Ensemble learning with feature selection for Alzheimer's disease prediction

Kyu S. Cho

College of Arts and Sciences, University of Missouri-St. Louis, USA

ARTICLE INFO

Article history:
December 14 2016

Keywords:
Alzheimer's disease prediction
Ensemble learning
Feature selection

ABSTRACT

Class prediction models have been shown to have varying performances in clinical gene expression datasets. The accuracy of class prediction models differs from dataset to dataset and depends on the type of classification models. While a substantial amount of information is known about the characteristics of classification functions, little has been done to determine which characteristics of gene expression data have impact on the performance of a classifier. *G*ene expression is an important aspect of many genetic diseases in the context of genetic disorders as the disorder affects only few gene expressions. Therefore, gene-expression is important in identifying disease-gene associations. Hence this paper focuses on building a classification models that accurately predicts the Alzheimer's disease (AD) using machine learning technique called ensemble learning and feature selection algorithms to find out those few disease-gene-expressions.

1. Introduction

In order to build the powerful and consistent predictive model, it's important to select right combination of models and right combination of important features. Combining multiple predictive model usually gives better performance.

An ensemble learning is a supervised learning technique for combining multiple predictions generated by different algorithms would normally deliver superior prediction power and more stable model. It is due to the diversification or independent nature as compared to each other. [1] The key to creating a powerful ensemble is model diversity and low correlation. An ensemble with two techniques that are very similar in nature will perform poorly than a more diverse model set. [2]

Too many features would result overfitting problems, and too less features would lead to miss the important features. [3] Therefore, feature selection is an extremely crucial part of modeling. The gene expression dataset consists of many features; this sounds good for building good robust model but it is a challenge to identify highly significant feature(s) out of over 8500. In such cases, feature selection and dimensionality reduction algorithm help to identify core feature(s) those highly contribute on predicting the outcome. In this study various statistical tests are used to identify such gene-expressions such as Information Gain, Chi-Squared Test, and Mean Decrease Gini Test.

Kyu S Cho | Data Scientist | Machine Learning Engineer
M.S. Computer Science | B.S. Computer Science & Mathematics
www.qcho1010.com | linkedin.com/in/qcho1010 | qcho1010@gmail.com

2. Methodology

2.1 Feature Cleaning / Selection for removing unnecessary data

Feature selection / dimensionality reduction is widely used to improve models' accuracy scores or to boost their performance on very high-dimensional datasets. Feature selection technique must be performing in this case with following reasons: [4]

1. Inputting over 8500 features may takes too much times to train for some machine learning algorithms.
2. Simplification of models to make them easier to interpret by researchers/users.
3. Enhanced generalization by reducing overfitting (formally, reduction of variance).
4. The data may contain many features that are either redundant (strongly correlated, linearly dependent) or irrelevant.

These features can be removed without incurring much loss of information

2-1-1. Removing Highly Correlated & Linearly Dependent Features

Multicollinearity (also collinearity) is a phenomenon in which two or more predictor variables in a multiple regression model are highly correlated, meaning that one can be linearly predicted from the others with a substantial degree of accuracy. [5]

Pearson correlation is used to find linear correlation between two variables X and Y. In short it is the covariance of the two variables divided by the product of their standard deviations:

$$\rho X, Y = \frac{cov(X, Y)}{\sigma X \sigma Y}$$

The result is a value between +1 and −1 inclusive, where 1 is total positive correlation, 0 is no correlation, and −1 is total negative correlation. [6] Multi-collinearity does not reduce the predictive power or reliability of the model as a whole, at least within the 'train' data set. thus, all the features that shows more than 80% of correlation are removed. From this correlated test, the number of more than 80% correlated features is 1,536 and the number of linearly dependent features is 6,736 after removing correlated features resulting 287 features remained out of 8,561 features.

