Crime Rate Dataset analysis

Patrick Cullinane

05/16/2022

Crime Rate Dataset

```
library(dplyr)
library(ggplot2)
library(car)
library(ggpubr)
library(tidyverse)
library(rstatix)
library(ggstatsplot)
library(FSA)
library(boot)

setwd("C:/Users/Laura/Documents/code/MA-541")
df <- read.csv("Crime_R.csv")


#str(df)
```

Introduction

The Crime Rate dataset is comprised of data crime rate and associated variables collected in the United States at two different periods of time. The dataset is broken down into 27 columns and 47 rows. The dataset was developed by the University of Sheffield

```
dim(df)
```

```
## [1] 47 27
```

```
names(df)
```

```
##  [1] "CrimeRate"           "Youth"               "Southern"
##  [4] "Education"           "ExpenditureYear0"    "LabourForce"
##  [7] "Males"               "MoreMales"           "StateSize"
## [10] "YouthUnemployment"   "MatureUnemployment"  "HighYouthUnemploy"
## [13] "Wage"                "BelowWage"           "CrimeRate10"
## [16] "Youth10"             "Education10"         "ExpenditureYear10"
## [19] "LabourForce10"       "Males10"             "MoreMales10"
## [22] "StateSize10"         "YouthUnemploy10"     "MatureUnemploy10"
## [25] "HighYouthUnemploy10" "Wage10"              "BelowWage10"
```

The first 13 columns consist of crime rate data and other measurements taken at a point in time, and the next set consists of the same data taken a decade later. Additionally there is a column called Southern which consists of a binary variable, 1 if a Southern state 0 otherwise. The Southern column applys to both sets of columns within the dataset. We are not told the exact date at which the data has been collected. Overall the data is a mixture of discrete, binary, continuous variables.

```
# split data into year 0 and year + 10

df0 <- df %>%
  select(-ends_with('10'))
df10 <- df %>%
  select(ends_with('10'))

head(df0,2)
```

```
##   CrimeRate Youth Southern Education ExpenditureYear0 LabourForce Males
## 1      45.5   135        0      12.4               69         540   965
## 2      52.3   140        0      10.9               55         535  1045
##   MoreMales StateSize YouthUnemployment MatureUnemployment HighYouthUnemploy
## 1         0         6                80                 22                 1
## 2         1         6               135                 40                 1
##   Wage BelowWage
## 1  564       139
## 2  453       200
```

```
tail(df0,2)
```

```
##    CrimeRate Youth Southern Education ExpenditureYear0 LabourForce Males
## 46     157.7   136        0      15.1              149         577   994
## 47     161.8   131        0      13.2              160         631  1071
##    MoreMales StateSize YouthUnemployment MatureUnemployment HighYouthUnemploy
## 46         0       157               102                 39                 0
## 47         1         3               102                 41                 0
##    Wage BelowWage
## 46  673       167
## 47  674       152
```

Experiment 1

The first question we will examine will be whether there is a relationship between Males per 1000 females and states classified as Southern. To accomplish this we will use 3 columns; Southern, Males, and Males10. As mentioned previously Southern is a binary variable while Males and Males10 are discrete variables. Males and Males10 both refer to the number of males per 1000 females in counted in a US State.

To get an idea of the make-up of the data we look at the summary statistics.

```
str(df[,c("Southern","Males","Males10")])
```

```
## 'data.frame':    47 obs. of  3 variables:
##  $ Southern: int  0 0 1 1 0 0 0 0 0 0 ...
##  $ Males   : int  965 1045 962 968 989 972 984 977 968 1024 ...
##  $ Males10 : int  974 1039 959 983 989 983 993 973 968 1024 ...
```

```
summary(df[,c("Southern","Males","Males10")])
```

```
##     Southern          Males           Males10
##  Min.   :0.0000   Min.   : 934.0   Min.   : 935.0
##  1st Qu.:0.0000   1st Qu.: 964.5   1st Qu.: 969.5
##  Median :0.0000   Median : 977.0   Median : 983.0
##  Mean   :0.3404   Mean   : 983.0   Mean   : 986.9
##  3rd Qu.:1.0000   3rd Qu.: 992.0   3rd Qu.: 994.0
##  Max.   :1.0000   Max.   :1071.0   Max.   :1079.0
```

Next let's look at how the data's shape using some plots.

```
df1 <- df[,c("Southern","Males")]
df2 <- df[,c("Southern","Males10")]
stem(df$Southern)
```

```
##
##   The decimal point is 1 digit(s) to the left of the |
##
##    0 | 00000000000000000000000000000000
##    2 |
##    4 |
##    6 |
##    8 |
##   10 | 0000000000000000
```

```
stem(df$Males)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    92 | 48
##    94 | 803356
##    96 | 24445688992223478
##    98 | 1244556690468
##   100 | 228
##   102 | 498
##   104 | 59
##   106 | 1
```

```
stem(df$Males10)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    92 | 5
##    94 | 5899269
##    96 | 28890134688
##    98 | 022233377992333359
##   100 | 131
##   102 | 4499
##   104 | 08
##   106 | 9
```

We can see from the stem plots that we have more Southern states. We should note that although it is not explicity stated in the data what constitutes a Southern state the data seems to align with the US census bureau's definition of a Southern state. The US census counts 16 states in total as Southern States, which appears to align with our data. Alternatively we can see there are 31 "0" states which in total make 47 states counted in this dataset. We are not told what comprise the 31 non-Southern states.

