SVA Simulation

Christopher Lo

9/6/2022

Simulation Set-Up

Simulation studies inspired from *"A general framework for multiple testing dependence"* (Leek et al. 2008)

We generate X from the following model:

$$X = BS + \Gamma G + U$$

We have $m = 1000$ genes (tests), $n = 20$ samples, and $r = 2$ latent variables.

Sampling noise: $U_{m,n} \sim N(0, 1)$.

The design matrix S is 10 cases and 10 controls: $S_{1,n} = 1$ for $n = 1 : 20$. Then, $S_{2,n} = 0$ for $n = 1 : 10$, $S_{2,n} = 1$ for $n = 11 : 20$.

Control effect for all genes: $b_{m,1} \sim N(0, 1), m = 1 : 1000$

Case effect for DE genes $m = 1 : 300$: $b_{m,2} \sim N(3, 1)$

Case effect for Non-DE genes $m = 301 : 1000$: $b_{m,2} \sim N(0, 2)$

Latent design matrix (kernel) G: $G_{r,n} \sim Bernoulli(.2), n = 1 : 10$. $G_{r,n} \sim Bernoulli(.7), n = 11 : 20$, where $r = 1, 2$. (This ensures correlation between the two design matrices.)

Latent effect 1: $\Gamma_{m,1} \sim N(0, 1), m = 1 : 300$, $\Gamma_{m,1} \sim N(1, 2.5), m = 301 : 1000$. (Overlaps with Non-DE genes, will lead to FPs if not corrected)

Latent effect 2: $\Gamma_{m,2} \sim N(-1, 2.5), m = 1 : 300$, $\Gamma_{m,2} \sim N(0, 1), m = 301 : 1000$. (Overlaps with Non-DE genes, will lead to FNs if not corrected)

Run SVA and regression to estimate parameters and SVs

We look at the number of SVs estimated, whether the latent variables are spanned by the estimated SVs, the recovered regression coefficients, the null p-value distribution, and the ranks of top genes.

Todo: ranks of top genes code unsure right now.

```
n.sv = num.sv(X, t(S), method = "be")
cat("Method be: Number of SVs: ", n.sv, "\n")
```

```
## Method be: Number of SVs:  2
```

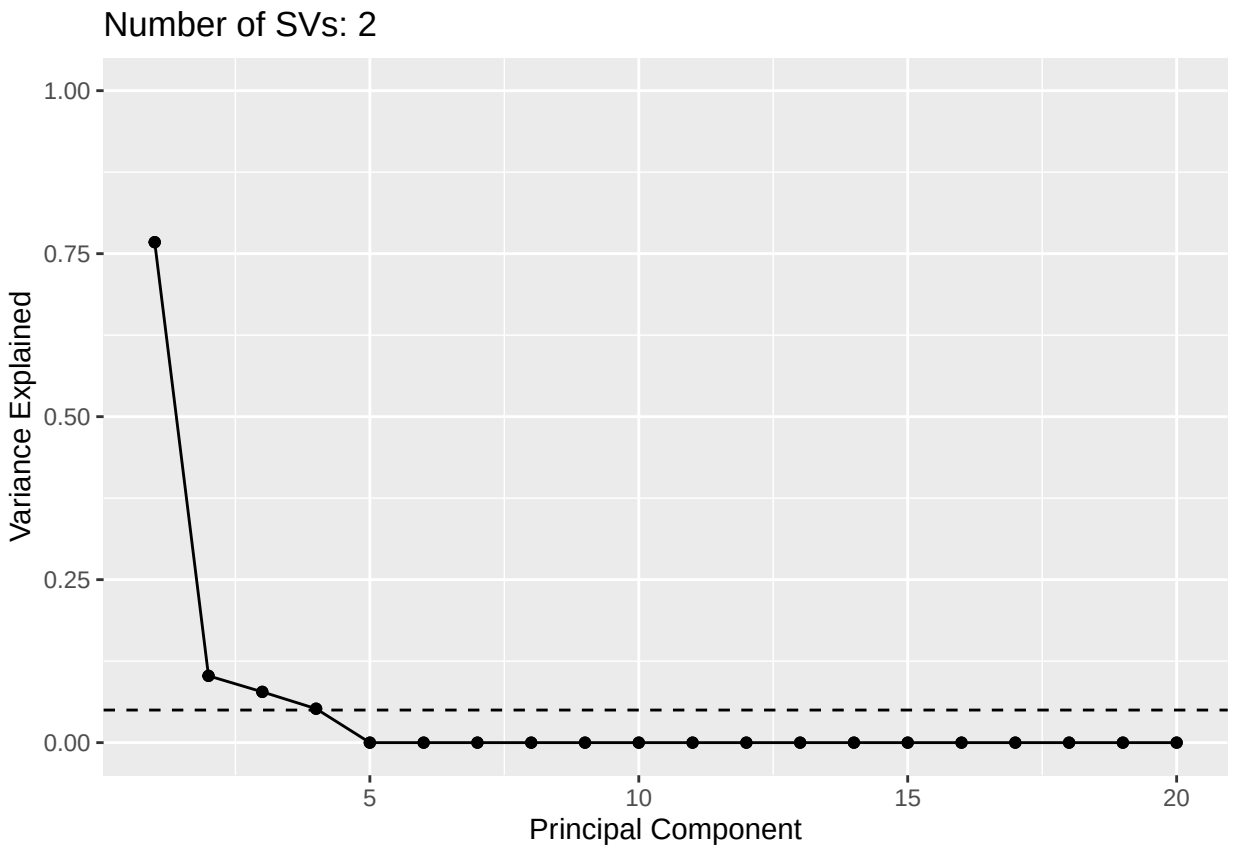
```
cat("Correlation of primary vs. latent variable 1: ", cor(G[1 ,], S[2 ,]), "\n")
```

```
## Correlation of primary vs. latent variable 1:  0.4082483
```

```
cat("Correlation of primary vs. latent variable 1: ", cor(G[2 ,], S[2 ,]), "\n")
```

```
## Correlation of primary vs. latent variable 1:  0.1048285
```

