**Predicting Online News Articles Popularity: A Classification Approach**

Aditya Kumar, Sasank Reddy, Nischay Sai

 2022-12-20

**1. Project Background**

With the help of the Internet, online news can be instantly spread around the world. Social media, such as Twitter and Facebook, has made it easy for people to read and share news online. The popularity of an article is usually determined by its number of readers, likes, or shares. In our dataset, we use shares as the metric to measure popularity.

In the online news industry, content providers or advertisers benefit greatly from the ability to accurately predict the popularity of content before it is published. Thus, it is interesting and meaningful to use machine learning techniques to predict the popularity of these articles. The popularity of news depends upon various features like the usage of different keywords, relevance to a trending topic, and perhaps even the day the article is published. It is necessary to know what makes one online news article more popular than another. There are two main popularity prediction approaches, those that use features only known after publication and those that use features known before publication. The dataset we have used consists of known features before publication. Though the prediction performance is usually low in the latter case, it is of more value addition to authors and content creators to flag whether their produced article is going to be unpopular before publishing it. As a preemptive next step, articles predicted to be unpopular can be enhanced to promote popularity by tweaking identified key features.

**2. About the Data**

Machine Learning Repository of the University of California at Irvine provides a dataset with a heterogeneous set of features about articles published by Mashable, a popular blog site in the world, during a period of two years, January 7, 2013 – January 7, 2015.

A copy of this dataset is taken from [Kaggle](https://www.kaggle.com/datasets/thehapyone/uci-online-news-popularity-data-set) instead, due to difficulties in downloading from the original source. This dataset consists of 39,664 observations for 61 different variables, with the number of shares as our dependent variable. A Classification problem is developed from this continuous variable by first identifying the median value of all available shares as a threshold and engineering another feature that used this threshold to mark articles as popular (1) or unpopular (0).

**2.1 SMART Questions**

Some of the SMART questions we initially framed were:

1. Does the day of the week an article is published have any correlation with the number of shares it receives? If yes, then which day receives the maximum shares and which day receives the minimum shares?
2. Which features of an article affect the prediction of shares an article receives, positively and negatively?
3. Which classification methodology produces the best results to predict whether an article is popular or not?
4. For the classification problem, is there a popularity imbalance in the dataset, and how to go around it? If yes, then what is the best metric to evaluate our model?
5. With what accuracy/precision/ recall can we predict, using the best model, whether an article is popular or not popular?

By using Statistical Modeling and Machine learning methods, we will attempt to predict if an article is popular or not, based on features that are available prior to the publication of the article.

Now we will begin our analysis.

**3. Initial Data Analysis**

Glossary of all the columns in the dataset:

