IN4320 Machine Learning: Final Assignment
The E Corp Challenge

Name: Georgios Dimitropoulos
Netid: georgedimitrop
Student Number: 4727657

July 6, 2018

1 Introduction

In this work, we deal with a two-class classification problem in 204 dimensions, where we are given a small labeled train set and a large unlabeled test that has undergone a corruption of some sort in the form of additive uniform noise. Our goal is to implement the best performing classifier on the unlabeled data in terms of the error on this test set. Our approach is based on an Active Learning approach.
The main task of Active learning is to query as few data as possible in order to minimize the annotation cost while maximizing the learning performance through selecting the most valuable samples as labeled instances are very difficult, time-consuming, or expensive to be obtained [1]. Thus, our strategy is to try to identify the most informative subset of the unlabeled test set through adopting an appropriate criterion that measures the usefulness of these unlabeled instances. After that we pose queries for the labels of these informative unlabeled data instances to an "oracle". Finally, we feed these examples, together with the labels that have been predicted by the "oracle", in combination with the labeled train data which we have to our classifier in order to enhance its performance.
The rest of our work is organized as follows: Section 2 demonstrates the preprocessing steps and the Principal Component analysis (PCA) that we applied to our data. Section 3 shows an analysis that we made in order to evaluate several classifiers and to select and proceed with the one that achieves the best performance. In section 4 the Retraining-based Active Learning Procedure that we employ is presented. Section 5 contains the experiments that we made in order to build our final best performing classifier. In Section 6, we discuss the final procedure that we followed in order to estimate the performance of our work on the 20,000 unlabeled data set. Finally, Section 7 gives a discussion and summarizes our conclusions that are drawn from our work, followed by the References of the literature that we studied for this assignment.

2 Preprocessing and Principal Component Analysis (PCA)

2.1 Preprocessing of the data

Initially we observed that the scales of the values of our data varies widely among the different features in the labeled train set and in the unlabeled test set. Thus, for classifiers that are based on a distance metric in the features space like the Nearest Neighbor classifier or for the PCA method that is sensitive in the scaling of the features, the difference in the scaling among the features may cause bias to the objective function and to the choice of the principal components. Thus our first step is to normalize all features in order to have zero mean and unit standard deviation.

2.2 Principal Component Analysis (PCA)

Our second step is to perform PCA in our data in order to reduce the dimensionality by obtaining the most discriminative features and for computation efficiency. In order to decide how many dimensions to keep, we decided to keep the principal components that capture the 95% of the variability in the data. It can be depicted in Figure 1 that the 95% of the Cumulative Variance

is captured by the first 125 Principal Components. Hence, we decided to project our data in this 125-dimensional space and to preserve 125 features for our data.

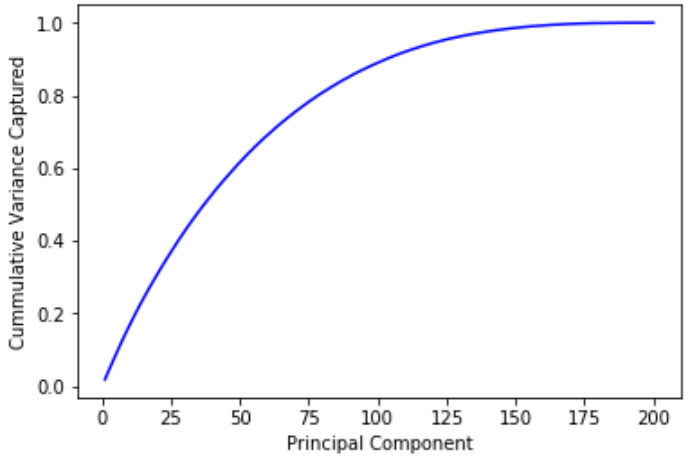

Figure 1: Cumulative Variance Captured vs Number of Principal Components

3 Initial Evaluation of Different Classifiers

In this section, we tried to evaluate several classifiers and to select and proceed with the one that achieves the best performance. In order to be able to do this we used a 10-fold cross-validation procedure to find the accuracy of several classifiers in the labeled training set. Thus we split this labeled training set of 200 samples in 10 folds where we train the classifiers with the first 9 folds and evaluate them with the last one and in each step we change the evaluation fold. The 10-fold cross-validation accuracy of each classifier can be depicted in Table 1. Based on these results we decided to proceed with the Logistic Regression classifier since it achieves the best 10-fold cross-validation accuracy.

