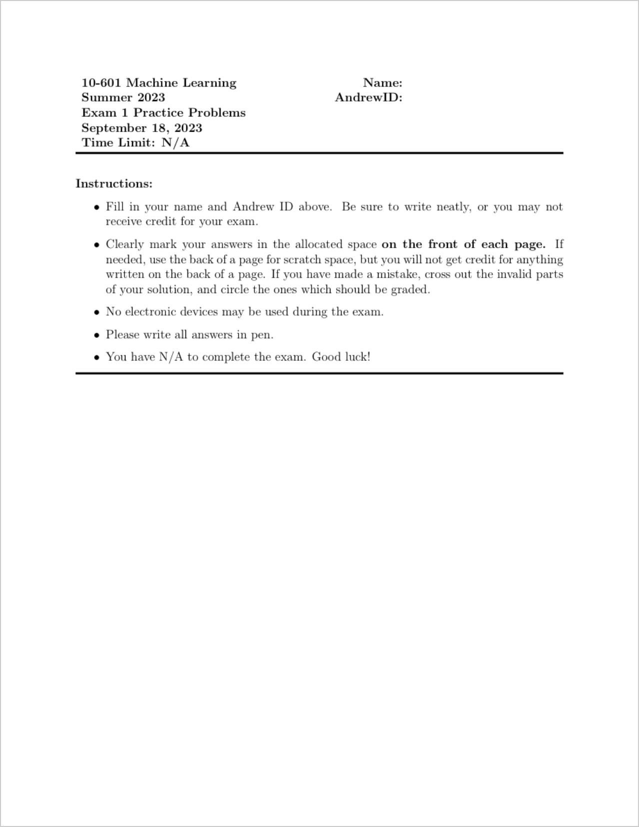
Exam 1 Practice Exam - Section B

Thursday, September 21, 2023 6:23 PM



10-601 Machine Learning
Summer 2023
Exam 1 Practice Problems
September 18, 2023
Time Limit: N/A

Name:
AndrewID:

Instructions:

- Fill in your name and Andrew ID above. Be sure to write neatly, or you may not receive credit for your exam.
- Clearly mark your answers in the allocated space **on the front of each page.** If needed, use the back of a page for scratch space, but you will not get credit for anything written on the back of a page. If you have made a mistake, cross out the invalid parts of your solution, and circle the ones which should be graded.
- No electronic devices may be used during the exam.
- Please write all answers in pen.
- You have N/A to complete the exam. Good luck!

Instructions for Specific Problem Types

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

Select One: Who taught this course?

- ● 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?

- ● 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
- ☐ I don’t know

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

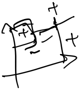
---

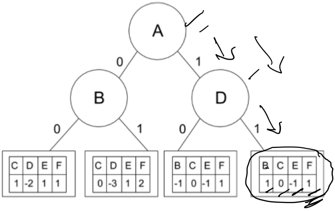
1 Decision Trees

1. To exploit the desirable properties of decision tree classifiers and perceptrons, Adam came up with a new algorithm called the “perceptron tree” that combines features from both. Perceptron trees are similar to decision trees, but each leaf node contains a perceptron rather than a majority vote.

   To create a perceptron tree, the first step is to follow a regular decision tree learning algorithm (such as ID3) and perform splitting on attributes until the specified maximum depth is reached. Once maximum depth has been reached, at each leaf node, a perceptron is trained on the remaining attributes which have not yet been used in that branch. Classification of a new example is done via a similar procedure. The example is first passed through the decision tree based on its attribute values. When it reaches a leaf node, the final prediction is made by running the corresponding perceptron at that node.

Assume that you have a dataset with 6 binary attributes {A, B, C, D, E, F} and two output labels {-1, 1}. A perceptron tree of depth 2 on this dataset is given below. Weights of the perceptron are given in the leaf nodes. Assume bias $b = 1$ for each perceptron.





$1*1 + 0*0 + 0*-1$

$1*1 + 1 = 3$

Figure 1: Perceptron Tree of depth 2

(a) **Numerical answer:** What would the given perceptron tree predict as the output label for the sample $\mathbf{x} = [1, 1, 0, 1, 0, 1]$?

+1

(b) **True or False:** The decision boundary of a perceptron tree will *always* be linear.

