Data 100, Final

Summer 2021

Name: _____

Email: _____@berkeley.edu

Student ID: _____

Exam Time: _____

All work on this exam is my own ***(please sign)***: _____

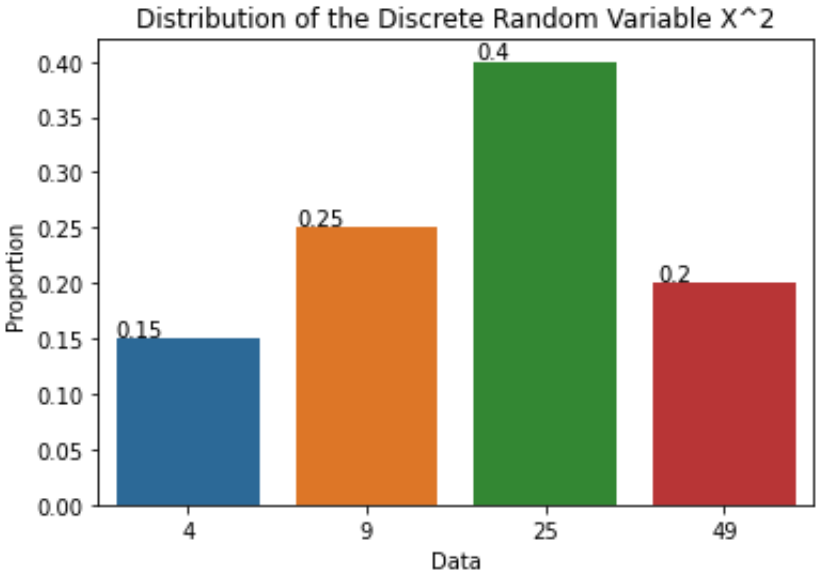
Honor Code [1 Pt]

1. *As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others. I will not communicate with any other individual during the exam, current student or otherwise. All work on this exam is my own.*
   (a) Please confirm your agreement with the above statement by writing your name in the space below.

   _____

Probability Potpourri [8 Pts]

2. Suppose X is a discrete, positively valued random variable. The following graph describes the probability distribution of X^2.



(a) [2 Pts] What is the expected value of X? Round your answer to two decimal places.

Solution: 4.45. First note that the question states that X is positively valued. Therefore, the possible values of X are the positive square root values of the current data points (2, 3, 5, 7), and the probabilities are the same as denoted on the graph (.15, .25, .4, .2). Thus,

$$E(X) = \sum_k kP(X = k) = 2 \ :15 + 3 \ :25 + 5 \ :4 + 7 \ :2 = 4:45$$

(b) [2 Pts] Following your answer to the previous question, what is the variance of X? Round your answer to two decimal places.

Solution: 2.85.

$$\text{Var}(X) = E(X^2) \quad E(X)^2$$

We can calculate $E(X^2)$ using the plot above:

$$E(X^2) = \sum_k k^2P(X = k) = 4 \ :15 + 9 \ :25 + 25 \ :4 + 49 \ :2 = 22:65$$

From the previous part, we have $E(X)^2 = 4:45^2$. Thus, $\text{Var}(X) = 22:65 \quad 4:45^2$ 2:85.

3. Oh no! Our friend Kanu has decided to take the Data 100 final without studying at all. He believes he can pass the course by simply guessing uniformly at random on every question. Assume Kanu needs a **10%** on the final to pass. The test consists of **20 MCQ** questions and **4 FRQ** questions. The grading scheme is as follows:

   - MCQ
     - 5 points are awarded for each correct answer.
     - $\frac{1}{3}$ points are awarded for each incorrect answer.
     - 0 points are awarded for each blank answer.
   - FRQ
     - 10 points are awarded for each correct answer.
     - $\frac{1}{3}$ points are awarded for each incorrect answer.
     - 0 points are awarded for each blank answer.

   There are 140 points available, so Kanu needs at least a 14 to pass.

   (a) [4 Pts] Each MCQ question has **4** possible answers, one of which are correct. Each FRQ question has **10** possible answers, one of which is correct. On average, which of the following test taking strategies will help Kanu pass the class? Select all that apply.

   Guess randomly on all MCQ and FRQ.

   Guess randomly on all MCQ and leave the FRQ blank.

   Guess randomly on all FRQ and leave the MCQ blank.

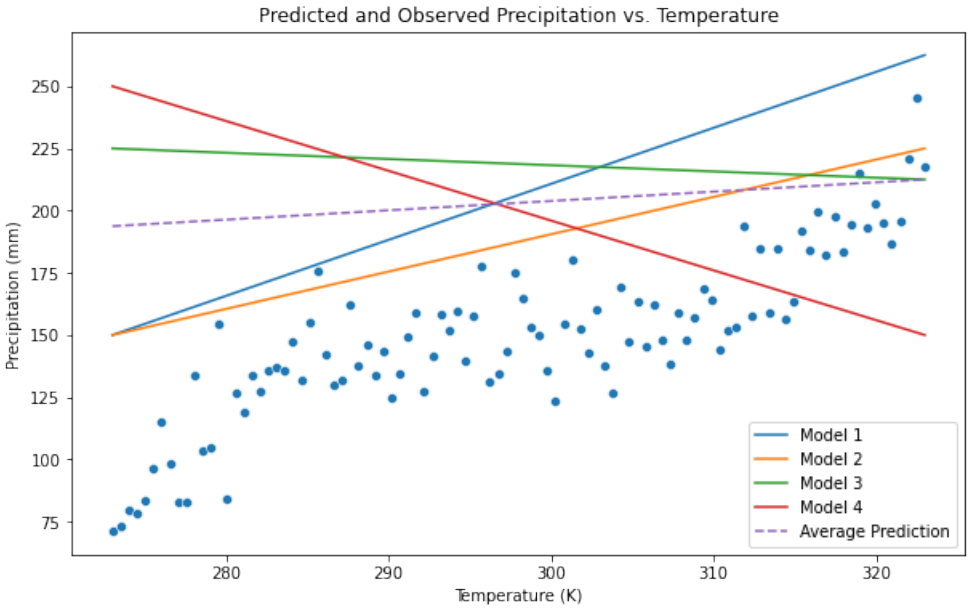
   Guess randomly on all MCQ and $\frac{1}{2}$ of the FRQ. Leave the other $\frac{1}{2}$ of the FRQ blank.

   Guess randomly on $\frac{3}{4}$ of the MCQ and all the FRQ. Leave the other $\frac{1}{4}$ of the MCQ blank.

Extreme Tradeoffs [12 Pts]

4. Mr. Bean wants to model extreme precipitation events, which are historically difficult to predict accurately. To attempt to create the world's best model, he tries training multiple models using bootstrap sampling and regularization.

   (a) [2 Pts] Mr. Bean designs 4 different models by bootstrap sampling 30% of the total training data to train each model on. On the test set, he creates the following plot displaying the temperature feature against the model's predictions and true observed values. The dotted line shows the average prediction across all 4 models. Which of the following does the figure indicate? Select all that apply.



High model variance

Low model variance

High model bias

Low model bias

> **Solution:** This displays high model variance since for any particular point, we have 4 predictions of precipitation that are vastly different.
>
> This displays high model bias since on average, our predictions are far away from the ground truth distribution and we don't have the necessary model complexity to model a non-linear distribution.

   (b) [4 Pts] Mr. Bean decides to diagnose the issue further. He increases the number of trained models to 100 and evaluates the models on the point $(x_i; y_i)$. Using historical data, he assumes that measurement errors follow a normal distribution with mean 0 and standard deviation = 4 mm. Given the below statistics calculated using `pd.describe` on the predictions and loss for these models, estimate the magnitude of the empirical bias.

Round to 3 decimal places.

Hint: Think of how bias is calculated in our bias-variance decomposition and relate the quantities below to the terms in the decomposition.

This question is difficult, so if you are not sure how to start then skip it for now and come back to the question later.

	preds (mm)	MSE
count	100.000000	100.000000
mean	104.130417	101.930101
std	8.255889	112.418423
min	90.819579	0.023824
50%	103.941588	51.771039
max	119.474415	367.888568

Solution: 4.215 mm

In the box below, show how you obtained the value above. Specifically, write down the bias-variance decomposition, substituting in the relevant quantities. No LaTeX is required, you can use plain English.

