Springboard Introduction to Data Science Capstone Project

Machine Learning Classifier for Detecting Email Spams

By

Savita Shrivastava

26 Dec 2017

Introduction

Email is one of the most efficient and effective mode of communication with one another. Today a serious problem for web users and web services is caused by inflow of large number of spam emails. Spam emails are called the unwanted emails or unsolicited emails or bad emails which user receives without any prior information of the sender. Spam emails are usually trying to get the recipient to buy some product or services, spreading viruses, advertisements, for fraud in banking and for phishing. An estimation shows that close to 80% of all the emails are spam.

A spam filter is a software that keeps spam emails from entering the in-box. Hence, it predicts if an email is considered spam or no-spam, and decides if the email should be displayed in the in-box or be junked.

Existing spam filtering techniques use classification. Classification is a type of data analysis that extracts models describing important data classes or concepts. Classification mainly consists of two steps. First is the learning step: where a classification model is constructed and second is the classification step: in this step the extracted model is used to predict the class labels for new data or unknown data depending on the learning step. Machine learning algorithms are used for classification of objects of different classes. Such algorithms have proved to be efficient in classifying emails as spam or not spam.

Objective

The objective of this project is to build a model using a machine learning method which can predict the outcome if an email is spam or no-spam and based on that the spam emails can be filtered out. The project will try to give answers to the following questions :

- How can we construct a spam filter, given the data set?
- What factors alter the probability of an email being a spam-email?
- How to create an accurate model that can predict if an email is spam?

- What is the risk of the model making false predictions?

Data Acquisition

The Spambase data set was acquired from UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/spambase) will be used for this project.

Data Exploration

The Spambase dataset contains:

- Number of Instances: 4601
- Number of attributes: 58
- Number of missing data points: None
- The last column of 'spambase.data' named 'spam' denotes whether the email was considered spam (1) or not spam (0).
- Most of the attributes indicate whether a particular word or character was frequently occuring in the email. The run-length attributes (55-57) measure the length of sequences of consecutive capital letters. The definitions of the attributes are described in table below:

Table 1 : Attributes in Spambase Data

Attribute Column Number	Attribute name	Attribute Type	Attribute Description
1 to 48	word_freq_WORD	continuous real [0,100]	percentage of words in the email that match WORD
49 to 54	char_freq_CHAR	continuous real [0,100]	percentage of characters in the email that match CHAR
55	capital_run_length_average	continuous real [0,100]	average length of uninterrupted sequences of capital letters
56	capital_run_length_longest	continuous integer [1,...]	length of longest uninterrupted sequence of

			capital letters
57	capital_run_length_total	continuous integer [1,...]	total number of capital letters in the email
58	spam	nominal {0,1}	denotes whether the email was considered spam (1) or not (0)

Outcome or dependent variable will be 'spam' and all other attributes from column 1 to 57 will be independent variables. Below is the summary of spam variable:

Table 2: Summary of Outcome Variable 'spam'

Spam	Frequency	Percent
0 (not spam)	2788	60%
1 (spam)	1813	39%

Data Analysis

The Logistic Regression algorithm was used to classify if an email is spam or not a spam.

What is Logistic Regression?

Logistic regression is a simple classification algorithm to analyze a dataset in which there are one or more independent variables that determine an outcome. In logistic regression the outcome or dependent variable is coded a 1 (TRUE) or 0 (FALSE).

Logit Transformation

The goal of logistic regression is to find the best fitting model to describe the relationship between the dependent variable (response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression generates the coefficients (and its standard errors and significance levels) of a formula to predict a *logit transformation* of the probability of an email being spam:

$$logit(p) = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \ldots + b_kX_k$$

where p is the probability of presence of characteristic of interest (an email being spam). The logit transformation is defined as the logged odds:

$$odds = \frac{p}{1-p} = \frac{probability\ of\ presence\ of\ characteristic}{probability\ of\ absence\ of\ characteristic}$$

And *logit transformation* of probability p is:

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Rather than choosing parameters that minimize the sum of squared errors (like in ordinary regression), estimation in logistic regression chooses parameters that maximize the likelihood of observing the sample values.

Odds-ratio

Equation 2 shows that if the probability of the outcome variable spam is between [0,1], the odds will be non-negative. If *odds > 1* the probability of an email being spam is greater than the probability of an email being no-spam.

Definitions:

Dependent variable

The variable whose values you want to predict. The dependent variable must be binary or dichotomous, and should only contain data coded as 0 or 1.

Independent variables

The independent variables are the variables which are expected to influence the dependent variable.

