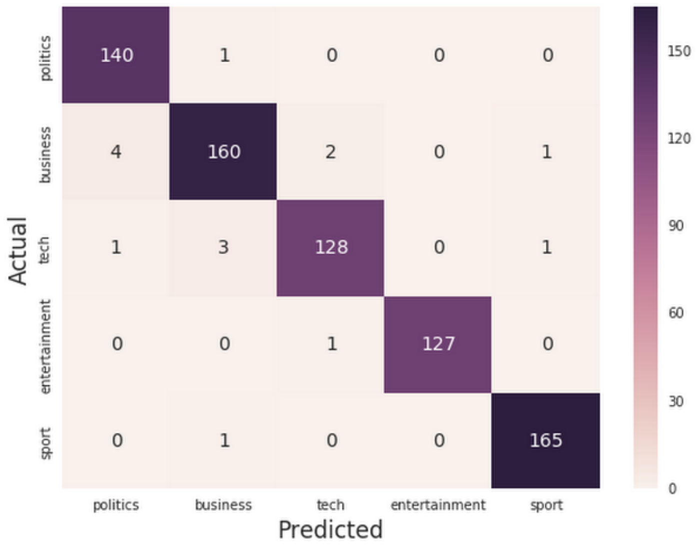
CSCE822 Homework 2

Problem 1: Classifier evaluation measures

Given the following confusion matrix, manually calculate all the following performance measures of the classier

Accuracy, precision and recall (for each class), AUC score, true positive rate, false positive rate, specificity, sensitivity



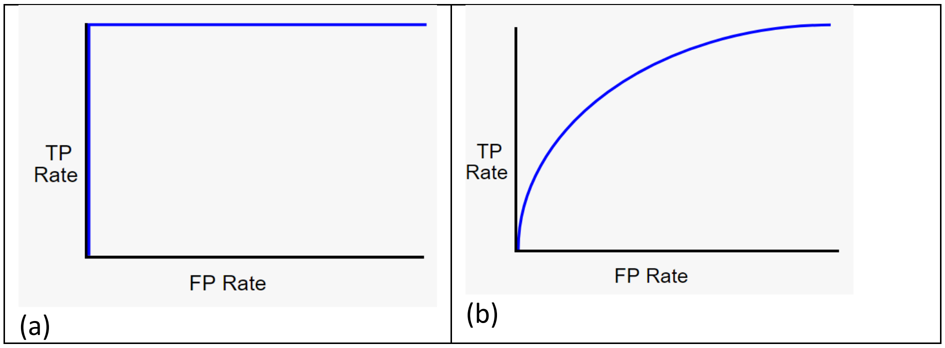
	politics	business	tech	entertainment	sport
politics	140	1	0	0	0
business	4	160	2	0	1
tech	1	3	128	0	1
entertainment	0	0	1	127	0
sport	0	1	0	0	165

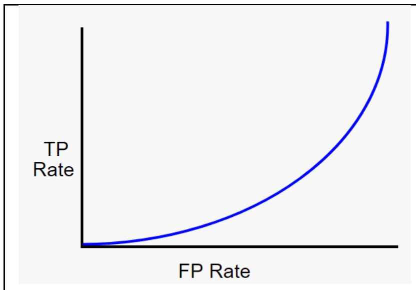
Check your understanding by taking this test:

https://developers.google.com/machine-learning/crash-course/classification/check-your-understanding-accuracy-precision-recall

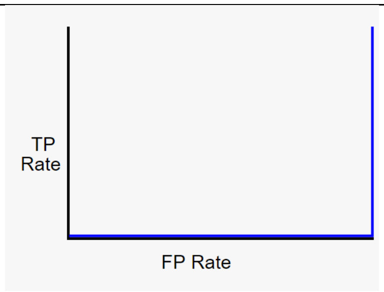
put down how many questions you answered correctly.

Describe the property of the classifiers that have the following ROC curves.





(c)



(d)

https://developers.google.com/machine-learning/crash-course/classification/check-your-understanding-roc-and-auc

Problem 2: Classification of e-tailer customers (Real-world problem) using Support vector machines. You can use weka or Scikit-learn python programming.

Objectives: E-commerce Customer Identification (Raw). Try to get the best performance using preprocessing, feature selection, data balancing, and parameter tuning.

The task involves binary classification to determine customers of the e-tailer. The training data contains 334 variables for a known set of 10000 customers and non-customers with a ratio of 1:10, respectively. The test data consists of a set of examples and is drawn from the same distribution as the training set.

Data: The feature data is train.csv and the label data is train_label.csv with corresponding labels for the records in train.csv. The test.csv is the test data.

