HOMEWORK 9: LEARNING PARADIGMS

10-301/10-601 Introduction to Machine Learning (Fall 2021)
http://www.cs.cmu.edu/~mgormley/courses/10601/
OUT: Sunday, November 21st
DUE: Wednesday, December 1st
TAs: Sana, Abhi, Sami, Helena, Chi

This is the final homework assignment. This assignment revisits **Bayes Nets** (from HW7) and **Reinforcement Learning** (from HW8). The new topics covered are **Ensemble Methods**, **K-Means**, **PCA**, and **Recommender Systems**.

START HERE: Instructions

- **Collaboration Policy**: Please read the collaboration policy here: http://www.cs.cmu.edu/~mgormley/courses/10601/syllabus.html
- **Late Submission Policy:** See the late submission policy here: http://www.cs.cmu.edu/~mgormley/courses/10601/syllabus.html
- **Submitting your work:** You will use Gradescope to submit answers to all questions and code. Please follow instructions at the end of this PDF to correctly submit all your code to Gradescope.
    - **Written:** For written problems such as short answer, multiple choice, derivations, proofs, or plots, please use the provided template. Submissions can be handwritten onto the template, but should be labeled and clearly legible. If your writing is not legible, you will not be awarded marks. If your scanned submission misaligns the template, there will be a 5% penalty. Alternatively, submissions can be written in LaTeX. Each derivation/proof should be completed in the boxes provided. If you do not follow the template, your assignment may not be graded correctly by our AI assisted grader.

Instructions for Specific Problem Types

For “Select One” questions, please fill in the appropriate bubble completely:

Select One: Who taught this course?

● Matt Gormley / Henry Chai

○ Marie Curie

○ Noam Chomsky

If you need to change your answer, you may cross out the previous answer and bubble in the new answer:

Select One: Who taught this course?

● Matt Gormley / Henry Chai

○ Marie Curie
~~●~~ Noam Chomsky

For “Select all that apply” questions, please fill in all appropriate squares completely:

Select all that apply: Which are scientists?

□ Stephen Hawking

■ **Albert Einstein**

□ Isaac Newton

□ None of the above

Again, if you need to change your answer, you may cross out the previous answer(s) and bubble in the new answer(s):

Select all that apply: Which are scientists?

■ Stephen Hawking

■ Albert Einstein

■ Isaac Newton
~~■~~ I don’t know

For questions where you must fill in a blank, please make sure your final answer is fully included in the given space. You may cross out answers or parts of answers, but the final answer must still be within the given space.

Fill in the blank: What is the course number?

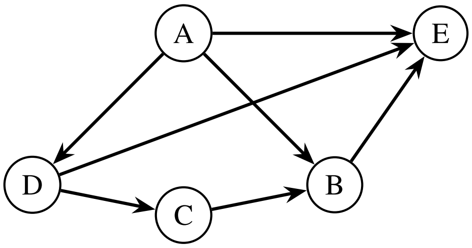
10-601

10-~~7~~601

Written Questions (75 points)

1 Bayes Nets, Revisited (4 points)

Consider the joint distribution over the binary random variables A, B, C, D, E represented by the Bayesian Network shown in the figure.



1. (1 point) **[SOLO]** Write the joint probability distribution for $P(A, B, C, D, E)$ factorized as much as possible using the conditional independence assumptions expressed by the Bayesian Network.

Your Answer

$P(A, B, C, D, E) = P(E \middle| A, B, D)P(B \middle| A, C)P(C \middle| D)P(D \middle| A)P(A)$

2. (1 point) **[SOLO]** Which nodes are in the Markov boundary of B? Note that the Markov boundary is the smallest possible Markov blanket.

Select all that apply:

☒ A

☒ C

☒ D

☒ E

□ None of the above

3. (1 point) **[SOLO]** Which nodes are in the Markov boundary of C?

Select all that apply:

☒ A

☒ B

☒ D

□ E

□ None of the above

4. (1 point) **[SOLO] True or False:** E is conditionally independent of C given $\{A, B, D\}$. That is, $E \perp C \mid \{A, B, D\}$.

