Natural Language Processing and Loan Default

Philip Rundall

May 23, 2020

Abstract

Using Supervised and Unsupervised machine learning methods we extract vital information from an eight of a million news articles. Identifying relevant words in each category allows us to achieve an 88% accuracy rate. Testing subsets of information, our team compares using 250 relevant words to using an order of magnitude more. Similarly we look at the predictive loss by only feeding in the titles of the article. Our results illustrate the trade off of high performance and computational efficiency. Furthermore we compare k-means separation of articles with their labeled categories.

Part I

Natural Language Classification of News

1 Introduction

Natural Language Processing is the science of training Artificial Intelligence in Linguistics. Understanding, interpreting and generating insight from natural language is a key step in the furtherment of automation. Classification of various writings can expedite the work of lawyers, business analysts and researchers by assigning documents to certain buckets. We illustrate machine learning capability to solve this problem on a dataset provided by AWS.

We use the accuracy of four models in cross-validation as the metric of interest. Accuracy is insightful here because the groups are equally balanced. We also analyze the efficiency of these models only using the title, as opposed to the full article. The difference in accuracy between these two approaches is only 6.4%. A considerable decrease when the computational speed is increased by around half.

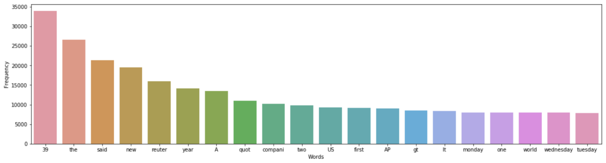
Clustering of news articles can expose valuable correlation to market sentiment and momentum. It can also allow us to see if the way articles are classified have similar features found by clustering. This is non-trivial as the elbow method identified an extra eight clusters that have no correspondence with the categories given.

2 Data

From an AWS database, our team sourced a data set with 127600 news articles. Each news article consists of a title (mean: 42 words), an article (mean: 193 words), and a class. There are four classes that correspond to the topic the article relates to: Sports, Business, Sci/Tech, or World. The classes are evenly distributed between the four categories.

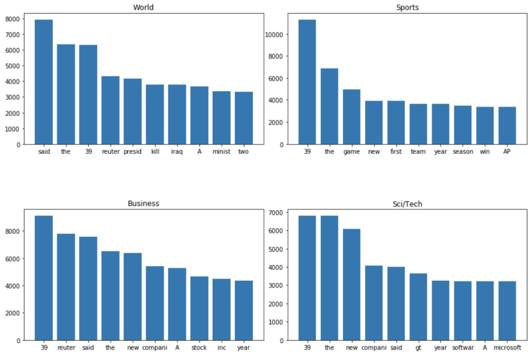
Removing all the punctuations and capitalization we next stem all the words. Stemming is a process that converts words into their root word i.e. killing, killed, kills all become kill. The next step in data processing consists of tokenizing the words into factor variables. The most significant words across all categories are seen in Figure 1.

Figure 1: Most Common Words



We must compare across categories to see which words are a significant indicator of an article belonging to a specific category. Certain stems like 39 and gt are in every category equally prevalent within the categories. These irrelevant variables can be accounted for in 2 ways: removal or regularization within the classification methods. Our team performs the latter.

Figure 2: Common Words by Category



3 Methods

3.1 Logistic Regression

Multinomial Logistic regression estimates a percentage probability for an observation to be in each class. The class that has the highest probability is the prediction. Due to the large amount of factor variables it is customary to add a $||1||_1$ regularization term to send non-informative variables' coefficients to zero.

3.2 LDA

Linear discriminant analysis creates linear combinations of features to divide classes to best explain the data. Here certain combinations of words could become new components giving better insight than the words alone.

3.3 QDA

Quadratic Discriminant Analysis is a variant of LDA where every class of observation has a corresponding covariance matrix that allows the algorithm to be divided into conic sections.

