Ensemble based Clustering of Plasmodium falciparum genes

[1, 2]Itunuoluwa Isewon, [1, 2]Jelili Oyelade, [1, 2]Ezekiel Adebiyi
[1]Department of Computer and Information Sciences and [2]Covenant University Bioinformatics Research (CUBRe)
Covenant University
Ota, Ogun State, Nigeria.
itunu.isewon@covenantuniversity.edu.ng

Benedict Brors
Division of Applied Bioinformatics
German Cancer Research Center (DKFZ)
Heidelberg, Germany.

***Abstract*—Ensemble learning is a recent and extended approach to the unsupervised data mining technique called clustering which is used for finding natural groupings that exist in a dataset. Here, we applied an ensemble based clustering algorithm called Random Forests with Partition around Medoids (PAM) to multiple time series gene expression data of Plasmodium falciparum. The Random Forest algorithm is most common ensemble learning approach that uses decision trees. Random Forest consists of large number of classification trees (ranging from hundreds to thousands) built from several bootstrap sampling of the dataset. We also applied the following internal cluster validity measures; Silhouette Width index, Connectivity Index and the Dunn Index to select the optimal number of final clusters. Our results show that ensemble based clustering is indeed a good alternative for cluster analysis with the promise of an improved performance over traditional clustering algorithms.**

Keywords-Random Forests, Plasmodium falciparum, Ensemble Clustering, Cluster Validity

I. INTRODUCTION

Clustering is an unsupervised data mining technique that is useful in finding natural groupings or inherent structure that exists in a dataset. In contrast to classification, it does not require predefined classes. This explains its ability to detect previously unknown relationships among unlabeled data objects. It also has the ability to describe unknown properties of these data objects (i.e. detection of natural data types) and discover unusual data objects (i.e. outlier detection). The goal of clustering is to group data points (i.e. objects) with similar numerical values together into disjoint or overlapping groups. This grouping is done such that members in the same group/cluster are more identical to each other than objects in another cluster. A cluster can therefore be referred to as a set of objects having high similarity between them as well as a high dis-similarity to other objects in other clusters. Clustering can either be hard or soft. In hard clustering, objects belong to one and only one cluster, while in soft clustering; an object belongs to a cluster to a certain degree (i.e. degree of membership). Soft clustering is also referred to as fuzzy clustering and is concerned with the probability of an object belonging to a cluster. An example of soft clustering is the Fuzzy C-Means (FCM) algorithm [1]. Clustering has a wide range of applications in the following domains; pattern recognition, image processing, data mining (information retrieval and text mining), market analysis/research, spatial data analysis, Web analysis (e.g. document classification, social network analysis), machine learning, medical diagnostics, bioinformatics and lots more.

Many clustering algorithms exist which employ different approaches/methods to group similar data objects into partitions. These include;

- Hierarchical methods: builds a hierarchy/tree of clusters called dendogram using a distance metric and a linkage criterion. Examples include agglomerative (bottom-up) and divisive (top-down) clustering algorithms (e.g. AGNES [2], DIANA [2]), BIRCH algorithm [3], CURE [4], CHAMELOEN [5], etc.
- Partitioning methods: assigns data objects into partitions and iteratively relocates them between these partitions in order to reduce a given clustering criterion. Examples include K-means algorithm [6], K-medoids algorithm [7] (e.g. PAM - Partition Around Medoids), k-mode algorithm [8], CLARA [2], CLARANS [9] and Expectation Maximisation algorithm [10] among others.
- Grid-based methods: creates clusters by mapping the data points onto a multi-resolution grid based structure and selecting contiguous groups of dense cells. The selling point of grid based clustering algorithms is that they have a significantly reduced computational complexity noticeably with large data sets because they do not require the computation of a distance metric Examples include STING [11], Wave Cluster [12], OPTIGRID [13], CLIQUE [14], e.t.c.
- Density-based methods: clusters data objects based on density i.e. using density connected points or a density function. It has the ability to handle noisy data and can discover clusters of arbitrary shapes. Examples include DENCLUE [15], DBSCAN [16] and OPTICS [17] among others.
- Model-based methods: here, clusters are created by hypothesizing a model for each cluster with the hope to find the best fit of the model to the data. Model based methods can be further divided into neural network approaches (e.g. SOM - Self

Organizing Maps [18]), probability density based approaches (e.g. COBWEB [19]) and statistical approaches (e.g. AutoClass [20] - which is a Bayesian clustering procedure based on mixture models).