Solution: Students received credit for writing the bias-variance tradeoff, either mathematically, or in plain English—for example, risk = model variance + (model bias) ^ 2 + observation variance". Points were also awarded for pinpointing the values of each of the three known quantities. The math is below:

$$\begin{aligned} \mathrm{E}[(y - f(x))^2] &= {}^2 + \mathrm{Var}[f(x)] + (\mathrm{E}[f(x)] - g(x))^2 \\ 101.9301 &= 4^2 + 8.2559^2 + \mathrm{bias}^2 \\ 101.9301 - 4^2 - 8.2559^2 &= \mathrm{bias}^2 \\ |\mathrm{bias}| &= 4.215 \end{aligned}$$

(c) [2 Pts] He decides to change his models to add L2 regularization. What behavior is expected in the training set compared to the unregularized models?

The model bias will decrease.

The model bias will increase.

The model variance will decrease.

The model variance will increase.

The observational variance will decrease.

The observational variance will increase.

Solution: The model bias increases and model variance decreases as per the bias-variance tradeoff when model complexity decreases. The observational variance doesn't change since the dataset remains the same.

(d) [2 Pts] Regardless of your answer to the previous question, assume that after implementing regularization, the model bias is too high. Which of these solutions helps reduce the model bias?

Add an intercept term.

Use a decision tree with the same features.

Increase the regularization hyperparameter.

Decrease the regularization hyperparameter.

Solution: Adding an intercept term adds model complexity, which decreases model bias.

A decision tree has lower bias than linear regression, even with the same features, as a decision tree tends to fit training points perfectly.

The regularization hyperparameter controls the model complexity. A larger hyperparameter reduces the model complexity, so reducing the regularization hyperparameter will increase model complexity, which decreases model bias.

(e) [2 Pts] Assume that we fixed the previous issue by changing to a *different* unspecified regression model, and the model bias decreased. Which of the following could have happened as a result?

The model variance increased.

The model variance decreased.

The model variance stayed the same.

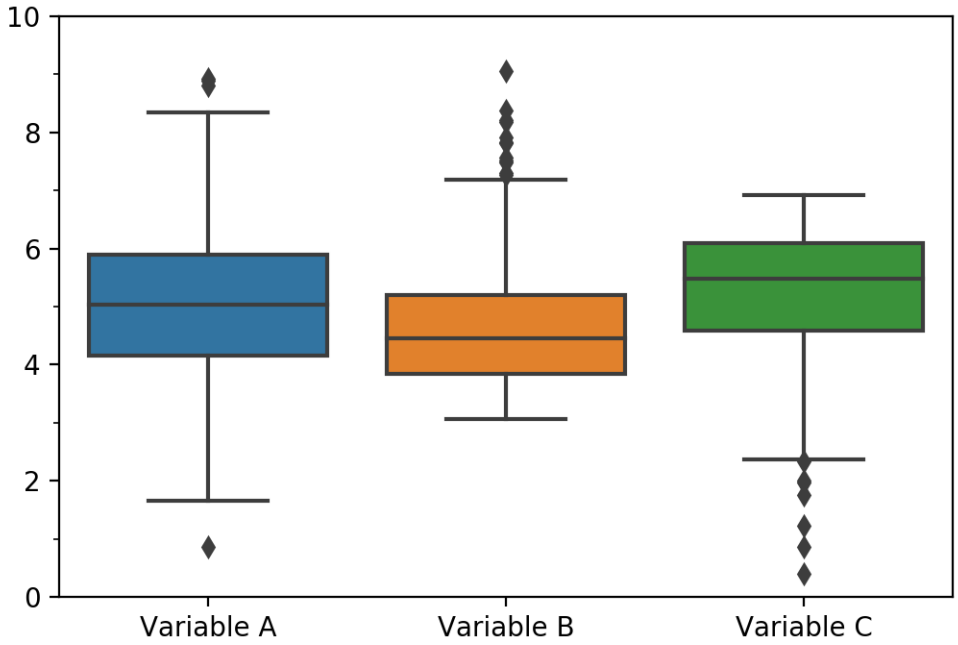
The observational variance decreased.

Solution: Since we change to a different type of model, we might have a completely different (or the same) expected loss in the bias-variance decomposition. Therefore, our variance could have increased, decreased, or stayed the same.

The observational variance does is not affected by our model because the dataset remains the same.

Thinking Inside the Box [6 Pts]

5. Below are boxplots showing the distributions for three different quantitative variables. We will name these variables Variable A, Variable B, and Variable C.



Some of these distributions may be skewed—if a distribution is skewed, we want to apply a transformation to symmetrize it.

The following three parts will ask which transformations may be suitable for symmetrizing each distribution. If no transformation is necessary, select "No transformation necessary."

(a) [1 Pt] Which of the following transformations may symmetrize Variable A?

$\log(x)$ x^2 $p_{\overline{x}}$ x^3 **No transformation necessary.**

(b) [1 Pt] Which of the following transformations may symmetrize Variable B?

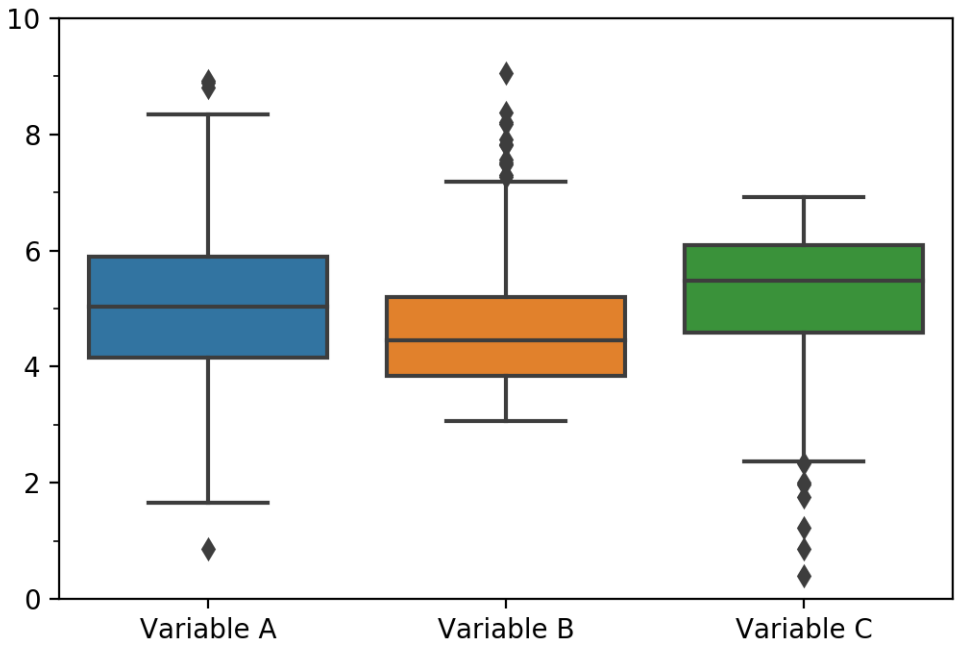
$\log(x)$ x^2 $p_{\overline{x}}$ x^3 No transformation necessary.

(c) [1 Pt] Which of the following transformations may symmetrize Variable C?

$\log(x)$ x^2 $p_{\overline{x}}$ x^3 No transformation necessary.

Solution: Variable A is already symmetric, so it needs no transformation. Variable B is right-skewed, so log and square root transformations may help. Variable C is left-skewed, so power transformations may help.

The boxplots are repeated here for your convenience.



In each of the following parts, you will see a statement about the boxplots above. Determine if each statement is True, False, or Impossible to tell.

(d) [1 Pt] Variable B has the lowest first quartile among all three variables.

True False Impossible to tell

Solution: True. Variable B has the lowest bottom of the box across all three variables.

(e) [1 Pt] Variable A is unimodal.

True False **Impossible to tell**

Solution: Impossible to tell. We know that variable A is symmetric, but the boxplot conceals information about the actual shape of the distribution. It could very well be unimodal, bimodal, or have even more modes.

(f) [1 Pt] Variable C contains zero points greater than 1.5 IQR above its median.

True False **Impossible to tell**

Solution: On this exam, we accepted both "True" and "impossible to tell". This was due to a typo in the question—it really should have read "Variable C contains zero points greater than 1.5 IQR above its **third quartile**." The question as it was written was still answerable, because it is straightforward to see that the top whisker does lie below 1.5 IQR above the median. However, we did not intend to introduce this layer of analysis into the question, so we accepted both "True" and "impossible to tell."

Night Owl or Early Bird? [13 Pts]

6. Suriya and Meghna are Data 100 students, and they have a prediction task where we wish to predict whether people are night owls or early birds using their favorite color. They're given a shortened training set with 5 data points where X = ['blue'; 'green'; 'pink'; 'purple'; 'red'] and they wish to predict $y = [0; 1; 1; 1; 0]$.

   (a) [1 Pt] What type of variables does X contain?

