Data 100/200, Midterm

Spring 2024

Name: ____

Email: ____@berkeley.edu

Student ID: ____

Name and SID of the person on your right: ______________________

Name and SID of the person on your left: ______________________

Instructions:

This exam consists of **70 points** spread out over **6 questions** and the Honor Code certification. The exam must be completed in **110 minutes** unless you have accommodations supported by a DSP letter.

Note that some questions have circular bubbles to select a choice. This means that you should only **select one choice**. Other questions have boxes. This means you should **select all that apply**. These will always have at least one correct answer. Please shade in the box/circle to mark your answer.

You must write your Student ID number at the top of each page.

Points Breakdown:

Question	Q1	Q2	Q3	Q4	Q5	Q6
Points	14	11	12	11	11	10

Honor Code [1 Pt]:

As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others. I am the person whose name is on the exam, and I completed this exam in accordance with the Honor Code.

Signature: ____

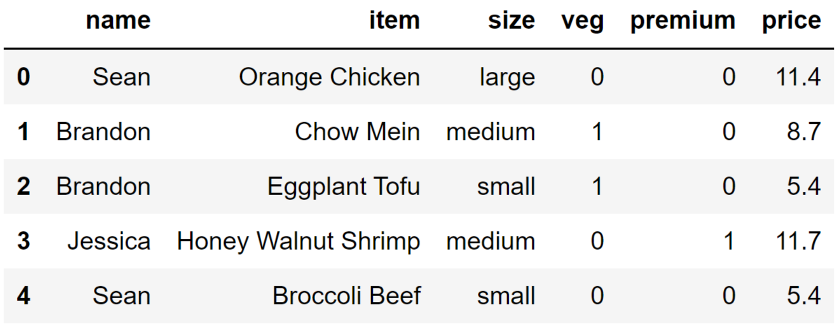
This page has been intentionally left blank.

1 Pandas Express [14 Pts]

This question involves coding. All code for each part, where applicable, must be written in Python. Assume that the `pandas` library has been imported as `pd`, the `numpy` library has been imported as `np`, and the Python RegEx library has been imported as `re`.

Milad is a big fan of the data-science-themed restaurant, Pandas Express. He tracks Data 100 staff members' orders at the restaurant in a `DataFrame` called `orders`. Descriptions of its columns and its first five rows can be found below:

- `name`: Name of a Data 100 staff member (type = `str`).
- `item`: The food item they ordered (type = `str`).
- `size`: The portion size of the item - small, medium, or large (type = `str`).
- `veg`: 0 if the item is not vegetarian, 1 if the item is vegetarian (type = `numpy.int64`).
- `premium`: 0 if the item is not a premium item, 1 if the item is a premium item (type = `numpy.int64`).
- `price`: The price of the order in dollars (type = `numpy.float64`).



	name	item	size	veg	premium	price
0	Sean	Orange Chicken	large	0	0	11.4
1	Brandon	Chow Mein	medium	1	0	8.7
2	Brandon	Eggplant Tofu	small	1	0	5.4
3	Jessica	Honey Walnut Shrimp	medium	0	1	11.7
4	Sean	Broccoli Beef	small	0	0	5.4

(a) [2 Pts] Which of the following lines of code will output a `DataFrame` only containing rows with dishes that are both vegetarian **AND** have a `price` above $7.50? **Select all that apply.**

□ A. `orders[(orders["veg"]==1) and (orders["price"]>7.5)]`

□ **B.** `orders.loc[(orders["veg"]==1) & (orders["price"]>7.5)]`

□ C. `orders[orders.isin(["veg" == 1, "price" > 7.5])]`

□ D. `orders[(orders["veg"]==1) | (orders["price"]>7.5)]`

Solution:
Option A is incorrect because you cannot use the word `and` to perform multi-condition filtering.

Option B is the correct way to perform a filter that satisfies multiple conditions.

Option C is an incorrect use of `.isin()` and would throw an error.

Option D is incorrect, as it only filters for rows that satisfy one of the specified conditions.

(b) [2 Pts] Fill in the blanks below to create `above_5`: a `DataFrame` with the same structure as `orders` but only containing rows that include an `item` that was ordered more than 5 times.

```
above_5 = orders.____A____(____B____).____C____(lambda sf: _____D_____)
```

(i) Fill in blank `A`:

Solution: `groupby`

(ii) Fill in blank `B`:

Solution: `"item"`

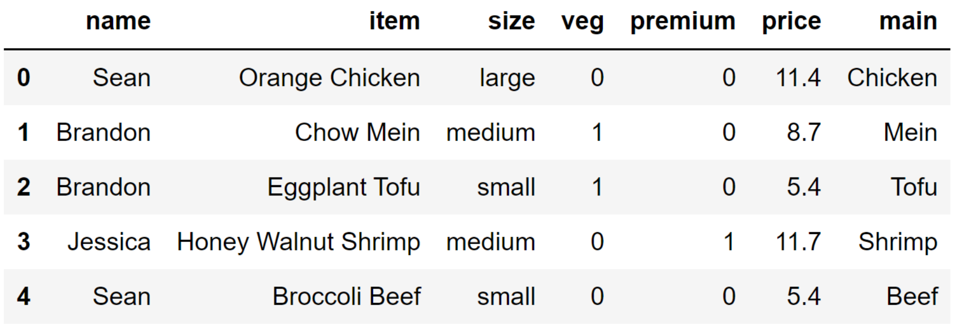
(iii) Fill in blank `C`:

Solution: `filter`

(iv) Fill in blank `D`:

Solution: `len(sf) > 5` or `sf.shape[0] > 5`

(c) [2 Pts] Milad determines that the main ingredient of each `item` is its last word (i.e., the main ingredient of `"Orange Chicken"` is `"Chicken"`). He wishes to store each main ingredient in a new column in `orders` called `main`, as seen below:



	name	item	size	veg	premium	price	main
0	Sean	Orange Chicken	large	0	0	11.4	Chicken
1	Brandon	Chow Mein	medium	1	0	8.7	Mein
2	Brandon	Eggplant Tofu	small	1	0	5.4	Tofu
3	Jessica	Honey Walnut Shrimp	medium	0	1	11.7	Shrimp
4	Sean	Broccoli Beef	small	0	0	5.4	Beef

Which of the following options will correctly fill in the blank to create the `main` column? **Select all that apply.**

```
orders["main"] = ______________________________
```

□ **A.** `orders["item"].str.split().str[-1]`

□ B. `orders["item"].str[-1]`

□ C. `orders["item"].str.contains(r"\s*(\w+)")`

□ **D.** `orders["item"].str.extract(r"(\w+)$")`

Solution:
Option A is correct. `str.split()` separates each `item` by the spaces (essentially splitting into individual words), and `str[-1]` goes and gets the last of these words.

Option B is incorrect, as this would only get the last character of each `item`.

Option C is incorrect, as this would return a `Series` of `True` or `False` rather than the strings we are looking for.

Option D is correct. The RegEx pattern will look for a series of characters followed by the end of the string (a.k.a. the last word) and extract that from each `item`.

Note: Option D was removed from the exam because the reference sheet incorrectly stated that `str.extract()` returned a `Series`, not a `DataFrame`. However, it would still run correctly with no bugs or errors.

(d) [2 Pts] Suppose we are given a list called `exclude_mains`. Fill in the blank to create `entrees`: a `DataFrame` with the same structure as `orders` but only containing rows with a `main` **NOT** in `exclude_mains`. Assume that the `main` column was created correctly.

```
entrees = orders[______________________________________]
```

Fill in the blank. Only solutions which do not iterate through `exclude_mains` will receive full credit.

Solution: ~ `orders["main"].isin(exclude_mains)`

(e) [2 Pts] Fill in the blanks below to create `main_revenue`: a `Series` that displays the total money that each `main` generated across all orders. Assume that the `main` column was created correctly.

```
main_revenue = orders.____A____(___B___)[____C____].____D____
```

(i) Fill in blank A:

Solution: `groupby`

(ii) Fill in blank B:

Solution: `"main"`

(iii) Fill in blank C:

Solution: `"price"`

(iv) Fill in blank D:

Solution: `sum()`, `agg(sum)`, `agg("sum")`, or `agg(np.sum)`

(f) [2 Pts] Using `main_revenue` from the previous part, fill in the blanks below to find the `main` which generated the **median** amount of money across all orders. Assume that there are an odd number of entries in `main_revenue`.

Hint: `a // b` returns `a` divided by `b` rounded down to the nearest integer.

```
position = _________________A_________________ // 2

main_revenue._____B_____()._____C_____[_____D_____]
```

(i) Fill in blank `A`:

Solution: `len(main_revenue)` or `main_revenue.shape[0]` or any code that gets the length of `main_revenue`.

(ii) Fill in blank `B`:

Solution: `sort_values`

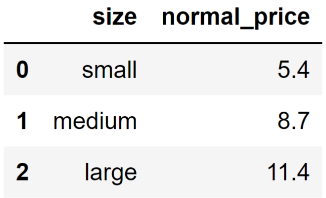
(iii) Fill in blank `C`:

Solution: `iloc` or `index`
Note: the wording of this question caused some confusion as to whether students were supposed to get the actual median money amount, or its corresponding `main`. As a result, both options were accepted.

(iv) Fill in blank `D`:

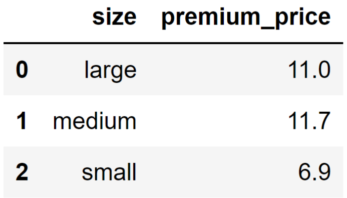
Solution: `position`

(g) [2 Pts] Milad creates two new `DataFrame` objects: `normal` and `premium`, which contain normal and premium prices for each size, respectively.



	size	normal_price
0	small	5.4
1	medium	8.7
2	large	11.4

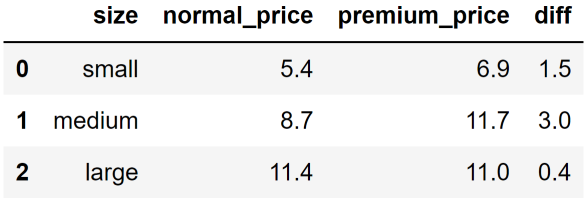
`normal`



	size	premium_price
0	large	11.0
1	medium	11.7
2	small	6.9

`premium`

Fill in the blanks to create a `DataFrame` called `prices` (shown below), which includes a new column `diff` displaying the `absolute` difference between each size.



	size	normal_price	premium_price	diff
0	small	5.4	6.9	1.5
1	medium	8.7	11.7	3.0
2	large	11.4	11.0	0.4

```
prices = normal.______A______(________________B________________)
prices["diff"] = _______________________C_______________________
```

(i) Fill in blank `A`:

Solution: `merge`

(ii) Fill in blank `B`:

Solution: `right=premium, left_on="size", right_on="size"` or any combination of arguments that correctly achieves this merge.