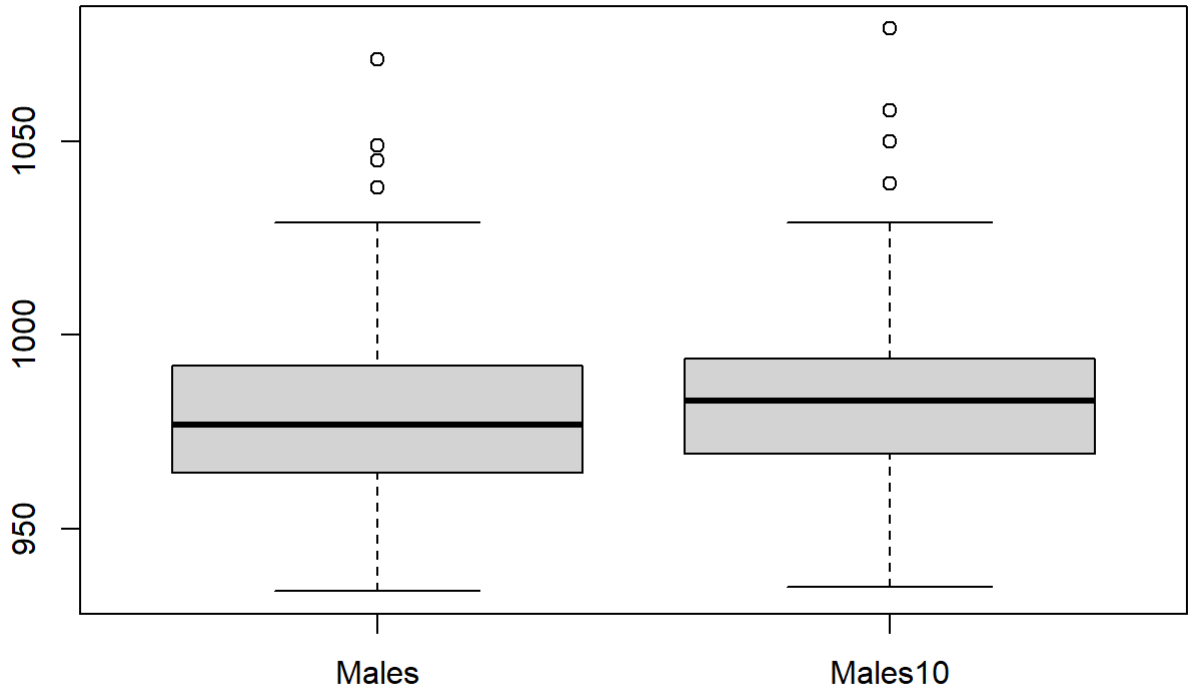
```
sum(df$Southern == 1)
```

```
## [1] 16
```

```
sum(df$Southern == 0)
```

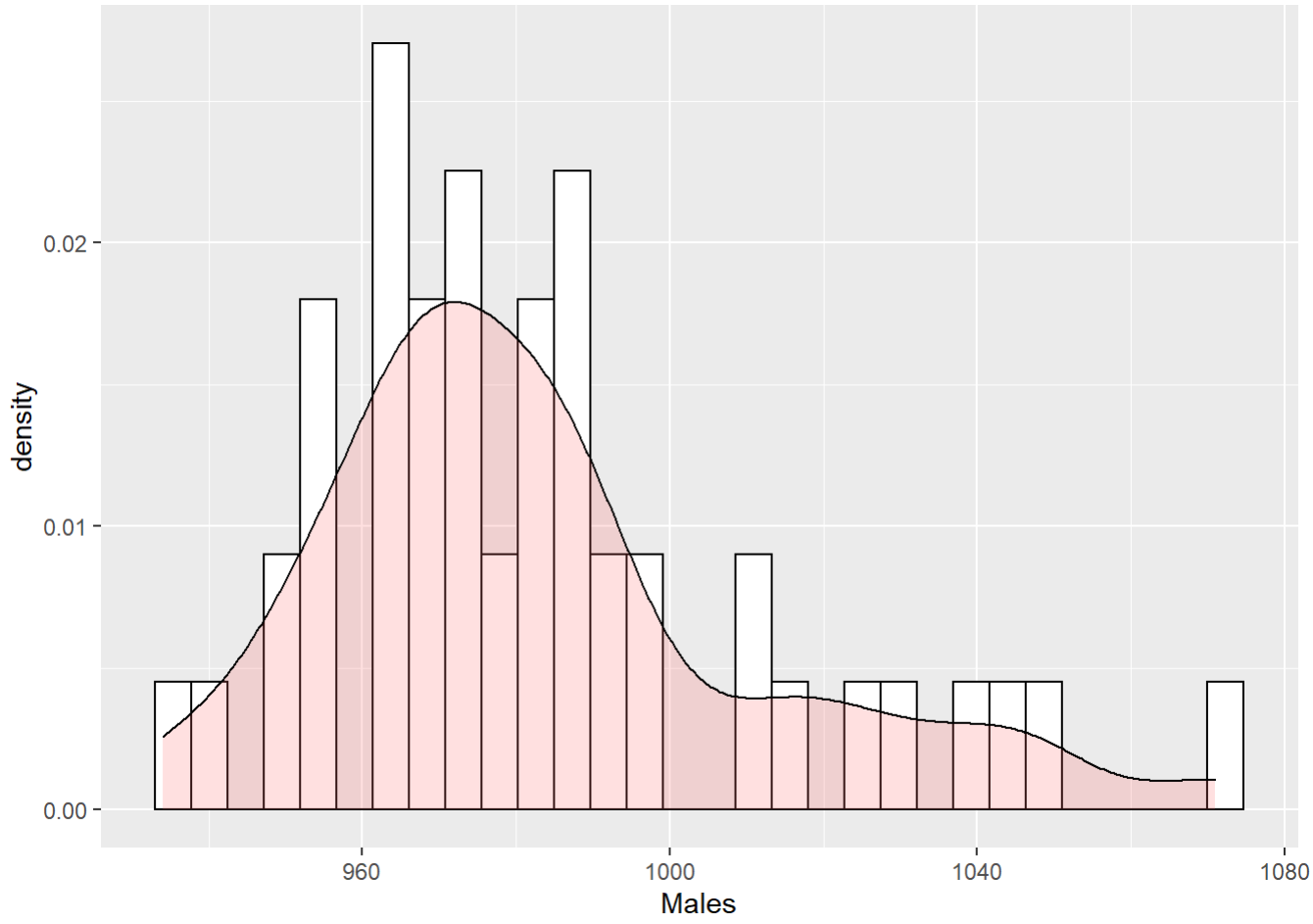
```
## [1] 31
```

```
boxplot(df[,c("Males","Males10")])
```

