STAT639 (Data Mining): Course Project Report

Ritwik Bera (127005833)

Spring 2021

1 Introduction

The projects is composed of two problems: a supervised learning problem and an unsupervised learning problem. Each is discussed in detail below in individual sections that cover the problem scope, methods and results.

2 Supervised Learning

This is a binary classification problem that maps 500-dimensional vectors to either of two classes. 400 samples of training data along with their classes (x, y) are provided. This training data is used to estimate testing accuracy and select an appropriate classification model. The parameters of the classification model are also to be tuned based on validation performance using subsets from the training data. Predictions are also generated for 1000 input samples (xnew).

2.1 Candidate Models

The models considered for the classification problem are discussed below:

2.1.1 K-Nearest Neighbors

Package Used: *class, caret*

KNN is a type of lazy learning as it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbours of each point.

Pros:

- Simple to implement
- Robust to noisy training data, and effective if training data is large

Cons:

- Need to determine the value of K
- Computation cost is high as it needs to compute the distance of each sample to all the remaining training samples

2.1.2 Logistic Regression

Package Used: *glmnet, boot*

Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function.

Pros:

- Logistic regression is designed for this purpose (binary classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.

Cons:

- Works only when the predicted variable is binary.

- High prediction variance and prone to overfitting when low number of samples present.
- Logistic Regression can lack stability when the classes are well-separated.

2.1.3 Linear Discriminant Analysis

Package Used: *MASS*

LDA is a generative modelling technique and uses Bayes' Theorem to estimate probabilities. They make predictions based upon the probability that a new input dataset belongs to each class. The class which has the highest probability is considered the output class and then the LDA makes a prediction.

Pros:

- Logistic Regression can lack stability when the classes are well-separated. This is where LDA comes in.
- If there are few examples from which the parameters are to be estimated, logistic regression becomes unstable and LDA is preferred

Cons:

- It requires normal distribution assumption on features/predictors.

2.1.4 Support Vector Machine

Package Used: *e1071*

A SVM is a representation of the training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

Pros:

- Effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient.

Cons:

- The algorithm does not directly provide probability estimates since its a discriminative classifier based in spatial distances.

2.1.5 Random Forest

Package Used: *randomForest*

A RF is an estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

Pros:

- Reduction in over-fitting and is more accurate than decision trees in most cases.

Cons:

- Slow real time prediction
- Complex algorithm due to lots of subsampling and ensembling involved.

2.1.6 Gradient Boosting Machine

Package Used: *gbm*

Ensemble learning involves building a strong model by using a collection (or "ensemble") of "weaker" models. Gradient boosting falls under the category of boosting methods, which iteratively learn from each of the weak learners to build a strong model.

Pros:

- High predictive accuracy
- No data pre-processing required, each feature individually handled.

Cons:

- GBM training generally takes longer because of the fact that trees are built sequentially
- Prone to overfitting. More generalized trees require tuning learning rate (shrinkage) and depth of tree. Generally shrinkage and tree depth are kept on the lower side to allow for slow learning and better generalization.

2.2 Parameter Tuning and Model Selection

Some of the candidate classification models had tunable parameters. These included:

- *KNN*: Number of nearest neighbors to consider
- *SVM*: Cost of constraint violation
- *Random Forest*: Number of trees to grow
- *Gradient Boosted Tree*: Number of trees to fit

For N total data points,

$$Classification\ Accuracy = \sum_{n=1}^{N} \mathbb{1}(\text{Predicted!} = \text{Truth})/N$$

$$Test\ Error\ Estimate\ (TEE) = 1 - Test\ Accuracy\ Estimate(TAE)$$

Testing accuracy was evaluated through different estimation techniques for different models, this included:

- *K-Fold Cross Validation* (Fig.1): for KNN, Logistic Regression, SVM, Gradient Boosted Tree
- *Leave-One-Out Cross Validation* (Fig.2): for LDA
- *Out-of-Bag Error* (Fig.3): for Random Forest

