HOMEWORK 5: APPLIED MACHINE LEARNING

CMU 10701: MACHINE LEARNING (FALL 2017)
OUT: Nov 15, 2017
DUE: Dec 4, 2017 2:00 pm
TAs: Easwaran, Logan, Guoquan
Student: INCLUDE NAME AND ANDREW ID IN WRITEUP

START HERE: Instructions

- **Collaboration Policy**: This assignment is to be done individually: each student must hand in their own answers. It *is* acceptable, however, for students to collaborate in figuring out how to solve the problems. We will be assuming that, as participants in a graduate course, you will be taking the responsibility to make sure you personally understand the solution to any work arising from such collaboration. You also must indicate on each homework with whom you collaborated.

- **Late Submission Policy:** You will be allowed 2 total late days without penalty for the entire semester. You may be late by 1 day on two different homeworks or late by 2 days on one homework. Weekends and holidays are also counted as late days. Late submissions are automatically considered as using late days.

  Once those days are used, you will be penalized according to the following policy:

  - Homework is worth full credit before the deadline.
  - It is worth half credit for the next 24 hours.
  - It is worth zero credit after that.

  You must turn in at least $n - 1$ of the n homeworks, even if for zero credit, in order to pass the course.

- **Submitting your work:**

  - **Gradescope:** For written problems such as derivations, proofs, or plots we will be using Gradescope. You can access the site here: https://gradescope.com/. Submissions can be handwritten, but should be labeled and clearly legible. If your writing is not legible, you will not be awarded marks. Alternatively, submissions can be written in LaTeX. Upon submission, label each question using the template provided. Regrade requests can be made, however this gives the TA the chance to regrade your entire paper, meaning if additional mistakes are found then points will be deducted.
  - **Autolab:** You can access the 10701 course on Autolab by going to https://autolab.andrew.cmu.edu/ The deadline displayed on Autolab may not correspond to the actual deadline for this homework, since we are allowing late submissions (as discussed in the late submission policy on the course site). Any attempt to "hack" Autolab or any other kind of code cheating will be dealt with according to university policy on student cheating.

- **Points:** there are a total of 100 points in this assignment.

1 Background and summary

A gene is a sequence of DNA containing instructions for building a single molecule. While most cells from the same individual organism share (roughly) the same genes, different cells use different subsets of these genes, and activate (build the molecule corresponding to) these genes different numbers of times. Single-cell RNA sequencing (scRNA-seq) provides measurements, for a single cell, of the activity levels of a set of annotated genes, called a gene expression profile. The type of a cell (e.g., lung cell, brain cell, skin epidermis) is closely connected with its gene expression profile. The ability to determine cell types based on the expression of the genes is currently a major human health challenge. The Chan Zuckerberg Initiative (CZI), founded by Mark Zuckerberg from Facebook, has recently pledged hundreds of millions of dollars to support the creation and analysis of the 'human cell atlas' — a catalog of the set of genes and processes that are active in all cells in our body. In this problem set you would develop and test methods that are critical for the success of this initiative — the ability to use the expression profiles to infer the types of cells being studied. Your task for this homework will be to

1. perform dimensionality reduction on single cell gene expression profile data, then
2. predict the types of certain cells using the reduced-dimension version of their gene expression profiles.

2 Input data

The datasets used for this homework are drawn from 104 separate scRNA-seq experiments, each of which profiles cells from different individuals or sets of individuals, and may focus on different cell types. **There will be no overlap in the set of experiments used in the training set and the set of experiments used in the test set;** however, we guarantee that the set of possible cell types seen in the test data will be a subset of those appearing in the training data. This is to ensure that the classifier does not learn to predict on **experimental biases** instead of gene expression profiles. On Autolab, you have access to a `.tar.gz` file containing the following data files in tab-separated-value (`.tsv`) format with `"\n"` newlines, with column names included and row names excluded:

- `train_covariates.tsv` — gene expression profiles (covariates) for the training set; it has $1 + n$ lines and p columns; its p columns correspond to genes; the first line contains column "names" *which are integers* denoting particular genes, and the following n rows correspond to cells; the (i, j)th entry is measurement of the gene expression level of the jth gene in the ith cell in the training set;
- `train_observed_labels.tsv` — cell types (labels) for the training set; it has $1 + n$ lines and a single column; the first line contains the column name "Cell Type", and the following n rows correspond to cells; the $(i, 1)$th entry is the cell type of the ith cell in the training set;
- `train_experiment_ids.tsv` — auxiliary information; experiment ID numbers ("accession numbers") for the training set; it has $1 + n$ lines and a single column; the first line contains the column name "Experiment ID", and the following n rows correspond to cells; the $(i, 1)$th entry is the ID number of the experiment that generated the gene expression profile for the ith training cell; **You may want to use this information when estimating the performance of a model you try**;
- `test_covariates.tsv` — gene expression profiles (covariates) for the test set; it has $1 + m$ lines and p columns; its p columns correspond to genes (the same ones as in the training set); the first line contains column "names" *which are integers* denoting particular genes, and the following m rows correspond to cells; the (i, j)th entry is measurement of the gene expression level of the jth gene in the ith cell in the test set; the rows will not be grouped or ordered by experiment ID and/or cell type as they are in the training data;

Warning: please note that this dataset is large; both coding and computation may take considerably longer than the implementation problems in other homework.

3 Tasks

To query the human genome atlas, researchers and clinicians would need to upload their data to a server that stores the classifiers. However, working with the large ($\sim$20K) dimension datasets maybe both time consuming and require huge bandwidth. One way to solve this is to reduce the dimension of the data prior to using the classifier. Therefore, there are two tasks for this assignment: dimensionality reduction and subsequent classification.

3.1 Dimensionality reduction

Learn a mapping from p-dimensional gene profiles to $\mathbb{R}^{100}$. Use this mapping to transform the training data into a $(n \times 100)$-dimensional matrix, and the test data into an $(m \times 100)$-dimensional matrix. (You will not submit these matrices on Autolab, but will use them in the next part.)

3.2 Classification

Learn a mapping from the $\mathbb{R}^{100}$ vectors from the previous part to cell types. The goal is to minimize the misclassification rate on the test set. Autolab will **NOT** indicate what your performance is on the test set, so you may want to estimate your out-of-sample misclassification rate using subsets of the training set. Prepare a `.tsv` file named `test_predicted_labels.tsv` in the same format as `train_observed_labels.tsv` (but with $1 + m$ lines rather than $1 + n$) containing the column name "Cell Type" followed by the predicted labels for each of the m test examples in the same order that they appear in `test_covariates.tsv`.

4 Submission

Submit the following in a folder named `hw5` within a `.tar.gz` file (try `tar -zcvf some_name.tar.gz hw5`) on Autolab:

- `test_predicted_labels.tsv` — the predicted labels for the test set as described above;
- `code` — a directory containing the code used to generate these labels from the input files provided for this assignment (for reference); may also include additional exploratory code, other approaches investigated, misclassification rate estimation code, etc.

Submit the following on Gradescope:

- A `.pdf` writeup, with at most 2 pages, describing the approaches you investigated and the ones that you finally used to reduce the dimensionality and produce the cell type predictions, any misclassification rate estimation approaches and results, any important observations about characteristics of the dataset, etc. The writeup should be in NIPS format.[1]

5 Tools

You may use any programming language(s) and libraries for this assignment. It should be possible to process this dataset with a laptop with 8GB of RAM (or less when using more RAM-efficient approaches or more swap).

6 Grading

Autolab checks: 0 pts (but required) Scores are assigned on Autolab simply to differentiate between things that pass the checks and things that don't; these autograder points don't actually correspond to this assignment's grade. You should at least receive a full score on Autolab for your submission to be considered complete; if any Autolab checks fail, then there is a high chance that the actual grading code will fail to read or will misread your files and assign a much lower score than it would to correctly formatted versions (and it may assign a score of 0).

- The format check makes sure that the submitted `.tsv` file indeed appears to be a `.tsv`-formatted file with a single column named "Cell Type", contains only labels that appear in the training set, and has the right number of rows.
- The autograder also checks to see if it looks like you remembered to include your code in the submission.

For partial scores, please consult the autograder feedback (by clicking the score for any part); some output from the autograder may specify the exact issue.

[1] https://nips.cc/Conferences/2017/PaperInformation/StyleFiles

Test accuracy: 50 pts A correct-classification rate above 0.35 on the test set will receive at least full credit in this category; other accuracy levels may as well. Other accuracy levels will be assigned scores based on the distribution among students. Beware: you may obtain optimistic estimates of performance on a validation set selected from the training data. Only your last submission's test accuracy is guaranteed to be considered for scoring.

Ideas: 25 pts Well-justified and novel ideas will generally receive higher scores.

Writeup: 25 pts A good writeup will describe methods and observations clearly and concisely, while allowing a reader to understand and reproduce important procedures and details.