DAI_ASSIGNMENT_1_Q2_Report.pdf

Dependable AI CSL7370 – Assignment 1

Task 2:

Contents:

Theory➔

1. Bias
2. Fashion MNIST
3. Cross validation
4. Linear SVM
5. 5 Layer Neural Network (MLP)
6. Testing Performance (mean ± STD)
7. Confusion Matrix
8. TP,FP,TN,FN, TPR, FPR,TNR,FNR
9. ROC and Curve importance
10. EER Equal Error Rate and curve importance
11. Precision Recall and curve importance
12. Which curve to prefer for imbalanced data and Why?

Results➔

1. Testing Performance (mean ± STD)
2. Testing Accuracy comparison of SVM vs ANN
3. Confusion Matrices for SVM and ANN
4. ROC Curve
5. Precision Recall Curve
6. Inferences from observed results and comparison of approaches.

Theory

1. **Bias:** Originally, bias is anything that if its value is high then the Machine Learning model will oversimplify the training and test cases. Bias in machine learning models turns high error. "*Model with high bias pays very less attention to the training data.*" In paradigm of Dependable AI we introduce bias as a partiality of a machine learning model towards the query data. This is because the model has learned such patterns during the training. May be because of the training samples are inadequate or called as data bias. Apart from this there are numerous biases devised out by research community. Some of them are: Representation bias, Evaluation bias, Population bias, Sample bias, Prejudice bias, Confirmation bias, Hypothesis bias. Specifically, Bias in data includes, Data Gathering or Geographical factors bias (example where data is gathered specifically from freely available resources mainly western country data and then applied on different parts of the world), Data pre-processing bias, Confirmation bias.
2. **Fashion MNIST**: Originated from a European based E-commerce company, the dataset is freely available with MNIST. Zalando SE published 10 class of items with a total of 70000 example images (28x28) spread over 10 classes. The dataset can be accessed bias tensor-flow package as well.
3. **Cross Validation**: Evaluating the test dataset (How the model works on an unseen dataset) Sometimes with limited data examples we resample the same dataset with different thresholds such that we get to learn more knowledge about the dataset. One can think that cross validation will overfit the dataset since it is learning complete dataset iteratively. Howebver this is not the case, because you are not considering whole dataset at a time. You try to create multiple train test splits so as to fine tune your learning model.
4. **Linear SVM**: A faster Support Vector Machine model with linear kernel. (Mostly used when the data is linearly separable.)
5. **5 Layer ANN/MLP**: Before jumping what we require to use in our dataset let us know the differences between MLP and ANN. MLP is a subset of ANN. Most of the times both words are used interchangeably. The inherent difference is that, MLP are always feed forward while ANN can have loops. More on : here
6. **Testing Performance** (mean ± STD)**:** Mean accuracy and Deviation tells the metrics about the average accuracy of the model (Since we trained with 3 different training sets our each of the SVM and ANN models). This tells the average accuracy of the models. Additionally, Deviation is the percentage amount that the model is not performing well.
7. **Confusion matrix:** Confusion Matrix tells about the true positives, true negatives, false positive, false negatives. In simple terms, Confusion matrix tells about how many times "Coat" class is classified as pullover and coat actually. And same for "Pullover". This is best way to evaluate your model. That is How many times our model was confusion with pullover but it was coat in reality. This is where we need to create a predicted label array along with the original labels. So as to identify the parameters.

8. **TP,FP,TN,FN, TPR, FPR,TNR,FNR:** for better performance, TPR and TNR should be high and rest low. Because we want prediction to be similar to the ground truth.
    a. True Positive: # of times coat is predicted as coat
    b. False Positive: # of times pullover is predicted as coat
    c. True Negative: # of times pullover is predicted as pullover
    d. False Negative: # of times coat is predicted as pullover.
    e. True Positive Rate: #tp/#tp+#fn
    f. False Positive Rate: #fp/#tn+#fp
    g. True Negative Rate: #tn/#tn+#fp
    h. False negative Rate: #fn/#tp+#fn
9. ROC Curve and its importance: ROC Curve is simpler measure of confusion matrices. Receiver Operator Characteristics curve is simple way to summarize. The vertical axis of the graph is sensitivity (TPR) and x axis is (1-specificity) FPR with varying thresholds. Threshold is a kind of decision boundary which enables the samples to classify into their respective classes. With increasing threshold you allow more samples of class 0 (in our case 0 is pullover) The importance of ROC curve is that, Classifiers that give curves closer to the top-left corner indicate a better performance. We envision higher TPR because we want more predictions to match with ground truth. ROC curve is suitable for balanced datasets. Because ROC curve may give you wrong hopes of model being accurate. However in reality you were dealing with 90% data of coats and 10% data of pullover.

