Homework 4: Energy Based Models

DS-GA 1008 Deep Learning

Fall 2021

The goal of homework 4 is to test your understanding of Energy-Based Models, and to show you one application in structured prediction.

In the theoretical part, we'll mostly test your intuition. You'll need to write brief answers to questions about how EBMs work. In part 2, we will implement a simple optical character recognition system.

In part 1, you should submit all your answers in a pdf file. As before, we recommend using LaTeX.

For part 2, you will implement Energy Based Models by adding your code to the provided ipynb file.

As already mentioned before, please use numerator layout.

The due date of homework 4 is `23:55 EST` of `11/25`. Submit the following files in a zip file `your_net_id.zip` through NYU Brightspace:

- `hw4_theory.pdf`
- `hw4_practice.ipynb`

Note: we will subtract points for Campuswire posts containing solutions to problems. Campuswire shouldn't be a platform where you can get your solution checked, the goal is to help you with any misunderstandings associated with the homework

The following behaviors will result in penalty of your final score:

1. 5% penalty for submitting your files without using the correct format. (including naming the zip file, PDF file or python file wrong, or adding extra files in the zip folder, like the testing scripts from part 2).
2. 20% penalty for late submission within the first 24 hours. We will not accept any late submission after the first 24 hours.
3. 20% penalty for code submission that cannot be executed using the steps we mentioned in part 2. So please test your code before submit it.

1 Theory (50pt)

1.1 Energy Based Models Intuition (15pts)

This question tests your intuitive understanding of Energy-based models and their properties.

(a) (1pt) How do energy-based models allow for modeling situations where the mapping from input x_i to output y_i is not 1 to 1, but 1 to many?

(b) (2pts) How do energy-based models differ from models that output probabilities?

(c) (2pts) How can you use energy function $F_W(x, y)$ to calculate a probability $p(y \mid x)$?

(d) (2pts) What are the roles of the loss function and energy function?

(e) (2pts) Can loss function be equal to the energy function?

(f) (2pts) Why using only positive examples for energy (pushing down energy of correct inputs only) may lead to a degenerate solution?

(g) (2pts) Briefly explain the three methods that can be used to shape the energy function.

(h) (2pts) Provide an example of a loss function that uses negative examples. The format should be as follows $\ell_{\text{example}}(x, y, W) = F_W(x, y)$.

1.2 Negative log-likelihood loss (20pts)

Let's consider an energy-based model we are training to do classification of input between n classes. $F_W(x, y)$ is the energy of input x and class y . We consider n classes: $y \in \{1, \ldots, n\}$.

(a) (2pts) For a given input x, write down an expression for a Gibbs distribution over labels y that this energy-based model specifies. Use β for the constant multiplier.

(b) (5pts) Let's say for a particular data sample x, we have the label y . Give the expression for the negative log likelihood loss, i.e. negative log likelihood of the correct label (don't copy expressions from the slides, show step-by-step derivation of the loss function from the expression of the previous subproblem). For easier calculations in the following subproblem, multiply the loss by $\frac{1}{\beta}$.

(c) (8pts) Now, derive the gradient of that expression with respect to W (just providing the final expression is not enough). Why can it be intractable to compute it, and how can we get around the intractability?

(d) (5pts) Explain why negative log-likelihood loss pushes the energy of the correct example to negative infinity, and all others to positive infinity, no matter how close the two examples are, resulting in an energy surface with really sharp edges in case of continuous y (this is usually not an issue for discrete y because there's no distance measure between different classes).

1.3 Comparing Contrastive Loss Functions (15pts)

In this problem, we're going to compare a few contrastive loss functions. We are going to look at the behavior of the gradients, and understand what uses each loss function has. In the following subproblems, m is a margin, $m \in \mathbb{R}$, x is input, y is the correct label, $\bar{y}$ is the incorrect label. Define the loss in the following format: $\ell_{\text{example}}(x, y, \bar{y}, W) = F_W(x, y)$.

(a) (3pts) **Simple loss function** is defined as follows:

$$\ell_{\text{simple}}(x, y, \bar{y}, W) = [F_W(x, y)]^+ + [m - F_W(x, \bar{y})]^+$$

Assuming we know the derivative $\frac{\partial F_W(x,y)}{\partial W}$ for any x, y, give an expression for the partial derivative of the ℓ_{simple} with respect to W.

(b) (3pts) **Hinge loss function** is defined as follows:

$$\ell_{\text{hinge}}(x, y, \bar{y}, W) = [F_W(x, y) - F_W(x, \bar{y}) + m]^+$$

Assuming we know the derivative $\frac{\partial F_W(x,y)}{\partial W}$ for any x, y, give an expression for the partial derivative of the ℓ_{hinge} with respect to W.

(c) (3pts) **Square-Square loss function** is defined as follows:

$$\ell_{\text{square-square}}(x, y, \bar{y}, W) = \left([F_W(x, y)]^+\right)^2 + \left([m - F_W(x, \bar{y})]^+\right)^2$$

Assuming we know the derivative $\frac{\partial F_W(x,y)}{\partial W}$ for any x, y, give an expression for the partial derivative of the $\ell_{\text{square-square}}$ with respect to W.

(d) (6pts) **Comparison**:

(a) Explain how NLL loss is different from the three losses above.

(b) What is the role of the margin in hinge loss? Why do we take only the positive part of $F_W(x, y) - F_W(x, \bar{y}) + m$?

(c) How are simple loss and square-square loss different from hinge loss? In what situations would you use simple loss, and in what situations would you use square-square loss?

2 Implementation (50pts + 30pts)

Please make a copy of this notebook hw4_practice.ipynb and add your solutions. Please use your NYU account to access the notebook. The notebook contains parts marked as TODO , where you should put your code or explanations. The notebook is a Google Colab notebook, you should copy it to your drive, add your solutions, and then download and submit it to Brightspace. You're also free to run it on any other machine, as long as the version you send us can be run on Google Colab.

There are 3 parts in the notebook:

1. (50pts) Part - 1 deals with training the energy based model with your viterbi implementation.
2. (15pts, Extra Credits) Part - 2 introduces the GTN framework which are popular in Automatic Speech Recognition and Handwriting Recognition.
3. (15pts, Extra Credits) Part - 3 is an open ended part. Here, you will be experimenting with what you have coded on the handwritten data.