Cancer Immunotherapy Submission Writeup

Team LIDS

1 Overview

We observe that given gene expression data, it is easy to classify its state with high precision. Therefore the main idea is to generate enough gene expression vectors, and compute the proportion vectors by putting them into a classifier.

We make the assumption that for any gene perturbation, the gene expression vector is distributed as a high-dimensional Gaussian distribution. This seems to be a valid assumption because we observe that, using true mean and covariance, this method gives a small ℓ_1 error for the prediction of proportion vectors.

So the main goal is the estimate the parameters (mean and covariance) of the Gaussian distribution. We make estimation by translation from corresponding genes in human gene expression data.

2 Dataset and Preprocessing

Apart from the given dataset of the mouse gene expressions, an external dataset we used is the human cell dataset, at

`https://plus.figshare.com/articles/dataset/_Mapping_information-rich_genotype-phenotype_landscapes_with_genome-scale_Perturb-seq_Replogle_et_al_2022_processed_Perturb-seq_datasets/20029387`.

For both the given (mouse) dataset and human gene dataset, we converted both datasets into PCA form of 50 components, fitting only the unperturbed data for each of them.

In addition, for quality control, we only consider training genes that have significant representation in the human dataset (i.e. gene knockouts that occur at least 50 times).

3 Method

The main idea is to find a mapping $f : (\mu_h, \Sigma_h) \to (\mu_m, \Sigma_m)$, where the μ_h, Σ_h are the mean and covariance of a human gene perturbation, and μ_m, Σ_m are those for mouse cells corresponding to the same gene.

1. Compute (μ_m, Σ_m) on the mouse dataset for each of the gene knockouts.
2. Using K-nearest classifier (with $K = 15$), fit the (X, y) pair (denoting the post-PCA gene expression, and the state of a given gene, respectively) into a classifier C.
3. Determine f by fitting the (human, mouse) data-pair (using only the training gene knockouts that fulfill our quality control criterion) into a linear map f. Due to lack of enough data, the output Σ_m of f is fixed to be the covariance for unperturbed cells.
4. Using this fitted linear map f, output $f(\mu_h, \Sigma_h)$ on the validation and test knockouts.
5. Based on the predicted (μ_m, Σ_m)-pair for each validation and test knockout, we generate 1000 gene expressions for each of the knockouts, and fit each of these expressions into the KNN classifier C that we trained before.