GWAS 9: Meta-analysis and summary statistics

Matti Pirinen, University of Helsinki

Latest update: 2.12.2020; first version: 20-Feb-2019

The slide set referred to in this document is "GWAS 9".

Consider GWAS data on n individuals and p SNPs. **GWAS summary statistics** can include a varying combination of the following information, for each variant l, or region reg,

- **Association statistics** $A_l = \left(\text{EA}_l, \widehat{\beta}_l, \text{SE}_l, P_l\right)$ where EA is the effect allele for which $\widehat{\beta}_l$ is reported. These statistics are of size $4p$, and hence their proportion compared to the whole data is $4/n$.
- **Information statistics** $I_l = (\text{EAF}_l, \text{INFO}_l, \text{QC}_l, \ldots)$, where EAF is the effect allele frequency (sometimes given only in controls), INFO is an imputation information score, and QC includes quality control measures, such as Hardy-Weinberg P-value and missingness rate of the genotype calls. Proportion to raw data is around $10/n$ (assuming 10 pieces of information per variant).
- **LD-matrix** R_{reg} for certain region(s) of the genome. For a region of size p_{reg}, the size of R_{reg} is about $\frac{1}{2}p_{\text{reg}}^2$, whereas that of the raw data is np_{reg}. Thus, the ratio is $p_{\text{reg}}/(2n)$.

When n is of order 1e+5, the summary statistics take only a tiny fraction of the space required by the raw data. Additionally, raw genotype-phenotye data are sensitive, personal data and cannot be shared freely, whereas usually there is no legal restrictions on sharing the GWAS summary statistics. For these reasons, large consortia, such as GIANT or CARDIoGRAMplusC4D, are distributing their results as summary statistics. Therefore there is a need for methods that can further analyze the summary statistics, e.g., in fine-mapping, in imputation, in heritability estimation or in gene-level testing. The utilization of summary statistics is reviewed by Pasaniuc & Price 2017.

Example 9.1. Reminder how the central association statistics are related to each other.

- If we are given `b.est`=$\widehat{\beta}$ and `se`=SE, we can compute the P-value as `pchisq((b.est/se)^2, df=1, lower=F)`.
- If we are given $\widehat{\beta}$ and P-value `pval`, we can compute SE as `sqrt(b.est^2 / qchisq(pval, df=1, lower=F))`.
- If we are given SE and P-value, we can compute `b.est`=$\widehat{\beta}$ for the trait increasing allele as `sqrt(se^2*qchisq(pval, df=1, lower=F))`. (We need to know which allele is the trait increasing from external information.)
- Assuming that no (strong) covariates have been applied, we can further infer one of sample size, MAF or case-proportion from SE given that the other two are known, according to the formulas in GWAS3. We can also estimate an SE from these study parameters, and use it with a known effect estimate to derive a P-value.

In these notes, we go through how the association summary statistics are produced through *meta-analysis* of individual GWAS and how some analyses, that we have so far done from raw data, could be done using only the summary data.

9.1. Meta-analysis

Suppose we have results from K independent GWAS on the same phenotype. (Here independent means that the samples of the GWAS are not overlapping.) Hence, for each variant l, we have K sets of GWAS association statistics $A_{kl} = \left(\widehat{\beta}_{kl}, \mathrm{SE}_{kl}, P_{kl}\right)$. How do we combine these K pieces of information into a single combined estimate of the effect size, SE and P-value?

A combination of summary-level results from two or more studies into a single estimate is called **meta-analysis** ("meta" refers to something happening at a higher level, meta-analysis is an "analysis of analyses"). A review of meta-analysis in GWAS by Evangelou & Ioannidis 2013.

In practice, all large GWAS are nowadays meta-analyses carried out by international consortia and a consortium may contain even over hundred individual studies. Often each study runs a GWAS and shares the summary statistics with a centralized analysis group that carries out the meta-analysis. While this approach avoids sharing sensitive individual-level genotype-phenotype data and also operates only with the light-weight summary data, it also restricts the set of possible downstream analyses, since only the marginal additive effect estimates are available. It has become clear that in order to maximize the scientific output from the consortia efforts, future meta-analyses should be designed so that all raw data will be collected in a single place. Unfortunately, this is not always easy because of legal issues related to the sharing of genotype-phenotype data.

Let's get back to the question how do we combine K sets of GWAS association statistics on same (or at least similar) phenotype: $A_{kl} = \left(\widehat{\beta}_{kl}, \mathrm{SE}_{kl}, P_{kl}\right)$.

The answer depends on what we assume about the possible variation between the true underlying effects β_{kl} for $k = 1, \ldots, K$.

9.1.1 Fixed-effects model The most common assumption is that all studies are measuring the same underlying quantity, i.e., $\beta_{1l} = \ldots = \beta_{Kl}$ and there are no (noticeable) differences in phenotype definitions and no distinct biases between the studies. This is called the fixed-effects model because the effect size is the same across the studies. In this case, the statistically most efficient unbiased linear estimator of the common effect size β is the **inverse-variance weighted (IVW)** estimator (here denoted by F, Fixed-effect estimator):

$$\widehat{\beta}_{l,F} = \frac{w_{1l}\widehat{\beta}_{1l} + \ldots + w_{Kl}\widehat{\beta}_{Kl}}{w_{1l} + \ldots + w_{Kl}} \tag{1}$$

$$\mathrm{SE}_{l,F} = (w_{1l} + \ldots + w_{Kl})^{-\frac{1}{2}}, \quad \text{where the weight} \tag{2}$$

$$w_{kl} = \frac{1}{\mathrm{SE}_{kl}^2} \text{ is the inverse-variance of study } k. \tag{3}$$

In statistics, the inverse of the variance is called **precision**. With that in mind, the above formulas are easy to remember:

- The weight given to each study in IVW estimator is proportional to the precision of the study and the weights sum to 1.
- The precision of the IVW estimator is the sum of the precisions of the individual studies.

