ST552 Homework 1

Nick Sun

January 17, 2019

Question 1: Initial Data Analysis

Some numerical and graphical summaries that Faraway discusses are useful include statistics such as means, standard deviations, and quartiles, histograms (or alternatively, kernel density estimates). We can start by looking at the data set as a whole using the summary() function.

```
head(teengamb)
```

```
##   sex status income verbal gamble
## 1   1     51   2.00      8    0.0
## 2   1     28   2.50      8    0.0
## 3   1     37   2.00      6    0.0
## 4   1     28   7.00      4    7.3
## 5   1     65   2.00      8   19.6
## 6   1     61   3.47      6    0.1
```

```
teengamb$sex <- factor(teengamb$sex)
levels(teengamb$sex) <- c("male", "female")

summary(teengamb)
```

```
##      sex         status          income           verbal
##  male  :28   Min.   :18.00   Min.   : 0.600   Min.   : 1.00
##  female:19   1st Qu.:28.00   1st Qu.: 2.000   1st Qu.: 6.00
##              Median :43.00   Median : 3.250   Median : 7.00
##              Mean   :45.23   Mean   : 4.642   Mean   : 6.66
##              3rd Qu.:61.50   3rd Qu.: 6.210   3rd Qu.: 8.00
##              Max.   :75.00   Max.   :15.000   Max.   :10.00
##      gamble
##  Min.   :  0.0
##  1st Qu.:  1.1
##  Median :  6.0
##  Mean   : 19.3
##  3rd Qu.: 19.4
##  Max.   :156.0
```

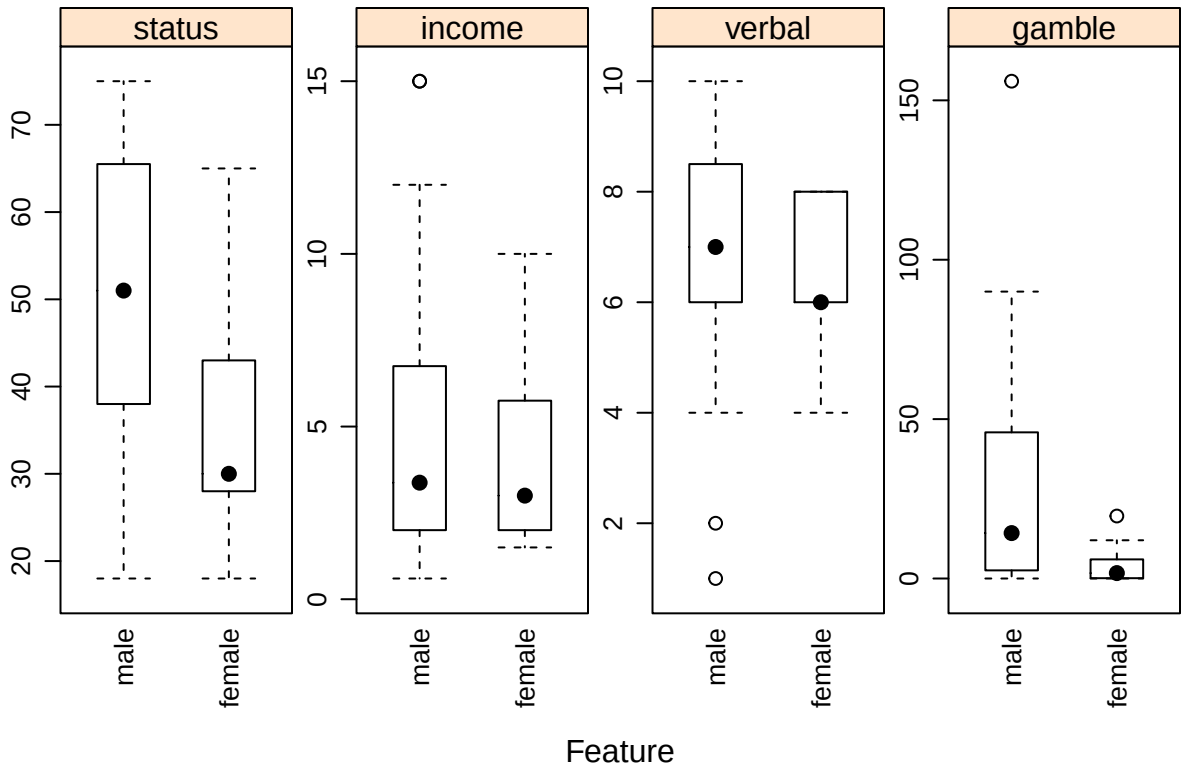
We have a categorical variable *sex* and four numerical variables: *status*, *income*, *verbal*, and *gamble*. It might be interesting to split this data up by sex and see how those individual groups compare. We can do this using dplyr.

```
teengamb %>% group_by(sex) %>% summarise(meanincome = mean(income),
                                         meanstatus = mean(status),
                                         meanverbal = mean(verbal),
                                         meangamble = mean(gamble))
```

```
## # A tibble: 2 x 5
##   sex    meanincome meanstatus meanverbal meangamble
##   <fct>       <dbl>      <dbl>      <dbl>      <dbl>
## 1 male         4.98       52         6.82      29.8
## 2 female       4.15       35.3       6.42       3.87
```