○ True

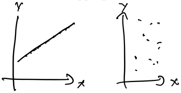
● False

(c) **True or False:** For small values of max depth, decision trees are *more* likely to underfit the data than perceptron trees.

● True

○ False

2. (2 points) **Select all that apply:** Given an input feature vector $\mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^n$, you are tasked with predicting a label for y, where $y = 1$ or $y = -1$. You have no knowledge about the distributions of $\mathbf{x}$ and of y. Which of the following methods are appropriate?



☐ Perceptron

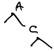
☒ k-Nearest Neighbors

☐ Linear Regression

☒ Decision Tree with unlimited depth

☐ None of the Above

3. **True or False:** The ID3 algorithm is guaranteed to find an optimal decision tree.



○ True

● False

4. **True or False:** One advantage of decision trees is that they are not easy to overfit.

○ True

● False

5. **True or False:** Neural the Narwhal flips two fair coins. Let Y denote the number of heads that are facing up. Then, the entropy of Y is 1.

○ True

● False

C_1 C_2 #H

$-\sum_i p \log_2 p$

$-\left(\frac{1}{4}\log_2 \frac{1}{4} + \frac{1}{2}\log_2 \frac{1}{2} + \frac{1}{4}\log_2 \frac{1}{4}\right)$

T T → 0 → 1/4
T H → 1 } → 1/2
H T → 1 }
H H → 2 → 1/4

2 K Nearest Neighbors

1. **Select one:** A k-Nearest Neighbor model with a large value of k is analogous to...
   - ○ A *short* Decision Tree with a *low* branching factor
   - ○ A *short* Decision Tree with a *high* branching factor
   - ○ A *long* Decision Tree with a *low* branching factor
   - ○ A *long* Decision Tree with a *high* branching factor
2. **Select one:** Imagine you are using a k-Nearest Neighbor classifier on a dataset with lots of noise. You want your classifier to be *less* sensitive to the noise. Which of the following is likely to help and with what side effect?
   - ○ Increase the value of $k \rightarrow$ Increase in prediction time
   - ○ Decrease the value of $k \rightarrow$ Increase in prediction time
   - ○ Increase the value of $k \rightarrow$ Decrease in prediction time
   - ○ Decrease the value of $k \rightarrow$ Decrease in prediction time
3. **Select all that apply:** Identify the correct relationship(s) between bias, variance, and the hyperparameter k in the k-Nearest Neighbors algorithm:
   - ☐ Increasing k leads to increase in bias
   - ☐ Decreasing k leads to increase in bias
   - ☐ Increasing k leads to increase in variance
   - ☐ Decreasing k leads to increase in variance
   - ☐ None of the above

4. Consider the following training dataset for a regression task:

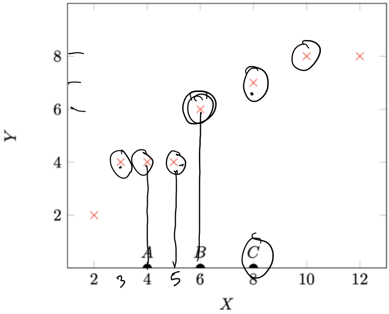
$$\mathcal{D} = \left\{ \left(x^{(1)}, y^{(1)}\right), \left(x^{(2)}, y^{(2)}\right), \cdots, \left(x^{(N)}, y^{(N)}\right) \right\}$$

with $x^{(i)} \in \mathbb{R}$ and $y^{(i)} \in \mathbb{R}$.

For regression with k-nearest neighbors, we make predictions on unseen data points similar to the classification algorithm, but instead of a majority vote, we take the mean of the output values of the k nearest points to some new data point x. That is,

$$h(x) = \frac{1}{k} \sum_{i \in \mathcal{N}(x, \mathcal{D})} y^{(i)}$$

where $\mathcal{N}(x, \mathcal{D})$ is the set of indices of the k closest training points to x.



In the above dataset, the red ×'s denote training points and the black semi-circles A, B, C denote test points of unknown output values. For convenience, all training data points have integer input and output values.

Any ties are broken by selecting the point with the lower x value.

(a) **Numerical answer:** When $k = 1$, what is the mean squared error on the training set?

0

(b) **Numerical answer:** When $k = 2$, what is the predicted value at A?

