Notes on: Combining evidence from different GWAS studies for binary traits

Ziang Zhang, Lei Sun

1 Introduction:

For most human traits, genetic effects from specific SNP only have small effect sizes (Evangelou and Ioannidis 2013). Therefore, practitioners often aggregate the GWAS results from different studies through meta-analysis or mega-analysis in order to achieve higher power. Aggregation through p-values is a commonly used example of meta-analysis method. For example when there are several different datasets, practitioners sometimes will first carry out a comprehensive study using one dataset, and only follow up with the SNPs that have the highest significance levels (Begum et al. 2012). Another example would be multi-trait analysis which contains a joint analysis of multiple related traits, in order to boost the statistical power (Turley et al. 2018). In this note, we would like to consider some problems that practitioners may encounter when applying such procedures on the analysis of binary traits.

With examples of Wald test, we presented the phenomenon that the same hypothesis test on different binary traits can have very different distributions of p-value under the alternative hypothesis, hence very different powers, even if the two traits have the same true SNP effect and are analyzed using the same dataset.

In the next section, we will give a brief overview of Wald test for Generalized Linear Models, with a short explanation of the rationale behind the phenomenon we mentioned above. Then, we will follow with two simulation studies to illustrate how this phenomenon is affecting the analysis of binary trait, but not the analysis of continuous trait.

2 Wald test for Generalized Linear Models:

Assume the generalized linear model (glm) has the following form:

$$\mathbb{E}(Y|X) = \mu = g^{-1}(\beta_0 + \beta_1 X_1 + \beta_2 X_2) = g^{-1}(\eta),$$

where $g(.)$ is a specific link function connecting the linear predictor η with the mean function of Y. In this case, the fisher information matrix at β can be written as

$$I_n(\beta) = X^T W(\beta) X,$$

where X denotes the design matrix and $W(\beta)$ is a diagonal matrix with each diagonal term depending on the value of β unless g is identity function. Specifically, the i^{th} diagonal term of W can be computed as

$$w_i = (\frac{\partial u_i}{\partial \eta_i})^2 / \mathrm{Var}(Y_i|X).$$

If the question of interest is to test the hypothesis $H_0 : \beta_2 = 0$ using Wald test, the test statistic can be written as

$$T = \frac{(\hat{\beta}_2)^2}{I_n^{-1}(\hat{\beta})_{[3,3]}},$$

where $I_n^{-1}(\hat{\beta})_{[3,3]}$ denotes the third diagonal term of the matrix $I_n^{-1}(\hat{\beta})$ and $\hat{\beta}$ is the MLE estimator. Under the null hypothesis, T asymptotically follows a Chi-Square distribution with 1 degree of freedom.

Under the alternative hypothesis that $\beta_2 \neq 0$, the non-centrality parameter of Wald test above can be computed as

$$\frac{\beta_2^2}{I_n^{-1}(\beta)_{[3,3]}},$$

where β is the vector of true values for the regression parameters β. Since $I_n^{-1}(\beta)_{[3,3]}$ will not solely depend on β_2 unless g is identity. Therefore the power function of this Wald test will not only depend on β_2, but the whole vector β.

Define $d = -\frac{\beta_2}{\sqrt{I_n^{-1}(\beta)_{[3,3]}}}$, the theoretical power of this Wald test at β can be computed as

$$1 - \Phi(d + z_{a/2}) + \Phi(d - z_{a/2}),$$

where Φ is the CDF of standard normal and $z_{a/2}$ is the $a/2$ quantile of standard normal.

In summary, this means that if we utilize Poisson regression to analyze count traits (e.g. number of cancers) or Logistic/Probit regression to analyze binary traits (e.g. disease status), powers of Wald test from different studies can be dramatically different, even if the two studies have the same effect size (i.e. $|\beta_2|$) and the same set of covariates $\{X_{1i}, X_{2i}\}_{i=1}^n$. The rationale behind this is actually the classical contrast between **statistical significance** measured by p-values and **practical significance** measured by the size of the effect $|\beta_2|$. In Wald test, p-values are determined by both the practical significance $|\beta_2|$ and the standard error $\sqrt{I_n^{-1}(\beta)_{[3,3]}}$. Since the standard errors of the MLE estimator will be different on the two studies, the conclusion drawn from statistical significance may be inconsistent with the practical significance of effects in the two studies.

