An empirical comparison between polygenic risk scores and machine learning for case/control classification

Muhammad Muneeb (muneebsiddique007@gmail.com)
Khalifa University of Science and Technology
Samuel Feng
Khalifa University of Science and Technology
Andreas Henschel
Khalifa University of Science and Technology

Research Article

Keywords: polygenic risk scores, genotype-phenotype prediction, genetics, bioinformatics, applied machine learning

Posted Date: February 24th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1298372/v1

RESEARCH

An empirical comparison between polygenic risk scores and machine learning for case/control classification

Muhammad Muneeb[1,2*†], Samuel Feng[2,3†] and Andreas Henschel[1]

*Correspondence:
muneebsiddique007@gmail.com
[1]Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology, Al Saada St - Zone 1, Abu Dhabi, United Arab Emirates
[2]Department of Mathematics, Khalifa University of Science and Technology, Al Saada St - Zone 1, Abu Dhabi, United Arab Emirates
Full list of author information is available at the end of the article
†Equal contributor

Abstract

Background: We compared the procedure to calculate polygenic risk scores and machine learning for simulated data, devised a way to compare machine learning results with PRS, and highlighted the required files formats for PRS calculation and machine learning model training. For PRS calculation, we used three tools: Plink, PRSice, and Lassosum, and for the machine learning algorithm, we used artificial neural networks.

Results: Based on our survey, we cannot say machine learning is better or polygenic risk scores because it depends on the phenotype under consideration. The average classification AUC of PRSice, Plink, Lassosum, and Machine learning was 0.27, 0.3, 0.35, and 0.87 on simulated data.

Conclusion: This article presents the comparison method in an automated way, ultimately assisting in various analyses. For instance, datasets with different heritability or genetic variations can be generated, and the effect on machine learning algorithms' accuracy and PRS's accuracy can be studied. Such analyses may require the generation of multiple datasets, calculation of PRS, and training machine learning model, which can be done quickly using the code segments and scripts provided in this manuscript. Apart from that, we compared the steps of PRS calculation with machine learning and found some steps are optional in machine learning.

Keywords: polygenic risk scores; genotype-phenotype prediction; genetics; bioinformatics; applied machine learning

Background

Among various diseases, some are purely genetic diseases [1, 2]. Genetic disorders risk computation method [3, 4] like polygenic risk score (PRS) [5] is employed to find people susceptible to a particular disease. A similar analysis can be performed using machine learning, as illustrated in this manuscript. PRS is a more substantial quantity than classification in machine learning because PRS predicts a particular person's tendency to have a specific disease or trait [6, 7]. In contrast, machine learning classifies people into traits or categories [8].

Results generated by the machine learning (ML) algorithm can be interpreted in a particular way to compare ML and PRS. In machine learning, samples are given a classification score that indicates how strongly any sample relates to a specific category [9]. Suppose all people above 0 are cases and below 0 control, where 0 is the threshold. If two people have a classification score of 0.5 and 1, they both are

cases, but the person who has a classification score of 1 is more strongly related to a particular trait or disease. Such a comparison can be performed to compare PRS and machine learning classification scores. Although the machine learning model classifies people into cases and controls, prediction probability can be normalized between 0 and 1 if we use sigmoid [9] in the last layer and normalize the values between 0 and 1. Similarly, polygenic risk scores generated by various tools can be normalized between 0 and 1 for comparison purposes.

Following are the major scientific and computational contributions of this research.

- This article compared machine learning and PRS for simulated genotyped data.
- We devised a way to convert the machine learning probability score to a normalized polygenic risk score.
- Calculating PRS is a highly complex procedure, and it involves a lot of several methods and quality control steps that should be performed for high-quality results. Some steps are compulsory in PRS calculation, but some can be relaxed for machine learning without affecting the final results. We compared these steps with the machine learning technique.
- The article focuses on the methodology and provides scripts that can be used to perform analyses like genetic variation effects on PRS and machine learning.