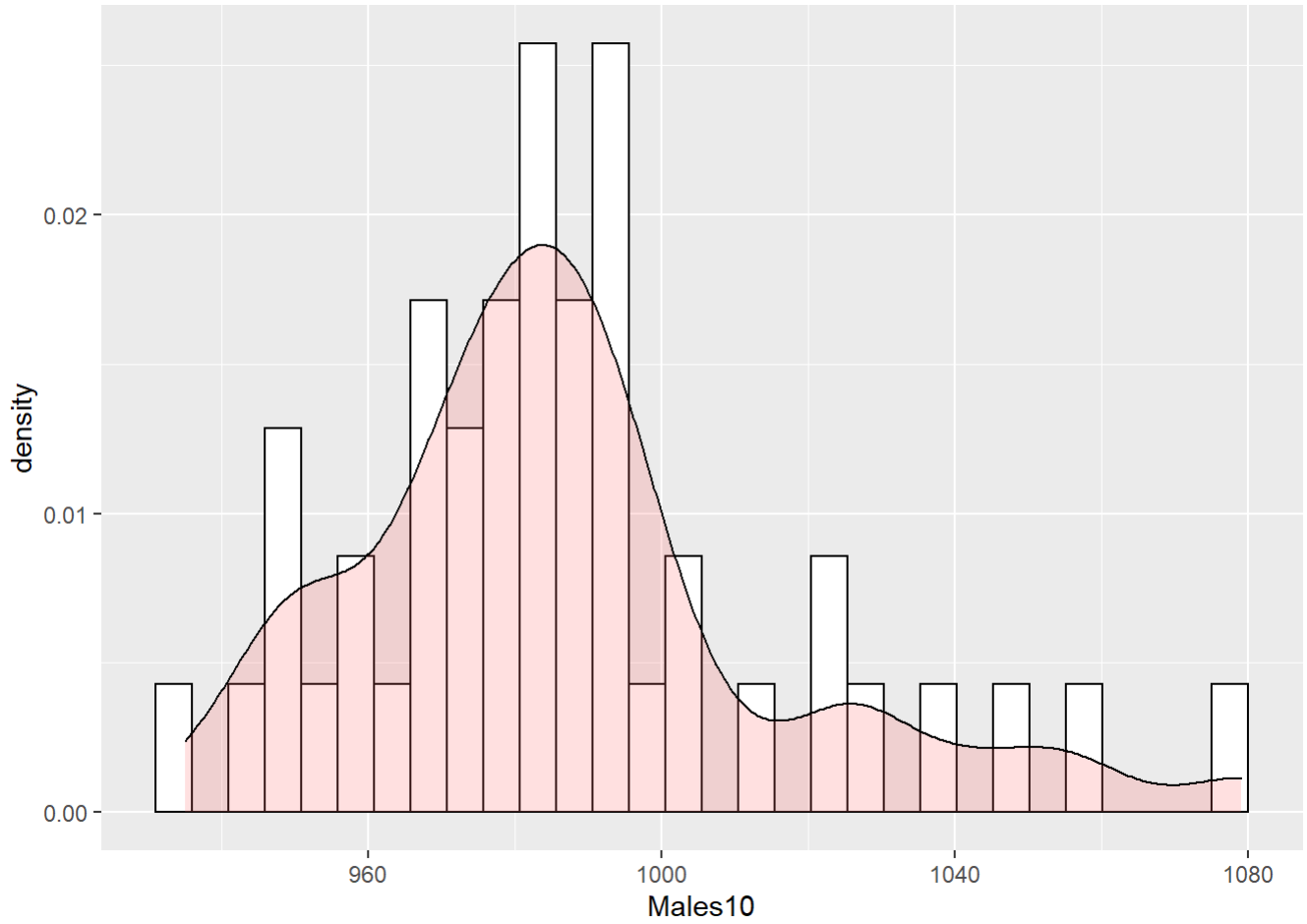


From the stem plots we can see that Males and Males10 both appear to be rightly skewed. We can examaine this in more detail using a histogram.

```
ggplot(df, aes(x=Males)) +
 geom_histogram(aes(y=..density..), colour="black", fill="white")+
 geom_density(alpha=.2, fill="#FF6666")
```




```
ggplot(df, aes(x=Males10)) +
 geom_histogram(aes(y=..density..), colour="black", fill="white")+
 geom_density(alpha=.2, fill="#FF6666")
```



More detailed examination with our histograms and density plots again show us that data may be skewed to the right. This will be important to note as we further examine whether there is a difference in means between groups of data.

With this intuition we will examine using anova whether there is a significant difference in group means. To accomplish this we must first encode our the Southern column to match Males and Males10, and then stack the data frames on top of eachother. Next we will run an anova test on the data.

Next we formulate the following hypothesis: \[H_0: \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 \] \[H_1: \alpha_I \neq \alpha_k \] The null hypothesis states that all groups measured will be the same while the alternative hypothesis states there is a difference in atleast two of the groups.

To proceed further we need to make four groups of data to compare. To do this we will map a new variable from Southern to Males10. We will stick with the current mapping of the Males and Southern column which is “1” if Southern otherwise “0”.

