Replication validation based on reviewer comments

*Johann Hawe | johann.hawe@tum.de*

Contents

Load data

We load all previously generated data, including the collected priors and the model fits for KORA and LOLIPOP.

- Number of sentinels to be processed: 551

Prior importance for replication

We investigate whether available prior information largely drives replication performance. In brief, we create a contingency table from all sentinels, where we check 1) whether an edge is replicated or not and 2) whether the edge has a prior assigned it or not.

```
## # A tibble: 4,408 x 17
##    graph_model estimate  p.value conf.low conf.high method alternative
##    <chr>          <dbl>    <dbl>    <dbl>     <dbl> <chr>  <chr>
## 1 bdgraph         58.8 1.79e- 8    19.5       Inf Fishe~ greater
## 2 bdgraph_no~      0   1.00e+ 0     0         Inf Fishe~ greater
## 3 bdgraph_no~      0   1.00e+ 0     0         Inf Fishe~ greater
## 4 irafnet          0   1.00e+ 0     0         Inf Fishe~ greater
## 5 genenet          0   1.00e+ 0     0         Inf Fishe~ greater
## 6 glasso         307.  6.46e-25   113.        Inf Fishe~ greater
## 7 glasso_no_~      0   1.00e+ 0     0         Inf Fishe~ greater
## 8 genie3           0   1.00e+ 0     0         Inf Fishe~ greater
## 9 bdgraph         85.6 2.27e-29    44.4       Inf Fishe~ greater
## 10 bdgraph_no~    10.1 2.47e- 6     4.63      Inf Fishe~ greater
## # ... with 4,398 more rows, and 10 more variables: set1_in_background <dbl>,
## #   set1_not_in_background <dbl>, set2_in_background <dbl>,
## #   set2_not_in_background <dbl>, sentinel <chr>, prior_and_replicated <dbl>,
## #   prior_not_replicated <dbl>, not_prior_and_replicated <dbl>,
## #   not_prior_not_replicated <dbl>, is_prior_based <lgl>

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 702 rows containing non-finite values (stat_bin).
```
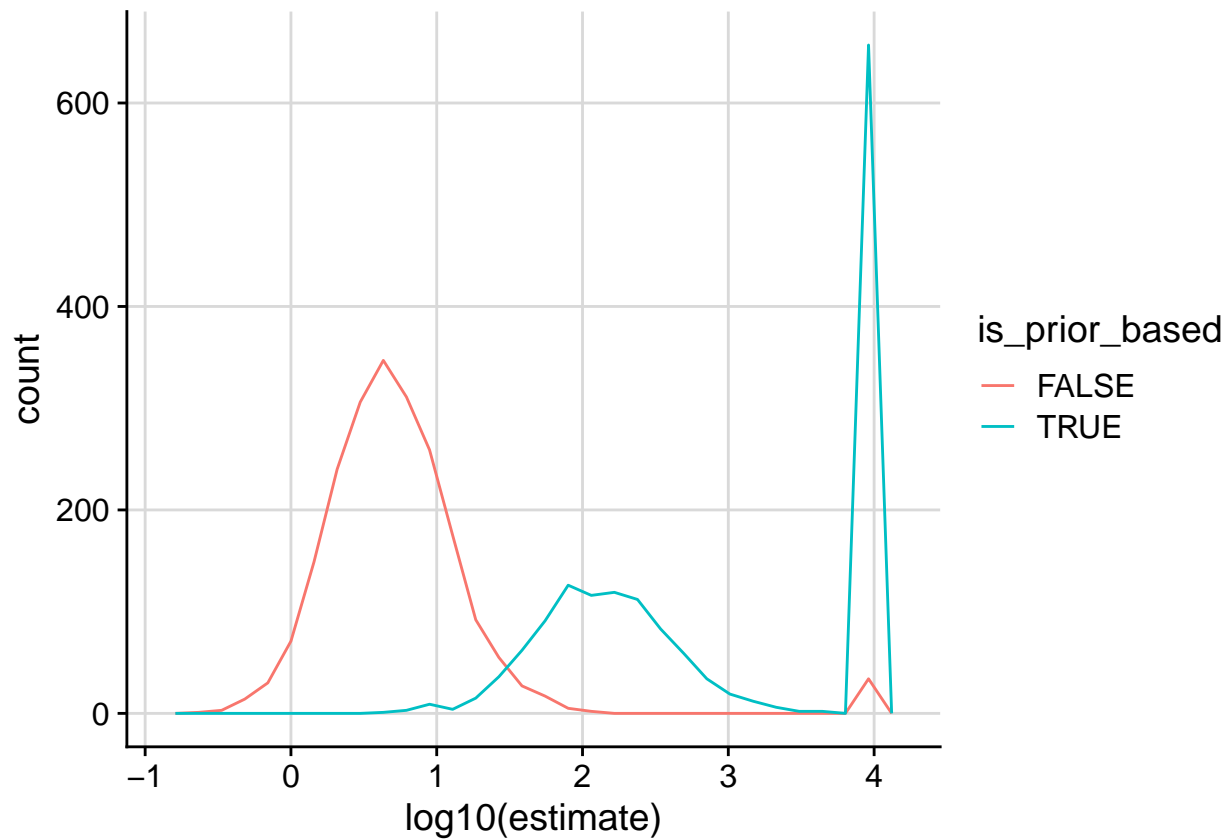
Above are the initial 'full' results, i.e. for ea ch graph model and sentinel, we see the evaluation by use of the fisher test.

