Online News Popularity Regression

Mohammad Rouintan

Computer Science, Shahid Beheshti University

Abstract

With the expansion of the Internet, more and more people enjoys reading and sharing online news articles. The number of shares under a news article indicates how popular the news is. In this project, we intend to find the best model and set of feature to predict the popularity of online news, using machine learning techniques. Our data comes from Mashable, a well-known online news website. We implemented 4 different learning algorithms on the dataset, Linear Regression, Ridge Regression, Lasso Regression and Polynomial Regressoin. Their performances are recorded and compared. Feature selection methods , GridSearchCv and RandomizedSearchCv are used to improve performance.

Introduction

we are going to work with the News Popularity Prediction dataset. we will implement a regression models using the Scikit-Learn package to predict the popularity of new articles (the number of times they will be shared online) based on about 60 features. In this way we use kind of scaling method and transformation for get better result for example:

- MinMax Scaler
- Standard Scaler
- Quantile Transformation
- Log Transformation
- Power Transfromation

In the other hand we use feature selection with forward selection and backward elimination method that implementated from scratch for find best features and use GridSearchCv and RandomizedSearchCv for tune our model's hyperparameters.

Methodologies

Data Analysis

first, for data analysis we should read description of dataset and know features and target of dataset in detail.

we found "url" and "time delta" are non-predictive features in dataset so they are not usage and we should remove them from dataset. Next step is trimming outliers because the dataset has many outliers that make our analysis wrong. For example, the dataset has samples with zero n-tokens-content that means there are news with zero number of words that have other features and number of shares. In the other hand there is a news with 840,000 number of shares despite that the 99th percentile of shares is 31,500. Therefore we use a method for trimming outlier that trim samples that have number less than 1st percentile of current column or more than 99th percentile of current column.

second, we divide the features into several categories so that the members of each category are almost related to each other. These categories are:

- Word Features
- Media Features
- Keyword Features
- Reference Features
- Positive and Negative Features
- Title Features
- Polarity Features
- Topic Features

We use the pair plot to show the approximate relationship of each of the features of some categories.

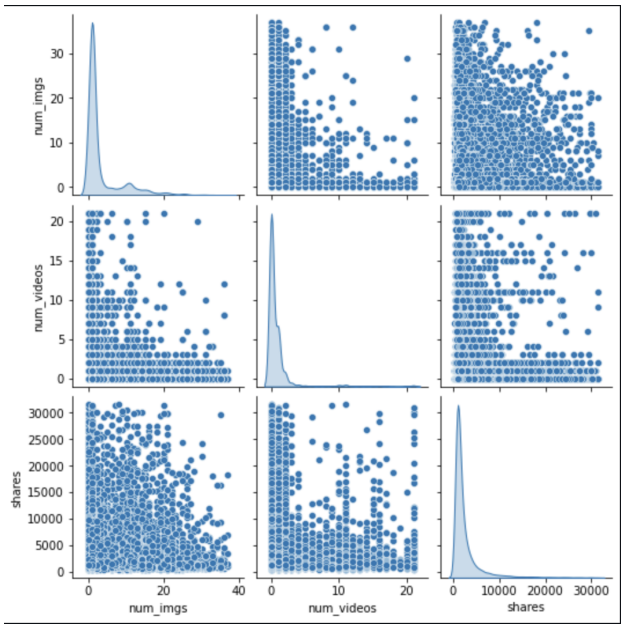

Figure 1: *Conclusion of this plot is, With the increase of number of images in the content, the number of videos in the content decreases slowly and With the increase of number of images in the content, the number of shares decreases slowly.*

