|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Machine learning | | | | |
|  |  | | |  |
| MATRIX | | | | |
|  | | for potential insecticide gene targets in the gut |  | |

# algorith Description

## Intoduction

This study aims to present a novel way of analyzing genomic data and classifying the out coming genes as favorable potential gene targets for insecticide use or poor ones .In order to perform such a task a pipeline was created which functions in the following way .

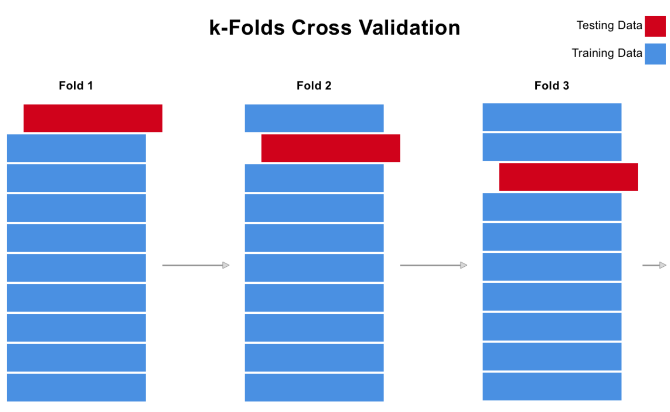
## Training and model selection

Initially we have to provide our algorithm a set of genes which will be our training set . The training set we give the algorithm will actually be the data based upon which our algorithm will ‘learn’ and recognize which characteristics or combination of characteristics of the genes classifies them as a good or a bad potential gene targets .So for example in our dataset we used the expression levels in the gut and the carcass, whether this gene was reported lethal in *Drosophila Melanogaster* or *Tribolium Castaneum* ,how many orthologues and paralogues it has, its existence in the human genome etc. We gather as much information as possible for every gene hopping that within these variables lies the characteristics of what qualifies a gene as a good or a bad potential target . It must be mentioned that the training phase of the algorithm is a very crucial one . If you teach your classifier to recognize some characteristics as good ones while they are not that will affect the final output . So the training set must and should be carefully chosen so that we get as precise as possible results . The existing scarcity of examples of good potential gene target for pesticide use in the gut didn’t make this whole task easy .

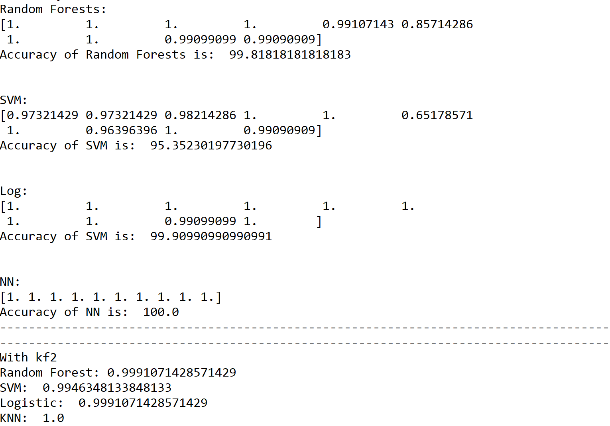
\*The training file provided here is an example ,DO NOT use this training set to train your data

Once given our algorithm the training data the process is the following . This specific algorithm aims to classify as mentioned before the genes into two separate categories .In order to achieve this therefore we must use some classification models . In machine learning classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, here for example if the tested gene could work efficiently as an insecticide , based on the training set of data containing observations whose category membership is known. With that being said our pipeline provides some of the most commonly used classifiers in RNA-seq analyses .

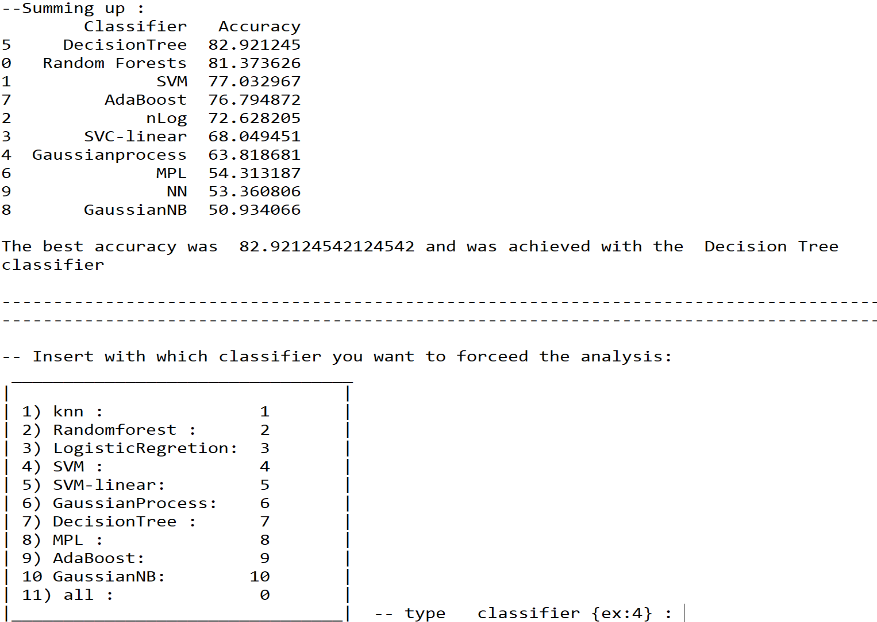
How however should can we choose between the which one is the most performing out of these algorithms ? This is the part were we perform a **Kfold cross validation** . The k-fold cross validation technique is designed to give an accurate estimate of the true error without wasting too much data. In *k*-fold cross validation the original training set is partitioned into *k*subsets (folds) of size *m/k*(for simplicity, assume that *m/k*is an integer). For each fold, the algorithm is trained on the union of the other folds and then the error of its output is estimated using the fold. Finally, the average of all these errors is the estimation of the true error.



As pictured bellow kfold cross validation actually gives us the mean estimation of one algorithms accuracy. So for example as seen below based on the training data we provided our algorithm ,when classified with knn our data was classified correctly 100 % of the times which means that all the genes in the test set where predicted correctly while with svm 99,46 % of the times .

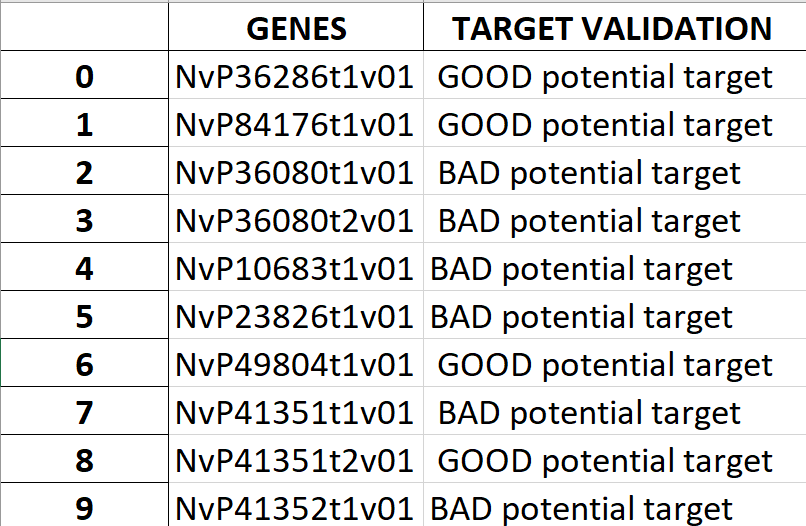


Based on the given accuracies of the existing classification models the user is called to decide with which ones will be chosen to classify the genomic data . If however someone prefers to use all the classification models and later on cross-alidade the results they were given this option is available just by typing 11 as mentioned bellow .



## Results and discussion

Based on the classifier of your choice the classification will take place . With that being said all of the genes provided as input will be classified and characterized as good or bad potential targets based on what our algorithm was taught on the training phase . As output an excel file (or four excel files if you chose to undergo the full analysis with all of the classifiers ) will be created in your working directory with the following format (one column with the name of the gene you are testing and one other with its classification ) :



All the script was performed in python 3.6 (older or new versions of python do not guarantee the function of the script ). The script is compatible both on windows and linux (ubuntu) . The classification algorithms were used by sckinit-learn .

* Your output is only as good as our training data : if the training data we provided is not representative of what consists of a good potential insecticide then our results will not be representative either
* Variable limitation: in our script we analyzed from 10 to 17 parameters (depending on the dataset ) with the potential ability to include more . No one guarantees that there is any correlation between these parameters or their combination and whether they can predict successfully the use of this gene as an insecticide .As much information however we have the most likely we are to include the required information for that prediction . So when someone intents to use this script if the option of more vs little information arrises it would be wise to go with the first option
* Format compatibly : before running the script please do make sure that your training data has the exact same fields as the data you aim to classify . You will be unable to run the classifiers if your training data has more or less variables than your data of interest .