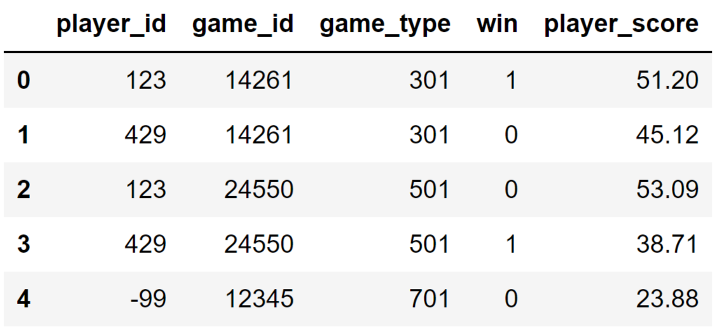
(iii) Fill in blank `C`:

Solution:
`abs(prices["premium_price"] - prices["normal_price"])`

2 Exploratory Dart-a Analysis [11 Pts]

A local arcade has given Nikhil access to data from their electronic dart boards. The arcade records every game played and the players who participated (players can only play each game once). He stores this data in a `DataFrame` called `darts`. Descriptions of its columns and its first five rows can be found below:

- `player_id`: Randomly-generated unique identifier for each player (type = `numpy.int64`).
- `game_id`: Randomly-generated unique identifier for each game of darts (type = `numpy.int64`).
- `game_type`: The type of darts game played: 301, 501, or 701 (type = `numpy.int64`).
- `win`: 1 if the player won that given game, 0 if not (type = `numpy.int64`).
- `player_score`: Player's average score for a given game (type = `numpy.float64`).



	player_id	game_id	game_type	win	player_score
0	123	14261	301	1	51.20
1	429	14261	301	0	45.12
2	123	24550	501	0	53.09
3	429	24550	501	1	38.71
4	-99	12345	701	0	23.88

(a) [1 Pt] What is the minimum set of columns that form the primary key of this dataset? Write the label(s) of the column or group of columns.

Solution: `player_id` and `game_id`. There is one entry for each player in each game.

(b) [2 Pts] Which **variable type** best describes each of the following columns of `darts`?

	Quantitative Continuous	Quantitative Discrete	Qualitative Ordinal	Qualitative Nominal
(i) `player_id`	○	○	○	[✓ ○]
(ii) `player_score`	[✓ ○]	○	○	○

Solution:
`player_id` is qualitative nominal because it describes categories without an inherent order.

`player_score` is quantitative continuous because it represents numerical scores that can take on any value.

(c) [2 Pts] Some values of `player_id` are missing, represented by `-99`. Which method below is **NEVER** appropriate to deal with missing values in `player_id`?

○ A. Keeping it as is (don't do anything).

○ B. Removing rows with missing values.

○ **C. Imputing with the mean of the column.**

○ D. Imputing with the mode of the column.

Solution:
Option A is incorrect. If we wanted to perform analysis at the game level, then we might not need to fill in missing player IDs.

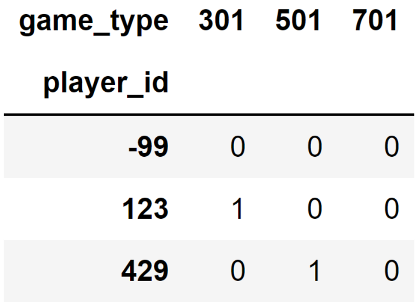
Option B is incorrect. If only a few rows had missing player IDs, it may be a good idea to drop them entirely.

Option C is correct. `player_id` is a qualitative variable, so imputing with a numerical mean doesn't make much sense here.

Option D is incorrect. If the missing data occurs randomly, one can assume that the players who play the most may also be the players who wind up having their `player_id` go unrecorded the most.

(d) [4 Pts] Nikhil wants to view each player's proportion of wins for each `game_type` and generates each `DataFrame` shown below. Assume that only the first five rows of `darts` were used.

(i) Which method **by itself** was used to generate this `DataFrame`?



game_type	301	501	701
player_id			
-99	0	0	0
123	1	0	0
429	0	1	0

○ A. `.groupby()` with `.mean()`

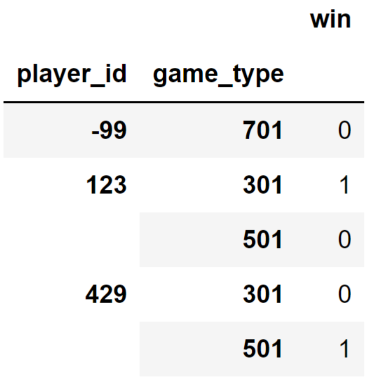
○ B. `.value_counts()`

○ [✓ **C.** `.pivot_table()`]

○ D. `.agg()`

Solution: This table has the `player_id` as the index and one column for each unique `game_type`. Additionally, if a player never played a certain `game_type`, they still have an entry of 0 present in the table. Both are behaviors that are only achieved by a call to `.pivot_table()`.