In the second group we will use “3” to denote a Southern state and “2” otherwise. In this way we will have 4 groups of Males. At this point we will have four groups: 0:Non-Southern Males, 1:Southern Males, 2:Non-Southern Males10, 3:Southern Males. To run the test we then stack the data on top of eachother to create one long column of Males and Southern variables. Additionally we establish that we will reject the null hypothesis at an alpha of 0.05.

```
df2$Southern <- ifelse(df2$Southern == 0, 2,3)
names(df2)[2] <- "Males"

df_stack <- rbind(df1,df2)
df_stack$Southern <- as.factor(df_stack$Southern)

aov_test <- aov(Males ~ Southern, data=df_stack)
summary(aov_test)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Southern     3   8765  2921.8   3.627  0.016 *
## Residuals   90  72501   805.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
glm_test <- glm(Males ~ Southern, data=df_stack, family=quasipoisson)

summary(glm_test)
```

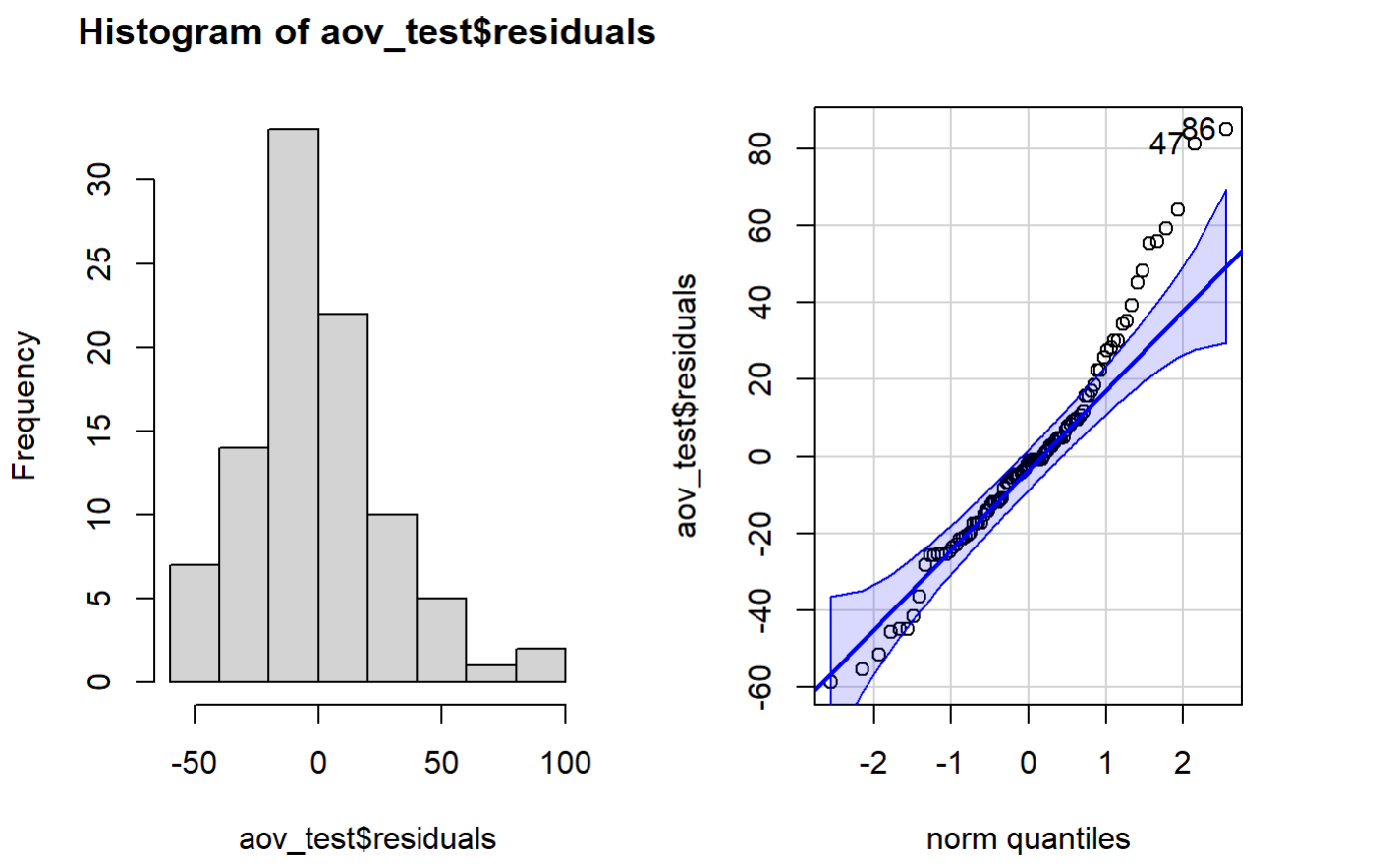
```
##
## Call:
## glm(formula = Males ~ Southern, family = quasipoisson, data = df_stack)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8863  -0.5603  -0.0562   0.3359   2.6631
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  6.897314   0.005150 1339.279   <2e-16 ***
## Southern1   -0.019760   0.008885   -2.224   0.0286 *
## Southern2    0.004294   0.007275    0.590   0.5566
## Southern3   -0.016609   0.008875   -1.871   0.0645 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 0.8136634)
##
##     Null deviance: 81.727  on 93  degrees of freedom
## Residual deviance: 72.810  on 90  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 3
```

As we can see from our test there appears to be a signficant difference between atleast two of the groups as the p-value is below our threshold of 0.05. Our test does not tell us which groups are different so we need to perform further tests to gain more insight.

Although it appears that we can proceed with post-hoc testing to compare groups from our previous examination of the underlying data's distribution we still need to check that the underlying assumption of normality is met.

To do this we will use both graphical methods and formally with the Shapiro-Wilk test. We will use the residuals of the anova test to test this assumption.

```
par(mfrow = c(1,2))
hist(aov_test$residuals)
qqPlot(aov_test$residuals)
```




```
## [1] 86 47
```

Graphically it appears that the residuals are not nornally distributed but we need to examine this more formally with a kruskal test.

