**Improving Polygenic Risk Score Prediction by Phenotype-Agnostic Dimensionality Reduction**

Yarden Hochenberg1, Nadav Rappoport2

1Industrial Engineering and Management, Ben-Gurion University of the Negev, ISRAEL

2Software and Information Systems Engineering, Ben-Gurion University of the Negev, ISRAEL

**Abstract**

**Motivation:** Progress in sequencing technologies enables early detection of genetic diseases and the development of personalized medicine. The “missing heritability” problem means that the variance explained by genetic variation is typically small compared to family-based heritability estimates. As such, it is highly desirable to develop improved models for polygenic risk score (PRS) prediction.

**Results:** In this study, we propose a new approach to computing PRS that enhances the heritability variance explanation of complex traits. Our method is not limited to an additive model and can incorporate high-dimensional genomic interaction. We use a two-stage approach. The first stage is phenotype-agnostic, an unsupervised approach to dimensionality reduction. The second stage is training a prediction model using a supervised machine learning algorithm. Our approach enables PRS to be computed without the need for variable selection techniques, while maintaining a computationally feasible model. Moreover, the first stage, which is computationally resource intensive, is independent of phenotype—that is, its output can be used as input for a prediction model for any chosen trait or disease and therefore need only be trained once. We evaluated the approach using two dimensionality reduction models, deep autoencoder and principal component analysis, and two phenotype prediction models, deep neural network and extreme gradient boosting. The models were trained using the UK Biobank dataset with over 340,000 subjects and 460,000 SNPs. Moreover, we evaluated the approach on two phenotypes, height and hypertension, and compared the results to the PRS baseline model. Our model results outperform the base model results for both phenotypes.

**Availability and Implementation:** The data underlying this article were provided by the UK Biobank under license. Data will be shared on request via application to the UK Biobank. The trained dimensionality reduction models are now available at: <https://github.com/nadavlab/genotyping_dimensionality_reduction>

**Contact:** nadavrap@bgu.ac.il

**Supplementary Information:** [Supplementary data](https://oup.silverchair-cdn.com/oup/backfile/Content_public/Journal/bioinformatics/38/18/10.1093_bioinformatics_btac509/1/btac509_supplementary_data.zip?Expires=1666304004&Signature=kaWls9~M7-tR~XNhc6YICdilYArrxJx1TAFz2rowZ5~cZfaw~3aEc44idxqRodIZJ7BgA0Yevihnzwo6Rmc0H17oXdRDI6dXuCKEOElhU1p0Uf5sSa5ZvD~Nq-qqkR0FVriazbn2XaaHzsfej7GL9P3CJlFpzPb1CWyqzPSVE7lDaebJCtLWpcQ7lRp9zlbpI-wIF9nHvZXMaXuMauPyoFbmGw5UdWASB4DuyJbFA6ba6OrUzq7ht~lQkooMSLvBwrpcZ3YjffjYMrcrImV3RuqyocU8D8JSHCEIE55vTwwURfra~9HLodlrlGi1tIqtlo0F1Gc32d-XPkq9GJgCMQ__&Key-Pair-Id=APKAIE5G5CRDK6RD3PGA) are available in the [Supplementary\_Data](https://oup.silverchair-cdn.com/oup/backfile/Content_public/Journal/bioinformatics/38/18/10.1093_bioinformatics_btac509/1/btac509_supplementary_data.zip?Expires=1666304004&Signature=kaWls9~M7-tR~XNhc6YICdilYArrxJx1TAFz2rowZ5~cZfaw~3aEc44idxqRodIZJ7BgA0Yevihnzwo6Rmc0H17oXdRDI6dXuCKEOElhU1p0Uf5sSa5ZvD~Nq-qqkR0FVriazbn2XaaHzsfej7GL9P3CJlFpzPb1CWyqzPSVE7lDaebJCtLWpcQ7lRp9zlbpI-wIF9nHvZXMaXuMauPyoFbmGw5UdWASB4DuyJbFA6ba6OrUzq7ht~lQkooMSLvBwrpcZ3YjffjYMrcrImV3RuqyocU8D8JSHCEIE55vTwwURfra~9HLodlrlGi1tIqtlo0F1Gc32d-XPkq9GJgCMQ__&Key-Pair-Id=APKAIE5G5CRDK6RD3PGA).docx file.

**Keywords:** UK Biobank; polygenic risk score; machine learning; dimensionality reduction; population genetics

**Introduction**

In recent years, much has been written about the challenges in discovering the full heritability variance of a trait (Maher, 2008; Manolio *et al.*, 2009). Genome‐wide association studies (GWAS) (Collins *et al.*, 1998) attempt to identify genetic loci associated with a complex trait by ranking the loci by the statistical significance (*p*-values) of association with the trait. The most widely tested markers in GWAS are single nucleotide polymorphisms (SNPs), which are genetic variations of a single base pair that show natural variation in the population (Collins *et al.*, 1998; Lewis and Knight, 2012).

