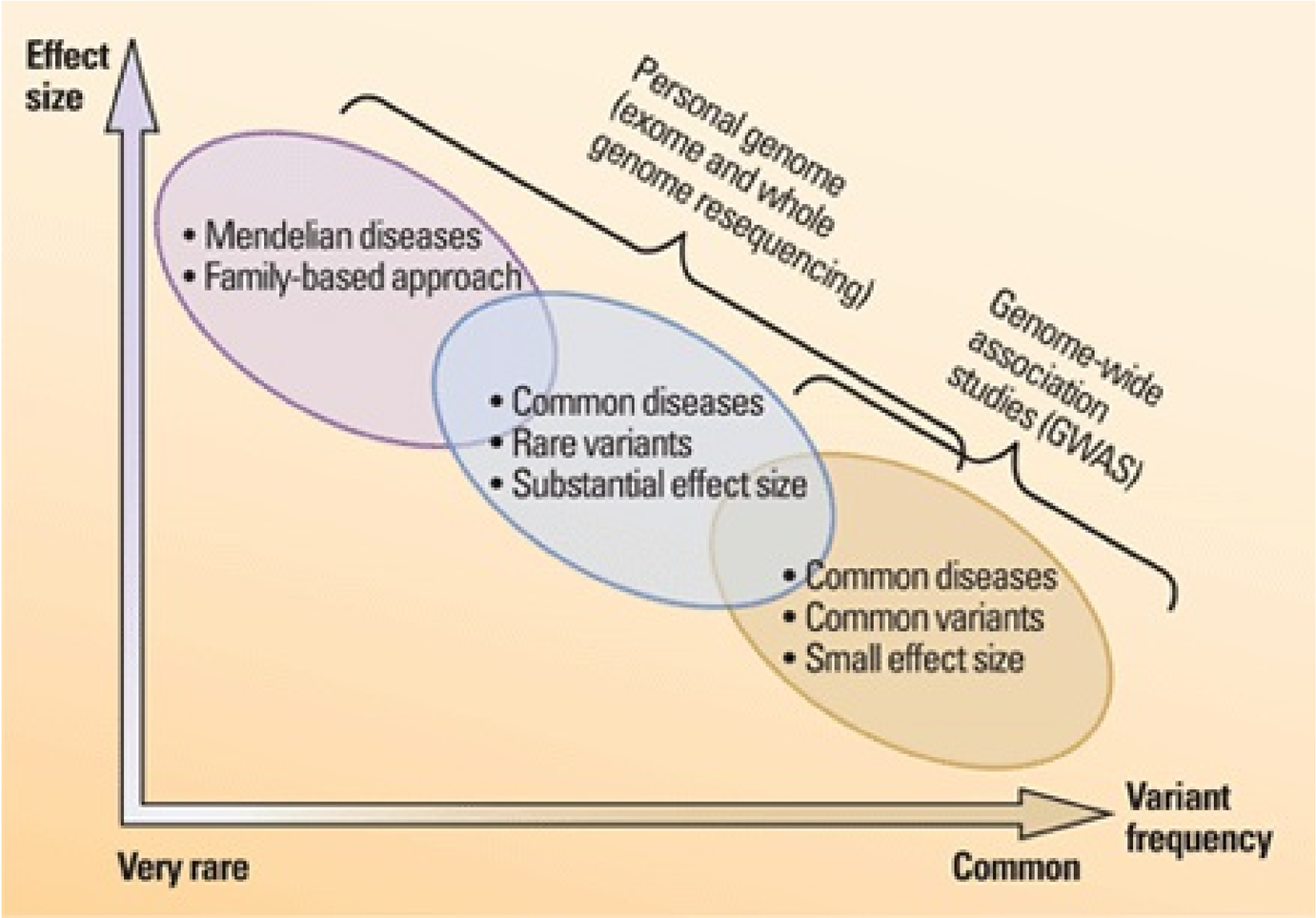
Predicting complex diseases: performance and robustness

Florian Privé, Hugues Aschard, Michael G.B. Blum

RECOMB-Genetics 2018

Introduction

Disease architectures



Source: 10.1126/science.338.6110.1016

How to predict common diseases based on common variants with small effects?

Interest in prediction: polygenic risk scores (PRS)

- Wray, Naomi R., Michael E. Goddard, and Peter M. Visscher. "**Prediction of individual genetic risk** to disease from genome-wide association studies." Genome research 17.10 (**2007**): 1520-1528.

- Evans, David M., Peter M. Visscher, and Naomi R. Wray. "Harnessing the information contained within genome-wide association studies to improve **individual prediction of complex disease risk**." Human molecular genetics 18.18 (**2009**): 3525-3531.

- Wray, Naomi R., et al. "Pitfalls of **predicting complex traits** from SNPs." Nature Reviews Genetics 14.7 (**2013**): 507.

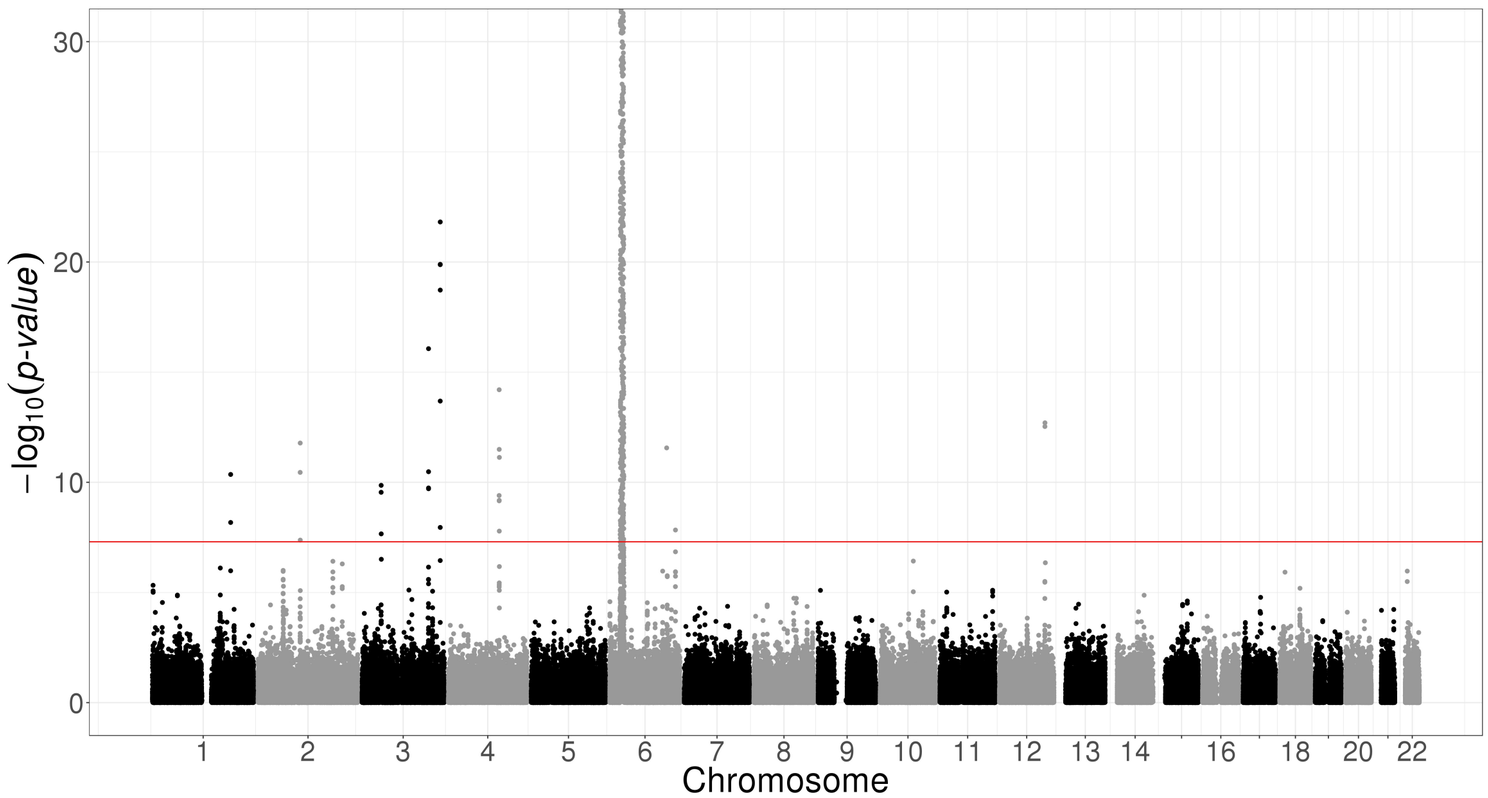
- Dudbridge, Frank. "Power and **predictive accuracy of polygenic risk scores**." PLoS genetics 9.3 (**2013**): e1003348.

- Chatterjee, Nilanjan, Jianxin Shi, and Montserrat García-Closas. "Developing and evaluating **polygenic risk prediction** models for stratified disease prevention." Nature Reviews Genetics 17.7 (**2016**): 392.

- Martin, Alicia R., et al. "Human demographic history impacts **genetic risk prediction** across diverse populations." The American Journal of Human Genetics 100.4 (**2017**): 635-649.

Genome-wide association studies (GWAS)

In case-control studies, a GWAS test each single-nucleotide polymorphism (SNP) **independently**, computing an **effect size** β and a **p-value** p.

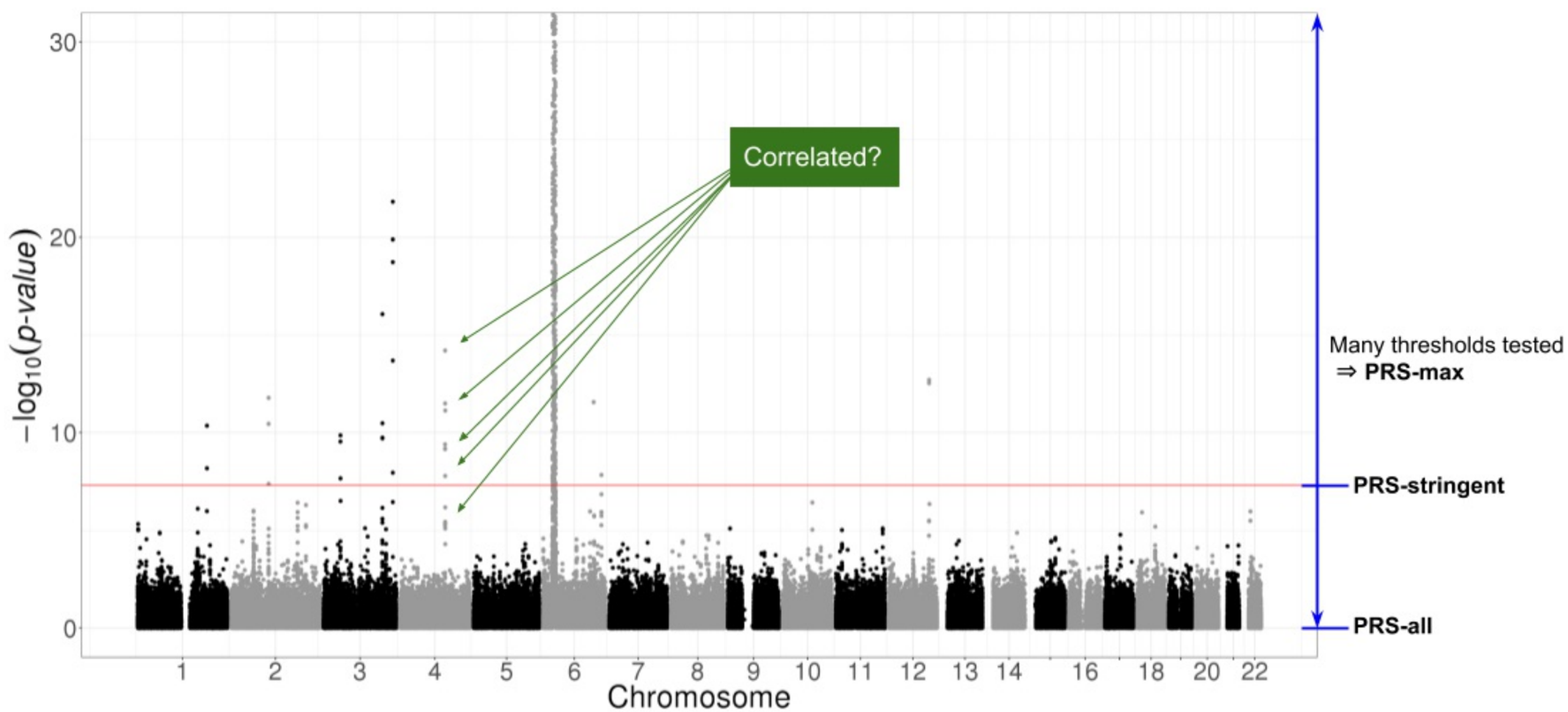


Then, we can **combine these effects in a polygenic risk score (PRS)** of disease.

Widely-used method to compute PRS

GWAS + **Clumping** + **Thresholding** **("C+T" or just "PRS")** (Chatterjee et al. 2013; Dudbridge 2013; Wray et al. 2007)

$$PRS_i = \sum_{\substack{j \in S_{\text{clumping}} \\ p_j < p_T}} \beta_j \cdot G_{i,j}$$



Weights learned independently and heuristics for correlation and regularization.

A more optimal approach to computing PRS?

Statistical learning

- joint models of all SNPs at once
- use regularization to account for correlated and null effects
- already proved useful in the litterature (Abraham et al. 2013; Okser et al. 2014; Spiliopoulou et al. 2015)

Our contribution

- a memory- and computation-efficient implementation to be used for **biobank-scale data**
- an automatic choice of the regularization hyper-parameter
- a comprehensive comparison for different disease architectures

Methods

Penalized Logistic Regression

$$\underset{\beta_0,\,\beta}{\operatorname{argmin}}(\lambda,\alpha)\left\{\underbrace{-\sum_{i=1}^{n}\left(y_i\log(p_i)+(1-y_i)\log(1-p_i)\right)}_{\text{Loss function}}+\underbrace{\lambda\left((1-\alpha)\frac{1}{2}\|\beta\|_2^2+\alpha\|\beta\|_1\right)}_{\text{Penalization}}\right\}$$

- $p_i = 1/\left(1+\exp\left(-(\beta_0+x_i^T\beta)\right)\right)$
- x is denoting the genotypes and covariables (e.g. principal components),
- y is the disease status we want to predict,
- λ is a regularization parameter that needs to be determined and
- α determines relative parts of the regularization $0 \leq \alpha \leq 1$.

Efficient algorithm

- Sequential strong rules for discarding predictors in lasso-type problems (Tibshirani et al., 2012)
- implemented in R package {biglasso} (Zeng et al., 2017)
- reimplemented in R package {bigstatsr} (Privé et al., 2018) with Cross-Model Selection and Averaging (CMSA)

Efficient analysis of large-scale genome-wide data with two R packages: bigstatsr and bigsnpr

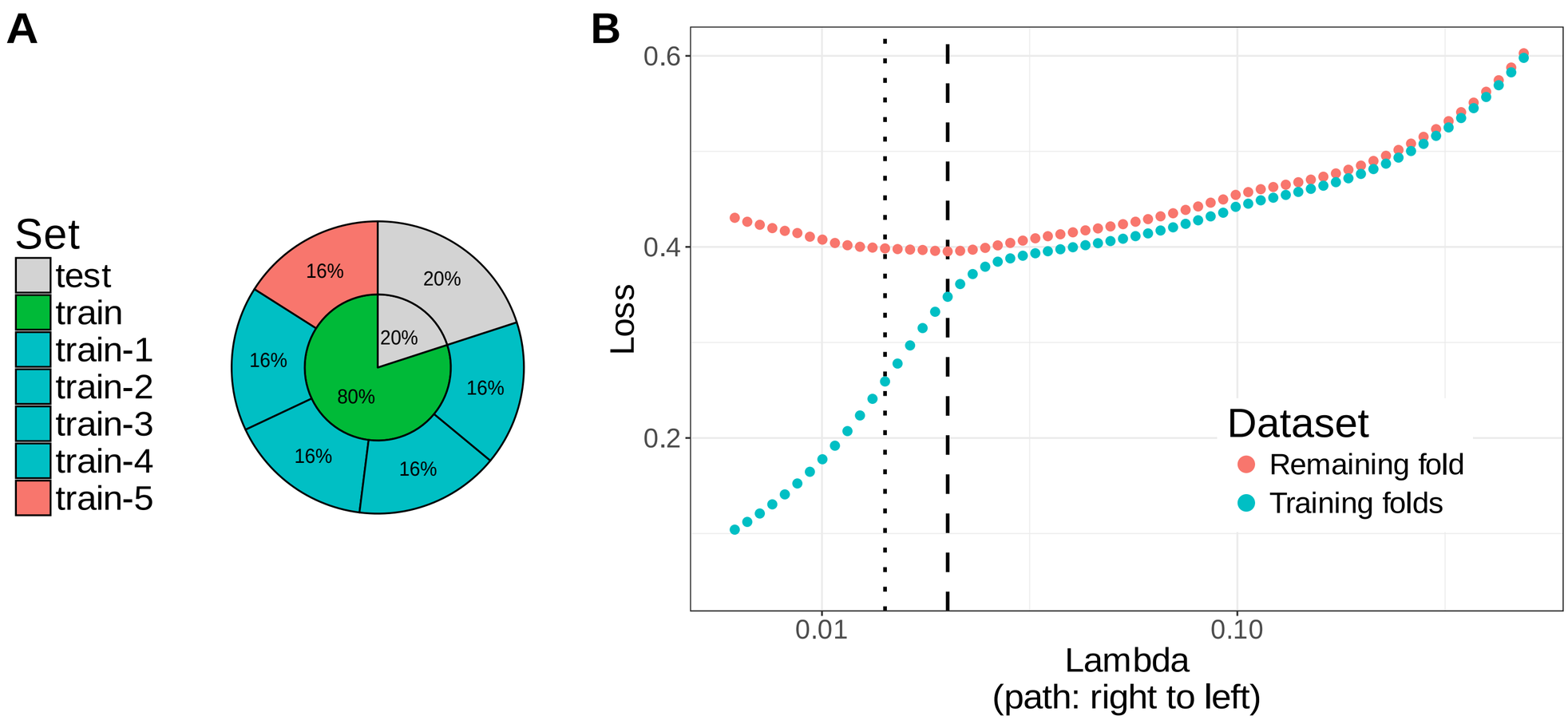
Florian Privé, Hugues Aschard, Andrey Ziyatdinov, Michael G B Blum

Bioinformatics, bty185, https://doi.org/10.1093/bioinformatics/bty185

Package {bigstatsr} and {bigsnpr} use memory-mapping to matrices stored on risk to handle biobank-scale data.

Cross-Model Selection and Averaging (CMSA)

Automatic choice of the regularization hyper-parameter, in turn



- dashed line: λ corresponding to model used
- dotted line: early stopping λ

Comprehensive simulations: varying many parameters

Numero of scenario	Dataset	Size of training set	Causal SNPs (number and location)	Distribution of effects	Heritability	Simulation model	Methods
1	**All 22 chromosomes**	**6000**	**30 in HLA** **30 in all** **300 in all** **3000 in all**	**Gaussian** Laplace	**0.8** 0.5	**simple** fancy	**PRS** **logit-simple** logit-triple (T-Trees)
2	Chromosome 6 only	-	-	-	-	simple	PRS logit-simple
3	**All 22 chromosomes**	**1000** **2000** **3000** **4000** **5000**	**300 in all**	-	-	-	-

Models

- simple: only additive effects
- fancy: additive, dominant and interaction-type effects

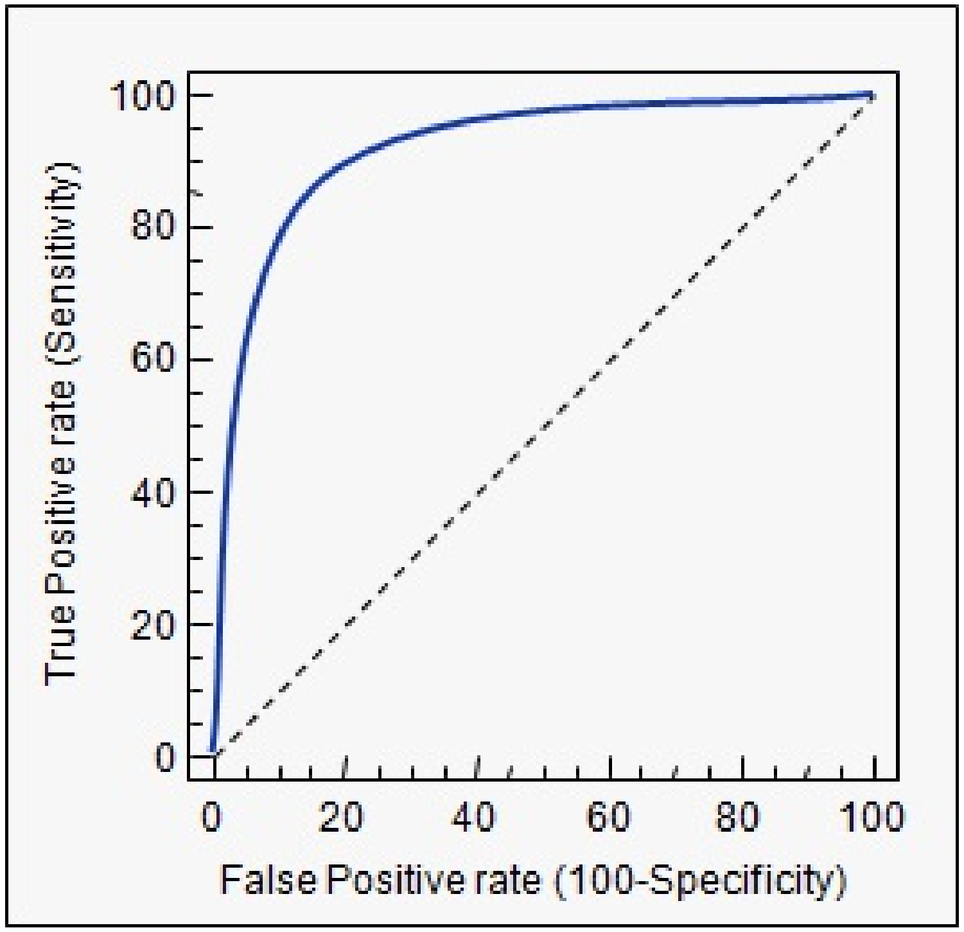
Methods

- PRS: PRS-all (no p-value thresholding), PRS-stringent (GWAS threshold of significance) and PRS-max (best prediction for all thresholds, considered as an upper-bound)

- logit-simple: penalized logistic regression with $\alpha = 0.5$ and CMSA

Predictive performance measures

AUC (Area Under the ROC Curve) is used.

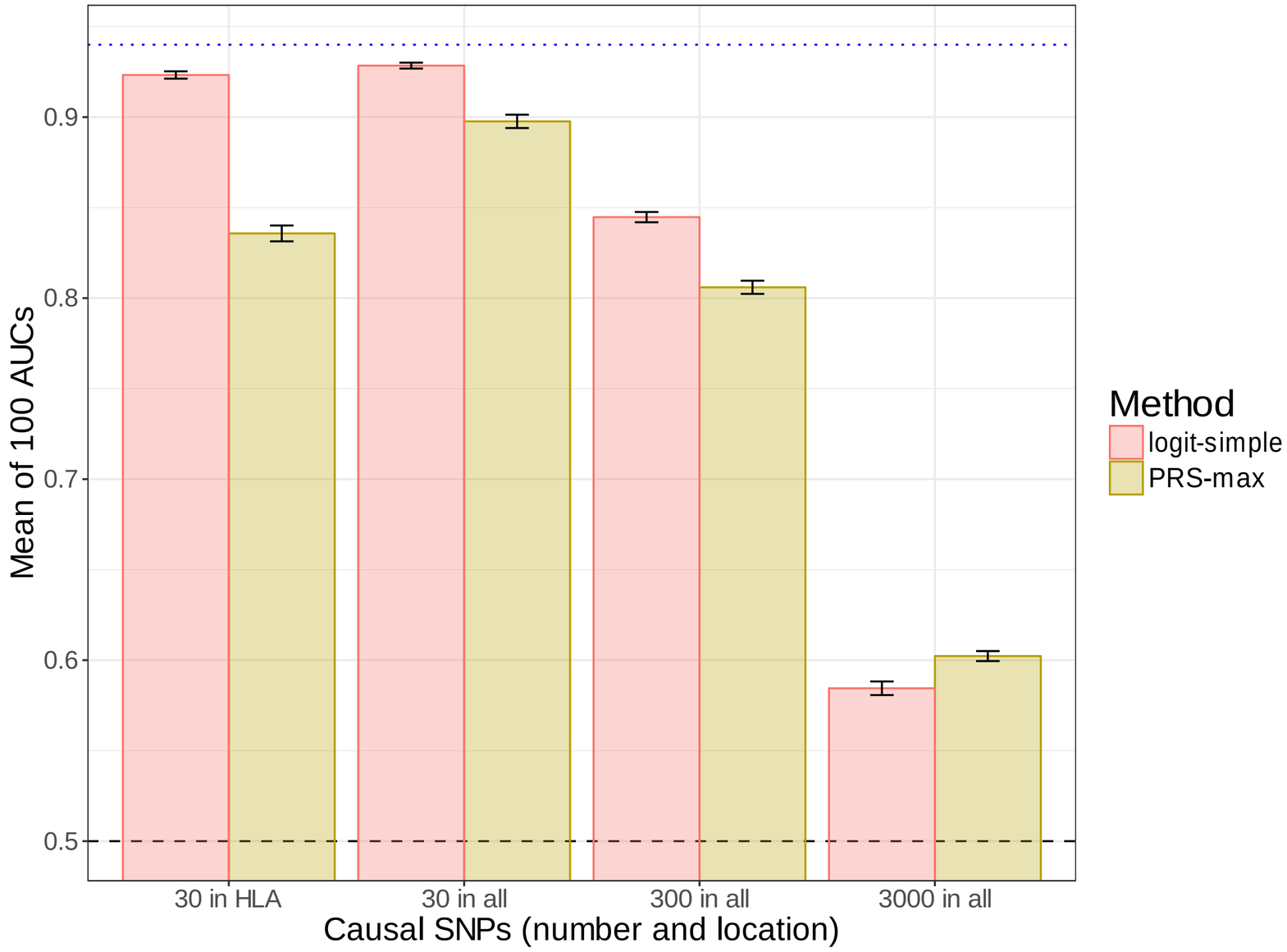


$$\text{AUC} = P(S_{\text{case}} > S_{\text{control}})$$

As a second measure, the **partial AUC** for specificities between 90% and 100% is also reported.

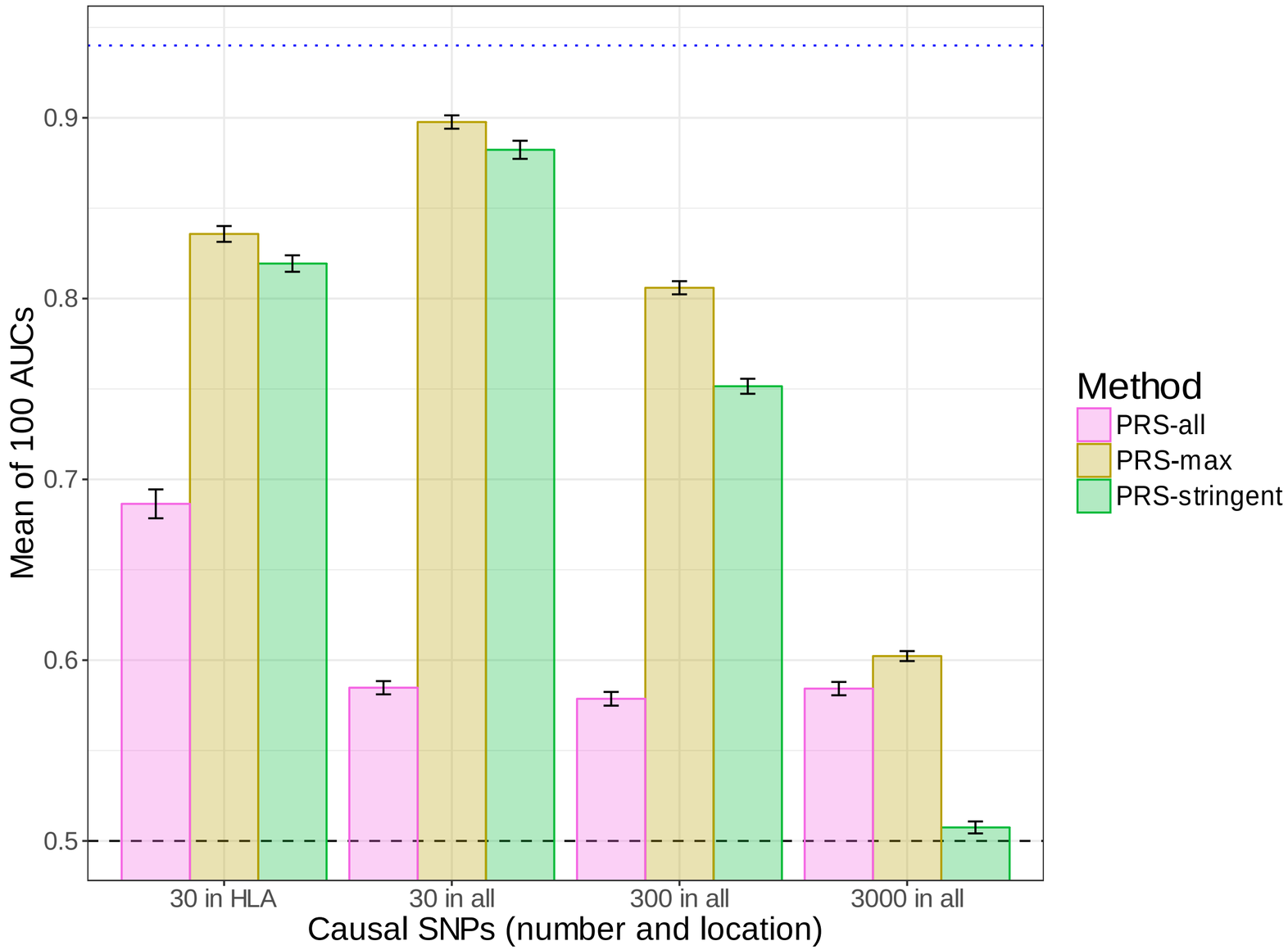
Results

Higher predictive performance with logit-simple



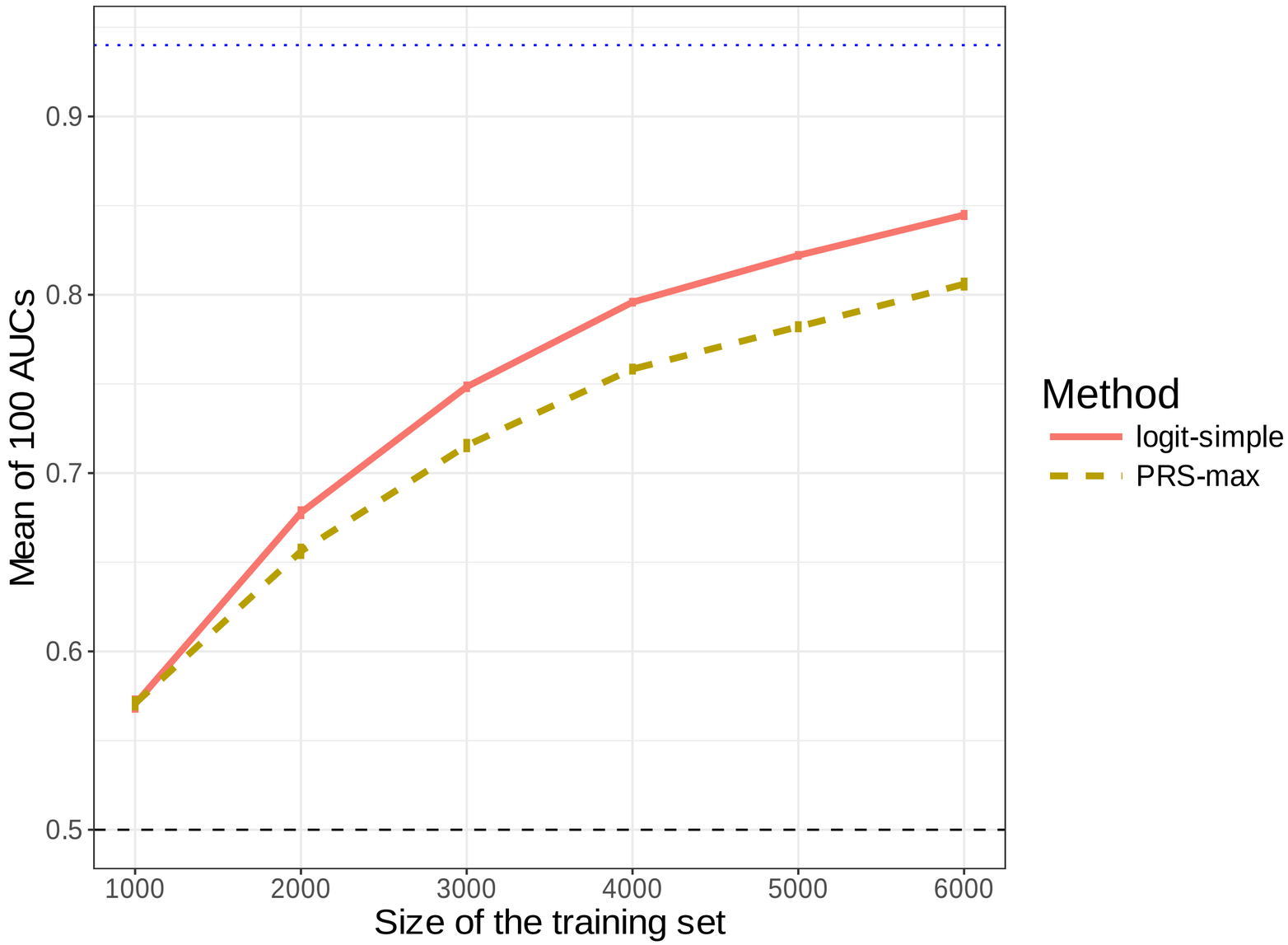
Penalized logistic regression provides higher predictive performance in the cases that matter, especially when there are correlated variables.

Predictive performance of C+T method varies with threshold



Recall that prediction of PRS-max is an upper-bound of the prediction provided by the C+T method.

Prediction with logit-simple is improving faster



Performance of methods improve with larger sample size. Yet, penalized logistic regression is improving faster than the C+T method.

Real data

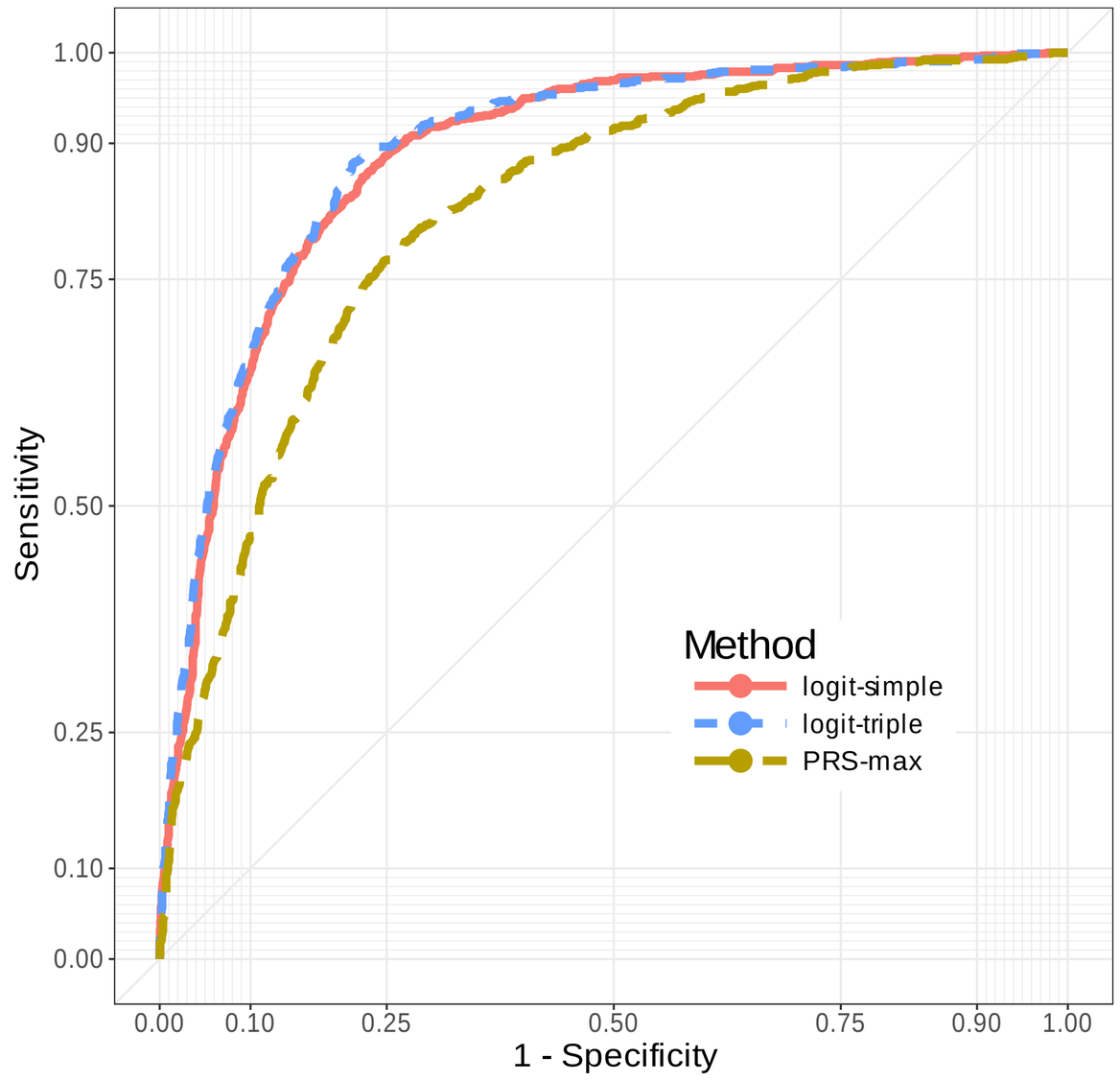
Celiac disease

- intolerance to gluten
- only treatment: gluten-free diet
- heritability: 57-87% (Nisticò et al. 2006)
- prevalence: 1-6%

Case-control study for the celiac disease (Dubois et al. 2010)

- ~15,000 individuals
- ~280,000 SNPs
- ~30% cases

Results: real Celiac phenotypes



Method	AUC	pAUC	# predictors	Execution time (s)
PRS-max	0.824 (0.000704)	0.0286 (0.00016)	9850 (781)	148 (0.414)
logit-simple	0.888 (0.000468)	0.0414 (0.000164)	3220 (62)	83.8 (1.27)
logit-triple	0.892 (0.000488)	0.0429 (0.000174)	4470 (80.6)	141 (1.85)

Discussion

Summary of our penalized regression as compared to the C+T method

- A more **optimal** approach for predicting complex diseases
- models that are **linear** and very **sparse**
- very **fast**
- **automatic choice** for the regularization parameter
- can be extended to capture also recessive and dominant effects

Prospects: future work with the UK Biobank

- use of external summary statistics to improve models
- generalization to external populations
- integration of clinical and environmental data

Thanks!

Presentation: https://privefl.github.io/thesis-docs/recomb18.html

R package {bigstatsr}: https://github.com/privefl/bigstatsr

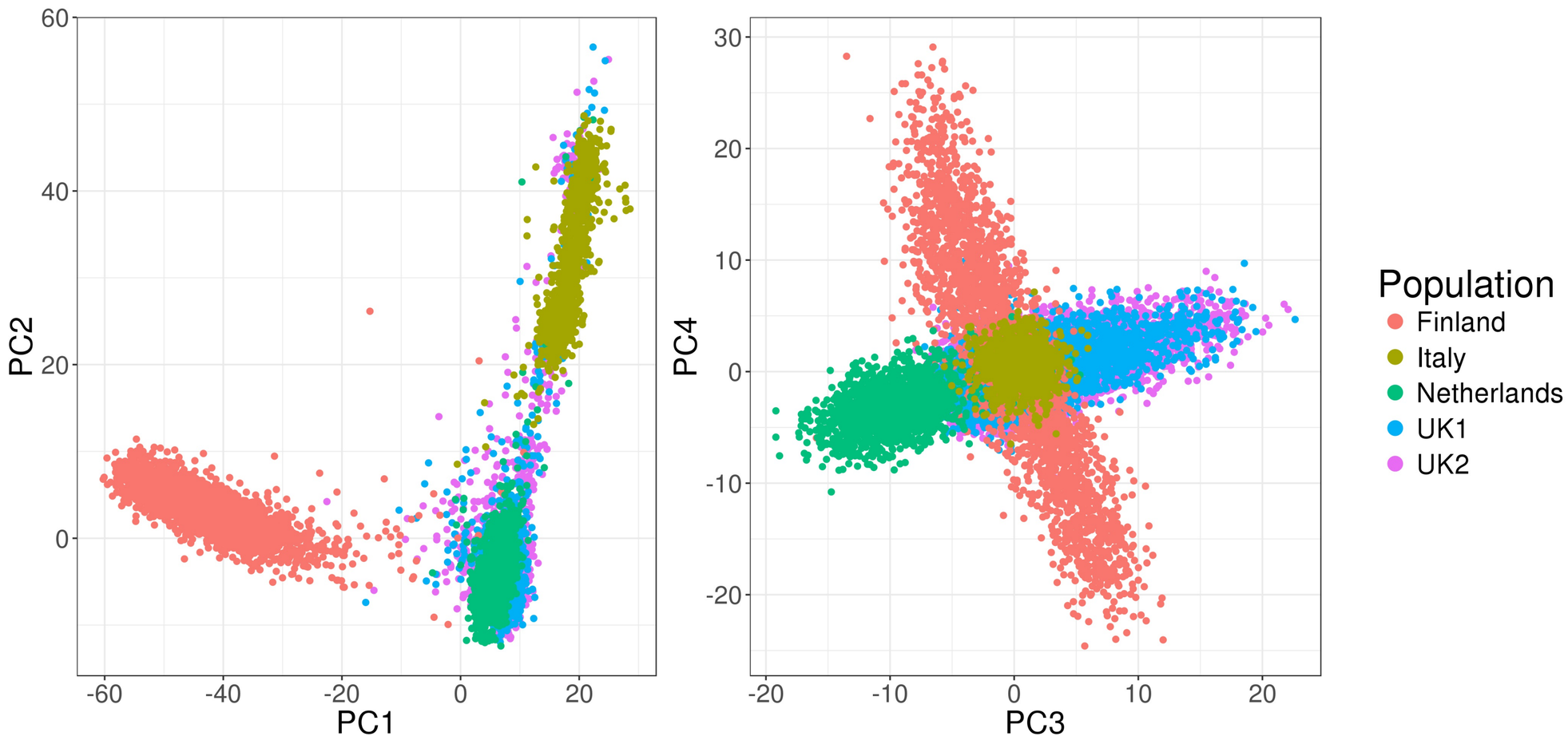
R package {bigsnpr}: https://github.com/privefl/bigsnpr

privefl privefl F. Privé

Slides created via the R package **xaringan**.

Real genotype data

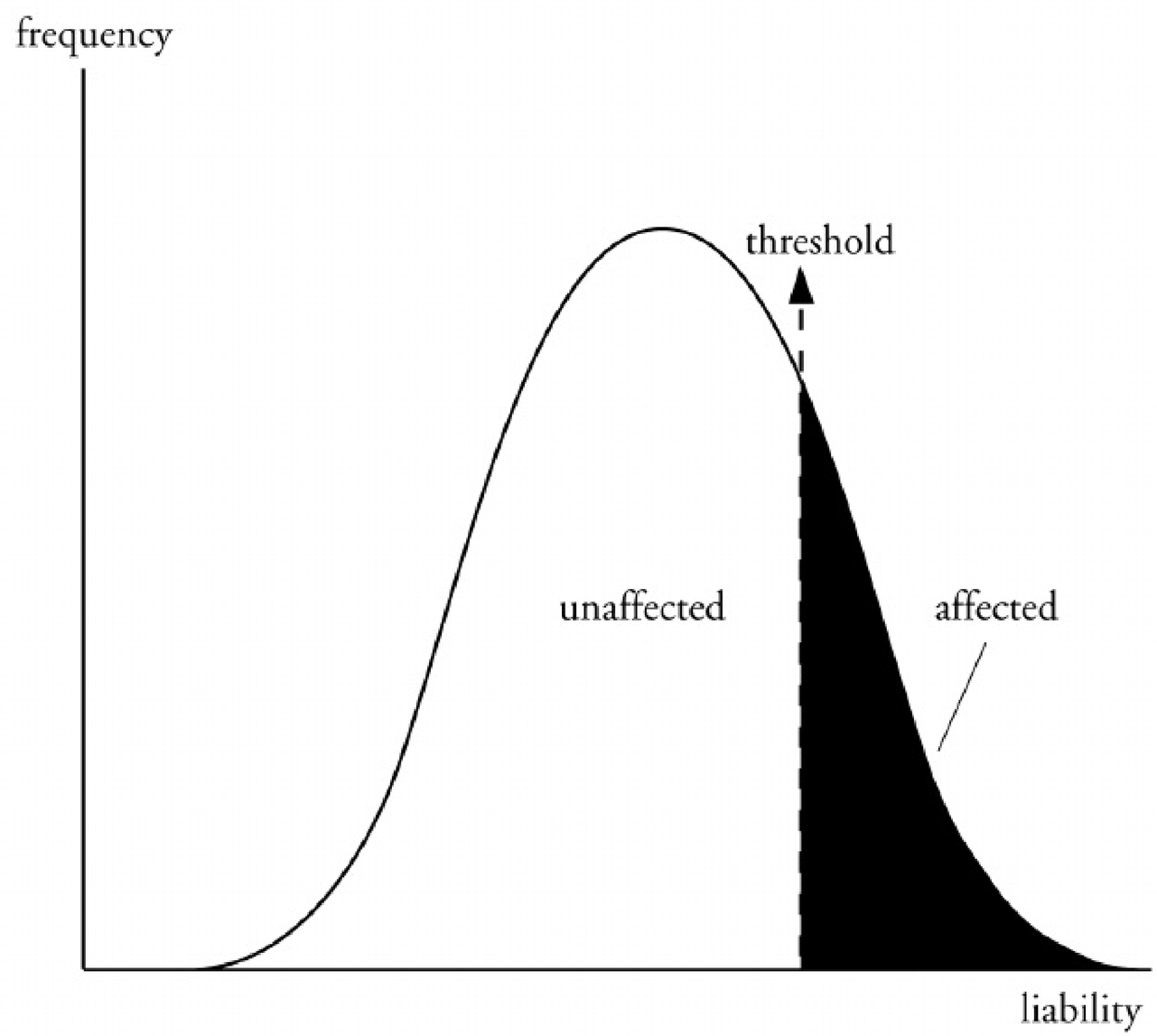
Use real data from a case-control study for the Celiac disease.



Keep only **controls** from the **UK** and **not deviating from the robust Malahanobis distance**.

Simulate new phenotypes

The liability-threshold model



Two models of liability

A "simple" model

$$y_i = \underbrace{\sum_{j \in S_{\text{causal}}} w_j \cdot \widetilde{G_{i,j}}}_{\text{genetic effect}} + \underbrace{\epsilon_i}_{\text{environmental effect}}$$

A "fancy" model

$$y_i = \underbrace{\sum_{j \in S^{(1)}_{\text{causal}}} w_j \cdot \widetilde{G_{i,j}}}_{\text{linear}} + \underbrace{\sum_{j \in S^{(2)}_{\text{causal}}} w_j \cdot \widetilde{D_{i,j}}}_{\text{dominant}} + \underbrace{\sum_{\substack{k=1 \\ j_1 = e_k^{(3.1)} \\ j_2 = e_k^{(3.2)}}}^{k = \left|S^{(3.1)}_{\text{causal}}\right|} w_{j_1} \cdot \widetilde{G_{i,j_1} G_{i,j_2}}}_{\text{interaction}} + \epsilon_i$$

- w_j are **weights** (generated with a Gaussian or a Laplace distribution)
- $G_{i,j}$ is the **allele count** of individual i for SNP j
- $D_{i,j} = \mathbf{1}\left\{G_{i,j} \neq 0\right\}$

Extension via feature engineering

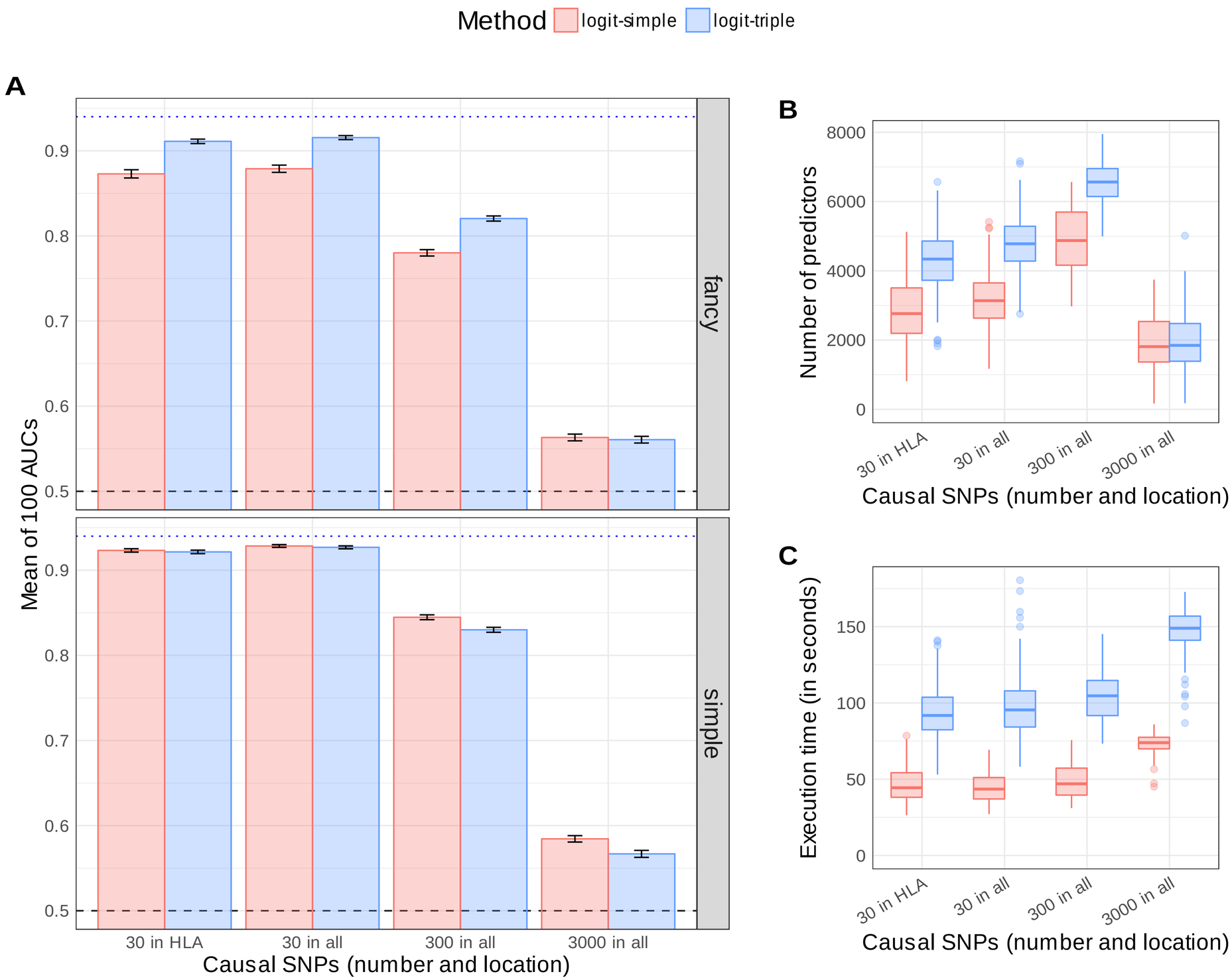
We construct a separate dataset with, for each SNP variable, two more variables coding for recessive and dominant effects.

	SNP1	SNP2
[1,]	0	2
[2,]	0	1
[3,]	1	1
[4,]	0	2
[5,]	0	0
[6,]	1	0
[7,]	1	1
[8,]	0	1
[9,]	0	1
[10,]	0	2

	SNP1.1	SNP1.2	SNP1.3	SNP2.1	SNP2.2	SNP2.3
[1,]	0	0	0	2	1	1
[2,]	0	0	0	1	1	0
[3,]	1	1	0	1	1	0
[4,]	0	0	0	2	1	1
[5,]	0	0	0	0	0	0
[6,]	1	1	0	0	0	0
[7,]	1	1	0	1	1	0
[8,]	0	0	0	1	1	0
[9,]	0	0	0	1	1	0
[10,]	0	0	0	2	1	1

We call these two methods "logit-simple" and "logit-triple".

Feature engineering improves prediction



Prediction with logit-simple is improving faster