An **ROC curve** (**receiver operating characteristic curve**) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

- True Positive Rate
- False Positive Rate

True Positive Rate (**TPR**) is a synonym for recall and is therefore defined as follows:

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate (**FPR**) is defined as follows:

$$FPR = \frac{FP}{FP + TN}$$

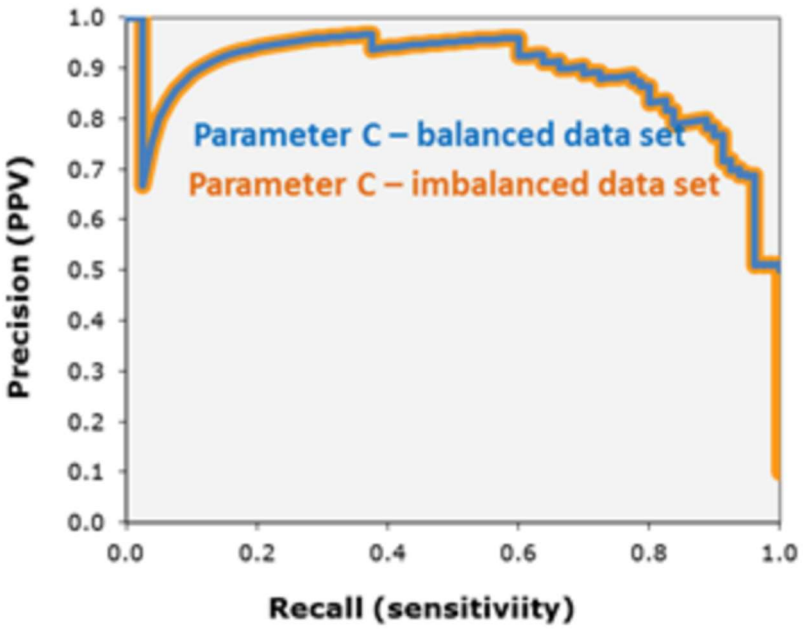
Figure 1 from google developers machine learning crash course

10. EER Equal Error Rate and curve importance: The point where the graph intersects is the point where the # of false acceptance is equal to # of false rejections. The lower the equal error rate value, the higher the accuracy of the out model. Below the intersection point, we accept low false predictions and higher false rejections, and above the intersection point, we accept more false predictions and less false rejections.
11. Precision Recall and curve importance: Suitable for imbalanced data. (From question 1) In more simpler words, **Precision** is the number of times the coat got correctly classified divided by number of times the model assumed it was coat. However (TPR) **Recall** is number of times model classified it was coat divided by the number of times it was actually coat.

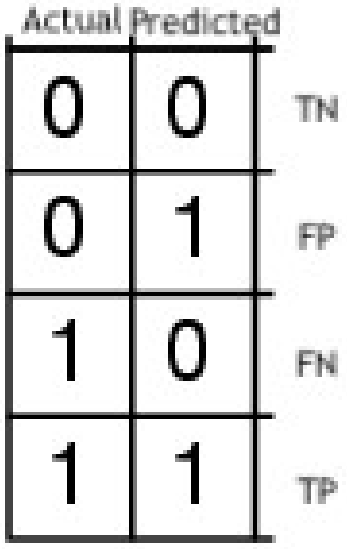
$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

