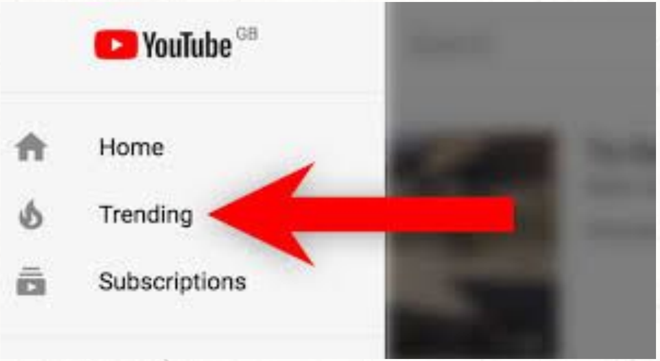
Trending YouTube Video Statistics

Khushboo Harjani / Chia-Han Chiang

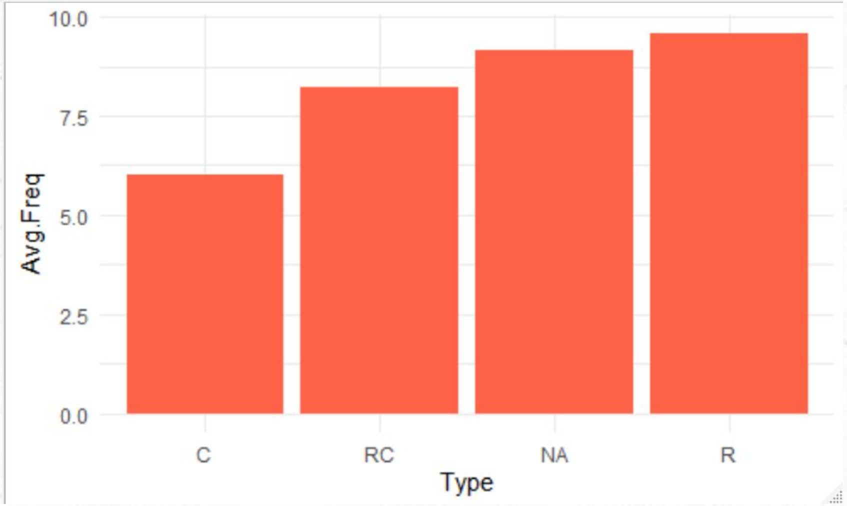
Problem Definition

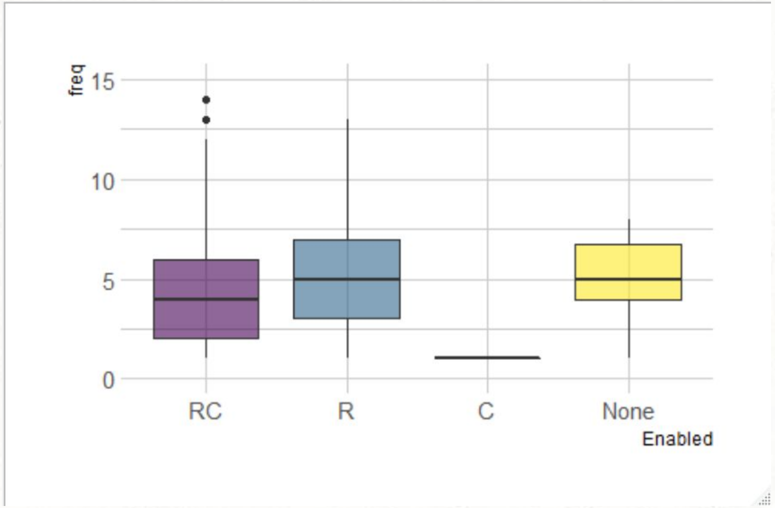
- YouTube channels are paid for marketing purposes, only when they have a large enough fan base.

- Goal: Foresee YouTube video trendiness given that a video has already shown up on the platform’s trending list at least once.



