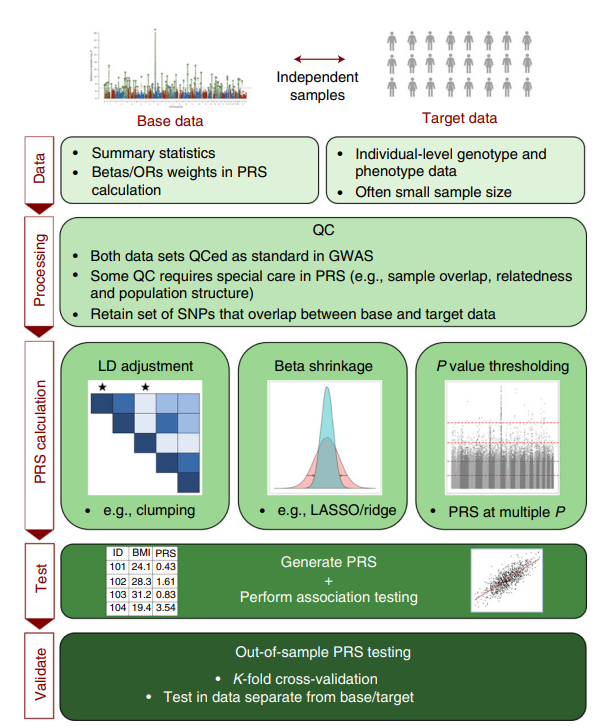
# PGS

## Summary

* Estimate of individual’s genetic liability to trait or disease
  + Currently explains a small proportion of trait variance
* sum of risk alleles an individual has, weighted by risk allele effect sizes estimated by GWAS on the phenotype
* Base data: GWAS summary statistics
* Target data: genotype-phenotype data of individuals in whom PGS are calculated

## Method

[Basic Tutorial for Polygenic Risk Score Analyses (choishingwan.github.io)](https://choishingwan.github.io/PRS-Tutorial/)



### QC for Base Data Only

1. Heritability check: use GWAS data with h^2 > 0.05
2. Effect allele: make identity of effect allele clear

### QC for Target Data Only

1. Target sample size > 100 individuals

### QC for Both Base and Target

1. Genome build: ensure base and target data SNPs have genomic positions assigned on same genome build (ex hg18)
2. Standard GWAS QC:
   1. Shing Wan Choi recommends: genotyping rate >0.99; sample missingness <0.02; HWE p>1x10^-6; heterozygosity within 3 stdev of mean; MAF >1%; info score >0.8
3. Ambiguous SNPs: remove
4. Mismatching SNPs: remove
5. Duplicate SNPs: remove
6. Sex Chromosomes: remove
7. Sample overlap:
8. Relatedness:

### Shrinkage of GWAS effect size estimates

1. Statistical shrinkage of all SNPs (ex. LASSO, ridge regression, Bayesian)
2. Clumping and thresholding (effectively shrinks all excluded SNPs to 0)
   1. Also controls for LD

### Population Structure

* Using covariates to adjust: 1st 10 PCs, birth year, and sex

### Interpret Results

* After PGSs are calculated, regression is performed on target sample
  + PGS is predictor of target trait, include covariates as appropriate
* Test for association between PGS and target trait
  + Get p-value, phenotypic variance explained (R^2), effect size estimate, etc
  + Nagelkerke R^2 estimates proportion of variance explained by PGS

## Tutorial Scripts

#### QC Target Data



#### PGS



#### PGS p-value Plot

#!/usr/bin/env Rscript

p.threshold <- c(0.001,0.05,0.1,0.2,0.3,0.4,0.5)

# Read in the phenotype file and PCs

#

setwd("~/Research/GWAS-frontera/1000G")

phenotype <- read.table("EUR.height", header=T)

pcs <- read.table("EUR.eigenvec", header=F)

setwd("~/Research/GWAS-frontera/GWAS\_Results/height/PGS")

# add the appropriate headers (6 PCs)

colnames(pcs) <- c("FID", "IID", paste0("PC",1:6))

# merge the files

pheno <- merge(phenotype, pcs, by=c("FID", "IID"))

# calculate the null model (model with PGS) using a linear regression

null.model <- lm(Height~., data=pheno[,!colnames(pheno)%in%c("FID","IID")])

# R2 of the null model

null.r2 <- summary(null.model)$r.squared

PGS.result <- NULL

for(i in p.threshold){

    PGS <- read.table(paste0("EUR\_female\_height\_PGS.",i,".profile"), header=T)

    print(head(PGS))

    # Merge PGS with phenotype matrix - only taje FID, IID and PGS

    pheno.PGS <- merge(pheno, PGS[,c("FID","IID", "SCORE")], by=c("FID", "IID"))

    # linear regression on Height with PGS and the covariates, ignoring the FID and IID

    model <- lm(Height~., data=pheno.PGS[,!colnames(pheno.PGS)%in%c("FID","IID")])