12. PR Curve is a plot of Precision (vertical axis) and Recall (horizontal axis) measures at different thresholds. Because of this we can rely on PR Curve for imbalanced dataset. Even if the dataset is not balanced or the dataset is biased, we will not deal with different proportions of the data. We will check how our model performed within that particular class. As precision checks for correctness of the classification to the assumption for that class of classifier whereas recall checks for correctness of the classification to actual ground truth of that class of classifier. In general PR Curve overlaps for balanced and imbalanced datasets. However, the PR curve drop point tells the proportion of the classes you incorporated. In simpler terms, PR Curve tells about the biasness by the droppage of the curve to some value very high or very low according to the class your envision. However for balanced dataset it drops at 0.5. Additionally, we can say that when recall = 1, we shall check the precision such that we get to know about balanced and imbalanced datasets.



Whereas it is not a possible way to detect biasness in ROC Curves. Because of the biased dataset you will see a beautiful and perfect ROC Curve such that the curve for balanced dataset will be lower top left than to the curve of imbalanced dataset which will be higher top left. Thus we would first look at the recall values, then precision values. We are more cantered towards the actual values.



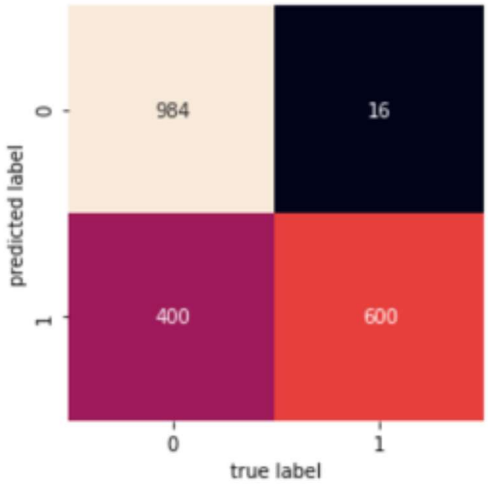
Results

1 and 2
Testing Performance (Mean ± STD) and Testing Accuracy comparison of SVM vs ANN:
SVM1 Accuracy = 71.3%
SVM2 Accuracy = 73.2%
SVM3 Accuracy = 69.55%
Mean Accuracy= (71.3+73.2+69.55)/3 = 71.35%
SVM1 Deviation = 71.3%-71.35% = -0.5
SVM2 Deviation = 73.2%-71.35% = 1.85
SVM3 Deviation = 69.55%-71.35% = -1.8 (This one was only shuffled) clearly we can see the change/deviation with shuffled data

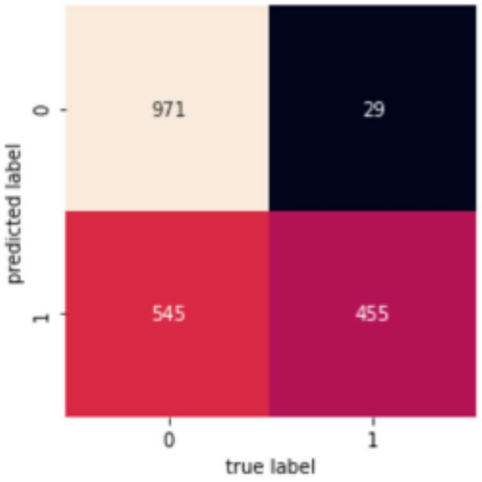
ANN1 Accuracy = 79.2%
ANN2 Accuracy = 80.9%
ANN3 Accuracy = 75.75%
Mean Accuracy= (**79.2**+**80.9**+**75.75**)/3 = 78.616%
ANN1 Deviation = 79.2%-78.616%**= 0.584**
ANN2 Deviation = 80.9%-78.616%**= 2.284**
ANN3 Deviation = 75.75%-78.616% **= -2.866 (This one was only shuffled)** clearly we can see the change/deviation with shuffled data