   Quantitative continuous

   Quantitative discrete

   Qualitative discrete

   Qualitative nominal

   Qualitative ordinal

   Solution: X contains colors, which are not quantitative. They don't have ordering, so the correct response is qualitative nominal.

   (b) [1 Pt] They decide to one-hot encode the data in X into a design matrix X^0, with the categories being ordered alphabetically from left to right. How many values in X^0 are zero?

   Solution: The matrix will contain a 1 value for each matching color in a 5x5 matrix. Since there are 5 colors, there will be 25 - 5 = 20 zero values.

   Suriya and Meghna decide to use logistic regression with no intercept term, where the predicted probabilities are rounded to the nearest whole number. Suriya decides to try L0 regularization for their logistic regression model. Unlike L1 and L2 regularization, L0 regularization does not add a term to the loss function. Instead, it specifies a constraint that only some k elements in our parameter for the model can be non-zero.

   Hint: $(0) = :5$

   (c) [2 Pts] Suppose he applies L0 regularization where $k = 5$, and finds the optimal for X^0 and y using logistic regression. How many points does he misclassify?

   Solution: Since this means that all of the values can be nonzero, this is effectively logistic regression without regularization. As a result, the first and last will tend towards 1, and the middle three will tend towards $+1$, resulting in 0 misclassified points on the fitted data.

   (d) [2 Pts] Suppose he applies L0 regularization where $k = 1$ and find the optimal for X^0 and y using logistic regression. How many points does he misclassify?

Solution: Given the hint, we know that $(0) = :5$ and based on our threshold, that would be a positive (1) prediction. Therefore, even if our values are all 0, we only misclassify the two negative predictions (0s). Since we have one non-zero value, we can set the first $_1$ to any number less than 0, which yields a predicted probability of less than 0.5. We will now only misclassify the other negative prediction.

Therefore, the answer is 1.

Since linear regression using mean square error is easier to solve than logistic regression, Meghna tries to use that instead to create a quick model.

(e) [1 Pt] Is $X^{0T} X^0$ invertible? **Yes** No

Solution: Yes, it is full rank. In fact, it's orthonormal!

(f) [2 Pts] What is the optimal value of if we use mean square error as the loss function? Your answer should be a sequence of 5 elements, e.g. $[1; 2; 3; 4; 5]$.

Solution: The design matrix X^0 is the identity matrix. Therefore, our is simply the y values exactly: $[0; 1; 1; 1; 0]$.

(g) [2 Pts] For the optimal value of , what is the mean squared error on X^0?

Solution: With the optimal above, we can predict every point perfectly, so the mean squared error is 0.

Since they can't possibly train a great model with 5 data points, they seek out the full training set and discover that it has 1,000,000 training points with the same colors from before. They decide to use logistic regression, without regularization and without an intercept term, for the remaining parts of the question.

(h) [1 Pt] What is the column rank of the new one-hot encoded dataset?

Solution: Since there are 5 OHE columns (which are linearly independent since they are orthogonal), the rank is 5.

(i) [1 Pt] Meghna wants to use a gradient method to discover the optimal . Which of the following options is the best suited to this training set and problem?

Stochastic gradient descent (batch size = 1)

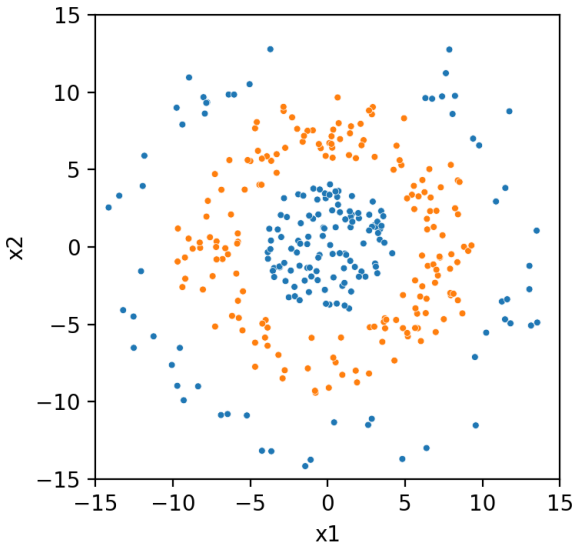
Gradient descent, on the complete dataset

Stochastic gradient descent (batch size = 32)

Solution: Our training set is too large for batch gradient descent on the entire dataset and using a batch size of 1 is likely to lead to fluctuation and oscillation. The middle ground that is appropriate is SGD with a batch size of 32.

Donut Decisions [9 Pts]

7. Below is a dataset from which we want to create a classifier. We have two features, x_1 and x_2, and two classes. Assume the orange points are in class 1 and the blue points are in class 0. The points displayed here are training data.



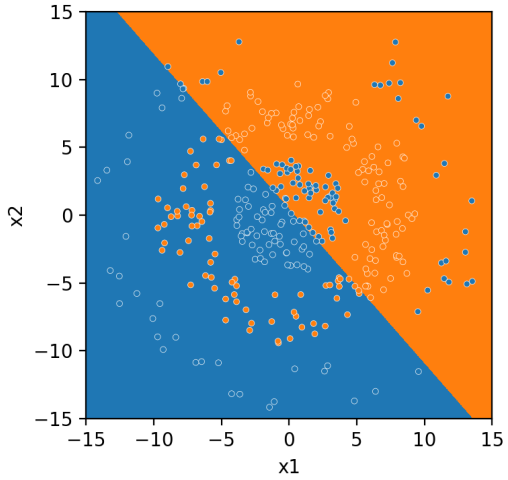
(a) [1 Pt] Is this dataset linearly separable?

Yes **No**

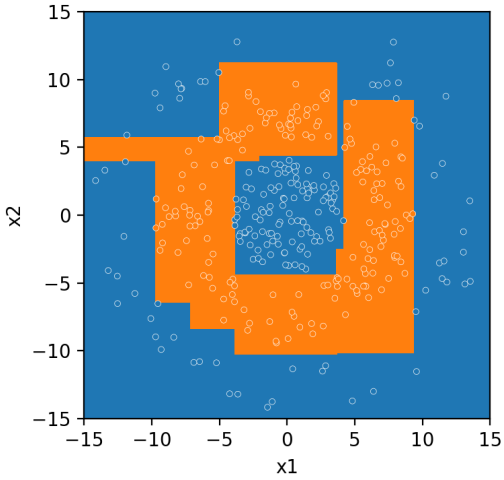
Solution: No line can be drawn that perfectly separates the two classes, so this dataset is not linearly separable.

Below are 4 different possible decision boundaries (a.k.a. classifiers) we can generate. The orange regions are areas where new points would be classified as class 1, and the blue regions are areas where new points would be classified as class 0.

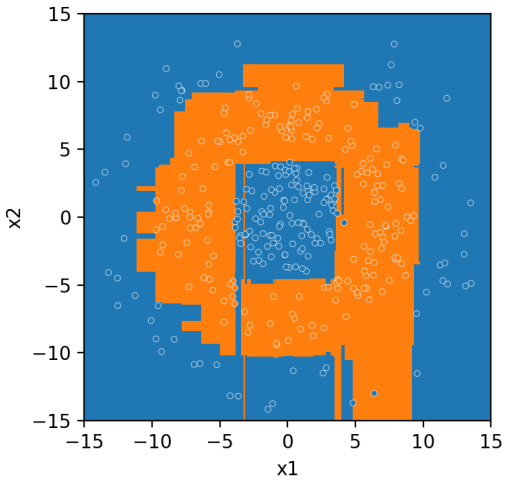
A.



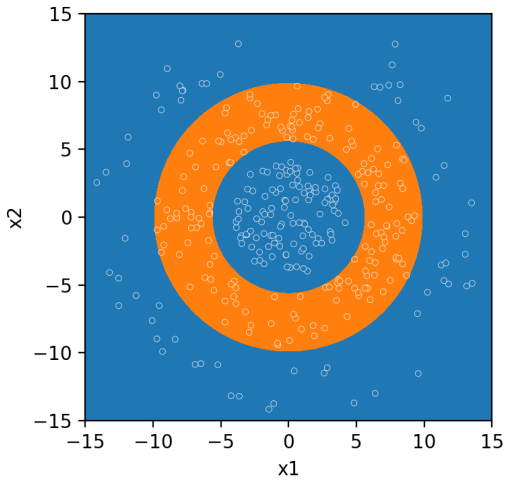
B.



C.



D.



(b) [2 Pts] Which of the above classifiers (A, B, C, D) has perfect accuracy on the training set? Select all that apply.

Note: Do not try to distinguish borderline points—you may assume points right on the boundary are classified correctly.

