RiVIERA-beta: Risk Variants Inference using Epigenomic Reference Annotations

Yue Li
liyue@mit.edu

July 9, 2016

1 What is RiVIERA

RiVIERA is a novel Bayesian framework is able to jointly model GWAS summary statistics in terms of p-values in multiple traits using large-scale reference datasets. It then outputs (a) posterior probability of association (PPA) c_{vd} that variant v is causal in disease d; (b) the Bayesian fold-enrichment estimates f_{kd} based on the ratio between the prior model with annotation k over the null model without annotation k. This vigentte demonstrates how to use RiVIERA to (1) fine-map risk variants in a given trait; (2) test for fold-enrichment for each annotations.

```
> library(RiVIERAbeta)
> set.seed(100)
```

2 Data preparation

To run RiVIERA, we will need three types of data: (1) GWAS summary statistics in terms of p-values of V variants for at least one single trait; (2) $V \times K$ annotaitons for V variants and K annotations; (3) the size of the loci. We assume the SNPs in both GWAS and annotation input matrices are ordered based on the the their genomic locations such that knowing the start and end of each SNP indices for each locus is sufficient to retrieve both information for the SNPs within that locus.

For ease of demonstration, we load the simulated data generated based on 1000 Genome Data (Phase 1 release 3) from 397 CEU individuals and 100 epigenomic datasets from Roadmap. For each locus, there is 1 causal variant. The summary statistics were generated by ordinary least square regression on the continuous phenotype [1]. For multi-trait causal inference, we simply add columns (one per trait) to the GWAS p-value matrix (not shown here).

```
> load(system.file("extdata/sim_seed15.RData", package="RiVIERAbeta"))
> gwasPval <- simdata$gwasPval
> annotMat <- simdata$ann
> locus_index <- make_locus_index(locus_cnt = simdata$blocksize)
> locus_cnt <- simdata$blocksize
> causal_snpid <- simdata$causal_snpid
```

3 Running RiVIERA

We now train RiVIERA on the simulated data. Here we specify the HMC step size and step number to be 1e-3 and 100 per MCMC iteration, respectively. Also, we specify the burnin to be 20%, meaningful we wwill discard 20% of the initial sampled models to estimate PPA and fold-enrichments.

```
> nsteps <- 100
> max_epoch <- 1e3
> step <- 1e-3
> burnFrac <- 0.2
> set.seed(100)
> ensemble_fit <- rivieraBeta(gwasPval=gwasPval,
+
+                          annMat=annotMat,
+
+                          positiveAnnotConstraint=TRUE,
+
+                          locus_index=locus_index,
+
+                          max_epoch=max_epoch,
+
+                          step=step, nsteps=nsteps,
+
+                          printfreq=10, verbose=FALSE)

Settings:
max_epoch = 1000
mu0 = 0.1; phi0 = 2
causalfrac =0.01; useAnn =1
step = 0.001; nsteps = 100
printfreq = 10; verbose = 0
Starting MCMC inference ...
MCMC inference completed.

> burnin <- round(burnFrac * slot(ensemble_fit, "fit_info")$ensembleSize)
```

4 Causal inference

To finemap causal variants, we infer their PPA as well as the 95% credile set, which is the minimal number of SNPs that add up to 0.95 of PPA in each locus:

```
> riviera_ppa <- finemap(ensemble=ensemble_fit,
+                        gwasPval=gwasPval,
+                        annMat = annotMat,
```

```
+                          locus_index=locus_index,
+                          burnin=burnin)

Total accepted models: 858
Discarded burnin models: 172
Final accepted models: 686

> block_idx <- unlist(sapply(1:length(locus_cnt), function(b) rep(b, locus_
> credset <- getCredibleSet(riviera_ppa, block_idx, thres = 0.95, indexOnly
```

5 Fold enrichment analysis

To perform fold-enrichments, we issue the following command, which will test for enrichment of each annotation based on the likelihood ratio between the full model and the model without the annotation k:

```
> enrich_obj <- enrichTest(ensemble_fit,
+                          annotMat, burnin,
+                          riviera_ppa,
+                          block_idx,
+                          cred_thres=0.95, cred_qt=0.95, verbose=FALSE)

Total accepted models: 858
Discarded burnin models: 172
Final accepted models: 686

> foldenrich_mu <- enrich_obj$w_mu
> foldenrich_lo <- enrich_obj$w_lo
> foldenrich_hi <- enrich_obj$w_hi
> foldenrich_ci <- enrich_obj$w_ci
```

The four outputs are the mean, 95% lower bound, 95% upper bound, and indicator for whether the corresponding annotation is signfiicant, respectively.