Two variables immediately jump out at us as being noticeably different between the sexes: *status* and *gamble*. It would be helpful to make a visualization. We can make some nice exploratory plots using caret::featurePlot()

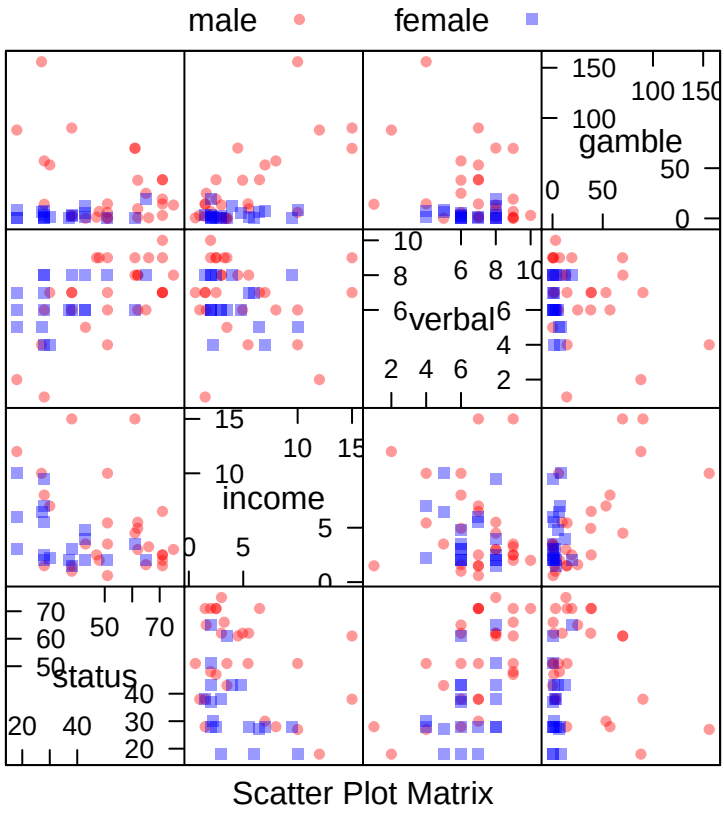
```
featurePlot(x = teengamb[, 2:5],
            y = teengamb$sex,
            plot = "box",
            scales = list(y = list(relation = "free"),
                          x = list(rot = 90)),
            layout = c(4, 1),
            auto.ket = list(columns = 2))
```



From these boxplots, the first thing we notice is that men definitely seem to gamble a lot more than females! Next we might be interested in showing all the data in a scatterplot matrix to really get a handle on how all pairs of the variables look when plotted against each other. We can also do this using the caret::featurePlot() command.

```
transparentTheme(trans = .4, pch = .7)
featurePlot(x = teengamb[, 2:5],
            y = teengamb$sex,
```

```
            plot = "pairs",
            auto.key = list(columns = 2))
```



We see that men and women don't appear to be too different (except for status and gambling which we already established).

Question 2

This question asks us to produce confidence and prediction intervals for certain observations in a given linear model.

```
data(GaltonFamilies, package = "HistData")
slr <- lm(childHeight ~ midparentHeight, data = GaltonFamilies)
summary(slr)
```

```
##
## Call:
## lm(formula = childHeight ~ midparentHeight, data = GaltonFamilies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9570 -2.6989 -0.2155  2.7961 11.6848
##
```

```
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     22.63624    4.26511   5.307 1.39e-07 ***
## midparentHeight  0.63736    0.06161  10.345  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.392 on 932 degrees of freedom
## Multiple R-squared:  0.103,  Adjusted R-squared:  0.102
## F-statistic:   107 on 1 and 932 DF,  p-value: < 2.2e-16
```

a) 95% Confidence interval for the slope parameter

The output of the summary() function gives us the standard error estimates for the intercept and slope parameters. We could use these to build our confidence interval. Alternatively, R has a built-in confint() function that will help us with this.

```
confint(slr, "midparentHeight", level = .95)
```

```
##                     2.5 %    97.5 %
## midparentHeight 0.5164552 0.7582666
```

```
c(slr$coefficients[2] - 1.96*.06161, slr$coefficients[2] + 1.96*.06161)
```

```
## midparentHeight midparentHeight
##       0.5166053       0.7581165
```

Both of these methods output the same confidence interval.

95% Confidence interval for mean child height when the midparentHeight is 72in.

