Data Science Project

Healthcare - Persistency of a drug

Group Name: BetterHealth Analytics	
Member Details	Name: Enias Vontas Country: Greece Email: vondas100@gmail.com Specialization: Data Science

Feature Selection

Having an output variable, 'Persistent' or 'Non Persistent' and almost all our predictor variables as categorical, leaves us with two choices when it comes to selecting our features.

- Chi Square test
- Mutual Information

We saw a Chi Square implementation in the section above, so we will not explain it further here. As far as mututal information is concerned, it comes from the Information Theory world, where Information will quantify how surprising an event is in bits. Low probability events have more information, while higher probability events have less information. We would like to quantify how much information there is in a random variable's probability distribution (Entropy). A skewed distribution has low entropy but a distribution where events have a more balanced probability of happening has a larger entropy.

The entropy of a dataset could be in terms of the probability distribution of observations in the dataset belonging to one class or the other. A dataset with a 50/50 split of samples for the two classes would have a maximum entropy (maximum surprise) of 1 bit, whereas an imbalanced dataset with a split of 10/90 would have a smaller entropy as there would be less surprise for a randomly drawn example from the dataset. In this way, enropy can be used as a calculation of how balanced the distribution of classes happens to be. In our case, for example, where we have 1713 'Non Persistent'(class0) and 1026 'Persistent' (class1) patients out of a total of 2739, we calculate the entropy as:

$$entropy = -\big(class0 * log2(class0) + class1 * log2(class1)\big) = 0.9541$$

which indicates that the values are in between the classes.

We will apply both methods and check all possible numbers of features: from just 1 all the way to 104, the total number of our predictors. Our criteria for choosing the number of predictors will be according to the Accuracy of a Logistic Classification model. As the number of features increases, so does the Accuracy of our classification model. But there is a point where the number of predictors become too many, increasing the complexity of our model and thus decreasing its overall Accuracy. We would like to find an optimal number of features in order to maximize our Accuracy but not have too many predictors. For better understanding we present some results in the table below, where we have various numbers of features and Accuracy of the model for both Chi Square and Mutual Information methods:

Number of Features	Chi Square Accuracy	Mutual Information Accuracy
15	0.8573	0.85348
20	0.86172	0.85847
30	0.86291	0.86380
31	0.86579	0.86281
35	0.86917	0.86850
39	0.86982	0.86623
40	0.87037	0.86798
41	0.87015	0.86249
45	0.86978	0.86520
80	0.86969	0.86867
100	0.87188	0.87133

We can see a similar picture from both methods. Also as the number of features increases, so does the Accuracy, but after a point it decreases again, only to increase toward the end. We should note that out of the 101 (102 – 1, because we cannot compute for 0 features) in 66 cases the Chi Square method Accuracy was greater than Mutual Information, but the differences were not very large and if we round on the 2nd or 3rd decimal point we reach 87% with about 30 predictors or so. Mutual Information was slower than Chi Square, and since Chi Square seems to reach about 87% Accuracy with fewer predictors, we will keep the result from Chi Square method with 31 predictors.

Model Building

Now that we have the number of features we will start building various models and see which one seems to classify our 'Test' set patients best. From now on we will only consider the predictors chosen in the previous step. The methods for our Classification problem will be Logistic Regression, RandomForest, k-nearest neighbors, Gradient Boosting and ExtraTrees.

The Logistic Regression has the capability of modeling our probabilities with a function that produces values between 0 and 1, which in a binary classification problem is very helpful.

The Decision Tree methods are non-parametric supervised learning methods. It uses a tree-like model of decisions and their possible consequences. It is helpful in our situation since it requires little data preparation (such as Encoding) and is able to handle both numerical and categorical variables.

The K-nearest neighbors method classifies a data point according to the classes of its neighbors. The number of neighbors can be tuned to find the optimal solution. It works with numeric values, hence Encoding is need in this method.

The Gradient Boosting Classifier works with an ensemble of weak learners, usually Decision Trees. It is seen as a numerical optimization problem where the goal is to minimize the loss of the model by adding weak learners using a gradient like procedure.

We can check our model's performance by comparing our actual values and the ones we got as predicted values and then count the correct and incorrect predictions. This will be done with the 'test set', the number of observations that were left out of the computation process. We will compare those values with the ones we got from our model and see how accurate we are, or not. We do this with the help of a confusion matrix:

$$\begin{pmatrix} TP & FP \\ FN & TN \end{pmatrix}$$

Where:

- TP(True Positives): correctly predicted ones ('Persistent'),
- TN(True Negatives): correctly predicted zeroes ('Non Persistent'),
- FN(False Negatives): incorrectly predicted zeroes,
- FP(False Positives): correctly predicted ones

From here we will calculate the Accuracy and Precision based on:

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$Precision = \frac{TP}{TP + FP}$$

Accuracy can be seen as the fraction of predictions our model predicted correctly. Precision can be seen as the ability of the Classifier not to label as positive a sample that is negative.

Another metric we can use to determine the diagnostic ability of a binary classifier is the Receiver Operating Characteristic (ROC) curve. It is constructed by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR):

$$\mathrm{TPR}=\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$$

And

$$\mathrm{FPR}=\mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$$

The ROC curve shows the trade-off between sensitivity (or TPR) and specificity (1 – FPR). Classifiers that give curves closer to the top-left corner indicate a better performance. A random classifier would give points lying along the diagonal (FPR = TPR)(our baseline).

