Part 1: Multiple Choice Questions

For each question, select all answer options that apply. Some questions may have more than one correct answer. To penalize guessing, for each incorrect answer you select, you will lose 1 point. But the lowest score you can earn for any question or question part (i.e., 2a, 2b, etc.) is 0.

You must ensure your bubbles are **completely filled in** or you may not receive credit:
● Good!
✓ Bad! Do not use check marks for your answers!

1. A browser requests:

   ```
   https://www.youtube.com/results?search_query=networking
   ```

   Which pair correctly identifies the path and query string?

   √ **Path: `/results`; query: `search_query=networking`**
   ◯ Path: `/results?search_query=networking`; query: none
   ◯ Path: `/`; query: `results?search_query=networking`
   ◯ Path: `/results/search_query`; query: `networking`

2. A student deploys Boogle at IP address `137.110.42.18`. The HTTP server listens on port 8000 and serves the homepage at `/`. Which URL reaches that homepage?

   ◯ `http://137.110.42.18/8000/`
   ◯ `http://137.110.42.18:80/`
   ◯ `http://137.110.42.18/?port=8000`
   √ **`http://137.110.42.18:8000/`**

3. Which of the following are ways that web browsers can make requests to web servers? *(Select all that apply.)*

   √ **The user types a URL into the browser bar**
   √ **The user clicks on a link**
   √ **The web server responds with a redirect**
   √ **A script within a webpage issues a request**
   ◯ The web server sends a new page to the browser without any request having been made

4. The `Products` table contains these rows:

productID	category
1	Books
2	Books
3	Games
4	Books

   What does the following query return? Ignore row order.

   ```
   SELECT DISTINCT category FROM Products;
   ```

   ◯ Four rows: Books, Books, Games, Books
   √ **Two rows: Books, Games**

- ◯ Two rows containing the counts 3 and 1
- ◯ One row: Books

5. What happens if you run this query?

```
SELECT top_category, dwell_time FROM user_logs GROUP BY top_category;
```

- ◯ It automatically applies a `DISTINCT` operator to resolve duplicate values.
- ◯ It returns non-deterministic values of `dwell_time` for each `top_category` group.
- ◯ It throws an error because `GROUP BY` cannot be used unless at least one aggregate function is present in the `SELECT` clause.
- √ **It throws an error because `dwell_time` is neither in the `GROUP BY` clause nor in an aggregate function.**

6. When implementing PA1 Part 2, you notice that your `boogle_id` cookie keeps resetting whenever you restart the browser. What is the problem?

- ◯ You have third-party cookies disabled on your browser
- ◯ You have an adblocker installed
- ◯ You forgot to set the `SameSite` property of the cookie
- √ **You forgot to set the `Max-Age` property of the cookie**
- ◯ Nothing; all cookies will always reset when browsers restart

7. A tutorial tells you to add `<script src="jquery.com/min.js"></script>` to your page. According to lecture, loading JavaScript like this means the script can:

- √ **Read your page's cookies**
- √ **Modify any variables you store on the client**
- √ **Read most browser properties**
- √ **Send the data it collects anywhere it wants on the web**
- ◯ Run on jquery.com's web server rather than in the user's browser

8. A company wants to predict 90-day Customer Lifetime Value (CLV) to set real-time bidding limits in ad auctions. They frame this as a supervised machine learning task where the target variable is continuous net revenue. Exploratory analysis shows that 85% of newly targeted users spend $0 over 90 days, while a small fraction of high-value buyers spend over $2,000, creating extreme right-skewness.

   Which approach correctly addresses feature/target preprocessing and model evaluation for this task?

- ◯ Treat CLV as a binary target variable using One-Hot Encoding, then evaluate the linear regression model using a confusion matrix and F1 score.
- √ **Apply One-Hot Encoding to categorical predictor features, log-transform the continuous target variable ($\ln(\text{CLV} + 1)$) to stabilize variance, and evaluate model performance using regression metrics (e.g., RMSE or MAE).**
- ◯ Use K-means clustering to predict the continuous CLV target variable directly, treating cluster centroids as precision metrics for bid optimization.
- ◯ Convert the linear regression task into an unsupervised feature extraction step to automatically filter out zero-value target entries prior to inference.

9. Boogle wants to deploy a classification model to target users predicted to have a high Average Basket Value.

   - Cost to serve an ad impression: $1.00 per user.
   - Net profit from a converting user (True Positive): $50.00 after product costs and ad expense.
   - Net loss from a non-converting targeted user (False Positive): $-\$1.00$ wasted ad spend.

   You tuned the model's decision threshold to maximize the F1 score, yielding the following confusion matrix on a test cohort:

		Predictions	
		Positive	Negative
Ground truth	Positive	200	50
	Negative	100	9650

   Your coworker Zi-qing suggests that lowering the classification threshold (predicting positives more aggressively) would increase total net revenue, even if it lowers the model's F1 score. Zi-qing's reasoning is sound.

   √ **True**

   ◯ False

   ◯ Not enough information to answer

10. An SSP receives an HTTP request from a publisher and then sends an HTTP request to an ad exchange. In its interaction with the exchange, the SSP is acting as:

    √ **A client**

    ◯ A server

    ◯ Both a client and a server

    ◯ Neither a client nor a server

11. A publisher uses waterfall bidding with networks contacted in the order A, B, C, and D. The floor is $2.00 CPM. If contacted, the networks would offer $1.86, $2.35, $3.61, and $2.80 CPM, respectively. Which network wins?