Preprocessing steps to do:

- You may use excel or write a simple script to merge the feature data file with label data file and save as csv file, then you can import into weka system.
- **Missing values**: Check if there are any missing values inside the dataset, if so, use Weka's missing value estimation filter to estimate the missing values to make the data complete
- **Normalization**: since the features have very different value ranges, apply weka's normalization procedure to make them comparable.
- **Attribute/Feature selection**: Since there are 334 features in the dataset, it may be useful to use some feature/attribute selection to reduce the dataset before training classifiers. Select one method

(weka->filters->supervised->attribute->attributeSelection) to do feature selection. Describe your selected method and explain how it works briefly.

- **Hint1**: after you import the merged csv file into weka, the class label 1/0 is regarded as numeric value rather than nominal labels. You need to use the weka->filter->unsupervised->attribute->numeric2Nominal filter to convert that column to nominal class. (you need to specify which column is your class label to apply this conversion) Also note that weka take first line as feature names!! So need to add a line of feature names.
- **Hint2**: The dataset is a severely unbalanced dataset. You may want to balance the data before training the classifier.
- **Hint3**: if your training data has been applied a set of normalization or feature selection, you need to do the same with test dataset, otherwise the feature values are not consistent, and you will get absurd results on test data.
- **Hint5**: The best AUC value for this problem is 0.6821. See what u can get.

Experiments to do:

1) Experiments on the training dataset

You will need to build a classifier using a SVM algorithms to classify the data into customers and non-customers and evaluate their performance.

- Pick one decision tree algorithm from Weka such as J48graft and describe it. (there are many decision tree algorithms)
- Explain pre-processing filters in the table below. Run your decision tree algorithm with the default parameters. This is to learn how the preprocessing affects performance.
- Write down the corresponding performance measures for class 1 (customer) in the following table for each processing
- All measures are based on **10-fold cross-validation results (except the last row)**. Put your results in Table 1 (below)

2) Use your best classifier you trained in step one, predict the class labels for the test dataset test10000.csv. Save your prediction labels into the predict.csv file.

   Write a program to calculate precision, recall, MCC (check the definition here http://en.wikipedia.org/wiki/Receiver_operating_characteristic#Further_interpretations) using the true labels in the test10000_label.csv and the predicted labels in your predict.csv file.

Table 1. Comparing performances of classifiers on test dataset

Algorithm performance	Precision	Recall	MCC	ROC area
SVM (10-fold CV)				
Result on test data				

Report requirement:

1) Describe the preprocessing methods you used in the above experiments: missing value estimation, normalization, attribute selection, random forest
2) Report the performance results in Table 1
3) Submit the program to calculate the performance measures: Precision, Recall, MCC from two label files.

References on unbalanced data handling

1. https://medium.com/james-blogs/handling-imbalanced-data-in-classification-problems-7de598c1059f
2. https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28
3. https://medium.com/strands-tech-corner/unbalanced-datasets-what-to-do-144e0552d9cd
4. https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/

Problem 3: Regression using deep neural networks.

The problem here is to develop a regression model that can beat a theory model.

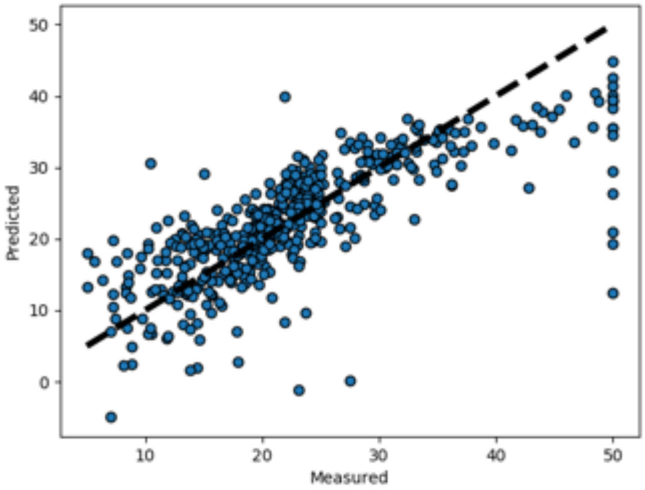
Attached thermal-data.xlsx contain a dataset for material thermal conductivity.

Develop a deep neural network regression program to predict the thermal conductivity (y-exp) using the all the features before it.
(V,M,n,np,B,G,E,v,H,B',G',ρ,vL,vS,va,Θe,γel,γes,γe,A,).

Report the MSE, RMSE, MAE, R^2 of 10-fold cross-validation.
Compare the MSE, RMSE, MAE, R^2 of the theoretical model using the values in column y-theory

Try to tune your parameters of the models to achieve the best performance.

Plot the final scatter plot for your best model/result. The better the points are around the diagonal line the better your model is.



Cross-validation example code can be found here

https://www.analyticsvidhya.com/blog/2018/05/improve-model-performance-cross-validation-in-python-r/

If you have any question about the requirements of the assignment, please send me email hujianju@gmail.com

Submit all your code and reports to:

http://dropbox.cse.sc.edu