# Predicting Party Affiliation Based on Twitter Engagement Numbers

Eyob Tadele

December 12, 2021

## Abstract

Predicting political party affiliation from Twitter engagement numbers such as replies, re-tweets, and favorites is an interesting problem to explore. Through the application of clustering and logistic regression, we can gain insight into whether it is possible to make reliable predictions. Interpretations of the results indicates that K-means and glm are not particularly useful in predicting group affiliation.

## Problem Definition

Twitter is one of the foremost social media platforms across the world. It has become the primary source for information, entertainment, and communication. With its widespread user base and availability, shaping opinions and associating with groups based on various interests come naturally. This poses an interesting question on whether we can predict some kind of behavioral pattern or group to which users belong to by analyzing twitter use habits.

For this specific problem, the motivation is to find out whether Twitter accounts of US politicians (i.e. Senators, Presidents) and their engagement numbers can be useful in predicting their political party affiliations. The main engagement metrics applied here will be reply, re-tweet, and favorite numbers.

## Data

The dataset to be used for this project was originally obtained from FiveThirtyEight website's github repository (https://github.com/fivethirtyeight/data/tree/master/twitter-ratio) and re-posted on Kaggle. It consists of tweets from US Senators, and two former Presidents between 2008 to 2017. Overall, it consists of over 290,000 rows of data with seven features. Namely, created_at, text tweeted, url, replies, re-tweets, favorites,and user. A label, which will be used as a predictive value has been added. A summary of these features and label are indicated below in [figure 1]. The main features that will be used for this project are number of replies, re-tweets, and favorites.

```
## 'data.frame':    295054 obs. of  8 variables:
##  $ created_at: chr  "2/17/09 16:49" "2/17/09 16:59" "2/17/09 18:24" "2/17/09 18:25" ...
##  $ text      : chr  "I\xe4\xf3\xbbm traveling around Florida this week, this morning I'm in Ocala at
##  $ url       : chr  "https://twitter.com/SenBillNelson/status/1219483896" "https://twitter.com/SenBil
##  $ replies   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ retweets  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ favorites : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ user      : chr  "SenBillNelson" "SenBillNelson" "SenBillNelson" "SenBillNelson" ...
##  $ party     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

# Methodology

The approach applied to tackle this problem is two-tiered. Initially, a logistic regression model is applied to the original twitter dataset as a baseline for performance comparison. Next, a K-means clustering algorithm is used to cluster the dataset into groups. The clustered data will then be used to predict 'party affiliation' using logistic regression. The added cluster will be used as an additional feature to the already existing ones. The assumption here is that, appending clusters will improve the predictive capability of the logistic regression model.

The ease with which it can be implemented and its efficiency in training data makes logistic regression an ideal candidate for this problem. Additionally, it is very fast at classifying and can have a good accuracy for simple datasets. In reality however, there may not be linearity between the features and the label. The choice for K-means clustering is its ability to work well with large datasets and its simple application. Outlier data may be an issue of concern, as K-means can be sensitive to it.

# Analysis

Initial import and observation of the dataset indicates there is a balanced distribution of the 'party' label and not heavily skewed to one end. It is indicated in [Figure 2] below. Here, the Democratic party is represented as 0, and Republicans as 1.

```
## party
##      0      1
## 145544 149510
```

Features such as 'created_at', 'text', 'url', 'user', 'bioguide_id', and, 'state' were dropped since they have little to no use in predicting the label (dependent variable). Initially, a logistic regression model was created on a training set and applied it to a test set. The partition was done in a 70-30 split. Result of this initial model suggests an accuracy of about 56.28%. This will be used as a baseline to compare if the model accuracy can be improved by applying a layer of clustering.