* Url: URL of the article (non-predictive)
* timedelta: Days between the article publication and the dataset acquisition (non-predictive)
* ntokenstitle: Number of words in the title
* ntokenscontent: Number of words in the content
* nuniquetokens: Rate of unique words in the content
* nnonstop\_words: Rate of non-stop words in the content
* nnonstopuniquetokens: Rate of unique non-stop words in the content
* num\_hrefs: Number of links
* numselfhrefs: Number of links to other articles published by Mashable
* num\_imgs: Number of images
* num\_videos: Number of videos
* averagetokenlength: Average length of the words in the content
* numkeywords: Number of keywords in the metadata
* datachannelislifestyle: Is data channel 'Lifestyle'?
* datachannelis\_entertainment: Is data channel 'Entertainment'?
* datachannelis\_bus: Is data channel 'Business'?
* datachannelis\_socmed: Is data channel 'Social Media'?
* datachannelis\_tech: Is data channel 'Tech'?
* datachannelis\_world: Is data channel 'World'?
* kwminmin: Worst keyword (min. shares)
* kwmaxmin: Worst keyword (max. shares)
* kwavgmin: Worst keyword (avg. shares)
* kwminmax: Best keyword (min. shares)
* kwmaxmax: Best keyword (max. shares)
* kwavgmax: Best keyword (avg. shares)
* kwminavg: Avg. keyword (min. shares)
* kwmaxavg: Avg. keyword (max. shares)
* kwavgavg: Avg. keyword (avg. shares)
* selfreferencemin\_shares: Min. shares of referenced articles in Mashable
* selfreferencemax\_shares: Max. shares of referenced articles in Mashable
* selfreferenceavg\_sharess: Avg. shares of referenced articles in Mashable
* weekdayismonday: Was the article published on a Monday?
* weekdayistuesday: Was the article published on a Tuesday?
* weekdayiswednesday: Was the article published on a Wednesday?
* weekdayisthursday: Was the article published on a Thursday?
* weekdayisfriday: Was the article published on a Friday?
* weekdayissaturday: Was the article published on a Saturday?
* weekdayissunday: Was the article published on a Sunday?
* is\_weekend: Was the article published on the weekend?
* LDA\_00: Closeness to LDA topic 0
* LDA\_01: Closeness to LDA topic 1
* LDA\_02: Closeness to LDA topic 2
* LDA\_03: Closeness to LDA topic 3
* LDA\_04: Closeness to LDA topic 4
* global\_subjectivity: Text subjectivity
* globalsentimentpolarity: Text sentiment polarity
* globalratepositive\_words: Rate of positive words in the content
* globalratenegative\_words: Rate of negative words in the content
* ratepositivewords: Rate of positive words among non-neutral tokens
* ratenegativewords: Rate of negative words among non-neutral tokens
* avgpositivepolarity: Avg. polarity of positive words
* minpositivepolarity: Min. polarity of positive words
* maxpositivepolarity: Max. polarity of positive words
* avgnegativepolarity: Avg. polarity of negative words
* minnegativepolarity: Min. polarity of negative words
* maxnegativepolarity: Max. polarity of negative words
* title\_subjectivity: Title subjectivity
* titlesentimentpolarity: Title polarity
* abstitlesubjectivity: Absolute subjectivity level
* abstitlesentiment\_polarity: Absolute polarity level
* shares: Number of shares (target)

*Source: [kaggle](https://www.kaggle.com/datasets/thehapyone/uci-online-news-popularity-data-set)*

Understanding the data at hand was extremely important. The dataset had been specifically curated to facilitate a research paper that is referenced toward the end. It was then donated to the Machine Learning Repository of UCI after extensive text feature extractions.

**3.1 Understanding the Data**

Some of the attributes in the dataset have already been encoded for machine learning. However, we will decode it into a single column for visualization purposes. Such columns include:

# 1. Data\_Channel: Type of article (Entertainment, Lifestyle, Media, Technology, World, etc.)

# 2. Publish\_Day: Day the article was published (Monday, Tuesday, etc.)

Some of the features are dependent on the particularities of the Mashable service (whose articles have been used as data source): articles often reference other articles published in the same service; and articles have meta-data, such as keywords, data channel type, and the total number of shares (when considering Facebook, Twitter, Google+, LinkedIn, Stumble-Upon and Pinterest). The minimum, average and maximum number of shares was determined for all Mashable links cited in the article and were extracted to prepare the data. Similarly, the rank of all article keyword average shares was determined, to get the worst, average, and best keywords. For each of these keywords, the minimum, average, and maximum number of shares was extracted as a feature. [1]

Several features are extracted by performing natural language processing on the original articles. The Latent Dirichlet Allocation (LDA) algorithm was applied to all Mashable articles in order to first identify the five top relevant topics and then measure the closeness of the current article to such topics. To compute the subjectivity and polarity sentiment analysis, the pattern web mining module was used. Subjectivity measures how subjective the article is – in terms of personal opinions vs factual information. It ranges from 0-1 with a higher score indicating more personal opinions and a lower score indicating factual inclination. Polarity sentiment analysis indicates the sentiment – negative, neutral, and positive. It ranges from –1 to 1.

