
CS-2750 PROJECT REPORT: EVALUATION METRICS FOR UNBALANCED DATA AND FOR DATA SYMMETRY

Swapnil Asawa

swal2@pitt.edu

1 ABSTRACT

The most popular reported metrics namely Accuracy, Precision, Recall, and F1-score are not good indicators when the data is imbalanced. We must know which evaluations we should be skeptical about and when. Also, some of these matrices are not symmetric i.e. if we exchange just the labels of positives and negatives, good measures may turn poor and vice versa. Ideally this is not a pleasant property as mostly we give equal importance to all the classes equally irrespective to the data unbalanced. The data mostly is available unbalanced naturally. Hence, it is important to have matrices which are robust to it. In this work, we do a robust comparison of various evaluation metrics on data imbalance and symmetry and analyse their results. Furthermore, we propose a evaluation metrics, TPNR, which is robust to these changes.

2 INTRODUCTION

Data imbalance is a naturally occurring phenomenon in many cases whereas the natural desire is to have models which are robust to these and don't produce results biased for classes which have more amount of data points comparatively. The models which completely give wrong results on other classes are of mere use and maybe doing the job of classifying everything as the majority class. Moreover, mostly it is desired to give equal importance, i.e. symmetry, to all the classes unless deliberately stated otherwise. Unfortunately, most of the metrics we report are biased for positive classes which even includes F-1 score!

We reviewed multiple works on data imbalance and symmetry. We do a robust comparison of various evaluation metrics on data imbalance and symmetry and analyse their results in this work in the experimentation section. Furthermore, we propose a evaluation metric in the methodology, TPNR, which is robust to these changes. We give a further discussion of why the way results are behaving in certain way in discussions of result section. Our final conclusions about each of the evaluation metrics performance in conclusions.

We used Dataset from the Kaggle Competition: Predicting pulsar star in the universe (1). HTRU2 is a data set which describes a sample of pulsar candidates collected during the High Time Resolution Universe Survey. Pulsars are a rare type of Neutron star that produce radio emission detectable here on Earth. They are of considerable scientific interest as probes of space-time, the inter-stellar medium, and states of matter. As pulsars rotate, their emission beam sweeps across the sky, and when this crosses our line of sight, produces a detectable pattern of broadband radio emission. As pulsars rotate rapidly, this pattern repeats periodically. Thus pulsar search involves looking for periodic radio signals with large radio telescopes. Each pulsar produces a slightly different emission pattern, which varies slightly with each rotation. Thus a potential signal detection known as a 'candidate', is averaged over many rotations of the pulsar, as determined by the length of an observation. In the absence of additional info, each candidate could potentially describe a real pulsar. However in practice almost all detections are caused by radio frequency interference (RFI) and noise, making legitimate signals hard to find. The data has the following attributes from the signal patterns: Mean of the integrated profile, Standard deviation of the integrated profile, Excess kurtosis of the integrated profile, Skewness of the integrated profile, Mean of the DM-SNR curve, Standard deviation of the DM-SNR curve, Excess kurtosis of the DM-SNR curve, Skewness of the

DM-SNR curve. The response are 2 classes : if the star is pulsar or not. The data is highly ubalanced with 1:10 ratio of positive to negative and hence highly suitable for our experimentation.

3 RELATED WORK

We reviewed multiple works on data imbalance. In the past the problem of data imbalance and symmetry has been addressed.

3.1 MATTHEWS CORRELATION COEFFICIENT

The MCC metric has been first introduced by B.W. Matthews to assess the performance of protein secondary structure prediction (2). Then, it becomes a widely used performance measure in biomedical research. MCC and Area Under ROC Curve(3) (AUC)(4) have been chosen as the elective metric in the US FDA-led initiative MAQC-II that aims to reach a consensus on the best practices for development and validation of predictive models for personalized medicine [16].