For a confidence interval, the only uncertainty we have to account for comes from the estimates of our slope and intercept:

$$Var(\hat{y_0}) = \sigma^2 \left[\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right]$$

The predict() function in R has the ability to make confidence intervals for us given values of the explanatory variable(s).

```
predvalue <- data.frame(midparentHeight = 72)
predict(slr, predvalue, interval="confidence")
```

```
##        fit      lwr      upr
## 1 68.52623 68.12437 68.92808
```

The upper and lower bounds for the confidence interval are (68.124in, 68.928in) respectively. The interpretation for this interval is that we are 95% confident that the mean response when midparentHeight is 72 inches should fall between 68.124inches and 68.928inches.

95% Prediction interval for mean child height when the midparentHeight is 72in.

When we are predicting a new response, we have to account for variation about the mean in addition to the uncertainty from our parameter estimates

$$Var(Pred(y_0)) = Var(\hat{y_0}) + \sigma^2$$

Lucky for us, the predict() function in R also has a prediction interval option.

```
predict(slr, predvalue, interval="prediction")
```

```
##        fit      lwr      upr
## 1 68.52623 61.85783 75.19462
```

The prediction interval for when midparentHeight is 72 inches is (61.86in, 75.194in). The interpretation of this is that for a new child whose midparentHeight is 72 inches, we have a 95% prediction interval to guess where their height will fall. Notice that this is significantly bigger than our confidence interval.

Extra Credit

midparentHeight is defined as the father's height plus 1.08 times the mother's height divided by 2. Where did the 1.08 come from?

Galton tried to adjust the height of mothers so that it would on average equal the height of the fathers. If you divide the mean height of the fathers by the mean height of the mothers, you will get a factor of ~1.08 Galton could have also mulitpled the fathers' height by .926 to get the same effect.

```
mean(GaltonFamilies$father)/mean(GaltonFamilies$mother)
```

```
## [1] 1.079698
```

Question 3

Here we are asked to compare using a equal variance t-test and a regression model with one categorical explanatory variable in analyzing a randomized experiment.

```
library(Sleuth3)
data <- case0101
```

The question of interest here asks "Is there a significant difference in score between the treatments (intrinsic vs. extrinsic)?" First let's use the t.test() function.

```
t.test(data[data$Treatment == "Intrinsic",]$Score, data[data$Treatment == "Extrinsic",]$Score, var.equal
```

```
##
##  Two Sample t-test
##
## data:  data[data$Treatment == "Intrinsic", ]$Score and data[data$Treatment == "Extrinsic", ]$Score
## t = 2.9259, df = 45, p-value = 0.005366
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.291432 6.996973
## sample estimates:
## mean of x mean of y
##  19.88333  15.73913
```

Now let's try answering the same question but using a linear model with a single categorical explanatory variable.

```
slr01 <- lm(Score ~ Treatment, data = data)
summary(slr01)
```

```
##
## Call:
## lm(formula = Score ~ Treatment, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.739  -2.983   1.061   2.961   9.817
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          15.739      1.012  15.550  < 2e-16 ***
## TreatmentIntrinsic    4.144      1.416   2.926  0.00537 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.854 on 45 degrees of freedom
## Multiple R-squared:  0.1598, Adjusted R-squared:  0.1412
## F-statistic: 8.561 on 1 and 45 DF,  p-value: 0.005366
```

Here we see something interesting, but also intuitive. The t-statistic returned by the t.test() function and the t-statistic for the slope parameter estimate is the same! The t.test() is trying to find a statistically significant difference between the treatments i.e. testing the null hypothesis $H_0 : \mu_{\text{intrinsic}} - \mu_{\text{extrinsic}} = 0$. For our model, we see that the intercept is actually the mean of our extrinsic group and that the parameter associated with the treatments is actually a dummy variable that takes on the value of the difference between the Intrinsic and Extrinsic groups *if* the subject was in the Intrinsic group (otherwise it is 0). If there was actually no difference between the treatment groups and this difference was really equal to 0 (notice that this is the same hypothesis that we are testing with the t.test!), then this parameter estimate should not be statistically significant from 0 and the associated t-test would have a high p-value.

```
library(dplyr)

means <- data %>% group_by(Treatment) %>% summarise(avg = mean(Score))
means
```

```
## # A tibble: 2 x 2
##   Treatment   avg
##   <fct>     <dbl>
## 1 Extrinsic  15.7
## 2 Intrinsic  19.9
```

```r
means[2, 2] - means[1, 2]
```

```
##        avg
## 1 4.144203
```

Extra Credit

In simple linear regression, what hypothesis is the p-value on the intercept testing? Is there an equivalent t-test?

First, what is the intercept telling us in the model? The intercept is a constant term that is the expected mean value of the response when the predictor variable is 0. If the predictor never does or never can equal 0, then the intercept doesn't really have an interpretation.

The t-test on the intercept is testing if the intercept is significantly different from 0. In our case, since our model only has one categorical predictor with two levels the real-world interpretation of our intercept is actually simpler; is the mean of the Extrinsic group equal to 0? Since it would be very unlikely to have an entire group of people average around 0 for an exam, we would expect that this p-value is very small. This particular t-test has an accompanying p-value of 2e-16, which is indeed super small!