All of these clustering algorithms largely employ some distance or proximity metrics in their computation to find similarity/distance between objects in the given data set. The choice of a right distance metric is a very important one as it largely affects the possibility of obtaining the correct clustering results. Several reports have carried out experiments on the use of different distance metrics in clustering algorithms and its attendant effects [21-25]. The most commonly used distance / proximity metrics include; Euclidean distance, Manhattan distance, Edit distance, Chebyshev distance, Minkowski distance, Chi-square distance, Pearson correlation coefficient, Spearman correlation coefficient, Kendall correlation coefficient and a host of others.

II. ENSEMBLE BASED CLUSTERING

Ensemble learning is a recent and extended approach to clustering involving the use of multiple learning approaches on the same data set for an improved performance. It is a two-phase approach; **the generation phase** – a collection of partitions are generated from the same dataset either with different runs of the same clustering algorithm (using different configurations) or using multiple clustering algorithms and **the consensus phase** – a consensus function is applied to the combined partitions to get a consensus clustering of the dataset. Ensemble methods generally have a stronger generalization ability compared to a single learning algorithm. Ensemble clustering has a better predictive performance in terms of stability, robustness, flexibility and accuracy when compared to any of its composing algorithms. Other strengths of ensemble learning include assigning a confidence to the decision made by the model, selecting optimal (or near optimal) features, data fusion, incremental learning, non-stationary learning and error-correcting [26].

Ensemble clustering provides more flexibility as the user is not constrained to the choice of a single clustering algorithm but can harness the strengths of multiple clustering approaches into one clustering solution and also avoid the possibility of making a poor choice of clustering algorithm. Reports have shown that ensemble based learning obtained better results when its composing models possess a significant diversity [27 – 29]. This is because when different models are used in an ensemble, a different type of error is made by each model as it learns the data by its own defined hypothesis and when the results are meaningfully combined, the overall error is drastically reduced. Ensemble based methods are also very useful in dealing with cases of large amount of data as well as insufficient data. With insufficient data, different random samplings with replacement of the data are drawn and fed into the models to be used. This process is called bootstrapping. On the other hand, large datasets are carefully partitioned into smaller chunks which are fed into the models and then later combined following some defined rules.

The following are commonly used ensemble based learning methods;

- Boosting [30] – creates an ensemble such that its composing models are built incrementally and trained by resampling the data. Successive models are built from previous ones as extra weight is awarded to miss-classified points of the previous model. The consensus prediction is gotten through a weighted vote. An extension of this method is called Adaptive Boosting (AdaBoost [31]).
- Bagging [32] – the composing models of an ensemble are built independently by different random sampling (bootstrap sample) of the data. Successive models do not rely on previous ones and the consensus prediction is gotten through a majority vote.
- Random Forest [33] – can be seen as a variant or an extension of bagging as they introduce an extra layer of randomness in bagging. They employ the use of large number of decision trees (ranging from hundreds to thousands) in order to achieve better prediction results.
- Stacking [34] – also known as stacked generalization creates ensembles in two phases. In the first phase, the bootstrapped samples of the dataset are fed into different models and their predictions are passed to the second phase where a combiner algorithm is used to merge them and produce a more refined result which ultimately corrects the miss classifications of the models used in the first phase.
- Mixtures of Experts [35] – several models are generated to learn the data and a generalized linear rule is used to combine their predictions. An Expectation Maximisation algorithm is used to train a gating network which is responsible for assigning weights to this combination. Jordan and Jacobs [36] introduced the concept of 'hierarchical mixture of experts' which is a combination of different models of mixtures of experts.

Reports presented in [37 - 43] provide more details and reviews on ensemble based learning methods in machine learning.

