MIT CSAIL

6.819/6.869 Advances in Computer Vision

Spring 2022

Problem Set 6

Posted: Thursday, March 31, 2022 **Due:** Thursday 23:59, April 7, 2022

6.869 and 6.819 students are expected to finish all problems unless there is an additional instruction.

We provide a python notebook with the code to be completed. You can run it locally or in Colab (upload it to Google Drive and select 'open in colab') to avoid setting up your own environment. Once you finish, run the cells and download the notebook to be submitted.

Submission Instructions: Please submit a .zip file named your $\langle kerberos \rangle$.zip containing 1) a pdf document named **report.pdf** including your answers to all required questions with images/plots. Copy the relevant lines into your PDF writeup. 2) the python notebook provided, with all the cells executed and the relevant source code, which should be named **notebook.ipynb**. No other files or folders may be included, and not following the naming convention will result in grade penalties.

Special Submission Instructions for this Pset: 1) This Pset has longer text, so please be patient and read carefully. 2) This Pset requires model training (in Problem 1 and Problem 5) so please start early. 3) In preparing your write-up, please also explain briefly your code to avoid false negative grading.

Late Submission Policy: If your Pset is submitted within 7 days (rounding up) of the original deadline, you will receive partial credit. Such submissions will be penalized by a multiplicative coefficient that linearly decreases from 1 to 0.5.

Owing to the success of machine learning, vision algorithms have made rapid headway in various aspects of our life, from video surveillance, automated resume screening to clinical diagnostics. Due to their usage in a wide variety of situations, ethical concerns arise during each step of development, including data collection, model development, and model deployment. All these issues oftentimes lead to unfair products: for example, racial and gender biases have been reported in many instances of face detection algorithms [1]. Similarly, a recent study has shown that some computer vision models detecting COVID-19 from chest radiographs rely on confounding factors rather than medical pathology, creating an undesired situation in which the systems may appear accurate, but could lead to catastrophic failure when tested in new hospitals [2]. Under certain circumstances, these biases can lead to harmful consequences, which makes it crucial for CV practitioners to be aware of the Social and Ethical Responsibilities of Computing (SERC) [1].

[1] https://computing.mit.edu/cross-cutting/social-and-ethical-responsibilities-of-computing/

The aim of this PSET is to study some common issues that may appear when developing a clinical decision support model. We will use the CBIS-DDSM/DDSM dataset, which is a database of scanned film mammography images. It contains normal, benign, and malignant cases[2]. You will train a simple model to classify these images, learn to evaluate the model using different metrics, and analyze some of the problems that arise when using these systems in practice. This Pset is not a comprehensive summary of the potential sources of bias/harm, and the problems studied are heavily simplified so that they can be studied concisely. However, we hope that this Pset and the references provided will give you a start to delve deeper into this important set of topics.

Problem 1 *Train an image classifier model (2 points)*

To begin with, you will use the experience you gained in Pset 5 to train a simple image model, that learns to produce a diagnosis in the categories: normal, malignant and benign. Experiment with some of the hyperparameters and techniques that are proposed in the notebook.

For the final set of techniques you choose, explain how they affect performance and (optionally), train using different hyperparameters. Your trained network should achieve a test accuracy of at least 90% to get a full score on the problem. Report the train, val and test Top-1 accuracy of your final network and your design choices (such as ResNet-18, data augmentation by rotation, SGD optimizer, learning rate of 0.5, 10% dropout).

For the rest of the Pset, you can either use your model (if you achieved a test accuracy higher than 93%) or the one trained by the instructors that is automatically downloaded in the notebook.

Problem 2 *Metrics (4 points)*

In the previous question, you may have achieved high accuracy, but that doesn't necessarily mean that the model is useful for clinic decision support in practice. Suppose that your model just learns nothing and outputs normal for every image. If 90% of the patient images in your dataset are normal, the accuracy is still as high as 90%! In this question, we will introduce some other metrics commonly used in biomedical imaging tasks. **For a better understanding of these metrics, please don't use pre-defined metrics function from other libraries.**

(a) Confusion matrix (1 point).

