Searching for Similarity: Regression

Diego Ochoa

CSV File and Data

https://www.kaggle.com/datasets/krantiswalke/bankfullcsv

```
  #Load data
  set.seed(1301)
  bank <- read.csv("C:/Users/Diego/Desktop/bank-full.csv")
  bank <- bank[-sample(nrow(bank), 31083), ] #keep 14128 observations, the k-map was having trouble wit
  bank$job <- factor(bank$job)
  bank$housing <- factor(bank$housing)
  bank$marital <- factor(bank$marital)

  #Divide data
  i <- sample(1:nrow(bank), nrow(bank)*0.8, replace=FALSE)
  train <- bank[i,]
  test <- bank[-i,]
```

Graphical Analysis

For this section of the project we will create a regression model of the bank data to predict a person's balance based on their job, marital status, education, and job. In the box plots below you can see that there is a consistent median between each of the categories but there is also a drastic range which will be a problem later on.

```
  #plot data
  install.packages("ggplot2", repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/Diego/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

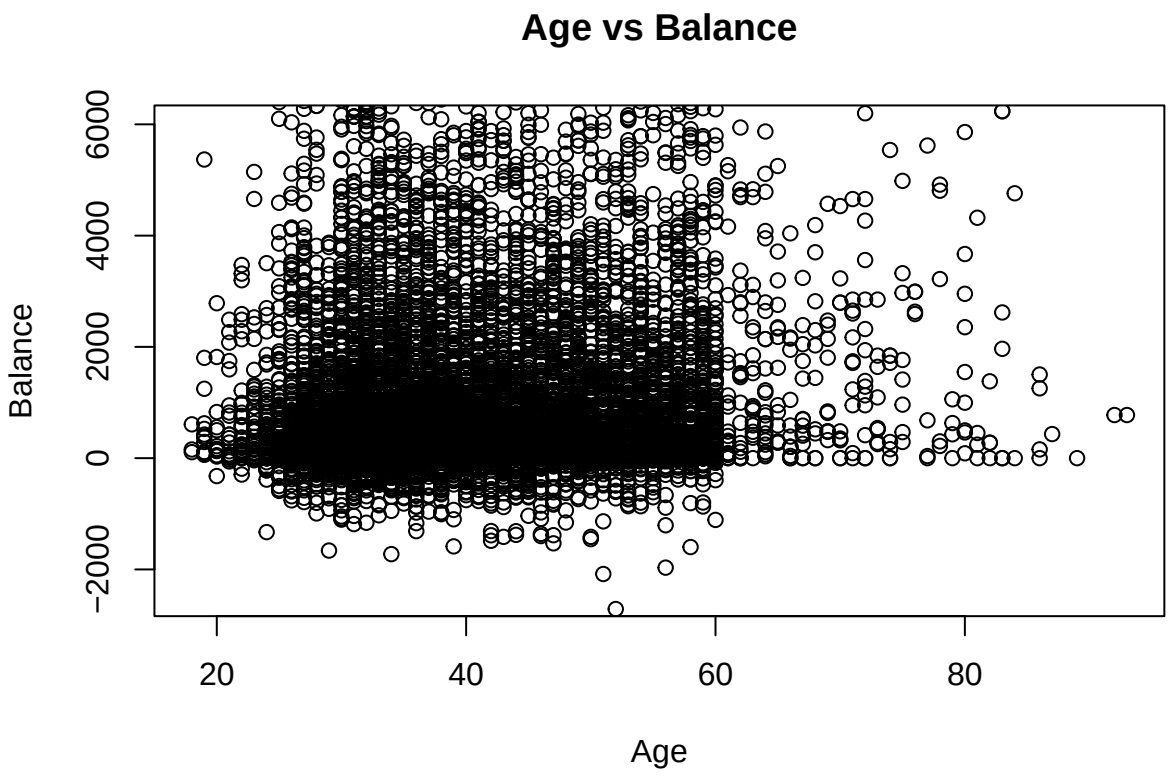
```
## package 'ggplot2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\Diego\AppData\Local\Temp\RtmpWonfZJ\downloaded_packages
```

