CS464 Machine Learning
Spring 2019
Homework 2

Due: April 28 11:59 pm

Instructions

- Submit a soft copy of your homework of all questions to Moodle in PDF format. Add your code at the end of the your homework file and upload it to the related assignment section on Moodle. Submitting a hard copy or scanned files is NOT allowed. You have to prepare your homework digitally(using Word, Excel, Latex etc.).
- Follow the honor code while doing this homework. This is an individual assignment for each student. That is, you are NOT allowed to share your work with your classmates.
- For this homework, you may code in any programming language you would prefer. In submitting the homework file, please package your file as a gzipped TAR file or a ZIP file with the name `CS464_HW2_Section#_Firstname_Lastname`. If your name is Oğuzhan Karakahya and you are from Section 1 for instance, then you should submit a file with name `CS464_HW2_1_Oguzhan_Karakahya`. Please do not use Turkish letters in your package name. The file you will upload must contain the following parts:
  - the soft copy of your report in PDF format
  - your source code file(s) in a format easy to run, i.e. with a main script to call other functions.
  - the README file that tells us how we can execute/call your program.
- Unless it is stated otherwise in the questions, you are NOT allowed to use ANY machine learning packages, libraries or toolboxes for this assignment (such as scikit-learn, MATLAB Statistics and Machine Learning Toolbox functions, etc.)
- If you do not follow the submission routes, deadlines and specifications (codes, report, etc), it will lead to significant grade deduction.

1 PCA & Eigenfaces [25 pts]

In this question, you are expected to compress face images using PCA and decompress eigenfaces back to restore original images (please mind that unless you use ALL eigenfaces, you will lose some information while reconstructing the original images). For this question, you will use LFW (Labeled Faces in the Wild) dataset [1] which is provided to you within homework zip file as `lfwdataset` folder. To reduce computational time, only first 1000 images are included. Each image file is named using the person's name.

An eigenface is an eigenvector of the normalized data matrix $\bar{X}$ with each row denoting a different image and each column consists of the normalized pixel intensities. To obtain eigenfaces, first you need to create a data matrix X by reading each image that is given to you. Then, you have to subtract mean from X using the row axis to obtain $\bar{X}$. Then, extract eigenvalues of the $\bar{X}^T\bar{X}$ matrix. After sorting eigenvalues

in descending order, starting from the highest one, you have to compute eigenvector corresponding to that eigenvalue. First k eigenvectors you got this way are the first k eigenfaces. You may use built-in PCA functions to obtain eigenfaces.

Question 1.1 [3 pts] What changes do you need to make in order to obtain eigenfaces using SVD instead of PCA? How are these 2 methods related?

Question 1.2 [5 pts] Obtain first k eigenfaces where k $\in \{16, 32, 64, 128, 256, 512, 1024, 2048, 4096\}$ and for each of these k values, find the percentage of the variance explained. Then, plot k vs. explained variance graph and comment on it. Describe what is the significance of explained variance and how you can obtain that value.

Question 1.3 [12 pts] Describe how you can reconstruct an original image using eigenfaces that are obtained using PCA. In a 5x10 grid, plot original photos of first 6 people (6 photos: Aaron Eckhart, Aaron Guiel, Aaron Patterson, Aaron Peirsol, Aaron Pena and Aaron Sorkin) on the first row, first 6 eigenfaces on the second row, reconstructed images of the 6 people specified above on the third row using first 32 eigenfaces, reconstructed images of the 6 people specified above on the fourth row using first 128 eigenfaces and the reconstructed images of the 6 people specified above on the fifth row using first 512 eigenfaces. What happens when you increase the number of eigenfaces?

Question 1.4 [5 pts] If we would like to keep 95% of the original dataset variance, what would be the minimum number of eigenfaces required? For that many eigenfaces, assuming a single pixel is contained within an integer and both integers and floats are 4 bytes, what is the compression ratio (uncompressed size / compressed size) when 95% of the variance is kept and assuming you are just keeping the necessary information required to reconstruct each image?

