Data Mining

Homework 4

Due: 10/1/2021, 23:59

Instructions

You must hand in the homeworks electronically and before the due date and time.

The first homework has to be done by each **person individually**.

Handing in: You must hand in the homeworks by the due date and time by an email to Andrea (`mastropietro@diag.uniroma1.it`) that will contain as attachment **(not links to some file-uploading server!)** a .zip file with your answers. The filename of the attachment should be
`DM_Homework_1__StudentID_StudentName_StudentLastname.zip`;
for example:
`DM_Homework_1__1235711_Robert_Anthony_De_Niro.zip`.
The email subject should be
`[Data Mining] Homework_1 StudentID StudentName StudentLastname`;
For example:
`[Data Mining] Homework_1 1235711 Robert Anthony De Niro`.
After you submit, you will receive an acknowledgement email that your project has been received and at what date and time. If you have not received an acknowledgement email within 2 days after you submit then contact Andrea.

The solutions for the theoretical exercises must contain your answers either typed up or hand written clearly and scanned.

For any questions on the homework, clarifications, and so on, contact Andrea (`mastropietro@diag.uniroma1.it`).

For information about collaboration, and about being late check the web page.

Problem 1.

We will prove some facts for the Frobenius norm. Given a matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$, we define the *trace* of $\mathbf{X}$ to be the sum of the diagonal elements:

$$\mathrm{Tr}(\mathbf{X}) = \sum_{i=1}^{n} \mathbf{X}_{ii}.$$

In the following, $\mathbf{A} \in \mathbb{R}^{n \times d}$.

1. Prove that $\|\mathbf{A}\|_F^2 = \mathrm{Tr}(\mathbf{A}^T\mathbf{A})$.
2. Let $\mathbf{U} \in \mathbb{R}^{n \times n}$ and $\mathbf{V} \in \mathbb{R}^{d \times d}$ be orthogonal matrices. Prove that $\|\mathbf{UAV}\|_F = \|\mathbf{A}\|_F$.
3. Prove that $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{r} \sigma_i^2}$, where r is the rank of matrix $\mathbf{A}$.

Problem 2.

In this problem we will show what we discussed in class, that the best rank-k approximation to a matrix $\mathbf{A}$ of rank at least k is the matrix $\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$, where σ_i, $\mathbf{u}_i$ and $\mathbf{v}_i$ are the ith singular value, left singular vector, and right singular vector, respectively.

Let $\mathbf{A} \in \mathbb{R}^{n \times d}$. We will prove that among all matrices $\mathbf{X} \in \mathbb{R}^{n \times d}$ of rank at most k, $\|\mathbf{A} - \mathbf{X}\|_F$ is minimized for $\mathbf{X} = \mathbf{A_k}$.

To do that, we will try to explicitly create another matrix of rank k.

Consider the singular-value decomposition of $\mathbf{X}$. Argue that it is of the form

$$\mathbf{X} = \mathbf{U} \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{V}^T,$$

where $\mathbf{U}, \mathbf{V}$ are orthogonal matrices and $\mathbf{D}$ is a diagonal matrix of dimensions $k \times k$.

Assume that

$$\mathbf{A} = \mathbf{U} \begin{bmatrix} \mathbf{L} + \mathbf{E} + \mathbf{R} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \mathbf{V}^T,$$

for some matrices $\mathbf{L}$,$\mathbf{E}$,$\mathbf{R}$,$\mathbf{F}$,$\mathbf{G}$,$\mathbf{H}$, where $\mathbf{L}$ has dimensions $k \times k$ and is strictly lower triangular, $\mathbf{R}$ is strictly upper triangular, and $\mathbf{E}$ is diagonal.

1. We will first show that $\mathbf{L}$, $\mathbf{R}$, $\mathbf{F}$, and $\mathbf{G}$ are all equal to $\mathbf{0}$. Consider the matrix

$$\mathbf{Y} = \mathbf{U} \begin{bmatrix} \mathbf{L} + \mathbf{D} + \mathbf{R} & \mathbf{F} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{V}^T.$$

   (a) What is the rank of $\mathbf{Y}$?

