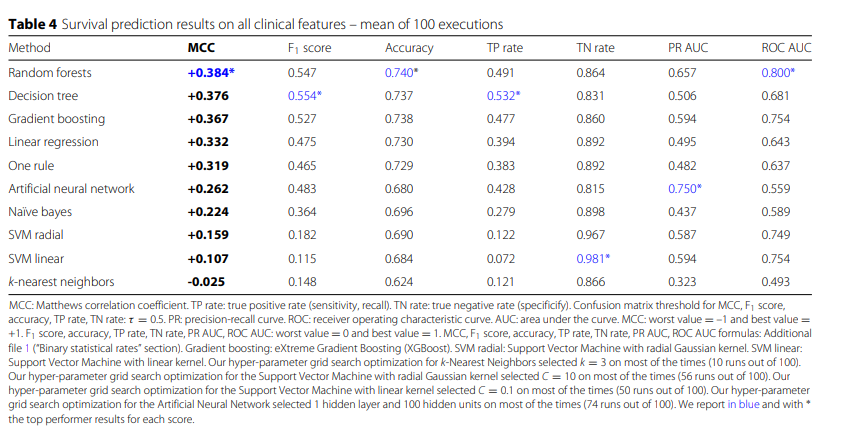
**Improving Research Paper**



| Method | MCC | F1 score | Accuracy | TP rate | TN rate | PR AUC | ROC AUC |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Random Forest | 0.385 | 0.538 | 0.746 | 0.642 | 0.872 | 0.474 | 0.676 |
| Decision Tree | 0.302 | 0.444 | 0.696 | 0.471 | 0.745 | 0.393 | 0.658 |
| Gradient Boosting | 0.387 | 0.554 | 0.746 | 0.62 | 0.854 | 0.854 | 0.683 |
| Linear Regression | 0.353 | 0.491 | 0.735 | 0.655 | 0.892 | 0.460 | 0.652 |
| One rule | 0.336 | 0.503 | 0.728 | 0.608 | 0.865 | 0.454 | 0.652 |
| Artificial neural Network | 0.283 | 0.444 | 0.75 | 0.461 | 0.847 | 0.331 | 0.638 |
| Naive bayes | 0.084 | 0.444 | 0.666 | 0.363 | 0.837 | 0.302 | 0.536 |
| SVM radial | 0.117 | 0.125 | 0.76 | 0.5 | 0.97 | 0.25 | 0.52 |
| SVM linear | 0.235 | 0.133 | 0.783 | 1 | 1 | 0.288 | 0.535 |
| K-NN | -0.246 | 0 | 0.60 | 0 | 0.78 | 0.23 | 0.39 |

Random Forest

Procedures:

-No feature selection applied to dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of random forest with default parameters with 80/20 train/test split each time. I then average the results of 100 runs to create my metrics. I ran the method multiple times and noticed a sizable variation in scores from 0.36 MMC to 0.38. This variation can be attributed to a number of factors. The dataset has an unbalanced ratio target with around ⅓ being positive with ⅔ being negative. This increases the chance for a highly skewed train/test data subset which would alter the results. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article.

Decision Tree

Procedures:

-No feature selection applied to dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of Decision Tree with default parameters with 80/20 train/test split each time. I then average the results of 100 runs to create my metrics. I ran the method multiple times and noticed a sizable variation in scores. This variation can be attributed to a number of factors. The dataset has an unbalanced ratio target with around ⅓ being positive with ⅔ being negative. This increases the chance for a highly skewed train/test data subset which would alter the results.There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed worse than the article.

Gradient Boosting

Procedures:

-No feature selection applied to dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of Gradient Boosting with default parameters with 80/20 train/test split each time. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article.

Linear Regression

Procedures:

-No feature selection applied to dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- post processing of model included bucketing threshold of 0.5> “positive” / 0,5 < “negative”

By following the article I applied 100 model runs of Linear Regression with default parameters with 80/20 train/test split each time. I then average the results of 100 runs to create my metrics. The predictions are then binned into 2 variables “0” and “1” based on the 0.5> threshold. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article.

One rule

Procedures:

- feature selection/preprocessing applied to dataset include binning dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of One rule with default parameters with 80/20 train/test split each time. I then average the results of 100 runs to create my metrics.Based on running the model, One Rule model would perform poorly without scaling the dataset and it is a possibility that the author achieved the results with preprocessing. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article when applying preprocessing methods to the dataset.

Naive bayes

Procedures:

-No feature selection applied to dataset

- data split 80%/20% train/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of Naive bayes with default parameters with 80/20 train/test split each time. I then average the results of 100 runs to create my metrics. I ran the method multiple times and noticed a sizable variation in scores. There is a strong possibility that hyperparameters were optimised to achieve better results . Based on MCC which is the main metric the article used to judge models, my model performed worse than the article.

Artificial neural Network

Procedures:

- feature selection/preprocessing applied to dataset include scaling entire dataset

- data split 60%/20%/20% train/validate/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of Artificial neural Network with default parameters with 60%/20%/20% train/validate/test split each time. I then average the results of 100 runs to create my metrics. Based on running the model, Artificial neural Network the model would perform poorly without scaling the dataset and it is a possibility that the author achieved the results with preprocessing. MCC metric was used to score the hyperparameters instead of accuracy due to the main metric used by the author. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article when applying preprocessing methods to the dataset.

SVM radial

Procedures:

- feature selection/preprocessing applied to dataset include scaling entire dataset

- data split 60%/20%/20% train/validate/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of SVM radial with default parameters with 60%/20%/20% train/validate/test split each time.MCC metric was used to score the hyperparameters instead of accuracy due to the main metric used by the author. I then average the results of 100 runs to create my metrics. Based on running the model, Artificial neural Network the model would perform poorly without scaling the dataset and it is a possibility that the author achieved the results with preprocessing. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article.

