

CMPE 188 Homework Assignment

Instructor: Jahan Ghofraniha

Decision Trees using scikit-learn

1. What are the definitions of precision and recall? Explain why there is a tradeoff between the two? You can use graphs or any other tools to answer this question.
2. What is the definition of F1 score and how do you interpret a high F1 score?
3. Use the baseball salary dataset and the exploratory data analysis to determine visually which are the candidate features for the model. Use the $\log(\text{salary})$ as your output and pick six features as input for your data (use the exploratory analysis as a basis for the choice of input features).
4. Develop a regression decision tree model for the dataset base on default setting of the regressor, (i.e. use `DecisionTreeRegressor()` without any input. Check out the documentation for `DecisionTreeRegressor()` in Scikit learn library). You can save the image of the decision tree by right clicking on the console and save image/copy image and paste it into a paint or any other graphic package and save it as a .png file so you can look at it.
5. How many levels does the default decision tree have based on the six features (tree depth)?
6. Run the system again with a depth of 2 and compare the performance measures in the two cases. Explain the difference in the context of performance measure as it relates to variance/bias trade off.
7. Try the code a third time with a depth of 3 and make the same comparison as in step 6.
8. If you were to optimize this system using the tree depth (`max_depth`) as the hyper-parameter what would be your suggestion?
9. Use the bank note authentication dataset as given in the sample code to build a classification tree.
10. Use the sample code given on Canvas to plot the recall/precision curve for authentication dataset.
11. Based on step 10 choose a suitable threshold to achieve a high precision. Would this strategy work in every case? It seems with a high enough threshold you can achieve a good precision, is this method a good approach to optimizing your classifier?

12. Plot the ROC curve and use the `roc_auc_score` function (please check out the description and documentation in Scikit learn library; https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html) to calculate ROC score.
13. Include your code, the results and explanation of the results either as a Jupyter file (.ipynb) or submit your Python code plus your plots as screen shots/pictures in a word/pdf file along with your Python code (*.py).