The identification of top-ranking SNPs with trait association has led to new types of models that predict polygenic risk scores (PRS). The PRS method aggregates information from SNPs across the genome, by the weighted sum of the trait-associated alleles, into a single score that can be used as a trait value or disease risk prediction per subject (Euesden *et al.*, 2015; Torkamani *et al.*, 2018; Yanes *et al.*, 2020; Collister *et al.*, 2022). The growth and progress in sequencing technologies in recent years has enabled the early detection of genetic diseases and the development of personalized medicine strategies (Chatterjee *et al.*, 2016; Torkamani *et al.*, 2018).

However, it was found that the genetic variation explained by trait-associated SNPs does not match the expectation from previous family studies. That is, only a small fraction of the genetic risk can be explained by those identified SNPs (van der Sluis *et al.*, 2010). For example, for the height phenotype, more than 80% of the variation within a population is attributable to additive genetic factors, but previous studies were able to find only a small fraction of this variation (Visscher *et al.*, 2006). This problem is called “missing heritability” and has been studied in recent years (Maher, 2008; Manolio *et al.*, 2009).

First, it has been shown that some factors can substantially influence the power of PRS prediction. Association testing’s power is directly affected by the size of the dataset used as a basis for identifying statistically significant SNPs, the effect size weighting methods, and the selection of the *p-*value threshold for inclusion in the PRS calculation (Dudbridge, 2013; Chatterjee *et al.*, 2013).

Second, the main limitation of standard PRS is that it assumes independence between SNPs—that is, trait-associated SNPs are mostly discovered using single-locus analysis, where variants are evaluated individually for association with phenotypes. In this type of analysis, factors with interaction effects but no marginal effects will not get included in the PRS calculation (Culverhouse *et al.*, 2002; Moore, 2003). Thus, PRS will exclude a single locus with miscorrelation to the phenotype. While in recent years much has been discussed about an increase in the heritability variance of phenotypes when combining them within gene–gene interaction techniques (Culverhouse *et al.*, 2002; Moore, 2003; Lehner, 2007; Cordell, 2009; McKinney and Pajewski, 2012).

There is a significant challenge when considering gene–gene interaction. In the phase of detecting factors that display interaction effects, a major challenge is the limited sample size compared to the huge number of genetic loci. For complex diseases, we might also expect not only two-locus interactions but higher-level interactions (multi-locus genotype combinations). This will lead to additional challenges, such as the huge number of parameters of the models and the consequent extremely large datasets required to accurately estimate these parameters (Cordell, 2009).

Regression-based methods are widespread data mining methods for detecting gene–gene interactions (Cordell, 2009) and for the prediction of complex traits (Makowsky *et al.*, 2011; Dudbridge, 2013; Khera *et al.*, 2018; Lello *et al.*, 2019). For example, the Plink tool (Purcell *et al.*, 2007), which provides an option to assume an allelic model for both the main effects and the interactions, uses a regression model. Regression-based methods have been criticized due to their inability to deal with a high-dimensional dataset that may contain multi-locus genotype combinations, nonlinear problems, and sparse data, and they have therefore been replaced by more sophisticated machine learning models (Moore and Williams, 2002; McKinney *et al.*, 2006; Cordell, 2009).

Artificial neural networks (ANNs) have become practical data mining models in the study of associations between genomic data and complex phenotypes because of their ability to learn linear as well as nonlinear phenotype–genotype relationships. Moreover, these models can also take into account gene–gene interactions in addition to the main effects (McKinney *et al.*, 2006; González-Recio *et al.*, 2014). Not only do ANN models not rely on most of the prior assumptions that underlie parametric models, but they can also capture complex signals from the data and deliver superior predictive accuracy (Ehret *et al.*, 2015).

As large genomic datasets accumulate, it is now possible to train even deeper and more complex models to gain insights and perform difficult genomic tasks (Danilevsky and Shomron, 2021). However, due to ANNs’ lack of interpretability, one of their limitations is that such models are usually no use for inferring the effects of SNPs on phenotypes. In high-dimensional genomic data, ANNs typically have more parameters (weights) than samples, and they may be too computationally demanding when the number of neurons is large (Ehret *et al.*, 2015). Moreover, one of their limitations is that they require a sample size larger than the number of features and may even require a sample size of the order of the number of features squared (Hua *et al.*, 2005; Figueroa *et al.*, 2012).

Therefore, many studies have used variable selection techniques in which only subsets of SNPs are used as predictors (Ehret *et al.*, 2015; Bellot *et al.*, 2018; Peng *et al.*, 2021), while other studies have demonstrated that traits are influenced by both common and rare SNPs with small effects (Gibson, 2012). Such feature selection is performed using prior knowledge, for example selecting only SNPs in proximity to genes that are suspected to be related or associated with the phenotype of interest. Another approach to SNP subsetting is to select only SNPs with a direct effect on the phenotype (Thomas *et al.*, 2020).

In the field of genetics, an autoencoder, which is a type of [ANN](https://en.wikipedia.org/wiki/Artificial_neural_network), can be used to reduce the gene space by combining multi-locus genotypes to give a smaller number of variables (Xie *et al.*, 2017). Part of the ANN’s advantages is that it is a nonparametric model—i.e., no hypothesis about the value of a statistical parameter is made—and it is free of any assumed genetic model.