   (b) Express $\|\mathbf{A} - \mathbf{X}\|_F^2$ as a function of $\|\mathbf{A} - \mathbf{Y}\|_F^2$, $\|\mathbf{L}\|_F^2$, $\|\mathbf{R}\|_F^2$, and $\|\mathbf{F}\|_F^2$. Use this expression, and the fact that $\|\mathbf{A} - \mathbf{X}\|_F^2$ is as small as possible, to show that $\mathbf{L}$, $\mathbf{R}$, and $\mathbf{F}$ all equal to $\mathbf{0}$.

   (c) Show similarly, that $\mathbf{G} = \mathbf{0}$.

2. Argue that $\|\mathbf{A} - \mathbf{X}\|_F = \left\|\mathbf{U}^T\mathbf{A}\mathbf{V} - \mathbf{U}^T\mathbf{X}\mathbf{V}\right\|_F$. Using this, compare the matrices $\mathbf{U}^T\mathbf{A}\mathbf{V}$ and $\mathbf{U}^T\mathbf{X}\mathbf{V}$ (write them in explicit form as we did above). Given that we want $\|\mathbf{A} - \mathbf{X}\|_F$ to be minimum, what should the diagonal values of $\mathbf{E}$ be? What should the singular values of $\mathbf{H}$ be? Use these facts to show that $\mathbf{X} = \mathbf{A}_k$.

Problem 3.

You will play a bit with neural nets. In particular, you have to develop a neural network able to perform text classification in PyTorch. Techniques such as feedforward neural networks and convolutional neural nets can be used (especially CNNs), even though the literature demonstrated RNNs (Recurrent Neural Networks) and LSTM (Long Short-Term Memory Networks) to be the most powerful neural nets to work with text. So, for the most adventurous, I advise to develop an RNN or an LSTM for the homework (not compulsory, anyway). The task you are asked to perform is about emergency awareness enhancement using tweets. The dataset your are going to use is the **Disaster Tweets** from Kaggle. It contains tweets collected from several distaster situations (volcanic eruption of Taal, coronavirus, etc), and it's available here: `https://www.kaggle.com/vstepanenko/disaster-tweets`. For the homework, given *only the text* of the tweet, you have to predicted whether it is a disaster-related tweet or not; it's a binary classification task.

The homework is divided into the following points:

Part 1. Feature Extraction and Classification

1. Download the dataset and perform feature extraction from the text. You can use any feature you want (bag of words, word count, tf-idf, word embeddings obtained by a neural network, etc.). You can also combine different features together. Do whatever you think can represent the dataset in the best way possible.

2. Build a neural network that is able to perform text classification. The input of the network will be the features extracted from the tweets and the output the most probable target value (disaster tweet or not).

Part 2. Transfer Learning using BERT

1. For the second part of the homework you will download, using PyTorch facilities, a pretrained and recent neural network developed by Google, called BERT, that is able to perform a very powerful contextual text embedding taking into consideration information such as semantics, syntax, morphology and so on. So, you will do a bit of transfer learning. You will feed the network with your data and you will select the proper output layer for the task. I suggest you check this link: `https://github.com/nlptown/nlp-notebooks/blob/master/Text%20classification%20with%20BERT%20in%20PyTorch.ipynb`, to see how to use BERT.

2. Compare the results obtained using the method developed by yourselves and BERT. To evaluate the models use the classic evaluation metrics, such as **accuracy**, **recall**, **precision** and **F1 score**, plot also a **confusion matrix**. Can you do better then BERT? ;)

Write a SHORT report (max 4/5 pages) in which you describe all the steps (plots are welcome). Describe the features used and why you thought they could represent your data properly. Describe the neural net, the layers, the loss function and the optimizer used (not deeply) and why you think your network can model the problem properly. Comment the results obtained with any observation you think to be important and try to explain why your network did (or did not) perform better than BERT.
Hint: The dataset is unbalanced, so be careful on how you weigh the classes or balance the dataset, since it can impinge on the results.