III. RANDOM FOREST

The random forest (RF) algorithm was developed by Leo Breiman and described in his paper published in 2001 [33]. He combined his concept of bagging with Amit and Geman's idea of random selection of features described in [44], in the construction of decision tress. Random forest is used for classification, clustering and regression by constructing a multitude of unpruned decision trees and outputting the mode prediction (classification/clustering) or the mean prediction (regression) of the composing trees. The RF algorithm involves the construction of a forest of trees by projecting the dataset onto a random subset of the data to grow each tree. Then splitting of the nodes of the trees is done by randomized optimization where the best of a randomly selected subset of predictors is chosen to split each node. This approach is different

from the usual approach for standard trees where each node is split using the best of all available variables (deterministic optimization). This counterintuitive approach out performs many other well-known classification models such as neural networks, discriminant analysis, support vector machines among others. The properties of the RF algorithm include;

1) It uses the OOB data (Out-Of-Bag) to estimate the generalization error,

2) It defines a measure for ranking the importance of the predictor variables. The different methods for variable importance measurement are further described by Breiman in [45].

3) It also defines a proximity/distance measure. Each cell of an RF proximity matrix represents the ratio of tress in which the represented objects appear in the same terminal nodes. The reasoning here is that similar objects are expected to appear in more often in the same terminal nodes than dissimilar objects.

Among the strengths of random forest are; robustness - it corrects the over-fitting problem of decision trees, simplicity and user-friendliness – only two parameters are involved in its computation (i.e. number of predictor variables and the number of trees) as well as capacity for handling high dimensionality data i.e. large p and small n datasets with limited sample sizes which are common in genomic data.

IV. RELATED WORK

Over the years, clustering has been applied largely in bioinformatics. An example is in microarray data analysis to find sets of genes with similar expression patterns to infer sharing of similar function (functional annotation), imply common regulation and predict cis-regulatory promoter sequences. Sharan *et al.,* in [46] presented a review of cluster analysis methods and their applications to analysis of gene expression data such identification of regulatory motifs and tissue classification. Li *et al.,* in [47] demonstrated that clustering can also be large biological databases. They developed a fast program to cluster highly homologous sequences stored in a protein database in order to reduce the size of the database. A number of tools, programs and applications have also been developed applying clustering to other bioinformatics related problems. An example is the HCPM developed by Grant *et al.,* in [48] for clustering protein models using hierarchical clustering. Clustering has also been used in the identification of clusters (i.e. protein complexes) in protein-protein interaction (PPI) networks. Brohee & van Helden in [49] presented a comparative analysis report of four major algorithms (i.e. MCODE - Molecular Complex Detection algorithm, RNSC - Restricted Neighborhood Search Clustering algorithm, MCL - Markov Clustering algorithm, SPC - Super Paramagnetic Clustering algorithm) used in clustering PPI networks. Several other methods asides from the traditional clustering algorithms have been developed and applied to gene expression data analysis such as Bi-clustering by Cheng & Church in [50], Multi-objective genetic algorithms for clustering and its applications in bioinformatics was discussed by Maulik *et al.,* in [51], MOGA-SVM (Multi-Objective Genetic Algorithms with Support Vector Machines) by Maulik *et al.,* in [52], Tri-clustering by Li and Tuck in [53], and a host of others.

Ensemble based methods have also been applied in bioinformatics tasks. Chen et al., in [54] presented a systematic review of the use of classification trees in bioinformatics. Asur et al., in [55] proposed an ensemble clustering framework for PPI networks. Quite a number of successful applications of random forest to genomic data abound in literature. Some include; the work of Shi *et al.,* in [56] where RF was used as a dissimilarity measure on protein expression profiles from tumor marker data to cluster renal cell carcinoma patients, the work of Lunetta et al., in [57] was one of the first to apply RF to gene wide association studies (GWAS) data to rank/prioritize SNPs (Single Nucleotide Polymorphisms) and Pang & Zhao in [58] used RF to build pathway clusters using three human breast cancer gene expression datasets.

V. MATERIALS AND METHODS

A. Gene Expression Dataset

Pre-processed gene expression time course data from Otto *et al.,* in [59] was used in this study. They applied RNA-Seq to seven time points every 8 h for 48 h, thus capturing the entire asexual intra-erythrocytic developmental cycle of P. falciparum from the ring stage to mature schizonts. The depth of sequence obtainable with highly parallel sequencing technologies make it possible to obtain high coverage of all transcribed genes. The dataset had expression for a total of 5270 transcribed *P. falciparum* genes.

