Optimizing and benchmarking polygenic risk scores with GWAS summary statistics

Zijie Zhao[1], Tim Gruenloh[1], Yixuan Wu[1], Zhongxuan Sun[1], Jiacheng Miao[1], Yuchang Wu[1,2], Jie Song[3], Qiongshi Lu[1,2,3,#]

[1] Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, WI
[2] Center for Demography of Health and Aging, University of Wisconsin-Madison, Madison, WI
[3] Department of Statistics, University of Wisconsin-Madison, WI

To whom correspondence should be addressed:
Dr. Qiongshi Lu (qlu@biostat.wisc.edu)

Abstract

We introduce an innovative statistical framework to optimize and benchmark polygenic risk score (PRS) models using summary statistics of genome-wide association studies. This framework builds upon our previous work and can fine-tune virtually all existing PRS models while accounting for linkage disequilibrium. In addition, we provide an ensemble learning strategy named PUMA-CUBS to combine multiple PRS models into an ensemble score without requiring external data for model fitting. Through extensive simulations and analysis of many complex traits in the UK Biobank, we demonstrate that this approach closely approximates gold-standard analytical strategies based on external validation, and substantially outperforms state-of-the-art PRS methods. We argue that PUMA-CUBS is a powerful and general modeling technique that can continue to combine the best-performing PRS methods out there through ensemble learning and could become an integral component for all future PRS applications.

Introduction

Genetic risk prediction is a main focus in human genetics research and a key step towards precision medicine[1-3]. Continued success in genome-wide association studies (GWAS) in the past decade has facilitated the development of polygenic risk scores (PRS) that aggregate the effects of millions of single nucleotide polymorphisms (SNPs) for many complex traits[4-6]. Compared to earlier statistical methods that require individual-level data for model training[7-10], PRS which only relies on GWAS summary data is much more generally applicable due to the wide availability of GWAS summary statistics. Although earlier PRS models struggled to produce accurate prediction results, recent and more sophisticated PRS methods have achieved substantially improved prediction accuracy through statistical regularization and biological data integration[11-17]. In numerous studies, PRS has shown promising performance in stratifying disease risk and great potential in informing early lifestyle changes or medical interventions[18-21].

Despite the progress, several lingering challenges create a significant gap between PRS methodology and applications. A main recurring issue we highlight (and address) throughout the paper is that PRS modelers often assume the existence of independent individual-level datasets that can be used for additional model tuning. But in practice, GWAS summary statistics are used for PRS model training, meaning that conventional sample splitting schemes cannot be used. Additional datasets that are independent from both training and testing samples also rarely exist. This suggests that model-tuning samples will have to come from the precious testing dataset which inevitably reduces the sample size and statistical power in downstream applications.

This disconnection between impractical method requirements and limited data availability can lead to a variety of problems. For example, many PRS methods have tuning parameters that could substantially swing model performance when not chosen properly[12-15,22-24]. Conventionally, these parameters need to be fine-tuned on a separate dataset with individual-level genotypes and phenotypes. Although some recent methods employ full Bayesian or empirical Bayesian techniques to bypass model fine-tuning[25-27], these hyperparameter-free PRS do not always outperform fine-tuned models, trading predictive accuracy for computational feasibility[28,29]. Second, no PRS method universally outperforms all other approaches. The empirical performance of a PRS model depends on GWAS sample size, genetic architecture of the phenotype, quality of GWAS summary statistics, and heterogeneity between training and testing samples[30-33]. Thus, it is of great interest to systematically and impartially benchmark various PRS methods for each trait, ideally in an independent dataset[11,30,34]. Third, several recent studies have applied ensemble learning which combines multiple PRS models via another regression[28,29]. This brute-force approach has shown superior performance compared to any single PRS method but is data-demanding – the second level regression model needs to be fit on a separate dataset. Finally, we note that it may be of interest to combine all these tasks in practice, e.g., benchmarking an ensemble learner that combines multiple PRS models which all need to be tuned separately. Now this truly becomes mission impossible.

In this paper, we seek a solution to these problems. We base our statistical framework on PUMAS, a method we recently introduced to perform Monte Carlo cross-validation (MCCV) using GWAS summary statistics[35]. We have shown that PUMAS can effectively fine-tune PRS models with clumped SNPs[36] and the approach has since been adopted in other applications[37-39]. Here, we first demonstrate that PUMAS can fine-tune and benchmark state-of-the-art PRS models without SNP pruning. Second, we introduce an extension to the PUMAS framework named PUMA-CUBS which is a highly innovative strategy to perform ensemble learning using GWAS summary data alone. Taken together, we showcase a sophisticated statistical framework for fine-tuning, benchmarking, and combining PRS models using GWAS summary statistics as input. We

demonstrate the performance of our approach through extensive simulations and analysis of 19 complex traits in UK Biobank (UKB). On average, the PUMA-CUBS ensemble PRS achieves a 6.54% relative gain in predictive R^2 compared to LDpred2 and a 15.00% gain compared to PRS-CS, respectively. We also apply our method to 31 well-powered GWAS with publicly available summary statistics and provide a catalog of ensemble PRS with benchmarked predictive performance.

Results

Method Overview

First, we present an overview of the PUMA-CUBS workflow. Statistical details and technical discussions are presented in the **Methods** section. For illustration, first we assume individual-level data is available. In this case, we would divide the samples into 4 independent sets for PRS training, model fine-tuning, constructing ensemble PRS, and benchmarking model performance, respectively (**Figure 1A**). The main goal of our new approach is to mimic this procedure when only summary statistics are available. Using PUMAS, we could sample marginal association statistics for a subset of individuals in the GWAS[35]. Doing this repeatedly, we could divide the full GWAS summary data to corresponding training, tuning, ensemble learning, and testing summary statistics (**Figure 1B**). Using these four sets of sub-sampled summary statistics, we train a series of PRS models, fine-tune each PRS model to select the besting tuning parameters, apply PUMA-CUBS to combine PRS models through linear regression, and finally evaluate the predictive performance of PRS models. The entire procedure only requires GWAS summary statistics and linkage disequilibrium (LD) references as input.

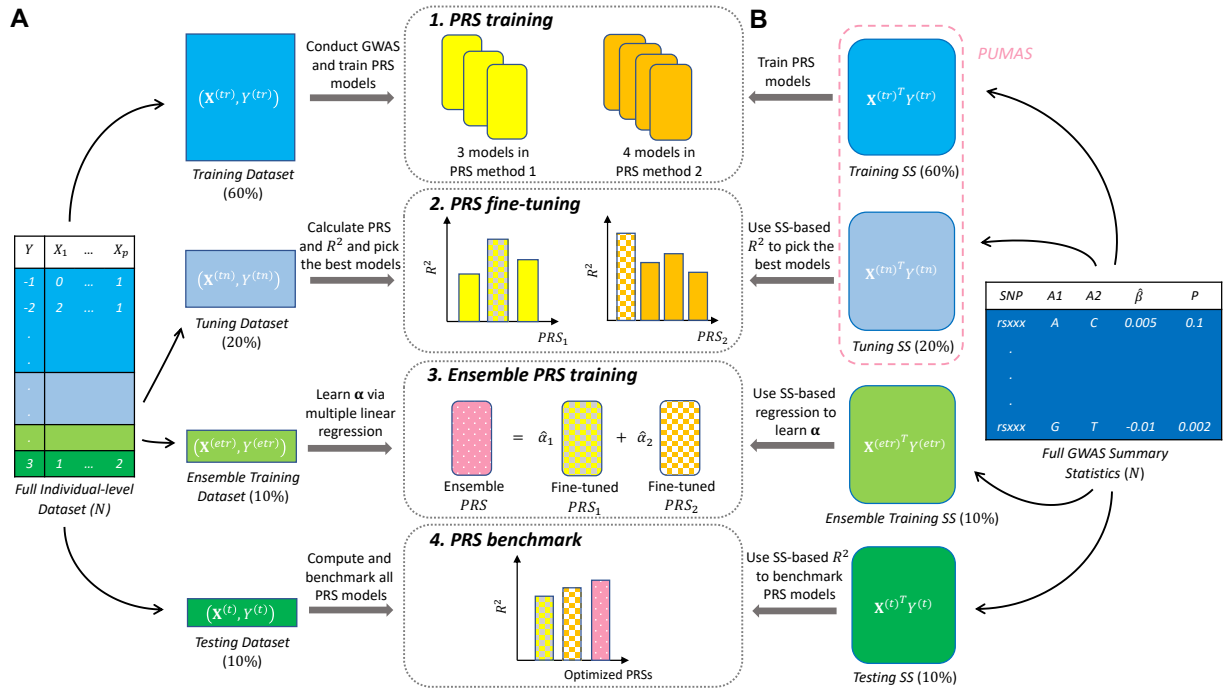

Figure 1. Workflow of PRS construction and evaluation. (**A**) Conventional approach divides the entire individual-level dataset to different subset of samples for each of 4 stages of PRS analysis. (**B**) PUMA-CUBS directly partitions the full summary-level data to corresponding summary statistics for different analytical purposes.

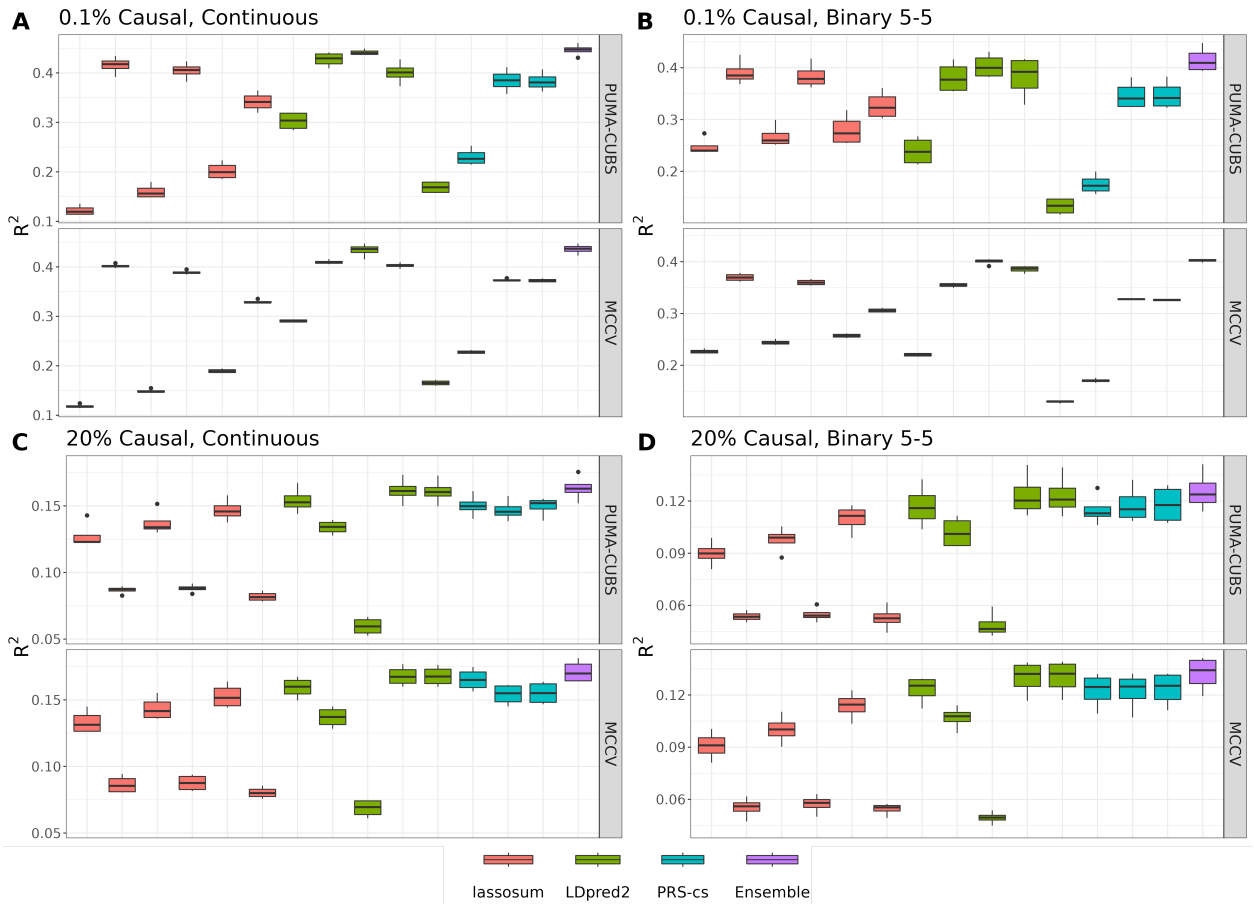

Figure 2. Comparison of PUMA-CUBS and MCCV in UKB simulation. (**A** and **C**) Simulation results for quantitative traits. (**B** and **D**) Simulation results for binary traits with balanced case-control ratio. Proportion of causal variants is 0.1% in **A** and **B**, and 20% in **C** and **D**. The heritability is set to be 0.5 in all panels. Y-axis: predictive R^2 across 4 repeats of MCCV; X-axis (left to right): lassosum models (red boxes) with tuning parameter settings: s=0.2 and λ=0.005, s=0.2 and λ=0.01, s=0.5 and λ=0.005, s=0.5 and λ=0.01, s=0.9 and λ=0.005, s=0.9 and λ=0.01. LDpred2 models (green boxes): non-infinitesimal with p=0.1, non-infinitesimal with p=0.01, non-infinitesimal with p=0.001, non-infinitesimal with p=auto, and infinitesimal model. PRS-CS (blue boxes): ϕ=0.01, 0.001, and auto. Finally, the purple box shows the results of ensemble PRS. Results for remaining simulation settings are summarized in **Supplementary Figures 1-7** and **Supplementary Tables 1-4**.