We formulate the null hypothesis that the data comes from a normal distribution and the alternative hypothesis that the data does not come from a normal distribution and we establish an alpha of 0.05.

```
kruskal.test(Males ~ Southern, data = df_stack)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Males by Southern
## Kruskal-Wallis chi-squared = 10.624, df = 3, p-value = 0.01394
```

From our test we can see that our p-value is below our establish alpha level so we can reject the null hypothesis that the data comes from a normal distribution. With this bit of information we will proceed with a non-parametric test. In this case we will use the Dunn test. We formulate the same hypothesis we used for the anova test: The null hypothesis that the groups are equal and the the alertnative hypothesis that they

```
dunnTest(Males ~ Southern,
  data = df_stack,
  method = "holm"
)
```

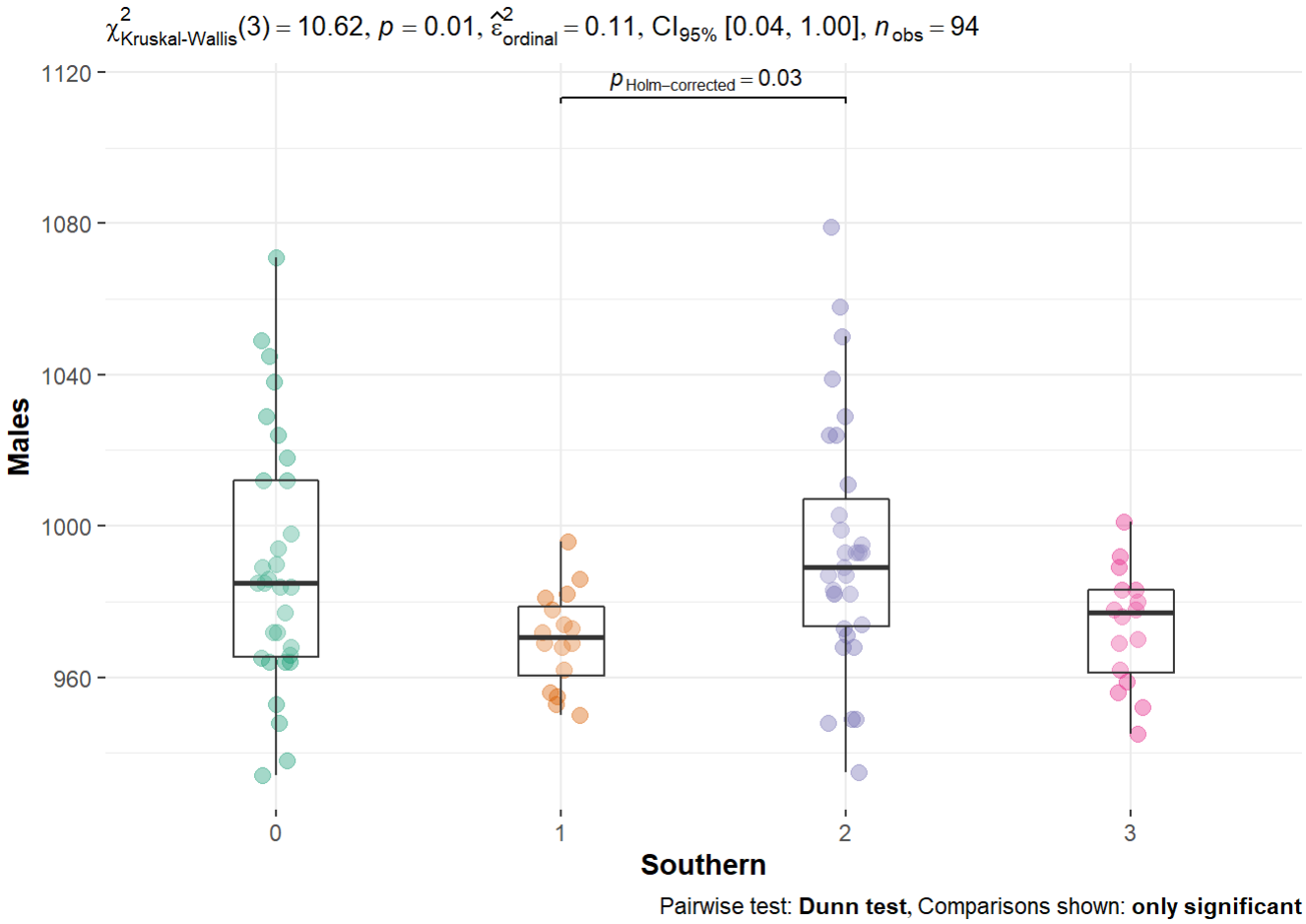
```
##   Comparison          Z     P.unadj     P.adj
## 1      0 - 1  2.1534446 0.031283751 0.1251350
## 2      0 - 2 -0.8056132 0.420465920 0.8409318
## 3      1 - 2 -2.8181859 0.004829584 0.0289775
## 4      0 - 3  1.5950908 0.110691931 0.3320758
## 5      1 - 3 -0.4861414 0.626866936 0.6268669
## 6      2 - 3  2.2598321 0.023831676 0.1191584
```

Results Discussion

After running our non-parametric tests we see that there is a significant difference between group 1 and 2 ($p < 0.05$). Where group 1 corresponds to Non-Southern States & Males/1000 Females and group 2 are Southern States Males/1000 Females 10 years later. Therefore we accept the alertnative hypothesis that there is a difference in group 1 and group 2.