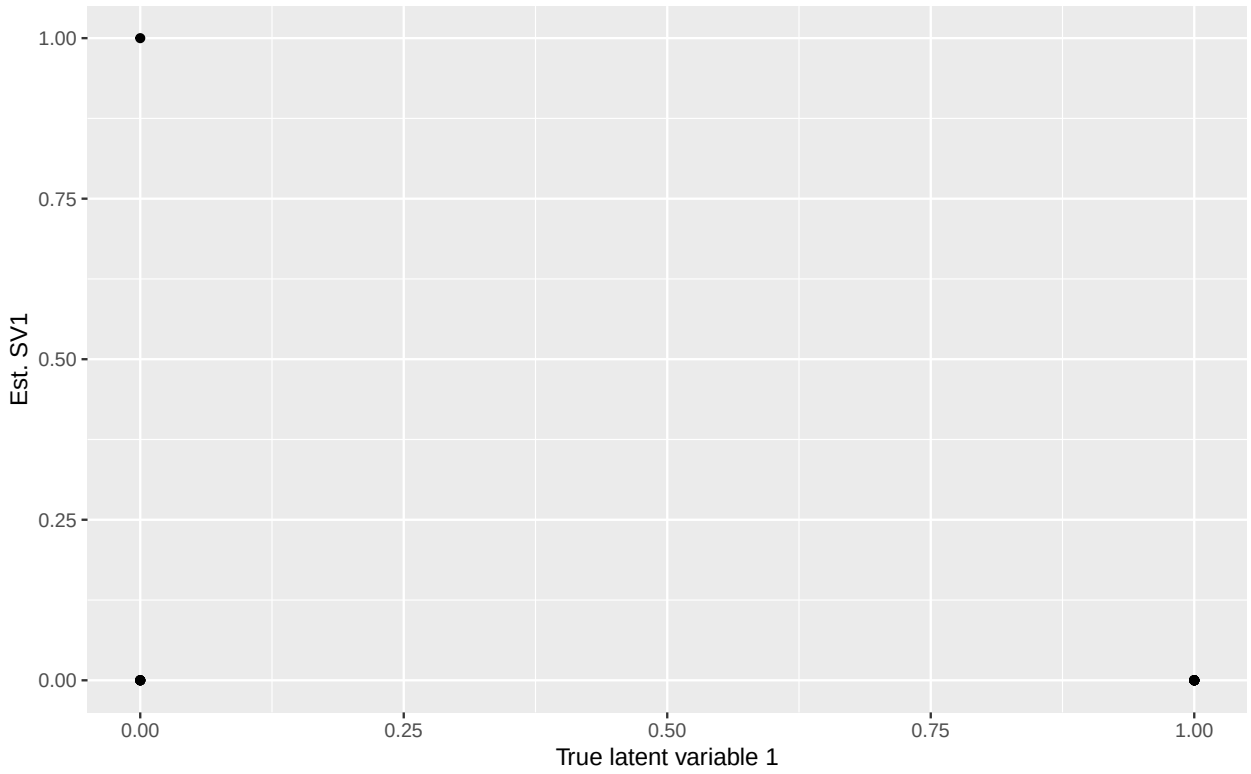
```
pca = prcomp(X)
variance = pca$sdev^2 / sum(pca$sdev^2)
qplot(c(1:length(variance)), variance) + geom_line() + geom_point() +
  geom_hline(yintercept=1/ncol(X), linetype = "dashed") +
  xlab("Principal Component") + ylab("Variance Explained") + ggtitle(paste0("Number of SVs: ", n.sv)) +
```



