DMWAS: Deep & Machine learning omics Wide Association Study & Feature set optimization by clustering & univariate association for Biomarkers discovery as tested on GTEx pilot dataset for death due to heart-attack

Abhishek N Singh (absingh@student.uef.fi)
Schiller International University

Research article

Keywords: SNP – Single Nucleotide Polymorphism, SVs – Structural Variations, MLP – Multi-Layer Perceptron, DNN – Deep neural network, DNA – Deoxyribonucleic acid, DIP – Deletion and Insertion Polymorphism, InDel – Insertion Deletion, DMWAS – Deep and Machine learning omics Wide Association Study, GWAS – Genome Wide Association Study, NGS – Next Generation Sequencing, ExhaustiveDNN – Exhaustive Deep Neural Network, TP – True Positive, TN – True Negative, FN – False Negative, FP – False Positive , CNV – Copy Number Variation, MLCSB – Machine Learning in Computational and Systems Biology, AUC – Area under curve, ROC – Receiver Operating Characteristic, PR-curve – Precision-recall curve, MHHRTATT - people who died of 'heart attack, acute myocardial infarction, acute coronary syndrome

Posted Date: May 26th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-413000/v2

Abstract

Univariate and multivariate methods for association of the genomic variations with the end-or-endo phenotype have been widely used for genome wide association studies. In addition to encoding the SNPs, we advocate usage of clustering as a novel method to encode the structural variations, SVs, in genomes, such as the deletions and insertions polymorphism (DIPs), Copy Number Variations (CNVs), translocation, inversion, etc., that can be used as an independent feature variable value for downstream computation by artificial intelligence methods to predict the endo-or-end phenotype. We introduce a clustering based encoding scheme for structural variations and omics based analysis. We conducted a complete all genomic variants association with the phenotype using deep learning and other machine learning techniques, though other methods such as genetic algorithm can also be applied. Applying this encoding of SVs and one-hot encoding of SNPs on GTEx V7 pilot DNA variation dataset, we were able to get high accuracy using various methods of DMWAS, and particularly found logistic regression to work the best for death due to heart-attack (MHHRTATT) phenotype. The genomic variants acting as feature sets were then arranged in descending order of power of impact on the disease or trait phenotype, which we call optimization and that also uses top univariate association into account. Variant Id P1_M_061510_3_402_P at chromosome 3 & position 192063195 was found to be most highly associated to MHHRTATT. We present here the top ten optimized genomic variant feature set for the MHHRTATT phenotypic cause of death.

Background

Genomes of individuals are said to be more than 99% similar. This small variation of less than 1% in the DNA accounts for vast amount of differences in endo-and-end phenotype and behavior of the person. Variations of single letters in nature, such as the letters A, T, G, C, N can be easily encoded, while numerical representation of variations of DNA of more than 1 letter need more complicated and logical methods. GWAS [12] has been used for univariate methods of association of these variations to end phenotype until now. A univariate method for association of the genomic variations with the end-or-endo-phenotype has been widely used through software tools such as snptest [23] and p-link [22]. Methods of multivariate GWAS where there are multiple phenotypes to associate with as dependent variables, which are claimed to perform better, have been suggested [24]. However, these associations still take one independent variable at a time for genome wide association, therefore are less stringent resulting in spurious results. We see lately that overall contribution of these loci to heritability of complex diseases is often less than 10% [13]. As pointed out from McCLellan and King, Cell 2010[14]:

To date, genome-wide association studies (GWAS) have published hundreds of common variants whose allele frequencies are statistically correlated with various illnesses and traits. However, the vast majority of such variants have no established biological relevance to disease or clinical utility for prognosis or treatment.
More generally, it is now clear that common risk variants fail to explain the vast majority of genetic heritability for any human disease, either individually or collectively (Manolio et al., 2009).

Until present, nobody has attempted to encode SVs in genomes larger than one base, here we call DIPs. As an example, deep learning has been deployed to predict gene expression from histone modifications [2]. Genome-wide assays of breast cancer by denoising autoencoders (DAs) employs a data-defined learning objective independent of known biology [3].So by using this independent system the information is not captured for any advantage. Convolution neural network has been used for classifying various kinds of tumors [4]. Deep learning has been used for pathology image classification [5] and does not tap into the SV of the genome information for the purpose. Recurrent neural network without deploying SVs of the genome has been used for heart failure onset [6].Articles [7, 8] act as a review article for deep learning in pharmaceutical research and drug discovery-SVs of genome for any advantageous role are not mentioned. Brain disorders, such as Alzheimer's disease, are evaluated using brain images using artificial intelligence techniques in article [9], yet heart related disorders use deep learning for magnetic resonance information in [10]. Article [11] tries to make use of transcriptomics data along with deep learning for drug prediction and repositioning, again SVs of the genomic data are not mentioned. Recently in 2019, paper [25], visits the idea of machine learning by SNP-only based approach, which fails to point out the impact of DIPs and its appropriate encoding to facilitate machine and deep learning.

SVs[20] in genomic data are obtained after comparing the patient's DNA sequence with a reference sequence and finding matches and mismatches using tools, such as GenomeBreak [16, 17]. Incorporation of DIPs or InDels to MLCSB cannot be avoided, as we are generating more and more sequences and the data is routinely being downstream analyzed for SVs. As DIPs essentially have all information for CNVs, inversions, translocations and other SVs on genome, encoding them would also indirectly encode the other SVs. Article [18], discusses utilizing tools for these SV detections, then comparing these variations to databases and conducting a knowledge mining [19] where these variations are known to be associated with a disease. DNA sequencing for individuals is becoming increasingly cheaper to obtain, for example via NGS at sequencing centers where it can be done at a scale thereby distributing the fixed cost [15].

Method

Paper [21], showed qualitatively that the deviation of the sum of the nucleotides in DIPs was generally higher than the deviation of the sum of the nucleotides of the SNPs for the whole genome. In other words, deviations in DIPs were more representative of the individual differences among them and could thus attribute to their differences in endo-phenotype or end-phenotype. As an example, paper [21] took the gender as the end phenotype and showed that the variance (and the standard deviation) between the set of structural variations (DIPs) was much higher than that of the sum of nucleotides of SNPs (Fig. 1.), and stating that structural variations were a stronger means to determine the phenotype, i.e. gender here. Inspired by the article, this paper is about fine tuning and quantifying individual DIPs, so we introduce a deviation from consensus score to quantify the differences in SVs for these letters while also using one-hot encoding for the single nucleotide bases.

We developed DMWAS suite to simulate genomic data for genomic co-ordinates as a combination of A, T,G C or a 4-letter combination for a larger letter. Comprised of a Python script genSampleData.py, it can

be used to generate genomic variation data specifying quantity of genomic loci, number of patients, frequency of occurrence of DIPs, and the maximum size of a DIP. Details of usage of script are specified in the downloadable ReadMe.md file from GitHub. For illustration purpose we are using 400 genomic co-ordinates.

In Fig. 2 are the simulated data for 40 columns and 8 patients. The simulated genotype data for 200 loci is provided as file multiColumnSample.csv. Once the simulated data is generated, and then we use the script splitMultiColDIPs.py to split each feature column into two columns, one column for 1 letter variants and another column for DIPs variants. The split file is available by name multiColumnSplitSample.csv. In Fig. 3 is an example of data with each column doubled as per described method.

