Summaries of power and ranking simulations

2023-05-05

Contents

1 Introduction

- In this document we read in the simulation results output by `simalt.R` and prepare summaries that can be included in our paper.
- The "output" of this document is four LaTeX tables of power estimates and their SEs in Section 4.1.6 and a graphical summary of the ranking results in Section 4.2.6.
- For publication, the graph is saved as an encapsulated postscript file named `rankres.eps` (saved in this directory).
- Before getting to the "output" we read in the data, and do some exploratory summaries of the simulation results to get a feel for the data and look for possible errors in the simulation code.
- **Note:** Other RMarkdown documents in our workflow were written as documentation for a corresponding R script that was intended to be run on the cluster using a SLURM script. By contrast, this RMarkdown document is intended to be run, or "knitted" on your PC, and there are no corresponding R and SLURM scripts.

2 Read in simulation results

- `simalt.R` returns files of p-value results and ranking results.
- Simulations were run on the Compute Canada cluster as an array of 200 jobs each containing 10 studies (of three pedigrees). Results were saved on the cluster in the `/project/def-jgraham/FJdata` directory. You must first copy the results files from the cluster to your PC. There are several options for doing the file transfer that are discussed in Appendix A.1.4 of the Graham and McNeney Labs Workflow document. I use the `rsync` method, for which I (i) open a terminal on my Mac and set its working directory to the directory that contains this .Rmd file, and (ii) run the following command from the terminal:

```
rsync -avz jgraham@cedar.computecanada.ca:project/FJdata/ FJdata
```

- p-values to estimate power are in the files `FJdata/pvalresi.csv` for i=1,. . . ,200 and ranking results are in the files `FJdata/rankresi.csv`, for i=1,. . . ,200.
- The format of the output files was described in `simalt.Rmd` and this description is repeated in the Appendix of this document.
- We read the results into R in the following code chunk:

```
njobs <- 200
pvalres <- read.csv("FJdata/pvalres1.csv") # start with first batch
for(i in 2:njobs){
  pvalres <- rbind(pvalres,read.csv(paste0("FJdata/pvalres",i,".csv")))
}
rankres <- read.csv("FJdata/rankres1.csv")
for(i in 2:njobs){
  rankres <- rbind(rankres,read.csv(paste0("FJdata/rankres",i,".csv")))
}
rankres[,ncol(rankres)] <- as.numeric(rankres[,ncol(rankres)])
```

3 Exploratory summaries of simulations

- Our first set of summaries are meant to explore the data. Exploratory summaries give us a feel for the data and can identify errors in the simulation code.

3.1 Sampling of pedigrees

- Check that pedigrees from our pool of 55 are being sampled uniformly over the 2000 simulated studies.
- We take the information about which pedigrees were sampled from the study information in `pvalres`.

```
# Read in the IDs of the pedigrees in our pool of 55
pedpool <- scan("FJdata/pedpool/pedpool.txt")
pedpool
```

```
##  [1]  2  3  4  5  8 11 12 13 15 16 18 19 20 22 23 24 25 27 29 30 31 33 34 37 38
## [26] 39 40 41 43 44 45 46 49 50 51 52 53 54 55 56 57 58 59 61 62 63 66 67 69 70
## [51] 73 74 75 76 77
```

```
# Read in the IDs of the pedigrees sampled across the 2000 studies.
peds <- c(pvalres[,"studyped_1"], pvalres[,"studyped_2"],pvalres[,"studyped_3"])
# Check whether every pedigree in the pool of 55 was sampled
all(pedpool %in% peds) # every ped in the pool of 55 was sampled at least once
```

```
## [1] TRUE
```

```
# See if pedigrees in the pool of 55 have been sampled roughly uniformly. There
# are 6000 study peds (2000 studies containing 3 peds each), so each pedigree
# should be sampled about 6000/55 = 109 times.
table(peds) # some sampled more than others by chance
```

```
## peds
##   2   3   4   5   8  11  12  13  15  16  18  19  20  22  23  24  25  27  29  30 
## 120 113 121 110 112 106  99 110 114 108 113 117 110 100  97 113 107 111 107 113 
##  31  33  34  37  38  39  40  41  43  44  45  46  49  50  51  52  53  54  55  56 
## 107 102 102 108  96 111 114 101 122  94 106 102 104 119  98 107 110 127 103 112 
##  57  58  59  61  62  63  66  67  69  70  73  74  75  76  77 
## 119 115 108 108 110  95 109 101 106 114 120  99 107 122 121
```

```
sum(table(peds)) # should be 6000=2000*3.
```

```
## [1] 6000
```

3.2 Number of cRVs sampled per study

- There can be 1, 2 or 3 cRVs in the three pedigrees that make a study. How many cRVs were sampled per study in our simulated data?
- In the code chunk below we find 34 studies with only 1 cRV, 612 studies with 2 cRVs and 1359 with 3 cRVs, for a total of 5320 cRVs over the 2000 simulated studies.
- In the majority of simulated studies (1359/2000=68%), the pedigrees harbour distinct cRVs.

```
numcRVs <- rep(NA,nrow(pvalres))
for(i in 1:nrow(pvalres)){
  # Coding note on next command: When we subset the data frame pvalres
  # R returns a data frame. We want to coerce to a vector, which we do
  # with unlist().
  cRVstudy <- unlist(pvalres[i,c("cRV_1","cRV_2","cRV_3")])
  numcRVs[i] <- length(unique(cRVstudy))
}
table(numcRVs)
```

```
## numcRVs
##    1    2    3 
##   34  612 1354
```

3.3 cRV sampling proportions

- Christina's `sim_RVstudy()` is supposed to sample a cRV for a pedigree in proportion to its population frequency. Specifically, each cRV should be sampled according to its conditional probability given that it is one of the 10 cRVs. Here we check the frequency of each cRV appearing in the 2000 simulated studies against its conditional population frequency.
- In the following code chunk we find that the empirical frequencies from the simulated data are roughly similar to the conditional population frequencies.

```
cRVs <- c(pvalres[,"cRV_1"],pvalres[,"cRV_2"],pvalres[,"cRV_3"])
empfreq <- table(cRVs)/length(cRVs) # ordered alphabetically
load("FJdata/chr8.RData")
aa <- chr8$Mutations[chr8$Mutations$is_CRV,
                                 c("afreq","SNV")]
aa$afreq <- aa$afreq/sum(aa$afreq) # normalize pop freqs to conditional pop freqs
aa <- aa[order(aa$SNV),] # order rows alph by SNV
aa$empfreq <- empfreq # RVs in empfreq are in same order as in aa
aa # roughly similar conditional population (afreq) and sample freqs (empfreq)
```

```
##              afreq         SNV    empfreq
## 24059 0.11764706 8_118923860 0.11450000
## 24074 0.11764706 8_118933213 0.12283333
## 5725  0.05882353  8_23020319 0.06216667
## 5726  0.05882353  8_23020346 0.05683333
## 5729  0.11764706  8_23020454 0.11983333
## 5740  0.17647059  8_23021159 0.17933333
## 5748  0.05882353  8_23021957 0.05550000
## 5756  0.05882353  8_23022671 0.05700000
## 5770  0.11764706  8_23068278 0.11183333
## 5772  0.11764706  8_23068413 0.12016667
```

3.4 Number of RVs per study

- Both Christina and Nirodha used the software SLiM (see http://benhaller.com/slim/SLiM_Manual.pdf) to simulate SNV sequences in the population. These population sequences were then used to "seed" the pedigree founders. They ran their SLiM simulations differently and I was curious about how this would affect the number of RVs in each study. The number of RVs in a study is the number that are observed in the affected individuals, and so varies randomly from one study to the next. The distribution of the number of RVs should reflect the total number of variants in the simulated population of chromosome 8's. I don't know the number of RVs in Christina's population, and am unsure about whether Christina's simulation study should have more RVs than ours.
  - Christina simulated selectively-neutral markers in a population of constant size 50,000 diploid individuals. Her simulations yielded an average of 120.4 RVs per study with a SD of 15.3.
  - Nirodha's SLiM simulation was much more realistic (see the description in her Supplementary Materials 1-A document), imposing negative selection and an established demographic model for a North American admixed population. The negative selection is expected to lead to *more* RVs than Christina's, but the demographic model would lead to *fewer* RVs, because overall Nirodha's population size is smaller than Christina's. Nirodha's population size was between 15,000 and 30,000 individuals from the beginning up to 12 generations ago, before reaching a present-day size of 53,876 individuals.
- I also wondered about any association between the number of RVs and the number of sampled cRVs in a study. I look for an association informally with boxplots of the number of RVs by the number of cRVs.
- From the following code chunk we find an average of 84.8 RVs per study (SD 12.0) which is smaller than in Christina's study. The boxplots suggest no association between the number of cRVs and the number of RVs in a study.

```
summary(rankres$numRV)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   51.00   77.00   84.00   84.85   93.00  125.00
```

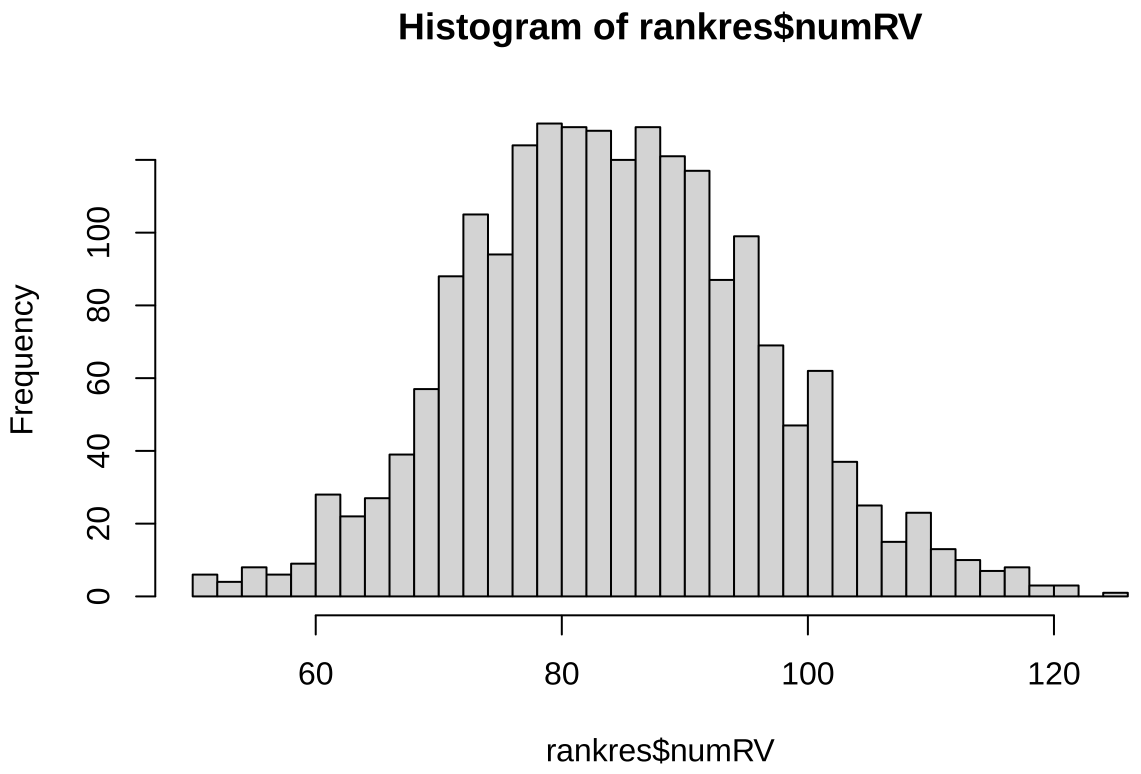
```
mean(rankres$numRV) # 84.85, so smaller than Christina's 120.4
```

```
## [1] 84.8485
```