```
  library("ggplot2")
```

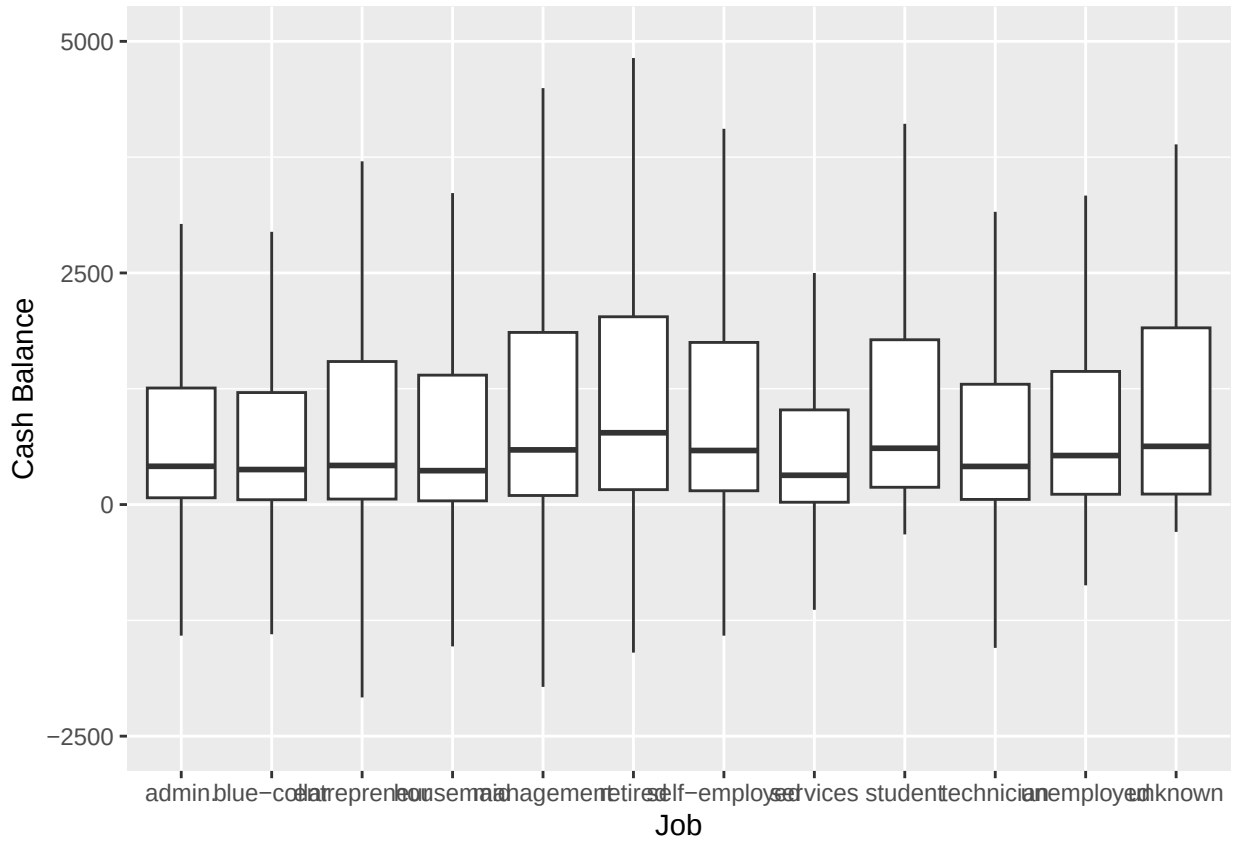
```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
  plot(train$age, train$balance, xlab = "Age", ylab = "Balance", main = "Age vs Balance")
```

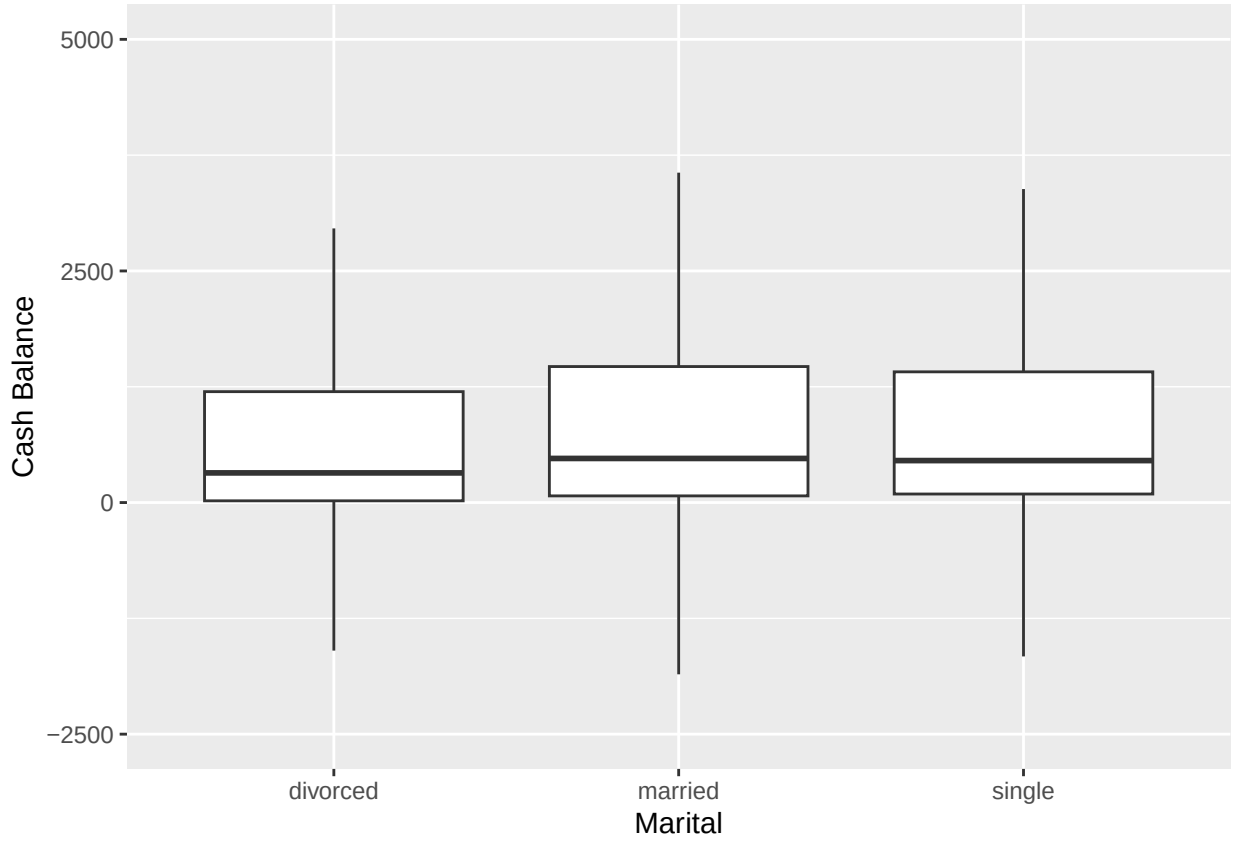

