Analyzing Tony Gitter's chemical data set

Available on https://zenodo.org/record/1411506#.X7w4gs1KhPY

Data set

- n_train = 72,423 training samples
- p = 1024
- Response = % inhibition
- Predictors = binary vectors of Morgan fingerprints
- Both predictors and response are standardized.

Training

- *blasso* (‘monomvn’ package) just froze and crashed.
- *basad* (‘basad’ package) works.
- 50,000 MCMC samples + 50,000 burn-in take ~ 3 hours+

- *glmnet* (1 for common penalty, 1 for different penalty factor) is time-consuming.
- 2,000 iterations on 7 parallel CPU cores take ~ 11hours.

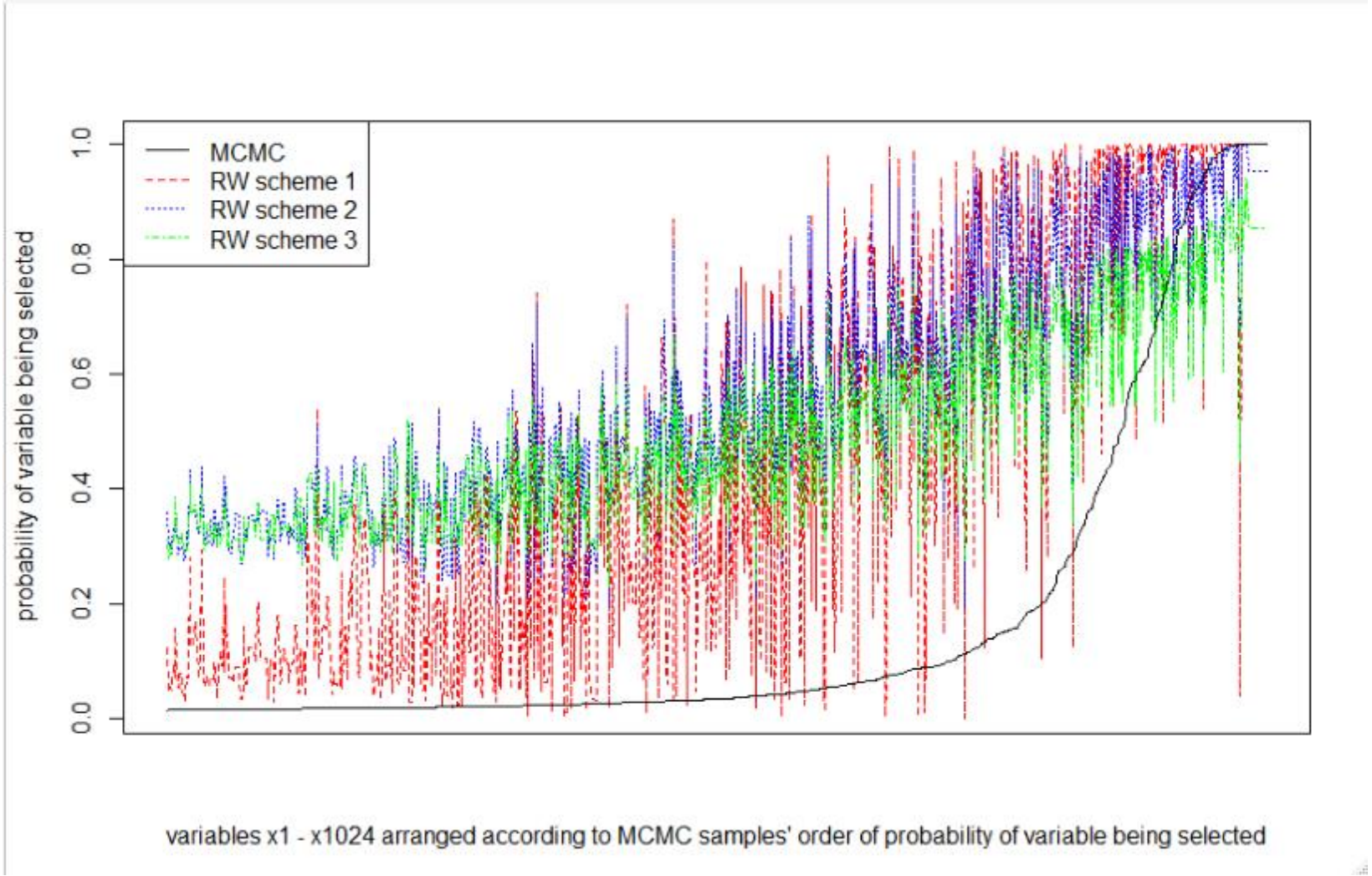
- Randomly pick 2,000 MCMC samples from the last 10,000 MCMC samples.

