Homework 3
Decision Tree, KNN, Perceptron, Linear Regression

10-301/10-601 Introduction to Machine Learning (Spring 2019)

piazza.com/cmu/spring2019/1030110601
OUT: Wednesday, Feb 6th, 2019

DUE: Friday, Feb 15th, 2019, 11:59pm
TAs: Longxiang Zhang, Subhodeep Mitra, Weng Shian Ho, Jiaqi Liu

Submission Template

Please use this template for submission on Gradescope. You can print this file out and manually fill in the answers and rescan your solution into pdf files (make sure to NOT mix up the page order); but we STRONGLY recommend that you modify the provided "HW3_template.tex" file directly. Examples on how to modify the LaTeX code to answer questions are provided in the next section.

IMPORTANT: Please do NOT forget to answer the collaboration questions in the end, it's not graded but it's required!

Examples on Using LaTeX Template to Answer Questions

In this section, We provide you LaTeX code examples on how to change the .tex file directly to answer questions of different types. Remember, you need to read the .tex file to actually see the codes.

Example: how to select choices for multiple-choie questions (circle and squares)

Select One: Who taught this course?

- ● Matt Gormley
- ○ Marie Curie
- ○ Noam Chomsky

Select all that apply: Which are scientists?

- ■ Stephen Hawking
- ■ Albert Einstein
- ■ Isaac Newton
- ☐ I don't know

Example: how to fill in written answers for textbox-style questions.

Fill in the blank: What is the course number?

This is where you input your answer. Long answers will be auto-wrapped, but extremely long answers may cause formatting problem. So keep your language concise.

1 Decision Tree (Revisited) [10 pts]

1. **[3pt]** Suppose you are the 10-601 instructor trying to predict students' letter grade (A+, A, A-, B+, B, B-, C+, C and C-) using only homework grades. You decide to use historic records on this course to build a predictive model. However, someone messed up CMU's academic records system and the only information you have on students from past semesters is (i) if a student has submitted all homework (ii) if a student has attained maximum score on any of the homework (iii) if a student has scored 0 on any of the homework and (iv) the students' letter grades. You decide to use Decision Tree model.

   Will your model be able to do a perfect job (i.e., make no mistakes on students' letter grades of current semester) given enough past records?

   ○ Yes

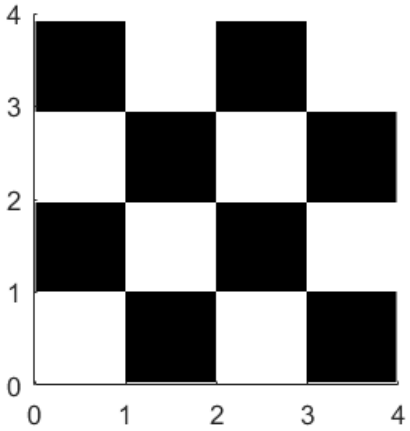
   ● No

   Why or why not? Explain your reason briefly (you can use mathematical expressions).

   NOTE: Please do not change the size of the following text box, and keep your answer in it. Thank you!

   It only use homework grade to predict letter grade which will be not correct. Also, there are three attributes: if a student has submitted all homework; if a student has attained maximum score; if a student has scored 0 in any homework. The results of all of the attribute will be binary. So it can only predict eight results, but the number of letter grade is nine. So it must make mistakes.

2. [**2pt**] Consider the following 4×4 checkerboard pattern.



What is the minimum depth of decision tree that perfectly classifies the 4×4 colored regions, using x and y coordinates as two separate features?

