Code ▾

Problem Set 4, Solutions

Stats 506, Fall 2018

Due: Monday December 10, 5pm

Instructions

- Submit the assignment by the due date via canvas. There is a maximum of 1 late day for this assignment.
- Use Rmarkdown to create and submit a single html or pdf with your answers to question 1-2 along with supporting evidence in the form of tables and graphs.
- All tables and graphs should be neatly labeled and appear polished.
- Question 1 and 2 ask you to use *R*. You should submit your code for each problem as `ps4_q1.R` and `ps4_q2_X.R` .
- You should submit a single compressed archive (`.zip`) which contains the following files:
  - `ps4.pdf` or `ps4.html`
  - `ps4.Rmd`
  - `ps4_q1.R`
  - `ps4_q2_funcs.R` , `ps4_q2a.R` , `ps4_q2b.R` , `ps4_q2c.R`
  - `run_ps4_q2b.pbs` , `run_ps4_q2c.pbs`
  - `ps4_q2b.Rout` , `ps4_q2c-X.Rout` (X = 1, 2, 4).
  - `ps4_q3.sas` , `ps4_q3c.csv` , `ps4_q3d.csv`

  All files should be executable without errors.
- All files read, sourced, or referred to within scripts should be assumed to be in the same working directory (`./`).
- Your code should be clearly written and it should be possible to assess it by reading it. Use appropriate variable names and comments. Your style will be graded using the style rubric (./StyleRubric.html) [15 points].
- Some of these exercises may require you to use commands or techniques that were not covered in class or in the course notes. You can use the web as needed to identify appropriate approaches. Part of the purpose of these exercises is for you to learn to be resourceful and self sufficient. Questions are welcome at all times, but please make an attempt to locate relevant information yourself first.

Question 1 [20 points]

Use the Lahman baseball data previously seen in the SQL notes (https://jbhender.github.io/Stats506/F18/Intro_to_SQL.html) to answer this question. Your answer should be a single SQL query, but may require anonymous tables created using nested queries.

Write an SQL query to construct a table showing the *all-time* leader in *hits* ("H" from the "batting" table) for each birth country ("birthCountry" in the "master" table). An *all-time* leader is the player ("playerID") with the most total hits across all rows (e.g. seasons/stints). Limit your table to players/countries with at least 200 hits and order the table by descending number of hits. Create a nicely formatted table with the following columns as your final output: Player (nameFirst nameLast), Debut (debut), Country of Birth (birthCountry), Hits (H).

Solution: Here is the query. For the full R script, see `ps4_q1.R` at the Stats506_F18 git repo.

Code

Below is a formatted table.

Code

Player	Debut	Country of Birth	Hits
Pete Rose	Apr 08, 1963	USA	4,256
Rod Carew	Apr 11, 1967	Panama	3,053
Ichiro Suzuki	Apr 02, 2001	Japan	3,030
Rafael Palmeiro	Sep 08, 1986	Cuba	3,020
Roberto Clemente	Apr 17, 1955	P.R.	3,000
Adrian Beltre	Jun 24, 1998	D.R.	2,942
Omar Vizquel	Apr 03, 1989	Venezuela	2,877
Chili Davis	Apr 10, 1981	Jamaica	2,380
Edgar Renteria	May 10, 1996	Colombia	2,327
Patsy Donovan	Apr 19, 1890	Ireland	2,253
Larry Walker	Aug 16, 1989	CAN	2,160
Tom Brown	Jul 06, 1882	United Kingdom	1,951
Andruw Jones	Aug 15, 1996	Curacao	1,933
Vinny Castilla	Sep 01, 1991	Mexico	1,884
John Anderson	Sep 08, 1894	Norway	1,841
Joe Quinn	Apr 26, 1884	Australia	1,797
Elmer Valo	Sep 22, 1940	Czech Republic	1,420
Horace Clarke	May 13, 1965	V.I.	1,230
Shin-Soo Choo	Apr 21, 2005	South Korea	1,206
Glenn Hubbard	Jul 14, 1978	Germany	1,084
Marvin Benard	Sep 05, 1995	Nicaragua	714
Eddie Ainsmith	Aug 09, 1910	Russia	707
Andre Rodgers	Apr 16, 1957	Bahamas	628
Xander Bogaerts	Aug 20, 2013	Aruba	528
Didi Gregorius	Sep 05, 2012	Netherlands	451