    ◯ A

    √ **B**

    ◯ C

    ◯ D

Part 2: Short Answer Questions

Provide a short response to each question. **Keep your responses brief.**

12. What happens when you try to connect to an unconfigured port?

> **Solution:** Connection not established, RST in the case of TCP, connection refused... anything that suggests that you will not be getting a connection back, and why

13. What is the concept of an attention economy? What are some of its implications?

> **Solution:**
>
> Attention economy –> human attention is the scarce resource to be mined for profit.
>
> Implications include changes to design, an increased support for advertising platforms, a move towards dark patterns, information quality decline (e.g., sensationalism...), anything that highlights the changing landscape

Part 3: Implementing Twotter

14. **Building Twotter**

The Booglers are back, and this time, they're building *Twotter*, a Twitter clone. Like Boogle, it tracks each user with a first-party cookie, whose value is the user's `client_id`. Twotter wants to sell targeted ads, so it is building classifiers that infer the same nine attributes you predicted in PA1 Part 3 (gender, age, parental status, etc.). Instead of searches, its training data is users' *posts*.

Unlike PA1, Twotter's training data has *one record per post*, not one per user, for example:

```
{"client_id": "9f2c...", "text": "ugh 8am lecture again #midterms",
 "ts": "2026-10-02T15:04:11Z", "hashtags": ["midterms"]}
```

Labels, however, need to be ascribed to each user, keyed by `client_id`.

Code is shown in Python, but no particular library is assumed. Do not worry about getting syntax 100% correct—we are primarily focused on the logic. But if your syntax is unreadable, we will not be able to give you any points.

(a) The company is trying to group their users by posts, but having a hard time with some of the logic. They've written the following code, but have a few blanks:

```
posts_by_user = {}                          # client_id -> list of post texts
for post in load_posts("train.jsonl.gz"):   # one dict per post, as shown above
    key = [A]_____________________
    if key not in posts_by_user:
        posts_by_user[key] = []
    posts_by_user[key].append(post["text"])

labels = load_labels("train_labels.csv")    # client_id -> {"age": ..., ...}
users  = list(posts_by_user)
docs   = [" ".join(posts_by_user[u]) for u in users]
y_age  = [B]_____________________________
```

i. (2 points) Fill in blanks **A** and **B**:
A: the value that groups each post with the other posts by the same person.
B: the list of age labels, such that `y_age[i]` is the true age of the user whose posts are in `docs[i]`.

> **Solution:** **A:** `post["client_id"]`.
> **B:** `[labels[u]["age"] for u in users]`.

ii. (3 points) A teammate writes this for blank **B**:

```
y_age = [row["age"] for row in labels.values()]
```

`labels` keeps the rows in the order they appear in the CSV. Because every user in the CSV has at least one post, the code runs without errors, but something about this logic is wrong. Explain what goes wrong and what the dev-set macro-F1 for age will look like.

> **Solution:** + 1 order of the ages are not aligned to the client_ids, so this is basically random assignments of age to users + 2 M-F1 will go down to a trivial baseline (no better than random)

(b) A second teammate skips the grouping in part (a) altogether. Instead, they treat *each post* as one example labeled with its author's attributes, then make a random 80/20 train/dev split *of the posts* instead of the users. Their dev macro-F1 for user age is 0.91, which is excellent. But on a fresh set of new users, it falls to 0.38.

i. (2 points) Explain why the dev score was so misleading.

Solution: +1 posts included across train and dev per user, so leakage / overfitting is going on
+1 model learns user-specific phrases but fails to generalize
+1 reidentification, not generalization

ii. (2 points) In one sentence, how should the split be done instead?

Solution: +1 split by user based on user IDs (just like in part (a))

(c) (4 points) **Evaluating the classifier.** On 100 dev users, the gender classifier produces this confusion matrix (rows = true label, columns = predicted label):

	pred. male	pred. female	pred. unknown
true male	30	8	2
true female	7	33	0
true unknown	9	7	4

Compute the accuracy and the macro-F1 (to two decimal places), showing per-class precision, recall, and F1. Which class is hurting macro-F1 most, and is the problem mainly precision or recall?

Solution: Accuracy $= (30 + 33 + 4)/100 = 0.67$.
male: $P = 30/46 = 0.65$, $R = 30/40 = 0.75$, $F_1 = 0.70$.
female: $P = 33/48 = 0.69$, $R = 33/40 = 0.83$, $F_1 = 0.75$.
unknown: $P = 4/6 = 0.67$, $R = 4/20 = 0.20$, $F_1 = 0.31$.
Macro-F1 $= (0.70 + 0.75 + 0.31)/3 \approx 0.59$.
`unknown` hurts macro-F1 most. The problem is *recall*: the model seldom predicts `unknown`—so users that are unknown also get put in incorrect classes.

(d) **Knowing when you don't know.** To fix the problem from (c), Twotter predicts `unknown` whenever the model isn't confident:

```
1 TAU = 0.6
2
3 def predict_gender(model, user_vec):
4     # most likely label, and the model's probability for it
5     label, p = model.top_prediction(user_vec)
6     if [E]_____________________:
7         return "unknown"
8     return label
```

i. (1 point) Fill in blank **E**:

Solution: **E:** `p < TAU`.

ii. (2 points) As `TAU` increases, what happens to the *recall* and the *precision* of the `unknown` class? What happens if `TAU` is set close to 1?

Solution: +1 recall will rise, precision will fall +1 as Tau gets close to 1, nearly everyone becomes unknown and all other classes will fail

iii. (2 points) A user who isn't a parent posts *"Happy Mother's Day to the best mom ever, love you!!"* The model confidently labels them `parent`. Explain why a TF-IDF + logistic regression model makes this mistake. Then say what it implies if Twotter sends this label as an ad segment.

Solution: +1 TF-IDF does not have context – it just treats all words as the same
+1 Ad targeting would fail, because you would send a "parent" ad to someone who is not a parent