    # model R2

    model.r2 <- summary(model)$r.squared

    # R2 of PGS is simply calculated as the model R2 minus the null R2

    PGS.r2 <- model.r2-null.r2

    # We can also obtain the coeffcient and p-value of association of PGS as follow

    PGS.coef <- summary(model)$coeff["SCORE",]

    PGS.beta <- as.numeric(PGS.coef[1])

    PGS.se <- as.numeric(PGS.coef[2])

    PGS.p <- as.numeric(PGS.coef[4])

    # We can then store the results

    PGS.result <- rbind(PGS.result, data.frame(Threshold=i, R2=PGS.r2, P=PGS.p, BETA=PGS.beta,SE=PGS.se))

}

# Best result is:

print(PGS.result[which.max(PGS.result$R2),])

#### PLOT ####

library(ggplot2)

# generate a pretty format for p-value output

PGS.result$print.p <- round(PGS.result$P, digits = 3)

PGS.result$print.p[!is.na(PGS.result$print.p) &

                    PGS.result$print.p == 0] <-

    format(PGS.result$P[!is.na(PGS.result$print.p) &

                            PGS.result$print.p == 0], digits = 2)

PGS.result$print.p <- sub("e", "\*x\*10^", PGS.result$print.p)

# Initialize ggplot, requiring the threshold as the x-axis (use factor so that it is uniformly distributed)

ggplot(data = PGS.result, aes(x = factor(Threshold), y = R2)) +

    # Specify that we want to print p-value on top of the bars

    geom\_text(

        aes(label = paste(print.p)),

        vjust = -1.5,

        hjust = 0,

        angle = 45,

        cex = 4,

        parse = T

    )  +

    # Specify the range of the plot, \*1.25 to provide enough space for the p-values

    scale\_y\_continuous(limits = c(0, max(PGS.result$R2) \* 1.25)) +

    # Specify the axis labels

    xlab(expression(italic(P) - value ~ threshold ~ (italic(P)[T]))) +

    ylab(expression(paste("PGS model fit:  ", R ^ 2))) +

    # Draw a bar plot

    geom\_bar(aes(fill = -log10(P)), stat = "identity") +

    # Specify the colors

    scale\_fill\_gradient2(

        low = "dodgerblue",

        high = "firebrick",

        mid = "dodgerblue",

        midpoint = 1e-4,

        name = bquote(atop(-log[10] ~ model, italic(P) - value),)

    ) +

    # Some beautification of the plot

    theme\_classic() + theme(

        axis.title = element\_text(face = "bold", size = 18),

        axis.text = element\_text(size = 14),

        legend.title = element\_text(face = "bold", size =

                                        18),

        legend.text = element\_text(size = 14),

        axis.text.x = element\_text(angle = 45, hjust =

                                    1)

    )

# save the plot

ggsave("EUR\_female.height.bar.png", height = 7, width = 7)

# Log

### 9/8/2021

* Proportion for cross-validation: 400,000 samples (both); 230,000 (female); 200,000 (male)
  + 50,000 for target: 25,000 male/female
* Age + PCs + sex (to control for M-F diff)

### 9/9/2021

* Remove 3rd degree relatives or closer; 2 options
  + Keep only ids used in genetic principal components,
  + <https://elifesciences.org/articles/48376#s4> GWAS by sample characteristics section
  + <https://biobank.ctsu.ox.ac.uk/crystal/field.cgi?id=22020> data field
  + Calculate kinship coefficients, and remove coefficients greater than 0.0442 (3rd degree) using –king-cutoff
  + <https://plink.readthedocs.io/en/latest/GWAS/> Relatives section
* Split into test and training set; for phenotype data, get 50k, 25k female, 25k male
  + Total iids in ukb: 502,491
  + Total iids in height pheno: 499952
  + Total iids in sex\_ids.txt: 488249
  + **Steps**: Keep white British, remove relatives, merge on iids with sex\_ids.txt, random sample 25k for each sex
* Finally, if PLINK 2 determines that any samples and covariates are irrelevant to all regressions (e.g. a covariate could be zero-valued for all but one sample), they are removed before any variants are processed. You can use the '**pheno-ids**' modifier to make PLINK 2 report the remaining samples to (per-phenotype) [.id](https://www.cog-genomics.org/plink/2.0/formats#id) files. (When the sample set changes on chrX or chrY, .x.id and/or .y.id files are also written.)

### 9/10/2021

* How –score works <https://zzz.bwh.harvard.edu/plink/profile.shtml>
  + score is simply a sum across SNPs of the number of reference alleles (0,1 or 2) at that SNP multiplied by the score for that SNP
    - score for that SNP is the effect size

### 9/27/2021

* updates PGS scripts – mash v additive
* r = correlation between observed and expected
* r^2 = coefficient of determination, explained variation/total variation
* lm() produces the coefficient of determination – how good is our model