Data 6
Summer 2026

Abby Brooks-Ramirez
Final Exam

Solutions last updated: Tuesday, August 25, 2026

Your name: _____

Your student ID: _____

Your Berkeley email: _____

Your room location: _____

Student ID of the person to your left: _____

Student ID of the person to your right: _____

You have 110 minutes. There are 5 questions of varying credit. (82 points total)

Question:	HC	1	2	3	4	5	Total
Points:	1	10	10	34	27	0	82

For questions with **circular bubbles**, you may select only one choice.

- ○ Unselected option (Completely unfilled)
- ◌✓ Don't do this (it will be graded as incorrect)
- ● Only one selected option (completely filled)

For questions with **square checkboxes**, you may select one or more choices.

- ■ You can select
- ■ multiple squares
- ☑ (Don't do this)

Anything written outside the answer boxes or ~~crossed out~~ will not be graded. If you write multiple answers, your answer is ambiguous, or the bubble/checkbox is not entirely filled in, we will grade the worst interpretation. For coding questions with blanks, you may write at most one statement per blank, unless otherwise stated.

> As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others. I will follow the rules of this exam.

Honor Code (HC): I have read and agree to the honor code above.

(1 point) Sign your name: _____

Q1 *What Would Python Do (WWPD)* **(10 points)**

Q1.1 Consider the below code:

```
1  n = 5
2  def bump(n):
3      n = n * 2
4      return n
5  bump(n)
6  print(n)
```

Q1.1.1 (1 point) On line 5, what does **bump(n)** evaluate to? If Python will error, select "Error".

○ **5** ● **10** ○ **20** ○ **None** ○ Error

Q1.1.2 (1 point) After the code is run, what is printed? If Python will error, select "Error".

● **5** ○ **10** ○ **20** ○ **None** ○ Error

Q1.2 Recall that **print** displays arguments to the screen but returns **None**.

```
1  n = 7
2  n //= 2
3  flag = print(n) is None
4  n -= int(flag)
```

Q1.2.1 (1 point) What does **flag** evaluate to on line 3? If Python will error, select "Error".

● **True** ○ **False** ○ **3.5** ○ **3** ○ **None** ○ Error

Q1.2.2 (1 point) After the code is run, what is printed? If Python will error, select "Error".

○ **7** ● **3** ○ **3.5** ○ **True** ○ **None** ○ Error

Q1.2.3 (1 point) After the code is run, what is the value of **n**? If Python will error, select "Error".

○ **7** ○ **3** ● **2** ○ **3.5** ○ **None** ○ Error

Q1.3 (1 point) What does the expression evaluate to?

```
1  max(abs(-7), 3) * min(4, 6) // 5
```

Solution: 5

Q1.4 (2 points) Recall that **print** displays arguments separated by spaces (' '), and that \ acts as an escape character.

```
1 langs = make_array("R", "SQL", "Python")
2 fav = "I \"heart\" " + langs.item(-1)
3 print(fav, "and", langs.item(0) + str(len(langs)))
```

When the code is run, what is printed?

- ○ **I \"heart\" Python and R3**
- ○ **I "heart" SQL and R3**
- ● **I "heart" Python and R3**
- ○ None of these
- ○ **I "heart" Python and R 3**

Q1.5 Consider the **check** function:

```
1 def check(x):
2     return len(x) and x[0]
```

What is the result of evaluating each call below? If Python will error, select "Error".

Q1.5.1 (1 point) **check("data")**

- ○ **True**
- ● **"d"**
- ○ **None**
- ○ **False**
- ○ **4**
- ○ Error

Q1.5.2 (1 point) **check(make_array(0, 5))**

- ○ **True**
- ○ **4**
- ○ **None**
- ○ **False**
- ● **0**
- ○ Error

SID: _____

Q2 *Potpourri* **(10 points)**

Q2.1 (1 point) **pivot** allows you to aggregate by 1 or more categorical variables.

