

# Polygenic Risk Scores based on statistical learning

Thesis follow-up n°2

Florian Privé

October 1, 2018

# Outline

Introduction

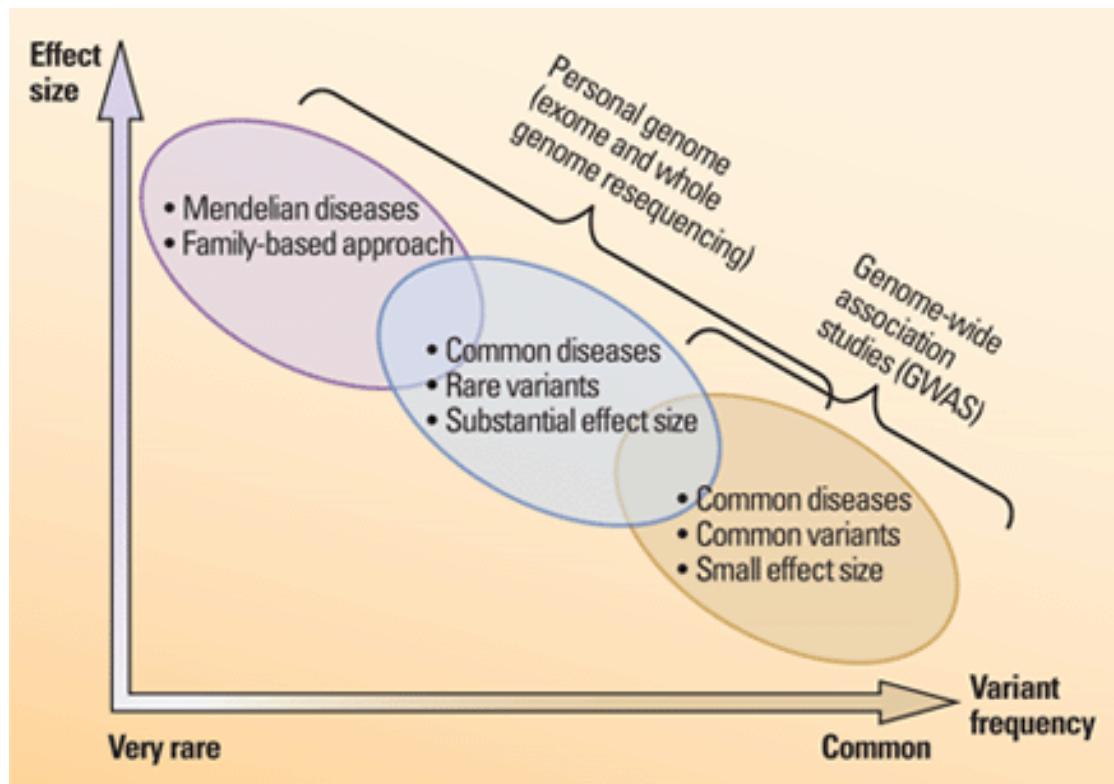
Recall of 1st year

Results of 2nd year

Planning for 3rd year

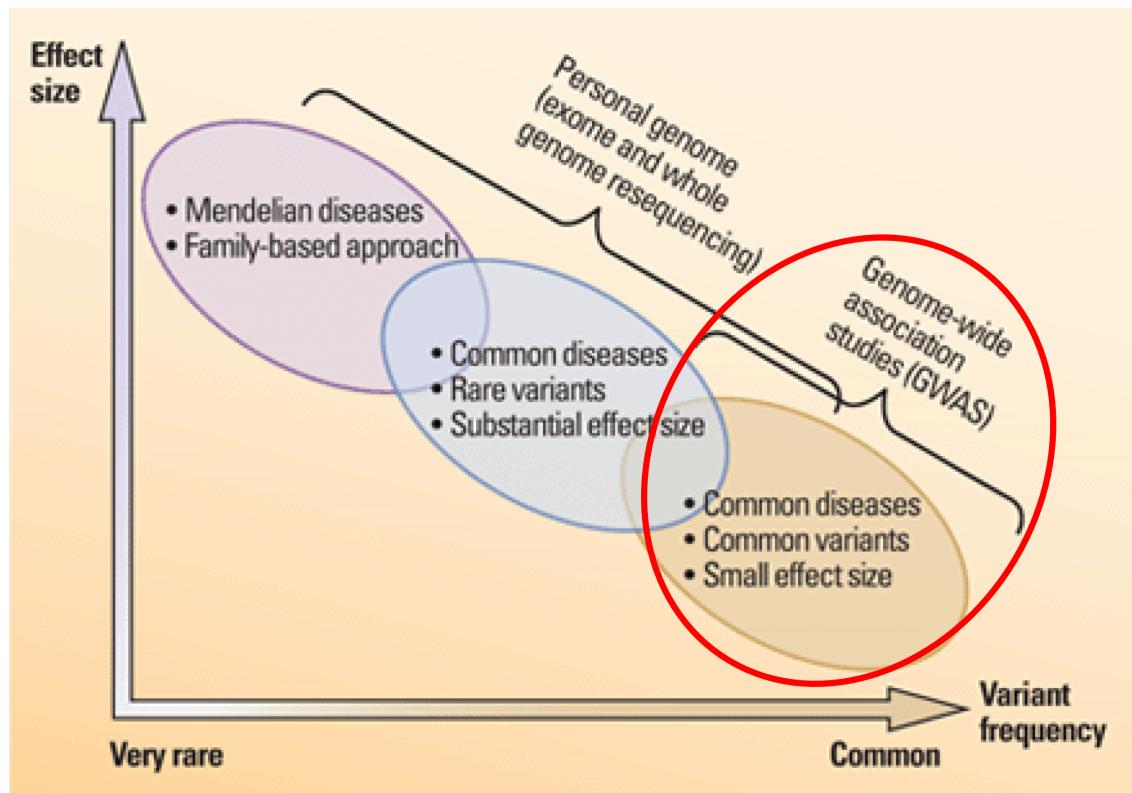
# Introduction

# Disease architectures



Source: 10.1126/science.338.6110.1016

# Disease architectures



How to derive a genetic risk score for 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.
- 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.

Still a gap between current predictions and clinical utility.  
Need more optimal predictions + larger sample sizes.

# Recall of 1st year

Developing tools to help  
researchers (me!) in their analysis

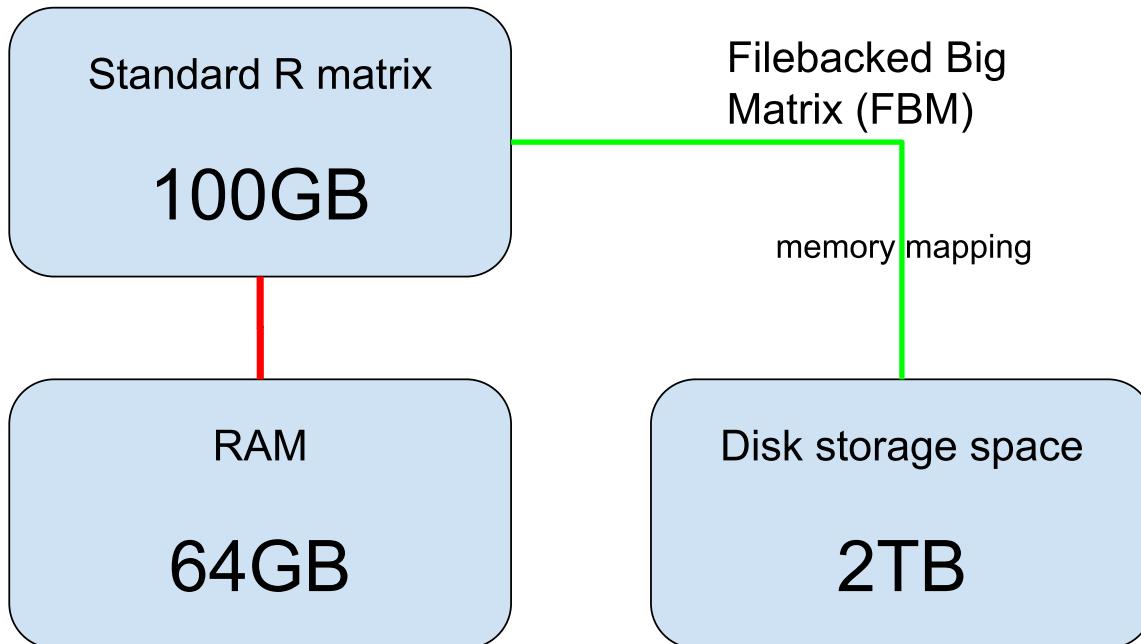
# What I want to be able to do

## Data analysis on large-scale genotype matrices!

- Be fast to test many ideas quickly
  - code should be fast
  - I shouldn't have to make many conversions between formats
  - easily combine multiple functions
- Not be restricted in my analysis
  - Basically use all I already know in 
- Work on my computer (interactively)
  - I have 64 GB of RAM and 12 cores
  - Working on a server is not as easy as on my computer