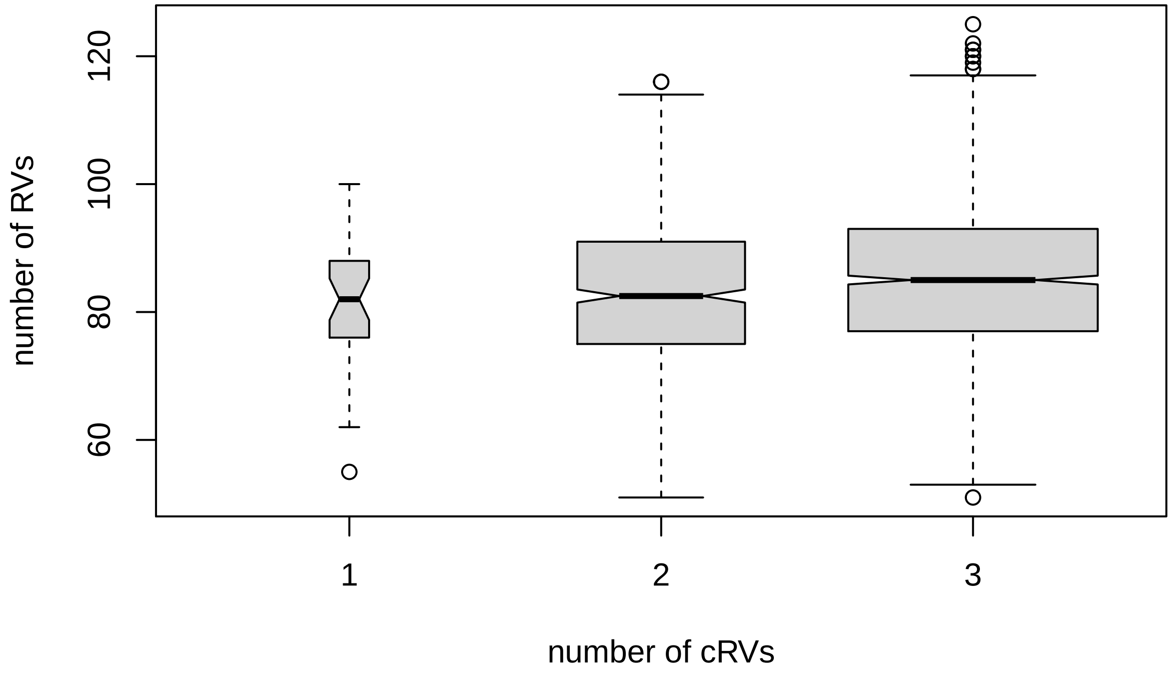
```
sd(rankres$numRV) # 12.0, also smaller than Christina's 15.3
```

```
## [1] 12.01744
```

```
hist(rankres$numRV,nclass=30) # roughly symetric distribution
```



```
boxplot(split(rankres$numRV,numcRVs),varwidth=TRUE, notch=TRUE,
        xlab="number of cRVs",ylab="number of RVs")
```



```
# No obvious differences in the numver of RVs by number of cRVs
```

4 Summaries for the paper

- The second set of summaries are of estimated power and ranking. These are to be included in the paper.
- Christina estimated power as the proportion of *all* cRV tests that reject the null hypothesis that $\tau_a = \tau_b = 1/2$ in favour of the alternative hypothesis that $\tau_a < \tau_b \leq 1/2$ at the 5% level. Over her 1000 simulated studies she sampled 2543 cRVs and hence had 2543 p-values for her power estimates.
  - My recollection is that she also looked at power using the cRVs that had the smallest p-value in each study, but found that all methods had very high power when measured this way, and so there was very little difference between the methods.
  - Below I've estimated power the same way as Christina, over all cRVs, but also, out of curiosity, power with (i) the cRVs from each study that had the smallest p-value, (ii) the cRVs that had the second-smallest p-value, if applicable, and (iii) the cRVs that had the third-smallest p-value.
- For the ranking results, Christina showed the median and IQR of the ranks for the top-ranked, second-ranked (if applicable) and third-ranked (if applicable) cRV in each study; see her Table 4.4, page 53. In addition, I've included the median and IQR of the *average* rank over the cRVs in each study.
- To augment the numerical summaries I've added a graphical display of the medians and IQRs for each ranking method, because I think the graph makes comparison of the different methods easier than the numerical summaries.

4.1 Power results

- The global LR and global transmission tests depend on the value of the carrier probability. Recall that we considered five values, leading to five global LR and five global transmission tests. In addition there are the three local tests (local LR, RVS and modified RVS) for a total of 13 tests.

```
true_carrier_prob <- 0.00032
carrier_probs <- true_carrier_prob*c(1/10,1/2,1,2,10)
tests <- c(paste0("globalLR",1:length(carrier_probs)),
           paste0("globaltrans",1:length(carrier_probs)),
           "localLR","RVS","modRVS")
tests
```

```
##  [1] "globalLR1"    "globalLR2"    "globalLR3"    "globalLR4"    "globalLR5"
##  [6] "globaltrans1" "globaltrans2" "globaltrans3" "globaltrans4" "globaltrans5"
## [11] "localLR"      "RVS"          "modRVS"
```

