Regression on the Kaggle Dataset Medals Data Set

Sam Ofiaza

The Kaggle Dataset Medals data set contains information on datasets including the medal it has received (if any) and how many votes, views, downloads, etc. it has.

Credit to Niek van der Zwaag for the data set (link (https://www.kaggle.com/datasets/niekvanderzwaag/kaggle-dataset-medals?select=dataset_medal_total.csv)).

Linear Regression Overview

Linear regression is a parametric model that predicts quantitative values from parameters. It estimates a line of best fit for the data by estimating a slope and intercept. The line of best fit does not have to be straight - polynomial, logistic, etc. lines are also possible. Linear regression is best used when the data has a Gaussian distribution and when both the predictor and target variables are both continuous. Linear regression's pitfalls are that it will underfit the data and that interaction between predictors, variables that correlate with both predictors and the target, and hidden variables will interfere with the model.

```
df <- read.csv("./dataset_medal_total.csv")
df <- df[, c(2, 5, 6, 7, 8)]
df$Medal <- as.factor(df$Medal)
str(df)
```

```
## 'data.frame':    42955 obs. of  5 variables:
##  $ Medal         : Factor w/ 4 levels "Bronze","Gold",..: 4 2 4 2 1 2 1 1 1 4 ...
##  $ Views         : int  69099 233338 63116 335275 18782 274664 18768 14891 25631 41586 ...
##  $ Votes         : int  96 513 254 893 58 688 41 31 36 82 ...
##  $ Votes_Advanced: int  29 90 47 158 16 135 6 11 6 21 ...
##  $ Downloads     : int  4466 25261 9294 60607 1895 40130 1903 1761 1415 2853 ...
```

```
sapply(df, function(x) sum(is.na(x)==TRUE))
```

```
##          Medal          Views          Votes Votes_Advanced      Downloads
##              0              0              0              0              0
```

Divide into 80/20 train/test

```
set.seed(1234)
i <- sample(1:nrow(df), nrow(df)*0.8, replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

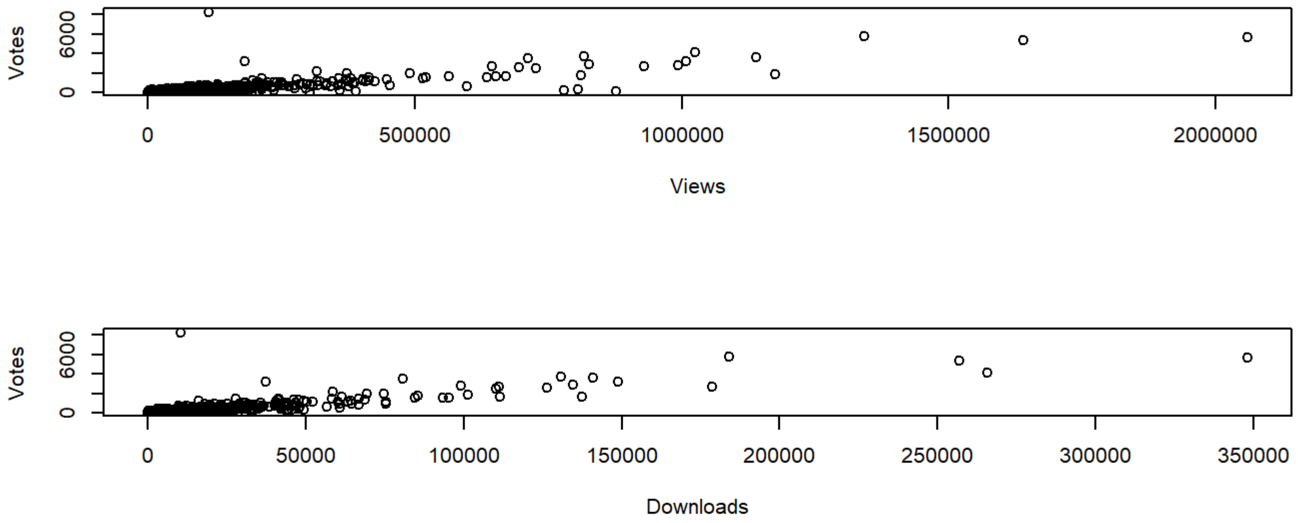
Data Summary