**Smooth and fast analysis!**

# Memory solution when working in



# My first paper

## 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>

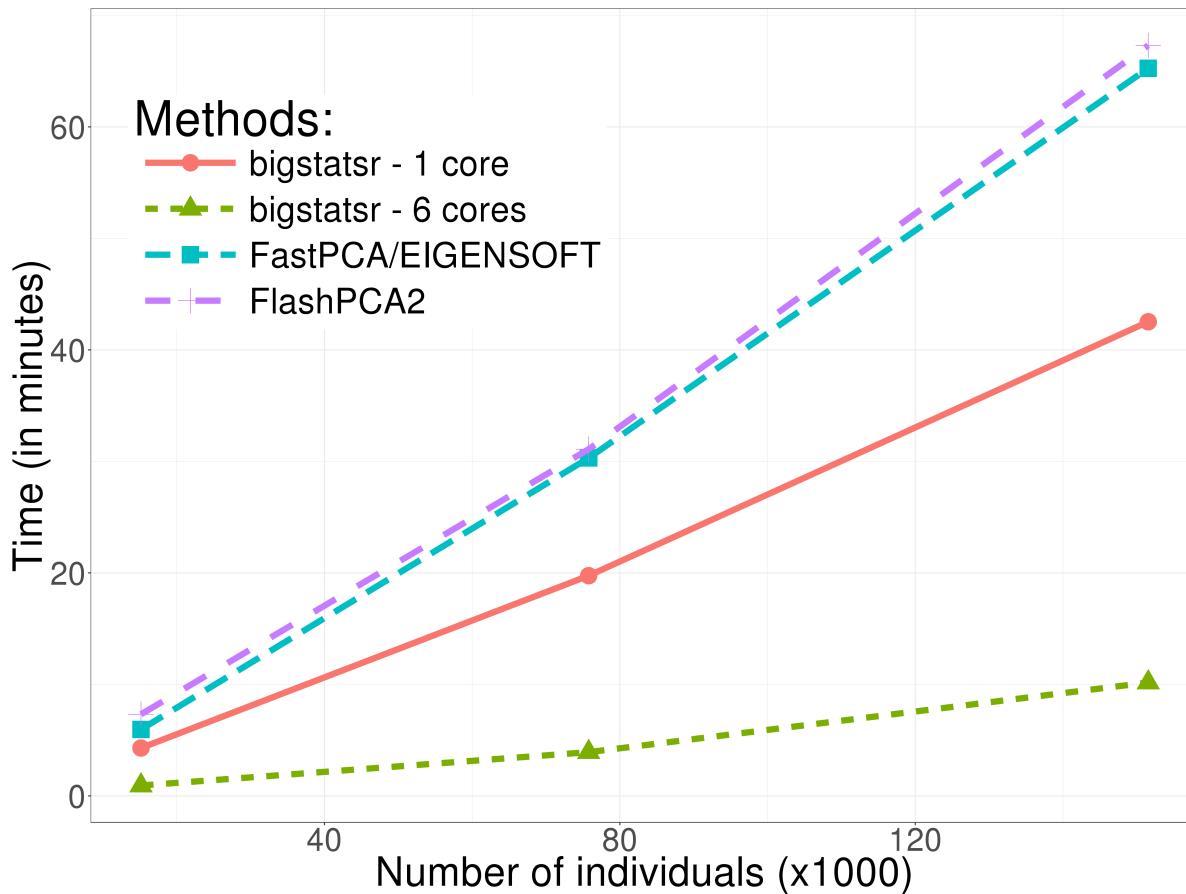
## Two R packages: `{bigstatsr}` and `{bigsnpr}`

- `{bigstatsr}` for many types of matrix, to be used by any field of research
- `{bigsnpr}` for functions which are specific to the analysis of SNP arrays

I've presented `{bigstatsr}` at 3 🎓 conferences.

# Comparative performance

## Computing partial SVD



# Results of 2nd year

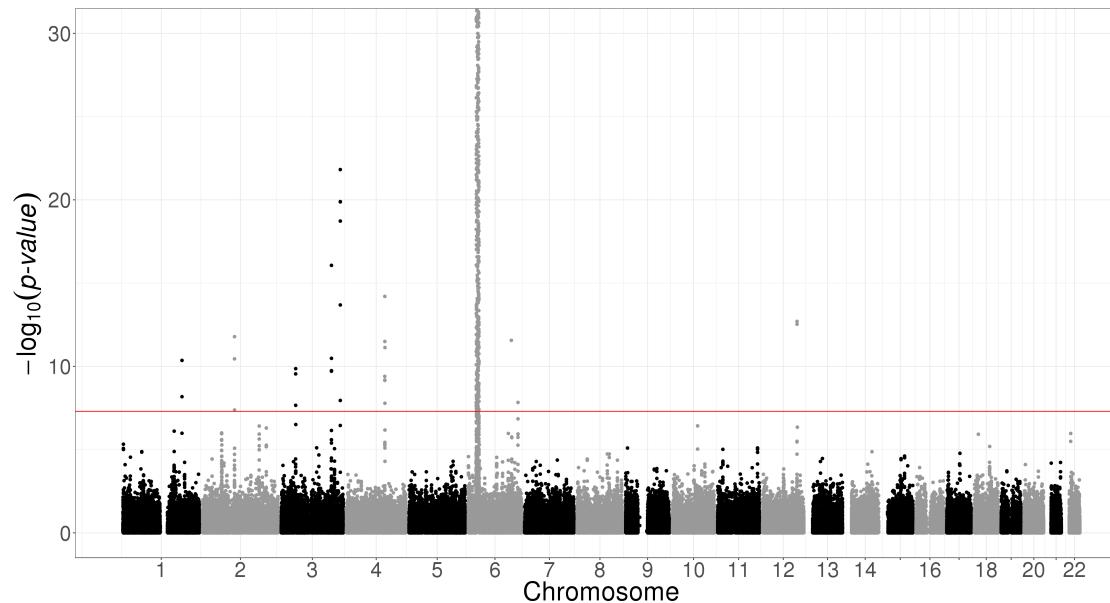
Comparing models for computing

Polygenic Risk Scores (PRS)