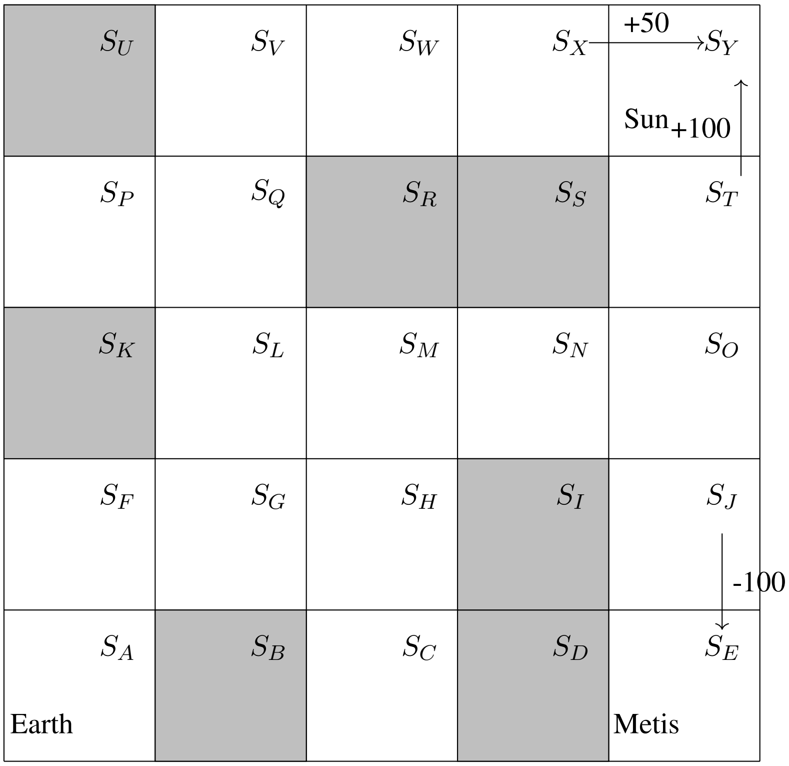
   Select one:

   ● True

   ○ False

2 Reinforcement Learning, Revisited (9 points)

While attending the ML conference *The Fellowship of the Ring*, you meet Elon Musk, founder of SpaceX. He has a new idea for destroying the evil lord Sauron's precious ring: fly the ring directly into the Sun. Elon has asked you to develop a reinforcement learning agent capable of carrying out the space-flight from Earth to the Sun. You model this problem as a Markov decision process (MDP). The figure below depicts the state space.



Here are the details:

1. Each grid cell is a state $S_A, S_B, ..., S_Y$ corresponding to a position in the solar system.
2. The action space includes movement up/down/left/right. Transitions are deterministic. It is not possible to move into blocked states, which are shaded grey, since they contain other planets.
3. The start state is S_A (Earth). The terminal states include both the S_Y (Sun) and S_E (asteroid Metis, home to Sauron's cousin).
4. Non-zero rewards are depicted with arrows. Flying into the Sun from the left gives positive reward $R(S_X, \text{right}) = +50$. Flying into the Sun from below gives positive reward $R(S_T, \text{up}) = +100$. Flying to Metis is inadvisable and gives negative reward $R(S_J, \text{down}) = -100$. All other rewards are zero.
5. The discount factor is $\gamma = 0.5$.

Below, let $V^*(s)$ denote the value function for state s using the optimal policy $\pi^*(s)$. Let $Q^*(s, a)$ denote the Q function for π^*.

1. (1 point) **[SOLO]** What is the value $V^*(S_T)$?

   Your Answer

   100

2. (1 point) **[SOLO]** What is the value $V^*(S_O)$?

   Your Answer

   50

3. (1 point) **[SOLO]** What is the value $V^*(S_A)$?

   Your Answer

   0.78125

4. (1 point) **[SOLO]** What is the value $Q^*(S_T, \text{up})$?

   Your Answer

   100

5. (1 point) **[SOLO]** What is the value $Q^*(S_T, \text{down})$?

   Your Answer

   25

6. (1 point) **[SOLO]** What action does the optimal policy take from state S_Q (i.e. what is $\pi^*(S_Q)$)? (Note: If the optimal policy is not unique and there are multiple optimal actions, select them all.)

   Select all that apply:

   ☒ Up

   □ Down

   □ Left

   □ Right

Now suppose you employ Q-Learning to learn table values $Q(s, a)$ for each state s and action a. The table is initialized to all zeros. On the very first episode of training, you begin at state S_A (Earth), take eight steps, and arrive in state S_E (Metis). At each step, you perform a Q-Learning update of the appropriate entry in $Q(s, a)$. Assume a learning rate $\alpha = 1$.

7. (1 point) **[SOLO]** What is the new table value found in $Q(S_J, \text{down})$ after this episode?

   Your Answer

   -100

8. (1 point) **[SOLO]** What is the new table value found in $Q(S_O, \text{down})$ after this episode?

   Your Answer

   0

9. (1 point) **[SOLO] True or False:** The Q-function is guaranteed to converge to the true Q-values in this environment given the specified initialization and assuming that the Q-Learning algorithm visits each state-action pair infinitely often.