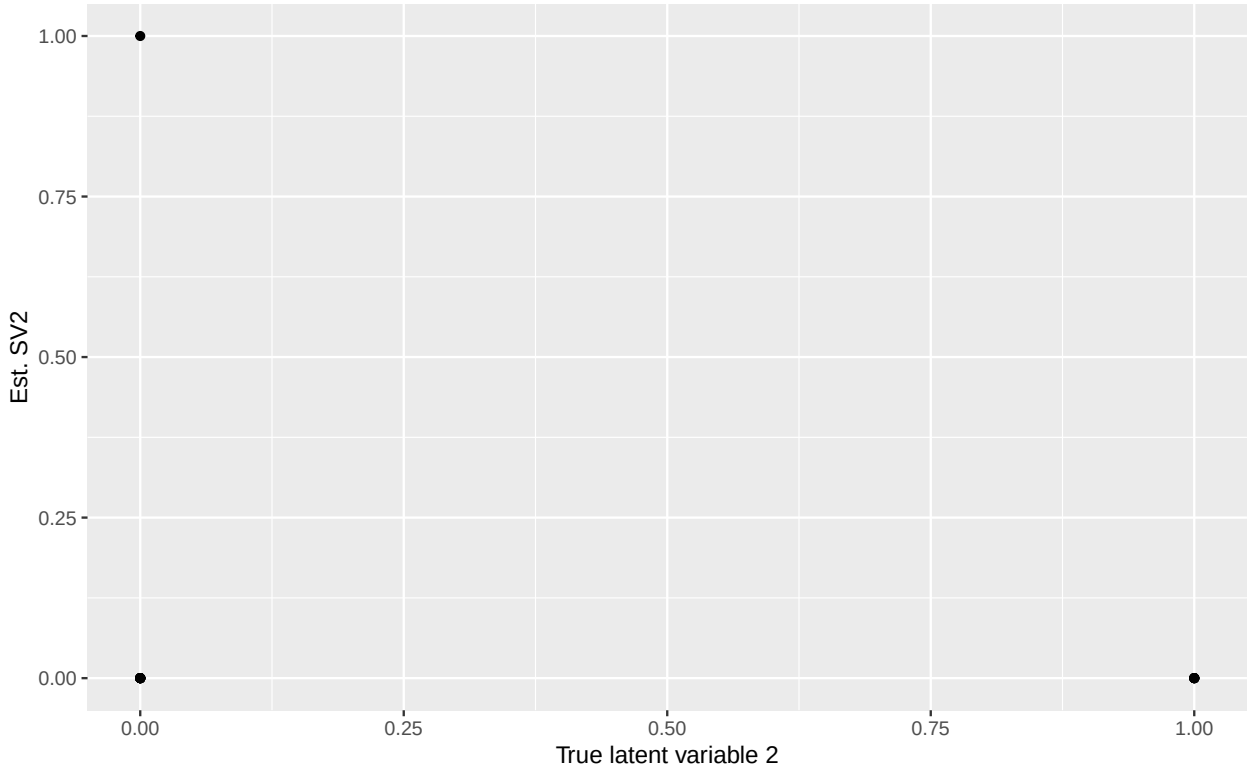

```
nullMod = t(S)[, 1]
svobj = sva(X, t(S), nullMod, n.sv = n.sv)
```

```
## Number of significant surrogate variables is:  2
## Iteration (out of 5 ):1  2  3  4  5
```