A **B** C **D**

Solution: Classifier A clearly does not predict every point correctly. Classifier C has 2 misclassified points, both blue points in orange regions. One is at about $(7; 13)$, and the other is at around $(5; 0)$. Classifiers B and D have no misclassified points.

The following parts will ask you about which model(s) could generate each of the boundaries. For each part, *assume x_1 and x_2 are the only features in our model.*

(c) [1 Pt] Which of the following models could have generated boundary A?

Logistic Regression Decision Tree Random Forest None of the above

Solution: Boundary A could only have been generated by logistic regression, because decision trees (and therefore random forests) only allow for axis-aligned splits.

(d) [1 Pt] Which of the following models could have generated boundary B?

Logistic Regression **Decision Tree** **Random Forest** None of the above

Solution: Boundary B only contains axis-aligned splits, so it could easily have been made by a decision tree, and random forests can create any boundary a decision tree can. Logistic regression could NOT have made this boundary, because logistic regression only returns a linear boundary. This boundary, while it consists of lines, is piecewise linear, not strictly linear.

(e) [1 Pt] Which of the following models could have generated boundary C?

Logistic Regression Decision Tree **Random Forest** None of the above

Solution: A decision tree could not have made Boundary C, because it classifies some points incorrectly. A random forest could have made this boundary due to the piecewise linear splits.

(f) [1 Pt] Which of the following models could have generated boundary D?

Logistic Regression Decision Tree Random Forest **None of the above**

Solution: None of logistic regression, decision trees, or random forests can create curved decision boundaries. With just these two features x_1 and x_2, none of the models could create this boundary.

(g) [2 Pts] Suppose we add a new feature x_3, which is some function of x_1 and x_2. **Now, assume x_3 is a feature in our model.** Which of the following models could have generated boundary D?

Logistic Regression **Decision Tree** **Random Forest** None of the above

Solution: If we select an appropriate function, we can make this dataset linearly separable in 3D space, allowing for all 3 models to classify all training points correctly. In this example, any function that is small for points either too close or too far from the origin, and large for points a certain distance from the origin (or vice versa), would do. The function used to generate this figure was

$$e^{\left(\sqrt{x_1^2+x_2^2} - 7.75\right)^2}$$

although any function meeting the properties described above would do. Note that a student would not have needed to come up with a specific function to correctly the answer the question, only recognizing that such a function exists would have sufficed.

When we add a new feature x_3, boundaries that are linear or piecewise linear in the 3D space can look curved in the original 2D space, leading to the boundary seen in figure D.

Feature Engineering [8 Pts]

8. The following dataset contains information about passengers on the Titanic. There are 20 rows in this dataset, and you may assume there are no missing or null values in the dataset. The first 5 rows are shown below.

	sex	age	fare	class	embark_town
0	male	22.0	7.2500	Third	Southampton
1	female	38.0	71.2833	First	Cherbourg
2	female	26.0	7.9250	Third	Southampton
3	female	35.0	53.1000	First	Southampton
4	male	35.0	8.0500	Third	Southampton

A brief description of the columns:

- `age` and `fare` are strictly positive
- `sex` takes on values $\{$male, female$\}$
- `class` takes on values $\{$First, Second, Third$\}$
- `embark_town` takes on values $\{$Southampton, Cherbourg, Queenstown, London, Oxford$\}$

(a) [2 Pts] Suppose we one-hot encode the `sex` column to get a design matrix X_1 with 2 columns, `sex_male` and `sex_female`, where values can be 0 or 1 within each column. Note that X_1 does NOT contain an intercept term.

Select all of the following statements that are true about X_1.

- **X_1 has 20 rows**
- **X_1 is full column rank**
- **$X_1^T X_1$ is invertible**
- None of the above

> **Solution:** The first choice is correct since one-hot encoding does not change the number of rows of the matrix, only the number of columns.
>
> The columns are linearly independent, so choices 2 and 3 are also correct.

(b) [2 Pts] Suppose we one-hot encode the `sex` and `embark_town` column and include an intercept term in the model. This results in a design matrix X_2 with 8 columns.

Select all of the following statements that are true about X_2.

- **X_2 has 20 rows**
- X_2 is full column rank

$X_2^T X_2$ is invertible

None of the above

Solution: The first choice is correct since one-hot encoding does not change the number of rows of the matrix, only the number of columns.

The sum of all the one-hot encoded columns resulting from one categorical feature (e.g. the sum of the `sex_male` and `sex_female` columns) is a column of all 1's, which the design matrix already contains due to the bias term. Therefore, X_2 is not full column rank. This makes choices 2 and 3 incorrect.

(c) [2 Pts] **(Hard)** Suppose we one-hot encode the `sex` and `embark_town` column and do **NOT** include an intercept term in the model. This results in a design matrix X_3 with 7 columns.

Select all of the following statements that are true about X_3.

X_3 **has 20 rows**

X_3 is full column rank

$X_3^T X_3$ is invertible

None of the above

Solution: The first choice is correct since one-hot encoding does not change the number of rows of the matrix, only the number of columns.

Choices 2 and 3 are wrong because there is linear dependence in the columns of X_3. Let town_i be the i^{th} column created by one-hot encoding the `embark_town` column. Then we can write

$$\text{sex\_male} = \left(\sum_{i=1}^{7} \text{town}_i\right) - \text{sex\_female}$$

(d) [2 Pts] Suppose we one-hot encode all the categorical columns (`sex`, `class`, and `embark_town`) and compute the following nonlinear transformations for the quantitative columns `age` and `fare`:

- x^2
- x^3
- $\log(x)$
- $\sin(x)$
- $\cos(x)$

Additionally, we include an intercept term in the model. This results in a design matrix X_4.

We cannot use the normal equations to minimize the MSE loss and solve for $\hat{\theta}$ because X_4 is not full rank. Identify one reason why X_4 is not full rank.

Solution: Acceptable answers:

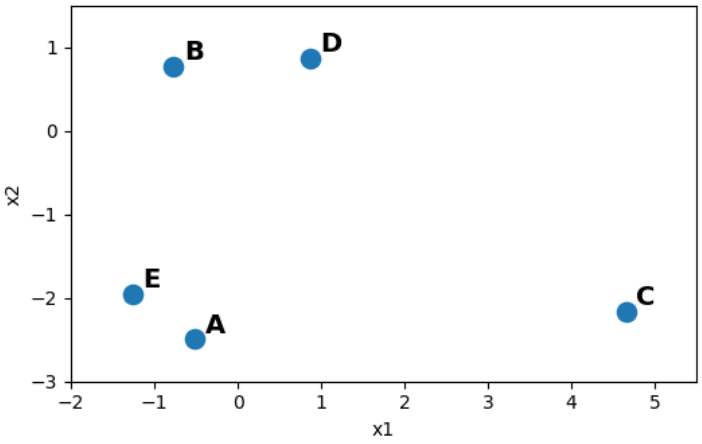
- Linear dependence due to one-hot encoded columns
- More features than data points (20 data points, 21 features)

Agg(ravating) Clustering [7 Pts]

9. Below is a dataset of 5 points, which we want to group into $k = 3$ clusters.

   We will use agglomerative clustering. Our criterion for the distance between clusters is the maximum distance between points from each cluster, as used in lecture.

   Below is a scatterplot of the data. Note that both axes have the same scale. Although relative distances should be discernible from the plot, we have provided the Euclidean distance matrix as well. For example, the distance between points **A** and **B** is 3.3.



	A	B	C	D	E
A	0.0	3.3	5.2	3.6	0.9
B	3.3	0.0	6.2	1.6	2.8
C	5.2	6.2	0.0	4.9	5.9
D	3.6	1.6	4.9	0.0	3.5
E	0.9	2.8	5.9	3.5	0.0

In this question, we will walk through each step of the agglomerative clustering algorithm. When two clusters are merged together, the name of the new cluster should be the lowest letter in the cluster. For example, if we merge clusters **Y** and **Z** together, the new cluster should be called **Y**.

(a) [2 Pts] Which two clusters will be merged together first? Your answer should be two letters in alphabetical order. For example, a possible answer would read **YZ**.

Solution: Clusters A and E are closest together, so the answer should read **AE**.

(b) [2 Pts] Which two clusters will be merged together next? Your answer should be two letters in alphabetical order.

Solution: Cluster B and Cluster D are closest together so the answer should read **BD**.

(c) [1 Pt] Have we completed the algorithm?

Yes No

Solution: After the first two steps, our dataset is now grouped into 3 clusters, so the algorithm has finished.

(d) [2 Pts] Now, we want to cluster this dataset with spectral clustering instead. Remember that when presented with Euclidean point data, we need to construct a graph. In this graph, each vertex corresponds to a point, and the weight of the edge between any two vertices is some function of the Euclidean distance between those points.