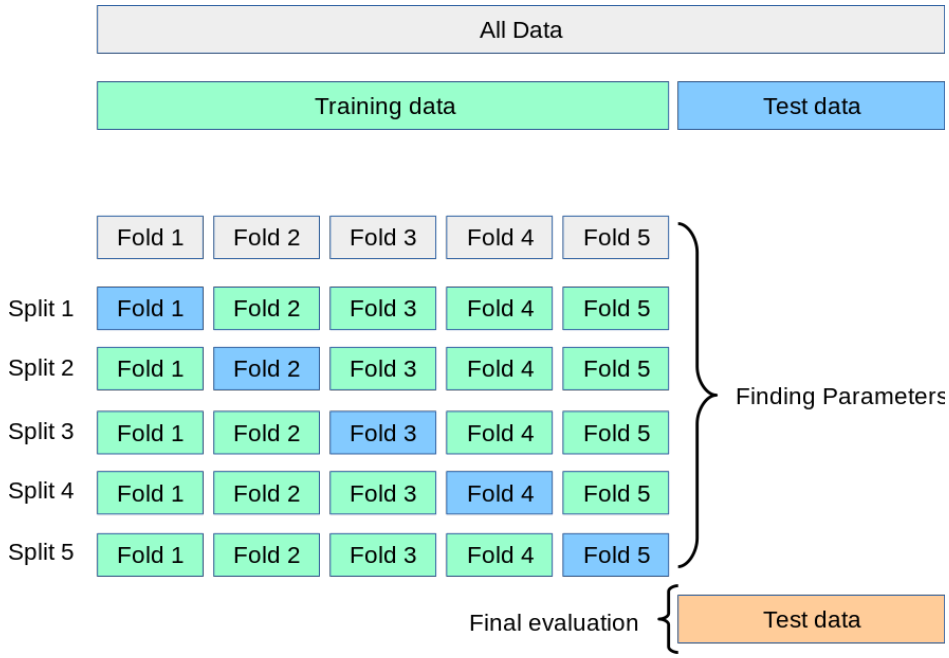

Figure 1: K-Fold Cross Validation

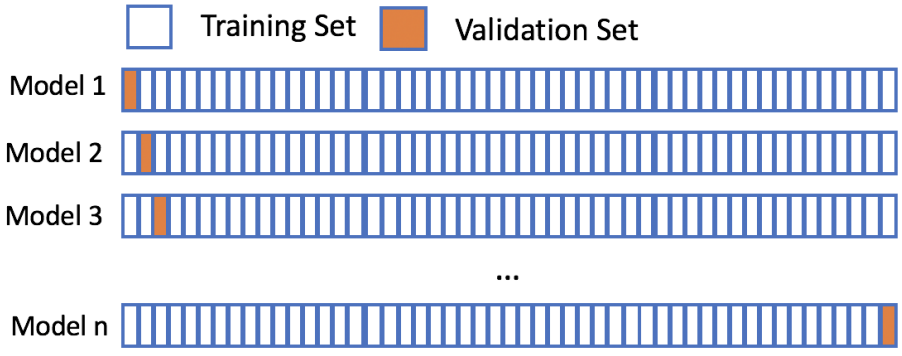

Figure 2: Leave-One-Out Cross Validation

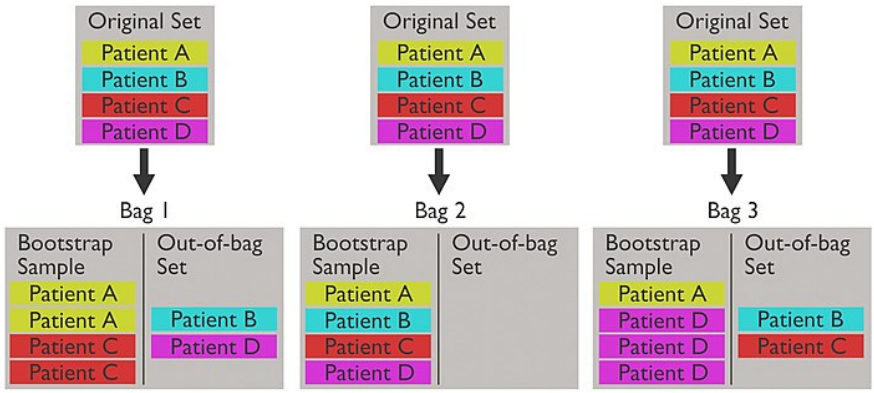

Figure 3: Out-of-Bound Error Prediction Mechanism

2.3 Results

2.3.1 Comparison of Models

Model	Parameter Tuned	Candidate Values	Optimal Value	TEE
Logistic Regression	N/A	N/A	M/A	0.481
LDA	N/A	N/A	N/A	0.47
KNN	K	1:10	5	0.292
SVM	cost	1,4,6,8,10	8	0.435
Random Forest	ntrees	300,500,1000	500	0.441
Gradient Boosting	iter	1:3000	564	0.29

2.3.2 Best Model

Description: **Gradient Boosted Trees** were found to be the best model.

Best parameter value: Predictions were generated using this model utilizing 564 trees (564 trees iteratively fit to the training data).

Estimated Testing Error: The estimated value of testing error was **0.29** (29% misclassification rate).

