STA4026S Analytics – Neural Networks.

Assignment 2 2024 (35 Marks)

Spam Bot Data

You are a Data Scientist consulting for a social media platform. A major issue for the present social media platform is the proliferation of spam comments (any unwanted or unsolicited communication) on their video hosting platform. (Think YouTube.) You've been tasked with developing a spam detection algorithm which aims to distinguish between human and non-human ('robot') comments based on text processing heuristics. These heuristics are stand-alone statistics, scores, or flags calculated by separate algorithms on the platform. You are given a number of observations on the following variables (the heuristics):

Variable	Description
`Spam_Bot`	Response, outcomes encoded as `'Human'` or `'Robot'`.
`Grammar_Score`	Continuous: A score of grammatical accuracy. Low = poor grammar, high = good grammar.
`Emoji_Score`	Continuous: A measure of abnormal use of emojis (picture characters). Low scores = normal, high scores = unusual or abnormal.
`Lure_Flag`	Categorical heuristic returned by lure detection algorithm which returns `'grey'` if text is deemed innocuous, `'blue'` if small probability of containing lure, `'red'` if high probability of a lure attempt. Lures are direct or indirect techniques for luring users to unsafe web domains.
`Sentiment`	Categorical heuristic indicating perceived sentiment of the text. Three levels `'Negative'`, `'Neutral'`, `'Positive'`.

For purposes of modelling the present dataset, we consider a type of double hidden layer Siamese neural network, which we will term a split-brain ($2m$,2)-network, where the hidden layers are split into two **independent** sub-networks or hemispheres with an identical number of nodes (here, m) on the first hidden layer, each taking as input a distinct subset of features of the network, followed by a single node on each hemisphere for the second layer. Once evaluated, the activations from the hemispheres are treated as usual for the remainder of the forward pass. The two activations from each of the hemispheres enter the output layer. Figure 1 gives a directed graph of the neural network.

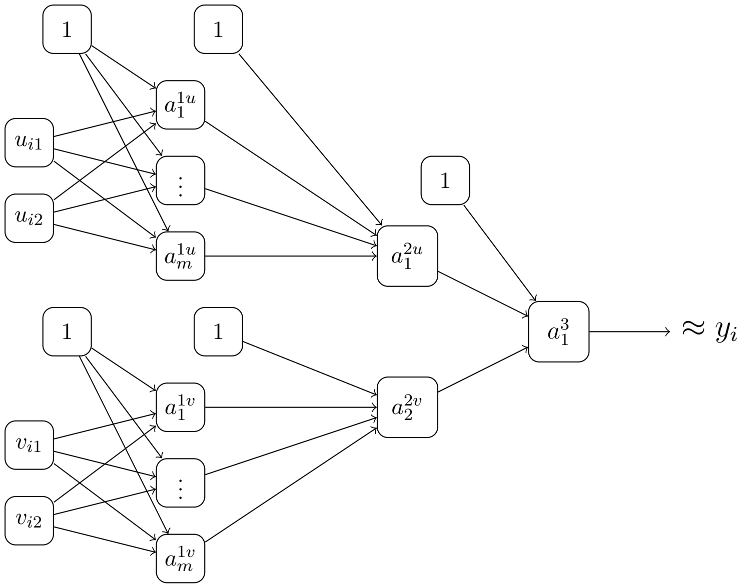

Figure 1: Directed graph of a 'split-brain' network. Note there are two sub-networks or 'hemispheres' in the initial layers of the network, hence activations incur the additional superscripts u and v to indicate which sub-network (and inputs) they pertain.

Note: I've added leading indicators to each question telling you what should be reflected in your compiled write-up. To that end, here's what they mean explicitly:

- Render code: This means, I want to see your R code typeset at that point in your document. So make sure the code-chunk options are set such that the code renders.
- Render plots: This means, I want to see a plot rendered in the write up.
- In-text response: This means I want a paragraph-style response where you type out your response in one or a few sentences.
- Combinations of the above: If you see R code + plots, that means render the R code and the plot, and no typed response is required. If you see R code, that means just render the R code and no other response is required.