- ○ 1
- ● 2
- ○ 4
- ○ 16

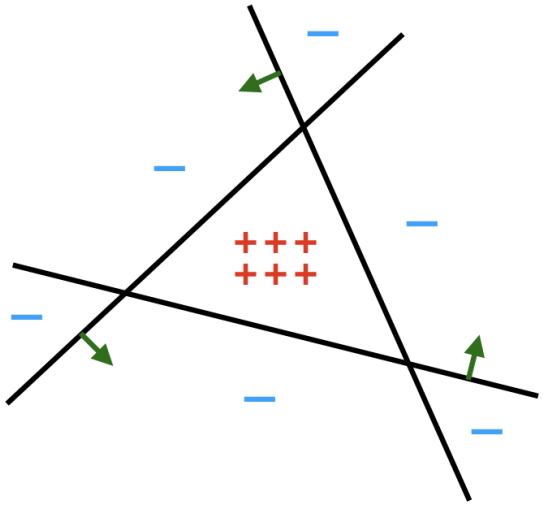
What is the minimum depth of decision trees to perfectly classify the colored regions, using ANY features?

- ● 1
- ○ 2
- ○ 4
- ○ 16

3. **[3pt] Ensemble of Decision Tree.** Say we have a data set shown below. In total, there are 12 data points, with 6 in label "-" and 6 in label "+". We would like to use Decision Tree to solve this binary classification problem. However, in our problem setting, each Decision Tree has access to only ONE line. That is to say, our Decision Tree would have access to only one attribute, and so has max-depth of 1.

   By accessing this line, the Decision Tree could know (and only know) whether the data point is on the right side of this line or the left side. (Unofficial definition: let's assume the right side of a line shares the same direction with the **green** normal vector of that line.)

   Finally, please use majority vote strategy to make classification decision at each leaf.



(a). If we train only one Decision Tree, what is the best/lowest error rate? Note that we have in total 12 data points. (Please round to 4 decimal.)

0.2500

(b). If we could use two Decision Trees, what is the best/lowest error rate? Let's say, if we have two Decision Trees, then each would predict each data point with label like '+' or '-'. Then we would like to combine these predictions as the final result. If these two all predict '+', then the result is '+'. The same with '-'. However, if one predicts '+' while one predicts '-', then to break tie, we always choose '-' as the final result. (Please round to 4 decimal.)

0.0833

(c). Now let's train three Decision Trees as a forest, what is the best/lowest error rate? The ensemble strategy is now the unanimous voting. **That is, if every Decision Tree agree, then the final result is positive. However, if one of them has a different answer from the other two, then our prediction is negative**. (Please round to 4 decimal.)

0.0000

4. **[2pt]** Consider a binary classification problem using 1-nearest neighbors. We have N 1-dimensional training points $x_1, x_2, \ldots x_N$ and corresponding labels $y_1, y_2, \ldots y_N$ with $x_i \in \mathbb{R}$ and $y_i \in \{0, 1\}$. Assume the points $x_1, x_2, \ldots x_N$ are in ascending order by value. If there are ties during the 1-NN algorithm, we break ties by choosing the label of the x_i with lower value. Assume we are using the Euclidean distance metric. Is it possible to build a decision tree in which the decision at each node takes the form of "$x \leq t$ or $x > t$" ($t \in \mathbb{R}$) such that the tree classifies new points in exactly the same way as the 1-nearest neighbor classifier? [1]

   ● Yes

   ○ No

   If your answer is yes, please explain how you will construct the decision tree. If your answer is no, explain why it's not possible.

   NOTE: Please do not change the size of the following text box, and keep your answer in it. Thank you!

We could set t equals to (X1 + X2)/2 at first time and split the y1 and the rest of data. The branch of the node will be x less or equal to (X1 + X2)/2 and x greater than (X1 + X2)/2. And then we could set t equals to (X2 + x3)/2 and split the y2 and the rest of data. We keep doing the same thing N-1 times to build a tree.

[1] The ID3 algorithm taught in class implies that one attribute cannot be reused after splitting, here we relax that restraint. Each node in this decision tree will be reusing attribute x. Another way to recomcile "reusing x" and not reusing attribute is to think of the attributes we have are a collection of N identical copies of x, where N is the number of training samples.

2 K-Nearest Neighbors [27 pts]

1. **[1pt]** Consider the description of two objects below:

	Object A	**Object B**
Feature 1	3	9.1
Feature 2	2.1	0.7
Feature 3	4.8	2.2
Feature 4	5.1	5.1
Feature 5	6.2	1.8

We can reason about these objects as points in high dimensional space.