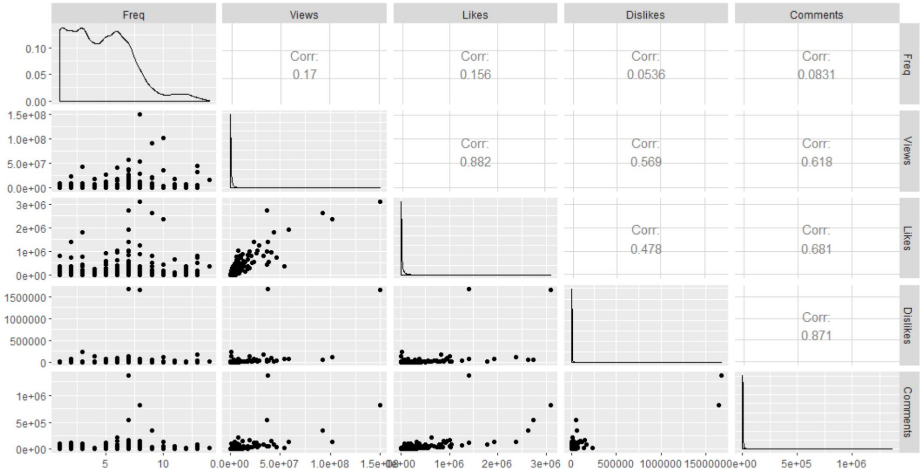
Data Exploration – Enabled Comments Or Ratings



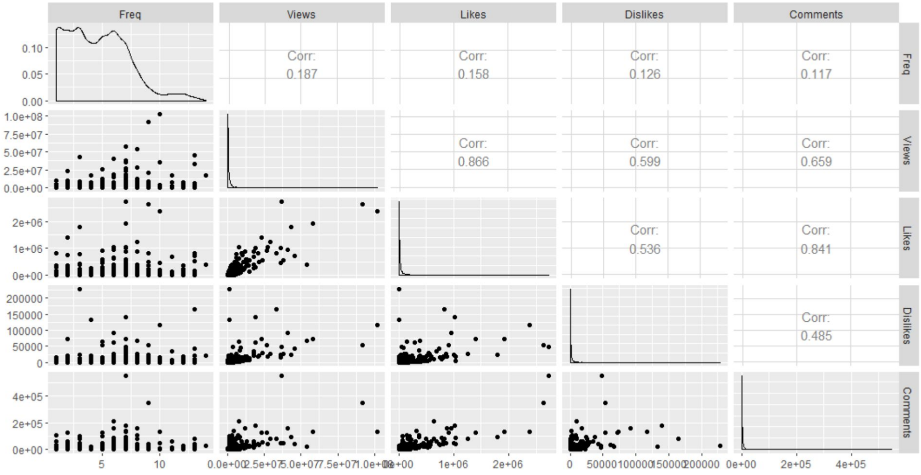


- RC: Rating and Comments / R: Rating / C: Comments / NA: Nothing enabled
- Videos with only ratings have the highest average number of times on the trend list.
- However, videos with “R”, “C”, and “RC” make up less than 2% of the proportion of records