3 Difference between continuous trait and binary trait:

For the analysis of continuous trait, a natural option would be using ordinary Gaussian linear regression model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon,$$

where $\epsilon \sim N(0, \sigma^2)$.

In this case, suppose the question of interest is testing $\beta_2 = 0$, the Wald test will have non-centrality parameter being

$$\frac{\beta_2^2}{\sigma^2[X^T X]^{-1}_{[33]}}.$$

Because the (inverse) information matrix $\sigma^2[X^T X]^{-1}$ only depends on the nuisance parameter σ, power of this test will not change as β_0 or β_1 change, as long as the nuisance parameter is the same. This implies, if a SNP G has true effect β_G being constant across two traits, then the power of testing $\beta_G = 0$ will be constant across two traits as well (assume same σ^2). Therefore, the aggregation of evidences across traits will be straightforward and smaller p value can be associated with larger SNP effect.

On the other hand, if the target is to combine evidence across several binary traits, the result will no longer be as straightforward. If traits are generated from the following logistic regression model:

$$\mathrm{P}(Y = 1|X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2,$$

In this case, the corresponding non-centrality parameter of testing $\beta_2 = 0$ becomes

$$\frac{\beta_2}{[X^T W_\beta X]^{-1}_{[33]}}.$$

Since $g^{-1}(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)}$ for logistic regression, we can compute the i-th term of the diagonal matrix W_β as

$$[W_\beta]_{ii} = f(\eta_i) = \frac{\exp(-\eta_i)}{(1+\exp(-\eta_i))^2}.$$

The function $f(.)$ represents the density function of standard logistic distribution, and the term η_i is the i-th linear predictor. This implies the non-centrality parameter, hence the power will depend on every parameter in the model, not just on the parameter of interest β_2.

An important consequence of this phenomenon is that, the magnitude of p values (statistical significance) will not reflect the magnitude of the SNP effects (pratical significance), even if the two studies are carried out using one common dataset. For example, if a SNP has effect β_G on both traits, it may show significance on only one trait because of the difference in the covariate effect such as gender or race.

4 Simulation with Gaussian linear regression model:

Assume a dataset of size $n = 1000$ has been collected. The dataset contains the information of a continuous covariate Z, a SNP G, and two continuous traits (A and B). For simplicity, assume this SNP follows hardy weinberg equilibrium (HWE) with minor allele frequency (MAF) 0.3, and the covariate Z has been centered such that $Z \sim N(0, \sigma = 3)$.

Furthermore, we assume that the generating models for each trait are the followings:

$$\mathbf{A} : Y = -0.5 + 0.8Z + 0.3G + \epsilon,$$

$$\mathbf{B} : Y = -0.5 + 0.1Z + 0.3G + \epsilon,$$

where the noise term ϵ follows $N(0, 3)$ in both models.