The plot indicates the distribution (log10) of estimates over all models and sentinels. `estimates` amounging to `Inf` were substituted by `10e4` and result in the peaks located at the far right of the plot.

Now we look at a simple summary, where we sum up all contingency tables and calculate a single fisher test for each of the graph models.

```
## # A tibble: 8 x 12
##   graph_model estimate p.value conf.low conf.high method alternative
##   <chr>          <dbl>   <dbl>    <dbl>     <dbl> <chr>  <chr>
## 1 bdgraph         238.       0   230.        Inf Fishe~ greater
## 2 bdgraph_no~     4.62       0     4.51      Inf Fishe~ greater
## 3 bdgraph_no~     1.87       0     1.85      Inf Fishe~ greater
## 4 genenet         6.38       0     6.04      Inf Fishe~ greater
## 5 genie3          2.58       0     2.54      Inf Fishe~ greater
## 6 glasso          373.       0   365.        Inf Fishe~ greater
## 7 glasso_no_~     2.94       0     2.87      Inf Fishe~ greater
## 8 irafnet        1257.       0  1078.        Inf Fishe~ greater
## # ... with 5 more variables: prior_and_replicated <dbl>,
## #   prior_not_replicated <dbl>, not_prior_and_replicated <dbl>,
## #   not_prior_not_replicated <dbl>, is_prior_based <lgl>
```
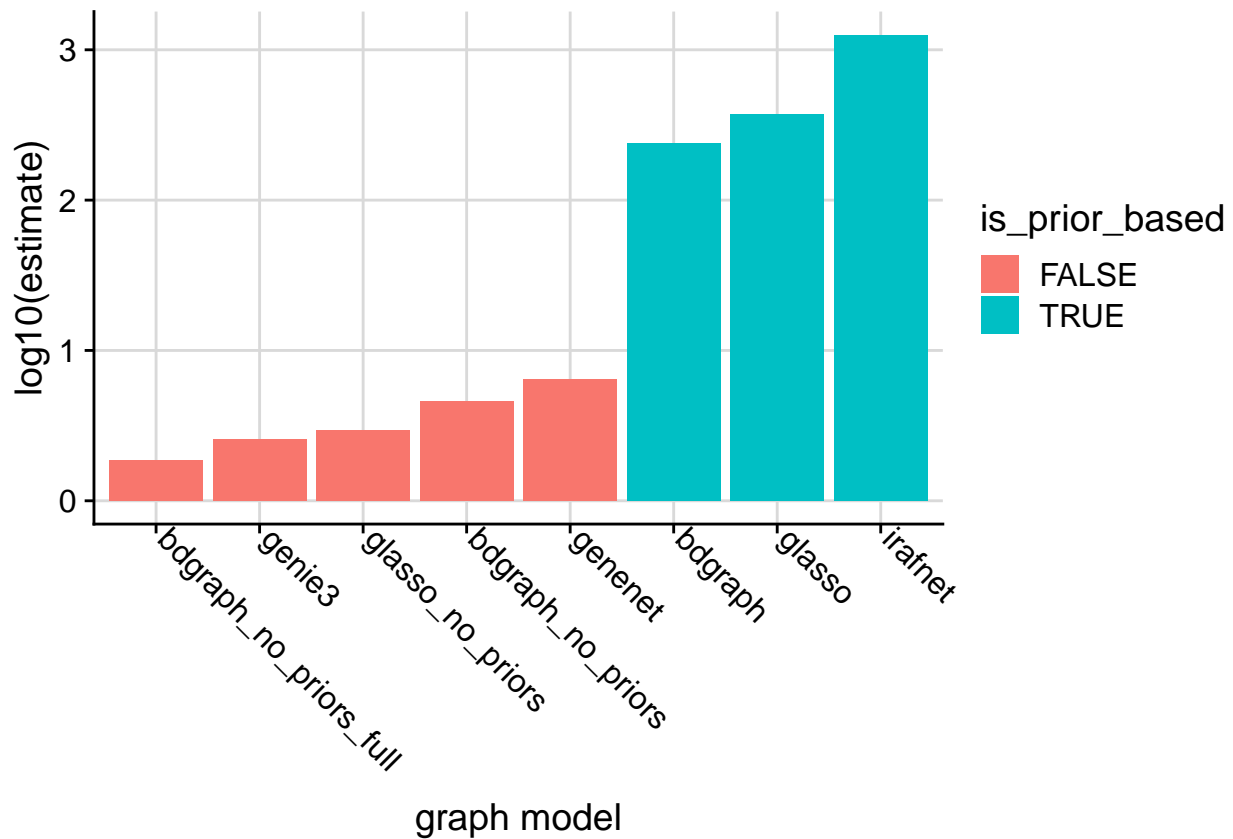
Prior importance inferred edge set