Simulation results

We performed simulations using imputed genotype data from UKB to demonstrate that PUMAS and PUMA-CUBS can fine-tune, combine, and benchmark PRS models. We included 100,000 independent individuals of European descent and 944,547 HapMap3 SNPs in the analysis. We simulated phenotypes with heritability of 0.2, 0.5, and 0.8 and randomly assigned causal variants under sparse and polygenic settings to mimic different types of genetic architecture (**Methods**). We performed GWAS and obtained marginal association statistics. We then implemented PUMAS and PUMA-CUBS to conduct a 4-fold MCCV to train, optimize, and evaluate lassosum, PRS-CS, LDpred2, and an ensemble PRS which combines all three methods[22,25,26]. For comparison, we also implemented a MCCV procedure using individual-level UKB data. We partitioned the UKB dataset into 4 mutually exclusive datasets. We used datasets 1 and 2 to train

and fine-tune each PRS method, then used the third dataset to fit a regression to combine multiple PRS. We evaluated each PRS method in the fourth dataset and reported PRS prediction accuracy quantified by R^2. We describe implementation details of both summary statistics-based and individual-level data-based MCCV in **Methods**.

Overall, we observed highly consistent results between PUMAS/PUMA-CUBS and MCCV for both quantitative and binary phenotypes (**Figure 2**; **Supplementary Figures 1-7**; **Supplementary Tables 1-4**). In addition, summary statistics-based approaches can closely approximate R^2 values obtained from model-tuning and benchmarking techniques using individual-level data. PUMA-CUBS also constructed scores that were highly concordant with ensemble PRS built from individual-level data which universally outperformed all PRS models used as input.

PUMAS can fine-tune and benchmark PRS methods

Next, we demonstrate that PUMAS effectively fine-tunes PRS models and performs accordantly with the gold standard external validation approach based on individual-level data. We applied PUMAS to 16 quantitative traits and 3 diseases in UKB (**Supplementary Tables 5-6**)**.** After quality control, the UKB dataset contained 375,064 independent individuals and 1,030,187 SNPs (**Methods**). We applied a 9-to-1 data split to hold out 10% of the samples for external validation, and performed GWAS for all traits using 90% of the samples. We applied 4-fold MCCV implemented in PUMAS to train and fine-tune three PRS models (i.e., LDpred2, lassosum, and PRS-CS which have been demonstrated to achieve high prediction accuracy in a recent benchmark study[22,25,26,29]) using only summary statistics. For external validation, we trained PRS models using summary statistics and calculated PRS prediction accuracy on the holdout dataset. We report the best tuning parameters for LDpred2, lassosum, and PRS-CS and corresponding R^2 obtained from both PUMAS and external validation.

Our summary-statistics-based approach showed highly consistent model-tuning performance for all analyzed traits compared to external validation (**Figure 3**, **Supplementary Figures 8-26**; **Supplementary Tables 7-8**). Among 19 traits, PUMAS and external validation selected the same best tuning parameters 19, 17, and 11 times for lassosum, LDpred2, and PRS-CS, respectively. When the model tuning results were different between PUMAS and external validation, they still selected models with very similar prediction accuracy. In addition, PUMAS provided precise R^2 estimates for all models compared to external validation, advocating the use of our summary-statistics-based approach for PRS model benchmarking.

We also observed that the parameter-tuning results are accordant with the analyzed traits' genetic architecture. For both height and monocyte count, PUMAS accurately selected the best tuning parameters based on external validation (**Figure 3A-B**), but the selected models were not the same between these two traits. Height is known to be extremely polygenic with more than 12,000 independent GWAS signals in the latest GWAS[40]. In comparison, fewer loci have been found to significantly associate with monocyte count[41]. Our model-tuning results suggest that polygenic prediction models fit best for height (e.g., LDpred2-Infinitesimal and PRS-CS with $\phi = 0.01$) while sparser PRS models with stronger regularization (e.g., PRS-CS with $\phi = 0.0001$) provide better prediction accuracy for monocyte count.

Finally, PUMAS can also effectively estimate predictive R^2 for binary traits (**Figure 3C**-**D**). To calculate interpretable R^2 for binary outcomes, PUMAS first transforms GWAS summary statistics obtained from logistic regressions to the linear regression scale, and then computes R^2 on the

observed scale[42-44]. To show that such transformation is valid, we trained two sets of PRS models using both transformed and original logistic regression summary statistics for 3 disease traits and observed nearly identical PRS performance between two approaches (**Supplementary Table 8**; **Supplementary Figures 24-26**). Details in the implementation of binary trait analysis and summary statistics transformation are presented in **Methods**.

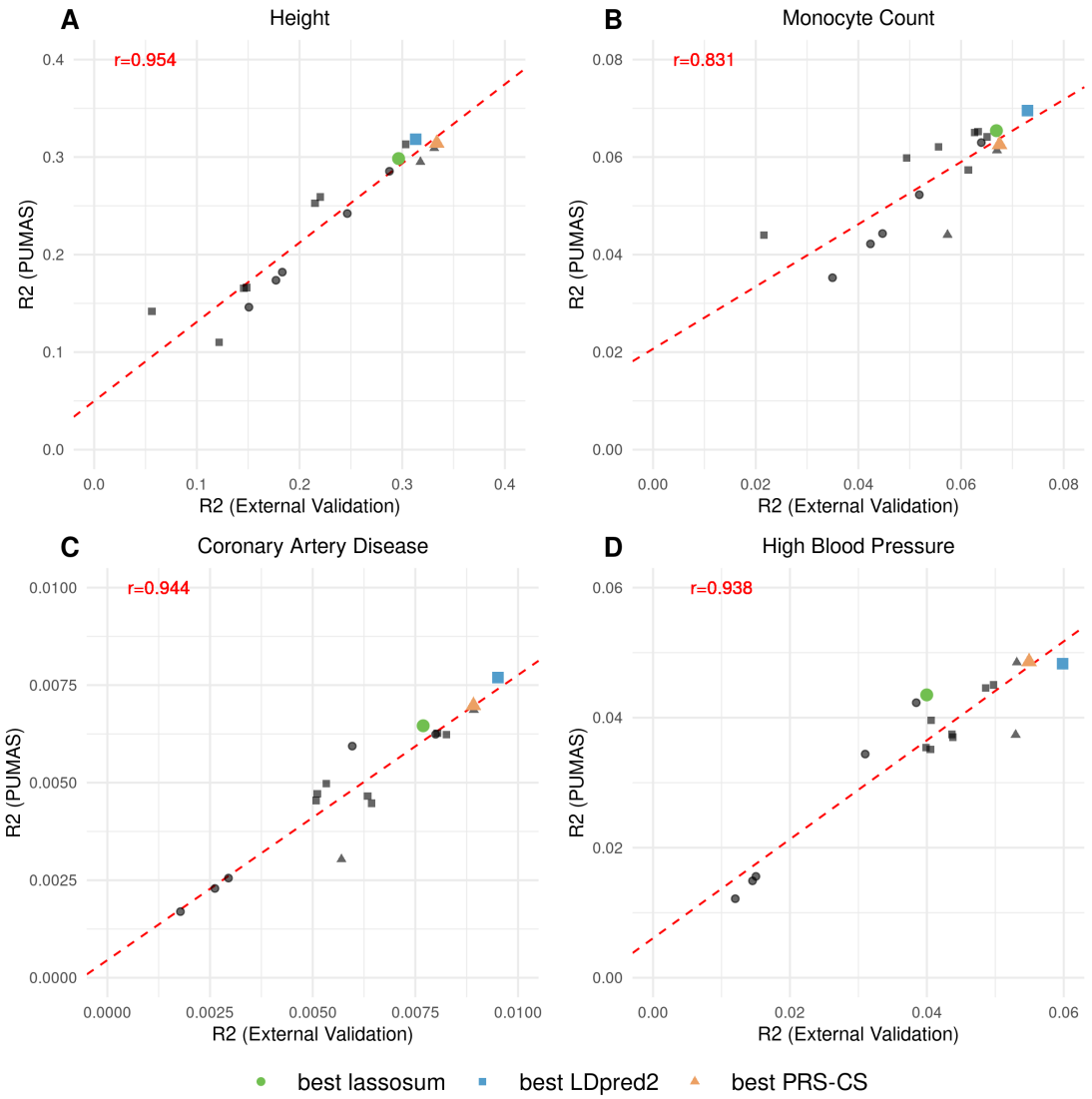

Figure 3. Comparing PUMAS results with external validation. Four panels show the model-tuning results for (**A**) height, (**B**) monocyte count, (**C**) coronary artery disease, and (**D**) high blood pressure. Y-axis: average predictive R^2 across 4-fold replications from PUMAS; X-axis: predictive R^2 evaluated by external validation on the holdout dataset. Each data points represents a PRS model with different tuning parameters and the shape of data points indicate three different PRS methods: LDpred2, PRS-CS, and lassosum. The best tuning parameter setting suggested by PUMAS for each PRS method is highlighted and colored. The dashed red line is fitted regression line between PRS R^2 from PUMAS and external validation. Pearson correlations between two sets of results are shown in each panel. Detailed model-tuning results for all 19 traits are summarized in **Supplementary Tables 7-8** and **Supplementary Figures 8-26**.

Ensemble learning via PUMA-CUBS substantially improves PRS prediction accuracy

Here we apply PUMA-CUBS, the ensemble learning extension of PUMAS, to UKB traits and show that ensemble PRS has superior prediction accuracy compared to each PRS method and our summary statistics-based approach is comparable to ensemble learning results based on individual-level data. We constructed linearly combined scores of lassosum, PRS-CS, and LDpred2. Using individual-level data, we split the 10% UKB holdout dataset into two equally sized subsets. We fitted a multiple regression on the first holdout set to aggregate the best-performing PRS models trained and tuned from GWAS summary statistics, and then evaluated the ensemble score's prediction accuracy using the second holdout set. For comparison, we implemented PUMA-CUBS to conduct 4-fold MCCV to perform ensemble learning and assess its performance using summary statistics alone.

Our approach showed almost identical performance compared to individual-level data results (**Figure 4A**), showcasing PUMA-CUBS' ability to benchmark and construct ensemble PRS without requiring additional datasets. In addition, ensemble PRS achieved the highest prediction accuracy for all traits compared with three input PRS models (**Supplementary Figure 27**; **Supplementary Table 9**). The ensemble PRS using individual-level data as input had an average 15.77% and 7.25% relative gain in R^2 compared to PRS-CS-auto and LDpred2-auto while the PUMA-CUBS ensemble PRS delivered a similar 15.00% and 6.54% R^2 increase respectively (**Figure 4B**), highlighting the substantial gain in prediction accuracy from ensemble learning.