```
##
## Call:
## glm(formula = party ~ ., family = "binomial", data = twTrain,
##     maxit = 100)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4904  -1.1847   0.1264   1.1693   8.4904
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.759e-02  4.508e-03   3.901 9.58e-05 ***
## replies      1.802e-03  5.380e-05  33.504  < 2e-16 ***
## retweets    -2.099e-04  1.154e-05 -18.192  < 2e-16 ***
## favorites   -8.095e-05  4.970e-06 -16.288  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 286285  on 206537  degrees of freedom
```

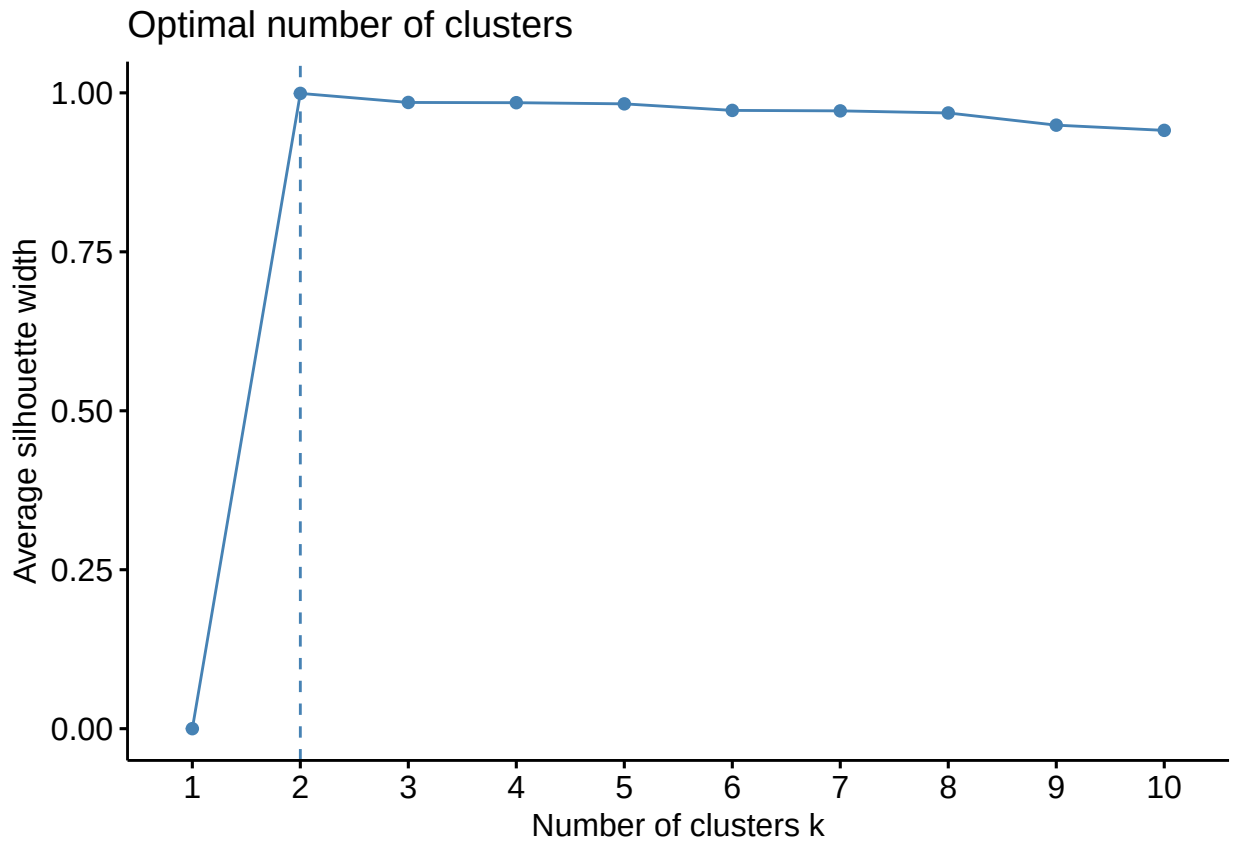
```
## Residual deviance: 281778  on 206534  degrees of freedom
## AIC: 281786
##
## Number of Fisher Scoring iterations: 9
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction      0      1
##          0  15117   3608
##          1  86764 101049
##
##                Accuracy : 0.5624
##                  95% CI : (0.5603, 0.5646)
##     No Information Rate : 0.5067
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.1152
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.14838
##             Specificity : 0.96553
##          Pos Pred Value : 0.80732
##          Neg Pred Value : 0.53803
##              Prevalence : 0.49328
##          Detection Rate : 0.07319
##    Detection Prevalence : 0.09066
##       Balanced Accuracy : 0.55695
##
##        'Positive' Class : 0
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0  6551  1506
##          1 37112 43347
##
##                Accuracy : 0.5637
##                  95% CI : (0.5604, 0.567)
##     No Information Rate : 0.5067
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.1177
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.15004
##             Specificity : 0.96642
##          Pos Pred Value : 0.81308
##          Neg Pred Value : 0.53875
##              Prevalence : 0.49328
```

