Examination
Software Engineering for AI Systems
DIT821

Software Engineering and Management
Chalmers | University of Gothenburg

Tuesday January 3, 2023

Time	08:30-12:30
Location	Lindholmen
Responsible teacher	Daniel Strüber (mobile: 0760475434)
Total number of pages	**5** (including this page)
Teacher visits exam hall:	At circa 9:30 and at circa 11:30
Exam (4.5 HEC)	Max score: 20 pts
Grade limits (4.5 HEC)	G: at least 10 pts
	VG: at least 15 pts

ALLOWED AID:

- English dictionary
- **NOT ALLOWED:** Anything else not explicitly mentioned above (including additional books, other notes, previous exams, or any form of electronic device: dictionaries, agendas, computers, mobile phones, etc.)

PLEASE OBSERVE THE FOLLOWING:

- This exam is composed by four exam tasks, divided into further sub-tasks, roughly corresponding to the four main topic areas of the lecture.
- Start each task on a new paper;
- Sort your answers in order (by task and sub-task) before handing them in;
- Write your student code on each page and put the number of the question on **every** paper;
- Points are denoted for each task and sub-task. The point distribution can give you an indicator of how much time to spend on each task and sub-task.
- Activities in the lectures in form of participation in breakout rooms will be considered, in the form of bonus points (1 pt. max).

Task 1: Linear Regression, Gradient Descent, Normal Equation (5 pts.)

a) Explain the *overfitting* and *underfitting* problems. (1 pt.)

Overfitting: Good performance on the training data, poor generliazation to other data.
Underfitting: Poor performance on the training data and poor generalization to other data

Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the models ability to generalize.

Underfitting refers to a model that can neither model the training data nor generalize to new data. An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data. Underfitting is often not discussed as it is easy to detect given a good performance metric. The remedy is to move on and try alternate machine learning algorithms. Nevertheless, it does provide a good contrast to the problem of overfitting.

b) Explain how regularization helps to address the overfitting problem. (1 pt.)

A loss function is involved in the fitting process. It is computed as the difference between the actual and predicted output from a model. Based on the training data, the loss function will adjust the coefficients. If the presence of noise or outliers is found in the training data, the approximated coefficients will not generalize well to the unseen data. Regularization comes into play and shrinks the learned estimates towards zero. In other words, it tunes the loss function by adding a penalty term, that prevents excessive fluctuation of the coefficients. Thereby, reducing the chances of overfitting.

c) Compare *gradient descent* and the *normal equation* approach by explaining at least three differences. These differences could be, for example, about their input, how they work, their performance, and the quality of their results. (1 pt.)

Gradient Descent	Normal Equation

In gradient descenet , we need to choose learning rate.	In normal equation , no need to choose learning rate.
Gradient descent works well with large number of features.	Normal equation works well with small number of features.
Feature scaling can be used.	No need for feature scaling.
No need to handle non-invertibility case.	If (X^TX) is non-invertible , regularization can be used to handle this.
It is an iterative algorithm	It is analytical approach.
Algorithm complexity is $O(kn^2)$.	Algorithm complexity is $O(n^3)$

d) Suppose you have a dataset with m=50 examples and n=200,000 features. You want to use multivariate linear regression to fit the parameters θ to our data. Would you prefer gradient descent or the normal equation? Explain why! (1 point)

With n = 200000 features, you have to invert a 200001 x 200001 matrix to compute the normal equation. Inverting such a large matrix is computationally expensive, so gradient descent is a good choice.

e) Let $f(\theta_0, \theta_1)$ be a function that takes two numbers and outputs a number. Assume that f is an arbitrary smooth function, in particular, it may have local optima. Suppose we use gradient descent to try to minimize $f(\theta_0,\theta_1)$ as a function of θ_0 and θ_1. Which of the following statements are true or false? Explain with reasons. (1 pt. in total, 0.5 each)

a. Even if the learning rate α is very large, every iteration of gradient descent will decrease the value of $f(\theta_0, \theta_1)$.

False: If the learning rate is too large, one step of gradient descent can actually vastly "overshoot" and actually increase the value of $f(\theta_0,\theta_1)$.

b. If θ_0 and θ_1 are initialized so that $\theta_0=\theta_1$, then by symmetry (because we do simultaneous updates to the two parameters), after one iteration of gradient descent, we will still have $\theta_0=\theta_1$.

The updates to θ_0 and θ_1 are different (even though we're doing simulaneous updates), so there's no particular reason to update them to be same after one iteration of gradient descent.

Task 2: Classification and Clustering (5 pts.)

a) Sir Jamine Lannister has given up sword fighting. Therefore, he has started to take up machine learning as a hobby. Sir Jamie has fit a logistic regression model of the form:

$$y = g(w_0 + w_1 x + w_2 x^2)$$

The values of the parameters he found are $w_0 = 6, w_1 = -5, w_2 = 1$. Now, Sir Jamie wants to find a decision boundary, but has forgotten how to do so. Help Sir Jamie and specify the equation of the decision boundary. (1 pt.)

1. The equation for the boundary is $6 - 5x + x^2$

b) Using logistic regression and a labelled dataset, Sir Jamie trained a one-vs-all model:

$$\mathcal{L} = \{(X_i, Y_i)\}_{i:=1}^{m}$$

where the target Y can have three different labels 0, 1, 2.