Efficiency of IVW. It sounds intuitively reasonable to weight each estimate by its precision, but what is the mathematical argument behind this? Let's consider the case of two studies and assume that both yield unbiased estimates of the common effect size β, with precisions $w_i = 1/\mathrm{SE}_i^2$ for $i = 1, 2$. This means that

$$\mathrm{E}\left(\widehat{\beta}_i \,|\, \beta\right) = \beta \quad \text{and} \quad \mathrm{Var}(\widehat{\beta}_i) = \mathrm{SE}_i^2 = \frac{1}{w_i}, \quad \text{for } i = 1, 2.$$

Consider all possible linear estimators $t(u) = u\widehat{\beta}_1 + (1-u)\widehat{\beta}_2$ determined by value of $u \in [0,1]$. Estimator $t(u)$ is unbiased for all u as

$$\mathrm{E}(t(u) \mid \beta) = u\mathrm{E}\left(\widehat{\beta}_1 \mid \beta\right) + (1-u)\mathrm{E}\left(\widehat{\beta}_2 \mid \beta\right) = u\beta + (1-u)\beta = \beta.$$

Thus, on average, any weighting scheme gives the correct answer, and the question is, which one of these weightings gives the most precise combined estimate (= smallest variance).

$$\mathrm{Var}(t(u)) = \mathrm{Var}\left(u\widehat{\beta}_1\right) + \mathrm{Var}\left((1-u)\widehat{\beta}_2\right) = u^2\frac{1}{w_1} + (1-u)^2\frac{1}{w_2} = u^2\left(\frac{1}{w_1} + \frac{1}{w_2}\right) - \frac{2}{w_2}u + \frac{1}{w_2}.$$

This is a second order polynomial with respect to u and has its minimum where the derivative is 0, i.e., at $u_0 = \frac{2/w_2}{2(1/w_1+1/w_2)} = \frac{w_1}{w_1+w_2}$, which is the IVW. We conclude that IVW is **the minimum variance unbiased linear estimator** of the fixed effect model.

Example 9.2. Suppose that we do a fixed-effect meta-analysis using IVW of two studies on LDL-cholesterol where the sample sizes of the studies are $n_1 = 5,000$ and $n_2 = 10,000$. If both studies have applied similar covariates and hence have similar error variance σ^2_ε, then the precisions of the studies are $w_i = 2n_i f_i(1-f_i)/\sigma^2_\varepsilon$ for $i = 1, 2$. At a SNP that has same MAF f in both studies, the precisions are proportional to n_i and hence the weights of the IVW are $\frac{w_1}{w_1+w_2} = 0.333$ and $\frac{w_2}{w_1+w_2} = 0.666$ and the precision of the IVW is $w_F = w_1 + w_2 = 2(n_1+n_2)f(1-f)/\sigma^2_\varepsilon$, which is the same as precision from a study with $n_1 + n_2 = 15,000$ samples. Indeed, with linear model, the precision from splitting the data into any subsets and then combining them with IVW is (approximately) the same as doing a joint analysis of all the data with separate intercept terms for each subset (Exercise). (If subsets are small and there are covariates involved, then random noise causes some numerical differences between the approaches.)

Example 9.3. Suppose that we do IVW meta-analysis of two studies on Parkinson's disease where $n_1 = 10,000$ of which $3,000$ are cases ($\phi_1 = 0.3$) and $n_2 = 6,000$ of which $3,000$ are cases ($\phi_2 = 0.5$). Thus the effective sample sizes are $n_{e1} = 10000 \cdot 0.3 \cdot 0.7 = 2100$ and $n_{e2} = 6000 \cdot 0.5 \cdot 0.5 = 1500$. If we assume that the MAF of the SNP is the same in both studies, then the precisions of the studies are $w_i = 2n_{ei}f(1-f)$ for $i = 1, 2$ and the weights of the IVW are $\frac{w_1}{w_1+w_2} = \frac{2100}{3600} = 0.583$ and $\frac{w_2}{w_1+w_2} = \frac{1500}{3600} = 0.417$ and the precision of the IVW estimator is $w_F = w_1 + w_2 = 2(n_{e1}+n_{e2})f(1-f)$, which is the same as precision from a study with effective sample size of $n_{e1} + n_{e2}$. It can be shown that by splitting the case-control data into subsets, the sum of the effective sample sizes over the subsets is always $\leq$ the effective sample size of the whole data (and the equality holds when the case proportion is constant across the subsets). This gives a technical explanation why, in logistic regression, an inclusion of a binary covariate, such as sex or population label, causes a decrease in precision, and hence increase in SE compared to a single joint analysis of all data without the covariate. This follows because a use of a binary covariate is approximately equivalent to splitting the data by the covariate value, analyzing subsets separately and combining the results using IVW (Exercise).

Example 9.4. Consider the association statistics at SNP rs11984041 on the large vessel subtype of Ischemic Stroke in Bellenguez et al. (2012). They reported a discovery OR 1.50 (1.25-1.79) and replication1 OR 1.38 (1.17-1.63) and replication2 OR 1.39 (1.15-1.68), all for the same allele. What is the combined estimate, SE and P-value using the fixed-effects meta-analysis? (They report 1.42 (1.28-1.57), P=1.87e-11.)

Answer. Let's make a function `meta.F()` that does the IVW meta-analysis for given estimates and SEs.

```
meta.F <- function(b.est, se){
  #returns inverse-variance weighted meta-analysis estimate, SE and P-value.
  b.F = sum(b.est / se^2) / sum(1 / se^2)
  se.F = 1 / sqrt(sum(1 / se^2))
  p.F = pchisq( (b.F / se.F)^2, df = 1, lower = F)
  return(list(b.F = b.F, se.F = se.F, p.F = p.F))
}
```

With these data, we need to compute the SEs from the 95%CIs and then use IVW.

```
b.est = log(c(1.50, 1.38, 1.39)) #beta is logOR for case-control data
ci = log(matrix(c(1.25, 1.79,
                  1.17, 1.63,
                  1.15, 1.68), byrow = T, ncol = 2))
se = (ci[,2] - ci[,1])/(2*1.96) #length of 95%CI is 2*1.96*SE
meta.res = meta.F(b.est, se)
meta.res
```

```
## $b.F
## [1] 0.3513526
##
## $se.F
## [1] 0.05227737
##
## $p.F
## [1] 1.805663e-11
```