3.4 KNN

KNN in the context of this problem will calculate the euclidean distance from a new observation to a previous observation. If two articles have the same word that the dimension of the feature space will contribute zero to the distance. If a word appears in one but not the other, a length is added to the euclidean distance. By simple geometry the euclidean $distance = \sqrt{i}$ for i being the number of different words. The classes of the k nearest observations is the prediction.

3.5 K-Means

K-Means is a clustering algorithm that attempts to group all observations into "K" number of groups. It attempts to do this by minimizing the sum of all pair - wise squared Euclidean distances between all observations in a cluster.

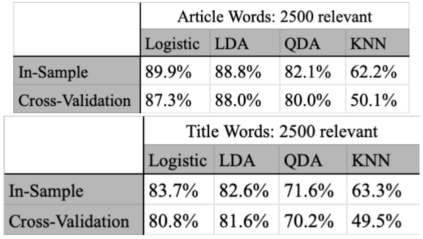
$$\underset{C_1,...,C_K}{Minimize}\left\{\sum_{k=1}^{K}\frac{1}{|C_k|}\sum_{i,i'\epsilon C_K}\sum_{j=1}^{p}(x_{ij}-x_{i'j})^2\right\}$$

4 Results

4.1 Supervised

Cross-Validation is used as a proxy for out of sample evaluation. We perform a 5-fold cross-validation on each model for consistent comparison. Due to the difference in word amount for titles compared to summaries, we run all methodologies on both to see if the same accuracy could be accomplished using the titles opposed to the summary, in turn saving a lot of computational efficiency.

Figure 3: Accuracy Scores



	Article Words: 2500 relevant			
	Logistic	LDA	QDA	KNN
In-Sample	89.9%	88.8%	82.1%	62.2%
Cross-Validation	87.3%	88.0%	80.0%	50.1%

	Title Words: 2500 relevant			
	Logistic	LDA	QDA	KNN
In-Sample	83.7%	82.6%	71.6%	63.3%
Cross-Validation	80.8%	81.6%	70.2%	49.5%

A marginal improvement of 6.4% gives a license to run the models on the Summary data. Between these two methods the performance ranking stays consistent, from best to worst LDA, Logistic Regression, QDA, and KNN. After reevaluating the methods using only the most popular 250 words our highest accuracy is in the low 60% proving the magnitude of common words needed.

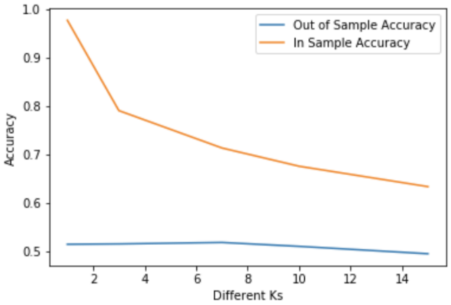
KNN stays consistent with its Cross Validated accuracy as k increases around .5. This is significantly better than random chance but not nearly as good as the other models.

Due to the poor performance of the two non-linear models we have shown that a linear fit for this problem is more accurate.

4.2 Unsupervised

There are a few ways to choose the number of clusters, K. First we attempted to fit 4 clusters due to

Figure 4: In-n-Out of Sample Accuracy of various Ks



there being 4 classifications in this dataset. Below are the results of how many observations from each classification were in each cluster.

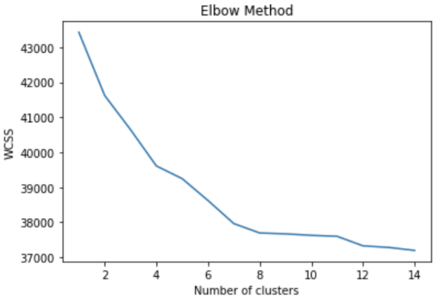
Table 1: Clusters Equalling Categories

Cluster \ Classification	1	2	3	4
1	45	36	39	31
2	359	294	373	350
3	553	371	316	282
4	943	1199	1172	1237

Most observations were in the fourth cluster, and it does not look like any classification necessarily matches with a cluster.