```
##          Detection Rate : 0.07401
##    Detection Prevalence : 0.09102
##       Balanced Accuracy : 0.55823
##
##        'Positive' Class : 0
##
```

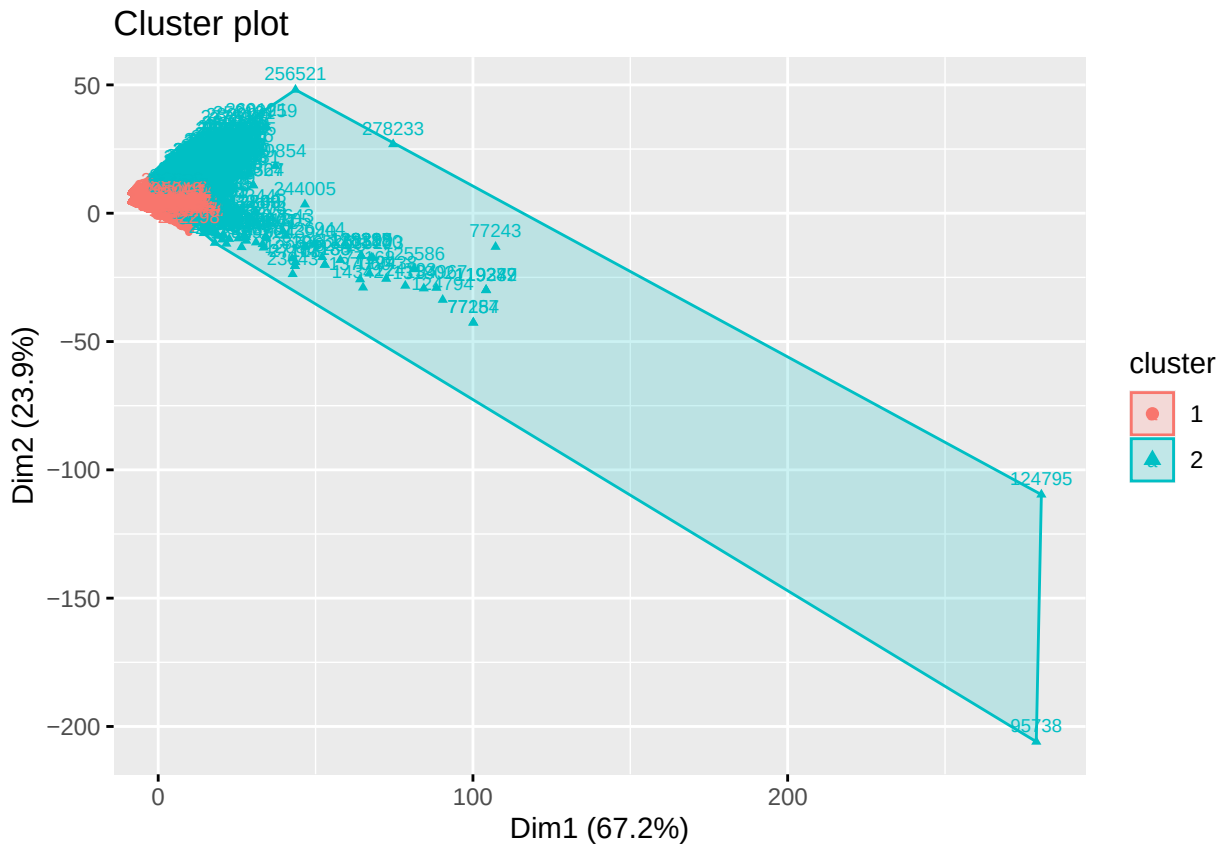
The next stage is to cluster the dataset using K-means clustering. This requires normalizing the continuous features (i.e. replies, re-tweets, and favorites). Using the silhouette method, we identify that k=2 is the optimal k for the dataset [Figure 3]. With the optimal k identified, applying a kmeans function results in a clustering that is lopsided, 1222 in cluster1 and 293832 in cluster2. This may potentially render the clustering uninformative as the features might be too close to each other and incorporating outliers. Cluster visualization is indicated below in [Figure 4].