Classifier	**10-fold cross-validation accuracy**
Logistic Regression	**76.5%**
Fisher	75.5%
Multi-layer Perceptron (MLP)	72.5%
Bayes	65.5%
5-Nearest Neighbor	58.5%
Decision Tree	54%
Random Forest	47%

Table 1: 10-fold cross-validation accuracy of different classifiers

4 Retraining-based Active Learning Procedure

In this section we present the Retraining-based Active Learning Procedure that we employ. Initially we studied the approaches that are proposed in [2], [3] and [4]. After that, we decided to employ the method (Algorithm 1) that is proposed in [2] and aims to make use of the uncertainty information in order to enhance the performance of the retraining-based model. The pseudocode of this Retraining-based Active Learning Procedure can be depicted in Figure 2. More specifically, in our case this procedure checks all the points in unlabeled pool (unlabeled test set of 20000 samples) over all the possible labels and we update the labeled training set of 200 samples and we retrain the classifier (Logistic Regression) we use. Based on this new trained classifier, we measure some kind of selection criteria (i.e. average-case [5], worst-case [6] and best-case criterion [7]. Finally, we query the instance that leads to the best value in terms of the criterion we are interested in. Following the procedure in [2] and taking into consideration that it is possible for [5],

[6] and [7] to may fail to take into account some potentially valuable information, we incorporate the uncertainty information within the following min-max framework for retraining-based models: $min_{x_i \epsilon U} max_{y_i \epsilon C} P_L(y_i|x_i)V(x_i, y_i)$, where $x_i \in U$ corresponds to every instance of the unlabeled test set of 20000 samples, $y_i \in C$ correspond to every possible label (1 or 2) that each x_i can have, $P_L(y_i|x_i)$ contains the pre-trained label information and for the $V(x_i, y_i)$ we employ the value of the objective function. Thus. after this fusion of the uncertainty sampling with the retraining-based model that is achieved through this method we are able to identify the most informative subset of the unlabeled test set of the 20000 samples and for which we are going to request its real labels through queries in an "oracle".

Algorithm 1 General Retraining-based Active Learning Procedure

Input: Labeled data $\mathcal{L}$, unlabeled data $\mathcal{U}$
repeat
  Train the classifier on $\mathcal{L}$ and calculate $P_{\mathcal{L}}(y_i|x_i)$ for each $x_i \in \mathcal{U}$, each $y_i \in C$;
  for each $x_i \in \mathcal{U}$ **do**
    for each $y_i \in C$ **do**
      Re-train the model on $\mathcal{L} \cup \{x_i, y_i\}$;
      Calculate some criterion $V(x_i, y_i)$, (*e.g.*, error or variance);
    end for
  end for
  Compute some kind of performance based on $P_{\mathcal{L}}(y_i|x_i)$ and $V(x_i, y_i)$;
  Query the instance x^* which leads to the best performance and label it y^*, update $\mathcal{L} \leftarrow \mathcal{L} \cup \{x^*, y^*\}, \mathcal{U} \leftarrow \mathcal{U} \backslash \{x^*\}$;
until Stopping criterion is satisfied

Figure 2: Pseudocode of the Retraining-based Active Learning Procedure

However, we should mention here that our approach has a significant shortcoming. Namely, we neither have access to the real labels of the unlabeled test set of 20000 samples nor we have in our disposal an "oracle" to provide us with real labels of the subset of the unlabeled test samples that we are interested in. In order to be able to overcome this obstacle we employ the following method: Initially we split the labeled training set (200 samples) into a 80%(160 samples)-20%(40 samples) training and validation set and we validate the Logistic Regression classifier on this validation set. After that, for each unlabeled sample of the unlabeled test (20000 samples) and more specifically (for a specific number of samples that belong to the aforementioned most informative subset) we use a majority-voting committee rule. More particularly, we use the 5 classifiers (Logistic Regression, Fisher, Multi-layer Perceptron (MLP), Bayes and 5-Nearest Neighbor) that perform better in our initial 10-fold cross-validation accuracy to predict its label and we assign as a true label to this sample the label that is based on a majority vote (i.e. at least 3 classifiers agree on this label). In the sequel we append this sample in the training set and we re-evaluate our classifier (Logistic Regression) on the validation set. In a similar manner we incrementally append each sample of the unlabeled test set to our training set with the same procedure and we re-evaluate our classifier (Logistic Regression) on the validation set. Our stopping criterion is a 2-fold criterion. Firstly, we keep track of the accuracy of the classifier in the validation set for every extra unlabeled sample that we use in order to train the classifier. Secondly, due to the high computational cost of this procedure, we cannot do it for a very large number of unlabeled samples and thats why we found in the previous step the most informative subset of the unlabeled samples that enhance the performance of our classifier. Moreover, in a real situation that we obtain the real labels of the unlabeled samples through requesting them from an "oracle" or "expert", it is very possible to have a limited budget for only a specific number of queries. An experimental justification of our decision is given in the next section.