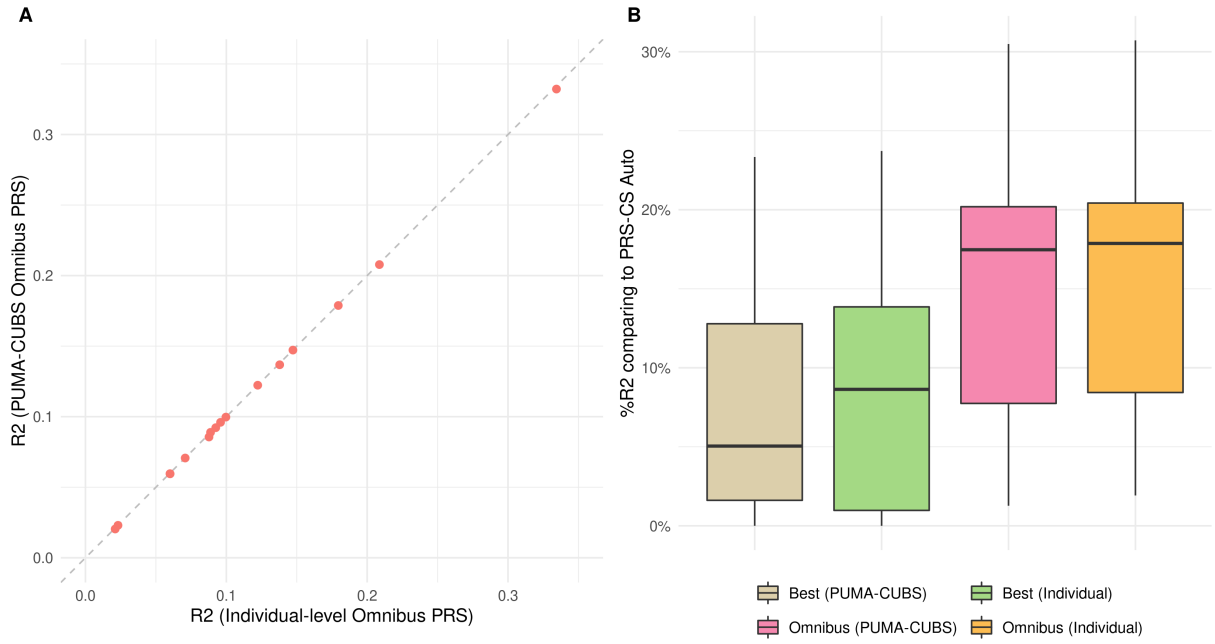

Figure 4. Constructing ensemble PRS for UKB traits. (**A**) Comparing two sets of ensemble PRS obtained from PUMA-CUBS and individual-level data. The gray dashed line is the diagonal line. (**B**) Comparing ensemble PRS with input PRS methods. Y-axis: relative percentage increase in R^2 compared to PRS-CS-auto; X-axis: 4 sets of PRS models, including the best single PRS suggested by PUMAS, the best single PRS selected based on the first individua-level holdout set, the ensemble PRS obtained from PUMA-CUBS, and the ensemble PRS trained from individual-level data. All R^2 values were computed using the second half of holdout dataset.

Constructing and benchmarking ensemble PRS for 31 complex traits

Finally, we applied PUMA-CUBS to provide a comprehensive catalog of ensemble PRS for 31 publicly available GWAS summary statistics with varying sample size and genetic architecture. The detailed information and selecting criteria for GWAS summary-level data are summarized in **Methods** and **Supplementary Tables 10-11**. We employed extensive quality controls to pinpoint and calibrate misspecifications in GWAS summary statistics following a recent study[31]. We also transformed logistic summary statistics to linear scale to produce interpretable R^2 for binary traits[42,44]. For each trait, we report prediction accuracy of the best performing PRS model and ensemble PRS. The full results of the PRS catalog are presented in **Supplementary Table 12**. The predictive performance of ensemble PRS is correlated with estimated trait heritability, and the predictive R^2 ranged from 0.001 to 0.227 across 31 traits, showing highly diverse predictive performance of genetic risk prediction. We also note that ensemble PRS improved predictive R^2 for every trait in the analysis with an average increase of 31.36% compared to PRS-CS-auto (**Supplementary Figure 28**). Among 31 complex diseases and traits, we observed the highest prediction improvement for rheumatoid arthritis (103.8%), Alzheimer's disease (71.96%, 83.68%, and 98.82% on three datasets), ischaemic stroke (69.48% and 78.35% on two datasets), and Parkinson's disease (75.55%).

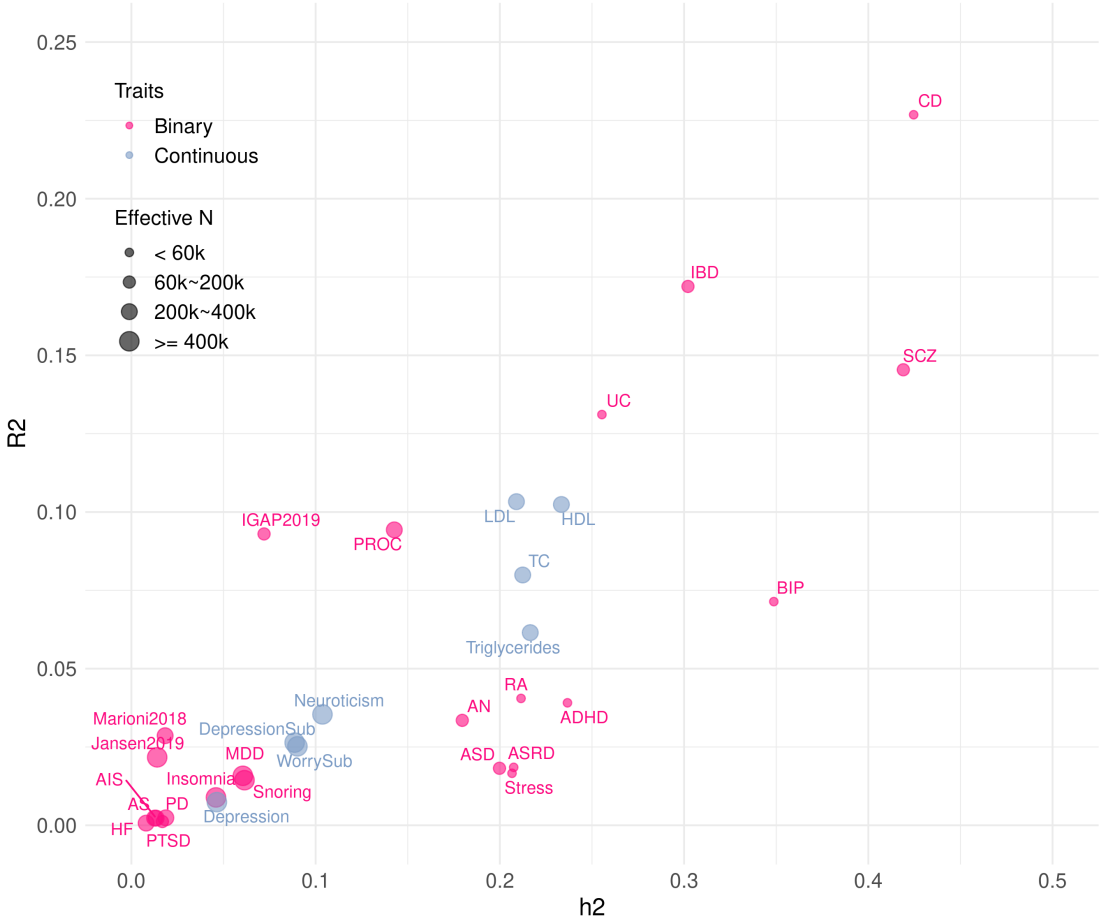

Figure 5. An ensemble PRS catalog for 31 complex traits. Y-axis: predictive R^2 of PUMA-CUBS ensemble PRS; X-axis: heritability estimates from LD score regression[45]. Size of data points indicates the effective sample size of each GWAS. Binary traits and continuous traits are highlighted with different colors. Detailed PRS benchmark results are presented in **Supplementary Table 12**.

Another observation is that the ensemble PRS R^2 exceeded the estimated trait heritability for all three Alzheimer's disease GWAS. To demonstrate that this is not an artifact from overestimating predictive R^2, we conducted additional analysis (**Methods**) using IGAP 2019 Alzheimer's GWAS summary statistics[46] and compared our results with external validation based on 2,600 Alzheimer's disease cases and 5,200 healthy controls in UKB (**Supplementary Table 13**). The R^2 of AD PRS obtained from external validation also exceeded estimated heritability (h^2=0.072, SE=0.012) and the results were consistent with PUMAS R^2 estimation (**Supplementary Figure 29**; **Supplementary Table 14**). We hypothesized that this is driven by the *APOE* region which contributes an unusually large fraction of AD risk[47-49]. Indeed, after removing 383 SNPs in the *APOE* region from IGAP 2019 AD summary statistics (**Methods**), we observed a steep decline in R^2 for both external validation and PUMAS. Both R^2 values became substantially lower than the estimated h^2 of 0.066 without *APOE* region (SE=0.009; **Supplementary Table 14**).

Discussion

Fine-tuning and benchmarking PRS models are challenging tasks due to the need of external individual-level datasets that are independent from the input GWAS. In this work, we extended our PUMAS approach to incorporate LD and fine-tune state-of-the-art PRS methods. In both simulations and analysis of UKB traits, we observed high concordance between PUMAS and results based on external validation using hold out samples. In addition, we presented a novel framework named PUMA-CUBS to perform ensemble learning and create combined PRS using only GWAS summary statistics. We showed that ensemble PRS created by PUMA-CUBS closely approximates scores built from hold out samples. Further, these ensemble scores substantially outperformed state-of-the-art PRS methods for all complex traits we analyzed in the study. Finally, we applied PUMA-CUBS to a collection of publicly available GWAS summary statistics and provided a comprehensive catalog of benchmarked and optimized PRS.

Our work presents several major advances that will impact future PRS applications. First, our method fills an important gap between PRS methodological research and its real-world applications. Currently, many PRS methods still have tuning parameters and grid search on external individual-level datasets remains the most common technique for fine-tuning these models. In practice, this kind of data can either be impossible to obtain, or need to be split from testing samples which could hurt statistical power in PRS applications[32]. Our method provides a universal solution to PRS model fine-tuning. Second, model benchmarking is another major challenge in the field which conventionally relies on external validation data. Comprehensive and unbiased benchmarking allows researchers to compare the effectiveness of different PRS methods for particular traits of interest, and importantly, estimate PRS predictive accuracy without using testing samples. We note that although some advanced PRS approaches do not require model fine-tuning anymore, no existing methods could benchmark model performance using GWAS summary data, which is crucial for model selection, power calculation, and study design. Our approach now provides a solution to this problem. Third, the ensemble learning approach which combines multiple predictive models through a second level regression has been viewed as a highly effective but data-demanding approach[28,29,33]. A major advance in this study is the introduction of PUMA-CUBS which allows ensemble learning on GWAS summary statistics. We note that this approach not only showcased a substantial gain over existing PRS methods, but is generally applicable to future PRS developments. If a future PRS approach shows promising improvements compared to older methods, that new approach can also be incorporated into the ensemble PRS. In our view, PUMA-CUBS is not a competing approach for any existing PRS

model, but instead is a flexible and general modeling technique that combines the best-performing methods out there and should be applied to all future PRS applications.

Our study has several limitations. First, we have constrained statistical analysis in this study to the European ancestral population. PRS is known to transfer poorly in terms of prediction accuracy for non-European populations which could exacerbate the disparity in genomic medicine between ancestral groups[50,51]. It is an important future direction to systematically optimize and benchmark PRS for diverse ancestral populations which would require incorporation of multiple sets of ancestry-specific GWAS and LD references. Although we did not explore this topic in this paper, our recent work introduced parallel ideas to tackle the challenges in multi-ancestry genetic risk prediction[39]. Second, analyses in this study were limited to GWAS summary statistics computed from independent samples. It remains to be investigated whether application of these approaches will be affected if the input GWAS summary statistics were obtained from linear mixed models with related samples or family-based designs[52-54]. Future work will focus on developing statistical methods to correct for sample relatedness or demonstrate robustness to these issues. That said, we expect PRS model-tuning to remain valid even with sample relatedness since the inflation in R^2 should be uniform across various tuning parameter settings, although biases may be introduced to the predictive R^2 which could affect benchmarking efforts. Third, our current analyses focused only on lassosum, PRS-CS, and LDpred2. While it serves to support the superiority of ensemble PRS as a proof of concept, more PRS methods need to be jointly modeled and evaluated in the future, including scores that leverage auxiliary information including functional annotation[13,14] or multiple phenotypes[15,17,55]. Finally, collinearity among PRS models could arise when using multiple regression to combine a large number of scores since some PRS methods tend to yield similar results. Therefore, another future direction is to incorporate variable selection strategies into our ensemble learning framework which could also involve penalized regression.