From the results it appears that there is a difference between Southern and Non-Southern Males/Females a decade apart although no evidence was found to show a difference within the same year. Although a difference was found through these methods future analysis

```
ggbetweenstats(
  data = df_stack,
  x = "Southern",
  y = "Males",
  type = "Non-parametric",
  plot.type = "box",
  pairwise.comparisons = TRUE,
  pairwise.display = "significant",
  centrality.plotting = FALSE,
  bf.message = FALSE
)
```



Experiment 2

The next experiment deals with examining if a relationship exists between variables of the dataset and CrimeRate and if we can build a model to predict Crimerate.

```
stem(df$CrimeRate)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    4 | 627
##    6 | 0481358
##    8 | 023568245779
##   10 | 034677892457
##   12 | 0237268
##   14 | 15948
##   16 | 2
```

To get started we look at the year 0 data to build any intuition on relationships between CrimeRate and any other variables.

```
model <- lm(CrimeRate~.,data=df0)
summary(model)
```

```
##
## Call:
## lm(formula = CrimeRate ~ ., data = df0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.204 -10.557   2.919  10.391  32.707
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -258.30363  192.43539  -1.342  0.18866
## Youth                 0.86498    0.35319   2.449  0.01980 *
## Southern              0.56966   12.04365   0.047  0.96256
## Education             6.43119    3.75033   1.715  0.09575 .
## ExpenditureYear0      0.71271    0.20199   3.528  0.00125 **
## LabourForce           0.10771    0.12281   0.877  0.38680
## Males                -0.18383    0.23656  -0.777  0.44265
## MoreMales            17.33920   15.83577   1.095  0.28147
## StateSize            -0.09895    0.11444  -0.865  0.39349
## YouthUnemployment    -0.09173    0.46132  -0.199  0.84361
## MatureUnemployment    0.68776    0.99491   0.691  0.49423
## HighYouthUnemploy    -4.49806   10.82134  -0.416  0.68035
## Wage                  0.19189    0.08950   2.144  0.03950 *
## BelowWage             0.55336    0.20693   2.674  0.01156 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.17 on 33 degrees of freedom
## Multiple R-squared:  0.6842, Adjusted R-squared:  0.5598
## F-statistic:   5.5 on 13 and 33 DF,  p-value: 3.616e-05
```

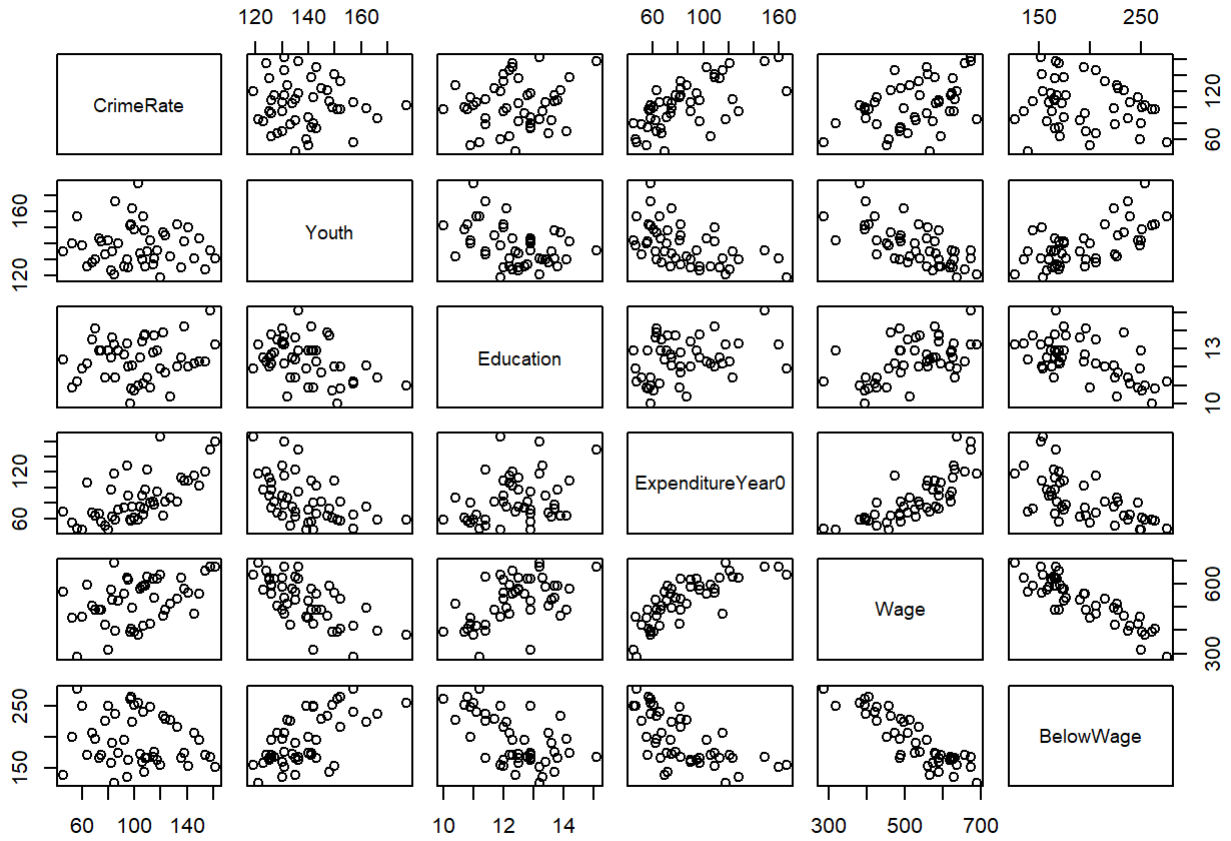
We can see from running the linear model on the data that it appears some relationship exists between CrimeRate, Youth, Education, ExpenditureYear0, Wage, and BelowWage. We will take these five features and perform further investigation on them to see if how their structure.

```
features <- df[,c("CrimeRate","Youth","Education","ExpenditureYear0","Wage","BelowWage")]

str(features)
```

```
## 'data.frame':    47 obs. of  6 variables:
##  $ CrimeRate       : num  45.5 52.3 56.6 60.3 64.2 67.6 70.5 73.2 75 78.1 ...
##  $ Youth           : int  135 140 157 139 126 128 130 143 141 133 ...
##  $ Education       : num  12.4 10.9 11.2 11.9 12.2 13.5 14.1 12.9 12.9 11.4 ...
##  $ ExpenditureYear0: int  69 55 47 46 106 67 63 66 56 51 ...
##  $ Wage            : int  564 453 288 457 593 507 486 487 489 425 ...
##  $ BelowWage       : int  139 200 276 249 171 206 196 166 170 225 ...
```

```
pairs(features)
```



Based on the structure of the features it appears they would benefit from some transformation. We will conduct further experiments on what features to keep/transform.