Classification accuracy alone can hide important details if you have an unequal number of observations in each class or if you have more than two classes in your dataset. A confusion matrix helps you summarize the performance of a classification algorithm and can give you a better idea of what your classification model is getting right and what types of errors it is making. **Please plot the confusion matrix of your selected model, and include it in your report**. Your confusion matrix should follow the template given in Figure 1.

(b) AUPRC curve (1 point).

[2] `https://wiki.cancerimagingarchive.net/display/Public/CBIS-DDSM`

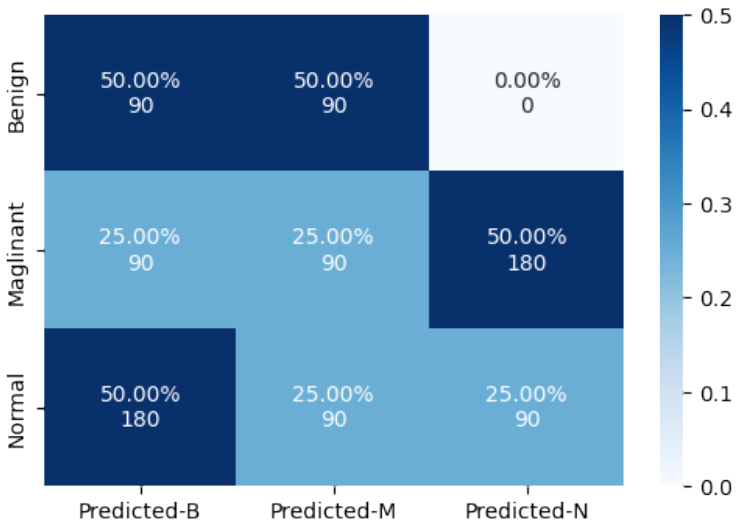

Figure 1: Confusion matrix template

Our dataset has 3 classes and in order to use some definitions in binary classification tasks, you need to create binary labels from the given class labels. Imagine a scenario where you really care about malignant cases. Please consider the malignant class (label 1) to be the positive class. And both benign (label 0) and normal (label 2) classes are negative.

Before proceeding with the following problems, please get familiar with the following terms commonly used in binary classification: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Besides, there are also Precision (P = TP/(TP+FP)) and Recall (R = TP/(TP+FN)). So which one should you look at when you are trying to improve the model? It really depends on the problem at hand. For example, if you are finding people with allergies, you don't want to alert too many people who may not have allergies therefore you want a high precision. However, in cases such as cancer prevention, a false negative is usually more disastrous than a false positive so you want a high recall. You could have 100% recall by predicting positive for everyone but then you have a completely uninformative model.

So this is where the precision-recall curve can be helpful in balancing the trade-off between precision and recall by using different probability cut-offs.

Plot the precision-recall curve of your selected model and make sure the area under the precision-recall curve (AUPRC, a number) is visible on the plot, and include it in your report.

(c) Fairness metric (2 points)

Modern computer vision algorithms are trained from data and thus are influenced by the statistical make-up of the training data. It's important to understand your input data as much as possible (while respecting privacy of course). Is your data sampled in a way that represents real-world settings?

The risk for breast cancer increases with age. Most breast cancers are diagnosed after age 50[3]. In this question, you will analyze 2 trained VGG models (`fairness_vgg_1.pt` and `fairness_vgg_2.pt`) together with the age distribution for the malignant class of the dataset that these 2 models were trained on. Besides accuracy, we will introduce a fairness metric to evaluate the model.

Please answer the following questions.

i) Which model has a higher accuracy when evaluated using the testset `fairness_testset`?

ii) To date, a number of algorithmic fairness metrics have been proposed. In this question, we want you to evaluate the model performance using a fairness metric which is called equalized odds, which is TP / (TP + FN). Equalized odds are achieved if the sensitivities in the subgroups are close to each other.

