Final Exam - EECS 398, Winter 2025

Full Name:

Uniqname:

UMID:

Room: ◯ BBB 1670 ◯ CSRB 2246 ◯ DOW 1010 ◯ SSD/Alt

Instructions:

- You have 120 minutes to complete this exam.
- **This exam consists of 9 questions, worth a total of 100 points.**
- Write your uniqname in the top right corner of each page in the space provided.
- Please write **clearly** in the provided answer boxes; we will not grade work that appears elsewhere. Completely fill in bubbles and square boxes; if we cannot tell which option(s) you selected, you may lose points.

  ◯ A bubble means that you should only **select one choice**.

  ☐ A square box means you should **select all that apply**. Blank answers will receive no credit.
- You may refer to two double-sided handwritten notes sheets. Other than that, you may not refer to any other resources or technology during the exam (no phones, watches, or calculators).

You are to abide by the University of Michigan/Engineering Honor Code. To receive a grade, please sign below to signify that you have kept the Honor Code pledge.

I have neither given nor received aid on this exam, nor have I concealed any violations of the Honor Code.

Signature:

Data Overview: Gone Fishin'

In this exam, we'll work with the DataFrame `fish`, which contains information about various fish for sale at a market in Finland.

The first few rows of `fish` are shown below, but `fish` has many more rows than are shown.

	Height	Weight	Species	Width
0	5.20	78.0	Perch	3.12
1	2.43	13.4	Smelt	1.27
2	5.57	200.0	Pike	3.38
3	8.38	270.0	Whitefish	4.25
4	5.22	150.0	Perch	3.63
5	18.96	1000.0	Bream	6.60

Each row in `fish` contains information about a single fish. The columns in `fish` are as follows:

- `"Height" (float)`: The height of the fish, in inches.
- `"Weight" (float)`: The weight of the fish, in grams.
- `"Species" (str)`: The species of the fish. There are **7** possible species, 5 of which are shown in the example above.
- `"Width" (float)`: The width of the fish, in inches.

Assume that:

- All necessary import statements have been run.
- Unless otherwise specified, $\|\vec{v}\|$ refers to the L_2 norm of $\vec{v}$, i.e. $\|\vec{v}\| = \|\vec{v}\|_2 = \sqrt{v_1^2 + v_2^2 + ... + v_n^2}$.
- If $c \in \mathbb{R}$ is a scalar (i.e. a single number), then $\vec{c}$ is a vector in which every element is c, e.g. $\vec{2}$ is a vector of all 2s.

uniqname: _____

Question 1 (9 pts)

Suppose we use the code below to build a multiple linear regression model to predict the width of a fish, given its height and weight.

```
X, _, y, _ = train_test_split(fish["Height", "Weight"], fish["Width"])
model = LinearRegression()
model.fit(X, y)

# Used in the grid below.
ws = np.append(model.intercept_, model.coef_)
preds = model.predict(X)
squares = X.shape[0] * mean_squared_error(y, preds)
```

Assume that:

- X and `X` both represent the design matrix used to train the model, and that X **is full rank**.
- $\vec{y}$ and `y` both represent the observation vector used to train the model.
- $\vec{w}^*$ represents the optimal parameter vector found by `sklearn`.

In **each column** of the grid below, **select all** mathematical expressions that have the same value as the code expression provided in the column header. The first column has been done for you as an example. Some guidance:

- It is possible that some rows are left empty, but there should be at least one square filled in per column. *Tip: Look at one column at a time.*
- Assume, just for this part, that $\vec{0}$ and 0 are the same value, e.g. if an expression produces an array of 0s, you would select the 0 option in the second row.

		a)	*b)*	*c)*	*d)*
	`X.shape[1]`	`preds`	`ws`	`squares`	`np.sum(y - preds)`
3	■	☐	☐	☐	☐
0	☐	☐	☐	☐	☐
$\|\vec{y} - X\vec{w}^*\|^2$	☐	☐	☐	☐	☐
$X^TX\vec{w}^* - X^T\vec{y}$	☐	☐	☐	☐	☐
$\vec{1}^T(\vec{y} - X\vec{w}^*)$	☐	☐	☐	☐	☐
$(X^TX)^{-1}X^T\vec{y}$	☐	☐	☐	☐	☐
$X(X^TX)^{-1}X^T\vec{y}$	☐	☐	☐	☐	☐