```
str(df)
```

```
## 'data.frame':    42955 obs. of  5 variables:
##  $ Medal         : Factor w/ 4 levels "Bronze","Gold",..: 4 2 4 2 1 2 1 1 1 4 ...
##  $ Views         : int  69099 233338 63116 335275 18782 274664 18768 14891 25631 41586 ...
##  $ Votes         : int  96 513 254 893 58 688 41 31 36 82 ...
##  $ Votes_Advanced: int  29 90 47 158 16 135 6 11 6 21 ...
##  $ Downloads     : int  4466 25261 9294 60607 1895 40130 1903 1761 1415 2853 ...
```

Data Visualization

```
par(mfrow=c(3,1))
plot(train$Views, train$Votes, xlab="Views", ylab="Votes")
plot(train$Downloads, train$Votes, xlab="Downloads", ylab="Votes")
```



Linear Model 1: Votes vs Views

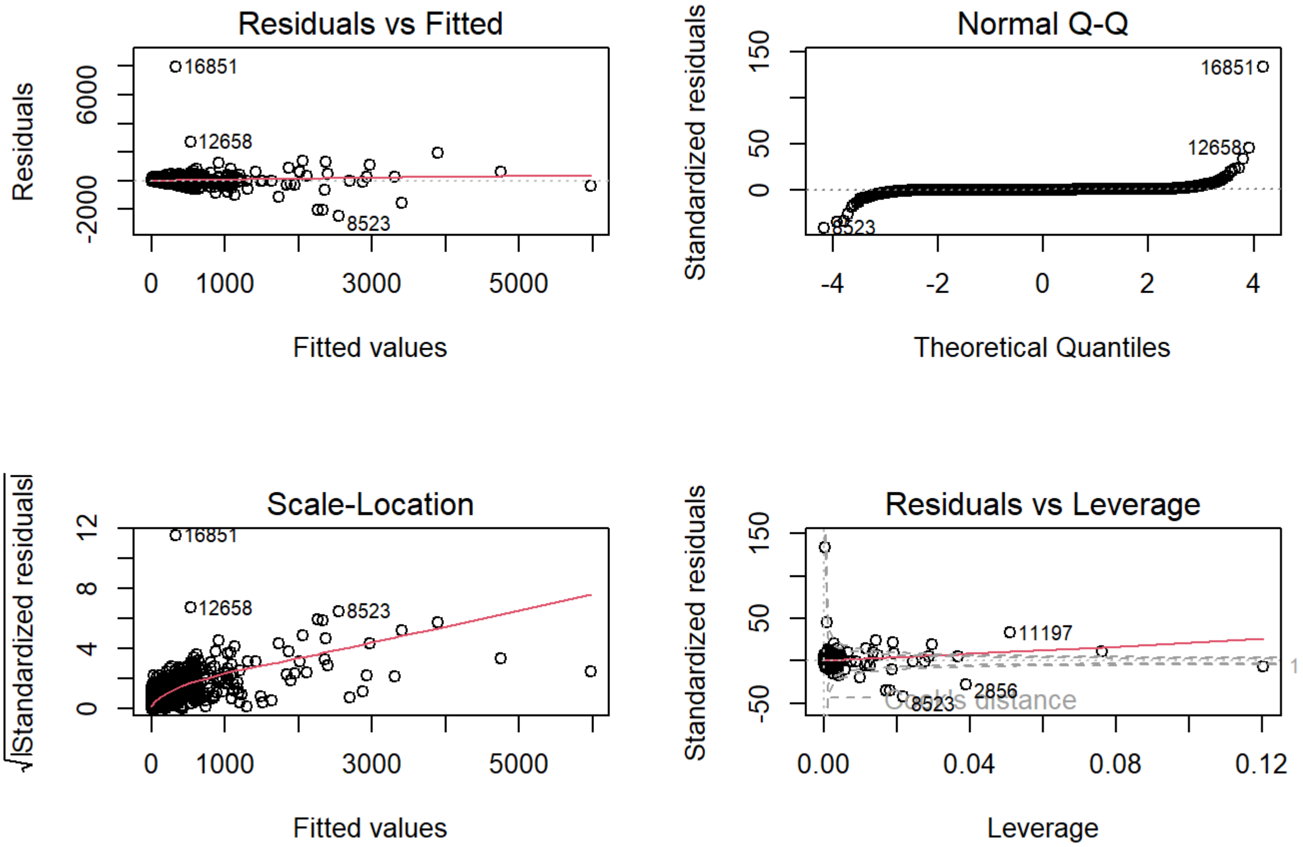
```
lm1 <- lm(Votes~Views, data=train)
summary(lm1)
```