Confusion Matrices

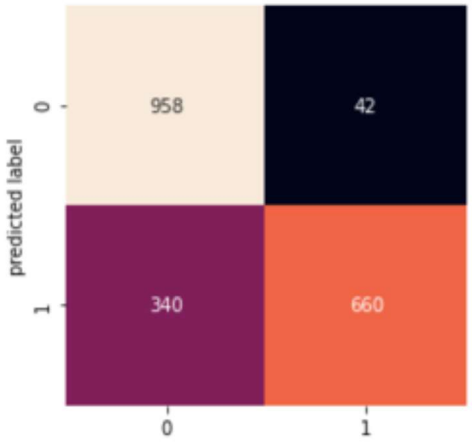
ANN1



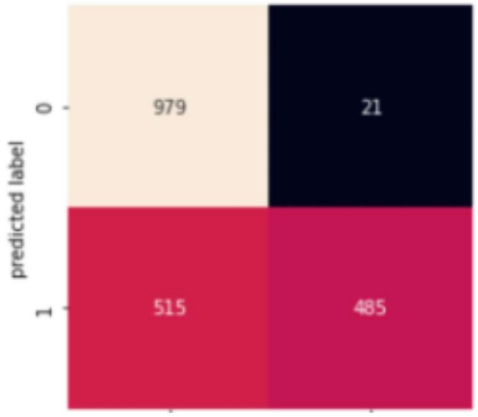
SVM1



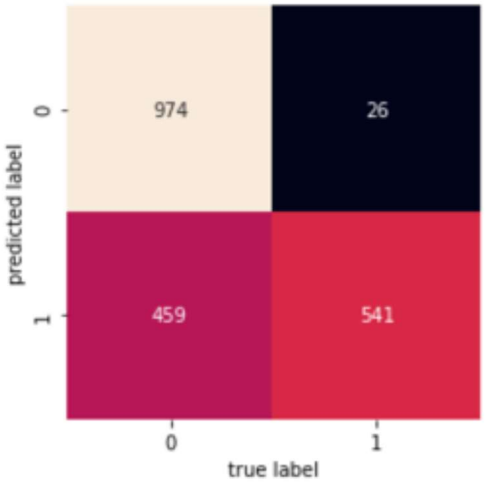
ANN2



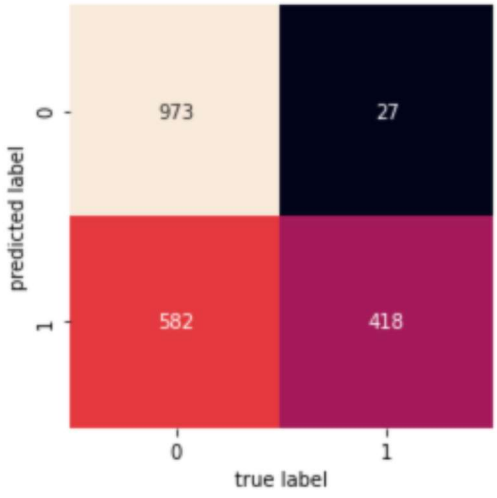
SVM2



