University of Glasgow

Wednesday 15th of December 2021
9:00 am — 11:30am
Duration: 2 hours
Additional time: 30 minutes
Timed exam — fixed start time

DEGREE OF MSc

INTRODUCTION TO DATA SCIENCE AND SYSTEMS (M)
COMPSCI5089

Answer all 3 questions

This examination paper is worth a total of 60 marks.

1. Computational linear algebra and optimisation

(a) Given a collection of N documents $\mathscr{D} = \{D_1, \ldots, D_N\}$, your task is to implement a functionality that provides a list of suggested 'more like this' documents. With this problem context, answer the following questions.

(i) Explain how would you represent each document $D \in \mathscr{D}$ as a (real-valued) vector $\mathbf{d}$. What is the dimension of each vector? [2]

(ii) What does the L_0 norm of a document vector indicate (in plain English) as per your definition of the document vectors in the previous question? [1]

(iii) How would you define the L_p distance between two document vectors $\mathbf{d}$ and $\mathbf{d}'$? [2]

(iv) What distance or similarity measure would you use for finding the set of 'more like this' documents for a current (given) document vector $\mathbf{d}$, and why. [2]

(b) The probability distribution function of an n dimensional Gaussian is given by

$$f(\mathbf{x}) = (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu),$$

where $\mu \in \mathbb{R}^n$ is the mean vector and $\Sigma \in \mathbb{R}^{n \times n}$ is a square and invertible matrix, called the *covariance* matrix. Consider the particular case of $n = 2$. Answer the following questions.

(i) Plot the contours of the following Gaussians. For each contour plot, show the conditional distributions along the two axes.

$$\mu = (0,0)$$
$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mu = (0,0)$$
$$\Sigma = \begin{pmatrix} 0.5 & 0 \\ 0 & 2 \end{pmatrix}$$

$$\mu = (0,0)$$
$$\Sigma = \begin{pmatrix} 0.5 & 0.1 \\ 0.5 & 2 \end{pmatrix}$$

$$\mu = (0,0)$$
$$\Sigma = \begin{pmatrix} 0.5 & 0.1 \\ -0.5 & 2 \end{pmatrix}$$

[2]

(ii) Which one/ones of the above 4 Gaussian distributions can be reduced to a a single dimensional Gaussian with PCA on the covariance matrix without too much loss of information. Note that you do not need to explicitly compute the Eigenvalues. You should rather derive your answer from a visual interpretation of the contour plots. Clearly explain your answer. [2]

(c) With respect to linear regression, answer the following questions.

(i) Derive the expression for stochastic gradient descent for linear regression with the squared loss function. Clearly introduce your notations for the input/output instances, and the parameter vector. [4]

(ii) Explain how linear regression can be extended to polynomial (higher order) regression? What is the problem of using high degree polynomials for regression? How can that problem be alleviated? [3]

(iii) A common practice in stochastic gradient descent is to use a variable learning rate α for the parameter updates

$$\theta_j^{(t+1)} \leftarrow \theta_j^{(t)} + \alpha^{(t)} \frac{\partial L}{\partial \theta_j},$$

where $\theta_j^{(t)}$ denotes the j^{th} component of the parameter vector θ at iteration t, and $\alpha^{(t)}$ denotes the value of the learning rate at iteration t. Which of the following alternatives of the learning rate update would you prefer (α is a constant) and why?

$$a)\, \alpha^{(t)} = \frac{\alpha}{t}, \quad b)\, \alpha^{(t)} = \alpha + t.$$

[2]

2. Probabilities & Bayes rule

Consider a card game where you have 4 suits (heart, diamonds, clubs and spades) and in each suit the cards 7, 8, 9, 10, Jack (J), Queen (Q), King (K) and Ace (A). In this question we will use the following commonly used terms:

- the *the pack*: is the set of all cards that have not been drawn yet.
- to *draw*: is to pick a card at random amongst the pack of remaining cards, removing it from the pack.
- the *hand*: is the set of cards a player has drawn from the pack.
- a *payout*: is the amount of points you get for a given hand.
- to *fold*: is to stop playing and put back your cards in the pack, forfeiting any payout for this game.

(a) Assuming that you draw a single card at random from the pack, give the probabilities for the following events

(i) Drawing an Ace?

(ii) Drawing a red card?

(iii) Drawing a diamonds?

(iv) Drawing a royalty figure (Jack, Queen or King)?

(v) Drawing the Ace of spades?

[5]

(b) Now assume that you have already drawn the three cards: 10,J,Q. When drawing two more cards from the pack, what is the probability to obtain:

(i) A pair of two cards with the same value (eg, two Jacks).

(ii) Two pairs (eg, two Jacks and two Queens).

(iii) Three of a kind (eg, three Jacks).

(iv) A sequence of 5 cards (eg, 10, J, Q, K, A). Note that the cards can be of any suit, but there cannot be a break in the sequence.

[4]

(c) Now let us assume the following payout table for each hand of 5 cards:

hand	payout
sequence of 5 cards	50
three of a kind	30
two pairs	20
one pair	10
anything else	0

As before, you have the cards 10, J, Q in hand.

(i) If you draw two more cards randomly from the deck, what is the expected value of the payout for this hand? [3]

(ii) Assuming that you need to pay 5 every time you draw a card (hence you would need to pay 10 to draw two cards), should you fold your hand or draw cards? [2]

(iii) Should you fold after drawing the first card (and having paid 5), if the card is: (i) the 7 of heart, (ii) the 8 of spades or (iii) the Queen of diamonds? [6]

3. Database systems

An online retail company is trying to assess the performance of its DB systems and has asked you to investigate some of the operations. Consider a relation Seller (ID, Name, Country) – abbreviated as S – where the primary key (ID) is a 32-bit integer, the Name attribute is a 54-byte (fixed length) string, and Country is a 16-bit integer. Further consider a relation Product(ID, ProductID, ManufacturerID, Price) – abbreviated as P – with ID being a foreign key to Seller's ID, ProductID and ManufacturerID being 64-bit integers, Price being a 32-bit float, and the first three attributes making up the relation's (composite) primary key.

Assume that both relations are stored in files on disk organised in 512-byte blocks, with each block having a 10-byte header. Assume that S has $r_S = 1,000$ tuples and that P has $r_P = 100,000$ tuples. Last, assume that Product is stored organised in a sequential file sorted by its primary key and Student is stored organised in a heap file. Finally, note that the database system adopts fixed-length records – i.e., each file record corresponds to one tuple of the relation and vice versa.

(a) Compute the blocking factors and the number of blocks required to store these relations. Show your work.

[2]

(b) Consider the following query:

```
SELECT S.Name, P.ID, P.Price FROM Seller as S, Product as P
WHERE S.ID = P.ID AND S.ID >= 6,000 and S.ID <= 6,199;
```

Assume that the memory of the database system can accommodate $n_B = 22$ blocks for processing, that all seller IDs in the query range exist in the database, and that all sellers have the same number of products on average.

Explain the query processing algorithm, taking care to include the file organisation in your reasoning. Then estimate the total expected cost (in number of block accesses) of the query above (Disregard the cost associated with the writing of the result set), of the following two approaches:

(i) First, assume that S is scanned at the outer loop and show your work: [9]

(ii) Second, assume instead that P is scanned at the outer loop and show your work: [9]

END OF QUESTION PAPER