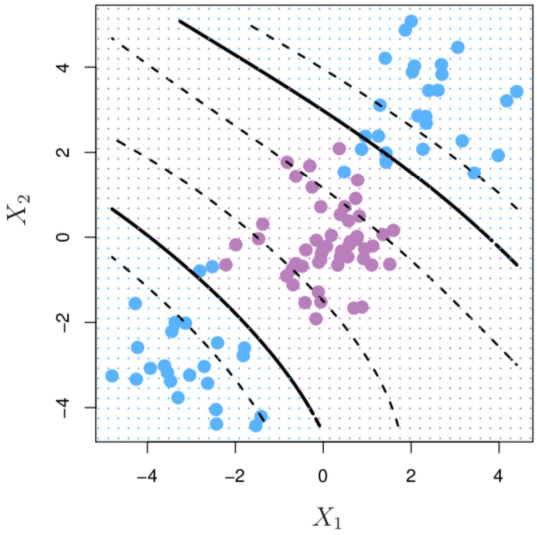
Final Project — Due Wednesday, Dec. 13th

*Submit your report on Canvas and append the codes you used to your submission.

Problem 1: Basics Knowledge (200 pts)

1. What is the difference between *random forest* and *gradient boosting trees*?
2. Does decision tree have good interpretability? How about random forest? Explain.
3. *Multiclass exponential loss.* For a K-class classification problem, consider the coding $Y = (Y_1, \cdots, Y_K)^\top$ with
$$Y_k = \begin{cases} 1, & \text{if } G = \mathcal{G}_k \\ -\frac{1}{K-1}, & \text{otherwise} \end{cases}$$
Let $f = (f_1, \cdots, f_K)^\top$ with $\sum_{k=1}^K f_k = 0$, and define
$$L(Y, f) = \exp\Big(-\frac{Y^\top f}{K}\Big)$$
   (a) Using Lagrange multipliers, derive the population minimizer f^* of $L(Y, f)$, subject to the zero-sum constraint, and relate these to class probabilities.
   (b) Show that a multiclass boosting using the loss function leads to a reweighting algorithm similar to Adaboost.
4. Prove that, in maximal margin classifier, the margins on the two sides of the optimal separating hyperplane must be equal.
5. In the following SVM classifer, which data points are the support vectors?



6. Show that the support vector classifier

$$\min_{\beta,\beta_0} \ \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{n}\xi_i$$
$$\text{subject to } \xi_i \geq 0,\ y_i(x_i^\top \beta + \beta_0) \geq 1 - \xi_i, \forall i$$

   can be equivalently formulated as

$$\min_{\beta_0,\beta} \ \sum_{i=1}^{n}\left(1 - y_i(x_i^\top \beta + \beta_0)\right)_+ + \frac{\lambda}{2}\|\beta\|^2$$

7. What is the difference between K-means and KNN? Explain.

8. PCA and Auto-Encoder both can performance dimension reduction. What are their differences?

9. When the number of layers increases in neural networks, how do the bias and variance behave?

10. Write the pseudocodes of a backward propagation stochastic gradient algorithm for training a multi-layer neural network.

Problem 2: Classification on 20newsgroup Data (100 pts)

Dataset: 20newsgroup.zip
Description: The goal is to classify the types of postings based on their context. The dataset is a tiny version of the 20newsgroups data, with binary occurrence data for 100 key words across 16242 postings. The file "wordlist.txt" lists the 100 key words. The file "documents.txt" is essentially a 16242x100 occurrence matrix where each row is corresponding to 1 posting and each column is corresponding to 1 keyword. The occurrence matrix has binary entries where the (i,j)-th entry is 1 if and only if the i-th posting contains the j-th keyword. Since the occurrence matrix is extremely sparse, the "documents.txt" is a sparse representation of the occurrence matrix. Basically, each line in "documents.txt" represents 1 non-zero entry of the occurrence matrix. For instance, the first line of "documents.txt" is "1 23 1" which means that the entry (1,23) of the occurrence matrix is 1, i.e., the 1st posting contains the 23th keyword. The file "newsgroup.txt" has 16242 lines where i-th line stands for the group labels of i-th posting. There are 4 different groups which means "comp.", "rec.", "sci." and "talk." respectively. The goal is predict the type, i.e. 4 different group, of the posting based on the words in this posting.