Training

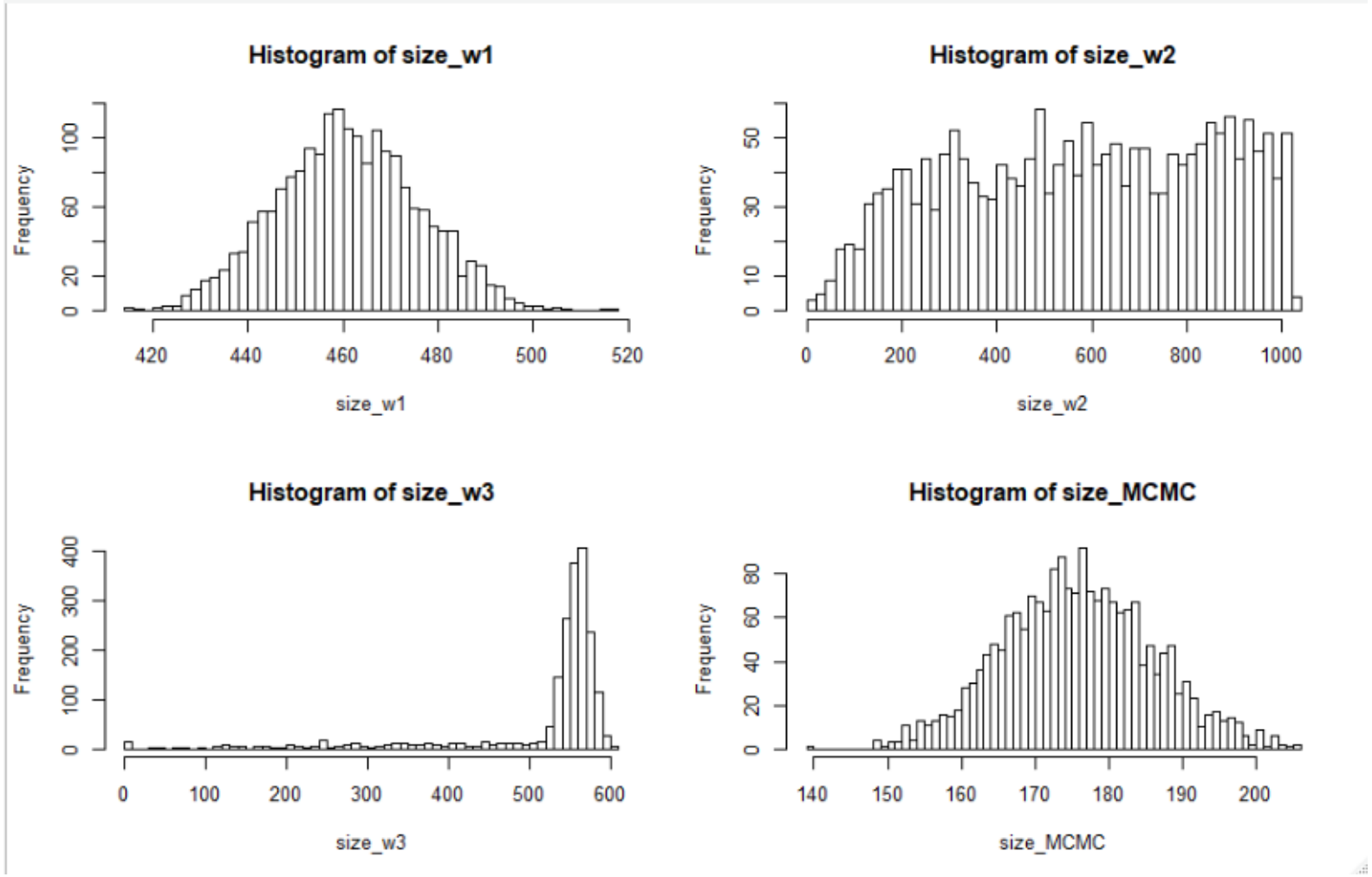
- *basad*
  - Use default package setting
  - Set corresponding MCMC β samples to be zero where median probability model dictates latent variable Z = 0, to induce sparsity in MCMC β samples