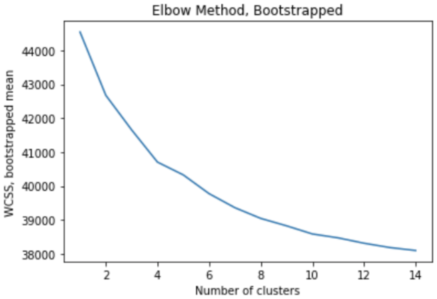
Next, we chose the number of clusters by looking at the average within cluster square distance. By plotting this against the number of clusters, we can determine how many clusters are optimal to our dataset. This is also called the "Elbow Method" due to the fact that we are looking for a kink in the graph to determine what the optimal number of clusters is.

Figure 5: Elbow method



By using the elbow method in Figue 5, 8 clusters is optimal for this dataset. To further validate this, we also bootstrapped the dataset and took the average within cluster square distance across bootstrapped samples. Below are the results

Figure 6: Bootstrapped Elbow Method



From bootstrapping, we can see that 12 clusters may fit our dataset. We fit both 8 and 12 clusters to our dataset to see if there were any interesting splits of classifications among clusters. However after analyzing 12 clusters some of the clusters contain as few as four observations, lending the data to fewer splits as not to be waned by outliers.

Table 2: Eight Clusters

Classification	1	2	3	4
Cluster				
1	45	36	39	31
2	122	82	182	130
3	175	186	165	186
4	807	1010	922	1017
5	13	25	33	23
6	122	159	120	96
7	354	168	173	141
8	262	234	266	276

After fitting 8 clusters to the data, it again does not look like any classifications necessarily coincide with particular clusters. Cluster 4 does have significantly more observations than the rest of the clusters though.

5 Conclusion

An accuracy of 88% using cross-validation shows Linear Discriminant Analysis is (very successful in classification of news articles. Logistic regression having a similar accuracy demonstrates this data is appropriately described by Linear models. Testing the use of 250 words compared to 2500 words the former reduces accuracy by 30% making the size of independent variables necessary. Similarly running the models on titles instead of articles produces a decrease of 6.4%.

One may consider this fault as a necessary trade off for computational efficiency.

Using K-means to cluster the data into 4 partitions, the same amount of classes yielded a scattered classification matrix. Showing the partitions generated by K-means do not align with the actual classification of the articles. Applying the elbow method the optimal amount of clusters suggested is 12, far more than the classes.

Future work in this field would consist of identifying correlation within global issues to the cluster of articles released to test any predictive power of a cluster i.e. predicting recession, increased volatility in the stock market or civil unrest.

Part II

Car Loan Default Using Regularization

Abstract

By employing data analysis methods, financial institutions build various models to optimize profitability and reduce losses. The following paper explores regularization methods to fit classification models on high-dimensional vehicle loan information. The elastic net method ultimately proves to be the best regularization method to classify defaults trained by this dataset.

6 Introduction

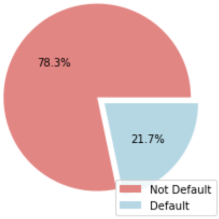
Determining if a customer will default on a loan is important for financial institutions to analyze the risk in their portfolio of loans. With the rise of big data, there are many variables that can be taken into account to determine if a customer will default on their loan. However, not all of these variables are useful. This is where regularization comes into play. Regularization is a technique in machine learning to reduce the dimensionality of a model, hopefully improving its accuracy.

To test all of this, we fit multiple classification algorithms to a high dimensional dataset for vehicle loan defaults. Some of these algorithms include regularization while one does not. We also fit a regularization algorithm called Principal Component Analysis to the dataset, strictly to try and reduce its dimensionality.