○ True ● False

Q2.2 (1 point) **group** allows you to aggregate by 1 or more categorical variables.

● True ○ False

Q2.3 (1 point) Bag of Words can be used to create vectors (numerical representations) of documents.

● True ○ False

Q2.4 (1 point) All **while** loops can be written as **for** loops.

○ True ● False

Q2.5 (2 points) An array is defined as **arr** = **make_array(5, 37, 39, 14, 2)**.

The p-th percentile of an array is the smallest value that is at least as large as p% of the values. What is the 25th percentile of **arr**?

○ 2 ○ 14 ○ 39

● 5 ○ 37 ○ None of the above

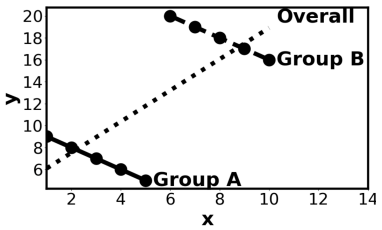
Q2.6 (1 point) In a left-skewed distribution, the mean is typically greater than the median.

○ True ● False

Q2.7 (1 point) A kitchen thermometer always reads 10°F higher than the true temperature. This thermometer is reliable but not valid.

● True ○ False

Q2.8 (1 point) Consider the graph below, which plots the overall trend of a dataset and the trends of the dataset when split into groups A and B.



True or False: The graph above indicates the presence of Simpson's Paradox.

● True ○ False

Q2.9 (1 point) Suppose you want to learn more about the ways college students study. You share a voluntary survey with 500 randomly sampled student emails at your school, asking each student to describe, in their own words, what a typical study session looks like for them. This is an example of what type of research?

○ Quantitative Research ● Qualitative Research

Q3 *By the Book* **(34 points)**

OCLC, a library organization representing over 16,000 member libraries worldwide, compiled the following dataset of the top 500 novels of all time. Some of the dataset's variables, such as **genre**, reflect the collective expertise of OCLC's member libraries, while others—such as **avg_rating**—were pulled from public sources like Goodreads, a website where users rate and review books.

rank	title	author	year	genre	avg_rating	num_ratings
463	Nights in Rodanthe	Nicholas Sparks	2002	romance	3.85	172005
124	The Magician's Nephew	C.S. Lewis	1955	fantasy	4.05	537714
286	Mockingjay	Suzanne Collins	2010	scifi	4.09	3203569
454	The Chocolate War	Robert Cormier	1974	bildung	3.49	44750

Table 1: Four rows from the **novels** table (500 rows total).

Variable descriptions:
- **rank**: Numeric rank of text according to OCLC's top 500 novels list (integer).
- **title**: Title of text(string).
- **author**: Author of text (string).
- **year**: Year of first publication of text (integer).
- **genre**: Genre of text (string).
- **avg_rating**: Average star rating for a text on Goodreads (float).
- **num_ratings**: Total number of ratings for a text on Goodreads (integer).

Q3.1 (1 point) What is the variable type of **author**?

○ Discrete numerical ○ Ordinal categorical

○ Continuous numerical ● Nominal categorical

Q3.2 (1 point) What is the variable type of **avg_rating**?

○ Discrete numerical ○ Ordinal categorical

● Continuous numerical ○ Nominal categorical

Q3.3 (1 point) Which of the below would be best for visualizing the relationship between **year** and **num_ratings** in the **novels** table?

○ Bar chart ● Scatter plot ○ Box plot

○ Histogram ○ Line plot ○ None of the above

Q3.4 (1 point) Which of the below would be best for visualizing the number of books in each genre?

● Bar chart ○ Scatter plot ○ Box plot

○ Histogram ○ Line plot ○ None of the above

Q3.5 (2 points) You show your friend the **novels** table you've been working with. Your friend wants to use the table to answer the question: "In which years did Goodreads users read the most novels?" Can this question be answered using the **novels** table? In 1 to 2 sentences, explain why or why not.