Problem Set

1. (a) **Render code and output**: Write R code that encodes the responses and input matrices appropriately. For the `Lure_Flag` and `Sentiment` variables, encode as numeric with three levels (1,2,3). Use the `head(.,4)` function to print the first 4 rows of the input matrices to show that they are correctly encoded. (2)

   (b) **Render code**: Write an R-function (call it `g(AL,Y)` as in the backprop lectures; `AL` = `Yhat`) that evaluates the cross-entropy error function: (3)

$$\text{Obj} = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i))$$

   where $y_i \in \{0, 1\}$.

   Render typed response: Are there any potential numerical pitfalls in evaluating this particular objective function? (If so, make sure to adress this in your code.)

   (c) **Write-up**: Let $\mathbf{U}$, and $\mathbf{V}$ denote the $N \times 2$ and $N \times 2$ input matrices corresponding to the 'u' and 'v' sub-networks respectively, $\mathbf{Y}$ the $N \times 1$ matrix of responses. Then let $\mathbf{W}_{1u}$ and $\mathbf{b}^{1u}$ denote the weight matrix and bias terms for the first hidden layer in the u-hemisphere, $\mathbf{W}_{2u}$ and $\mathbf{b}^{2u}$ for the second hidden layer on the u-hemisphere and so on. Explain how this network can be evaluated as a standard $(2m, 2)$-network and give appropriate expressions/structures for the elements of the standard matrix-form updating equations in terms of those given here. (5)

   (d) **Render code**: Write an R-function that evaluates a forward pass of the split-brain network with m nodes on each of the hemispheres of the first hidden layer and one on the second. Use tanh activations on all hidden nodes. This function must also include evaluation of an appropriate objective function and regularisation mechanism for the present dataset (let ν denote the regularization parameter and use the L1 norm). For purposes of the present dataset, we will group the `Gramar_Score` and `Emoji_Score` variables into inputs for the left hemisphere, and `Lure_Flag` and `Sentiment` for the other. (5)

   Failover: If you cannot figure out (b) and (c) you may use a standard $(m, 2)$-network to conduct the analysis but you will be forgoing some marks for that. In any case, see if you can get the analysis to work for the standard case first...

   (e) **Render code + plot + in text response**: Conduct a validation analysis for $m = 5$ using standard R optimization routines in order to fit the models. Note: no scaling is required for the input variables here. Use `set.seed(2024)` before splitting the data into 80-20 training vs. validation. Plot the validation error vs. ν and use this figure to motivate your choice of regularization level. Indicate the level of regularisation chosen and report the value. (5)

   (f) **Render code + plot + in text**: Plot the magnitudes of the parameters of the regularized model and comment on what the resulting figure implies about connections between nodes in the final model network. Hint: `type = 'h'`. (3)

   (g) **Render code + plot + in-text response**: Use the regularized model to construct response curves over `Gramar_Score`, `Emoji_Score`, and **Lure_Flag** (4) for neutral sentiment. Superimpose the observat Hint: You'll have to make three plots. If you are unfamiliar with response curves, I show how to construct a 2D lattice in https://youtu.be/2ki7Yi2Hx3U – time stamp 57:30.

(h) **In-text response**: Are there any practical advantages you can envisage for using a split-brain network such as the above over a standard feed-forward architecture? (2)

(i) **In-text response**: Explain how the backpropagation algorithm can be used to calculate the partial derivatives of the output of the network with respect to the inputs. Two marks for explaining correctly. Full marks if you can give the appropriate equations. (3)

(j) Three marks for the quality of your write-up. These are (some of) the things I'll be looking for: (3)

- Are your figures 1-1 aspect ratio or close to it and vector graphics? (That is, not jpeg or png.)
- Are your figure captions full sentences and punctuated?
- Are your equations punctuated and properly typeset? Equations should read as sentences and thus should be punctuated.

Notes, Instructions, Etc.

Submission instructions:
For submission, I'd like you to give me

- A pdf containing your write-up and R code in an appendix. The write-up has an 8-page limit (excluding appendix). Use the naming convention `STDNUM001_STDNUM002_Analytics_2` for your file. Note the underscores.

- Your R code. Use the naming convention `STDNUM001_STDNUM002_Analytics_2024_A2.R` for your code file. Your R code should NOT contain any of the following:

  ```
  install.packages()
  rm()
  setwd()
  ```

  I want to be able to run your code on my computer without having to manually edit your code, installing libraries or calling to external files.

ANY deviation from the above conventions WILL be penalised. Doing so wastes my time and I've grown quite weary of such simple instructions being ignored at my expense.