1. Build a random forest for this dataset and report the 5-fold cross validation value of the misclassification error. Note that you need to train the model by yourself, i.e., how many predictors are chosen in each tree and how many trees are used. There is no benchmark. Stop tuning when you feel appropriate. Report the best CV error, the corresponding confusion matrix and tuning parameters. What are the ten most important keywords based on variable importance?

2. Build a boosting tree for this dataset and report the 5-fold cross validation value of the misclassification error. Similarly, report the best CV error, the corresponding confusion matrix and tuning parameters. Note that the R example in the textbook only considers binary classification.

3. Compare the results from random forest and boosting trees.

4. Build a multi-class LDA classifier. Report the 5-fold CV error of misclassification.

5. Build a multi-class QDA classifier. Report the 5-fold CV error of misclassification.
6. Train an SVM on the given dataset. Report the 5-fold CV error of misclassification.
7. Compare the performances of all above methods and give your comments.

Problem 3: Spectral Clustering (PCA + K-means)on 20newsgroup Data (100 pts)

Dataset: 20newsgroup.zip

1. Implement the K-means clustering algorithm by yourself. Then use your algorithm for the next several steps.
2. Apply PCA on the binary occurrence matrix and apply K-means clustering. Basically, take the top 4 left singular vectors of the occurrence matrix (of size 16242x100) and apply K-means on the rows of these singular vectors with K=4. Report the mis-clustering error rate and running time.
3. Now take the top 5 left singular vectors of the occurrence matrix and apply K-means on the rows of these singular vectors with K=4. Report the mis-clustering error rate and running time.
4. Compare with the performances from Problem 2.

   (Note that the true cluster labels are already provided in the data. Using the truth and the results of K-means, you can compute the mis-clustering error rate.)
5. Visualize the two-dimensional or three-dimensional projection of the given data. Can you observe revealing cluster structure?

Problem 4: Classification on MNIST Data (100pts)

Dataset: MNIST/train_resized.csv, MNIST/test_resized.csv

Description: train_resized.csv has 30000 rows and 145 columns, test_resized.csv has 12000 rows and 145 columns. Each row is corresponding to 1 handwriting digits. The first column label denotes the actual digit that can be 0,1,. . . ,9. The remaining 144=12*12 column are the pixels of one image, so each image is of size 12x12. Some example images are as follows.

Note that the original image is of size 28x28. I have downsized it to 12x12 to make your computation faster. As a result, the image pixel values are not 0 or 1 anymore.

1. Use only the digit images of 3 and 6 from train_resized.csv and test_resized.csv to build an SVM classifier for binary classification. More specifically, use a linear kernel and choose the best cost (the data size is large so a large cost value is suitable) parameter (called budget in our course) by 5 fold cross validation. Apply your model on the test data and report the misclassification error, confusion matrix. Also report the time cost of training your model.

2. Use only the digit images of 3 and 6 from train_resized.csv and test_resized.csv to build an SVM classifier for binary classification. More specifically, use a radial kernel and choose the best cost parameter, gamma parameter by 5 fold cross validation. Apply your model on the test data and report the misclassification error, confusion matrix. Also report the time cost of training your model.

3. Compare the results of the above two models and report your comments.

4. Use only the digit images of 1,2,5 and 8 from train_resized.csv and test_resized.csv to build an SVM classifier for multi-class classification. More specifically, use a linear kernel and choose the best cost parameter (called budget in our course) by 5 fold cross validation. Apply your model on the test data and report the misclassification error, confusion matrix. Also report the time cost of training your model.

5. Use the complete dataset of train_resized.csv and test_resized.csv to build an SVM classifier for classifying all 10 classes. You can use any SVM model and tune the parameters by yourself. Report the best test performance (misclassification error) you can get, the model you used and the time cost of training your model.

Problem 5: Deep learning on MNIST Data (100pts)

Dataset: MNIST/train_resized.csv, MNIST/test_resized.csv

1. Train a convolution neural network and tune the parameters for the best performance. Report the test error, running time, and model details.

2. Train an Auto-Encoder on the combined dataset (train+test) and visualize their two-dimensional representation. Report the results and model details.