It is not possible to answer this question with the novels table for a variety of reasons, one of which being that the table only includes information for Goodreads users for novels in the table, not Goodreads users *overall*.

You and your friend want to explore the total number of Goodreads reviews each author has received, summed across all of their books in **novels**. Fill in the code below to create a table called **authors** with two columns: **author** and **total_reviews**, where **total_reviews** is the sum of **num_ratings** across all of that author's books.

author	total_reviews
Agatha Christie	2230891
Alan Paton	74518
Albert Camus	1446463
Aldous Huxley	1843997
Aleksandr Isaevich Solzhenitsyn	113811

Table 2: First five rows of the **authors** table.

```
authors = novels._____(a)_____._____(b)_____.relabeled(_____(c)_____, 'total_reviews')
```

Q3.6 (2 points) Which of the below goes in blank (a)?

- ○ **group('author', sum)**
- ● **select('author', 'num_ratings')**
- ○ **'num_ratings sum'**

Q3.7 (2 points) Which of the below goes in blank (b)?

- ● **group('author', sum)**
- ○ **select('author', 'num_ratings')**
- ○ **'num_ratings sum'**

Q3.8 (1 point) Which of the below goes in blank (c)?

- ○ **.group('author', sum)**
- ○ **select('author', 'num_ratings')**
- ● **'num_ratings sum'**

Complete the code below to reassign **authors** so that it contains only the 10 authors with the highest **total_reviews**, sorted from highest to lowest.

```
authors = authors._____(a)_____._____(b)_____(_____(c)_____)
```

Q3.9 (2 points) Fill in blank (a)

sort('total_reviews', descending=True)

Q3.10 (1 point) Fill in blank (b)

- ○ **percentile**
- ○ **item**
- ● **take**
- ○ None of the above

Q3.11 (2 points) Fill in blank (c)

np.arange(0,10)

Using **authors**, we create the following visualization:

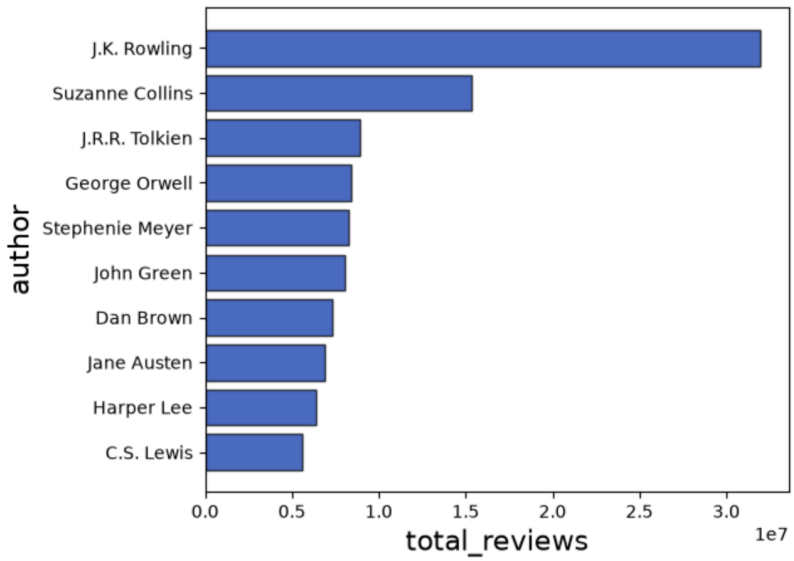


Figure 2: Visualization generated from Table 2

Q3.12 (1 point) What type of visualization is this?

- ● Bar chart
- ○ Line plot
- ○ Histogram
- ○ Box plot
- ○ Scatter plot
- ○ None of the above

Q3.13 (1 point) The value 1e7 is shorthard for 10^7 and it is used to scale large numbers. In Figure 2, which of the below is being scaled by 10^7?