Contrary to the replicated edge set above, we can also have a look at the 'inferred' edge set, which are defined as the union of the edge sets from KORA and LOLIPOP.

```
## # A tibble: 4,408 x 17
##     sentinel graph_model estimate  p.value conf.low conf.high method alternative
##     <chr>    <chr>          <dbl>    <dbl>    <dbl>     <dbl> <chr>  <chr>
##  1 rs10103~ bdgraph        22.9   1.24e- 8    9.55      Inf Fishe~ greater
##  2 rs10103~ bdgraph_no~     0.596 8.16e- 1    0.0290    Inf Fishe~ greater
##  3 rs10103~ bdgraph_no~     0.620 8.04e- 1    0.0301    Inf Fishe~ greater
##  4 rs10103~ irafnet        46.7   1.83e- 3    5.95      Inf Fishe~ greater
##  5 rs10103~ genenet         1.43  5.17e- 1    0.0690    Inf Fishe~ greater
##  6 rs10103~ glasso        208.    2.06e-24   67.9       Inf Fishe~ greater
##  7 rs10103~ glasso_no_~     0     1.00e+ 0    0         Inf Fishe~ greater
##  8 rs10103~ genie3          0     1.00e+ 0    0         Inf Fishe~ greater
##  9 rs10120~ bdgraph       119.    2.61e-46   62.6       Inf Fishe~ greater
## 10 rs10120~ bdgraph_no~    15.4   3.31e-17    9.07      Inf Fishe~ greater
## # ... with 4,398 more rows, and 9 more variables: set1_in_background <dbl>,
## #   set1_not_in_background <dbl>, set2_in_background <dbl>,