With information for the DIPs as second column, we can extract them separately and conduct a clustering of the data getting the divergence score for each DIPs from the consensus. This method of encoding the letters based on divergence from a mean, median or consensus score is called 'DivScoreEncoding'. DivScoreEncoding is different than one-hot, word embedding, index based encoding and other kind of label encoding methods as described in the article 'Text Encoding: A Review' [26].We realize that the InDels or DIPs can be different from each other and the difference in biological relevance such as by means of frameshift of codon or mutation at a point need to be given a score in biological context. The traditional means of encoding texts, such as those discussed at [26], do not take biological evolutionary distance into account when encoding for DIPs or InDels. This method of DivScoreEncoding is applicable to larger insertions and deletions as well as for other SVs in the genome like CNVs, translocations, etc. Cross-species multiple sequence alignment has been tried using phylogenetic tree construction in article [30]. Papers [28, 29] generate pathogenic scores of InDels throughout the non-coding genome to classify them into pathogenic or not, and would be clearly very different in terms of method and application, although the similarity remains in terms of the concept of giving a score to the InDels based on a biological role. Figure 4 shows a sample clustering by multiple alignments done words of varying length with consensus and divergence score for each sequence.

For implementing DivScoreEncoding method by clustering, we have chosen T_coffee[1] application to get the divergence score. This third-party software is available online. A wrapper Python script multiColDIPsDiv.py is provided, which automates extraction of the DIPs from multiCoumnSplitSample.csv file, then passes it to T_coffee software for multiple sequence alignment and divergence score determination. Script reverseReadMulti.py is provided to reverse the scores obtained, and script ReplaceMultiColDIPsNew.py can be used to replace the DIPs with appropriate scores. This would lead to file with content such as in Fig. 5. The resulting file is also provided as MultiColDIPsScored.txt. The Python script encodeSNPs.py has been provided for this purpose; the resulting final scored and encoded file MultiColDIPsScoredEncoded.txt is also provided. Figure 6 shows a sample scored and encoded file snippet.

Since we had 40 individuals or rows, we generated 40 y-values 0–39, with the 1st row left as that of feature column variable names. File is named as Phenotype.txt. We decided to use several machine

learning methods, such as logistic regression, naïve bayes, gradient boosting, bagging, and adaboost, and deploy enhanced form of exhaustive multi-layer perceptron (MLP = in the form of DNN by incorporating early stopping criteria to avoid overfitting, using rectified linear unit (*ReLU*) as activation function to reduce weight adjustment time and addressing the vanishing gradient problem. We also introduce an exhaustive nature of exploration for the right hidden layer and hidden units by varying the number of layers and number of hidden units in the DNN in a loop. Each time the best scores were chosen for its number of hidden layers and units. This exhaustive nature of DNN, when the range was given in realistic bound proved more useful than simply adding hidden layers as in a typical DNN, and thereby gave profound results, so this approach is called 'ExhaustiveDNN'. The scripts ExhaustiveDNN.ipynb and ExhaustiveDNN.py are provided in DMWAS and feeds in MultiColDIPsScoredEncoded.txt as input file. The script internally looks for all columns with any null values that are removed before modeling. The data file was also separately checked for Null values and minor allele frequency (MAF) of at least 5% and the resulting encoded file is available at DMWAS as NullMafMultiColDIPsScoredEncoded.txt, which can be used as an alternative. From this file applying F-Test criteria for each of the feature columns we chose the top 1% of the feature set as the final data that the deep and machine learning scripts would work on. Feature set optimization has been an active area of research recently such as what we see in article [31]. Article [27] talks about various applications of deep learning in different spheres of biology, and to which ExhaustiveDNN as part of DMWAS with the DivScoreEncoding methodology can play vital role as it is exhibited in the results section later.

Results

ExhaustiveDNN proved very useful. 30% of data was used for test and prediction purpose, results shown as a confusion matrix. In less than a minute it resulted in model that was 100% accurate on the test data with the following configuration of hidden layers and hidden units, and the score on average for each training batch as 96%:hidden units: 8, hidden layers: 2, avg_score:0.9600000023841858. The confusion matrix is shown in Fig. 7.

Here accuracy is defined as:

Accuracy = (Tp + tn)/(Tp + fp + fn + tn)

Machine learning techniques, mentioned in previous section, were applied as well for which Fig. 8 shows their corresponding confusion matrices. Each script took less than 1 minute to produce the confusion matrix, precision-recall curve, ROC-curve and list the dominant features. The scripts are available in DMWAS as createLogitReg.py, createAdaBoost.py, createBagging.py, createGradientBoosting.py and createNaiveBayes.py, extratreeclassifier.ipynb, randomforest.ipynb, support vector.ipynb. When there would be imbalance in distribution of cases and control, then PR-Curve metrics would be worthwhile to discuss, as we later plan to scale up the work for larger dataset analysis in future. All results for ROC-

curve, PR-Curve, list of dominant column variables, etc. are made available at DMWAS GitHub. Table 1 below summarizes the accuracy values obtained from these machine and deep learning software tools.

Using these approaches, the Naïve Bayes method seems to have the highest positive hits detected with 75% accuracy in this simulated data.

However, using the ExhaustiveDNN approach with a variation of number of layers and hidden units, with early stopping conditions, gave the best result with an accuracy of 100% almost immediately. The trick is to set the initial set of hidden layers and hidden units large enough while running ExhaustiveDNN. The initial opinion of 100% accuracy would be that the model has perhaps done over-fitting, the early stopping condition ensures that over-fitting does not take place. This is further substantiated by the fact that ExhaustiveDNN does not give 100% accuracy in real GTEx data, as discussed later. ExhaustiveDNN when allowed to continue after the 1st model has been generated can lead to multiple models, each with different average accuracy score such as that shown below in Table 2 at epoch (cycles) of 100. The model with best average score is saved for its configuration to be used on test and real data.

Application Of Dmwas To Gtex V7 Pilot Dataset

We used the scripts of DMWAS for Genotype-Tissue Expression (GTEx) project V7 pilot dataset of 185 individuals, for the phenotype coded as MHHRTATT for the people who died of ‘heart attack, acute myocardial infarction, acute coronary syndrome’, and were able to see that most of the machine learning based algorithms could perform remarkably better for real case data. As an example, Fig. 9 shows the AUC for ROC-curve for logistic regression for the MHHRTATT phenotype. A score of 97.3% accuracy was obtained using logistic regression model of DMWAS as shown through the confusion matrix in Fig. 10 for which the test data was taken as the entire GTEx V7 Pilot dataset. The results obtained have been summarized in Table 3. The plots for various confusion matrices are shown as well in Fig. 11.

Discussion

Evidently for the given dataset of GTEx V7 Pilot the methods of support vector and logistic regression substantially outperformed other methods in terms of accuracy. Since true positives in the dataset were significantly less, once scaled up from pilot dataset to whole dataset analysis, such as for GTEx V8 data, the method would help determine the precision rather than just depend on accuracy.

For illustration purpose on real data, we showed how the results improve drastically as we achieve accuracy of 97.3 % for real case data of GTEx V7 Pilot, for logistic regression, compared to only 58.3% as in randomly generated simulated dataset. This 97.3% of accuracy was generated when the entire GTEx V7 Pilot data was used for testing purpose.

The tools and techniques discussed in DMWAS can be applied for solving other data science problems as once the encoding work is completed, the user can use any algorithm of his choice and not be locked

into using those provided or suggested in this paper. This applies to the clustering DivScoreEncoding method as well. For instance, we can give each letter a value, then calculate a mean or median score for the complete word, or use other sophisticated clustering method which use fuzzy logic for instance.

