Data C100/200, Midterm

Fall 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 **66 points** spread out over **7 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 you should **select one choice** for questions with **circular bubbles**. There is always at least one correct answer. Please **fully** shade in the circle to mark your answer. For all math questions, **please simplify your answer**. Please also **show your work** if a large box is provided. For all coding questions, you may use commas and/or one or more function calls in each blank.

For all Python questions, you may assume `Pandas` has been imported as `pd`, `NumPy` has been imported as `np`, the Python RegEx library has been imported as `re`.

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

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 Using Data to Learn Who Asked [5 points]

DATA C100 staff just finished hosting a hackathon that was open to the general public. They collected a dataset, `participant_info`, containing **all** hackathon participants' information.

Each participant gets exactly one row describing them in the dataset, with the following columns:

- `participant_id`: The participant's ID. Each participant has a randomly assigned unique ID. Each row of the dataset corresponds to one unique participant ID. (type = `numpy.int64`)
- `first_name`: The first name of the participant. (type = `str`)
- `last_name`: The last name of the participant. (type = `str`)
- `project_topic`: The project topic that the participant is interested in. Project topics can be one of: `"AI"`, `"Education"`, or `"Health"`. (type = `str`)

A sample of `participant_info` is shown below:

	participant_id	first_name	last_name	project_topic
0	4	James	Geronimo	AI
1	14	Yewen	Xu	Health
2	17	Julianna	Lee	AI
3	18	Vladyslav	Shevkunov	Health
4	19	Ella	Hammond	Education

Note: You can assume that at least one but not all UC Berkeley students and at least one non-UC Berkeley student participated in this hackathon.

(a) [1 Pt] Select the correct statement regarding the dataset `participant_info` from the following choices:

- ◯ **The column `participant_id` can serve as a primary key for this dataset.**
- ◯ There exist several columns that can serve as unique identifiers of each row, such as `participant_id` or `first_name`.
- ◯ If 30% of the data from `project_topic` column is missing, it would be best to drop rows with missing values rather than imputing them.
- ◯ The granularity of this dataset is such that each row represents one unique project topic.

Solution: The first option is correct, as the participant ID is a unique identifier for each participant, where each row of the dataset describes one participant.

The second option is incorrect, as participants can have overlapping first names.

The third option is correct, as 30% is a large proportion for the amount of missing values to make any interpolation-based imputations, and the missing data is better left dropped out.

The fourth option is incorrect, as the dataset's granularity should be that each row represents one participant.

(b) [1 Pt] Xiaorui is interested in learning what hackathon participants think about the dinner food quality. He distributed a survey about dinner food quality to participants when he happened to be on campus from 8 pm to 9 pm, right after the dinner event. Which of the following is an accurate description of this sampling method?

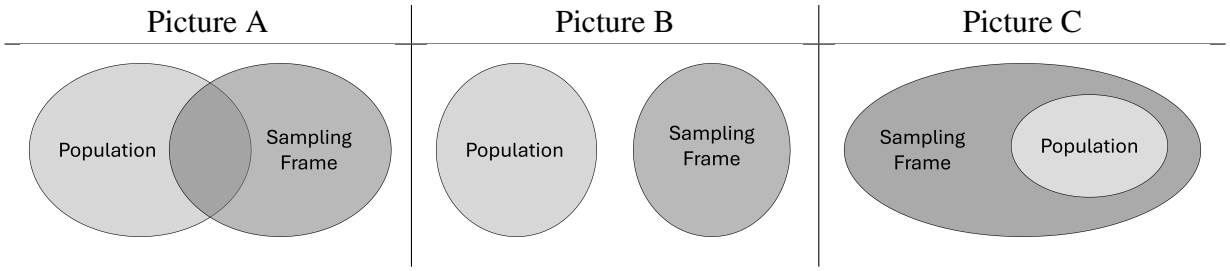
- ◯ The sampling method guarantees a representative sample of its population.
- ◯ The sampling method guarantees to prevent response bias from its respondents.
- ◯ **This method is an instance of convenience sampling.**
- ◯ This method is an instance of simple random sampling with replacement.

Solution: No sampling method prevents response bias, as participants are always subject to some degree of interference from answering with their true opinions. This can be due to multiple reasons, including but not limited to an unwillingness to answer the survey, specific social customs, and non-anonymity. The inevitability of response bias makes the second choice incorrect.

The described method is an instance of convenience sampling, as the participants are just taken within a convenient range of the sampler's availability at an arbitrary available location. Therefore, the third option is correct, and the fourth option is incorrect.

Lastly, as convenience sampling does not guarantee a representative sample of hackathon participants, the first choice is incorrect.

(c) [1 Pt] Xiaorui changes the sampling method for the survey described in the previous subpart. This time, he distributed this survey to all UC Berkeley students the day after the dinner event. Which of the following pictures best describes the relationship between the sampling frame and the population of this survey?



◯ **Picture A**

◯ Picture B

◯ Picture C

Solution: Berkeley students may be hackathon participants, as described in the introduction of this question. However, some Berkeley students may not be hackathon participants.

Given that the population of this survey is hackathon participants (open to the general public) and the sampling frame of this survey is Berkeley students (as the survey was distributed to Berkeley students), individuals in the sampling frame may or may not be part of the population. Picture A describes this correctly.

(d) [1 Pt] Suppose we want to use a visualization to capture the exact number of participants involved in each project topic. Which of the following is a suitable type of visualization?

◯ Hexplot

◯ Contour Plot

◯ **Barplot**

◯ Histogram

Solution: A hexplot visualizes the joint distribution of two quantitative variables, which is unsuitable. Thus, the first option is incorrect.

A contour plot serves a similar purpose as a hexplot, making this an inappropriate choice. So, the second option is incorrect.