6 Power analysis

Because we simulated the data, where the causl variants are known, we can perform power analysis on the detecting the causal variants comparing RiVIERA with the input p-values as the baseline model. To not clutter the vignette, we do not show the R code that generate the plots but interested users can refer to the source code.

As we can see, RiVIERA conferred much higher causal detection power compared to GWAS p-values (Fig. 1). The superior performance is attributable to its ability to takes into account both the genetic signals from GWAS and the effects of the annotations that it identified the causal annotations. In particular, we can examine the quality of the inferred fold-enrichments by them with the ground-truth of the simulated annotation effects as in Fig. 2.

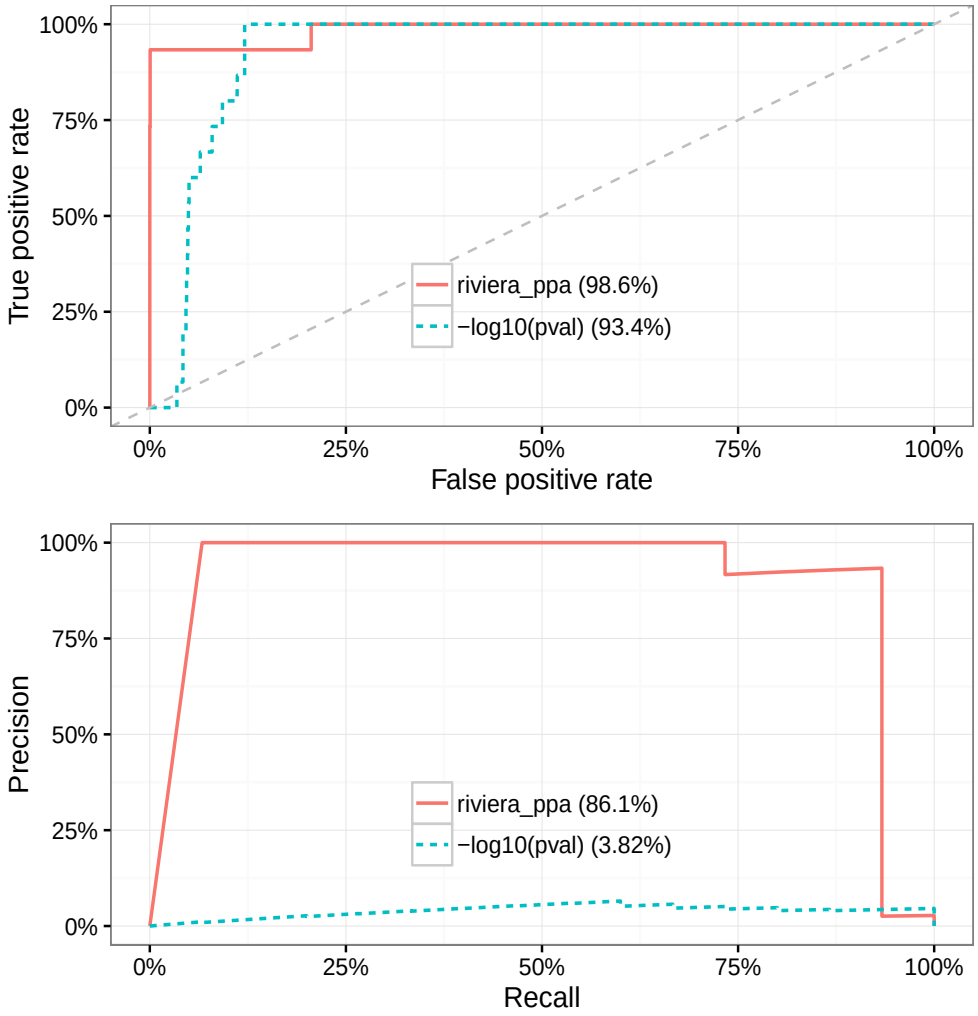

Figure 1: Power analysis on detecting causal variants from simulated case-control GWAS data. Results on a single simulation run shown as receiver operating characteristic (ROC) and precision-recall (PRC). Here we evaluated RiVIERA using posterior probability of association to prioritize variants. As a comparison, we applied p-values to prioritize SNPs as well. Figure insets indicate the areas under the curves (AUC) of ROC and PRC.