```
plot(train$age, train$balance, xlab = "Age", ylab = "Balance", main = "Age vs Balance", ylim = c(-250
```



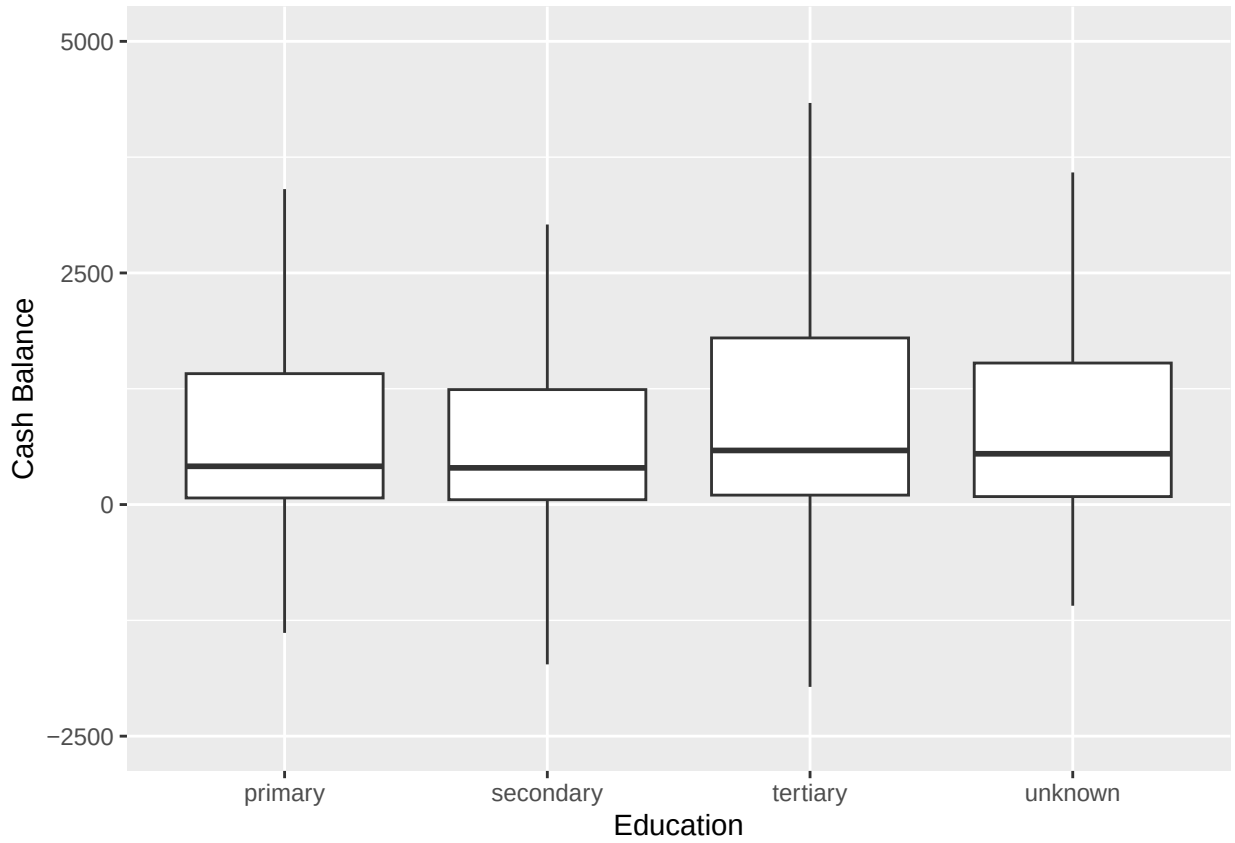