(ii) Which method **by itself** was used to generate this `DataFrame`?



		win
player_id	**game_type**	
-99	**701**	0
123	**301**	1
	501	0
429	**301**	0
	501	1

○ [✓ **A.** `.groupby()` **with** `.mean()`]

○ B. `.value_counts()`

○ C. `.pivot_table()`

○ D. `.agg()`

Solution: This table is similar to the last one, except now both `player_id` and `game_type` form the index. This is a clear sign that `.groupby()` was used.

(e) [2 Pts] Nikhil writes the following lines of code as part of his exploratory data analysis. Under what condition will Nikhil's code print `True`?

```
counts = darts[darts["win"] == 1]["game_id"].value_counts()
print(sum(counts != 1) == 0)
```

○ A. Each player only plays in each game once.

○ B. Each player has won at least one game.

○ [✓ **C. There is exactly one winner per game.**]

○ D. There are no repeated values of `game_id`.

Solution: This code first sets `counts` equal to a `Series` displaying the number of winners for each `game_id`. It then checks that each of these values is 1. In other words: it checks that each game had exactly one winner.

3 Free Samples [12 Pts]

The professors want to conduct a mid-semester survey. They wish to gather feedback from all the students enrolled in Data 100/200 but have determined that it will be easier to take a small sample instead. They propose a couple of different methods to record this survey.

(a) [3 Pts] Suppose the professors survey by asking the first 20 students who attend Professor Norouzi's office hours to fill out a form.

(i) **In one sentence or less**: what is the population of interest?

Solution: All students enrolled in Data 100/200, as stated at the beginning of the question.

(ii) Which of the following statements are true regarding this procedure? **Select all that apply.**

☐ A. This is a simple random sample.

☐ **B. This is a convenience sample.**

☐ **C. The results will likely have selection bias.**

☐ **D. The results will likely have chance error.**

Solution:
Option A is incorrect. There is no random selection being done here.

Option B is correct. Choosing the first individuals you see is a common example of a convenience sample.

Option C is correct. There is always a strong chance of selection bias when there is a convenience sample.

Option D is correct. There is always a strong likelihood of chance error when taking a small sample from a population.

(b) [4 Pts] The professors opt for a different method. At the end of the most recent lecture, the professors assign a unique number to each person in the crowd, then uniformly and randomly select 20 numbers and have the corresponding individuals fill out the survey.

(i) **In one sentence or less**: what is the sampling frame of this method?

Solution: All people in attendance at the most recent lecture.

(ii) Which of the following groups of individuals are in the sampling frame but **NOT** in the population of interest? **Select all that apply.**

□ A. Students who did not attend the lecture.

□ B. Students who attended the lecture online.

□ **C. Individuals who are not enrolled in the course but attended the lecture.**

□ **D. TAs who were at the lecture.**

Solution:
Option A is incorrect. These individuals are in the population of interest.

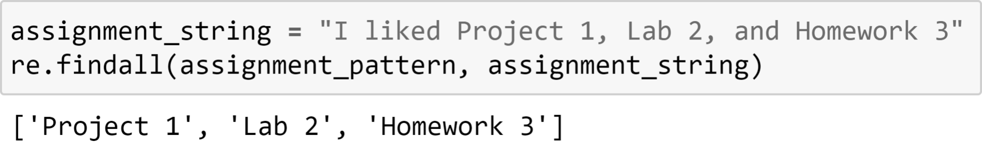
Option B is incorrect. These individuals are in the population of interest.

Option C is correct. Being present at the lecture, they are in the sampling frame. Not being enrolled in the class, they are not in the population of interest.

Option D is correct for the same reason as Option C.

For the remainder of this question, the Python RegEx library has been imported as `re`. **For all parts, you will only need to worry about the example strings given to you and may assume that these examples cover all edge cases.**

(c) [2 Pts] The professors want to analyze feedback on particular assignments (projects, labs, and homeworks) and create a RegEx pattern called `assignment_pattern`. Given the example output below, which RegEx patterns could be `assignment_pattern`? **Select all that apply.**



```
assignment_string = "I liked Project 1, Lab 2, and Homework 3"
re.findall(assignment_pattern, assignment_string)
```

```
['Project 1', 'Lab 2', 'Homework 3']
```

□ **A.** `r"Homework\s\d+|Lab\s\d+|Project\s\d+"`

□ B. `r"[Homework|Lab|Project]\s\d+"`

□ C. `r"Homework|Lab|Project\s\d+"`

□ **D.** `r"\w+\s\d+"`

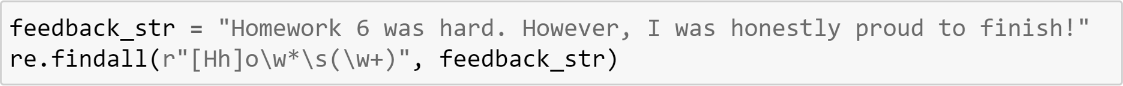
Solution:
Option A is correct. There needs to be a space and number following each of `Homework`, `Lab`, and `Project`.

Option B is incorrect, as it would only look for a singular letter that belongs in any of `Homework`, `Lab`, or `Project`, rather than the whole word.

Option C is incorrect, as it would only capture the assignment numbers after `Project`, not `Homework` or `Lab`.

Option D is correct. It looks for at least one letter, followed by a space, then a digit, which fits our needs here.