7 Data

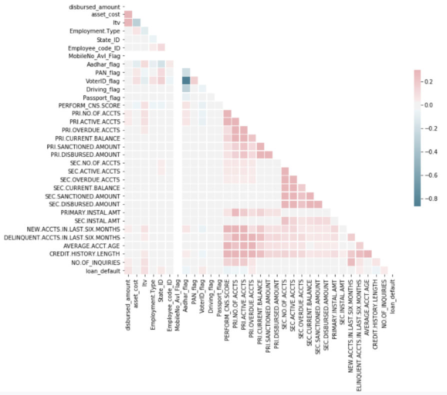
The dataset for analysis targets whether vehicle loans will default. The dataset includes about one-fifth default loans.

Figure 7: Distribution of Loans



The following variables and relative correlations are included in the dataset.

Figure 8: Correlation Plot



8 Techniques

8.1 Lasso

Lasso, least absolute shrinkage and selection operator, operates under the following formula:

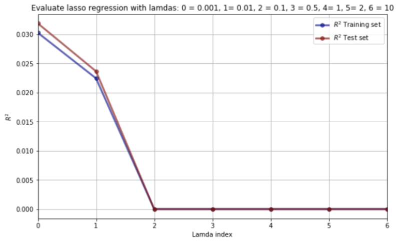
$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

The lasso method can reduce the number of variables for a linear model through L1 regularization, which applies a penalty equivalent absolute value of the coefficients. A tuning parameter, λ, controls the penalty, or “shrinkage”, where increasing λ, increases the bias of the trained model.

We attempt several methods and evaluate R^2 the at each λ. It seems that the most optimal initial λ value is 0.001 or a lower number. However, the

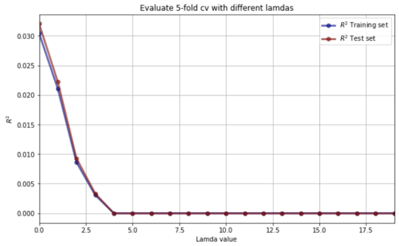
R^2 value does seem quite small at this most optimal value.

Figure 9: Varying llambdas



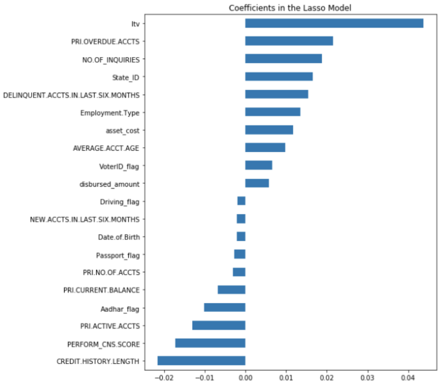
Next, we evaluate the 5-fold cross-validation values. Again it seems that the optimal λ value is 0.001 or a lower number. Additionally, the graphs for both the training and testing set seem to overlap, possibly indicating the absence of over bias in the training model.

Figure 10: Varying llambdas using cross validation



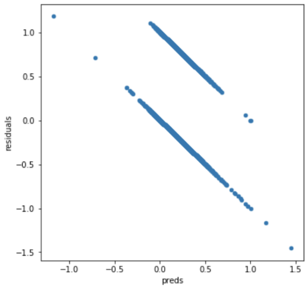
The Lasso method with $\lambda = .001$ is given by mean squared error of 0.16542. Ultimately, the Lasso method is able to eliminate 8 variables and retain 26 variables. The following plot represents the remaining coefficients:

Figure 11: Remaqining Coefficients



As mentioned above, the R^2 for the model is quite low. Moreover, the following residuals plot indicates that the linear model is not suitable for the training (and testing) dataset.

Figure 12: Residuals Plot



8.2 Ridge

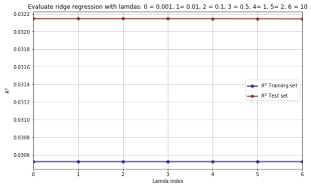
Ridge regression follows L2 regularization, which applies the L2 penalty, equivalent to the square of the magnitude of the coefficients. The formula is given by,

$$\beta^{ridge} = \frac{argmin}{\beta \epsilon \mathbb{R}} ||Y - X\beta||_2^2 + \lambda ||\beta||_2^2$$

Unlike the Lasso model, the Ridge method does not result in sparse models.