```
ggplot(bank, aes(x = job, y = balance)) + geom_boxplot(outlier.shape = NA) + labs(x = "Job", y = "Cas
```



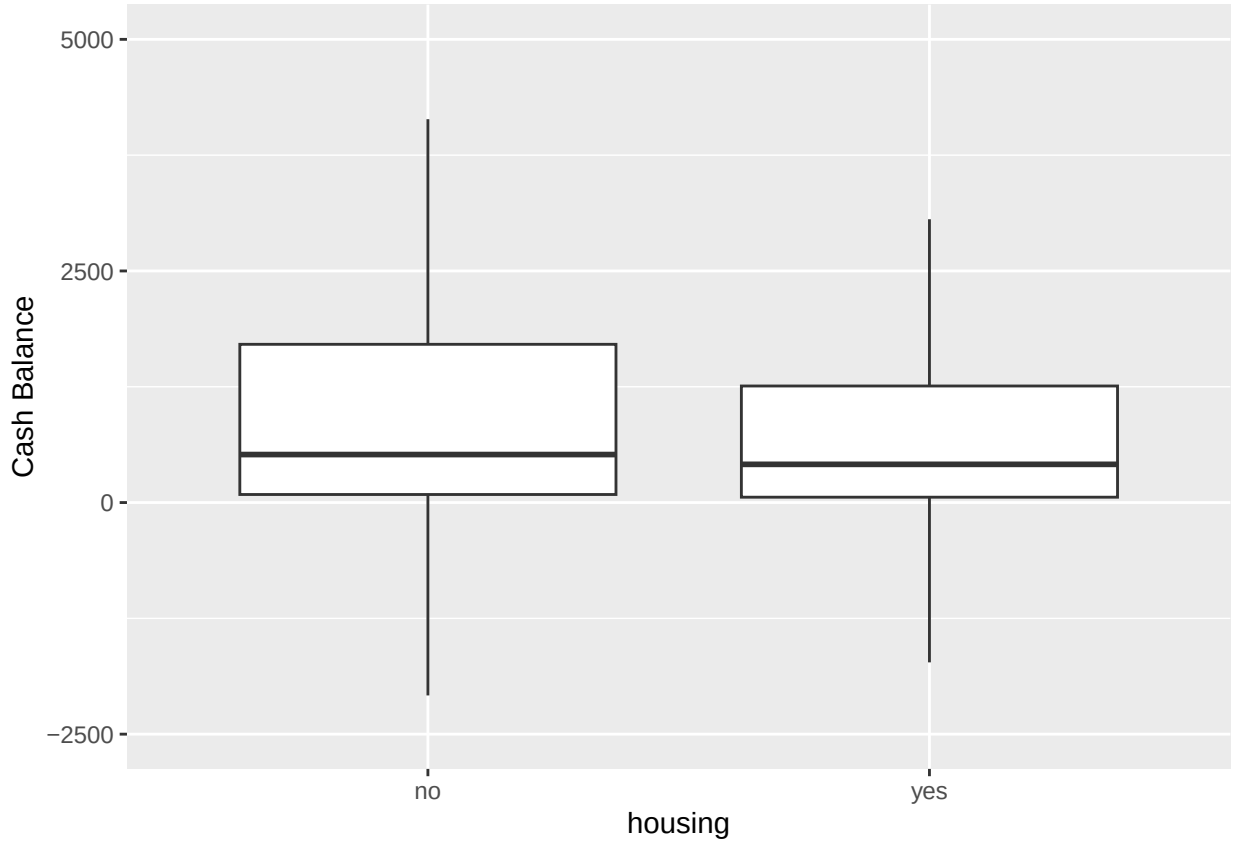

```
ggplot(bank, aes(x = marital, y = balance)) + geom_boxplot(outlier.shape = NA) + labs(x = "Marital",
```




```
ggplot(bank, aes(x = education, y = balance)) + geom_boxplot(outlier.shape = NA) + labs(x = "Educatio
```




```
ggplot(bank, aes(x = housing, y = balance)) + geom_boxplot(outlier.shape = NA) + labs(x = "housing", y
```



Linear Regression

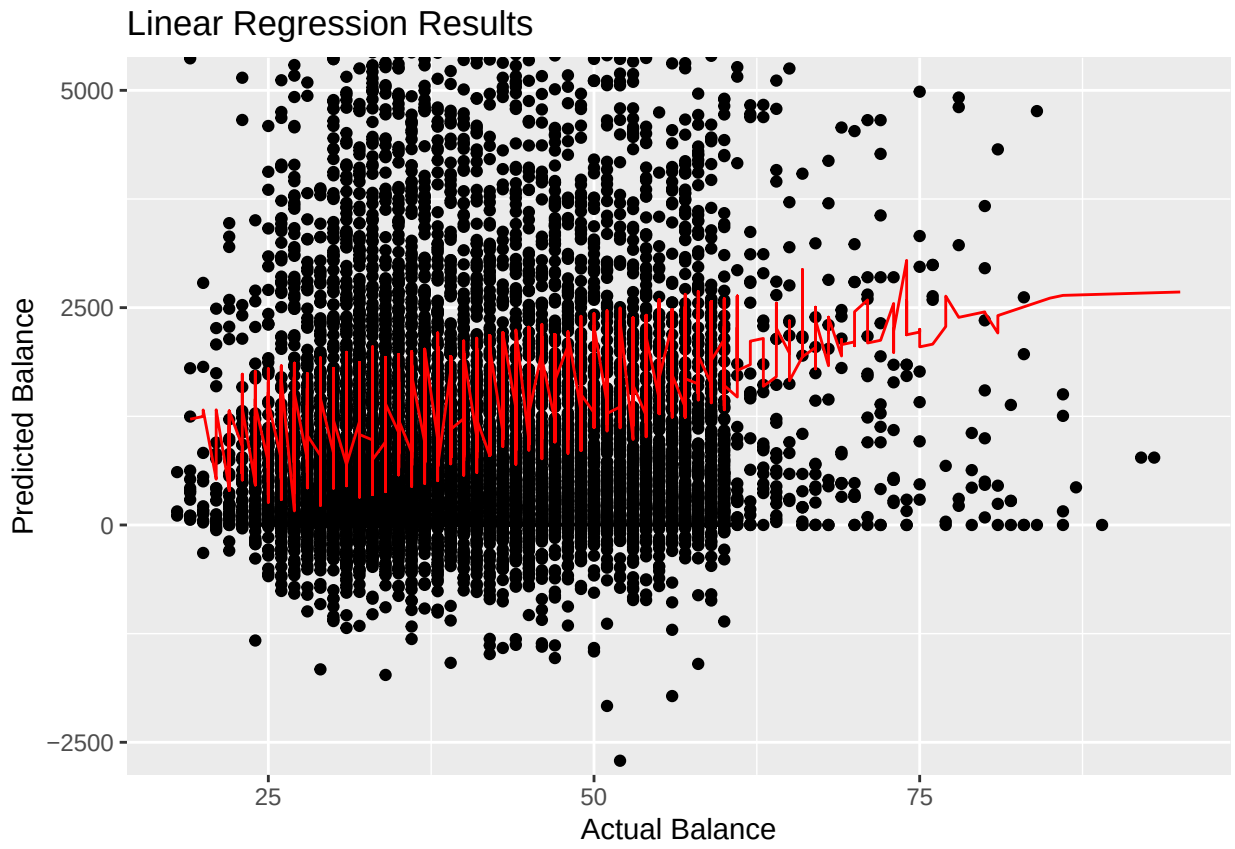
In our linear regression model we can see that it resulted in a very low correlation and a very high mean squared error. The summary shows that most of our variables were not good predictors except for age. Certain sub categories however were significantly better than others which most likely means that those sub categories are have a higher certainty of a person's balance. A real world explanation for this would be that other life styles are unpredictable in terms of wealth. Because of the wide ranges of outliers in each sub category the linear regression model is having a harder time finding correlations between the data.