B. Clustering

Time courses for *P. falciparum* transcribed genes derived from the dataset above were clustered. The clustering was done using Random forest algorithm as a proximity measure. The resulting RF proximity matrix was used as a distance measure for clustering using Partitioning Around Medoids (PAM) algorithm. All analyses were done in R version 3.1.1 [60], Bioconductor version 2.14 [61] and Rstudio version 0.98.978 [62]. The RF algorithm is implemented in the R package "randomForest" [63] and the PAM algorithm is implemented in the R package "cluster" [64].

C. Cluster Validity

Clustering is an unsupervised pattern classification method that partitions the input space into clusters. Many clustering algorithms are not able to determine the number of natural clusters that exist in the data. As a result, these algorithms require this information to be supplied—known as the k parameter. As this information is rarely previously known, the usual approach is to run the algorithm several times with a different k value for each run. Then, all the partitions are evaluated and the partition that best fits the data is selected.

In order to select the optimal clusters for the data set, clustering was done with many runs of the PAM algorithm using different k values. We then applied the following internal cluster validity measures; Silhouette Width index implemented in R package "clues" [65],

Connectivity Index and the Dunn Index both implemented in R package "clValid" [66].

VI. RESULTS AND DISCUSSION

The time courses of the 5270 transcribed *P. falciparum* genes from the dataset described above were used to get bootstrapped samples which we passed into the RF algorithm for training. In order to get a stable proximity matrix, we built a forest of 1000 trees. The (i, j) element of an RF proximity matrix represents the fraction of trees where the ith and jth objects appear in the same terminal node. The multi-dimensional scaling representation of the proximity matrix generated from the RF algorithm is presented in Fig. 1.

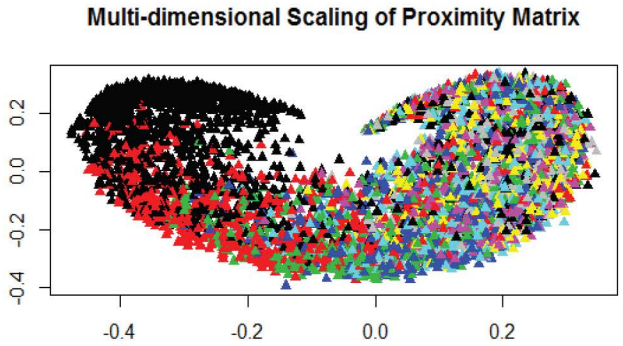

Fig. 1. A multi-dimensional scaling representation of the RF proximity matrix for 5270 P. falciparum genes

We then used this proximity matrix as a distance metric for the PAM clustering algorithm using the formula Distance = 1 – proximity. As the no of clusters (k) is unknown, we used different values of k ranging from 15 to 22 to run the PAM algorithm. As already mentioned in the methods section, three cluster validity indices were used to determine the no of optimal clusters out of the 8 different clustering from the PAM algorithm. The results are presented in TABLE I. and Fig. 2.

TABLE I. CLUSTER VALIDITY INDICES FOR DIFFERENT K VALUES FOR RF & PAM ALGORITHM

K / Cluster Validity	15	16	17	18	19	20	21	22
Silhouette	0.07 17	0.0 77 9	0.07 05	0.0 73 1	0.0 72 5	0.0 73 15	0.07 30	0.070 0
Dunn	5.12 e-6	5.0 9e- 6	5.12 e-6	2.6 3e- 6	2.6 3e- 6	2.6 3e- 6	2.63 e-6	2.63e -6
Connecti vity	417 6.01	41 23. 38	4483 .76	44 58. 94	46 47. 12	47 29. 78	4749 .98	5077. 62