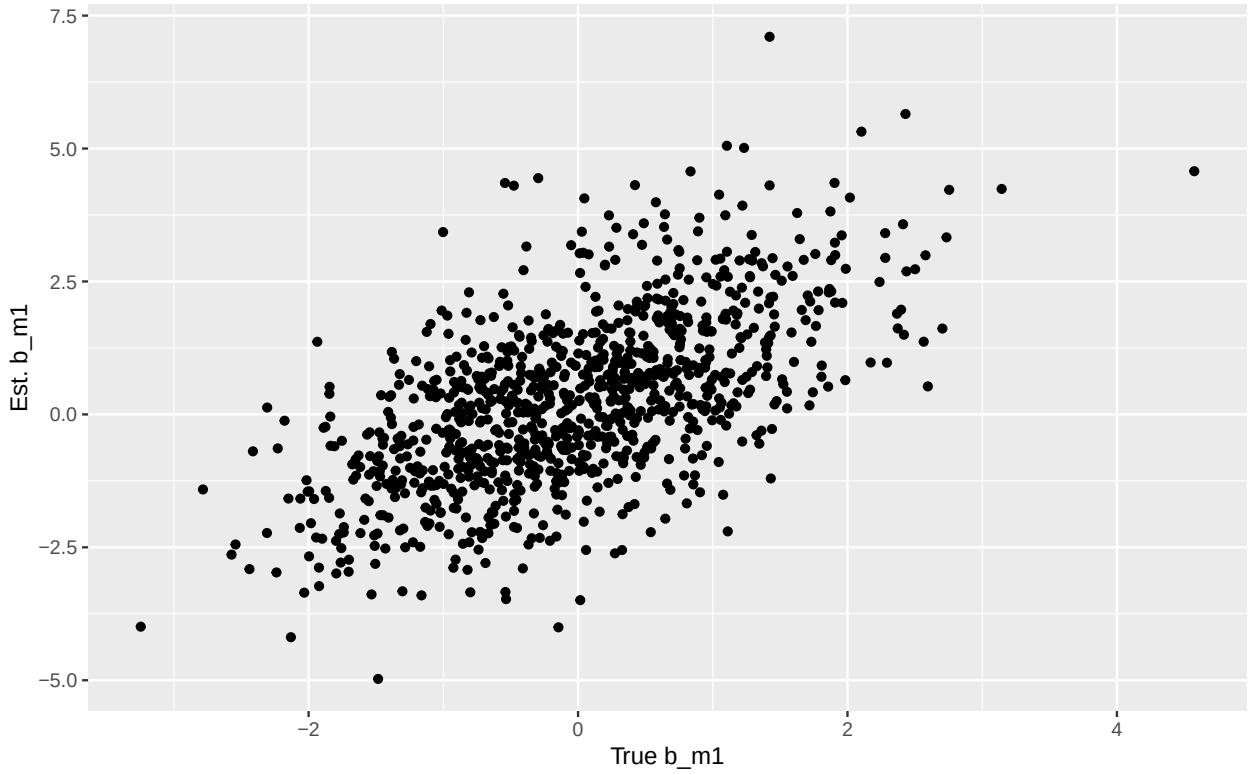
```
#visually look at predicted SVs.
qplot(as.numeric(G[1 ,]), svobj$sv[, 1], xlab = "True latent variable 1", ylab = "Est. SV1")
```