Question 2 (5 pts)

Suppose we'd like to build a regression model to predict the width of a fish, given its height and weight.

a) (3 pts) For each of several model instances, we train a model twice — once with the features as-is, and once with standardized features. In other words:

```
# Without standardization
model = ModelClass() # e.g. model = Lasso(alpha=10000)
model.fit(X, y)

# With standardization
model_std = make_pipeline(StandardScaler(), ModelClass())
model_std.fit(X, y)
```

We say a model is **standardization invariant** if its training mean squared error is **guaranteed** to be the same, with or without standardization. Which of the models below are standardization invariant? **Select all** that apply.

- ☐ `LinearRegression()`
- ☐ `Ridge(alpha=10)`
- ☐ `Lasso(alpha=1)`
- ☐ `Lasso(alpha=10000)`
- ☐ `DecisionTreeRegressor(max_depth=3)`
- ☐ `KNeighborsRegressor(n_neighbors=5)`
- ☐ `KNeighborsRegressor(n_neighbors=8)`

b) (2 pts) Consider the two model instances below.

```
once = make_pipeline(StandardScaler(), ModelClass())
twice = make_pipeline(StandardScaler(), StandardScaler(), ModelClass())
```

True or False: As long as `ModelClass()` is a valid regression model in `sklearn` that behaves deterministically*, `once` and `twice` are **guaranteed** to have the same training mean squared error.

◯ True ◯ False

**By this, we mean that the model makes the same predictions every time it is fit on the same training set. Technically, not all models behave this way.*

uniqname: _____

Question 3 (6 pts)

Suppose $A \in \mathbb{R}^{n \times d}$ is a matrix, $\vec{b} \in \mathbb{R}^n$ is a vector, θ is a **negative** number, and that $\vec{v}^*$ is **a** vector that minimizes $\|\vec{b} - A\vec{v}\|^2$. In other words:

$$\vec{v}^* = \underset{\vec{v}}{\text{argmin}} \|\vec{b} - A\vec{v}\|$$

Furthermore, suppose that one of the columns in A is $\vec{\theta} = \begin{bmatrix} \theta \\ \theta \\ \vdots \\ \theta \end{bmatrix}$.

a) (3 pts) What is the value of $\vec{\theta}^T(\vec{b} - A\vec{v}^*)$?

◯ 0 ◯ $\vec{0}$ ◯ 1 ◯ $\vec{1}$ ◯ θ ◯ $\vec{\theta}$ ◯ None of these

b) (3 pts) Select the true statement below.

*Hint: Remember that θ is a **negative** number, n is the number of rows in A, and I is the identity matrix, a square matrix in which the diagonal entries are 1 and all non-diagonal entries are 0.*

◯ If $A^TA - n\theta I$ is invertible, then the value of $\vec{v}^*$ is unique.

◯ If $A^TA + n\theta I$ is invertible, then the value of $\vec{v}^*$ is unique.

◯ If $A^TA + n\theta I$ is **not** invertible, then the value of $\vec{v}^*$ is **not** unique.

◯ If the value of $\vec{v}^*$ is unique, then $A^TA + n\theta I$ is invertible.

◯ If the value of $\vec{v}^*$ is unique, then $A^TA + n\theta I$ is **not** invertible.

Question 4 (14 pts)

Suppose we'd like to build a regression model to predict the width of a fish, given its height, weight, and species. Consider the following two possible approaches to building linear regression models:

- **Approach 1**: One hot encode species using `OneHotEncoder(drop="first")`, use height and weight as-is, and fit a linear regression model.
- **Approach 2**: Fit 7 separate sub-models, one for each unique value of species, where each sub-model is a linear regression model that uses height and weight only.
  - For instance, one of the 7 sub-models would be for the Perch species; we'd query the training data to keep only the rows corresponding to the species Perch, then fit the Perch sub-model on just this subset of the training data.
  - To predict the width of a new fish, we use the sub-model corresponding to the new fish's species.

