Introduction to Computational and Systems Genomics – challenge

Goals:

1. Build a classifier that, given features from a particular patient's genome, will return the type of cancer that patient has for Squamous Cell Lung Carcinoma (LUSC) and Head-Neck Squamous Cell Carcinoma (HNSC) based on 100 genes.

2. Build a classifier that, given features from a particular patient's genome **and in addition to their methylation levels,** will return the type of cancer that patient has for the same types of cancer in the previous section based on data from those 100 genes.

Mission description

The following files are attached:

First file (csv.data_muts_train – (contains mutations for 80% of patients (805 patients))

Second file (csv.data_muts_test – (contains mutations for 20% of patients (201 patients))

The above files contain the following data:

1. Patient index (id_case)
2. Name of the gene in which the mutation is present (name_Gene)
3. The chromosome number where the mutation and gene are located (Chromosome)
4. Start position of the mutation (Position_Start)
5. Mutation end position (Position_End)
6. In which strand is the mutation located (Strand_Mut)
7. Nature of the mutation (Classification_Variant)
8. The sequence in the original gene at the mutation location (Allele_Reference)
9. The sequence of the mutation in allele 1 (1Allele_Seq_Tumor)
10. The sequence of the mutation in allele 2 (2Allele_Seq_Tumor)

• Note that a particular patient may have several mutations, i.e., several rows of data. • Also, for the TRAIN set there is also a column called Label with a label (=label) equal to -1 for patients LUSC and 2- for HNSC patients

Third file (csv.feats_train) – contains features and labels for the TRAIN set of patients

Fourth file (csv.feats_test – contains features for the TEST set of patients

These files contain the features of the amount of mutations for each of the 100 selected genes.

Please note that this is just an example of possible features that can be created. You should think of additional features and create them. them

Fifth file (csv.genes_100 – contains the 100 selected human genes with the gene name, its strand, its coding sequence (for the longest transcript of that gene), and information about that protein/transcript (including the locations of the coding sequence in the genome).

Sixth file (csv.data_meth_train – contains methylation data for the same patients from the first file

Python code that, given the location of a mutation on a particular chromosome, extracts X nucleotides before the mutation and Y nucleotides after the mutation (sequences_extract)

The chromosome sequence corresponding to the gene (note that in order to use the attached Python code, the chromosome number is in the attached files) must be downloaded from the NCBI website as taught in class (https://www.ncbi.nlm.nih.gov/genome/?term=human)

Seventh file (csv.data_train_meth) – (contains methylation data for the patients in the TRAIN set. Note that you will not be able to see all of its rows if you look at it in Excel (but when you read it in Python it will be read in its entirety).

The eighth file (csv.data_test_meth) contains methylation data for the patients in the TEST set.

The files with methylation data contain:

1. Patient index (id_case)
2. The ID of the probe that measured the methylation values (probeID)
3. Beta values measuring methylation values (val_beta)
4. The chromosome with the methylation (chrm_CpG)
5. The starting position of methylation (beg_CpG)
6. The end position of methylation (end_CpG)
7. The strand on which the probe was (strand_probe)
8. Name of the gene in which the methylation occurs (genes_matching)

Here too, note that:

• A particular patient may have several locations where he or she had metastases, i.e., several rows of data. • Also, for the TRAIN set there is a column called Label with a label (=label) equal to -1 for patients LUSC and 2- for HNSC patients

Ninth file (pdf.npj_text_Main – (One article recommended for reading to get ideas for features (search for more))

Task 1 (classified based on mutations only):

A. Create the following features for each patient:

1. The amount of all mutations (sum over all genes without considering their nature) 2. The amount of mutations of a particular nature (Missense, Intron, etc.) for all possible types of mutation nature (sum over all genes) 3. The amount of mutations of a particular nature (Missense, Intron, etc.) for all possible types of mutation nature **for each gene out of the 100 genes.**

B. Create additional features as you see fit (as explained and demonstrated in class)

C. Create a graph where the Y-axis depicts the number of mutations and the X-axis depicts the type/location of mutation for each The data. The division of the X-axis is according to Figure b1 of the attached article.

D. Based on the features you received and produced, create a classifier (e.g. SVM) and train it on the TRAIN set as follows:

What we learned in class

E. Evaluate the performance of your classifier according to the following error on the Test Set (Known from the :)TRAIN=Known

Error =

sum(df_known_test['Label']!=df_known_test['predict_label'])/len(df_known_test)

When the classifier's prediction is in the column in -label_predict.

F. Finally, show the patients in the TEST set which type of cancer is appropriate for them using the classifier. that you built

Task 2 (classified based on mutations and methylation):

A. Create the following features for each patient:

1. Average methylation values for all probes belonging to a particular gene for each gene.

B-F – Same as in Assignment 1 (Section C is not relevant in this assignment). Be sure to also use the features from the previous assignment.

Submission Description –

1. Create a PDF document that contains: • What additional features you created and an explanation of why you chose to add them • The graph you created in section C of assignment 1 • Detailed flowcharts of your classifiers' operation. Discussion – How successful are your classifiers? How did you measure it? How can you improve your classifiers? 2. • Attach the code files for your two classifiers. Write a neat ME_READ file about how the code works (PDF format) for each. 3. Two csv files for the TEST set (one with the classification of the classifier from task 1 and one with the classification of the classifier from task 2) that will contain two columns: • Patient index (in the id_case column)

• The corresponding classification, i.e. 1 or 2 (in the label_predict column)