PEKING UNIVERSITY, GUANGHUA SCHOOL OF MANAGEMENT
Machine Learning and Artificial Intelligence

Assignment 3

Prof. Yingjie Zhang

Due date: Nov 28, 2021 11:59pm

Instruction

- This homework includes both conceptual questions and implementation questions (all are described in this .pdf file)
- Deliverables: Please submit **one pdf file** (with all your answers to the conceptual questions) and **python files** (including one with a .ipynb extension and one with a .html extension) that include all coding work (with necessary comments).
- You may discuss the questions with fellow students, however you must always write your own solutions and must acknowledge whom you discussed the question with. Do not copy from other sources, share your work with others, or search for solutions on the web. Plagiarism will be penalized according to university rules.

1 HIDDEN MARKOV MODEL

We denote the data set as $\mathbf{X} = \{x_1, x_2, ..., x_T\}$. And the hidden states are $\mathbf{Z} = \{z_1, z_2, ..., z_T\}$.

In class, we learned how to compute

$$\alpha(z_t) = P(x_1, ..., x_t, z_t) \tag{1.1}$$

and

$$\beta(z_t) = P(x_{t+1}, ..., x_T | z_t) \tag{1.2}$$

a. Show that

$$\begin{aligned} \xi(z_{t-1}, z_t) &= P(z_{t-1}, z_t | \mathbf{X}) \\ &= \frac{\alpha(z_{t_1}) P(x_t | z_t) P(z_t | z_{t-1}) \beta(z_t)}{P(\mathbf{X})} \end{aligned} \tag{1.3}$$

b. How can you use α or β definitions to compute $P(\mathbf{X})$?

2 PRINCIPLE COMPONENT ANALYSIS

In NLP, words such as "the", "an", "of", etc. that occur commonly in any English document are called *stopwords*. Because such words occur throughout a text dataset (often called a corpus), they typically have high corpus-wide counts. As a result, their corresponding BoW features are often dominant in the first principal component of the dataset because they often explain a lot of the variance in the data. In this question, we will discover stopwords from the top principal component of a dataset of text articles, each coming from either the serious European magazine *The Economist*, or from the not-so-serious American magazine *The Onion*. The data is provided to you as 'onion_vs_economist.zip' on Canvas, containing five files which can be loaded using appropriate commands in your programming language to yield the following variables:

- `Vocabulary` is a V dimensional list that that contains every word appearing in the documents. When we refer to the j^{th} word, we mean `Vocabulary(j)`.

- `XTrain` is a $n \times V$ dimensional matrix describing the n documents used for training your Naive Bayes classifier. The entry `XTrain(i,j)` is 1 if word j appears in the i^{th} training document and 0 otherwise.

- `yTrain` is a $n \times 1$ dimensional matrix containing the class labels for the training documents. `yTrain(i,1)` is 0 if the i^{th} document belongs to The Economist and 1 if it belongs to The Onion.

- Finally, `XTest` and `yTest` are the same as `XTrain` and `yTrain`, except instead of having n rows, they have m rows. This is the data you will test your classifier on and it should not be used for training.

a Find the top principal component of `XTrain`. Note that the length of the principal component is equal to the number of words in `Vocabulary`. Provide a list of the top-30 words from `Vocabulary` sorted in decreasing order of the *absolute* value of their coefficient in the top principal component.

b many of the top-30 words in the list are stopwords i.e. occur in the `stopwords.txt` file provided with the data?

3 Deep Learning

Describe in the iPython file.