  Assume that in **both** approaches, the linear regression models being considered include **intercept terms**.

a) (2 pts) Suppose the mean squared error on the training set for approaches 1 and 2 are MSE_1 and MSE_2, respectively. Fill in the ???:

$$\text{MSE}_1 \quad \boxed{???} \quad \text{MSE}_2$$

◯ $\geq$ ◯ $>$ ◯ $=$ ◯ $<$ ◯ $\leq$ ◯ Impossible to tell

b) (6 pts) Suppose we visualize both models in 3 dimensions, where one axis represents height, one axis represents weight, and one axis represents predicted width.

Fill in the blanks: In the 3 dimensional space described above,
the fit model in **approach 1** looks like __(i)__ __(ii)__ __(iii)__,
while the fit model in **approach 2** looks like __(iv)__ __(v)__ __(vi)__.

(i) ◯ 2 ◯ 3 ◯ 4 ◯ 5 ◯ 6 ◯ 7 ◯ 8
(ii) ◯ identical ◯ possibly non-parallel ◯ orthogonal ◯ parallel
(iii) ◯ lines ◯ planes ◯ points ◯ spheres ◯ vectors

(iv) ◯ 2 ◯ 3 ◯ 4 ◯ 5 ◯ 6 ◯ 7 ◯ 8
(v) ◯ identical ◯ possibly non-parallel ◯ orthogonal ◯ parallel
(vi) ◯ lines ◯ planes ◯ points ◯ spheres ◯ vectors

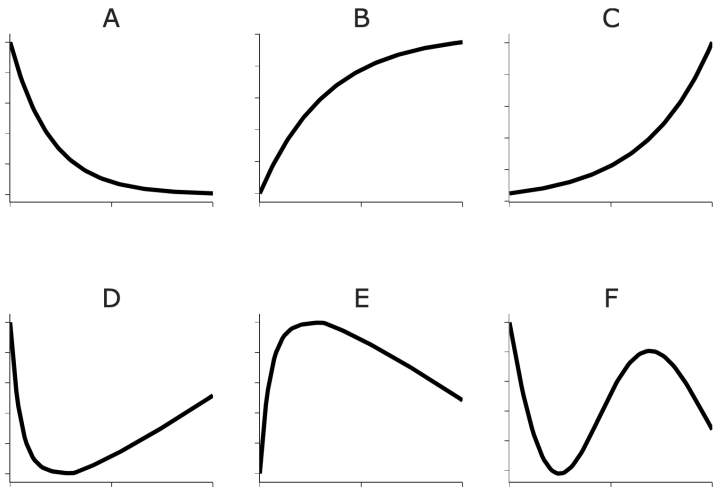
It is possible to construct a design matrix X' such that approach 2 is implemented as a single linear regression model, rather than 7 separate linear regression models.

c) (2 pts) How many columns are in the design matrix, X'?

d) (4 pts) In 1-2 English sentences, describe how to create X'. Then, sketch an example of how X' might look, including at least two example rows, one of which should correspond to a fish of the Perch species with a height of 50 and weight of 25. Feel free to use ellipses, ..., in your answer.

Question 5 (14 pts)

Suppose we'd like to build a regression model to predict the width of a fish, given various features. Consider the six line graphs shown below.



In each part, select the graph that **best** represents the relationship between model performance (drawn on the y-axis) and the provided hyperparameter (drawn on the x-axis).

First, suppose we build a **k-nearest neighbors regression** model, as seen in Homework 8.

a) (2 pts) Which graph best represents **training mean squared error** (y-axis) vs. k (x-axis)?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

b) (2 pts) Which graph best represents **test set mean squared error** vs. k?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

c) (2 pts) Which graph best represents **model variance** vs. k?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

Now, suppose we build a **linear regression model with degree-d polynomial features**.

d) (2 pts) Which graph best represents **training mean squared error** vs. d?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

e) (2 pts) Which graph best represents **test set** R^2 vs. d? *Hint: Recall from Homework 8,* $0 \leq R^2 \leq 1$*, where larger values of* R^2 *indicate better predictions.*

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

f) (2 pts) Which graph best represents **model variance** vs. d?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

Finally, suppose we build a **ridge regression model with hyperparameter λ**.

g) (2 pts) Which graph best represents **model variance** vs. λ?