# Standard PRS - part 1: estimating effects

## Genome-wide association studies (GWAS)

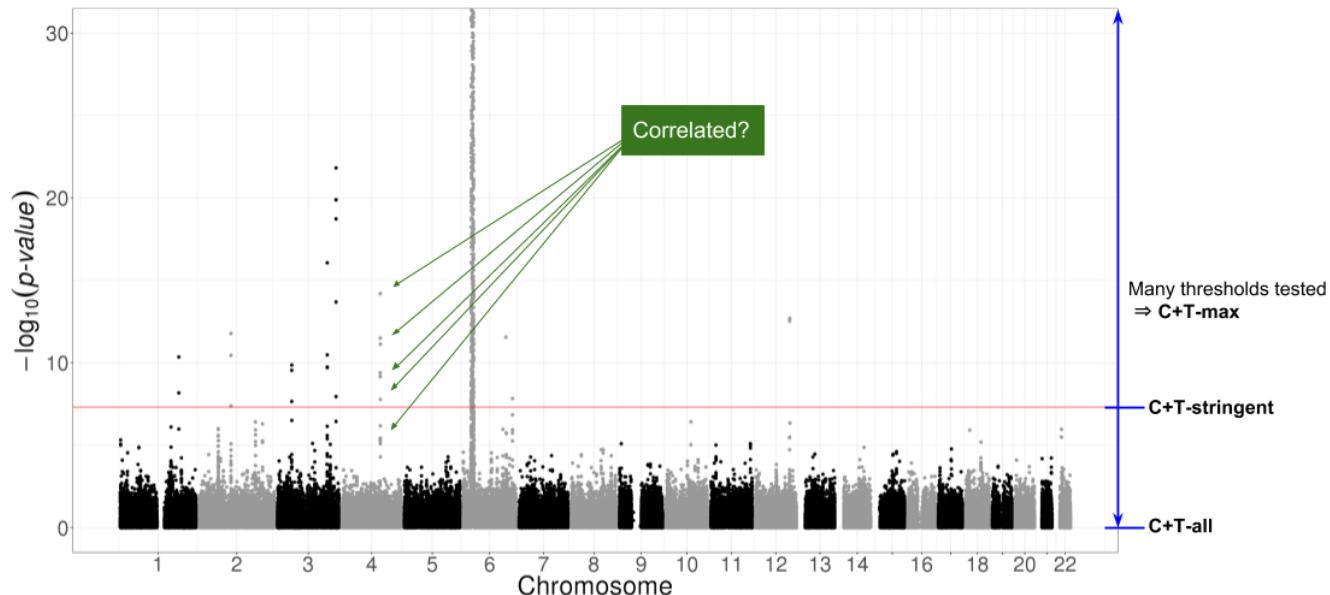
In a GWAS, each single-nucleotide polymorphism (SNP) is tested **independently**, resulting in one **effect size**  $\hat{\beta}$  and one **p-value**  $p$  for each SNP.



Easy combining:  $PRS_i = \sum \hat{\beta}_j \cdot G_{i,j}$

# Standard PRS - part 2: restricting predictors

## Clumping + Thresholding (C+T)



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

# A more optimal approach to computing PRS?

In C+T: weights learned independently and heuristics for correlation and regularization.

## 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 (PLR)

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

- 
- $p_i = 1 / (1 + \exp(-( \beta_0 + x_i^T \beta)))$
  - $x$  is denoting the genotypes and covariates (e.g. principal components),
  - $y$  is the disease status we want to predict,
  - $\lambda$  is a regularization parameter that needs to be determined and
  - $\alpha$  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; Zeng et al. 2017)
- implemented in our R package {bigstatsr}

**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 

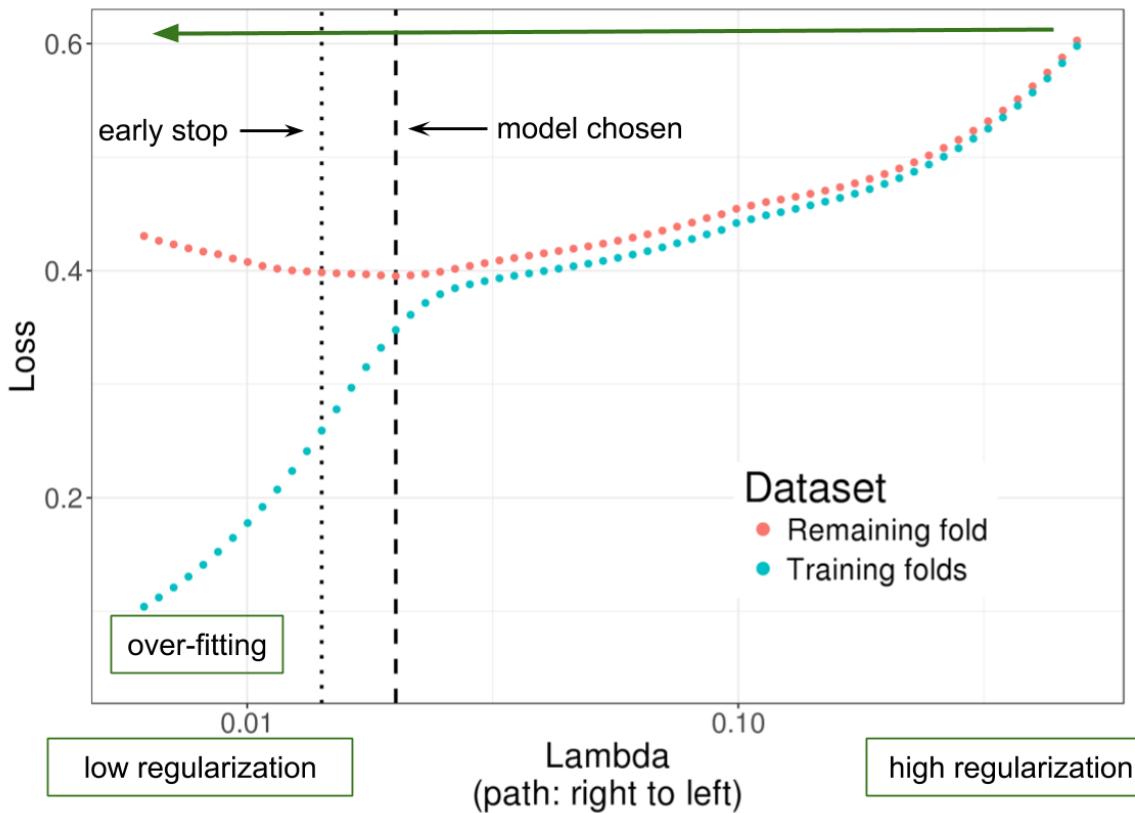
*Bioinformatics*, bty185, <https://doi.org/10.1093/bioinformatics/bty185>

## Our implementation

- uses memory-mapping to matrices stored on disk
- functioning choice of the hyper-parameter  $\lambda$  + embedded grid-search for  $\alpha$
- early stopping criterion to fasten the overall procedure

# Choice of the hyper-parameter $\lambda$

TRAINING	VALIDATION	TEST
learn a 'regularization path' of many models	choose one model and early stop	check this model



# Comprehensive simulations: varying many parameters

## Simulation models

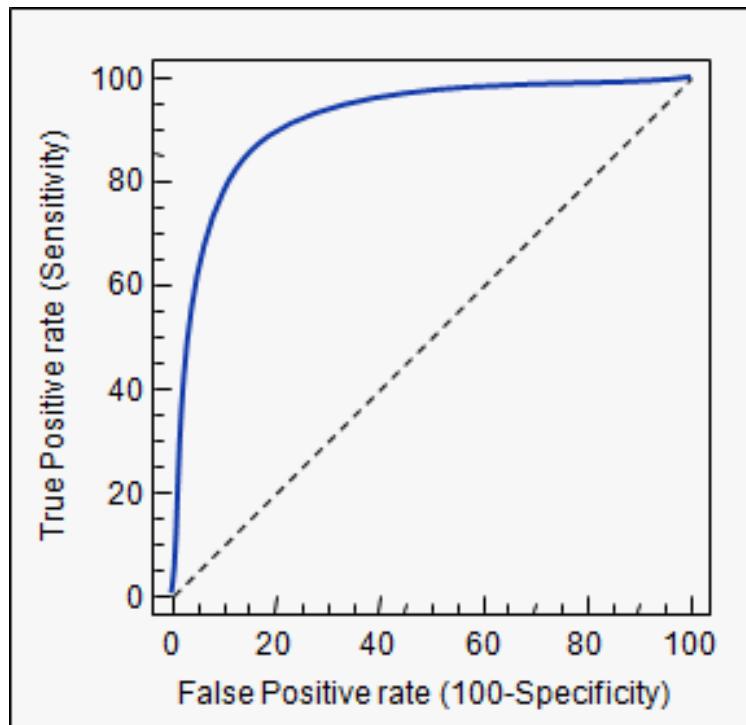
Numero of scenario	Dataset	Size of training set	Causal SNPs (number and location)	Distribution of effects	Heritability	Simulation model	Methods
1	All 22 chromosomes	6000	30 in HLA 30 in all 300 in all 3000 in all	Gaussian Laplace	0.5 0.8	ADD COMP	C+T PLR PLR3 (T-Trees)
2	Chromosome 6 only	-	-	-	-	ADD	C+T PLR
3	All 22 chromosomes	1000 2000 3000 4000 5000	300 in all	-	-	-	-

## Methods

- C+T:
  - C+T-all (no p-value thresholding),
  - C+T-stringent (GWAS threshold of significance) and
  - C+T-max (best prediction for all thresholds, considered as an upper-bound)
- PLR: penalized logistic regression with automatic selection of hyper-parameters