Based on our survey, we cannot say machine learning is better or polygenic risk score is better because it depends on the phenotype under consideration. For instance, in this article [10], researchers benchmarked PRS and compared it with machine learning algorithms like logistic regression, naïve Bayes, random forests, support vector machines, and gradient boosting on CAD disease, and PRS outperformed other algorithms. Whereas in this article [11], researchers used a deep neural network to improve the estimation of polygenic risk scores for breast cancer, and a deep neural network outperformed alternative techniques like LDpred.

In this article [8], researchers benchmarked 9 deep/machine learning algorithms for eye-color and type-2 diabetes prediction. In this article [12], researchers benchmarked PRS calculation using 4 tools: PRScise, Plink, LDpred-2, and Lassosum, which we also followed when comparing the machine learning with PRS.

This article focuses on the computational part, not the actual phenotype classification.

Methodology and Implementation

This section explains the dataset generation, dataset division into training and test set, quality control steps on both sets, p-value thresholding and generation of subdatasets, PRS calculation, machine learning model training, and finally, how to interpret and compare machine learning results with PRS.

Dataset Generation

The documentation associated with the article explains the dataset generation part in a detailed way **(See Supplementary material, Step 0 - Generate Data)**. Using hapgen2 [13], we generated 10000 controls for chromosome 21 and passed to phenotypesimulator (PS) [14] to generate the phenotype. The phenotype generated

by PS for each person is continuous, which we converted to binary phenotype by thresholding on 0. Figure 1 shows the overall process of generating simulated data used for benchmarking, and figure 2 shows the final files generated for the PRS calculation and machine learning.

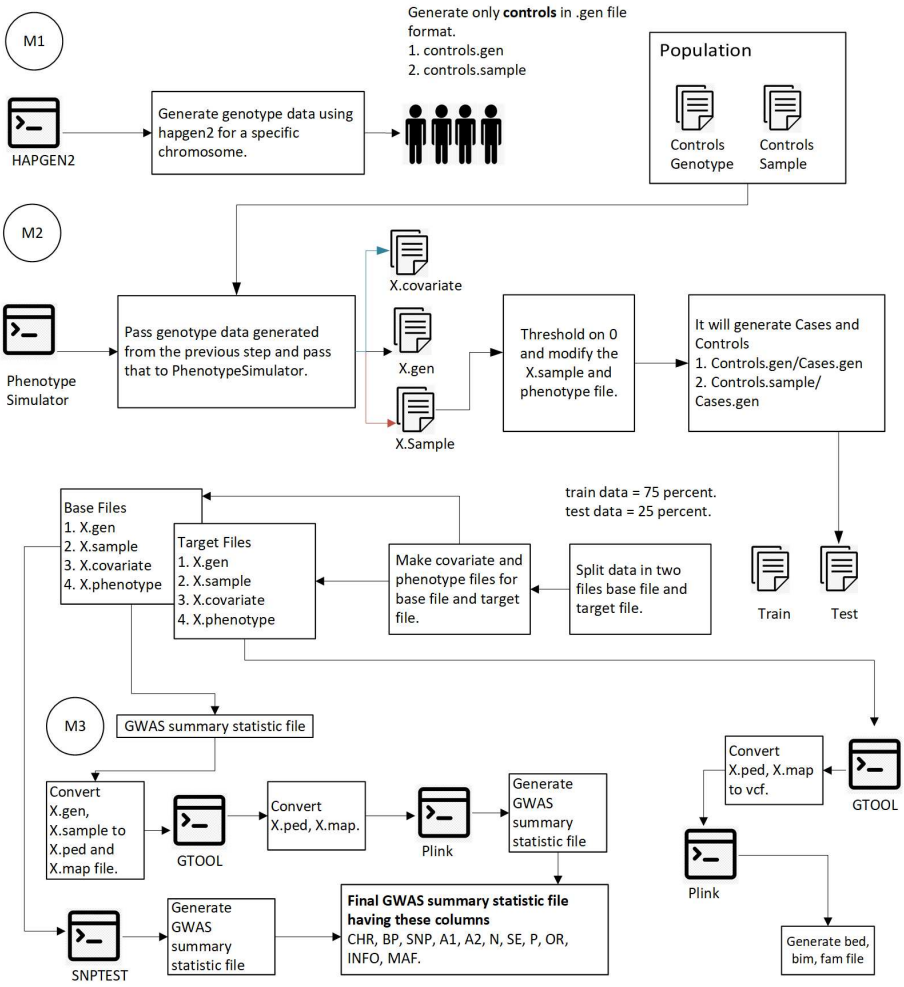

Figure 1: This diagram shows the process of dataset preparation. Generate data using Hapgen2, pass it to PhenotypeSimulator, convert continuous phenotype to binary phenotype, split data into training/test set, and calculate GWAS file.

Dataset Division

We followed this article [12] for PRS calculation [15], and for the PRS calculation, two files are required: Base file (the GWAS summary statistic file) and target file (Samples for which PRS is to be calculated). There are two ways to obtain the GWAS file: The first is to divide the data into training and test sets and use training samples to calculate GWAS, which is the only way to proceed with simulated data. The second is to use the GWAS summary statistic file available online at the GWAS catalog for a particular phenotype (like depression) for the real dataset. We used 10000 samples for the comparison, and the following is the division of the dataset.

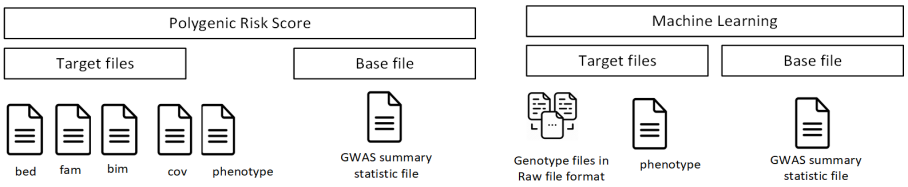

Figure 2: This diagram shows the files required for machine learning and PRS. Genotype files, phenotype, and GWAS for p-value threshold is required to train machine learning algorithm. Genotype files in encoded format (aa = 0, aA//Aa = 1, AA = 2), which can be generated using plink and the name of the format is .RAW format. For PRS, target data should be in BED, BIM, BAM format. The phenotype file should contain phenotype for the target data, and the COV file contains the covariate like sex and age. GWAS summary statistic file is passed to PRS tools, and these tools use ODD ratios to calculate the PRS and p-value column for SNPs selection.

It is important to note that genotype data and phenotype data are the same for both processes, but the format they should be processed is entirely different.

1 Training Data / Base data / Data use to generate the GWAS summary statistic file = 75 percent / 7500 samples
2 Test Data / Target data / Samples for which PRS is calculated = 25 percent / 2500 samples

PRS calculation is a well-established procedure, and it involves some quality control steps to ensure that the score is accurate enough to make any statement about the status of the disease in any patient.

Generate GWAS summary statistic file

The next step is to calculate the GWAS summary statistic file from the base data, which can be done using Plink and Snptest, and it contains specific columns which are shown on this link [15]. Columns are renamed based on the information they contain **(See Supplementary material, Step 1 - Divide Data into Base and Target sets)**, so PRS tools can easily process that file. Whereas for machine learning, only p-values are required to extract the SNPs above a specific threshold.

Quality control steps for base file/training data/GWAS summary statistic file and target file/test data

The base file can also be called training data (machine learning) or GWAS summary statistic file, and the target file can also be called test data (machine learning).