```
## List of 9
##  $ cluster     : int [1:295054] 1 1 1 1 1 1 1 1 1 1 ...
##  $ centers     : num [1:2, 1:3] -0.0539 12.9706 -0.02 4.8051 -0.0326 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:2] "1" "2"
##   .. ..$ : chr [1:3] "replies" "retweets" "favorites"
##  $ totss       : num 885159
##  $ withinss    : num [1:2] 61367 513485
##  $ tot.withinss: num 574852
##  $ betweenss   : num 310307
##  $ size        : int [1:2] 293832 1222
##  $ iter        : int 1
```

```
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"
```

```
##
##      1      2
## 293832   1222
```



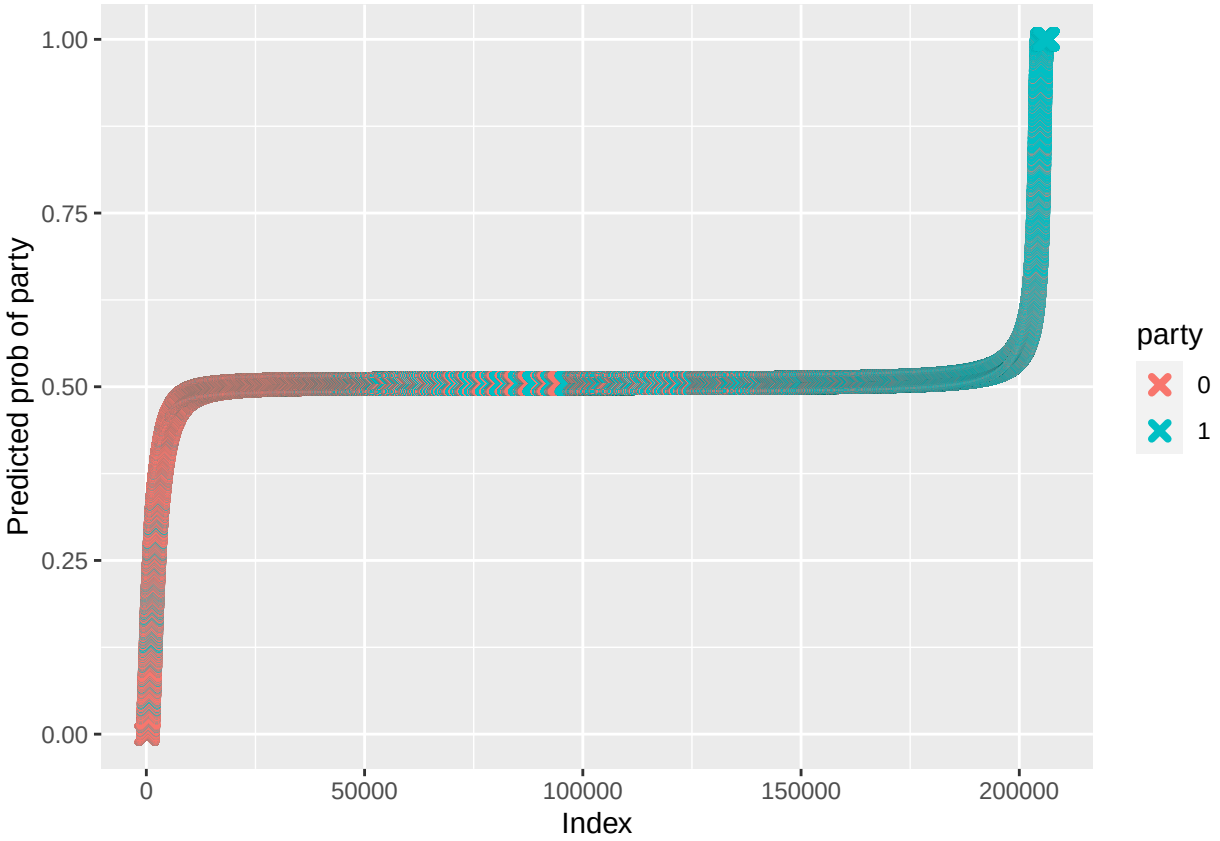

Cluster values are added to the dataset, thereby having reply, re-tweets, favorites, and cluster as the features for predicting party affiliation. As a result, re-running the logistic regression model on this transformed dataset may improve the model. Looking at the model created by using the training set and applied to the test set (still 70-30 split), there is only a very slight improvement in accuracy to about 56.44%. This indicates that the clustering layer applied did not have a major factor in improving predictive accuracy of the logistic regression model.