- ● The x-axis
- ○ The total_reviews column
- ○ The y-axis
- ○ None of the above

Q3.14 (2 points) In 1 sentence, explain why we use 1e7 for scaling. Consider what would happen if we didn't scale in Figure 2.

The x-axis would be stretched out, making it difficult to interpret the visualization.

Questions 3.15-3.17 refer to the following code. When looking at Figure 2, your friend notices that many of the authors with the most reviews have written books within the last several decades and wonders whether "modern" novels receive more ratings on Goodreads than "classic" ones. Your friend writes the following code to create a new column, **era**, labeling each book as "modern" if it was published between 1950 and 2026 (inclusive), or "classic" otherwise. However, the code contains two bugs. For each of the two bugs below, identify the line of code responsible and write a corrected version of that line.

```
1 def split_year(year):
2     if year <= 1950:
3         return "modern"
4     else:
5         return "classic"
6
7 era_arr = authors.apply(split_year, 'year')
8 novels = novels.with_column('era', era_arr)
```

Q3.15 (3 points) **Bug 1:** A bug that will cause the code to error when run.

Q3.15.1 Identify the line number where this error occurs.

Line 7

Q3.15.2 Rewrite the line of code identified above to correct the code.

era_arr = novels.apply(split_year, 'year')

Q3.16 (3 points) **Bug 2:** A bug that will not cause an error, but produces incorrect results due to a logic mistake.

Q3.16.1 Identify the line number where this error occurs.

Line 2

Q3.16.2 Rewrite the line of code identified above to correct the code.

if year >= 1950:

Q3.17 (2 points) Your friend wants to rename the function **split_year(year)** to **type(year)**. When, if ever, would this name change cause errors later in the notebook? Limit your response to 1-2 sentences.

*Hint: think about what happens to the global scope when we define a name, and when else we might want to use **type**.*

Defining a function named **type** will overwrite the Python built in **type** function. This will then cause errors if we call **type** on an argument (such as a string, list, or any value that is not an int/float) because instead of the built in **type** behavior the name is overshadowed by the function defined above Q3.15.

Q3.18 Assume that we have correctly resolved the bugs in the previous code, and **novels** now contains a column **era**. Consider the figure below, which shows two separate distributions based on the value in the **era** column. Fill in the code below to produce the visualization shown. *You do not need to replicate the color or shading shown-it is included only because the exam is in black and white.*