◯ A ◯ B ◯ C ◯ D ◯ E ◯ F

uniqname: _____

Question 6 (9 pts)

Suppose we'd like to build a L_1-regularized (LASSO) regression model to predict the width of a fish, given its height and weight. To choose a value of λ (the regularization hyperparameter) from the list $[0.01, 0.1, 1, 10]$, we perform k-fold cross-validation with $k = 4$.

a) (3 pts) Consider the following optimal parameter vectors, each of which came from minimizing L_1-regularized empirical risk using a different value of λ.
(In each parameter vector, the 0th component represents the intercept term, the 1st component represents the coefficient on height, and the 2nd component represents the coefficient on weight.)

$$\vec{w}_a^* = \begin{bmatrix} 2.6999 \\ 0.0105 \\ 0.0041 \end{bmatrix} \qquad \vec{w}_b^* = \begin{bmatrix} 3.0674 \\ 0.0000 \\ 0.0034 \end{bmatrix} \qquad \vec{w}_c^* = \begin{bmatrix} 2.0728 \\ 0.1236 \\ 0.0031 \end{bmatrix}$$

(i) Which optimal parameter vector resulted from choosing $\lambda = 0.01$?
◯ $\vec{w}_a^*$ ◯ $\vec{w}_b^*$ ◯ $\vec{w}_c^*$

(ii) Which optimal parameter vector resulted from choosing $\lambda = 0.1$?
◯ $\vec{w}_a^*$ ◯ $\vec{w}_b^*$ ◯ $\vec{w}_c^*$

(iii) Which optimal parameter vector resulted from choosing $\lambda = 1$?
◯ $\vec{w}_a^*$ ◯ $\vec{w}_b^*$ ◯ $\vec{w}_c^*$

b) (3 pts) Suppose, just for this part, that:

- Our entire dataset has N rows.
- To split the dataset of N rows into training and test sets, we use `train_test_split` with `test_size=0.2`.

While performing 4-fold cross-validation, each time a model is trained, 90 points are used to train the model. What is the value of N?

◯ 100 ◯ 108 ◯ 120 ◯ 150 ◯ 180 ◯ None of these

c) (3 pts) Average validation mean squared errors are shown in the table below.

	Fold 1	Fold 2	Fold 3	Fold 4
$\lambda = 0.01$	15	9	12	12
$\lambda = 0.1$	12	18	6	12
$\lambda = 1$	3	12	15	m
$\lambda = 10$	18	6	3	9

Given that $\lambda = 1$ is the hyperparameter with the best cross-validation performance above, provide the best possible upper-bound for a value of m. That is, find M such that, as long as $m < M$, $\lambda = 1$ is the best choice of λ. Give your answer as a **constant with no variables**.

Question 7 (12 pts)

Let $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Consider the function $f(\vec{x}) = (x_1^2 + x_2 - 3)^2 + (x_1 + x_2^2 - 4)^2$.

a) (6 pts) $\nabla f(\vec{x})$, the gradient of f, can be written in the form $\boxed{\nabla f(\vec{x}) = A\vec{g}}$, where $A \in \mathbb{R}^{2\times 2}$ and $\vec{g} \in \mathbb{R}^2$. Fill in the blanks to complete the definitions of A and $\vec{g}$. All blanks should be filled with expressions involving x_1, x_2, and/or constants.

$$A = \begin{bmatrix} 2x_1 & \text{--(i)--} \\ 1 & \text{--(ii)--} \end{bmatrix} \qquad \vec{g} = 2\begin{bmatrix} \text{--(iii)--} \\ \text{--(iv)--} \end{bmatrix}$$

(i) :

(iii):

(ii):

(iv):

b) (4 pts) We'd like to use gradient descent to minimize f. We choose an initial guess of $\vec{x}^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and learning rate/step size β. Given that $A = \begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix}$ and $\vec{g} = \begin{bmatrix} -4 \\ -6 \end{bmatrix}$ when evaluated on $\vec{x}^{(0)}$, perform one iteration of gradient descent. In other words, what is $\vec{x}^{(1)}$?