Quality controls steps are well explained in this article [12], and the corresponding online documentation. We discussed which quality control steps can be ignored for base and target simulated data. See figure 3 to understand the following discussion. QC (Quality control) 1 can be ignored for the simulated data as the simulation tool handles the genome build. QC 2 involves standard quality controls like minor allele frequency thresholding (In simulated data, it should not affect because data is simulated for all the SNPs) and linkage disequilibrium calculation. We ignored the

linkage disequilibrium step for machine learning because SNPs containing related information will be automatically considered as one feature. QC 3 to 5 can also be ignored because, in simulated data, there is no missing, duplicate, or ambiguous SNP. Ignore QC 6 because we simulated genotype data for chromosome 21 and do not have sex information. Although in the implementation, this part is included. QC 7 is necessary to avoid overfitting. QC 8 - 9 should always be considered before starting.

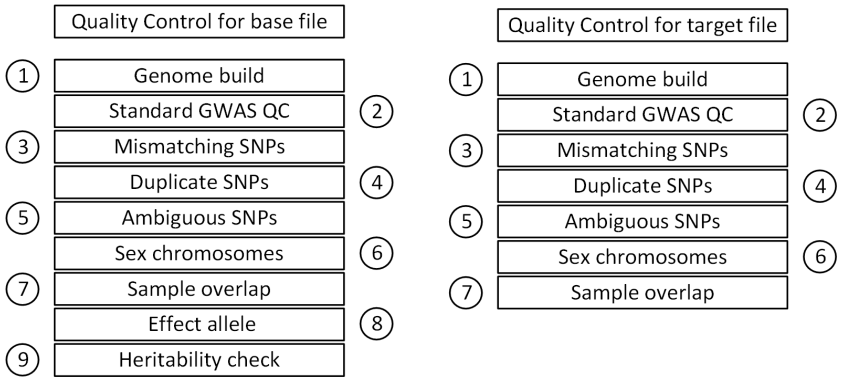

Figure 3: Quality control steps for Polygenic Risk Score calculation.

Covariates

For calculating PRS, sex, age, and principal components are the essential parameters along with genotype data. Now the question arises what if we want to include that information for machine learning. For ML, we generally do not need sex or age information. Sometimes for more accurate prediction, it is important to include that information. Such information must be included with the features passed to the machine learning algorithm. Suppose we pass feature set like this SNP1, SNP2, and SNP3, then after including age, sex, and PCA information, it will be like this SNP1, SNP2, SNP3, age, sex, PCA(1), PCA(2), and PCA(N).

Snps preselection

To reduce the number of SNPs, we used a p-value threshold. After splitting the data into train/test, we calculated the GWAS summary statistic file, which contains p-values for each SNP. Any number of p-value thresholds can be considered, and the sub-dataset will be stored in a separate folder.

PRS calculation

We noticed the following essential points when calculating PRS.

1 Various tools use different mechanisms and computation techniques for PRS calculation.
2 The input file format in which PRS tools accept input differs.
3 The output file in which PRS tools save the best PRS score can vary. Moreover, the data structure in which tools and scripts are used to store and process data can vary.

4 Some tools can automatically perform a few quality control steps like LDpred-2.

5 The PRS tools iteratively and automatically determine p-value thresholds. In contrast, in machine learning, we have to specify the threshold, and we recommend starting from thresholds yielding fewer SNPs and moving towards higher orders.

Machine learning

We used ANN (Artificial Neural Network) for classification (Architecture shown in table 1). There are two issues with the machine learning approach.

1 The first is that when using a machine learning model, iteratively increasing the SNPs can delay the computation process, but some heuristics can be used, such as arbitrary p-value thresholding can be explored, and the threshold which yields the best validation accuracy should be further investigated. PRS tools also rely on heuristics to find the optimal p-value, which can also be used to determine the optimal number of SNPs ultimately, leading to hard benchmarking.

2 In machine learning, overfitting, finding the optimal number of neurons, optimal number of layers, and hyper-parameter can cost a lot of time. Selecting models like lstm, 1DCNN, or ANN can yield different results. In such cases, multiple runs should be performed on the same dataset.

Sometimes, the dataset is an imbalance (unequal number of cases and controls), so rather than using accuracy, the area under the curve (AUC) should be considered to assess the model's performance. Table 2 compares the machine learning steps with PRS calculation.

