IITM-CS5691 : Pattern Recognition and Machine Learning Release Date: 22 Oct, 2021
New Method Writeup Due Date: 10 Nov, 2021

Roll No: ME18B030, ME18B046 Name: S.Jeeva, G.Deepak

- Dear Student, You may have tried different methods for predicting each of the clinical descriptors in the data contest. Submit a write-up of the methods chosen for the data contest in the template provided below. **You will have to add the details in your own words and submit it as a team in gradescope**.
- We will run plagiarism checks on codes/write-up, and any detected plagiarism in writing/code will be strictly penalized.

1. (2 points) [Name of the Method chosen for each descriptor, and its Paradigm: paradigm could be linear/non-linear models, etc.)]:

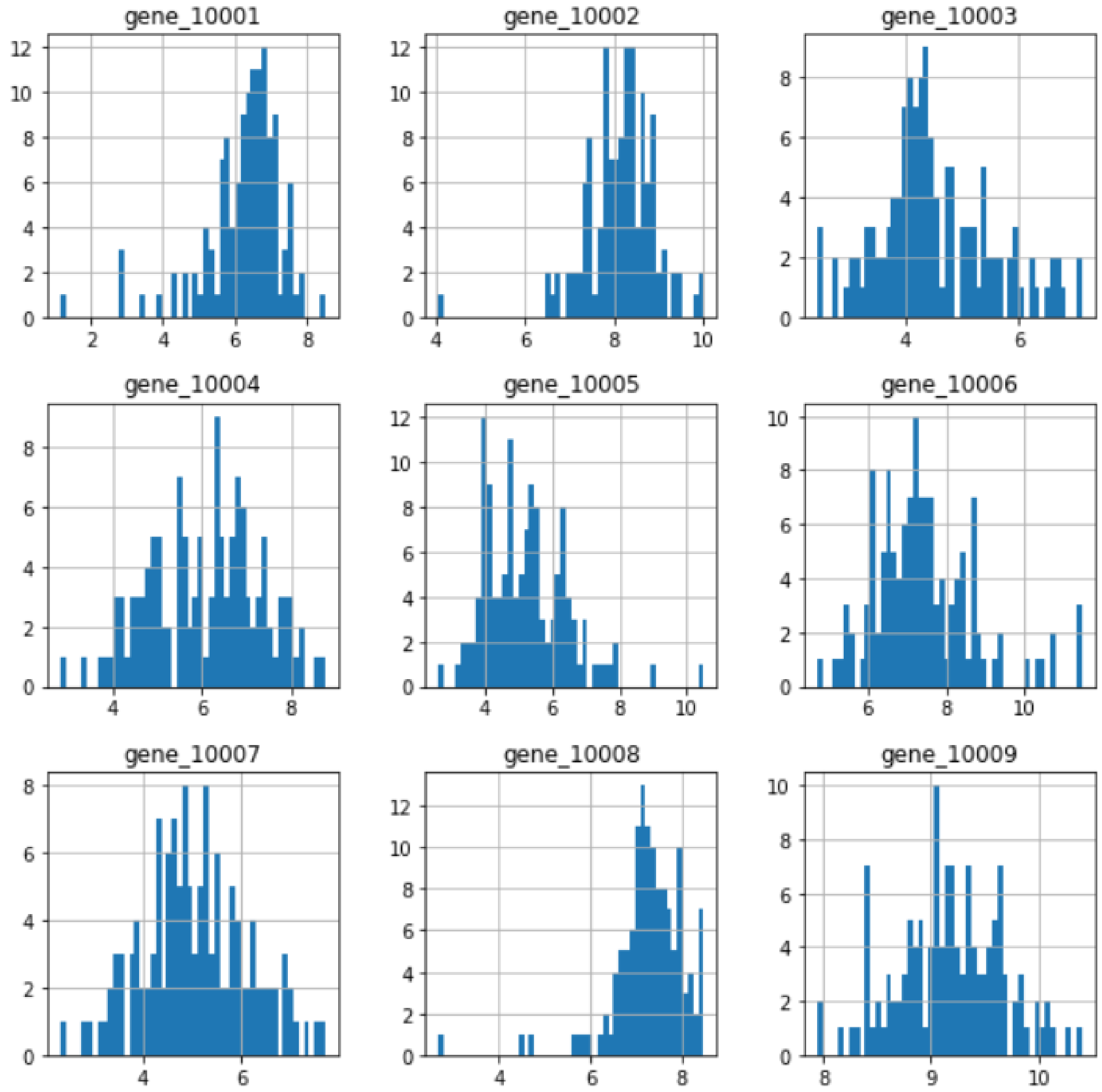
Solution:

Descriptor	**Model**	**Paradigm**
′CO: 1′	RandomForestClassifier	Non-Linear model
′CO: 2′	RandomForestClassifier	Non-Linear model
′CO: 3′	RandomForestClassifier	Non-Linear model
′CO: 4′	RandomForestClassifier	Non-Linear model
′CO: 5′	RandomForestClassifier	Non-Linear model
′CO: 6′	Support Vector Classifier - Poly. kernel	Linear Model

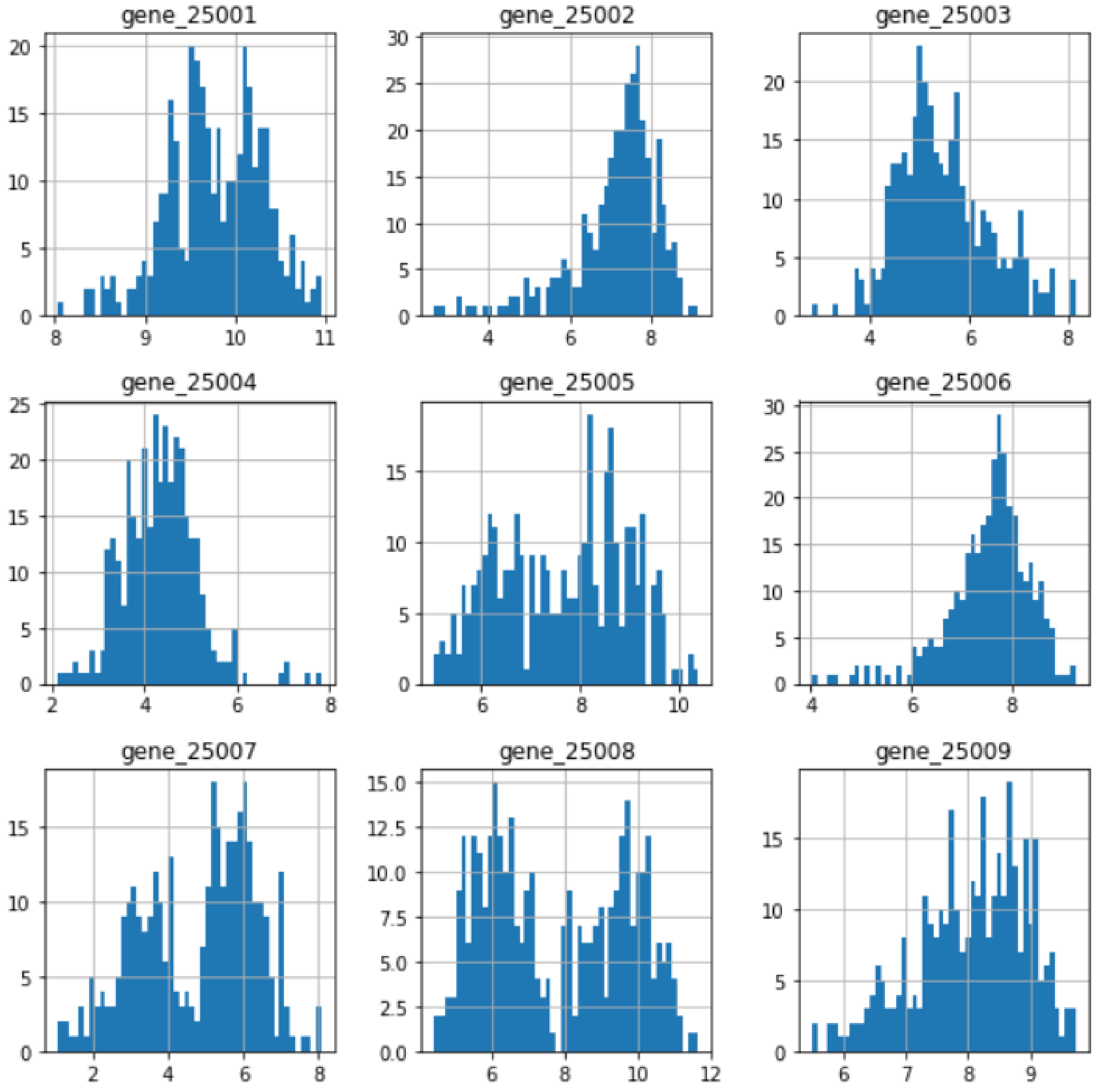
2. (5 points) [Brief description on the dataset: could show graphs illustrating data distribution or brief any additional analysis/data augmentation/data exploration performed]

Solution:

Dataset 1 contains 22283 genes as features and 2 clinical descriptors ′CO: 1′ and ′CO: 2′. It contains 130 training samples and 100 test samples. While visualizing the training set we can observe that in general, genes approximately follow a normal distribution. We used $StandardScaler$ to normalize the data where we fitted the scaler on Training dataset and further transformed both the training and the test datasets.



On the other hand, **Dataset 2** contains 54675 genes as features and 4 clinical descriptors 'CO: 3','CO: 4','CO: 5' and 'CO: 6'. It contains 340 training samples and 214 test samples. While the genes in dataset 1 approximately followed a normal distribution, several genes in dataset 2 had multi-modal distributions. Both the datasets are complete, without any missing or 'NAN' values, and hence we need not do any data cleaning to compensate for the missing values.

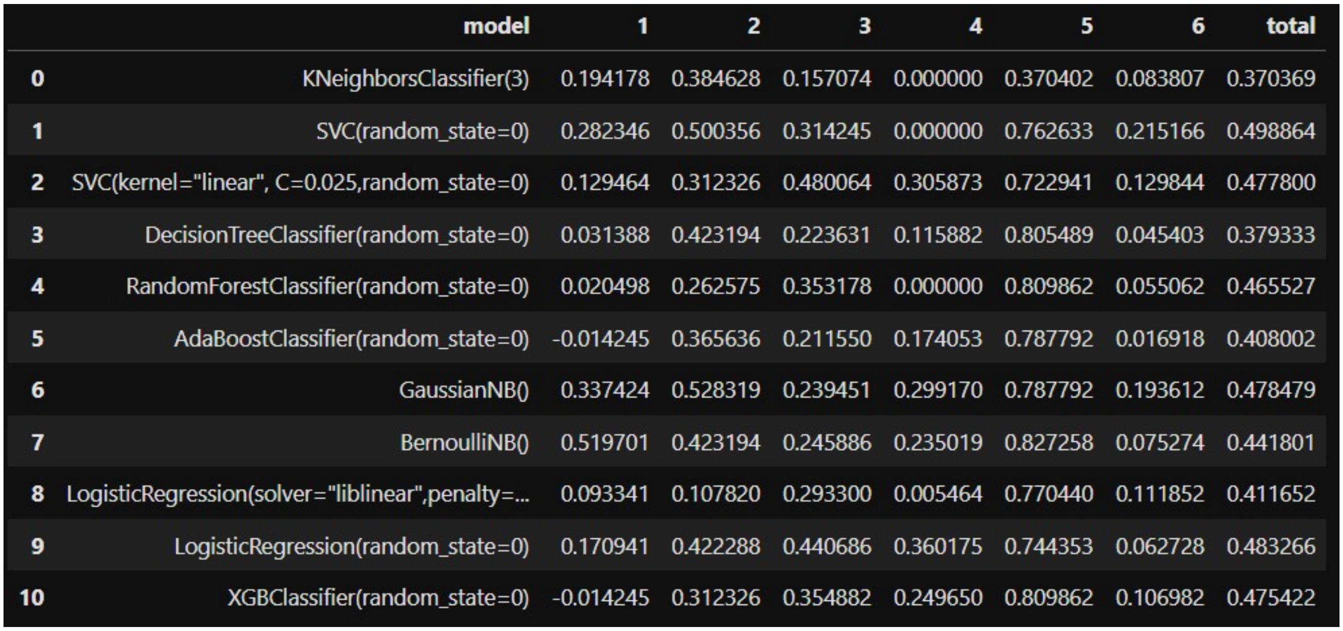


The datasets are imbalanced with value counts as follows,

	CO: 1	CO: 2	CO: 3	CO: 4	CO: 5	CO: 6
Label 0	97	77	257	289	194	140
Label 1	33	53	83	51	146	200

3. (5 points) [Brief introduction/motivation: Describe briefly the reason behind choosing a specific method for predicting each of the descriptors (could show plots or tables summarizing the scores of training different model)]

