

10-601: Homework 8 Worksheet  
Due: 24 November 2014 11:59pm (Autolab)  
TAs: Kuo Liu, Yipei Wang  
Name: Dawei Wang  
Andrew ID: daweiwan@andrew.cmu.edu

Please answer to the point, and do not spend time/space giving irrelevant details. Please state any additional assumptions you make while answering the questions. For Questions 1 to 5, 6(b) and 6(c), you need to submit your answers in a single PDF file on autolab, either a scanned handwritten version or a  $\text{\LaTeX}$ pdf file. Please make sure you write legibly for grading. For Question 6(a), submit your m-files on autolab.

You can work in groups. However, no written notes can be shared, or taken during group discussions. You may ask clarifying questions on Piazza. However, under no circumstances should you reveal any part of the answer publicly on Piazza or any other public website. The intention of this policy is to facilitate learning, not circumvent it. Any incidents of plagiarism will be handled in accordance with [CMU's Policy on Academic Integrity](#).

---

**★: Code of Conduct Declaration**

---

- Did you receive any help whatsoever from anyone in solving this assignment? No.
- Did you give any help whatsoever to anyone in solving this assignment? No.

---

**1: PCA I (TA:- Kuo Liu)**

---

(a) Remember that the key assumption of a naive Bayes (NB) classifier is that features are independent, which is not always desirable. Suppose that linear principle component analysis (PCA) is used to transform the features, and NB is then used to classify data in this low-dimension space. Is it true for the following statement? Give reasons to explain why it is true or false. The independent assumption of NB would now be valid with PCA transformed features because all principle components are orthogonal hence independent. [7 points]

FALSE. Principle components being orthogonal means that they're uncorrelated; but the features in the dimension-reduced space could still be dependent.

(b) We usually treat SVD and PCA to be synonymous, can SVD and PCA really produce the same projection result? If yes, under what condition they are the same? If no, please explain why. [7 points]

YES. SVD can be employed to complete PCA. Decomposing  $\mathbf{X}$  using SVD gives  $\mathbf{X} = \mathbf{U}\mathbf{V}\mathbf{T}^T$ , and the covariance matrix  $\mathbf{C} = n^{-1}\mathbf{X}\mathbf{X}^{-1} = n^{-1}\mathbf{U}\mathbf{T}^2\mathbf{U}^T$ , where  $\mathbf{U}$  is a  $n \times m$  matrix. If  $n < m$ , the first  $n$  columns in  $\mathbf{U}$  correspond to the sorted eigenvalues of  $\mathbf{C}$ , and if  $m \geq n$ , the first  $m$  columns correspond to the non-zero eigenvalues of  $\mathbf{C}$ . The projection can thus be written as:

$$\mathbf{Y} = \mathbf{U}^T\mathbf{X} = \mathbf{U}^T\mathbf{U}\mathbf{T}\mathbf{V}^T \quad (1)$$

REFERENCE: *Singular Value Decomposition and Principal Component Analysis*, Rasmus Elsberg Madsen, Lars Kai Hansen and Ole Winther, February 2004.

---

**2: PCA II (TA:- Kuo Liu)**

---

Given 3 data points in 2-d space, (1,1), (2,2) and (3,3)

(a) What is the first principle component? [5 points]

The first principle component is  $(2^{-0.5}, 2^{-0.5})$ .

(b) If we want to project the original data points into a 1-d space by principle component you choose, what is the variance of the projected data? [5 points]

The project data are  $2^{0.5}, 2 \cdot 2^{0.5}, 3 \cdot 2^{0.5}$ . The variance is  $\frac{4}{3}$ .

(c) For the projected data in (b), now if we represent them in the original 2-d space, what is the reconstruction error? [5 points]

The reconstruction error is zero, since the data points perfectly line up.

**3: Collaborative Filtering (TA:- Kuo Liu)**

The problem of collaborative filtering is to predict how well a user will like an item that he has not rated given a set of historical preference judgments for a community of users.

	$i_1$	$i_2$	$i_3$	$i_4$	$i_5$	$i_6$	$i_7$	$i_8$	$i_9$	$i_{10}$
$u_1$	1	2	3	4	5	1	2	3	4	5
$u_2$	5	4	3	2	1	5	4	3	2	1
$u_3$	1	5	1	5	1					
$u_4$	1			5						
$u_5$	4	2								
$u_n$			1	4						

Each row can be viewed as a user profile (a bag of items).

Each column can be viewed as an item profile (a bag of users).

Each cell is the rating (if provided) by a user on an item.

**In reality(exp. online shopping platform Amazon.com), the matrix would have many rows and columns(e.g., millions or billions), and be very sparse**

The following questions are based on the user-item table given above. Here we apply memory-based approach to predict the user profile of the new user( $u_n$ ).

Memory-based approach: Find the k-nearest neighbors (kNN) in the training set, and make inference about the query from there. The formula is as follows:

$$\hat{u}_n = \sum_{u_i \in kNN(u_n)} w(u_n, u_i) u_i$$

(a) Suppose we treat empty cells as zero, use  $\cos(u_n, u_i)$  to represent  $w(u_n, u_i)$  and set k as 2 in kNN(don't consider the user himself as one of the nearest neighbors), what is the user profile for  $u_n$  predicted by this method? [8 points]

The cosine similarities are  $\cos(u_n, u_1) = 0.43937$ ,  $\cos(u_n, u_2) = 0.25437$ ,  $\cos(u_n, u_3) = 0.69961$ ,  $\cos(u_n, u_4) = 0.95130$ ,  $\cos(u_n, u_5) = 0$ . Therefore the nearest two neighbors are  $u_3$  and  $u_4$ , and the user profile for  $u_n$  would be  $\cos(u_n, u_3)u_3 + \cos(u_n, u_4)u_4 = 1.65, 3.50, 0.70, 8.25, 0.70, 0, 0, 0, 0, 0$ .

(b) So far, we have treated the empty cells as zero's in similarity comparison (when computing cosine similarity). Is this a problem? And can you think out of a way to deal with this problem? [8 points]

This is a problem if most of the vectors are sparse. To deal with this problem, neglect the dimensions where no data are available.