In this new graph, which edge will have the smallest weight? Your answer should be two letters in alphabetical order.

_____

Solution: Remember that larger distances between points are represented by smaller edge weights. Points B and C are farthest apart, so the answer should be BC.

Clustering Facts [7 Pts]

10. Fill in the blank for the following statements about clustering algorithms.

(a) [1 Pt] When clustering, the algorithm _____ sees data labels.

always sometimes never

(b) [1 Pt] A _____ is what we use to represent data when an individual is not represented by its numerical features, but by its relationship to other individuals.

graph design matrix relational database

(c) [1 Pt] K-means clustering will _____ find the optimal clustering in terms of inertia.

always sometimes never

(d) [1 Pt] When choosing k for k-means clustering, we want to pick a k with a _____ average silhouette score.

smaller larger

(e) [1 Pt] In spectral clustering, the number of clusters is _____.

predetermined returned by the algorithm

(f) [1 Pt] As part of the spectral clustering algorithm, we find the eigenvalues and eigenvectors of the _____ matrix.

distance adjacency Laplacian

(g) [1 Pt] After calculating the spectral coordinates of each vertex, we use _____ to determine the clusters.

agglomerative clustering k-means clustering

In nite Descent [10 Pts]

11. Curious George decides to use gradient descent to minimize his loss function with respect to a few times, each time using a different set of hyperparameters. Unfortunately, he refreshed his Jupyter Notebook too early and forgot to save his learning rates and initializations. However, he wrote down his loss function, and plotted it with respect to (displayed below). In the following parts, we will investigate the behavior of his chosen hyperparameters. Assume the derivative at 0 with respect to is 0.

$$L(\ ) = \begin{cases} 2\ ^2 + \frac{3}{16} & \leq 0 \\ \ -\frac{1}{2} + \frac{3}{16} & > 0 \end{cases}$$

(a) [1 Pt] George remembers his rst set of hyperparameters! He used a learning rate of 0.1 and initialized $^{(0)}$ to 1. After the rst iteration of gradient descent, what is the new value of ? In other words, what is $^{(1)}$?

Solution: $^{(1)} = {}^{(0)} \quad \frac{d}{d}L(\ ) = \quad 1 \quad (0{:}1)(\ \ 4) = \quad 0{:}6$

(b) [1 Pt] After the second iteration of gradient descent, what is our value of? In other words, what is $^{(2)}$?

Solution: $^{(2)} = {}^{(1)} \quad \frac{d}{d}L(\ ) = \quad 0{:}6 \quad (0{:}1)(\ \ 2{:}4) = \quad 0{:}36$

(c) [2 Pts] For his second set of hyperparameters, George remembers he used a learning rate of $= 0{:}5$, but he cannot remember the initialization of (denoted as $^{(0)}$), where $^{(0)} > 0$. However, he notices after many updates, keeps oscillating between the two unique values $^{(0)}$, the initial value, and $^{(1)}$, the value after one gradient update.

Which of the following could cause gradient descent to oscillate between two unique values in this example?

The sign of the gradient oscillates.

The sign of the gradient does not oscillate.

The magnitude of the gradient oscillates.

The magnitude of the gradient does not oscillate.

The learning rate is inappropriate.

Solution: The sign of the gradient must oscillate, because if the sign of the gradient was constant instead, the values would only move in one direction

If the magnitude of the gradient does not change, but the sign does, it is possible for gradient descent to oscillate back and forth between two values, assuming a constant learning rate.

The learning rate can affect this behavior, because a different learning rate might ensure that the magnitude of the gradient does in fact change. Another possible x could be to use a decaying learning rate instead.

(d) [2 Pts] For his third set of hyperparameters, suppose George used a xed learning rate with gradient descent, and he initialized at 4. Assume convergence to the global minimum took 12 steps—that is, $^{(12)} = 0$. Given this information, what is a xed learning rate that he used? Round to 3 decimal places, or write your answer as a fraction - e.g. "10=9".

Solution: Given $^{(0)} = 4$, where our derivative is a constant $\frac{1}{2}$, the iterative gradient descent process collapses as follows.

$$^{(1)} = {}^{(0)} \quad \frac{}{2}$$

$$^{(2)} = {}^{(1)} \quad \frac{}{2}$$

:::

Note that in general $^{(i)} = {}^{(0)} \quad \frac{i}{2}$, which means that we can plug in = 12 as follows:

$$0 = 4 \quad \frac{12}{2}$$

$$4 = 6$$

$$= \frac{2}{3}$$

After these experiments, George decides to upgrade his gradient descent techniques to the next level.

(e) [2 Pts] George doesn't want to use a fixed learning rate for gradient descent anymore. Assuming t represents the timestep corresponding to each gradient step ($t = 1$ corresponding to the first gradient step and $t = 2$ corresponding to the second, which of the following function(s) are reasonable to describe the learning rate $R(t)$ such that gradient descent will typically converge?

Hint: Think about how the learning rate should behave as we keep iterating.

$R(t) = 2t$

$R(t) = -2t$

$R(t) = \frac{1}{2t}$

$R(t) = e^{2t}$

$R(t) = e^{-2t}$

Solution: The first and fourth option monotonically increase, which make for increasing learning rates. This is likely to fluctuate or oscillate.

The second option is always negative, and learning rates need to be positive (otherwise we're going the direction of the gradient—gradient ascent)!

The remaining options are all decreasing, positive functions for $t > 0$.

(f) [2 Pts] George modifies his loss function and learning rate, while using stochastic gradient descent. He notices that gradient descent converges to a real number every time, but depending on his random initialization, the resulting θ changes, with different corresponding values of the loss function. Which of the following could cause this behavior?

Hint: Recall that gradient descent "converges" when the model weights do not change from one iteration to the next.

The loss function is convex.

The batch size is too small.

The batch size is too large.

The loss function is not convex.

Solution: If the loss function is convex, there wouldn't be local minima such that the model weights did not change (i.e. where the gradient or derivative was zero). The loss function being non-convex could cause this behaviour.

The batch size being too small (for instance, 1) can cause behaviour where within our batch, we compute a gradient of 0 for that data point and "converge".

Decisions, Decisions [13 Pts]

12. Unfortunately, Clippy and some of his classmates did not reach the required accuracy on their spam-ham email model from Homework 11. To get partial credit, they are required to collect their own data and build their own classi ers. Clippy decides to surf his email inbox for data and comes up with a new training dataframe, called `train`, that consists of 7 rows and 2 features. Similarly, he has a testing dataframe, called `test`, with 3 rows and 2 features.

    - The feature `A` represents the polarity of the email `body`, as determined by VADER
    - The feature `B` represents the polarity of the email `subject`, as determined by VADER

    (a) [2 Pts] Say Clippy wanted to visualize his training data using a 2D scatter plot, with feature A on the x-axis and feature B on the y-axis. He wants to color his data points differently to distinguish between emails labeled as spam or ham. Write a single line of code using Seaborn to return the following visualization.

Solution: sns.scatterplot(data=train, x="A", y="B", hue="Spam")

(b) [2 Pts] What is the minimum number of linear splits that completely separate the two classes?

Solution: 4

(c) [2 Pts] What is the minimum depth of the decision tree that completely separates the two classes? Assume a tree with exactly one split has depth 1.

Solution: 3

(d) [2 Pts] Among all possible decision trees that Clippy can train from his training set, what is the maximum testing accuracy Clippy can achieve from any of these models? Assume the first two emails are truly non-spam (ham) emails, and the third email is spam. Write your answer as a percentage, including the % symbol.

Solution: 100%

(e) [2 Pts] What is the weighted entropy of the child nodes after splitting on the feature B < 1.00? Round your answers to 3 decimal places, and omit the leading 0.

Solution:

$$\frac{4}{7}\left(\frac{3}{4}\log_2\frac{3}{4}+\frac{1}{4}\log_2\frac{1}{4}\right)+\frac{3}{7}\left(\frac{2}{3}\log_2\frac{2}{3}+\frac{1}{3}\log_2\frac{1}{3}\right)=\frac{6}{7}\approx .857$$

(f) [1 Pt] Fill in the blank. Using entropy as an indicator of split effectiveness, the proposed split on $A < 1.00$ is _____ to that of the proposed split on $A > 1.25$.

Better than Worse than Equal Cannot be determined

Solution: Splitting on $A < 1$ results in one node with 3 points in class 0 and 2 points in class 1, and the other node with one point in each class. Splitting on $A > 1.25$ results in one node with 4 points in class 0 and 2 points in class 1, and the other node with just 1 point in class 1. In the latter split, both nodes each have lower entropy than their counterparts in the former split. Even without entropy calculations, it is clear to see that $A > 1.25$ does a better job splitting the data than $A < 1$.