Data Exploration – Scatter Plot



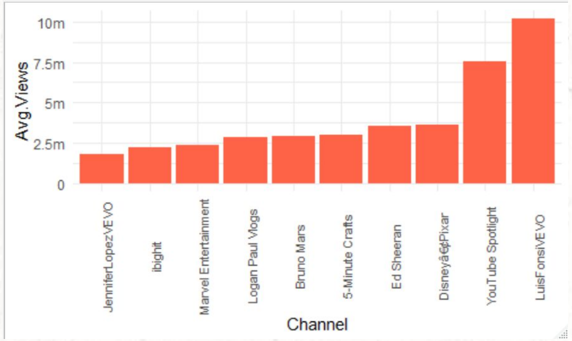
Scatter plot of all training data

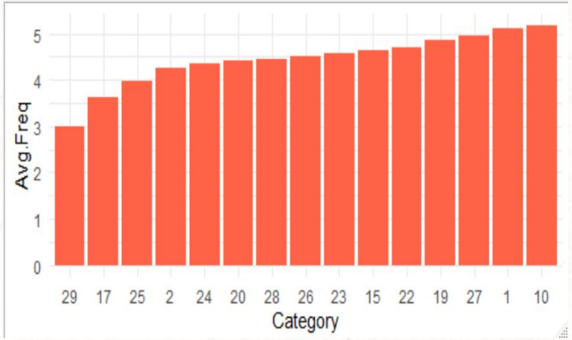


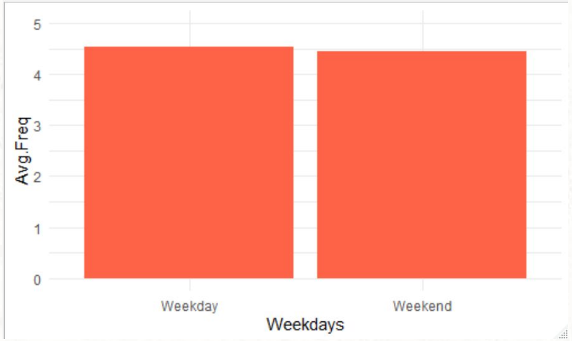
Scatter plot after outliers are removed

- “Likes” have a strong correlation of 0.866 with “Views”
- “Dislikes” and “Comments” have correlations with “Views”, of 0.599 and 0.659 respectively
- There is a very small correlation between any of the attributes and the response variable

Data Exploration – Category, Channel, Weekday







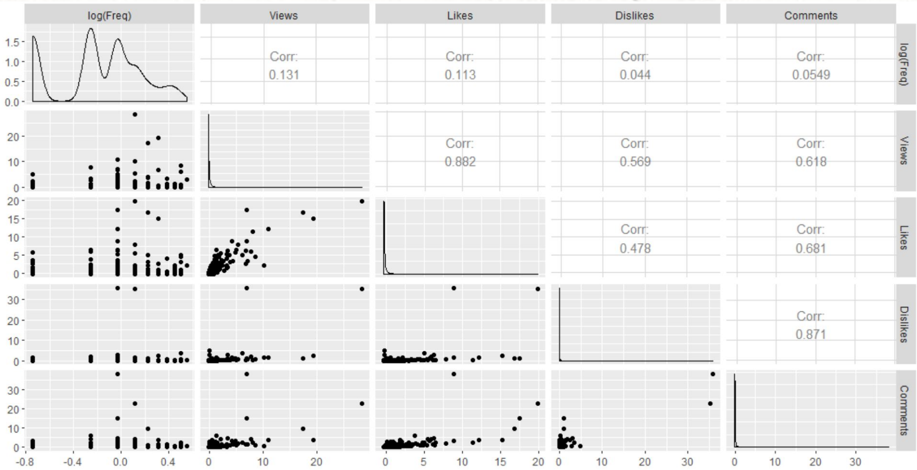
- **Popularity amongst channels in terms of average video views**
- **Videos evenly distributed across categories (Trendiness level ranges from 3 to 5)**
- **Category 29 has the least frequent trendiness**
- **The average frequency of a video on the trend list is the same, whether the videos were published on a weekday or weekend**

Data Mining Tasks

Multiple Linear Regression (Prediction)
K-NN Neighbors (Classification)
Classification trees (Classification)

Multiple Linear regression

- The log of each variable is taken to create the following new scatterplot
- No apparent linear, polynomial or exponential trend between any predictor and the response variable “Frequency”
- It is expected that the multiple linear regression model will not perform well
- **Principal Component Analysis (PCA)** is explored to observe if fewer principal component variables can be used as better predictors



Multiple Linear regression (PCA)

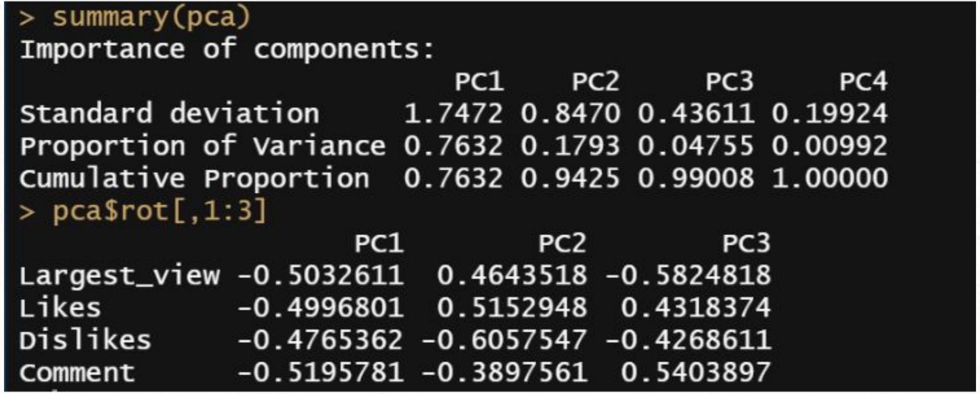
Original Numerical Variables:

- Views
- Likes
- Dislikes
- Comment Count

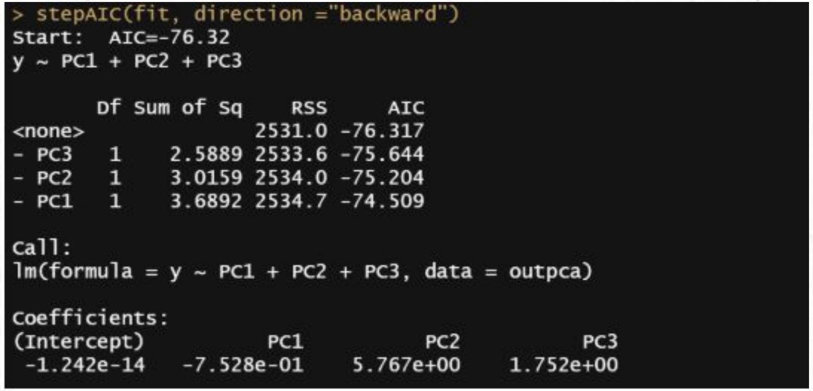
- 99% of Variance is Captured by new Product Components (PC1, PC2, PC3)

The model will be:
y = −1.242e-14−7.528e-01*PC1+5.767e+00*PC2 +1.752e+00*PC3

- MLR of PC's result in a model with RMSE of 0.982
- **Model accuracy is 1.8%** (Extremely weak), lower than Naive benchmark



```
> summary(pca)
Importance of components:
                          PC1    PC2     PC3     PC4
Standard deviation     1.7472 0.8470 0.43611 0.19924
Proportion of Variance 0.7632 0.1793 0.04755 0.00992
Cumulative Proportion  0.7632 0.9425 0.99008 1.00000
> pca$rot[,1:3]
                    PC1        PC2        PC3
Largest_view -0.5032611  0.4643518 -0.5824818
Likes        -0.4996801  0.5152948  0.4318374
Dislikes     -0.4765362 -0.6057547 -0.4268611
Comment      -0.5195781 -0.3897561  0.5403897
```

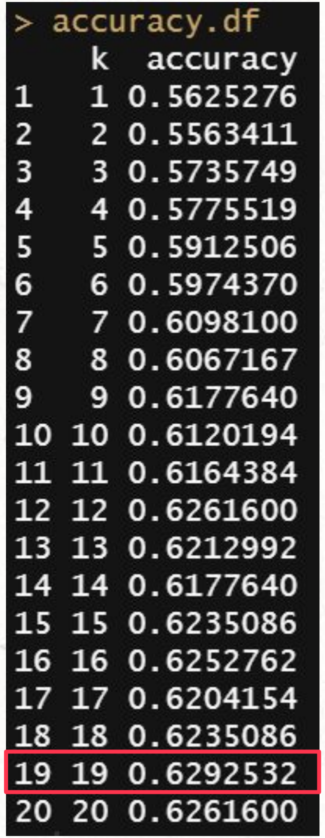


```
> stepAIC(fit, direction ="backward")
Start:  AIC=-76.32
y ~ PC1 + PC2 + PC3

       Df Sum of Sq    RSS     AIC
<none>              2531.0 -76.317
- PC3   1    2.5889 2533.6 -75.644
- PC2   1    3.0159 2534.0 -75.204
- PC1   1    3.6892 2534.7 -74.509

Call:
lm(formula = y ~ PC1 + PC2 + PC3, data = outpca)

Coefficients:
(Intercept)          PC1          PC2          PC3
 -1.242e-14   -7.528e-01    5.767e+00    1.752e+00
```

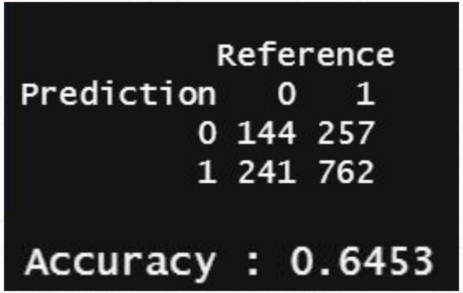
K-NN Neighbors

- K-NN is expected to be better than MLR
- Predictors are highly correlated. There’s a high likelihood for clusters to present in the data
- Converted “category” into a numerical predictor: calculated the average frequency of a video in each category and normalized the average frequencies of each category
- Did not include the variable “Channel”: there are a large number of channels in new data are not present in the training data
- The response variable “Frequency” is classified into two outcomes: **Success** (> Median) and **Failure** (< Median)
- Normalization is done separately for training, validation, and test data