- *glmnet*
  - *cv.glmnet* gives lambda.min = 0.001 and lambda.1se = 0.006
  - Try lambda.1se to have slightly larger penalization

Compare probability of variables selected

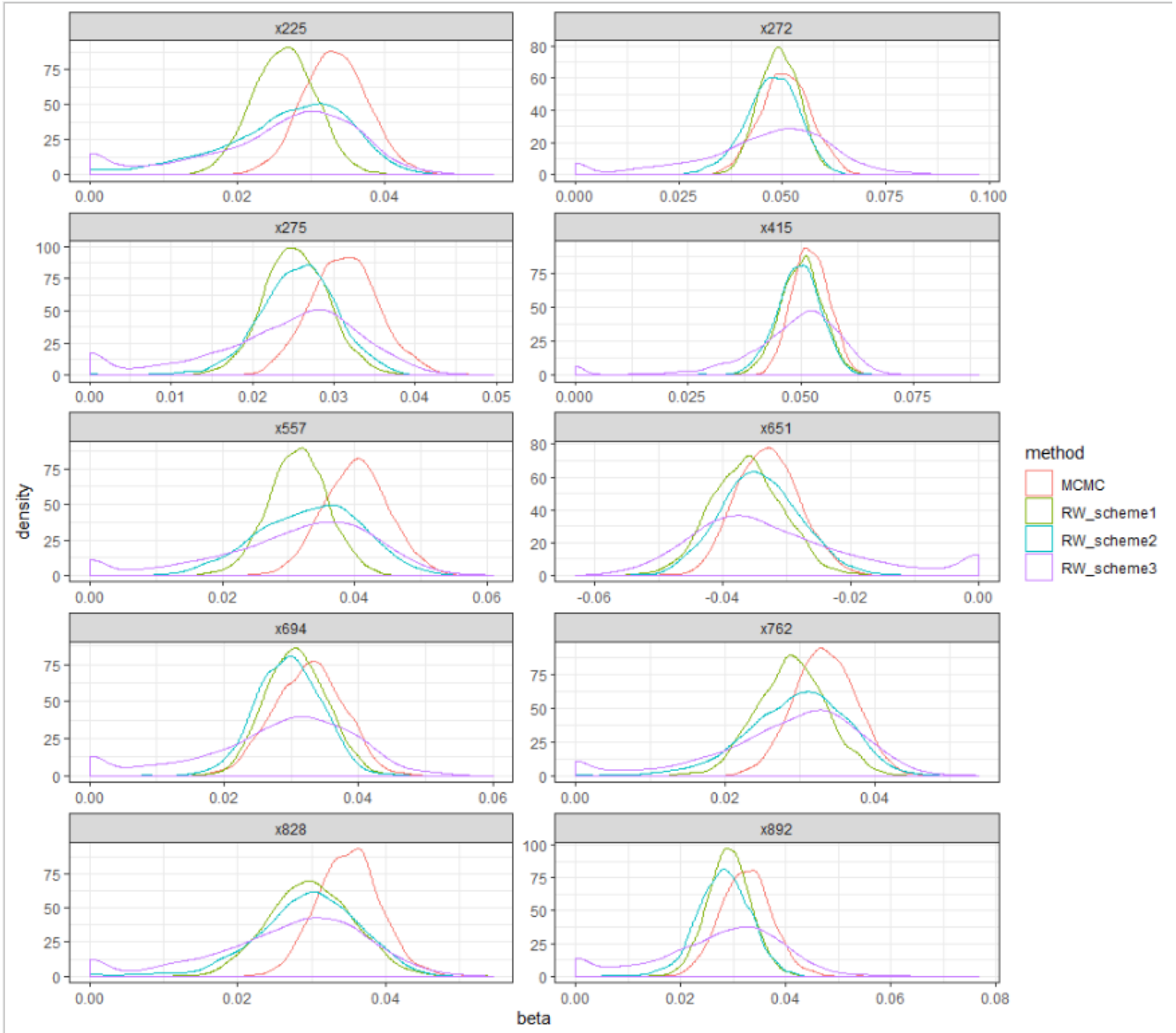


Compare model sizes

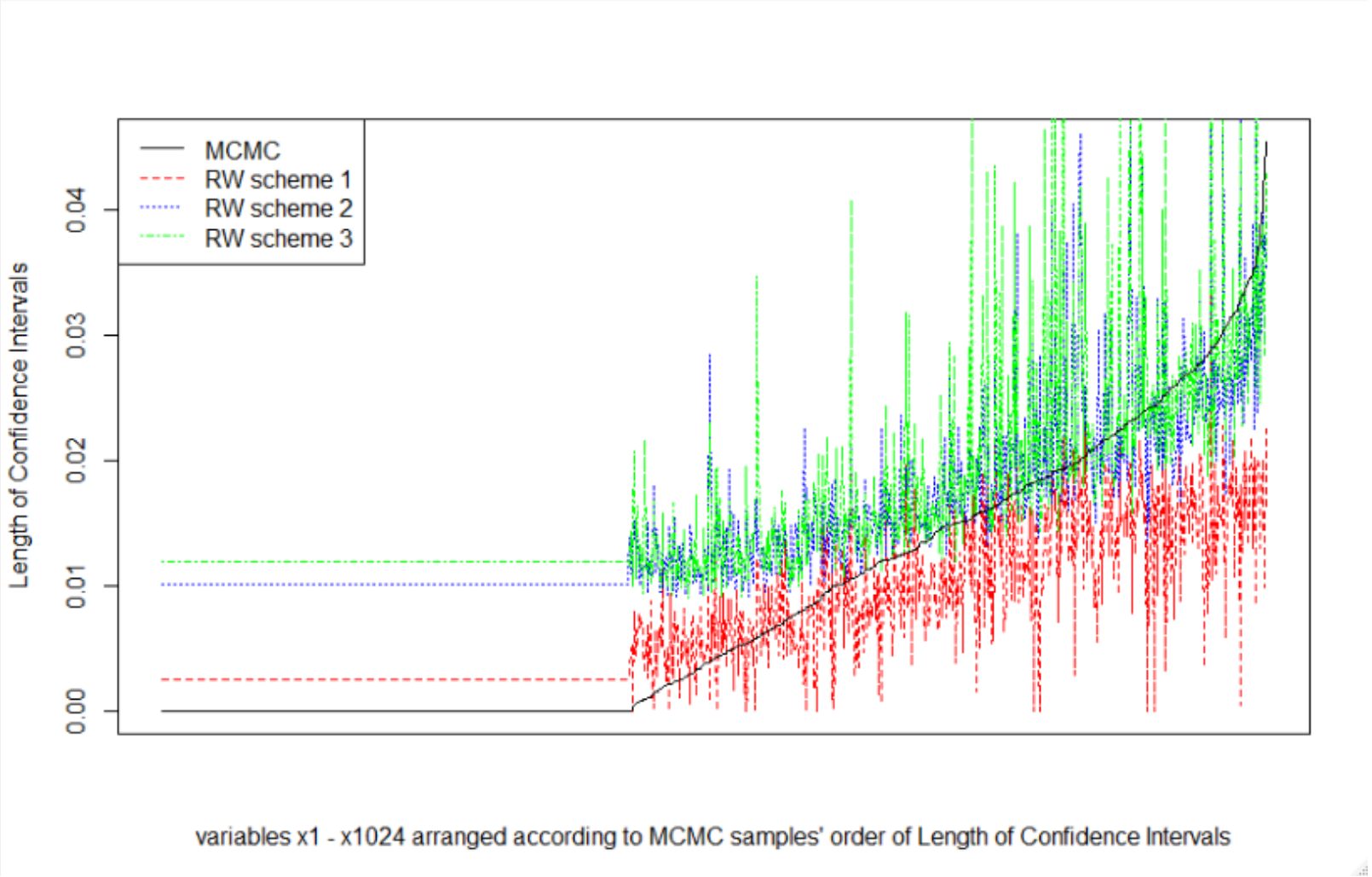


Compare marginal posterior distributions of β's

- Pick variables whose probability of being selected is larger than 0.93 for all *basad*'s MCMC and RW schemes 1, 2 and 3.
- There are 10 of them.
- Plot marginal distributions of their sampled β's.


x225
x272
x275
x415
x557
x651
x694
x762
x828
x892
density
beta
method
MCMC
RW_scheme1
RW_scheme2
RW_scheme3

