
Linear Regression for Binary Classification

Adam Payzant
101082175

Aidan Crowther
100980915

Alex Cornish
101053176

Abstract

This investigative assignment was performed with the intended purpose of exploring the implementation of linear classification and the effect of various variables on its performance. The resultant logistical classifier is to then be modified through the use of hyperparameters as well as modifying the data on which it is to train so as to explore the impact of various variables on the performance of the classifier. Parameters such as the learning rate and the number of iterations were observed to have a significant impact on the performance of the implemented classifier. In addition the elimination of largely biased and non-descriptive features was observed to have a significant impact on the performance and accuracy of the model by reducing the number of consequent calculations, as well as by removing noisy data from the classifier.

1 Introduction

The first task to be accomplished within this project is to perform some basic analysis of the data contained within the datasets supplied in order to observe patterns of feature distribution, and using this data isolate potential targets for isolation when performing classification using fewer features. Following this primary analysis stage, a linear classification model is to be constructed, utilizing logistic regression on the supplied data. This models performance is to then be observed by the metrics of classification speed and accuracy, and is to be altered through the use of various modification techniques to observe their effects on the measured performance. These include the modification of input features as previously mentioned, as well as the use of various hyperparameters for the model, such as the learning rate and number of iterations to train on each subset of the data. The impacts of these modifications were observed to have a significant impact on the models performance, such as the impact of a large learning rate on the dataset with more features resulting in worse performance while a larger learning rate resulted in improved performance for the dataset with fewer features. It was also observed that more iterations are generally better for training, however, this does not hold completely true, as training on the same dataset began to negatively impact the performance past a certain point. This phenomenon is believed to be related to the overfitting of the model on these individual datasets, resulting in poor general performance. This is all to be performed using two provided datasets for analysis, each with a different number of features and a binary classification, providing information on bankruptcy of certain individuals and the diagnosis of hepatitis.

31 2 Datasets

32 Provided for this evaluation are two datasets with a variety of features and a resultant binary
33 classification for each set of these features. The first of these datasets provides various statistics on
34 and about patients who either were or were not diagnosed with hepatitis. These include metrics such
35 as the patients age, their pre-existing conditions, and various measurements of their general health.
36 This dataset consists of 19 features, with 143 different samples, and a general distribution of positive
37 to negative diagnoses of 116 positive to 27 negative. This dataset is relatively poorly distributed,
38 however it does allow for significant linking between strongly correlated features and their final
39 classification.

40
41 The second dataset focuses on the occurrence of bankruptcy among a group of individuals given a
42 number of unspecified features. This dataset consists of 64 features, and a total of 453 samples. The
43 distribution of bankruptcies in this dataset is far more balanced than in the hepatitis dataset, with
44 203 bankruptcies and 250 non-bankruptcies. This dataset unlike the hepatitis dataset features a more
45 balanced distribution of its results, however, it is also notable in that there are a number of features
46 that contain data that is completely unrelated to the outcome, or at worst completely skewed to one
47 side in a way that would potentially have negative effects on the performance of the final model. To
48 this end, feature manipulation is expected to have a significant impact on the performance of this
49 model, by removing the seemingly irrelevant data the effect of feature manipulation can be observed
50 during this investigation.

51 3 Results

52 The linear classification model implemented for this report has performed surprisingly well in
53 classifying for both datasets. Initial tests using small, static, hyperparameter values provided
54 troubling results, with low accuracies; however, this was due to the use of fixed hyperparameters for
55 the task of building a functional model, as these values allowed the model to iterate quickly. Upon
56 constructing a model that appeared to be learning, despite its abysmal performance, the K-Fold
57 process was then modified to run through multiple iterations using various values for the learning rate
58 and number of iterations. Doing this it was evident how much of an impact each parameter had on
59 the model. Iterations were tested running from 10 times per sample to 5000 times per sample, with
60 major impacts on the runtime. Iterations were fast to run when below 500, however runtime rapidly
61 became unbearable when running over 5000 iterations. Thankfully, the runtime growth was linear in
62 complexity, and as such didn't become absurdly long. The results of varying the iterations showed
63 that 1000 iterations resulted in the best performance for both datasets; this was a surprising result due
64 to the sample sizes between the datasets varying by a wide degree. Although the results for this
65 parameter converged on the same number of iterations for both datasets, it is believed that the cause of
66 this may be more related to insufficient variance between the number of iterations tested, as the next
67 lowest was 500 and the next highest was 5000. These range samples were selected to minimize time
68 spent waiting on results, however it may be warranted to further investigate this parameter to ideally
69 learn data from each sample set. As expected, excessively small numbers of iterations resulted in
70 poor accuracy due to underfitting the data, while excessively large numbers resulted in poor ac-
71 curacy due to overfitting, where the model was specializing on the sample and not the general problem.

72

73 The other parameter that was tested was the learning rate of the model; of which rates from 0.01 to
74 0.8 were tested. Unlike with the iterations, varying the learning rate did not affect runtime of the
75 model during training, and as such was easier to test with larger variance in its values. This parameter
76 has provided interesting results into the learning of the resultant linear classification model, as the
77 learning rates that provided the best performance were not affected by the number of features as
78 much as by the range of values within these features. It is now apparent as to why this would be
79 the case, since the bias which is modified by these steps is directly applied to the input features. It
80 was thought that the larger feature set of the bankruptcy dataset would result in larger step values
81 performing better, however it has become apparent that the local minimums found for these samples
82 are narrower than imagined. As a result of this testing, the results for the optimization of learning rate
83 becomes intuitively clear with this further understanding. The rates upon which the model performed
84 best for each dataset were 0.01 and 0.1 for the bankruptcy and hepatitis datasets respectively. As with
85 the number of iterations, fixed values were tested for these learning rates, and as a result it may be
86 possible that these values are not in fact ideal for the sample sets. Despite this, these values were able
87 to provide very high accuracy for the datasets and as such are deemed to be sufficiently performant
88 for our purposes.