(d) [3 Pts] What will be the output of the code snippet below? Recall that `re.findall` returns a list of matches.



```
feedback_str = "Homework 6 was hard. However, I was honestly proud to finish!"
re.findall(r"[Hh]o\w*\s(\w+)", feedback_str)
```

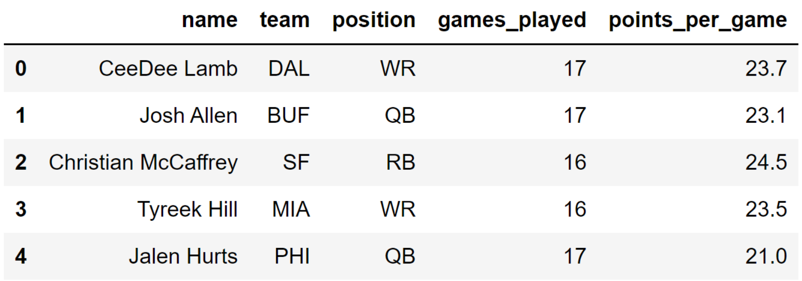
Put your answer in list format.

Solution: `["6", "proud"]`
This pattern looks for a string that starts with an `H` or `h`, followed by an `o`, then some other characters, and a space. This would be `Homework` and `honestly` (there is a comma, not a space, right after `However`). It then captures the string after, which are `6` and `proud`.

4 Box Scores and Box Plots [11 Pts]

For this question, assume that the seaborn library has been imported as sns and the matplotlib pyplot library has been imported as plt.

Ishani is a fantasy sports champion who wants to use her data science skills to get a head start next season. She records last season's players' team, position, number of games played, and average number of fantasy points per game, storing the results in a DataFrame called players. The first 5 rows are given to you below.



	name	team	position	games_played	points_per_game
0	CeeDee Lamb	DAL	WR	17	23.7
1	Josh Allen	BUF	QB	17	23.1
2	Christian McCaffrey	SF	RB	16	24.5
3	Tyreek Hill	MIA	WR	16	23.5
4	Jalen Hurts	PHI	QB	17	21.0

(a) [2 Pts] Which of the following visualization types are appropriate to visualize the distribution of games_played? **Select all that apply.**

□ **A. Histogram**

□ **C. Violinplot**

□ B. Hexplot

□ D. Scatterplot

Solution:
Options A and C are both common choices to display the distribution of one quantitative column.

Options B and D both display the relationship between two quantitative variables.

(b) [2 Pts] Ishani creates a DataFrame called output using the following code:

```
total = players["games_played"] * players["points_per_game"]
players["total"] = total
output = players.groupby("team")[["total"]].mean().reset_index()
```

Which of the following lines of code are best suited to visualize the relationship between both columns of output? **Select all that apply.**

□ A. `sns.violinplot(data = output, x = "total");`

□ **B.** `sns.boxplot(data = output, x = "team", y = "total")`

□ **C.** `plt.bar(x = output["team"], height = output["total"])`

□ D. `sns.countplot(data = output, x = "team")`

Solution:
`output` displays the average number of points accrued by players of each team. A good visualization will include both the team and their points.

As such, Option B will create a boxplot (with one item) for each team, while Option C will create a bar chart showing the average points per team.

Option A does not include any information about the teams, and Option D only shows how many times each team shows up in `output`.

(c) [2 Pts] Which of the following pieces of information can you **always** determine by looking at a box plot? **Select all that apply.**

□ **A. Quartiles** □ C. Mean

□ **B. Existence of outliers** □ D. Mode

Solution:
Option A is correct. The central box captures the first, second, and third quartiles.

Option B is correct. Values outside the whiskers are considered outliers and plotted as independent dots.

Option C is incorrect. Boxplots do not plot the mean.

Option D is incorrect. This is not something a boxplot can do.

(d) [3 Pts] For the subparts of this question, select the best choice to fill in each blank of the below statement:

The first step of making a Kernel Density Estimate (KDE) curve is to place a kernel at ____`A`____. Next, you normalize each kernel so that ____`B`____ and sum the normalized kernels. One key value to set is the bandwidth parameter α, which determines the ____`C`____ of our KDE curve.

(i) Fill in Blank A:

○ A. The minimum and maximum
○ **B. Every data point**
○ C. Some data points, depending on α
○ D. None of the above

(ii) Fill in Blank B:

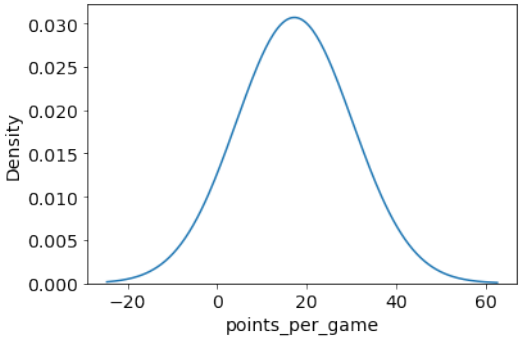
○ A. Each kernel has an area of α
○ B. Each kernel has an area of 1
○ **C. The total area under the KDE curve is 1**
○ D. None of the above

(iii) Fill in Blank C:

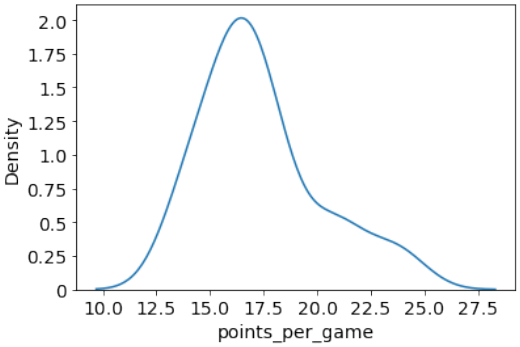
○ A. Height
○ **B. Smoothness**
○ C. Area
○ D. None of the above

(e) [1 Pt] Ishani generates the following KDE curves from the `points_per_game` column:

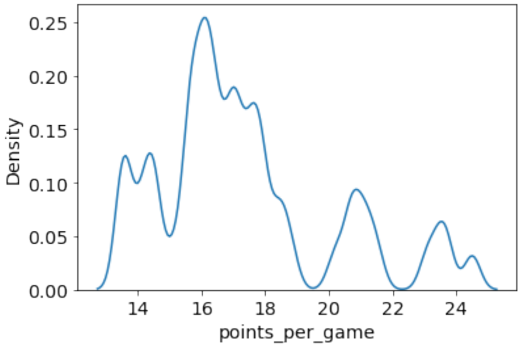
A.



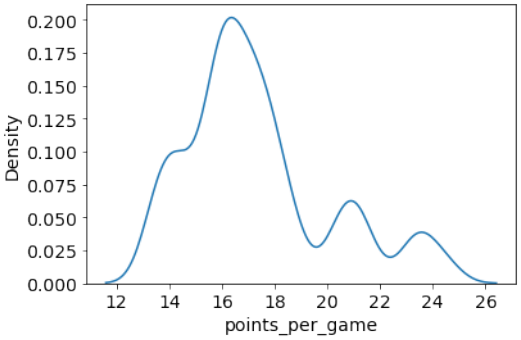
B.



C.



D.



Which of the above plots is **NOT** a valid KDE curve?

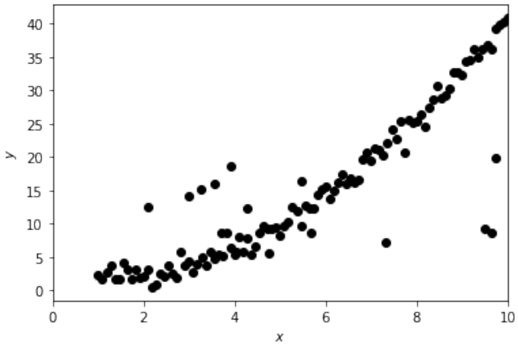
○ A ○ **B**

○ C ○ D

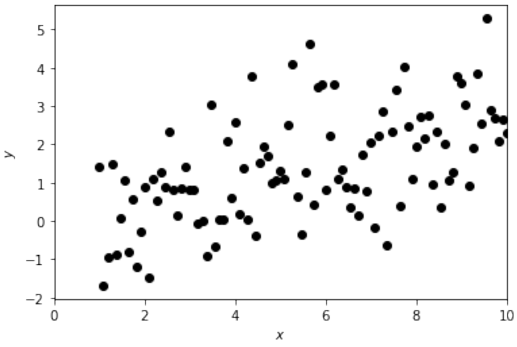
Solution: The total area under a KDE curve has to be 1, which is violated by Option B.

(f) [1 Pt] Suppose you have two variables, x and y, whose relationship can be linearized by performing a log transformation on x. Which plot most likely represents the original relationship between x and y?

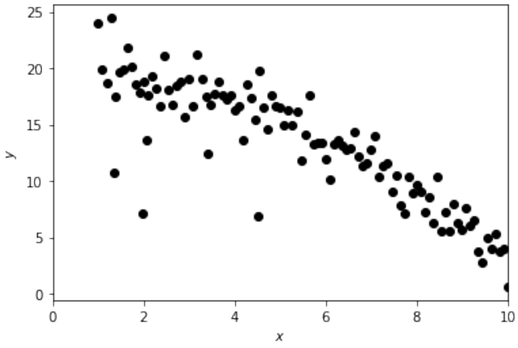
A.



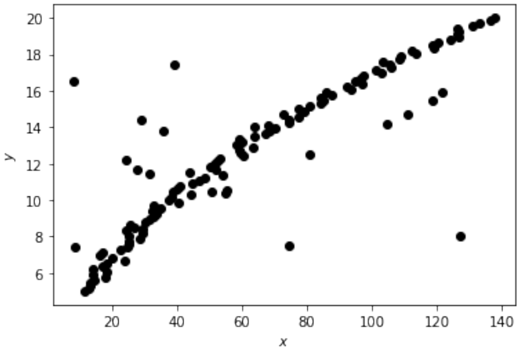
B.



C.



D.



○ A ○ B

○ C ○ **D**

Solution: From the Tukey-Mostellar Bulge Diagram, the relationship in plot D is the best candidate for this transformation.

5 Loss Boss [11 Pts]

Angela and Shiny have a dataset with n datapoints: $\{(x_i, y_i)\}_{i=1}^{n}$. For each datapoint i, $x_i \in R$ is the feature and $y_i \in R$ is the target output.

(a) [5 Pts] Angela proposes to model some particular observation y_i as follows:

$$\hat{y}_i = \frac{1}{\theta_1} x_i$$

(i) Take the derivative with respect to θ_1 of the Mean Squared Error (MSE) of Angela's model. **Please simplify your answer in terms of** x_i, y_i, θ_1 **and** n.