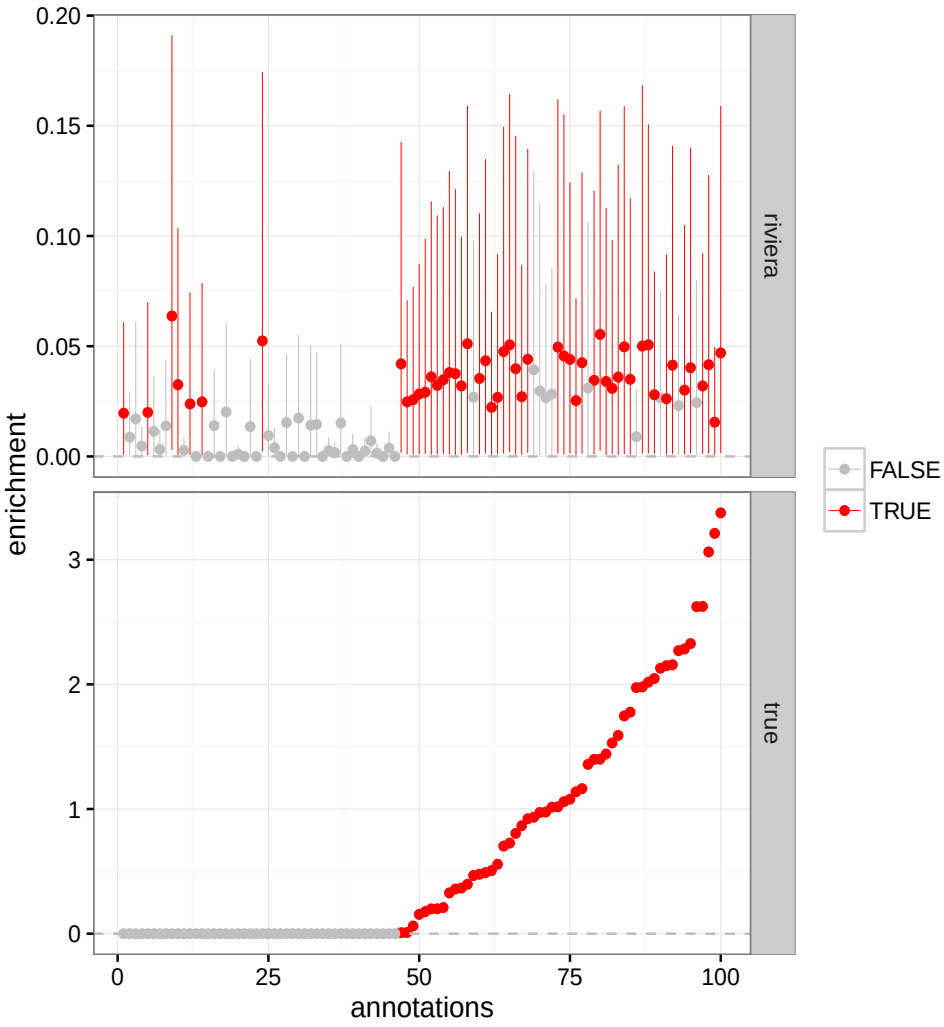

Figure 2: Barplot show the enrichment score for each of the 100 annotations. The top panel shows the Bayesian estimates from RiVIERA centered at 95% credible interval. The bottom panel indicates the true functional influence of each epigenomic annotations.

To gain further intuition of the RiVIERA's superior performance on the simulated data, we can visualize the GWAS p-values and the inferred PPA side-by-side as illustrated for 4 of the 15 loci in Fig. 4. Fig. **??** illustrates the underlying annotations that gave rise to the causal variants (red vertical bar).