2-2-2. Information Gain

The entropy characterizes the impurity of an arbitrary collection of samples. Information Gain is the expected reduction in entropy caused by partitioning the samples according to a given feature which is the way of measuring association between inputs and outputs. [7] Thus, higher the information gain is the better.

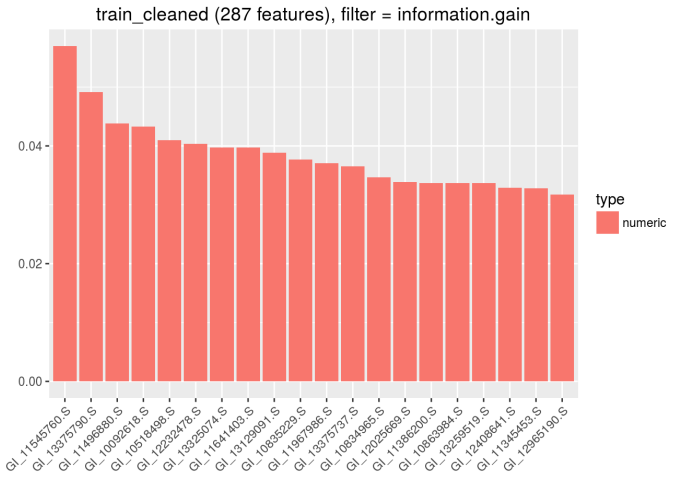

Fig. 1. Information Gain Bar Chart

From Fig. 1. features with information gain value above 0.03 are selected for the future analysis.

2-2-3. Chi-Squared Test

Pearson's chi-squared test ($\chi 2$) is a statistical test applied to sets of categorical data to evaluate how likely it is that any observed difference between the sets arose by chance and to test the independence of two events,

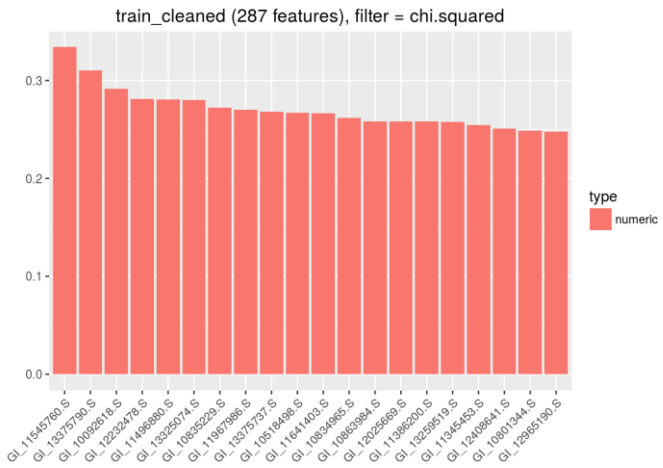

Fig. 2. Chi Squared Bar Chart

in another words, it is used to test whether the occurrence of a specific term and the occurrence of a specific class are independent. [8] Thus after estimating the following quantity for each term and features are ranked by the score. Higher scores indicate that the null hypothesis (H0) of independence should be rejected and thus that the occurrence of the term and class are dependent. If they are dependent, then those features are selected. From Fig. 2. features with chi squared value above 0.24 are selected for the future analysis.

2-2-4. Mean Decrease Gini

The mean decrease in Gini coefficient is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. [9] Each time a particular variable is used to split a node, the Gini coefficient for the child nodes are calculated and compared to that of the original node. The Gini coefficient is a measure of homogeneity from 0 (homogeneous) to 1 (heterogeneous).