## #   set2_not_in_background <dbl>, prior_and_inferred <dbl>,
## #   prior_not_inferred <dbl>, not_prior_and_inferred <dbl>,
## #   not_prior_not_inferred <dbl>, is_prior_based <lgl>

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 339 rows containing non-finite values (stat_bin).
```
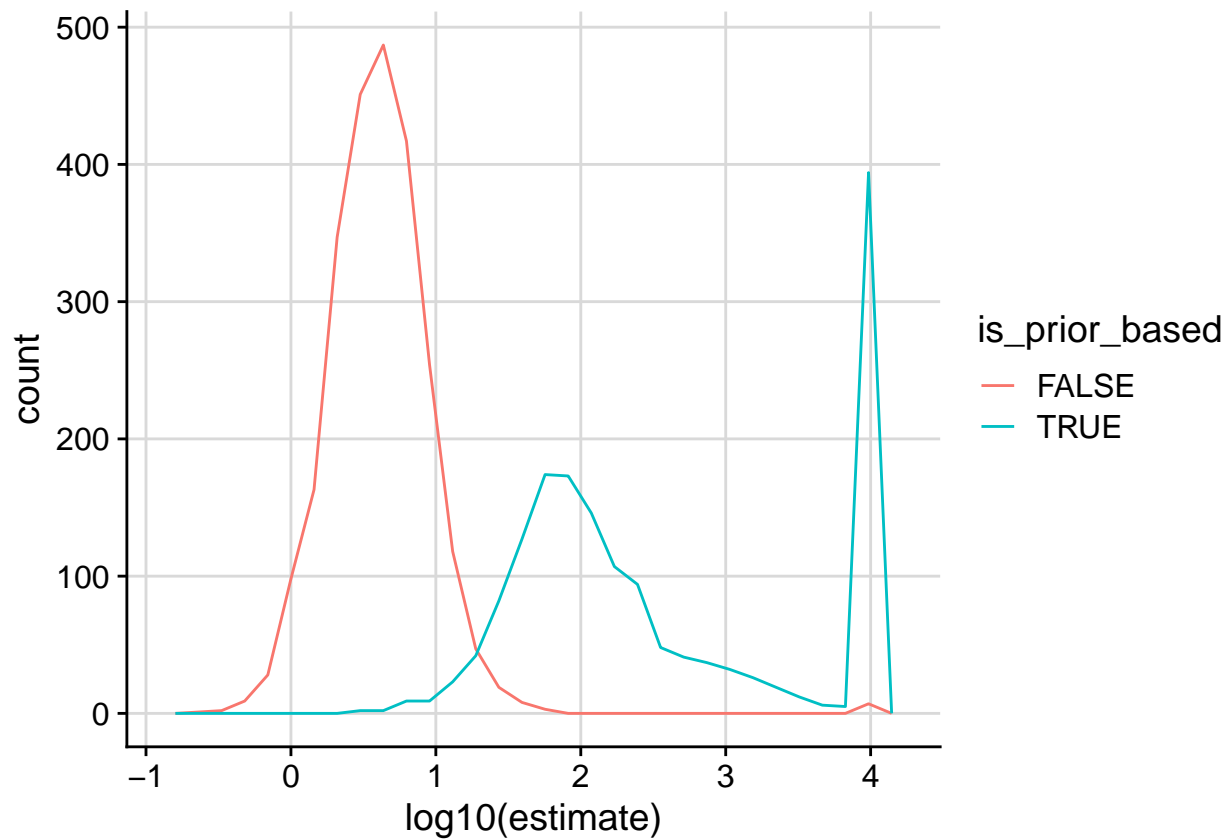
```
## # A tibble: 8 x 12
##   graph_model estimate p.value conf.low conf.high method alternative
##   <chr>          <dbl>   <dbl>    <dbl>     <dbl> <chr>  <chr>
## 1 bdgraph        148.        0   146.        Inf Fishe~ greater
## 2 bdgraph_no~      3.37      0     3.33      Inf Fishe~ greater
## 3 bdgraph_no~      2.83      0     2.80      Inf Fishe~ greater
## 4 genenet          4.39      0     4.26      Inf Fishe~ greater
## 5 genie3           2.82      0     2.80      Inf Fishe~ greater
## 6 glasso         139.        0   136.        Inf Fishe~ greater
## 7 glasso_no_~      2.79      0     2.76      Inf Fishe~ greater
## 8 irafnet        685.        0   646.        Inf Fishe~ greater
## # ... with 5 more variables: prior_and_inferred <dbl>,
## #   prior_not_inferred <dbl>, not_prior_and_inferred <dbl>,
## #   not_prior_not_inferred <dbl>, is_prior_based <lgl>
```
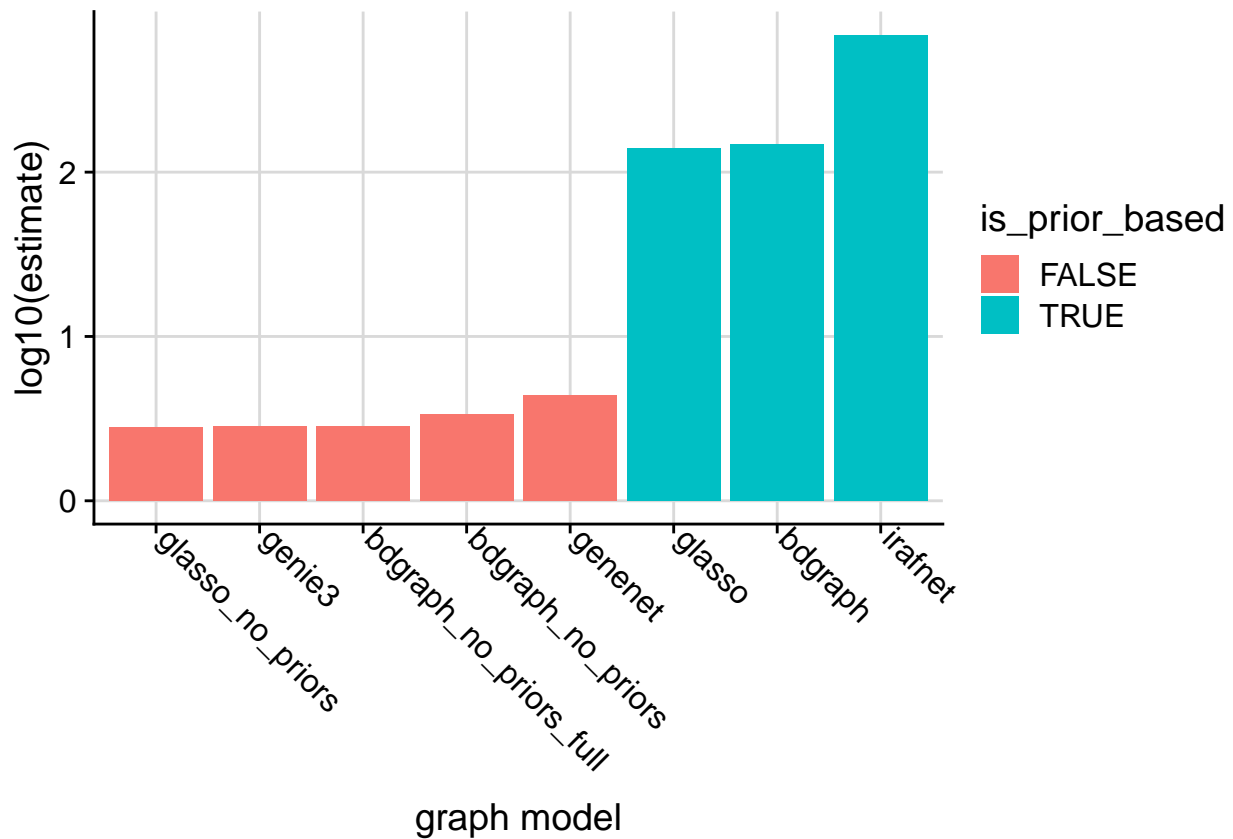
Replication with prior noise