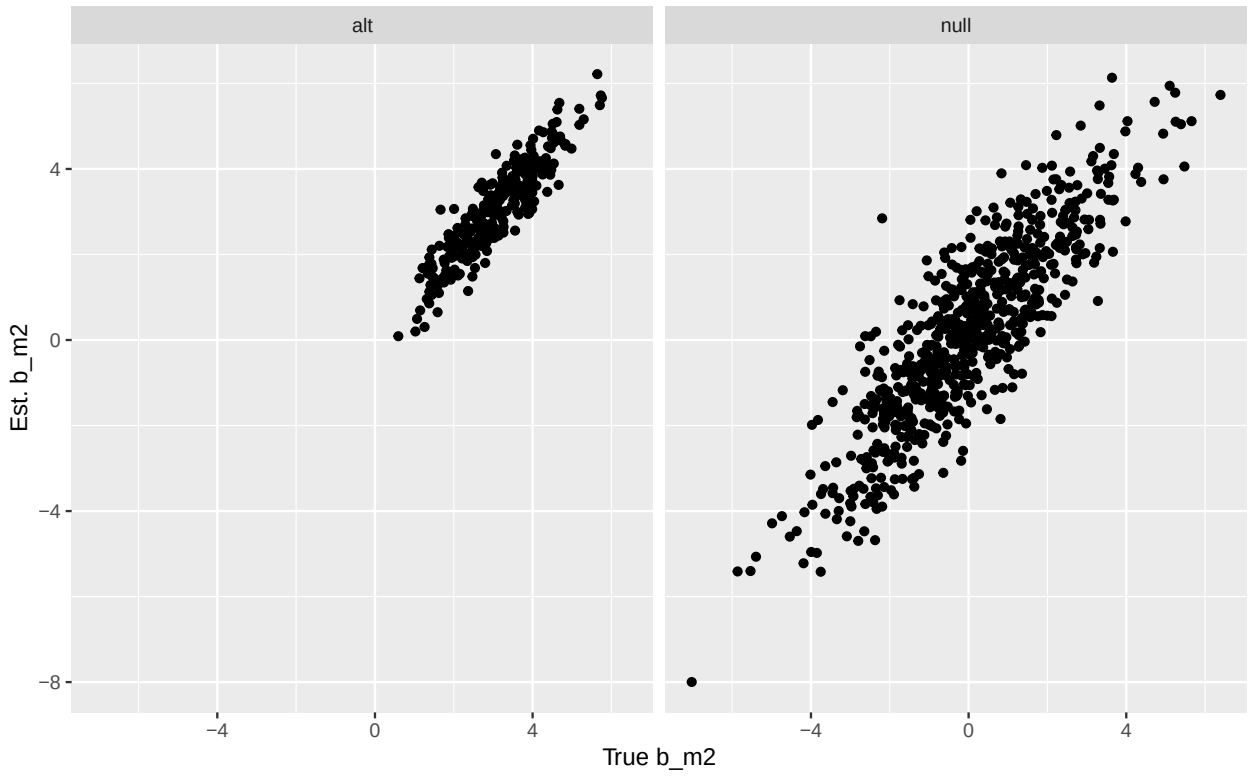

```
qplot(as.numeric(G[2 ,]), svobj$sv[, 2], xlab = "True latent variable 2", ylab = "Est. SV2")
```