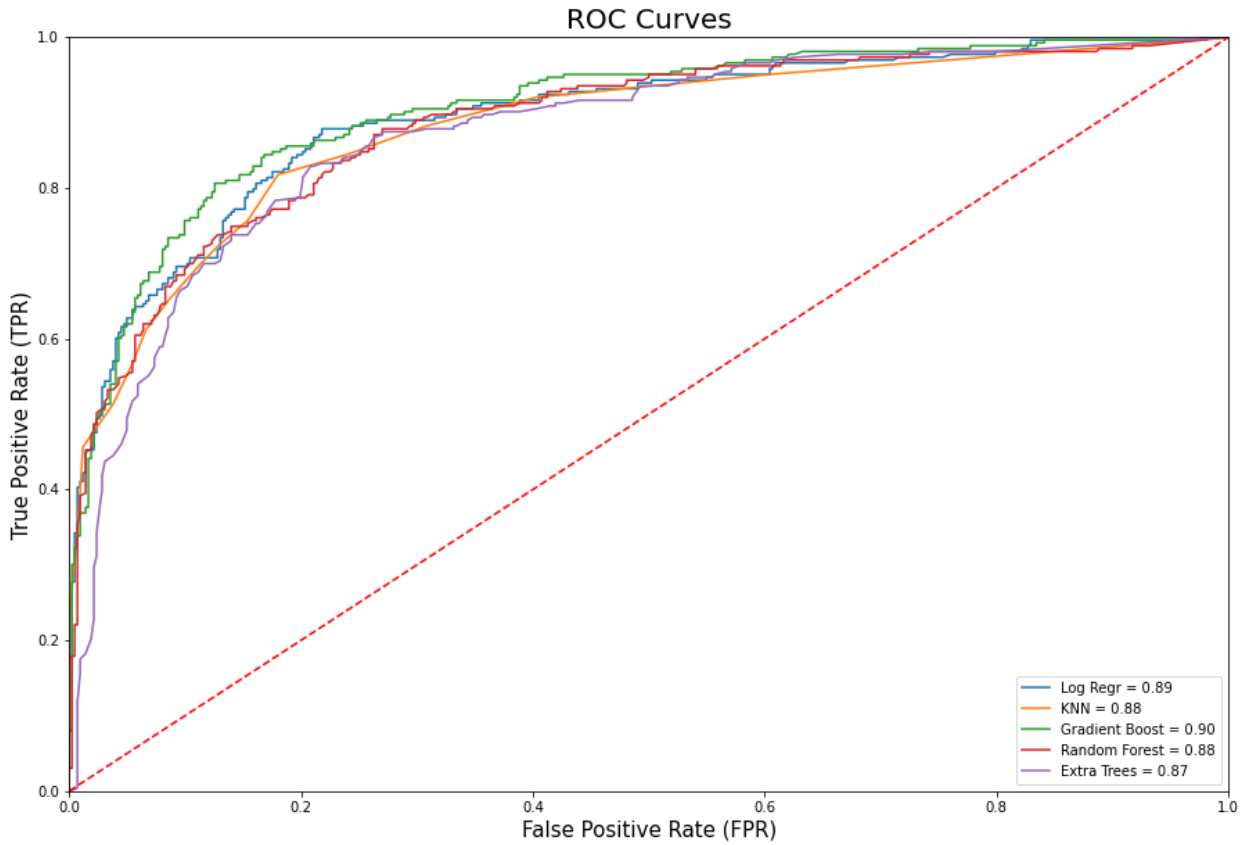
Area Under the Curve (AUC) is used to compare different classifiers, where we summarize the performance of each classifier into a single measurement. AUC measures the entire two-dimensional area underneath the ROC curve. It can be interpreted as the probability that the model ranks a random positive example more highly than a random negative example.

We present a table of our Classifiers and their performance metrics below:

Classification Method	**Accuracy**	**Precision**	**AUC**
Logistic Regression	80.44	80.57	88.55
RandomForest	81.31	81.26	88.44
K Nearest Neighbors	81.31	81.47	88.18
Gradient Boosting	84.10	84.00	90.05
ExtraTrees	80.09	80.08	87.40

From the table we can see that there are not very large differences between the models, when it comes to Accuracy and Precision, GradientBoosting is the best Classification method, while it falls very little behind Logistic Regression in the AUC column. Our recommendation would be the Gradient Boosting Classifier, since we are not interested in coefficients of predictors and their meaning. If that was the case, we would opt for the Logistic Regression Classifier, whose coefficients can be interpreted as logarithms of odds of events.

A plot of the ROC curves of the five models is shown below. No large differences between the models can be observed just by looking at the curves.



References

1. https://www.sciencedirect.com/science/article/pii/S1098301510604950
2. https://curanthealth.com/top-barriers-to-patient-persistence/
3. https://www.pharmexec.com/view/top-barriers-patient-persistence
4. https://faculty.wharton.upenn.edu/wp-content/uploads/2012/04/FS8.Lee-Fader-Hardie.pdf
5. 1976. “Inference and Missing Data.” *Biometrika* 63 (3): 581–90
6. https://scikit-learn.org/stable/modules/impute.html#knnimpute
7. https://www.lung.org/lung-health-diseases/lung-disease-lookup/nontuberculous-mycobacteria/diagnosing-and-treating-ntm
8. https://www.nof.org/patients/diagnosis-information/bone-density-examtesting/
9. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4470303
10. https://journals.lww.com/md-journal/Fulltext/2019/11080/Incidence,_comorbidities,_and_treatment_patterns.46.aspx