COMP 3105 Introduction to Machine Learning
Practice Final

Instructor: Junfeng Wen (junfeng.wen [AT] carleton.ca)

Fall 2023
School of Computer Science
Carleton University

Note:

- This exam has **14 pages** (including this one)
- It has **7 questions**
- Exam time: **3 hours**
- You can bring one A4 page of your note to the final exam
- Fill your **name**, **ID** and **email** address to the following table
- Good luck!

First Name:	
Last Name:	
Student ID:	
Carleton Email Address:	

Question	Max Score	Score
Q1: True/False	20	
Q2: Multiple Choice	30	
Q3: Logistic Regression	10	
Q4: SVM	10	
Q5: Backprop	9	
Q6: Decision Trees	10	
Q7: MLP	11	
Total	100	

Question 1 (20 points) True/False

Instruction: Check either "True" or "False" as your answer. Each question is worth 2 points. 0 point will be given if the question is answered wrong, not attempted or if your answer is ambiguous/illegible.

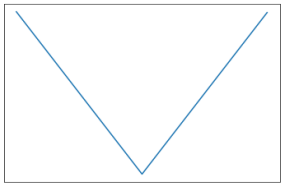
1. One can use the performance on the test data to choose the hyper-parameters of the algorithm.
☐ True ☑ False

2. Linear regression is an unsupervised learning algorithm.
☐ True ☑ False

3. Support vector machine is a classification algorithm.
☑ True ☐ False

4. In linear regression, using L_1 loss is more robust to outliers than using L_2 loss.
☑ True ☐ False

5. The following function is convex.



☑ True ☐ False

6. The objective of k-means is guaranteed to increase after each iteration.
☐ True ☑ False

7. Principle component analysis finds the projecting directions with the minimum variations.
☐ True ☑ False

8. Using the L_2 loss is equivalent to imposing a Gaussian likelihood on the data.
☑ True ☐ False

9. The derivatives of the parameters in a neural network are computed during forward propagation.
☐ True ☑ False

10. Gradient clipping can be used to prevent vanishing gradient problem in RNN.
☐ True ☑ False

Question 2 (30 points) Multiple Choice Questions

Instruction: Each question is worth 2 points and has **exactly one** correct choice. Circle your answer. You get 2 points if your answer is correct. 0 point will be given if the question is answered wrong, not attempted or if your answer is ambiguous/illegible.

1. Which of the following is NOT used to prevent overfitting?
A. Regularization
B. Cross-validation
C. Early stopping
D. Feature expansion

2. Suppose you are using L_2 regularization (i.e., $\frac{\lambda}{2}\|\mathbf{w}\|_2^2$) to prevent overfitting for a binary classification problem. As you increase the hyper-parameter λ, the accuracy on the validation dataset also increases. What should you try next?
A. Keep increasing λ
B. Decrease λ
C. Take a look at the accuracy of the test data
D. Not enough information given

3. Alice and Bob are both training a linear model for logistic regression. Alice initializes the model weights with zeros, while Bob initializes them with ones. They both train the model via gradient descent with a proper step-size / learning rate. Who will get better training accuracy at the end?
A. Alice will have significantly better accuracy
B. Bob will have significantly better accuracy
C. They will achieve about the same accuracy
D. Not enough information given

4. Which of the following is the correct gradient of the objective

$$J(\mathbf{w}) = \|X\mathbf{w} - \mathbf{y}\|_2^2?$$

A. $2X^\top(X\mathbf{w} - \mathbf{y})$
B. $2X^\top X\mathbf{w} - X^\top\mathbf{y}$
C. $2(X^\top X\mathbf{w} - \mathbf{y})$
D. $2(X\mathbf{w} - \mathbf{y})$

5. Which of the following polynomials performs the best on the test data $(x, y) = (1, 0.5)$ with the smallest L_1 loss $|\widehat{y} - y|$?
A. $\widehat{y} = -x$
B. $\widehat{y} = x^2 + 0.5$
C. $\widehat{y} = x^3 - x$
D. $\widehat{y} = 2x$

6. Recall that for binary classification, the misclassification loss can be represented as $L_{0/1}(\widehat{z}, y) = \mathbb{I}(\widehat{z}y < 0)$, where $\widehat{z} = \mathbf{x}^\top\mathbf{w}$ is the linear prediction and $y \in \{-1, +1\}$ is the ground-truth label. One can replace the 0/1 loss with a surrogate loss. Which of the following properties is NOT true for the surrogate loss $L(\widehat{z}, y) = \exp(-\widehat{z}y)$?
A. It is an upper bound of the 0/1 loss
B. It is a convex loss
C. It is robust to outliers
D. It is smooth/differentiable