A barplot can visualize quantities related to qualitative variables, making the third option correct.

A histogram is not a suitable choice here, as it visualizes the distribution of quantitative variables. Therefore, the fourth option is incorrect.

(e) [1 Pt] Evaluate the correctness of each of the following descriptions regarding `participant_info` dataframe.

◯ True ◯ **False** The variable `participant_id` is a quantitative variable.

◯ True ◯ **False** The variable `project_topic` is a qualitative ordinal variable.

Solution: The variable `participant_id` is not quantitative because performing arithmetic operations with `participant_id` doesn't grant meaningful operations. For example, adding or dividing the values of `participant_id` is meaningless. Therefore, the first choice is False.

`project_topic` is not an ordinal variable, as the IDs were randomly assigned to each participant and assume no inherent order. Therefore, the second option is False.

2 Submitting `pandas` Code at `11:59:59 PM` [16 points]

To assess the traffic that the hackathon website observed during the hackathon, Sarah collected the dataset, `portal_traffic`, with the following columns:

- `visitor_id`: The ID of the website visitor. Hackathon participants have an ID that begins with `"100"`, and judges have an ID that begins with `"101"`. A visitor is either a hackathon participant or a judge, but not both. (type = `str`)
- `hour_of_visit`: The hour that the user activity occurred at; Values must be an integer within $[0, 23]$. (type = `np.float32`)
- `activity`: The activity of the visit. The activity can be one of: `"submit"`, `"visit"`, or `"grade"`. (type = `str`)

A sample of `portal_traffic` is shown below:

	visitor_id	hour_of_visit	activity
0	101-Sammie-10053	4.0	grade
1	100-James-10258	NaN	submit
2	101-Willy-12930	8.0	visit
3	101-Nehal-12100	2.0	visit
4	100-Brie-11003	19.0	visit

(a) [2 Pts] Shreya wants to create a `DataFrame` called `portal_traffic_filtered`, such that there are only records of hackathon participants from `portal_traffic`. The resulting `portal_traffic_filtered` should include all columns from `portal_traffic` for such activities. Write one line of `pandas` code that achieves this task.

You should use at least one of the following variables when constructing your solution:

```
mask_1 = portal_traffic["visitor_id"].str.contains("100")
mask_2 = portal_traffic["visitor_id"].str.startswith("100")
mask_3 = portal_traffic["visitor_id"].str[:3].isin(["100"])
```

Solution: Several possible solutions exist by using `mask_2` and `mask_3`; for example:
`portal_traffic_filtered = portal_traffic.loc[mask_2]`

(b) [1.5 Pts] Malavikha wants a `DataFrame` of only entries with the activity `"visit"` and the columns `visitor_id` **and** `hour_of_visit`. She decides to do this by running the following code:

```
df = portal_traffic.copy()
is_visit = df["activity"]=="visit"
_____________________A_____________________
```

For each option below, determine whether filling it into blank `A` outputs the desired `DataFrame`:

◯ True ◯ **False** `df.loc[is_visit, [0, 1]]`

◯ **True** ◯ False `df[is_visit].iloc[:, [0, 1]]`

◯ **True** ◯ False `df[is_visit][["visitor_id", "hour_of_visit"]]`

> **Solution:** The first option is incorrect. Specifically, it will result in an indexing bug as `.loc` needs the column labels, but no columns are named `0` or `1`.
>
> The second option is correct, as `.iloc` supports its provided indexing, and columns at index `0` and `1` are indeed, respectively, the participant's ID and first name.
>
> The third option is correct, as it successfully uses the brackets to filter the desired rows and columns.

(c) Using `portal_traffic`, help Abby create a histogram with 24 bins for the count of visits per hour. Make the **title** of the resulting plot `visits_per_hour`. Assume that all missing entries are already dropped from the `hour_of_visit` column within the following code.

```
import matplotlib.pyplot as plt
plt._________A_________(
    portal_traffic["hour_of_visit"].dropna(), bins=24
)
plt._________B_________
```

(i) [0.5 Pts] Fill in blank `A`:

> **Solution:** `hist`

(ii) [1 Pts] Fill in blank `B`:

> **Solution:** `title("visits_per_hour")`

(d) Using `portal_traffic`, fill in the blanks to filter out the missing entries from the `hour_of_visit` column, and create a `Series` that contains the count of activities for each hour from `hour_of_visit`. Assume no missing entries are in the `activity` column.

```
(
    portal_traffic[~_________A_________._________B_________()]
        ._____________C_____________
        ._____________D_____________["activity"]
)
```

(i) [1 Pts] Fill in blank `A`:

Solution: `portal_traffic["hour_of_visit"]`

(ii) [1 Pts] Fill in blank `B` by selecting the correct choice below:

- ◯ `isnull`
- ◯ `dropna`
- ◯ `fillna`

Solution: The correct solution here is `isnull`.
Using `dropna` and `fillna` here does not result in a `Series` of boolean values that filter out all missing entries in the `hour_of_visit` column.

(iii) [1 Pts] Fill in blank `C`:

Solution: `groupby("hour_of_visit")`

(iv) [1 Pts] Fill in blank `D`:

Solution: Equivalents of `count()` are accepted. However, `size` is not accepted, as it returns a `Series` and causes indexing errors in the provided code.

(e) Fill in the blanks below to create `less_than_three_grade`, a `DataFrame` containing only rows where its corresponding `hour_of_visit` has less than 3 `grade` activities.

```
def less_than_three_grading(df):
    return df[df["activity"]=="grade"]._____A_____
less_than_three_grade = (
    portal_traffic.groupby(_____B_____)._____C_____
)
```

(i) [1 Pts] Fill in blank `A`:

Solution: `shape[0] < 3`

(ii) [1 Pts] Fill in blank `B`:

Solution: `"hour_of_visit"`

(iii) [1 Pts] Fill in blank `C`:

Solution: `filter(less_than_three_grading)`

(f) Claire wants to impute the missing entries of `hour_of_visit` with the mode of its same activity category.