   Select one:

   ● True

   ○ False

3 PCA (15 points)

Some PCA Theory

1. (2 points) **[SOLO]** Assume we apply PCA to a matrix $X \in R^{n \times m}$ and obtain a set of PCA features, $Z \in R^{n \times m}$ .We divide this set into two, $Z1$ and $Z2$. The first set, Z1, corresponds to the top principal components. The second set, Z2, corresponds to the remaining principal components. Which is more common in the training data:

   Select one:

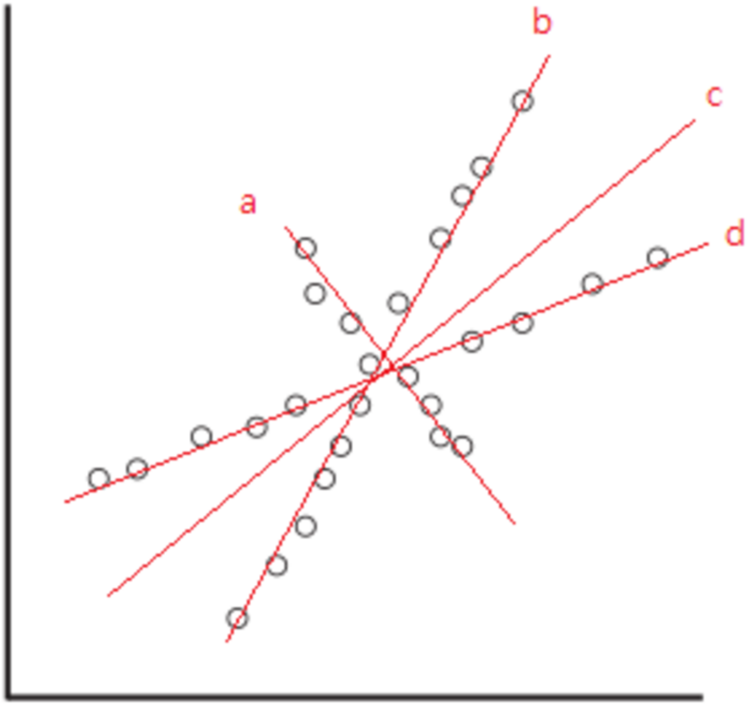
   ● a point with large feature values in $Z1$ and small feature values in $Z2$

   ○ a point with large feature values in $Z2$ and small feature values in $Z1$

   ○ a point with large feature values in $Z2$ and large feature values in $Z1$

   ○ a point with small feature values in $Z2$ and small feature values in $Z1$

2. (1 point) **[SOLO]** For the data set shown below, assume the data are centered at the **origin**. Further, refer to the x-axis as component 1, and the y-axis as component 2. What is its first principal component, if it exists?



Select one:

- ○ a
- ○ b
- ● c
- ○ d
- ○ None of the above

3. (1 point) **[SOLO] NOTE : This is continued from the previous question.** What is the second principal component in the figure from the previous question, if it exists?

   Select one:

   - ● a
   - ○ b
   - ○ c
   - ○ d
   - ○ None of the above

4. (1 point) **[SOLO] NOTE : This is continued from the previous question.** What is the third principal component in the figure from the previous question, if it exists?

   Select one:

   - ○ a
   - ○ b
   - ○ c
   - ○ d
   - ● None of the above

PCA in Practice

For this section, refer to the PCA demo linked here. In this demonstration, we have performed PCA for you on a simple four-feature dataset. The questions below have also been added to the colab notebook linked for ease of access. Run the code in the notebook, then answer the questions based on the results. Once you've answered the following questions, feel free to make a copy of the notebook to further explore how PCA works.

5. (2 points) **[GROUP]** We begin by normalizing each of the columns in our data (ie: adjusting the data to be between 0 and 1, here implemented through min-max scaling. Why is this a good idea? Additionally, why is it potentially a bad idea to standardize our features (ie: adjust all the features to have a variance of 1)?

Your Answer

PCA tries to capture the direction in which the variance is maximized. The higher the variance captured, the better that component is able to represent the data distribution. However, if the features have different scales, their variances will not be a true representation of their importance in capturing the data distribution. Hence, it's preferred to normalize the features. Similarly, you wouldn't want to standardize the features as that would lead to all having a variance of 1. If all features have a variance 1 then, PCA would be incorrect as the data representation would be affected due to standardization.'

6. (2 points) **[GROUP]** Below this question (in the colab file), we plot each of the features against each other. Do you see any special relationships between any of the features? In particular, take a look at the `petal_length` feature. How would you describe its association with each of the other features? Note: there is no need to copy the plots to your submission.

Your Answer

Some of the feature pairs are successfully able to group the data properly. n particular, `petal_length`. `petal_length` with each of the other features is able to successfully capture most of the distribution information. Especially, `petal_length` and `petal_width`. The corresponding plot shows that these 2 features are able to represent the data of 3 classes with minimal overlap based on the plot.

7. (2 points) **[GROUP]** To get the principal components of the features, we will calculate the eigenvalues and eigenvectors of the covariance matrix. If we took the dot product of any two eigenvectors, what should it be? What does it mean to obtain this value and why is this property beneficial?

Your Answer

The dot product of any 2 eigen vectors should be 0. This means that an eigen vector is orthogonal to other eigen vectors. This property ensures that the eigen vectors or principal components are uncorrelated and thus, there's no redundancy in the feature information they capture or represent. The principal components independently represent certain data characteristics.

8. (2 points) **[GROUP]** If we wanted to find k principal components such that we preserve **at least** 95% of the variance in the data, what would be the value of k? Hint: it is helpful here to look at the cumulative variance in the first k components, which we have calculated for you.