7. Given 3 data points in 2D space as follows: $[0,0], [2,2]$, and $[-1,-1]$. Which of the following can be the first principal component direction?
A. $[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}]$
B. $[-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}]$
C. $[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}]$
D. $[1,0]$

8. Suppose that the first principal component direction of a data set is given by $[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0]$. Which of the following CANNOT be the second principal component direction?
A. $[0,0,1,0]$
B. $[-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0]$
C. $[0,0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}]$
D. $[1,0,0,0]$

9. Which of the following is NOT a clustering algorithm?
A. k-means
B. Spectral clustering
C. Hierarchical clustering
D. Principal component analysis

10. What's the difference between batch gradient descent and stochastic gradient descent?
A. Batch gradient descent uses one single example to compute the gradient while stochastic gradient descent uses the whole dataset
B. Batch gradient descent uses the whole dataset to compute the gradient while stochastic gradient descent uses one single example
C. Batch gradient descent uses a small subset of examples to compute the gradient while stochastic gradient descent uses the whole dataset
D. Batch gradient descent uses a small subset of examples to compute the gradient while stochastic gradient descent uses one single example

11. When training a neural network for a binary classification problem, which of the following should be used as the output activation function?
A. ReLU
B. Softplus
C. Sigmoid
D. Identity

12. Suppose you build an MLP model with 2 hidden layers, each hidden layer has 10 hidden units. There are 10 input neurons and one single output neuron. **Including bias parameters**, how many parameters are there in the model?
A. 1000
B. 231
C. 210
D. 31

13. Suppose the input to a convolutional layer is $32 \times 32 \times 16$ and the layer has one single 1×1 filter. **Including bias parameters**, how many parameters are there in this 2d convolutional layer?
A. 2
B. 17
C. 1025
D. 16384

14. Which of the following is the most suitable for predicting time series data such as stock prices?
A. Gated recurrent unit
B. Support vector machine
C. Multilayer perceptron
D. Linear regression

15. Alice implemented a neural network model with a few hidden layers. She noticed that some values of the hidden neurons are negative after activation. Which of the following is a possible choice for the activation function in those hidden layers?
A. ReLU
B. Sigmoid
C. Tanh
D. Exponential

Question 3 (10 points) Logistic Regression

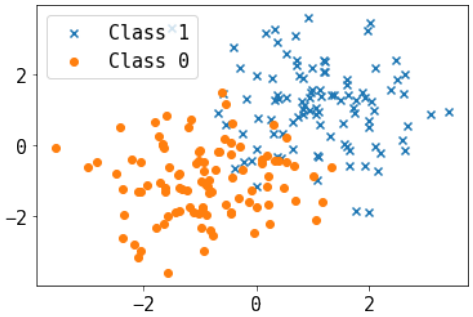


Figure 1: Data from two classes

(a) (3 points) As shown in Fig. 1, there are two sets of data coming from two different classes. Can you find a linear model that separate them perfectly with 100% accuracy? Why or why not?

No, because there is sufficient overlap between the two classes

(b) (2 points) The logistic/sigmoid function is given by

$$y = \sigma(z) = \frac{1}{1 + \exp(-z)}.$$

Calculate the gradient of the sigmoid function w.r.t. input z, and represent the gradient using only y.

$$\begin{aligned}
y &= (1 + \exp(-z))^{-1} \\
\frac{\partial y}{\partial z} &= (-1) \cdot (1 + \exp(-z))^{-2} \cdot \exp(-z) \cdot (-1) \\
&= \frac{\exp(-z)}{(1 + \exp(-z))^2} \\
&= \frac{\exp(-z) + 1 - 1}{(1 + \exp(-z))^2} \\
&= \frac{1}{1 + \exp(-z)} - \frac{1}{(1 + \exp(-z))^2} \\
&= \frac{1}{1 + \exp(-z)} \left(1 - \frac{1}{(1 + \exp(-z))}\right) \\
&= y\,(1 - y)
\end{aligned}$$

(c) (3 points) Suppose you learn a classifier that achieves the following confusion matrix:

	Predicted class 0	Predicted class 1
Actual class 0	80	20
Actual class 1	12	88

where each row represents the numbers of data points in an actual class (ground-truth labels), while each column represents the numbers of data points in a predicted class (predicted labels). What is the accuracy of the model?