Data Cleaning

The data cleaning was steps includes as described below:

- Find and remove any missing values
- Change the name of the below attributes which have special characters in their name as below:
    a. char_freq_; to char_freq_semic

b. char_freq_(to char_freq_openp
c. char_freq_[to char_freq_openb
d. char_freq_! to char_freq_excl
e. char_freq_$ to char_freq_dollar
f. char_freq_# to char_freq_pound

- Since the column names were not present in the data, the column names were added to the data after above step.

Building the Predictive Model and Performance Evaluation

Since the outcome variable spam has binary levels (0 or 1), logistic regression was used to build the predictive model using all of the independent variables (attributes 1 to 57). The data was divided into a training and testing set with 75/25 ratio. The set.seed variable was used to make sure the dependent variable was well balanced in both the training and testing sets. Once the model was developed with the training data, it was subsequently tested on the test data to determine its accuracy.

Performance Evaluation Parameters

The performance evaluation was done using classification matrix as shown in the Table 3:

Table 3: Classification Matrix:

	Predicted = 0	**Predicted = 1**
Actual = 0	True Negatives (TN)	False Positive (FP)
Actual = 1	False Negative (FN)	True Positive (TP)

The description of parameters are as follows:

- True Positive (TP): Spam emails are correctly predicted as spams
- True Negatives (TN) : No-spam emails are correctly predicted as no-spam emails
- False Positive (FP) : No-spam emails are incorrectly predicted as spam emails
- False Negative (FN) : Spam emails are incorrectly predicted as no-spam emails

- Accuracy : (True Negatives (TN) + True Positive (TP)) / Total number of observations
- Sensitivity (True Positive Rate) = True Positive (TP) / (True Positive (TP) + False Negative (FN))
- Specificity (False Positive Rate) = True Negatives (TN) / True Negatives (TN) + False Positive (FP))
- Error rate = (False Positive (FP) + False Negative (FN)) / Total number of observations

The below coefficient output of training model shows the independent variables which are significantly affecting the model and outcome variable. The negative estimate value of some of the significant variables such as 'word_freq_george', 'word_freq_hp', 'word_freq_hpl' and 'word_freq_edu' clearly showing that these are no-spam email related words as opposite to some of the high positive estimate values such as 'word_freq_free', 'word_freq_000', 'word_freq_our', 'word_freq_remove', 'word_freq_re' and 'char_freq_dollar'.

```
------------------------------------------------------------------------------------------------------------------------
Coefficients:
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                -2.006e+00  2.112e-01  -9.499  < 2e-16 ***
word_freq_make             -4.501e-01  2.887e-01  -1.559 0.119051
word_freq_address          -1.516e-01  8.264e-02  -1.834 0.066600 .
word_freq_all               6.749e-02  1.459e-01   0.462 0.643753
word_freq_3d                3.616e+00  2.249e+00   1.607 0.107957
word_freq_our               8.113e-01  1.360e-01   5.968 2.41e-09 ***
word_freq_over              1.329e+00  3.193e-01   4.164 3.13e-05 ***
word_freq_remove            2.998e+00  5.080e-01   5.902 3.59e-09 ***
word_freq_internet          4.624e-01  1.574e-01   2.939 0.003295 **
word_freq_order             3.857e-01  3.787e-01   1.018 0.308453