4

(c) **Numerical answer:** When $k = 2$, what is the predicted value at B?

5

(d) **Numerical answer:** When $k = 3$, what is the predicted value at C?

7

(e) **Numerical answer:** When $k = 8$, what is the predicted value at C?

5.375

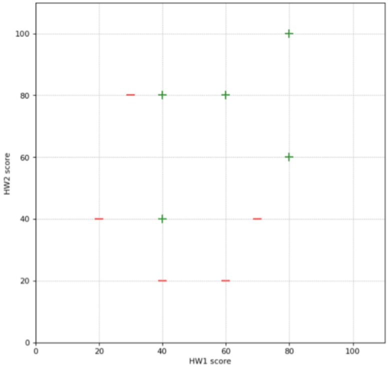
(f) **Math:** With $k = N$, for any dataset $\mathcal{D}$ with the form specified in the beginning of this question, write down a mathematical expression for the predicted value $\hat{y} = h(x)$. Your response shouldn't include a reference to the neighborhood function $\mathcal{N}()$.

$\frac{1}{N}\sum_{i} y^{i}$

5. You have just enrolled into your favourite course at CMU - Introduction to Machine Learning 10-301/601 - but you have not yet decided if you want to take it for a grade or as pass/fail. You want to use your performance in HW1 and HW2 to make this decision. You follow a general rule that if you can get at least a B in the course, you will take it for a grade, and if not, you will take it as pass/fail.

You have just learnt the new classification technique, k-NN, and wish to employ it to make your decision. You start by collecting data on prior student performance in HW1 and HW2, along with their final letter grades. You then create a binary label (1/0) based on the final grades such that you assign a label of 1 if the final grade is at least a B and 0 otherwise. Next, you train the model on this data set and calculate the training error. The distance measure you use is **Euclidean distance**.

Note: For ease of computation, we will use only 10 randomly selected training data points as plotted below. Label 1 is represented by '+' and green color; label 0 is represented by '-' and red color.



(a) **Numerical answer:** What will be the training error if you choose $k = 1$?

(b) **Numerical answer:** What will be the training error if you choose $k = 3$?

(c) **True or False:** Using Euclidean distance as the distance measure, the decision boundary of k-NN for $k = 1$ is a piece-wise straight line, that is, it contains only straight line segments. **Justify your answer.**

○ True

○ False

(d) **Drawing:** In the image above, draw a rough decision boundary for k = 1. Clearly label the + and - sides of the decision boundary.

(e) You have scored 60 in HW1 and 40 in HW2, and you now want to predict if your final grade would be at least a B.

i. **Numerical answer:** What would be the predicted class (1/0) for $k = 1$?

ii. **Numerical answer:** What would be the predicted class (1/0) for $k = 3$?

(f) **Short answer:** Looking at the training errors, you choose the model with k = 1 as it has the lowest training error. Do you think this is the right approach to select a model? Why or why not?

6. **Select all that apply:** Please select all that apply about k-NN in the following options. Assume a point can be its own neighbor.

- ☐ k-NN works great with a small amount of data, but struggles when the amount of data becomes large.
- ☐ k-NN is sensitive to outliers; therefore, in general we decrease k to avoid overfitting.
- ☐ k-NN can be applied to classification problems but not regression problems.
- ☐ We can always achieve zero training error (perfect classification) with k-NN, but it may not generalize well in testing.

3 Model Selection and Errors

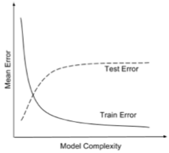
1. **Train and test errors:** In this problem, we will see how you can debug a classifier by looking at its train and test errors. Consider a classifier trained until convergence on some training data $\mathcal{D}^{\text{train}}$, and tested on a separate test set $\mathcal{D}^{\text{test}}$. You look at the test error, and find that it is very high. You then compute the training error and find that it is close to 0.

   (a) **Short Answer:** What is this scenario called?

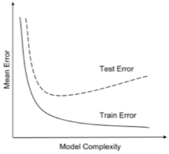
   (b) **Select all that apply:** Which of the following are expected to help?