In a next step, we investigate replication performance under consideration of differing levels of priors noise. First we have a look at the enrichment of prior edges over non-prior edges within the set of replicated edges.

> NOTE: at the beginning we defined the edges sets using the ORIGINAL (not noisified) prior matrix. We now switched to using the noisified one. Naturally, this changes our previous obesrvation of decreased enrichment with increase in prior noise (we rather observed only the 'decrease in true signal in the priors' than any performance changes...)

```
## Warning: Removed 19 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 19 rows containing non-finite values (stat_summary).
```

Enrichment of prior over non-prior edges in replicated edges
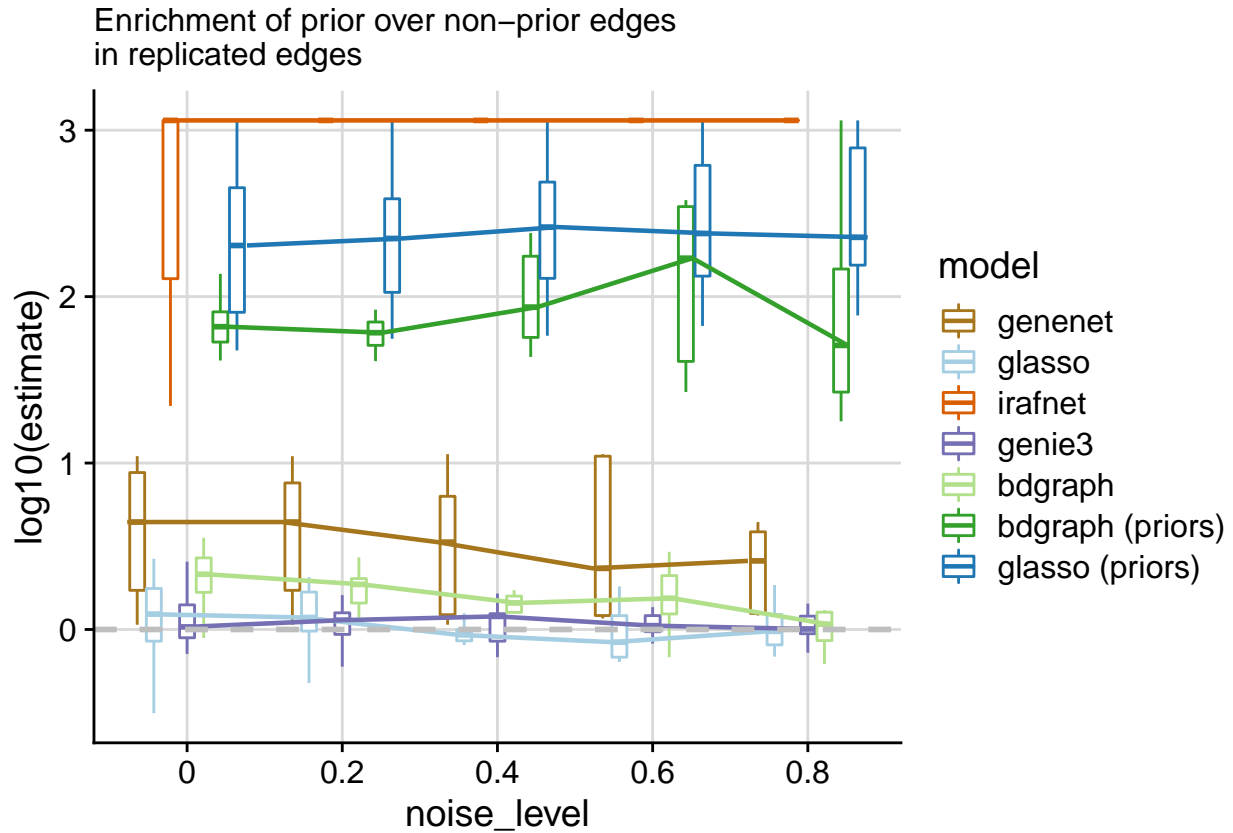
Now we check out the enrichment of **false prior** edges over the **true prior** edges within the set of replicated edges.

```
## Warning: Removed 19 rows containing non-finite values (stat_boxplot).

## Warning: Removed 19 rows containing non-finite values (stat_summary).

## Warning: Removed 19 rows containing non-finite values (stat_boxplot).

## Warning: Removed 19 rows containing non-finite values (stat_summary).

## Warning: Removed 52 rows containing non-finite values (stat_boxplot).

## Warning: Removed 52 rows containing non-finite values (stat_summary).
```