3 Unsupervised Learning

A dataset of 1000 784-dimensional data points is provided. The objective is to find the optimal number of clusters that this dataset can be clustered into. The methods investigated for this purpose are described below. The R package *fviz_nbclust* was extensively used for this analysis.

3.1 Method Selection

3.1.1 Elbow Method

The basic idea behind partitioning methods, such as k-means clustering, is to define clusters such that the total intra-cluster variation [or total within-cluster sum of square (WSS)] is minimized. The total WSS measures the compactness of the clustering and we want it to be as small as possible. The Elbow method looks at the total WSS as a function of the number of clusters: One should choose a number of clusters so that adding another cluster doesn't improve much better the total WSS. However, as evident from Fig. 4, no knee/elbow can be found.

3.1.2 Silhouette Method

This method measures the quality of a clustering. Average silhouette method computes the average silhouette of observations for different values of k. The optimal number of clusters k is the one that maximize the average silhouette over a range of possible values for k [1]. The silhouette score for each data pointed is computed as:

$$s(o) = \frac{b(o) - a(0)}{max(b(0), a(0))}$$

where,

$a(o)$ is the average distance between o and all the other data points in the cluster to

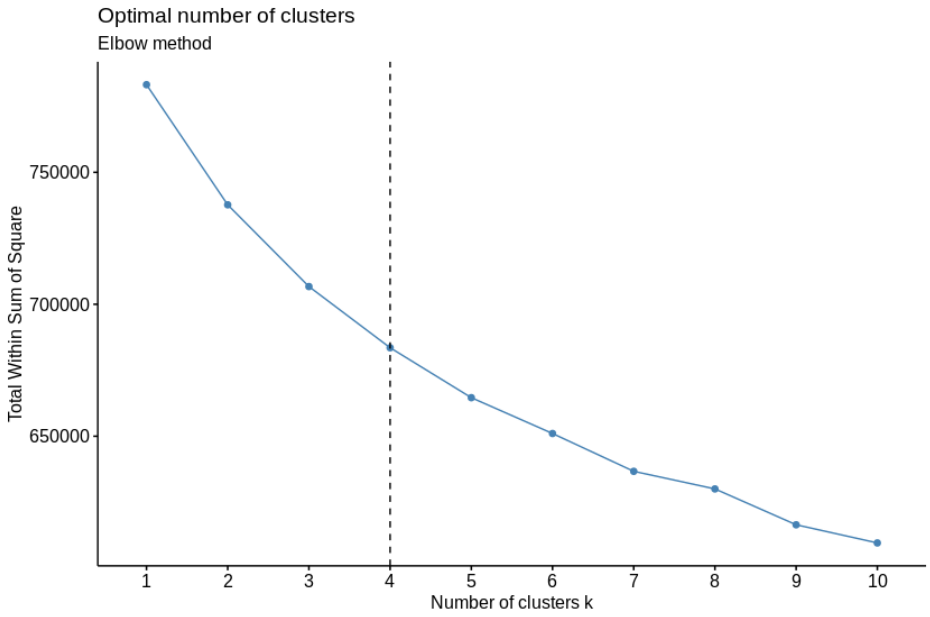

Figure 4: No clear elbow evident in the WSS/Distortion vs k plot

which o belongs.

$b(o)$ is the minimum average distance from o to all clusters to which o does not belong.

As seen in Fig. 6, the average silhouette score is maximised from k=2.

3.1.3 Gap Statistic Method

The gap statistic [2] compares the total within intra-cluster variation (distortion) for different values of k with their expected values under null reference distribution of the data. The estimate of the optimal clusters will be value that maximize the gap statistic (i.e, that yields the largest gap statistic). This means that the clustering structure is far away from the random uniform distribution of points.

The algorithm is described as:

- Cluster the observed data, varying the number of clusters from $k = 1, \ldots, k_{max}$, and compute the corresponding total within intra-cluster variation W_k.
- Generate B reference data sets with a random uniform distribution. Cluster each of these reference data sets with varying number of clusters $k = 1, \ldots, k_{max}$, and compute