   - ☐ Increasing the training data size.
   - ☐ Decreasing the training data size.
   - ☐ Increasing model complexity (For example, if your classifier is an SVM, use a more complex kernel. Or if it is a decision tree, increase the depth).
   - ☐ Decreasing model complexity.
   - ☐ Training on a combination of $\mathcal{D}^{\text{train}}$ and $\mathcal{D}^{\text{test}}$ and test on $\mathcal{D}^{\text{test}}$
   - ☐ None of the above

   (c) **Select one:** Say you plot the train and test errors as a function of the model complexity. Which of the following two plots is your plot expected to look like?



(a)



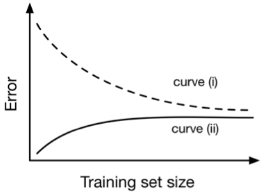
(b)

- ○ Plot A
- ○ Plot B

---

2. **Training Sample Size:** In this problem, we will consider the effect of training dataset size n on a logistic regression classifier with d features. The classifier is trained by optimizing the conditional log-likelihood. The optimization procedure stops if the estimated parameters perfectly classify the training data or they converge.

   The following plot shows the general trends in training and testing error as we increase the sample size $n = |S|$.



(a) **Short Answer:** Which curve represents the training error? Provide 1-2 sentences of justification.

(b) **Short Answer:** In one word, what does the gap between the two curves represent?

3. What are the effects of the following on overfitting? Choose the best answer.

(a) Increasing decision tree max depth.

○ Less likely to overfit

○ More likely to overfit

(b) Increasing decision tree mutual information split threshold.

○ Less likely to overfit

○ More likely to overfit

(c) Increasing decision tree max number of nodes.

○ Less likely to overfit

○ More likely to overfit

(d) Increasing k in k-nearest neighbor.

○ Less likely to overfit

○ More likely to overfit

(e) Increasing the training data size for decision trees. Assume that training data points are drawn randomly from the true data distribution.

○ Less likely to overfit

○ More likely to overfit

(f) Increasing the training data size for 1-nearest neighbor. Assume that training data points are drawn randomly from the true data distribution.

○ Less likely to overfit

○ More likely to overfit

4. Consider a learning algorithm that uses two hyperparameters, γ and ω, and it takes 1 hour to train *regardless* of the size of the training set.

We choose to do random subsampling cross-validation, where we do K runs of cross-validation and for each run, we randomly subsample a fixed fraction αN of the dataset for validation and use the remaining for training, where $\alpha \in (0, 1)$ and N is the number of data points.

(a) **Numerical answer:** In combination with the cross-validation method above, we choose to do grid search on discrete values for the two hyperparameters.

Given $N = 1000$ data points, $K = 4$ runs, and $\alpha = 0.25$, if we have 100 hours to complete the entire cross-validation process, what is the maximum number of discrete values of γ that we can include in our search if we also want to include 8 values of ω? Assume that any computations other than training are negligible.

$\boxed{3}$

(b) **Short answer:** In one sentence, give one advantage of increasing the value of α.

100h

25h
25h
25h
25h

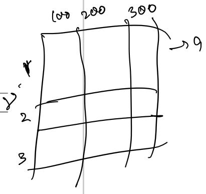
1,2,3

$\gamma \to 3$

$w \to 100, 200, 1000$

$w \to 8$

$w \to \gamma$? $8\gamma \leq 25$ $\gamma = 2$



---

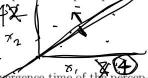
4 Perceptron

1. **Select all that apply:** Let $S = \{(\mathbf{x}^{(1)}, y^{(1)}), \cdots, (\mathbf{x}^{(n)}, y^{(n)})\}$ be n linearly separable points by a separator through the origin in $\mathbb{R}^d$. Let S' be generated from S as: $S' = \{(c\mathbf{x}^{(1)}, y^{(1)}), \cdots, (c\mathbf{x}^{(n)}, y^{(n)})\}$, where $c > 1$ is a constant. Suppose that we would like to run the perceptron algorithm on both data sets separately, and that the perceptron algorithm converges on S. Which of the following statements are true?