The first experiment we will run is removing the weakest variable to see if we have any improvment on the R-squared value and if we can see if the model without education is significantly better using anova.

```
lm.fit <- lm(
  CrimeRate ~
    Education + Youth + Wage + BelowWage
    + ExpenditureYear0, data=df0)

lm.fit2 <- lm(
  CrimeRate ~
    Youth + Wage + BelowWage
    + ExpenditureYear0, data=df0
)

summary(lm.fit)
```

```
##
## Call:
## lm(formula = CrimeRate ~ Education + Youth + Wage + BelowWage +
##     ExpenditureYear0, data = df0)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -43.32 -12.69   3.12  10.78  32.52
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -338.74486   90.91882  -3.726 0.000588 ***
## Education           4.72597    3.05412   1.547 0.129450
## Youth               0.78508    0.29627   2.650 0.011387 *
## Wage                0.20208    0.08097   2.496 0.016679 *
## BelowWage           0.55952    0.15831   3.534 0.001029 **
## ExpenditureYear0    0.69979    0.15487   4.519  5.2e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.55 on 41 degrees of freedom
## Multiple R-squared:  0.6326, Adjusted R-squared:  0.5878
## F-statistic: 14.12 on 5 and 41 DF,  p-value: 4.872e-08
```

```
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = CrimeRate ~ Youth + Wage + BelowWage + ExpenditureYear0,
##     data = df0)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -46.02 -12.06   3.09  12.70  33.83
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -265.89320   79.06065  -3.363 0.001653 **
## Youth               0.76376    0.30082   2.539 0.014913 *
## Wage                0.21169    0.08206   2.580 0.013475 *
## BelowWage           0.49014    0.15432   3.176 0.002797 **
## ExpenditureYear0    0.66540    0.15579   4.271 0.000109 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.86 on 42 degrees of freedom
## Multiple R-squared:  0.6111, Adjusted R-squared:  0.5741
## F-statistic:  16.5 on 4 and 42 DF,  p-value: 3.367e-08
```

```
anova(lm.fit, lm.fit2)
```

```
## Analysis of Variance Table
##
## Model 1: CrimeRate ~ Education + Youth + Wage + BelowWage + ExpenditureYear0
## Model 2: CrimeRate ~ Youth + Wage + BelowWage + ExpenditureYear0
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     41 14110
## 2     42 14934 -1   -824.05 2.3945 0.1294
```

We can see that the model without the Education feature does not perform significanlty better. So we will move on to comparing a two models where we have log transformed 4 features in the first model (Youth, Wage, BelowWage and ExpenditureYear0) and model 2 has the same data but this time we remove Wage.

```
lm.fit3 <- lm(
  CrimeRate ~
    + log10(Youth)
    + log10(Wage)
    + log10(BelowWage)
    + log10(ExpenditureYear0), data=df0
)

lm.fit4 <- lm(
  CrimeRate ~
  + log10(Youth)
  + log10(BelowWage)
  + log10(ExpenditureYear0), data=df0
)

summary(lm.fit3)
```

```
##
## Call:
## lm(formula = CrimeRate ~ +log10(Youth) + log10(Wage) + log10(BelowWage) +
##     log10(ExpenditureYear0), data = df0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.094 -12.065   0.593  12.248  27.813
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1719.42     364.93  -4.712 2.70e-05 ***
## log10(Youth)               248.00      89.06   2.785 0.008003 **
## log10(Wage)                168.54      75.88   2.221 0.031799 *
## log10(BelowWage)           218.64      56.90   3.843 0.000405 ***
## log10(ExpenditureYear0)    176.39      29.61   5.958 4.57e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.07 on 42 degrees of freedom
## Multiple R-squared:  0.6815, Adjusted R-squared:  0.6512
## F-statistic: 22.47 on 4 and 42 DF,  p-value: 5.63e-10
```

```
summary(lm.fit4)
```

```
##
## Call:
## lm(formula = CrimeRate ~ +log10(Youth) + log10(BelowWage) + log10(ExpenditureYear0),
##     data = df0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.539 -12.120   3.539  11.659  28.879
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1032.55     202.39  -5.102 7.25e-06 ***
## log10(Youth)               195.90      89.76   2.182  0.03458 *
## log10(BelowWage)           134.11      44.18   3.035  0.00407 **
## log10(ExpenditureYear0)    215.46      24.88   8.659 5.64e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.83 on 43 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6193
## F-statistic: 25.94 on 3 and 43 DF,  p-value: 9.768e-10
```

```
anova(lm.fit3, lm.fit4)
```

```
## Analysis of Variance Table
##
## Model 1: CrimeRate ~ +log10(Youth) + log10(Wage) + log10(BelowWage) +
##     log10(ExpenditureYear0)
## Model 2: CrimeRate ~ +log10(Youth) + log10(BelowWage) + log10(ExpenditureYear0)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     42 12231
## 2     43 13668 -1   -1436.5 4.9328 0.0318 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Our R-squared has improved slightly from the non-log transformed models and it looks like the second model in this group performs signficanly better. With this in mind we will now use a validation set approach to estimate the test error rate. We will split our data up roughly in half in this case because we don't have many training examples to work with. We will be using MSE to test the effectiveness of our model.