An interesting point was raised by Prof Edwin Lo in his feedback on the project proposal. Is the number of shares a moving target? Is it a function of the number of days that have passed since when an article was published that determines the number of shares it receives? After reading through a couple of research papers on this topic, one suggested that there is a convergence of shares reached and the number of days it took that to happen was roughly 21.

There is a feature in the dataset: timedelta; which gives us the No. of days between the article being published and this dataset being collected. To reinforce the above assumption, it was important to see whether there was any relation between the timedelta and the number of shares the article received within our dataset.

We plotted this using a scatter plot:

A picture containing chart

Description automatically generated

Text

Description automatically generated

Two Observations here:

1. It seems like there is no concrete relationship between when the article was published and the number of shares it receives if data is acquired after 3 weeks, that is, there appears to be some convergence in the number of shares an article receives a definite number of days (21) after it has been published.

2. In the dataset at hand, the majority of the articles (~87%) have shares in the range of 0-5000. Whereas the maximum number of shares received is as high as 843300.

In addition, to eliminate the problem of class imbalance in our target variable, we used the median value of shares, instead of the mean to engineer our target variable.

**3.2 Exploratory Data Analysis**

While looking at the dataset, we had a few curious questions. we will help answer them with EDA.

Q1. What is the effect of digital content on the popularity of an article (hence the number of shares it receives)?

There are two primary digital content features – number of images and number of videos in an article. We’ll look at their distribution first.

Number of Videos:

Chart

Description automatically generated

Majority of the number of videos is distributed between 0 and 1. Looking at their value counts:

Graphical user interface, text

Description automatically generated

Here, True represents if the count is 0 or 1 and False represents any other number of videos. We see again that ~87% of the data is in the former category.

So, we decided to change our analysis a bit. Now we wanted to see if having 0 number of videos is more beneficial than having 1 or vice versa. We changed the scope of analysis to between just these two groups. To see this, we plotted boxplots:

Chart, box and whisker chart

Description automatically generated

From the above plot, we see that the average of num\_video = 0 and 1 is about the same. We can assume that having a greater number of videos does not have a direct effect on the number of shares.

Next, we plotted the relationship between Number of images and shares:

Chart, scatter chart

Description automatically generated

Looking at this plot, we see that there isn’t any concrete relationship between number of images to the number of shares. Although, articles that tend to have a number of images in the range of 0-20 seem to have a higher number of shares, crudely speaking.

Q2. Is there a relationship between the number of words in the content and the number of words in the title in the article’s popularity?

We first decided to see the distribution of the Number of words in the title:

Chart, histogram

Description automatically generated

Wow, it appears to be a nice normally distributed graph with an average of 10 words!

Next, we saw the distribution of the number of words in the content:

Chart, histogram

Description automatically generated

Upon checking the value counts of this feature, we identified an anomaly. Some articles appeared to have 0 words in their content! As the next step, we subset these mystery articles from the dataset and decided to have a look at the articles through their links directly.

We found that these were errors in the dataset. These articles did in fact have words in their content. This was an important catch because we’ll have to remove these erroneous records from the dataset before we train our model.

Before we proceeded to see the relationship of these two features with the target, out of curiosity, we wanted to see how they were related to each other. We plotted this relationship through a scatter plot:

Chart, scatter chart

Description automatically generated

It seems like, except for a few outliers, number of words in the content peak when the Number of words in the title is between 8-14 and they fall off gradually as it increases or decreases from this range. This is interesting.

Plotting relationship between number of words in title to popularity:

Chart

Description automatically generated

The unpopular line is not of that much interest as the number of shares for unpopular articles remains constant. But with popular articles, we see a constant increase in shares when the title increases and there is a sharp rise between 15-20. But this is also followed by a sharp decrease once ~18 or 19 words are crossed. Something to keep in mind for content creators.