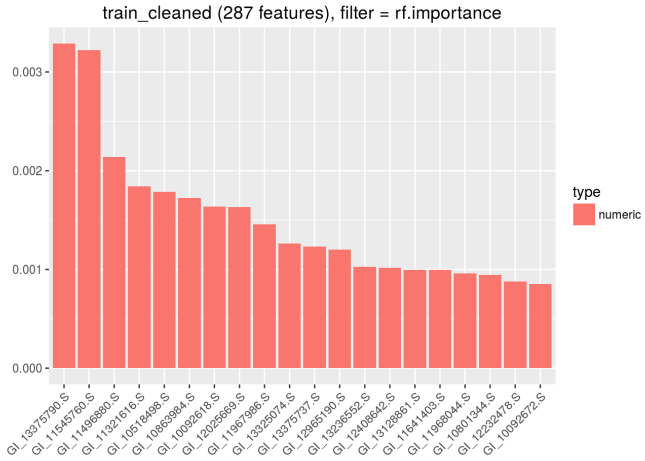

Fig. 3. Mean Decreased Gini Bar Chart

The changes in Gini are summed for each variable and normalized at the end of the calculation. Variables that result in nodes with higher purity have a higher decrease in Gini coefficient. From the Fig. 3. features with mean decreased Gini value above 0.001 are selected for the future analysis.

2-2-5. Combine Selected Features

After performing various feature cleaning / selection techniques, resulting number of important features are 30. Which is significant amount of dimensionality reduction from over 8,500 features. Those 30 features are used for building predictive model.

2-3 Building Models

In order to build the ensemble learning model, which is combination of multiple models with weights on each models, it requires various types of models. In this study, logistic regression (regression type), extreme gradient boosting (boosting type), random forest (ensemble type), recursive partitioning (tree base type), Gaussian processes (Bayesian based type), and support vector machine (coordinate based type) are used.

2.3-1. Logistic Regression (logreg): base model

Before building such powerful model, building base model is essential to compare the performance of the newly built model. Logistic regression model is one of the classification model that performs appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). It is used to describe data and to explain the relationship between one dependent binary variable and one or more metric (interval or ratio scale) independent variables. [10] Thus, this paper uses it as base model.

2.3-2. Extreme Gradient Boosting (xgb)

Boosting is an ensemble learning algorithm which combines the prediction of several base estimators in order to improve robustness over a single estimator. It combines multiple weak or average predictors to a build strong predictor. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. [11] Since, it implements machine learning algorithms under the Gradient Boosting framework it has both linear model solver and tree learning algorithms; also, it provides a parallel tree boosting that solve many data science problems in a fast and accurate way.

Pros. XGBoost is known as 'regularized boosting' technique which is used to avoid over-fitting by introducing a cost term for bringing in more features with the objective function. Hence, it tries to push the coefficients for many variables to zero and hence reduce cost term. [12] In addition, it supports various objective functions, including regression, classification and ranking.

Cons. Introducing regularization parameter to user is another deciding factor variable, faulty variables can bring the overfitting problem as well. Beside regularization parameter, there are many more parameters which needs to be controlled to optimize the model.

2-3-3. Random Forest (rf)

Random forest is an ensemble tool which takes a subset of observations and a subset of variables to build a decision trees.
It builds multiple such decision tree and amalgamate them together to get a more accurate and stable prediction. It is direct consequence of the fact that by maximum voting from a panel of independent judges, it results the final prediction better than the best judge. [13]
Each tree is planted & grown as follows:

1. If the number of cases in the training set is N, then sample of N cases is taken at random but with replacement. This sample will be the training set for growing the tree.
2. If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
3. Each tree is grown to the largest extent possible. There is no pruning.

Pros. Random forest is one of the most commonly used algorithm due to its great predictive power, Random forest model are pretty simple to build. [14]

Cons. Although, it gives high performance, users generally don't understand how they actually work. Not knowing the statistical details of the model is not a concern however not knowing how the model can be tuned well to clone the training data restricts the user to use the algorithm to its full potential. Also, it is very CPU intensive algorithm which can take up quite a long time to get the result.

2-3-4. Recursive Partitioning (rp)

Rpart is a statistical method that builds classification or regression models of a very general structure for multivariable analysis which creates a decision tree that strives to correctly classify members of the population by splitting it into sub-populations based on several dichotomous independent variables. [15] The process is termed recursive because each sub-population may in turn be split an indefinite number of times until the splitting process terminates after a particular stopping criterion is reached. Rpart methods have become popular and widely used tools for non-parametric regression and classification in many scientific fields.