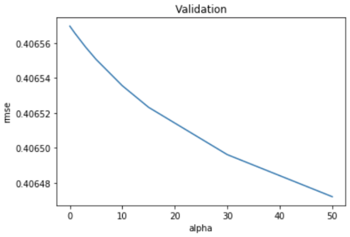
First, we experiment with changing the lambda values. However, it does not seem that changing the alpha values has any effect on increasing the R^2 values of the model, thus questioning the validity of a Ridge model. Identically employing a five-fold cross validation method, althering values does not affect the values.

Figure 13: Evaluating Ridge



Overall, the Ridge method does not seem suitable for the model, but significantly increasing the value may be helpful for further studies. The minimum RMSE value that can be obtained for the training set is 0.4065.

Figure 14: Validation



8.3 Elastic Net

Elastic net is a form of regularization that combines L1 and L2 regularization. It has a λ parameter just as lasso and ridge, but also has another hyperparameter, α, which is a weight between the L1 and L2 regularization.

Because our dataset is a classification problem, we use elastic net to regularize logistic regression. To determine the alpha and lambda parameters, we used gridsearch and crossfold validation. This gives us a of 0.0001 and an α of 0.15. Elastic net set all of the variables in our dataset equal to 0 except one, LTV.]

We assessed the accuracy of our model on our training and testing dataset.

Table 3: Evaluation

	Train	Test
Accuracy	0.784	0.781

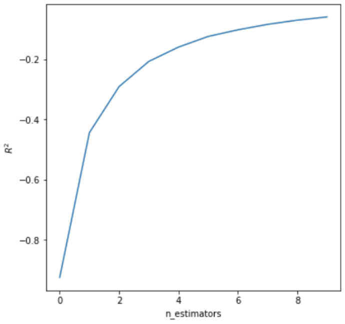
We also used cross fold validation to further assess the accuracy of our model. Which has a consistent accuracy across all folds of 0.7835.

By using cross fold validation, we can see that we are not overfitting our dataset and we have a decent accuracy, meaning we do not have too much bias or variance. Finally, we used the ROC curve to measure our model's goodness of fit because this is a binary classification problem. Our AUC ROC score for elastic net was 0.57, which is not very high.

8.4 Random Forest

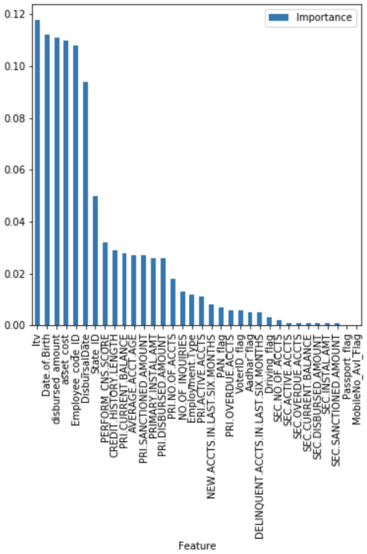
The random forest method utilizes bagging and feature randomness to implement classification. The random forest method is able to produce the following results: a MSE of 0.2204, RMSE of 0.4694, and accuracy of 0.7796. The accuracy increases with increased number of trees, but is marginally less effective after about 3 estimators into the model.

Figure 15: Varying number of Trees



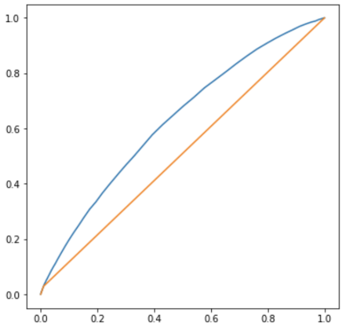
In addition, the Random Forest method can select features based on importance, as shown in the following image.