We then plotted the relationship between number of words in the content and number of shares:

Chart, scatter chart

Description automatically generated

Here, we don’t see a significant relationship, but it does appear as though keeping the content small (0-2000 words), the number of shares could be higher. This is probably because readers would prefer reading a small catchy article over a long boring one and are more likely to share it.

Q3. Evaluating whether Day of Week has any effect on the popularity, in the number of shares?

To analyze this effectively, we created count plots on the Days of the Week, and we kept the hue as to whether the article was popular or not. This is what the plot looked like:

Chart, bar chart

Description automatically generated

This plot gives us highly useful insight. An article published over the weekend is more likely to be popular as opposed to an article that is published over the weekday. This makes intuitive sense since people have more time to read articles over the weekend as opposed to weekdays. To showcase this, we’ll plot the percentage chance of popularity on all weekdays according to our data:

Chart, bar chart

Description automatically generated

We observe a trend here. Near the weekends, the percentage of popular articles increases, and it peaks on Saturday, however, it gradually decreases then and is the lowest during mid-week on Wednesday.

Q4. Evaluating whether Data Channel has any effect on the popularity.

We took a similar approach to visualize this as above:

Chart, bar chart

Description automatically generated

We can observe that in the Data Channel of technology and social media, the number of popular news is much larger than unpopular ones, and in the Data Channel of world and entertainment, the number of unpopular news is larger than popular ones. This reflects that the readers of "Mashable.com" prefer the channel of technology and social media over the channel of world and entertainment.

To reinforce this further, we’ll plot the percentages like before:

Chart, bar chart

Description automatically generated

Q5. Title sentiment vs the number of shares.

This feature tells us whether the article expresses a negative sentiment, neutral or positive sentiment. We plotted a simple relation with the number of shares using a scatter plot:

Chart, scatter chart

Description automatically generated

Mostly the articles have titles that are not too positive or negative. It lies within the range of -0.5 to 0.5. However, the highest concentration can be seen in the 0 axes i.e., high no. of articles is neutral in nature with the higher number of shares. We also observe two small peaks at -1 and 1 in the number of shares.

Q6. Global subjectivity’s effect on the number of shares.

This feature gives us information on the subjectivity of the article. A high rating indicates more personal opinions whereas, a low rating indicates more factual information. Again, we plotted this using a simple scatter plot:

Chart, scatter chart

Description automatically generated

Maximum shares are received by articles that have values for Global Subjectivity between 0.3 to 0.7. Hence, we conclude that most of the articles with neutral Global Subjectivity receive maximum shares, that is, the articles contain a good blend of personal opinions and factual information.

**4. Pre-processing and Modeling**

**4.1 Pre-processing**

The following pre-processing steps were performed:

* Subset –
  + We first subset our dataset to remove all articles that had a timedelta of less than 21.
  + We also subset the dataset to remove those articles that had 0 words in their content, as identified during the EDA.
* Dropping irrelevant columns – Further, irrelevant columns like URL and timedelta were removed from the dataset.
* Correlation heatmap – We created a correlation heatmap between all the features of our dataset to eliminate features with high correlation. We chose our threshold value as 0.7. This was done to avoid the problem of any multicollinearity in modeling. This heatmap was a 45 x 45 matrix and therefore, we haven’t included it here. It can be found in the technical analysis.
* Outlier treatment – Since most of our columns were numerical, we decided to treat the outliers with the following rule:
  + Outliers clipped at Q1 – 1.5 IQR and Q3 + 1.5 IQR
  + Any value below or above this range was clipped at this boundary level, not removed from the dataset.
* Lastly, to deal with the imbalanced scale/range of features, we decided to employ StandardScaling as opposed to Log transformations. This was done because few features contained negative values and we also wanted to preserve the distribution of our dataset.