90 On running this analysis of hyperparameter values, it was possible to achieve classification accuracies
91 of 83.86% on the bankruptcy dataset, using a step size of 0.01 over 1000 iterations per sample; and
92 86.57% on the hepatitis dataset, using a step size of 0.1 over 1000 iterations. These accuracies are
93 significantly higher than random for both datasets, and perform better than always predicting the
94 same result even on the hepatitis dataset, which as was mentioned has heavily skewed results. This
95 lends credence to the likelihood that on the hepatitis dataset, the model has not overfitted to always
96 return the same result, where as for the bankruptcy dataset it is relatively clear that the model has not
97 become skewed in its classification.

99 As there is a lower number of attributes in the dataset for hepatitis.csv, the changes made by
100 pruning the dataset of columns that diminish the accuracy of the prediction can be seen more
101 easily. This generally occurs when the distribution of the values of a column are similar regardless
102 of if the class label is 0 or 1, meaning that the column itself does not contribute valuably to the
103 prediction. For example, by removing 5 out of 19 columns from hepatitis (bilirubin ,albumin
104 ,protime ,steroid ,and ascites), an improvement in the accuracy of predictions from around 83% to
105 88.7%, whilst this improvement is not too significant, any improvement in the resulting accuracy
106 is good. However, the same cannot be said of bankruptcy.csv, with its 64 attributes, which makes
107 improving the accuracy by the most logical method rather difficult. In this instance, the method
108 used to improve the accuracy of predictions was to run the initialisation of the model, then
109 remove one attribute at a time, perform the testing, and find the resulting accuracy of that set
110 of testing with the missing attribute, and then move on to the next column. Once that round of
111 testing was complete, the highest resulting accuracy was compared against the original accuracy
112 and if it was deemed an improvement, that attribute was removed more permanently and the
113 process was repeated until no more accurate model than the current one could be found. A
114 problem arose in this method when removing one attribute at a time yielded no improvement to
115 the bankruptcy model with the accuracy of most attributes being removed being about the same,
116 which would indicate that all of them are either equally important or equally unimportant to the model.

118 4 Discussion and Conclusion

119 The result of this investigation has been a successful linear classifier implementation, while also
120 providing a strong learning opportunity into the function and operation of linear classifiers as
121 well as logistic regression. The observations of particular note are the impact of paramaters and
122 feature pruning on classifier accuracy and performance. It was observed that iterating over each
123 sample has performed better when iterations do not exceed 10x the sample size, while iteration
124 ranges equal to or less than the sample size show similar results. Surprisingly, the variance in
125 accuracies between these ranges is relatively marginal, which would seem to imply diminishing re-
126 turns with larger iteration counts, especially when considering the risk of overfitting in these situations.

128 As for learning rate, it has become apparent that learning rate is heavily dependant on the actual
129 data contained within each feature set rather than on the features themselves. These results are also
130 enforced by the observation of performance of the model when changing the learning rate. By far the
131 largest variance observed in model performance has been as a result of varying the learning rate, with
132 significant impact on final accuracy.

133

134 The usefulness of pruning seems to be heavily dependent on the data itself, mainly regarding the
135 number of attributes or the values of the dataset itself as for some sets, such as hepatitis.csv, it is
136 obvious using the method used which attributes when removed improve the accuracy of predictions
137 by a noticable amount; however, for some datasets, such as bankruptcy.csv, such things are not
138 feasible due to the number of attributes and the very limited impact that their absence has on the
139 accuracy of the model. In retrospect, it might work better by starting with the attributes that when
140 removed lower the accuracy of predictions and start by adding the other ones back in.

141

142 As a result of this investigation, we have come to learn more on the subject of linear classification,
143 and the importance of features and their distribution on the final classifier performance. We have seen
144 the impact of hyperparameters on the learning of a model, and have been directed to possible avenues
145 of improvement for the future. This would include further analysis into iteration count optimization,
146 as overfitting is the major risk of large numbers of iterations, while the actual poerformance increase
147 from them is poor. As a result finding a more ideal ratio between the sample size and number of
148 iterations could prove fruitful. Similarly, further investigation into optimizing the learning rate may
149 be ideal, as it was determined that this parameter is directly affected by the data contained within
150 the dataset, perhaps it may be possible to determine this value as a function of the range of values
151 contained. It may also be an interesting concept to explore the use of a changing learning rate,
152 allowing the gradient descent algorithm to better fit into narrow minima.

153

154 Furthermore, some experimentation into pruning to find better ways to consistently determine what
155 attributes improve or worsen the accuracy of predictions would be interesting and especially useful in
156 some cases where certain attributes cloudy the data and make accurate prediction more difficult.

157

158 **5 Statement of Contributions**

159 Alex Cornish: Ones and zeros table modifications, data pruning, k-fold implementation.

160 Adam Payzant: Wrote the initial classifier implementation and k-fold 2

161 Aidan Crowther: Generated graphics and corrected classifier. Wrote report

162 **6 Appendix**

163 https://github.com/AdamPayzant/comp4900_mp1/tree/master/plots