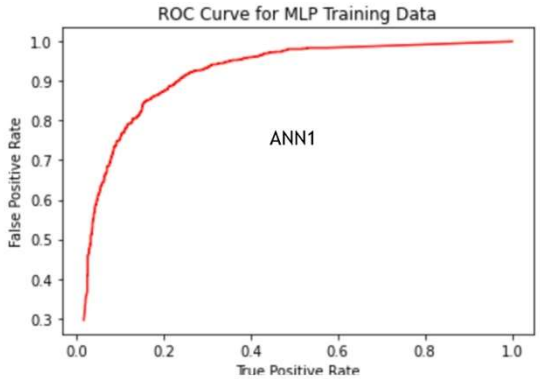
ANN3

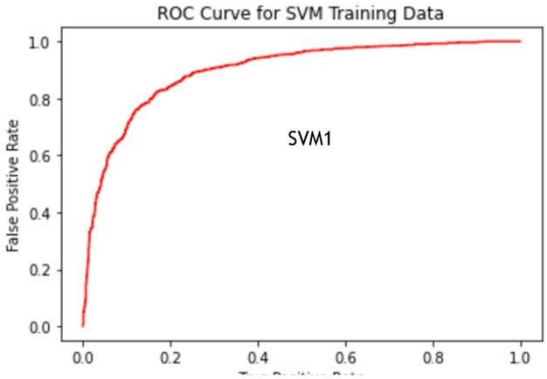


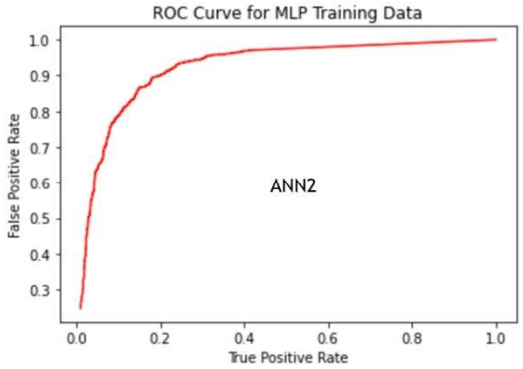
SVM3

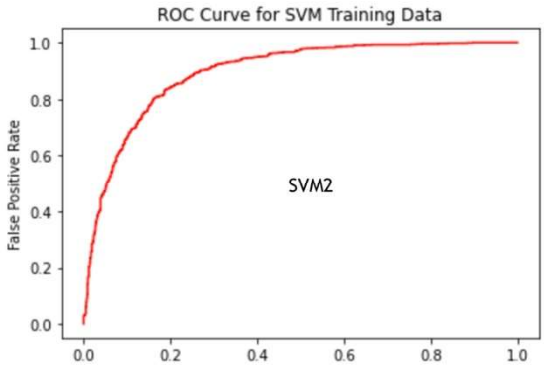


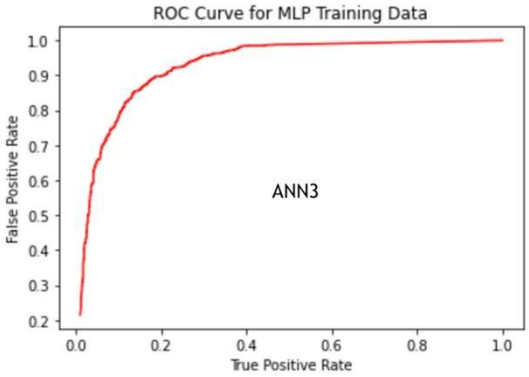
ROC Curves:

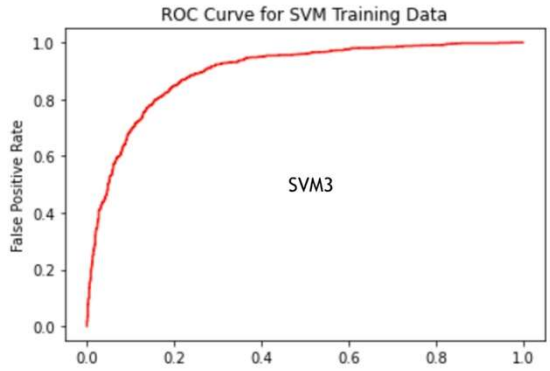

ROC Curve for MLP Training Data
ANN1
False Positive Rate
True Positive Rate


ROC Curve for SVM Training Data
SVM1
False Positive Rate

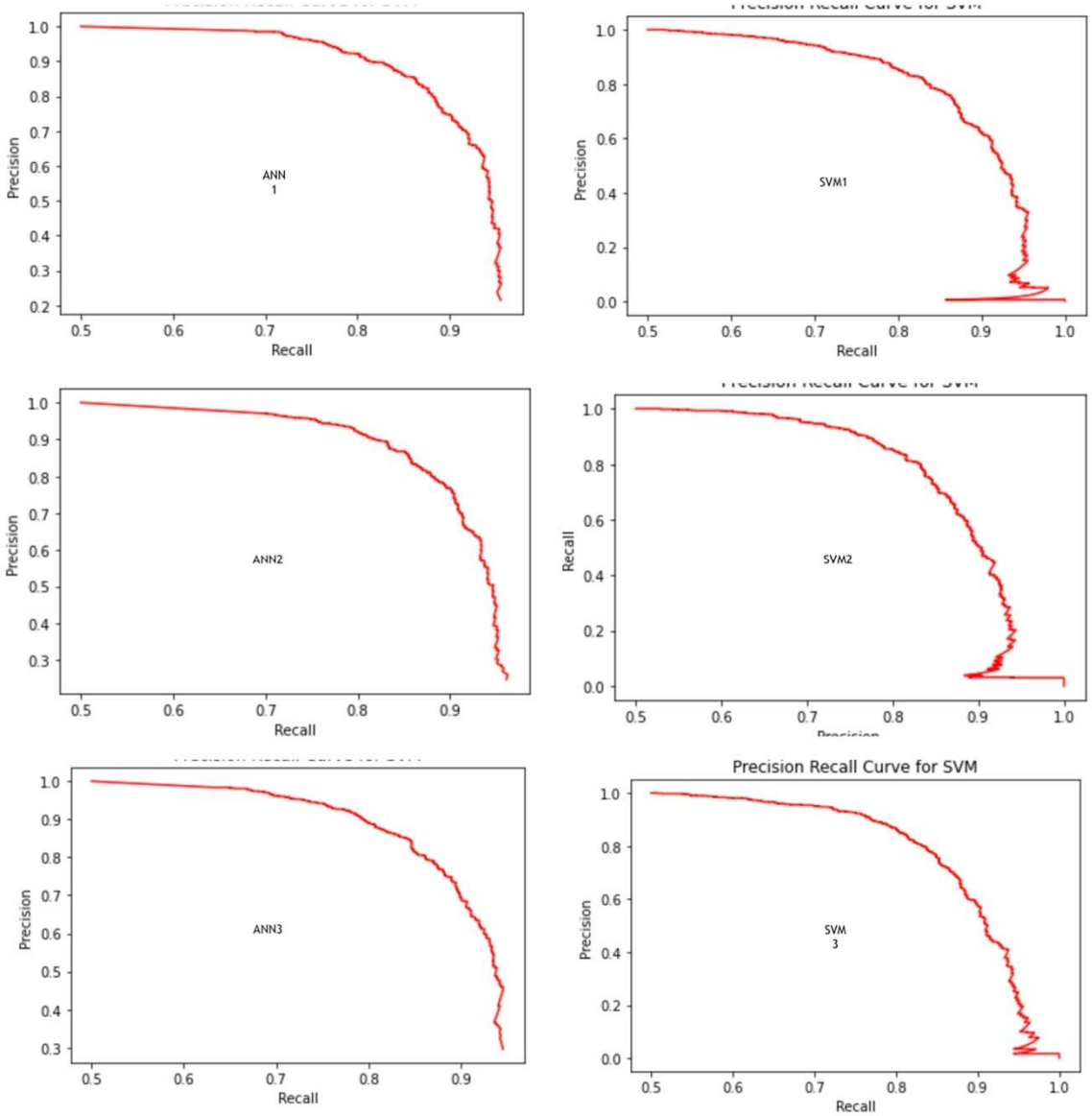

ROC Curve for MLP Training Data
ANN2
False Positive Rate
True Positive Rate


ROC Curve for SVM Training Data
SVM2
False Positive Rate


ROC Curve for MLP Training Data
ANN3
False Positive Rate
True Positive Rate


ROC Curve for SVM Training Data
SVM3
False Positive Rate

PR Curves:



Inferences:

From the complete experiment we come to conclusions:

1. Having large number of samples of a specific class creates a bias.
2. DNN performs better than SVM because Linear SVMs just uses a basic hyperplane methodologies and DNN learns the features. Yes, you can create non linear hyperplanes with different kernels however, DNN still outperforms without overfitting the dataset.
3. When we shuffled the dataset we got smaller deviations than non shuffled dataset.
4. ROC, PR and Confusion matrix are drawn to evaluate the metrics of different models.
5. Out of which we use ROC for balanced dataset and PR Curve for Imbalanced dataset.

Reference:

1. http://www.davidsbatista.net/blog/2018/08/19/NLP_Metrics/

2. https://acutecaretesting.org/en/articles/precision-recall-curves-what-are-they-and-how-are-they-used