Now, Sir Jamie wishes to find the labels of three new unlabeled points x_1, x_2, x_3. Help Sir Jamie predict the labels y_1, y_2, y_3 for x_1, x_2, x_3 respectively based on the following information: (1 pt.)

$$h^{(1)}(x_1) = 0.53, h^{(2)}(x_1) = 0.29, h^{(3)}(x_1) = 0.28$$
$$h^{(1)}(x_2) = 0.21, h^{(2)}(x_2) = 0.23, h^{(3)}(x_2) = 0.56$$
$$h^{(1)}(x_3) = 0.75, h^{(2)}(x_3) = 0.20, h^{(3)}(x_3) = 0.05$$

2. $y_1 = 1,\ y_2 = 3, y_3 = 1$

c) One important algorithm for clustering is K-means. At minimum, the K-means algorithm requires two inputs: the training data and a certain parameter. What is the parameter and what does it do? (l pt.)

K, number of clusters

d) The first step of the K-means algorithm is to initiate cluster centroids. What are the second and third steps? You can give the answer in words or by writing it as pseudo-code. (Hint: Each step can be seen as a for-loop.) (1 pt.)

- calculate distance between each centroid and data point
- assign each data point to the nearest cluster

e) K-means minimizes the following cost function:

$$J\left(c_j^{(1)},\dots,c_j^{(m)},\mu_1,\dots,\mu_K\right) = \frac{1}{m}\sum_{i=1}^{m}\|x^{(i)} - \mu_{c^{(i)}}\|^2$$

Explain the cost function. (1pt.)

Average squared distance of objects from the respective centroids

Task 3: Neural Networks **(6 pts.)**

a) Compare traditional machine learning to deep learning by explaining at least three differences. These differences could be, for example, about their input, how they work, their performance, and the quality of their results. (1 pt.)

- DL can handle and deal with **more complex data formats** (image, text, audio signal ..etc.) rather the tabular data in ML.
- In ML, we follow the **feature engineering by a human**, and in DL, we **extract the features automatically** through different architectures.
- ML is limited for the **amount of data** because the **parameters in ML are limited**. As for DL, the architecture is scalable for the amount of data we have, which is why the more data we have, the more accurate the model we obtain.

b) In a neural network, what is the purpose of non-linearity? Name at least one type of function typically used for non-linearity. (1 pt.)

One function is ReLue. The data may contain many features and information like many objects, colours, backgrounds, etc., so we must add more complexity.

c) In a convolutional neural network, why is the kernel size generally smaller than the image size? (1 pt.)

to locate the local features information in the image matrix

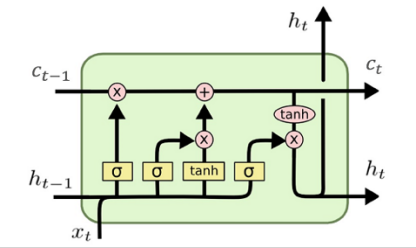
d) In a convolutional neural network, what is the functionality of the pooling layer? (1 p.)

Downsampling by reducing the dimension of the feature map and preserving the essential features.

e) What is the vanishing gradient problem and how do recurrent neural networks solve it? (2 pt.)

What it is: when we have s long dependencies, the slight gradient when it propagated back caused a minor update for the W, which caused insufficient training.
Example: (The tree colour is "**green**") this example is for predicting the next word. The dependencies between the target word "green" and the corresponding contextual information "tree colour" is small. (I live in Gothenburg in Southern Sweden. where I live, the weather is generally "**cold**" most of the year) here we have very long dependencies between the predicted word and contextual information.

How to solve it. By changing the cell of RNN and adding more gates.
The students must either explain the architecture or write the mathematical formula (a bit easier than explaining it by text



Task 4: ML Engineering **(4 pts.)**

a) Name two sources of data quality problems, in particular:
- One source of problems that can appear during data collection, and
- One source of problems that only arises over time, *after* data collection. (1 pt.)

During collection: sensor inaccuracies
After collection: concept drift, e.g, user behavior changes

b) During feature engineering, it can become necessary to convert categorical data into numerical data. Explain why that is (with details, e.g., giving an example). Explain how one-hot encodings can be used to address this task. (1 pt.)

The algorithm might only work on numerical data (e.g., NN). One-hot encoding provides a way to encode each category as a combination of numerical values.

c) The formula for the IOU score is:

$$IOU = \frac{Area\ of\ Intersection\ of\ two\ boxes}{Area\ of\ Union\ of\ two\ boxes}$$

Explain what the formula is used for, in the context of data management, and how it works: What are the boxes? How is the score interpreted? (1 pt.)

Used, e.g., to assess inter-annotator agreement. Boxes = bounding boxes specified by several annoators, higher IO is good

d) Discuss the following statement about the "model requirements" and the "model evaluation" phases in the ML engineering workflow:

> Model requirements are connected to the evaluation metrics used for model evaluation, in the following way: for each model requirement, there should be at least one metric that directly measures how well the model fulfills it.

Is this statement always applicable? Give at least one example and (if there is one) one counterexample to this statement. (1 pt.)

Not, it is not always applicable. It is generally desirable to have metrics for as many requirements as possible. For example, good predictive performance could be a requirement, and then relevant metrics would be precision, recall, F1, and accuracy. However, some requirements cannot be directly measured by metrics, e.g., regarding explainability.