ANN architecture used for the classification	
Layers	**Parameters**
Layer 1	**Input layer**
Layer 2	100 Neurons
Layer 3	Activation and dropout
Layer 4	50 Neurons
Layer 5	Activation and dropout
Layer 6	1 Neuron
Layer 7	**Sigmoid**

Table 1: Machine learning model used for classification. The last layer of the model should be sigmoid. Machine learning models are subjected to change depending on the model's performance on the validation set. Depending on the results, variations can be performed in the number of layers and the number of neurons in each layer.

Following is the reason to benchmark PRS vs. machine learning for the simulated data.

In some studies, it is necessary to mutate the parameters used to generate the simulated data to see the effects on the predictor's (machine learning or PRS) performance. For instance, we can study the AUC of machine learning algorithm and polygenic risk score for various heritability values, and genetic variations affect

Comparison of PRS and ML			
Step number	**Step name**	**Polygenic risk scores**	**Machine learning**
1	Files required	Base (GWAS) / Target (Test.bed, Test.bim, Test.fam)	Training (GWAS) and (Train.bed, Train.bim, Train.fam) / Test (Test.bed, Test.bim, Test.fam)
1.1	Covariates (includes gender, age)	Covarites included, but gender and age are missing	none
2	Quality controls steps on base file	yes	yes
3.1	Quality controls steps on Target file	yes	no
3.2	Linkage disequilibrium	yes	no
3.3	minor allele frequency	yes	no
3.4	missing/ duplicate SNPs	yes	no
4	P-value thresholding	Automatic	Manual/Heuristic
5	Algorithm	PRS tools (Individual Snps)	Machine learning model Non-linearity, but requires optimization in layers, neurons, and hyper-parameters
5.1	PCA	yes	no
6	Comparison	Normalized polygenic risk scores between 0 and 1.	Use sigmoid on model's last layer and normalize the probability between 0 and 1.

Table 2: This table shows which operations can be skipped for machine learning (Step number: 3.1,3.2,3.3,3.4). Secondly, it shows the alternative of PRS calculation steps for machine learning classification (Step number: 4,5,5.1,6)

can also be examined. There are some phenotypes for which heritability is high, and it is very low for some. There is a possibility that machine learning may generate high AUC for high heritability or yield low AUC. For such studies, it is necessary to generate multiple datasets with various parameters and execute ML vs. PRS pipeline to see the effect on ML AUC and PRS AUC. The proposed pipeline can easily work in such situations, and we also explained each process in detail and mentioned the steps where user input is required.

Results

This section elaborates on results interpretation. Results may change based on the dataset and explore the output of each tool for further investigation. Table 3 shows the results (Accuracy) of the machine learning model. Table 4 shows the AUC of PRSice, plink, lassosum, and machine learning. The final PRS score by tools is normalized between 0 and 1. The person having $PRS > 0.5$ is considered a case, and $PRS < 0.5$ is considered a control. After thresholding, the final results are compared

with the actual test data values (in machine learning) or target data (in PRS) to find AUC. We used the same process for machine learning. The classification probabilities were normalized between 0 and 1, and after the same thresholding, we compared it with the test data.

Figure 4 shows the distribution of PRS.

P-values	Average Training AUC	Average Test AUC	Number of SNPs
pv_5e-60	0.96	0.92	521
pv_5e-55	0.96	0.91	794
pv_5e-50	0.96	0.92	1184
pv_5e-45	0.96	0.91	1644
pv_5e-40	0.96	0.91	2319

Table 3: This table shows the machine learning model's average training and test accuracies for five iterations. The first column shows the p-value threshold, the second shows the training accuracy, the third shows the average test accuracy, and the fourth indicates the number of SNPs in each threshold.