Pros. It is very simple to implement and it is one of the basic tree based algorithm in machine learning.

Cons. While it is a method that is used extensively throughout the field of pattern recognition, a potential flaw exists. At each splitting node, algorithm observes immediate effects on the children to choose the one which gives us the best purity in the immediate children, instead of looking through each and every possible way to partition meaning that if it chooses a splitting criteria which is not optimal, [16] it might have better choices much later in the tree which it cannot see in the immediate children, meaning increasing the computational time. Also, this algorithm prunes to overfitting.

2-3-5. Gaussian Processes (gpr)

Gaussian Processes is a non-parametric classification method which is based on a Bayesian methodology. It assumes some prior distribution on the underlying probability densities that guarantees some smoothness properties. The final classification is then determined as the one that provides a good fit for the observed data, while at the same time guaranteeing smoothness. This is achieved by taking the smoothness prior into account, while factoring in the observed classification of the training data.

Pros. One of the few machine learning algorithms that is based on a Bayesian methodology. It is a very effective and fast classifier, also in unsupervised learning.

Cons. The performance of the method varies significantly depends on the data input.

2.3-6. Support Vector Machine (svm)

Support Vector Machine is a supervised machine learning algorithm which can be used for both classification or regression challenges. Support Vector (frontier) is simply the co-ordinates of individual observation in the hyper-plane which best segregates or differentiates the two classes. [17]

Pros. It works really well with clear margin of separation. It is effective in high dimensional spaces. It is effective in cases where number of dimensions is greater than the number of samples. [18] It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

Cons. Computation time is usually longer than normal machine learning algorithm. It under perform with noise data such as data with lots of highly correlated features.

2-3-7. Ensemble Learning (esmbl)

Boosting is an approach to calculate the output using several different models and then weight the result using a weighting approach. One of the most common challenge with ensemble modeling is to find optimal weights to ensemble base models. [19] There are various methods to find the optimal weight for combining all base learners. It starts by classifying original data set and giving equal weights to each observation. If classes are predicted incorrectly using the first learner, then it gives higher weight to the missed classified observation. Being an iterative process, it continues to add classifier learner until a limit is reached in the number of models or accuracy.

Pros. By combining the advantages and pitfalls of these approaches by varying the weighting formula, it results with a good predictive force for a wider range of input data, using different narrowly tuned models. It aims to decrease bias, not variance and it also suitable for low variance high bias models.

Cons. It also tends to over-fit the training data as well.

3. Results analysis

3-1. Data preprocessing

Data preprocessing is required to properly feed data to each machine learning algorithms. In addition, the performance of the machine learning algorithm is heavily dependent on the quality of the data. [20] The format of gene expression dataset is not the best format to be fitted into the algorithms, thus data transformation has been performed. This gene expression dataset is formatted as row (gene expression) x column (individual); however, it needs to formatted as row (individual) x column (gene expression).

3-2. Performance assessment

Learning the parameters of a prediction function and testing it on the same data is a methodological mistake: a model that would just repeat the labels of the samples that it has just seen would have a perfect score but would fail to predict anything useful on yet-unseen data. This situation is called overfitting. [21] In this study uses cross-validation (CV) to avoid the overfitting problem and uses accuracy and area under the curve (AUC) for the performance measurement.

3-2-1. Accuracy

Each prediction has two values, one is the probability of having a disease which is '1' another value is the probability of not having a disease which is '0' The accuracy is based on 0.5 threshold, meaning if the value has 0.5 than it is '1' otherwise '0'. 0.5 threshold assuming that the cost of false positive and false negative is equal.

3-2-2. Area Under the Curve (AUC)

AUC is a measure of the accuracy based on ROC curve. An ROC curve shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity). The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.

