**CS6301.001 (S’19): Midterm project report |Prof. Kunapuli**

Siva Saket Sripada - SXS180054@utdallas.edu - 50% | Anusha Devi - AXT180026@utdallas.edu - 50%

**1 Pre-processing**

Pre-processing step used is standard scalar for z-score-gaussian normalization, to avoid scale-based discrimination among features during learning weights. All features are represented by their mean and variance rather than absolute values.

**2 Machine Learning Methods**

The training data set is split into training and validation dataset (val\_frac=0.3). Pipelines were constructed for different algorithms viz. decision tree, logistic regression, K-nearest neighbors, random forest and SVM. The task is a 12-dimensional binary classification. Random forest, SVM, Logistic Regression, Decision trees and k-Nearest neighbors were chosen because:

1. Perform generally best on almost all publicly available datasets [[Olson et al, 2017]
2. Simpler to implement/run on low computational resources
3. Easier to use with sklearn's grid-search, pipeline and ROC implementations

Below is a table describing the different classification algorithms, their best parameters (courtesy GridsearchCV), Accuracy, precision and recall:

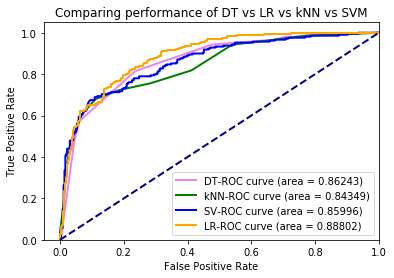
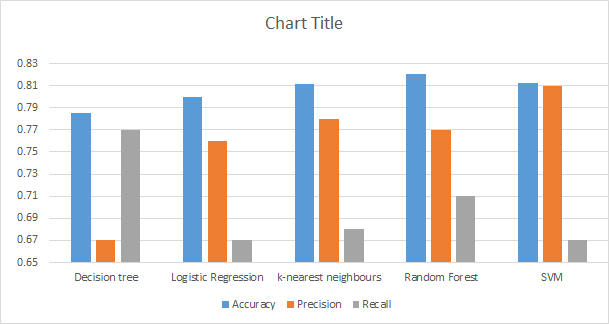
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithms | Parameters | Accuracy | Precision | Recall |
| Decision tree | max\_depth': 4 | 0.785211268 | 0.67 | 0.77 |
| Logistic Regression | 'C': 10.0 | 0.799295775 | 0.76 | 0.67 |
| k-nearest neighbours | 'n\_neighbors': 12 | 0.811619718 | 0.78 | 0.68 |
| Random Forest | 'max\_depth': 7, n\_estimators=500 | 0.820422535 | 0.77 | 0.71 |
| SVM | C': 1, 'gamma': 0.1, 'kernel': 'rbf' | 0.812664908 | 0.81 | 0.67 |

**3 Results**

Cross validation using Grid search was performed on 70% of the training data, for 10 cycles. The rest of the 30% of the training data was reserved as explicit pseudo-test/validation set to evaluate the models.

Based on the best accuracy and precision scores, SVM is chosen as the best model. It gets >81% of the validation labels right, and classifies least number of negative labels as positive (avoiding bad recommendations or i.e. it is preferred to not classify a not-so-great-song among top 100)

(PTO)



ROC for Logistic regression (jagged orange), SVM(jagged blue) and Decision trees (smooth pink) almost all look equally good, however, a close analysis of the model-performance based on precision and accuracy shows that SVM is clearly supreme (and is as good or better than Random Forest).

**4 Future Work**

* Finer and more exhaustive hyper parameter optimisation can be performed granted access to supreme computing resources.
* A more comprehensive search of several variants of SVMs and other algorithms can be tried, such as platforms like [TPOT.](https://github.com/EpistasisLab/tpot/)
* Principle component Analysis and similar dimensionality reduction techniques can be used to make the classification easier to visualise, and the model more interpretable.
* AdaBoost or similar enhancing algorithms can be used to build upon a baseline SVM in order to better capture the factors that truly make a song a top-100 song.
* An online/interactive/semi-supervised-Reinforcement learning approach could make the predictor more human-like in its ability to "enjoy" the song it tries to classify.