(g) [2 Pts] Clippy realizes the data he's working with is not very representative of spam and ham emails. He is thinking of possible solutions to build a more accurate decision tree which is generalizable to the larger dataset provided in Homework 11. When implemented alone, select all of the following possible solutions.

Increase the number of features in his training dataset.

Increase the size of his test set to get a more accurate testing error.

Collect more emails to use for his training and testing dataset.

Use LASSO regularization.

Solution: The first option was removed from the rubric.

The second option is incorrect, because increasing the size of the test set would do nothing to affect the actual model that is trained, on the training data.

The third option is correct, because more emails could very well lead to a better model. However, beware of sampling—if these new emails come from the same source as the original data, they may not improve generalizability much at all (see, 1936 election case study from Lecture 2).

The fourth option is incorrect because we are building a decision tree, not a regularized linear model.

(R)ussian (O)lympic (C)ommittee [10 Pts]

13. It's almost time for the gold-medal match of the Data Science Olympiad, with the United States set to take on Russian Olympic Committee.

We have collected lots of data from other matches at this tournament, which we will use to create a logistic regression classi er to predict the outcome of the gold-medal match. Consider the training data to be all previous matches in the tournament, and the gold-medal match as a single test point.

For the gold-medal match, if our classi er outputs 1, we will predict Russian Olympic Committee will win the match, and if our classi er outputs 0, we will predict United States wins the match.

However, we do not know what threshold, T, to use for our model.

(a) [1 Pt] True or False? One way to select T is to try different candidate values of T, create a ROC curve for each one, and pick the T with the largest area under the curve.

True False

Solution: A single ROC curve displays the true positive rate and false positive rate for different thresholds, so it does not make sense to draw multiple ROC curves to help pick T. Area under the curve is a function of the model as a whole, not a speci c threshold.

(b) [1 Pt] Suppose we decide that predicting Russian Olympic Committee to win, and being wrong, is a better scenario than predicting the United States to win, and being wrong. Of the following three possible thresholds, which makes the most sense to use for our model?

$T = .3$ $T = .5$ $T = .7$

Solution: The scenario described above says that we consider a false positive to be "less bad" than a false negative. As lower thresholds are more likely to lead to false positives than false negatives, we want to pick a lower threshold.

Below is the output of our model on 10 randomly selected training points. The rst column contains the true Y, and the second column contains our model's estimate of $P(Y = 1)$.

(c) [1 Pt] If we set $T = .5$, what is our model's accuracy on these ten training points?

Solution: $\frac{8}{10} = .8$. With $T = .5$, we predict two points incorrectly—the fifth point and the seventh point.

(d) [2 Pts] If we set $T = .5$, what is our model's precision on these ten training points?

Solution: Precision is equal to the total number of true positives our model predicts, divided by the total number of points predicted as positive. More concisely, precision equals $\frac{TP}{TP+FP}$. With $T = .5$, the number of true positives is 4, the number of false positives is 1, so precision equals $\frac{4}{5} = .8$.

(e) [2 Pts] If we set $T = .5$, what is our model's recall on these ten training points?

Solution: Recall is equal to the total number of true positives our model predicts, divided by the total number of points that are actually as positive. More concisely, recall equals $\frac{TP}{TP+FN}$. The total number of positives is 5, and with $T = .5$, the number of true positives is 4, so recall equals $\frac{4}{5} = .8$.

(f) [3 Pts] Give a value of T that maximizes the accuracy on these 10 training points.

Solution: Any value between the fourth point, .361822, and the fifth point, .488413 would do. Here, the only incorrect prediction would be the seventh data point, so our accuracy would be $\frac{9}{10} = .9$.

Rare Red Rabbits [20 Pts]

14. Kermit the Frog and Miss Piggy join a job as data science consultants at a national reserve where there live rare red and blue rabbits. They are tasked with exploring yearly data so that they can develop models to help predict future red and blue rabbit population. They are provided with the following DataFrame, rabbit. Specifically, rabbit includes the location, year, unique ID, name and color of each rabbit.

Additionally, they are provided with another DataFrame, metadata. metadata contains the year in consideration, carrots eaten in that year, and the number of animal tracks found.

Fill out the Pandas expression below to generate a DataFrame where each row corresponds to a unique year. Each row should also contain the number of animal tracks, carrots eaten, total red rabbit population, and total blue rabbit population for that year. Fill all missing values with 0. You may not need to use all the provided blanks.

```
rabbit.__A__(_____B_____) \
        .merge(_____C_____) \
        .__D__(0)
```

Here is what your output should look like:

(a) [2 Pts] What goes in the blank indicated by the letter A?

_____

(b) [2 Pts] What goes in the blank indicated by the letter B?

_____

(c) [2 Pts] What goes in the blank indicated by the letter C?

_____

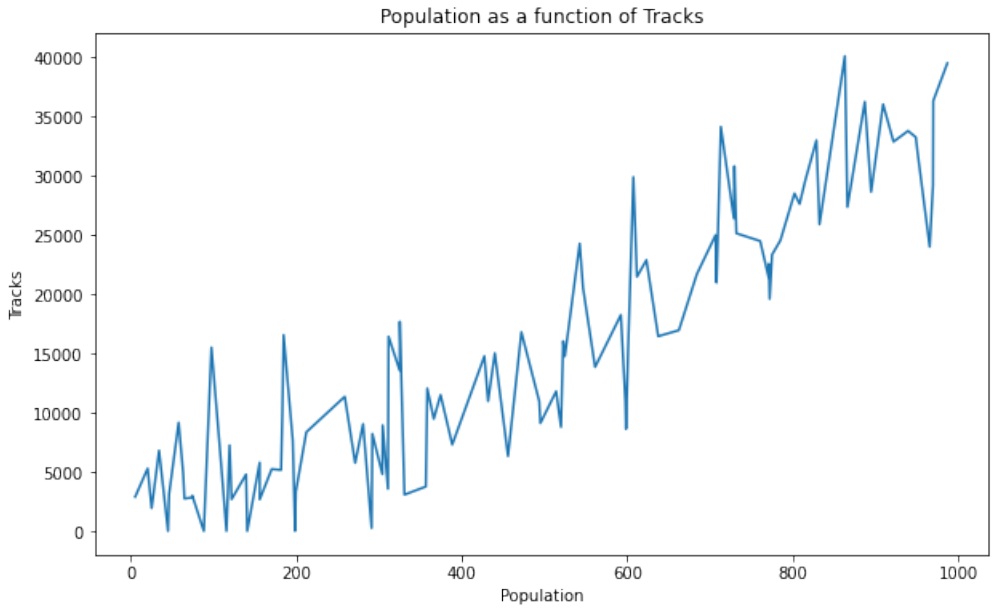
(d) [1 Pt] What goes in the blank indicated by the letter D?

_____

Solution:

```
rabbit.pivot_table(index = 'Year', columns = 'Color',
                   values = 'ID', aggfunc = 'count') \
      .merge(metadata, how = 'outer',
                   left_index = True, right_on = 'Year') \
      .fillna(0)
```

Kermit decides to perform some EDA before diving into some modeling. He decides to focus on the relationship between the population and the number of tracks.



(e) [2 Pts] Describe an issue with the visualization above.

Solution:

- The choice of plot is incorrect. This should be represented by a scatterplot.
- The overall trend is masked by the jagged edges similar to the graph shown in Discussion 5 (Bremorse).

(f) [1 Pt] Which of the following is most likely the correlation coefficient r between population and number of tracks?

1 :9 :3 0 :3 :9 1

Solution: This displays a positive correlation, so any non-positive correlations are not possible (-1, -0.9, -0.3 and 0). It isn't perfectly correlated, so a correlation of 1 is impossible. Visually, it is clear that the correlation is closer to 0.9 than to 0.3 since a correlation coefficient of 0.3 would display much more scatter (and essentially indicates that there is a very weak correlation between the two variables). The answer is therefore 0.9.

Using his findings from above, Kermit wishes to predict the total rabbit population, p, using the number of animal tracks, t, found in the year. He decides to use Ridge regression **without an intercept term**, and with regularization hyperparameter .

(g) [2 Pts] Which of the following is the optimal ^?

