Financial Analytics Project

Part 1 Report

Computing the feature importance scores

for a clustered malware dataset.

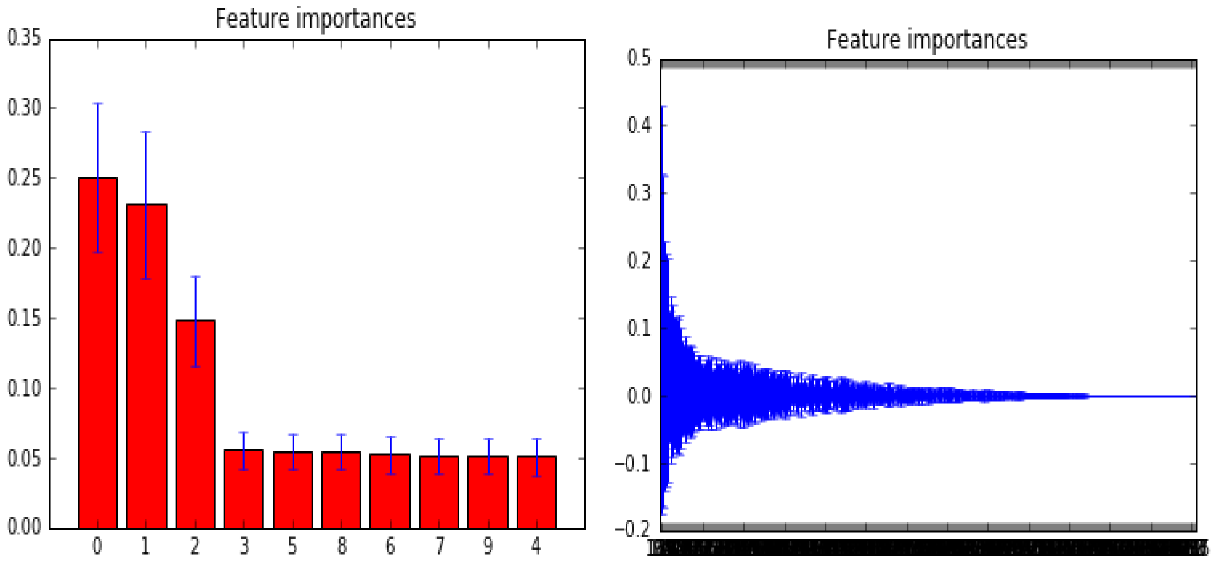
Ezeanaka Kingsley Uchechukwu

Introduction

In the area of machine learning and statistics, feature selection is a term commonly used to refer to the process of selecting a subset of relevant features, which could be variables or predictors, for using when developing a training model. The process of grouping the entire set of features in some order of relevance is termed **feature ranking.** The values associated with each rank are called feature importance values or feature ranking values. The idea is based on the fact that datasets are bound to contain many features that are either redundant or irrelevant, and thus can be removed without incurring much of information[1].

Feature selection techniques simplify models for easier interpretation by researchers and users, reduce model training times, avoid unnecessary complications associated with dimensionality and also to enhance generalization of dataset by reducing overfitting.[2]

There are a number of algorithms that are used to compute the importance scores of various features of dataset. The most common ones which are utilized in the project include: Linear Regression, Ridge method, Lasso algorithm, MIC model, and Random forest.



(a)

(b)

Fig.1 – Feature importance graph for (a) 10-feature dataset, (b) 350-feature dataset

Figure 1 above illustrates a visual picture of the relative importances of features in a dataset. The vertical axis represents the range of importance scores and the horizontal axis represent the labels for the features. The features are sorted with the most important features coming first.

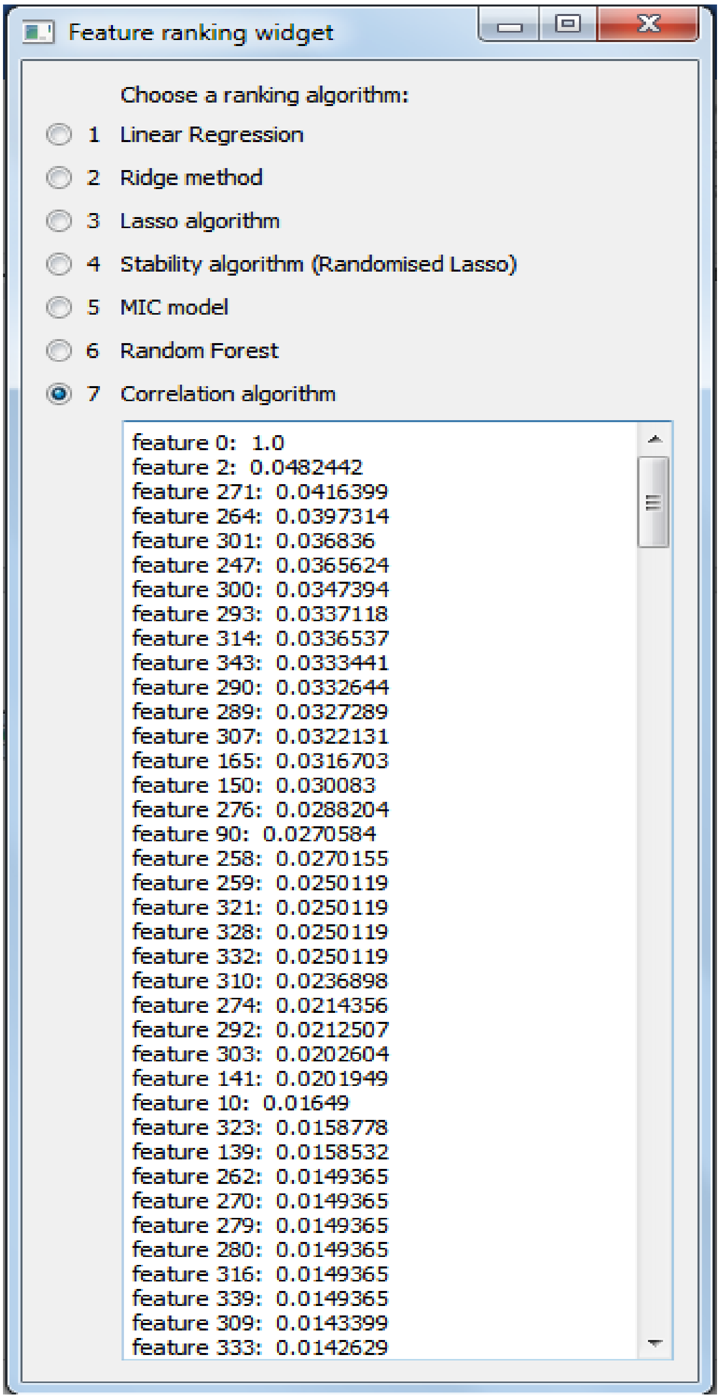
The error bars are reflective of the OOB or out of bag error estimate which calculates the mean prediction error on each data point of the dataset being fit into the random forest and is averaged over the entire forest. The importance score for the j-th feature is computed by averaging the difference in out-of-bag error before and after the permutation over all trees. The score is normalized by the standard deviation of these differences.