To test the null hypothesis $H_0 : \beta_G = 0$, a Wald test can be carried out for each trait (with $\alpha = 0.05$ for simplicity). Using the formula from above, we can compute the theoretical power of each trait:

```
### Simulated the common Z and G
set.seed(100,sample.kind = "Rounding")
N <- 1000
G <- sample(c(0,1,2),size = N, replace = T, prob = c(0.49,0.42,0.09))
Z <- rnorm(N,sd = 3)

### Simulate each trait's disease status based on Z and G
## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
yA <- beta0 + betaG*G + betaZ*Z + rnorm(N,sd = 3)

## B:
beta0 <- -0.5
betaZ <- 0.1
betaG <- 0.3
yB <- beta0 + betaG*G + betaZ*Z + rnorm(N, sd = 3)

## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
### Theoretical Power
```

```
mod_A <- lm(yA~Z + G)
#### Get the design matrix:
X <- cbind(rep(1,N),mod_A$model[,-1])
### Compute the weight matrix W:
beta <- c(beta0,betaZ,betaG)
I <- (1/9)* as.matrix(t(X)) %*% as.matrix(X)
#### Invert to get the true covariance matrix
V <- solve(I)
### Compute the power function:
delta <- sqrt(1/V[3,3])*(0-beta[3])
alpha <- 0.05
Power_A <- 1- pnorm(delta - qnorm(alpha/2)) + pnorm(delta + qnorm(alpha/2))
Power_A
```

```
## [1] 0.548751
```

```
## B:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
### Theoretical Power
mod_B <- lm(yB~Z + G)
#### Get the design matrix:
X <- cbind(rep(1,N),mod_B$model[,-1])
### Compute the weight matrix W:
beta <- c(beta0,betaZ,betaG)
I <- (1/9)* as.matrix(t(X)) %*% as.matrix(X)
#### Invert to get the true covariance matrix
V <- solve(I)
### Compute the power function:
delta <- sqrt(1/V[3,3])*(0-beta[3])
alpha <- 0.05
Power_B <- 1- pnorm(delta - qnorm(alpha/2)) + pnorm(delta + qnorm(alpha/2))
Power_B
```

```
## [1] 0.548751
```

Based on the formula from previous section, we can compute the power to be 0.549 in both studies. As we expected, since in both traits the SNP effect is the same, powers should be the same for the two traits as well. To make sure the computed theoretical powers are indeed correct, we can compare them with empirical powers obtained from repeated simulations (K = 2000):

```
set.seed(12345,sample.kind = "Rounding")
## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
p1 <- c()
for (i in 1:2000) {
  yA <- beta0 + betaG*G + betaZ*Z + rnorm(N, sd = 3)
  mod <- lm(yA~Z+G)
  p1[i] <- summary(mod)$coefficient[3,4]
}
emp_power <- mean(p1 <= alpha)
emp_power
```

```
## [1] 0.5505
```

```
## B:
beta0 <- -0.5
betaZ <- 0.1
betaG <- 0.3
p2 <- c()
for (i in 1:2000) {
  yB <- beta0 + betaG*G + betaZ*Z + rnorm(N, sd = 3)
  mod <- lm(yB~Z+G)
  p2[i] <- summary(mod)$coefficient[3,4]
}
emp_power <- mean(p2 <= alpha)
emp_power
```

```
## [1] 0.5395
```

Based on the 2000 resampling results, the empirical powers are basically the same in the two studies (0.55 for trait A and 0.54 for trait B), which is consistent to the result from theoretical powers.

5 Simulation with logistic regression model:

Assume the same setting as before, except now the two traits of interest are both binary instead of continuous. Assume their generating models are the followings:

$$\mathbf{A}: \text{logit}(\text{P}(Y=1|G,Z)) = -0.5 + 0.8Z + 0.3G,$$

$$\mathbf{B}: \text{logit}(\text{P}(Y=1|G,Z)) = -0.5 + 0.1Z + 0.3G.$$

All the regression parameters are the same as in the example of continuous traits. The case-control ratio for each trait is displayed at below:

```
### Simulated the common Z and G
set.seed(100,sample.kind = "Rounding")
N <- 1000
G <- sample(c(0,1,2),size = N, replace = T, prob = c(0.49,0.42,0.09))
Z <- rnorm(N,sd = 3)


### Simulate each trait's disease status based on Z and G
## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
ylat_A <- beta0 + betaG*G + betaZ*Z + rlogis(N)
y_A <- ifelse(ylat_A >=0, 1, 0)

## B:
beta0 <- -0.5
betaZ <- 0.1
betaG <- 0.3
ylat_B <- beta0 + betaG*G + betaZ*Z + rlogis(N)
y_B <- ifelse(ylat_B >=0, 1, 0)
```

```
### Case control counts across traits:
t <- rbind(table(y_A),table(y_B)) %>% as_tibble()
rownames(t) <- c("A","B")
kableExtra::kable(t, caption = "Case Control Counts across traits") %>%
  kable_styling(latex_options = "HOLD_position", font_size = 10)
```