A Enrichment of prior over non–prior edges in replicated edges
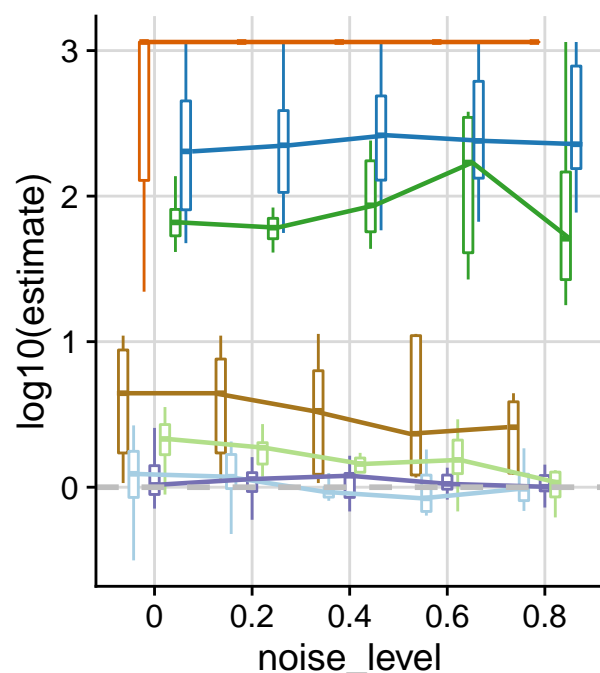
B Enrichment of false priors over true in replicated edges.
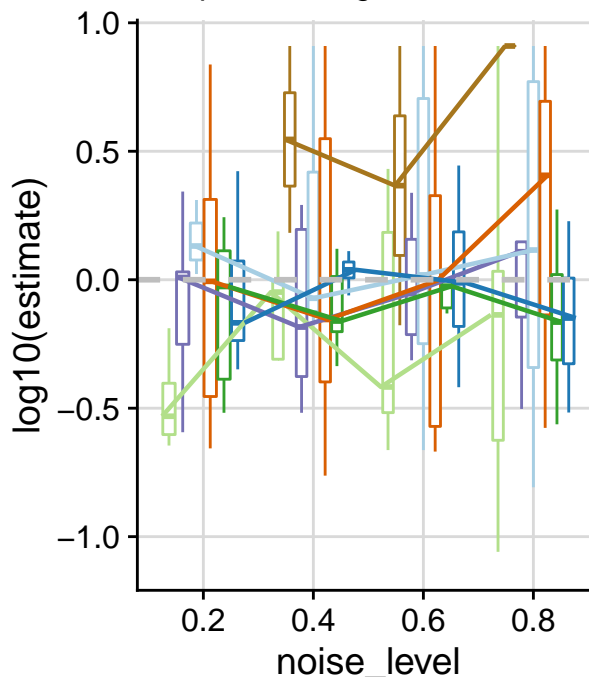
model: genenet, irafnet, bdgraph, glasso (priors), glasso, genie3, bdgraph (priors)