Show your work, and put a $\boxed{\text{box}}$ around your final answer, which should be a vector with two components, both of which are expressions involving β and/or constants, but no other variables (i.e. x should not appear in your answer).

c) (2 pts) Suppose $Q(\vec{x})$ is a convex function. True or False: Given that gradient descent converges to the global minimum of Q in T iterations, it is guaranteed that:

$$\|\vec{x}^{(1)} - \vec{x}^{(0)}\| > \|\vec{x}^{(t+1)} - \vec{x}^{(t)}\|, \quad \text{for } t = 1, 2, ..., T-1$$

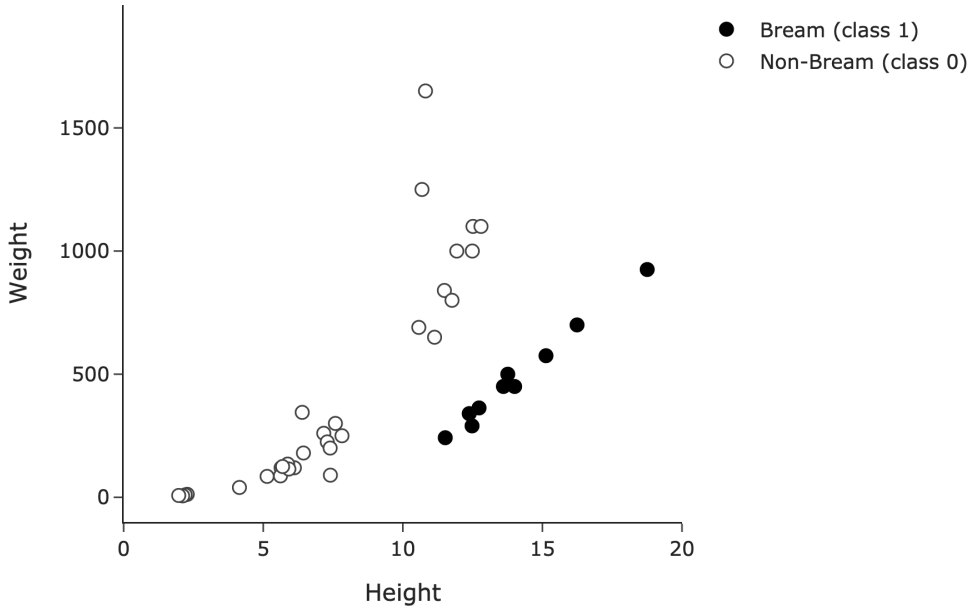
◯ True ◯ False

uniqname: _____

Question 8 (20 pts)

Suppose we build a classifier to predict whether a fish is of species Bream (class 1) or not (class 0), given its height and/or weight.

Consider the training set of $n = 40$ points below. Note that the 10 points in black correspond to class 1, and the 30 outlined points correspond to class 0. Assume there are no black points hidden under outlined points and vice versa.



a) (1 pt) Suppose we use height **only** to predict species. In the 1-dimensional feature space, is the training set linearly separable?

◯ Yes ◯ No

b) (1 pt) Suppose we use **both** height and weight to predict species. In the 2-dimensional feature space, is the training set linearly separable?