K-NN Neighbors



```
> accuracy.df
    k  accuracy
1    1 0.5625276
2    2 0.5563411
3    3 0.5735749
4    4 0.5775519
5    5 0.5912506
6    6 0.5974370
7    7 0.6098100
8    8 0.6067167
9    9 0.6177640
10  10 0.6120194
11  11 0.6164384
12  12 0.6261600
13  13 0.6212992
14  14 0.6177640
15  15 0.6235086
16  16 0.6252762
17  17 0.6204154
18  18 0.6235086
19  19 0.6292532
20  20 0.6261600
```



```
          Reference
Prediction   0   1
         0 144 257
         1 241 762

 Accuracy : 0.6453
```

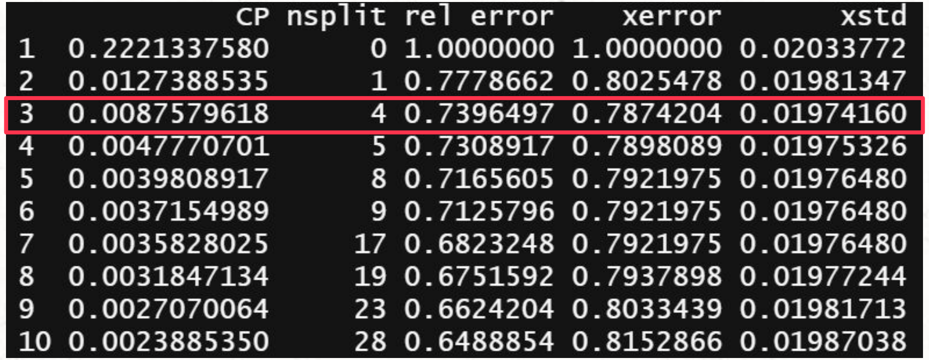
- Test k=19 on the “test data” and calculate the accuracy of the model

- The Accuracy level of the final K-nn model is 64.5%

- K-nn is a relatively good model to use for this dataset.

Classification Tree

- Identify single variable (bin) associations with higher frequency (trendiness)
- The smallest xerror is #3, which has 0.7874, and has 4 splits for a best pruned tree

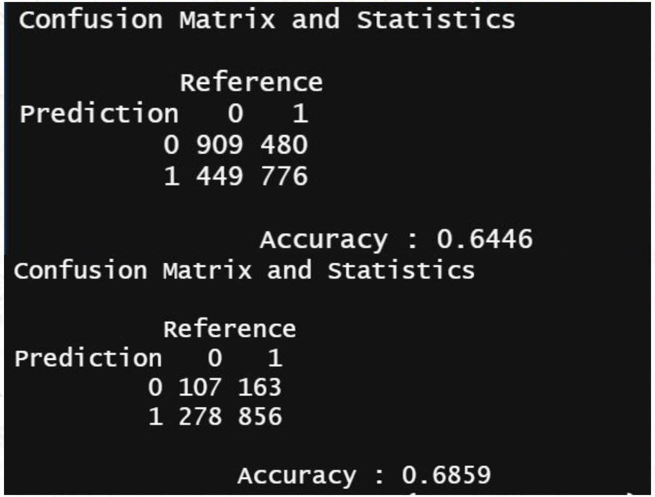


```
           CP nsplit rel error    xerror       xstd
1  0.2221337580      0 1.0000000 1.0000000 0.02033772
2  0.0127388535      1 0.7778662 0.8025478 0.01981347
3  0.0087579618      4 0.7396497 0.7874204 0.01974160
4  0.0047770701      5 0.7308917 0.7898089 0.01975326
5  0.0039808917      8 0.7165605 0.7921975 0.01976480
6  0.0037154989      9 0.7125796 0.7921975 0.01976480
7  0.0035828025     17 0.6823248 0.7921975 0.01976480
8  0.0031847134     19 0.6751592 0.7937898 0.01977244
9  0.0027070064     23 0.6624204 0.8033439 0.01981713
10 0.0023885350     28 0.6488854 0.8152866 0.01987038
```