Consider the two different distance functions below. Under which scheme are they closer in 5-D space?

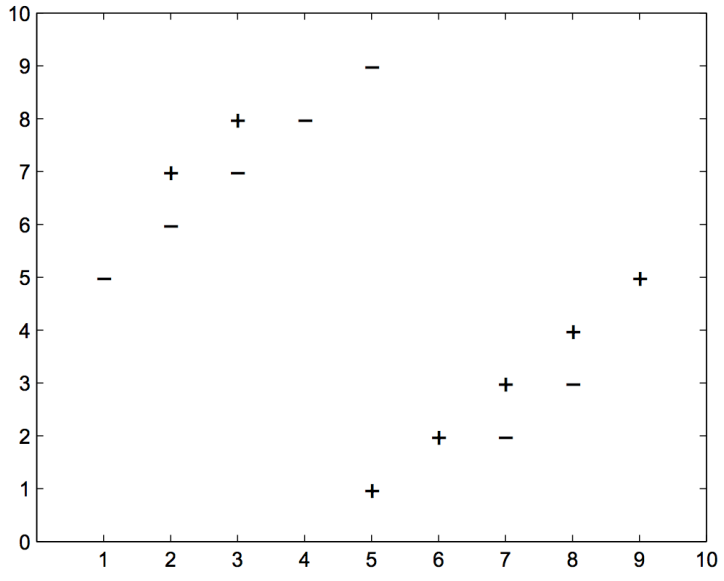
(a) Euclidean Distance: $d(x, y) = \sqrt{\sum_{i-1}^{n}(x_i - y_i)^2}$

(b) Manhattan Distance: $d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$

Select one:

● Euclidean Distance

○ Manhattan Distance



2. **[3pt]** Consider a k-nearest neighbors binary classifier which assigns the class of a test point to be the class of the majority of the k-nearest neighbors, according to a Euclidean distance metric. Using the data set shown above to train the classifier and choosing

$k = 5$, what is the classification error on the training set? Assume that a point can be its own neighbor.

Answer as a decimal with precision 4, e.g. (6.051, 0.1230, 1.234e+7)

0.2857

3. **[3pt]** In the data set shown above, what is the value of k that minimizes the training error? Note that a point can be its own neighbor. Let's assume we use random-picking as the tie-breaking algorithm.

1

4. **[3pt]** Assume we have a training set and a test set drawn from the same distribution, and we would like to classify points in the test set using a k-NN classifier.

(4.1) In order to minimize the classification error on this test set, we should always choose the value of k which minimizes the training set error.

Select one:

○ True

● False

(4.2) Instead of choosing the hyper-parameters by merely minimizing the training set error, some people would like to further split the training dataset into two parts: training and validation datasets, and choose the hyper-parameters that lead to lower error on the validation set. How do you think of this method? Justify your opinion with no more than 3 sentences.

Select one:

○ Good

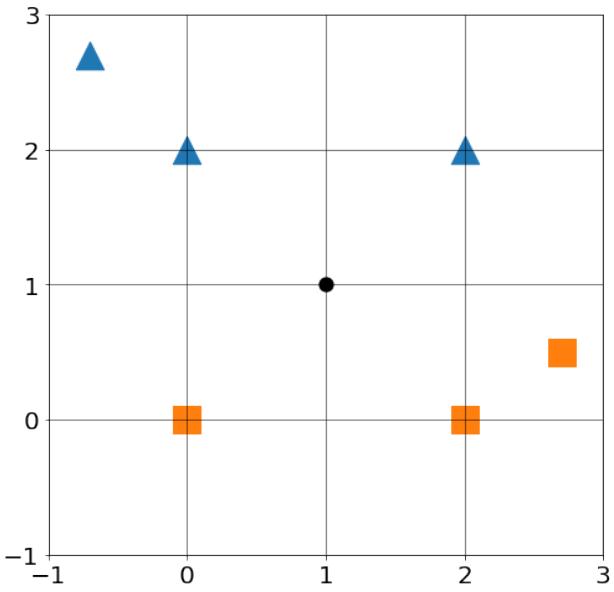
● No good

NOTE: Please do not change the size of the following text box, and keep your answer in it. Thank you!