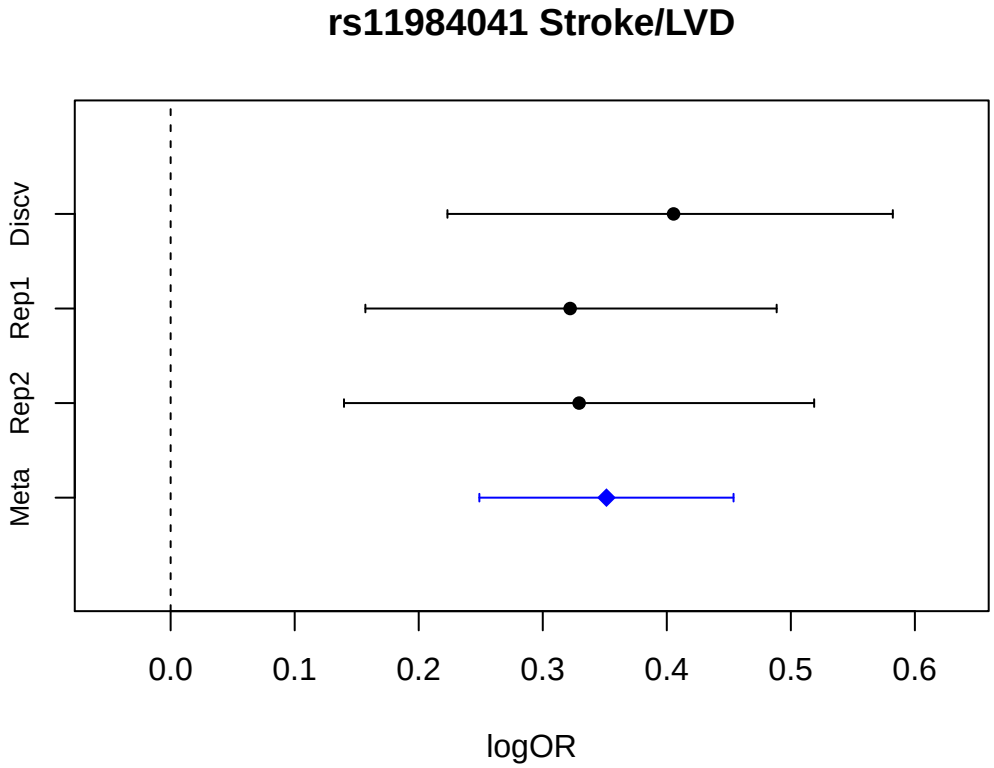
```
cbind(OR = exp(meta.res$b.F), low = exp(meta.res$b.F - 1.96*meta.res$se.F),
      up = exp(meta.res$b.F + 1.96*meta.res$se.F), pval = meta.res$p.F)
```

```
##            OR    low       up         pval
## [1,] 1.420988 1.2826 1.574309 1.805663e-11
```

The results match well given the accuracy of two decimals in CIs to compute SE. Let's visualize them by a forest plot.

```
forest.plot <- function(x, intervals, labels = NULL, main = NULL, xlab = "Effect size",
                        pchs = rep(19,length(x)), cols = rep("black", length(x)),
                        cexs = rep(1,length(x))){
  K = length(x)
  stopifnot(nrow(intervals) == K)
  plot(0, col="white", xlim = c( min(c(intervals[,1],0) - 0.05), max(c(intervals[,2],0) + 0.05)),
       ylim = c(0, K+1), xlab = xlab, ylab = "", yaxt = "n",main = main)
  axis(2, at = K:1, labels = labels, cex.axis = 0.8)
  arrows(intervals[,1], K:1, intervals[,2], K:1,
         code = 3, angle = 90, length = 0.02, col = cols)
  points(x, K:1, pch = pchs, cex = cexs, col = cols)
  abline(v = 0,lty = 2)
}
```

```
b.est = c(b.est, meta.res$b.F)
ci = rbind(ci, c(meta.res$b.F + c(-1,1)*1.96*meta.res$se.F))
labs = c("Discv", "Rep1", "Rep2", "Meta")
main.txt = "rs11984041 Stroke/LVD"
forest.plot(b.est, ci, labels = labs, main = main.txt, xlab = "logOR",
            pchs = c(19, 19, 19, 18), cexs = c(.8, .8, .8, 1.3), cols = c(1, 1, 1, 4))
```



The plot shows that the estimates are very consistent with each other. We also see how the uncertainty has decreased in the combined estimate compared to the individual studies. (In practice, visualizations are recommended to be done with existing R-packages such as `meta` that have many more options.)

9.1.2 Heterogeneity When we talk about *heterogeneity* in meta-analysis we mean that the true effect sizes between studies are different. How can we assess that from the observed data? Let's first list heterogeneity measures that are typically reported in meta-analyses.

Cochran's Q. Assume as the null hypothesis that all K studies are measuring the same effect β and that the errors are Normally distributed: $\widehat{\beta}_k \sim \mathcal{N}(\beta, \mathrm{SE}_k^2)$, for study k. Then each $z_k = (\widehat{\beta}_k - \beta)/\mathrm{SE}_k \sim \mathcal{N}(0,1)$, and, assuming that the studies are independent, the sum of the squares of these K independent standard Normals has a chi-square distribution with K degrees of freedom: $\sum_{k=1}^{K} z_k^2 \sim \chi_K^2$. When we replace true β with the fixed-effect estimate $\widehat{\beta}_F$ from IVW method, then we have a heterogeneity measure called **Cochran's Q**:

$$Q = \sum_{k=1}^{K} \left(\frac{\widehat{\beta}_k - \widehat{\beta}_F}{\mathrm{SE}_k} \right)^2 = \sum_{k=1}^{K} w_k \left(\widehat{\beta}_k - \widehat{\beta}_F \right)^2,$$

that under the null hypothesis of the fixed-effect assumption has distribution χ^2_{K-1} (where one degree of freedom is lost as we have used the data to estimate the common mean $\widehat{\beta}_F$ to replace the true β). This distribution can be used to derive P-values for heterogeneity. However, when K is small (say < 5), there is little power to see any heterogeneity with this test, and when K is large (say > 100), then even a small heterogeneity becomes statistically significant.