- The response variable “Frequency” is classified into two outcomes: **Success** (> Median) and **Failure** (< Median)
- Tested on the test data to ensure that it is not overfit to the training data

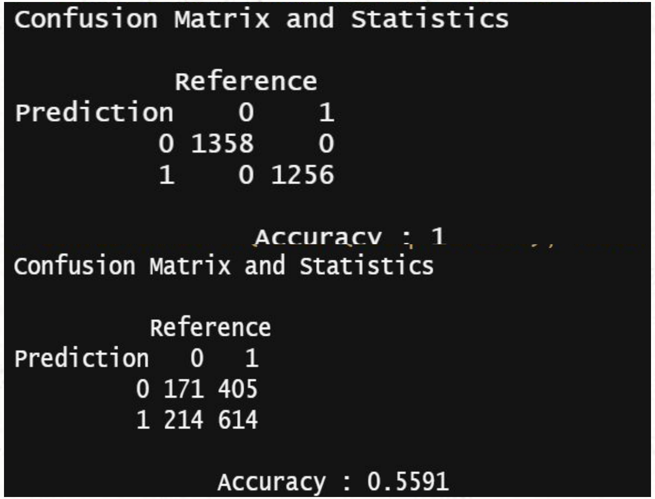
Classification Tree

Best-pruned tree



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 909 480
         1 449 776

               Accuracy : 0.6446
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 107 163
         1 278 856

               Accuracy : 0.6859
```

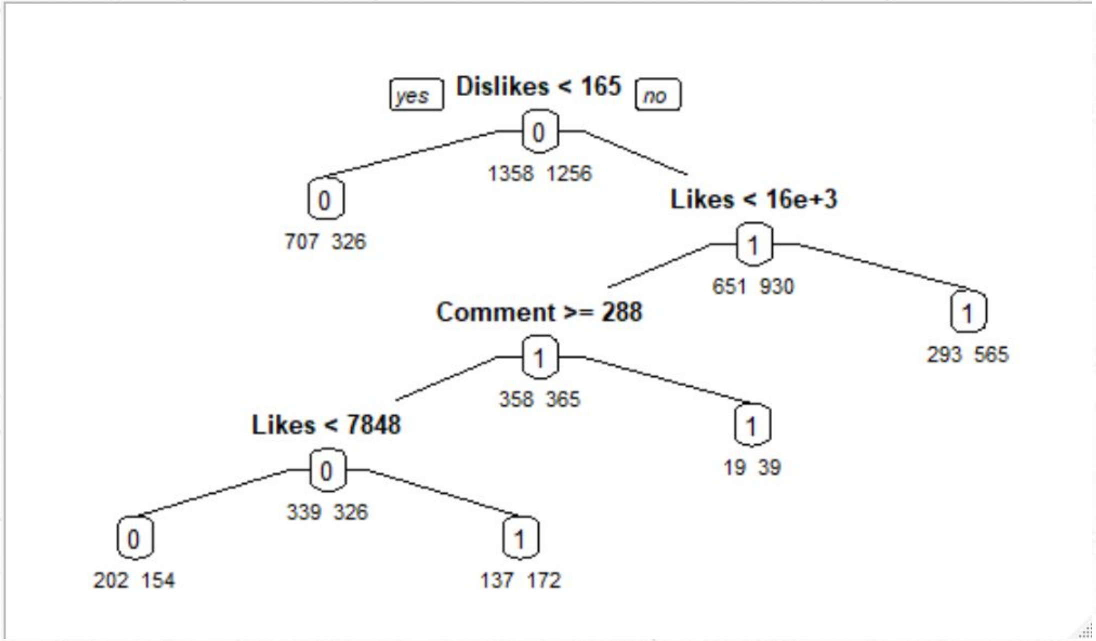
Deeper tree



```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1358    0
         1    0 1256

               Accuracy : 1
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 171 405
         1 214 614

               Accuracy : 0.5591
```

The chosen tree is the default tree because it captures the trend rather than the noise.

Classification Tree



- Classifications of the validation data have an **accuracy of 68.6%**
- **This is the best model to predict trending videos that with frequency >5**

Result

- Multiple Linear Regression (PCA): **1.8%**
- K-NN neighbors: **64.5%**
- Classification trees: **68.6%**

- Highest predictive performance from the three models: **Classification trees → 68.6%**

- Higher than the naive benchmark (50%) and would therefore create a lift given predictor information.

Thank you