Written Report – 6.419x Module 2

Name: ptoche

Long calculations and code snippets are in the appendix at the end of this report.

Problem 2.2: Larger unlabeled subset

Now we will work with the larger, unlabeled subset in `p2_unsupervised`*. This dataset is has not been processed, so you should process using the same log transform as in Problem 1.*

Part 1: Visualization

A scientist tells you that cells in the brain are either excitatory neurons, inhibitory neurons, or non-neuronal cells. Cells from each of these three groups serve different functions within the brain. Within each of these three types, there are numerous distinct sub-types that a cell can be, and sub-types of the same larger class can serve similar functions. Your goal is to produce visualizations which show how the scientist's knowledge reflects in the data.

As in Problem 1, we recommend using PCA before running T-SNE or clustering algorithms, for quality and computational reasons.

1.

(3 points) Provide at least one visualization which clearly shows the existence of the three main brain cell types described by the scientist, and explain how it shows this. Your visualization should support the idea that cells from a different group (for example, excitatory vs inhibitory) can differ greatly.

2.

(4 points) Provide at least one visualization which supports the claim that within each of the three types, there are numerous possible sub-types for a cell. In your visualization, highlight which of the three main types these sub-types belong to. Again, explain how your visualization supports the claim.

Part 2: Unsupervised Feature Selection

Now we attempt to find informative genes which can help us differentiate between cells, using only unlabeled data. A genomics researcher would use specialized, domain-specific tools to select these genes. We will instead take a general approach using logistic regression in conjunction with clustering. Briefly speaking, we will use the `p2_unsupervised` *dataset to cluster the data. Treating those cluster labels as ground truth, we will fit a logistic regression model and use its coefficients to select features. Finally, to evaluate the quality of these features, we will fit another logistic regression model on the training set in* `p2_evaluation`*, and run it on the test set in the same folder.*

1.

(4 points) Using your clustering method(s) of choice, find a suitable clustering for the cells. Briefly explain how you chose the number of clusters by appropriate visualizations and/or numerical findings.

2.

(6 points) We will now treat your cluster assignments as labels for supervised learning. Fit a logistic regression model to the original data (not principal components), with your clustering as the target labels. Since the data is high-dimensional, make sure to regularize your model using your choice of l_1, l_2, or elasticnet, and separate the data into training and validation or use cross-validation to select your model. Report your choice of regularization parameter and validation performance.

Multi-class logistic regression: *When the underlying data has more than two classes involved, we can adapt Logistic Regression which is usually used for binary classification by* ***one-versus-rest*** *approach. In particular, if we have K classes, we train K separate binary classification models using logistic regression. Each classifier f_k for $k \in \{1, \ldots, K\}$ is trained to determine the probability of a data point belonging to class k. To predict the class for a new point x, we run all K classifiers on x and choose the class with the highest probability, i.e.,*

$$\hat{y} = argmax_{k \in \{1,\ldots,K\}} f_k(x)$$

Python tip: You may use liblinear solver in LogisticRegression or LogisticRegressionCV for one-versus-rest logistic regression.

Note: *Recall that the* `p2_unsupervised_reduced` *and* `p2_evaluation_reduced` *folders contain datasets with a reduced number of genes, in case you are unable to run some of the procedures on the larger versions. In particular, a full logistic regression could take* 1 *or* 2 *GB of memory to run.*

3.

(9 points) Select the features with the top 100 *corresponding coefficient values (since this is a multi-class model, you can rank the coefficients using the maximum absolute value over classes, or the sum of absolute values). Take the evaluation training data in* `p2_evaluation` *and use a subset of the genes consisting of the features you selected. Train a logistic regression classifier on this training data, and evaluate its performance on the evaluation test data. Report your score. (Don't forget to take the log transform $\log_2(x+1)$ before training and testing.)*

Compare the obtained score with two baselines: random features (take a random selection of 100 *genes), and high-variance features (take the* 100 *genes with highest variance). Finally, compare the variances of the features you selected with the highest variance features by plotting a histogram of the variances of features selected by both methods.*

Note: *The histogram should show the distribution of the variances of features selected by both methods. You could show the comparison by overlaying both histograms in the same plot.*

Problem 2.3: Influence of Hyper-Parameters

Introduction

The hyper-parameter choices used in data analysis techniques can have a large impact on the inferences made. As you may have encountered, finding the best choice of parameter such as perplexity in T-SNE or the number of clusters can be an ambiguous problem. We will now investigate the sensitivity of your results to changes in these hyper-parameters, with the goal of understanding how your conclusions may vary depending on these choices.

1.

(3 points) When we created the T-SNE plot in Problem 1, we ran T-SNE on the top 50 PCs of the data. But we could have easily chosen a different number of PCs to represent the data. Run T-SNE using 10, 50, 100, 250, and 500 PCs, and plot the resulting visualization for each. What do you observe as you increase the number of PCs used?

2.

(13 points) Pick three hyper-parameters below and analyze how changing the hyper-parameters affect the conclusions that can be drawn from the data. Please choose at least one hyper-parameter from each of the two categories (visualization and clustering/feature selection). At minimum, evaluate the hyper-parameters individually, but you may also evaluate how joint changes in the hyper-parameters affect the results. You may use any of the datasets we have given you in this project. For visualization hyper-parameters, you may find it productive to augment your analysis with experiments on synthetic data, though we request that you use real data in at least one demonstration.

Some possible choices of hyper-parameters are:

Category A (visualization):

- *T-SNE perplexity.*
- *T-SNE learning rate.*
- *T-SNE early exaggeration.*
- *T-SNE initialization.*
- *T-SNE number of iterations/convergence tolerance.*

Category B (clustering/feature selection):

- *Effect of number of PCs chosen on clustering.*
- *Type of clustering criterion used in hierarchical clustering (single linkage vs ward, for example).*
- *Number of clusters chosen for use in unsupervised feature selection and how it affects the quality of the chosen features.*

- *Magnitude of regularization and its relation to your feature selection (for example, does under or over-regularizing the model lead to bad features being selected?).*
- *Type of regularization (L^1, L^2, elastic net) in the logistic regression step and how the resulting features selected differ.*