```
set.seed(1)
train <- sample(47,24)

lm.fit4 <- lm(
  CrimeRate ~
  + log10(Youth)
  + log10(BelowWage)
  + log10(ExpenditureYear0), data=df0, subset = train
)

mean((df$CrimeRate - predict(lm.fit4, df0))[-train]^2)
```

```
## [1] 360.7829
```

The MSE estimate for this approach is 360.78

```
lm.fit4 <- lm(
  CrimeRate ~
  + poly(log10(Youth),2)
  + poly(log10(BelowWage),2)
  + poly(log10(ExpenditureYear0),2), data=df0, subset = train
)

mean((df$CrimeRate - predict(lm.fit4, df0))[-train]^2)
```

```
## [1] 275.9393
```

Adding the poly feature to our features reduces the MSE to 275.93.

Our next approach is to see if a dimensionality reduction technique can improve our model even more. In this case we are using PCA along with the three features from the model above.

```
new_features <- df[,c("Youth","BelowWage","ExpenditureYear0")]
pca <- prcomp(features, scale. = TRUE)


lm_pca <- lm(df0$CrimeRate ~ pca$x[,1] + pca$x[,2], subset=train)
mean((df0$CrimeRate - predict(lm_pca, df0))[-train]^2)
```

```
## [1] 81.53669
```

PCA showed a signficant drop in MSE as compared to our previous attempts. However, to get a more robust idea of how our model performs across the whole dataset.

Next we will use the caret package to create a model pipeline were we can combine our preprocessing techique (PCA), cross-validation, and our linear regression model.

```
library(caret)

set.seed(12)
train_index <- sample(1:nrow(df0), 0.6 * nrow(df0))
X_train <- df0[train_index, ]
X_test <- df0[-train_index, ]
head(X_train)
```

```
##    CrimeRate Youth Southern Education ExpenditureYear0 LabourForce Males
## 2       52.3   140        0      10.9               55         535  1045
## 26     105.9   130        0      13.4               90         623  1049
## 16      88.0   140        0      12.9               71         632  1029
## 27     106.6   157        1      11.1               65         553   955
## 5       64.2   126        0      12.2              106         599   989
## 43     145.4   131        1      12.2              115         542   969
##    MoreMales StateSize YouthUnemployment MatureUnemployment HighYouthUnemploy
## 2          1         6               135                 40                 1
## 26         1         3               113                 40                 0
## 16         1         7               100                 24                 1
## 27         0        39                81                 28                 0
## 5          0        40                78                 25                 1
## 43         0        50                79                 35                 0
##    Wage BelowWage
## 2   453       200
## 26  588       160
## 16  526       174
## 27  421       239
## 5   593       171
## 43  472       206
```

```
fit2 <- train(
    CrimeRate ~
      Youth + BelowWage + ExpenditureYear0
    , data=X_train
    , method = "lm"
    , preProcess=c("pca")
    , trControl = trainControl(method = "cv")
)

fit2_pred <- predict(fit2, X_test)
fit2
```

```
## Linear Regression
##
## 28 samples
##  3 predictor
##
## Pre-processing: principal component signal extraction (3), centered (3),
##  scaled (3)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 25, 25, 25, 25, 26, 26, ...
## Resampling results:
##
##   RMSE     Rsquared   MAE
##   22.1224  0.7245961  19.39399
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
postResample(pred = fit2_pred, obs = X_test$CrimeRate)
```

```
##       RMSE   Rsquared        MAE
## 22.1239478  0.3561878 17.7719769
```

```
mean(
  (X_test$CrimeRate - predict(
            fit2, X_test[,c("Youth","BelowWage","ExpenditureYear0")])
   )^2
)
```

```
## [1] 489.4691
```

After putting it into a pipeline and testing

```
fit3 <- train(
    CrimeRate ~
      log10(Youth) + log10(BelowWage) + log10(ExpenditureYear0)
    , data=X_train
    , method = "BstLm"
    , preProcess=c("center", "scale", "YeoJohnson", "pca")
    , trControl = trainControl(method = "cv")
)

fit3_pred <- predict(fit3, X_test)
fit3
```

```
## Boosted Linear Model
##
## 28 samples
##  3 predictor
##
## Pre-processing: centered (3), scaled (3), Yeo-Johnson transformation
##  (3), principal component signal extraction (3)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 26, 25, 25, 25, 25, 25, ...
## Resampling results across tuning parameters:
##
##   mstop  RMSE      Rsquared   MAE
##    50    18.36448  0.8627987  16.79130
##   100    18.59407  0.8481486  16.68138
##   150    18.67319  0.8460444  16.66383
##
## Tuning parameter 'nu' was held constant at a value of 0.1
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were mstop = 50 and nu = 0.1.
```

```
postResample(pred = fit3_pred, obs = X_test$CrimeRate)
```

```
##       RMSE   Rsquared        MAE
## 20.3121927  0.4507671 16.8576018
```

```
mean(
  (X_test$CrimeRate - predict(
            fit3, X_test[,c("Youth","BelowWage","ExpenditureYear0")])
   )^2
)
```

```
## [1] 412.5852
```

Conclusions

We started off by running some basic linear regression on the dataset to see if we could notice any correlations that we could quickly iterate on. We did see that some variables, mainly Youth, Wage/BelowWage, ExpenditureYear0, appeared to be better correlated with CrimeRate. From there we looked at how some basic transformations impacted the data. It appeared that log transformation did well. This was confirmed through comparing models using anova.

After exploring some basic feature elimination/transformation we moved into modeling our data. We started out with a basic linear model and compared models using log transformation and quadratic polynomials. Ultimately PCA appeared to perform best and we moved into the final phase which involved creating a predictive pipeline using the caret library.

The first pipeline method used a simple combination of PCA and linear regression. 10-fold cross validation was applied to make sure we were not overfitting and we did a 50% training and test split. Our final model used the Boosted Linear Model from caret, with preprocessing steps that involved centered, scaled, Yeo-Johnson transformation, and pca. Tweaking the training set slighly we chose 60/40 as the train/test split as this produced the best results.

Ultimatley we were unable to produce a model with that was able to have a high R^2 and low MSE value. This may be due to the limited dataset size. Future work can be conducted on finding similar datasets that have more data, as this will allow us

References:

1. www.statstutor.ac.uk
2. US Census