```
## 
## Call:
## lm(formula = Votes ~ Views, data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -2479.8    -2.3    -1.2     0.8  7959.8 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.949e+00  3.248e-01   6.002 1.97e-09 ***
## Views       2.898e-03  1.005e-05 288.239  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 59.61 on 34362 degrees of freedom
## Multiple R-squared:  0.7074, Adjusted R-squared:  0.7074 
## F-statistic: 8.308e+04 on 1 and 34362 DF,  p-value: < 2.2e-16
```

Model Summary Analysis

The model estimated the intercept to be 1.949 and the slope to be 2.898e-3 with an extremely low p-value, meaning that these estimates are most likely accurate. The residual standard error was 59.61, so the model was off by an average of 59.61 votes per entry. A good R-squared value is close to -1 or 1, so a value of 0.7074 is not bad. Because the p-value of the overall model is so small, it is safe to reject the null hypothesis and say that the results are statistically significant.

```
par(mfrow=c(2,2))
plot(lm1)
```



Residual Plots

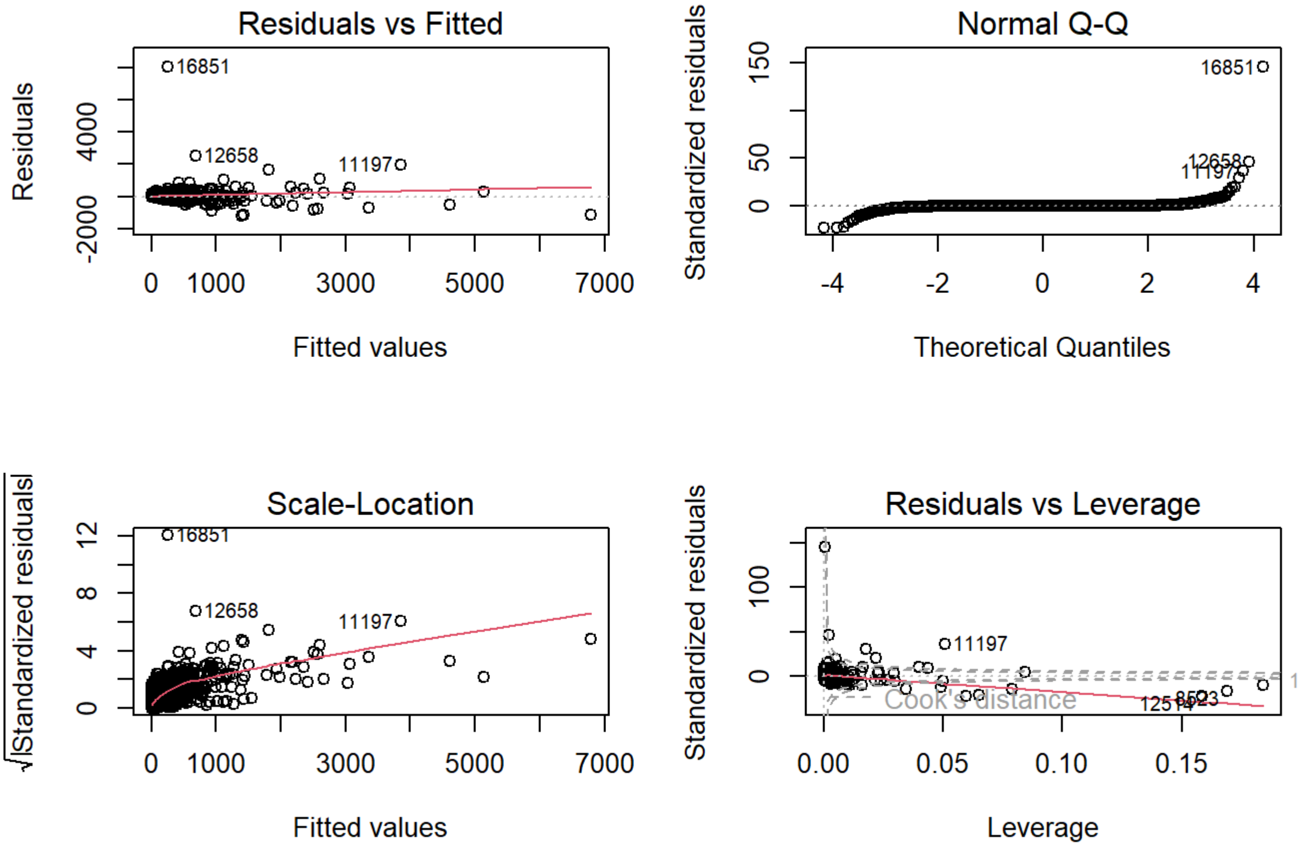
The Residuals vs Fitted plot seems to indicate linearity but there is a clear density on the left so it is possible that some improvement can be made with the model. The Normal Q-Q plot indicates a decently strong normal distribution with some variation at the ends. Data point 16851 seems to be an outlier. The Scale-Location plot does not indicate equal variance in the model as the points are not equally spread and the line is not exactly horizontal. The Residuals vs Leverage plot points out some influential data points that are seen as clearly outside the dotted grey lines, such as 11197, 2856, and 8523. If we excluded these points, the results will be significantly altered.

Linear Model 2: Votes vs Views+Downloads