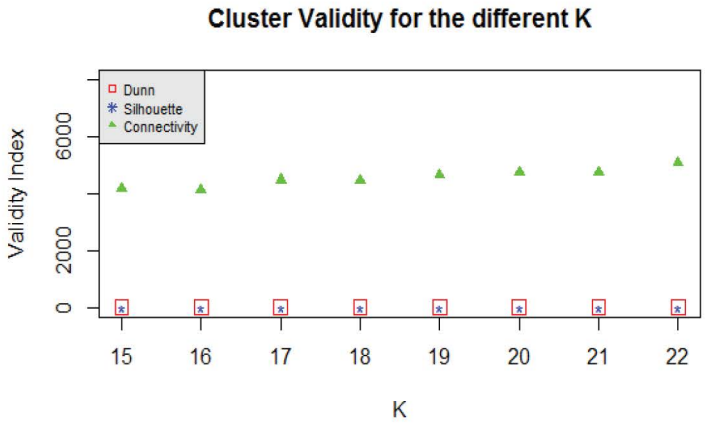

Fig. 2. Graphical representation of the validity indices

The cluster validity results show a gross difference in the range of values for the connectivity index compared the silhouette and Dunn indices. To choose the best k, connectivity index should be minimized (i.e. the lowest value), while silhouette and Dunn indices should be maximized (i.e. the highest). K = 16 was the best from connectivity and silhouette indices. We used 16 as the optimal number of clusters. The result of the PAM algorithm for K = 16 is presented in Table II. The result shows a balanced partitioning.

TABLE II. PAM CLUSTERING FOR 16 CLUSTERS

Cluster No	SIZE	Average Distance
1	279	0.52995
2	334	0.53020
3	461	0.53890
4	274	0.55035
5	395	0.55332
6	314	0.55646
7	295	0.56452
8	230	0.57233
9	298	0.57291
10	336	0.58760
11	394	0.59316
12	321	0.59503
13	337	0.60207
14	279	0.64025
15	393	0.64444
16	330	0.73132

In order to give credence to our results, we used another partitioning algorithm, the K-means algorithm on our dataset and set K =16. The result of the K-means algorithm is presented in TABLE III.

TABLE III. K-MEANS CLUSTERING FOR 16 CLUSTERS

Cluster no	Size
1	79
2	42
3	4239
4	4
5	1
6	9
7	5
8	13

9	682
10	10
11	1
12	1
13	34
14	113
15	33
16	4

The result of k-means clustering shows a skewed partitioning as some clusters are seen having very few objects (Cluster 4, 5, 11, 12 and 16) while some are seen having too many objects (Cluster 3 and 9). Table 4 presents the comparison of the partitioning between the PAM algorithm (using RF as a distance metric) and the K-means algorithm. It shows RF and PAM corrected the miss-classification of k-means. For instance Cluster 4, 5, 11, 12 and 16 that had very few objects in k-means reported a higher no of objects in RF and PAM. Also Cluster 3 that had 4239 objects which is about ¾ of the total no objects in the dataset was split into smaller partitions by RF and PAM as well as Cluster 9.

Figure 3 shows a subspace of the cluster plot colored by cluster id. It further shows the balanced nature of the partitioning by RF and PAM and the skewed nature of that of k-means as more colors are seen in the former than the latter.

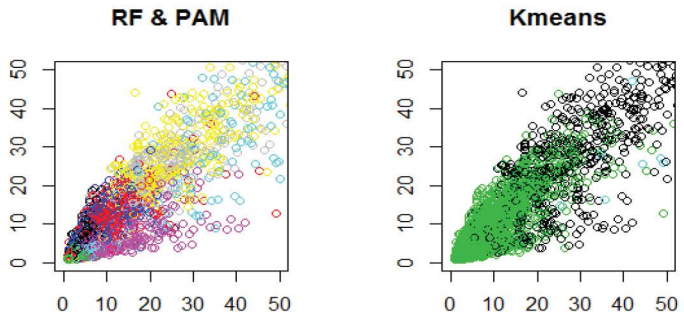

Fig. 3. Subspace of the cluster plot colored by cluster id.

VII. CONCLUSION

We have shown that ensemble based clustering is indeed a good alternative for cluster analysis with the promise of an improved performance over traditional clustering algorithms. The clusters generated by our classification ensemble method were insensitive to miss-classification from using a single clustering algorithm. Our results further shows ensemble clustering has a better predictive performance in terms of stability, robustness and accuracy when compared to any of its composing algorithms.

ACKNOWLEDGMENT

This work was supported by study grants from the German Academic Exchange Service (DAAD), Covenant University and German Cancer Research Centre (DKFZ).

REFERENCES

[1] J. C. Bezdek, "Pattern recognition with fuzzy objective function algorithms," Plenum Press, New York, 1981.

[2] L. Kaufman and P. J. Rousseeuw, "Finding Groups in Data: an Introduction to Cluster Analysis," New York: John Wiley & Sons, 1990.

[3] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," Data Mining and Knowledge Discovery 1997, Volume 1, Issue 2 , pp 141-182.

[4] S. Guha, R. Rastogi, K. Shim, "Cure: an efficient clustering algorithm for large databases, " Information Systems, Volume 26, Issue 1, March 2001, Pages 35-58.

[5] G. Karypis, Eui-Hong Han, V. Kumar, "Chameleon: hierarchical clustering using dynamic modelling," Computer, Volume 32 Issue 8, August 1999, Page 68-75.

[6] J. MacQueen, "Some methods for classification and analysis of multivariate observations," In: 5th Berkeley symposium on mathematics, statistics and probability, pp. 281–296, 1967.

[7] L. Kaufman, P. Rousseeuw, "Clustering by means of medoids," Statistical Data Analysis Based on the L1-Norm and Related Methods, 1987.

[8] Z. Huang, "Extensions to the K-means algorithm for clustering large data sets with categorical values", Data Mining Knowl. Discov., vol. 2, pp.283 -304 1998

[9] R. Ng and J. Han, "Efficient and effective clustering method for spatial data mining," In Proc. of the 20th VLDB Conference, pages 144–155, Santiago, Chile, 1994.

[10] A.P. Dempster, N.M. Laird, D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," Journal of the Royal Statistical Society, 1977, Series B 39 (1): 1–38.

[11] W. Wang, J. Yang, R. Muntz, "STING: A Statistical Information grid Approach to Spatial Data Mining," VLDB '97 Proc.23rd Int.Conf. on Very Large Data Bases, Pages 186-195

[12] G. Sheikholeslami, S. Chatterjee, and A. Zhang, "WaveCluster: A multi-resolution clustering approach for very large spatial databases," VLDB '98 Proc. of the 24th Int. Conf. on Very Large Data Bases, Pages 428-439

[13] A. Hinneburg, D.A. Keim, "Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering," Proc. 25th VLDB Conf., Edinburgh, Scotland, 1999.

[14] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," Proc. of the 1998 ACM SIGMOD international conference on Management of data, Pages 94-105

[15] A. Hinneburg, D.A. Keim, "An Efficient Approach to Clustering in Large Multimedia Databases with Noise," Proc. 4th Int. Conf. on Knowledge Discovery and Data Mining (KDD-98), AAAI Press, August 1998, pages 58—65.

[16] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases," Proc. 3rd Int. Conf. on Knowledge Discovery and Data Mining, AAAI Press, 1996.

[17] M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," Proc. of the 1999 ACM SIGMOD international conference on Management of data, Pages 49-60

[18] T. Kohonen, "Self-Organizing Maps," Springer, Berlin, Heidelberg, 1995, vol. 30.

[19] D. H. Fisher, "Knowledge Acquisition Via Incremental Conceptual Clustering", Machine Learning, September 1987, Volume 2, Issue 2, pp 139-172

[20] F. Achcar, J-M. Camadro, D. Mestivier, "AutoClass@IJM: a powerful tool for Bayesian classification of heterogeneous data in biology," Nucleic Acids Research. 2009;37(Web Server issue):W63-W67.

[21] O. A. Mohamed Jafar and R. Sivakumar, "A Comparative Study of Hard and Fuzzy Data Clustering Algorithms with Cluster Validity Indices," Proc. Int. Conf. on Emerging Research in Computing, Information, Communication and Applications, ERCICA 2013, pp 775 – 782.

[22] A. Vimal, S. R. Valluri, K. Karlapalem, "An Experiment with Distance Measures for Clustering," International Conference on Management of Data COMAD, December 2008.