For example, if any row with activity `"grade"` has a missing value for `hour_of_visit`, the missing values should be replaced by the mode of `hour_of_visit` for `"grade"` activities. Fill in the blanks to accomplish this.

```
modes_of_each_group = (
    portal_traffic.groupby_________A_________
        .agg(lambda series: series.value_counts().index[0])
)
replacement_values = modes_of_each_group[
    portal_traffic[portal_traffic["hour_of_visit"].isna()]["activity"]
].values
portal_traffic.loc[
    _________B_________._________C_________(), _________D_________
] = replacement_values
```

Hint: Below is `modes_of_each_group`, a `Series` where each row corresponds to the mode of `hour_of_visit` for each activity category. Note that the following values are for the entire `portal_traffic` dataframe:

```
activity
grade     13.0
submit     2.0
visit      8.0
Name: hour_of_visit, dtype: float64
```

(i) [1 Pts] Fill in blank `A`:

Solution: `("activity")["hour_of_visit"]` or `("activity")`

(ii) [1 Pts] Fill in blank `B`:

Solution: `portal_traffic["hour_of_visit"]`

(iii) [1 Pts] Fill in blank `C`:

Solution: `isna`

(iv) [1 Pts] Fill in blank `D`:

Solution: `"hour_of_visit"`

3 How The Tables Have Turned. . . [14 points]

To learn more about hackathon participants, Sarika collected another dataset called `other_info` with the following columns:

- `participant_id`: The participant's ID. Each participant has a randomly assigned unique ID. Each row of the dataset corresponds to one unique participant ID. (type = `numpy.int64`)
- `age`: The age of the participant. (type = `np.int64`)
- `level_of_profession`: Level of profession of the participant. (type = `str`)
- `email_address`: The email address of the participant. (type = `str`)

A sample of `other_info` is shown below:

	participant_id	age	level_of_profession	email_address
0	4	20	undergrad	james.geronimo@berkeley.edu
1	14	23	undergrad	yewen.xu@berkeley.edu
2	17	30	newgrad	julianna.lee@dataChundred.net
3	18	20	newgrad	vladyslav.shevkunov@dataChundred.net
4	19	25	newgrad	ella.hammond@dataChundred.net

(a) [1 Pt] Fill in the blank to create a `Series` with participants' email suffixes.

Note: The suffix of an email address is any text that comes after the first appearance of the character `"@"`. For example, the suffix of `"john.doe@berkeley.edu"` is `"berkeley.edu"`.

```
email_suffix = other_info["email_address"]._____A_____.str[1]
```

Fill in blank `A`:

Solution: `str.split("@")`

(b) Recall the dataset used in question 1, `participant_info`. For reference, a sample of this dataset is shown again below:

	participant_id	first_name	last_name	project_topic
0	4	James	Geronimo	AI
1	14	Yewen	Xu	Health
2	17	Julianna	Lee	AI
3	18	Vladyslav	Shevkunov	Health
4	19	Ella	Hammond	Education

where the `participant_id` column in `participant_info` matches the `participant_id` column in `other_info`.

Fill in the blanks to produce `sorted_merged_info`, a `DataFrame` where each row represents one participant with information from both `participant_info` and `other_info`. In addition, `sorted_merged_info` should be sorted by the last letter of the participant's first name in descending order and contain a new column `last_letter`.

```
participant_info.loc[:, "last_letter"] = (
    participant_info["first_name"].str.lower()._____A_____
)
sorted_merged_info = participant_info._____B_____(
    _____C_____ # Feel free to use commas here.
)._____D_____
```

(i) [1 Pts] Fill in blank `A`:

Solution: `str[-1]`

(ii) [1 Pts] Fill in blank `B`:

Solution: `merge`; an alternative solution using `join` exists with an unchanged blank B.

(iii) [1 Pts] Fill in blank `C`:

Solution: `other_info, on="participant_id"` is the most concise form. The `on=` can be expanded into `left_on=` and `right_on=`.

(iv) [1 Pts] Fill in blank `D`:

Solution: `sort_values(by="last_letter", ascending=False)`

(c) [1 Pt] Let `sorted_merged_info` be the correct result of the previous subpart.

Consider this code snippet:

```
sorted_merged_info.pivot_table(
    index="level_of_profession", columns="project_topic",
    values="age", aggfunc="median"
)
```

Which image below is the output of the above code snippet?

project_topic	level_of_profession	age
AI	gradstudent	27.0
	newgrad	24.0
	undergrad	20.0
Education	gradstudent	32.0
	newgrad	22.5
	undergrad	22.0
Health	gradstudent	32.0
	newgrad	27.0
	undergrad	21.0

Picture A

level_of_profession	gradstudent	newgrad	undergrad
project_topic			
AI	27.0	24.0	20.0
Education	32.0	22.5	22.0
Health	32.0	27.0	21.0

Picture B

project_topic	AI	Education	Health
level_of_profession			
gradstudent	27.0	32.0	32.0
newgrad	24.0	22.5	27.0
undergrad	20.0	22.0	21.0

Picture C

◯ Picture A.

◯ Picture B.

◯ **Picture C.**

Solution: Here, the code snippet produces a pivot table where the columns are the project topic, and indices (row indices) are the level of the profession. Therefore, picture C is the correct choice.