```
lm2 <- lm(Votes~Views+Downloads, data=train)
summary(lm2)
```

```
##
## Call:
## lm(formula = Votes ~ Views + Downloads, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1212.9    -2.5    -1.8     0.4  8031.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.720e+00  3.009e-01    9.04   <2e-16 ***
## Views       1.037e-03  2.629e-05   39.45   <2e-16 ***
## Downloads   1.335e-02  1.765e-04   75.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.19 on 34361 degrees of freedom
## Multiple R-squared:  0.7492, Adjusted R-squared:  0.7492
## F-statistic: 5.133e+04 on 2 and 34361 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm2)
```

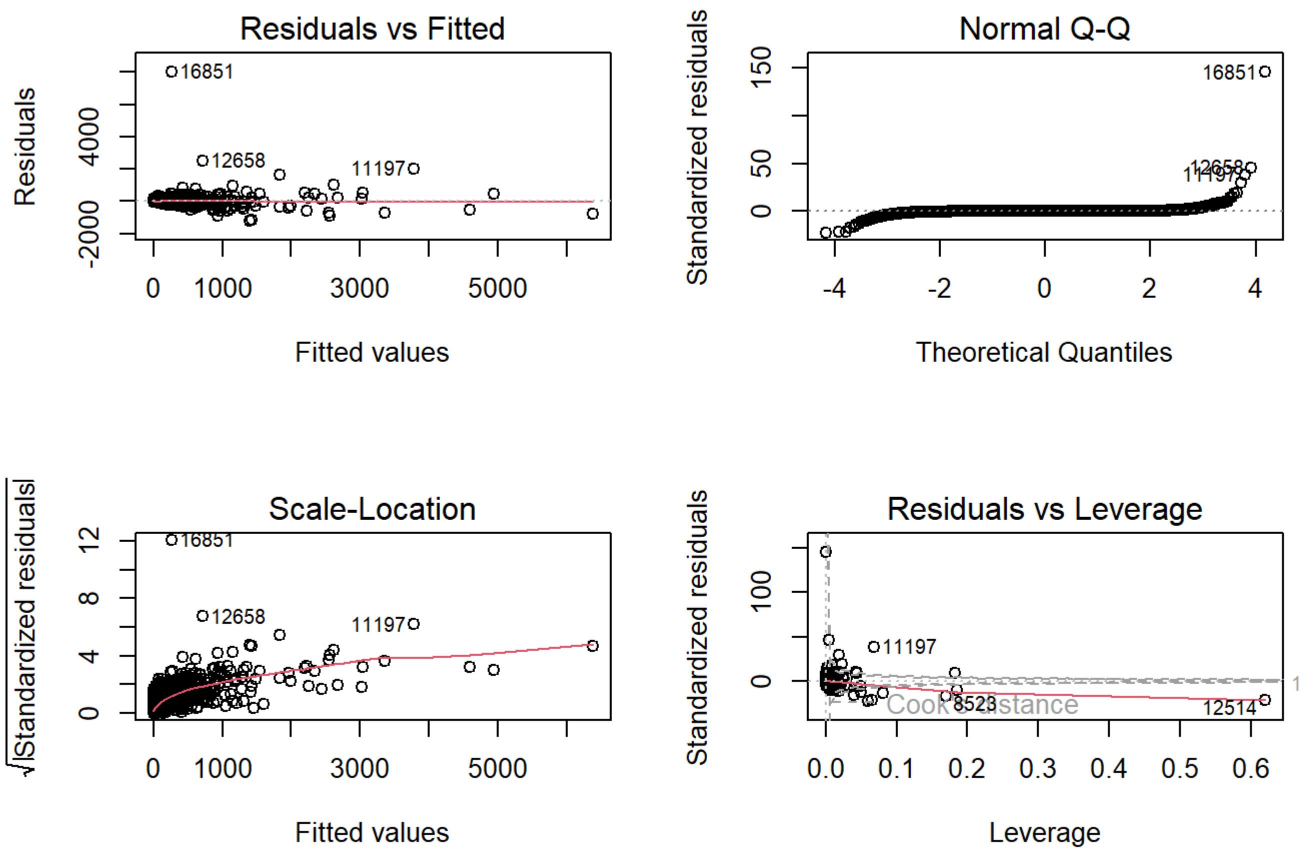


Linear Model 3: Votes vs Views*Downloads