Figure 16: Importance of Variables



The ROC curve shows the relationship between the sensitivity and specificity of a model. Since the curve of this model comes quite close to the 45-degree diagonal, the validity of the model is less than a more accurate model.

Figure 17: ROC



8.5 Support Vector Machines

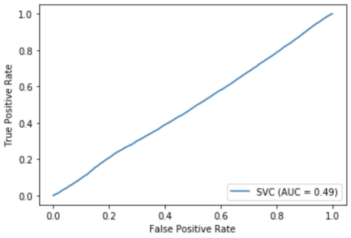
The last classification model we fit to our dataset was SVM. SVM works by trying to fit a hyperplane between the classifications in our dataset. We fit both linear and polynomial SVM's to our dataset. The polynomial fit had a higher accuracy so we proceeded with it. Below are the training and testing accuracies for our polynomial fit SVM.

Table 4: In-n-Out of Sample Accuracy

Data	Train	Test
Accuracy	0.7835	0.7813

As we can see, there is not much of a drop off in accuracy between training and testing dataset, meaning we are not overfitting our data. To further validate this we again used crossfold validation. Cross Fold validation yields an accuracy of .783 across all five folds. Again, there is not much variance in accuracy meaning we are not overfitting the data. Finally, we fit an AUC ROC curve to validate our model because this is a classification problem.

Figure 18: SVM ROC

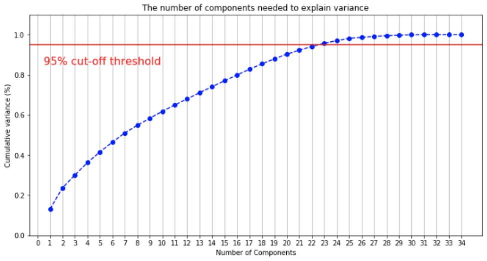


Our AUC ROC score is 0.49 which is worse than random guessing. This means that our model poorly distinguishes between defaulting and not defaulting observations. By this metric, SVM performs very poorly on the dataset.

8.6 Principal Component Analysis

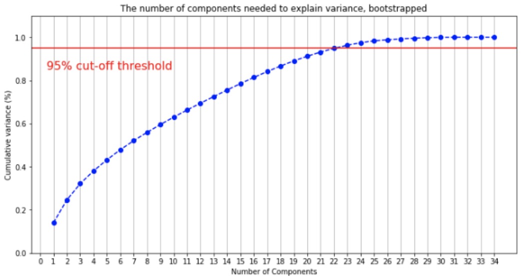
PCA is a form of dimension reduction that works by creating new variables which are linear combinations of the original variables but with no overlap of information. These new variables are called components. The number of components can be chosen by determining how much variance in the dataset is accounted for with the number of components selected. Usually, the optimal number of components accounts for 95% of the variance in a dataset.

Figure 19: Component Analysis



As we can see from above, about 23 components accounts for 95% of the variance in the dataset. To test the robustness of this result, we also bootstrapped our dataset and took the average variance of fitting each number of components across the bootstrapped samples. Below are the results.

Figure 20: Bootstrapped Component Analysis



Even after bootstrapping, 23 is the number of components needed to explain the variance in the dataset.

9 Conclusion

After fitting all of these classification and regularization models, we are unsure if this dataset is suitable for regularization. When using PCA to reduce the dimension of our dataset, the optimal number of components was 23. Our highest performing model was the logistic regression with elastic net regularization which set all of our variables equal to 0 except for LTV. Even though the elastic net model was our best performing model, it was not by a large margin. All models had similar accuracies and low AUC ROC scores.

While none of our models stood out in terms of performance, we believe that they all had a good bias-variance trade off. All of the models had consistent accuracy scores when using the training and testing datasets, as well as using crossfold validation.

References

[1] https://course.fast.ai/datasets

[2] https://www.kaggle.com/avikpaul4u/vehicle-loan-default-prediction