Table 1: Case Control Counts across traits

	0	1
A	522	478
B	573	427

```
### Case control ratio across genotypes:
t <- cbind(c(y_A,y_B),c(G,G)) %>% as_tibble()
colnames(t) <- c("Y","G")
t <- t %>% group_by(G) %>% summarise(ratio = sum(Y)/n())
kableExtra::kable(t, caption = "Case Control Ratio across genotypes") %>%
  kable_styling(latex_options = "HOLD_position", font_size = 10)
```

Table 2: Case Control Ratio across genotypes

G	ratio
0	0.4238901
1	0.4683841
2	0.5200000

We can use Wald test to test the hypothesis $\beta_G = 0$ (i.e. G is a casual SNP) for each trait:

```
## A:
mod_A <- glm(y_A~Z + G, family = binomial(link = "logit"))
kableExtra::kable(summary(mod_A)$coefficients, caption = "Fitted Model for Trait A") %>%
  kable_styling(latex_options = "HOLD_position", font_size = 10)
```

Table 3: Fitted Model for Trait A

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.4645384	0.1250578	-3.714589	0.0002035
Z	0.8137269	0.0508021	16.017584	0.0000000
G	0.3242170	0.1334258	2.429942	0.0151012

```
## B:
mod_B <- glm(y_B~ Z + G, family = binomial(link = "logit"))
kableExtra::kable(summary(mod_B)$coefficients, caption = "Fitted Model for Trait B") %>%
  kable_styling(latex_options = "HOLD_position", font_size = 10)
```

Table 4: Fitted Model for Trait B

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.4625221	0.0898641	-5.146908	0.0000003
Z	0.0538062	0.0210894	2.551332	0.0107312
G	0.2547557	0.0975127	2.612539	0.0089872

Note that the p-values are 0.015 for trait A, and 0.009 for trait B. It is typically expected that for the trait with smaller p-value, the magnitude of the association (i.e. $|\beta_G|$) should be larger. However, in this simulation example the true value of β_G is $\beta_G = 0.3$ for both traits, and even the covariates are exactly the same.

Again, assume that the hypothesis $\beta_G = 0$ will be tested using Wald test with $\alpha = 0.05$, then we can compute the theoretical powers of the two Wald test using the simulated data $\{G_i, Z_i\}_n$ and the true parameters vectors:

```
## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
### Theoretical Power
mod_A <- glm(y_A~Z + G, family = binomial(link = "logit"))
#### Get the design matrix:
X <- cbind(rep(1,N),mod_A$model[,-1])
### Compute the weight matrix W:
beta <- c(beta0,betaZ,betaG)
#beta <- as.numeric(mod_A$coefficients)
w <- c()
for (i in 1:N) {
  si <- as.numeric(as.numeric(X[i,]) %*% beta)
  w[i] <- dlogis(si)
}
I <- as.matrix(t(X)) %*% diag(w,nrow = N,ncol = N) %*% as.matrix(X)
#### Invert to get the true covariance matrix
V <- solve(I)
### Compute the power function:
delta <- sqrt(1/V[3,3])*(0-beta[3])
alpha <- 0.05
Power_A <- 1- pnorm(delta - qnorm(alpha/2)) + pnorm(delta + qnorm(alpha/2))
Power_A
```

```
## [1] 0.6193771
```

```
## B:
beta0 <- -0.5
betaZ <- 0.1
betaG <- 0.3
### Theoretical Power
mod_B <- glm(y_B~ Z + G, family = binomial(link = "logit"))
#### Get the design matrix:
X <- cbind(rep(1,N),mod_B$model[,-1])
### Compute the weight matrix W:
beta <- c(beta0,betaZ,betaG)
#beta <- as.numeric(mod_B$coefficients)
w <- c()
for (i in 1:N) {
```

```
  si <- as.numeric(as.numeric(X[i,]) %*% beta)
  w[i] <- dlogis(si)
}
I <- as.matrix(t(X)) %*% diag(w,nrow = N,ncol = N) %*% as.matrix(X)
#### Invert to get the true covariance matrix
V <- solve(I)
### Compute the power function:
delta <- sqrt(1/V[3,3])*(0-beta[3])
alpha <- 0.05
Power_B <- 1- pnorm(delta - qnorm(alpha/2)) + pnorm(delta + qnorm(alpha/2))
Power_B
```

```
## [1] 0.8618099
```