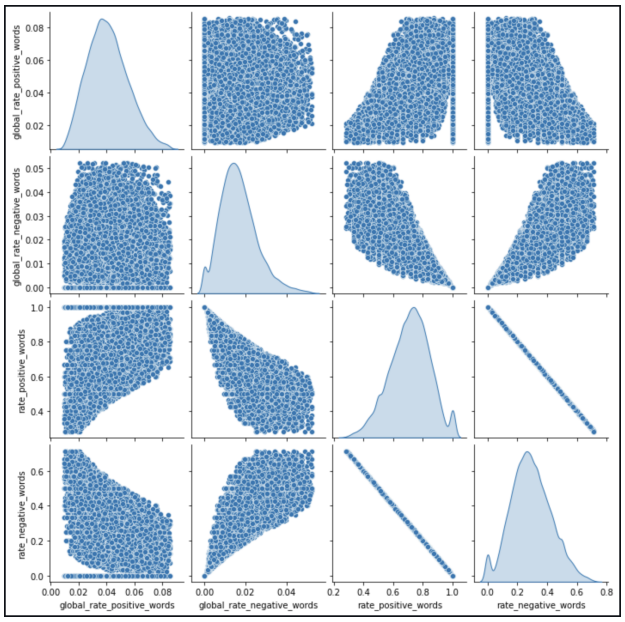

Figure 2: *Conclusion of this plot is There is a linear relationship between rate-positive-words and rate-negative-words.*

for Weekday Features and Channel Features we convert weekday features to a single feature with name "published-day" and channel features to a single feature with name "data-channel". We show count of news shares in each day and each data channel.

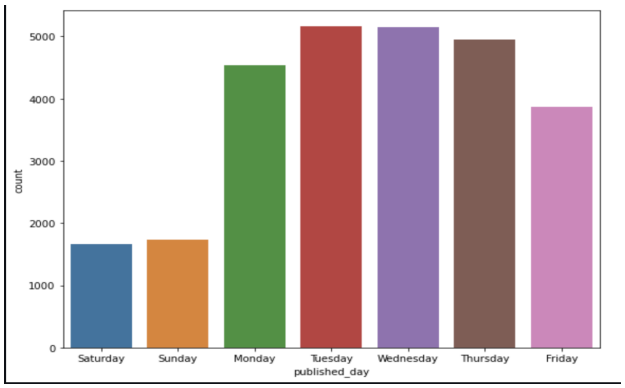

Figure 3: *Most of News published in Monday, Tuesday, Wednesday and Thursday.*

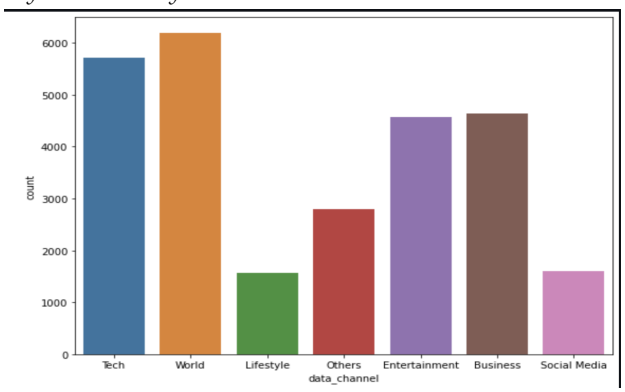

Figure 4: *Most of News are about Tech, World, Entertainment and Business.*

Last plot for analysis data is Violon Plot. A violin plot is a hybrid of a box plot and a kernel density plot, which shows peaks in the data. It is used to visualize the distribution of numerical data. Based on plots we found many of features and target don't have normal distribution. Because number of columns of dataset is large we show two example of this.