Player	Debut	Country of Birth	Hits
Gerald Young	Jul 08, 1987	Honduras	446
Reno Bertoia	Sep 22, 1953	Italy	425
Yan Gomes	May 17, 2012	Brazil	367
Steve Jeltz	Jul 17, 1983	France	367
Tony Solaita	Sep 16, 1968	American Samoa	336
Jack Quinn	Apr 15, 1909	Slovakia	248

Question 2 [40 points]

In this question you will modify your answer to Problem Set 3, Question 2 (PS3 Q2) to practice parallel, asynchronous, and batch computing. Copy the functions from part a and c of PS3 Q2 to a new file `ps4_q2_funcs.R`

In each of the parts below, let $\beta \in \mathbb{R}^{100}$ be defined so that

$$\beta_i = \begin{cases} .1, & i \leq 10, \\ 0, & \text{else.} \end{cases}$$

and Σ be block diagonal with $\Sigma_{ij} = \rho\beta_i\beta_j$ when $i \neq j$ and $\Sigma_{ii} = 1$. (You may also use β as in PS3 Q2 and rescale in any other way that results in a positive definite Σ.)

Create a table or plot for your results from each part.

a. Write an R script `ps4_q2a.R` that sources `ps4_q2_funcs.R`, and then uses `mclapply` to run parallel simulations for $\rho \in \{.25i\}_{i=-3}^{3}$.
   Let $\sigma = 1$ and use 10,000 Monte Carlo replications. Reorganize the results into a long data frame `results_q4a` with columns: “rho”, “sigma”, “metric”, “method”, “est”, and “se”. “Metric” should contain the assessment measure: FWER, FDR, Sensitivity, or Specificity and “method” the multiple comparison method used. The columns “est” and “se” should contain the Monte Carlo estimate and its standard error, respectively.