```
##     replies            retweets         favorites       party
##  Min.   :     0.0   Min.   :      0   Min.   :      0   0:101881
##  1st Qu.:     1.0   1st Qu.:      3   1st Qu.:      3   1:104657
##  Median :     3.0   Median :      8   Median :     13
##  Mean   :   200.4   Mean   :    478   Mean   :   1426
##  3rd Qu.:    15.0   3rd Qu.:     37   3rd Qu.:     84
##  Max.   :142859.0   Max.   :3644423   Max.   :2108869
##     cluster
##  Min.   :1.000
##  1st Qu.:1.000
##  Median :1.000
##  Mean   :1.004
```

```
##  3rd Qu.:1.000
##  Max.   :2.000

##     replies            retweets           favorites       party
##  Min.   :     0.0   Min.   :      0.0   Min.   :      0   0:43663
##  1st Qu.:     1.0   1st Qu.:      3.0   1st Qu.:      3   1:44853
##  Median :     3.0   Median :      8.0   Median :     13
##  Mean   :   195.1   Mean   :    456.4   Mean   :   1380
##  3rd Qu.:    15.0   3rd Qu.:     37.0   3rd Qu.:     84
##  Max.   :104556.0   Max.   :1712802.0   Max.   :4603556
##     cluster
##  Min.   :1.000
##  1st Qu.:1.000
##  Median :1.000
##  Mean   :1.004
##  3rd Qu.:1.000
##  Max.   :2.000

## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0  6616  1477
##          1 37047 43376
##
##                Accuracy : 0.5648
##                  95% CI : (0.5615, 0.568)
##     No Information Rate : 0.5067
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.1199
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.15152
##             Specificity : 0.96707
##          Pos Pred Value : 0.81750
##          Neg Pred Value : 0.53935
##              Prevalence : 0.49328
##          Detection Rate : 0.07474
##    Detection Prevalence : 0.09143
##       Balanced Accuracy : 0.55930
##
##        'Positive' Class : 0
##
```




## Conclusion

Predicting human behavioral patterns is often an uphill task, especially when it is hard to quantify. It can be inferred from the results of the models applied, both logistic regression and K-means clustering, engagement metrics such as replies, re-tweets, and favorites are not strong predictors of political party affiliation among elected officials. Even reducing the dimension of the features by dropping some variables does not lead to improvement of the predictive capability. It is also clear from the results that the the K-means clustering approach has not produced the expected level of result. This might be due to the dataset itself and how similar it is. It was only able to differentiate two clusters where one is significantly larger than the other and the second only include what appears to be outliers. In general, it is very difficult to predict human behavior that requires analytical and emotional reasoning. Twitter engagement metrics, especially for political content may significantly differ depending on which party is in power, the poll number, and the current hot issues. If there is a way to incorporate all that into the dataset, there is a potential to improve the predictive capability of the model. But there are no guarantees.