Derivative = ________________________

Solution: The equation for the MSE of Angela's model can be written as:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \frac{1}{\theta_1}x_i)^2$$

To take the derivative with respect to θ_1, we treat all other terms in our model as constants and differentiate using the chain rule. The result is seen below:

$$\frac{1}{n}\sum_{i=1}^{n} 2(y_i - \frac{1}{\theta_1}x_i)(\frac{1}{\theta_1^2}x_i)$$

$$= \frac{1}{n}\sum_{i=1}^{n}(\frac{2y_ix_i}{\theta_1^2} - \frac{2x_i^2}{\theta_1^3})$$

(ii) What is the value of $\hat{\theta}_1$ that minimizes MSE? The critical point you find is guaranteed to be a minimum, so you do **not** need to prove that this point is a minimum. **Please simplify your answer in terms of x_i, y_i and n.**

$\hat{\theta}_1$ = ______________________

Solution: To find the optimal $\hat{\theta}_1$, we take the derivative from the previous part and set it equal to 0:

$$\frac{1}{n}\sum_{i=1}^{n}(\frac{2y_ix_i}{\theta_1^2} - \frac{2x_i^2}{\theta_1^3}) = 0$$

From here, we perform some algebra to solve for θ_1:

$$\frac{1}{n}\sum_{i=1}^{n}(\frac{2y_ix_i}{\theta_1^2} - \frac{2x_i^2}{\theta_1^3}) = 0$$

$$\frac{1}{n}\sum_{i=1}^{n}(\frac{2y_ix_i}{\theta_1^2}) - \frac{1}{n}\sum_{i=1}^{n}(\frac{2x_i^2}{\theta_1^3}) = 0$$

$$\frac{1}{n}\sum_{i=1}^{n}\frac{2y_ix_i}{\theta_1^2} = \frac{1}{n}\sum_{i=1}^{n}\frac{2x_i^2}{\theta_1^3}$$

$$\frac{1}{\theta_1^2}\sum_{i=1}^{n}2y_ix_i = \frac{1}{\theta_1^3}\sum_{i=1}^{n}2x_i^2$$

Multiply both sides by θ_1^3 :

$$\theta_1\sum_{i=1}^{n}2y_ix_i = \sum_{i=1}^{n}2x_i^2$$

$$\hat{\theta}_1 = \frac{\sum_{i=1}^{n}2x_i^2}{\sum_{i=1}^{n}2y_ix_i}$$

$$= \frac{\sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} y_i x_i}$$

(b) [2 Pts] Shiny collects three data points, as shown in the table below. She wishes to use the same model from part (a) with $\theta_1 = \frac{1}{3}$. What is the Mean Absolute Error (MAE) of Shiny's model? **Please simplify your answer.**

x	y
1	5
2	2
4	9

As a reminder, this is the model from part (a): $\hat{y}_i = \frac{1}{\theta_1} x_i$

MAE = ______________________

Solution: Our model with these weights would be $\hat{y}_i = 3x_i$.
So $x_i = 1$ predicts $\hat{y}_i = 3$, $x_i = 2$ predicts $\hat{y}_i = 6$, and $x_i = 4$ predicts $\hat{y}_i = 12$.

To get the MAE, we first sum the absolute errors (add up each $|y_i - \hat{y}_i|$):

$$|5-3| + |2-6| + |9-12| = 2 + 4 + 3 = 9$$

Next, to get the mean, we divide this sum by the number of data points (3):

$$9/3 = 3$$

(c) [2 Pts] Angela wishes to use a constant model to predict each $\hat{y}_i$. Which of the following methods could she utilize to always find the $\hat{\theta}_0$ that minimizes MSE? **Select all that apply.**

□ **A. Setting $\hat{\theta}_0$ to the mean of y.**

□ B. Setting $\hat{\theta}_0$ to the median of y.

□ C. Setting $\hat{\theta}_0$ to the mode of y.

□ **D. Taking a derivative of the MSE function with respect to θ_0 and solving for the minima.**

Solution:
Option A is correct. If you use the traditional method of finding the minima of the MSE using partial derivatives, then the calculation will show that the mean is the minimizing constant model.

Options B and C are incorrect. The median and mode are not always equal to the mean.

Option D is correct. This method will always work for any convex and differentiable loss function (which MSE is).

(d) [2 Pts] Consider the loss function Mean Cubed Error: $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^3$. Which of the following statements are true for this loss function when used for a linear model? **Select all that apply.**

□ A. Overpredictions are always penalized more than underpredictions.

□ **B. Underpredictions are always penalized more than overpredictions.**

□ **C. Predictions that are closer to their true target value are sometimes penalized more than predictions that are farther from their true target value.**

□ D. There is a single, finite value of θ which minimizes this loss function.

Solution:
Option A is incorrect. Overpredictions will result in a negative loss for a given data point, while underpredictions will result in a positive loss for a given data point.

Option B is correct. See above.

Option C is correct. A massive overprediction would have a lower loss than a slight underprediction.

Option D is incorrect. This loss function has no finite minimum.

6 SOLS [10 Pts]

It's the year 2124, and Yuerou and Mir are excited to lead an expedition to the planet Mars. They decide to use Ordinary Least Squares (OLS) to predict how many Martian days (also known as sols) their mission can last.