Based on the results above, we know in this simulation study, the power of Wald test will be 0.619 for the trait A, and 0.861 for the trait B. Note that Wald test on trait B has quite larger power compared to on trait A, despite the fact that the two samples are generated with same $\beta_G = 0.3$ and generated by the same set of $\{G_i, Z_i\}_n$. This suggests the p-values of Wald test may have very different distributions on the two traits. We can double check that our theoretical powers for both tests are correct using empirical powers:

To compute the empirical powers, we re-simulated each type of binary trait for $K = 2000$ times, and compute the 2000 p-values in each trait:

```
set.seed(100,sample.kind = "Rounding")
## A:
beta0 <- -0.5
betaZ <- 0.8
betaG <- 0.3
p1 <- c()
for (i in 1:2000) {
  ylat_A <- beta0 + betaG*G + betaZ*Z + rlogis(N)
  y_A_rep <- ifelse(ylat_A >=0, 1, 0)
  mod <- glm(y_A_rep~Z+G, family = binomial(link = "logit"))
  p1[i] <- summary(mod)$coefficient[3,4]
}
emp_power <- mean(p1 <= alpha)
emp_power
```

```
## [1] 0.6115
```

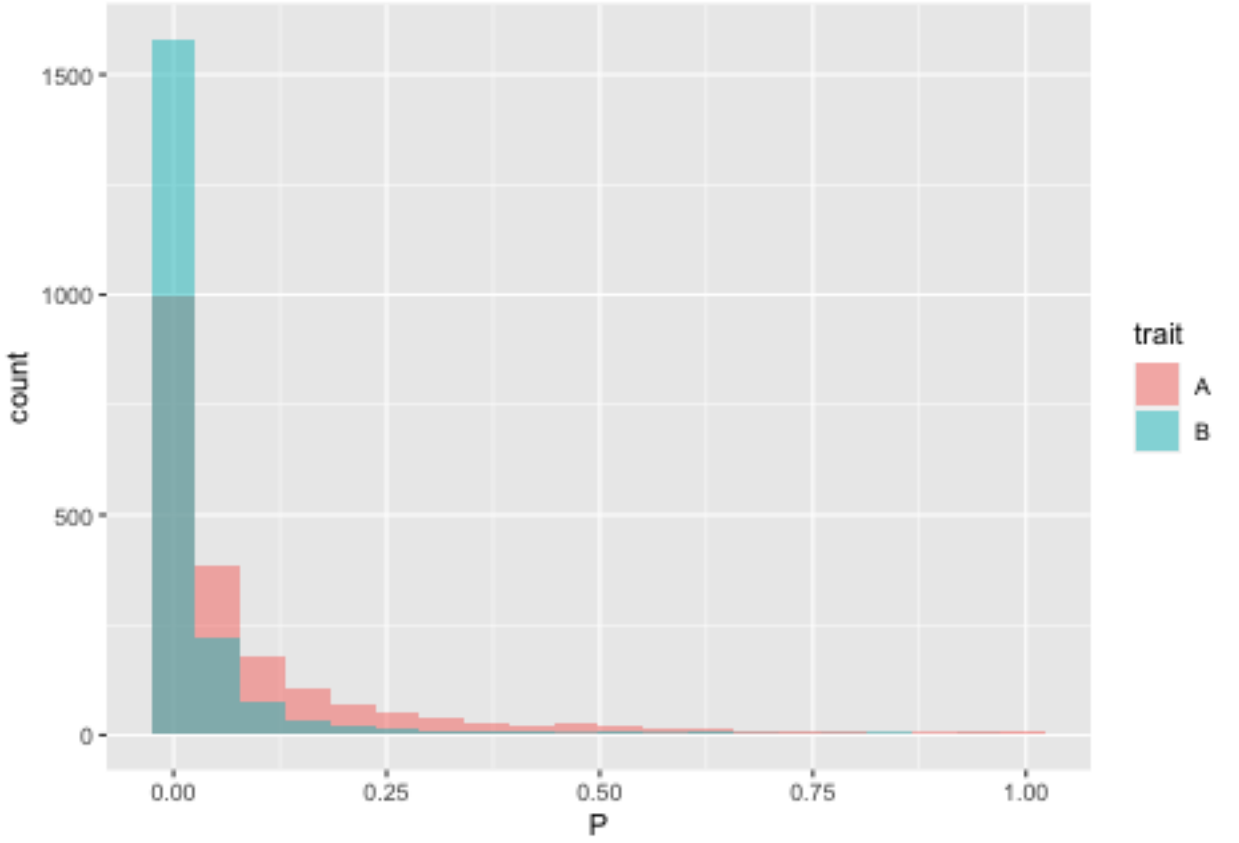
```
## B:
beta0 <- -0.5
betaZ <- 0.1
betaG <- 0.3
p2 <- c()
for (i in 1:2000) {
  ylat_B <- beta0 + betaG*G + betaZ*Z + rlogis(N)
  y_B_rep <- ifelse(ylat_B >=0, 1, 0)
  mod <- glm(y_B_rep~Z+G, family = binomial(link = "logit"))
  p2[i] <- summary(mod)$coefficient[3,4]
}
emp_power <- mean(p2 <= alpha)
emp_power
```

```
## [1] 0.8635
```

Based on the 2000 resampling results, the empirical powers are respectively 0.612 for trait A and 0.864 for