# of correct 80+88=168
# of all 80+20+12+88=200
accuracy=168/200=0.84

(d) (2 points) One can use two different approaches to learn binary classifiers

- With sigmoid, $P(y=1) = \sigma(\mathbf{x}^\top \mathbf{w}_s)$ and $P(y=0) = 1 - \sigma(\mathbf{x}^\top \mathbf{w}_s)$
- With softmax, $P(y=1) = \exp(\mathbf{x}^\top \mathbf{w}_1)/Z$ and $P(y=0) = \exp(\mathbf{x}^\top \mathbf{w}_0)/Z$ where $Z = \exp(\mathbf{x}^\top \mathbf{w}_1) + \exp(\mathbf{x}^\top \mathbf{w}_0)$ is for normalization so that $P(y=1) + P(y=0) = 1$.

Is the softmax method more powerful than the sigmoid method?

- If yes, show a configuration of $(\mathbf{w}_1, \mathbf{w}_0)$ that is NOT representable by a single $\mathbf{w}_s$ (i.e., there exist $P(y=1), P(y=0)$ that are reresentable by some $\mathbf{w}_1, \mathbf{w}_0$ but not by $\mathbf{w}_s$ when $P(y=1) + P(y=0) = 1$).
- If no, show an equation for $\mathbf{w}_s$ using $\mathbf{w}_0, \mathbf{w}_1$ that achieves the same $P(y=1), P(y=0)$.

Hint: For simplicity, you can focus on the scalar case here where x, w_s, w_1, w_0 are all scalars.

No.

Sigmoid

$$P(y=1) = \frac{1}{1+\exp(-xw_s)}$$

Softmax

$$\begin{aligned} P(y=1) &= \frac{\exp(xw_1)}{\exp(xw_1)+\exp(xw_0)} \\ &= \frac{1}{1+\frac{\exp(xw_0)}{\exp(xw_1)}} \\ &= \frac{1}{1+\exp(xw_0 - xw_1)} \\ &= \frac{1}{1+\exp(-x(w_1 - w_0))} \end{aligned}$$

Thus there exists $w_s = w_1 - w_0$ that achieves the same predictions.

Question 4 (10 points) Support Vector Machine

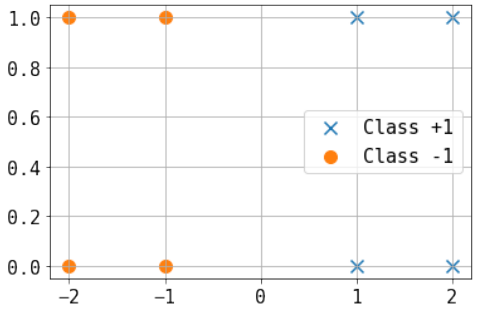


Figure 2: Data from two classes

(a) (2 points) Suppose we are using hard SVM to classify the data in Fig. 2 with $(1,0),(1,1),(2,0),(2,1)$ from class $+1$ and $(-1,0),(-1,1),(-2,0),(-2,1)$ from class -1. Which points are support vectors?

The decision boundary provides the maximum margin, so it should be x1=0.

Then the SVs are the points closest to the boundary, which are (1, 0), (1, 1), (-1, 0), (-1, 1)

(b) (3 points) Which parameters correctly represent the SVM decision boundary? And what's the margin it achieves?

- $\mathbf{w}_A = (0,1), b_A = 0$
- $\mathbf{w}_B = (1,0), b_B = 0$
- $\mathbf{w}_C = (1,1), b_C = 0$
- $\mathbf{w}_D = (-1,-1), b_D = 0$

Hint: Recall that the decision boundary of SVM is given by the following

$$\widehat{y} = \text{sign}(\mathbf{x}^\top \mathbf{w} + b) = \begin{cases} +1 & \text{if} \quad \mathbf{x}^\top \mathbf{w} + b \geq 0 \\ -1 & \text{if} \quad \mathbf{x}^\top \mathbf{w} + b < 0 \end{cases}$$

and the margin is given by $\gamma = \frac{1}{\|\mathbf{w}\|_2}$.

Since the boundary is x1=0, the corresponding normal vector should be w=(1, 0)

Note that it is not (0, -1) because the normal vector should point to the +1 class so that the points are correctly classified here.