[23] P. A. Jaskowiak, R. Campello and I. G. Costa, "On the selection of appropriate distances for gene expression data clustering," BMC Bioinformatics 2014, 15(Suppl 2):S2

[24] S-H. Cha, "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions," INT. J. MATHEMATICAL MODELS AND METHODS IN APPLIED SCIENCES 2007, Issue 4, Volume 1.

[25] P. Grabusts, "The choice of metrics for clustering algorithms, " Proc. 8th Int. Scientific and Practical Conference, 2011, Volume 11, pp. 70–76.

[26] R. Polikar, "Ensemble learning", Scholarpedia 2009, 4(1):2776.

[27] L. Kuncheva, and C. Whitaker, "Measures of diversity in classifier ensembles," Machine Learning, 51, pp. 181-207, 2003

[28] G. Brown, J. Wyatt, R. Harris, and X.Yao, "Diversity creation methods: a survey and categorisation," Information Fusion, 6(1), pp.5-20, 2005.

[29] J. G. Adeva, U. Cerviño, and R. Calvo, "Accuracy and Diversity in Ensembles of Text Categorisers," CLEI Journal, Vol. 8, No. 2, pp. 1 - 12, December 2005.

[30] R.E. Schapire, "The strength of weak learnability," Machine Learning 5(2) (1990) 197–227

[31] Y. Freund and R. E. Schapire, "Decision-theoretic generalization of on-line learning and an application to boosting," Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119-139, 1997.

[32] L. Breiman, "Bagging predictors," Machine Learning, vol. 24, no. 2, pp. 123-140, 1996.

[33] L. Breiman, "Random Forests," Machine Learning, vol. 45, no 1, pp. 5–32, 2001.

[34] D.H. Wolpert, "Stacked generalization," Neural Networks, vol. 5, no. 2, pp. 241–260, 1992

[35] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," Neural Computation, vol. 3, pp. 79-87, 1991.

[36] M. J. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," Neural Computation, vol. 6, no. 2, pp. 181-214, 1994.

[37] E. Bauer, R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, Boosting, and variants," Machine Learning 36(1-2) (1999) 105–139

[38] K.M. Ting, I.H. Witten, " Issues in stacked generalization," Journal of Artificial Intelligence Research 10 (1999) 271–289

[39] D. Opitz, R. Maclin, "Popular ensemble methods: An empirical study," Journal of Artificial Intelligence Research 11 (1999) 169–198

[40] A. Strehl, J. Ghosh, "Cluster ensembles - a knowledge reuse framework for combining multiple partitionings," Journal of Machine Learning Research 3 (2002) 583–617

[41] T.G. Dietterich, "Ensemble Methods in Machine Learning," Int. Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, Vol. 1857, pp. 1-15, 2000, Springer-Verlag.

[42] C. M. Bishop, "Pattern Recognition and Machine Learning," Vol. 4. No. 4. New York:Springer 2006

[43] H. B. Mitchell, "Ensemble Learning." Data Fusion: Concepts and Ideas. Springer Berlin Heidelberg, 2012. 295-321.

[44] Y. Amit, D. Geman, "Shape quantization and recognition with randomized trees," Neural Computation Vol. 9, No. 7, pp 1545–1588, 1997.

[45] L. Breiman, "Manual on setting up, using, and understanding Random Forests v3.1,". Technical Report, 2002, http://oz.berkeley.edu/users/breiman.

[46] R. Sharan, R. Elkon, R. Shamir, "Cluster analysis and its applications to gene expression data," Ernst Schering Res Found Workshop 2002, 83-108.

[47] W. Li, L. Jaroszewski, and A. Godzik, "Clustering of highly homologous sequences to reduce the size of large protein databases," Bioinformatics, Vol. 17,No. 3, pp 282-283, 2001.

[48] S. Brohee, J. van Helden, "Evaluation of clustering algorithms for protein–protein interaction networks," BMC Bioinformatics Vol. 7, No. 488, 2006

[49] S. Brohee, and J. Van Helden, "Evaluation of clustering algorithms for protein-protein interaction networks." BMC bioinformatics Vol. 7, No. 1, pp 488, 2006

[50] C, Yizong, and G. M. Church, "Biclustering of expression data." ISMB. Vol. 8. 2000.