**Logistic Regression**

Having over 500,000 is too much to process in the model and is causing the session to slow down. To prevent this, we are going to further subset the dataset before processing and also store certain features as factor variables.

We will include the following variables in our initial development of our model to predict delayed status: \* Airline \* Origin Airport \* Destination Airport \* Month \* Day of Week \* Time of Day (by hour of departure) \* Distance

Performing a 4:1 Train-test Split of Dataset. Then we go ahead and build the logistic regression model first.

The following coefficients are significant:

## Estimate Pr...z..

## (Intercept) -4.2650 1.65e-03

## airlineAtlantic Southeast Airlines 0.4264 1.90e-03

## airlineFrontier Airlines Inc. 0.6956 2.84e-05

## airlineHawaiian Airlines Inc. 0.9542 1.04e-03

## airlineJetBlue Airways 0.4359 3.01e-03

## airlineSkywest Airlines Inc. 0.5766 1.02e-05

## airlineSouthwest Airlines Co. 0.4674 1.65e-04

## airlineSpirit Air Lines 0.8018 6.35e-07

## airlineVirgin America 0.6223 6.99e-04

## originDHN 3.0847 2.72e-02

## originDRO 2.5555 2.89e-02

## originISN 2.1976 4.35e-02

## originLRD 2.2347 3.93e-02

## originMGM 2.6300 1.33e-02

## originSGU 2.9412 1.03e-02

## destAGS 2.7435 3.63e-02

## destCOS 2.4227 3.43e-02

## destGSO 2.3179 4.63e-02

## destGSP 2.3925 3.99e-02

## destILM 2.9902 1.94e-02

## destMFR 2.7964 4.05e-02

## destMOB 2.3943 4.65e-02

## destSPS 3.3169 3.41e-02

## destSRQ 2.3943 4.80e-02

## MONTH02 0.1483 4.32e-02

## MONTH04 -0.1522 3.32e-02

## MONTH06 0.1605 2.22e-02

## MONTH09 -0.5159 4.66e-12

## MONTH10 -0.5091 4.51e-12

## MONTH11 -0.2604 3.48e-04

## dow.Q -0.1833 3.94e-06

## dow.C -0.1083 5.10e-03

## dow^4 -0.1577 2.66e-05

## hour 0.0629 8.08e-91

*ADD THE REST* The following Airlines are significant: \* Frontier Airlines \* Skywest Airlines \* Southwest Airlines \* Spirit Airlines

The following months are significant: \* February \* June \* October \* December

The following days of the week: \* Q? \* 6? (For Dys of the week, we were seeing some absurd results and we weren’t sure as to what this means.)

Hour was also a significant predictor.

Let’s plot the ROC for this model used on our training set first.

## Area under the curve: 0.654

Chart

Description automatically generated

Next we’ll plot the ROC curve for our testing set.

## Area under the curve: 0.6

Chart, line chart

Description automatically generated

Area under the curve is 0.6543 on the training data, and 0.60 on the testing data.

Lets look at other model metrics.

## fitting null model for pseudo-r2

## llh llhNull G2 McFadden r2ML r2CU

## -1.41e+04 -1.49e+04 1.67e+03 5.58e-02 7.06e-02 9.66e-02

McFadden value is 0.058

Confusion Matrix:

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 3240 1732

## 1 418 409

##

## Accuracy : 0.629

## 95% CI : (0.617, 0.642)

## No Information Rate : 0.631

## P-Value [Acc > NIR] : 0.602

##

## Kappa : 0.088

##

## Mcnemar's Test P-Value : <2e-16

##

## Sensitivity : 0.886

## Specificity : 0.191

## Pos Pred Value : 0.652

## Neg Pred Value : 0.495

## Prevalence : 0.631

## Detection Rate : 0.559

## Detection Prevalence : 0.857

## Balanced Accuracy : 0.538

##

## 'Positive' Class : 0

##

Accuracy on model: 62.9% Sensitivity / Recall: 19.1% - 409 / (1731+409) Specificity: 88.57% Precision: 49.5%

The specificity on this model is pretty good, but the other metric could use some improvement. Let’s try with a different cutoff value of 0.4.