```
lm <- lm(formula = balance ~ poly(age) + job + marital + education + housing, data = train)
lmpred <- predict(lm, newdata = test)
lmcor <- cor(lmpred, test$balance)
lmmse <- mean((lmpred - test$balance)^2)

predictions <- data.frame(age = test$age, predicted_balance = predict(lm, newdata = test))

sprintf("Correlation:%#.4f   MSE:%#.4f", lmcor, lmmse)
```

```
## [1] "Correlation:0.1913   MSE:8946571.0236"
```

```
ggplot(train, aes(x = train$age, y = train$balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Actual Balance", y = "Predicted Balance", title = "Linear Regression Results")
```


```
ggplot(train, aes(x = train$age, y = train$balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Actual Balance", y = "Predicted Balance", title = "Linear Regression Results") +
  coord_cartesian(ylim=c(-2500, 5000))
```




```
summary(lm)
```

```
##
## Call:
## lm(formula = balance ~ poly(age) + job + marital + education +
##     housing, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -9102  -1254   -731    113  99662
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1072.311    150.450   7.127 1.09e-12 ***
## poly(age)          35406.490   3914.321   9.045  < 2e-16 ***
## jobblue-collar       -70.846    110.603  -0.641 0.521834
## jobentrepreneur      186.282    180.560   1.032 0.302238
## jobhousemaid        -111.155    203.004  -0.548 0.584011
## jobmanagement        362.061    123.146   2.940 0.003288 **
## jobretired          -102.374    169.443  -0.604 0.545737
## jobself-employed     245.364    178.927   1.371 0.170306
## jobservices         -233.038    129.202  -1.804 0.071309 .
## jobstudent           516.477    225.965   2.286 0.022293 *
## jobtechnician          1.554    109.616   0.014 0.988688
## jobunemployed        173.593    189.695   0.915 0.360150
## jobunknown           203.072    375.346   0.541 0.588500
```

```
## maritalmarried       245.078     92.740   2.643 0.008238 **
## maritalsingle        392.112    107.312   3.654 0.000259 ***
## educationsecondary   -74.290     92.860  -0.800 0.423715
## educationtertiary    317.409    116.437   2.726 0.006420 **
## educationunknown     -29.689    167.297  -0.177 0.859149
## housingyes          -166.356     61.167  -2.720 0.006544 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3047 on 11283 degrees of freedom
## Multiple R-squared:  0.02257,    Adjusted R-squared:  0.02101
## F-statistic: 14.48 on 18 and 11283 DF,  p-value: < 2.2e-16
```

NN Regression

The KNN regression model is largely similar to that of the linear regression model in terms of results, still bad but are slightly better. The linear regression model did find some sense of correlation in the data, albeit rather low, however it seems as though the KNN model was able to be far more precise in the age ranges. As you can see in both graphs the linear regression model had a line that tried to mimic the wide ranges of the original data and the KNN regression model does the same but is able to closely copy the sporadic ranges at each age interval. Because KNN relies on the distance between points it must have had an easier time finding and using the medians of each variable thus making it better than linear regression.

```
install.packages("caret", repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/Diego/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'caret' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\Diego\AppData\Local\Temp\RtmpWonfZJ\downloaded_packages
```