◯ Yes ◯ No

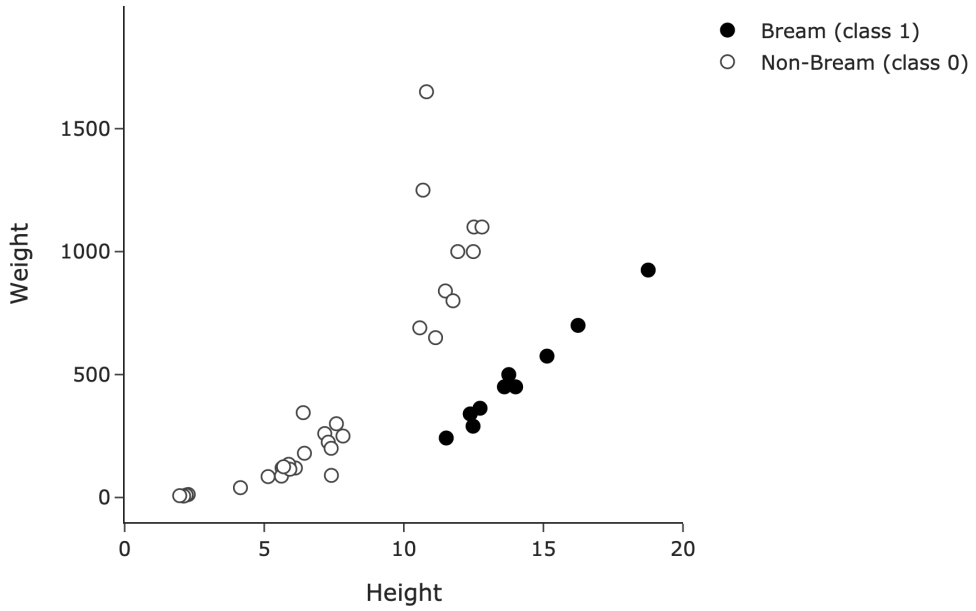
c) (3 pts) Suppose we use **both** height and weight to predict species. What is **minimum** depth d required for a decision tree classifier to achieve a training accuracy of 100%?

◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 ◯ 10 ◯ 40

d) (2 pts) Suppose we use **both** height and weight to predict species. If we use a k-nearest neighbors classifier with $k = 1$, which class would be predicted for a fish with height 20 and weight 1250?

◯ Class 1 (Bream) ◯ Class 0 (non-Bream)

For your convenience, we show the training set of $n = 40$ points again below, in which 10 points belong to class 1.



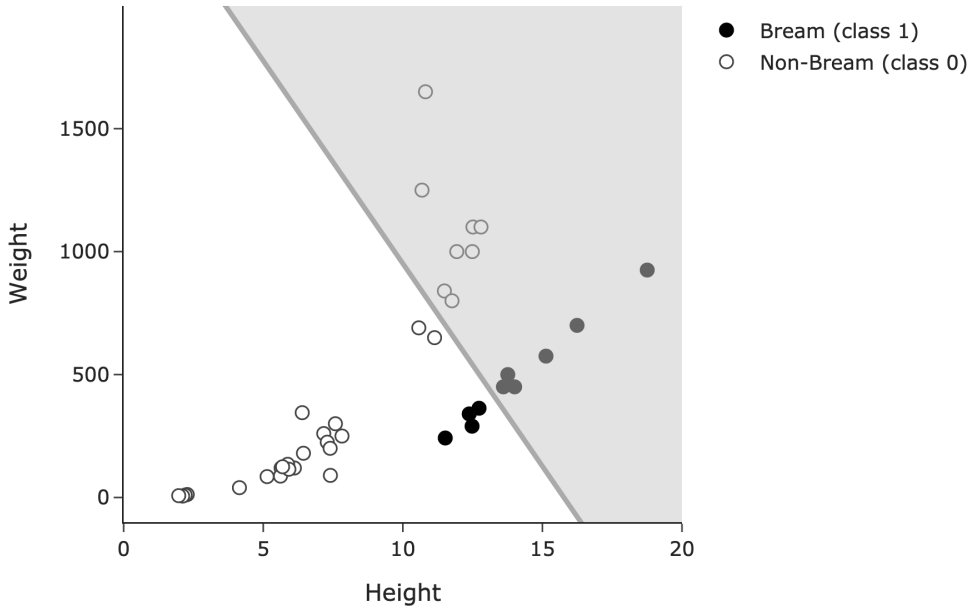
e) (3 pts) Suppose we use height (x_i) **only** to predict species. If we use logistic regression without regularization, which option best describes the fit model, $P(y_i = 1|x_i)$?