I^2. To change the focus from P-value to the amount of heterogeneity, a heterogeneity index I^2 has been proposed:

$$I^2 = \frac{Q - (K-1)}{Q} = 1 - \frac{K-1}{Q}.$$

This value is between 0 and 1 and often reported as percentage (negative values are rounded up to 0). The idea is that if Q shows only the null hypothesis expectation of variation (which is $K-1$), then $I^2 = 0\%$

indicating no heterogeneity, whereas values $> 50\%$ are interpreted as moderate amount of heterogeneity and $> 75\%$ high heterogeneity. With a small number of studies, the uncertainty around estimate of I^2 is large and little can inferred statistically.

Between study variance T^2. Often a model for heterogeneity between the true effects is a two-stage hierarchical model where the heterogeneity is defined by a variance parameter T^2. We assume that first each true effect is sampled from $(\beta_k|\beta, T^2) \sim \mathcal{N}(\beta, T^2)$ and then our estimates are formed by adding some noise around these values as $(\widehat{\beta}_k|\beta_k) \sim \mathcal{N}(\beta_k, \mathrm{SE}_k^2)$. From this model, a commonly-used estimate for T^2 is

$$\widehat{T^2} = \frac{Q - (K-1)}{\sum_{k=1}^K w_k - \frac{\sum_{k=1}^K w_k^2}{\sum_{k=1}^K w_k}}.$$

This is on the same scale as the (squared) effect sizes, which makes it different from Q and I^2 that are independent of the effect size scale.

When $T^2 > 0$ in the model formulation above, we have defined the standard **random-effects** meta-analysis model where the effect sizes across studies are not assumed to be exactly the same but still they are possibly quite similar (namely, when T is small compared to the common mean β of all effects). In statistics literature, it is common to derive a P-value assuming $\beta = 0$ from such a model and call that the random-effects model's P-value. Such a test is not suitable for GWAS where the relevant null hypothesis is that all effects are 0, rather than that only their mean is 0. This issue is explained by Han & Eskin 2011 and they also propose a modification that tests the null hypothesis of exactly 0 effect in every study.

As a conclusion, the three quantities listed above to measure heterogeneity are often reported in meta-analyses, but they are often not that informative in cases where there are only a handful of studies.

9.1.3 Bayesian meta-analysis The question of heterogeneity between studies can be more flexibly defined in the Bayesian framework where a set of models with different assumptions about heterogeneity can be directly compared against each other and the interpretation of the results of the model comparison does not depend on the sample size.

Let's remind ourselves how, in GWAS 4, we compared the model with a non-zero effect to the null model using the approximate Bayes factor, ABF.

We assumed that under the alternative hypothesis H_1, there was a non-zero effect $\beta \sim \mathcal{N}(0, \tau^2)$ whereas under the null hypothesis $\beta = 0$. Then we derived the marginal likelihoods that these models give for the observed data, and these marginal likelihoods were proportional to Normal densities:

$$P(\mathrm{Data} \,|\, H_1) = c \cdot \mathcal{N}(\widehat{\beta}; 0, \mathrm{SE}^2 + \tau^2) \tag{4}$$
$$P(\mathrm{Data} \,|\, H_0) = c \cdot \mathcal{N}(\widehat{\beta}; 0, \mathrm{SE}^2) \tag{5}$$

From this we got an approximate Bayes factor in favor of H_1 vs. H_0 as

$$\mathrm{ABF}_{1:0} = \frac{P(\mathrm{Data} \,|\, H_1)}{P(\mathrm{Data} \,|\, H_0)} = \frac{\mathcal{N}(\widehat{\beta}; 0, \mathrm{SE}^2 + \tau^2)}{\mathcal{N}(\widehat{\beta}; 0, \mathrm{SE}^2)}.$$

Finally, we derived the posterior probability of H_1 under the assumption that one of H_0 and H_1 is true and that the prior probability of it being H_1 was $p_1 = 1 - p_0$:

$$P(H_1 \,|\, \mathrm{Data}) = \frac{p_1 \cdot \mathrm{ABF}_{1:0}}{p_0 + p_1 \cdot \mathrm{ABF}_{1:0}}.$$

Thus, in order to get to a probabilistic model comparison, we needed to set values for two parameters: τ that determines how large effects we expect to see under H_1, and p_1 that determines how probable we think that the alternative hypothesis is *a priori*, before we have seen the data.

Let's see how we generalize this to multiple studies. We use a multivariate Normal distribution as the prior distribution of the effect size vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^T \sim \mathcal{N}_K(0, \boldsymbol{\Theta})$, where the prior matrix $\boldsymbol{\Theta}$ is assumed to take the form

$$\boldsymbol{\Theta} = \tau^2 \begin{bmatrix} 1 & \theta_{12} & \ldots & \theta_{1K} \\ \theta_{12} & 1 & \ldots & \theta_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{1K} & \theta_{2K} & \ldots & 1 \end{bmatrix}.$$

Thus, the parameter τ^2 still defines the prior variance of any one effect size parameter, but now effect sizes from two studies may be correlated as defined by prior correlation θ_{ij}. For example,

- the fixed-effect model H_F results if we set all $\theta_{ij} = 1$,
- the independent-effect model H_I results if we set $\theta_{ij} = 0$,
- the standard random-effect model $H_R(\rho)$ results if we set $\theta_{ij} = \rho$ for some value of $\rho > 0$, where values close to 1 assume only little heterogeneity and values close to 0 assume almost independent effects; our default is $\rho = 0.9$,
- the null model H_0 is defined by setting $\tau^2 = 0$ (and then the values of θ_{ij} do not matter).

The likelihood function defined by the observed data is also proportional to a multivariate Normal density $\mathcal{N}_K(\widehat{\boldsymbol{\beta}}; \boldsymbol{\beta}, \boldsymbol{\Sigma})$. If we assume that the studies are independent (no overlapping samples), then $\boldsymbol{\Sigma}$ is simply a diagonal matrix where the diagonal is $(\mathrm{SE}_1^2, \ldots, \mathrm{SE}_K^2)$.