Compare length of 95% CI for β's



Predictions

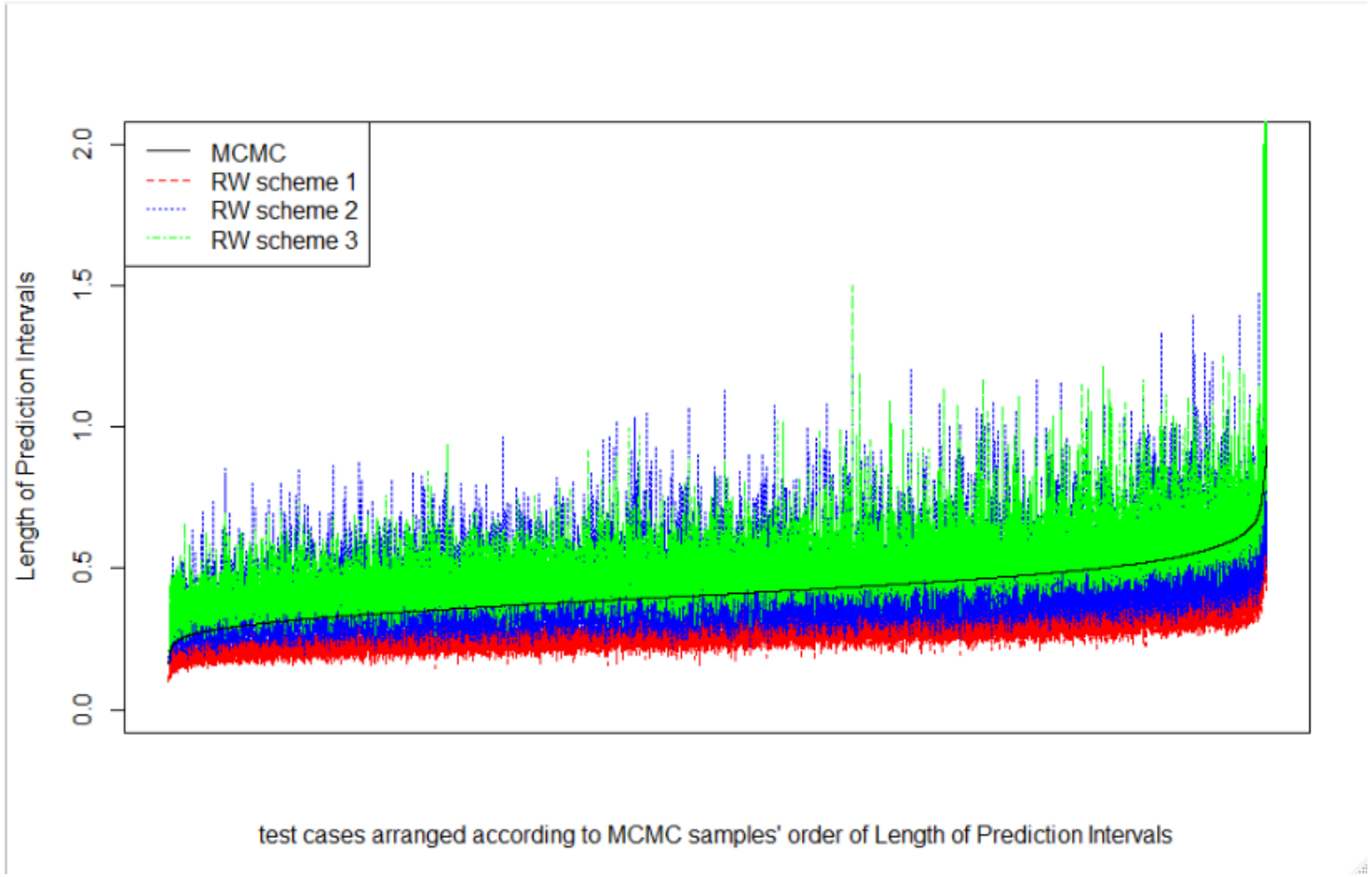
- n_test = 22,423
- For predictions, take sampled β's from MCMC or RW, and multiply with test cases' predictors.
- Each test case has a distribution of predictions.