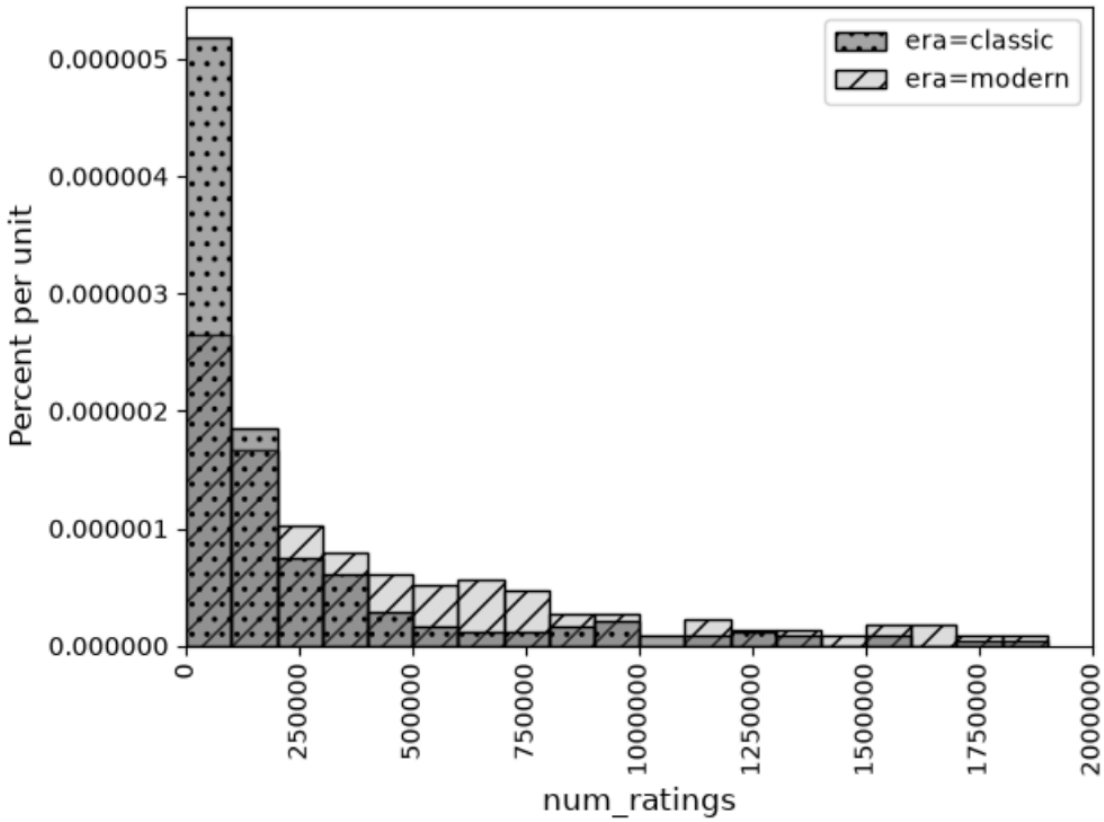


Figure 3: Visualization generated from Table 2

```
novel_bins = np.arange(0, 2000000, 100000)
novels.___(a)___(_______(b)_______)
```

Q3.18.1 (1 point) Fill in blank (a).

- ○ **barh**
- ○ **boxplot**
- ○ **scatter**
- ● **hist**
- ○ **plot**
- ○ None of the above

Q3.18.2 (2 points) Fill in blank (b).

```
'num_ratings', group='era', bins=novel_bins
```

Based on Figure 3, evaluate the following statements.

Q3.19 (1 point) Both modern and classic novels have a right skewed distribution for **num_ratings.**

- ● True
- ○ False

Q3.20 (1 point) The percentage of classic novels with fewer than 250,000 ratings is higher than the percentage of modern novels with fewer than 250,000 ratings.

● True ○ False

Q3.21 (1 point) Modern novels have fewer outliers along the x-axis.

○ True ● False

Q4 *Plot Twist!* **(27 points)**

We'll continue our exploration of **novels**. As a reminder, the **novels** table contains the following columns:

- **rank**: Numeric rank of text according to OCLC's top 500 novels list (integer).
- **title**: Title of text(string).
- **author**: Author of text (string).
- **year**: Year of first publication of text (integer).
- **genre**: Genre of text (string).
- **avg_rating**: Average star rating for a text on Goodreads (float).
- **num_ratings**: Total number of ratings for a text on Goodreads (integer).

Fill in the code below to create a dictionary **genre_counts**. The keys should be the genres present in **novels**, and the values should be the number of times that genre appears. After the code runs, **genre_counts** should evaluate to the following dictionary.

```
{'fantasy': 48,
 'romance': 33,
 'bildung': 27,
 'scifi': 21,
 'mystery': 18,
 'na': 353}
```

```
genre_counts = {}

for ___(a)___ in ___(b)___:
    if ___(c)___:
        ___(d)___
    else:
        ___(e)___
```

Q4.1 (1 point) Fill in blank (a).

genre

Q4.2 (2 points) Fill in blank (b).

novels.column('genre')

Q4.3 (2 points) Fill in blank (c).

genre in genre_counts

Q4.4 (2 points) Fill in blank (d).

genre_counts[genre] += 1

Q4.5 (2 points) Fill in blank (e).

genre_counts[genre] = 1

When looking at **genre_counts**, your friend points out that several of the genre labels are not immediately clear, such as books with a "bildung" genre. "Bildung" is a German word meaning "education" or "formation" and often refers to coming-of-age stories. You also note that the most common genre is "na", which is assigned to novels for which no genre was provided. Your friend suggests that the two of you determine a new set of genres and relabel the books.