3.2 COHAN'S KAPPA

Cohan's Kappa (5) is the proportion of agreement corrected for chance, and scaled to vary from -1 to +1 so that a negative value indicates poorer than chance agreement, zero indicates exactly chance agreement, and a positive value indicates better than chance agreement. A value of unity indicates perfect agreement. The use of kappa implicitly assumes that all disagreements are equally serious.

Matthews Correlation Coefficient and Cohan's kappa evaluation matrices are found in the experiments to take care of symmetry but not ideally data imbalance.

4 METHODOLOGY

We first pre-processing the data. We standardize all the 8 attributes. Then we visualization and Analysis of data. We found correlations with of all attributes as well as the correlation with the attributes and target variable. Then we used top 2 variables for the analysis. We also used PCA and used top 2 PCs for further analysis. The results from PCA were better so we choose report only PCA results. The aim here is to compare evaluation matrices but not to improve the models so we do not need cross validation and other techniques for the same. Simple train test split is suffice thus we perform train and test set split. The Machine Learning model we use is variations of Decision Trees. The models get trained on the training set. We finally evaluate those results on the test set by calculating and visualizing evaluation metrics(6) and test multiple Evaluation Matrices. We compare these results when we swap the target variables to understand symmetry issues with the evaluation Matrices. We also compare these results with ones got from the balanced test set data to analyse the robustness of these matrices to data imbalance.

4.1 STANDARDIZE AND VISUALISE

We standardize all the 8 attributes to avoid any bias from data scales. We visualise the attributes w.r.t target variables.

We also visualize correlations. We found correlations with of all attributes as well as the correlation with the attributes and target variable. We found that Attribute 1,3,4 are highly correlated with the data and attribute 1 and 3 are least correlated from then and hence chose those 2 to train the model.

We also used PCA and used top 2 PCs for further analysis. The results from PCA were better so we choose report only PCA results for the learning and testing parts.

4.2 TRAINING

The Machine Learning model we use is variations of Decision Trees. The models get trained on the training set. Running multiple configuration of DECISION TREES. 1st tree has 1 split. 2nd tree has 5 split. 3rd tree has 10 split. 4th tree can split as many times on training data to make final leafs completely pure.