$\frac{t^T p}{t^T t + n}$

$\frac{t^T t}{t^T p + n}$

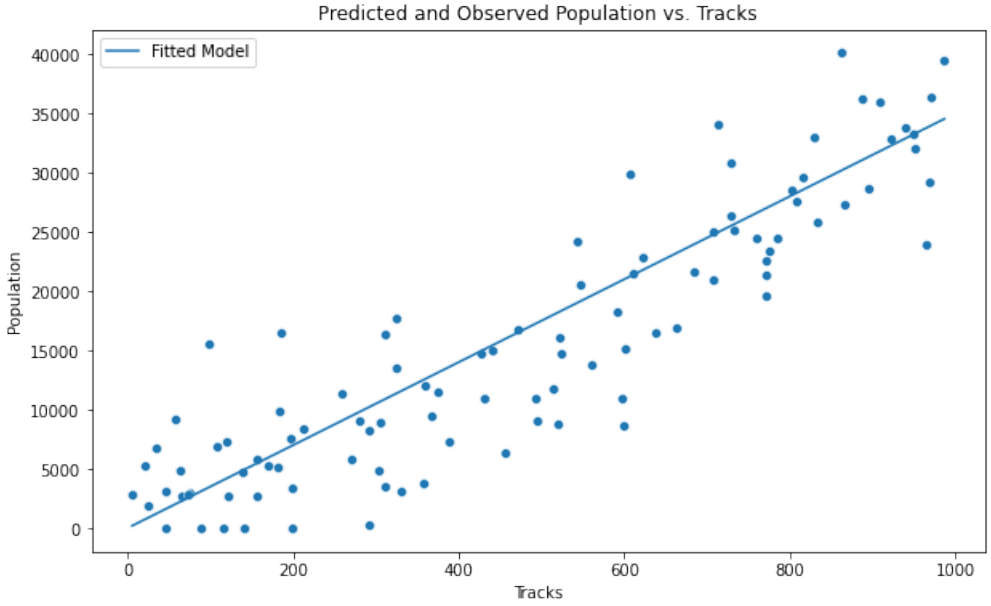
$\frac{1}{t^T p + \quad n}$

Solution: This is the analytical solution for Ridge regression where $X = t$ and $y = p$. Since $t^T t$ is one dimensional, we know the dimensionality of I, the identity matrix, is 1x1 (or a scalar). Note that the inverse of a scalar is just the multiplicative inverse.

$$(X^T X + \quad nI)^{\ 1} X^T y = (t^T t + \quad n)^{\ 1} t^T p$$

$$= \frac{t^T p}{t^T t + \quad n}$$

(h) [2 Pts] After fitting his model, he plots his predictions against his observations as shown below. Which of the following are true? Select all that apply.



The fit is inaccurate due to the large amounts of scatter around the line, which suggests that linear regression is inappropriate.

There is a curvature in the relationship, which suggests that linear regression is inappropriate.

The training loss would decrease if the regularization hyperparameter was 0.

The plot displays high variance, which suggests that should be increased to reduce model complexity.

Solution: While there is scatter around the line, the general trend suggests that linear regression is appropriate since there is no curvature in the relationship. This rules out the first two options.

Since restricts the magnitude of our parameters, removing regularization by setting

$= 0$ would improve our fit and therefore reduce training loss.

The plot does display high variance in the observations themselves, not the predictions necessarily. The variance in the observations is not linked to model complexity or the regularization hyperparameter. Further, being increased would lead to a poorer fit.

15. Using the same data we constructed in part a of the previous question, Kermit wishes to create a second model to predict the *proportion* of red rabbits given the previous year's proportion of red rabbits. Using the proportion of red rabbits from this new model and the total population from the model trained in part (b), he can calculate the red and blue rabbit population for any year!

    (a) [2 Pts] Which machine learning model(s) would **not** be suitable for predicting the proportion of red rabbits?

    KMeans

    LASSO Regression

    Logistic Regression

    Ridge Regression

    PCA

    Solution: The first and last options are unsupervised algorithms, so they do not predict any quantities based on known labels. Logistic regression is a classification algorithm that trains on binary labels, so it is not applicable in this problem. Ridge and LASSO are both acceptable predictor for a continuous variable.

    (b) [2 Pts] Regardless of your answer to the previous question, Miss Piggy invents her own (bad!) loss function to train a linear model **with an intercept term**. Recall that her only feature x represents the proportion of red rabbits from the previous year. The linear model is written as $f_{\sim}(x) = {}_0 + {}_1 x$ and the loss function is written as:

$$L(y; f_{\sim}(x)) = \frac{1}{n} \sum_{i=1}^{n} (f\ (x_i) \quad {}_1 y_i) + \sum_{j=0}^{1} y e^{\ j}$$

    Calculate the optimal value for $\hat{}_1$.

    $\ln(1 + \frac{x}{y})$

    $\ln(1 \quad \frac{x}{y})$

    $\ln(1 + \frac{y}{x})$

    $\ln(1 \quad \frac{y}{x})$

Solution: We solve this question by minimizing with respect to $_1$ using the gradient and setting it equal to 0.

$$\frac{dL}{d\ _1} = \frac{d}{d\ _1}(\frac{1}{n}\sum_{i=1}^{n}(f\ (x_i) \quad {}_1y_i) + \sum_{j=0}^{1} ye^{\ j})$$

$$\frac{dL}{d\ _1} = (\frac{1}{n}\sum_{i=1}^{n}\frac{d}{d\ _1}(f\ (x_i) \quad {}_1y_i) + \sum_{j=0}^{1}\frac{d}{d\ _1}ye^{\ j})$$

$$\frac{dL}{d\ _1} = \frac{1}{n}\sum_{i=1}^{n}(x_i \quad y_i) + ye^{\ 1}$$

We set the value equal to 0.

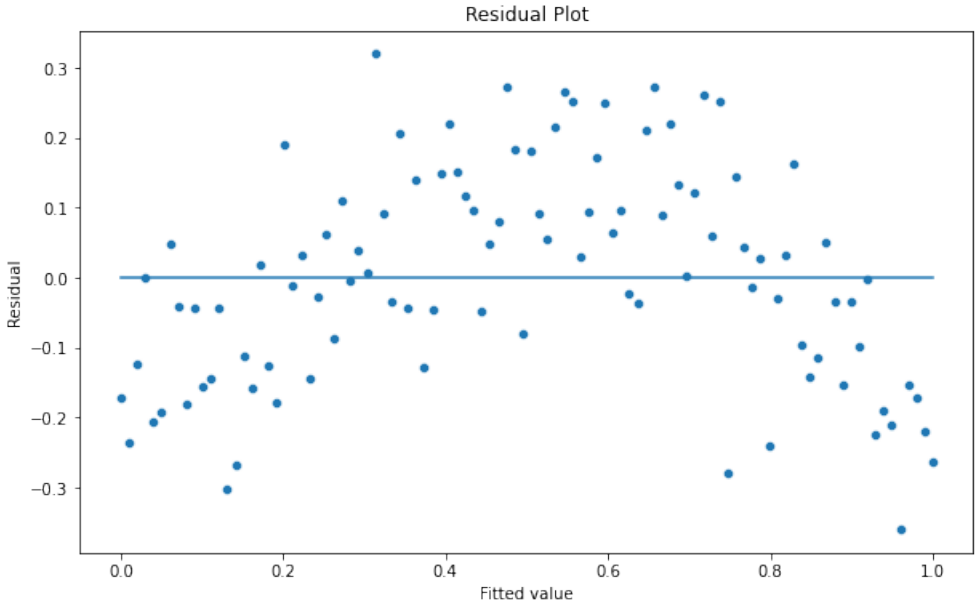
$$\frac{1}{n}\sum_{i=1}^{n}(x_i \quad y_i) + ye^{\ 1} = 0$$

$$ye^{\ 1} = \frac{1}{n}\sum_{i=1}^{n}(y_i \quad x_i)$$

$$ye^{\ 1} = y \quad x$$

$$_1 = \ln(1 \quad \frac{x}{y})$$

(c) [2 Pts] Unfortunately, Miss Piggy's loss function didn't work, so she resorts to minimizing the MAE on her linear model instead. She generates predictions on the test set and create a residual plot as shown below. What does this visualization indicate?



The residual plot displays a roughly normal distribution, which suggests that linear regression is appropriate.

The residual plot displays roughly uniformly random scatter, which suggests that linear regression is appropriate.

The residual plot has many outliers, which suggests that linear regression is inappropriate.

The residual plot has a weak relationship with the fitted value, which suggests that linear regression is inappropriate.

The residual plot has curvature, which suggests that linear regression is inappropriate.

Solution: The residual plot should display uniformly random scatter, but this residual plot displays clear curvature in the form of a parabola. The outliers in the residual plot just indicate that our model performed poorly on those points, but it doesn't suggest that linear regression is inappropriate. The residual plot should have no relationship with the fitted value.

Rare Red Rabbits Return [10 Pts]

16. Kermit wants to apply PCA to the rare rabbit dataset from the previous question to understand patterns in rabbit population per location as a function of the year. Provided is a Pandas DataFrame, `rabbit_pop` (shown below), which contains the rabbit population for every particular year and location. Note that not every year and location is shown here.