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 2494 1166

## 1 1164 975

##

## Accuracy : 0.598

## 95% CI : (0.585, 0.611)

## No Information Rate : 0.631

## P-Value [Acc > NIR] : 1.000

##

## Kappa : 0.137

##

## Mcnemar's Test P-Value : 0.983

##

## Sensitivity : 0.682

## Specificity : 0.455

## Pos Pred Value : 0.681

## Neg Pred Value : 0.456

## Prevalence : 0.631

## Detection Rate : 0.430

## Detection Prevalence : 0.631

## Balanced Accuracy : 0.569

##

## 'Positive' Class : 0

##

At a lower cutoff value, the accuracy is slightly lower at 59.8% but there is an improvement in the sensitivity of the model at 45.5%

Can go even further with a lower cutoff value of 0.3.

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 1320 494

## 1 2338 1647

##

## Accuracy : 0.512

## 95% CI : (0.499, 0.525)

## No Information Rate : 0.631

## P-Value [Acc > NIR] : 1

##

## Kappa : 0.11

##

## Mcnemar's Test P-Value : <2e-16

##

## Sensitivity : 0.361

## Specificity : 0.769

## Pos Pred Value : 0.728

## Neg Pred Value : 0.413

## Prevalence : 0.631

## Detection Rate : 0.228

## Detection Prevalence : 0.313

## Balanced Accuracy : 0.565

##

## 'Positive' Class : 0

##

Recall with this model is **76.9%**.

**4.1.1 Trying with other variables**

In this section, we’ll build this model again, but now we include those extra variables that help predict delay even better because they are known after the flight either departs from the origin airport or arrives at the destination airport, dropping the value addition of this prediction. But just to see how the prediction power increases.

Significant Coefficients:

## (Intercept) airlineAmerican Airlines Inc.

## 3.25e-08 4.98e-02

## airlineAtlantic Southeast Airlines airlineDelta Air Lines Inc.

## 3.35e-02 9.05e-03

## airlineFrontier Airlines Inc. airlineHawaiian Airlines Inc.

## 6.05e-08 1.72e-02

## airlineJetBlue Airways airlineSkywest Airlines Inc.

## 2.85e-09 2.23e-02

## airlineSouthwest Airlines Co. airlineSpirit Air Lines

## 8.15e-16 2.91e-11

## airlineUnited Air Lines Inc. airlineVirgin America

## 4.58e-04 1.41e-09

## originABI originABQ

## 1.04e-03 9.67e-04

## originACV originADQ

## 2.09e-02 1.23e-02

## originAEX originAGS

## 2.01e-02 4.36e-02

## originAMA originANC

## 3.73e-02 5.16e-06

## originASE originAUS

## 2.40e-02 3.97e-02

## originBET originBIL

## 6.14e-04 2.50e-02

## originBMI originBOI

## 4.49e-02 7.35e-03

## originBRD originBTR

## 1.32e-02 3.10e-03

## originBUR originBZN

## 7.21e-04 3.43e-03

## originCID originCOS

## 4.76e-02 5.99e-03

## originCRP originDAL

## 6.54e-03 6.78e-03

## originDEN originDFW

## 3.01e-03 8.19e-03

## originDHN originDLH

## 3.27e-03 3.16e-02

## originDRO originDSM

## 1.47e-03 1.70e-02

## originELP originEUG

## 1.95e-02 5.33e-03

## originFAR originFAT

## 4.32e-03 4.93e-04

## originFLG originFSD

## 7.51e-03 4.01e-02

## originFSM originGEG

## 3.00e-03 3.03e-04

## originGRB originGRI

## 2.70e-02 2.56e-02

## originGTF originHNL

## 8.01e-03 7.57e-07

## originHOU originHSV

## 1.09e-02 1.33e-02

## originIAH originICT

## 1.62e-02 6.03e-03

## originIDA originILM

## 4.51e-02 1.97e-02

## originINL originISN

## 3.23e-02 1.34e-02

## originITO originJAC

## 3.98e-04 5.46e-03

## originJAN originJNU

## 2.34e-02 2.36e-02

## originKOA originKTN

## 1.51e-05 2.29e-02

## originLAS originLAX

## 9.06e-04 2.67e-04

## originLBB originLFT

## 1.77e-03 4.09e-02

## originLGB originLIH

## 2.99e-03 2.88e-07

## originLNK originLRD

## 7.19e-04 2.03e-03

## originMAF originMCI

## 1.66e-02 4.72e-02

## originMFE originMGM

## 5.04e-03 1.27e-03

## originMHK originMKG

## 2.07e-02 1.45e-02

## originMLU originMOT

## 8.53e-03 9.67e-03

## originMSO originMTJ

## 1.34e-02 2.05e-02

## originOAK originOGG

## 1.22e-04 1.01e-07

## originOKC originOME

## 1.17e-02 1.59e-03

## originONT originOTZ

## 2.63e-04 4.85e-04

## originPAH originPDX

## 2.26e-02 4.80e-04

## originPHX originPIB

## 6.71e-04 4.90e-02

## originPIH originPSC

## 2.94e-04 1.01e-03

## originPSE originPSP

## 3.11e-02 1.33e-03

## originRAP originRDM

## 1.21e-02 1.20e-02

## originRHI originRNO

## 2.65e-02 3.89e-04

## originROW originSAF

## 1.07e-02 3.01e-02

## originSAN originSBA

## 5.39e-04 1.04e-02

## originSBN originSCC

## 4.82e-02 4.74e-03

## originSEA originSFO

## 7.93e-05 4.68e-04

## originSGU originSHV

## 3.50e-05 1.61e-02

## originSIT originSJC

## 1.01e-03 2.63e-04

## originSJT originSLC

## 4.68e-03 2.33e-03

## originSMF originSNA

## 2.51e-04 1.92e-03

## originSPS originSTL

## 9.65e-03 2.82e-02

## originSUN originTUL

## 3.81e-03 4.44e-02

## originTUS originXNA

## 3.17e-03 5.80e-03

## originYUM MONTH06

## 2.44e-02 2.96e-05

## MONTH07 MONTH09

## 2.67e-02 4.05e-05

## MONTH10 MONTH11

## 3.03e-06 1.67e-02

## dow.Q dow^4

## 4.45e-03 1.77e-05

## hour distance

## 2.21e-90 0.00e+00

## actual\_elapsed\_time

## 0.00e+00

McFadden:

## fitting null model for pseudo-r2

## llh llhNull G2 McFadden r2ML r2CU

## -1.19e+04 -1.49e+04 6.16e+03 2.06e-01 2.37e-01 3.25e-01

Mc Fadden is 0.206

Plotting ROC.

## Area under the curve: 0.786

Diagram

Description automatically generated

Confusion Matrix:

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 12758 3957

## 1 1694 4353

##

## Accuracy : 0.752

## 95% CI : (0.746, 0.757)

## No Information Rate : 0.635

## P-Value [Acc > NIR] : <2e-16

##

## Kappa : 0.432

##

## Mcnemar's Test P-Value : <2e-16

##

## Sensitivity : 0.883

## Specificity : 0.524

## Pos Pred Value : 0.763

## Neg Pred Value : 0.720

## Prevalence : 0.635

## Detection Rate : 0.560

## Detection Prevalence : 0.734

## Balanced Accuracy : 0.703

##

## 'Positive' Class : 0

##

**4.2 Random Forest**

Train-test splitting of 4:1 ratio again.