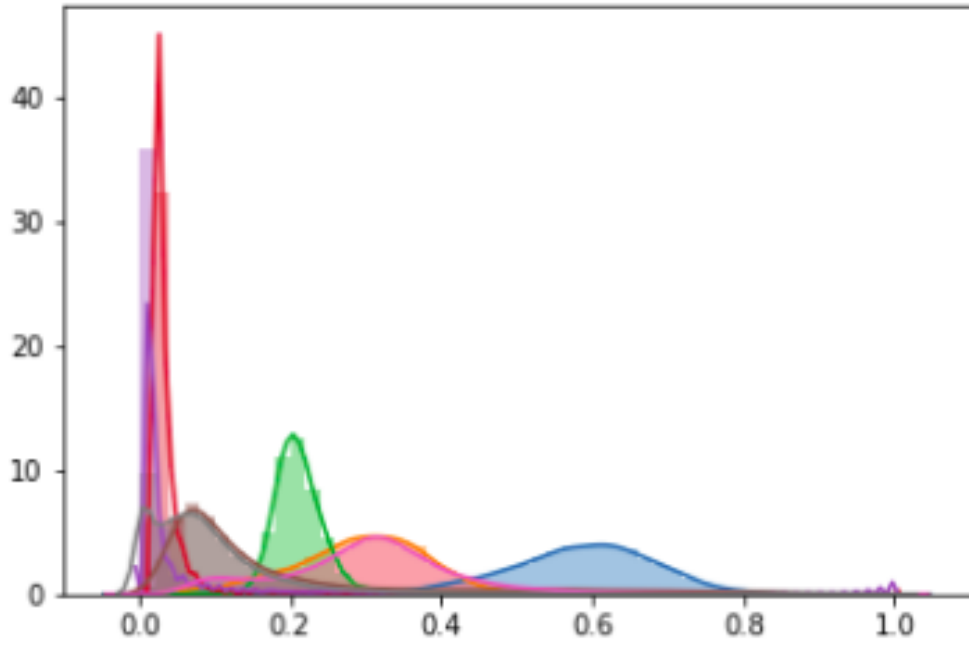


Figure 1: Attributes densities

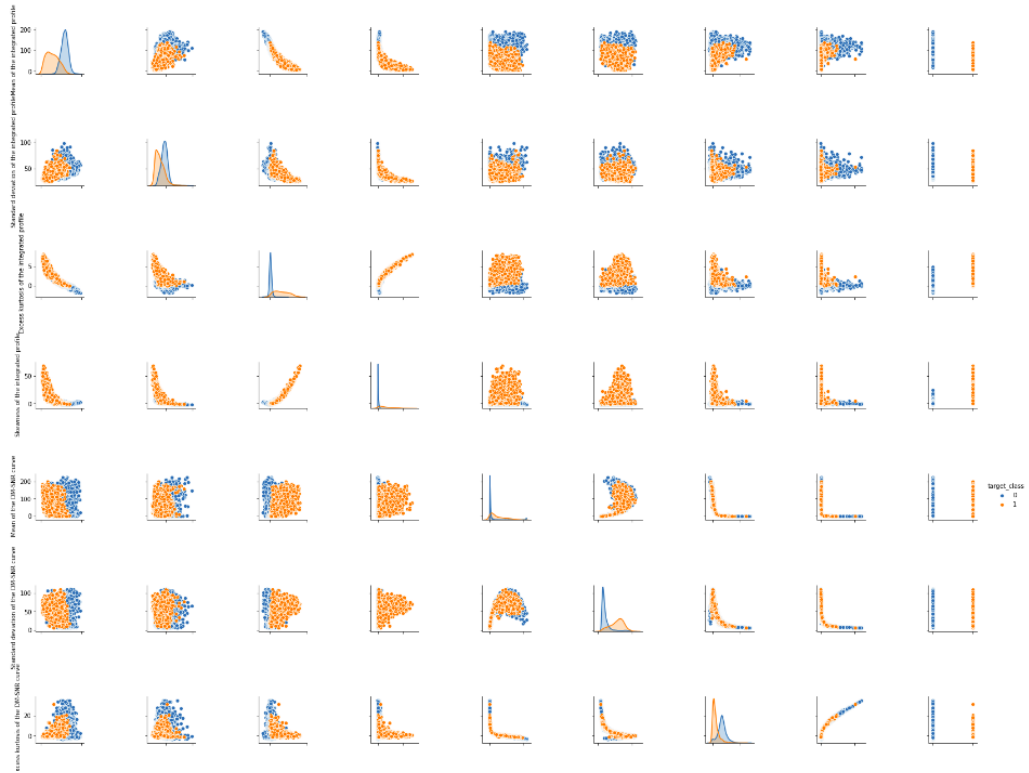


Figure 2: Correlations

4.3 EVALUTION

Scoring is done on various different evaluation matrix namely famous ones Accuracy, precision, recall, F1-score, specificity, ROC, auc, not so famously reported matthew, cohans kappa and, our invented TPNR

4.4 TPNR

We designed novel evaluation matrix, TPNR which is simply : True Positive Rate* True Negative Rate TPR is essentially the ratio of Positives detected, and TNR is the ratio of Negatives detected. We beleive this metric is very good for data imbalance as well as for symmetry because it gives positive and negative class equal important. It gives equal weight to both positive and negative classes. It makes evaluation unbiased of data imbalance because ratios are used instead of actual numbers. We will also test this with the other famous and published metrics.

5 EXPERIMENTAL RESULTS

5.1 RESULTS ON IMBALANCED DATA

All the decision trees are tested on test data. Figure 3 shows the actual evaluation Metrics values on different decision trees on the unbalanced data.