We should use N-fold cross validation method to split the training data many times and average the error of each validation dataset to find the lowest error in order to choose the best K.

5. **[3pt]** Consider a binary k-NN classifier where $k = 4$ and the two labels are "triangle" and "square".

   Consider classifying a new point $\mathbf{x} = (1, 1)$, where two of the $\mathbf{x}$'s nearest neighbors are labeled "triangle" and two are labeled "square" as shown below.



   Which of the following methods can be used to break ties or avoid ties on this dataset?

   (a) Assign x the label of its nearest neighbor

   (b) Flip a coin to randomly assign a label to $\mathbf{x}$ (from the labels of its 4 closest points)

   (c) Use $k = 3$ instead

   (d) Use $k = 5$ instead

   Select one:

   ○ a only

   ○ b only

   ○ b,c,d

   ● b,d

   ○ d only

   ○ a,b,c,d

   ○ None of the above

6. **[2pt]** Consider the following data concerning the relationship between academic performance and salary after graduation. High school GPA and university GPA are two numerical variables (predictors) and salary is the numerical target. Note that salary is measured in thousands of dollars per year.

Student ID	High School GPA	University GPA	Salary
1	2.2	3.4	45
2	3.9	2.9	55
3	3.7	3.6	91
4	4.0	4.0	142
5	2.8	3.5	88
6	3.5	1.0	2600
7	3.8	4.0	163
8	3.1	2.5	67
9	3.5	3.6	unknown

Among Students 1 to 8, who is the nearest neighbor to Student 9, using Euclidean distance?

Answer the Student ID only.

3

7. **[3pt]** In the data set shown above, our task is to predict the salary Student 9 earns after graduation. We apply k-NN to this regression problem: the prediction for the numerical target (salary in this example) is equal to the average of salaries for the top k nearest neighbors.

If $k = 3$, what is our prediction for Student 9's salary?

Round your answer to the nearest integer. Be sure to use the same unit of measure (thousands of dollars per year) as the table above.

132

8. **[3pt]** Suppose that the first 8 students shown above are only a subset of your full training data set, which consists of 10,000 students. We apply KNN regression using Euclidean distance to this problem and we define training loss on this full data set to be the mean squared error (MSE) of salary.

Now consider the possible consequences of modifying the data in various ways. Which of the following changes **could** have an effect on training loss on the full data set as measured by mean squared error (MSE) of salary? Select all that apply.

Select all that apply:

■ Rescaling only "High School GPA" to be a percentage of 4.0

■ Rescaling only "University GPA" to be a percentage of 4.0

☐ Rescaling both "High School GPA" and "University GPA", so that each is a percentage of 4.0 (scale by the same percentage).

☐ None of the above.

9. **[3pt]** In this question, we would like to compare the differences among KNN, the perceptron algorithm, and linear regression. Please select all that apply in the following options.

   Select all that apply:

   ■ For classification tasks, both KNN and the perceptron algorithm can have linear decision boundaries.

   ☐ For classification tasks, both KNN and the perceptron algorithm always have linear decision boundaries.

   ■ All three models can be susceptible to overfitting.

   ☐ In all three models, after the training is completed, we must store the training data to make predictions on the test data.

   ☐ None of the above.

10. **[3pt]** Please select all that apply about kNN in the following options.

    Select all that apply:

    ■ Large k gives a smoother decision boundary

    ■ To reduce the impact of noise or outliers in our data, we should increase the value k.

    ☐ If we make k too large, we could end up overfitting the data.

    ■ We can use cross-validation to help us select the value of k.

    ☐ We should never select the k that minimizes the error on the validation dataset.

    ☐ None of the above.

3 Perceptron [28 pts]

1. **[2pt]** Consider running the online perceptron algorithm on some sequence of examples S (an example is a data point and its label). Let S' be the same set of examples as S, but presented in a different order.