Optimized Feature Set For Mhhrtatt Biomarkers

These models can help us ‘optimize the feature set’ i.e., identify dominant variants that are strongly associated to the model - and thus to the disease. The possibility was explored on the simulated dataset as well as prediction was made for MHHRTATT trait for the GTEx V7 pilot dataset to see if we get a score for the DIPs variant columns. Table 4 lists a partial dependence score generated for the simulated data in which the variant columns were also captured. The partial dependence score is calculated for genomic variant columns having single nucleotide variant eg. 214_C would mean the C nucleotide at 214th column in the genome variant file, for showing that just the presence of DIP at a position eg. 232_I means that the 232nd genomic variant column having an insertion, for showing the effect of the insertion variants at that column simply the column number is stated eg. 377. For the real case data the top 10 optimized features were all belonging to InDel class as shown in Table 5 for the logistic regression; the column variable name and numbering is as per the GTEx data with extension filename .PED and the actual co-ordinates can be found by looking at the corresponding rows of .MAP file. Note that since the .PED file comprise of one major allele and another minor allele information, the number of columns with regard to the genotype information is twice that of the number of rows in .MAP file and so tracing back of the corresponding genomic map coordinate should be done accordingly. As an example if the optimized feature has name 16,830,168_G, then the .PED file corresponding feature co-ordinate removing the initial 6 columns is 16,830,168 and the genomic variant that is having an effect is G. The corresponding genomic map coordinate line number is CEILING (16,830,168 / 4) = 4,207,542. This in the .MAP file corresponds to variant Id kgp30994055 and at position 52587347 of chromosome 23. The list of top associated and least associated genomic variants with their chromosome number, variant Id, and genomic loci are stated in Table 5 and Table 6 respectively. Apparently, the lowest scoring features were all SNPs (Table 6) however the relative difference in the tops scorer and bottom scorer were not huge indicating a rheostat model of combined effect of the variants on the phenotype.

Conclusion & Future Work

This paper has demonstrated and advocates use of clustering divergence score as a new way of genomic variant encoding particularly for structural variants larger than point mutations and demonstrated in for InDels, though the technique can well be applied to other SVs such as copy number variants, etc. Several machine learning algorithms were experimented and MLP (multi-layer perceptron) script with alterations to gain properties of deep learning was developed in Python, such as early stopping condition to avoid over-fitting. This led us to 100% accuracy using ExhaustiveDNN for the simulated data while accuracy for the real data was lower; confirming no case of over-fitting as far as the script logic is concerned. Other machine learning techniques such as bagging gave results lower than DNN with highest being 75% using

Naïve Bayes for the simulated data. The concept of clustering score is central to the ideas discussed in this paper and once the divergence scores are obtained, the downstream modeling advance algorithm need not be just restricted to those mentioned in DMWAS GitHub page but could also use many other deep and machine learning algorithms such as even genetic algorithm as has been used for GARBO [31].

We gave results for optimized feature for the top 10 genomic variants for MHHRTATT heart disease related death. Future work requires up-scaling the analysis for the entire dataset such as GTEx V8 since the number of cases of individuals having the trait in pilot sample is very limited, thereby having considerable under performance for most of the deep and machine learning models. Future work also asks for exploring and comparing performance of other similar tools that deploy machine and deep learning for GWAS, such as CADD [30] even though it was used in cross-species context, or GARBO[31] which uses fuzzy logic and genetic algorithm, and see if there are complementary aspects that DMWAS can benefit from, in a future version of the tool. The purpose of the current work was to not just describe a method, but also list top genomic variants associated to MHHRTATT. The idea is also to make DMWAS available and deployable for the purpose of deep learning and machine learning application to GWAS.

Open-Source Development & Supplementary:

The PR-Curve, ROC-Curve, PDValues (partial dependency scores based on the model for the genomic variant columns) for the simulated data and python scripts including script to simulate data is publicly accessible here: https://github.com/abinarain/DMWAS. Supplementary materials can be downloaded from https://sites.google.com/a/iitdalumni.com/abi/educational-papers .

Declarations

Open-Source Development & Supplementary:

The PR-Curve, ROC-Curve, PDValues (partial dependency scores based on the model for the genomic variant columns) for the simulated data and python scripts including script to simulate data is publicly accessible here: *https://github.com/abinarain/DMWAS*. Supplementary materials can be downloaded from https://sites.google.com/a/iitdalumni.com/abi/educational-papers .

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and material

Genotype-Tissue Expression (GTEx) dataset https://gtexportal.org/home/index.html .

Competing interests

None.

Funding

No funding was obtained for this study.

Authors' contributions

The complete work, idea and manuscript preparation was done by Abhishek.

Acknowledgement

The author thanks all developers and teachers of Python programming language. The data was used as part of an authorized access request while the author was employed at A.I. Virtanen Institute employed and working for Genotype-Tissue Expression (GTEx) dataset https://gtexportal.org/home/index.html. The author is grateful to Dr Minna Kaikkonen for giving him the opportunity to work on the data. The author is also thankful to arXiv [32] for enabling a preprint of the article.

References

1. T-Coffee. A novel method for multiple sequence alignments. Notredame,Higgins,Heringa,JMB,302(205–217)2000.
2. Singh R, Lanchantin J, Robins G, Qi Y. DeepChrome: deep-learning for predicting gene expression from histone modifications, Bioinformatics, Volume 32, Issue 17, 1 September 2016, Pages i639–i648, https://doi.org/10.1093/bioinformatics/btw427.
3. Tan J, Ung M, Cheng C, Greene CS. (2015). Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders. Pacific Symposium on Biocomputing,132–143.
4. Benjamin Q, Huynh H, Li ML, Giger, "Digital mammographic tumor classification using transfer learning from deep convolutional neural networks," J. Med. Imag. 3(3) 034501 (22 August 2016) https://doi.org/10.1117/1.JMI.3.3.034501.
5. Janowczyk A, Madabhushi A. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. J Pathol Inform. 2016;7:29.
6. Choi E, Schuetz A, Stewart WF, Sun J. Using recurrent neural network models for early detection of heart failure onset. J Am Med Inform Assoc. 2017;24(2):361–70. doi:10.1093/jamia/ocw112.
7. Ekins S. The Next Era: Deep Learning in Pharmaceutical Research. Pharm Res. 2016;33:2594–603. doi:10.1007/s11095-016-2029-7.
8. Gawehn E, Hiss JA, Schneider G. Deep Learning in Drug Discovery. Mol Inf. 2016;35:3–14. doi:10.1002/minf.201501008.
9. Ortiz A, Munilla J, Gorriz JM. J. RamirezEnsembles of deep learning architectures for the early diagnosis of the Alzheimer's disease Int. J. Neural Syst., 26 (07) (2016).

10. Ngo TA, Lu Z, Carneiro G. Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance. Med Image Anal. Jan. 2017;35:159–71.
11. Deep Learning Applications for Predicting Pharmacological Properties of Drugs and Drug Repurposing Using Transcriptomic. DataAlexander Aliper, Sergey Plis, Artem Artemov, Alvaro Ulloa, Polina Mamoshina, and Alex ZhavoronkovMolecular Pharmaceutics 2016 13 (7), 2524–2530 DOI: 10.1021/acs.molpharmaceut.6b00248.
12. Manolio TA. "Genomewide association studies and assessment of the risk of disease". The New England Journal of Medicine. July 2010;363(2):166–76. doi:10.1056/NEJMra0905980.
13. Peter MVisscher, Mark MABrown, JianYang IMcCarthy, Five Years of GWAS Discovery, AJHG, Volume 90, Issue 1, 13 January 2012, Pages 7–24.
14. McClellan J, King MC. Genetic heterogeneity in human disease Cell, 141 (2010), pp. 210–217.
15. Schwarze K, Buchanan J, Fermont JM, et al. The complete costs of genome sequencing: a microcosting study in cancer and rare diseases from a single center in the United Kingdom. Genet Med. 2020;22:85–94. doi:10.1038/s41436-019-0618-7.
16. Abhishek N, Singh, Comparison of Structural Variation between Build 36 Reference Genome and Celera R27c Genome using GenomeBreak, Poster Presentation, The 2nd Symposium on Systems Genetics, 29–30 September 2011, Groningen.
17. 22. Singh A. GENOMEBREAK: A versatile computational tool for genome-wide rapid investigation, exploring the human genome, a step towards personalized genomic medicine, Poster 70, Human Genome Meeting 2011, Dubai, March 2011.
18. Abhishek N, Singh. A105 Family Decoded: Discovery of Genome-Wide Fingerprints for Personalized Genomic Medicine, Poster, 2–5 Feb UPCP 2012, Florence, Italy http://f1000.com/posters/browse/summary/1089898.
19. Singh AN. Knowledge Mining and Bioinformatics Tools to Advance Personalized Diagnostics and Therapeutics, USISTF organized Workshop, Florence Nov 2012, Italy http://tinyurl.com/biomining http://hit.fiu.edu/W/pre-report.pdf.
20. Singh AN. Variations in Genome Architecture, Poster, International Congress on Personalized Medicine, 2–5 Feb UPCP 2012, Florence, Italy http://f1000.com/posters/browse/summary/1089896.
21. Singh AN, Informatics CB, Nature S, Analytics BMC,BData, May 2018 https://bdataanalytics.biomedcentral.com/articles/10.1186/s41044-018-0030-3.
22. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MAR, Bender D, Maller J, Sklar P, de Bakker PIW, Daly MJ, Sham PC. (2007), PLINK: a toolset for whole-genome association and population-based linkage analysis. American Journal of Human Genetics, 81.
23. Marchini J, Howie B, Myers S, McVean G, Donnelly P. A new multipoint method for genome-wide association studies by imputation of genotypes. Nat Genet. 2007 Jul;39(7):906–13. Epub 2007 Jun 17. PubMed PMID: 17572673.