# Predictive performance measures

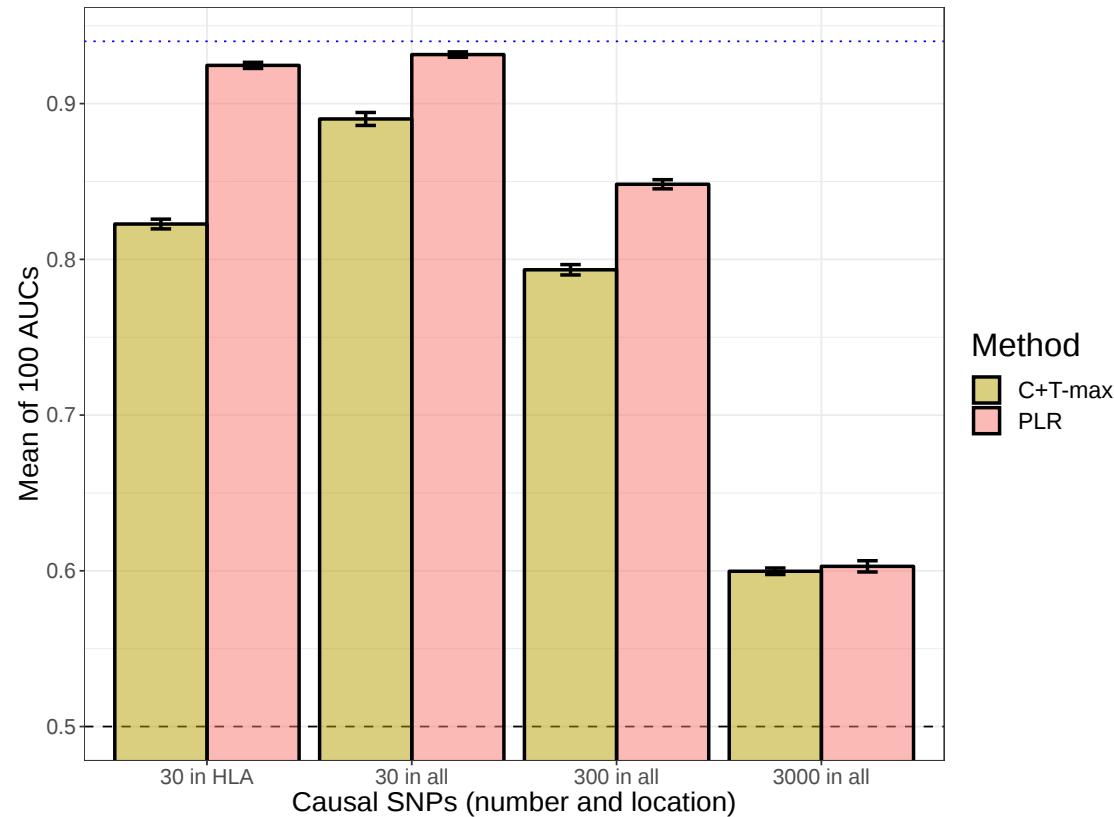
AUC (Area Under the ROC Curve) and partial AUC (FPR < 10%) are used.



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

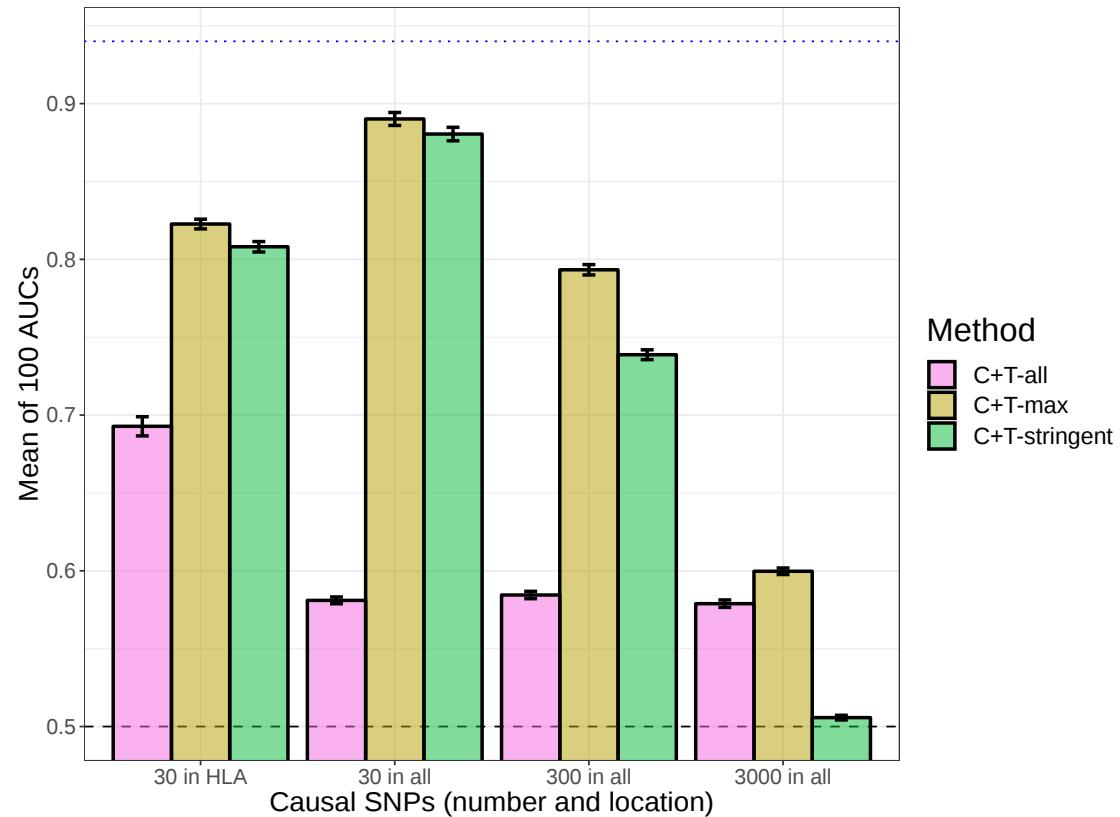
# Results

# Higher predictive performance with PLR



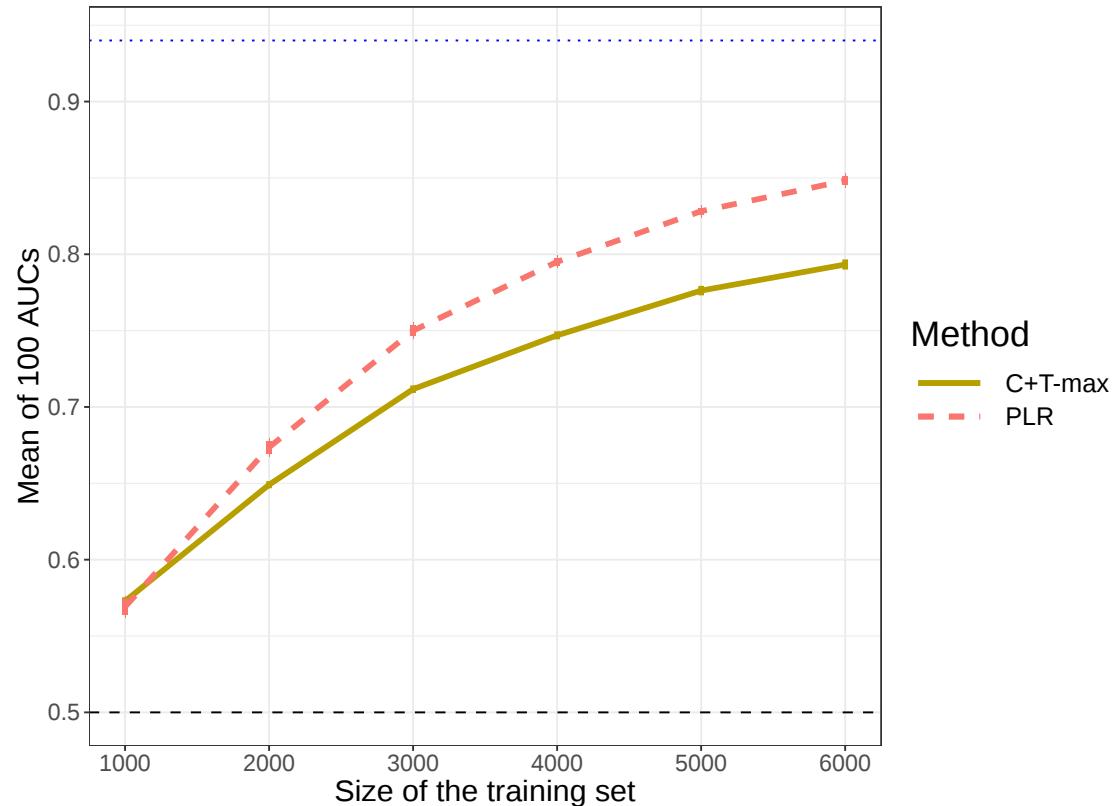
Penalized logistic regression consistently provides higher predictive performance, especially when there are correlated variables.