SVM linear

Procedures:

- feature selection/preprocessing applied to dataset include scaling entire dataset

- data split 60%/20%/20% train/validate/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of SVM linear with default parameters with 60%/20%/20% train/validate/test split each time.MCC metric was used to score the hyperparameters instead of accuracy due to the main metric used by the author. I then average the results of 100 runs to create my metrics. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed better than the article.

K-NN

Procedures:

- feature selection/preprocessing applied to dataset include scaling entire dataset

- data split 60%/20%/20% train/validate/test

-No hyperparameter optimization to model

-default parameters for the model

- 100 runs of the model

- No post processing of model

By following the article I applied 100 model runs of K-NNwith default parameters with 60%/20%/20% train/validate/test split each time. Based on running the model, K-NN would perform poorly without scaling the dataset and it is a possibility that the author achieved the results with preprocessing. MCC metric was used to score the hyperparameters instead of accuracy due to the main metric used by the author. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article. Based on MCC which is the main metric the article used to judge models, my model performed worse than the article.

Afterthoughts

The dataset has an unbalanced ratio target with around ⅓ being positive with ⅔ being negative. This increases the chance for a highly skewed train/test data subset which would create large variances in the results. There is a strong possibility that preprocessing such as standardization or binning was applied on the dataset but not written down on the article.Based on running the 9 models, some would perform poorly without scaling the dataset and it is a possibility that the author achieved the results with preprocessing the dataset.There is also a possibility the author included some form of hyperparameter optimization not included in the models. This can include prior domain knowledge. Some of my models performed better and others worse than the author.

Q.2

i) The dataset has an unbalanced ratio target with around ⅓ being positive with ⅔ being negative. Optimal predictions may not be achieved through basic splitting of the small data set. The author incorporated a pseudo leave 1 out validation into his model to make up for small sample size. I propose to reduce bias and variance in our predictions using K-fold cross validation. I want to use hyperparameter optimization with each data subset to increase our prediction results. To this end I will be appalling a technique called nested cross validation to the authors best performing model random Forest.

ii)

Nested cross-validation is used to overcome bias in prediction performance. It involves finding optimal hyperparameters as well as fitting the model. The author Random forest model had no hyperparameter optimization method as well as a basic random sampling split 80%/20%, My model uses a cross validation nested inside another cross validation to find optimal hyperparameters for the inner loop while model fitting occurs in the outer loop.One of my models uses a dataset that has been scaled.

iii) The dataset is scaled using standard scale. Random forest is the main model. The custom score uses MMC as the scoring metric to decide hyperparameters. The inner k fold (k = 5) using gridsearch to find optimal hyperparameters which includes number of estimators [300,500,600,700] and max features as [2,11]. The outer cross validation loop takes the best hyperparameters and fits the rest of the dataset to predict the results and the metrics are recorded.

iv)

-scale transform dataset

- create custom score (MCC)

-define number of k for inner cross validation

- fit gridsearch of hyperparameters into inner fold

-define outer cross validation fold

- fit the training data into outer cross validation fold

- predict the testing data based hyperparameters in inner fold and model in outer fold

- record metrics

v) The Evaluation metrics used for this model is the same as the authors

| MCC | F1 score | Accuracy | TP rate | TN rate | PR AUC | ROC AUC |
| --- | --- | --- | --- | --- | --- | --- |

vi)

| Random Forest | MCC | F1 score | Accuracy | TP rate | TN rate | PR AUC | ROC AUC |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No scale | 0.409 | 0.579 | 0.752 | 0.637 | 0.794 | 0.489 | 0.694 |
| Scale | 0.646 | 0.75 | 0.849 | 0.8 | 0.869 | 0.660 | 0.812 |

These are the results of both my models with/without scaled dataset. The scaled dataset performed better in all metrics than the model without the scaled dataset.

vii)

| Random Forest | MCC | F1 score | Accuracy | TP rate | TN rate | PR AUC | ROC AUC |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No scale | 0.409 | 0.579 | 0.752 | 0.637 | 0.794 | 0.489 | 0.694 |
| Scale | 0.646 | 0.75 | 0.849 | 0.8 | 0.869 | 0.660 | 0.812 |
| author | +0.384\* | 0.547 | 0.740\* | 0.491 | 0.864 | 0.657 | 0.800\* |

The running theme through this assignment is that preprocessing the dataset is extremely important to achieve optimal prediction metrics. Some models without preprocessing could not tell the difference between positive and negative targets. Using the main metric MCC, Nested cross validation and scaled data doubled the score with the authors random Forest model. There was a slight increase in MMC score using just nested cross validation. Scaled data with nested outperformed every metric compared to the author.

## References

[1]J. Brownlee, "Nested Cross-Validation for Machine Learning with Python", *Machine Learning Mastery*, 2021. [Online]. Available: https://machinelearningmastery.com/nested-cross-validation-for-machine-learning-with-python/?fbclid=IwAR2Q4bNA42WDb1dPTZcHM5v7El8VOQve0GZfm\_-RNWQLdoD4OPNOhzpNt9E. [Accessed: 08- Oct- 2021].

[2]S. Raschka, "Home - mlxtend", *Rasbt.github.io*, 2021. [Online]. Available: http://rasbt.github.io/mlxtend/. [Accessed: 09- Oct- 2021].

[3]"scikit-learn: machine learning in Python — scikit-learn 1.0 documentation", *Scikit-learn.org*, 2021. [Online]. Available: https://scikit-learn.org/stable/. [Accessed: 07- Oct- 2021].