Your Answer

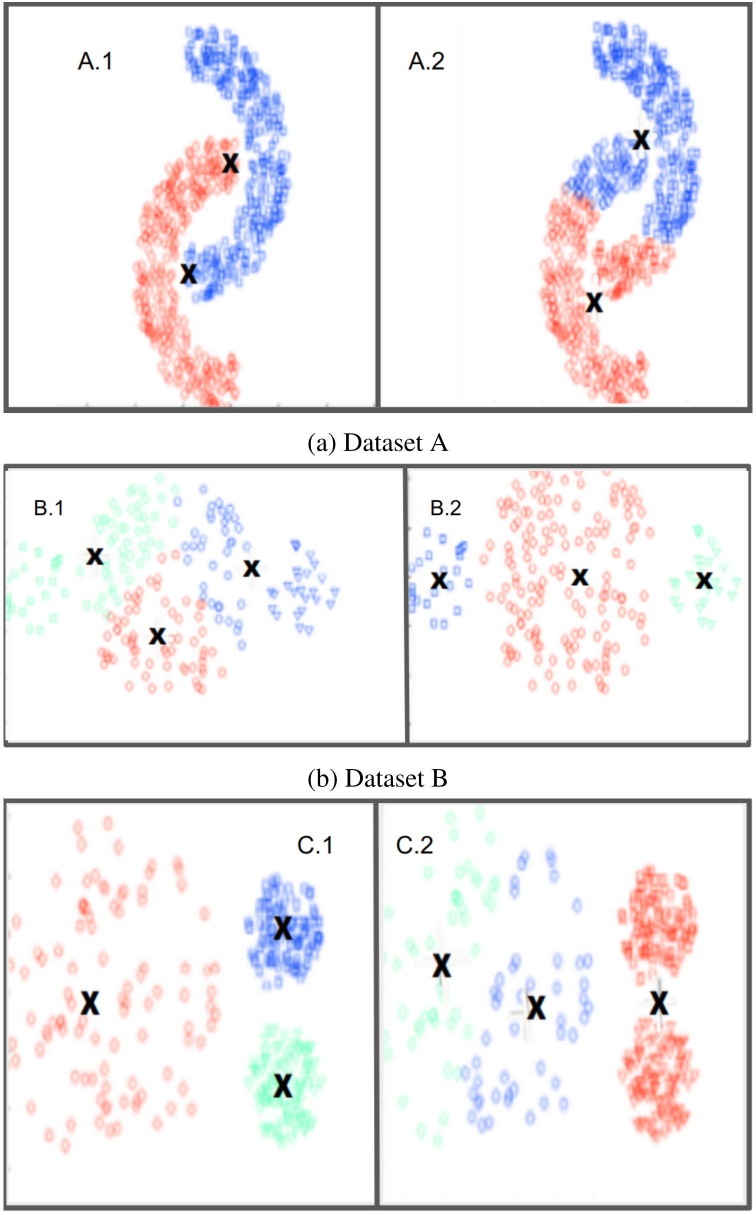
2

9. (2 points) **[GROUP]** Taking note of the principal components plot, let's take a look back at the scatter matrix. If we wanted to perform dimensionality reduction to have just two features, we could pick any two features from the dataset, and train a classifier on just those. What is one reason we could prefer the PCA features to just choosing two of the original features to represent our data?

Your Answer

The distribution as per the principal components has a greater variance and thus, captures the spread of the data much better than that by any 2 of the normalized features. Principal components are formed Thus, it would lead to fewer misclassifications on the test set as compared to that using 2 features.

4 K-Means (20 points)



(a) Dataset A

(b) Dataset B

(c) Dataset C

Figure 1: Datasets

1. Consider the 3 datasets A, B and C as shown in Figure 1. Each dataset is classified into k clusters as represented by different colors in the figure. For each dataset, select the image with cluster centers (denoted by an X) that is generated by K-means. The distance measure used here is the Euclidean distance.

   (a) (1 point) **[SOLO]** Dataset A **(Select one)**

   ○ A.1

   ● A.2

   (b) (1 point) **[SOLO]** Dataset B **(Select one)**

   ● B.1

   ○ B.2

   (c) (1 point) **[SOLO]** Dataset C **(Select one)**

   ○ C.1

   ● C.2

2. Consider a Dataset $\mathcal{D}$ with 5 points as shown below. Perform a K-means clustering on this dataset with $k = 2$ using the Euclidean distance as the distance function. Remember that in the K-means algorithm, an iteration consists of performing following tasks: Assigning each data point to it's nearest cluster center followed by recomputating those centers by taking an average based on all the data points assigned to it. Initially, the 2 cluster centers are chosen randomly as $\mu0$ = (5.3, 3.5), $\mu1$ = (5.1, 4.2). Parts (a) through (d) refer only to the first iteration of K-means clustering performed on $\mathcal{D}$.

$$\mathcal{D} = \begin{bmatrix} 5.5 & 3.1 \\ 5.1 & 4.8 \\ 6.6 & 3.0 \\ 5.5 & 4.6 \\ 6.8 & 3.8 \end{bmatrix}$$

   (a) (2 points) **[SOLO]** Which of the following points will be the center for cluster 0? **Select one:**

   ○ (5.7 , 4.1)

   ○ (5.6 , 4.8)

   ● (6.3 , 3.3)

   ○ (6.7 , 3.4)

(b) (2 points) **[SOLO]** Which of the following points will be the center for cluster 1? **Select one:**

○ (6.1 , 3.8)

○ (5.5 , 4.6)

○ (5.4 , 4.7)

● (5.3 , 4.7)

(c) (1 point) **[SOLO]** How many points will belong to cluster 0?