To sum up, we presented a sophisticated statistical framework to fine-tune, combine, and benchmark PRS methods using only GWAS summary statistics. This is a statistically novel and computationally efficient approach with flexible implementation that can handle a variety of applications. We have demonstrated its performance through careful and comprehensive analyses, and we argue that this framework presents highly innovative and generally applicable features that should become the default in many future PRS studies.

Methods

Sampling distribution of summary statistics

We adopt a commonly used linear model framework to quantify the relationship between a quantitative trait and SNP genotypes:

$$Y = \mathbf{X}\boldsymbol{\beta} + \epsilon$$

Here, Y denotes the trait, $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_p)$ denotes the genotypes of p SNPs, $\boldsymbol{\beta} \in \mathbb{R}^p$ denotes their true effect sizes, and ϵ denotes the random error that is independent from $\mathbf{X}$ and follows a normal distribution with mean zero and some variance σ_e^2. Let $\mathbf{y}$ and $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$ denote the observed values for Y and $\mathbf{X}$ from N independent individuals. For simplicity, we assume both $\mathbf{y}$ and $\mathbf{x}_j$ ($j = 1, \ldots, p$) are centered. Then, GWAS summary statistics can be denoted as:

$$\hat{\beta}_j = (\mathbf{x}_j^T \mathbf{x}_j)^{-1}(\mathbf{x}_j^T \mathbf{y}) \quad (1)$$

$$SE(\hat{\beta}_j) = \sqrt{\frac{\hat{\boldsymbol{\epsilon}}_j^T \hat{\boldsymbol{\epsilon}}_j}{(N-1)\mathbf{x}_j^T \mathbf{x}_j}} \quad (2)$$

where $\hat{\boldsymbol{\epsilon}}_j = \mathbf{y} - \mathbf{x}_j \hat{\beta}_j$ are the residuals from the marginal linear regression between the trait and the j-th SNP. To train, fine-tune, combine, and benchmark PRS models, independent datasets are required to avoid overfitting. We have previously proposed a flexible statistical framework to generate training and fine-tuning datasets when only GWAS summary statistics are available[35]. Here, we generalize this statistical framework in two different directions. First, we allow our method to incorporate LD information. We note that this extension is similar to some recent work built on our initial PUMAS paper[37,39]. Second, we allow the method to partition full GWAS summary statistics into more than two datasets for various analytical purposes. Let $y^{(s)}$ and $x^{(s)}$ denote phenotype and genotype data for any arbitrary subset of N individuals with sample size $N^{(s)}$. When N is large enough, we have previously shown that by central limit theorem[35]:

$$\mathbf{x}^T\mathbf{y} \sim \mathbf{N}\big(NE(\mathbf{X}^T Y), NVar(\mathbf{X}^T Y)\big)$$

$$\mathbf{x}^{(s)^T}\mathbf{y}^{(s)} \sim \mathbf{N}((N-N^{(s)})E(\mathbf{X}^T Y), (N-N^{(s)})Var(\mathbf{X}^T Y))$$

where $\mathbf{X}^T Y = (X_1 Y, \ldots, X_p Y)^T$. Then, given the observed summary-level data from GWAS, the conditional distribution of summary statistics of a subset of GWAS samples is

$$\mathbf{x}^{(s)^T}\mathbf{y}^{(s)} | \mathbf{x}^T\mathbf{y} \sim \mathbf{N}\left(\frac{N^{(s)}}{N}\mathbf{x}^T\mathbf{y}, \frac{\big(N-N^{(s)}\big)N^{(s)}}{N}\widehat{\boldsymbol{\Sigma}}\right) \quad (3)$$

where $\widehat{\boldsymbol{\Sigma}}$ is the observed variance-covariance matrix for $\mathbf{X}^T Y$. To subsample summary statistics $\mathbf{x}^{(s)^T}\mathbf{y}^{(s)}$, we need to estimate $\mathbf{x}^T\mathbf{y}$ and $\boldsymbol{\Sigma}$ first. Recall formula *(1)* for marginal regression coefficient estimation, $\mathbf{x}_j^T\mathbf{y}$ can be calculated using $\hat{\beta}_j$ and $\mathbf{x}_j^T\mathbf{x}_j$ which is proportional to SNP variance and can be estimated by minor allele frequency reported from GWAS or imputed from LD reference panel. On the other hand, deriving $\boldsymbol{\Sigma}$ is more complicated and we discuss how $\widehat{\boldsymbol{\Sigma}}$ is estimated using summary statistic and an LD reference panel in the following section.

Estimate variance-covariance matrix of summary statistics

Let D denote the SNP correlation matrix and d_{jk} denote the correlation between the j-th and the k-th SNPs. Let $\boldsymbol{\Sigma}$ be the true covariance matrix of summary statistics with diagonal and off-diagonal elements denoted as $\boldsymbol{\Sigma}_j$ and $\boldsymbol{\Sigma}_{jk}$, respectively. For convenience, we write $Y = \mathbf{X}\boldsymbol{\beta} + \epsilon = X_1\beta_1 + \ldots + X_p\beta_p + \epsilon = X_j\beta_j + \epsilon_j$, where $\epsilon_j = \sum_{i:i\neq j} X_i\beta_i + \epsilon$. Then the diagonal terms of the $\boldsymbol{\Sigma}$ can be written as

$$
\begin{aligned}
\boldsymbol{\Sigma}_j &= Var(X_j Y) \\
&= Var[X_j(X_j\beta_j + \epsilon_j)] \\
&= \beta_j^2 Var(X_j^2) + Var(X_j\epsilon_j) + 2\beta_j Cov(X_j^2, X_j\epsilon_j) \\
&= \beta_j^2 Var(X_j^2) + Var[X_j(\sum_{i:i\neq j} X_i\,\beta_i + \epsilon)] + 2\beta_j Cov(X_j^2, X_j\epsilon_j)
\end{aligned}
$$

We partition all SNPs in the genome into 2 sets. Let S_1 be the index set that contains all SNPs that are independent from the j-th SNP and S_2 be the set with all remaining SNPs that are in LD with the j-th SNP. Then we can further expand $\boldsymbol{\Sigma}_j$ by

$$
\begin{aligned}
\boldsymbol{\Sigma}_j = \;& \beta_j^2 Var(X_j^2) + Var[X_j(\sum_{g\in S_1} X_g\beta_g + \epsilon)] + Var[X_j(\sum_{g'\in S_2} X_{g'}\beta_{g'})] + \\
& 2Cov[X_j(\sum_{g\in S_1} X_g\beta_g + \epsilon), X_j(\sum_{g'\in S_2} X_{g'}\beta_{g'})] + 2\beta_j Cov(X_j^2, X_j\epsilon_j) \\
= \;& \beta_j^2 Var(X_j^2) + Var[X_j(\sum_{g\in S_1} X_g\beta_g + \epsilon)] + \sum_{g'\in S_2} \beta_{g'}^2\, Var(X_j X_{g'}) + \\
& 2\sum_{g_1'\in S_2, g_2'\in S_2, g_1'\neq g_2'} \beta_{g_1'}\beta_{g_2'} Cov(X_j X_{g_1'}, X_j X_{g_2'}) + 2\sum_{g'\in S_2} \beta_{g'} Cov[X_j\epsilon, X_j X_{g'}] + \\
& 2\sum_{g\in S_1}\sum_{g'\in S_2} \beta_g\beta_{g'} Cov[X_j X_g, X_j X_{g'}] + 2\beta_j Cov(X_j^2, X_j\epsilon_j)
\end{aligned}
$$

We can simplify $\boldsymbol{\Sigma}_j$ based on two commonly made assumptions. First, any given SNP should be in linkage equilibrium with the vast majority of SNPs in the genome. Therefore, we can safely assert $|S_1| \gg |S_2|$. Second, each individual SNP's effect on the phenotype is typically very small such that the products of any effect sizes are negligible in practice. Taken together, we can reduce the expansion of $\boldsymbol{\Sigma}_j$ by discarding SNPs in S_2 which eventually allows us to treat X_j and ϵ_j as independent in practice:

$$
\begin{aligned}
\boldsymbol{\Sigma}_j &\approx Var[X_j(\sum_{g\in S_1} X_g\beta_g + \epsilon)] \\
&\approx Var[X_j\epsilon_j] \\
&= E(X_j^2\epsilon_j^2) - [E(X_j\epsilon_j)]^2 \\
&\approx E(X_j^2)E(\epsilon_j^2)
\end{aligned}
$$

Note that $E(X_j^2)$ can be easily approximated using an MAF-based estimator, denoted as $\hat{\sigma}_j^2$, that may be obtained either from the full GWAS summary statistics or the LD reference data. For $E(\epsilon_j^2)$, we can estimate its value by standard error of effect size estimation from GWAS summary data using formula *(2)*. In this way we can obtain an estimator of $\boldsymbol{\Sigma}_j$ as

$$
\hat{\boldsymbol{\Sigma}}_j = N[SE(\hat{\beta}_j)\hat{\sigma}_j^2]^2 \qquad (4)
$$

To estimate off-diagonal terms $\boldsymbol{\Sigma}_{jk}$, we now write $Y = \mathbf{X}\boldsymbol{\beta} + \epsilon = X_1\beta_1 + \ldots + X_p\beta_p + \epsilon = X_j\beta_j + X_k\beta_k + \epsilon_{jk}$, where $\epsilon_{jk} = \sum_{i:i\notin\{j,k\}}^{p} X_i\,\beta_i + \epsilon$. Under the same assumption where the magnitude of SNP effects is very small, we can simplify $\boldsymbol{\Sigma}_{jk}$ by:

$$
\begin{aligned}
\boldsymbol{\Sigma}_{jk} = \;& Cov[X_j(X_j\beta_j + X_k\beta_k + \epsilon_{jk}), X_k(X_j\beta_j + X_k\beta_k + \epsilon_{jk})] \\
= \;& Cov(X_j^2\beta_j, X_jX_k\beta_j) + Cov(X_jX_k\beta_k, X_jX_k\beta_j) + Cov(X_j\epsilon_{jk}, X_jX_k\beta_j) + \\
& Cov(X_j^2\beta_j, X_k^2\beta_k) + Cov(X_jX_k\beta_k, X_k^2\beta_k) + Cov(X_j\epsilon_{jk}, X_k^2\beta_k) + \\
& Cov(X_j^2\beta_j, X_k\epsilon_{jk}) + Cov(X_jX_k\beta_k, X_k\epsilon_{jk}) + Cov(X_j\epsilon_{jk}, X_k\epsilon_{jk}) \\
\approx \;& Cov(X_j\epsilon_{jk}, X_jX_k\beta_j) + Cov(X_j\epsilon_{jk}, X_k^2\beta_k) + Cov(X_j^2\beta_j, X_k\epsilon_{jk}) + \\
& Cov(X_jX_k\beta_k, X_k\epsilon_{jk}) + Cov(X_j\epsilon_{jk}, X_k\epsilon_{jk}) \\
\approx \;& Cov(X_j\epsilon_{jk}, X_k\epsilon_{jk})
\end{aligned}
$$