trait B. These values are quite close to the theoretical values 0.619 and 0.861 we computed above. The distributions of p-values in each trait can be visualized as well:

```
### Comparison:
pcomp <- tibble(P = c(p1,p2), trait = c(rep("A",2000),rep("B",2000)))
pcomp %>% ggplot(aes(x = P, fill = trait)) + geom_histogram(bins = 20, alpha=0.5, position="identity")
```



Based on the figure above, we can conclude that the distribution of p values in trait B is stochastically smaller than the distribution in trait A, even if their underlying β_G are both 0.3. Therefore, it shows that the magnitudes of p-values of different studies are not directly comparable, unless the generalized linear regression model being used is the ordinary linear regression model with g being identity function.

6 Bibliography

Begum, Ferdouse, Debashis Ghosh, George C. Tseng, and Eleanor Feingold. 2012. "Comprehensive literature review and statistical considerations for GWAS meta-analysis." *Nucleic Acids Research* 40 (9): 3777–84.

Evangelou, Evangelos, and John Ioannidis. 2013. "Evangelou e, Ioannidis JP.meta-Analysis Methods for Genome-Wide Association Studies and Beyond. Nat Rev Genet 14:379-389." *Nature Reviews. Genetics* 14 (May).

Turley, Patrick, Raymond K Walters, Omeed Maghzian, Aysu Okbay, James J Lee, Mark Alan Fontana, Tuan Anh Nguyen-Viet, et al. 2018. "Multi-Trait Analysis of Genome-Wide Association Summary Statistics Using MTAG." *Nature Genetics* 50 (2): 229–37.