CSE 258, Fall 2018: Midterm

Name:	Student ID:

Instructions

The test will start at 6:40pm. Hand in your solution at or before 7:40pm. Answers should be written directly in the spaces provided.

Do not open or start the test before instructed to do so.

Note that the final page contains some algorithms and definitions. Total marks = 26

Section 1: Regression and Ranking (6 marks)

Suppose you wanted to predict *Air Quality* measurements (generally measured using an index of particulate matter called ‘PM2.5’) for a large city. Suppose you have a dataset containing thousands of hourly measurements to do so. Examples of previous measurements look like:

Observation	Date	Time	**PM2.5**	Temp (c)	Wind sp. (km/h)	wind direction	humidity
1	10/03/2004	18.00.00	150	24.4	4.1	NNW	76
2	10/03/2004	19.00.00	112	22.8	5.8	NW	84
3	10/03/2004	20.00.00	88	20.7	2.2	NW	87
4	10/03/2004	21.00.00	80	16.5	4.4	NNW	89
5	10/03/2004	22.00.00	51	15.5	2.1	W	90
6	10/03/2004	23.00.00	38	12.8	0.4	W	83
7	11/03/2004	00.00.00	31	11.8	0.6	SW	78
8	11/03/2004	01.00.00	31	10.9	1.3	S	69

1. Both the time and the date could be useful for this type of prediction. Suggest a scheme for representing the date and time, and write down the resulting features for the first two observations (2 marks).

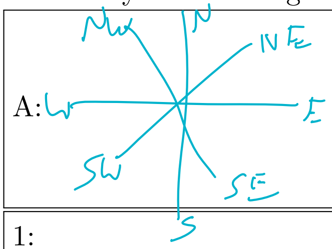
A: 10/3/04 18.00 one-hot day, month, hour

$[0 \ldots \underset{10}{1} \ldots 0 \mid 0 0 \underset{3}{1} 0 \ldots \mid 0 \ldots \underset{18}{1} \ldots 0]$

1: $[1 0 0 \ldots 1 \ldots 0 \mid 0 0 1 \ldots \mid 0 \ldots 1 0 \ldots]$

2: $[1 0 0 \ldots 1 \mid 0 0 1 \ldots \mid 0 \ldots 0 \underset{19}{1} \ldots]$

2. Similarly, suppose you wanted to incorporate the wind direction[1] and wind speed into your predictor. Describe your encoding and write down the features of the first two datapoints (2 marks).

A: 

one-hot

NW	N	NE	E	SE	S	SW	W	NW
$\cancel{0}$	0	0	0	0	0	0	0	1
$\cancel{0}$	0	0	0 0		0	0	0	1

1:

2:

3. (Critical thinking) Even though we’ve solved a regression problem above, suppose that you only cared about predicting a *binary* outcome, namely whether PM2.5 > 50 on a given day. Consider the following two options for solving this problem:

   (a) Train a regressor $f(X) \to \mathbb{R}$ as we have done above, and predict ‘True’ whenever the output of the regressor is greater than 50

   (b) Train an SVM classifier $f(X) \to \{0, 1\}$ (using the same features) based on the binary outcome

   Are these two approaches equivalent? Briefly explain why or why not, and if not, which you would expect to obtain better performance (i.e., accuracy) on this task and why (2 marks).

   A: No, not equivalent. For the regressor 250 vs 200 is same as 0 vs. 50
   not an error error
   so, regressor is not minimizing # of errors

[1] NNW = North-North-West, etc.