- In the code chunk below we loop over the tests and find all the p-values for each one.
- For a given test, the simulation output from `simalt.R` includes p-values for each cRV in each study. The different cRVs are distinguished by the suffix `_1` for the first, `_2` for the second and `_3` for the third.
  - Here "first", "second" and "third" refer to the order the cRVs were sampled in the study, not to the size of the p-values.
  - When there are only two cRVs in a study the third p-values are `NA` and when there is only one cRV in a study the second and third p-values are `NA`.
- For a given test we record all p-values (for estimating power as the proportion of all cRV tests that reject the null hypothesis) as well as the smallest, second-smallest and third-smallest for each study.

```
# Initialize matrices to hold 1st-, 2nd- and 3rd-smallest pvals,
# and also a matrix to hold all pvals. Rows of each matrix are
# studies and columns are tests.
p1 <- matrix(NA,nrow=nrow(pvalres),ncol=length(tests))
p2 <- matrix(NA,nrow=nrow(pvalres),ncol=length(tests))
p3 <- matrix(NA,nrow=nrow(pvalres),ncol=length(tests))
pall <- matrix(NA,nrow=nrow(pvalres)*3,ncol=length(tests))
for(i in 1:length(tests)) {
  pvalcols <- paste0(tests[i],"_",1:3)
  p <- pvalres[,pvalcols] # p is a data frame of p-values for this test
  pall[,i] <- unlist(p) # vector of all p-values over all cRVs in all studies
  # Loop over studies and find the smallest, second-smallest
  # and third-smallest p-value for each study
  for(j in 1:nrow(pvalres)) {
    p[j,] <- sort(as.numeric(p[j,]),na.last=TRUE)
  }
  p1[,i] <- p[,1]
  p2[,i] <- p[,2]
  p3[,i] <- p[,3]
}
colnames(p1) <- colnames(p2) <- colnames(p3) <- colnames(pall) <- tests
powerres<- function(pmat){
  ests <- apply(pmat,2,FUN=function(x) mean(x<=0.05,na.rm=TRUE))
  ses <- apply(pmat,2,FUN=se)
  res <- cbind(ests,ses); colnames(res) <- c("Estimates","SEs")
  return(res)
}
se <- function(x) {
  n <- length(x)
  p <- mean(x<=0.05,na.rm=TRUE)
  return(sqrt(p*(1-p)/n))
}
```

4.1.1 Power of tests of all cRVs

- Recall from the summaries of the number of cRVs per study that we sampled 5320 cRVs over the 2000 studies. The estimated power of the tests of all 5320 cRVs is as follows.

```
round(powerres(pall),3)
```

```
##                Estimates   SEs
## globalLR1          0.837 0.005
## globalLR2          0.836 0.005
## globalLR3          0.834 0.005
## globalLR4          0.833 0.005
## globalLR5          0.823 0.005
## globaltrans1       0.830 0.005
## globaltrans2       0.830 0.005
## globaltrans3       0.829 0.005
## globaltrans4       0.828 0.005
## globaltrans5       0.821 0.005
## localLR            0.758 0.006
## RVS                0.676 0.006
## modRVS             0.766 0.005
```

These estimated powers are summarized in Table 1 of this document.

4.1.2 Power of tests of cRV with smallest p-value

We estimate the power of tests of the cRV with smallest p-value as follows.

```
round(powerres(p1),3)
```

The estimated power of tests of the cRV with smallest p-value are not printed here but are summarized in Table 2 of this document.

4.1.3 Power of tests of cRV with second-smallest p-value

We estimate the power of tests of the cRV with second-smallest p-value as follows.

```
round(powerres(p2),3)
```

The estimated power of tests of the cRV with second-smallest p-value are not printed here but are summarized in Table 3 of this document.

4.1.4 Power of tests of cRV with third-smallest p-value

We estimate the power of tests of the cRV with third-smallest p-value as follows.

```
round(powerres(p3),3)
```

The estimated power of tests of the cRV with third-smallest p-value are not printed here but are summarized in Table 4 of this document.

4.1.5 LaTeX tables

- When you run your simulations you should get the following results for the power of the various tests.
- Christina didn't show power results for different values of the carrier probability. (However, Table 4.2 on page 51 of her thesis shows type-1 error rates under different assumed values of the carrier probabilities.)
- I organized the power estimates and SEs into LaTeX-formatted tables (see below) that are similar to Christina's table of type-1 error rates.
- Note: These tables are *not* automatically generated. They were done by hand from an empty template table (shown below) with the cells of the table filled in by cutting-and-pasting from the R output.

```
\begin{table}
\caption{Template table}
\begin{tabular}{lccccccc}
       & & \multicolumn{6}{c}{Method} \\ \cline{3-8}
       & & \multicolumn{3}{c}{Local} & & \multicolumn{2}{c}{Global} \\ \cline{3-5} \cline{7-8}
Assumed $p_c$ & & RVS & ModRVS & LR &   & LR & Transm. \\ \hline
NA & & e & e & e &    & -- & -- \\
$0.000032$& & -- & -- & -- &  & e & e \\
$0.000160$& & -- & -- & -- &  & e & e \\
$0.000320^*$& & -- & -- & -- &  & e & e \\
$0.000640$& & -- & -- & -- &  & e & e \\
$0.003200$& & -- & -- & -- &  & e & e \\ \hline
\multicolumn{8}{l}{$^*$ True carrier probability is $p_c = 0.000320$}
\end{tabular}
\end{table}
```

Table 1: Estimated power (SE), all cRVs (5320 assessed in total)

	Method				
	Local			Global	
Assumed $p_c$	RVS	ModRVS	LR	LR	Transm.
NA	0.676 (0.006)	0.766 (0.005)	0.758 (0.006)	–	–
0.000032	–	–	–	0.837 (0.005)	0.830 (0.005)
0.000160	–	–	–	0.836 (0.005)	0.830 (0.005)
0.000320*	–	–	–	0.834 (0.005)	0.829 (0.005)
0.000640	–	–	–	0.833 (0.005)	0.828 (0.005)
0.003200	–	–	–	0.823 (0.005)	0.821 (0.005)

* True carrier probability is $p_c = 0.000320$

Table 2: Estimated power (SE), top-ranked cRV (2000 in total)

	Method				
	Local			Global	
Assumed p_c	RVS	ModRVS	LR	LR	Transm.
NA	0.966 (0.004)	0.991 (0.002)	0.985 (0.003)	–	–
0.000032	–	–	–	0.992 (0.002)	0.993 (0.002)
0.000160	–	–	–	0.992 (0.002)	0.993 (0.002)
0.000320*	–	–	–	0.992 (0.002)	0.992 (0.002)
0.000640	–	–	–	0.992 (0.002)	0.992 (0.002)
0.003200	–	–	–	0.992 (0.002)	0.992 (0.002)

* True carrier probability is $p_c = 0.000320$

Table 3: Estimated power (SE), second-ranked cRV (1971 in total)

	Method				
	Local			Global	
Assumed p_c	RVS	ModRVS	LR	LR	Transm.
NA	0.670 (0.011)	0.801 (0.009)	0.785 (0.009)	–	–
0.000032	–	–	–	0.885 (0.007)	0.874 (0.007)
0.000160	–	–	–	0.885 (0.007)	0.874 (0.007)
0.000320*	–	–	–	0.883 (0.007)	0.874 (0.007)
0.000640	–	–	–	0.883 (0.007)	0.872 (0.007)
0.003200	–	–	–	0.871 (0.008)	0.866 (0.008)

* True carrier probability is $p_c = 0.000320$

- The following conclusions apply to the power estimated from the tests of all the cRVs in a study as well as the tests of the cRVs with the smallest, second smallest and third smallest p-values in a study.
- The estimated power of the global tests (LR and transmission) is very similar for all values of the carrier probability p_c, suggesting that the methods are robust to the choice of p_c.
  - There is some suggestion that the power of the global tests may even be slightly improved by under-specifying the value of p_c.
- The estimated power of the global LR test is similar to the estimated power of the global transmission test, and both are larger than the estimated power of the local tests (local LR, RVS and modified RVS).
- The estimated power of the local LR test is similar to the estimated power of the modified RVS test, and both are larger than the estimated power of the RVS test which is as expected the worst of the five tests (because it doesn't account for two subtypes).

4.2 Ranking results

- Note that the value of the global transmission statistic does not depend on the value of the carrier probability (though its *p*-value does).

```
statistics <- c(paste0("globalLR",1:length(carrier_probs)),
              "globaltrans","localLR","RVS","modRVS")
```

- The following code chunk summarizes the ranking results for each statistic. It's very similar to the code chunk above that summarizes the *p*-value results.
- We consider each test in turn.
- To account for different numbers of RVs in each study, the raw rank of a cRV, relative to all the other RVs in a study, is normalized by dividing it by the total number of RVs in that study.
- For a given test and study, the simulation output from `simalt.R` includes the normalized ranks for each cRV.
- The different cRVs in a study are distinguished by the suffix `_1` for the first, `_2` for the second and `_3` for the third that are sampled.

Table 4: Estimated power (SE), third-ranked cRV (1359 in total)

	Method				
	Local			Global	
Assumed p_c	RVS	ModRVS	LR	LR	Transm.
NA	0.256 (0.010)	0.385 (0.011)	0.383 (0.011)	–	–
0.000032	–	–	–	0.538 (0.011)	0.526 (0.011)
0.000160	–	–	–	0.535 (0.011)	0.525 (0.011)
0.000320*	–	–	–	0.530 (0.011)	0.524 (0.011)
0.000640	–	–	–	0.529 (0.011)	0.523 (0.011)
0.003200	–	–	–	0.505 (0.011)	0.503 (0.011)

* True carrier probability is $p_c = 0.000320$

  - Thus "first", "second" and "third" refer to the order the cRVs were sampled in the study, not to the size of the ranks.
  - When there are only two cRVs in a study the third p-values are `NA` and when there is only one cRV in a study the second and third p-values are `NA`.
- For a given test and study, we consider the normalized rankings for cRVs and take their study-specific average. For example, suppose a study has three cRVs with normalized ranks .1, .2 and .3 for a given test. The study-specific average of the normalized ranking for this test would then be (.1+.2+.3)/3 = .2. For a given test and study we also record the smallest, second-smallest and third-smallest normalized ranking for the cRVs. In the example, these would be .1, .2 and .3, respectively.

```
# Initialize matrices to hold 1st-, 2nd- and 3rd-smallest ranks
# and average of the tree ranks (recall that small ranks are best).
r1 <- matrix(NA,nrow=nrow(rankres),ncol=length(statistics))
r2 <- matrix(NA,nrow=nrow(rankres),ncol=length(statistics))
r3 <- matrix(NA,nrow=nrow(rankres),ncol=length(statistics))
ravg <- matrix(NA,nrow=nrow(rankres),ncol=length(statistics))
for(i in 1:length(statistics)) {
  rankcols <- paste0(statistics[i],"_",1:3)
  r <- rankres[,rankcols] # r is a data frame of ranks for this test
  # Find the average rank for this test over all cRVs in the study
  ravg[,i] <- apply(r,1,mean,na.rm=TRUE)
  # Loop over studies and find the smallest, second-smallest
  # and third-smallest rank for each study
  for(j in 1:nrow(rankres)) {
    r[j,] <- sort(as.numeric(r[j,]),na.last=TRUE)
  }
  r1[,i] <- r[,1]
  r2[,i] <- r[,2]
  r3[,i] <- r[,3]
}
colnames(r1) <- colnames(r2) <- colnames(r3) <- colnames(ravg) <- statistics
```