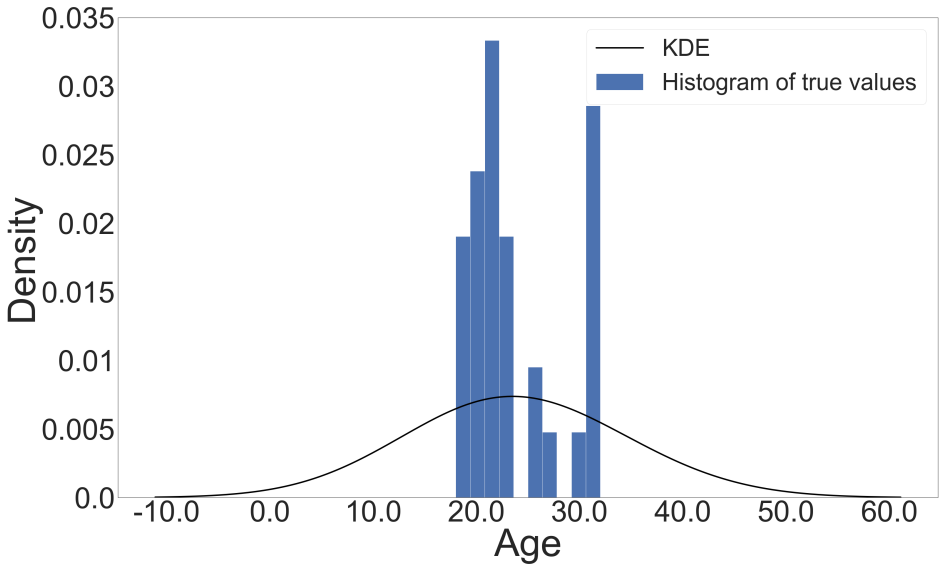
(d) [1 Pt] Rose is considering a variety of visualizations to use. For each of the following descriptions regarding visualizations, determine whether it is true or false.

◯ **True** ◯ False In contour plots, contour lines represent datapoints with the same density.

◯ True ◯ **False** The skew of a histogram refers to the ratio between its peak and number of bins.

Solution: The first option is correct, as it is the definition of contour lines.
The second option is incorrect, as the skew of the histogram refers to the direction in which its "tail" extends, not the ratio between its peak and number of bins.

(e) [1 Pt] Rose generates the following KDE plot to estimate the distribution of participants' ages:



For each of the following descriptions, determine whether it is true or false.

◯ **True** ◯ False The α of this KDE plot is too high, resulting in a high estimated density for numeric values way below 18.

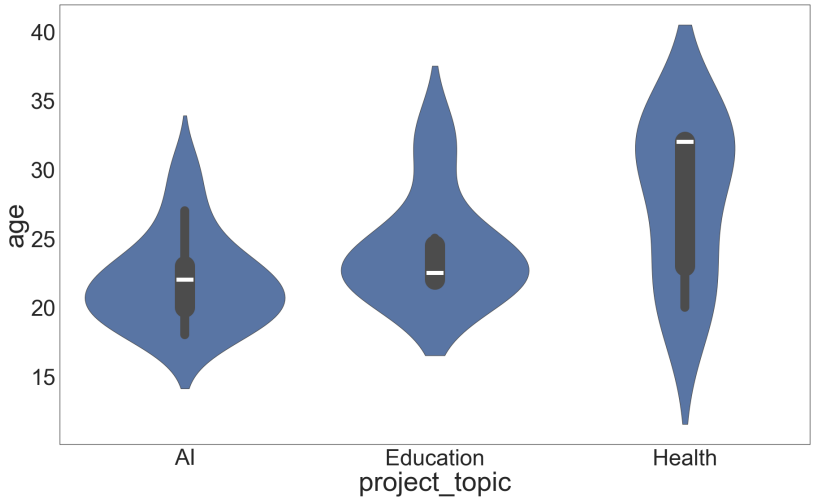
◯ True ◯ **False** Based on the histogram, the true distribution of participant age is unimodal.

Solution: The first option is correct, as an overly high α, causes over-smoothing of the estimated distribution.
The second option is incorrect, as the distribution shown by the histogram suggests a bimodal distribution with peaks at roughly 20.0 and 30.0.

(f) Angela noticed that side-by-side violin plots could help effectively visualize the variable `age` across participants of different project topics.

Recall `sorted_merged_info`, a `DataFrame` where each row represents one participant with information from both `participant_info` and `other_info`.

(i) [2 Pt] Let `sorted_merged_info` be the correct result of subpart 3(b). Write one line of `Python` code to recreate the following visualization with the `seaborn` library:



Assume the following line of code has been run:

```
import seaborn as sns
```

Solution:

```
sns.violinplot(
    data=sorted_merged_info, x="project_topic", y="age"
)
```

(ii) [1 Pts] For each of the following descriptions regarding the violinplots, determine whether it is true or false.

◯ True ◯ **False** Violinplots show the mean of each variable's distribution.

◯ **True** ◯ False Based on the violinplot, the 75th percentile of `age` for participants with project topic `"Health"` is higher than that for participants in project topic `"Education"`.

Solution: The first choice is incorrect, as the white line represents the median, not the mean.

The 75th percentile of a variable is represented by the top of its center vertical bar. As the top of the center vertical bar for `Health`'s violinplot is higher than that for `Education`'s violinplot, we can conclude the second choice is correct.

(g) [3 Pts] Provided three participants' age: $(20, 22, 23)$, use a boxcar kernel of width $\alpha = 2$ to perform KDE. What is the estimated density at age 22.5?

A boxcar kernel is formulated as follows:

$$B_\alpha(x, x_i) = \begin{cases} \frac{1}{\alpha}, & \text{if } -\frac{\alpha}{2} \leq x - x_i \leq \frac{\alpha}{2} \\ 0, & \text{otherwise} \end{cases}$$

Grading will be done based on the work you show in the box below.

Density at age 22.5 = _____

Solution: ***Calculating for the kernel of each datapoint***.
The kernel situated at datapoint with age 20 is provided as follows:

$$B_{x_i=20,\alpha=2}(x) = \begin{cases} \frac{1}{2}, & \text{if } 19 \leq x \leq 21 \\ 0, & \text{otherwise} \end{cases}$$

The kernel situated at datapoint with age 22 is provided as follows:

$$B_{x_i=22,\alpha=2}(x) = \begin{cases} \frac{1}{2}, & \text{if } 21 \leq x \leq 23 \\ 0, & \text{otherwise} \end{cases}$$

The kernel situated at datapoint with age 23 is provided as follows:

$$B_{x_i=23,\alpha=2}(x) = \begin{cases} \frac{1}{2}, & \text{if } 22 \leq x \leq 24 \\ 0, & \text{otherwise} \end{cases}$$

Aggregating the kernel of each datapoint.
The next step is to sum all kernels together, which provides the following pre-normalized piecewise function:

$$B_{\text{prenormalized}} = \begin{cases} \frac{1}{2}, & x \in [19, 22) \\ 1, & x \in [22, 23] \\ \frac{1}{2}, & x \in (23, 24] \end{cases}$$

Normalizing the final result.
Finally, remember to normalize the calculation outcome, as the estimated distribution must have its area under the curve sum of 1. Provided we have three kernels, we should divide the entire function's output value by 3:

$$B_{\text{prenormalized}} = \begin{cases} \frac{1}{6}, & x \in [19, 22) \\ \frac{1}{3}, & x \in [22, 23] \\ \frac{1}{6}, & x \in (23, 24] \end{cases}$$

Therefore, the density at age 22.5 is $\frac{1}{3}$.

4 Pur+egex! [6 points]

To alleviate participants' stress, Aneesh decided to host cat-petting sessions during the hackathon. People provided feedback for this event, and Aneesh is trying to process this text data.

For subparts (a) and (b), you will be provided a table with the following format:

Match all of these below	Match none of these below
✓`case 1`	✗`case 3`
✓`case 2`	✗`case 4`

You will be asked to provide patterns that match strings in the left column after the ✓and not match strings in the right column after the ✗. For example, for the above table, you should provide a pattern that matches `case 1` and `case 2` but not `case 3` and `case 4`.

(a) Aneesh decides to do the following RegEx exercises before processing feedback.

(i) [2 Pts] In the following blank, write a RegEx pattern that only matches words that contain the substring `"cat"` and does not contain any uppercase letters or space characters.

Fill in the blank with only the RegEx pattern, **do not make your solution a raw string**. A raw string is in the format of `r"_____"`.

Match all of these below	Match none of these below
✓`cats`	✗`Cat`
✓`concatenate`	✗`CA Tacoma`
✓`10cats10`	✗`CATcatCAT`

Solution: `[^\sA-Z]*cat[^\sA-Z]*`

(ii) [1 Pts] Assume the correct answer to the previous question is stored in a raw string `pat`, and `cat_responses` is a `Series` of `string` objects. Write a line of `pandas` code that replaces any word that matches `pat` with `"dog"`.

Solution: `cat_responses.str.replace(pat, "dog", regex=True)`

(b) [2 Pts] Here is the provided table:

Match all of these below
✓`total cats petted:60,102,305`
✓`my cats' names are:amy,baron,carol`
✓`the current time is 20:59`

For each of the following RegEx patterns, determine if it is true that they fully match all cases listed in the "Match all of these below" column.

◯ **True** ◯ False `.*:\w+(,\w+)*`

◯ **True** ◯ False `.*:[A-Za-z\d]+(,[A-Za-z\d]+)*`

◯ True ◯ **False** `.*:\w+(,\w+)+`

◯ True ◯ **False** `\D*:\w+(,\w+)+`

Solution: Per the reference sheet, `\w` matches any word character, equivalent to the character class pattern `[a-zA-Z0-9_]`.

For the first choice, it captures any string that ends with whatever matches the latter pattern `:\w+(,\w+)*`, which is a colon directly followed by a comma-separated list of word-character sequences or just one word-character sequence. This fits what all cases describe, so the first choice is correct. By the same logic, the second choice is correct.

The third and fourth choices need commas to appear after the colon and some sequence of word characters, which causes them to fail case (c). Therefore, both of the third and fourth choices are incorrect.

(c) [1 Pt] Given the provided variables:

```
case = "Can you let the mouse go, Jordan?"
pattern = r"\w?$"
```

What is the output of the following function call?

```
re.search(pattern, case)[0]
```

Select the correct option from below.

◯ `"Jordan"`

◯ `"n"`

◯ `"n?"`

◯ `""`

Solution: The pattern asks for one or no word character preceding the end of the line. However, the character that precedes the end of the line is `?`, which can only be matched by an escaped special character `\?`. Therefore, the fourth choice is correct: the pattern matches for a substring with zero word characters.

5 Spending a Night in Jupyter [9 points]

It is unhealthy to sleep for less than 8 hours. Regardless, several groups tried working on their projects overnight. Sam collected a dataset to investigate whether the number of hours worked overnight is influential to the project's final grade:

- `team_name`: The team name of the submitted project. (type = `str`)
- `project_score`: The final grade of the submitted project is out of 30 points. It must be a positive value. (type = `np.float32`)
- `hours_spent`: Number of hours the team spent on their project. It must be a positive value. (type = `np.float32`)
- `proportion_overnight`: The proportion of time spent between 12 AM and 8 AM on their project, between 0 and 1 inclusively. (type = `np.float32`)

A sample of the dataset is shown below:

	team_name	project_score	hours_spent	proportion_overnight
0	Pandeys	24.0	10.5	0.30
1	Seabirth	18.0	9.0	0.35
2	Numthon	28.0	11.0	0.27

(a) [1 Pt] Nikhil wants to use a constant model to predict `proportion_overnight`. What should Nikhil pay attention to before fitting the constant model? Select the correct description from below:

- ◯ For a constant model, the L1 loss is generally more sensitive to outliers than the L2 loss.
- ◯ For a constant model, the optimal solution to L1 loss is always the mode of the predicted variable's actual values.
- ◯ **For a constant model, the optimal solution to L2 loss is always the mean of the predicted variable's actual values.**
- ◯ Instead of L1 loss, Nikhil can choose to use a loss function: $\mathcal{L}(y, \hat{y}) = y - \hat{y}$.