Section 2: Classification and Diagnostics (9 marks)

Suppose you wish to build a classifier to detect malicious e-mails (e.g. spam, phishing, etc.). You collect 10,000 e-mails, and obtain ground-truth labels indicating which e-mails are malicious (i.e., malicious e-mails are labeled *True*). You then train three classifiers, whose performance is as follows:

Classifier 1:

False Positives	150
False Negatives	21
True Positives	35
True Negatives	9794

Classifier 2:

False Positives	3828
False Negatives	6
True Positives	50
True Negatives	6135

Classifier 2:

False Positives	843
False Negatives	40
True Positives	16
True Negatives	9101

4. How many of the 10,000 instances have a positive label (i.e., $y_i = True$) (1 mark)?

A: $TP + FN = 21 + 35$

5. How many of the 10,000 instances have a positive prediction **for Classifier 1** (i.e., $f(X) = True$) (1 mark)?

A: $TP + FP = 35 + 150$

6. Compute the following statistics **for Classifier 1**. You can leave your results as unsimplified expressions (2 marks):

Accuracy: A: $(35 + 9794) / 10000$

BER: A: $1 - \frac{1}{2}\left(\frac{35}{35+21} + \frac{9794}{9794+150}\right)$

Precision: A: $35 / (35 + 150)$

Recall: A: $35 / (35 + 21)$

$TP + TN / 10000$

$1 - \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)$

$\frac{TP}{TP + \boxed{FP}}$

$\frac{TP}{TP + \boxed{FN}}$

Which of the three classifiers would you select if your goal is to optimize the measures below? Assume that content where the prediction is positive is filtered/blocked (e.g. moved to a spam folder). **Briefly state your reasoning for each answer**. (1 mark each).

7. The classifier with the highest accuracy:

A: TP + TN = C1

8. The classifier that lets the *fewest malicious e-mails* through the filter:

A: lowest FN = C2

9. The classifier that filters the *fewest non-malicious e-mails*:

A: low FP = C1

10. (Critical thinking) You train a classifier based on the 5,000 most common words in spam e-mails (i.e., you use a 5,000 dimensional feature vector with binary features indicating which common words appear in each e-mail). You use half of your data for training and half for testing. After training the classifier you diagnose the following issues:

- The classifier has strong training performance, but weak performance on the test set.
- Even though the classifier has high accuracy, the classifier identifies nearly all spam e-mails as 'non-spam'

Suggest steps you might take to address the above issues (e.g. modifications to your classifier or features, etc.) (2 marks):

A:

① overfitting → regularizing, or use a smaller dictionary

② dataset imbalance → use a balanced classifier

Section 3: Clustering / Communities (5 marks)

Suppose you collect a dataset of taxi rides in New York, containing pickup and dropoff locations, among other features. After generating a scatterplot of the data you obtain the following result:[2]

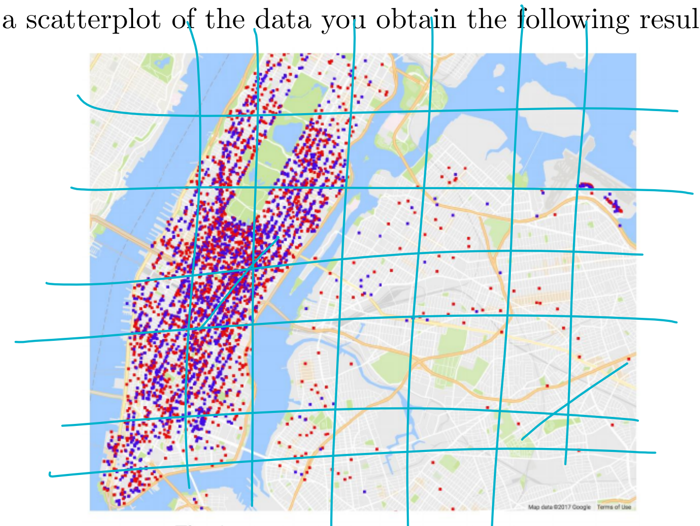


Fig. 1: Mapping of pick-up and drop-off locations

Suppose your goal is to predict the *total tip* that a given fare will receive.

You consider three alternative techniques to incorporate the geographical location into your model:

- (Grid:) Split the data into a grid (using latitude and longitude values), and include a feature indicating which grid position each datapoint belongs to.
- (Nearest Neighbor:) For each new trip, identify the 'most similar' trip in the training data in terms of the distance between start and end locations. Predict the tip for the new trip to be the same as the tip for this previous trip (this is known as 'nearest neighbor' classification).
- (Clustering:) Run a clustering algorithm (e.g. k-means or hierarchical clustering) to obtain feature representations of each point.