   True or False: the online perceptron algorithm is guaranteed to make the same number of mistakes on S as it does on S'.

   Select one:

   ○ True

   ● False

2. **[3pt]** Suppose we have a perceptron whose inputs are 2-dimensional vectors and each feature vector component is either 0 or 1, i.e., $x_i \in \{0, 1\}$. The prediction function $y = \text{sign}(w_1 x_1 + w_2 x_2 + b)$, and

$$\text{sign}(z) = \begin{cases} 1, & \text{if } z \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

   Which of the following functions can be implemented with the above perceptron? That is, for which of the following functions does there exist a set of parameters w, b that correctly define the function. Select all that apply.

   Select all that apply:

   ■ AND function, i.e., the function that evaluates to 1 if and only if all inputs are 1, and 0 otherwise.

   ■ OR function, i.e., the function that evaluates to 1 if and only if at least one of the inputs are 1, and 0 otherwise.

   □ XOR function, i.e., the function that evaluates to 1 if and only if the inputs are not all the same. For example

$$\text{XOR}(1, 0) = 1, \text{ but } \text{XOR}(1, 1) = 0.$$

   □ None of the above.

3. **[2pt]** Suppose we have a dataset $\left\{\left(\mathbf{x}^{(1)}, y^{(1)}\right), \ldots, \left(\mathbf{x}^{(N)}, y^{(N)}\right)\right\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^M$, $y^{(i)} \in \{+1, -1\}$. We would like to apply the perceptron algorithm on this dataset. Assume there is no bias term. How many parameter values is the perceptron algorithm learning?

   Select one:

   ○ N

   ○ $N \times M$

   ● M

4. **[3pt]** Which of the following are true about the perceptron algorithm? Select all that apply.

   Select all that apply:

   ☐ The number of mistakes the perceptron algorithm makes is proportional to the size of the dataset.

   ☐ The perceptron algorithm converges on any dataset.

   ■ The perceptron algorithm can be used in the context of online learning.

   ■ For linearly spearable data, the perceptron algorithm always finds the separating hyperplane with the largest margin.

   ☐ None of the above.

5. **[3pt]** Suppose we have the following data:

$$\mathbf{x}^{(1)} = [1, 2] \qquad \mathbf{x}^{(2)} = [-1, 2] \qquad \mathbf{x}^{(3)} = [-2, 3] \qquad \mathbf{x}^{(4)} = [1, -1]$$
$$y^{(1)} = 1 \qquad y^{(2)} = -1 \qquad y^{(3)} = -1 \qquad y^{(4)} = 1$$

   Starting from $\mathbf{w} = [0, 0]$, what is the vector $\mathbf{w}$ after running the perceptron algorithm with exactly one pass over the data? Assume we are running the perceptron algorithm without a bias term. If the value of the dot product of a data point and the weight vector is 0, the algorithm makes the prediction 1.

   Select one:

   ● $[1, -2]$

   ○ $[2, 0]$

   ○ $[-1, 1]$

   ○ $[1, -3]$

6. **[3pt]** Please refer to previous question for the data. Assume we are running perceptron in the batch setting. How many passes will the perceptron algorithm make before converging to a perfect classifier, i.e., one that does not make false prediction on this dataset?

   Select one:

   - ● 2
   - ○ 3
   - ○ 5
   - ○ Infinitely many (the algorithm does not converge)

7. **[3pt]** We can view the perceptron algorithm as trying to minimize which of the following loss functions with stochastic gradient descent? Assume that we apply the notation where $x_0 = 1$. θ_0 is the bias term, and N is the number of data points. You may use the notation

$$(x)_+ = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

   Select one:

   - ○ $J(\boldsymbol{\theta}) = \sum_{i=1}^{N} -y^{(i)} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)}\right)$
   - ○ $J(\boldsymbol{\theta}) = \sum_{i=1}^{N} y^{(i)} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)}\right)$
   - ● $J(\boldsymbol{\theta}) = \sum_{i=1}^{N} \left(-y^{(i)} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)}\right)\right)_+$
   - ○ $J(\boldsymbol{\theta}) = \sum_{i=1}^{N} \left(y^{(i)} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)}\right)\right)_+$

8. **[3pt]** Continuing with the above question, what is the gradient of the correct loss function when the current data we are seeing is $\left(\mathbf{x}^{(i)}, y^{(i)}\right)$?