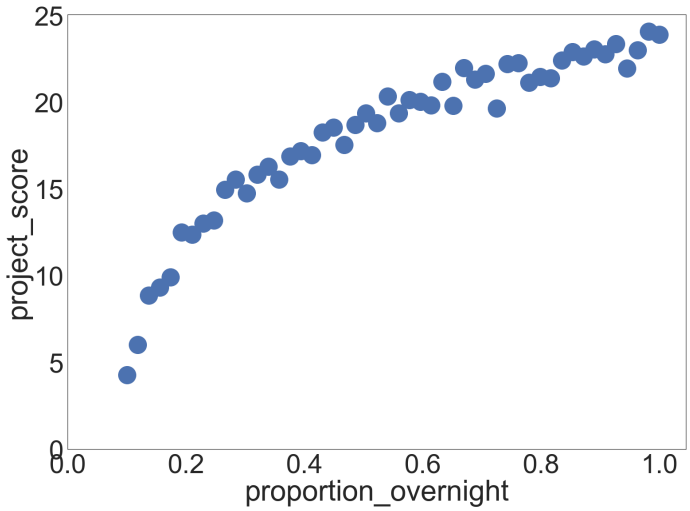
Solution: The first choice is incorrect. The L1 loss is generally less sensitive to outliers than the L2 loss in fitting a constant model.

The second choice is incorrect, as the optimal solution to L1 loss should be the median of the predicted variable's actual values.

The third choice is correct. The concrete proof was done in lectures.

The fourth choice is incorrect. For any positive real number k, Nikhil's loss function punishes the predicted values $\hat{y} = y - k$ but not $\hat{y} = y + k$. An accurate prediction $\hat{y} = y$ obtains a loss value of 0. However, the loss function for an inaccurate prediction $\hat{y} = y + k$ is negative, which means this loss function provides more punishment for the accurate prediction than the inaccurate one. This loss function is, therefore, illegitimate, as it should've punished any inaccurate predictions more than an accurate prediction.

(b) [2 Pts] Dan builds a linear regression model to predict `project_score` using the variable `proportion_overnight`, and made a scatterplot as shown below:



For each of the following combination of transformations, determine whether applying that combination alone would linearize the data:

◯ True ◯ **False** Applying a cubic root to the `proportion_overnight` variable, and a logarithmic transformation to the `project_score` variable.

◯ True ◯ **False** Raising the `proportion_overnight` variable to the third power, and a logarithmic transformation to the `project_score` variable.

◯ **True** ◯ False Applying a cubic root to the `proportion_overnight` variable, and raising the `project_score` variable to the second power.

◯ True ◯ **False** Raising the `proportion_overnight` variable to the third power, and raising the `project_score` variable to the second power.

Solution: To linearize the data, we need to reduce the power of the x variable and increase the power of the y variable. Therefore, the third choice is correct.

(c) [2 Pts] We are provided the following view of our dataset:

	project_score	hours_spent
mean	22.0	8.0
variance	16.0	1.0

The correlation between `project_score` (y) and `hours_spent` (x) is 0.5. Using Simple Linear Regression (SLR), write the model's equation predicting `project_score` using `hours_spent`. **Grading will be done based on the work you show in the box below.**

The model equation: _____

Solution: Recall from the midterm reference sheet that the formula for simple linear regression solutions is:

$$\begin{cases} \hat{\theta}_0 &= \bar{y} - \hat{\theta}_1 \bar{x} \\ \hat{\theta}_1 &= \frac{r\sigma_y}{\sigma_x} \end{cases}$$

for the model equation $\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x$.

The standard deviation of a variable is the square root of its variance. Therefore, we may conclude that $\sigma_x = \sqrt{1} = 1$ and $\sigma_y = \sqrt{16} = 4$.

Therefore, let us obtain $\hat{\theta}_0$ and $\hat{\theta}_1$ by plugging in $\sigma_x = 1$, $\sigma_y = 4$, $r = 0.5$, $\bar{x} = 8$, $\bar{y} = 22$:

$$\hat{\theta}_1 = \frac{1}{2}\frac{4}{1} = 2$$
$$\hat{\theta}_0 = 22 - 2 \times 8 = 6$$

The model equation is hence $\hat{y} = 6 + 2x$.

(d) [1 Pt] Minoli proposes an alternative loss function for the linear regression problem between `project_score` (y) and `hours_spent` (x) as:

$$\mathcal{L}(y_i, \hat{y}_i) = \begin{cases} 3|y_i - \hat{y}_i|, & \text{if } y_i > \hat{y}_i \\ |y_i - \hat{y}_i|, & \text{otherwise} \end{cases}$$

Select the correct description regarding Minoli's proposed loss function.

- ◯ Minoli's loss function punishes underprediction and overprediction equivalently.
- ◯ **Minoli's loss function punishes underprediction more than overprediction.**
- ◯ Minoli's loss function is concave.
- ◯ The optimal solution to Minoli's loss function cannot be found.

Solution: The situation where $y_i > \hat{y}_i$ is underprediction, where the situation $y_i < \hat{y}_i$ is overprediction. The loss function's value is tripled where underprediction occurs. Therefore, the second option is correct. By that logic, the first option is incorrect.