In a similar fashion, we further partition all SNPs in the genome other than the j-th and the k-th SNP into two sets. Let S_3 denote the collection of SNPs that are independent from both the j-th and the k-th SNPs, and S_4 includes the remaining SNPs that are in LD with either the j-th or the k-th SNP. Based on a similar rationale, we can safely assume that $|S_3| \gg |S_4|$. Then, by ignoring SNPs in S_4 and thus treating X_j and X_k as being independent from ϵ_{jk}, we express $\boldsymbol{\Sigma}_{jk}$ as:

$$
\begin{aligned}
Cov(X_j\epsilon_{jk}, X_k\epsilon_{jk}) = \;& Cov[X_j(\sum_{l\in S_3} X_l\beta_l + \epsilon), X_k(\sum_{l\in S_3} X_l\beta_l + \epsilon)] + Cov[X_j(\sum_{l'\in S_4} X_{l'}\beta_{l'}), X_k(\sum_{l\in S_3} X_l\beta_l + \epsilon)] + \\
& Cov[X_j(\sum_{l\in S_3} X_l\beta_l + \epsilon), X_k(\sum_{l'\in S_4} X_{l'}\beta_{l'})] + Cov[X_j(\sum_{l'\in S_4} X_{l'}\beta_{l'}), X_k(\sum_{l'\in S_4} X_{l'}\beta_{l'})] \\
\approx \;& Cov[X_j(\sum_{l\in S_3} X_l\beta_l + \epsilon), X_k(\sum_{l\in S_3} X_l\beta_l + \epsilon)] \\
\approx \;& Cov[X_j\epsilon_{jk}, X_i\epsilon_{jk}] \\
\approx \;& E(X_jX_k)E(\epsilon_{jk}^2)
\end{aligned}
$$

where $E(X_jX_k)$ can be directly estimated by the LD correlation matrix and MAF-based SNP variance estimator. For $E(\epsilon_{jk}^2)$, it is the residual variance from a two-SNP regression model and should be smaller than both $E(\epsilon_j^2)$ and $E(\epsilon_k^2)$. In practice, we can approximate it by the smaller value between $\frac{\hat{\boldsymbol{\epsilon}}_j^T\hat{\boldsymbol{\epsilon}}_j}{N-1}$ and $\frac{\hat{\boldsymbol{\epsilon}}_k^T\hat{\boldsymbol{\epsilon}}_k}{N-1}$. Therefore, the numerical approximation for $\boldsymbol{\Sigma}_{jk}$ becomes

$$
\hat{\boldsymbol{\Sigma}}_{jk} = Nmin\left[\left(SE(\hat{\beta}_j)\hat{\sigma}_j, SE(\hat{\beta}_k)\hat{\sigma}_k\right)\right]^2 d_{jk}\hat{\sigma}_j\hat{\sigma}_k \qquad (5)
$$

Now we can then generate summary statistic from the multivariate normal distribution in formula *(3)*. Note that our earlier subsampling framework is a special case where SNPs are independent and its only difference with the current method is the estimation of $\hat{\boldsymbol{\Sigma}}_{jk}$. In the next session we will discuss how to subsample summary statistics efficiently from a multivariate normal distribution.

Strategy for subsampling summary statistics

Next, we discuss how to partition full GWAS summary statistics into K independent subsets of GWAS samples, denoted as $\mathbf{x}^{(1)^T}\mathbf{y}^{(1)}, \ldots, \mathbf{x}^{(K)^T}\mathbf{y}^{(K)}$ for $K > 2$. When $K = 2$, formula *(3)* can be directly applied to divide GWAS summary statistics into two independent sets. Otherwise, let $N^{(1)}, \ldots, N^{(K)}$ denote the corresponding sample size for each subset of individuals and $N = \sum_{s=1}^{K} N^{(s)}$. By formula *(3)*, we can subsample $\mathbf{x}^{(1)^T}\mathbf{y}^{(1)}$ from $\mathbf{x}^T\mathbf{y}$ observed in the complete GWAS summary data. After that, we calculate summary statistics excluding $N^{(1)}$ individuals from the first

subset as $\mathbf{x}^{(-1)^T}\mathbf{y}^{(-1)} = \mathbf{x}^T\mathbf{y} - \mathbf{x}^{(1)^T}\mathbf{y}^{(1)}$. To generate summary statistics for any following subset numbered $t+1$ (i.e., $\mathbf{x}^{(t+1)^T}\mathbf{y}^{(t+1)}$) for $t = 1, \dots, K-2$, we update the conditional distribution in *(3)* with the new "full" GWAS summary statistics and correspondent total sample size:

$$\mathbf{x}^{(t+1)^T}\mathbf{y}^{(t+1)} | \mathbf{x}^{(-t)^T}\mathbf{y}^{(-t)} \sim \mathbf{N}\left(\frac{N^{(t+1)}}{N - \sum_{s=1}^{t} N^{(s)}}\mathbf{x}^{(-t)^T}\mathbf{y}^{(-t)}, \frac{\left(N - \sum_{s=1}^{t+1} N^{(s)}\right)N^{(t+1)}}{N - \sum_{s=1}^{t} N^{(s)}}\widehat{\boldsymbol{\Sigma}}\right) \qquad (6)$$

where $\mathbf{x}^{(-t)^T}\mathbf{y}^{(-t)}$ represents summary statistics excluding first t subsets of individuals. This subsampling strategy guarantees that every subset is independent from each other and avoids overfitting when $K > 2$. Finally, for the last subset K, we can directly calculate its summary statistics by $\mathbf{x}^{(K)^T}\mathbf{y}^{(K)} = \mathbf{x}^T\mathbf{y} - \sum_{s=1}^{K-1}\mathbf{x}^{(s)^T}\mathbf{y}^{(s)}$. Together, this is a flexible framework for generating summary statistics and can be used for various types of PRS analyses as we discuss in later sections.

It is a difficult task to subsample summary statistics for all SNPs in the genome simultaneously given the large dimension of genotype and imputed data. Even if PRS modeling is restricted to HapMap3 SNPs, it remains challenging to subsample $\mathbf{x}^{(s)^T}\mathbf{y}^{(s)}$ for more than one million SNPs altogether[26]. To efficiently generate data, we partition the whole genome into approximately independent LD blocks and subsample summary statistics for SNPs in each LD block separately[56,57]. Then $\widehat{\boldsymbol{\Sigma}}$ becomes a sparse block-diagonal matrix, i.e., $\widehat{\boldsymbol{\Sigma}} = diag(\widehat{\boldsymbol{\Sigma}}_{D_i})$. Within each LD block, the empirical SNP correlation matrix may not always be positive-definite and thus making it impossible to randomly generate data from that LD block. A straightforward remedy is to conduct eigen decomposition for any $\widehat{\boldsymbol{\Sigma}}_{D_i}$that is negative definite, manually change negative eigenvalues to 0's, and obtain an approximation of $\widehat{\boldsymbol{\Sigma}}_{D_i}$ that is positive semi-definite. Note that this may not be the best approach and other methods for estimating LD blocks can also be applied[58,59].

Evaluate predictive performance of PRS

Here, we generalize the summary-statistics-based PRS evaluation scheme proposed in our previous work to incorporate LD. We denote PRS as a weighted sum of allele counts across many SNPs:

$$\hat{Y} = \mathbf{X}\boldsymbol{\omega}$$

where $\boldsymbol{\omega} \in \mathbb{R}^p$ is a vector of SNP weights, which can be marginal regression coefficients from GWAS or post-hoc effect size estimates. If individual-level data is available, then R^2 evaluated on any holdout dataset $\left(\mathbf{y}^{(s)}, \mathbf{x}^{(s)}\right)$ can be calculated as

$$R^2_{samples} = \frac{\left[Cov\left(\mathbf{y}^{(s)}, \hat{\mathbf{y}}^{(s)}\right)\right]^2}{Var(\mathbf{y}^{(s)})Var(\hat{\mathbf{y}}^{(s)})} = \frac{\left(\sum_{i=1}^{N^{(s)}} y_i^{(s)}\hat{y}_i{}^{(s)} - n\overline{\mathbf{y}^{(s)}}\ \overline{\hat{\mathbf{y}}^{(s)}}\right)^2}{\sum_{i=1}^{N^{(s)}}\left(y_i^{(s)} - \overline{\mathbf{y}^{(s)}}\right)^2 \sum_{i=1}^{N^{(s)}}\left(\hat{y}_i^{(s)} - \overline{\hat{\mathbf{y}}^{(s)}}\right)^2}$$

where $\hat{y}_i$ is the PRS for the i-th person, $\overline{\mathbf{y}^{(s)}}$ is the mean phenotypic value, and $\overline{\hat{\mathbf{y}}^{(s)}}$ is the mean PRS value in holdout dataset s. On the other hand, we have shown that when only summary statistics of the holdout dataset is available and SNPs are independent, $R^2_{samples}$ can be approximated by[35]:

$$\frac{1}{N^{(s)}}\sum_{i=1}^{N^{(s)}}\left(\hat{y}_i^{(s)}-\overline{\hat{\mathbf{y}}^{(s)}}\right)^2 \approx \sum\nolimits_{j=1}^{p} w_j^2\hat{\sigma}_j^2$$

$$\frac{1}{N^{(s)}}\sum_{i=1}^{N^{(s)}}\left(y_i^{(s)}-\overline{\mathbf{y}^{(s)}}\right)^2 \approx \max_j\left[\frac{\hat{\boldsymbol{\epsilon}}_j^{\top}\hat{\boldsymbol{\epsilon}}_j}{N-1}\right] \approx N\max_j\left[\mathrm{SE}\left(\hat{\beta}_j\right)^2\hat{\sigma}_j^2\right]$$

$$\hat{R}^2_{noLD} \approx \frac{\left(\frac{1}{N^{(s)}}\sum_{j=1}^{p} w_j\mathbf{x}_j^{(s)\top}\mathbf{y}^{(s)}\right)^2}{N\max_j\left[\mathrm{SE}\left(\hat{\beta}_j\right)^2\hat{\sigma}_j^2\right]\sum_{j=1}^{p} w_j^2\,\hat{\sigma}_j^2}$$

given that $\mathbf{x}^{(s)}$, $\mathbf{y}^{(s)}$, and $\hat{\mathbf{y}}^{(s)}$ are centered. In practice, we use the 90% quantile instead of $\max_j\left[\mathrm{SE}\left(\hat{\beta}_j\right)^2\hat{\sigma}_j^2\right]$ to get a robust estimate of $Var\left(\mathbf{y}^{(s)}\right)$. When LD is present, the approximations for $Cov\left(\mathbf{y}^{(s)},\hat{\mathbf{y}}^{(s)}\right)$ and $Var\left(\mathbf{y}^{(s)}\right)$ remain the same. For $Var\left(\hat{\mathbf{y}}^{(s)}\right)$, it can now be approximated by $\boldsymbol{\omega}^T Var\left(\mathbf{x}^{(s)}\right)\boldsymbol{\omega}$, with $Var\left(\mathbf{x}^{(s)}\right)$ estimated using the LD correlation matrix and MAF calculated from the reference panel. Taken together, we have

$$\hat{R}^2_{LD} = \frac{(\frac{1}{N^{(s)}}\sum_{j=1}^{p}\omega_j\,\mathbf{x}_j^{(s)^T}\mathbf{y}^{(s)})^2}{N\max_j[SE(\hat{\beta}_j)^2\,\hat{\sigma}_j^2](\sum_{j=1}^{p}\sum_{k\neq j} w_j^2\hat{\sigma}_j^2 + w_j w_k d_{jk}\hat{\sigma}_j\hat{\sigma}_k)} \qquad (7)$$

Note that similar versions of this formula have been tested and applied in the literature[22,37,38]. In practice, we can directly calculate PRS on the LD reference genotype data and use the sample variance of PRS to replace $\sum_{j=1}^{p}\sum_{k\neq j} w_j^2\hat{\sigma}_j^2 + w_j w_k d_{jk}\hat{\sigma}_j\hat{\sigma}_k$ for optimal computational efficiency.

