ECE 5268 - Theory of Neural Networks (Spring 2017)
Mini-Project #3

Dr. Georgios C. Anagnostopoulos*

April 2017

1 Objectives

The objective of Mini-Project III is to expose the students to (i) Backpropgation Through Time (BPTT) and (ii) time series prediction via an Elman Recurrent Neural Network (ERNN). Please make sure you consult first the OverLeaf notes on Lecture 26, which provide important details on BPTT. Note that these notes are more reliable than the corresponding in-class notes, which may have a few mistakes here and there. Rest assured that the forward- and back-propagation-through-time pseudo-codes of Lecture 26 are verified and tested.

As usual, standard preparation guidelines (Section 3) and submission instructions (Section 4) are provided, which the students are expected to strictly adhere to. Finally, at the end of this document, a few, possibly helpful, references can be found.

2 Assignments

● **Task 1. [50 total points]**

Task 1 asks you to implement the performance and training phases of an ERNN, which uses linear activation functions for its output layer and logistic activation functions for its hidden layer. Furthermore, its training will consider the usual average squared loss function $E \triangleq 1/N \sum_{t=1}^{N} E(t)$, where N is the number of training samples and the "instataneous error" $E(t)$ is given as $E(t) \triangleq \|\mathbf{y}_2(t) - \mathbf{d}(t)\|_2^2$, where $\mathbf{d}(t)$ is the desired/target output vector for time t; when the outputs are scalars, then this error, obviously, simplifies to the squared error.

(a) **[10 points]** Implement first the performance phase of the ERNN, *i.e.*, the forward-propagation algorithm for this network. After completion, generate a random sequence s of 10 data points that have been sampled uniformly from $[0, 1]$. Based on this sequence, create a lagged vector-valued sequence as follows: $\mathbf{x}(1) \triangleq [s(1), s(2), s(3), s(4)]^T, \mathbf{x}(2) \triangleq [s(2), s(3), s(4), s(5)]^T, \ldots$, etc. Next, feed this input sequence into a 4-2-1 ERNN, whose weights have been randomly initialized according to the rules-of-thumb used for MLPs; make sure that you save all those weights. Feed the input sequence into the network and record/plot its outputs for $\mathbf{y}_1(0) = \mathbf{0}$ and $\mathbf{y}_1(0) = [0.5 \quad 0.5]^T$. Confirm that the two output sequences differ.

(b) **[10 points]** Implement a stochastic gradient method for training the ERNN that uses a constant learning rate to update the network's weights via BPTT. Initially, in order to test your implementation, use the generated data from the previous sub-task as training data and train the network for at least 5000 epochs (iterations through the entire training set) by updating $\widetilde{\mathbf{W}}$ only, leaving the remaining weights ($\widetilde{\mathbf{V}}$ and $\mathbf{U}$) to the values that were used to generate the data (*i.e.*, their optimal/true values). By picking a learning rate of 0.1 or lower, produce a plot of log-base-10 of the average squared loss

*georgio@fit.edu

versus the epochs. If the learning rate is sufficiently low, you should see a monotonically-decreasing learning curve.

(c) **[10 points]** Repeat the same as the previous sub-task (include plot) by only changing $\mathbf{U}$ while holding the remaining weights constant at their optimal value and report the corresponding plot. How does this case compare to the previous one; comment accordingly.

(d) **[10 points]** Repeat the same as the previous sub-task (include plot) by allowing all weights to change. After 5000 epochs compare the resulting weight values to their optimal ones (the ones that generated the data) and comment accordingly. Did you get what you expected?

(e) **[10 points]** Derive a modified BPTT that also computes the instantaneous gradient for updating the initial condition $\mathbf{y}_1(0)$. After implementing this version, train the ERNN for 5000 epochs by using a random initial value of $\mathbf{y}_1(0)$ and using optimal weight values. Comment on your obtained results. What do you conclude and based on what?
In what follows next (task 2), do not use this modified BPTT version.