   Select one:

   - ● $\begin{cases} -y^{(i)}\mathbf{x}^{(i)}, & \text{if } -y^{(i)} \left(\theta \cdot \mathbf{x}^{(i)}\right) \geq 0 \\ 0, & \text{otherwise.} \end{cases}$
   - ○ $-y^{(i)}\mathbf{x}^{(i)}$
   - ○ $y^{(i)}\mathbf{x}^{(i)}$
   - ○ $\begin{cases} y^{(i)}\mathbf{x}^{(i)}, & \text{if } -y^{(i)} \left(\theta \cdot \mathbf{x}^{(i)}\right) \geq 0 \\ 0, & \text{otherwise.} \end{cases}$

9. **[3pt]** Please select the correct statement(s) about the mistake bound of the perceptron algorithm. Select all that apply.

   Select all that apply:

   ☐ If the minimum distance from any data point to the separating hyperplane of the data is increased, the mistake bound will also increase.

   ■ If the maximum distance from any data point to the origin is increased, then the mistake bound will also increase.

   ☐ If the maximum distance from any data point to the mean all data points is increased, then the mistake bound will also increase.

   ☐ The mistake bound is linearly inverse-proportional to the minimum distance of any data point to the separating hyperplane of the data.

   ☐ None of the above.

10. **[3pt]** Suppose we have data whose elements are of the form $[x_1, x_2]$, where $x_1 - x_2 = 0$. We do not know the label for each element. Suppose the perceptron algorithm starts with $\boldsymbol{\theta} = [3, 5]$, which of the following will $\boldsymbol{\theta}$ never take on in the process of running the perceptron algorithm on the data?

    Select one:

    ○ $[-1, 1]$

    ○ $[4, 6]$

    ○ $[-3, -1]$

    ● $[5, 5]$

4 Linear Regression [35 pts]

1. **[3pt]** Suppose you have data $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$ and the solution to linear regression on this data is $y = w_1 x + b_1$. Now suppose we have the dataset $(x^{(1)} + \alpha, y^{(1)} + \beta), \ldots, (x^{(n)} + \alpha, y^{(n)} + \beta)$ where $\alpha > 0, \beta > 0$ and $w_1 \alpha \neq \beta$. The solution to the linear regression on this dataset is $y = w_2 x + b_2$. Please select the correct statement about w_1, w_2, b_1, b_2 below. Note that the statement should hold no matter what values α, β take on within the specified constraints.

   Select one:

   ○ $w_1 = w_2, b_1 = b_2$

   ○ $w_1 \neq w_2, b_1 = b_2$

   ● $w_1 = w_2, b_1 \neq b_2$

   ○ $w_1 \neq w_2, b_1 \neq b_2$

2. **[3pt]** We would like to fit a linear regression estimate to the dataset

$$D = \left\{ \left(\mathbf{x}^{(1)}, y^{(1)}\right), \left(\mathbf{x}^{(2)}, y^{(2)}\right), \cdots, \left(\mathbf{x}^{(N)}, y^{(N)}\right) \right\}$$

   with $\mathbf{x}^{(i)} \in \mathbb{R}^M$ by minimizing the ordinary least square (OLS) objective function:

$$J(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^{N} \left(y^{(i)} - \sum_{j=1}^{M} w_j x_j^{(i)} \right)^2$$

   Specifically, we solve for each coefficient w_k $(1 \leq k \leq M)$ by deriving an expression of w_k from the critical point $\frac{\partial J(\mathbf{w})}{\partial w_k} = 0$. What is the expression for each w_k in terms of the dataset $(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \cdots, (\mathbf{x}^{(N)}, y^{(N)})$ and $w_1, \cdots, w_{k-1}, w_{k+1}, \cdots, w_M$?