Another family of models that are not based on neural network architecture and that may capture nonlinear interactions between loci are random forest (RF), gradient boosted trees (GBT), and extreme gradient boosting (XGBoost). These algorithms have been used in the genetic analysis of complex diseases because of their capability to combine different predictors sequentially, i.e., taking into account interaction structures in the data (Goldstein *et al.*, 2010; Chen and Ishwaran, 2012; Paré *et al.*, 2017; Romagnoni *et al.*, 2019; Thomas *et al.*, 2020; Bracher-Smith *et al.*, 2022). The advantage of boosting algorithms is that they can mitigate the drawbacks associated with large datasets. Previous genome-wide studies have shown that such models can perform better than linear methods and deliver superior predictive accuracy (González-Recio and Forni, 2011; González-Recio *et al.*, 2013).

Boosting models iteratively learn from residual estimates from previous estimators. As the number of iterations increases, the selected SNP should contribute less at each iteration. Moreover, this model performs variable selection, as some SNPs may not be selected at all (González-Recio *et al.*, 2014). In high-dimensional data such as genomic data, some suggested methods subset features even before applying machine learning algorithms (Romagnoni *et al.*, 2019). However, previous works have shown that a model using the entire genome has better predictive performance than a model that selects features first. The reason for the lower performance of the latter models is that when SNPs are excluded from the model, a substantial amount of information may be lost, and that cannot be compensated for (Thomas *et al.*, 2020).

**Materials and Methods**

**Phenotype and Genotype Data**

In this study, we used the UK Biobank data (Allen *et al.*, 2014) with 487,409 samples and 93,095,623 SNPs from the imputed dataset. In addition, clinical and physiologic phenotypes were used, including height and hypertension, and socio-demographic covariates, including age (Data-Field 21003), ethnic group (Data-Field 22006), sex (Data-Field 22001), genotype measurement batch (Data-Field 22000), and the center where the patient’s examination was performed (Data-Field 54). In contrast to other UK Biobank PRS studies, we did not use the 40 PCA variables (Data-Field 22009) that were calculated by the UK Biobank based on the whole dataset, due to information leakage concerns.

**Preprocessing**

Quality control was implemented on participants and SNPs using the Plink2 tool (Purcell *et al.*, 2007). SNP quality control was performed by eliminating variants with duplicate IDs, those with missing values (keeping only genotyped SNPs with a dosage of 0, 1, or 2), variants that deviated from Hardy–Weinberg equilibrium (HWE) by *p*-value < 1e-6, and variants with minor allele frequency (MAF) < 0.001. SNP quality control operations were carried out in the order of mention, according to the default Plink2 order. Subject quality control was performed by eliminating family-related subjects as described in Sheppard *et al.* (2021). In addition, we used only subjects of European Caucasian ancestry to prevent population stratification (Lewis and Knight, 2012).

SNPs were coded as 0,1,2 for homozygote for the minor allele, heterozygote, and homozygote for the alternative allele, respectively, assuming additive allele effects. The samples were randomly split into 85% training set and 15% test set. From the training set, 1,000 random samples were taken to serve as the validation set. We chose a relatively low validation set size because we wanted to maximize the number of samples on which the model could be trained. Such a large dataset in high dimension could not be loaded into memory at once. Therefore, we decided to split the data into chunks of 1,000 samples each. We used iterative algorithms such that in each iteration the model was updated based on a batch of 1,000 samples.

**Phenotypes**

To examine the quality of the complex trait prediction models, we used two phenotypes, one quantitative and one binary trait. Height is a continuous trait, highly heritable, and a classic polygenic trait (Lango Allen *et al.*, 2010). Hypertension was used as a binary trait, as it is a common disease that has a complex multifactorial etiology (Moore and Williams, 2002).

The new variables after dimensionality reduction, which were used as input for the trait prediction model, were min-max normalized to be between zero and one. The trait prediction model was adjusted using the covariates listed above. We used the socio-demographic categorical features of sex, genotype measurement batch, and the patient’s assessment center as dummy variables. Age was min-max normalized to be between zero and one. In addition, for each model, we also filtered samples that had a missing value for the phenotype of interest.

**Dimensionality Reduction**

For the dimensionality reduction process, we trained separate dimensionality reduction models for each of the 22 chromosomes (without the sex chromosomes) due to the existing resource limit. We used two models: deep auotoencoder (autoencoder) (Rumelhart *et al.*, 1985) and principal component analysis (PCA) (Hotelling, 1933). Other dimensionality reduction alternative methods exist, such as truncated singular value decomposition (SVD) (Golub and Reinsch, 1971) and Lsomap (Tenenbaum *et al.*, 2000), and could easily be substituted. However, we chose the two mentioned as they are broadly used and can be trained with an iterative process using batch processing. We chose to reduce the dimension of the SNPs to 10% of the input SNP number in each chromosome, a percentage that was adjusted to the computational power available and allows the later union of the variables from all the chromosomes for the prediction model. For the autoencoder, this was achieved by setting the number of neurons in the encoding layer to 10% of the input SNP number. For PCA, we used the first 10% of the principal components.

The dimensionality reduction phase is independent of phenotype, i.e., training does not require labels, and its output can be used as an input for any type of prediction model and any chosen trait or disease. The dimensionality reduction models were evaluated using the validation set according to the coefficient of determination metric (see evaluation metrics section). The training run time of the dimensionality reduction phase is shown in Figure 1.