$\begin{bmatrix}1\\1\end{bmatrix}$ $\begin{bmatrix}2\\2\end{bmatrix}$

- ☐ The mistake bound of perceptron on S' is larger than the mistake bound on S ✗
- ☑ The perceptron algorithm when run on S and S' returns the same classifier, modulo constant factors (i.e., if $\mathbf{w}_S$ and $\mathbf{w}_{S'}$ are outputs of the perceptron for S and S', then $\mathbf{w}_S = c_1\mathbf{w}'_S$ for some constant c_1) 4✗
- ☑ The perceptron algorithm converges on S'.
- ☐ None of the above.

$C = 2$



$\left(\frac{R}{\gamma}\right)^2 \Rightarrow \left(\frac{2R}{2\gamma}\right)^2$

2

4

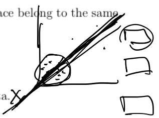
2. **True or False:** Given a linearly separable dataset, the convergence time of the perceptron algorithm depends on the sample size n.

$\left(\frac{R}{\gamma}\right)^2$

- ○ True
- ● False

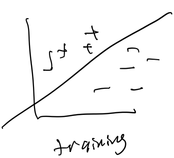
3. **Select all that apply:** Which of the following are inductive biases of the perceptron algorithm?

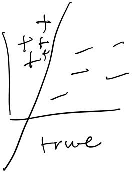
- ☐ Most of the cases in a small neighborhood in feature space belong to the same class. ✗
- ☑ The true decision boundary is linear.
- ☑ We prefer to correct the most recent mistakes.
- ☐ We prefer the simplest hypothesis that explains the data. ✗
- ☐ None of the above.



4. **True or False:** If the training data is linearly separable, the perceptron algorithm always finds the optimal decision boundary for the true distribution.

- ○ True
- ● False





5. (1 point) **True or False:** Consider two datasets $\mathcal{D}^{(1)}$ and $\mathcal{D}^{(2)}$, where $\mathcal{D}^{(1)} = \{(x_1^{(1)}, y_1^{(1)}), ..., (x_n^{(1)}, y_n^{(1)})\}$ and $\mathcal{D}^{(2)} = \{(x_1^{(2)}, y_1^{(2)}), ..., (x_m^{(2)}, y_m^{(2)})\}$ such that $x_i^{(1)} \in \mathbb{R}^{d_1}$, $x_i^{(2)} \in \mathbb{R}^{d_2}$, $d_1 > d_2$ and $n > m$. The maximum number of mistakes the perceptron algorithm will make is always higher for dataset $D^{(1)}$ than it is for dataset $D^{(2)}$.

$\left(\frac{R}{\sqrt{n}}\right)$ A

○ True

☒ False

6. Suppose you are given the following dataset:

Example Number	X_1	X_2	Y
1	-1	2	-1
2	-2	-2	+1
3	1	-1	+1
4	-3	1	-1

You wish to perform the Batch Perceptron algorithm on this data. Assume you start with initial weights $\theta^T = [0, 0]$ and bias $b = 0$, and that you pass through all of the examples in order of their example number.

i. **Numerical answer:** What would be the updated weight vector θ after we pass example 1 through the Perceptron algorithm?

$\boxed{[1,-2]}$

$0 \cdot -1 + 0 \cdot 2 + 0 = 0 \geq 0 \Rightarrow Y = +1$

misclassify neg $\underline{w = w - x}$ $b = b - 1$ $[0\ 0] - [-1\ 2] = [1, -2]$

ii. **Numerical answer:** What would be the updated bias b after we pass example 1 through the Perceptron algorithm?

$\boxed{-1}$

$b = 0 - 1 = -1$

iii. **Numerical answer:** What would be the updated weight vector θ after we pass example 2 through the Perceptron algorithm?

$\boxed{[1,-2]}$ ← $\theta^T = [1, -2]$

$[-2\ -2]$ $+1$ $b = -1$

$\theta x + b$
$= 1 \cdot \overset{-2}{-2} + \overset{4}{-2 \cdot -2} \overset{-1}{-1}$
$= 1 \geq 0 \Rightarrow +1$ correct

iv. **Numerical answer:** What would be the updated bias b after we pass example 2 through the Perceptron algorithm?

$\boxed{-1}$

v. **Numerical answer:** What would be the updated weight vector θ after we pass example 3 through the Perceptron algorithm?

$\boxed{[1,-2]}$

$\theta x + b \rightarrow \hat{y} \overset{=}{\neq} y_3$

---

vi. **Numerical answer:** What would be the updated bias b be after we pass example 3 through the Perceptron algorithm?