   Select one:

   ● $w_k = \frac{\sum_{i=1}^{N} x_k^{(i)} (y^{(i)} - \sum_{j=1, j \neq k}^{M} w_j x_j^{(i)})}{\sum_{i=1}^{N} (x_k^{(i)})^2}$

   ○ $w_k = \frac{\sum_{i=1}^{N} x_k^{(i)} (y^{(i)} - \sum_{j=1, j \neq k}^{M} w_j x_j^{(i)})}{\sum_{i=1}^{N} (y^{(i)})^2}$

   ○ $w_k = \sum_{i=1}^{N} x_k^{(i)} (y^{(i)} - \sum_{j=1}^{M} w_j x_j^{(i)})$

   ○ $w_k = \frac{\sum_{i=1}^{N} x_k^{(i)} (y^{(i)} - \sum_{j=1, j \neq k}^{M} w_j x_j^{(i)})}{\sum_{i=1}^{N} (x_k^{(i)} y^{(i)})^2}$

3. **[3pt]** Continuing from the above question, how many coefficients do you need to estimate? When solving for these coefficients, how many equations do you have?

   Select one:

   ○ N coefficients, M equations

   ○ M coefficients, N equations

   ● M coefficients, M equations

   ○ N coefficients, N equations

4. **[3pt]** We are trying to derive the closed form solution for linear regression:

   In the following, each row in $\mathbf{X}$ denotes one data point and Y is a column vector.

   First we take the derivative of the objective function $L = \frac{1}{2}(\mathbf{X}\mathbf{w} - Y)^T(\mathbf{X}\mathbf{w} - Y)$ with respect to $\mathbf{w}$ and set it to zero, arriving at equation $(*)$.

   Then after some algebraic manipulation, we get the solution $\mathbf{w} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T Y$. What should equation $(*)$ be?

   Select one:

   ● $(\mathbf{X}\mathbf{w} - Y)^T\mathbf{X} = 0$

   ○ $(\mathbf{X}\mathbf{w} + Y)^T\mathbf{X} = 0$

   ○ $\mathbf{X}^T\mathbf{X}\mathbf{w} + Y^T\mathbf{X} = 0$

   ○ $Y\mathbf{X}^T\mathbf{X}\mathbf{w} + \mathbf{X} = 0$

5. **[1pt]** Suppose we are working with datasets where the number of features is 3. The optimal solution for linear regression is always unique regardless of the number of data points that are in this dataset.

   Select one:

   ○ True

   ● False

6. **[1pt]** Assume that a data set has M data points and N variables, where $M > N$. As long as the loss function is convex, the regression problem will return the same set of solutions.

   Select one:

   ● True

   ○ False

7. **[1pt]** Consider the following dataset:

x	1.0	2.0	3.0	4.0	5.0
z	2.0	4.0	6.0	8.0	10.0
y	4.0	7.0	8.0	11.0	17.0

We want to carry out a multiple-linear regression between y (Dependent Variable) and x, z (Independent Variables). The closed-form solution given by $\mathbf{w} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T Y$ will return the unique solution.

Note: The i^{th} row of $\mathbf{X}$ contains the i^{th} data point (x_i, z_i) while the i^{th} row of $\mathbf{Y}$ contains the i^{th} data point y_i.

○ True

● False

8. **[3pt]** Identifying whether a function is a convex function is useful because a convex function's local minimum has the nice property that it has to be the global minimum. Please select all functions below that are convex functions. Note $dom(f)$ denotes the domain of the function f.

Select all that apply:

☐ $f(x) = x, dom(f) = \mathbb{R}$

☐ $f(x) = x^3 + 2x + 3, dom(f) = \mathbb{R}$

☐ $f(x) = \log x, dom(f) = \mathbb{R}_{++}$ (the set of positive real numbers)

■ $f(x) = |x|, dom(f) = \mathbb{R}$

■ $f(x) = ||\mathbf{x}||_2$, $dom(f) = \mathbb{R}^n$

☐ None of the above.

