1 Principle Component Analysis (15 points)

For a set of sample vectors $\vec{x}_1 \dots, \vec{x}_n$, PCA finds the first principle component vector $\vec{v}$ by minimizing the following objective function:

$$J(\vec{v}) = \sum_{i=1}^{n} ||\vec{x}_i - (\vec{v}^T\vec{x}_i)\vec{v}||^2$$

Where $\vec{v}$ is a unit vector which means $||\vec{v}|| = 1$.

We know that PCA is maximizing the variance of our data which have been projected onto $\vec{v}$. Assuming sample vectors have zero mean, mathematically this means maximizing the following function:

$$R(\vec{v}) = \sum_{i=1}^{n}(\vec{v}^T\vec{x}_i)^2 = \vec{v}^T XX^T\vec{v}$$

Please show that both the minimizing $J(\vec{v})$ and maximizing $R(\vec{v})$ tasks are equivalent.

2 HMM Algorithm (25 points)

Suppose that we use HMM to model a sequence of binary states $\{X_1, X_2, \cdots X_T\}$ (the possible states of X_t are labeled as S_1 and S_2) and binary observations $\{Y_1, Y_2, \cdots Y_T\}$, (the possible values of observation Y_t are 0 or 1). Assume the parameters of initial, transition, and emission probabilities for HMM $\theta = \{\pi_i, a_{i,j}, e_{i,j}\}$ are:

- Initial probability π_i:

$$\pi_1 = P(X_1 = S_1) = 0.6, \pi_2 = P(X_1 = S_2) = 0.4$$

- Transition probability $a_{i,j}$, (i.e. $P(X_{t+1} = S_j | X_t = S_i)$):

$$P(S_1|S_1) = 0.9, P(S_2|S_1) = 0.1$$
$$P(S_1|S_2) = 0.2, P(S_2|S_2) = 0.8$$

- Emission probability $e_{i,j}$:

$$P(y = 0|x = S_1) = 0.7, P(y = 1|x = S_1) = 0.3$$
$$P(y = 0|x = S_2) = 0.8, P(y = 1|x = S_2) = 0.2$$

(a) Using the forward algorithm, compute the probability that we observe the sequence $Y_1 = 0$, $Y_2 = 1$ and $Y_3 = 0$. Show your work (i.e., show each of your alphas).

(b) Using the backward algorithm, compute the probability that we observe the aforementioned sequence ($Y_1 = 0$, $Y_2 = 1$, and $Y_3 = 0$). Recall that $P(\{Y_t\}_{t=1}^T) = \sum_k \alpha_1^k \beta_1^k$. Again, show your work (i.e., show each of your betas).

(c) Do your results from the forward and backward algorithm agree?

(d) Using the forward-backward algorithm, compute (and report) the most likely setting for each state. Hint: you already have the alphas and betas from the above sub-problems.

(e) Use the Viterbi algorithm to compute (and report) the most likely sequence of states. Show your work (i.e., report a 2×3 table recording the maximal probabilities a path pass state S_0 or S_1 at time 1,2,3).

3 PCA Programming (35 points)

In this programming assignment, you will be implementing the Principal Component Analysis (PCA) algorithm on MATLAB for data (image) representation compression and then use the compressed representation for classification.

For this purpose, the datasets could be loaded from folder `hw5_pca`, which consists of labeled training dataset (`x_train` and `y_train`) and labeled test dataset (`x_test` and `y_test`). The training dataset `x_train` contains 9298 samples (9298 rows), each row being 1 sample. Each sample (each row) is a 16-by-16 grayscale pixel intensity (with possible values between 0 and 255, inclusive), which mostly represent a single digit handwritten number, and can be visualized on MATLAB. For example if you want to visualize the 199-th handwritten digit in the training dataset, you can use the following commands on MATLAB Command Window:

```
load('x_train.mat')
imshow(double(reshape(x_train(199,:), 16, 16)),[]);
```

which will display a handwritten number '3' rotated. The training label `y_train` contains the ground-truth label of this handwritten digit. The test dataset `x_test` and its label `y_test` are similar correspondingly to `x_train` and `y_train`, but with much lesser samples, 2200 in total.