3-2-3. Cross Validation (CV)

A test set should still be held out for final evaluation, but the validation set is no longer needed when doing CV. In the basic approach, called k-fold CV, the training set is split into k smaller sets and the following procedure is followed for each of the k "folds": [21]

1. A model is trained using k=5 of the folds as training data;
2. The resulting model is validated on the remaining part of the data (i.e., it is used as a test set to compute a performance measure such as accuracy).
3. The performance measure reported by k-fold cross-validation is then the average of the values computed in the loop.

3-3 Parameter setting

The study uses following hyper parameters for each models,

1. Extreme Gradient Boosting (xgb): eta=0.196; gamma=0.0507; max_depth=1; lambda=0.938; alpha=0.75; subsample=0.929; colsample_bytree=0.195; min_child_weight=7.42;
2. Random Forest (rf): ntree=691; mtry=2; sampsize=1; nodesize=4; maxnodes=26
3. *Recursive Partitioning (rp)*: minsplit=14; minbucket=93; cp=0.104; maxcompete=10; maxdepth=24
4. *Gaussian Processes (gpr):* sigma=0.331
5. *Support Vector Machine (svm):* C=0.895; nu=0.215; epsilon=-0.0403; sigma=0.069

3-3 Features Used

GI_10092618.S,	GI_10518498.S,	GI_10800415.S,	GI_10801344.S,
GI_10834965.S,	GI_10835229.S,	GI_10863984.S,	GI_11141898.S
GI_11345453.S,	GI_11386200.S,	GI_11496880.S,	GI_11545760.S,
GI_11612658.S,	GI_11641403.S,	GI_11967986.S,	GI_12025669.S,
GI_12232414.S,	GI_12232478.S,	GI_12408641.S,	GI_12408642.S,
GI_12965190.S,	GI_13129091.S,	GI_13162281.S,	GI_13259519.S,
GI_13259542.A,	GI_13325074.S,	GI_13375737.S,	GI_13375790.S,
GI_11321616.S,	GI_13236552.S		

3-4. Performance Results

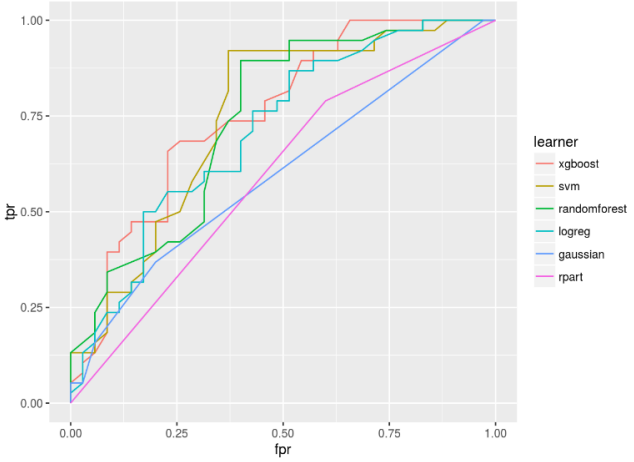

Fig. 4. AUC graph

	rp	logreg	gpr	xgb	rf	svm	esmbl
AUC	0.595	0.714	0.710	0.754	0.740	0.748	0.754
Accuracy	0.603	0.616	0.644	0.689	0.736	0.764	0.781
Overall	1.197	1.331	1.354	1.443	1.477	1.513	1.531

Table 1

	rp	logreg	gpr	xgb	rf	svm
Weight	0.05	0.05	0.40	0.00	0.00	0.50

Table 2

As Fig.4. and Table 1 are shown, at certain threshold, 'svm' and 'random forest' perform extremely good but overall 'xgb' performs the best for the AUC results. However, the accuracy of 'svm' performs the best as a single model with accuracy of 0.736. The difference AUC value between 'xgb' and 'svm' does not show the significance; whereas the accuracy of 'svm' overwhelming the accuracy of 'xgb'. Table 2 shows the weight distribution on each predictions to build the 'esmbl' model which is the combination of multiple models. It is interesting to see some bad model such as 'rp' and 'logreg' is contributing more weight than better models such as 'xgb' and 'rf', again ensemble model becomes the best with diverse and low correlated models. 'esmbl' model performs the best out of all models with accuracy of 0.781 and AUC value of 0.754. By comparison between the base model (logreg), and the ensemble model, the both accuracy (over 14%) and AUC (4%) are improved.