5 Experiments

In this Section we present the experiments that we made in order to build our final best performing classifier. We keep track of the accuracy of the classifier in the validation set for every extra unlabeled sample that we use in order to train the classifier and we also take into account the high computational cost. As far as the tracking of the accuracy is concerned, we made the following plots of the accuracy of the classifier in the validation set for every extra unlabeled sample that we use in order to train the classifier versus the number of the unlabeled points and the results can be depicted in Figures 3-6.

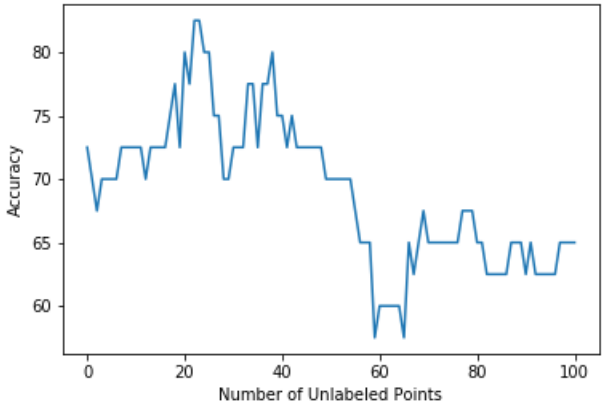

Figure 3: Accuracy vs unlabeled points(100)

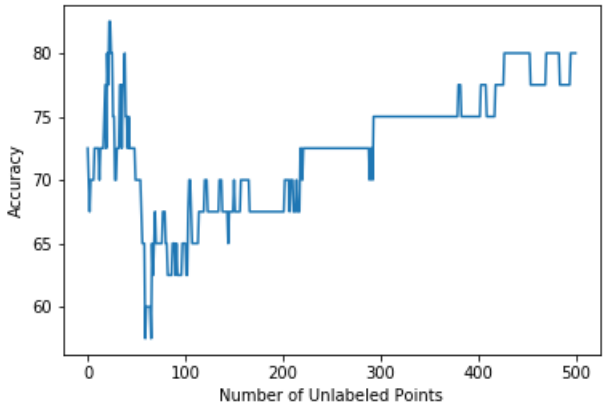

Figure 4: Accuracy vs unlabeled points(500)

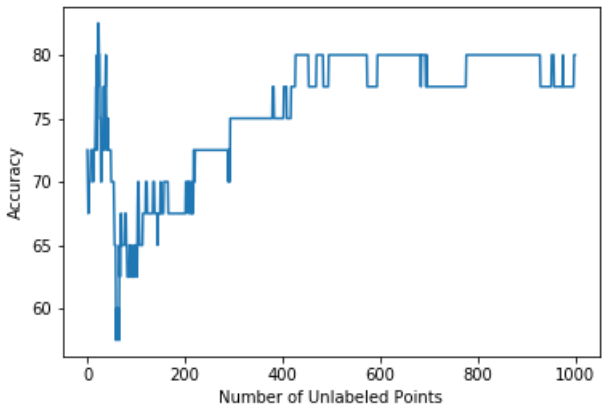

Figure 5: Accuracy vs unlabeled points(1000)

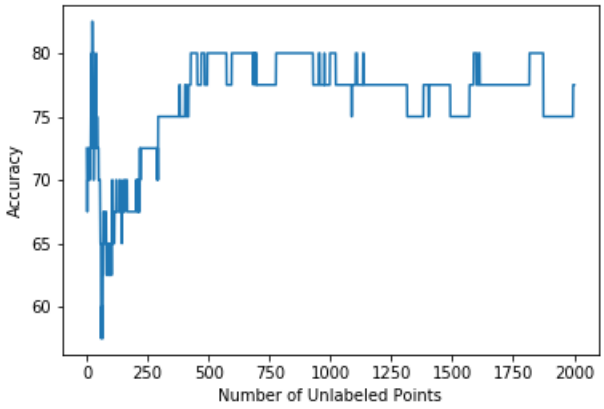

Figure 6: Accuracy vs unlabeled points(2000)