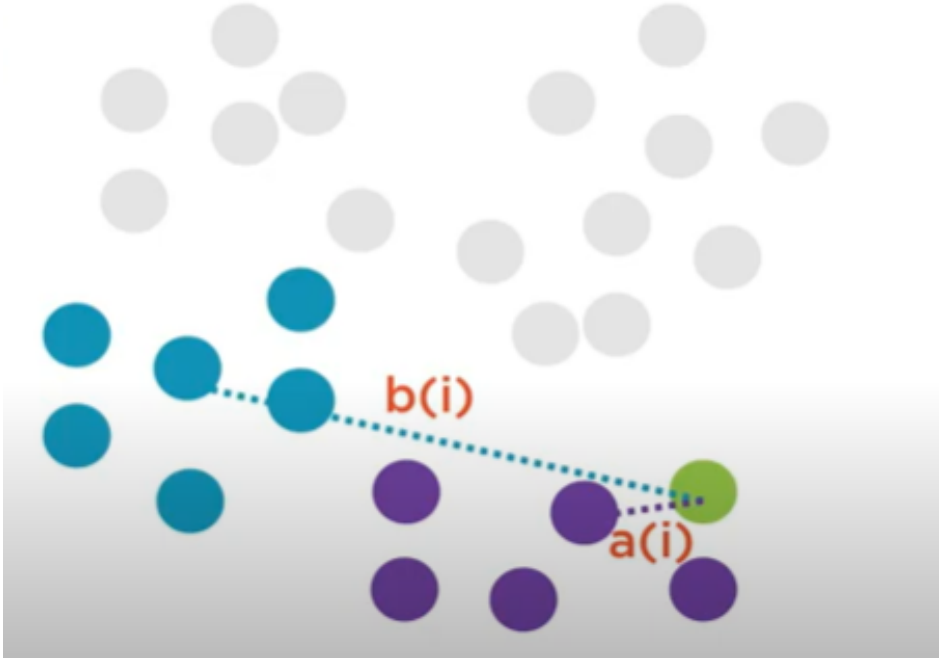

Figure 5: Silhouette Determination

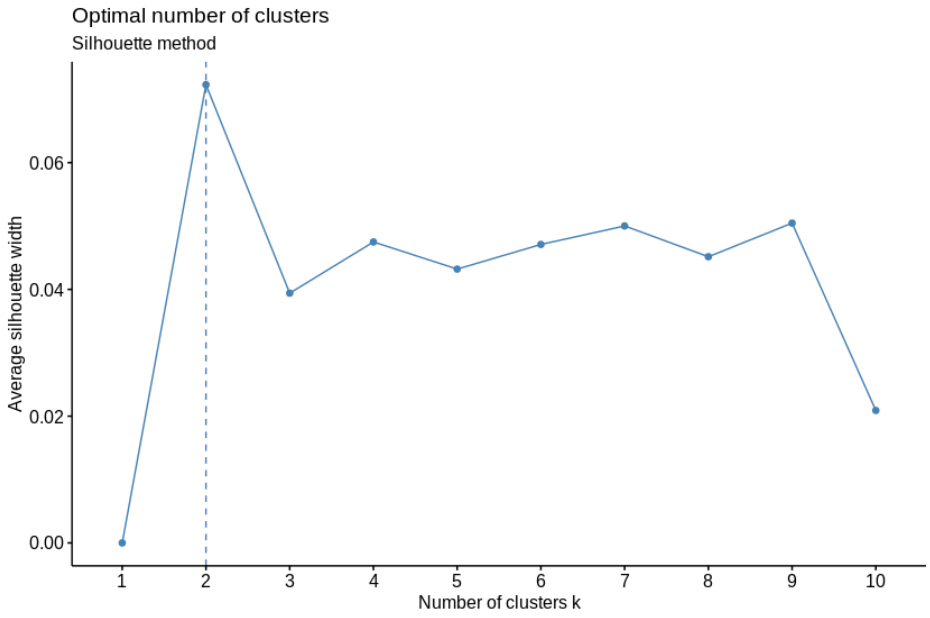

Figure 6: Determination of optimal number of clusters through Silhouette Method

the corresponding total within intra-cluster variation W_{kb}.

- Compute the estimated gap statistic as the deviation of the observed W_k value from its expected value W_{kb} under the null hypothesis:

$$Gap(k) = \frac{1}{B}\sum_{b=1}^{B} log(W_{kb}^*) - log(W_k)$$

- Choose the number of clusters as the smallest value of k such that the gap statistic is within one standard deviation of the gap at $k+1 : Gap(k) \geq Gap(k+1) - s_k + 1$.

However, no local such local maximum of the gap statistic method could be found and thus this method was not further used. Pham et. al. introduced a closely related method for the identification of the number of clusters [3], which is much more computationally efficient. The R package *kselection* provides an implementation. However no clear clustering was identified by this method.

3.2 Results

Only the silhouette method gave conclusive results on the number of clusters potentially present in the given dataset. Thus the number of clusters is recommended to be 2.

References

[1] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[2] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

[3] Duc Truong Pham, Stefan S Dimov, and Chi D Nguyen. Selection of k in k-means clustering. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 219(1):103–119, 2005.