[51] U. Maulik, S. Bandyopadhyay, and A. Mukhopadhyay, "Multiobjective Genetic Algorithms for Clustering: Applications in Data Mining and Bioinformatics." Springer Science & Business Media, 2011.

[52] U. Maulik, A. Mukhopadhyay, and S. Bandyopadhyay, Maulik, "Combining pareto-optimal clusters using supervised learning for identifying co-expressed genes." BMC bioinformatics 10.1 (2009): 27.

[53] A. Li and D. Tuck, "An Effective Tri-Clustering Algorithm Combining Expression Data with Gene Regulation Information," Gene Regul Syst Bio. Vol.3, pp 49–64, 2009.

[54] X. Chen, M. Wang, and H. Zhang. "The Use of Classification Trees for Bioinformatics." Wiley interdisciplinary reviews. Data mining and knowledge discovery 1.1 (2011): 55–63.

[55] S. Asur, D. Ucar and S. Parthasarathy, "An ensemble framework for clustering protein–protein interaction networks," Bioinformatics, Vol. 23, pp i29-i40, 2007.

[56] T., Shi, D. Seligson, A.S. Belldegrun, A. Palotie, S. Horvath, "Tumor classification by tissue microarray profiling: random forest clustering applied to renal cell carcinoma". Modern Pathology, Vol. 18, No. 4, pp 547–557, 2005.

[57] K.L. Lunetta, L.B. Hayward, J. Segal, P. Van Eerdewegh, "Screening large-scale association study data: exploiting interactions using random forests," BMC Genet., Vol. 5, p. 32, 2004.

[58] H. Pang, and H. Zhao. "Building pathway clusters from Random Forests classification using class votes." BMC bioinformatics 9.1 (2008): 87.

[59] T.D. Otto, D. Wilinski, S. Assefa, et al., "New insights into the blood-stage transcriptome of Plasmodium falciparum using RNA-Seq,". Mol Microbiol., Vol. 76, pp 12–24, 2010.

[60] R Core Team, "R: A language and environment for statistical computing," R Foundation for Statistical Computing, Vienna, Austria. 2014 URL http://www.R-project.org/

[61] R. Gentleman, V. J. Carey, D. M. Bates, et al., "Bioconductor: Open software development for computational biology and bioinformatics," Genome Biology, Vol.5, No.10:R80, 2004

[62] RStudio, " RStudio: Integrated development environment for R (Version 0.98.978) [Computer software],". Boston, MA. 2014. Retrieved July, 2014. URL http://www.rstudio.org/

[63] A. Liaw and M. Wiener, "Classification and Regression by randomForest," R News, Vol. 2, No. 3, pp 18—22, 2002.

[64] M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert and K. Hornik, "cluster: Cluster Analysis Basics and Extensions," R package version 2.0.1, 2015.

[65] F. Chang, W. Qiu, R. H. Zamar, R. Lazarus, X. Wang, "clues: An R Package for Nonparametric Clustering Based on Local Shrinking,". Journal of Statistical Software, Vol. 33, No. 4, pp 1-16, 2010.

[66] G. Brock, V. Pihur, S. Datta, S. Datta, "clValid: An R Package for Cluster Validation," Journal of Statistical Software, Vol. 25, No.4, pp 1-22, 2008

TABLE IV. COMPARISON BETWEEN THE PARTITION OF K-MEANS & RF AND PAM

		RF & PAM															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
K MEANS	1	0	0	0	0	76	0	0	0	0	0	0	0	1	0	2	0
	2	0	0	0	0	40	0	0	0	0	0	0	0	0	0	2	0
	3	279	334	461	274	0	172	295	224	295	331	394	303	335	272	42	228
	4	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	6	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0
	7	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
	8	0	0	0	0	10	0	0	0	0	0	0	0	0	0	2	1
	9	0	0	0	0	121	139	0	6	3	5	0	17	1	7	297	86
	10	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0
	11	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	12	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	13	0	0	0	0	32	0	0	0	0	0	0	0	0	0	0	0
	14	0	0	0	0	61	0	0	0	0	0	0	1	0	0	38	13
	15	0	0	0	0	21	1	0	0	0	0	0	0	0	0	9	2
	16	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0