CSE343/CSE543/ECE363/ECE563: Machine Learning (PG)
Winter 2023

Assignment-3 Rubrics (40 points)

Release: March 15, 2023 (Wednesday) | Submission: March 22, 2023 (Wednesday)

Instructions

- **Institute Plagiarism Policy Applicable.** Both programming and theoretical problems will be subjected to strict plagiarism checks.
- This assignment should be attempted individually. All questions are compulsory.
- **Theory [T]**: For theory questions, only hand-written solutions are acceptable. Attempt each question on a different sheet & staple them together (for ease of checking). Do not start a new question at the back of the previous one. Do not forget to mention the page number (bottom center) and your credentials (bottom right) on each sheet. It must be submitted in *Assignment submission box 4* kept outside B-609, R&D block. Scanned PDFs are not acceptable.
- **Programming [P]**: For programming questions, the use of any one programming language throughout this assignment is acceptable (Python/R/MATLAB). For python, you must submit a single *.py* file named as *A3_RollNo.py*. For other programming languages, submit the files accordingly. Make sure the submission is self-complete & replicable, i.e., you are able to reproduce your results with the submitted files only. Use random seeds wherever applicable to retain reproducibility. Further, save & submit (in the zip) the trained ML model using either pickle or joblib.
- **Report.pdf**: Create a *.pdf* report of programming questions that contain your applied approach, pre-processing, assumptions, analysis, visualizations, etc.. Anything not in the report will not be evaluated. Alternatively, a well-documented *.ipynb* file (in addition to a single *.py* file mentioned in the previous bullet) with answers to all the questions may be submitted as a report. The report must be named as *A3_RollNo_Report.pdf* or *A3_RollNo_Report.ipynb*.
- **File Submission**: Submit a *.zip* named A3_RollNo.zip (e.g., *A3_PhD22100.zip*) file containing the report and code files.
- **Submission Policy**: Turn-in your submission as early as possible to avoid late submissions. In case of multiple submissions, the latest submission will be evaluated. Expect **No Extensions**. Besides, submission within 24 hours of the passing of the deadline will incur a penalty of 1 mark out of the total 6 marks allocated to this assignment. Submission, between 24 and 48 hours of the passing of the deadline, will incur a penalty of 2.5 marks out of the total 6 marks allocated to this assignment. Beyond this, late submissions will not be evaluated and hence will be awarded zero marks.
- **Clarifications**: Symbols have their usual meaning. Assume the missing information & mention it in the report. You are allowed to use any machine learning library until exclusively mentioned in the question that it is supposed to be done from scratch. You can always use basic python libraries such as numpy, pandas, and matplotlib, unless specified otherwise. Use Google Classroom for any queries. In order to keep it fair for all, no email queries will be entertained. You may attend office/TA hours for personal resolutions. Start your assignment early. No queries will be answered in Google Classroom comments 12 hours before the submission deadline.
- There could be multiple ways to approach a question. Kindly justify your answers mathematically in theory questions and via commented text in the programming questions appropriately. Questions without justification will get zero marks.

---

1. **[T ‖ CO1] Optimizers** **(4 points)**
Write and explain the description, mathematical equation, advantages, and disadvantages of the following model optimizers.
   (a) SGD
   (b) RMSProp
   (c) AdaGrad
   (d) Adam

2. **[P ‖ CO2 & CO3] Autoencoder** **(6 points)**
Pick random 1000 images (Train- 900, Test-100) of 5 Superclasses of the CIFAR-100 dataset (https://www.cs.toronto.edu/~kriz/cifar.html, https://www.cs.toronto.edu/~kriz/cifar-100-python.tar.gz) and train an autoencoder to regenerate the images. Apply batch normalization and plot the loss vs. epoch training curve. Print a 5x2 grid containing 1 test image of each class, in which the first column contains the original image and the second column contains the autoencoder output of the same image.
Now, use the latent embeddings learned in the above autoencoder to build a five-class classifier. Show performance on train and test sets using accuracy as an evaluation metric.

3. **[P ‖ CO3 & C4] SVM** **(10 points)**
Use the following SVM strategies to classify sapodillas from kiwi for the dataset available here[1]. For each strategy, plot the learned decision boundary. Use an appropriate evaluation metric to present the test scores.
   (a) Linear SVM
   (b) Polynomial SVM
   (c) Kernel SVM

   Finally, write the individual and combined inferences of the obtained results.

4. **[P ‖ CO3 & C4] Transfer Learning** **(20 points)**
Use the FashionMNIST dataset (https://keras.io/api/datasets/fashion_mnist/) from Keras to build one classifier each based on the following pre-trained architectures trained on IMAGENET dataset (reference: https://keras.io/api/applications/). Remove the last layer, add your own few dense layers and the output layer. Train only these added layers from scratch on the data. Use an appropriate evaluation metric.
   (a) VGG16
   (b) VGG19
   (c) ResNet50V2
   (d) MobileNet
   (e) EfficientNetB0

   Finally, write the individual and combined inferences of the obtained results.

[1] https://drive.google.com/drive/folders/1nDb5K2-Nz0Er41-X6Fny11bjK-cTft-x?usp=sharing