3-5. Feature Exploration

As Fig. 1-4 are shown, feature 'GI_11545760.S' and 'GI_13375790.S' took top 2 in every chart. The mean of feature 'GI_11545760' with disease is 0.1126 and without disease is -0.1148. and the mean of feature 'GI_11375790.S" with disease is 0.1862 and without disease is -0.1743. Although, those two variables are taking top 2 on every test, those do not show the significant level of 0.05 > p, as Fig. 5-6 are shown. Thus, combination of multiple features does contribute more predictive power rather than single important variables.

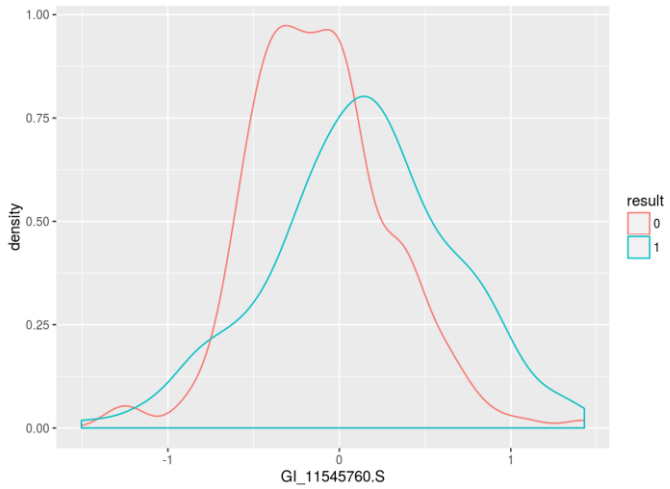

Fig. 5. 'GI_11545760.S' Density Graph

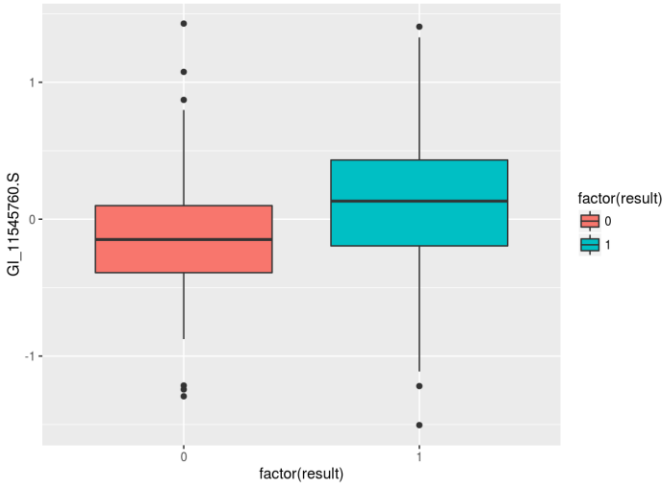

Fig. 6. 'GI_1154760.S' Box Chart

4. Conclusions and future research

An ensemble model, which is the combination of multiple models to increase the predictive force, show do outperform the best single model, which in this case support vector machine, with proper weight distribution on each model's prediction.

Various feature selection algorithms incredibly reduce the dimensionality of the data from over 8,500 features to 30 and increases the accuracy of the model from 63% to 78%, which also significantly reduce the computational intensity by selecting only the core combination of the features.