Tools/ iterations	1	2	3	4	5
PRSice	0.28	0.29	0.27	0.27	0.27
Plink	0.27	0.3	0.31	0.32	0.30
Lassosum	0.36	0.34	0.35	0.33	0.38
Machine Learning	0.88	0.87	0.88	0.87	0.87

Table 4: This table shows the AUC (Area under the curve) for five iterations on the same dataset. In our case, machine learning outperformed PRS tools, but it is subjected to dataset or phenotype under consideration and other parameters like genetic variation.

Discussion and conclusion

PRS is a well-established practice of measuring a person's tendency to suffer from a particular disease, whereas machine learning algorithms classify people into cases/controls. Machine learning scores can be normalized to compare them with PRS by changing the representation of the results. Though we showed the procedure for simulated binary phenotypes, it can be extended to the real dataset and continuous phenotypes with a mutation in the final step and converting the problem to multi-label classification when training the machine learning model. We presented the pipeline in an automated way, and it can assist in various studies. We plan to analyze the effect of quality control steps on PRS calculation and machine learning classification in the future.

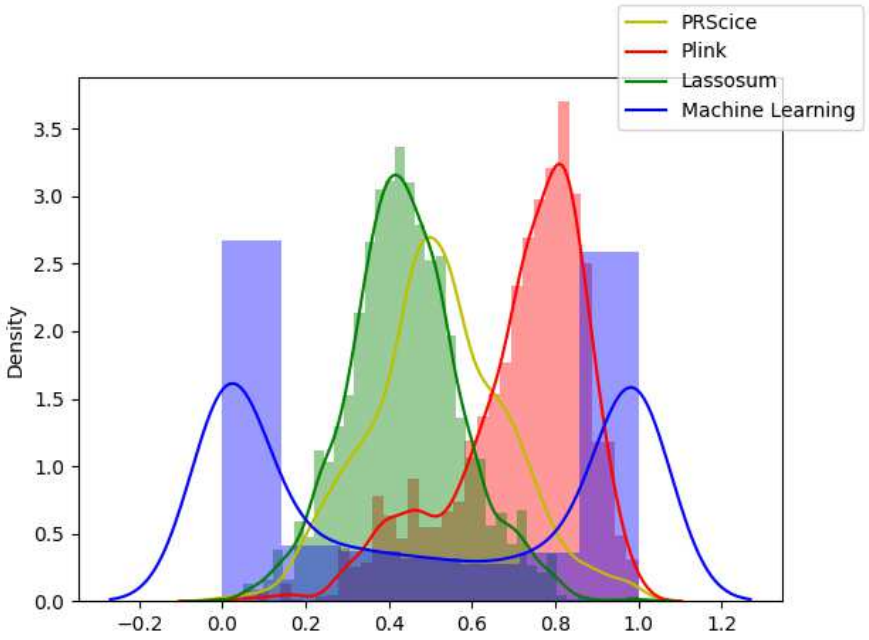

Figure 4: Distribution of risk scores using Plink, PRSice, Plink, and machine learning. Two purple lines show the real distribution of cases and controls. Yellow, Red, Green, and Blue lines show the distribution of normalized (between 0-1) PRS score generated using PRsice, Plink, lassosum, and machine learning. The distribution of machine learning resembles the actual distribution, and this figure is generated from the first iteration of classification.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Availability of data and materials
All data and code are publically available on this link https://1drv.ms/u/s!AIFVll05llt7gwXMJui4TFEuEIIt?e=a0voH8. Online documentation is available on this link shorturl.at/mtMN0.

Competing interests
The authors declare that they have no competing interests.

Funding
This publication is based upon work supported by the Khalifa University of Science and Technology under Award No. CIRA-2019-050 to SFF.

Author's contributions
M.M. and S.F. wrote the main manuscript text. All authors reviewed the manuscript.

Acknowledgements
This publication is based upon work supported by the Khalifa University of Science and Technology under Award No. CIRA-2019-050 to SFF.