2 Linear Regression, Logistic Regression & SVMs [75 pts]

Dataset

Your dataset contains the hourly count of rental bikes between years 2011 and 2012 in Capital bike-share system from UCI Machine Learning Repository [2, 3]. Through this system, user is able to easily rent a bike from a particular position and return back at another position. The usage information of the bikes are monitored for 17379 hours with weather and seasonal information.

The data set has the following attributes:

1. mnth : Month (1 to 12)
2. hr : Hour (0 to 23)
3. weekday : Day of the week
4. weathersit : Weather site
    1: Clear, Few clouds, Partly cloudy, Partly cloudy
    2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
5. temp : Normalized temperature in Celsius. The values are divided to 41 (max)
6. atemp: Normalized feeling temperature in Celsius. The values are divided to 50 (max)
7. hum: Normalized humidity. The values are divided to 100 (max)
8. windspeed: Normalized wind speed. The values are divided to 67 (max)

The data has been already split into two subsets: a 14000-entry subset for training and a 3379-entry subset for testing (consider this as your validation set and imagine there is another test set which is not given to you). You will use the following files:

- `question-2-train-features.csv`

- `question-2-train-labels.csv`
- `question-2-test-features.csv`
- `question-2-test-labels.csv`

The files that ends with features.csv contains the features and the files ending with labels.csv contains the ground truth labels. In the feature files each row contains the feature vector for an entry. The j-th term in the row i is the information related to j-th attribute of the i-th entry. The label files include the ground truth label, i.e. total count of the rental bikes, for the corresponding entry. The order of the entries (rows) are the same in the features and the labels files. That is, the i-th row in both files corresponds to the same entry.

Linear Regression [25 pts]

Question 2.1 **[3 pts]** Derive the general closed form solution for multivariate regression model using ordinary least squares loss function given in Eqn. 2.1. Briefly explain each matrix involved in calculation and how they are constructed.

$$J_n = ||y - X\beta||^2 = (y - X\beta)^T(y - X\beta) \tag{2.1}$$

Question 2.2 **[3 pts]** Find the rank of $X^T X$ for the given dataset in `question-2-train-features.csv` using built-in library functions of your language (rank() for MATLAB, numpy.linalg.matrix_rank() for numpy etc.). What does the rank tell you about the solution you have found for Question 2.1.

Question 2.3 **[12 pts]** You are NOT allowed to use any machine learning libraries to train and test your model for this question. Train your model using the entries in `question-2-train-features.csv` and test your model on the entries in `question-2-train-features.csv`. Report your trained model's coefficients (β values). Evaluate mean squared error on training and test set separately and report them.

Question 2.4 **[2 pts]** After training the model, comment on the coefficients you have found. What does it mean if a coefficient has negative sign? What is the relation between the magnitude of a coefficient and the predicted value? Plot the count of rental bikes vs. normalized humidity for the whole dataset (training set + test set) with normalized humidity on x axis and the count of rental bikes on y axis. Comment on the relation between the count of rental bikes and normalized humidity.

Question 2.5 **[5 pts]** You are NOT allowed to use any machine learning libraries to train and test your model for this question. Train your model one more time using **only** normalized humidity as your feature and plot the count of rental bikes vs. normalized humidity for the whole dataset (training set + test set) with normalized humidity on x axis and the count of rental bikes on y axis. Comment on the relation between the count of rental bikes and normalized humidity and compare your results with the what you have found in Question 2.4.