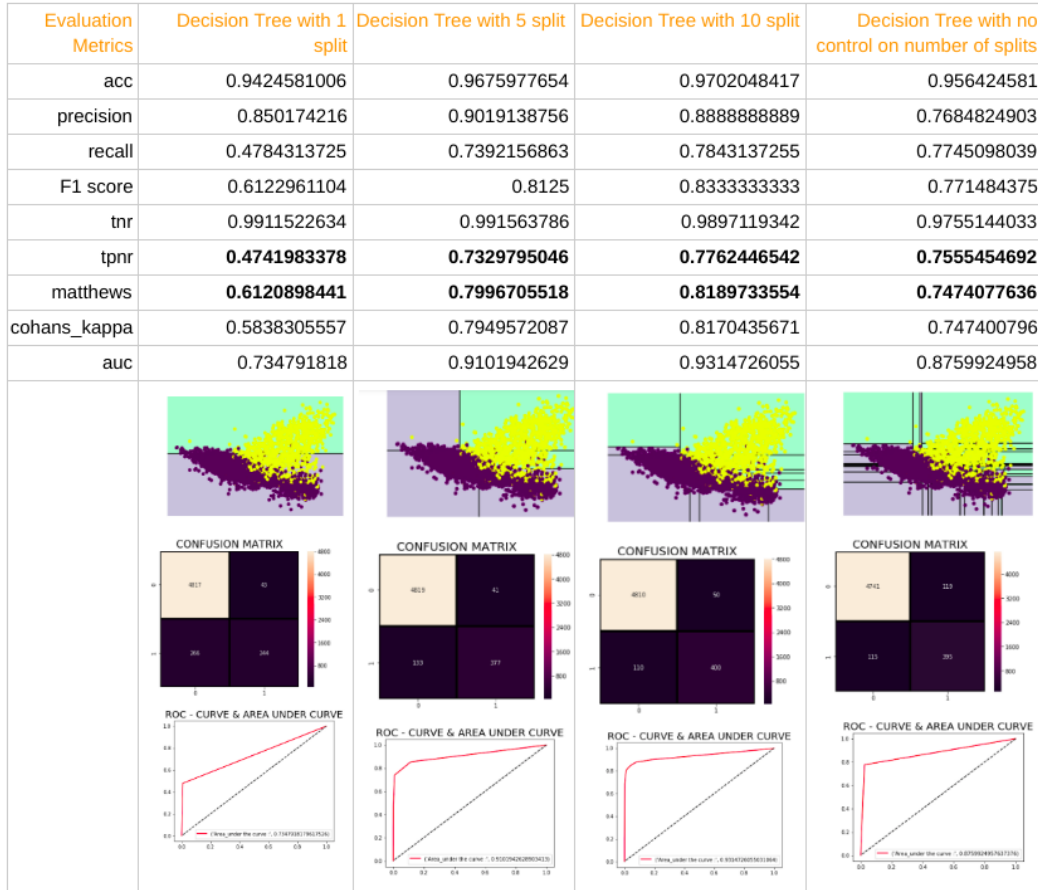


Figure 3: Various evaluation Metrics results for different Decision Trees results on test set. 1st column is of Decision tree with 1 split. 2nd column is of Decision tree with 5 split. 3rd column is of Decision tree with 10 split. Last column is of Decision tree with no control on number of split on training data.

5.2 CHANGES ON BALANCING DATA

Figure 4 shows change in values of evaluation matrices when the test data has been balanced from unbalanced. Notice that these are change in values and not original values of the matrices. As immunity to data imbalance is desired, if the change is 0 or near to 0, the metric is good otherwise not. 1st 2 columns are of Decision tree with 1 split. 2nd 2 columns are of Decision tree with 5 split. 3rd 2 columns are of Decision tree with 10 split. Last 2 columns are of Decision tree with no control on number of split on training data. 1st, 3rd, 5th, 7th columns of tree are having unbalanced test data. The 2nd, 4th, 6th, 8th columns of tree are having balanced test data.