The third option is incorrect, as the convexity of this function can be proved using Jensen's inequality.

The fourth option is incorrect, as we can find the optimal solution to this loss function via calculus.

(e) [3 Pts] Minoli wants to fit a constant model with equation $\hat{y}_i = \theta$, using the loss function $\mathcal{L}$ from part (d). What is the derivative of the following empirical risk function $\mathcal{R}$ with respect to θ?

$$\mathcal{R}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, \hat{y}_i)$$

Let y_+ be the count of values y_i where $y_i > \hat{y}_i$, and y_- be the count of y_i where $y_i \leq \hat{y}_i$. Express and simplify your final answer in terms of y_+ and y_-. **Grading will be done based on the work you show in the box below.**

$\frac{\mathrm{d}\mathcal{R}}{\mathrm{d}\theta}$ = _____

Solution: Because $\mathcal{R}$ is convex, its critical point should be the global minimum of the function. So, let us find the critical point by first computing $\frac{\mathrm{d}\mathcal{R}}{\mathrm{d}\theta}$.

Whenever $y_i > \hat{y}_i$, the derivative of the function is -3.
Meanwhile, whenever $y_i < \hat{y}_i$, the derivative of the function is 1.
We do not mind the case $y_i = \hat{y}_i$, where the derivative is 0.
So, integrating the above information, we deduce that:

$$\frac{\mathrm{d}\mathcal{R}}{\mathrm{d}\theta} = \frac{1}{n}(y_- - 3y_+)$$

6 Ordinarily, or Legendarily Satisfied? [7 points]

Rayna uses the following dataset to train a linear regression model for hackathon participant satisfaction. All variables in the dataset are integers (type = `np.int64`) between 0 and 5 inclusively:

- `overall`: The overall satisfaction of a participant.
- `food`: The participant's satisfaction regarding the provided food.
- `booth`: The participant's satisfaction regarding external booths.

The **full dataset** is shown below:

	overall	food	booth
Datapoint 0	4	2	0
Datapoint 1	2	0	1
Datapoint 2	4	0	0

(a) [2 Pts] Calculate the coefficients of an Ordinary Least Squares (OLS) model to predict `overall` with all the other features. Do not include a bias column in your design matrix. **Grading will be done based on the work you show in the box below.**

Hint: $\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}^{-1} = \begin{bmatrix} \frac{1}{a} & 0 \\ 0 & \frac{1}{b} \end{bmatrix}$

$\hat{\theta}_{\text{food}}$ = _____, $\hat{\theta}_{\text{booth}}$ = _____

Solution: We can begin by finding some components of our arithmetic:

$$\mathbb{X} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \mathbb{Y} = \begin{bmatrix} 4 \\ 2 \\ 4 \end{bmatrix}$$

Then,

$$(\mathbb{X}^T\mathbb{X})^{-1} = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{bmatrix}$$

And,

$$\mathbb{X}^T\mathbb{Y} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 4 \\ 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 8 \\ 2 \end{bmatrix}$$

Multiplying all components together:

$$\begin{aligned} (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbb{Y} &= \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 8 \\ 2 \end{bmatrix} \\ &= \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \vec{\hat{\theta}} \end{aligned}$$

Therefore, $\hat{\theta}_{\text{food}} = 2$, $\hat{\theta}_{\text{booth}} = 2$.

We have now collected more datapoints and will use this larger dataset for the following subparts.

(b) [2 Pts] Rayna processes the dataset before fitting the model. For each suggestion, determine whether it is true or false that Rayna should perform it before applying OLS to the provided dataset.

◯ **True** ◯ False If the design matrix is an n-by-p matrix, then it must be that $\mathbb{Y} \in \mathbb{R}^n$.

◯ **True** ◯ False Rayna should check if the resulting design matrix has more columns than rows. If so, OLS does not have a unique solution.

◯ True ◯ **False** We must have $\text{rank}(\mathbb{X}) = 1$ for OLS to have a unique solution.

◯ True ◯ **False** Rayna should check if the rank of $\mathbb{X}^T\mathbb{X}$ is equal to the rank of $\mathbb{X}$. If so, OLS cannot find a unique solution.

Solution: The first choice is correct, as we need the same number of observations and actual predicted variable values (n) for computing an OLS solution.

The second option is correct, as having more columns than rows guarantees the design matrix to have a nontrivial nullspace (which makes $\mathbb{X}^T\mathbb{X}$ non-invertible), which prevents a unique solution for ordinary least squares.

The third option is incorrect, as there are no limits in the derivation of ordinary least squares. The solution for the first subpart directly refutes these options, as the first subpart concerns a rank-3 design matrix.

The fourth option is incorrect, as having the rank of $\mathbb{X}^T\mathbb{X}$ equal to the rank of $\mathbb{X}$ results in a unique solution.

(c) [2 Pts] Rayna produces the OLS model after some work. For each option, determine whether the option holds true **only if** a **unique** OLS solution exists.

◯ True ◯ **False** The dot product of residuals $\vec{e}$ and all features is 0.

◯ **True** ◯ False The design matrix $\mathbb{X}$ has full column rank.

◯ **True** ◯ False For the design matrix $\mathbb{X}$, $\mathbb{X}^T\mathbb{X}$ is invertible.