Unable to use factor variables that are more than 53 categories, so we have to drop origin and destination airports

Building Model.

Making Confusion Matrix:

For the training set-

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 2884 5

## 1 0 1692

##

## Accuracy : 0.999

## 95% CI : (0.997, 1)

## No Information Rate : 0.63

## P-Value [Acc > NIR] : <2e-16

##

## Kappa : 0.998

##

## Mcnemar's Test P-Value : 0.0736

##

## Sensitivity : 1.000

## Specificity : 0.997

## Pos Pred Value : 0.998

## Neg Pred Value : 1.000

## Prevalence : 0.630

## Detection Rate : 0.630

## Detection Prevalence : 0.631

## Balanced Accuracy : 0.999

##

## 'Positive' Class : 0

##

99.9% Accuracy on the training set

Confusion Matrix for the Testing set

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 543 297

## 1 181 112

##

## Accuracy : 0.578

## 95% CI : (0.549, 0.607)

## No Information Rate : 0.639

## P-Value [Acc > NIR] : 1

##

## Kappa : 0.025

##

## Mcnemar's Test P-Value : 1.44e-07

##

## Sensitivity : 0.750

## Specificity : 0.274

## Pos Pred Value : 0.646

## Neg Pred Value : 0.382

## Prevalence : 0.639

## Detection Rate : 0.479

## Detection Prevalence : 0.741

## Balanced Accuracy : 0.512

##

## 'Positive' Class : 0

##

Accuracy: 57.8% Sensitivity / Recall: 27.4% Specificity: 75.0% Precision: 38.2%

Again, specificity is good but the other measures are not.

Trying with different amounts of trees.

## Confusion Matrix and Statistics

##

## Reference

## Prediction 0 1

## 0 543 297

## 1 181 112

##

## Accuracy : 0.578

## 95% CI : (0.549, 0.607)

## No Information Rate : 0.639

## P-Value [Acc > NIR] : 1

##

## Kappa : 0.025

##

## Mcnemar's Test P-Value : 1.44e-07

##

## Sensitivity : 0.750

## Specificity : 0.274

## Pos Pred Value : 0.646

## Neg Pred Value : 0.382

## Prevalence : 0.639

## Detection Rate : 0.479

## Detection Prevalence : 0.741

## Balanced Accuracy : 0.512

##

## 'Positive' Class : 0

##

Playing around with maxnodes, number of trees, and tree depth doesn’t seem to improve the model very well

**4.3 Notes on Data Limitations**

We would have been able to significantly improve the quality of our models if we had the following additional data:

* Weather Data
* Airline Staffing Data
* Data on International Flights
* Airport Security Data

We might need to wait for more data, and higher computational power, before we can predict a brave new world for air travel - one without long queues and outrageous wait times, without airport sleepovers and missed schedules.

**5 Summary and Conclusions**

In summary, the key lessons learnt from the dataset used in this project and key analysis performed were:

* Certain Airlines and Airports are more likely to have delays than others.
* As the day progresses, delays are more likely to occur.
* The average delay percentage of domestic flights have dropped from 36.51% in 2015 to 20.19% in 2022 (YTD).
* Flight delays are difficult to predict! Combining with more data would have been helpful.
* While more dataset would have provided more insight, higher computational power is required to develop better models!

**6 References**

1. Flight Delays and Cancellation Dataset. US Department of Transportation’s (DOT) Bureau of Transportation Statistics (2015). Retrieved November 22, 2022, from <https://www.kaggle.com/datasets/usdot/flight-delays>
2. Zieffler, Andrew (2018). R Markdown Theme Gallery, from <https://www.datadreaming.org/post/r-markdown-theme-gallery/>
3. K. Fernandes, P. Vinagre and P. Cortez. A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News. Proceedings of the 17th EPIA 2015 - Portuguese Conference on Artificial Intelligence, September, Coimbra, Portugal.