# Predictive performance of C+T method varies with threshold



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

# Prediction with PLR is improving faster (Scenario #3)



Performance of methods improve with larger sample size. Yet, PLR is improving faster than C+T.

# Real data

## Celiac disease

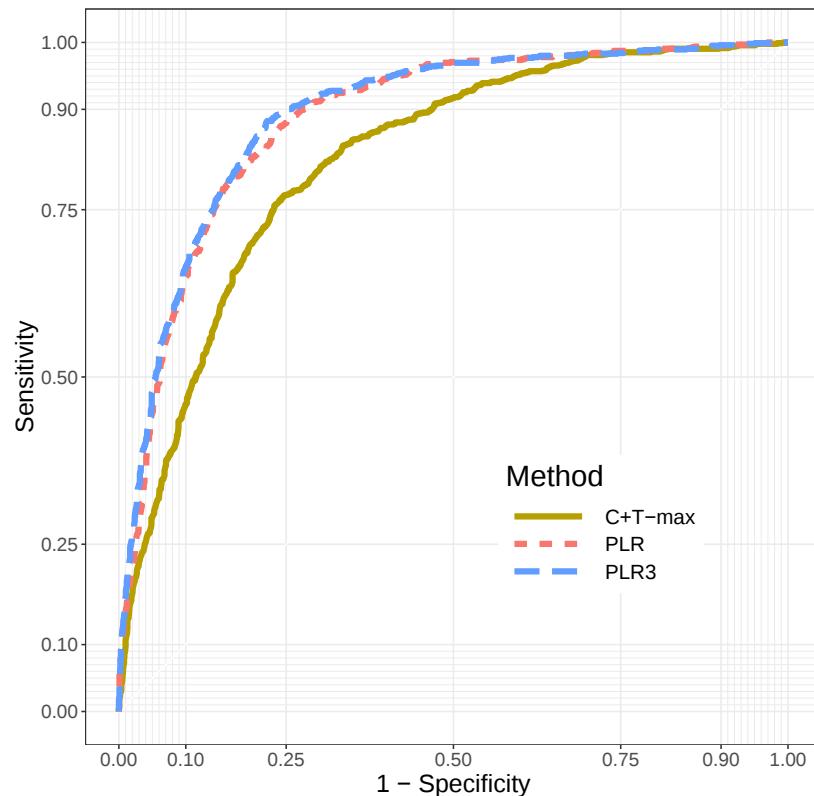
- intolerance to gluten
- only treatment: gluten-free diet
- heritability: 57-87% (Nisticò et al. 2006)
- prevalence: 1-6%

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

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

# Results: real Celiac phenotypes

Method	AUC	pAUC	# predictors	Execution time (s)
C+T-max	0.825 (0.000664)	0.0289 (0.000187)	8360 (744)	130 (0.143)
PLR	0.887 (0.00061)	0.0411 (0.000224)	1570 (46.4)	190 (1.21)
PLR3	0.891 (0.000628)	0.0426 (0.000219)	2260 (56.1)	296 (2.03)