The PUMAS framework

Given the flexible framework we introduced for subsampling GWAS summary data and evaluating PRS based on summary statistics, PUMAS becomes a special case where the entire GWAS summary-level data is partitioned into a training and a tuning dataset, denoted as $\mathbf{x}^{(tr)^T}\mathbf{y}^{(tr)}$ and $\mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)}$. PUMAS first draws $\mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)}$ from *(3)* and then calculates $\mathbf{x}^{(tr)^T}\mathbf{y}^{(tr)}$ by $\mathbf{x}^T\mathbf{y} - \mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)}$. For each SNP, the marginal effect size and its standard error from the training set can be calculated as

$$\hat{\beta}_j^{(tr)} = [N^{(tr)}\,\hat{\sigma}_j^2]^{-1}\mathbf{x}_j^{(tr)^T}\mathbf{y}^{(tr)}$$

$$SE(\hat{\beta}_j^{(tr)}) = \sqrt{\frac{N}{N^{(tr)}}}SE(\hat{\beta}_j)$$

Then these summary statistics from the training dataset can be used to train any PRS methods that use GWAS summary statistics as input. R^2 of the PRS model assessed on the fine-tuning dataset can be approximated by replacing $\mathbf{x}^{(s)^T}\mathbf{y}^{(s)}$ with $\mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)}$ and changing the corresponding sample size in formula *(7)*. This procedure can be repeated k times to implement a k-fold Monte Carlo cross-validation (MCCV) to select the best-performing tuning parameter. When there is a set of tuning parameters $\boldsymbol{\lambda}$ in a PRS framework, that is, $\hat{Y}(\lambda) = \mathbf{X}\boldsymbol{\omega}(\lambda)$, PUMAS chooses the optimal tuning parameter $\hat{\lambda}$ by

$$\hat{\lambda} = \underset{\lambda \in \boldsymbol{\lambda}}{\operatorname{argmax}} \bar{R}^2_{LD}(\lambda)$$

where $\bar{R}^2_{LD}$ denotes the mean $\hat{R}^2_{LD}$ across k-fold MCCV. This cross-validation technique also applies to models that are hyperparameter-free or fine-tuned in advance. When the goal is to pick the best PRS model among a total of M methods, the best model $\hat{m}$ can be selected by

$$\hat{m} = \underset{m=1,2,\ldots,M}{\operatorname{argmax}} \bar{R}^2_{LD}\left(m, \hat{\lambda}_m\right)$$

where $\hat{\lambda}_m$ is the besting tuning parameter for model m.

Combining multiple PRS with PUMA-CUBS

Next, we introduce PUMA-CUBS, an extension of PUMAS that applies ensemble learning to combine multiple PRS using GWAS summary statistics. To do this, PUMA-CUBS further partitions the full GWAS association results to 4 independent sets of summary statistics corresponding to training ($\mathbf{x}^{(tr)^T}\mathbf{y}^{(tr)}$), tuning ($\mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)}$), ensemble training ($\mathbf{x}^{(etr)^T}\mathbf{y}^{(etr)}$), and testing ($\mathbf{x}^{(t)^T}\mathbf{y}^{(t)}$) summary statistics. Using formula *(6)*, we subsample summary statistics iteratively and compute $\mathbf{x}^{(tr)^T}\mathbf{y}^{(tr)} = \mathbf{x}^T\mathbf{y} - \mathbf{x}^{(tn)^T}\mathbf{y}^{(tn)} - \mathbf{x}^{(etr)^T}\mathbf{y}^{(etr)} - \mathbf{x}^{(t)^T}\mathbf{y}^{(t)}$. Like PUMAS, PUMA-CUBS first conducts k-fold MCCV using training and tuning summary statistics to pick the best tuning parameter for each PRS method. Then, it trains each optimal PRS model's weight on the ensemble training data and evaluates the combined PRS on the testing summary statistics. A straightforward and intuitive way of combining PRS is through multiple linear regression. However, if individual-level genotype and phenotype data is not available, we cannot fit the regression in the conventional way. Below we illustrate how to calculate regression coefficients using summary-level data alone. We define the multiple linear regression model on the ensemble training dataset as:

$$Y^{(etr)} = \alpha_1 \times \hat{Y}_1^{(etr)} + \alpha_2 \times \hat{Y}_2^{(etr)} + \cdots + \alpha_q \times \hat{Y}_q^{(etr)} + \epsilon_{prs}$$

where $\boldsymbol{\alpha} = \left[\alpha_1\ \alpha_2\ \ldots\ \alpha_q\right]^T$ are PRS weights for q PRS methods. We also define

$$\mathbf{z} = \left[\hat{\mathbf{y}}_1^{(etr)}\ \hat{\mathbf{y}}_2^{(etr)}\ \ldots\ \hat{\mathbf{y}}_q^{(etr)}\right] = \mathbf{x}^{(etr)}\mathbf{W}$$

as the observed PRS matrix with dimension $N^{(etr)} \times q$, and $\mathbf{W} = \left[\mathbf{w}_1\ \mathbf{w}_2\ \ldots\ \mathbf{w}_q\right]$ are a $p \times q$ SNP weights matrix for p SNPs from q methods. To obtain the least squares estimator of $\boldsymbol{\alpha}$, that is $\hat{\boldsymbol{\alpha}} = \left(\mathbf{z}^{\mathrm{T}}\mathbf{z}\right)^{-1}\mathbf{z}^{\mathrm{T}}\mathbf{y}^{(etr)}$, we need to estimate $\mathbf{z}^{\mathrm{T}}\mathbf{z}$ and $\mathbf{z}^{\mathrm{T}}\mathbf{y}^{(etr)}$ separately. In fact, under the assumption that genotype and phenotype are both centered, we can show that

$$\mathbf{z}^T\mathbf{z} \approx N^{(etr)} \cdot \hat{\boldsymbol{\Sigma}}_{\mathbf{z}} \qquad (8)$$

$$\mathbf{z}^T\mathbf{y}^{(etr)} \approx \mathbf{W}^T\mathbf{x}^{(etr)^T}\mathbf{y}^{(etr)} \qquad (9)$$

where $\hat{\boldsymbol{\Sigma}}_{\mathbf{z}}$ is the empirical covariance matrix of the PRS matrix $\mathbf{z}$. In practice, we can estimate $\hat{\boldsymbol{\Sigma}}_{\mathbf{z}}$ by calculating PRS and its sample variance on a reference LD genotype dataset or approximate it by computing $\mathbf{W}^T\mathbf{D}\mathbf{W}$. Taken *(8)* and *(9)* together, we can estimate PRS weights using only summary statistics. Then we take the average PRS weights across k folds, i.e., $\bar{\boldsymbol{\alpha}} = \frac{1}{k}\sum_{j=1}^{k}\hat{\boldsymbol{\alpha}}_j$, and use it as the PRS weight to combine optimized PRSs. Finally, we modify equation *(7)* to calculate predictive R^2 for ensemble PRS on the testing summary-level data:

$$R^2_{ensemble} = \frac{\left(\sum_{i=1}^{N^{(t)}} y_i^{(t)} \hat{y}_i{}^{(t)} - n\overline{\mathbf{y}^{(t)}}\,\overline{\hat{\mathbf{y}}^{(t)}}\right)^2}{\sum_{i=1}^{N^{(t)}}\left(y_i^{(t)} - \overline{\mathbf{y}^{(t)}}\right)^2 \sum_{i=1}^{N^{(t)}}\left(\hat{y}_i^{(t)} - \overline{\hat{\mathbf{y}}^{(t)}}\right)^2} \approx \frac{\left[\frac{1}{N^{(t)}}\overline{\boldsymbol{\alpha}}^T \mathbf{W}^T \mathbf{x}^{(t)^T} \mathbf{y}^{(t)}\right]^2}{N \max_j [SE(\hat{\beta}_j)^2\, \hat{\sigma}_j^2][\overline{\boldsymbol{\alpha}}^T \hat{\boldsymbol{\Sigma}}_{\mathbf{z}} \overline{\boldsymbol{\alpha}}]} \quad (10)$$

In the end, PUMA-CUBS reports the average prediction accuracy of ensemble PRS across k folds. Note that PUMA-CUBS can benchmark all PRS models in addition to the ensemble PRS on the testing summary statistics since it is independent from training and tuning datasets. Therefore, PUMA-CUBS becomes a highly flexible framework to train, fine-tune, combine, and evaluate PRS models based on GWAS summary statistics.

Binary phenotypes

There are two challenges when applying PUMAS and PUMA-CUBS to binary phenotypes. First, summary statistics obtained from logistic regression frameworks violate the linear regression model assumption in our derivation. Therefore equations *(7)* and *(10)* are not directly appliable to subsampling summary statistics for binary traits because $\mathbf{X}^T Y$ calculation is non-trivial for log odds ratios. Second, squared Pearson correlation between a binary outcome and PRS using logistic regression coefficients as input is uninterpretable and rarely reported. On the other hand, area under the ROC curve (AUC) is often the preferred metric to quantify PRS accuracy for binary outcome. AUC calculation based on summary statistics has been developed but is not yet generalized to handle whole genome data, making it difficult to evaluate more sophisticated PRS methods that leverage contributions from millions of SNPs when individual-level data is not accessible[60]. Here we propose a simple solution that allows us to apply PUMAS and PUMA-CUBS to binary phenotypes and report interpretable R^2. For binary traits, R^2 on the observed scale (i.e., R^2_{obs}) has been defined and discussed in the literature as an alternative metric for evaluating PRS prediction accuracy[44]. R^2_{obs} is the squared correlation between PRS and 0-1 status where PRS uses effect sizes estimated from linear probability model (LPM, i.e., linear regression between the binary response and SNP allele counts) as inputs[61]. If GWAS summary-level data is acquired from linear probability model, then PUMAS and PUMA-CUBS can be directly applied to calculate R^2_{obs} for binary traits[53]. When LPM summary statistics is not available, since a single SNP has very weak effect on the phenotypic outcome in practice, we can still safely approximate LMP coefficient estimations using Z-score from logistic regression between SNP and binary phenotype[43,62]. Specifically, we can calculate $\hat{\beta}_{j,LPM} \approx Z_{j,logistic} \times \sqrt{\frac{v(1-v)}{X_j^T X_j}}$ where $Z_{j,logistic}$ is Z-score for the j-th SNP from logistic summary statistics and v is the sample prevalence. Then, we can use $\hat{\beta}_{j,LPM}$ and correspondent standard error $SE(\hat{\beta}_{j,LPM}) \approx \sqrt{\frac{v(1-v)}{X_j^T X_j}}$ to apply PUMAS and PUMA-CUBS to dichotomous phenotypes. Eventually, if it is preferred to transform R^2_{obs} to R^2 on the liability scale ($R^2_{liability}$) which can be comparable across different studies and phenotypes, such transformation has been developed using sample and population prevalence[44,63].

Sample size imputation

In this section, we discuss how to handle sample size misspecification in GWAS summary statistics when applying our approach. Sample size misspecification is common in published GWAS datasets since many studies often do not report SNP-specific sample size in summary-level data and only provides a maximum sample size for the entire study. This is sub-optimal for

PRS training if variant-level samples sizes differ substantially (e.g., in meta-analysis). A recent study has extensively investigated sample size misspecification in marginal association statistics and observed consistently decreased PRS prediction accuracy when the issue is not properly addressed[31]. For PUMAS and PUMA-CUBS, incorrect sample sizes will both affect the quality of subsampled summary statistics and bias the estimation of predictive R^2. To address this issue, we employed the approach proposed in Prive et al. to impute and conduct quality control on variant-specific sample size[31]. Specifically, when the summary-level data does not provide sample size information for each SNP, we first impute sample size and remove SNPs with imputed sample size smaller than 70% and larger than 110% of reported maximum sample size. For summary statistics that provides per-SNP sample sizes, we simply removed variants with sample size smaller than 70% of the largest sample size. On the other hand, to make sure formula *(7)* and *(10)* work for summary statistics with varying SNP-specific sample sizes, we enforce all summary statistics other than training summary statistics to be strictly rectangular with the same sample size for every SNP. We achieve this by subsampling all other summary statistics first where we can specify subset size and calculate $\mathbf{x}^{(tr)^T}\mathbf{y}^{(tr)}$ at last.

PRS training

We trained lassosum, PRS-CS, and LDpred2 models for all PRS analyses in this study[22,25,26]. lassosum is a penalized regression framework that trains lasso regression coefficients for SNPs in each LD blocks with tuning parameters s and λ, where s controls the sparsity of LD matrix and λ is the penalty term that regularizes shrinkage of effect sizes. PRS-CS and LDpred2 are both Bayesian PRS frameworks with different prior assumptions for the SNP effect size distribution. PRS-CS has a global shrinkage parameter ϕ that uniformly shrinks its continuous prior distribution for each SNP and includes a full Bayesian approach that automatically learns ϕ during model fitting. LDpred2 is an extension of LDpred that places a point normal prior on SNP effects based on tuning parameter p that represents the proportion of causal variants in the genome or a univariate normal prior on all SNPs that doesn't require model-tuning (LDpred/LDpred2-Inf)[12]. Like PRS-CS, LDpred2 can also employ an empirical Bayesian approach to optimize p on the training summary statistics. For implementation, we trained PRS-CS SNP effect sizes (v1.0.0) with the default prior distribution. We followed PGS server pipeline to implement lassosum (R package 'lassosum' v0.4.5) and LDpred2 (R package 'bigsnpr' v1.9.11)[28,64]. Due to the large computational burden, we implemented LDpred2 on each chromosome separately and only used the estimated heritability from LD-score regression as the tuning parameter h^2 in LDpred2[45]. For real data analysis in UKB we constructed LDpred2 models with both non-sparse and sparse versions of posterior effect sizes. We only trained PRS models on HapMap3 SNPs in all analyses throughout this study. The best tuning parameter for lasso was obtained through grid search. For LDpred2 and PRS-CS, we compared grid search with empirical Bayesian models to find the best parameter.