Session Info

```
## - Session info ---------------------------------------------------------------
##   setting  value
##   version  R version 3.6.1 (2019-07-05)
##   os       Debian GNU/Linux 9 (stretch)
##   system   x86_64, linux-gnu
##   ui       X11
##   language (EN)
##   collate  en_US.UTF-8
##   ctype    en_US.UTF-8
##   tz       Etc/UTC
##   date     2021-04-14
##
## - Packages -------------------------------------------------------------------
##   package     * version date       lib source
##   assertthat    0.2.1   2019-03-21 [1] CRAN (R 3.6.1)
##   backports     1.1.5   2019-10-02 [1] CRAN (R 3.6.1)
##   BDgraph     * 2.62    2019-12-05 [1] CRAN (R 3.6.1)
##   BiocGenerics * 0.32.0 2019-10-29 [1] Bioconductor
##   broom         0.5.2   2019-04-07 [1] CRAN (R 3.6.1)
##   callr         3.3.2   2019-09-22 [1] CRAN (R 3.6.1)
##   cli           1.1.0   2019-03-19 [1] CRAN (R 3.6.1)
##   colorspace    1.4-1   2019-03-18 [1] CRAN (R 3.6.1)
##   cowplot     * 1.0.0   2019-07-11 [1] CRAN (R 3.6.1)
```

```
## crayon       1.3.4   2017-09-16 [1] CRAN (R 3.6.1)
## desc         1.2.0   2018-05-01 [1] CRAN (R 3.6.1)
## devtools     2.2.1   2019-09-24 [1] CRAN (R 3.6.1)
## digest       0.6.23  2019-11-23 [1] CRAN (R 3.6.1)
## dplyr      * 0.8.3   2019-07-04 [1] CRAN (R 3.6.1)
## ellipsis     0.3.0   2019-09-20 [1] CRAN (R 3.6.1)
## evaluate     0.14    2019-05-28 [1] CRAN (R 3.6.1)
## fansi        0.4.0   2018-10-05 [1] CRAN (R 3.6.1)
## farver       2.0.1   2019-11-13 [1] CRAN (R 3.6.1)
## fs           1.3.1   2019-05-06 [1] CRAN (R 3.6.1)
## generics     0.0.2   2018-11-29 [1] CRAN (R 3.6.1)
## ggplot2    * 3.2.1   2019-08-10 [1] CRAN (R 3.6.1)
## ggpubr       0.2.4   2019-11-14 [1] CRAN (R 3.6.1)
## ggsignif     0.6.0   2019-08-08 [1] CRAN (R 3.6.1)
## glue         1.3.1   2019-03-12 [1] CRAN (R 3.6.1)
## graph      * 1.64.0  2019-10-29 [1] Bioconductor
## gridExtra    2.3     2017-09-09 [1] CRAN (R 3.6.1)
## gtable       0.3.0   2019-03-25 [1] CRAN (R 3.6.1)
## hms          0.5.2   2019-10-30 [1] CRAN (R 3.6.1)
## htmltools    0.4.0   2019-10-04 [1] CRAN (R 3.6.1)
## igraph     * 1.2.4.2 2019-11-27 [1] CRAN (R 3.6.1)
## knitr        1.26    2019-11-12 [1] CRAN (R 3.6.1)
## labeling     0.3     2014-08-23 [1] CRAN (R 