```
library(caret)
```

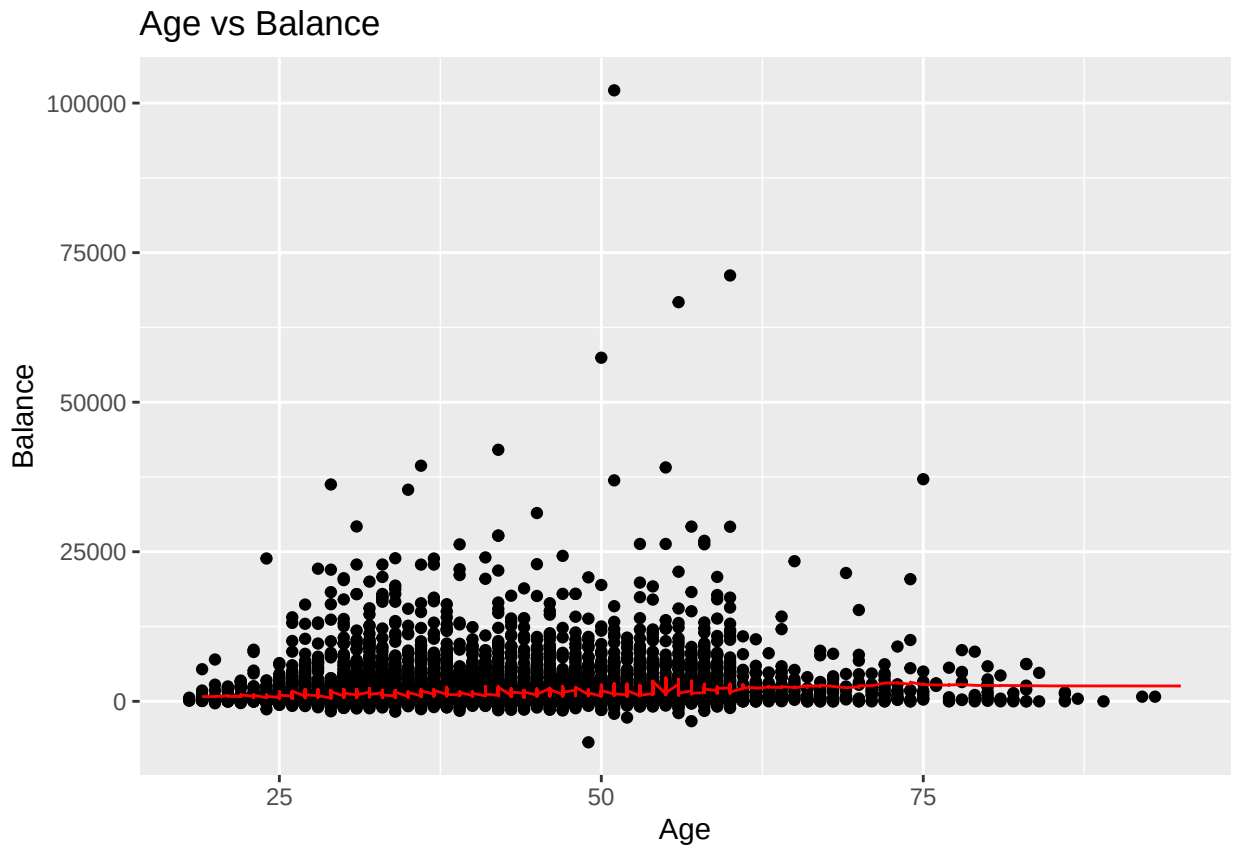
```
## Warning: package 'caret' was built under R version 4.2.3
```

```
## Loading required package: lattice
```

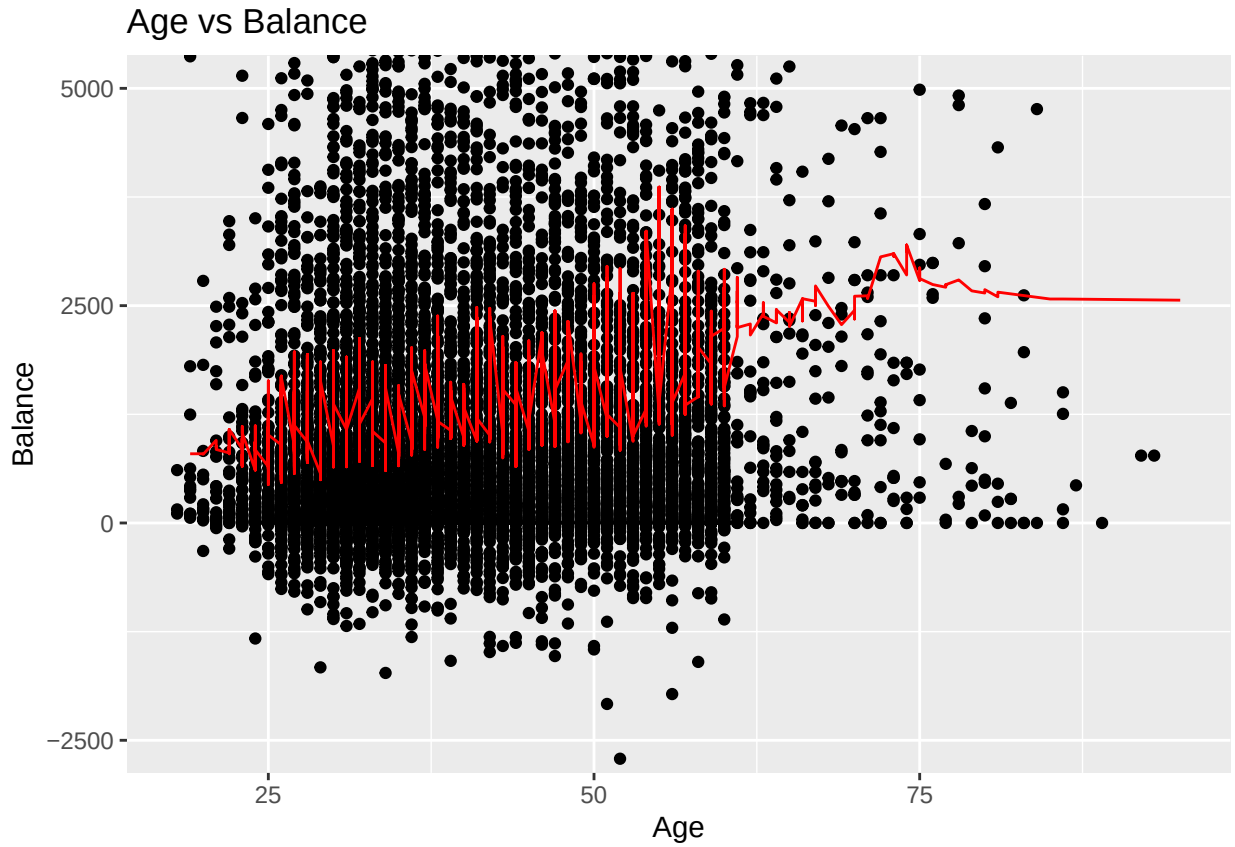
```
knn <- knnreg(formula = balance ~ age + job + marital + education + housing, data = train, k=89)
knnpred <- predict(knn, test)
knncor <- cor(knnpred, test$balance)
knnmse <- mean((knnpred - test$balance)^2)