(a) [2 Pts] Yuerou begins by researching past missions to Mars. They store features in the design matrix $\mathbb{X}'$, with the first column representing the intercept term, the second column representing the number of portions of food brought on the missions, and the third column representing the number of astronauts. The vector $\mathbb{Y}'$ stores the duration of the missions in sols.

Yuerou finds the ideal weight vector to be $\hat{\theta} = [-3, 0.25, 2]^T$.

(i) Holding all other variables constant, how many more sols would a mission last for each additional astronaut in the crew, according to this model?

________ additional sols

Solution: 2 is the correct answer. The first item in $\hat{\theta}$ represents the intercept term, the second item is the weight of the food portions, and the third item is the weight of the number of astronauts.

(ii) Suppose Yuerou stumbles across a mission with 20 food portions and 3 astronauts lasting 5 sols. What would be the residual of this model's prediction on this mission?

Residual = ________________________

Solution: Based on these parameters, the model would predict that this mission lasted $-3 + 0.25(20) + 2(3) = 8$ sols. The correct value is 5 sols, so our residual is $5 - 8 = -3$.

For the remainder of this question, Mir creates a new dataset with more observations and more features. The new $n \times p$ design matrix is denoted as $\mathbb{X}$, and the new target variables are stored in the vector $\mathbb{Y}$.

(b) [2 Pts] Which of the following statements are true regarding an OLS model involving $\mathbb{X}$, $\mathbb{Y}$, and the ideal weight vector $\hat{\theta}$? **Select all that apply.**

□ A. $\mathbb{Y}$ and $\hat{\theta}$ have the same dimensionality.

□ **B. The residual vector ($\vec{e}$) is orthogonal to the span of $\mathbb{X}$.**

□ C. The prediction vector ($\hat{\mathbb{Y}}$) is orthogonal to the span of $\mathbb{X}$.

□ **D. $\hat{\mathbb{Y}}$ has a shape of $n \times 1$.**

Solution:
Option A is incorrect. $\mathbb{Y}$ is $n \times 1$ (one item per row of $\mathbb{X}$) while $\hat{\theta}$ is $p \times 1$ (one item per feature of $\mathbb{X}$).

Option B is correct. This is a core property of OLS.

Option C is incorrect. $\hat{\mathbb{Y}}$ is within the span of $\mathbb{X}$.

Option D is correct. $\hat{\mathbb{Y}}$ needs to have one prediction for each row of $\mathbb{X}$.

(c) [4 Pts] Mir has some troubles with his OLS model. For the following subparts, **assume each scenario happens independently of one another.**

(i) Mir calculates the ideal weight vector $\hat{\theta}$ which minimizes MSE. He reruns OLS and finds a different weight vector $\hat{\theta}'$, which yields the same MSE. Mir is confident that this is the lowest possible MSE an OLS model can achieve. What is most likely the issue here?

○ **A. One or more of the features is a linear combination of other features.**

○ B. $\hat{\mathbb{Y}}$ is not a linear combination of the columns of $\mathbb{X}$.

○ C. The prediction vector $\hat{\mathbb{Y}}$ is not in the span of the design matrix.

○ D. The model does not have an intercept term.

Solution: If the feature matrix is not full column rank (i.e., one or more features is a linear combination of other features), then there is not a unique optimal solution for OLS.

(ii) Mir runs OLS but finds all his residuals are negative. What is most likely the issue here?

○ A. One or more of the features is a linear combination of other features.

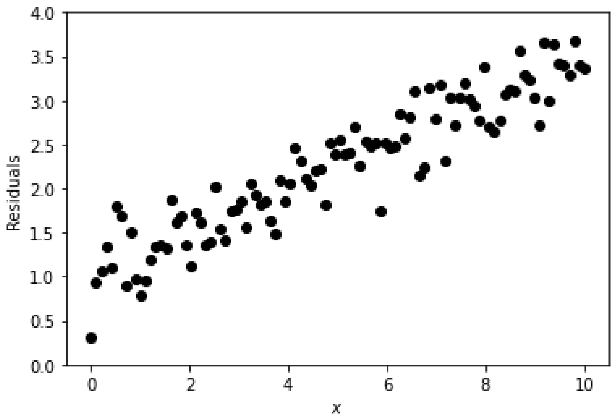
○ B. $\hat{\mathbb{Y}}$ is not a linear combination of the columns of $\mathbb{X}$.

○ C. The prediction vector $\hat{\mathbb{Y}}$ is not in the span of the design matrix.

○ **D. The model does not have an intercept term.**

Solution: OLS is only guaranteed to have the residuals sum to zero if an intercept term is present.

(d) [2 Pts] Which conclusions could you make from the following residual plot resulting from a linear model? **Select all that apply.**



□ A. The model is consistently overpredicting.

□ **B. The model is consistently underpredicting.**

□ **C. As x increases, the model performs worse.**

□ D. A linear model works well for this data.

Solution:
Option A is incorrect. We see that almost all the residuals are above 0. The formula for residuals is: actual y - prediction $\hat{y}$, so a positive residual represents an underprediction.

Option B is correct for the previously listed reason.

Option C is correct. As x increases, we see the residuals increase as well.

Option D is incorrect. We cannot tell this from the residual plot alone.

Note: the original intention behind Option D was to focus more on the model used to generate these residuals. However, the language used for this choice could be interpreted as ”any linear model”. Students were awarded points for selecting or not selecting Option D.

You are done with the midterm! Congratulations!

Use this page to draw your favorite Data 100 moment!