The marginal likelihood for data given the model m (defined by prior variance matrix $\boldsymbol{\Theta}_m$) is

$$P(\text{Data} \,|\, H_m) = c \cdot \mathcal{N}_K(\widehat{\boldsymbol{\beta}}; \mathbf{0}, \boldsymbol{\Sigma} + \boldsymbol{\Theta}_m).$$

Approximate Bayes factor between any two models m and ℓ is

$$\mathrm{ABF}_{m:\ell} = \frac{P(\text{Data} \,|\, H_m)}{P(\text{Data} \,|\, H_\ell)} = \frac{\mathcal{N}_K(\widehat{\boldsymbol{\beta}}; \mathbf{0}, \boldsymbol{\Sigma} + \boldsymbol{\Theta}_m)}{\mathcal{N}_K(\widehat{\boldsymbol{\beta}}; \mathbf{0}, \boldsymbol{\Sigma} + \boldsymbol{\Theta}_\ell)}.$$

If model m is given a prior probability p_m (and $p_0 + \ldots + p_K = 1$), then we can compute the posterior probability for model m as

$$P(H_m \,|\, \text{Data}) = \frac{p_m \cdot \mathrm{ABF}_{m:0}}{\sum_{\ell=0}^{K} p_\ell \cdot \mathrm{ABF}_{\ell:0}},$$

where we have computed ABFs between all models and the null model (and $\mathrm{ABF}_{0:0} = 1$). Thus, the posterior probability of a model is proportional to the product of the prior probability of the model and ABF of the model.

Let's write a function `abf.mv()` that computes ABFs and posterior probabilities for any given set of prior matrices and prior probabilities. First, we need a density for the multivariate normal. (There is also a package `mvtnorm` for that.)

```
log.dmvnorm <- function(x, mu = rep(0, length(x)), S = diag(1, length(x)) ){
  #returns log of density of MV-Normal(mean = mu, var = S) at x
  K = length(mu)
  stopifnot(all(dim(S) == K))
```

```
  stopifnot(length(x) == K)
  chol.S = chol(S) #Cholesky decomposition
  log.det = 2*sum(log(diag(chol.S))) #log of det(S)
  inv.chol.S = solve(t(chol.S)) #inverse of cholesky^T
  return(-K/2*log(2*pi) - 0.5*(log.det + crossprod(inv.chol.S %*% (x-mu))))
}

abf.mv <- function(b.est, Sigmas, prior = rep(1,length(Sigmas))){
  #Returns posterior probabilities of the models listed in Sigmas by their
  #   total variance matrix (= sum of prior + likelihood variance matrices)
  #Returns also ABFs w.r.t the first model in Sigmas.

  M = length(Sigmas) #number of models
  K = length(b.est) #number of studies
  prior = prior/sum(prior)
  log.abf = sapply(Sigmas, function(x){log.dmvnorm(b.est, S = x)})
  abf = exp(log.abf - log.abf[1]) #abf w.r.t the first model
  posterior = prior*abf
  posterior = posterior/sum(posterior)
  return(list(posterior = posterior, abf = abf))
}
```

Example 9.5. Let's generate 4 data sets for 10 case-control studies where the effective sample size varies between 250 and 2500 and MAF varies between 0.4 and 0.5.

- 1st data set: all studies estimate the same effect $\beta = 0.1$.
- 2nd data set: there is heterogeneity and the true effects come from $\mathcal{N}(0.1, 0.04^2)$.
- 3rd data set: there is heterogeneity but no correlation in effects as they come from $\mathcal{N}(0, 0.1^2)$.
- 4th data set has a null SNP in all studies.

```
K = 10
n.eff = runif(K, 250, 2500)
f = runif(K, 0.4, 0.5)
se = 1/sqrt(2*n.eff*f*(1-f)) #SEs
w = 1/se^2 #precisions
b = 0.1 #true mean of effects in 1 and 2
B.est = cbind(rnorm(K, b, se), #fixed effects
              rnorm(K, b, sqrt(se^2 + 0.04^2)), #correlated random effects
              rnorm(K, rep(0,K), sqrt(se^2 + 0.1^2)), #independent effects
              rnorm(K, rep(0,K), se)) #null model
```

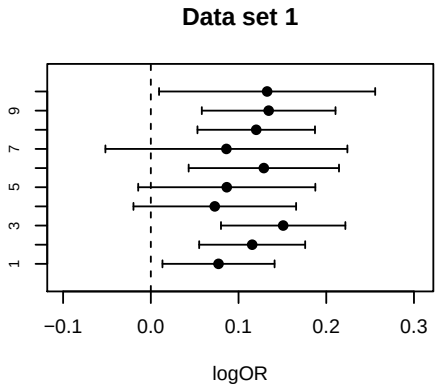
Let's then compare 4 models in these data sets: (1) fixed-effects, (2) correlated-effects, (3) independent-effects and (4) the null.

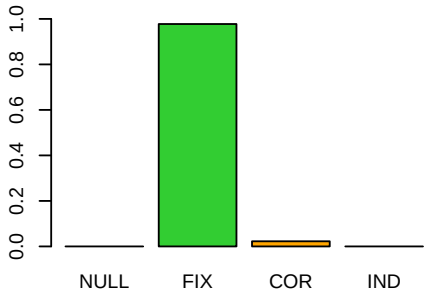
We specify these models by their matrices $(\mathbf{\Sigma} + \mathbf{\Theta}_m)$, run `abf.mv()` and print the forest plot of the data as well as a barplot of the posterior probability across the 4 competing models, assuming the prior probability of each model is the same (= 0.25).

Let's also print the standard heterogeneity measures: value of I^2 and the P-value from Cochran's Q-statistic.