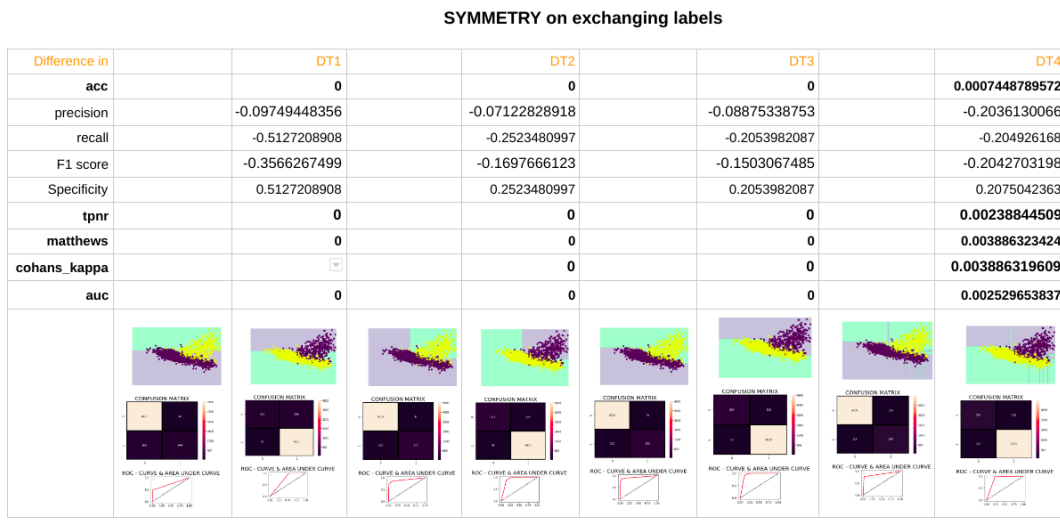


Figure 4: Change in values of evaluation matrices when the test data has been balanced from unbalanced. Notice that these are change in values and not original values of the matrices. As immunity to data imbalance is desired, if the change is 0 or near to 0, the metric is good otherwise not. 1st, 3rd, 5th, 7th columns of tree are having unbalanced test data. The 2nd, 4th, 6th, 8th columns of tree are having balanced test data.

We can see recall(TPR),specificity(TNR), TPNR, and area under ROC(AUC) are insensitive to data imbalance, so they are good for data imbalance.

5.3 CHANGES ON FLIPPING LABELS/SYMMETRY

Figure 5 shows change in values of evaluation matrices when the train as well as the test data has been just flipped the label values to evaluate symmetry. Notice that these are change in values and not original values of the matrices. As symmetry is desired for the cases where we give equal importance to each of the labels, if the change is 0 or near to 0, the metric is good otherwise not.

We see Accuracy, TPNR, Cohen's Kappa, Matthews CC and AUC are insensitive to data imbalance, so they are good for data symmetry.

5.4 RESULTS

The winning metrics are TPNR and Area under ROC which survived both data symmetry test as well as data imbalance test.

6 DISCUSSIONS

We wondered why AUC is also doing a good job? We figured that is because AUC is area under sensitivity* (1-specificity). This is same as area under sensitivity* specificity. As sensitivity is TPR and specificity is TNR, its Area under TPR*TNR But AUC has some cons as well. Firstly, it is not