4.2.1 Numerical summaries

- The five number summaries and means of the normalized study-specific average rankings for each method (contained in the object `ravg` from the code chunk above) are shown below.
- For future reference, notice that the summary statistics (e.g., median) for the global LR approach are virtually identical across all the values of p_c.

```
summary(ravg)
```

```
##    globalLR1         globalLR2         globalLR3         globalLR4
##  Min.   :0.01010   Min.   :0.01010   Min.   :0.01010   Min.   :0.01010
##  1st Qu.:0.03740   1st Qu.:0.03718   1st Qu.:0.03704   1st Qu.:0.03718
##  Median :0.05624   Median :0.05622   Median :0.05624   Median :0.05620
##  Mean   :0.07526   Mean   :0.07525   Mean   :0.07526   Mean   :0.07524
##  3rd Qu.:0.08333   3rd Qu.:0.08333   3rd Qu.:0.08333   3rd Qu.:0.08333
##  Max.   :0.51942   Max.   :0.51942   Max.   :0.51942   Max.   :0.51942
##    globalLR5        globaltrans         localLR              RVS
##  Min.   :0.01010   Min.   :0.01010   Min.   :0.01010   Min.   :0.01010
##  1st Qu.:0.03723   1st Qu.:0.03745   1st Qu.:0.03704   1st Qu.:0.04527
##  Median :0.05622   Median :0.05714   Median :0.05556   Median :0.07222
##  Mean   :0.07528   Mean   :0.07796   Mean   :0.07538   Mean   :0.08697
##  3rd Qu.:0.08333   3rd Qu.:0.08434   3rd Qu.:0.08248   3rd Qu.:0.10417
##  Max.   :0.51942   Max.   :0.51942   Max.   :0.51942   Max.   :0.50515
##      modRVS
##  Min.   :0.01010
##  1st Qu.:0.04116
##  Median :0.06410
##  Mean   :0.07000
##  3rd Qu.:0.09241
##  Max.   :0.26506
```

- Summaries for the smallest, second-smallest and third-smallest normalized rankings (contained respectively in the objects `r1`, `r2` and `r3` from the code chunk above) are shown in the Appendix. As with the study-specific average rankings, the summary statistics of these rankings for the global LR approach are virtually identical across all the values of p_c.

4.2.2 Graphical summaries

- The numerical summaries are difficult to compare and interpret. To help with interpretation, we graphically compare the ranking methods.
- Since the summary statistics for the global LR methods are nearly identical across values of the carrier probability (see the Appendix), we selected the global LR method with the value of $p_c = 0.00032$ used to simulate the data as a representative on the plots.
- Coding notes: We use `ggplot()` to do the plotting with the study-specific average, smallest, second-smallest and third-smallest normalized ranking results in separate panels. rows are unique combinations of study, rank type and ranks and columns are study, rank type ranks
  - For this plot, we provide `ggplot()` a data frame with rows for combinations of study, statistic (global LR, global transmission, local LR, modified RVS or RVS) and rank type (average, top, second or third). The columns of the data frame are study, statistic, rank type and observed rank.
  - We first create a data frame with rows for combinations of study and statistics and columns for the different rank types. We then “reshape” the data frame to get rows for combinations of study, statistics and rank type and columns for study, statistics, rank type and observed rank.
  - The commands for preparing the data frame for `ggplot()` were found through lots of stackoverflow searching, cutting-and-pasting, and trial-and-error!

```
plotravg <- data.frame(study = rep(1:nrow(ravg),times=length(statistics)),
                       statistic=factor(rep(statistics,each=nrow(ravg))),
                       top.rank = as.numeric(unlist(r1)),
                       second.rank = as.numeric(unlist(r2)),
                       third.rank = as.numeric(unlist(r3)),
                       average.rank = as.numeric(unlist(ravg)))
# Filter to just globalLR3, globaltrans, localLR, RVS and modRVS
plotravg <- plotravg[plotravg$statistic %in% c("globalLR3","globaltrans","localLR","RVS","modRVS"),]
# Drop unused factor levels in statistic and shorten names
# of those that remain so they fit on the plot
```

```r
plotravg$statistic <- droplevels((plotravg$statistic))
levels(plotravg$statistic) <- c("GlobalLR","GlobalTr","LocalLR","modRVS","RVS")
# Now "reshape".
library(tidyr)
plotlong <- pivot_longer(plotravg,cols=3:6,names_to="rank.type",values_to="rank")
# Finally, re-order the levels of rank.type so that the next ggplot orders
# its panels in a sensible way.
plotlong$rank.type <- factor(plotlong$rank.type,
                        levels=c("average.rank","top.rank","second.rank","third.rank"))
```

- We can now use `ggplot()` to plot our ranks with panels for average, top, second and third rank of the statistics.
- We plot the median normalized ranks as dots and the IQR as bars.
- Summary statistics are plotted on the log scale, but y-axis labels are on the original scale of the normalized ranks.
- The red-dashed horizontal line is at a normalized rank of 0.04. The y-axes limits of the panels have been enlarged so that 0.04 is included on each panel.
- Coding note: The commands for the plot were found through lots of stackoverflow searching, cutting-and-pasting, and trial-and-error!

```r
library(ggplot2)
scaleFUN <- function(x) sprintf("%.3f", x)
ggplot(plotlong,aes(x=statistic,y=rank)) + stat_summary(
  fun.min = function(z) { quantile(z,0.25) },
  fun.max = function(z) { quantile(z,0.75) },
  fun = median) + facet_wrap(vars(rank.type),scales="free_y") +
  scale_y_continuous(trans="log",labels=scaleFUN) +
  expand_limits(y=0.04) +
  geom_hline(yintercept=0.04, linetype="dashed", color = "red")
```

```
## Warning: Removed 3400 rows containing non-finite values (`stat_summary()`).
```

- The following conclusions apply to the results on the average-rank, top-rank, second-rank and third-rank results.
- The rankings based on the global LR statistic are very similar across specified values of the carrier probability p_c.
- The summary statistics of the global LR and global transmission rankings are very similar to each other.
- The local LR approach is surprisingly good at ranking, being almost identical to the global approaches and possibly even slightly better; e.g., slightly lower median rank.
- The LR-based methods appear to rank better (e.g., smaller median rank) than the RVS and modified RVS approaches.
- As expected, the RVS approach is the worst of the five ranking methods considered because it does not take into account the disease subtypes.
- The modified RVS approach can account for disease subtypes and performs better than the naive RVS approach but not as well as the likelihood approaches.
- The following code chunk saves the above plot to an EPS (encapsulated post script) file called `rankres.eps`.
  - See Appendix C.2 of the group's workflow document for information about EPS and why we use it.