Answer
3

(d) (1 point) **[SOLO]** How many points will belong to cluster 1?

Answer
2

3. Recall that in K-means clustering we attempt to find k cluster centers $c_j \in \mathbb{R}^d$ (where d is the dimension of the data), $j \in \{1, \dots, k\}$ such that the total distance between each datapoint and the nearest cluster center is minimized. Then the objective function is,

$$\sum_{i=1}^{n} \min_{j \in \{1,\dots,k\}} ||x_i - c_j||^2 \tag{1}$$

In other words, we attempt to find $c_1, \dots, c_k$ that minimizes Eq. (1), where n is the number of data points. To do so, we iterate between assigning x_i to the nearest cluster center and updating each cluster center c_j to the average of all points assigned to the j^{th} cluster. Instead of holding the number of clusters k fixed, your friend John tries to minimize Eq. (1) over k. Yet, you found this idea to be a bad one.

Specifically, you convinced John by providing two values α, the minimum possible value of Eq. (1), and β, the value of k when Eq. (1) is minimized.

(a) (1 point) **[SOLO]** What is the value of α when $n = 100$?

Answer
0

(b) (1 point) **[SOLO]** What is the minimum value of β when $n = 100$?

Answer
100

(c) (2 points) **[SOLO]** We want to see how K-means clustering works on a single dimension. Consider the case in which $k = 3$ and we have 4 data points $x_1 = 1, x_2 = 2, x_3 = 5, x_4 = 7$. What is the optimal value of the objective Eq. (1) when k is fixed at 3?

Answer

0.5

4. (3 points) **[GROUP]** Consider the following simple brute-force algorithm for minimizing the K-means objective:

   1. Enumerate all the possible partitionings of the n points into k clusters.
   2. For each possible partitioning, compute the optimal centers $c_1, ..., c_k$ by taking the mean of the points in each cluster and compute the corresponding K-means objective value.
   3. Output the best clustering found.

   This algorithm is guaranteed to output the optimal set of centers, but unfortunately its running time is exponential in the number of data points. For the case $k = 2$, justify that the running time of the brute-force algorithm above is exponential in the number of data points n.

   Your Answer

   Each of the n points could belong to either of the 2 clusters. Thus, $x^{(1)}$ can take 2 values, $x^{(2)}$ can take 2 values and similarly, $x^{(n)}$ can take 2 values. Thus, the number of combinations/partitionings possible would be 2^n and similary, the corresponding runtime of the algorithm would be $\mathbb{O}(n2^n)$. Thus, as n increases, the runtime increases exponentially.

5. Initializing the centers has a big impact on the performance of Lloyd's clustering algorithm. Usually, we randomly initialize k cluster centers. However, there are other methods, namely, furthest point initialization and k-means++ initialization.

   (a) (2 points) **[GROUP]** Essentially, in k-means++, the first cluster center is chosen uniformly at random from the data points, after which each subsequent cluster center is chosen from the remaining (not chosen) data points. The probability of each point to be chosen as the next center is proportional to the squared distance to closest existing cluster center.

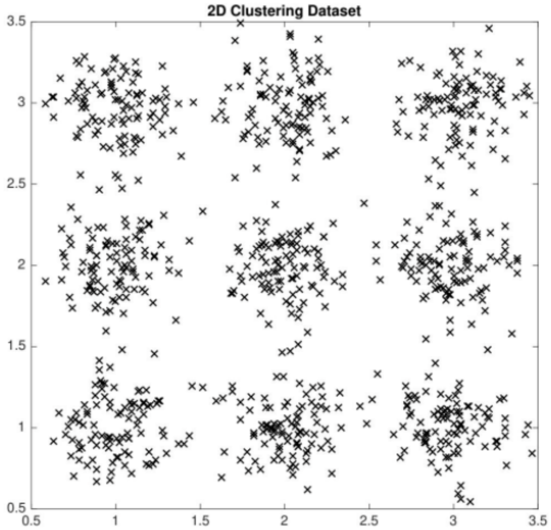


Figure 2: 2D Dataset

Explain in 1– 2 sentences why using k-means++ initialization is more likely to choose one sample from each cluster than random initialization from the dataset in Figure 2 above. Recall that random initialization is randomly choosing a set of data points as starting cluster centers.