It can be depicted from Figure 3 that the performance of the classifier is quite unstable due to the fact that we only use extra 100 samples from the unlabeled test set to retrain our classifier. Thus, it is one unreliable solution (we also make a 10-fold cross-validation accuracy in the sequel to justify our reasoning).
In Figure 4, where we use the 500 most informative unlabeled samples, we can see that our classifier already achieves a quite stable and very good performance (approximately 80%). It is worth mentioning here that our classifier starts with an accuracy of 72.5% in the previous 80%-20% split.
In Figure 5, where we use the 1000 most informative unlabeled samples, we can see that the accuracy of our classifier starts to plateau after the 500 samples and has some minor changes.
Finally, in Figure 6 where we use the 2000 most informative unlabeled samples, we can see that the accuracy of our classifier starts to drop after the 1000 samples and this consists an indication that we should stop adding extra unlabeled samples.
Thus through tracking the accuracy of the classifier in the validation set for every extra unlabeled sample that we use in order to train the classifier, we know that we should stop when its performance plateaus for a sufficient enough number of unlabeled samples. Based on this analysis, we should use this procedure for 500-1000 unlabeled samples. In order to be sure and to be able to build the best classifier, we evaluate our classifier through a 10-fold cross-validation accuracy for these combination of unlabeled samples. The results of this evaluation can be depicted in Table 2. We can observe that the 1000 unlabeled samples is the ideal choice in terms of accuracy, potential overfitting issues (if we choose 2000 unlabeled samples) and computational cost. Finally, it was worth mentioning that we have a tremendous increase in the accuracy of the classifier from 76.5% (200 samples) that we start to 95.5% (1200 samples).

Number of Unlabeled Samples	**10-fold cross-validation accuracy**
0	76.5%
100	79.6%
500	91.9%
1000	95.5%
2000	95.6%

Table 2: 10-fold cross-validation accuracy vs Number of unlabeled points

6 Benchmark

Based on our analysis of the previous sections we decided to use the Logistic classifier with 1200 training samples (initial 200 samples with their real labels and 1000 most informative unlabeled samples that we labeled through our method) in order to estimate the performance of our work on the 20,000 unlabeled data set. To summarize, this classifier achieved a performance with the 10-fold cross-validation that can be depicted in Table 3.

Error	**4.5%**
Accuracy	**95.5%**
Recall	96.4%
Precision	95.8%

Table 3: Evaluation metrics with 10-fold cross validation of the proposed classifier

7 Conclusion

In this work and after trying a lot of approaches, we finally based on an Active Learning approach in which we built a Logistic classifier with that we described in previous sections in order to estimate the performance of our work on the 20,000 unlabeled data set. We were able to achieve a 10-fold cross-validation error of rate **4.5%** on this dataset and we believe that our classifier would be able to achieve a very good error rate on the test set that is provided. It is worth mentioning, that in case that we had in our disposal an "oracle" or expert to provide us with real labels of the subset of the most informative unlabeled test samples that we are interested in, our performance would be even better.

8 Word Count

Total words: 1996

References

[1] *Settles, Burr. Active learning literature survey. University of Wisconsin, Madison, 52(55-66):11, 2010.*

[2] *Yang, Yazhou and Loog, Marco. (2016). Active learning using uncertainty information. 2646-2651. 10.1109/ICPR.2016.7900034.*

[3] *Yang, Yazhou and Loog, Marco. A variance maximization criterion for active learning. Pattern Recognition, 78: 358–370, 2018*

[4] *Y. Yang and M. Loog, "A benchmark and comparison of active learning methods for logistic regression," arXiv preprint, 2016*

[5] *Roy, Nicholas and Mccallum, Andrew. Toward optimal active learning through sampling estimation of error reduction. In In Proc. 18th International Conf. on Machine Learning, pp. 441–448, 2001.*

[6] *Huang, Sheng-Jun, Jin, Rong, and Zhou, Zhi-Hua. Active learning by querying informative and representative examples. IEEE Transactions on Pattern Analysis and Machine Intelligence, 10(36):1936–1949, 2014.*

[7] *Y. Guo and R. Greiner, "Optimistic active-learning using mutual information." in IJCAI, vol. 7, 2007, pp. 823–829.*