(c) (2 points) Recall that SVM is compatible with a kernel function to learn a non-linear decision boundary. A valid kernel $k(x_i, x_j)$ corresponds to an inner product $k(x_i, x_j) = \boldsymbol{\phi}(x_i)^\top \boldsymbol{\phi}(x_j)$ for some expanded features $\boldsymbol{\phi}$. Consider the feature mapping $\boldsymbol{\phi}(x) = [1, \sin(x), x^2]$. Find the corresponding kernel function $k(x_i, x_j)$.

$$\boldsymbol{\phi}(x_i) = \begin{bmatrix} 1 \\ \sin(x_i) \\ x_i^2 \end{bmatrix} \quad \boldsymbol{\phi}(x_j) = \begin{bmatrix} 1 \\ \sin(x_j) \\ x_j^2 \end{bmatrix}$$

$$k(x_i, x_j) = \boldsymbol{\phi}(x_i)^\top \boldsymbol{\phi}(x_j) = 1 + \sin(x_i)\sin(x_j) + x_i^2 x_j^2$$

(d) (3 points) Suppose we are using the following kernel function

$$k(x_i, x_j) = \exp\left(-\frac{(x_i - x_j)^2}{2\sigma^2}\right)$$

for one dimensional input where input x is a scalar (i.e., $x \in \mathbb{R}$). Show that this is a valid kernel.

Hint: We know that $k(x_i, x_j) = x_i x_j$ is a valid kernel because it can be represented as the product of $\phi(x_i) = x_i$ and $\phi(x_j) = x_j$. The following modified kernels are still valid when k, k_1, k_2 are valid kernels.

- $\widetilde{k}(x, y) = k(x, y) + c, c \geq 0$
- $\widetilde{k}(x, y) = \frac{k(x,y)}{\sqrt{k(x,x) \cdot k(y,y)}}$
- $\widetilde{k}(x, y) = k_1(x, y) + k_2(x, y)$
- $\widetilde{k}(x, y) = a \cdot k(x, y), a > 0$
- $\widetilde{k}(x, y) = k_1(x, y) k_2(x, y)$
- $\widetilde{k}(x, y) = \exp(k(x, y))$

We start from the basic kernel and the subsequent kernels will remain valid according to the rules above

$$\begin{aligned}
& k(x_i, x_j) = x_i x_j \\
\implies & k(x_i, x_j) = x_i x_j / \sigma^2 && \text{(Rule \#4)} \\
\implies & k(x_i, x_j) = \exp(x_i x_j / \sigma^2) && \text{(Rule \#6)} \\
\implies & k(x_i, x_j) = \frac{\exp(x_i x_j / \sigma^2)}{\sqrt{\exp(x_i^2/\sigma^2)}\sqrt{\exp(x_j^2/\sigma^2)}} && \text{(Rule \#2)} \\
& = \frac{\exp(x_i x_j / \sigma^2)}{\exp(x_i^2/2\sigma^2) \cdot \exp(x_j^2/2\sigma^2)} && ((\exp A)^c = \exp(cA); \text{Let } c = 1/2) \\
& = \exp\left(\frac{x_i x_j}{\sigma^2} - \frac{x_i^2}{2\sigma^2} - \frac{x_j^2}{2\sigma^2}\right) && \left(\frac{\exp A}{\exp B} = \exp(A - B)\right) \\
& = \exp\left(-\frac{1}{2\sigma^2}(x_i^2 - 2x_i x_j + x_j^2)\right) \\
& = \exp\left(-\frac{(x_i - x_j)^2}{2\sigma^2}\right) && ((A^2 - 2AB + B^2) = (A - B)^2)
\end{aligned}$$

Question 5 (9 points) Backprop

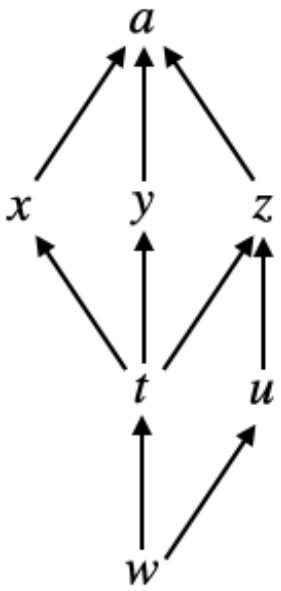


Figure 3: Computational graph

For the computational graph in Fig. 3, each node represents a variable, and each arrow pointing from one variable to another means the latter depends on the former during the forward computation. Use immediate derivatives (from **one single arrow**; for example $\frac{\partial a}{\partial y}$) to calculate the following.