24. Galesloot TE, van Steen K, Kiemeney LALM, Janss LL, Vermeulen SH. A Comparison of Multivariate Genome-Wide Association Methods. PLoS ONE. 2014;9(4):e95923. https://doi.org/10.1371/journal.pone.0095923.
25. Ho DSW, Schierding W, Wake M, Saffery R, O'Sullivan J. Machine Learning SNP Based Prediction for Precision Medicine. Front Genet. 2019;10:267. doi:10.3389/fgene.2019.00267.
26. Text Encoding: A Review Posted by Rosaria Silipo on February 11, 2019 at 3:09pm https://www.datasciencecentral.com/profiles/blogs/text-encoding-a-review.
27. Ching Travers. et. al., Opportunities and obstacles for deep learning in biology and medicine. 15. J. R. Soc. Interfacehttp://doi.org/10.1098/rsif.2017.0387.
28. Ferlaino M, Rogers MF, Shihab HA, Mort M, Cooper DN, Tom R, Gaunt C, Campbell. An integrative approach to predicting the functional effects of small indels in non-coding regions of the human genome. BMC Bioinformatics. 2017;18:442.
29. Hashem A, Shihab MF, Rogers J, Gough M, Mort DN, Cooper, Ian NM, Day TR, Gaunt. Colin Campbell. An integrative approach to predicting the functional effects of non-coding and coding sequence variation. Bioinformatics. 2015;31:1536–43.
30. Philipp Rentzsch D, Witten GM, Cooper J, Shendure M. Kircher, CADD: predicting the deleteriousness of variants throughout the human genome, *Nucleic Acids Research*, Volume 47, Issue D1, 08 January 2019, Pages D886–D894, https://doi.org/10.1093/nar/gky1016.
31. Fortino V, Scala G, Greco D. Feature set optimization in biomarker discovery from genome-scale data, *Bioinformatics*, btaa144, https://doi.org/10.1093/bioinformatics/btaa144.
32. arXiv:2102.13470 [q-bio.QM].

Tables

Table 1
ExhaustiveDNN outperforming some of the popular Machine Learning methods for simulated dataset

Algorithm	Accuracy %
Exhaustive Deep Neural Network	100
Logistic Regression	58.33
AdaBoost	66.67
GradientBoost	50
Naïve Bayes	75
Bagging	33.33
Support Vector	66.67
Random Forest	50
Extra Tree Classifier	66.67

Table 2
ExhaustiveDNN leading to several different average accuracy score for various combinations of hidden layers and hidden units.

# Hidden Layers	# Hidden Units in Each Layer	Average Score of K-fold (k = 10)
2	8	0.9600000023841858
3	8	0.9400000005960465
4	8	0.9600000023841858
5	8	0.9400000035762787
6	8	0.9600000023841858
7	8	0.9600000023841858
2	9	0.9800000011920929
3	9	0.9800000011920929
4	9	0.9600000023841858
5	9	0.9200000047683716
6	9	0.6333333551883698
7	9	0.9800000011920929
2	10	0.9400000005960465
3	10	0.9600000023841858
4	10	0.6333333551883698
5	10	0.9600000023841858
6	10	0.9600000023841858
7	10	0.6333333551883698
2	11	0.9600000023841858
3	11	0.9400000005960465
4	11	0.9400000005960465
5	11	0.9600000023841858
6	11	0.9400000035762787
7	11	0.9400000005960465

Table 3
List of various machine and deep learning algorithms with the score of their accuracy.

Algorithm	Accuracy %
Exhaustive Deep Neural Network	78.5
Logistic Regression	94.64
AdaBoost	76.78
GradientBoost	76.78
Naïve Bayes	76.78
Bagging	78.5
Support Vector	94.64
Random Forest	78.5
Extra Tree Classifier	78.5

Table 4

List of partial score and the corresponding column explanatory genomic variant variable

PDValues	ColumnName
0.690258855	289
0.690258855	232_I
0.690258855	53
0.690258855	9
0.690258855	3
0.690258855	288_I
0.690258855	233
0.690258855	377
0.690258855	267
0.690258855	259
0.690258855	145
0.690258855	392_I
0.690258855	214_C
0.690258855	356_I
0.690258856	226_I
0.690258856	234_I
0.690258857	196_C
0.690258857	380_T
0.690258857	68_I
0.690258858	296_I
0.690258858	396_T
0.690258858	110_A
0.690258858	206_G

PDValues	ColumnName
…	395
…	
…	
0.711782557	
0.712538851	137
0.712730046	81
0.712760458	399
0.713352305	121
0.713557698	183
0.714217146	197
0.714282215	349
0.714400629	389
0.716420812	149
0.719423753	329
0.724220414	113
0.72980916	187
PDValues	ColumnName

Table 5

List of top 10 partial score as per the logistic regression and the corresponding column explanatory genomic variant variable column number as per the GTEx V7 pilot data numbering. The corresponding genomic co-ordinates can be found using the .MAP and .PED file information from GTEx dataset as described in 'Optimized Feature set for MHHRTATT biomarkers' section and are also shown in the table

PDValues	ColumnName
0.13895276827249736	9395961
0.1389563978100227	2 7104275
0.13923530541027984	11354221
0.13927094791319042	11050029
0.13947943677072142	9281287
0.13949864891527605	6479351
0.13971383684704966	4671785
0.13977059245647452	2642209
0.14012947617522825	3610447
0.14211946648423188	3884145

GTEx Pilot 5M.PED.MAP File ROW Number	Genotype
2348991	Chromosome 9 position 95811874 and variant Id P1_M_061510_9_203_M
1776069	Chromosome 6 variant Id P1_M_061510_6_987_P position 162112867
2838556	Chromosome 12 variant Id P1_M_061510_12_59_P genomic position 5223453
2762508	Chromosome 11 variant Id P1_M_061510_11_420_M genomic position 93911243
2320322	Chromosome 9 variant Id P1_M_061510_9_163_M genomic position 78004294
1619838	Chromosome 6 variant Id P1_M_061510_6_181_P genomic position 48930947
1167947	Chromosome 4 variant Id P2_M_061510_4_715_M genomic position 137617593
660553	Chromosome 2 variant Id P1_M_061510_2_509_P genomic position 233364549
902612	Chromosome 3 variant Id P1_M_061510_3_309_M genomic position 145931899
971037	Chromosome 3 variant Id P1_M_061510_3_402_P genomic position 192063195
PDValues	ColumnName