**Figure 1. Training run times of the dimension reduction models (in minutes).** For the autoencoder model, the times shown are the average running time per 100 epochs for the 600 epochs that the model was trained.

We trained a fully connected autoencoder model with the TensorFlow Python package (Abadi *et al.*, 2016). Each chromosome’s autoencoder was designed as an encoder with two hidden layers, with the second layer being the feature compression layer (bottleneck), and a decoder with one hidden layer. We chose to add dropout layers to prevent overfitting and because this technique is known for its ability to efficiently combine many different neural network architectures (Srivastava *et al.*, 2014). Each hidden layer had a dropout layer with a rate of 0.1, except for the bottleneck. The number of neurons in each hidden layer was 20%, 10%, and 20% of the input SNP number, respectively. We performed hyperparameter tuning using the validation set for the activation functions and for the learning rate, where the best configuration was chosen according to the RMSE metric. The tuning was done on chromosome 22, assuming that the training on the other chromosomes would behave similarly and due to the long run times. We trained our model using the Adam optimizer (Kingma and Ba, 2014) and MSE loss function, with a batch size of 250 over 600 epochs. The chosen activation functions were PReLU (He *et al.*, 2015) for the hidden layers and linear for the output layer, and the best learning rate was 1e-5.

The second method for dimensionality reduction we used was PCA, implemented by the Scikit-learn package in Python (Pedregosa *et al.*, 2011). We projected the SNP data of each individual in each chromosome using the first 10% of the principal components.

**Phenotype Prediction**

We concatenated the new representations of each chromosome and adjusted them to the covariates matrix. The combined dataset was used as input for the phenotype prediction model. We used two models for prediction: deep neural network (DNN) (Werbos, 1982) and XGBoost (XGB) (Chen and He, 2015; Chen and Guestrin, 2016), since they can both be trained using an iterative process and following the advantages discussed in the introduction section. We trained the two prediction models for each phenotype and input variables driven by different dimensionality reduction models separately (see Figure 2). Each model had its hyperparameters tuned, where the best configuration (Table 1) was chosen using the validation set according to the RMSE metric for height and the log-loss metric for hypertension. Then the models were evaluated on the test set by different evaluation metrics. The training run times of the different models are shown in Table 2 (see resources section for information about the resources consumed).

**Table 1.** **Best configurations.** For each model, the chosen configuration was based on hyperparameter tuning. The best configuration was chosen using the validation set according to the RMSE metric for height and the log-loss metric for hypertension.

|  |  |  |  |
| --- | --- | --- | --- |
| Phenotype | Configuration | Hyperparameter tuned | Chosen hyperparameter value |
| Height | Autoencoder + DNN | DNN hidden layers’ activation function | ReLU |
| PCA + DNN | DNN hidden layers’ activation function | PReLU |
| Autoencoder + XGB | Tree’s max depth | 5 |
| PCA + XGB | Tree’s max depth | 5 |
| Hypertension | Autoencoder + DNN | DNN hidden layers’ activation function | PReLU |
| PCA + DNN | DNN hidden layers’ activation function | PReLU |
| Autoencoder + XGB | Tree’s max depth | 2 |
| PCA + XGB | Tree’s max depth | 2 |

**Table 2.** **Training run times of the trait prediction models in minutes.**

|  |  |  |
| --- | --- | --- |
| **PCA** | **Autoencoder** | **Dimensionality reduction model** |
| **XGB** | **DNN** | **XGB** | **DNN** | **Prediction model** |
| 1958 | 1924 | 1776 | 1892 | Height |
| 537 | 2474 | 692 | 2021 | Hypertension |



**Figure 2.** **Schematic overview of the study framework.** The dimensionality reduction phase is independent of the phenotype, i.e., its output can be used as an input for any type of prediction model and any trait or disease. Therefore, it need be trained only once. Then, we trained and compared multiple pipelines for height and hypertension phenotype prediction. Note that the dimensionality reduction phase was trained for each chromosome separately. Then, the variables from all chromosomes were merged, adjusted to the covariate matrix, and used as input for the prediction models.

We trained a fully connected DNN model with TensorFlow (Abadi *et al.*, 2016). The network was designed with two hidden layers, and each hidden layer had a dropout layer with a rate of 0.1. The number of neurons in each hidden layer was 20% and 10% of the number of input variables, respectively. We tuned the hidden layers’ activation function for each model (Table 1), and we chose a linear output layer activation function for height and a sigmoid for hypertension. We trained our model using the Adam optimizer (Kingma and Ba, 2014) with a batch size of 50 over 80 epochs. The learning rate used was 1e-7, and the loss function was MSE for height and binary-cross-entropy for hypertension.

For the second prediction model, we trained an XGB model with the XGBoost Python package. XGB can be used for regression (for height) as well as for classification (for hypertension). We trained the model in batch processing using the “xgb\_model” parameter, which allows training continuation. For each batch of 1,000 samples, the model built one gradient boosted tree, using a 0.2 subsample ratio of columns and a 0.01 minimum loss reduction required to make a further partition on a leaf node. We tuned the maximal depth of the tree for each model separately (Table 1), and we trained our model for 20 epochs for hypertension and 40 epochs for height prediction until convergence. The learning rate used was 1e-3, and the evaluation metric was RMSE for height and binary-cross-entropy for hypertension.

**The Base Model, Phenotype Prediction using PRSice-2**

The base model was executed for each phenotype separately on the original SNPs before the dimensionality reduction process (see Figure 2). First, we performed GWAS (Collins *et al.*, 1998; Marees *et al.*, 2018) using the Plink2 tool (Purcell *et al.*, 2007) on the training set. In this method, trait-associated SNPs are discovered using single-locus analysis, where each variant, adjusted to the covariate, is evaluated individually for association with a phenotype.

Then, we performed PRS analyses (Choi *et al.*, 2020; Collister *et al.*, 2022) using the PRSice-2 software (Choi and O’Reilly, 2019). In the classic PRS calculation method, only SNPs with a GWAS association *p*-value below a certain threshold are included in the calculation. The PRSice-2 software searches for a *p*-value threshold that generates the PRS best-fit that maximizes the phenotypic variation. To compute PRS, we used the training set and omitted SNPs that had constant dosage. In addition, we used the *p*-value and the coefficients obtained from the GWAS results for each SNP. Consider SNPs with a *p*-value smaller than the tested threshold; , a list of the number of effective alleles observed for each SNP, assuming an additive model; and , a list of the regression coefficients for height and log-odds ratio for hypertension. The PRS is then computed as follows:

Then, to adjust the PRS to the covariate matrix, we trained a linear regression model for height and a logistic regression model for hypertension using the Scikit-learn package in Python (Pedregosa *et al.*, 2011). The predictors were the PRS obtained from the PRSice-2 software and the covariate matrix. The model was evaluated on the test set by the same metrics mentioned above and compared to the new approach we suggest.

**Evaluation Metrics**

For evaluating and comparing the conserved information of dimensionality reduction models, we compared the correct SNP values and the reconstructed SNP values obtained from the decoder models. We did this using the coefficient of determination (R2) metric, calculated using the weighted variance aggregation, implemented by the Scikit-learn package in Python (Pedregosa *et al.*, 2011).

For evaluating and comparing the different algorithms for complex trait prediction, we chose the metrics according to the type of phenotype. For height, a quantitative phenotype, we used R2 and root mean square error (RMSE). Consider samples, a list of correct values , and a list of predicted values obtained from the estimated model ; the RMSE is calculated as follows:

R2 is calculated as follows:

For hypertension, a binary phenotype with unequal distribution between the two classes, we used the Average Precision metric. Since the ratio between the number of samples in the different classes was 23–77%, and to compare the results on a range of metrics, we also used the Area Under the Receiver Operating Characteristics (ROC AUC) and Cross Entropy Loss (log-loss) metrics. Consider samples, a list of estimated probabilities obtained from the estimation model , and a list of correct values ; the log-loss is calculated as follows:

Considering optional threshold, the Average Precision is calculated as follows:

where Precision is the proportion of true positive samples in the total predicted positive observations, and Recall is the portion of true positive samples in the total predicted negative observations.

**Resources**

The training of the prediction models was conducted on a high-performance computing cluster. For the DNN prediction models, we used an Intel Xeon Silver 4214 CPU @ 2.20 GHz p processor and Nvidia’s RTX-2080 GPU. For each DNN model we allocated a job with 6x hyper-threading and 80 GB RAM. For the XGB prediction models, we used an AMD EPYC 7702P 64-Core CPU. For each XGB model, we allocated a job using 4x hyper-threading and 15 GB RAM.

**Results**

**Preprocessing Results**

After quality control, 341,985 samples and 467,429 SNPs were used in the analyses (see supplementary table S2). The hypertension phenotype comprised 266,001 controls and 75,984 cases. The training set included 289,688 samples, the test set 51,297 samples, and the validation set 1,000 samples. In addition, for the height outcome prediction 745 samples were filtered out due to phenotype absence. For hypertension outcome prediction, no additional samples had a missing phenotype.

**Dimensionality Reduction Results**

The dimensionality reduction results were evaluated using the validation set, for each chromosome and each dimensionality reduction model. The weighted average R2 for the autoencoder achieved 0.5745, and for the PCA achieved 0.6642, where each chromosome’s R2 was weighted according to the number of SNPs in it (see supplementary table S1). The autoencoder’s R2 yielded lower results overall in all chromosomes than the PCA’s results. As shown in Figure 1, the average time for training the autoencoder for 100 epochs was much longer than the total time needed for training the PCA. Figure 3 shows the weighted average R2 value vs. the number of epochs for the autoencoder model. We trained the autoencoder for 600 epochs until it appeared to reach a plateau.



**Figure 3.** **The weighted average R2 value dependent on the number of epochs for the autoencoder model.**

**Predictive Ability Compared to the Base Model across Different Traits**

The number of features in the prediction model input matrix was 42,870, including 42,742 variables from the dimension reduction process unioned from all 22 chromosomes, and 128 variables from the covariate matrix. The covariate matrix included age, sex, 105 dummy variables of the genotype measurement batch, and 21 dummy variables of the assessment center. The performance of our new approach was then compared with that of the base model. The optimal *p*-value threshold, using the PRSice-2 software, was 1, both for height and hypertension, meaning all the SNPs were taken into account in computing the PRS. Table 3 details the results on the test set for height phenotype prediction, and Table 4 for hypertension phenotype prediction.