** Solution: ** See source code at the Stats506_F18 repo on github.

Code

Code

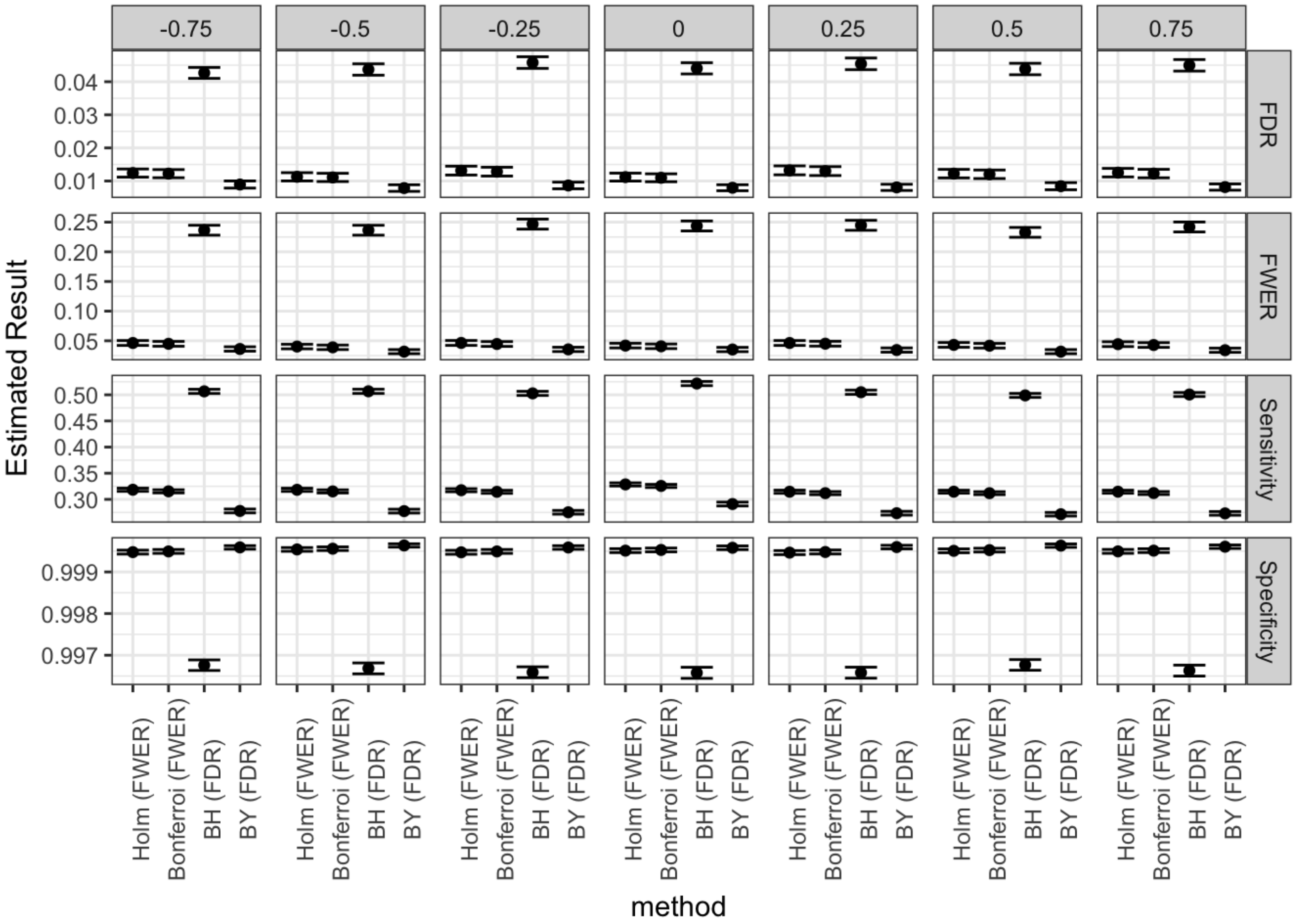

Figure 1. Performance measures for variable selection in a linear model from multiple hypothesis testing using four methods for controlling the family-wise error rate (FWER) or false discovery rate (FDR). Each interval shows the mean and 95% confidence interval computed from 10,000 Monte Carlo draws of the response, but conditional on a single draw of the covariate matrix. Because of this conditioning, results are comparable within columns but not between columns. Each column corresponds to a different model matrix in which the true-positive covariates all have pairwise correlation $ ho imes 1e-3$ with $ ho$ given in the column header.

b. Use your script from part a as the basis for a new script `ps4_q2b.R` . Setup a 4 core cluster using `doParallel` and then use nested foreach loops to run simulations for $\rho \in \{.25i\}_{i=-3}^{3}$ and $\sigma = \{.25, .5, 1\}$. Reshape the results as before into `results_q4b` saved to a file `results_q4b.RData` . Use a PBS file to run this script on the Flux cluster.

Solution:

Code

Code

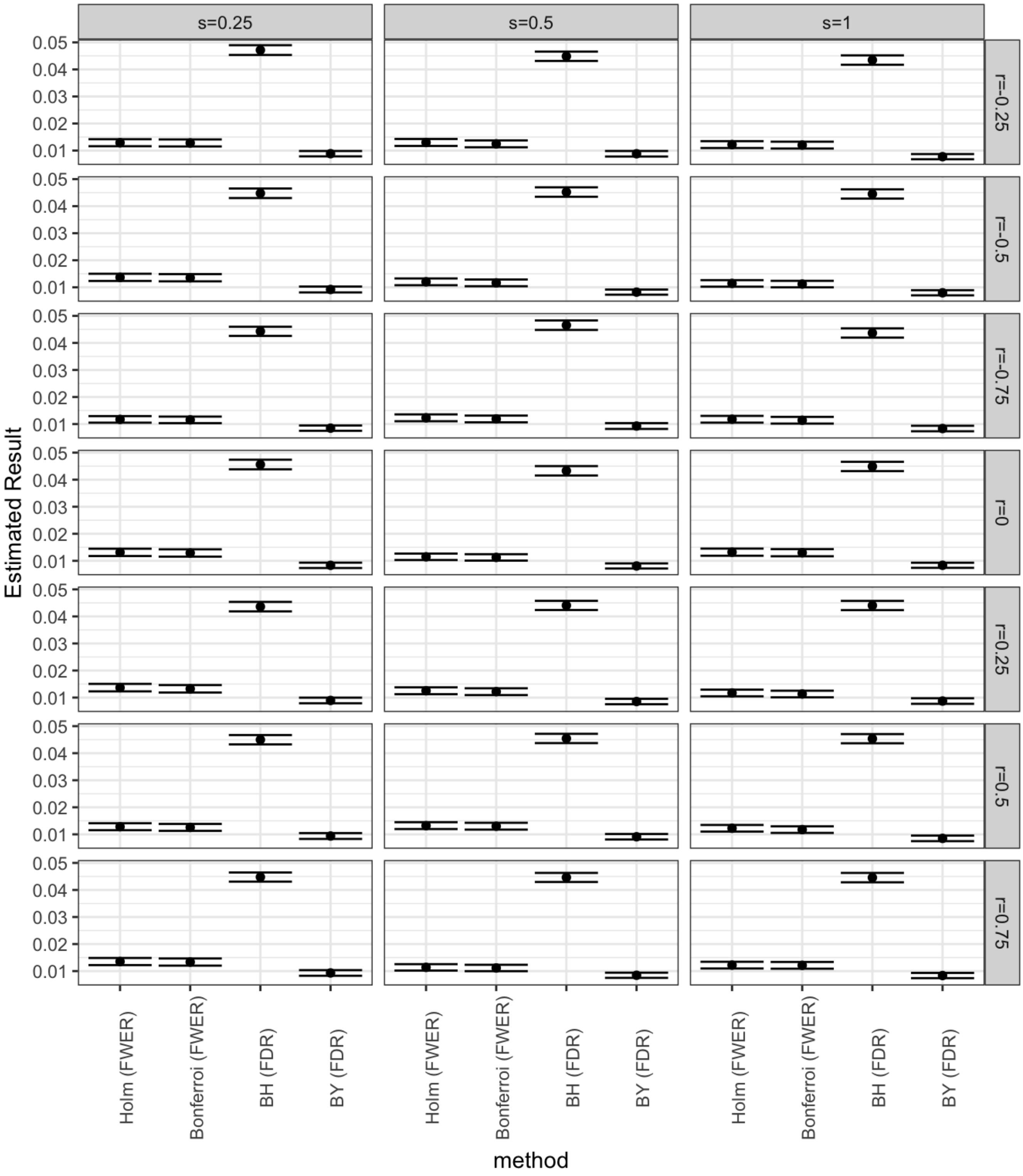

Figure 2a. *False discovery rates.*

Code

Code

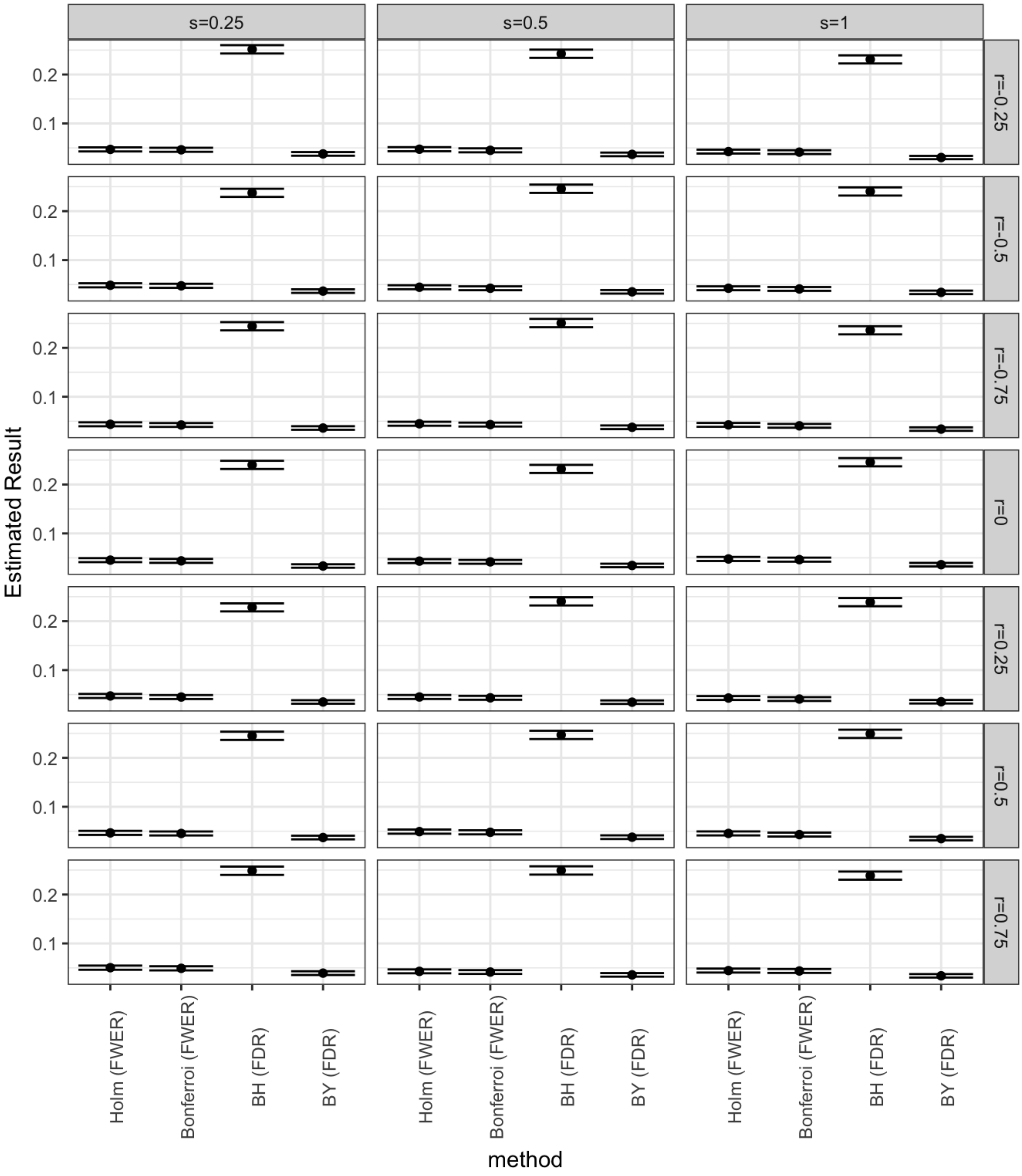

Figure 2b. *Family wise error rates.*

Code

Code

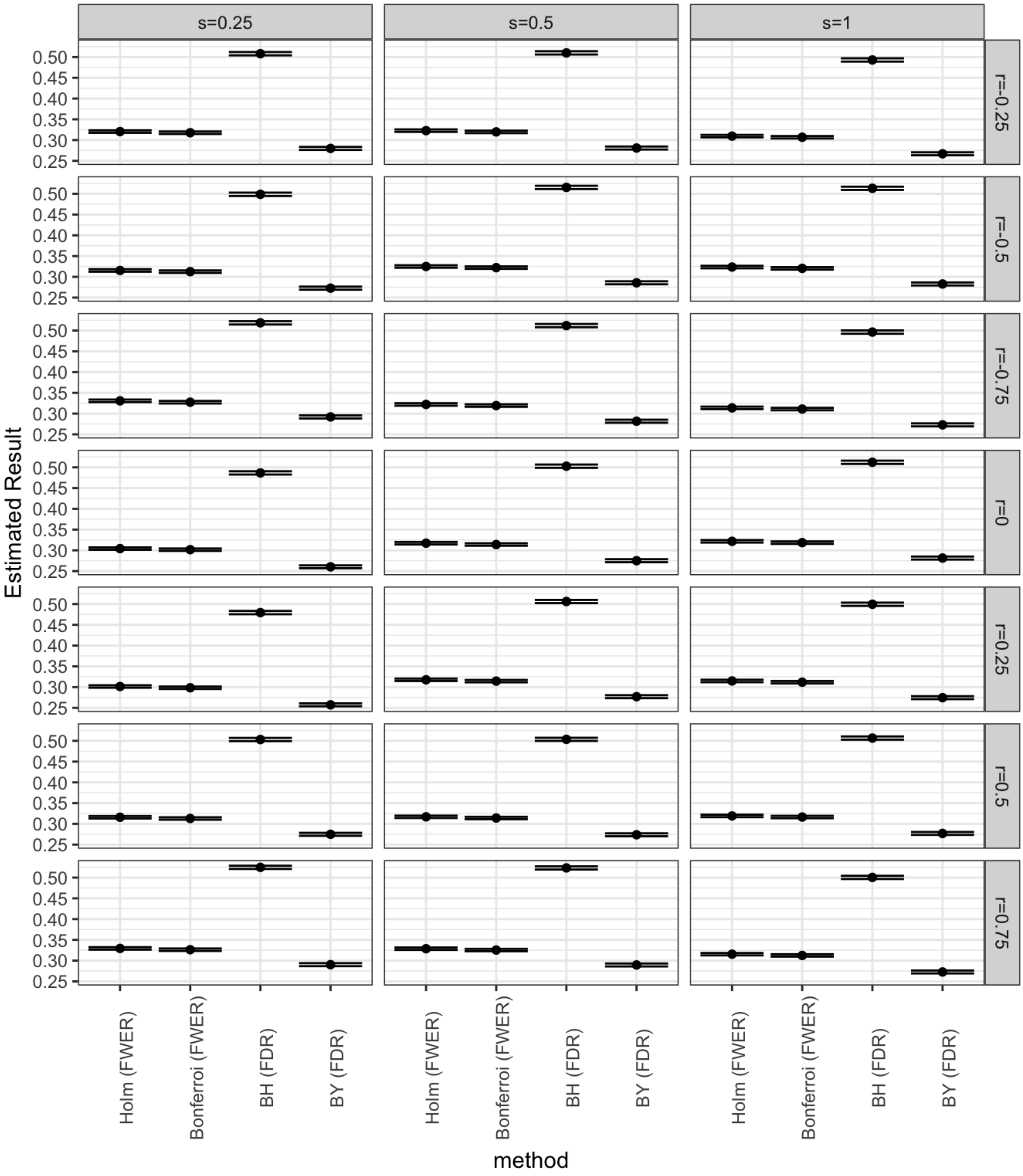

Figure 2c. *Sensitivity.*

Code

Code

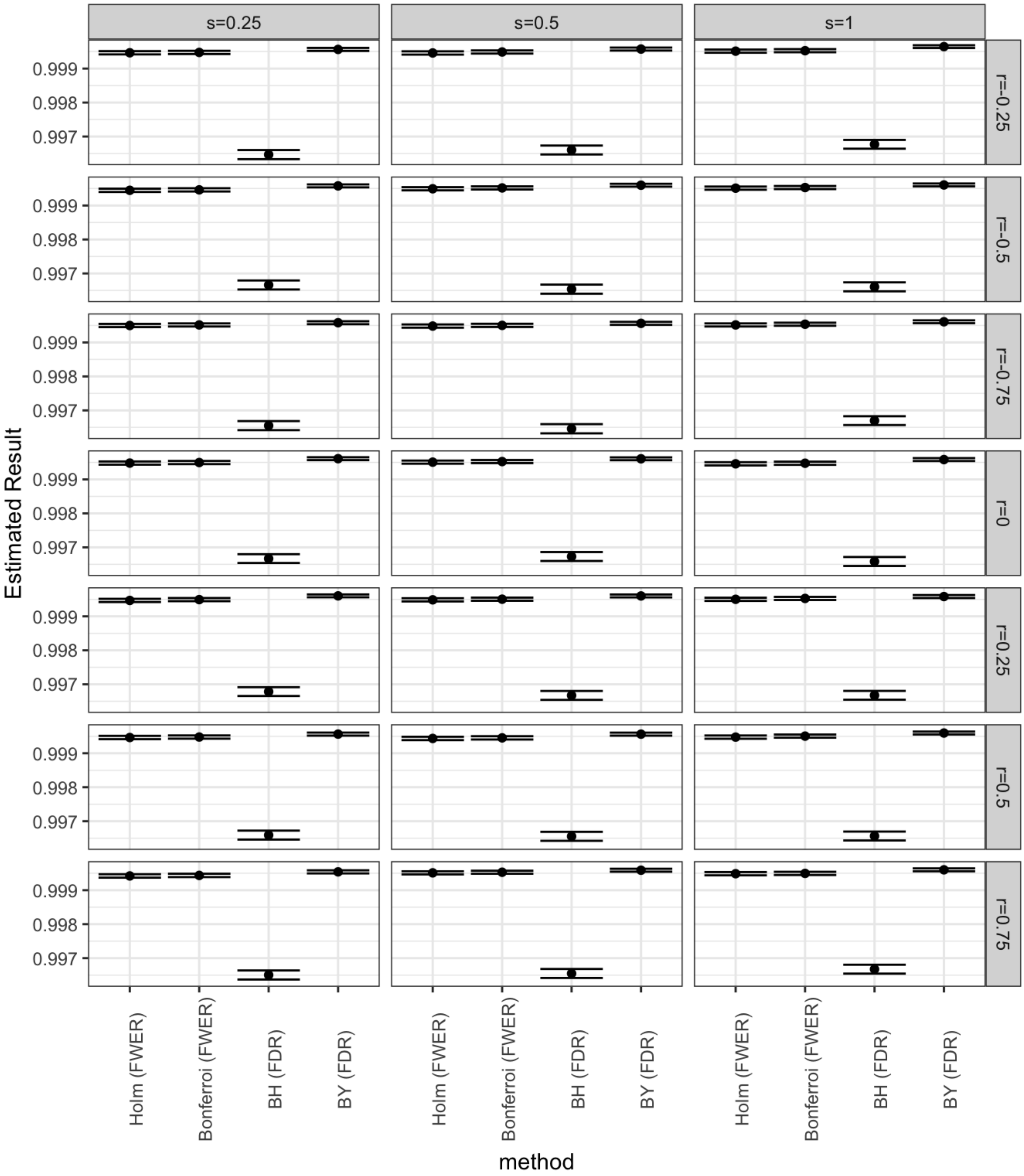

Figure 2d. *Specificity.*

Code

c. Modify your script from part a to create `ps4_q2c.R` which reads the following arguments from the command line: `sigma` , `mc_rep` , and `n_cores` . Also modify the script to use the `futures` package for parallelism. Use a PBS file to run this script as a job array for $\sigma = \{.25, .5, 1\}$. _Hint: see the answer at this page (https://stackoverflow.com/questions/12722095/how-do-i-use-floating-point-division-in-bash) for how to convert `$PBS_ARRAYID` to `sigma` .

Solution: See script at Stats506_F18 repo. Results should be similar to part b.

Question 3 [25 points]

For this question you should use the 2016 Medicare Provider Utilization and Payment data available here here (https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Provider-Charge-Data/Physician-and-Other-Supplier.html).

a. Put the data into a folder `./data` and then follow the instructions to read this data into SAS.

b. Use one or more data steps to reduce the data set to those rows with “MRI” in the ‘hcpcs_description’ field and where ‘hcpcs_code’ starts with a 7.

c. Use proc means or proc summary (as needed) to determine the MRI procedures with the highest volume, highest total payment, and highest average payment among the procedures represented here.

d. Repeat part b-c using PROC SQL.

e. Export the results from “c” and “d” to csv and verify that they match. You do *not* need to produce a nice table within your solution document.

Solution: See `ps4_q3_import.sas` for the solution to part a and `ps4_q3.sas` for parts b-e. These can be found at the Stats506_F18 git repo. Also available there are the results exported as csv files.