$\boxed{-1}$

example 4
$[-3, 1]$ $y = -1$
$-3 - 2 - 1 < 0$

vii. **True or False:** Your friend stops you here and tells you that you do not need to update the Perceptron weights or bias anymore; is this true or false?

● True

○ False

if labels every thing correctly ✓

$w = [1, -2]$ $b = -1$
example 1 $[-1, 2]$ $y = -1$
$-1 - 4 - 1 < 0$

7. **True or False:** Data (X, Y) has a non-linear decision boundary. Fortunately, there is a function $\mathcal{F}$ that maps (X, Y) to $(\mathcal{F}(X), Y)$ such that $(\mathcal{F}(X), Y)$ is linearly separable. We have tried to build a modified perceptron to classify (X, Y). Is the given (modified) perceptron update rule correct?

$X \rightarrow \mathcal{F}(X)$

if $\text{sign}(w\overbrace{\mathcal{F}(x^{(i)})}^{x} + b) \mathrel{!=} y^{(i)}$:
  $w' = w + y^{(i)}\mathcal{F}(x^{(i)})$ x
  $b' = b + y^{(i)}$

● True

○ False

8. (a) **True or False:** All *examples* $(\mathbf{x}, y)$ that the perceptron algorithm has seen are weighted equally.

○ True

● False

5 Linear Regression

$X^TX : 6\times6$

1. **Select one:** The closed form solution for linear regression is $\overset{6\times1}{\theta} = (\underbrace{X^T}_{6\times35}\underbrace{X}_{35\times6})^{-1}\overset{6\times35}{X^T}\overset{35\times1}{y}$. Suppose you have $n = 35$ training examples and $m = 5$ features (excluding the bias term). Once the bias term is folded in, what are the dimensions of X, y, θ?

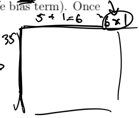
● X is 35×6, y is 35×1, θ is 6×1

○ X is 35×6, y is 35×6, θ is 6×6

○ ~~X is 35×5,~~ y is 35×1, ~~θ is 5×1~~

○ X is $35 \times$ ~~5, y~~ is 35×5, ~~θ is 5×5~~

35 6
35×6
35×1
$5+1=6$



2. Answer the following True/False questions, providing brief explanations to support your answers.

   (a) **True or False:** Consider a linear regression model with only one parameter, the bias, i.e., $y = \beta_0$. Then, given n data points (x_i, y_i) (where x_i is the feature and y_i is the output), minimizing the sum of squared errors results in β_0 being the median of the y_i values.

   β_0

   ○ True

   ● False

wo, minimizing this results in the mean of the y_i values

$$\frac{\sum_{i=0}^{n}(\beta_0 - y_i)^2}{n}$$

$$\Rightarrow (\beta_0 - y_0)^2 + (\beta_0 - y_1)^2 \dots$$

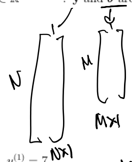
$$\Rightarrow 2(\beta_0 - y_0) + 2(\beta_0 - y_1) \dots$$

$$\Rightarrow \beta_0 = \frac{\sum y_i}{n}$$

---

3. **Select all that apply:** Which of the following are valid expressions for the mean squared error objective function for linear regression with dataset $\mathcal{D} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{N}$, with each $\mathbf{x}^{(i)} \in \mathbb{R}^M$ and the design matrix $\mathbf{X} \in \mathbb{R}^{N\times(M+1)}$? $\mathbf{y}$ and $\boldsymbol{\theta}$ are column vectors.

   - ~~☐ $J(\boldsymbol{\theta}) = \frac{1}{N}\|\mathbf{y} - \boldsymbol{\theta}\mathbf{X}\|_2^2$~~
   - ~~☐ $J(\boldsymbol{\theta}) = \frac{1}{N}\|\mathbf{y}^T - \boldsymbol{\theta}\mathbf{X}\|_2^2$~~
   - ☐ $J(\boldsymbol{\theta}) = \frac{1}{N}\|\mathbf{y}^T - \mathbf{X}\boldsymbol{\theta}\|_2^2$
   - ☑ $J(\boldsymbol{\theta}) = \frac{1}{N}\|\underset{N\times 1}{\mathbf{X}\boldsymbol{\theta}} - \mathbf{y}\|_2^2$ $N\times 1$
   - ☐ None of the Above