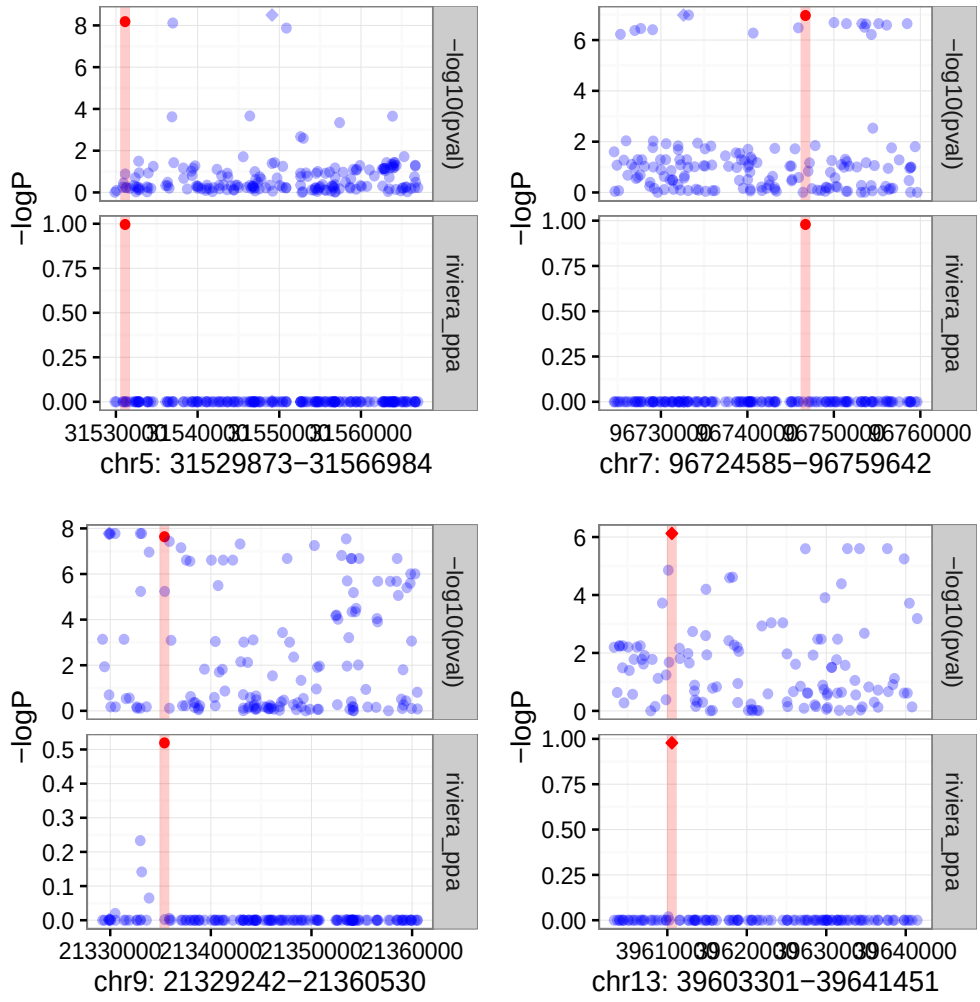

Figure 3: Manhattan plot for GWAS p-values and the predicted PPA. Diamond and red dot indicating the lead SNP and the underlying causal variant, respectively. Other prominent variants (including the lead SNP if it is not the causal one) are due to linkage-disequilibrium to the causal variant.