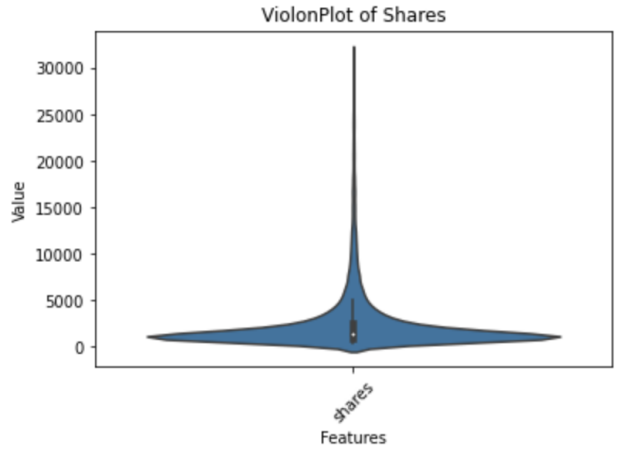

Figure 5: *Shares don't have normal distribution*

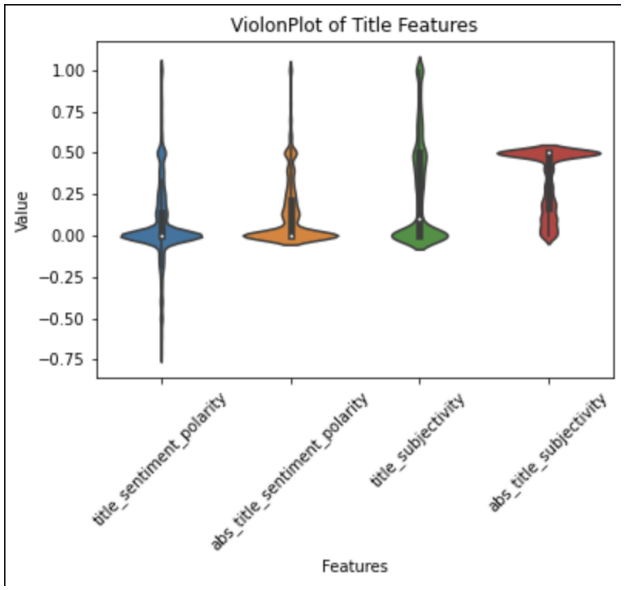

Figure 6: *Title Features don't have normal distribution*

Hypothesis Test

Question1: Average of shares is 2600? (Calculate the T-test for the mean of ONE group of scores. This is a test for the null hypothesis that the expected value (mean) of a sample of independent observations a is equal to the given population mean, popmean.)

$$H_0 : \mu = 2600$$

$$H_1 : \mu \neq 2600$$

```
stat : -0.06955787395104038 , p_value : 0.9445460809613277
accept null hypothesis
```

Figure 7: *Accept null hypothesis because p-value > alpha, alpha=0.05*

Question2: Average of shares in Tuesday and Wednesday is equal? (Calculate the T-test for the means of two independent samples of scores. This is a test for the null hypothesis that 2 independent samples have identical average (expected) values. This test assumes that the populations have identical variances by default.)

$$H_0 : \mu_{Tuesday} = \mu_{Wednesday}$$

$$H_1 : \mu_{Tuesday} \neq \mu_{Wednesday}$$

```
stat : 0.41937712866223426 , p_value : 0.6749492757571369
accept null hypothesis
```

Figure 8: *Accept null hypothesis because p-value > alpha, alpha=0.05*

Question3: Data Channel effects on shares ? (The one-way ANOVA tests the null hypothesis that two or more groups have the same population mean. The test is applied to samples from two or more groups, possibly with differing sizes.)

$$H_0 : \mu_{World} = \mu_{Tech} = \mu_{Entertainment}$$

$$H_1 : \mu_{World} \neq \mu_{Tech} \neq \mu_{Entertainment}$$

```
stat : 93.21732849382894 , p_value : 5.541377340495207e-41
reject null hypothesis
```

Figure 9: *Reject null hypothesis because p-value <= alpha, alpha=0.05*

Question4: Is there a relationship between published-day as categorical feature and data-channel as categorical feature? (Chi-square test of independence of variables in a contingency table. This function computes the chi-square statistic and p-value for the hypothesis test of independence of the observed frequencies in the contingency table observed. The expected frequencies are computed based on the marginal sums under the assumption of independence). First we calculate contingency table with crosstab attribute of pandas library.

$$H_0 : \textit{There isn't a relationship}$$

$$H_1 : \textit{There is a relationship}$$

```
chi2 stat : 244.42028069129918 , p_value : 8.297789120826792e-33
dof :  36
Reject H0,There is a relationship between 2 categorical variables
```

Figure 10: *Reject null hypothesis because p-value <= alpha, alpha=0.05*

Question5: Average of num-hrefs and num-self-hrefs? (Calculate the t-test on TWO RELATED samples of scores, a and b. This is a test for the null hypothesis that two related or repeated samples have identical average (expected) values.)

$$H_0 : \mu_{hrefs} = \mu_{selfhrefs}$$

$$H_1 : \mu_{hrefs} \neq \mu_{selfhrefs}$$

```
stat : 147.55984388610443 , p_value : 0.0
reject null hypothesis
```

Figure 11: *Reject null hypothesis because p-value <= alpha, alpha=0.05*

Scaling methods

MinMax Scaler: Transform features by scaling each feature to a given range. This estimator scales and translates each feature individually such that it is in the given range on the training set, e.g. between zero and one.