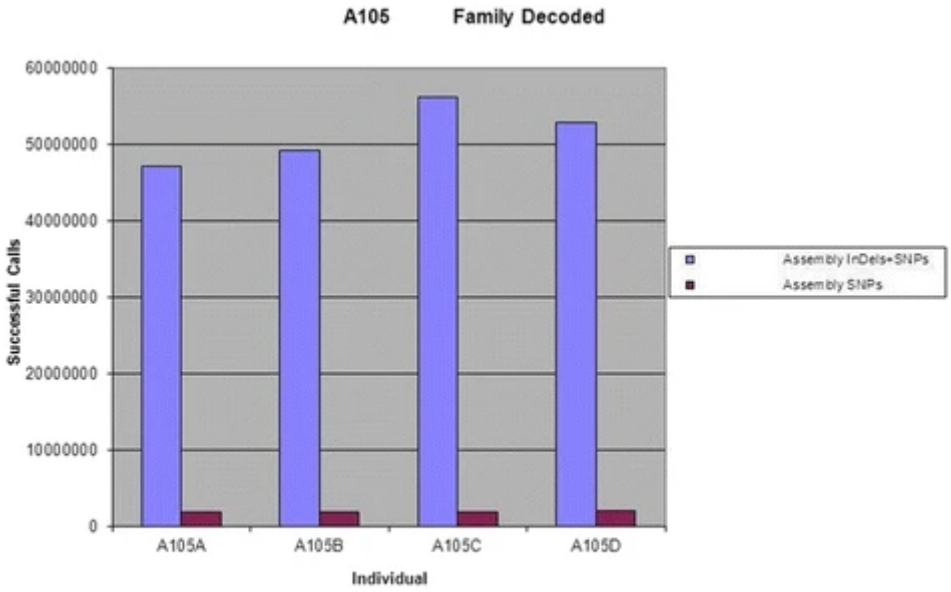
Table 6

List of bottom 10 partial score as per the logistic regression and the corresponding column explanatory genomic variant variable column number as per the GTEx V7 pilot data numbering. The corresponding genomic co-ordinates can be found using the .MAP and .PED file from GTEx dataset information as described in 'Optimized Feature set for MHHRTATT biomarkers' section and are also shown in the table.

PDValues	ColumnName
0.13240398739505238	16830168_G
0.13240398739505238	16830170_G
0.13240398739506198	7592676_T
0.1324039873952151	3591768_T
0.1324039873952151	3591288_C
0.1324039873952151	13241510_T
0.1324039873952151	5093676_G
0.1324039873952151	5093678_G
0.1324039873952151	14435950_A

GTEx Pilot 5M.PED.MAP File ROW Number	Genotype
4207542	Chromosome 23 variant Id kgp30994055 genomic position 52587347
4207543	Chromosome 23 variant Id kgp31134917 genomic position 52588392
1898169	Chromosome 7 Variant Id kgp11290556 genomic position 70226068
897942	Chromosome 3 Variant Id kgp5923265 genomic position 142797398
897822	Chromosome 3 Variant Id kgp18185020 genomic position 142711709
3310378	Chromosome 14 Variant Id kgp28093020 genomic position 97615238
1273419	Chromosome 5 Variant Id kgp22643217 genomic position 13809129
1273420	Chromosome 5 Variant Id kgp22679345 genomic position 13809146
3608988	Chromosome 17 Variant Id kgp5104948 genomic position 4991686

Figures



Structural Variations as detected for the whole Genome

Figure 1

Here we see the Figure 4. from paper [21] where clearly higher deviation is observed for sum of nucleotides of SNPs plus DIPs (blue) compared to sum of bases for the SNPs (magenta).

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39

A,C,G,G,G,A,A,G,G,A,G,C,T,G,G,G,G,A,A,G,T,C,A,T,T,C,G,A,A,T,G,C,T,A,C,G,T,T,A,G

C,A,A,T,A,G,A,C,G,C,A,C,G,G,T,C,TGAGAG,G,T,C,C,A,G,G,T,G,G,GCATTC,C,G,C,G,C,T,T,T,G,G,A,T

G,AGCATATA,T,A,C,CCTGGCCC,G,G,G,T,T,A,G,C,G,A,A,TGACA,A,A,T,G,T,G,G,T,G,A,T,A,C,A,G,G,A,T,A,G,T,A

C,C,C,A,G,C,TTTCG,C,A,T,T,A,C,A,A,A,T,A,A,C,G,GG,A,C,C,A,C,G,A,A,G,T,C,T,ATTAAGTCG,G,G,G,C,A

T,C,A,G,T,T,C,A,A,T,T,G,C,A,C,C,T,G,A,A,C,A,A,G,G,G,A,G,A,G,C,A,GGG,C,A,C,G,A,C,GG

C,T,G,G,A,C,T,C,A,A,T,C,A,C,A,G,G,G,C,G,A,GCGGG,T,A,C,T,C,C,T,G,T,C,G,T,G,C,T,G,T,T

A,C,T,G,G,T,A,G,G,T,C,G,G,A,A,A,T,G,T,T,A,G,G,G,C,G,T,G,T,A,G,A,T,A,G,C,C,G,A,A

C,A,C,A,T,A,T,A,G,G,T,TAAAAAT,T,A,G,T,T,ACCCATGA,G,A,A,G,T,G,T,A,A,T,G,C,G,C,C,A,A,G,A,G,C,C

Figure 2

Randomly generated Genotype data for 8 patients for illustration purpose. The example data chosen for simulation comprises of 40 individuals and 200 genotypic loci.

0	1	2	3	4	5	6	7	8	9
T	0	T	0	T	0	T	0	T	0
A	0	A	0	C	0	I	CATTCTAGC	C	0
C	0	T	0	C	0	C	0	T	0
G	0	C	0	A	0	T	0	C	0
T	0	C	0	A	0	A	0	C	0
G	0	G	0	C	0	G	0	A	0
I	CGCGAGGGAGCGT	A	0	C	0	A	0	T	0
T	0	C	0	G	0	G	0	G	0
I	CTAGA	C	0	C	0	T	0	C	0
G	0	T	0	C	0	A	0	A	0
I	CGATCTACAGACG/	G	0	G	0	G	0	A	0
A	0	A	0	T	0	A	0	C	0
G	0	A	0	G	0	T	0	A	0
G	0	T	0	A	0	G	0	A	0
T	0	C	0	C	0	C	0	C	0
C	0	I	ATTGTAGGCAGGC	A	0	C	0	A	0

Figure 3

For illustration purpose, we show how the DIPs columns are generated, by splitting each feature column variable into 2.