Your Answer

Say, the first center (c_1) is chosen from the top-left set of points. However, the subsequent center has a higher probability of being chosen from the set of points furthest from c_1. So, c_2 would be somewhere near the bottom right. Similarly, all subsequent centers would be chosen with high probability from a set of points away from centers leading to a center in every separate cluster. For example, c_3, c_4 have a higher chance of being along the other diagonal.

(b) (2 points) **[GROUP]** Another method of initialization is furthest point initialization. Essentially, the first cluster center is chosen at random. Then, pick the rest of the cluster centers, one by one, among the datapoints that are the furthest from the cluster centers chosen thus far.

Given this definition, is the furthest point initialization sensitive to outliers? Explain why or why not.

Your Answer

Yes, this would be sensitive to outliers. Consider points distributed within 2 circles of radius 1 close to each other and 1 outlier at a distance of 5 from their centers. The 1^{st} center has a high probability of being assigned to one of the points within the circles. However, the 2^{nd} center would be the outlier. Now, k-means would converge to the outlier being one cluster and the two circles being another whereas the ideal clustering would have separated the 2 circles.

5 Ensemble Methods (17 points)

1. (1 point) **[SOLO]** Which of the following is false?

   Select one:

   ○ In the weighted majority algorithm, the weights associated with the weak learners are learned during training.

   In the AdaBoost algorithm, the weights associated with the weak learners are learned during training.

   ● In the weighted majority algorithm, the weak learners are learned during training.

   ○ In the AdaBoost algorithm, the weak learners are learned during training.

2. (1 point) **[SOLO] True or False:** Provided enough iterations are performed, AdaBoost will give zero training error on any dataset regardless of the type of weak learner used.

   Select one:

   ● True

   ○ False

3. (1 point) **[SOLO] True or False:** Consider some training point (x_i, y_i) to the AdaBoost algorithm. If all weak learners during training correctly classify x_i as label y_i, there will eventually be a time t such that the weight assigned to x_i in the training distribution $\mathcal{D}_t$ reaches 0.

   Select one:

   ○ True

   ● False

4. (1 point) **[SOLO] True or False:** If AdaBoost reaches perfect training accuracy, all weak learners created in subsequent iterations will be identical.

   Select one:

   ○ True

   ● False

5. (1 point) **[SOLO]** Consider the following table containing recorded weights from iterations of AdaBoost. Which of the training points (A, B, C, D, E, F) must have been misclassified in Round 220 in order to produce the updated weights shown at the start of Round 221?

Round	$D_t(A)$	$D_t(B)$	$D_t(C)$	$D_t(D)$	$D_t(E)$	$D_t(F)$
...						
220	$\frac{1}{14}$	$\frac{1}{14}$	$\frac{7}{14}$	$\frac{1}{14}$	$\frac{2}{14}$	$\frac{2}{14}$
221	$\frac{5}{40}$	$\frac{5}{40}$	$\frac{14}{40}$	$\frac{2}{40}$	$\frac{10}{40}$	$\frac{4}{40}$
...						

Select all that apply:

☒ A

☒ B

□ C

□ D

☒ E

□ F

6. In the following question, we will examine the generalization error of AdaBoost using a concept known as the *classification margin*.

   Throughout the question, use the following definitions:

   - T: The number of iterations used to train AdaBoost.
   - N: The number of training samples.
   - $S = \{(x_1, y_1), \cdots, (x_N, y_N)\}$: The training samples with binary labels ($\forall i \in [N]\; y_i \in \{-1, +1\}$).
   - d: The VC-dimension of the weak learner hypothesis class.
   - $\mathcal{D}_t(i)$: The weight assigned to training example i at time t. Note that $\sum_i \mathcal{D}_t(i) = 1$.
   - h_t: The weak learner constructed at time t.
   - ϵ_t: The error of h_t on $\mathcal{D}_t$.
   - $Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)}$: The normalization factor for the distribution update at time t.
   - $\alpha_t = \frac{1}{2}\ln((1-\epsilon_t)/\epsilon_t)$: The weight assigned to the learner h_t in the composite hypothesis.
   - $f_t(x) = \left(\sum_{t'=1}^{t} \alpha_{t'} h_{t'}(x)\right) / \left(\sum_{t'=1}^{t} \alpha_{t'}\right)$: The majority vote of the weak learners, rescaled based on the total weights.
   - $H_t(x) = \text{sign}(f_t(x))$: The voting classifier decision function.
   - $\hat{\epsilon}_S(H)$: The training error of classifier H.
   - $\epsilon(H)$: The generalization error of classifier H.

   Consider a binary classification task where our classifier generates a distribution of weights, or confidence scores, over the possible labels. To form a valid distribution, such weights must fall into the range $[0, 1]$ and must sum to 1. The classifier output is the label which has the largest associated weight. We define the *classification margin* for an input as the difference between the weight assigned to the correct label and the incorrect label.

   (a) (1 point) **[GROUP]** What is the range of values the classification margin can take on?

