# A Comparison of Naïve Bayes (NB) and Random Forest (RF) on Predicting Employee Departures

# IDM431: Machine Learning – Edward St John Appendix

## Glossary

**Bias**

The tendency of a model to have prejudice from learning the wrong signals.

**Categorical variable**

A variable often without logical order containing a finite number of distinct groups.

**Class imbalance**

When a high proportion of the data belongs to a single class label instead of being split more equally. This tends leads to classifiers being more bias towards that particular class.

**Confusion matrix**

A 4-way-split table grouping predictions in order to assess performance.

**Correlation matrix**

A table displaying correlation coefficients between variables using all the possible pairs of variables in a table.

**Classifiers**

Models that predict a certain class / label based on various data points.

**Continuous variable**

Numerical variables that contain an infinite number of values (often between two values, i.e. a maximum and a minimum).

**Ensemble**

The aggregation of multiple models’ outputs, producing one optimal model.

**Hyperparameter**

A hyperparameter is an external parameter (i.e. not derived from the data) that define the underlying system of the model that get tuned in order to optimise model performance.

**Naïve Bayes (NB)**

A supervised classification model based on Bayes Theorem.

**Overfitting**

When a model (often the more complex models) corresponds too closely to the data it is trained on, leading to an unreliable prediction with future / unseen data.

**Random Forest (RF)**

An algorithm using a collection of decision trees grouping together, either for classification or regression problems.

**Variance**

The degree of the spread of how far a set of numbers vary from the mean.

References: datascienceglossary.org

mastersindatascience.org

## Intermediate results including any negative results

### Random Forest Trees:

During model tuning for the Random Forest model, it was noticed that the maximum allowed number of trees (‘NumTrees’) of 150 was returning as the optimal value. To investigate whether a higher maximum would yield more beneficial results, NumTrees was edited to allow up to 200 trees as a hyperparameter. This ended up having a worse F1 score at 0.8067 (vs 0.8139 in the final model used with 150 trees), which led to changing the parameter back to a maximum of 150 to keep for the final model.

### Naïve Bayes K-fold

K-fold cross-validation was used for the Naïve Bayes model, with the number of folds originally set at 10. Out of interest, during the model creation phase, the number of folds was changed from 10 to 5 to see how it affected model performance and how much of a difference there would be. As it turns out, it provided exactly the same accuracy score – probably due to having a relatively small dataset and not being large enough to see a tangible difference in using 10 folds. Due to this, the number of folds was kept at 5 for the final model as it took less training time.

## Implementation details including a brief description of main implementation choices

Initial data exploration and pre-processing was performed in Python. This included converting the categorical variables to numerical so that the NB could handle them (with new columns made splitting out each city as an example), as well as the basic statistics and correlation matrix. The other figures used in the ‘Initial analysis of data including basic statistics’ were created in Tableau (but included in the Python file).

In MATLAB, the ‘rng’ function was set to ‘default’ in order to ensure reproducibility in the code. For the models, the function used for training Naïve Bayes model was ‘fitcnb’, and the function used for training the Random Forest model was ‘TreeBagger’.