sprintf("Correlation:%#.4f   MSE:%#.4f", knncor, knnmse)
```

```
## [1] "Correlation:0.1813   MSE:8968043.4738"
```

```
predictions <- data.frame(age = test$age, predicted_balance = knnpred)

ggplot(train, aes(x = age, y = balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Age", y = "Balance", title = "Age vs Balance")
```




```
ggplot(train, aes(x = age, y = balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Age", y = "Balance", title = "Age vs Balance") +
  coord_cartesian(ylim=c(-2500, 5000))
```



Decision Tree Regression

Compared to the other two models the decision tree regression is the worst as it can not be properly executed. Due to how wide spread yet close the data is it is not able it create any divisions making the model unusable. Decision tree's are very reliant on grouping but because of the nature of this data set it is unable to create any groupings. Reducing the data might work however that would seem to be ignoring the problem rather than solve it as the balance range would still be high.

```
install.packages("tree", repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/Diego/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## package 'tree' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'tree'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\Diego\AppData\Local\R\win-library\4.2\00LOCK\tree\libs\x64\tree.dll to
## C:\Users\Diego\AppData\Local\R\win-library\4.2\tree\libs\x64\tree.dll:
## Permission denied

## Warning: restored 'tree'

##
## The downloaded binary packages are in
##  C:\Users\Diego\AppData\Local\Temp\RtmpWonfZJ\downloaded_packages
```

```
library(tree)
```

```
## Warning: package 'tree' was built under R version 4.2.3
```

```
dtree <- tree(formula = balance ~ age + job + marital + education + housing, data = train, mincut = 5)
```

```
## Warning in tree(formula = balance ~ age + job + marital + education + housing,
## : NAs introduced by coercion
```

```
dtpred <- predict(dtree, test)
```

```
## Warning in pred1.tree(object, tree.matrix(newdata)): NAs introduced by coercion
```

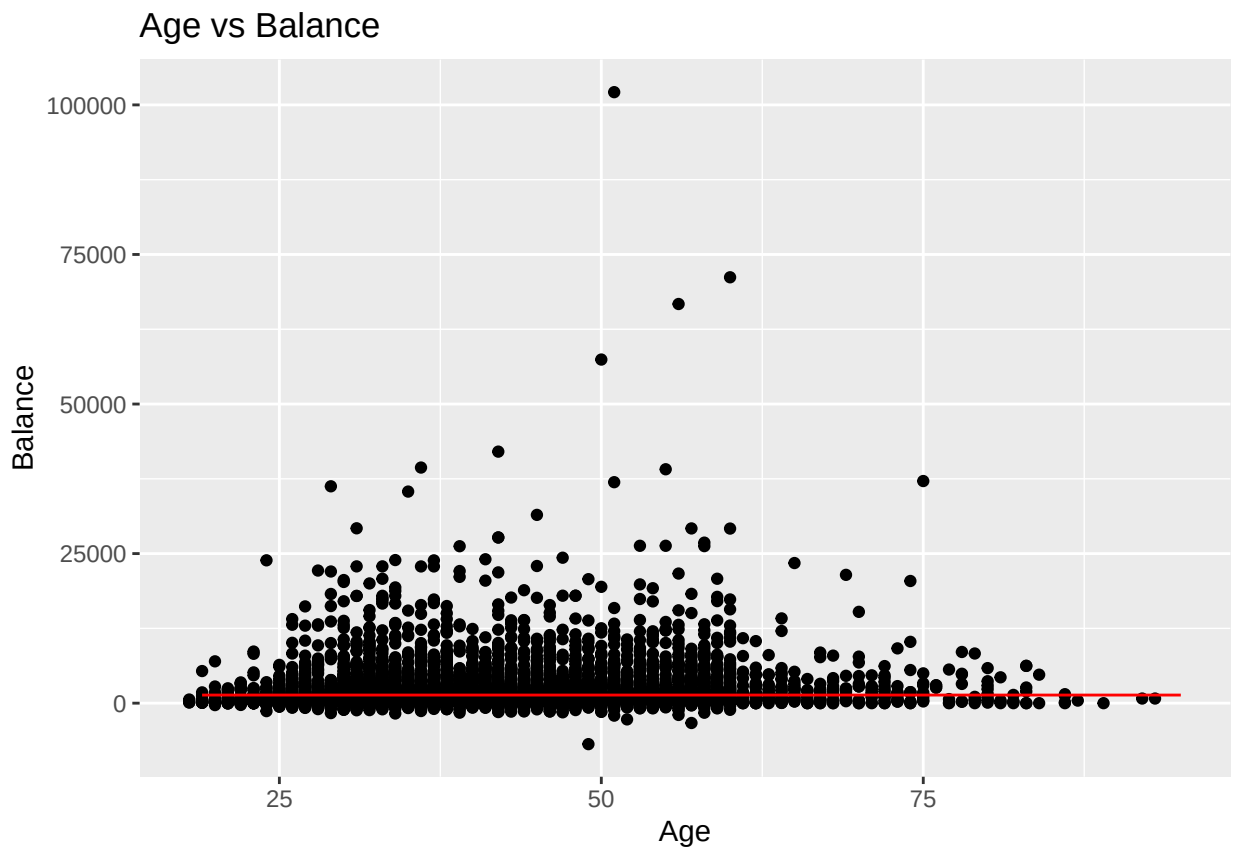
```
dtcor <- cor(dtpred, test$balance)
```