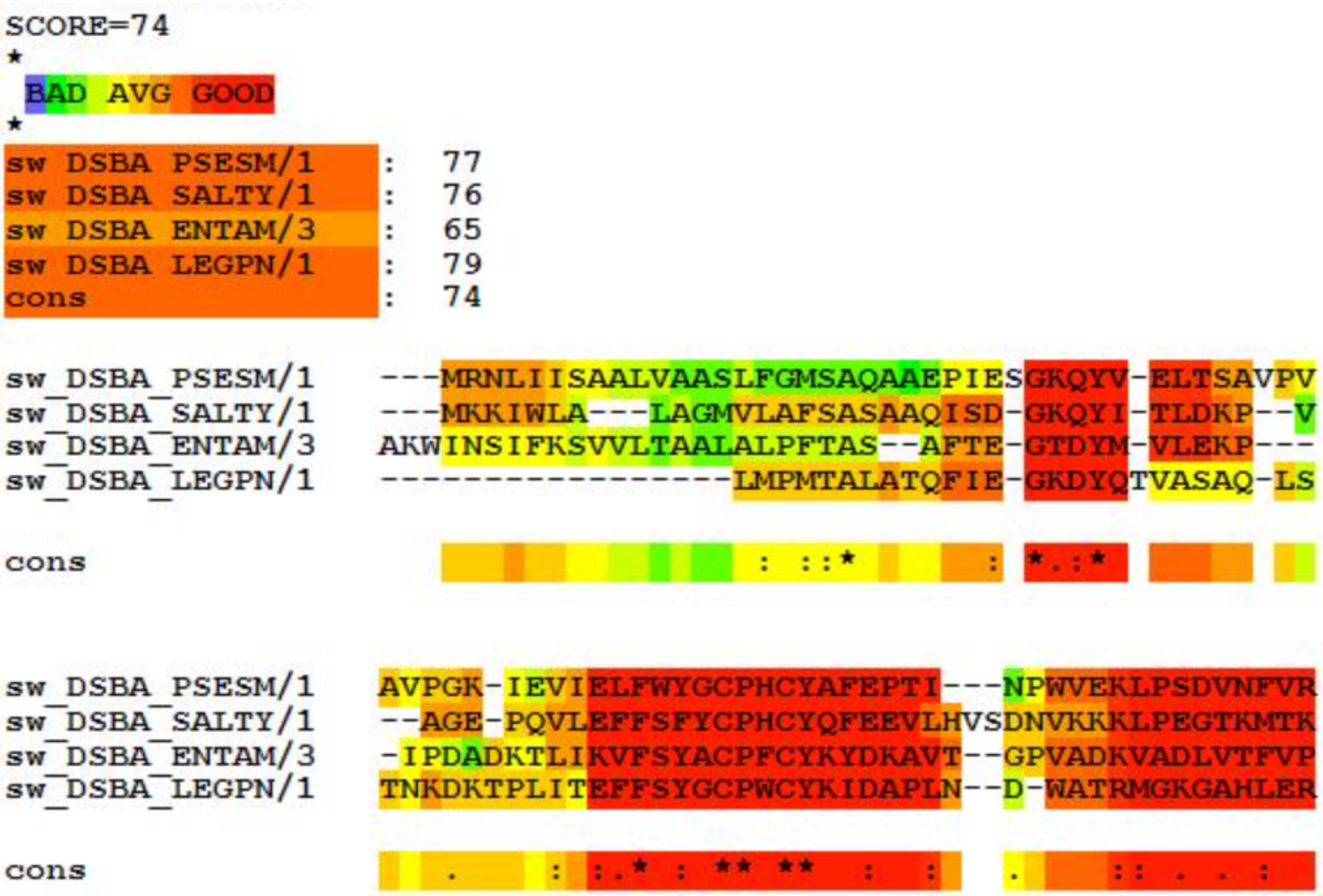


Figure 4

Example of a sample clustering by multiple sequence alignment with consensus regions and the consensus score with individual scores as well (which we call divergence score in this paper).

```
T       0.0     T       0.0     T       0.0     T       0.0     T
T       0.0     C       0.0     T       0.0     A       0.0     T
G       0.0     T       0.0     G       0.0     C       0.0     G
C       0.0     G       0.0     T       0.0     A       0.0     G
C       0.0     A       0.0     A       0.0     C       0.0     C
G       0.0     C       0.0     I       79.0    A       0.0     C
T       0.0     T       0.0     C       0.0     I       58.0    C
C       0.0     C       0.0     T       0.0     T       0.0     T
T       0.0     A       0.0     A       0.0     G       0.0     T
G       0.0     G       0.0     A       0.0     T       0.0     G
T       0.0     G       0.0     I       61.0    A       0.0     A
T       0.0     A       0.0     A       0.0     A       0.0     T
C       0.0     A       0.0     C       0.0     G       0.0     C
T       0.0     A       0.0     T       0.0     T       0.0     T
```

Figure 5

The DIPs are replaced by the corresponding divergence from consensus score, lying between 0 and 100.

```
0,0,0,0,1,0.0,0,0,0,0,1,0.0,0,0,0,0,1,0.0,0,0,0,1,0.0,0,0,0
,0,0,0,0,1,0.0,0,0,0,0,1,0.0,1,0,0,0,0,0.0,0,0,1,0,0,0.0,0,1
0,0,0,0,1,0.0,0,0,0,0,1,0.0,0,0,0,0,1,0.0,1,0,0,0,0,0.0,1,0,0
1,0,0,0,0,0.0,1,0,0,0,0,0.0,0,1,0,0,0,0.0,0,0,0,1,0,44.0,0,1
0,0,1,0,0,0.0,0,1,0,0,0,0.0,0,0,1,0,0,0.0,1,0,0,0,0,0.0,0,1,0
,1,0,90.0,1,0,0,0,0,0.0,0,1,0,0,0,0.0,0,0,0,1,0,34.0,1,0,0,0
0,1,0,0,0,0.0,0,0,0,0,1,0.0,0,1,0,0,0,0.0,0,1,0,0,0,0.0,0,0,0
,0,0,0,0,1,0.0,1,0,0,0,0,0.0,0,0,1,0,0,0.0,0,0,0,1,0,33.0,0,1
0,1,0,0,0,0,0.0,0,0,1,0,0,0.0,0,0,0,0,1,0.0,0,0,0,0,1,0.0,0,0
```