Logistic Regression [25 pts]

For this part of the question you will use the same dataset from Question 2 as described in section Dataset after discretizing the labels. To do so, use the average of all training set and test set labels ($mean(Y) \approx 190$) to decide if an instance belongs to "high usage" or "low usage" class.

$$Y_i = \begin{Bmatrix} \text{"}low\ usage\text{"}, & if & Y_i < mean(Y) \\ \text{"}high\ usage\text{"}, & if & Y_i \geq mean(Y) \end{Bmatrix} \tag{2.2}$$

Question 3.1 [10 points] You will implement full batch gradient ascent algorithm to train your logistic regression model. Initialize all weights to 0. Try different learning rates from the given logarithmic scale $[10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]$ and choose the one which works best for you. Use 1000 iterations to train your model. Report the accuracy and the confusion matrix using your model on the test set given. Calculate and report micro and macro averages of precision, recall, negative predictive value (NPV), false positive rate (FPR), false discovery rate (FDR), F1 and F2 scores.

Question 3.2 [12 points] You are NOT allowed to use any machine learning libraries to train and test your model for this question. You will implement mini-batch gradient ascent algorithm with *batch size = 32* and stochastic gradient ascent algorithm to train your logistic regression model. Initialize all weights to 0. Use the learning rate you have chosen in Question 3.1 and perform 1000 iterations to train your model. Report the accuracies and the confusion matrices using your models on the given test set. Calculate and report micro and macro averages of precision, recall, negative predictive value (NPV), false positive rate (FPR), false discovery rate (FDR), F1 and F2 scores.

Question 3.3 [3 points] In what cases, NPV, FPR, FDR, F1 and F2 would be a more informative metric compared to accuracy, precision and recall alone?

Support Vector Machines (SVMs) [25 pts]

In this question, you will train one soft margin and one hard margin SVM classifiers on the processed (discretized) dataset from Question 3. You must perform 10-fold cross validation WITHOUT using any libraries but you CAN use libraries or software packages to train your SVM.

Question 4.1 [12 points] In this part, you will train a linear SVM model with soft margin without using any kernels. Your model's hyper-parameter is C. Using 10-fold cross validation on your *training set*, find the optimum C value of your model. Look for the best C value with line search in the following range $[10^{-3}, 10^{-2}10^{-1}, 1010^{1}, 10^{2}]$ and calculate accuracy on the left-out fold. For each value of C, calculate mean cross validation accuracy by changing the left-out fold each time and plot it in a nice form. Report your optimum C value. Then, run your model on the *test set* with this C value and report test set accuracy with the confusion matrix. Calculate and report micro and macro averages of precision, recall, negative predictive value (NPV), false positive rate (FPR), false discovery rate (FDR), F1 and F2 scores.

Question 4.2 [13 pts] This time, use radial basis function (RBF) kernel to train your hard margin SVM model on the processed (discretized) dataset from Question 3. RBF kernel is defined as

$$K(x, x') = exp\left(- \frac{||x - x'||^2}{2\sigma^2} \right) \tag{2.3}$$

In RBF kernel formula, $\gamma = -\frac{1}{2\sigma^2}$ is a free parameter that can be fine-tuned. This parameter is the inverse of the radius the influence of samples selected by the model as support vectors. Similar to linear SVM part, train a SVM classifier with RBF kernel using same training and test sets you have used in linear SVM model above. In addition to the penalty parameter C, γ is your new hyper-parameter that needs be optimized. Using 10-fold cross validation and calculating mean cross validation accuracy as described in Question 4.1, find and report the best γ within the interval from the logarithmic scale $[2^{-4}, 2^{-3}, 2^{-2}2^{0}, 2^{1}]$. After tuning γ on your *training set*, run your model on the *test set* and report your accuracy along with the confusion matrix. Calculate and report micro and macro averages of precision, recall, negative predictive value (NPV), false positive rate (FPR), false discovery rate (FDR), F1 and F2 scores.

References

[1] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[2] H. Fanaee-T and J. Gama, "Event labeling combining ensemble detectors and background knowledge," *Progress in Artificial Intelligence*, pp. 1–15, 2013.

[3] D. Dua and C. Graff, "UCI machine learning repository," 2017.