```
## Warning in cor(dtpred, test$balance): the standard deviation is zero
```

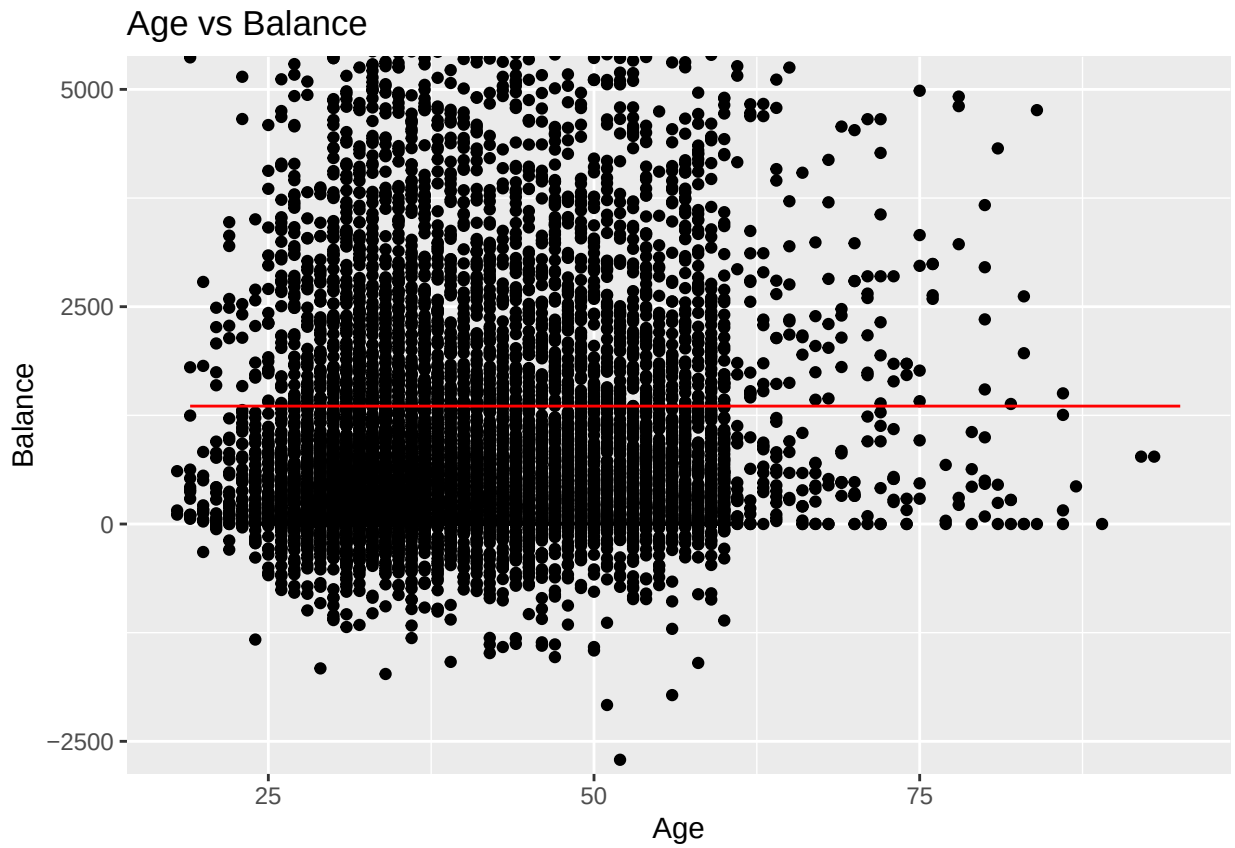
```
dtmse <- mean((dtpred - test$balance)^2)

sprintf("Correlation:%#.4f   MSE:%#.4f", dtcor, dtmse)
```

```
## [1] "Correlation:NA   MSE:9272544.7327"
```

```
predictions <- data.frame(age = test$age, job = test$job,predicted_balance = dtpred)

ggplot(train, aes(x = age, y = balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Age", y = "Balance", title = "Age vs Balance")
```



```r
ggplot(train, aes(x = age, y = balance)) +
  geom_point() +
  geom_line(data = predictions, aes(x = age, y = predicted_balance), color = "red") +
  labs(x = "Age", y = "Balance", title = "Age vs Balance") +
  coord_cartesian(ylim=c(-2500, 5000))
```



Conclusion The regression models preformed poorly mostly due to the data set having such large ranges in the predicted value. The linear and KNN regression models both understood the general direction that each graph was trying to go and tried to copy the data's wild changes but both were unable to do so accurately. The failure of the regression models is most certainly attributed do to people's balances being far too unpredictable.