● **Task 2. [50 total points]**

Download the MacKey-Glass time series dataset from http://web.cecs.pdx.edu/~mcnames/DataSets/MackeyG17.zip. This task is about predicting the series' next value based on its current and previous values. Despite being noise-free, the series is notoriously difficult to predict as it exhibits chaotic traits (*i.e.*, it is very sensitive on its initial conditions). More about this series and what it represents can be found online.

(a) **[10 points]** Plot the first 100 values of the time series versus time and comment on its nature.

(b) **[10 points]** From the first 50 samples of the original sequence, call it s, create a dataset, whose t^{th} input/output sample is given as $\mathbf{x}(t) \triangleq [s(t-\ell+1), s(t-\ell+2), \ldots, s(t-1), s(t)]^T$ and $d(t) \triangleq s(t+1)$, where the lag ℓ is set to 17. This dataset will be used for one-step-ahead prediction of s. For that purpose, use an ERNN with 1 hidden node and 1 output node. Train the network for 5000 epochs using a fixed $\mathbf{y}_1(0) = \mathbf{0}$ and a learning rate of 0.01 or lower in order to have a monotonic decrease in loss. Plot the obtained learning curve (log-base-10 of averaged squared loss versus epochs) and, in another plot, compare the obtained predicted sequence to the actual MacKey-Glass sequence values on a second "test" dataset that is based on the first 100 samples of the MacKey-Glass sequence.

(c) **[10 points]** Repeat the previous sub-task using a truncated BPTT with $\tau = 10$ and compare to the previous results.

(d) **[10 points]** Repeat sub-task (2.b) but, this time, for 2-step-ahead prediction. Compare the obtained results with ones from (2.b) and comment accordingly.

(e) **[10 points]** Consider a naive one- and 2-step-ahead model, which uses the current value $s(t)$ as the predictor for $s(t+1)$ and $s(t+2)$. Compute the average squared loss of these two naive models and compare them to similar results obtained from tasks (2.b) and (2.d) on the test dataset. Finally, comment on the differences you will observe.

3 Preparation Guidelines

Below are some general guidelines that should be followed, when compiling a Mini-Project report. I strongly encourage you to stick to them, so that you receive full credit for your correct responses.

- **Task Statements:** Before attempting to address a particular task, ensure that you completely understand what is asked from you to perform and/or to produce. When in doubt, come to ask me for clarifications! Also, make sure you did not omit your response to any of the parts that you have attempted. Finally, make sure that it is crystal clear, which response corresponds to which task/part.
- **Derivations & Proofs:** If you provide handwritten derivations and/or proofs, make sure you use your best handwriting. Each derivation should have a logical and organized flow, so that it is easy to follow and verify.
- **Code & Data:** The code that you author should be as well organized as possible and amply commented. This is very useful for assessing your work, as well as for you, while you are debugging/or modifying it, or if you have to go back to it in the near future. Regarding the data you generate, keep them organize and document somehow (*e.g.*, in a text file) the specifics of how they were generated. **Caution:** <u>You are not allowed to use any code and/or data that you have not produced</u> without my explicit prior permission, in which case the sources you have obtained these from must be clearly indicated in your code or data description as well in your report. You are deemed to be plagiarizing, if you fail to do so, which may have <u>dire consequences</u> to your academic tenure here at Florida Tech!
- **Figures, Plots & Tables:** Plots should have their axes labeled and, if featuring several graphs, an appropriate legend should be used. Whether figures, plots or tables, each one of these elements should feature a caption with sufficient information on what is being displayed and how were these results obtained (*e.g.*, under what experimental conditions or settings, etc.). You should ask yourself the question: if someone comes across it, will they understand about what is being depicted? Apart from a concise description, major, relevant conclusions stemming from the display should also be included in the caption text.
- **Observations, Comments & Conclusions:** When stating observations about a particular result, do not stop at the obvious that anyone can notice (*e.g.*, "*... we see that the curve is increasing.*"). Instead, assess whether the result is expected, either by theory or intuition (*e.g.*, "*... This is as expected, because X is the integral of ...*"), or, if it is unexpected, offer a convincing reasoning behind it (*e.g.*, "*... We expected a decreasing curve ... All points to that I must have not been calculating X correctly ...*"). The latter is more preferable (*i.e.*, expect partial credit) than stopping at the obvious, which happens to be wrong (*i.e.*, do not expect partial credit). Next, descriptions and comments on results should be sufficient. Be concise, but complete. Finally, conclusions that you draw must be well-justified; vacuous conclusions will be swiftly discounted.

4 Submission Instructions

Kindly adhere to the conventions and submission instructions outlined below. Deviations from what is described here may cause unnecessary delays, costly oversights and immense frustrations related to the assessment of your hard work.

First, store all your Mini-Project deliverables in a folder named **lastname_mpX**, where "lastname" should be your last name and X should be the number of the Mini-Project, like 1, 2, etc. The folder name should be all lower case. For example, my folder for Mini-Project 1 would be named *anagnostopoulos_mp1*. Secondly, your **lastname_mpX** folder should have the following contents:

- An Adobe PDF document named **lastname_report.pdf**, where, again, "lastname" should be replaced by your last name in all lower case, *e.g.*, *anagnostopoulos_report.pdf*. This document should contain <u>your entire</u> Mini-Project report as a <u>single document</u>. This will be the document that will be graded. Also, here are some important things to keep in mind:

- The report <u>must</u> include a signed & dated copy of the Work Origination Certification page. You can either scan such a page and include it in your document, or sign and date it electronically, as long as your signature is not typed. If this page is missing from your report, or it does not comply with the aforementioned conditions, I reserve the right not to accept the report and assign a score of 0/100 for the relevant Mini-Project.
- The Mini-Project may ask you to produce a variety of derivations, proofs, etc. You are not obliged to type such parts; it would be nice, but I realize that such effort would be quite time-consuming. Instead, you can import scanned images (or whole pages) of your handwritten work, as long as they are legible and well organized, so that the report has a clear logical flow. For example, it has to be clear where this hand-written work corresponds to (*e.g.*, which assignment it addresses).
- Having said all this, you may want to consider to print out your typed work, appropriately merge it with any handwritten pages (don't forget the signed Work Origination Certification page!) and then scan the whole compilation into a single PDF report, say, in the Library. **Caution:** when scanning, use a relatively low-resolution (DPI) setting, so your resulting PDF document does not become too big in size, which may prevent you from uploading your work to Canvas.

- A folder named **src**, which should contain all your MATLAB scripts that you authored and used for producing your results and the data sets that you created for this Mini-Project, if applicable.
- An optional folder named **docs**, in which you can include a MS Word version of your report and other ancillary material connected in one way or another to your Mini-Project report.

Next, compress your **lastname_mpX** folder into a single ZIP archive named **lastname_mpX.zip**; *e.g.*, mine would be called *anagnostopoulos_mp1.zip*.

Finally, upload your ZIP archive to Canvas by the specified deadline using the appropriate drop box. You are done!

References

WORK ORIGINATION CERTIFICATION

By submitting this document, I, ____, the author of this deliverable, certify that

1. I have reviewed and understood the Academic Honesty section of the current version of FITs Student Handbook available at http://www.fit.edu/studenthandbook/, which discusses academic dishonesty (plagiarism, cheating, miscellaneous misconduct, etc.)
2. The content of this Mini-Project report reflects my personal work and, in cases it is not, the source(s) of the relevant material has/have been appropriately acknowledged after it has been first approved by the courses instructor.
3. In preparing and compiling all this report material, I have not collaborated with anyone and I have not received any type of help from anyone but from the courses instructor.

Signature ____ Date ________________