Homework Assignment # 3

Due: Friday, November 4, 2022, 11:59 p.m.

Total marks: 100

Question 1. [30 MARKS]

In class, we have discussed the use of autoencoders to generate a new data representation. In this assignment, we will extend this idea, using a Supervised Autoencoder (SAE) in order to jointly learn a new representation of the data which is useful for prediction and obtains low generalization error. The variables used for our autoencoder are:

1. $\mathbf{x} \in \mathbb{R}^{1\times d}$ - input
2. $\mathbf{y} \in \mathbb{R}^{1\times m}$ - supervised label for m classes
3. $\mathbf{W}^{(2)} \in \mathbb{R}^{d\times h}$ - Weights for first layer
4. $\mathbf{W}_x^{(1)} \in \mathbb{R}^{h\times d}$ - Weights for second layer $\mathbf{x}$ head
5. $\mathbf{W}_y^{(1)} \in \mathbb{R}^{h\times m}$ - Weights for second layer $\mathbf{y}$ head

The supervised autoencoder which we use in this assignment minimizes the joint loss function

$$\ell(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)}, \mathbf{W}_y^{(1)}) = \ell_y(\text{softmax}(\mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_y^{(1)}), \mathbf{y}) + \beta\ell_x(\mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_x^{(1)}, \mathbf{x})$$

using the backpropagation algorithm for all samples $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}$. Notice that the predictions outputed by the networks are $\hat{\mathbf{y}} = \text{softmax}(\mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_y^{(1)})$ and $\hat{\mathbf{x}} = \mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_x^{(1)}$. We will use the mean squared loss for ℓ_x and the multinomial logistic regression cross-entropy loss for ℓ_y

$$\ell_x(\mathbf{x}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)}) = \frac{1}{2}\|\mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_x^{(1)} - \mathbf{x}\|_2^2$$

$$\ell_y(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_y^{(1)}) = \ln(\exp(\mathbf{x}_i\mathbf{W}^{(2)}\mathbf{W}_y^{(1)})\mathbf{1}) - \mathbf{x}_i\mathbf{W}^{(2)}\mathbf{W}_y^{(1)}\mathbf{y}^\top$$

(a) [10 MARKS] In order to perform this minimization, we must first compute the derivative of this loss. In this question, you will analytically compute several terms and report their dimensions in terms of d, m, and h. You may use these terms in your answers:

1. $\mathbf{h} \stackrel{\text{def}}{=} \mathbf{x}\mathbf{W}^{(2)}$
2. $\theta_x \stackrel{\text{def}}{=} \mathbf{h}\mathbf{W}_x^{(1)}$
3. $\theta_y \stackrel{\text{def}}{=} \mathbf{h}\mathbf{W}_y^{(1)}$

Notice that

$$\delta_y^{(1)} = \nabla_{\theta_y}\ell_y(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_y^{(1)}) = \text{softmax}(\mathbf{x}\mathbf{W}^{(2)}\mathbf{W}_y^{(1)}) - \mathbf{y}$$

which we derived for multinomial logistic regression. On Page 92 of the notes, we similarly re-use this part when computing these gradients for logistic regression with only two classes. We've started off the derivation for you by giving you this first term; your job is to complete the rest.
Compute the following derivatives and report their dimension. Show all work for full marks.

1. (2 marks) $\delta_x^{(1)} = \nabla_{\theta_x}\ell_x(\mathbf{x}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)})$.

2. (2 marks) $\nabla_{\mathbf{W}_y^{(1)}} \ell(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)}, \mathbf{W}_y^{(1)})$
3. (3 marks) $\delta^{(2)}$ (i.e. $\nabla_{\mathbf{h}} \ell(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)}, \mathbf{W}_y^{(1)})$ see Chapter 9.2.3 in the notes)
4. (3 marks) $\nabla_{\mathbf{W}^{(2)}} \ell(\mathbf{x}, \mathbf{y}; \mathbf{W}^{(2)}, \mathbf{W}_x^{(1)}, \mathbf{W}_y^{(1)})$

(b) [10 MARKS] Notice that when $\beta = 0$, the portion of the loss that depends on $\mathbf{W}_x^{(1)}$ drops out and we are left with a standard single hidden-layer neural network with a linear activation on the first layer. Let the number of classes equal 2, so $\mathbf{y} \in \{0, 1\}$. Show that this neural network is equivalent to logistic regression regardless of the number of hidden layers, $h > 0$.

(c) [5 MARKS] What does this result, in Part b, tell us about neural networks with linear activations and many hidden layers (e.g., 100 hidden layers)?

(d) [5 MARKS] Does our SAE ($\beta > 0$) suffer from the same problem? Why or why not? Justify your answer mathematically.

Question 2. [40 MARKS]

Implement the supervised autoencoder in A3.jl.

Question 3. [30 MARKS]

In this question, we will run and evaluate our SAE.

(a) [7 MARKS] Implement the evaluation metrics in A3.jl.

(b) [10 MARKS] Implement internal k-fold cross-validation for hyperparameter selection.

(c) [8 MARKS] Implement cross-validation using repeated random sampling (RRS), to evaluate the mode with that best hyperparameter.

(d) [5 MARKS] Explain how you would change the implementation of RRS in A3.jl, to do nested cross-validation. Be specific, including identifying how the inputs to the RRS procedure would have to change. Ideally, give a chunk of julia code or pseudocode showing the new implementation. But do not include that code inside A3.jl, only include it here in the written answers.

Homework policies:

Your assignment should be submitted on eClass as a single pdf document and a zip file containing: the code (a .jl file), a .html file of the pluto notebook with all the cells run. The answers must be written legibly and scanned or must be typed (e.g., Latex). All code should be turned in when you submit your assignment. This means submitting the completed Pluto notebook, where you took the Pluto notebook with `todos` and completed them with your implementation. You are not allowed to change any of the imports in the notebook.

Because assignments are more for learning, and less for evaluation, grading will be based on coarse bins. **The grading is atypical**. For grades between (1) 80-100, we round-up to 100; (2) 60-80, we round-up to 80; (3) 40-60, we round-up to 60; and (4) **0-40, we round down to 0**. The last bin is to discourage quickly throwing together some answers to get some marks. The goal for the assignments is to help you learn the material, and completing less than 50% of the assignment is ineffective for learning.

We will not accept late assignments. Plan for this and aim to submit at least a day early. If you know you will have a problem submitting by the deadline, due to a personal issue that arises, please contact the instructor as early as possible to make a plan. If you have an emergency that prevents submission near the deadline, please contact the instructor right away. Retroactive reasons for delays are much harder to deal with in a fair way.

All assignments are individual. All the sources used for the problem solution must be acknowledged, e.g. web sites, books, research papers, personal communication with people, etc. Academic honesty is taken seriously; for detailed information see the University of Alberta Code of Student Behaviour.

Good luck!