(a) (3 points) $\frac{\partial a}{\partial t}$ the derivative of a w.r.t. t.

$$\frac{\partial a}{\partial t} = \frac{\partial a}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial a}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial a}{\partial z}\frac{\partial z}{\partial t} \doteq \delta_t$$

(b) (3 points) $\frac{\partial z}{\partial w}$ the derivative of z w.r.t. w.

$$\frac{\partial z}{\partial w} = \frac{\partial z}{\partial t}\frac{\partial t}{\partial w} + \frac{\partial z}{\partial u}\frac{\partial u}{\partial w}$$

(c) (3 points) $\frac{\partial a}{\partial w}$ the derivative of a w.r.t. w.

$$\text{Let } \delta_u \doteq \frac{\partial a}{\partial u} = \frac{\partial a}{\partial z}\frac{\partial z}{\partial u}$$

$$\text{Then } \frac{\partial a}{\partial w} = \delta_t \frac{\partial t}{\partial w} + \delta_u \frac{\partial u}{\partial w} = \cdots$$

Question 6 (10 points) Decision Tree

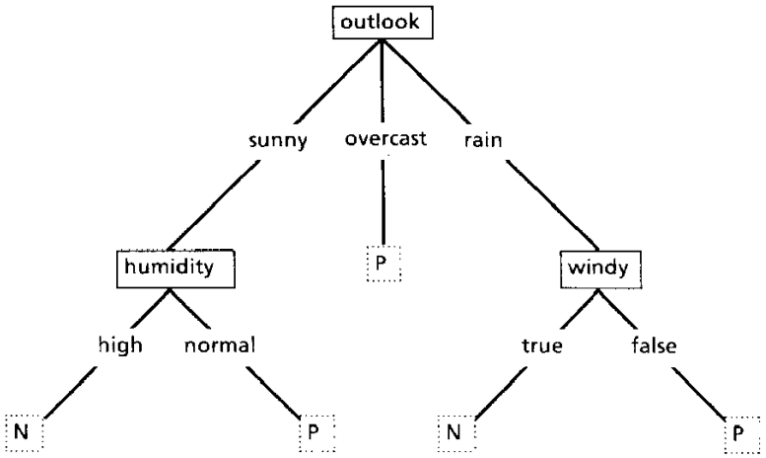


Figure 4: Decision tree

Day	Outlook	Temperature	Humidity	Windy	Class
1	sunny	hot	high	false	N
2	rain	mild	high	false	N
3	rain	cool	normal	false	P
4	sunny	mild	high	true	N
5	overcast	hot	normal	true	P
6	sunny	cool	normal	true	P
7	rain	mild	normal	false	P
8	sunny	hot	normal	true	N
9	overcast	cool	high	false	N
10	overcast	cool	normal	false	P

Figure 5: Dataset for decision tree

(a) (2 points) Consider the decision tree in Fig. 4, where the internal nodes are the attributes, the edges are the values of those attributes and the leaf nodes represent the binary prediction: positive (P) versus negative (N). What is the accuracy of this model on the given dataset in Fig. 5?

Correct examples: 1, 3, 4, 5, 6, 7, 10
Incorrect examples: 2, 8, 9
Accuracy: 7/10=0.7

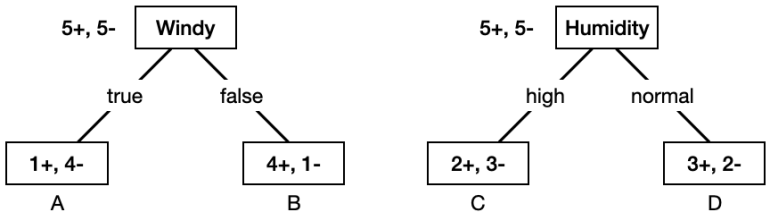


Figure 6: Split candidate

(b) (4 points) Consider the two possible split given in Fig. 6 where the numbers $(a+, b-)$ indicates that there are a positive examples and b negative examples in that node. Calculate the entropy at each of the nodes A, B, C and D.