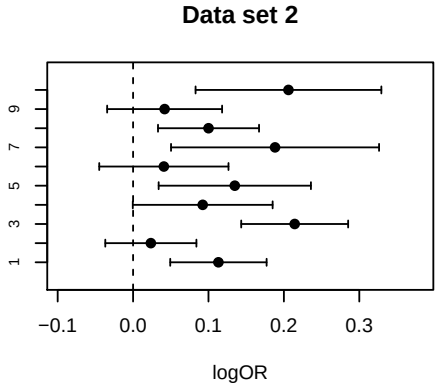
```
Sigma = diag(se^2) #this is the variance of likelihood function -- same for all models
tau2 = 0.2^2 #prior variance of effect size -- same for all non-null models
S.fix = tau2 * matrix(1, K, K) #fixed-effect model has 1s in the correlation matrix
```

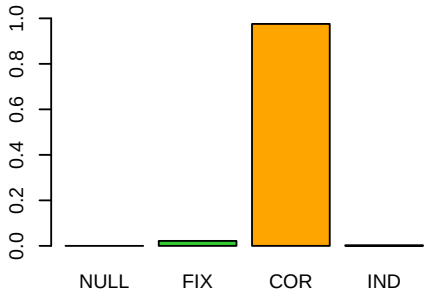
```r
S.cor = tau2 * matrix(0.9, K, K) #correlated effects has corr of 0.9 as off-diagonal
diag(S.cor) = tau2 #... and corr of 1 on the diagonal
S.ind = tau2 * diag(K) #diagonal matrix for independent effects model, off-diagonals = 0
S.null = matrix(0, K, K) #null model has 0 effects
Var.matrices = list(Sigma + S.null, Sigma + S.fix, Sigma + S.cor, Sigma + S.ind)

par(mfrow = c(4,2))
for(ii in 1:ncol(B.est)){
  #Standard heterogeneity measures:
  b.F = sum(w*B.est[,ii]) / sum(w) #IVW estimate under fixed-effect model
  Q = sum( w * (B.est[,ii] - b.F)^2 ) #Cochran's Q
  pval.Q = pchisq(Q, df = K-1, lower = F)
  I2 = 1 - (K-1)/Q #I^2 from Q

  #Bayesian model comparison:
  abf.out = abf.mv(B.est[,ii], Sigmas = Var.matrices) #by default, prior is uniform
  ci = cbind(B.est[,ii] - 1.96*se, B.est[,ii] + 1.96*se) #95%CIs
  forest.plot(B.est[,ii], ci, main = paste("Data set",ii), xlab = "logOR")
  barplot(abf.out$posterior, ylim = c(0,1), cex.sub = 1.3,
          sub = paste0("I2=",max(c(0,round(I2*100))),"%  het P=",signif(pval.Q,2)),
          names.arg = c("NULL", "FIX", "COR", "IND"),
          col = c("gray","limegreen","orange","dodgerblue"))
}
```



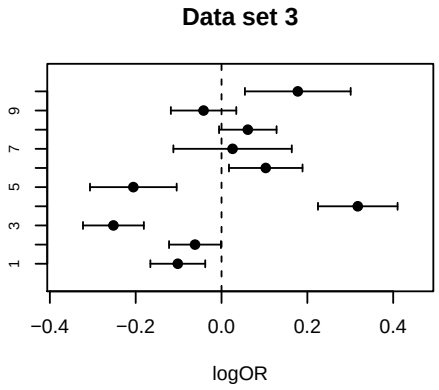


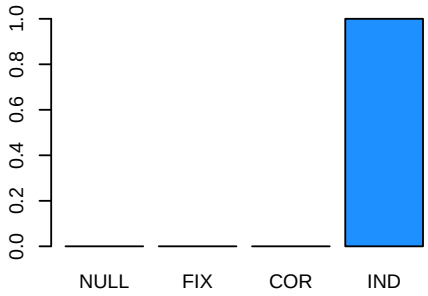
I2=0% het P=0.91



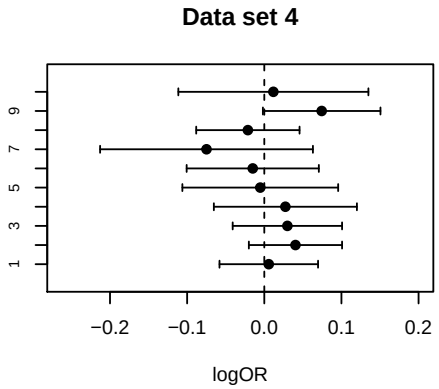


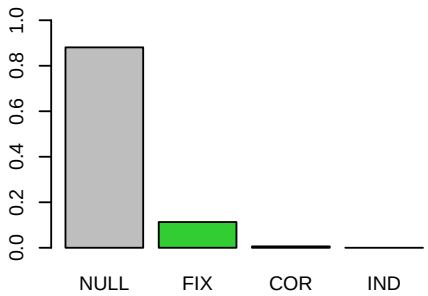
I2=64% het P=0.0027





I2=93% het P=4.2e−25





I2=0% het P=0.67

The Bayesian approach gives a way to assess whether there is heterogeneity in the effect sizes by comparing correlated effects model and/or independent effects model to the fixed effect model. Above it indicates heterogeneity in data sets 2 and 3, as expected. Similarly the I^2 value together with P-value from Q indicate heterogeneity in sets 2 and 3.

Extensions of the Bayesian approach to overlapping samples between studies, e.g., due to shared controls, and to the subset models, where the effect is non-zero only in particular studies, are discussed by Trochet et al. 2019.

As the final comment about heterogeneity: Whenever you have an interesting variant in a GWAS meta-analysis, make a forest plot over all the cohorts to see how the effects look like and don't rely only on some quantitative heterogeneity measures, especially if there are only a couple of studies included.

9.1.4 Publication bias A crucial part of all meta-analyses is to include in the analysis *all* the data available on the particular research question. In particular, one should never use the results of the studies to decide which studies to include or leave out since that will obviously bias the results of the meta-analysis. Studies can be left out because of quality issues or differences in phenotypes, for example, but these must be objective criteria that are not based on the results of the study in any one SNP. In general, meta-analyses in epidemiology and social science etc. are hampered by **publication bias** which means that only studies reporting statistically significant results are published whereas null studies never find their way to public. Consequently, a meta-analysis may report a significant effect based on published studies even though there could be another set of unpublished studies that could show that, when all information is combined, there is no effect. The pubication bias is less of a problem in GWAS, because GWAS results are published simultaneously genome-wide, not separately for the "significant" SNPs.

9.2. Further analyses with summary statistics

Meta-analysis yields a set of association statistics ($\widehat{\beta}$, SE, P-value). Let's look at how we can do some of the downstream analyses with these pieces of information without an access to the full raw genotype-phenotype data.