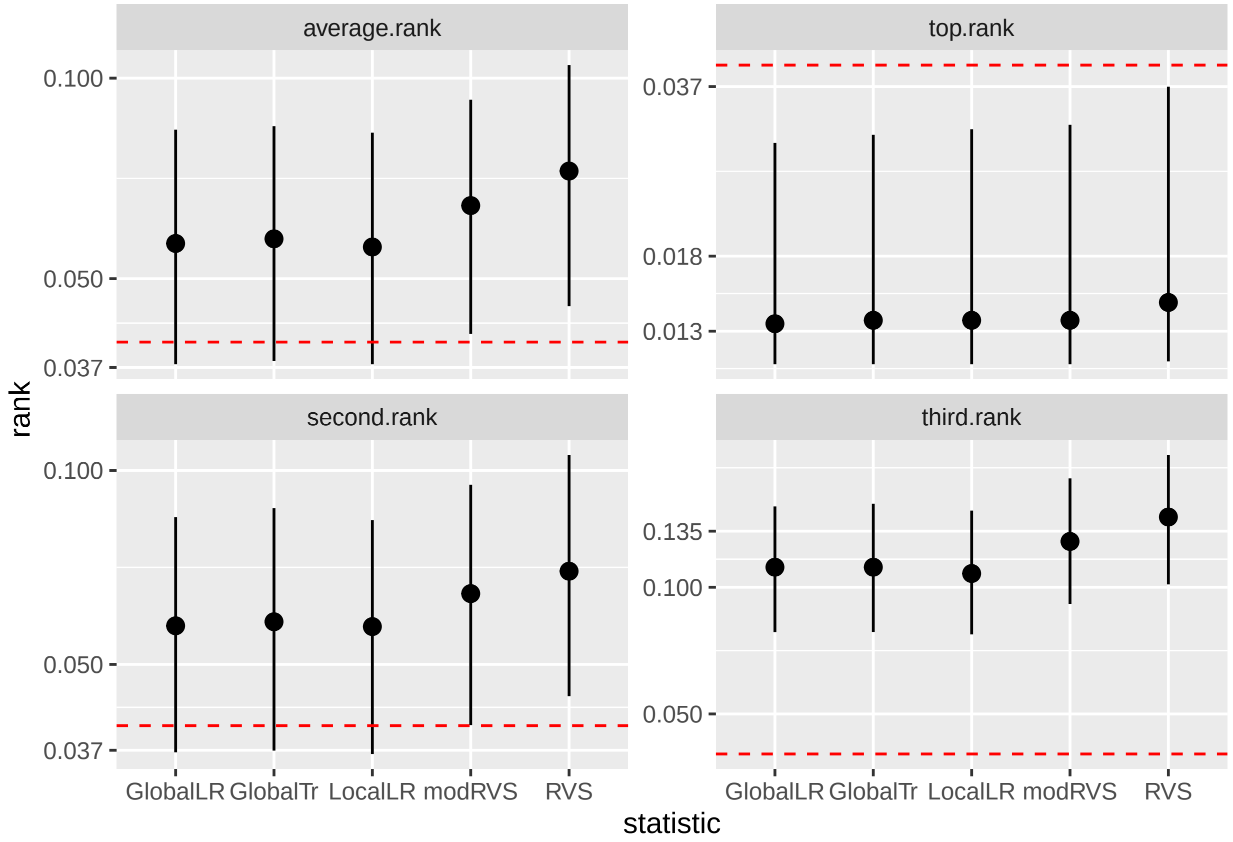


Figure 1: Median normalized ranks (dots) and IQR (bars) for the different ranking methods using the average (top left), top (top right), second (bottom left) or third (bottom right) rank in each study. Summary statistics are plotted on the log scale, but the y-axis labels are on the original scale of the normalized ranks. The red dashed line is at a normalized rank of 0.04. The global LR method shown in the plot is for $p_c = 0.00032$; the global LR method with different values of p_c had very similar rankings.

```
scaleFUN <- function(x) sprintf("%.3f", x)
rankplot <-  ggplot(plotlong,aes(x=statistic,y=rank)) + stat_summary(
  fun.min = function(z) { quantile(z,0.25) },
  fun.max = function(z) { quantile(z,0.75) },
  fun = median) + facet_wrap(vars(rank.type),scales="free_y") +
  scale_y_continuous(trans="log",labels=scaleFUN) +
  expand_limits(y=0.04) +
  geom_hline(yintercept=0.04, linetype="dashed", color = "red")
ggsave(rankplot,file="rankres.eps",device="eps")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 3400 rows containing non-finite values (`stat_summary()`).
```

A Appendix

A.1 Simulation output file format

- In both the p-value and ranking output files we include the following columns of information about the study replicate: its replicate number, the IDs of its three study pedigrees, and the IDs of the cRVs sampled in each pedigree (there will be duplicate IDs if the same cRV is sampled more than once).
- The n_p carrier probabilities have $3 \times (2 \times n_p + 3)$ p-values to record: for each of the three cRVs, we have n_p p-values for the global LR, n_p for the global transmission approaches, and one each for the three local approaches (local LR, RVS and modified RVS).
  - For the i^{th} carrier probability and j^{th} cRV, the column names are `globalLR`i`_`j for the global likelihood ratio test and `globaltrans`i`_`j for the global transmission test. For the local tests that do not depend on the carrier probability, the column names for the j^{th} cRVs are `localLR_`j, `RVS_`j and `modRVS_`j, respectively.
  - If there are fewer than three cRVs, the empty slots for p-values are encoded as `NA`.
- The n_p carrier probabilities have $3 \times (n_p + 4)$ rankings to record: for each of the cRVs, we have n_p rankings for the global LR and one each for the global transmission, local LR, RVS and modified RVS approaches. In addition, we record the number of chromosome 8 RVs that were observed in the sample (i.e., the number of RVs the cRVs were ranked against).
  - The column to hold the number of RVs per study is `numRV`.
  - For the i^{th} carrier probability and j^{th} cRV, the column names for the rankings are `globalLR`i`_`j for the global likelilihood ratio statistic. For the global transmission statistic and for the local statistics (which do not depend on the carrier probability), the column names for the j^{th} cRV are `globaltrans_`j, `localLR_`j, `RVS_`j and `modRVS_`j, respectively.
  - If there are fewer than three cRVs, the empty slots for the ranks are encoded as `NA`.