**Table3.** **Prediction results on height phenotype using the test set.** The RMSE metric is presented in centimeters.

|  |  |  |  |
| --- | --- | --- | --- |
| **---** | **PCA** | **Autoencoder** | **Dimensionality Reduction model** |
| **Base model** | **XGB** | **DNN** | **XGB** | **DNN** | **Prediction model** |
| 6.031 | 6.436 | **5.785** | 6.435 | 5.914 | RMSE |
| 0.574 | 0.515 | **0.608** | 0.515 | 0.591 | R2 |

**Table4.** **Prediction results on hypertension phenotype using the test set.**

|  |  |  |  |
| --- | --- | --- | --- |
| **---** | **PCA** | **Autoencoder** | **Dimensionality Reduction model** |
| **Base model** | **XGB** | **DNN** | **XGB** | **DNN** | **Prediction model** |
| 0.566 | 0.505 | **0.490** | 0.505 | 0.492 | Log-loss |
| 0.675 | 0.681 | **0.705** | 0.681 | 0.700 | ROC AUC |
| 0.360 | 0.347 | **0.394** | 0.345 | 0.384 | Average Precision |

When examining the prediction performance of the different models, we can observe that our model results outperform the base model results both in height and hypertension phenotypes.

For the height phenotype, the deep learning classifier trained on the output from the dimensionality reduction process achieved a best R2 score of 0.608 after the PCA process and 0.591 after the autoencoder process. For these models, the RMSE for height prediction was 5.785–5.914 cm. The XGB model shows a worse R2 score than the base model. While in the DNN model, the model that included the variables after PCA produced a better prediction than after the autoencoder, in XGB, the results are much the same for both types of dimensionality reduction models. Thus, our model result shows that although the weighted average R2 for the autoencoder was significantly lower than for PCA, the prediction results of the former can reach a compromise with the latter.

For the hypertension phenotype, both DNN and XGB prediction results surpassed those of the base model results according to ROC AUC and log-loss metrics. Based on the Average Precision metric, the base model outperforms the XGB model but not the DNN model. The deep learning classifier trained on the variables driven by the dimensionality reduction process achieved best ROC AUC scores of 0.705 after the PCA process and 0.7 after the autoencoder process. The PCA and the autoencoder performed quite similarly according to ROC AUC and log-loss metrics, while there was a slight gap of 0.01 in favor of the PCA in the Average Precision metric. Overall, the DNN classifier surpassed the results of both the base model and the XGB model, as seen in the receiver operating characteristic curve (ROC) and the precision–recall curve (Figure 4).



**Figure 4.** **Plots for hypertension phenotype**. **a.** Receiver operating characteristic (ROC) curve. **b.** Precision–recall curve.

In order to examine how trait values vary with increasing predicted phenotype target and to measure the elevated disease risk among individuals with the highest predicted values, we visualized the results using a quantile plot (Figures 5 & 6). The predicted phenotype was first split into 20 quantiles of increasing values, i.e., 20 equally sized quantiles, each comprising 5% of the predicted values’ sample distribution. Then, the quantiles as dummies were adjusted to the covariate matrix and were used as a predictor of the phenotype in linear or logistic regression models. The y-axis takes values of the coefficients for height prediction and odds ratio for hypertension prediction.

For the height phenotype, the quantile plots (Figure 5) reflect an S-shape, which means that the trait values are more spread out between quantiles at the tails, as expected in a normally distributed target trait. The trend in coefficients is significantly steeper in the DNN models than in the XGB and the base models. That is, the variation in the coefficients for the different quantiles in the DNN models has a wider range, and therefore we can assume that there is a stronger correlation between the quantile and the phenotype.

For the hypertension phenotype, the quantile plots (Figure 6) of the DNN and the base model reflect a disease risk that increases sharply in the right tail of the distribution. The plots are asymmetrical, with a marked inflection at the upper end, as expected when the cases are enriched in the higher quantiles. However, in the XGB prediction model, the plots are more symmetrical, which shows a lack of difference in disease risk between the different quantiles.



**Figure 5.** **Quantile plot for height phenotype.** The y-axis takes values of the coefficients of the quantiles. The error bars represent a 95% confidence interval.



**Figure 6. Quantile plot for hypertension phenotype.** The y-axis represents the odds ratio for the phenotype. The error bars represent a 95% confidence interval.

**Discussion**

We propose an improved approach for predicting individual trait value or disease risk using all available SNPs—an alternative to feature selection taking into account SNP interactions as predictors. The new approach can be used in the development of personalized medicine strategies and for improving healthcare. Moreover, this new approach can be generalized to any clinical and physiologic phenotypes without the need to train the dimensionality reduction part more than once.