# 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 using the UK Biobank

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

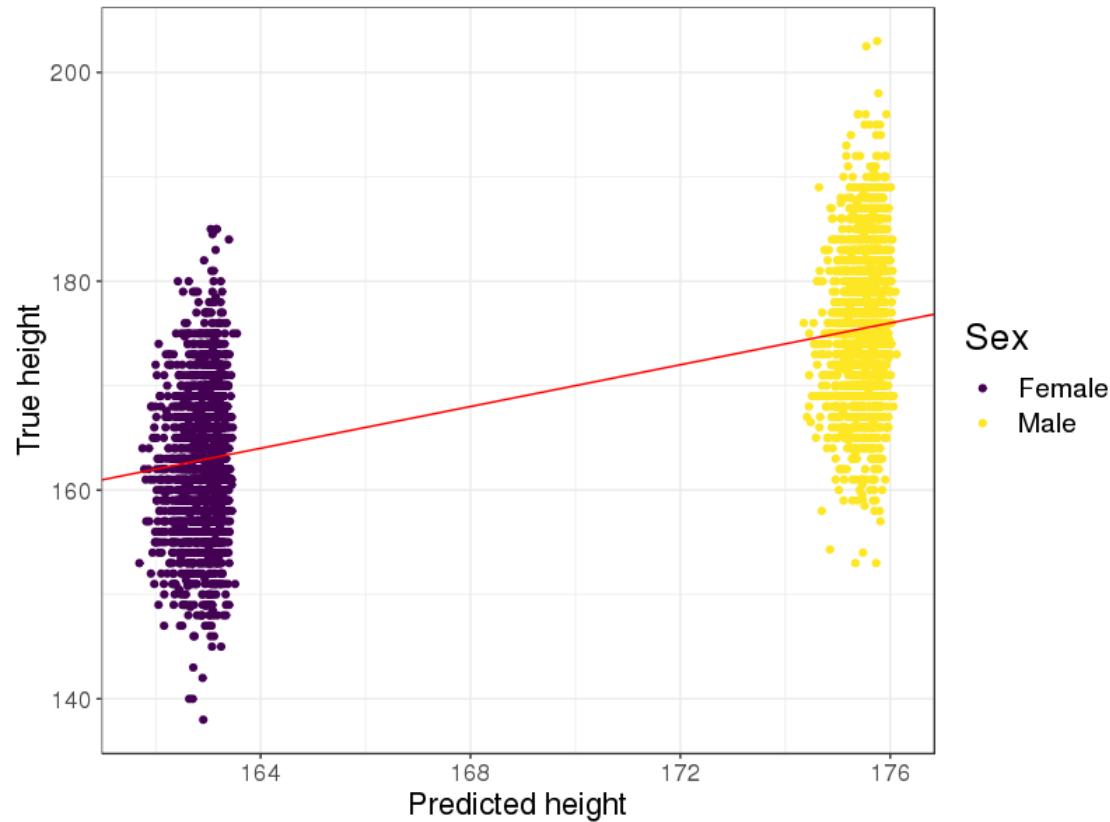
# Planning for 3rd year

+ early results usink the UK biobank

# Pitfalls of penalized regression

# Penalization is too strong when there are large effects (sex for height)

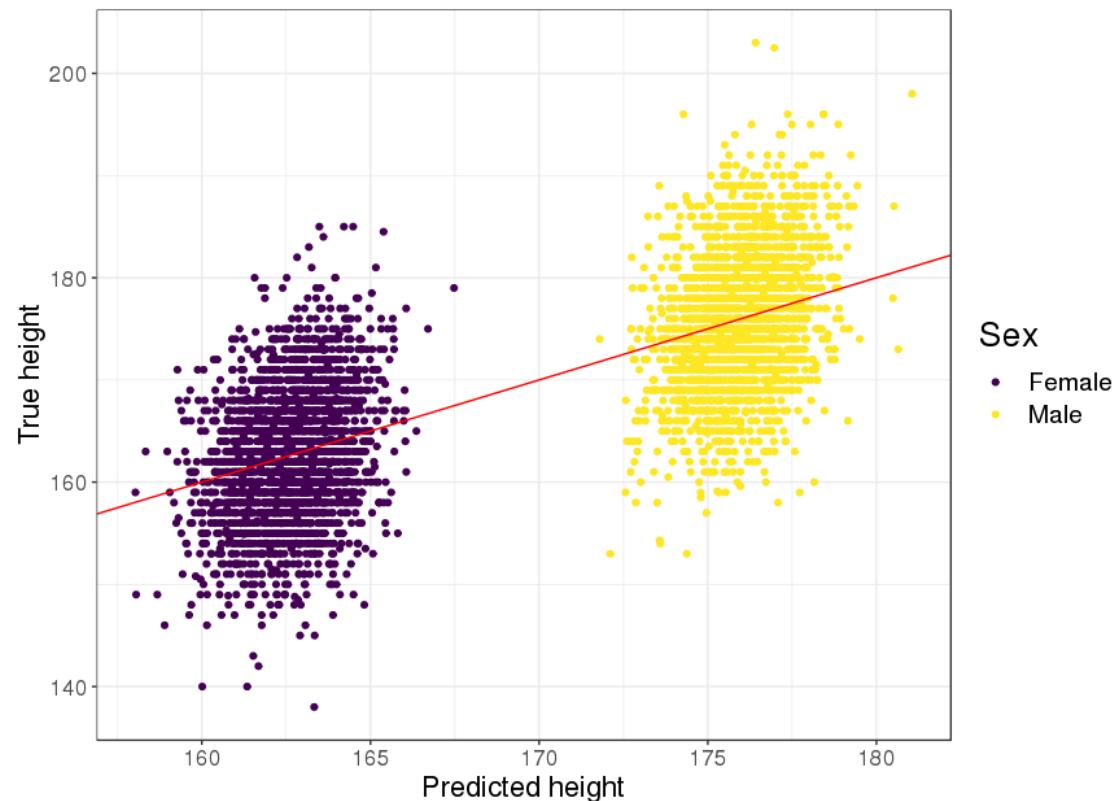
$\text{height} \sim \text{SNPs} + \text{PCs} + \text{sex}$



# Better prediction, but still too conservative

base:  $\text{height} \sim \text{sex}$

pred:  $\text{height} \sim \text{base} + \text{SNPs} + \text{PCs} + \text{sex}$

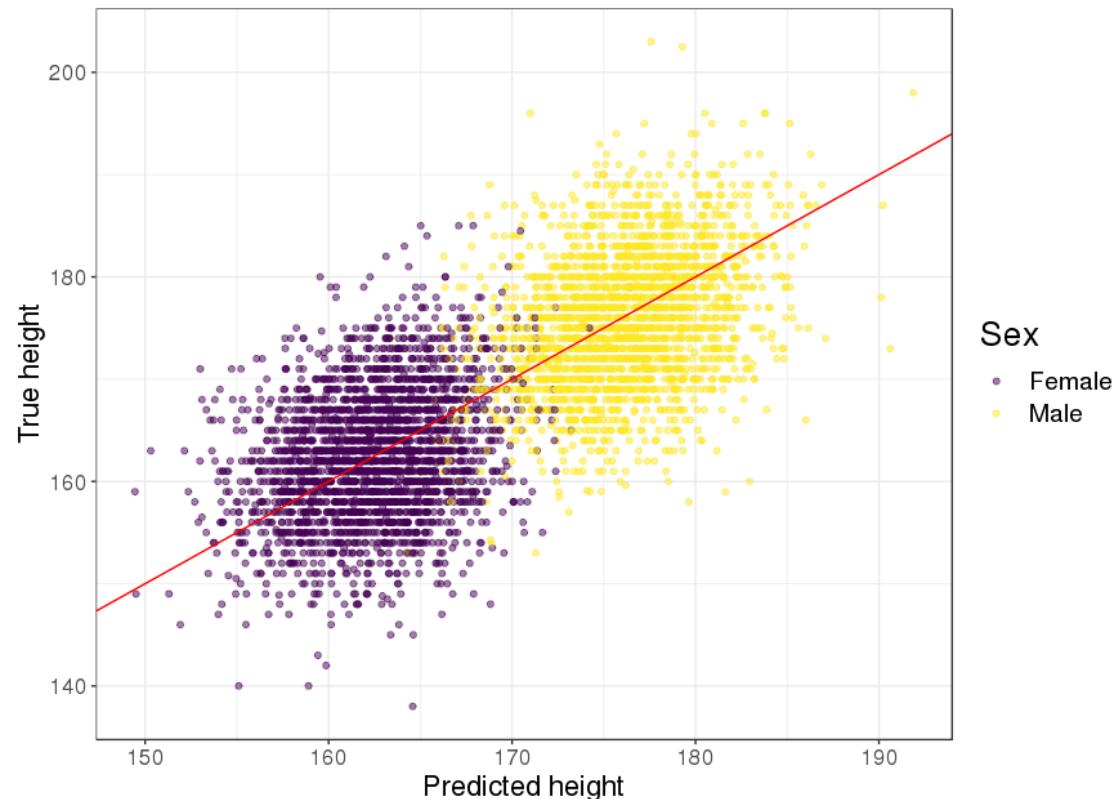


# Post-process with some linear regression

base:  $\text{height} \sim \text{sex}$

pred:  $\text{height} \sim \text{base} + \text{SNPs} + \text{PCs} + \text{sex}$

post\_pred:  $\text{height} \sim \text{pred} \times \text{sex}$



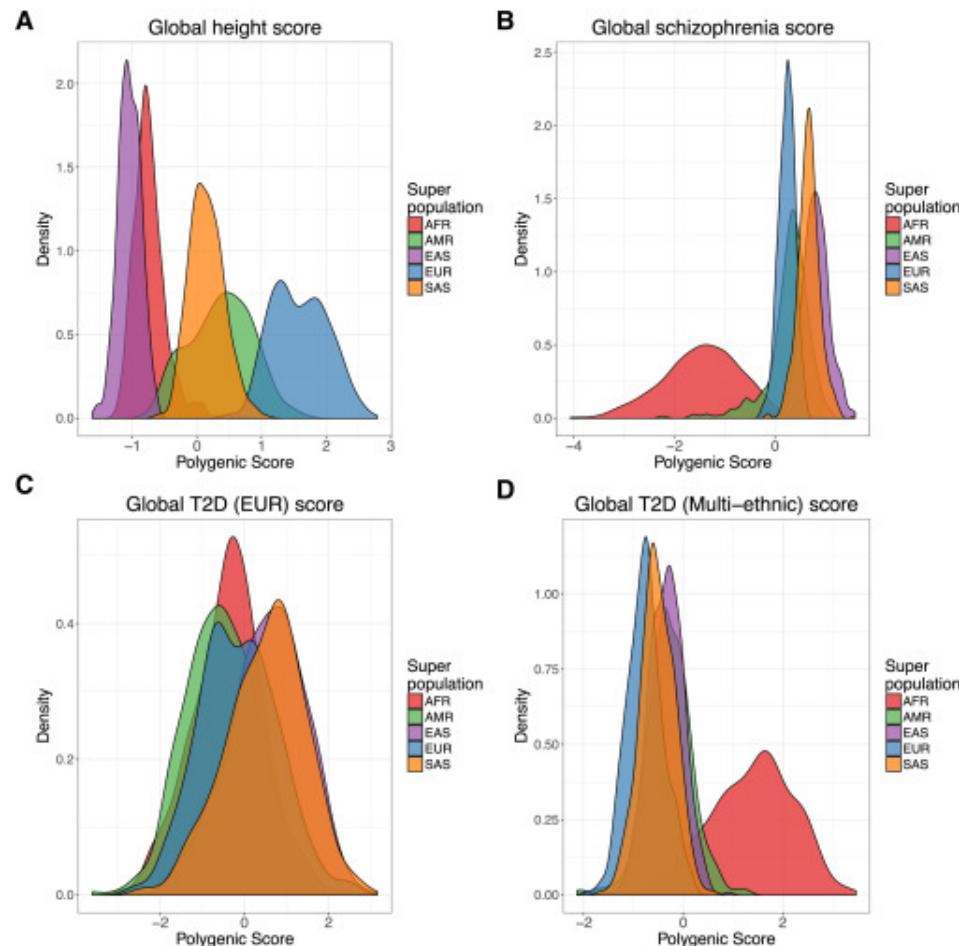
How to combine the information  
of multiple studies?  
(possibly of different populations)

# Genetics are different between populations



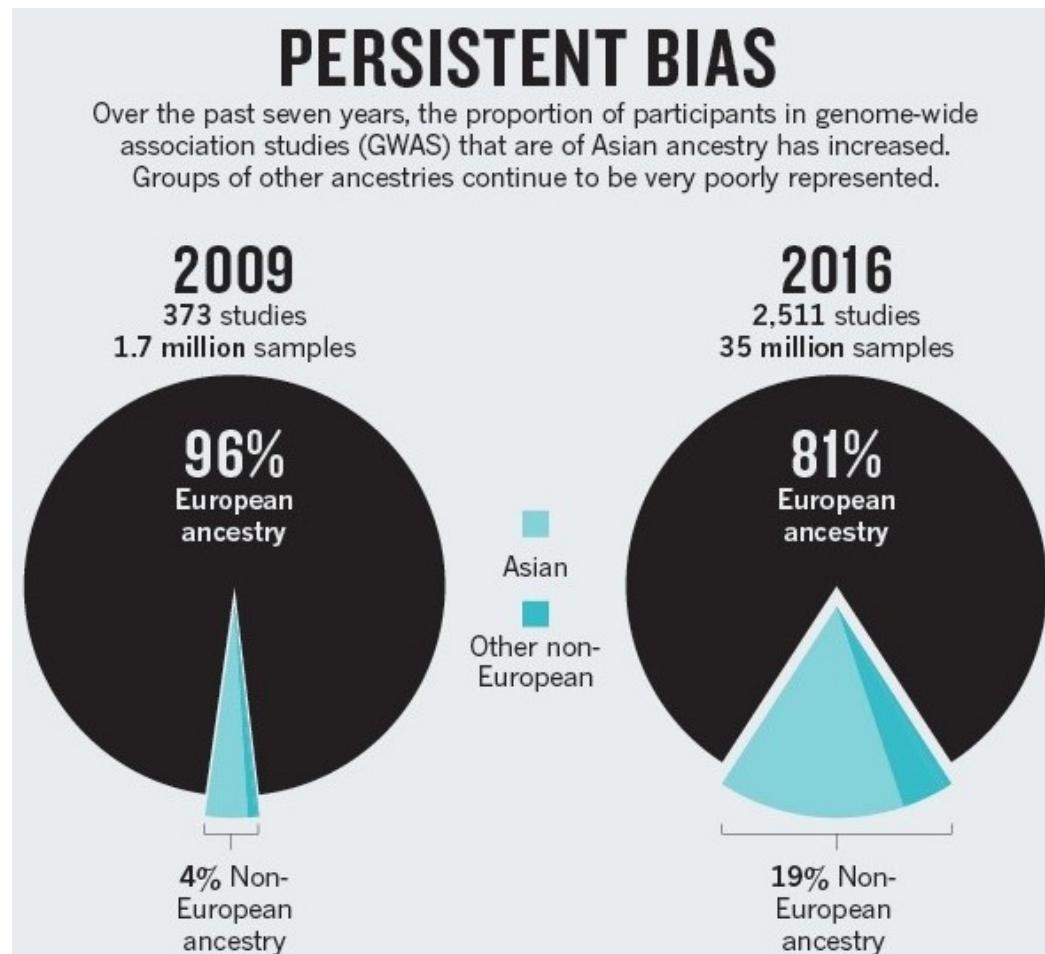
Produced with our R packages (450K people from the UK biobank)

# which makes predictions fail on external populations



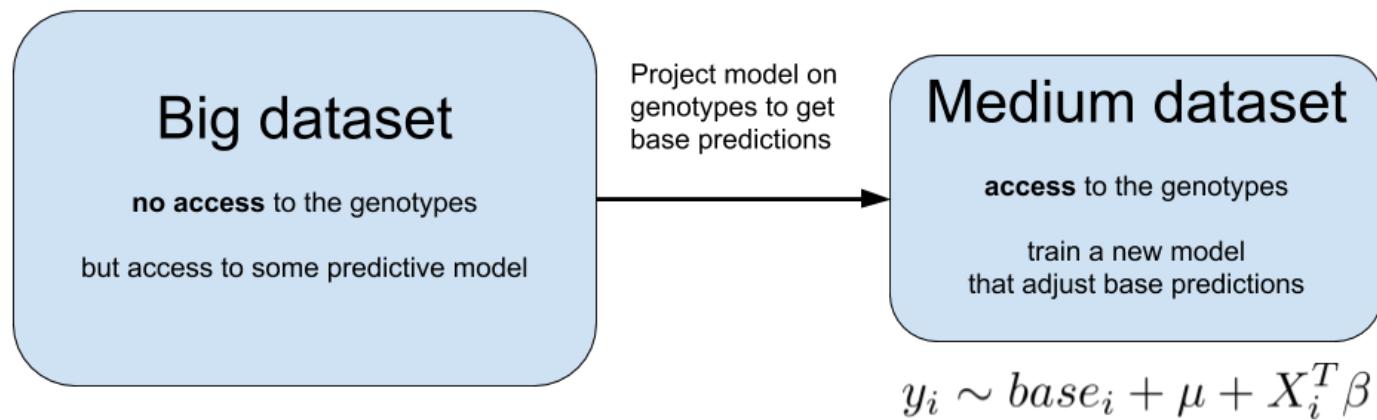
Source: 10.1016/j.ajhg.2017.03.004

# Genomics is failing on diversity



# What can we do about it?

We can use information from other studies (possibly in other populations)



Will this improve prediction?

# Planning for 3rd year

## End of 2018

- Publication of second paper
- Exploratory analysis for 3rd paper
- Starting thesis writing

## Early 2019

- Teaching (Advanced R for doctoral school + statistics at ENSIMAG)
- Final analysis + writing of third paper
- Getting my DNA sequenced and analyzing it!

## Until July 2019

- Submission of 3rd paper
- Finishing writing thesis
- International R Conference useR!2019 in Toulouse