θX

$\Rightarrow (M+1)\times 1$

$N \times (M+1)$

$y: N\times 1$

$1\times N$

$X\theta = N\times(M+1) \times (M+1)\times 1 = N\times 1$

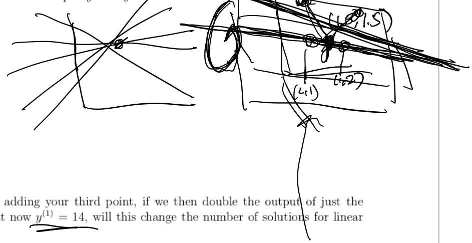
4. **Numerical answer:** We have 2 data points:

$$\mathbf{x}^{(1)} = [2, 1]^T \quad y^{(1)} = 7$$

$$\mathbf{x}^{(2)} = [1, 2]^T \quad y^{(2)} = 5$$

We know that for linear regression with a bias/intercept term and mean squared error objective function, there are infinite solutions with these two points.

Give a specific third point $\mathbf{x}^{(3)}, y^{(3)}$ such that, when included with the first two, will cause linear regression to still have infinite solutions. Your $\mathbf{x}^{(3)}$ should not equal $\mathbf{x}^{(1)}$ or $\mathbf{x}^{(2)}$ and your $y^{(3)}$ should not equal $y^{(1)}$ or $y^{(2)}$.



$x_1^{(3)}$

1.5

$x_2^{(3)}$

1.5

$y^{(3)}$

whatever

Select one: After adding your third point, if we then double the output of just the first point such that now $y^{(1)} = 14$, will this change the number of solutions for linear regression?

○ Yes

● No

5. Given that we have an input x and we want to estimate an output y, in linear regression we assume the relationship between them is of the form $y = wx + b + \epsilon$, where w and b are real-valued parameters we estimate and ϵ represents the noise in the data. When the noise is Gaussian, maximizing the likelihood of a dataset $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ to estimate the parameters w and b is equivalent to minimizing the squared error:

$$\arg\min_{w} \sum_{i=1}^{n} (y_i - (wx_i + b))^2.$$

Consider the dataset S plotted in Fig. 3 along with its associated regression line. For each of the altered data sets S^{new} plotted in Fig. 5, indicate which regression line (relative to the original one) in Fig. 4 corresponds to the regression line for the new data set. Write your answers in the table below.

Dataset	(a)	(b)	(c)	(d)	(e)
Regression line	b	c	b	a	a

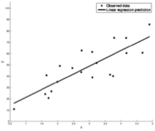
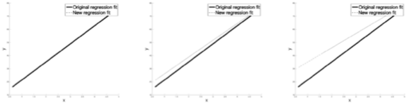


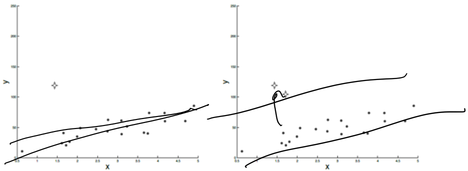
Figure 3: An observed data set and its associated regression line.



(a) Old and new regression lines. (b) Old and new regression lines. (c) Old and new regression lines.

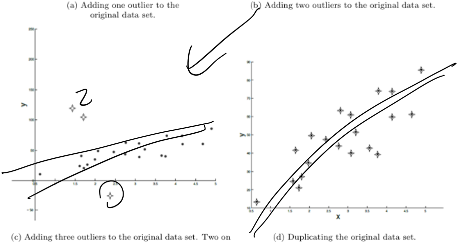
same shift a little up shift a lot up

Figure 4: New regression lines for altered data sets S^{new}.



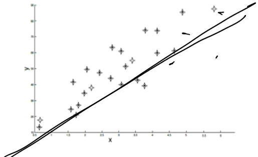
(a) Adding one outlier to the original data set.

(b) Adding two outliers to the original data set.



(c) Adding three outliers to the original data set. Two on one side and one on the other side.

(d) Duplicating the original data set.



(e) Duplicating the original data set and adding four points that lie on the trajectory of the original regression line.

Figure 5: New data set S^{new}.