word_freq_mail             -2.065e-02  1.241e-01  -0.166 0.867867
word_freq_receive          -2.970e-01  3.590e-01  -0.827 0.408124
word_freq_will             -1.279e-01  9.024e-02  -1.418 0.156268
word_freq_people           -5.160e-02  3.063e-01  -0.168 0.866241
word_freq_report            5.376e-02  1.470e-01   0.366 0.714624
word_freq_addresses         8.097e-01  7.600e-01   1.065 0.286692
word_freq_free              8.309e-01  1.753e-01   4.741 2.13e-06 ***
```

```
word_freq_business          1.083e+00  2.839e-01   3.815 0.000136 ***
word_freq_email             1.062e-01  1.606e-01   0.662 0.508269
word_freq_you               6.335e-02  4.493e-02   1.410 0.158555
word_freq_credit            1.335e+00  9.340e-01   1.429 0.153040
word_freq_your              1.575e-01  6.644e-02   2.370 0.017791 *
word_freq_font             -5.869e-02  2.086e-01  -0.281 0.778427
word_freq_000               3.237e+00  7.330e-01   4.417 1.00e-05 ***
word_freq_money             3.038e-01  1.343e-01   2.263 0.023664 *
word_freq_hp               -2.046e+00  3.884e-01  -5.269 1.37e-07 ***
word_freq_hpl              -1.004e+00  5.307e-01  -1.893 0.058418 .
word_freq_george           -1.850e+01  3.980e+00  -4.647 3.37e-06 ***
word_freq_650               6.413e-01  2.932e-01   2.187 0.028737 *
word_freq_lab              -2.350e+00  1.742e+00  -1.349 0.177233
word_freq_labs             -4.709e-01  5.470e-01  -0.861 0.389270
word_freq_telnet           -3.886e+00  3.131e+00  -1.241 0.214524
word_freq_857               1.940e+00  3.710e+00   0.523 0.601048
word_freq_data             -6.652e-01  3.640e-01  -1.827 0.067645 .
word_freq_415               1.641e-01  1.708e+00   0.096 0.923455
word_freq_85               -2.705e+00  1.022e+00  -2.647 0.008110 **
word_freq_technology        7.257e-01  3.831e-01   1.894 0.058172 .
word_freq_1999             -3.035e-01  2.522e-01  -1.203 0.228963
word_freq_parts            -5.981e-01  5.995e-01  -0.998 0.318437
word_freq_pm               -7.256e-01  5.117e-01  -1.418 0.156183
word_freq_direct           -3.735e-01  4.439e-01  -0.842 0.400063
word_freq_cs               -5.676e+02  2.071e+04  -0.027 0.978141
word_freq_meeting          -2.270e+00  8.923e-01  -2.544 0.010973 *
word_freq_original         -1.474e+00  1.173e+00  -1.257 0.208771
word_freq_project          -1.641e+00  7.555e-01  -2.172 0.029857 *
word_freq_re               -6.978e-01  1.627e-01  -4.290 1.79e-05 ***
word_freq_edu              -1.409e+00  2.963e-01  -4.756 1.98e-06 ***
word_freq_table            -4.017e+00  2.575e+00  -1.560 0.118749
word_freq_conference       -6.014e+00  3.249e+00  -1.851 0.064132 .
char_freq_semic            -1.085e+00  5.476e-01  -1.981 0.047570 *
char_freq_openp            -3.150e-01  3.654e-01  -0.862 0.388588
char_freq_openb            -2.397e+00  1.725e+00  -1.390 0.164645
char_freq_excl              7.186e-01  1.657e-01   4.337 1.44e-05 ***
char_freq_dollar            5.446e+00  9.474e-01   5.748 9.01e-09 ***
char_freq_pound             2.187e+00  1.645e+00   1.330 0.183522
```

```
capital_run_length_average  2.839e-01  7.102e-02   3.997 6.42e-05 ***
capital_run_length_longest -1.043e-03  3.413e-03  -0.306 0.759940
capital_run_length_total    1.515e-03  3.196e-04   4.739 2.15e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4329.5  on 3220  degrees of freedom
Residual deviance: 1159.7  on 3163  degrees of freedom
AIC: 1275.7

Number of Fisher Scoring iterations: 23
```