Although, the study was able to improve the accuracy of more than 14% from the base model, following ways can bring even more improvement,

1. Performing better missing value imputation
2. Using more machine learning algorithm to build the ensemble learning
3. Finding better hyper-parameters for each models
4. Adding new features by performing better feature engineering
5. Finding right number and the combination of important features

Acknowledgements

The author is very grateful for the constructive comments of Professor Sharlee Climer. This research is partially supported by College of Arts and Sciences, University of Missouri-St. Louis, USA

References

[1] Dietterich, T. G. (2000*). Ensemble Methods in Machine Learning*. In MCS '00: Proceedings of the First International Workshop on Multiple Classifier Systems (pp. 1–15).

[2] Webb, G. I., & Zheng, Z. (2004). *Multistrategy ensemble learning: Reducing error by combining ensemble learning techniques*. IEEE Transactions on Knowledge and Data Engineering, 16(8), 980–991.

[3] Kim, K. I., & Simon, R. (2014). *Overfitting, generalization, and MSE in class probability estimation with high-dimensional data*. Biometrical Journal, 56(2), 256–269.

[4] Saeys, Y., Inza, I., & Larrañaga, P. (2007). *A review of feature selection techniques in bioinformatics*. Bioinformatics.

[5] Graham, M. H. (2003). *Confronting multicollinearity in ecological multiple regression.* Ecology, 84(11), 2809–2815.

[6] Good, P. (2009). *Robustness of Pearson correlation*. InterStat, 15.

[7] Yu, Y., & Lee, T. S. (2005). *Adaptive contrast gain control and information maximization. Neurocomputing*, 65–66(SPEC. ISS.), 111–116.

[8] Hoey, J. (2012). *The Two-Way Likelihood Ratio (G) Test and Comparison to Two-Way Chi Squared Test*. arXiv:1206.4881v2, 2, 1–6.

[9] Gini, T., Gini, C., & Gini, I. (2008). *Gini coefficient.* Stat, 1–5.

[10] Gortmaker, S. L., Hosmer, D. W., & Lemeshow, S. (1994). *Applied Logistic Regression*. Contemporary

[11] Mason. (2000). *Boosting algorithms as gradient descent*. Nips, 3(1), 1–11.

[12] Chen, T., & Guestrin, C. (2016*). XGBoost : Reliable Large-scale Tree Boosting System*. arXiv, 1–6.

[13] Qi, Y. (2012). *Random forest for bioinformatics. In Ensemble Machine Learning: Methods and ApplicatiOns* (pp. 307–323).

[14] Ren, S., Cao, X., Wei, Y., & Sun, J. (2015). *Global refinement of random forest. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Vol. 7-12-NaN-2015, pp. 723–730).

[15] Zeileis, A., Hothorn, T., & Hornik, K. (2008*). Model-Based Recursive Partitioning. Journal of Computational and Graphical Statistics*, 17(2), 492–514.

[16] Zhu, F., & Agrafiotis, D. K. (2007). *Recursive distance partitioning algorithm for common pharmacophore identification. Journal of Chemical Information and Modeling*, 47(4), 1619–1625. h

[17] Jändel, M. (2010). *A neural support vector machine. Neural Networks*, 23(5), 607–613.

[18] Bottou, L., & Lin, C. J. (2007). *Support Vector Machine Solvers. Science*, 3(1)7477, 1–27.

[19] Oh, S., Lee, M. S., & Zhang, B. T. (2011). *Ensemble learning with active example selection for imbalanced biomedical data classification. IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(2), 316–325. h

[20] Yin, S., Wei, Z., Gao, H., & Peng, K. (2012). *Data-driven quality related prediction and monitoring*. In IECON Proceedings (Industrial Electronics Conference) (pp. 3874–3879).

[21] Horne, J. S., & Garton, E. O. (2006*). Likelihood cross-validation versus least squares cross-validation for choosing the smoothing parameter in kernel home-range analysis.* Journal of Wildlife Management, 70(3), 641–648.