   Your Answer

   $[-1, 1]$

(b) (1 point) **[GROUP]** Let $\text{margin}_t(x, y)$ represent the margin for our AdaBoost classifier at iteration t on the sample (x, y). Write a single inequality in terms of $\text{margin}_t(x, y)$ that is true if and only if the classifier makes a mistake on the input (x, y) (i.e., provide a bound on the margin in the case the classifier is incorrect). Assume the classifier makes a mistake on ties.

Your Answer

(c) (1 point) **[GROUP]** For a given input and label (x_i, y_i), write $\text{margin}_t(x_i, y_i)$ in terms of x_i, y_i and f_t.

Your Answer

(d) (1 point) **[GROUP]** Recall the update AdaBoost performs on the distribution of weights:

- $\mathcal{D}_1(i) = 1/N$
- $\mathcal{D}_{t+1}(i) = \mathcal{D}_t(i)\dfrac{\exp(-y_i\alpha_t h_t(x_i))}{Z_t}$.

Fill in the blank for the following expression for $\mathcal{D}_{t+1}(i)$. You may use x_i, y_i, and both of α, h at any iteration in your answer.

$$\mathcal{D}_{t+1}(i) = \frac{1}{N}\left(\prod_{t'=1}^{t}\frac{1}{Z_{t'}}\right)\exp(\underline{\hspace{3cm}})$$

Your Answer

$-y_i \sum_{t'=1}^{t} \alpha_{t'} h_{t'}(x_i)$

(e) (1 point) **[GROUP]** Let $\alpha = \sum_{t'=1}^{t} \alpha_{t'}$. Rewrite your above answer in terms of y_i, α, f_t, x_i.

Your Answer

$-y_i \alpha f_t(x_i)$

(f) (1 point) **[GROUP]** Rewrite your above answer in terms of $\text{margin}_t(x_i, y_i)$ and α.

Your Answer

(g) (1 point) **[GROUP]** Note that α is constant across the input points. Using the classification margin, describe which points AdaBoost assigns high weight to at time t.

Your Answer

Consider $\hat{\Pr}_{(x_i,y_i)\sim S}\left[\text{margin}_T(x_i, y_i) \leq \theta\right]$, the fraction of inputs with margin below a margin threshold $\theta > 0$ for the combined hypothesis. Suppose we are able to find weak learners below $1/2$ error at all time steps. As a result of the weighting behavior you discovered in the above parts, AdaBoost decreases $\hat{\Pr}_{(x_i,y_i)\sim S}\left[\text{margin}_T(x_i, y_i) \leq \theta\right]$ exponentially fast in the number of iterations T.

Now, consider the following high-probability bounds on the generalization (true) error of AdaBoost. The first bound stems from a traditional PAC learning analysis, while the second stems from a classification margin analysis.

$$\text{Bound 1}: \epsilon\left(H_T\right) \leq \hat{\epsilon}_S\left(H_T\right) + O\left(\sqrt{T \log T}\sqrt{d}\sqrt{\frac{\log N}{N}}\right)$$

$$\text{Bound 2}: \epsilon\left(H_T\right) \leq \hat{\Pr}_{(x_i,y_i)\sim S}\left[\text{margin}_T(x_i, y_i) \leq \theta\right] + O\left(\frac{1}{\theta}\sqrt{d}\sqrt{\frac{\log^2 N}{N}}\right)$$

(h) Suppose we have achieved $\hat{\epsilon}_S(H) = 0$. Considering only the first bound, and supposing we want as tight a bound on true error as possible, should we continue training? Why or why not?

i. (1 point) **[GROUP] Select one:**

○ Continue training

○ Stop training

ii. (1 point) **[GROUP]** Justify your selection:

Justification

(i) Suppose we have achieved $\hat{\epsilon}_S(H) = 0$. Considering only the second bound, and supposing we want as tight a bound on true error as possible, should we continue training? Why or why not?

i. (1 point) **[GROUP] Select one:**

○ Continue training

○ Stop training

ii. (1 point) **[GROUP]** Justify your selection:

Justification

(j) (1 point) **[GROUP] True or False:** Considering both bounds, it is reasonable to stop training AdaBoost as soon as $\hat{\epsilon}_S(H) = 0$.

Select one:

○ True

○ False

6 Recommender Systems (10 points)

1. (2 points) **[SOLO]** In which of the following situations will a collaborative filtering system be more appropriate learning algorithm compared to linear or logistic regression?

   Select all that apply:

   ■ You manage an online bookstore and you have the book ratings from many users. For each user, you want to recommend other books she will enjoy, based on her own ratings and the ratings of other users.

   ■ You run an online news aggregator, and for every user, you know some subset of articles that the user likes and some different subset that the user dislikes. You'd want to use this to find other articles that the user likes.

   □ You've written a piece of software that has downloaded news articles from many news websites. In your system, you also keep track of which articles you personally like vs. dislike, and the system also stores away features of these articles (e.g., word counts, name of author). Using this information, you want to build a system to try to find additional new articles that you personally will like.

