

Assignment 2 in Artificial Intelligence

Lakes Athanasios, 1115201800090

December 2022

Contents: In this folder you will find the source code Ai 2 Ergasia 2.ipynb, a folder containing learning curve screenshorts (learning curve) and a notepad file (report) that containins some parameters of the different models I used.

1 Introduction

The following document is supposed to shed light on some of my choices regarding the initialization of the linear model while also clarifying my thought process approaching the assignment.

2 Model

The model I finally concluded on is a linear 2-layer model with the following attributes:

- 1st layer has 128 units
- 2nd layer has 64 units
- output uses softmax function to produce probabilities

I followed the instructions given by the respective lab class for this assignment and used a feed forward linear model, reusing its loss in every batch with the function `loss.backward()`.Deciding on the model was not an easy task though. I tried many different parameters and made some interesting experimentation.

3 Experimentation

3.1 Layer

This was probably the trickiest part. Following the examination of the lab code, i started trying different models that were using 3 or more layers. However as you can see on the photo model6f1.png the results even though they started to converge on the last epochs, in the beggining had pretty much random behaviour. Another 4-layer model you can check its learning curve is model1f1 which had high overfit, probably due to its high batch size (more on that later).

4-layer model did not work for me in most occasions so I started taking out layers. In photo **model7f1.png** and **model4f1.png** you can see the learning curves of 2 3-layer models. The former, was overfitting and the latter was kinda ok.

Trying to find an even better model, I took out one more layer and made a 2-layer model. Photos model9f1 and model8f1 depict some learning curves produced by 2-layer models the former with layers 128, 32 and the later 128, 64. Eventually I decided to make use of this type of model.

3.2 Batch size

Batch size parameter was an essential part of the experimentation. I used different number for batches. 16 was alright but on the other hand 64 resulted in a slower increase of the score (model10f1.png). Eventually I decided to play it safe and went for batch = 32.

3.3 Learning Rate

The learning rate for my final model is 0.01 (1e-2) and was also a result of experimentation. Firstly, I was using very lower values such as (1e-4), (2e-4) and didnt make the model work. Using a rate 0.1 was also proved to be very effective (obsiously) but was overfitting a bit so a rate of 0.01 I feel was a more realistic value for the model while retaining a non-overfit character.

4 ROC Curves

Unfortunately, I could not imake ROC curves work. No matter what I tried I could not get them to sufficient results. The best I did can be found in the folder ROC. In the end the curves where just a diagonal and could not get my hands on the reason behind such a behaviour. I am afraid it had to do with the model.