Standard Scaler: Standardize features by removing the mean and scaling to unit variance. The standard score of a sample x is calculated as:

$$z = \frac{x - \mu}{s}$$

Quantile Transformer: One of the most interesting feature transformation techniques that I have used, the Quantile Transformer Scaler converts the variable distribution to a normal distribution. and scales it accordingly. Since it makes the variable normally distributed, it also deals with the outliers. Here are a few important points regarding the Quantile Transformer Scaler:

- It computes the cumulative distribution function of the variable.
- It uses this cdf to map the values to a normal distribution.
- Maps the obtained values to the desired output distribution using the associated quantile function.

A caveat to keep in mind though: Since this scaler changes the very distribution of the variables, linear relationships among variables may be destroyed by using this scaler.

Log Transformer: The Log Transform is one of the most popular Transformation techniques out there. It is primarily used to convert a skewed distribution to a normal distribution/less-skewed distribution. In this transform, we take the log of the values in a column and use these values as the column instead. A small caveat though – if our data has negative values or values ranging from 0 to 1, we cannot apply log transform directly – since the log of negative numbers and numbers between 0 and 1 is undefined, we would get error or NaN values in our data. In such cases, we can add a number to these values to make them all greater than 1. Then, we can apply the log transform.

Power Transformer: I often use this feature transformation technique when I am building a linear model. To be more specific, I use it when I am dealing with heteroskedasticity. Like some other scalers we studied above, the Power Transformer also changes the distribution of the variable, as in, it makes it more Gaussian(normal). We are familiar with similar power transforms such as square root, and cube root transforms, and log transforms.

However, to use them, we need to first study the original distribution, and then make a choice. The Power Transformer actually automates this decision making by introducing a parameter called lambda. It decides on a generalized power transform by finding the best value of lambda using either the:

- Box-Cox transform
- The Yeo-Johnson transform

While I will not get into too much detail of how each of the above transforms works, it is helpful to know that Box-Cox works with only positive values, while Yeo-Johnson works with both positive and negative values.

To train the model, we use a dataset with 59 features, we also remove the outliers with the method mentioned at the beginning and for testing a model we use train and test split method of scikit-learn to divide dataset to training set and testing set.

Linear Regression

MinMax Scaler : Convert scale of each column with MinMax Scaler method.

$$RMSE = 0.1077927007$$

$$Accuracy = 8.17\%$$

Standard Scaler : Convert scale of each column with Standard Scaler method.

$$RMSE = 0.971898051$$

$$Accuracy = 8.18\%$$

Quantile Transformation : Transform distribution of each column to uniform distribution or normal distribution.

$$RMSE_{uniform} = 0.2598998564$$

$$Accuracy_{uniform} = 17.81\%$$

$$RMSE_{normal} = 0.9123201056$$

$$Accuracy_{normal} = 15.95\%$$

Feature Selection :In this scale we use feature selection methods that implemented from scratch to find the best features. But the selection of features does not have a significant effect on the result, even if when we reduce the number of selected features, the accuracy of the model decreases. The results are as follows:

$$RMSE_{forward} = 0.9123757034$$

$$Accuracy_{forward} = 15.94\%$$

$$RMSE_{backward} = 0.912989981$$

$$Accuracy_{backward} = 15.83\%$$

Log Transformation : Transform distribution of each column to normal distribution.

$$RMSE = 0.7517094892$$

$$Accuracy = 14.00\%$$

Power Transformation (yeo-johnson) : Transform distribution of each column to normal distribution.

$$RMSE = 0.9099390439$$

$$Accuracy = 16.1\%$$

Ridge Regression

MinMax Scaler : Convert scale of each column with MinMax Scaler method.

$$RMSE = 0.1077942824$$

$$Accuracy = 8.18\%$$

Standard Scaler : Convert scale of each column with Standard Scaler method.

$$RMSE = 0.9718941728$$

$$Accuracy = 8.18\%$$

Quantile Transformation : Transform distribution of each column to uniform distribution or normal distribution.

$$RMSE_{uniform} = 0.2599012128$$

$$Accuracy_{uniform} = 17.81\%$$

$$RMSE_{normal} = 0.9123781046$$

$$Accuracy_{normal} = 15.95\%$$

Log Transformation : Transform distribution of each column to normal distribution.