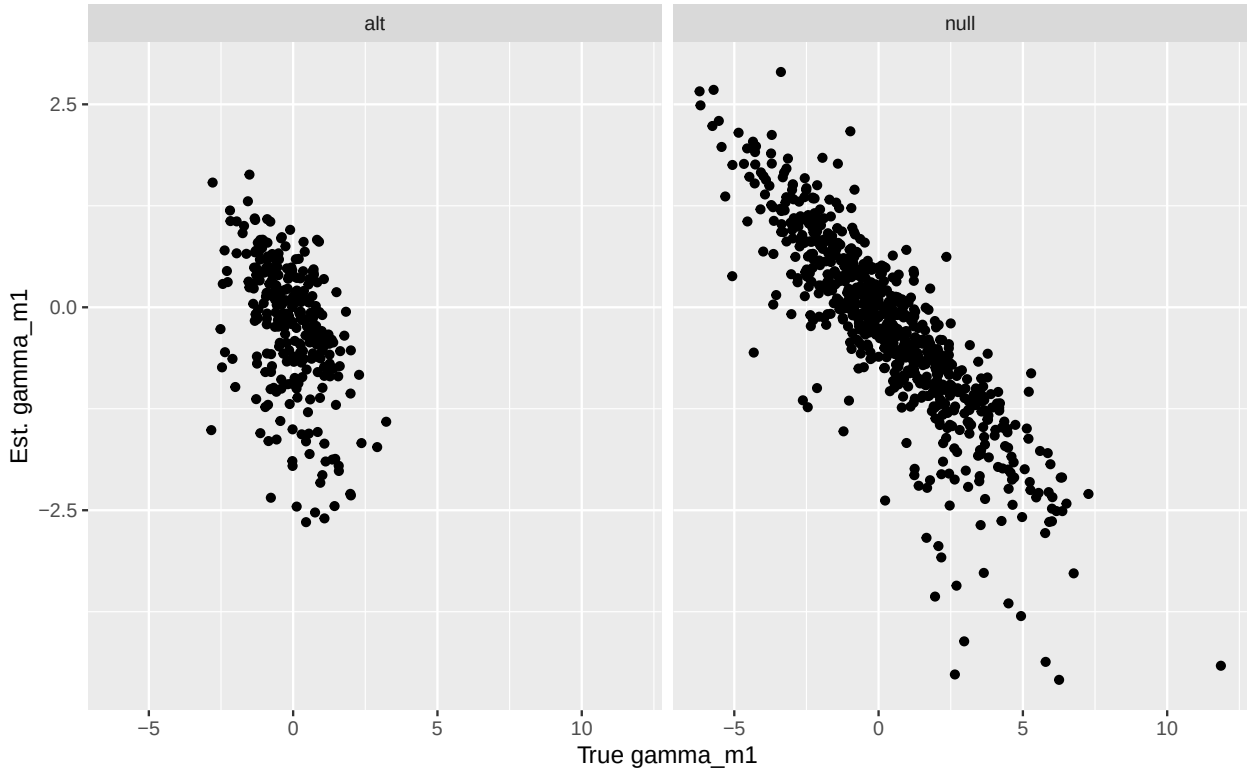
```
nullmodsv = cbind(nullMod, svobj$sv)
modsv = cbind(t(S), svobj$sv)
#run full regression.
fitsv = lm.fit(modsv, t(X))

#visually look at predicted coefficients
plot_df = data.frame(b1 = B[, 1],
                     b2 = B[, 2],
                     b1_hat = fitsv$coefficients[1 ,],
                     b2_hat = fitsv$coefficients[2 ,],
                     b2_labels = c(rep("alt", 300), rep("null", m - 300)),
                     gamma1 = Gamma[, 1],
                     gamma1_hat = fitsv$coefficients[3 ,],
                     gamma1_labels = c(rep("alt", 300), rep("null", m - 300)),
                     gamma2 = Gamma[, 2],
                     gamma2_hat = fitsv$coefficients[4 ,],
                     gamma2_labels = c(rep("null", 200), rep("alt", 200), rep("null", 600)))

ggplot(plot_df, aes(b1, b1_hat)) + geom_point() + labs(x = "True b_m1", y = "Est. b_m1")
```



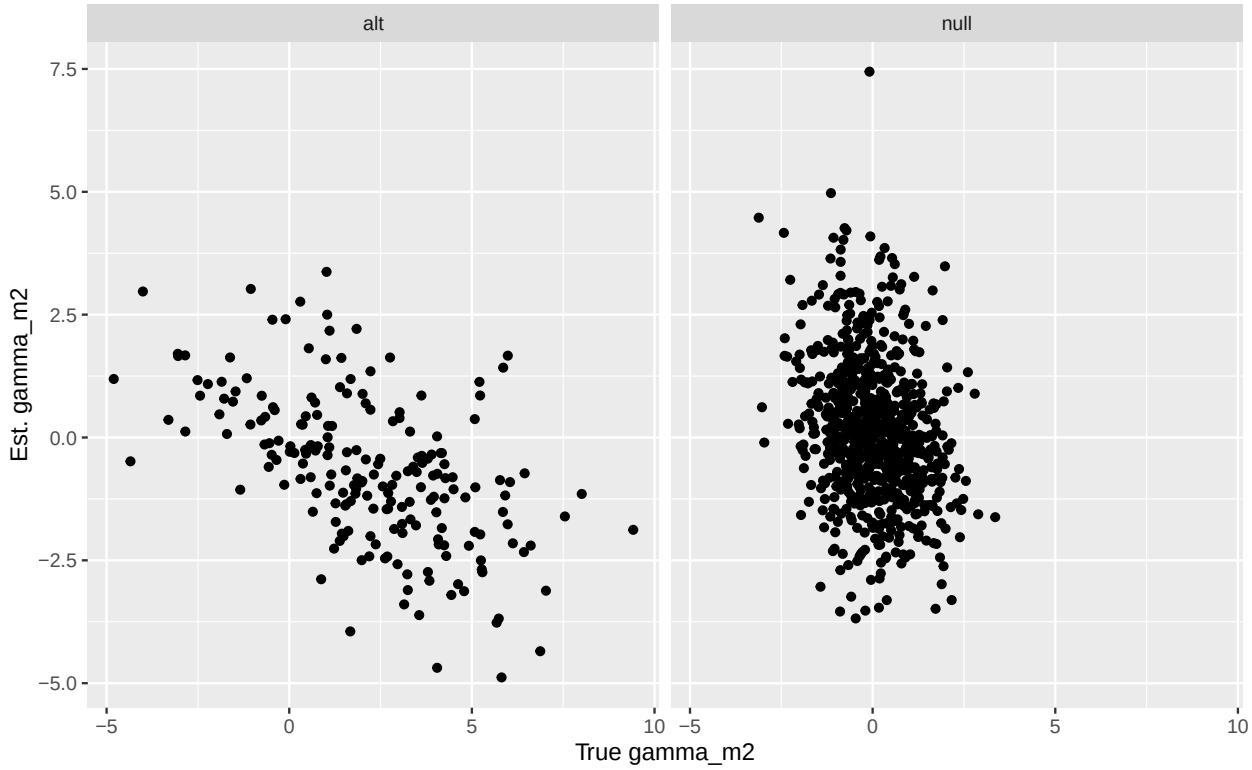

```
ggplot(plot_df, aes(b2, b2_hat)) + geom_point() + facet_wrap(~b2_labels) + labs(x = "True b_m2", y = "Es
```