7 Session Info

```
> sessionInfo()

R version 3.3.1 (2016-06-21)
Platform: x86_64-apple-darwin13.4.0 (64-bit)
```

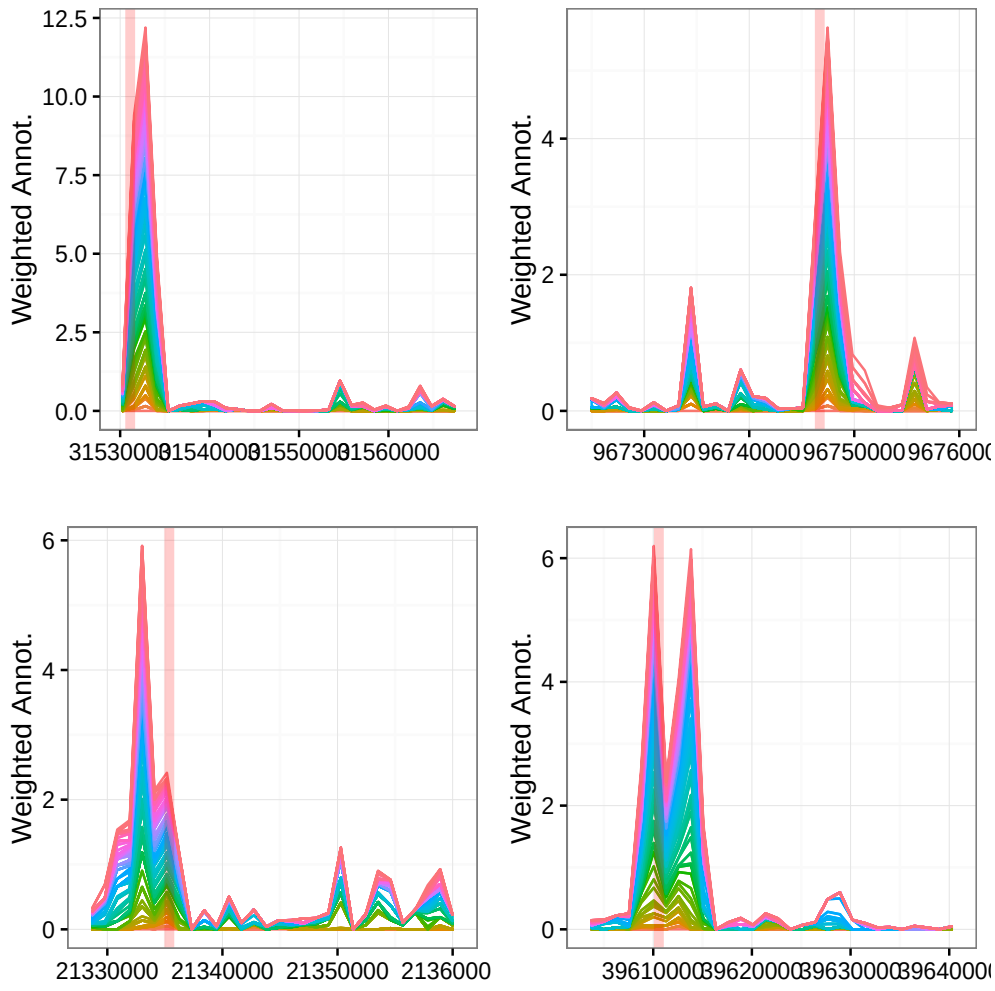

Figure 4: Underlying weighted annotations of the Manhanttan plots shown in Fig. 4.

```
Running under: OS X 10.11.5 (El Capitan)

locale:
[1] C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats4    parallel  stats     graphics  grDevices utils     datasets
[8] methods   base

other attached packages:
 [1] GenomicRanges_1.24.2 GenomeInfoDb_1.8.1   IRanges_2.6.1
 [4] S4Vectors_0.10.1     BiocGenerics_0.18.0  gridExtra_2.2.1
 [7] ggplot2_2.1.0        scales_0.4.0         ROCR_1.0-7
[10] gplots_3.0.1         RiVIERAbeta_0.9.1    reshape_0.8.5

loaded via a namespace (and not attached):
 [1] Rcpp_0.12.5         XVector_0.12.0      magrittr_1.5        zlibbioc_1.18
 [5] munsell_0.4.3       colorspace_1.2-6    stringr_1.0.0       plyr_1.8.4
 [9] caTools_1.17.1      tools_3.3.1         grid_3.3.1          gtable_0.2.0
[13] KernSmooth_2.23-15 gtools_3.5.0        digest_0.6.9        reshape2_1.4.1
[17] bitops_1.0-6        labeling_0.3        gdata_2.17.0        stringi_1.1.1
```

References

[1] Yue Li and Manolis Kellis. Joint bayesian inference of risk variants and tissue-specific epigenomic enrichments across multiple complex human diseases. *bioRxiv*, 1, 2015.