```
lm3 <- lm(Votes~Views*Downloads, data=train)
summary(lm3)
```

```
##
## Call:
## lm(formula = Votes ~ Views * Downloads, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1216.6    -2.1    -1.4     0.7  8022.6
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.298e+00  3.028e-01   7.591 3.24e-14 ***
## Views            1.030e-03  2.625e-05  39.237  < 2e-16 ***
## Downloads        1.441e-02  1.999e-04  72.055  < 2e-16 ***
## Views:Downloads -1.069e-09  9.611e-11 -11.125  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.09 on 34360 degrees of freedom
## Multiple R-squared:  0.7501, Adjusted R-squared:  0.7501
## F-statistic: 3.438e+04 on 3 and 34360 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm3)
```



Comparing the 3 Models

Model 1 with only Views as a predictor had an R-squared value of 0.7074. Model 2 with Views and Downloads as predictors had an R-squared value of 0.7492. Model 3 with Views, Downloads, and a possible interaction effect between Views and Downloads as predictors had an R-squared value of 0.7501. Given that all 3 models have the same miniscule p-value, Model 3 seems to be the best model based on its R-value.

Calculated Metrics for each Linear Model

```
pred1 = predict(lm1, newdata=test)
cor1 <- cor(pred1, test$Votes)
mse1 <- mean((pred1-test$Votes)^2)
rmse1 <- sqrt(mse1)
print(paste('correlation:', cor1))
```

```
## [1] "correlation: 0.908413134048508"
```

```
print(paste('mse:', mse1))
```

```
## [1] "mse: 1764.31470478177"
```

```
print(paste('rmse:', rmse1))
```

```
## [1] "rmse: 42.0037463184151"
```

```
pred2 = predict(lm2, newdata=test)
cor2 <- cor(pred2, test$Votes)
mse2 <- mean((pred2-test$Votes)^2)
rmse2 <- sqrt(mse2)
print(paste('correlation:', cor2))
```

```
## [1] "correlation: 0.883282732503103"
```

```
print(paste('mse:', mse2))
```

```
## [1] "mse: 2258.0269066085"
```

```
print(paste('rmse:', rmse2))
```

```
## [1] "rmse: 47.5187005989063"
```

```
pred3 = predict(lm3, newdata=test)
cor3 <- cor(pred3, test$Votes)
mse3 <- mean((pred3-test$Votes)^2)
rmse3 <- sqrt(mse3)
print(paste('correlation:', cor3))
```

```
## [1] "correlation: 0.871877879027364"
```

```
print(paste('mse:', mse3))
```

```
## [1] "mse: 2509.20436290527"
```

```
print(paste('rmse:', rmse3))
```

```
## [1] "rmse: 50.0919590643575"
```

Analysis of Prediction Results on Test Data

A correlation value close to either -1 or 1 and a low rmse value is ideal. Out of the 3, Model 1 seems to be the better model based on the the calculated metrics above with the highest correlation and lowest rmse. This indicates that considering Views as the only predictor results in the best predictor for Votes. My best guess on why this happened is that Downloads is not a good predictor for Votes and thus adding Downloads as a predictor in any way caused the model to perform worse.