◯ $P(y_i = 1|x_i) = \sigma\left(-15 - \frac{2}{3}x_i\right)$
◯ $P(y_i = 1|x_i) = \sigma\left(-15 + \frac{2}{3}x_i\right)$
◯ $P(y_i = 1|x_i) = \sigma\left(-20 + \frac{2}{3}x_i\right)$
◯ $P(y_i = 1|x_i) = \sigma\left(-15 - \frac{5}{4}x_i\right)$
◯ $P(y_i = 1|x_i) = \sigma\left(-15 + \frac{5}{4}x_i\right)$
◯ $P(y_i = 1|x_i) = \sigma\left(-20 + \frac{5}{4}x_i\right)$

f) (3 pts) Suppose we use height (x_i) **only** to predict species. If we use L_2-regularized logistic regression with a regularization hyperparameter of $\lambda = 10^{10}$, what is the value of w_0^* in the fit model $P(y_i = 1|x_i) = \sigma(w_0^* + w_1^* x_i)$?

◯ 3 ◯ 1/3 ◯ 4 ◯ 1/4
◯ $\log(3)$ ◯ $\log(1/3)$ ◯ $\log(4)$ ◯ $\log(1/4)$

For the rest of this question, suppose we use **both** height and weight to predict species. Suppose we use logistic regression — possibly with regularization — and choose a classification threshold of T, where $0 \leq T \leq 1$. The resulting decision boundary is shown below, along with the same training set as before (which has $n = 40$ points, 10 of which belong to class 1).



In the region above the line, shaded gray, our classifier predicts class 1 (Bream); in the region below the line, our classifier predicts class 0 (non-Bream).

g) (2 pts) Was regularization used when fitting the logistic regression model whose decision boundary is shown above?

◯ Yes, regularization was used ◯ No, regularization was not used

h) (3 pts) What are the precision and recall of the classifier above? Give your answers as simplified fractions. Answers that "swap" precision and recall will not be given any credit.

precision = _____ recall = _____

i) (2 pts) Suppose that each time we accidentally misclassify a fish's species as Bream, we must pay a fine to the local aquarium. Given this fact alone, is it more important for our classifier to have high precision or high recall?

◯ High precision ◯ High recall

Question 9 (11 pts)

Consider a dataset of n non-negative numbers, $x_1 < x_2 < \ldots < x_n$, that are evenly spaced. In other words, $x_{i+1} - x_i$ is some fixed positive constant, for all $i = 1, 2, ..., n-1$. Furthermore, assume that n **is an even integer that is not a multiple of 4**.

We'd like to use k-means clustering to cluster this dataset into $k = 2$ clusters, Cluster 1 and Cluster 2, defined by two centroids, μ_1 and μ_2, respectively.

Suppose we initialize the two centroids at $\mu_1 = x_1$ and $\mu_2 = x_2$.

a) (2 pts) In iteration **1**, **before** updating the locations of the centroids, how many points belong to each cluster? Give your answers in the form of expressions involving n and/or constants.

Cluster 1: [ ] Cluster 2: [ ]

b) (4 pts) In iteration **1**, **after** updating the locations of the centroids, both centroids are located exactly at values in the dataset. Which data point is each centroid now located at? Give your answers in the form of expressions involving n and/or constants, but **not** x. (For example, if you believe μ_2 is now located at x_{n-1}, put $n - 1$ in the second box.)

(i) μ_1 is now located at x_p, where $p =$ [ ]

(ii) μ_2 is now located at x_q, where $q =$ [ ]

c) (3 pts) In iteration **2**, **before** updating the locations of the centroids, how many points belong to each cluster? Give your answers in the form of expressions involving n and/or constants.

Cluster 1: [ ] Cluster 2: [ ]

d) (2 pts) In one English sentence, name **and** describe the objective function that k-means clustering minimizes.

[ ]

Make sure you've written your uniqname in the space provided in the top right corner of every page of this exam!

Congrats on finishing the course — we'll miss you! Feel free to draw us a picture about EECS 398 below :)