	2017	2018	2019	2020
Site A	8789	29372	49271	101822
Site B	18573	38317	102847	192742
Site C	402	3928	20212	80272
Site D	4392	28172	93172	203082

Kermit needs to preprocess his current dataset in order to use PCA.

(a) [1 Pt] Select all appropriate preprocessing steps used for PCA.

Transform each row to have a magnitude of 1 (Normalization)

Transform each column to have a mean of 0 (Centering)

Transform each column to have a mean of 0 and a standard deviation of 1 (Standardization)

None of the above

Solution: We can use standardization or centering for PCA. We cannot compute the covariance matrix correctly using SVD if the data is not centered with mean 0.

(b) [1 Pt] Kermit wishes to apply a transformation to the rabbit population at each site for each year so that he can apply PCA more effectively. What transformation function $f(p)$ would be most effective to apply to the rabbit population p for using PCA?

Hint: Notice the population at each site appears to grow exponentially every year.

```
f = np.log
f = np.exp
f = np.square
f = np.sqrt
f = lambda x:  x
```

Solution: Since the population is exponential, we use the `f = np.log` function to linearize the relationship for PCA.

(c) [2 Pts] Assume we decide to standardize the data, regardless of your answer to the previous subparts. Select the line of Pandas code that preprocesses the population for PCA into the DataFrame `rabbit_PCA`.

```
rabbit_PCA = f(rabbit_pop - rabbit_pop.mean()) /
rabbit_pop.std()
rabbit_PCA = f((rabbit_pop - rabbit_pop.mean()) /
rabbit_pop.std())
rabbit_PCA = f(rabbit_pop - rabbit_pop.mean()) /
f(rabbit_pop.std())
rabbit_PCA = (f(rabbit_pop) - f(rabbit_pop).mean()) /
f(rabbit_pop).std()
rabbit_PCA = f(rabbit_pop - rabbit_pop.mean()) -
f(rabbit_pop.std())
```

Solution: Note that we apply our feature transformations before centering our data for PCA. This is because we want to center the data on the dataset on the featurized dataset on which we want to run PCA. Therefore, option D is the only valid option.

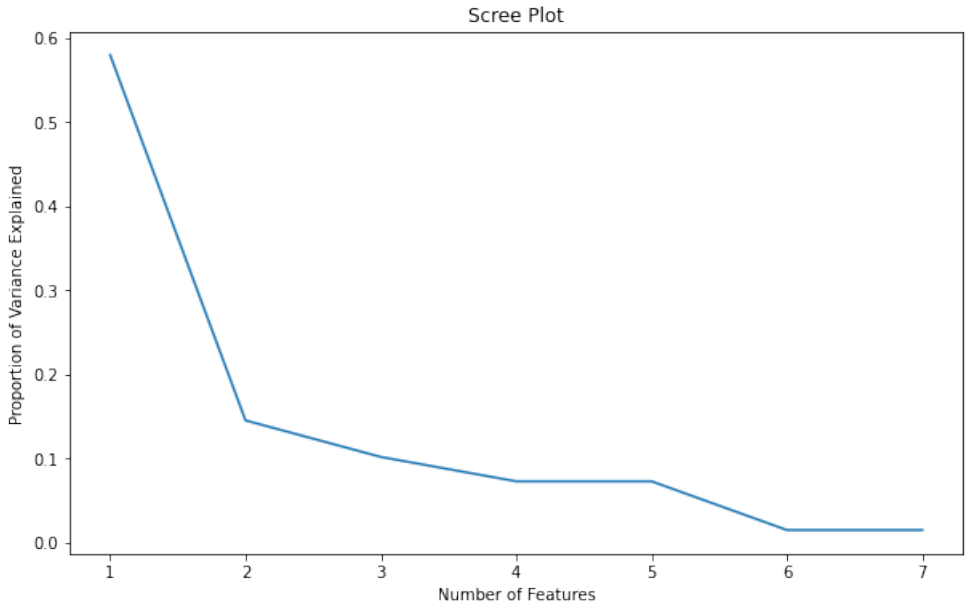
(d) [2 Pts] Write a line of code that returns the first 3 principal components assuming you have `rabbit_PCA` and the following variables returned by SVD.

```
u, s, vt = np.linalg.svd(rabbit_PCA, full_matrices = False)
first_3_pcs = ____________________________________
```

Solution:

```
(rabbit_PCA @ vt.T)[:, :3]
```

(e) [2 Pts] Kermit successfully applies PCA and makes a scree plot that is displayed below. How many principal components should Kermit use to capture at least 80% of the variance in the rabbit population data?

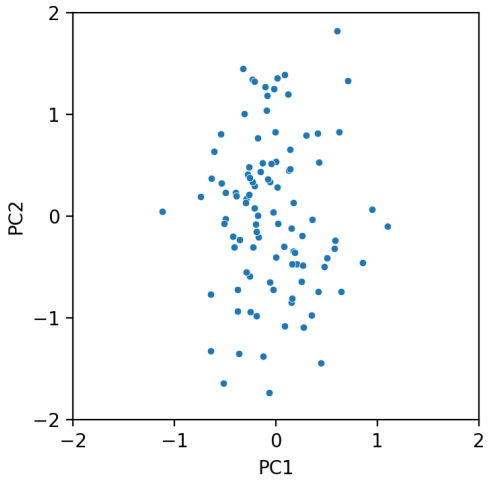


1 2 **3** 4 5 6 7

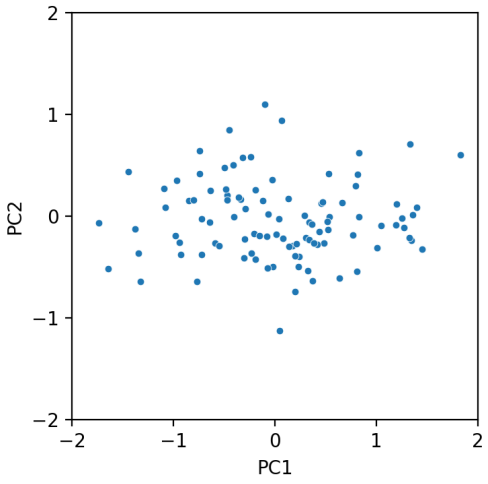
Solution: The first principal component captures around 58% of the variance, the second captures 15%, and the third component captures around 11%. This sums to around 84%, so we need 3 features.

(f) [2 Pts] We now wish to display the first two principal components in a scatterplot. Which of the following plots could potentially display the first two principal components?
Hint: The above scree plot may be helpful.

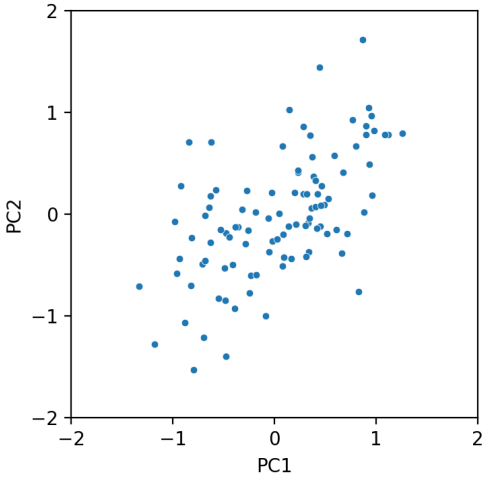
A.



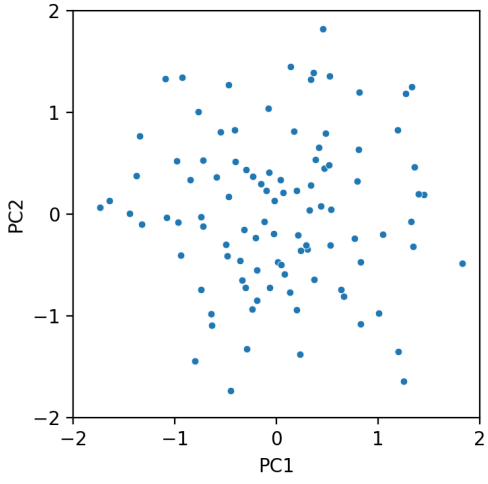
B.



C.



D.



A **B** C D

Solution: The first principal component PC1 must capture more variance than PC2 by the properties of PCA. The scree plot from the previous question tells us that PC1

captures nearly 4 times as much variance. Therefore, options A, C, and D don't work because PC1 doesn't have 4 times as much variance in any of them. Additionally, the principal components must be axis-aligned, which is another mark against C. Therefore, the answer is B.