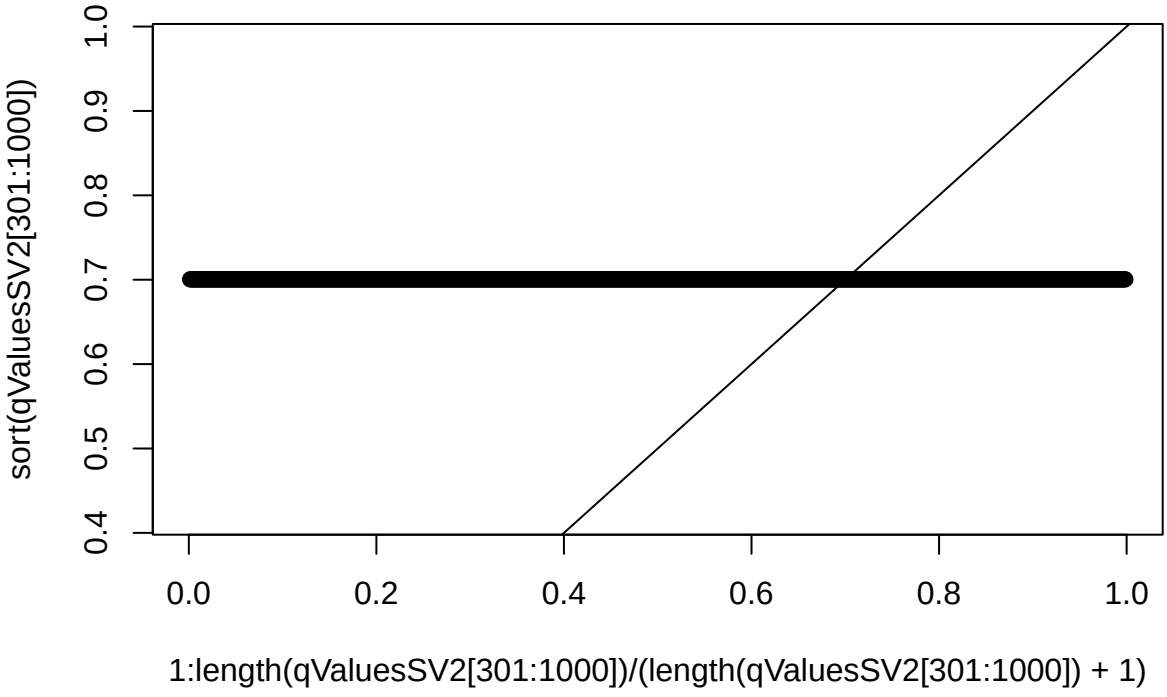

```
ggplot(plot_df, aes(gamma1, gamma1_hat)) + geom_point() + facet_wrap(~gamma1_labels) + labs(x = "True g
```



```
ggplot(plot_df, aes(gamma2, gamma2_hat)) + geom_point() + facet_wrap(~gamma2_labels) + labs(x = "True g
```



Not sure what's going on here yet regarding p-values and ranking.