A.2 Additional ranking results

- Here we show the numerical summaries for the smallest (`r1`), second-smallest (`r2`) and third-smallest (`r3`) normalized ranking for the cRVs.

```
# summary(ravg)
summary(r1)
```

```
##    globalLR1           globalLR2           globalLR3           globalLR4       
##  Min.   :0.008333   Min.   :0.008333   Min.   :0.008333   Min.   :0.008333  
##  1st Qu.:0.011765   1st Qu.:0.011765   1st Qu.:0.011765   1st Qu.:0.011765  
##  Median :0.013889   Median :0.013889   Median :0.013889   Median :0.013889  
##  Mean   :0.023836   Mean   :0.023825   Mean   :0.023820   Mean   :0.023837  
##  3rd Qu.:0.029092   3rd Qu.:0.029092   3rd Qu.:0.029092   3rd Qu.:0.029092  
##  Max.   :0.191176   Max.   :0.191176   Max.   :0.191176   Max.   :0.191176  
##    globalLR5         globaltrans          localLR              RVS
##  Min.   :0.008333   Min.   :0.008333   Min.   :0.008333   Min.   :0.008065
##  1st Qu.:0.011765   1st Qu.:0.011765   1st Qu.:0.011765   1st Qu.:0.011905
##  Median :0.013889   Median :0.014085   Median :0.014085   Median :0.015152
##  Mean   :0.023805   Mean   :0.024174   Mean   :0.024401   Mean   :0.027525
##  3rd Qu.:0.028986   3rd Qu.:0.030076   3rd Qu.:0.030769   3rd Qu.:0.036613
##  Max.   :0.161765   Max.   :0.191176   Max.   :0.176471   Max.   :0.192771
##      modRVS
##  Min.   :0.008065
##  1st Qu.:0.011765
##  Median :0.014085
##  Mean   :0.024980
##  3rd Qu.:0.031332
##  Max.   :0.164384
```

```
summary(r2)
```

```
##    globalLR1         globalLR2         globalLR3         globalLR4
##  Min.   :0.01724   Min.   :0.01724   Min.   :0.01724   Min.   :0.01724
##  1st Qu.:0.03642   1st Qu.:0.03636   1st Qu.:0.03636   1st Qu.:0.03642
##  Median :0.05714   Median :0.05714   Median :0.05714   Median :0.05714
##  Mean   :0.06984   Mean   :0.06982   Mean   :0.06980   Mean   :0.06977
##  3rd Qu.:0.08399   3rd Qu.:0.08399   3rd Qu.:0.08421   3rd Qu.:0.08421
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.00000
##  NA's   :34        NA's   :34        NA's   :34        NA's   :34
##    globalLR5        globaltrans          localLR             RVS
##  Min.   :0.01724   Min.   :0.01724   Min.   :0.01724   Min.   :0.01724
##  1st Qu.:0.03620   1st Qu.:0.03659   1st Qu.:0.03614   1st Qu.:0.04444
##  Median :0.05682   Median :0.05797   Median :0.05698   Median :0.06944
##  Mean   :0.06972   Mean   :0.07196   Mean   :0.06958   Mean   :0.08514
##  3rd Qu.:0.08431   3rd Qu.:0.08696   3rd Qu.:0.08333   3rd Qu.:0.10526
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.00000
##  NA's   :34        NA's   :34        NA's   :34        NA's   :34
##      modRVS
##  Min.   :0.01724
##  1st Qu.:0.04008
##  Median :0.06410
##  Mean   :0.07361
##  3rd Qu.:0.09459
##  Max.   :0.39189
##  NA's   :34
```

```
summary(r3)
```

```
##    globalLR1        globalLR2        globalLR3        globalLR4
##  Min.   :0.0265   Min.   :0.0265   Min.   :0.0265   Min.   :0.0265
##  1st Qu.:0.0778   1st Qu.:0.0778   1st Qu.:0.0779   1st Qu.:0.0778
##  Median :0.1111   Median :0.1111   Median :0.1111   Median :0.1111
##  Mean   :0.1725   Mean   :0.1726   Mean   :0.1727   Mean   :0.1726
##  3rd Qu.:0.1549   3rd Qu.:0.1549   3rd Qu.:0.1549   3rd Qu.:0.1547
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##  NA's   :646      NA's   :646      NA's   :646      NA's   :646
##    globalLR5       globaltrans        localLR            RVS
##  Min.   :0.0265   Min.   :0.0265   Min.   :0.0265   Min.   :0.0288
##  1st Qu.:0.0778   1st Qu.:0.0780   1st Qu.:0.0769   1st Qu.:0.1011
##  Median :0.1111   Median :0.1111   Median :0.1073   Median :0.1462
##  Mean   :0.1729   Mean   :0.1801   Mean   :0.1723   Mean   :0.1924
##  3rd Qu.:0.1549   3rd Qu.:0.1573   3rd Qu.:0.1514   3rd Qu.:0.2055
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##  NA's   :646      NA's   :646      NA's   :646      NA's   :646
##      modRVS
##  Min.   :0.0265
##  1st Qu.:0.0909
##  Median :0.1279
##  Mean   :0.1430
##  3rd Qu.:0.1806
##  Max.   :0.4875
##  NA's   :646
```