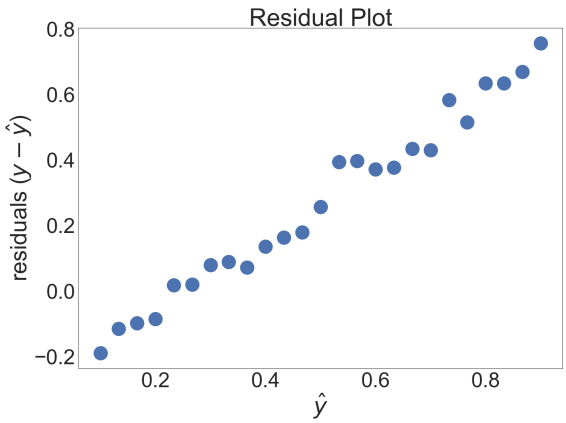
◯ True ◯ **False** For the design matrix $\mathbb{X}$, the equation $\mathbb{X}^T\mathbb{X}\theta = \mathbb{X}^T\mathbb{Y}$ holds.

Solution: The first choice is incorrect, as having a design matrix $\mathbb{X} = 0$ also allows it to hold, and the design matrix $\mathbb{X} = 0$ does not have a unique OLS solution.

The second and third choices are synonymous and correct, as they both promise a unique solution θ for the normal equation.

The fourth option is incorrect, as having $\mathbb{X} = 0$ also allows the described equation to hold.

(d) [1 Pt] Rayna generated a residual plot from her linear regression model, as shown below:



For each of the following descriptions regarding this plot, mark whether it is true or false.

◯ True ◯ **False** The above plot shows a trend of overprediction for higher $\hat{\mathbb{Y}}$ values.

◯ **True** ◯ False In an ideal residual plot, there should not be a particular pattern with systematic underpredictions or overpredictions.

Solution: The first option is incorrect, as the plot indicates that the real value is higher than the predicted value for higher values of $\hat{\mathbb{Y}}$, showing a trend of underprediction.

The second option is correct, as residual plots with patterns of overprediction and underprediction indicate that its models are systematically inaccurate. You may see more of this description from Lecture 11, Slide 63.

7 What is The Optimal Amount of Free Food? [8 points]

We chose not to offer free food to reduce the cost of hosting a hackathon, but maybe that's not the most optimal choice. Jessica tries to answer this concern via an optimization process.

(a) Jessica created the following loss function $\mathcal{L}$, which describes the overall operational cost of providing free food:

$$\mathcal{L}(\theta_f, \theta_c, \theta_s) = 25\theta_f\theta_c - \ln(\theta_s)$$

Here, θ_f, θ_c, and θ_s respectively represent the amount of free food, car transportation cost of food, and subsidies for food costs. For this subpart, **grading will be done based on the work you show in the boxes below.**

(i) [3 Pts] Express the gradient for $\mathcal{L}$ in terms of θ_f, θ_c, θ_s. Please simplify your answer.

$$\nabla\mathcal{L} = \Big[\qquad \Big]^T$$

Solution: Let us calculate the partial derivatives of $\mathcal{L}$, then assemble them as the gradient:

$$\frac{\partial\mathcal{L}}{\partial\theta_f} = 25\theta_c$$
$$\frac{\partial\mathcal{L}}{\partial\theta_c} = 25\theta_f$$
$$\frac{\partial\mathcal{L}}{\partial\theta_s} = -\frac{1}{\theta_s}$$
$$\nabla\mathcal{L} = \begin{bmatrix} 25\theta_c & 25\theta_f & -\frac{1}{\theta_s} \end{bmatrix}^T$$

(ii) [3 Pts] Ignoring the previous part, Jessica found the following current values for parameters and gradient at iteration t:

$$\theta_f^{(t)} = 0.5, \theta_c^{(t)} = -0.5, \theta_s^{(t)} = 1$$
$$\nabla\mathcal{L}(\theta_f, \theta_c, \theta_s) = \begin{bmatrix} 0.5 & 1.5 & -0.5 \end{bmatrix}^T$$

With a learning rate of $\alpha = 2$, calculate the value of each variable at iteration $t + 1$ of gradient descent on $\mathcal{L}$.

$\theta_f^{(t+1)}$ = _____, $\theta_c^{(t+1)}$ = _____, $\theta_s^{(t+1)}$ = _____

Solution: Recall that (with provided $\alpha = 2$), the gradient descent update rule is:

$$\vec{\theta}^{(t+1)} \leftarrow \vec{\theta}^{(t)} - \alpha \nabla \mathcal{L}(\vec{\theta}^{(t)})$$

Provided that the gradient above, we update the parameter as such:

$$\begin{aligned}\vec{\theta}^{(t+1)} &= \begin{bmatrix}0.5 & -0.5 & 1\end{bmatrix}^T - 2\begin{bmatrix}0.5 & 1.5 & -0.5\end{bmatrix}^T \\ &= \begin{bmatrix}-0.5 & -3.5 & 2\end{bmatrix}^T = \begin{bmatrix}\theta_f^{(t+1)}, \theta_c^{(t+1)}, \theta_s^{(t+1)}\end{bmatrix}\end{aligned}$$

(b) [2 Pts] What possible aspect(s) of gradient descent may lead to a suboptimal solution? For each option, determine whether it is true or false.

- ◯ **True** ◯ False The starting point of the gradient descent algorithm.
- ◯ **True** ◯ False The choice of gradient descent (batch, mini-batch, or stochastic gradient descent).
- ◯ **True** ◯ False Using a fixed number of iterations.
- ◯ **True** ◯ False The learning rate of gradient descent.

Solution: The first option is correct, as gradient descent can converge on different results for functions with multiple local minima.

The second option is correct, as the inherent randomness of mini-batch and stochastic gradient descent can either lead to parameter values with a higher loss than what batch gradient descent can find, or help it achieve a smaller local minimum by escaping out of other ones. This is especially true for functions with more than one local minimum.

The third option is correct, as not all functions can be optimized to their global minimum by a fixed number of gradient descent iterations.

The fourth option is correct, as the learning rate can cause problematic behavior in gradient descent, such as oscillation between two parameter values.

You are done with the midterm! Congratulations!

Use this page to draw your favorite Data 100 moment!