3.6.1)
## lattice      0.20-38 2018-11-04 [2] CRAN (R 3.6.1)
## lazyeval     0.2.2   2019-03-15 [1] CRAN (R 3.6.1)
## lifecycle    0.1.0   2019-08-01 [1] CRAN (R 3.6.1)
## magrittr     1.5     2014-11-22 [1] CRAN (R 3.6.1)
## memoise      1.1.0   2017-04-21 [1] CRAN (R 3.6.1)
## munsell      0.5.0   2018-06-12 [1] CRAN (R 3.6.1)
## nlme         3.1-140 2019-05-12 [2] CRAN (R 3.6.1)
## pillar       1.4.2   2019-06-29 [1] CRAN (R 3.6.1)
## pkgbuild     1.0.6   2019-10-09 [1] CRAN (R 3.6.1)
## pkgconfig    2.0.3   2019-09-22 [1] CRAN (R 3.6.1)
## pkgload      1.0.2   2018-10-29 [1] CRAN (R 3.6.1)
## plyr         1.8.4   2016-06-08 [1] CRAN (R 3.6.1)
## prettyunits  1.0.2   2015-07-13 [1] CRAN (R 3.6.1)
## processx     3.4.1   2019-07-18 [1] CRAN (R 3.6.1)
## ps           1.3.0   2018-12-21 [1] CRAN (R 3.6.1)
## purrr        0.3.3   2019-10-18 [1] CRAN (R 3.6.1)
## R6           2.4.1   2019-11-12 [1] CRAN (R 3.6.1)
## RColorBrewer 1.1-2   2014-12-07 [1] CRAN (R 3.6.1)
## Rcpp         1.0.3   2019-11-08 [1] CRAN (R 3.6.1)
## readr      * 1.3.1   2018-12-21 [1] CRAN (R 3.6.1)
## remotes      2.1.0   2019-06-24 [1] CRAN (R 3.6.1)
## reshape2   * 1.4.3   2017-12-11 [1] CRAN (R 3.6.1)
## rlang        0.4.2   2019-11-23 [1] CRAN (R 3.6.1)
## rmarkdown    1.17    2019-11-13 [1] CRAN (R 3.6.1)
## rprojroot    1.3-2   2018-01-03 [1] CRAN (R 3.6.1)
## scales       1.1.0   2019-11-18 [1] CRAN (R 3.6.1)
## sessioninfo  1.1.1   2018-11-05 [1] CRAN (R 3.6.1)
## stringi      1.4.3   2019-03-12 [1] CRAN (R 3.6.1)
## stringr      1.4.0   2019-02-10 [1] CRAN (R 3.6.1)
## testthat     2.3.0   2019-11-05 [1] CRAN (R 3.6.1)
## tibble       2.1.3   2019-06-06 [1] CRAN (R 3.6.1)
```

```
##  tidyr        1.0.0   2019-09-11 [1] CRAN (R 3.6.1)
##  tidyselect   0.2.5   2018-10-11 [1] CRAN (R 3.6.1)
##  usethis      1.5.1   2019-07-04 [1] CRAN (R 3.6.1)
##  utf8         1.1.4   2018-05-24 [1] CRAN (R 3.6.1)
##  vctrs        0.2.0   2019-07-05 [1] CRAN (R 3.6.1)
##  withr        2.1.2   2018-03-15 [1] CRAN (R 3.6.1)
##  xfun         0.11    2019-11-12 [1] CRAN (R 3.6.1)
##  yaml         2.2.0   2018-07-25 [1] CRAN (R 3.6.1)
##  zeallot      0.1.0   2018-01-28 [1] CRAN (R 3.6.1)
##
## [1] /usr/local/lib/R/site-library
## [2] /usr/local/lib/R/library
```