We showed that our method can predict an individual’s trait value or disease risk better than the PRS base model. We have found that dimensionality reduction using PCA, in addition to being a faster procedure and although it only learns linear patterns, can serve as a good basis for phenotype prediction. The autoencoder process approach sometimes equals PCA, but mostly shows slightly worse results. However, we wish to underline the potential of the autoencoder, which by design can learn and represent nonlinear relationships. In the dataset we have used, the dimensionality shrinkage of the SNPs using a linear method yielded the best results, but perhaps an autoencoder may prove more useful with a larger number of samples. For the trait prediction model, our results showed that deep learning methods are capable of predicting phenotypes with higher performance than boosting algorithms like XGBoost.

We acknowledge some limitations of the current work. First, the model was trained and evaluated using individuals of European descent only; hence, adjustments to the model may need to be made for other ancestral groups. Second, due to limitations in the computational resources available to us, our model was estimated using one validation set. We assume that an estimation of the model using *k*-fold cross-validation may lead to more stable, accurate, and comprehensive results. Third, we trained and evaluated our method using data from the UK Biobank; a better evaluation would have used an external independent dataset for validation.

**Acknowledgments**

This research has been conducted using the UK Biobank resource under Application Number 56774. We would like to thank Dr. Jonathan Rosenblatt for the fruitful discussion along the way.

**References**

Abadi, M. *et al.* (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv Prepr. arXiv1603.04467*.

Allen, N.E. *et al.* (2014) UK biobank data: Come and get it. *Sci. Transl. Med.*, **6**, 3–5.

Bellot, P. *et al.* (2018) Can deep learning improve genomic prediction of complex human traits? *Genetics*, **210**, 809–819.

Bracher-Smith, M. *et al.* (2022) Machine learning for prediction of schizophrenia using genetic and demographic factors in the UK biobank. *Schizophr. Res.*, **246**, 156–164.

Chatterjee, N. *et al.* (2016) Developing and evaluating polygenic risk prediction models for stratified disease prevention. *Nat. Rev. Genet.*, **17**, 392–406.

Chatterjee, N. *et al.* (2013) Projecting the performance of risk prediction based on polygenic analyses of genome-wide association studies. *Nat. Genet.*, **45**, 400–405.

Chen, T. and Guestrin, C. (2016) Xgboost: A scalable tree boosting system. In, *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*., pp. 785–794.

Chen, T. and He, T. (2015) Higgs boson discovery with boosted trees. In, *NIPS 2014 workshop on high-energy physics and machine learning*. PMLR, pp. 69–80.

Chen, X. and Ishwaran, H. (2012) Random forests for genomic data analysis. *Genomics*, **99**, 323–329.

Choi, S.W. *et al.* (2020) Tutorial: A guide to performing polygenic risk score analyses. *Nat. Protoc.*, **15**, 2759–2772.

Choi, S.W. and O’Reilly, P.F. (2019) PRSice-2: Polygenic Risk Score software for biobank-scale data. *Gigascience*, **8**, giz082.

Collins, F.S. *et al.* (1998) A DNA polymorphism discovery resource for research on human genetic variation. *Genome Res.*, **8**, 1229–1231.

Collister, J.A. *et al.* (2022) Calculating Polygenic Risk Scores (PRS) in UK Biobank: A Practical Guide for Epidemiologists. *Front. Genet.*, **13**.

Cordell, H.J. (2009) Detecting gene–gene interactions that underlie human diseases. *Nat. Rev. Genet.*, **10**, 392–404.

Culverhouse, R. *et al.* (2002) A perspective on epistasis: Limits of models displaying no main effect. *Am. J. Hum. Genet.*, **70**, 461–471.

Danilevsky, A. and Shomron, N. (2021) Deep learning applied on next generation sequencing data analysis. In, *Deep Sequencing Data Analysis*. Springer, pp. 169–182.

Dudbridge, F. (2013) Power and predictive accuracy of polygenic risk scores. *PLoS Genet.*, **9**, e1003348.

Ehret, A. *et al.* (2015) Application of neural networks with back-propagation to genome-enabled prediction of complex traits in Holstein-Friesian and German Fleckvieh cattle. *Genet. Sel. Evol.*, **47**, 1–9.

Euesden, J. *et al.* (2015) PRSice: Polygenic Risk Score software. *Bioinformatics*, **31**, 1466–1468.

Figueroa, R.L. *et al.* (2012) Predicting sample size required for classification performance. *BMC Med. Inform. Decis. Mak.*, **12**, 1–10.

Gibson, G. (2012) Rare and common variants: Twenty arguments. *Nat. Rev. Genet.*, **13**, 135–145.

Goldstein, B.A. *et al.* (2010) An application of Random Forests to a genome-wide association dataset: Methodological considerations & new findings. *BMC Genet.*, **11**, 1–13.

Golub, G.H. and Reinsch, C. (1971) Singular value decomposition and least squares solutions. In, *Linear Algebra*. Springer, pp. 134–151.

González-Recio, O. *et al.* (2014) Machine learning methods and predictive ability metrics for genome-wide prediction of complex traits. *Livest. Sci.*, **166**, 217–231.

González-Recio, O. *et al.* (2013) The gradient boosting algorithm and random boosting for genome-assisted evaluation in large data sets. *J. Dairy Sci.*, **96**, 614–624.