What you will need to do is consider 2 subgroups: age [60, 70) and age [70, max) and **generate a table similar to this template, which needs to be included it in your report**. Please also plot the age distribution for the malignant class used in training both models using `fairness_vgg_1.csv` and `fairness_vgg_2.csv`. If you have to choose a model that will be applied to an age group similar to our `fairness_testset`, **which model will you choose? Please briefly explain. You can also think about different scenarios where one model is preferred over the other.** (Be careful: `test_age.csv` has the age info for all three classes in the test set, and it only makes sense to compare malignant class age distribution)

	fairness_vgg_1		fairness_vgg_2	
	Age [60, 70)	Age [70, max)	Age [60, 70)	Age [70, max)
equalized odds				

Figure 2: Equalized odds template

There are many other fairness metrics such as `proportional parity`, which is (TP + FP) / (TP + FP + TN + FN) and also `predictive rate parity`, which is TP / (TP + FP). Based on your accuracy/confusion matrix/AUPRC/fairness evaluation in Problem 2, you should have a better idea that there are many different criteria in which a model can be performing well or poorly, and even if it's performing well in one criterion, it may be performing poorly in others. Different metrics can evaluate different types of success/failure, and it's important to use metrics that are sensitive to the things that you care about in your application, which may include metrics that are more than just Top-1 accuracy.

Problem 3 *Where is the model looking at? (1 point)*

A common technique used to delve deeper into trained models is visualizations. As studied in the previous PSET, filter visualizations are sometimes useful to understand how the network behaves, but it is hard for non-expert users to interpret them.

[3] `https://www.cdc.gov/cancer/breast/basic_info/risk_factors.htm`

Another technique previously studied is CAM visualizations, which provide a simple way to interpret what regions of the image contribute to the prediction for each output target. This is a useful tool to check whether the model is picking up spurious correlations or the predictions focus on regions of the image that are consistent with expert knowledge. Run Grad-CAM (or another CAM visualization) on several images for the different test sets, and explain the differences you see between correctly and incorrectly classified samples.

Problem 4 *Evaluate on external dataset (2 points 6.819, 3 points 6.869)*

In this question, we will study a problem that occurs frequently in practice when deploying systems. Oftentimes, the data distribution at test time is different from the one used in training.

(a) To begin with, compare the performance with the original model against an external dataset of the same type, provided in the notebook under *data/external_dataset* visually. (1 point)

(b) Compare samples of the two datasets visually, and report the differences that you appreciate. (Hint: if you need inspiration, check typical image transformations in PyTorch under *torchvision.transforms*) (1 point)

(c) **(6.869 only)** Implement a set of transformations that can be applied to the original dataset at train time that would mitigate this issue. Plot the original dataset with the proposed transformations, which should produce qualitatively similar samples as the external dataset. Although you don't need to retrain the model, you are free to do so to further check that the transformations are correct (performance on the external dataset should improve using the original data with the extra transformations). Use some of the available PyTorch transformations under *torchvision.transforms*. (1 point)

Problem 5 *Class imbalance (2 points 6.819, 4 points 6.869)*

An issue typically present in medical datasets is class imbalance: patients that do not present the disease are more frequent than patients that do have the disease. One way to make the trained model perform differently is by simulating a dataset that contains a different number of samples of each category. This can be achieved by at least two methods: reweighting the loss and changing the sampling procedure.

(a) Implement reweighting of the loss so that all classes contribute the same to the loss. (1 point)

(b) Modify the dataset class so that samples for the three classes are balanced. (1 point)

(c) **(6.869 only)** Finally, train your original model with the two modifications, plot the confusion matrix for them, and explain the empirical (if there are) and theoretical differences between the two techniques. Hint: think about gradient estimation in the extreme case where the number of samples of the less predominant class is low. (2 points)

Reading list

- Standard deep learning models can be trained to predict race from medical images with high performance even when models are optimized to perform clinically motivated tasks: Reading Race: AI Recognises Patient's Racial Identity In Medical Images(`https://arxiv.org/abs/2107.10356`)
- Understanding Potential Sources of Harm throughout the Machine Learning Life Cycle by Harini Suresh and John Guttag. (`https://mit-serc.pubpub.org/pub/potential-sources-of-harm-throughout-the-machine-learning-life-cycle/release/1`)

References

[1] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.

[2] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. Ai for radiographic covid-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 3(7):610–619, 2021.