Figure 6

Now, the single nucleotide variations, SNVs or SNPs, are also one-hot encoded

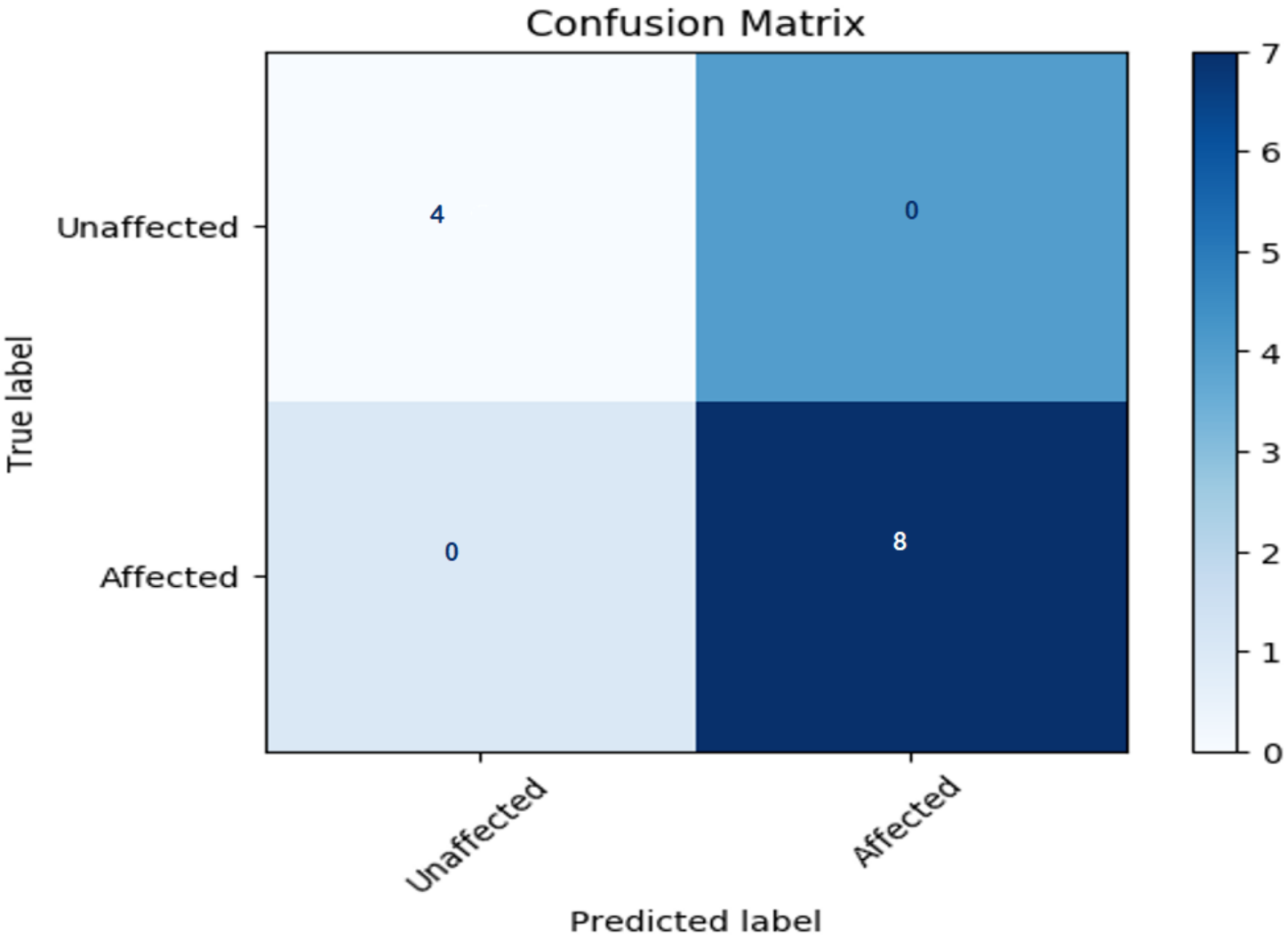


Figure 7

Confusion MATRIX for ExhaustiveDNN model for simulated dataset.

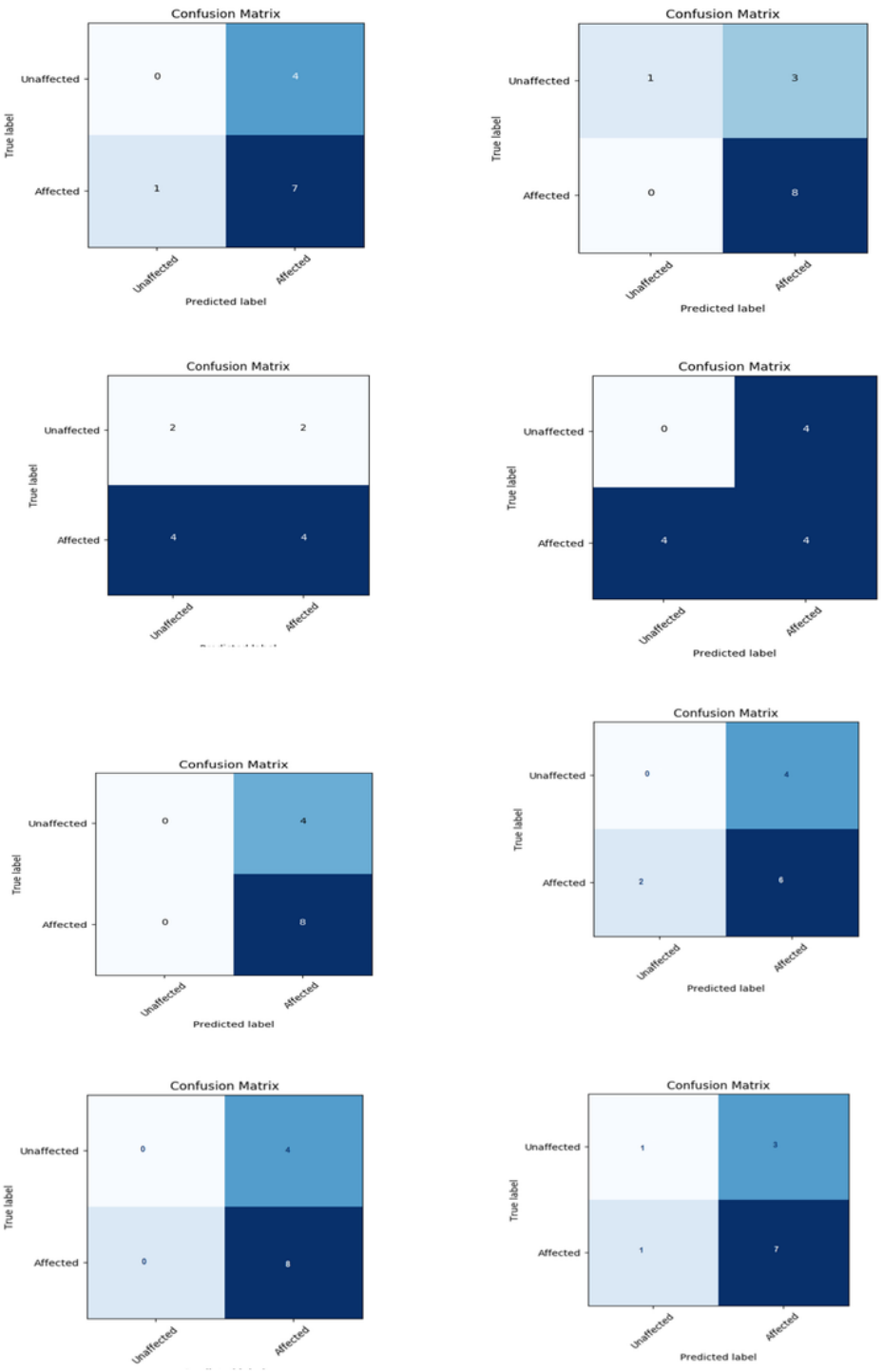


Figure 8

Top left to bottom right: Confusion MATRIX for logistic regression, Naïve Bayes, Gradient Boost, Bagging approach, AdaBoost, RandomForest, Support Vector & Extratree Classifier respectively for simulated dataset

