Helpsheet for Assignment 3. Use in conjunction with Helpsheet for Assignment 2.**

**Don’t forget to load packages** using **library(**package**)** and **read the data** using **read.csv().** Use **modelsummary()** to format tables.

1. **Recoding to factor variables**

# Inspect to see all the names /levels available

unique(your\_data$cat\_variable) #shows you all the diff names

your\_data$cat\_variable <- as.factor(your\_data$cat\_variable)

# Check levels and order

levels(your\_data$cat\_variable)

#use summary() to select appropriate reference group

summary(your\_data$cat\_variable)

# Set "Select Ref Group" as the reference level

your\_data$cat\_variable <- relevel(your\_data$cat\_variable, ref = "Select Ref Group")

# Check levels again to ensure appropriate order

levels(your\_data$cat\_variable)

1. **Summary statistics for categorical variables**

#deselect the variables that you don’t want to summarize in summary table

desc\_stats <- select(your\_data, -c(ncesid, ncesdistid\_geo, distname, schoolname))

#rename variables for table to intuitive options

names(desc\_stats) <- c("Per Pupil Expenditure", "Free/Reduced Lunch Eligibility", “...”)

**Table for categorical summary statistics**

datasummary\_skim(desc\_stats,

type = "categorical", # or “all” for only all selected variables

fun\_numeric = list(Mean = Mean, SD = SD, Min = Min, Median = Median, Max = Max))

**Descriptive statistics of continuous variable by group**

# Use ` backticks (tilda key) instead of single quotes for the variable names

datasummary(`Per-pupil Expenditure` ~ `Urbanacity` \* (N + Mean + SD), data = desc\_stats) # these are example names, sub in the names you have

1. **Figure to display descriptive comparisons by category— Violin plot**

# you can use your\_data or your desc data, just adjust variable names w/out backticks

ggplot(desc\_stats, aes(x= `School Type`, y= `Per-Pupil Expenditure`, fill= `School Type`)) +

geom\_violin(position = 'dodge') +

stat\_summary(fun = "median", geom = "crossbar", width = 0.5, colour = "red") +

labs(x = "X-axis Title", y = "Y-axis Title",

title = "Give title", caption = "Give caption") +

theme\_minimal(base\_size = 12) +

theme(legend.position = "none")

# some descriptive stats code for graph/personal viewing

**desc\_**stats %>%

group\_by(`Categorical Name`) %>%

summarise(mean(`Outcome Name`))

1. **Analysis of variance (ANOVA) test**

fit1 <- lm(`Per-Pupil Expenditure` ~ `School Type`, data = desc\_stats) #your data or desc\_stats data is fine here

anova(fit1)

# Already done ahead, but to change in regression code can relevel here too:

fit\_ref <- lm(outcome ~ relevel(cat\_variable1, ref= "Reference group name"), data= your\_data)

summary(fit\_ref)

1. **Creating new variable to compare 1 group vs all others**

your\_data <- your\_data %>%

mutate(**new\_variable** = ifelse(cat\_variable1== "Ref", 1, 0),

new\_variable = factor(new\_variable))

# can continue to use desc\_stat data with full names just adjust for back ticks, esle use original you\_data name

fit2 <- lm(outcome ~ new\_variable, data = your\_data)

summary(fit2)

**Summary table output**

modelsummary(list(fit2, fit3),

stars=T,

fmt=2,

vcov = "robust",

gof\_omit = "Adj.|AIC|BIC|Log|RMSE|RSE|Std.Err",

coef\_rename = c(`variable name` = "Name Given”,

....))

1. **Model summary comparison between bivariate analysis and multiple regression**

fit4 <- lm(outcome ~ predictor, data)

fit5 <- lm(outcome ~ predictor + categorical\_variable + covariate1 +covariate2.., data)

# can continue to use desc\_stat data with full names just adjust for back ticks, esle use original you\_data name; true throughout

modelsummary(list(fit4, fit5),

stars=T,

fmt=2,

vcov = "robust",

gof\_omit = "Adj.|AIC|BIC|Log|RMSE|RSE|Std.Err",

coef\_rename = c("variable" = "Intuitie Name",

....), # won’t have to rename if using desc\_stats data

notes= c("Notes."))

1. **Plotting prototypical values for categorical groups using margins()**

proto\_df <- margins :: margins(fit5, at =

list(cat\_variable = c( "group level1 name", "group level2 name”, "group level3 name")))

ggplot(data = proto\_df, aes(x = predictor, y = fitted, color = factor(cat\_variable))) +

geom\_smooth(method = ‘lm’ , se = F) +

labs(x = "X Title Here",

y = "Y Title Here",

title = “Give a Title”)+

scale\_color\_discrete(name = "Cat Levels") + # rename legend not “cat levels”

theme\_minimal()