Selecting a Threshold for Filter

By using a threshold value, the outcome of logistic regression which are probabilities can be converted into predictions. If the probability of an email being spam is greater than the threshold, then the prediction is that the email is spam. If it's below, then prediction is that the email is not a spam. Selecting a right threshold is often challenging. A Receiving Operator Characteristic (ROC) curve can help in deciding which value of the threshold could be best. The ideal spam filter should detect smaller number of no-spam emails as spam emails i.e., there should be lower number of false positives in our data to have higher specificity. The ROC curve from training data set was generated as shown in Figure 1 to get a threshold cutoff. The ROC curve is suggesting to have a threshold cutoff close to 1. Therefore threshold cutoff 0.9999 was selected. With threshold cutoff 0.9999, specificity was higher and sensitivity was lower. The Classification Matrix of training model is shown in Table 4 and sensitivity and specificity in Table 5.

Figure 1: ROC Curve to Select Threshold for Prediction.

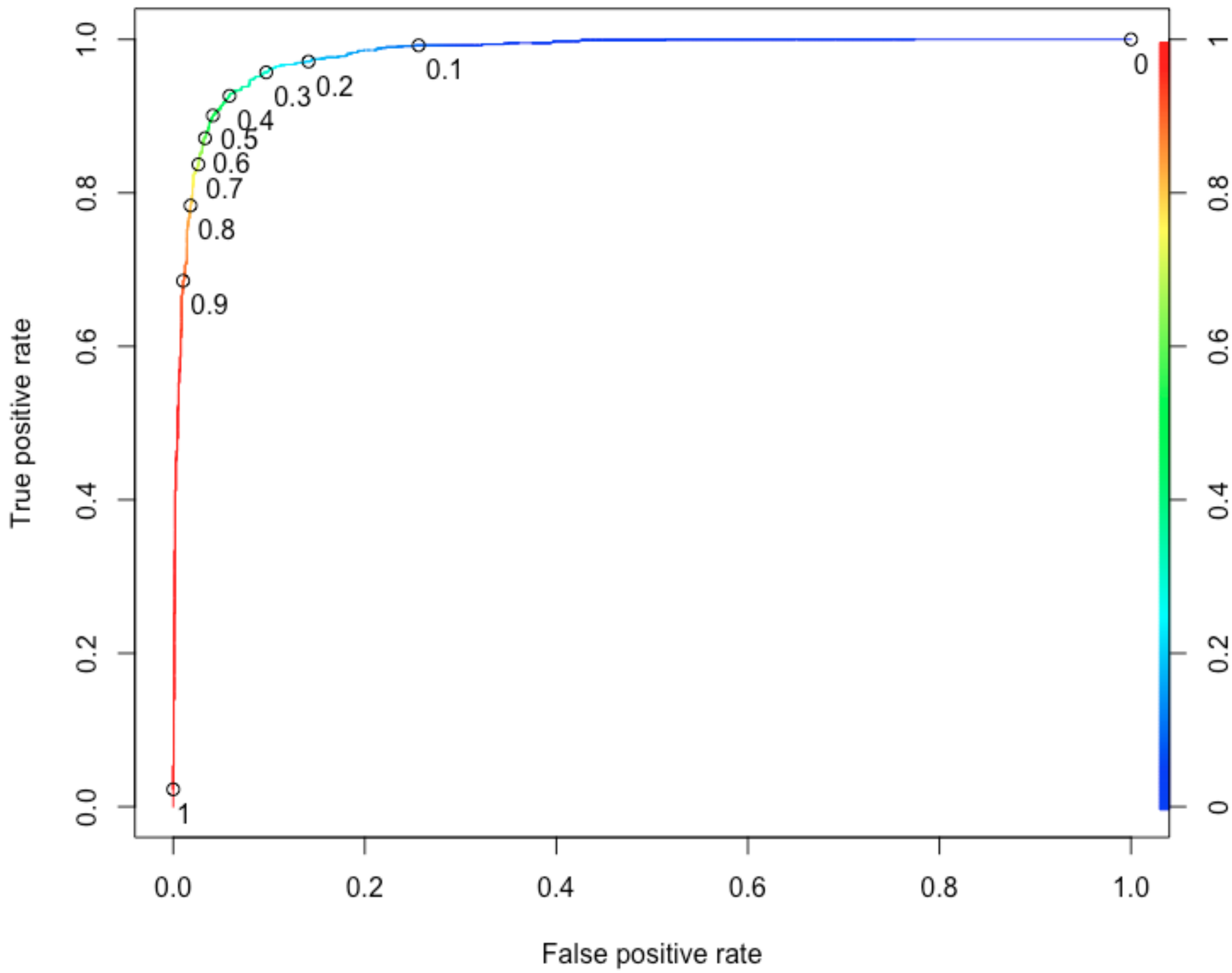

Table 4: Classification Matrix of Training Model using Threshold 0.99

	Predicted = 0	**Predicted = 1**
Actual = 0	1933	7
Actual = 1	697	584

Table 5: Accuracy, Sensitivity and Specificity of Training Model using Threshold 0.99

Accuracy	Sensitivity	Specificity
78.14%	45.6%	99.6%

Area Under the ROC (AUROC)

Another measurement to validate the accuracy of the training model could be the area under the ROC (AUROC). The metric for AUROC ranges from 0.50 to 1.00, and value above 0.80 indicate that the model does a good job in discriminating spam and no-spam emails. The AUROC for this training model is 0.98 which is above then 0.80.

Evaluating the Model on Testing Data Set

The testing data set contains 1380 observations and was used to evaluate the model. The below classification matrix was created to determine the accuracy of the model. The threshold value of 0.5 was used to create the classification matrix.

Table 6: Classification Matrix using Testing Data Set to Evaluate the Model

	Predicted = 0	Predicted = 1
Actual = 0	803	48
Actual = 1	45	484

Table 7: Accuracy, Sensitivity and Specificity using Testing Data Set to Evaluate the Model

Accuracy	Sensitivity	Specificity
93.26%	91.5%	94.35%

The accuracy of model is 93.26% to predict if an email is spam or not a spam with overall error rate of 6.7%. The model exhibit high sensitivity as well as high specificity for predicting spam and no-spam emails and exhibit that this model is a better predictor than the baseline method where accuracy was 60.8%

Conclusion

Email spam classification has received a tremendous attention by majority of the people as it helps to identify the unwanted information and threats. Therefore, most of the researchers pay attention in finding the best classifier for detecting spam emails. The results demonstrate that logistic regression model has 93.26 % accuracy in spam detection than baseline methods 60.8% accuracy. The careful selection of attributes in building model can increase the accuracy of the model.