Hint: Recall that the entropy is given by

$$H = -\frac{n_+}{n_+ + n_-} \log_2 \frac{n_+}{n_+ + n_-} - \frac{n_-}{n_+ + n_-} \log_2 \frac{n_-}{n_+ + n_-}.$$

where n_+, n_- are the numbers of positive and negative points respectively. The table below may be helpful.

x	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\log_2(x)$	-3.32	-2.32	-1.74	-1.32	-1.0	-0.74	-0.51	-0.32	-0.15

H(A) = - 1/5 log(1/5) - 4/5 log(4/5) = (-0.2)*(-2.32) + (-0.8)*(-0.32) = 0.718 = H(B)

(Note that H(A)=H(B) because they are symmetric)

H(C) = - 2/5 log(2/5) - 3/5 log(3/5) = (-0.4)*(-1.32) + (-0.6)*(-0.74) = 0.972 = H(D)

(c) (4 points) Calculate the conditional entropy of each split: $H(y|\text{windy})$ and $H(y|\text{humidity})$ where y is the label variable (P or N). Which split is preferred and why?

H(y|windy) = 5/10 H(A) + 5/10 H(B) = 0.718

H(y|humidity) = 5/10 H(C) + 5/10 H(D) = 0.972

The windy split is preferred because it has lower conditional entropy.

Question 7 (11 points) Feedforward Neural Net

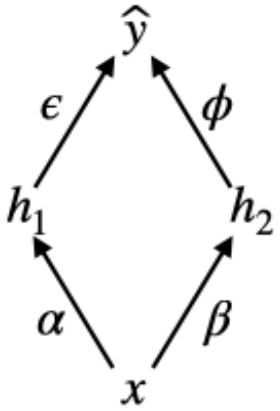


Figure 7: Simple MLP

Consider a simple feedforward model given by Fig. 7 where x is the input node, h_1, h_2 are the hidden nodes, $\widehat{y}$ is the output node, and all other letters are the parameters of the model. They are all scalars. For instance $h_1 = f(\alpha \cdot x)$ for some activation f.

(a) (2 points) Suppose that the hidden nodes and the output node all use tanh activation $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, and all the model parameters are initialized to be zeros. Before we train the model, what is the value of the output prediction $\widehat{y}$ for a training data point?

$$h_1 = \tanh(\alpha x) = \tanh 0 = 0$$
$$h_2 = \tanh(\beta x) = \tanh 0 = 0$$
$$\widehat{y} = \tanh(\epsilon h_1 + \phi h_2) = \tanh 0 = 0$$

(b) Suppose that the hidden nodes and the output node do not use any activation (e.g., $h_1 = \alpha x$). We use L_2 loss (i.e., $\frac{1}{2}(y - \widehat{y})^2$) to train the model for regression.

(b.1) (2 points) Show that this model is then equivalent to one single linear layer $\widehat{y} = w \cdot x$ for some parameter w.

$$\begin{aligned} \widehat{y} &= \epsilon h_1 + \phi h_2 \\ &= \epsilon\alpha x + \phi\beta x \\ &= (\epsilon\alpha + \phi\beta)x \end{aligned}$$
$$\text{So } w = \epsilon\alpha + \phi\beta$$
$$\widehat{y} = (\epsilon\alpha + \phi\beta)x$$

(b.2) (3 points) For the training point $(x, y) = (1, 1)$, which of the following configurations provides the best prediction $\widehat{y}$ with the least L_2 loss?

- $(\alpha, \beta, \epsilon, \phi) = (-1, -1, 0.5, 0.5)$
- $(\alpha, \beta, \epsilon, \phi) = (-0.5, 0.5, 1, 1)$
- $(\alpha, \beta, \epsilon, \phi) = (0.5, -0.5, -1, -1)$
- $(\alpha, \beta, \epsilon, \phi) = (1, -1, 0.5, -0.5)$

Based on the prediction formula (from above): $\widehat{y} = (\epsilon\alpha + \phi\beta)x$

their predictions are -1, 0, 0, and 1, respectively. Therefore the last configuration is the best for this training point

(b.3) (2 points) Calculate the derivative $\frac{\partial L_2}{\partial \epsilon}$ for a data point (x, y).

$$\frac{\partial L_2}{\partial \epsilon} = \frac{\partial L_2}{\partial \widehat{y}} \frac{\partial \widehat{y}}{\partial \epsilon} = (\widehat{y} - y)\alpha x$$

(b.4) (2 points) Calculate the derivative $\frac{\partial L_2}{\partial \alpha}$ for a data point (x, y).

$$\frac{\partial L_2}{\partial \alpha} = \frac{\partial L_2}{\partial \widehat{y}} \frac{\partial \widehat{y}}{\partial \alpha} = (\widehat{y} - y)\epsilon x$$