9.2.1. Joint model of multiple SNPs Let's consider the joint linear model with p SNPs with the mean centered phenotype y and standardized genotypes (and then we can drop the intercept term from the model):

$$\boldsymbol{y} = \boldsymbol{X^*}\boldsymbol{\lambda}^* + \boldsymbol{\varepsilon}.$$

The least squares estimator and its variance are

$$\widehat{\boldsymbol{\lambda}}^* = \left(\boldsymbol{X^*}^T\boldsymbol{X^*}\right)^{-1}\boldsymbol{X^*}^T\boldsymbol{y}, \tag{6}$$

$$\mathrm{Var}\left(\widehat{\boldsymbol{\lambda}}^*\right) = \sigma_J^2\left(\boldsymbol{X^*}^T\boldsymbol{X^*}\right)^{-1}, \tag{7}$$

where

- $\sigma_J^2 = \mathrm{Var}(\varepsilon) = \mathrm{Var}(y) - (\widehat{\boldsymbol{\lambda}}^*)^T\boldsymbol{R}\widehat{\boldsymbol{\lambda}}^*$ is the error variance from the Joint model and $\boldsymbol{R}$ is the LD matrix of the SNPs in $\boldsymbol{X}$.

It turns out that these quantities can be written using summary data from the marginal models $\boldsymbol{y} = \boldsymbol{x}_l^*\beta_l + \boldsymbol{\varepsilon}_l$, since

$$\left(\boldsymbol{X^*}^T\boldsymbol{X^*}\right) = n\boldsymbol{R} \tag{8}$$

$$\boldsymbol{X^*}^T\boldsymbol{y} = n\widehat{\boldsymbol{\beta}}^* \tag{9}$$

$$\mathrm{Var}(y) = (\widehat{\beta}_l^*)^2 + \widehat{\sigma}_l^2 = (\widehat{\beta}_l^*)^2 + 2nf_l(1-f_l)\cdot\mathrm{SE}_l^2, \tag{10}$$

where f_l is the MAF of SNP l and SE_l is the standard error of the *allelic* marginal effect $\widehat{\beta}_l$, as reported by GWAS.

With these formulas we have that

$$\widehat{\boldsymbol{\lambda}}^* = \boldsymbol{R}^{-1}\widehat{\boldsymbol{\beta}}^*, \tag{11}$$

$$\mathrm{Var}\left(\widehat{\boldsymbol{\lambda}}^*\right) = \frac{\widehat{\sigma_J^2}}{n}\boldsymbol{R}^{-1}, \tag{12}$$

$$\widehat{\sigma_J^2} = \mathrm{median}_{l=1}^{p}\left\{(\widehat{\beta}_l^*)^2 + 2nf_l(1-f_l)\cdot \mathrm{SE}_l^2\right\} - (\widehat{\boldsymbol{\beta}}^*)^T\boldsymbol{R}^{-1}\widehat{\boldsymbol{\beta}}^*, \tag{13}$$

where the median is taken over all the available SNPs in $\boldsymbol{X}$ and its function is to reduce noise compared to the corresponding variance estimate taken from any one l. In particular, we do not need an access to raw $\boldsymbol{X}$ and $\boldsymbol{y}$ in order to do a stepwise forward selection or probabilistic fine-mapping as long as we have the marginal GWAS association statistics and the LD matrix available.

If association statistics come from an IVW fixed-effect meta-analysis, then the LD-matrix is a weighted sum of the LD-matrices of individual studies, where the weights are proportional to the sample sizes of the studies Before the meta-analysis, one must make sure that all the studies have measured the effects on the same scale in order that the fixed-effect meta-analysis of the effect sizes makes sense. Typically, this is ensured by normalizing the trait to have a variance of 1 in each study before running the GWAS.

If the association statistics are coming from a **logistic regression model applied to case-control data**, we modify the above formulas by setting $\sigma_J^2 = 1$ and replacing the sample size n with the effective sample size n_e. If summary data come from one study, then $n_e = n\phi(1-\phi)$, where ϕ is the proportion of cases in data. If summary data are from a meta-analysis over several studies, then n_e is the sum of the effective sample sizes $n_e^{(i)} = n^{(i)}\phi_i(1-\phi_i)$ over individuals studies, where $n^{(i)}$ is the total sample size (cases + controls) of study i and ϕ_i is the proportion of cases in study i. In this meta-analysis case, the LD-matrix is a weighted sum of the LD-matrices of individual studies, where the weights are proportional to the effective sample sizes of the studies. This approximation works well when the effect sizes are not very large, MAF is not very small and ϕ is quite balanced, say, within (0.2,0.8).

The idea of computing the joint model from the summary statistics was introduced by Yang et al. 2012 and has been widely used through the conditional and joint analysis (COJO) module of the software package GCTA. The same idea is used in many fine-mapping software packages such as FINEMAP.

Let's try it with two SNPs and a quantitative trait. (Uses `geno.2loci()` from Section 7 to generate genotypes.)

```
n = 2000
maf = c(0.3, 0.4)
lambda = c(0.2, -0.05) #true causal effects
r = 0.5 #LD btw SNPs 1 & 2
X = geno.2loci(n, r, mafs = maf, return.geno = TRUE)
R = cor(X) #LD in data
f = colSums(X)/2/n #MAFs in data
sc = sqrt(2*f*(1-f)) #scaling constants
y = scale(X %*% lambda + rnorm(n, 0, sqrt(1 - var(X %*% lambda))))
b.est = rbind(summary(lm(y ~ X[,1]))$coeff[2,1:2],
              summary(lm(y ~ X[,2]))$coeff[2,1:2]) #col1 estimate, col2 SE
b.s = b.est[,1]*sc #scaled betas
l.s.sumstat = solve(R, b.s) #computes scaled lambdas as R^-1 * b.s
sigma2.J.sumstat = median(b.s^2 + n*sc^2*b.est[,2]^2) - as.vector(t(b.s) %*% solve(R, b.s))
l.s.se.sumstat = sqrt(sigma2.J.sumstat/n * diag(solve(R))) #SEs of scaled lambdas
cbind(lambda = l.s.sumstat/sc, se = l.s.se.sumstat/sc) #show on allelic scale
```

```
##           lambda         se
```

```
## [1,]  0.21911598 0.03996448
## [2,] -0.05144214 0.03695407
```

```
#Compare to the joint model on raw data
summary(lm(y ~ X))$coeff[2:3,1:2]
```

```
##       Estimate Std. Error
## X1  0.21883814 0.04014083
## X2 -0.05050388 0.03681416
```

Same results up to the 3rd decimal. (Binary trait case left as an exercise.)