   □ You manage an online bookstore and you have the book ratings from many users. You want to learn to predict the expected sales volume (number of books sold) as a function of the average rating of a book.

   □ None of the above

2. (2 points) **[SOLO]** What is the basic intuition behind matrix factorization?

   Select all that apply:

   □ That content filtering and collaborative filtering are just two different factorizations of the same rating matrix.

   □ That factoring user and item matrices can partition the users and items into clusters that can be treated identically, reducing the complexity of making recommendations.

   □ The user-user and item-item correlations are more efficiently computed by factoring matrices.

   ■ That user-item relations can be well described in a low dimensional space that can be computed from the rating matrices.

   □ None of the above

3. To plan your schedule for next semester, you decide to use the recommender system by checking the courses your 10-601 TAs have taken. The course chart below shows everyone's course status. For entry values, 1 indicates that the course has been taken and 0 indicates that the course has not been taken yet.

	15122	15150	15210	15213	10301	10403	10725	11344
You	1	1	0	0	1	0	0	0
Abhi	1	1	1	0	1	1	1	1
Chi	1	0	0	1	1	0	0	0
Helena	1	1	0	0	1	1	0	1
Sana	1	1	0	0	1	1	1	1
Sami	1	1	1	1	1	1	0	0

To use the content-based filtering system, the following similarity table is provided:

	15122	15150	15210	15213	10301	10403	10725	11344
15122	1	0.8	0.7	0.8	0.5	0.4	0.1	0.1
15150	0.8	1	0.9	0.6	0.3	0.1	0.1	0.1
15210	0.7	0.9	1	0.6	0.2	0.1	0.1	0.1
15213	0.8	0.6	0.6	1	0.2	0.1	0.1	0.1
10301	0.5	0.3	0.2	0.2	1	0.8	0.6	0.6
10403	0.4	0.1	0.1	0.1	0.8	1	0.9	0.6
10725	0.1	0.1	0.1	0.1	0.6	0.9	1	0.8
11344	0.1	0.1	0.1	0.1	0.6	0.6	0.8	1

(a) (1 point) **[SOLO]** Using the content-based filtering recommender system, which one of the courses that you (*assuming the first row of table 1*) have **not** taken is the most recommended?

Answer

15210

(b) (1 point) **[SOLO]** Using the collaborative filtering neighborhood method, which one of the courses that you have not taken is the most recommended?

Answer

10403

(c) (1 point) **[GROUP]** Which one of the recommender systems do you prefer? Choose one and justify your choice **in one sentence**.
Select one:

○ Content-based filtering

● Collaborative filtering neighborhood method

Answer

With a large number of users, recommending using collaborative filtering seems like a better choice. The system doesn't need to know any details about the item but just needs to know what other users liked the item. Using that, it can recommend other items.

4. (3 points) **[GROUP]** When building a recommender system using matrix factorization, the regularized objective function we wish to minimize is:

$$J(\mathbf{W}, \mathbf{H}) = \sum_{u,i \in \mathcal{Z}} (v_{ui} - \mathbf{w}_u^T \mathbf{h}_i)^2 + \lambda(\sum_u ||\mathbf{w}_u||^2 + \sum_i ||\mathbf{h}_i||^2)$$

where v_{ui} is the user u's rating of item i, $\mathbf{w}_u$ is the uth row of $\mathbf{W}$ and the vector representing user u; $\mathbf{h}_i$ is the ith row of $\mathbf{H}$ and the vector representing item i; $\mathcal{Z}$ is the index set of observed user/item ratings in the training set; and λ is the weight of the L2 regularizer. One method of solving this optimization problem is to apply Block Coordinate Descent. The algorithms proceeds as shown below:

```
• while not converged:
    – for $u$ in $\{1, \ldots, N_u\}$:
        ∗ $\mathbf{w}_{u'} \leftarrow \text{argmin}_{\mathbf{w}_{u'}} J(\mathbf{W}, \mathbf{H})$
    – for $i$ in $\{1, \ldots N_i\}$
        ∗ $\mathbf{h}_{i'} \leftarrow \text{argmin}_{\mathbf{h}_{i'}} J(\mathbf{W}, \mathbf{H})$
```

Doing so yields an algorithm called Alternating Least Squares (ALS) for matrix factorization. Which of the following is equal to the *transpose* of $\text{argmin}_{\mathbf{w}_{u'}} J(\mathbf{W}, \mathbf{H})$? Note: v_u is the vector of user u's ratings across all items.
Select one:

○ $v_u\mathbf{H}(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I})^{-1}$

○ $(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I})^{-T}v_u\mathbf{H}$

○ $v_u\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}$

6 Collaboration Questions

After you have completed all other components of this assignment, report your answers to these questions regarding the collaboration policy. Details of the policy can be found here.

1. Did you receive any help whatsoever from anyone in solving this assignment? If so, include full details.
2. Did you give any help whatsoever to anyone in solving this assignment? If so, include full details.
3. Did you find or come across code that implements any part of this assignment ? If so, include full details.

Your Answer

1 - No
2 - No
3 - No