11. Suggest one reason why clustering the data might be preferable to each of the 'grid' or 'nearest neighbor' models (2 marks):

Versus Grid:	Grid could be not granular enough in busy regions Clustering would add more clusters in dense regions.
Versus Nearest Neighbor:	Absolute coordinates matter (long trip in city ≠ long trip in suburbs), but NN ignores coordinates

12. (Design thinking) In addition to geographical features, suggest (at least three) additional features that may be useful in predicting tip amounts (3 marks):

A:

- total pre-tip cost
- Speed
- time of day / day of week

[2] Scatterplot taken from a previous CSE258 assignment on taxi tip prediction.

Section 4: Recommender Systems (6 marks)

Suppose you collect the following ratings of teen romance novels from *Goodreads*:

Item ID	Book	Read?					Rated?				
		Nathan	Thomas	Dhruv	Kevin	Prateek	Nathan	Thomas	Dhruv	Kevin	Prateek
1	*To All the Boys I've Loved Before*	1	1	0	1	1	5	3	?	1	4
2	*P.S. I Still Love You*	1	0	0	0	1	5	?	?	?	4
3	*Always and Forever, Lara Jean*	1	0	0	0	0	4	?	?	?	?
4	*It All Started with an Apple*	0	1	0	0	0	?	2	?	?	?
5	*The Kissing Booth*	1	0	1	1	1	1	?	1	2	4

$\|a\|$

$\sqrt{25+9+1+16} = \sqrt{51}$

$\sqrt{41}$

4

2

$\sqrt{22}$

13. You want to implement a feature of the form 'you'll like X because you liked Y,' that is based on maximizing the *cosine similarity* between ratings of the items X and Y, and only makes a recommendation if (a) the user's rating of Y is ≥ 4 stars, and (b) the user hasn't already rated X. Under this system, what would be the top recommendation for Prateek? Show which comparisons you considered (2 marks)

A: 1 vs. 4 $= \frac{3}{\sqrt{51}}$ | 1 vs. 3 $= \frac{5}{\sqrt{41}}$

You want to make a simple recommender that identifies the 'all time best' books, using a model of the form

$$\text{rating}(i) = \alpha + \beta_i.$$

Here α is a global term, and β_i is an item bias. You fit your model by setting α to the global mean of all ratings, and β_i to be the remainder. Finally, you make recommendations simply by identifying those items with the highest bias terms, i.e.,

$$\underset{i}{\text{argmax}}\, \beta_i.$$

14. What item would receive the highest ranking according to this global recommender (1 mark)?

A: I 2

highest av.

15. (Critical Thinking) Suppose you wanted to design a recommender system to estimate the compatibility between candidates and job openings. Describe what data you would collect from users, how you would model the problem, and any issues that make this problem different or unique compared to those we saw in class (3 marks).

A:

Data: qualifications (user) one-hot | requirements (job) one-hot | interactions adjacency

Model: Similarity (Jaccard) on interactions + classification on features w/ ensemble

Issues: Sparse – very few jobs per person

New users (how to recommend w/ no job history)

Precision:

$$\frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

$$\frac{TP}{TP + FP}$$

Recall:

$$\frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

$$\frac{TP}{TP + FN}$$

Balanced Error Rate (BER):

$$\frac{1}{2}(\text{False Positive Rate} + \text{False Negative Rate})$$

F-score:

$$2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Jaccard similarity:

$$\text{Sim}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Cosine similarity:

$$\text{Sim}(A, B) = \frac{A \cdot B}{\|A\|\|B\|}$$

Algorithm 1 Hierarchical clustering

```
Initially, every point is assigned to its own cluster
while there is more than one cluster do
    Compute the center of each cluster
    Combine the two clusters with the nearest centers
```

Algorithm 2 K-means

```
Initialize every cluster to contain a random set of points
while cluster assignments change between iterations do
    Assign each $X_i$ to its nearest centroid
    Update each centroid to be the mean of points assigned to it
```

Write any additional answers/corrections/comments here: