BUAD5742 Convolutional Neural Net Assignment

Instructions

**You are required to:**

* Please submit the Assignment write-up, in one MSWord document, to our Blackboard classroom.
* Use Python and Keras for the assignment.
* Please use our coding template for this assignment. Jupyter notebooks are not acceptable for the code submission.
* The assignment should be submitted in report format. If you are uncertain, please peruse the guidelines at this website: <https://www.examples.com/education/report-writing-format.html>
* All source code used in the assignment must be attached in the appendix.
* Please decorate the code with explanatory comments, as the code may be reused in the future.
* The late policy, as in the syllabus applies.
* The assignment is W&M Honor Code Category C. The use of discussions, course materials and homework solutions are allowed, but you should write the ﬁnal solutions alone. Books, notes, and Internet resources can be consulted, but not copied from.

**Convolutional Neural Networks**

In this assignment, you will design and implement a Convolutional Neural Network (CNN) in one of the modern deep learning frameworks, Keras. We will use the CIFAR-10 Dataset for this project. Please complete the following tasks:

1. You will ﬁnd the CIFAR-10 dataset in Keras. Sample code to import the data looks like this:

(x\_train, y\_train), (x\_test, y\_test) = cifar10.load\_data()

1. Your network should have **at least 3 convolutional layers and at least 1 fully connected layer.** Describe your network and show a table of your network with each layer and relevant parameters (you might use the .summary() method. You may use those shown in class as a guide.
   1. Train your network. Describe your training procedure.
   2. Plot the following charts:
      1. Training and test loss (not classiﬁcation accuracy!) vs. training iterations.
      2. Classiﬁcation accuracy on the test set vs. training iterations.
   3. Comment on the change of performance. Does the network train faster? Generalize better? Discuss your result.
2. Experiment with preprocessing the input data. Start from the network in part 2.
   1. Try data augmentation (as used in the classroom, or elsewhere) of your dataset. Train your (second) network.
   2. Plot the following charts:
      1. Training and test loss (not classiﬁcation accuracy!) vs. training iterations.
      2. Classiﬁcation accuracy on the test set vs. training iterations.
   3. Comment on the change of performance. Does the network train faster? Generalize better? Discuss your result.
3. Experiment with network structure. Create a network that only has one hidden convolutional layer.
   1. Train your (third) new network.
   2. Plot the following charts:
      1. Training and test loss (not classiﬁcation accuracy!) vs. training iterations.
      2. Classiﬁcation accuracy on the test set vs. training iterations.
   3. Comment on the change of performance. Is depth important?
4. Experiment with network structure, again. Create a fourth network that has at least one more hidden layer than your first network in 2
   1. Train your (fourth) new network.
   2. Plot the following charts:
      1. Training and test loss (not classiﬁcation accuracy!) vs. training iterations.
      2. Classiﬁcation accuracy on the test set vs. training iterations.
   3. Comment on the change of performance. Is depth important?
5. Add dropout to your network in part 2 to create your fifth network.
   1. Train your (fifth) new network.
   2. Plot the following charts:
      1. Training and test loss (not classiﬁcation accuracy!) vs. training iterations.
      2. Classiﬁcation accuracy on the test set vs. training iterations.
   3. Comment on the change of performance. Is dropout important?
6. What are your overall comments and conclusions on your results?
7. For each line of the code that you used for the assignment, other than those containing ‘from’ and/or ‘import’, please insert a comment above stating what each line does.
   1. Please place a comment block (code flow) below the ‘from/import’ block and above the code that describes in sentence form the overall flow of the code and the purpose of the code (what are we trying to accomplish?). You may use pseudocode, if desired.
   2. Please place a comment block below the ‘code flow’ block and above the code that lists each variable used and explains what each variable is used for.
   3. A single line comment may start with a #.
   4. A comment block should start with ‘’’ and end with ‘’’.
   5. If you are commenting two or more lines of code that are essentially identical, i.e. different variables but identical operations, you may use one comment above that code block. For code that has similar operations (model.add), but has different parameters, please comment each line.
   6. Please do not combine comment blocks.

Good, competent work would get a Proficient or Accomplished rating level on the grading rubric. Exemplary implies work that is over and above.