Given that the association summary statistics from large meta-analyses are publicly available, it would be very nice if we could do joint models and fine-mapping by combining those statistics with LD-information from some **reference database**, without needing the access to the original genotypes. Indeed, this is how GCTA-COJO analyses are done. Unfortunately, with recent large datasets, such as the UK Biobank, it has become clear that the accuracy of the LD-estimates must increase together with the GWAS sample size. Otherwise, the summary statistic methods start reporting false positives because of the inconsistency between the highly precise effect estimates and the LD information from the reference data (Benner et al. 2017). Hence, in general, we will need LD-information from the same data from which the GWAS summary statistics were calculated in order to do reliable fine-mapping and joint analysis. This is one reason why future meta-analyses should be planned in such a way that all data are collected in one place, and why we will need new ways to seamlessly distribute LD-information as another type of GWAS summary statistics.

9.2.2 Polygenic scores Our goal so far has been to identify **causal variants** that tell about the biology of the phenotype and propose ways for targeted treatments.

Another way to utilize GWAS results is to **predict** phenotypes. There is a difference between understanding the causes of a phenomenon and an ability to predict the phenomenon: While understanding typically implies a good prediction, a good prediction does not necessarily require understanding. For example, we do not need to know which of the two variants in high LD with each other is a causal one in order to do a good prediction: Either of the variants will do almost equally well when used in a prediction model, because, due to high LD, they carry almost the same information about the genetic differences between individuals.

Let's consider the standard additive model for the phenotype across the whole genome:

$$y_i = \eta_i + \varepsilon_i = \sum_{k=1}^{p} x_{ik}\lambda_k + \varepsilon_i.$$

If we knew the true causal effects λ_k, then we could do the perfect prediction of the genetic component $\eta_i = \sum_k x_{ik}\lambda_k$ for individual i given her/his genotypes. In the population, this perfect genetic prediction would explain the proportion h^2 (=additive heritability) of the total phenotypic variance and this would be as good as an additive genetic prediction ever gets.

By a **polygenic score** (PS) we mean an instance of the additive genetic predictor defined by a set of weights $\boldsymbol{\alpha} = (\alpha_k)_{k=1}^p$ that predicts the genetic component of individual i as

$$\mathrm{PS}_i(\boldsymbol{\alpha}) = \sum_{k=1}^{p} x_{ik}\alpha_k.$$

Typically, the weights α_k are obtained from the GWAS summary statistics $\widehat{\beta}_k$, possibly with some variable selection and/or shrinkage of the effect sizes and/or LD-adjustments to approximate the causal effects λ_k rather than the marginal effects β_k.

Such PS can then be tested against the known phenotype values in a test cohort to see how much phenotypic variation it explains.

A recent guide for making PS by Choi et al. 2020 and the corresponding online tutorial. These articles talk about developing and evaluating polygenic risk prediction models (Chatterjee et al. 2016) and personal and clinical utility of PS (Torkamani et al. 2018).

The methods to derive the best possible PS weights α_k are currently one of the hot topics in the GWAS field since the recent results have shown that the current GWAS summary statistics can already provide useful predictive discrimination for risk of several diseases (slides 17-21). Hope is that by more advanced modeling, the accuracy can be further improved.

New methods are typically compared to the two reference methods: **P-value clumping** and LDpred. Both of these take in the GWAS association statistics and produce PS-weights by accounting for LD. P-value clumping prunes away variants that are in high LD with each other whereas LDpred is a Bayesian method that outputs estimates for the causal effect sizes for each variant by accounting for the LD-structure around the variant.

P-value clumping (r^2,d,$P_{\mathbf{thr}}$). The simplest PS uses the marginal GWAS effect estimates $\widehat{\beta}_k$ as weights. Suppose that we have two variants in high LD. Their marginal effects are almost the same and including them both in the PS is likely to overestimate the joint contribution from these two SNPs and, hence, reduce the PS accuracy. To avoid this, we do some **LD-pruning** meaning that we will only include non-zero weights for SNPs that are not in high LD with each other. For example, we may require that $r^2 < 0.1$ between all pairs of variants that have non-zero weights in PS. In practice, such LD-pruning is applied only within certain window size (e.g., $d = 1$Mb) for computational reasons and because LD decays quickly with distance in homogeneous populations. **P-value clumping** means LD-pruning that prefers to leave in the data the variant with the lowest GWAS P-value and prune away its LD-friends that have higher GWAS P-values. This way our final LD-pruned data set contains as many of the top GWAS variants (in terms of the lowest P-values) as possible given the pruning parameter r^2. Typically, there is also a P-value threshold (between 0.05 and 5e-8) to ensure that all variants included in PS with non-zero weights will have GWAS P-value $< P_{\text{thr}}$.

Training, validation and testing. Let's put the generation of the PS-weights using P-value clumping method to the context of typical prediction model building having three independent data sets for each of training, validation and testing.

- Training data is an existing large GWAS on the phenotype of interest where we have access to the marginal association statistics. Ideally, we would also have access to the LD information of the training data, but when this is not possible, we use external reference data from the GWAS population as an approximation to the LD in training data.
- Validation data are genotype-phenotype data that we use to tune the parameters of the PS model, namely r^2 and P_{thr} (while d is often assumed fixed). This means that we will make a set of PS for a grid of values of r^2 and P_{thr}, and test in the validation set how each of them performs. We choose the best performing version of the PS as our final PS.
- Testing data are individual level genotype-phenotype data that are independent from training and validation data. Testing data are used only to test the final PS that was chosen at the validation step. The performance of the PS in testing data is expected to generalize to other data sets that have similar properties: same population, same phenotype etc. Note that the performance in validation data may overestimate the performance in some new data since the validation data were used for optimizing the PS parameters. The performance in testing data does not suffer from this problem and it is therefore the final result to report about the PS chosen.

PRSice2 is software to generate and validate PS given the summary statistics and validation data. Also PLINK2 does P-value clumping and computes PS.