95% prediction interval coverage for the ~22,000 test cases is similarly poor for all schemes

- RW scheme 1 = 12.5%
- RW scheme 2 = 19.3%
- RW scheme 3 = 23.0%
- MCMC = 19.2% (based on sparsity-induced β's)

- Because length of prediction intervals for RW schemes 2 and 3 are relatively wider

Compare 95% prediction interval length



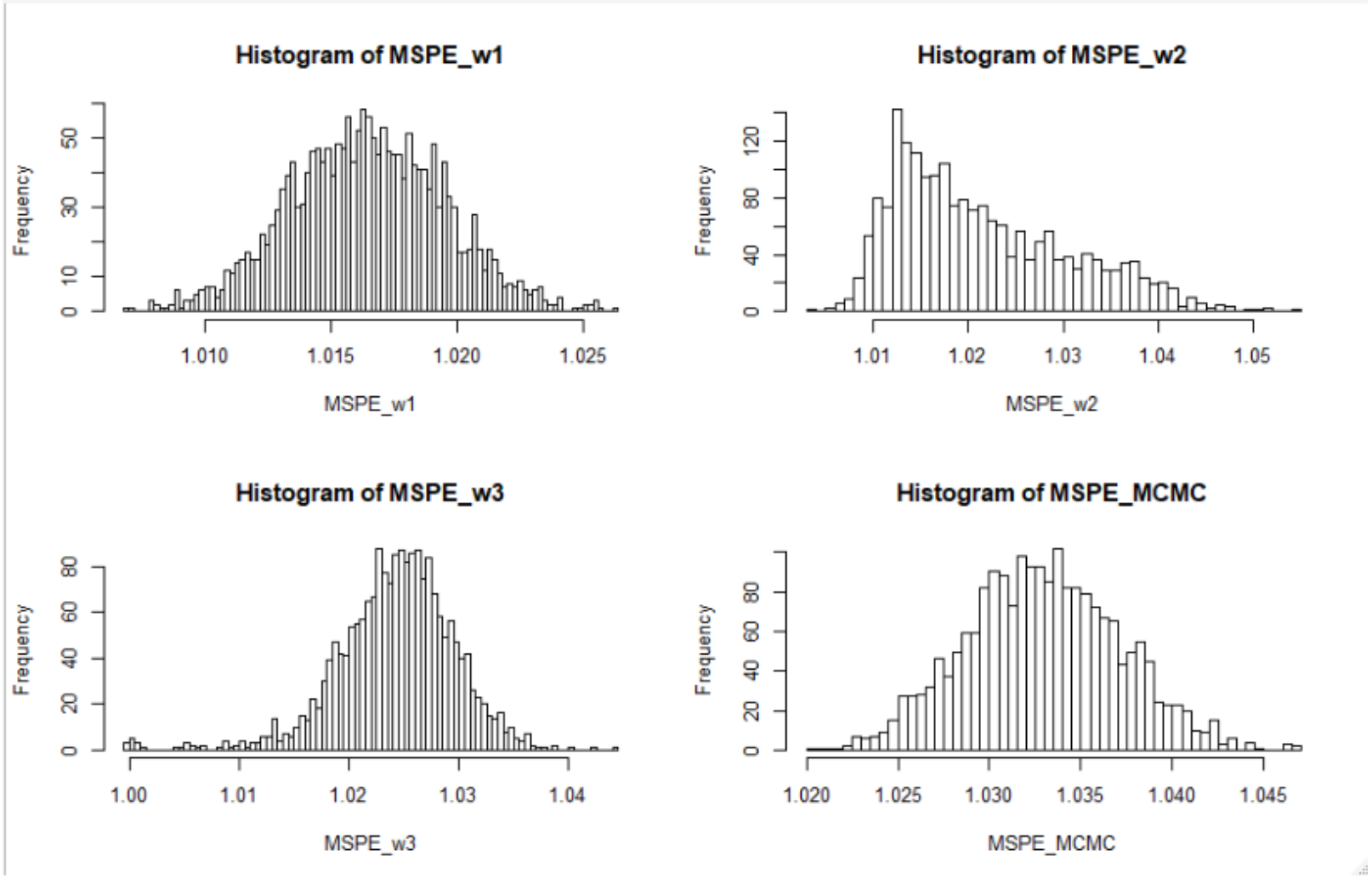
Prediction

- I considered 2 types of MSPE calculations

    1. For each of B=2,000 iterations, compute MSPE (squared prediction errors averaged across all 22,423 test cases)

    2. For each of n_test = 22,423 test cases, compute MSPE (squared prediction errors averaged across all 2,000 predictions)

Distribution of 2,000 MSPEs (averaged across all test cases)



Compare distribution of MSPE for each test case