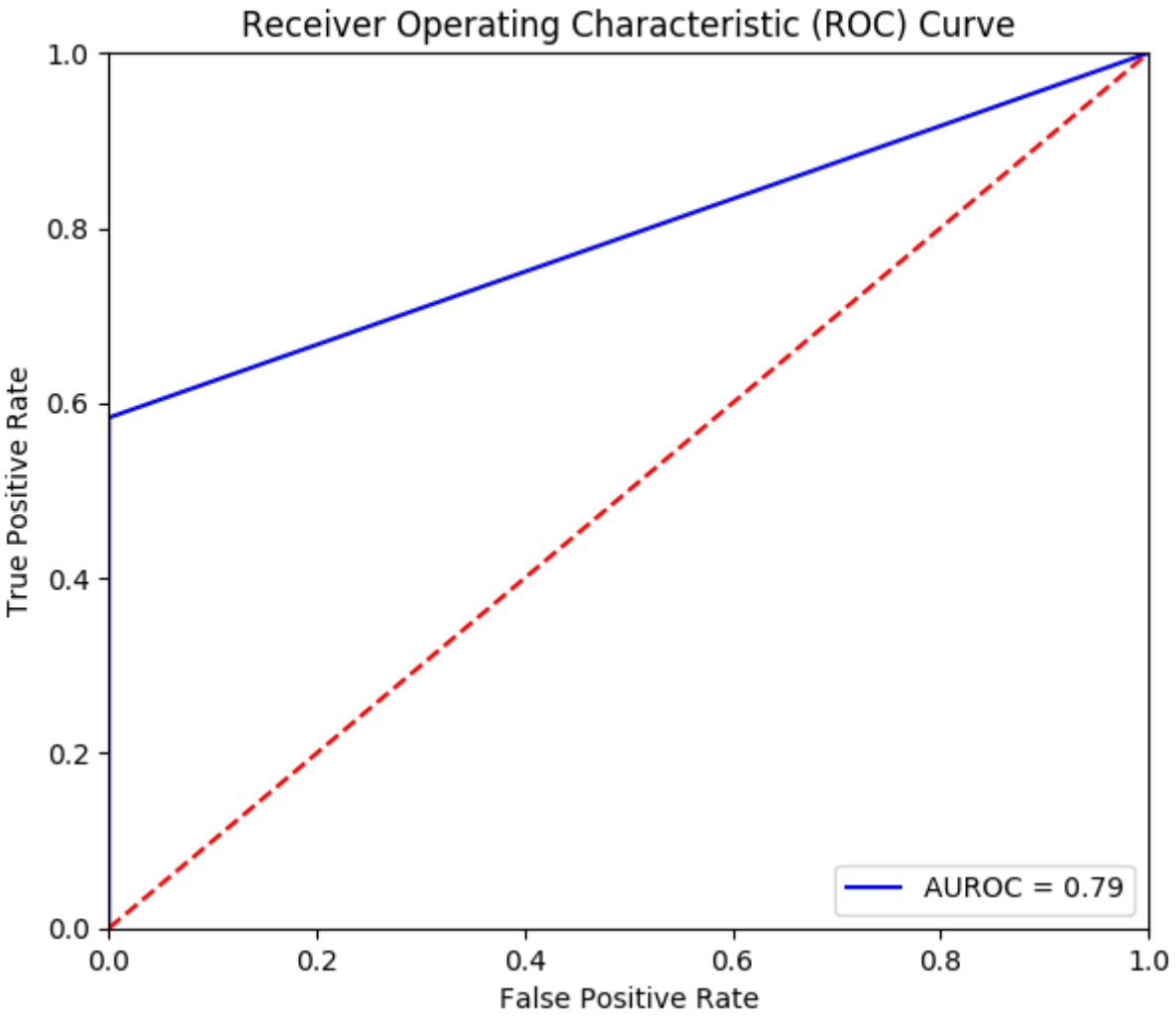


Figure 9

ROC curve for Logistic regression in DMWAS suite MHHRTATT trait for GTEx V7 pilot dataset

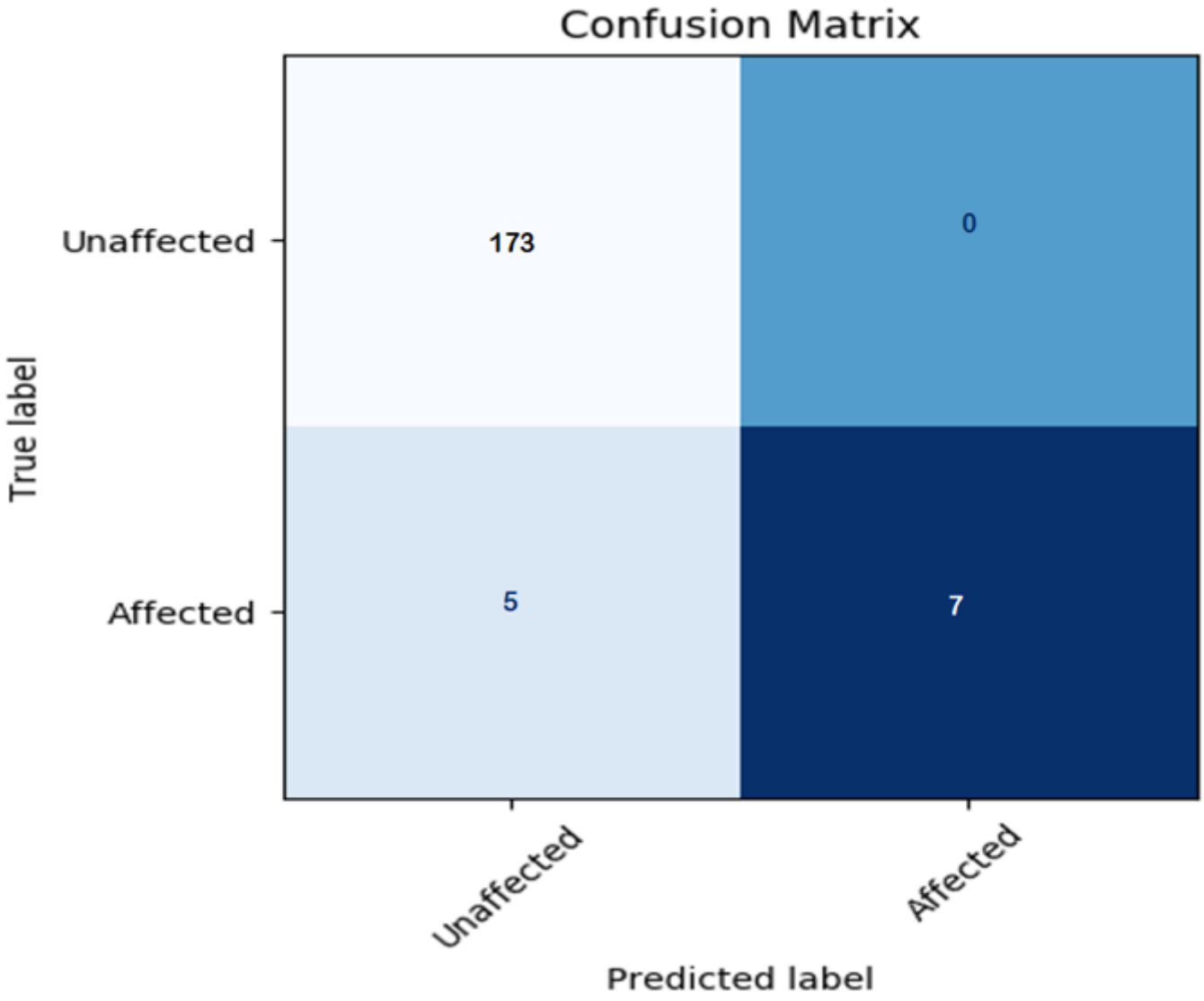


Figure 10

Confusion Matrix of Logistic Regression of DMWAS on GTEx V7 Pilot data for MHHRTATT trait giving accuracy of 97.3%

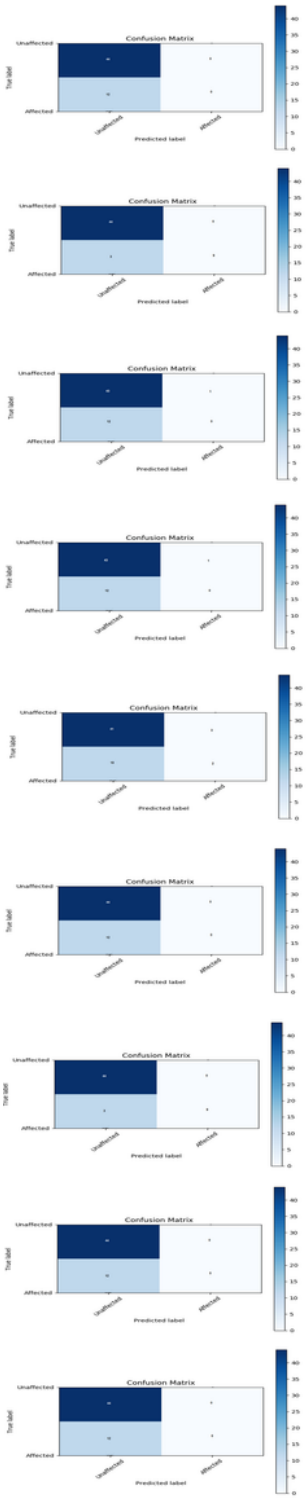


Figure 11

Confusion Matrices top to bottom for ExhaustiveDNN, Logistic Regression, AdaBoost, GradientBoost, Naïve Bayes, Bagging, Support Vector, Random Forest, ExtraTreesClassifier for MHHRTATT phenotype GTEx V7 Pilot dataset.