Simulation settings

We conducted simulations using UKB genotype data imputed to the Haplotype Reference Consortium reference. We removed samples who are not of European ancestry and genetic variants with minor allele frequency below 0.01, imputation R^2 below 0.9, Hardy-Weinberg equilibrium test p-value below 1e-6, or missing genotype call rate greater than 2%. We further extracted variants in HapMap3 SNP list and 1000 Genome Project Phase III LD reference data for European ancestry from PRS-CS. 377,509 samples and 944,547 variants remained after

quality control. Then, we randomly selected 100,000 samples to form the training dataset and 1,000 samples as the LD genotype reference for our summary-statistics-based approach. To generate trait values, we simulated true effect sizes from a widely used point normal distribution, i.e., $\beta_j \sim (1-p)\delta_0 + p\mathrm{N}(0, \frac{h^2}{Mp})$ where p is the proportion of causal variants, δ_0 is point mass at 0, h^2 is the total heritability of the phenotype, and M is the total number of SNPs[7,12]. We did not simulate associations between SNP true effects on the allelic scale and minor allele frequency since previous analysis has shown minimal difference in performance between PUMAS and PRS validation using individual-level data[35,65]. We chose p to be 0.1% and 20% that correspond to a sparse and a more polygenic genetic model, and $h^2 = 0.2, 0.5, 0.8$ to create a total of 6 simulation settings with various genetic architecture. Within each setting, we randomly selected and shuffled causal variants to be distributed across the whole genome. Then we simulated quantitative traits by adding up the SNP allele counts weighted by their true effect sizes and randomly generated gaussian noises scaled based on trait heritability. We fitted marginal linear regression in PLINK to obtain GWAS summary statistics in each setting[66].

We compared PUMAS with 4-fold MCCV. To implement MCCV, in each fold we randomly selected 60% of all samples to form the training dataset (N=60,000) and 20% as the tuning dataset (N=20,000). We conducted GWAS on the training data and used summary statistics to train PRS models and evaluated each PRS model's predictive R^2 on the tuning data. For PUMAS, we used all samples (N=100,000) to fit marginal linear regression and obtained the full summary statistics. In a similar fashion, we partitioned the full summary statistics to training summary data (N=60,000) for PRS training and tuning summary data (N=20,000) for PRS fine-tuning. The remaining summary-level data (N=20,000) was reserved for PUMA-CUBS simulation. 1000 Genomes Project EUR data provided by PRS-CS software were used for subsampling summary statistics. The holdout UKB LD data (N=1,000) was used as the LD reference in lassosum and LDpred2, while PRS-CS used its pre-calculated UKB LD matrices. We implemented lassosum with $s = 0.2, 0.5, 0.9$ and $\lambda = 0.005, 0.01$, PRS-CS with $\phi = 0.0001, 0.01, auto$, LDpred2 with $p = 0.001, 0.01, 0.1, auto$ and the infinitesimal model. We repeated this procedure 4 times and calculate average R^2 to pick the best set of tuning parameters for both approaches.

We also compared PUMA-CUBS with 4-fold MCCV that involves construction of ensemble PRS. For MCCV, in each fold we first used the same training dataset (N=60,000) and tuning dataset (N=20,000) to select the best PRS model for each method. Then, we randomly split the remaining 20,000 samples into ensemble training (N=10,000) and testing (N=10,000) datasets. We fitted multiple linear regression and obtained regression coefficients for best PRS models on the ensemble training dataset and computed PRS prediction accuracy on the testing dataset. For PUMA-CUBS, we further partitioned the reserved summary statistics (N=20,000) into ensemble training summary data (N=10,000) and testing summary data (N=10,000) for regression fitting and assessing predictive performance, respectively. This procedure was repeated 4 times for both approaches and we reported average R^2 for each PRS model.

We conducted additional simulations to demonstrate that PUMAS and PUMA-CUBS can be applied to binary traits. For each setting in the quantitative simulation study, we dichotomized the continuous phenotype (i.e., true liability value under a liability threshold model[63]) using either the median or 90% quantile to acquire a balanced setting (5-to-5) and an unbalanced setting (1-to-9). Therefore, we have a total of 12 binary simulation settings. We fitted logistic regressions in PLINK to obtain GWAS summary statistics in each setting and transformed logistic regression summary statistics to the linear scale. We then compared PUMAS and PUMA-CUBS with MCCV and MCCV-liability. For PUMAS, PUMA-CUBS and MCCV, we computed R^2 on the observed scale (i.e., R^2 between PRS and 0-1 status) and transformed it to R^2 on the liability scale by[44]:

$$R^2_{liability} = \frac{v(1-v)}{[\phi(\Phi^{-1}(1-v))]^2} R^2_{obs}$$

where v is sample prevalence, ϕ and Φ^{-1} are the pdf and inverse cdf of the standard normal distribution. For MCCV-liability, we directly calculated R^2 on the liability scale using true liability values.

UKB data analysis

We applied our approach to 16 quantitative traits and 3 diseases in UKB. The list of UKB phenotypes is presented in **Supplementary Tables 5-6**. The imputed UKB genotype data consists of 375,064 independent individuals of European ancestry and 1,030,187 variants after quality control. We used Hail (v0.2.57) to perform linear regression for quantitative traits while adjusting for sex, age polynomials to the power of two, interactions between sex and age polynomials, and top 20 principal components[67]. For 3 disease outcomes, we obtained GWAS summary statistics via regenie (v3.0.3) accounting for sex, age polynomials to the power of 3, interactions between sex and age polynomials, and top 10 principal components as recommended[68].

We compared PUMAS with external validation using a subset of UKB samples as the holdout dataset. For external validation of quantitative traits, we randomly selected 38,521 samples with non-missing phenotypic measurements for all traits as the holdout dataset and the remaining samples for each phenotype were used as training data. In this way, we implemented an approximate 9-to-1 training-testing split. Similarly for each binary outcome, we continued to employ a 9-to-1 sample partition while matching the case-control ratio between the training and holdout datasets. Detailed sample size information for each dataset and each trait is presented in **Supplementary Tables 5-6**. We further randomly picked 1000 samples from the holdout set of quantitative traits to create an LD reference data. Then, we conducted GWAS on the training data and obtained summary statistics for HapMap 3 SNPs. For quantitative traits, we computed and evaluated PRS models on the entire holdout set and reported predictive R^2 between PRS and phenotypes with covariates regressed out. For disease traits, we used both linear probability model summary statistics and logistic model summary statistics to train PRS models and calculated R^2 on the observed scale. For comparison, we applied PUMAS to partition the same GWAS summary-level data to a 75% training set and 25% tuning set. We computed approximately independent LD blocks on the holdout dataset for summary statistics subsampling and used the same LD reference panel in external validation for PRS training and evaluation[56]. This procedure was repeated 4 times and we reported the average R^2 for each PRS model. For both approaches, we implemented lassosum with $s = 0.2, 0.5, 0.9$ and $\lambda = 0.005, 0.01,$ PRS-CS with $\phi = 0.0001, 0.01, auto$, LDpred2 with $p = 0.001, 0.01, 0.1, auto$ and the infinitesimal model.

Next, we compared PUMA-CUBS with training-testing split on the holdout dataset for construction of ensemble PRS for each of 16 quantitative traits in UKB. For PUMA-CUBS, we partitioned full GWAS summary statistics into training (60%), tuning (20%), and ensemble training (10%) summary statistics to train PRS models based on a grid of tuning parameters, select the best tuning parameter setting for each PRS method, and fit a second level regression to obtain regression weights for fine-tuned PRS models. We then randomly partitioned the holdout dataset into two equally sized subsets. We used PUMA-CUBS to obtain PRS models' regression weights and then constructed and evaluated the ensemble PRS on the second half of the holdout set. PRS models with negative weights were removed from linear combination. In comparison, for the training-testing split based on individual-level data, we used the first half of the holdout set to fit

multiple linear regression to obtain regression coefficients for fine-tuned lassosum, LDpred2, and PRScs scores. Then we computed and evaluated the ensemble PRS models on the second half of the holdout data. For both PUMA-CUBS and training-testing split, we trained lassosum with $s = 0.2, 0.9$ and $\lambda = 0.001, 0.01, 0.1$, PRS-CS with $\phi = 0.0001, 0.01, auto$, LDpred2 with $p = 0.001, 0.01, 0.1, auto$ and the infinitesimal model.

Building a catalog of PUMA-CUBS ensemble scores

We applied PUMA-CUBS to a collection of publicly available GWAS summary statistics. We selected complex diseases and traits that have a minimal case sample size of 5,000 and total sample size of 50,000 respectively and have significant heritability (p-value below 0.05) from LD score regression[45]. We excluded studies that performed GWAS on related samples. We obtained a list of 31 GWAS summary statistics including 23 binary outcomes and 8 complex traits as summarized in **Supplementary Table 10**. For each summary statistics, we kept HapMap 3 SNPs that passed a series of quality control criteria listed in **Supplementary Table 11**, including transformation of logistic summary statistics and imputation of per-SNP sample size. Then we applied PUMA-CUBS to each phenotype to implement 4-fold MCCV by partitioning the summary statistics to training (60%), tuning (20%), ensemble training (10%), and testing (10%) datasets. We used 1000 Genomes Project Phase III European samples as the LD panel for summary statistics subsampling, PRS model fitting and benchmarking. We implemented lassosum with $s = 0.2, 0.5, 0.9$ and $\lambda = 0.005, 0.01$, PRS-CS with $\phi = 0.0001, 0.01, auto$, LDpred2 with $p = 0.001, 0.01, 0.1, auto$ and the infinitesimal model. We reported average predictive R^2 of ensemble PRS, the best single PRS model, and PRS-CS-auto on the testing summary statistics.

We conducted additional analysis to investigate the validity of predictive R^2 of ensemble PRS on Alzheimer's disease. We used IGAP 2019 Alzheimer's GWAS summary statistics as input, and 2,600 Alzheimer's disease cases of European ancestry in the UKB cohort as external validation dataset[46]. The data fields used for Alzheimer's cases extraction are presented in **Supplementary Table 13**. We randomly selected 5,200 independent UKB samples not diagnosed with Alzheimer's disease to use as healthy controls to match the case-control ratio in the IGAP 2019 study. Together, we obtained a UKB external validation dataset with 7,800 samples in total. We applied PUMAS to IGAP 2019 GWAS summary-level data and compared its performance with external validation. We compared R^2 from both approaches with and without removing the *APOE* region from GWAS summary statistics. We excluded the *APOE* region from PRS analysis by removing variants between base pairs 45,116,911 and 46,318,605 (hg19) on chromosome 19.

Data and code availability

PUMAS/PUMA-CUBS software is freely available at https://github.com/qlu-lab/PUMAS.

Author Contribution

Z.Z. and Q.L. conceived and design the study.
Z.Z. developed the statistical framework.
Z.Z. and T.G. performed statistical analyses.
Z.Z. and Y.W. wrote the software.
S.Z. assisted in preparing and curating summary statistics.

J.M. assisted in developing ensemble PRS approach.
J.M. and Y.W. assisted in UKB data preparation.
J.S. assisted in developing statistical method for subsampling summary statistics.
Q.L. advised on statistical and genetic issues.
Z.Z. and Q.L. wrote the manuscript.
All authors contributed in manuscript editing and approved the manuscript.