$$RMSE = 0.7522392809$$

$$Accuracy = 13.87\%$$

Power Transformation (yeo-johnson) : Transform distribution of each column to normal distribution.

$$RMSE = 0.9106862852$$

$$Accuracy = 15.96\%$$

GridSearchCv and RandomizedSearchCv : In this scale we use GridSearchCv and RandomizedSearch cv for tune our model hyperparameters (hyperparameter for Ridge is α). But using GridSearchCv and RandomizedSearchCv does not have a significant effect on the result.

$$\alpha_{best} = 0.204081632653061$$

$$RMSE_{Grid} = 0.9106158089$$

$$Accuracy_{Grid} = 15.97\%$$

$$\alpha_{best} = 0.09955363805627204$$

$$RMSE_{Randomized} = 0.9106008833$$

$$Accuracy_{Randomized} = 15.97\%$$

We see using Ridge Regression instead of Linear Regression has very little effect but not significant.

Lasso Regression

MinMax Scaler : Convert scale of each column with MinMax Scaler method.

$$RMSE = 0.112512931$$

$$Accuracy = -0.04\%$$

Standard Scaler : Convert scale of each column with Standard Scaler method.

$$RMSE = 1.0144664738$$

$$Accuracy = -0.04\%$$

Quantile Transformation : Transform distribution of each column to uniform distribution or normal distribution.

$$RMSE_{uniform} = 0.2869379532$$

$$Accuracy_{uniform} = -0.18\%$$

$$RMSE_{normal} = 0.9959507532$$

$$Accuracy_{normal} = -0.16\%$$

Log Transformation : Transform distribution of each column to normal distribution.

$$RMSE = 0.8110603126$$

$$Accuracy = -0.12\%$$

Power Transformation (yeo-johnson) : Transform distribution of each column to normal distribution.

$$RMSE = 0.9941839185$$

$$Accuracy = -0.16\%$$

Polynomial Regression

We use Polynomial Features of degree 2 for better result. Created dataset has 1770 columns(features). We Train this dataset with linear regression and ridge regression.

MinMax Scaler : Convert scale of each column with MinMax Scaler method.

$$Accuracy_{train-linear} = 18.17\%$$

$$Accuracy_{test-linear} = 0.85\%$$

$$Accuracy_{train-ridge} = 16.38\%$$

$$Accuracy_{test-ridge} = 4.77\%$$

Standard Scaler : Convert scale of each column with Standard Scaler method.

$$Accuracy_{train-linear} = 18.13\%$$

$$Accuracy_{test-linear} = 0.1\%$$

$$Accuracy_{train-ridge} = 17.46\%$$

$$Accuracy_{test-ridge} = 2.13\%$$

Quantile Transformation : Transform distribution of each column to uniform distribution or normal distribution.

Linear :

$$Accuracy_{train-linear-uniform} = 29.00\%$$

$$Accuracy_{test-linear-uniform} = 15.80\%$$

$$Accuracy_{train-linear-normal} = 27.71\%$$

$$Accuracy_{test-linear-normla} = 13.43\%$$

Ridge :

$$Accuracy_{train-ridge-uniform} = 27.80\%$$

$$Accuracy_{test-ridge-uniform} = 17.82\%$$

$$Accuracy_{train-ridge-normal} = 27.34\%$$

$$Accuracy_{test-ridge-normla} = 14.5\%$$

Results

The general conclusion that can be drawn from this dataset is that according to the features and the target of dataset, we found that this dataset is not suitable for prediction and such models.

But still We found that the best models for this dataset are linear and ridge, but lasso does not show good results with different scalings. Then, we saw that with polynomial regression of the 2nd degree, we were able to get a better result, which is still not a very good result. We also saw that the selection of features and the use of GridSearchCv and Randomized did not have much use in changing the accuracy of the model.

The positive point was that the different scalings were able to make more significant changes in the results of the model. Since this dataset is not suitable for prediction, in the end we do not get very good results and we have to try more complex models for better results.

References

[1] Preprocessing

[2] Visualization

[3] Feature Selection

[4] Statistical Hypothesis Testing

[5] Different types of Statistical Hypothesis Testing

[6] Feature Scaling