Related Work

Feature ranking and selection is a fairly popular concept in the field of machine learning, and as such, there are a number of other algorithms in the industry that also compute the importance scores of dataset features. A number of them worthy of note include the recursive feature elimination algorithm

Program Pipeline

The GUI takes a pair of datasets as inputs the first input is the raw dataset with as many features and any of the label or target dataset containing the clusters of output. The user is prompted to choose any of the listed algorithms with which to train the dataset. Choosing any of the algorithm buttons activates the corresponding algorithm, subsequently trains the dataset and produces the list of importance values sorted in descending order.



Feature ranking widget

Choose a ranking algorithm:

○ 1 Linear Regression

○ 2 Ridge method

○ 3 Lasso algorithm

○ 4 Stability algorithm (Randomised Lasso)

○ 5 MIC model

○ 6 Random Forest

● 7 Correlation algorithm

```
feature 0:  1.0
feature 2:  0.0482442
feature 271:  0.0416399
feature 264:  0.0397314
feature 301:  0.036836
feature 247:  0.0365624
feature 300:  0.0347394
feature 293:  0.0337118
feature 314:  0.0336537
feature 343:  0.0333441
feature 290:  0.0332644
feature 289:  0.0327289
feature 307:  0.0322131
feature 165:  0.0316703
feature 150:  0.030083
feature 276:  0.0288204
feature 90:  0.0270584
feature 258:  0.0270155
feature 259:  0.0250119
feature 321:  0.0250119
feature 328:  0.0250119
feature 332:  0.0250119
feature 310:  0.0236898
feature 274:  0.0214356
feature 292:  0.0212507
feature 303:  0.0202604
feature 141:  0.0201949
feature 10:  0.01649
feature 323:  0.0158778
feature 139:  0.0158532
feature 262:  0.0149365
feature 270:  0.0149365
feature 279:  0.0149365
feature 280:  0.0149365
feature 316:  0.0149365
feature 339:  0.0149365
feature 309:  0.0143399
feature 333:  0.0142629
```

Figure 2. Screenshot of the widget executing correlation ranking algorithm

As shown in the figure above, the feature ranking values are shown to be sorted from the important for the selected algorithm, down to the list important, hence the most important features are easily spotted at the top of the list.

Limitations and challenges

First and foremost, prior to taking up this project, I had little or beginner level experience with the python programming language. I had to learn on my feet. Most of what I coded was done after looking up resources online, implementing the things learned into ways suitable for my code. As a result, certain tasks that would have otherwise been easier to implement for seasoned python developers took a little extra time to come up with.

The feature ranking GUI is currently takes a while to train the data when some algorithms are utilized. For instance, the stability algorithm doesn't work well with our dataset as it produces 0's across for all the features. The GUI currently doesn't distinguish between a clustered dataset and a raw untampered dataset.

The code was developed in python 3.5 interpretation while the code of my team mates was implemented in python 2.7. So we are not able to easily combine both codes for this part of the project.

Future plan

The plan for the part 2 of the project is, among other things, to incorporate the clustering algorithms into the GUI. We intend to reconcile the differences in the python interpretations i.e to have everyone on the same page as regards to the python version to adopt for the project.

References

[1] Bermingham, Mairead L.; Pong-Wong, Ricardo; Spiliopoulou, Athina; Hayward, Caroline; Rudan, Igor; Campbell, Harry; Wright, Alan F.; Wilson, James F.; Agakov, Felix; Navarro, Pau; Haley, Chris S. (2015). *"Application of high-dimensional feature selection: evaluation for genomic prediction in man"*. *Sci. Rep.* .

[2] Gareth James; Daniela Witten; Trevor Hastie; Robert Tibshirani (2013). *An Introduction to Statistical Learning*. Springer. p. 204.