Solution:



	model	1	2	3	4	5	6	total
0	KNeighborsClassifier(3)	0.194178	0.384628	0.157074	0.000000	0.370402	0.083807	0.370369
1	SVC(random_state=0)	0.282346	0.500356	0.314245	0.000000	0.762633	0.215166	0.498864
2	SVC(kernel="linear", C=0.025,random_state=0)	0.129464	0.312326	0.480064	0.305873	0.722941	0.129844	0.477800
3	DecisionTreeClassifier(random_state=0)	0.031388	0.423194	0.223631	0.115882	0.805489	0.045403	0.379333
4	RandomForestClassifier(random_state=0)	0.020498	0.262575	0.353178	0.000000	0.809862	0.055062	0.465527
5	AdaBoostClassifier(random_state=0)	-0.014245	0.365636	0.211550	0.174053	0.787792	0.016918	0.408002
6	GaussianNB()	0.337424	0.528319	0.239451	0.299170	0.787792	0.193612	0.478479
7	BernoulliNB()	0.519701	0.423194	0.245886	0.235019	0.827258	0.075274	0.441801
8	LogisticRegression(solver="liblinear",penalty=...	0.093341	0.107820	0.293300	0.005464	0.770440	0.111852	0.411652
9	LogisticRegression(random_state=0)	0.170941	0.422288	0.440686	0.360175	0.744353	0.062728	0.483266
10	XGBClassifier(random_state=0)	-0.014245	0.312326	0.354882	0.249650	0.809862	0.106982	0.475422

Initially, we adopted the same model for all the six clinical descriptors to arrive at a baseline model. We used LogisticRegression to do predict the descriptors and the score was worse since all the six of them used the same model. Then we tried to visualize the training data better to realize that certain descriptors were very different from the others. Later we tried a plethora of classification models for each of the descriptors with KFoldValidation and the mean values of the MatthewsCorrelationCoefficient of the same can be seen from the above table. Moreover, while we tried to use RandomForestClassifier and observed that the model performed better in predicting the first 5 descriptors ′CO : 1′,′CO : 2′,′CO : 3′,′CO : 4′, and ′CO : 5′. Further, we used hyperparameter tuning inorder to get the optimal hyperparamter set for each of the first 5 descriptors. For the sixth descriptor ′CO : 6′, however, RandomForestClassifier performed bad but SupportVectorClassifier with $poly_kernel$ performed really well. Further hyperparameter tuning on this model gave us the optimal regularization co-efficient C.

4. (4 points) [As the data comes from a clinical setting, model interpretation is also an important task along with prediction. Describe briefly your preferred choice of methods if you are asked to determine the significant genes behind the regulation of the endpoints or restrict the number of features/genes.]:

Solution:

Because of the data's nature being from a clinical setting, there are thousands of features/attributes responsible for classifying the clinical descriptors. So, in this case it would be better to determine the significant genes by obtaining the features that are more important to the clinical descriptors, through the `feature_importances_` method of the `RandomForestClassifier` that we train. Basically, the features that are able to split the dataset better would naturally have more importance to the dataset. This is what we try to exploit inorder to better classify, using the models that we have chosen.

5. (2 points) [Share your thoughts on each of the endpoints: whether easy/difficult to predict]:

Solution:

Descriptor	Difficulty
'CO: 1'	Easy
'CO: 2'	Easy
'CO: 3'	Easy
'CO: 4'	Easy
'CO: 5'	Easy
'CO: 6'	Difficult

The descriptors from the first dataset gave a consistent performance with more than 75% accuracy. The same was observed for CO: 3 and CO: 4 descriptors however the predictions were skewed. On the other hand, CO: 5 performed very well with more than 90% accuracy. However, CO: 6 was no better than a random guess for various classifiers, except for Support Vector classifier, using which we were able to achieve around 65% accuracy.

6. (2 points) [Add any additional information: like challenges faced or some details that would help us to better understand the strategies that you have utilized for model development]:

Solution:

The models using `RandomForestClassifier` required extensive Hyperparameter tuning using `max_depth`, `n_estimators`, `max_features` and the `SupportVectorClassifier` required `kernel` type and `C` values (regularization parameter with L2 penalty) as the hyperparameters respectively posed a significant difficulty. Moreover, the Feature selection techniques using `SelectKBest` module from `sklearn` library was necessary to select the important features and the number of features were also treated as a hyperparameter and tuned along with the models' hyperparameters to ensure that the models generalized better. This significantly improved the time needed to tune the hyperparameters using `GridSearchCV`.