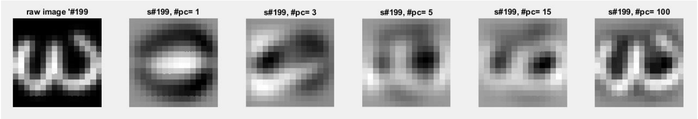
(a) To begin with, you will be implementing eigenvectors computation and sorting based on eigenvalues' magnitude in the provided template file `get_sorted_eigenvecs.m`. Please see the description inside the file for more details.

(b) Each of the computed eigenvectors is a vector $\in \mathbb{R}^{256\times 1}$, thus by itself it can be displayed as an 16×16 image, using the command:

```
imshow(double(reshape(eigenvecs(:,i), 16, 16)),[]);
```

to display the image of `i`-th eigenvector (let us call it an "eigendigit").
Please plot the top 8 eigendigits, corresponding to the top 8 biggest eigenvalues. You may use MATLAB commands `figure`, `hold on`, `subplot`, `imshow`, `hold off` for this purpose. Report this plot on your `*.pdf` file (as well as the hardcopy).

(c) Perform data representation compression on the training data. Since `x_train` is $\mathbb{R}^{9298\times 256}$, if you pick top K eigendigits to project into, you will get the compressed representation `X_compressed`, which is $\mathbb{R}^{9298\times K}$. Now, from this compressed representation `X_compressed`, you can reconstruct the data, producing `X_reconstruction`, which is $\mathbb{R}^{9298\times 256}$ again, but somewhat distorted as compared to the original image. The degree of distortion depends on how many eigendigits K that you used to represent the data in compression. The less K you used, the more severe the distortion is. Here is an example image of the reconstruction.



In the picture above, the first image from left is the raw image drawn from `x_train` for sample 199. The second to the sixth from left to right are the reconstructed digits, when using $K =$ 1, 3, 5, 15, 100 eigendigits (pc stands for principal components, or eigendigits), respectively.

For this part, you need to report similar images like the above, but for sample # 250, 300, 450, 500, and 3000, respectively, drawn from `x_train`. Again, you may use MATLAB commands `figure`, `hold on`, `subplot`, `imshow`, `hold off` for this purpose.
Report this plot on your `*.pdf` file (as well as the hardcopy).

(d) In this final part of programming assignment for PCA, we will do classification on the compressed data. Therefore you need to compress both `x_train` and `x_test` in the same manner (using the same number of eigendigits representer). **Do NOT forget to subtract each sample in `x_test` with the mean of `x_train`**, before projecting them to get the compressed representation. For classification, we will just use the simplest and readily-available-on-MATLAB Decision Tree algorithm. You may use either `ClassificationTree.fit` in MATLAB R2013b or `fitctree` in MATLAB R2015b. Use the following command to perform classification:

```
tree = ClassificationTree.fit(train_data,train_label,'SplitCriterion', 'deviance');
train_label_inferred = predict(tree,train_data);
test_label_inferred = predict(tree,test_data);
```

Please report the accuracy of the training prediction and test prediction, as well the amount of time (in seconds) required to finish the computation in your computer, **for 5 different choices of K (number of top eigendigits used in the representation): 1, 3, 5, 15, 100**. Also, provide the analysis on this results, why the accuracy and computation time differs between different choices of K.
Report both the results and the analysis on your `*.pdf` file (as well as the hardcopy).

Submission Instruction: You need to provide the followings:

- Provide your answers in PDF file, named as `CSCI567_hw5_spring16_yourUSCID.pdf`. You need to submit the homework in both hard copy (at CS567 Homework lockers by 4 pm of the deadline date) and electronic version as pdf file on Blackboard. If you choose handwriting instead of typing all the answers, 40% points will be deducted.
- Submit ALL the code and report via Blackboard. The only acceptable language is MATLAB. For your program, you MUST include the main function called `CSCI567_hw5_spring16.m` in the root of your folder. After running this main file, your program should be able to generate all of the results needed for the programming assignment, either as plots or console outputs. You can have multiple files (i.e your subfunctions), however, the only requirement is that once we unzip your folder and execute your main file, your program should execute correctly. Please double-check your program before submitting. You should only submit one `.zip` file. No other formats are allowed except `.zip` file. Also, please name it as `[lastname]_[firstname]_hw5_spring16.zip`.

Collaboration: You may collaborate. However, collaboration has to be limited to discussion only and you need to write your own solution and submit separately. You also need to list with whom you have discussed.