González-Recio, O. and Forni, S. (2011) Genome-wide prediction of discrete traits using Bayesian regressions and machine learning. *Genet. Sel. Evol.*, **43**, 1–12.

He, K. *et al.* (2015) Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In, *Proceedings of the IEEE international conference on computer vision*., pp. 1026–1034.

Hotelling, H. (1933) Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.*, **24**, 417.

Hua, J. *et al.* (2005) Optimal number of features as a function of sample size for various classification rules. *Bioinformatics*, **21**, 1509–1515.

Khera, A.V. *et al.* (2018) Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations. *Nat. Genet.*, **50**, 1219–1224.

Kingma, D.P. and Ba, J. (2014) Adam: A method for stochastic optimization. *arXiv Prepr. arXiv1412.6980*.

Lango Allen, H. *et al.* (2010) Hundreds of variants clustered in genomic loci and biological pathways affect human height. *Nature*, **467**, 832–838.

Lehner, B. (2007) Modelling genotype–phenotype relationships and human disease with genetic interaction networks. *J. Exp. Biol.*, **210**, 1559–1566.

Lello, L. *et al.* (2019) Genomic prediction of 16 complex disease risks including heart attack, diabetes, breast and prostate cancer. *Sci. Rep.*, **9**, 1–16.

Lewis, C.M. and Knight, J. (2012) Introduction to genetic association studies. *Cold Spring Harb. Protoc.*, **2012**, pdb-top068163.

Maher, B. (2008) The case of the missing heritability: When scientists opened up the human genome, they expected to find the genetic components of common traits and diseases. But they were nowhere to be seen. Brendan Maher shines a light on six places where the missing lo. *Nature*, **456**, 18–22.

Makowsky, R. *et al.* (2011) Beyond missing heritability: Prediction of complex traits. *PLoS Genet.*, **7**, e1002051.

Manolio, T.A. *et al.* (2009) Finding the missing heritability of complex diseases. *Nature*, **461**, 747–753.

Marees, A.T. *et al.* (2018) A tutorial on conducting genome‐wide association studies: Quality control and statistical analysis. *Int. J. Methods Psychiatr. Res.*, **27**, e1608.

McKinney, B.A. *et al.* (2006) Machine learning for detecting gene-gene interactions. *Appl. Bioinformatics*, **5**, 77–88.

McKinney, B.A. and Pajewski, N.M. (2012) Six degrees of epistasis: Statistical network models for GWAS. *Front. Genet.*, **2**, 109.

Moore, J.H. (2003) The ubiquitous nature of epistasis in determining susceptibility to common human diseases. *Hum. Hered.*, **56**, 73–82.

Moore, J.H. and Williams, S.M. (2002) New strategies for identifying gene-gene interactions in hypertension. *Ann. Med.*, **34**, 88–95.

Paré, G. *et al.* (2017) A machine-learning heuristic to improve gene score prediction of polygenic traits. *Sci. Rep.*, **7**, 1–11.

Pedregosa, F. *et al.* (2011) Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.

Peng, J. *et al.* (2021) A deep learning-based genome-wide polygenic risk score for common diseases identifies individuals with risk. *medRxiv*.

Purcell, S. *et al.* (2007) PLINK: A tool set for whole-genome association and population-based linkage analyses. *Am. J. Hum. Genet.*, **81**, 559–575.

Romagnoni, A. *et al.* (2019) Comparative performances of machine learning methods for classifying Crohn Disease patients using genome-wide genotyping data. *Sci. Rep.*, **9**, 1–18.

Rumelhart, D.E. *et al.* (1985) Learning internal representations by error propagation. California Univ San Diego La Jolla Inst for Cognitive Science.

Sheppard, B. *et al.* (2021) A model and test for coordinated polygenic epistasis in complex traits. *Proc. Natl. Acad. Sci.*, **118**, e1922305118.

van der Sluis, S. *et al.* (2010) Phenotypic complexity, measurement bias, and poor phenotypic resolution contribute to the missing heritability problem in genetic association studies. *PLoS One*, **5**.

Srivastava, N. *et al.* (2014) Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**, 1929–1958.

Tenenbaum, J.B. *et al.* (2000) A global geometric framework for nonlinear dimensionality reduction. *Science (80-. ).*, **290**, 2319–2323.

Thomas, M. *et al.* (2020) Genome-wide modeling of polygenic risk score in colorectal cancer risk. *Am. J. Hum. Genet.*, **107**, 432–444.

Torkamani, A. *et al.* (2018) The personal and clinical utility of polygenic risk scores. *Nat. Rev. Genet.*, **19**, 581–590.

Visscher, P.M. *et al.* (2006) Assumption-free estimation of heritability from genome-wide identity-by-descent sharing between full siblings. *PLoS Genet.*, **2**, e41.

Werbos, P.J. (1982) Applications of advances in nonlinear sensitivity analysis. In, *System Modeling and Optimization*. Springer, pp. 762–770.

Xie, R. *et al.* (2017) A deep auto-encoder model for gene expression prediction. *BMC Genomics*, **18**, 39–49.

Yanes, T. *et al.* (2020) The emerging field of polygenic risk scores and perspective for use in clinical care. *Hum. Mol. Genet.*, **29**, R165–R176.