Q4.6 You and your friend want to survey UC Berkeley students to see how they naturally refer to book genres or themes, so you can use everyday language in your relabeling. Your friend proposes the methods below.

Q4.6.1 (2 points) Which of the following violate one or more principles of the Belmont Report (Respect for Persons, Beneficence, and Justice)? Select all that apply.

- ☐ Post a link to an anonymous, voluntary survey in a public Berkeley student forum, asking students to describe books in their own words. The survey discloses the purpose of the study and clarifies anonymity and data protection procedures.
- ■ Sit in a campus library and quietly write down students' conversations about books without telling them they're being observed or recorded.
- ■ Offer a $50 gift card for completing the survey, but only to students who are declared English or Rhetoric majors.
- ■ Require students in a large introductory course to complete the survey for course credit, with no alternative assignment offered.
- ☐ Email a random sample of enrolled students, clearly stating that participation is voluntary, responses are anonymous, and data will only be reported in aggregate.

Q4.6.2 (2 points) Choose one of your answers from the previous part and explain which Belmont principle it violates and why. If it violates more than one principle, pick just one. (2–3 sentences)

Sitting in the library and recording conversations without telling students violates Respect for Persons, because it collects data from people without their knowledge or consent.

You and your friend ultimately decide to use public posts from the r/berkeley subreddit, a public social media page where UC Berkeley students often post.

Q4.7 (1 point) Which Python package below is most appropriate for extracting textual data from the posts on r/berkeley? You may assume the posts have already been "scraped"/pulled from the web.

- ● BeautifulSoup
- ○ requests
- ○ None of the above
- ○ datascience
- ○ numpy

For the next part, consider the **HTML element in lines 3-5** in the following snippet:

```
1 <div class="post">
2   <div class="username">avid-reader</div>
3   <p class="post-text">
4     Have you read Ninth House by Leigh Bardugo (dark academia vibes)?
5   </p>
6 </div>
```

Q4.8 (1 point) What is this element's tag?

- ● <p>
- ○ class = "post-text"
- ○
- ○ None of the above

After accessing public data to see what phrases students use to discuss literature, you move on to developing a labeling strategy. Your friend thinks the **novels** table doesn't have enough variables to properly label each book, and suggests adding a column containing each book's description. They propose requesting this data from an organization that maintains book information, such as Goodreads.

Q4.9 (2 points) Which of the following Python libraries could be used to access this data? Select all that apply.

- ■ **requests**
- ☐ **matplotlib**
- ☐ **BeautifulSoup**
- ☐ **datascience**
- ☐ **numpy**
- ○ None of the above

Q4.10 Your friend is able to create a new table **novel_descriptions**.

title	description
Pride and Prejudice	Elizabeth Bennet navigates courtship and class in...
1984	Winston Smith works for a totalitarian government that...
The Hunger Games	Katniss Everdeen volunteers to take her sister's...
Germinal	A young miner joins a coal workers' strike in...
Ninth House	A Yale freshman investigates a murder tied to...

Table 3: Five rows from the **novel_descriptions** table (5000 rows total).

You want to create a new table, **novels_uncoded**, containing each novel's title, author, and description; the first row of **novels_uncoded** is shown below. Fill in the code below to create the **novels_uncoded** table.

title	author	description
The Magician's Nephew	C.S. Lewis	Two children stumble into a wood between worlds...

Table 4: First row of the **novels_uncoded** table.

```
novels_uncoded = novels.select('title', 'author')._____(a)_____(_____(b)_____)
```

Q4.10.1 (1 point) Fill in blank (a).

- ○ **select**
- ○ **group**
- ● **join**
- ○ **pivot**

Q4.10.2 (2 points) Fill in blank (b).