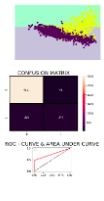
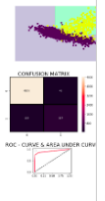
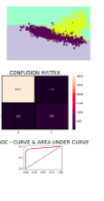
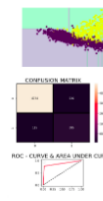
| Difference in values when balancing the unbalanced data | | | | | | | | |
|---|---|--|---|--|---|--|---|--|
| Difference in | DT1 | | DT2 | | DT3 | | DT4 | |
| acc | 0.2091247672 | | 0.1009310987 | | 0.0819695476 | | 0.08308138898 | |
| precision | -0.125825784 | | -0.09019138756 | | -0.101210121 | | -0.1914014885 | |
| recall | 0 | | 0 | | 0 | | -0.005882352941 | |
| F1 score | -0.02980915274 | | -0.03469101124 | | -0.04194018964 | | -0.08859039807 | |
| Specificity | 0.002916969257 | | -0.002553861051 | | -0.002444928589 | | 0.005543451949 | |
| tpnr | 0.001395569605 | | -0.001887854149 | | -0.00191759105 | | -0.001437623706 | |
| matthews | 0.06963750012 | | 0.04128540801 | | 0.02516777564 | | -0.01416853969 | |
| Cohen's Kappa | 0.117163889 | | 0.06162387536 | | 0.0405729789 | | 0.0003169298247 | |
| auc | 0.001458484628 | | -0.000999508736 | | -0.002204826254 | | 0.001071169208 | |
| |  | |  | |  | |  | |

Figure 5: shows change in values of evaluation matrices when the train as well as the test data has been just flipped the label values to evaluate symmetry. As symmetry is desired for the cases where we give equal importance to each of the labels, if the change is 0 or near to 0, the metric is good otherwise not.

a very intuitive measure. Secondly, AUC becomes complicated to extend to multi classes. It requires pair wise comparison of AUC(one vs all or 1 vs others). On the other hand, TPNR can be easily extended as product of ratios of correct detection of classes. Thirdly, AUC integrate the result on all the possible cutoffs to predict positive and negative classes which is not what acutally happens in the final model results. There is only one cutoff used while testing.

We suggest to use ROC first to find the accurate cutoff mark for classifying into positive and negatives and then report the TPNR values as one of the evaluation measure.

We forgot to include per class accuracy in the list of our evaluation measures. Per class accuracy is a good measure and could be the 3rd winner too. Per class accuracy also uses ratios to balance data like TPNR. Unlike TPNR, instead of multiplying ratios of true predictions of classes, it averages the ratios. We believe that choosing between TPNR or per class accuracy depends on usecase to usecase. If the use-case is 50 percent satisfied when one class is detected perfectly and other is detected completely incorrect, then per class accuracy is better because it doesn't penalize heavily based on individual class performance. On the other hand, when performance of each class should be good and one class performing bad is not acceptable, TPNR is the best.

7 CONCLUSION

We performed various forms of evaluations on the test data to see how they behave with data imbalance and to test if they are symmetric. We devised a new good evaluation metric TPNR which is robust to there effects. We found AUC and TPNR are the best performers. We analysed the relation between TPNR and AUC. We concluded that best would be to use ROC first to find the accurate cutoff mark for classifying into positive and negatives and then report the TPNR values as one of the evaluation measure. We also discussed per class accuracy and how it is useful in different scenarios than TPNR and where each one would be appropriate.

REFERENCES

- [1] Predicting a pulsar star. <https://www.kaggle.com/pavanraj159/predicting-a-pulsar-star>.
- [2] Brian W. Matthews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *biochimica et biophysica acta (bba)-protein structure* 405.2 (1975): 442-451.
- [3] David M. DeLong DeLong, Elizabeth R. and Daniel L. Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *biometrics* 44.3 (1988): 837-845.
- [4] James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *radiology* 143.1 (1982): 29-36.
- [5] Jacob. Cohen. A coefficient of agreement for nominal scales. *educational and psychological measurement* 20.1 (1960): 37-46.
- [6] Pranav Raj. Code for visualizing confusion metrics. <https://www.kaggle.com/pavanraj159/predicting-a-pulsar-star/kernels>.