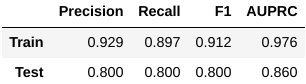
**Predicting Bedaquiline Resistance Based on Known Mutations for Rv0678**

Results



A gradient-boosted tree classifier was trained and optimised with mutations associated with resistance and susceptibility phenotypes. This was done to determine if mutations in *Rv0678* are useful in predicting the resistance phenotype. Features were engineered to represent various amino acid property changes due to mutation. The optimised model performed reasonably well with an area under the precision-recall curve (AUPRC) of 0.860. A permutation test was performed to determine how likely the observed AUPRC was due to chance. Permutation tests are useful because they break the associations between the resistance phenotype and mutations in *Rv0678* to produce a suitable null model. The *p*-value for AUPRC of the optimised model was 0.0022 suggesting that the optimised model could successfully differentiate between resistance and susceptibility phenotypes given information of changes to physiochemical properties as a result of mutation.

Models were interpreted using *TreeExplainer* (lundberg), which decomposes the predicted probability score of each prediction into individual feature contributions. That is, each feature is assigned a SHAP value which represents the change in predicted probability score in each prediction. The importance of features in “black box” models can thus be inferred from their mean absolute SHAP values. The results suggest that mutations of polar and positively charged residues are associated with Bedaquiline resistance. Additionally, transitions from negatively to positively charged residues are associated with susceptibility. Lastly, variants with small MW and that hydrophobic are associated with susceptibility.

Figure 1. Summary plot of SHAP values. Each point represents the SHAP value of a single prediction for a particular feature. Points are stacked vertically using density estimation. WT, mutant and MW denotes the wild-type amino acid, amino acid variant and molecular weight (Da) respectively. Property refers to whether an amino acid was non-polar, polar, positively charged or negatively charged.

Methods

A gradient-boosted tree classifier was developed using the *XGBoost API* (v0.90) (tianqi) and interpreted using *TreeExplainer* as part of the *shap* API (v0.34.0) (lundberg) to determine whether the phenotypes of genomes with unknown mutations could be predicted based on the change in amino acid properties for known mutations. The task of the classifier is to predict whether a mutation results in a susceptible or resistant phenotype given the change in AA properties. Eleven features were engineered based on the wild-type and mutant residues of each mutation. Two features represent the amino acid residue of the wild-type and of the mutant. Two features encode whether they are non-polar, polar, positively charged or negatively charged. Two features represent whether the wild-type and the mutant residues are hydrophobic (+1) or hydrophilic (-1). Two features encode the molecular weight of wild-type and mutant AA. One feature represents the change in charge from from the wild-type to mutant, where non-polar and polar residues are assumed to contribute a charge of zero. One feature represents the change in hydrophobicity where a hydrophobic→hydrophilic residue change is coded as +1 and the reverse as -1. One feature represents the change in molecular weight from the wild-type to mutant residue. Only mutations that are associated with a resistant or susceptible phenotype were used, resulting in 59 resistance and 32 susceptibility mutations. Model parameters were optimised to maximise the F1 score and the final model was evaluated using a 50% test dataset. AUPRC was used for model evaluation due to class imbalance in the dataset (CITE AUPRC paper). Statistical significance of the final model was determined via permutation testing. 10000 models were constructed using the previously optimised parameters and with permuted class labels. These models were then evaluated with the same test set. Lastly, a *p*-value was computed for the AUPRC metric as the proportion of models that had an AUPRC greater or equal to that of the final model.