9. **[3pt]** Typically we can solve linear regression problems in two ways. One is through direct methods, e.g. solving the closed form solution, and the other is through iterative methods, e.g. using stochastic or batch gradient descent methods. Consider a linear regression on data $(\mathbf{X}, \mathbf{y})$. We assume each row in $\mathbf{X}$ denotes one input in the dataset. Please select all options that are correct about the two methods.

Select all that apply:

☐ If the matrix $\mathbf{X}^T\mathbf{X}$ is invertible, the exact solution is always preferred for solving the solution to linear regression as computing matrix inversions and multiplications are fast regardless of the size of the dataset.

☐ Assume N is the number of examples and M is the number of features. The computational complexity of N iterations of batch gradient descent is $\mathcal{O}(MN)$.

■ When the dataset is large, stochastic gradient descent is often the preferred method because it gets us reasonably close to the solution faster than both the direct method and batch gradient descent.

☐ None of the above.

10. **[3pt]** A data scientist is working on a regression problem on a large data set. After trying stochastic gradient descent (gradient is evaluated on a portion of the data set in each step) and batch gradient descent (gradient is evaluated on the entire data set in each step), the scientist obtained the values of the loss function (in the table below) for the two methods with respect to training time. Note that the same learning rate is used in both cases.

Time in hours	Stochastic GD	Batch GD
1	102.34	120.12
2	80.45	92.37
3	65.23	73.64
4	56.77	58.23
5	52.33	49.21
6	50.74	45.98
7	49.88	43.64

Select all the choices consistent with this table.

Select all that apply:

☐ The table shows, in practice, that stochastic gradient descent can compute a more accurate gradient direction than batch gradient descent.

☐ Within the first 3 hours, the table suggests that batch gradient descent makes more progress in finding the optimum of the objective function than stochastic gradient descent.

■ In general, stochastic gradient descent does not necessarily take a descent step in each step. However, stochastic gradient descent takes much less time to evaluate per step. In this table, during the first hour, stochastic gradient descent makes more update steps to the weights while batch gradient descent makes less updates. Hence it is reasonable that using batch gradient descent likely results in a higher value for the loss function (worse performance) than that of stochastic gradient descent at the 1 hour time point.

☐ None of the above.

11. [**3pt**] Consider the following dataset:

x	1.0	2.0	3.0	4.0	5.0
y	3.0	8.0	9.0	12.0	15.0

If we initialize the weight as 2.0 and bias term as 0.0, what is the gradient of the loss function with respect to the weight w, calculated over all the data points, in the first step of the gradient descent update? Note that we do not introduce any regularization in this problem and and our objective function looks like $\frac{1}{N}\sum_{i=1}^{N}(wx_i + b - y_i)^2$, where N is the number of data points, w is the weight, and b is the bias term.

Fill in the blank with the gradient on the weight you computed, rounded to 2 decimal places after the decimal point.

-23.60

12. [**4pt**] Based on the data of the previous question, please compute the direct solution of the weight and the bias for the objective function defined in the previous question, rounded to 2 decimal places after the decimal point.

Weight: 2.80

Bias: 1.00

13. [**2pt**] Using the dataset and model given in question 11, perform two steps of batch gradient descent on the data. Fill in the blank with the value of the weight after two steps of batch gradient descent. Let the learning rate be 0.01. Round to 2 decimal places after the decimal point.

2.42

14. [**2pt**] Using the dataset and model given in question 11, which of the following learning rates leads to the most optimal weight and bias after performing two steps of batch gradient descent? (Hint: The most optimal learned parameters are the parameters that lead to the lowest value of the objective function.)

Select one:

- ○ 1
- ○ 0.1
- ● 0.01
- ○ 0.001

Collaboration Questions Please answer the following:

After you have completed all other components of this assignment, report your answers to the collaboration policy questions detailed in the Academic Integrity Policies found here.

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

Solution

1. No ; 2. No ; 3. No