Acknowledgements

The authors gratefully acknowledge research support from National Institutes of Health (NIH) grants U01 HG012039 and R21 AG067092, and support from the University of Wisconsin-Madison Office of the Chancellor and the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation (WARF). We also acknowledge use of the facilities of the Center for Demography of Health and Aging at the University of Wisconsin-Madison, funded by NIA Center Grant P30 AG017266. We thank members of the Social Genomics Working Group at University of Wisconsin for helpful comments. This research has been conducted using the UK Biobank Resource under Application 42148.

References

1. Torkamani, A., Wineinger, N.E. & Topol, E.J. The personal and clinical utility of polygenic risk scores. *Nat Rev Genet* **19**, 581-590 (2018).
2. Chatterjee, N., Shi, J. & Garcia-Closas, M. Developing and evaluating polygenic risk prediction models for stratified disease prevention. *Nat Rev Genet* **17**, 392-406 (2016).
3. Lewis, C.M. & Vassos, E. Polygenic risk scores: from research tools to clinical instruments. *Genome Med* **12**, 44 (2020).
4. Wray, N.R., Goddard, M.E. & Visscher, P.M. Prediction of individual genetic risk to disease from genome-wide association studies. *Genome Res* **17**, 1520-8 (2007).
5. International Schizophrenia, C. *et al.* Common polygenic variation contributes to risk of schizophrenia and bipolar disorder. *Nature* **460**, 748-52 (2009).
6. Yang, J. *et al.* Common SNPs explain a large proportion of the heritability for human height. *Nat Genet* **42**, 565-9 (2010).
7. Zhou, X., Carbonetto, P. & Stephens, M. Polygenic modeling with bayesian sparse linear mixed models. *PLoS Genet* **9**, e1003264 (2013).
8. Minnier, J., Yuan, M., Liu, J.S. & Cai, T. Risk Classification with an Adaptive Naive Bayes Kernel Machine Model. *J Am Stat Assoc* **110**, 393-404 (2015).
9. Wei, Z. *et al.* Large sample size, wide variant spectrum, and advanced machine-learning technique boost risk prediction for inflammatory bowel disease. *Am J Hum Genet* **92**, 1008-12 (2013).
10. Speed, D. & Balding, D.J. MultiBLUP: improved SNP-based prediction for complex traits. *Genome Res* **24**, 1550-7 (2014).
11. Wray, N.R. *et al.* Pitfalls of predicting complex traits from SNPs. *Nat Rev Genet* **14**, 507-15 (2013).
12. Vilhjálmsson, B.J. *et al.* Modeling Linkage Disequilibrium Increases Accuracy of Polygenic Risk Scores. *Am J Hum Genet* **97**, 576-92 (2015).
13. Hu, Y. *et al.* Leveraging functional annotations in genetic risk prediction for human complex diseases. *PLoS Comput Biol* **13**, e1005589 (2017).
14. Márquez-Luna, C. *et al.* Incorporating functional priors improves polygenic prediction accuracy in UK Biobank and 23andMe data sets. *Nature Communications* **12**, 6052 (2021).
15. Hu, Y. *et al.* Joint modeling of genetically correlated diseases and functional annotations increases accuracy of polygenic risk prediction. *PLoS Genet* **13**, e1006836 (2017).
16. Maier, R.M. *et al.* Improving genetic prediction by leveraging genetic correlations among human diseases and traits. *Nat Commun* **9**, 989 (2018).
17. Turley, P. *et al.* Multi-trait analysis of genome-wide association summary statistics using MTAG. *Nat Genet* **50**, 229-237 (2018).
18. Khera, A.V. *et al.* Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations. *Nat Genet* **50**, 1219-1224 (2018).
19. Meisner, A. *et al.* Combined Utility of 25 Disease and Risk Factor Polygenic Risk Scores for Stratifying Risk of All-Cause Mortality. *Am J Hum Genet* **107**, 418-431 (2020).
20. Hao, L. *et al.* Development of a clinical polygenic risk score assay and reporting workflow. *Nature Medicine* **28**, 1006-1013 (2022).
21. Kulm, S., Marderstein, A., Mezey, J. & Elemento, O. A systematic framework for assessing the clinical impact of polygenic risk scores. *medRxiv*, 2020.04.06.20055574 (2021).
22. Mak, T.S.H., Porsch, R.M., Choi, S.W., Zhou, X. & Sham, P.C. Polygenic scores via penalized regression on summary statistics. *Genet Epidemiol* **41**, 469-480 (2017).

23. Chen, T.-H., Chatterjee, N., Landi, M.T. & Shi, J. A penalized regression framework for building polygenic risk models based on summary statistics from genome-wide association studies and incorporating external information. *Journal of the American Statistical Association*, 1-19 (2020).
24. Chung, W. *et al.* Efficient cross-trait penalized regression increases prediction accuracy in large cohorts using secondary phenotypes. *Nat Commun* **10**, 569 (2019).
25. Privé, F., Arbel, J. & Vilhjálmsson, B.J. LDpred2: better, faster, stronger. *Bioinformatics* **36**, 5424-5431 (2020).
26. Ge, T., Chen, C.Y., Ni, Y., Feng, Y.A. & Smoller, J.W. Polygenic prediction via Bayesian regression and continuous shrinkage priors. *Nat Commun* **10**, 1776 (2019).
27. Lloyd-Jones, L.R. *et al.* Improved polygenic prediction by Bayesian multiple regression on summary statistics. *Nat Commun* **10**, 5086 (2019).
28. Yang, S. & Zhou, X. PGS-server: accuracy, robustness and transferability of polygenic score methods for biobank scale studies. *Brief Bioinform* **23**(2022).
29. Pain, O. *et al.* Evaluation of polygenic prediction methodology within a reference-standardized framework. *PLOS Genetics* **17**, e1009021 (2021).
30. Wang, Y., Tsuo, K., Kanai, M., Neale, B.M. & Martin, A.R. Challenges and Opportunities for Developing More Generalizable Polygenic Risk Scores.
31. Privé, F., Arbel, J., Aschard, H. & Vilhjálmsson, B.J. Identifying and correcting for misspecifications in GWAS summary statistics and polygenic scores. *Human Genetics and Genomics Advances* **3**, 100136 (2022).
32. Ni, G. *et al.* A Comparison of Ten Polygenic Score Methods for Psychiatric Disorders Applied Across Multiple Cohorts.
33. Ruan, Y. *et al.* Improving polygenic prediction in ancestrally diverse populations. *Nature Genetics* **54**, 573-580 (2022).
34. Ma, Y. & Zhou, X. Genetic prediction of complex traits with polygenic scores: a statistical review. *Trends in Genetics* **37**, 995-1011 (2021).
35. Zhao, Z. *et al.* PUMAS: fine-tuning polygenic risk scores with GWAS summary statistics. *Genome Biol* **22**, 257 (2021).
36. Picard, R.R. & Cook, R.D. Cross-Validation of Regression Models. *Journal of the American Statistical Association* **79**, 575-583 (1984).
37. Zhang, Q., Privé, F., Vilhjálmsson, B. & Speed, D. Improved genetic prediction of complex traits from individual-level data or summary statistics. *Nature Communications* **12**, 4192 (2021).
38. Yang, S. & Zhou, X. Accurate and Scalable Construction of Polygenic Scores in Large Biobank Data Sets. *Am J Hum Genet* **106**, 679-693 (2020).
39. Miao, J. *et al.* Quantifying portable genetic effects and improving cross-ancestry genetic prediction with GWAS summary statistics. *bioRxiv*, 2022.05.26.493528 (2022).
40. Yengo, L. *et al.* A saturated map of common genetic variants associated with human height. *Nature* (2022).
41. Vuckovic, D. *et al.* The Polygenic and Monogenic Basis of Blood Traits and Diseases. *Cell* **182**, 1214-1231.e11 (2020).
42. Grotzinger, A.D. *et al.* Genomic structural equation modelling provides insights into the multivariate genetic architecture of complex traits. *Nature Human Behaviour* **3**, 513-525 (2019).
43. Lloyd-Jones, L.R., Robinson, M.R., Yang, J. & Visscher, P.M. Transformation of Summary Statistics from Linear Mixed Model Association on All-or-None Traits to Odds Ratio. *Genetics* **208**, 1397-1408 (2018).
44. Lee, S.H., Goddard, M.E., Wray, N.R. & Visscher, P.M. A Better Coefficient of Determination for Genetic Profile Analysis. *Genetic Epidemiology* **36**, 214-224 (2012).

45. Bulik-Sullivan, B.K. *et al.* LD Score regression distinguishes confounding from polygenicity in genome-wide association studies. *Nat Genet* **47**, 291-5 (2015).
46. Kunkle, B.W. *et al.* Genetic meta-analysis of diagnosed Alzheimer's disease identifies new risk loci and implicates Abeta, tau, immunity and lipid processing. *Nat Genet* **51**, 414-430 (2019).
47. Corder, E.H. *et al.* Gene Dose of Apolipoprotein E Type 4 Allele and the Risk of Alzheimer's Disease in Late Onset Families. *Science* **261**, 921-923 (1993).
48. Bellenguez, C. *et al.* New insights into the genetic etiology of Alzheimer's disease and related dementias. *Nature Genetics* **54**, 412-436 (2022).
49. de Rojas, I. *et al.* Common variants in Alzheimer's disease and risk stratification by polygenic risk scores. *Nature Communications* **12**, 3417 (2021).
50. Martin, A.R. *et al.* Human Demographic History Impacts Genetic Risk Prediction across Diverse Populations. *Am J Hum Genet* **100**, 635-649 (2017).
51. Martin, A.R. *et al.* Clinical use of current polygenic risk scores may exacerbate health disparities. *Nat Genet* **51**, 584-591 (2019).
52. Zhou, W. *et al.* Efficiently controlling for case-control imbalance and sample relatedness in large-scale genetic association studies. *Nat Genet* **50**, 1335-1341 (2018).
53. Loh, P.-R. *et al.* Efficient Bayesian mixed-model analysis increases association power in large cohorts. *Nature Genetics* **47**, 284-290 (2015).
54. Truong, B. *et al.* Efficient polygenic risk scores for biobank scale data by exploiting phenotypes from inferred relatives. *Nat Commun* **11**, 3074 (2020).
55. Albiñana, C. *et al.* Multi-PGS enhances polygenic prediction: weighting 937 polygenic scores. *medRxiv*, 2022.09.14.22279940 (2022).
56. Berisa, T. & Pickrell, J.K. Approximately independent linkage disequilibrium blocks in human populations. *Bioinformatics* **32**, 283-5 (2016).
57. Zhang, Y. *et al.* SUPERGNOVA: local genetic correlation analysis reveals heterogeneous etiologic sharing of complex traits. *Genome Biology* **22**, 262 (2021).
58. Spence, J.P., Sinnott-Armstrong, N., Assimes, T.L. & Pritchard, J.K. A flexible modeling and inference framework for estimating variant effect sizes from GWAS summary statistics. *bioRxiv*, 2022.04.18.488696 (2022).
59. Xiang, Z. & Matthew, S. Bayesian large-scale multiple regression with summary statistics from genome-wide association studies. *The Annals of Applied Statistics* **11**, 1561-1592 (2017).
60. Song, L. *et al.* SummaryAUC: a tool for evaluating the performance of polygenic risk prediction models in validation datasets with only summary level statistics. *Bioinformatics* (2019).
61. Amemiya, T. Some Theorems in the Linear Probability Model. *International Economic Review* **18**, 645-650 (1977).
62. Grotzinger, A.D. *et al.* Genomic structural equation modelling provides insights into the multivariate genetic architecture of complex traits. **3**, 513-525 (2019).
63. Neale, B. Liability Threshold Models. *Wiley StatsRef: Statistics Reference Online* (2014).
64. Privé, F., Aschard, H., Ziyatdinov, A. & Blum, M.G.B. Efficient analysis of large-scale genome-wide data with two R packages: bigstatsr and bigsnpr. *Bioinformatics* **34**, 2781-2787 (2018).
65. Speed, D., Hemani, G., Johnson, M.R. & Balding, D.J. Improved heritability estimation from genome-wide SNPs. *American journal of human genetics* **91**, 1011-1021 (2012).
66. Purcell, S. *et al.* PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am J Hum Genet* **81**, 559-75 (2007).
67. Team, H. Hail 0.2.57.
68. Mbatchou, J. *et al.* Computationally efficient whole-genome regression for quantitative and binary traits. *Nature Genetics* **53**, 1097-1103 (2021).