Author details
[1]Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology, Al Saada St - Zone 1, Abu Dhabi, United Arab Emirates. [2]Department of Mathematics, Khalifa University of Science and Technology, Al Saada St - Zone 1, Abu Dhabi, United Arab Emirates. [3]Research And Data Intelligence Support Center R-DISC, Khalifa University of Science and Technology, Al Saada St - Zone 1, Abu Dhabi, United Arab Emirates.

References

1. Jiang, X., Holmes, C., McVean, G.: The impact of age on genetic risk for common diseases. PLOS Genetics **17**(8), 1009723 (2021). doi:10.1371/journal.pgen.1009723
2. Jackson, M., Marks, L., May, G.H.W., Wilson, J.B.: The genetic basis of disease. Essays in Biochemistry **62**(5), 643–723 (2018). doi:10.1042/ebc20170053
3. Igo, R.P., Kinzy, T.G., Bailey, J.N.C.: Genetic risk scores. Current Protocols in Human Genetics **104**(1) (2019). doi:10.1002/cphg.95
4. Schulz, C., Padmanabhan, S.: Methods to assess genetic risk prediction. In: Hypertension, pp. 27–40. Springer, ??? (2017). doi:10.1007/978-1-4939-6625-7_2. https://doi.org/10.1007/978-1-4939-6625-7_2
5. Lewis, A.C.F., Green, R.C.: Polygenic risk scores in the clinic: new perspectives needed on familiar ethical issues. Genome Medicine **13**(1) (2021). doi:10.1186/s13073-021-00829-7
6. Fritsche, L.G., Ma, Y., Zhang, D., Salvatore, M., Lee, S., Zhou, X., Mukherjee, B.: On cross-ancestry cancer polygenic risk scores. PLOS Genetics **17**(9), 1009670 (2021). doi:10.1371/journal.pgen.1009670
7. Homburger, J.R., Neben, C.L., Mishne, G., Zhou, A.Y., Kathiresan, S., Khera, A.V.: Low coverage whole genome sequencing enables accurate assessment of common variants and calculation of genome-wide polygenic scores (2019). doi:10.1101/716977
8. Muneeb, M., Henschel, A.: Eye-color and type-2 diabetes phenotype prediction from genotype data using deep learning methods. BMC Bioinformatics **22**(1) (2021). doi:10.1186/s12859-021-04077-9
9. Narayan, S.: The generalized sigmoid activation function: Competitive supervised learning. Information Sciences **99**(1-2), 69–82 (1997). doi:10.1016/s0020-0255(96)00200-9
10. Gola, D., Erdmann, J., Müller-Myhsok, B., Schunkert, H., König, I.R.: Polygenic risk scores outperform machine learning methods in predicting coronary artery disease status. Genetic Epidemiology **44**(2), 125–138 (2020). doi:10.1002/gepi.22279
11. Badré, A., Zhang, L., Muchero, W., Reynolds, J.C., Pan, C.: Deep neural network improves the estimation of polygenic risk scores for breast cancer. Journal of Human Genetics **66**(4), 359–369 (2020). doi:10.1038/s10038-020-00832-7
12. Choi, S.W., Mak, T.S.-H., O'Reilly, P.F.: Tutorial: a guide to performing polygenic risk score analyses. Nature Protocols **15**(9), 2759–2772 (2020). doi:10.1038/s41596-020-0353-1
13. Su, Z., Marchini, J., Donnelly, P.: HAPGEN2: simulation of multiple disease SNPs. Bioinformatics **27**(16), 2304–2305 (2011). doi:10.1093/bioinformatics/btr341
14. Meyer, H.V., Birney, E.: PhenotypeSimulator: A comprehensive framework for simulating multi-trait, multi-locus genotype to phenotype relationships. Bioinformatics **34**(17), 2951–2956 (2018). doi:10.1093/bioinformatics/bty197
15. 1. QC of Base Data - Basic Tutorial for Polygenic Risk Score Analyses. https://choishingwan.github.io/PRS-Tutorial/base/. (Accessed on 01/05/2022)

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInfo.pdf