'title', novel_descriptions

Q4.11 (2 points) You and your friend build a codebook with the labels **['futurism', 'dark academia', 'autobio', 'fantasy']** and agree on a coding strategy. You each independently label a sample of 30 books from the dataset. When you compare your labels, you calculate a Cohen's kappa of 0.20. Your friend thinks this level of agreement is good enough and suggests handing your coding strategy off to an LLM to label the remaining books. Do you agree? Justify your answer in 1–2 sentences.

Disagree—a Cohen's kappa of 0.20 indicates only slight agreement between you and your friend, meaning the coding strategy isn't well-defined or consistently applied enough for two trained humans to agree on labels, which suggests LLMs would not be able to reliably use the coding strategy.

You and your friend ask library staff for support with your research project, and together with expert librarians, you label 100 books with gold labels.

Q4.12 (2 points) With gold-standard labels for 100 books in hand, you prompt an LLM to label the remaining 400 books using the following prompt:

> "Label each book with one of the following: ['futurism', 'dark academia', 'autobio', 'fantasy'] based on the book's description."

This prompt is an example of:

- ● Zero-shot prompting
- ○ Chain-of-thought prompting
- ○ Few-shot prompting
- ○ None of the above

After the dataset has been labeled by the LLM, you randomly sample **100 rows** and compare the LLM's labels to the gold-standard labels, producing the confusion matrix below.

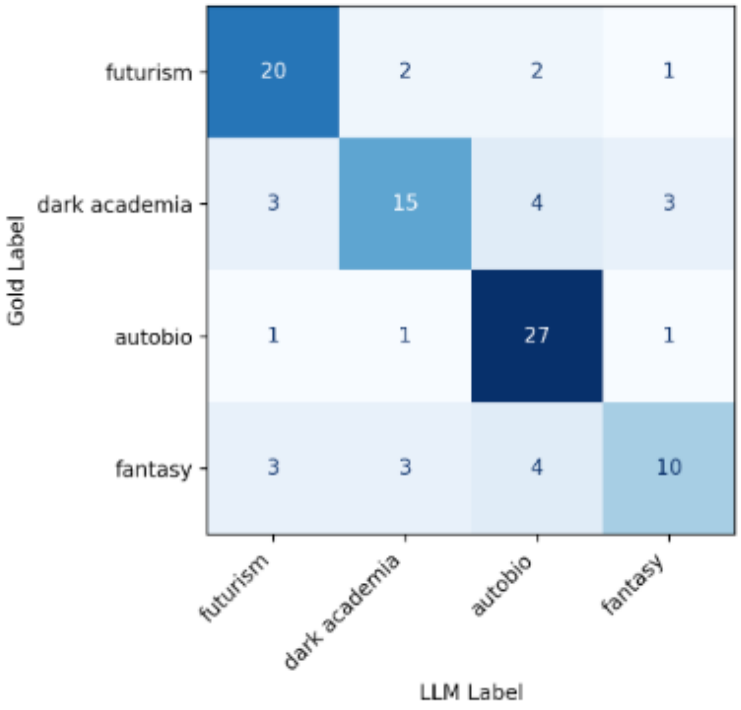


Figure 4: Confusion matrix comparing LLM labels to gold-standard labels for a random sample of 100 books.

Q4.13 (1 point) What is the class accuracy of "dark academia"? You may leave your answer as a fraction.

15 / 25

Q4.14 (1 point) What is the class accuracy of "fantasy"? You may leave your answer as a fraction.

10 / 20

Q4.15 (1 point) What is the overall accuracy of the LLM's labels? Your answer must be a percent.

(20 + 14 + 27 + 10 = 72 / 100 = 0.72

Q5 *Just for fun!* **(0 points)**

Q5.1 Draw something fun, or write a message for the staff! Or leave this blank!

Everyone had cool drawings/messages! :D

Thanks for a great semester!

—The Data 6 Summer 2026 Course Staff