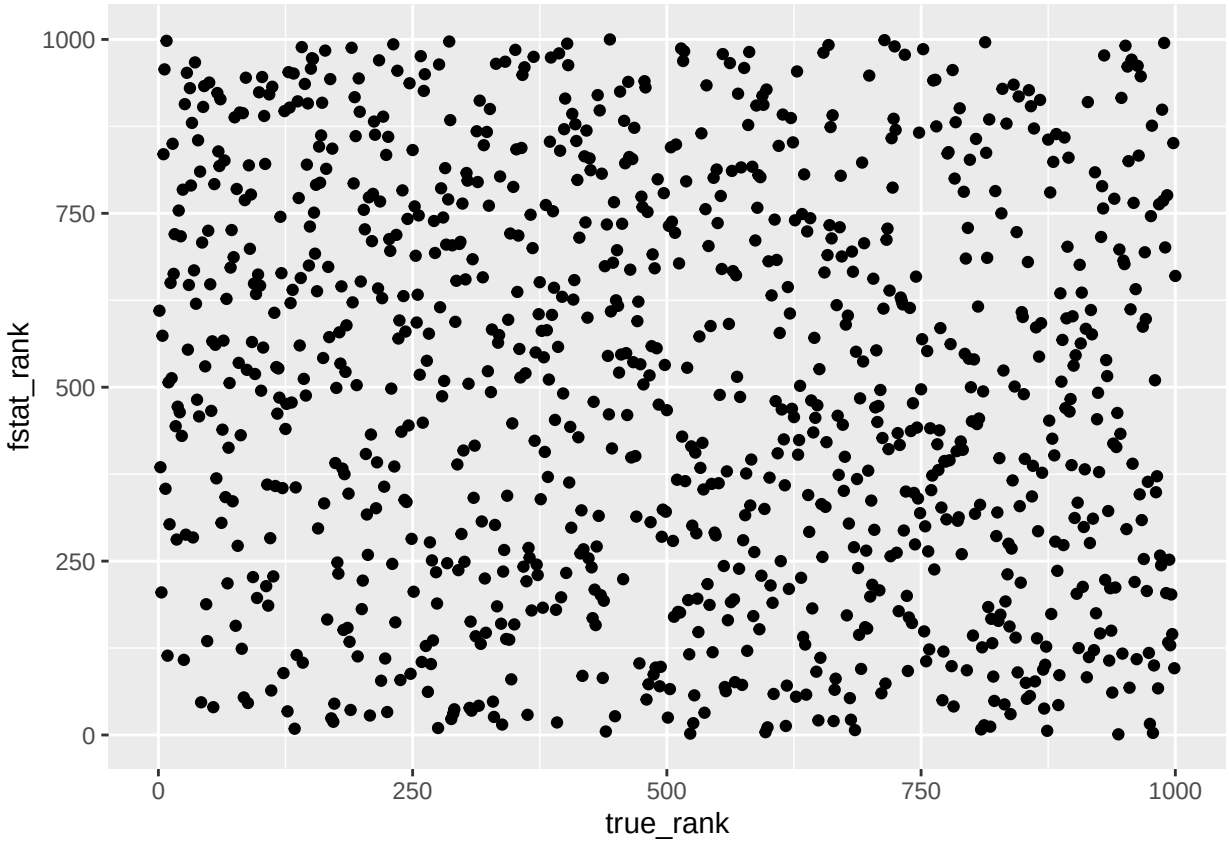



```
## Warning in ks.test(qValuesSV2[301:1000], "punif", 0, 1): ties should not be
## present for the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  qValuesSV2[301:1000]
## D = 0.70047, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
##
## FALSE
##   700
```

```
##
## FALSE
##   300
```



"Knobs to turn" in this experiment:

$\Gamma_{m,1}$: If strong effect relative to $b_{m,1}$, then this will generate noise on control samples.

$\Gamma_{m,2}$: If strong effect relative to $b_{m,2}$, then this will generate noise on case samples.

Our certainty of Γ to effect case or control samples depends on "the percentage of row space of S explained by G". We appropx that by looking at $cor(G_r, S_2), r = 1, 2$.

Speculating...

$\Gamma_{m,1}$	$\Gamma_{m,1}$	$cor(G_r, S_2)$	DE	Scree plot
strong	weak	strong	more FPs	more even PCA
weak	strong	strong	more FNs	more even PCA
weak	weak	strong	neutral	more dominated PCA
strong	strong	strong	more FPs and FNs	more even PCA