## The end

- Answer reviews for 3rd paper
- Thesis defense

# Thanks!

Presentation: <https://privefl.github.io/thesis-docs/suivi-these2.html>

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

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



privefl



privefl

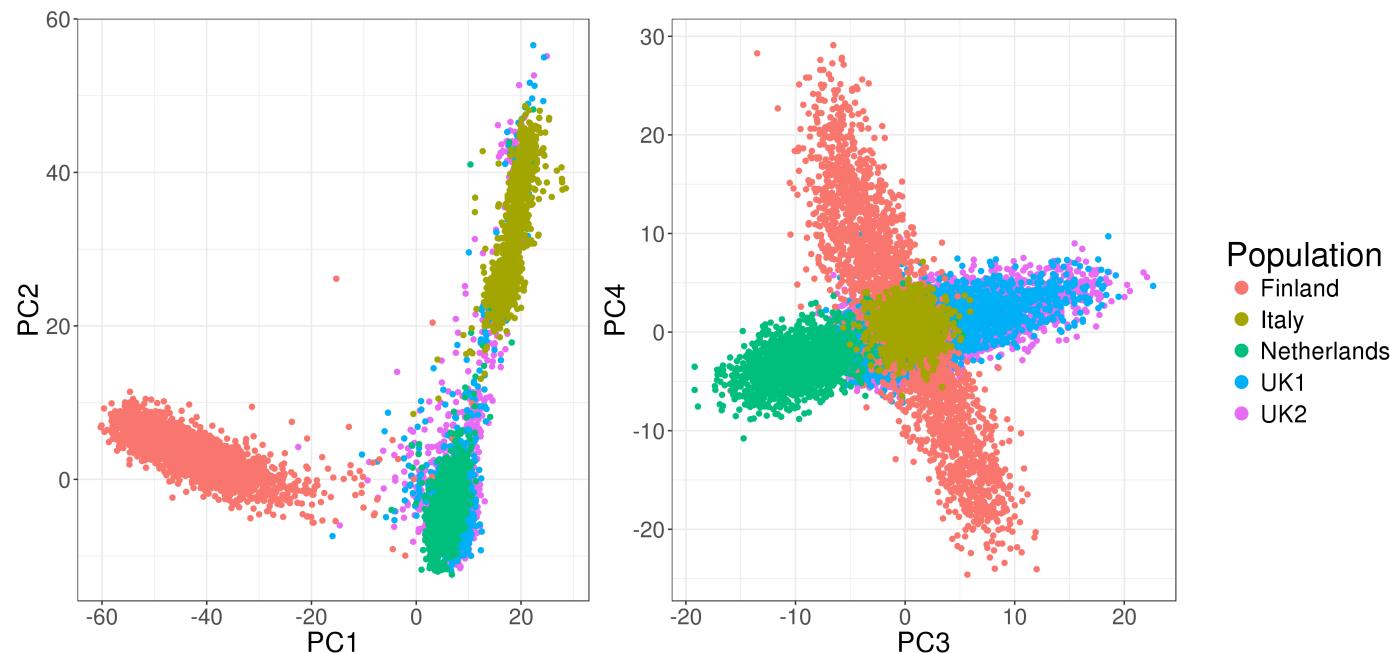


F. Privé

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

# Real genotype data

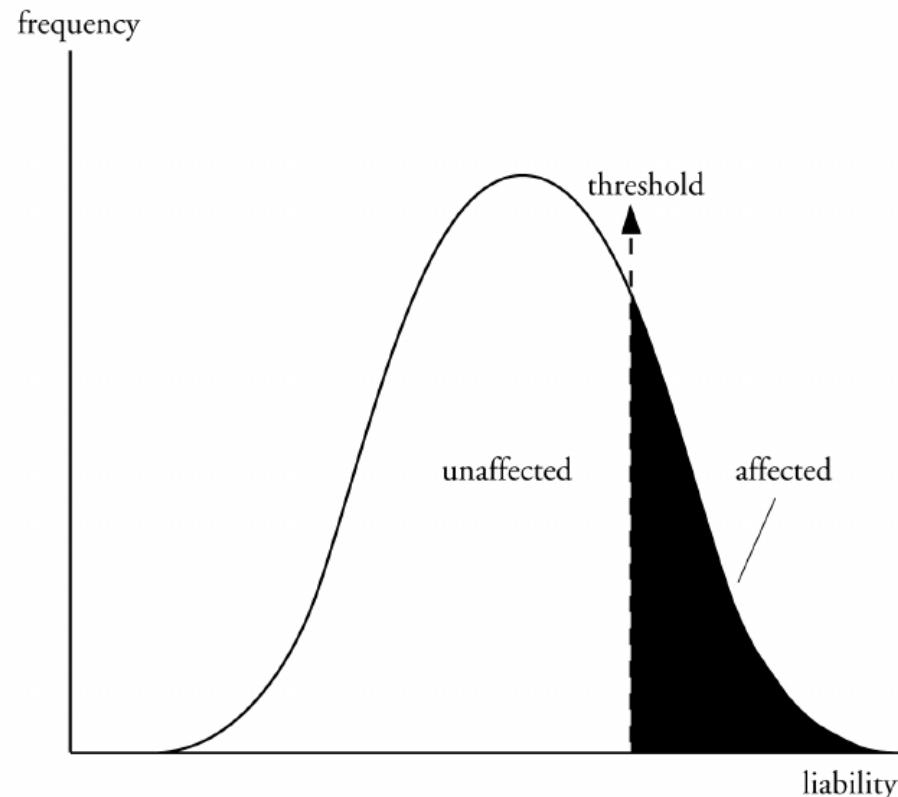
Use real data from a case-control study for the Celiac disease.



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

## Model "ADD"

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

## Model "COMP"

$$y_i = \underbrace{\sum_{j \in S_{\text{causal}}^{(1)}} w_j \cdot \widetilde{G}_{i,j}}_{\text{linear}} + \underbrace{\sum_{j \in S_{\text{causal}}^{(2)}} w_j \cdot \widetilde{D}_{i,j}}_{\text{dominant}} + \underbrace{\sum_{\substack{k=1 \\ j_1=e_k^{(3,1)} \\ j_2=e_k^{(3,2)}}}^{k=|S_{\text{causal}}^{(3,1)}|} w_{j_1} \cdot \widetilde{G}_{i,j_1} \widetilde{G}_{i,j_2}}_{\text{interaction}} + \epsilon_i$$

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

# Extension via feature engineering

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

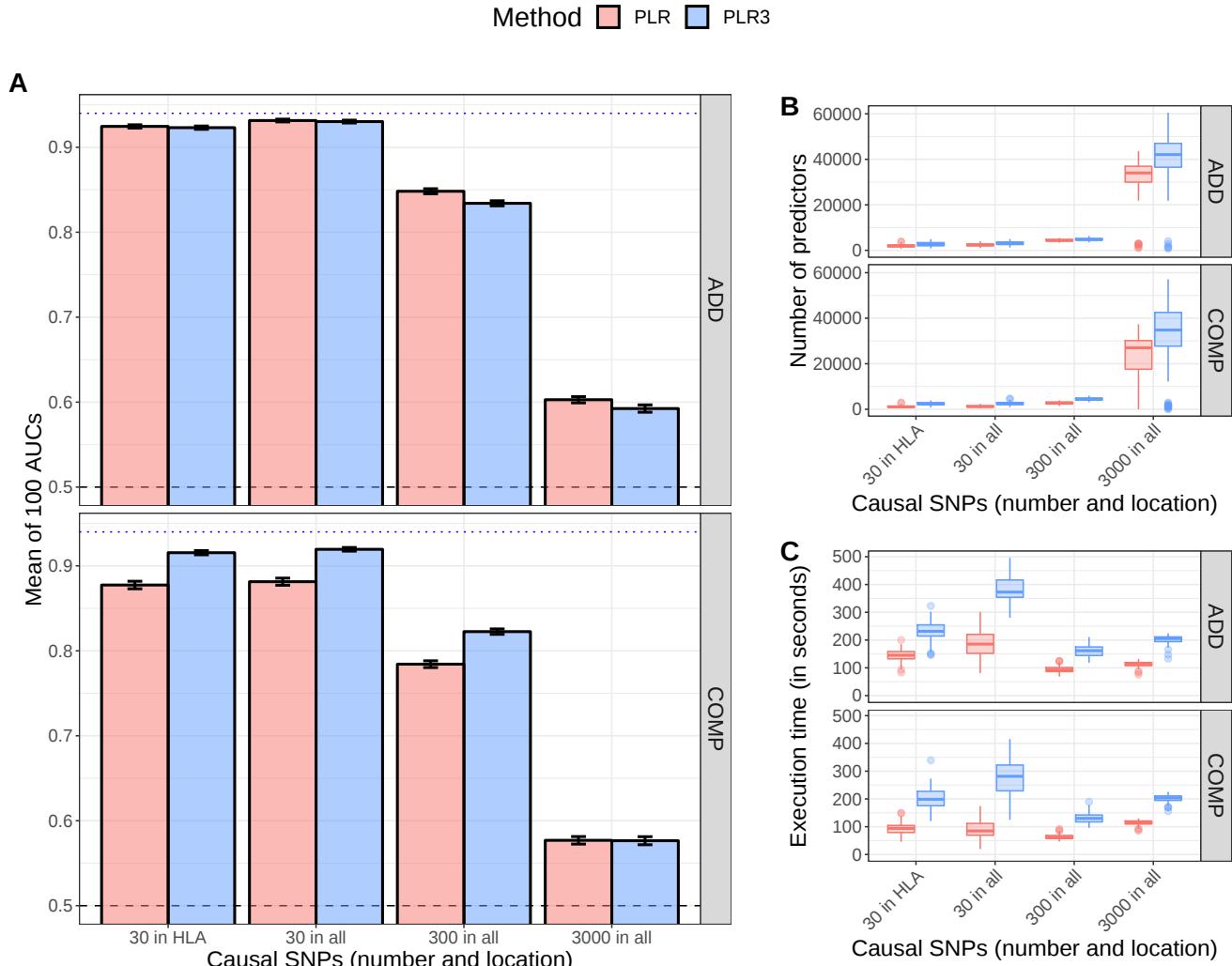
	SNP1	SNP2
[1,]	0	2
[2,]	0	1
[3,]	1	1
[4,]	0	2
[5,]	0	0
[6,]	1	0
[7,]	1	1
[8,]	0	1
[9,]	0	1
[10,]	0	2



	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 "PLR" and "PLR3".

# Feature engineering improves prediction



# Prediction with PLR is improving faster (Scenario #2)

