

7 Supplemental Materials

In addition to the graphical design of the leave-half-out cross validation design presented in Figure 3, we created pseudocode that describes the process.

Within- and between-batch cross-validation

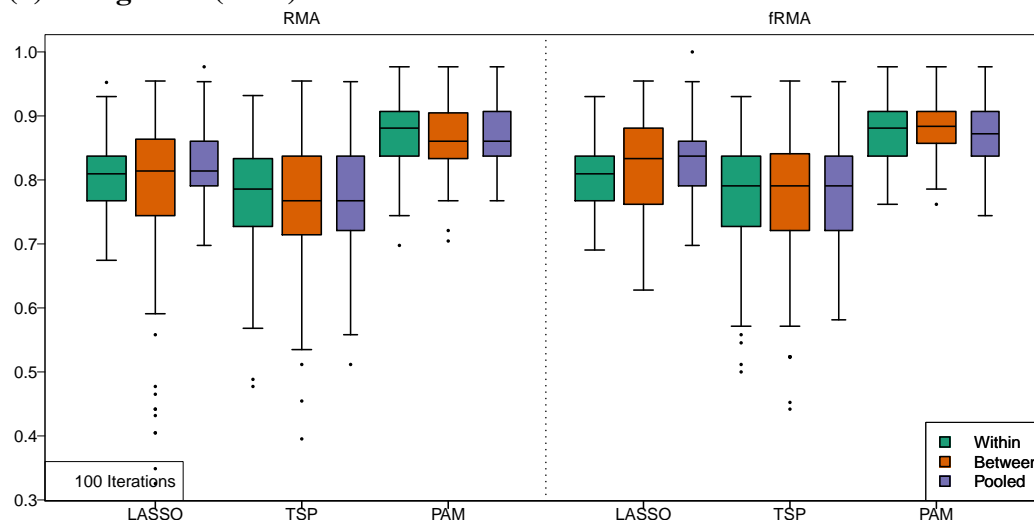
1. For the batch with the smaller sample size (batch A), randomly choose half of the arrays from outcome 1 (n_1) and half of the arrays from outcome 2 (n_2). This will be the training data for batch A.
2. For the batch with the larger sample size (batch B), randomly chose n_1 arrays from outcome 1, and n_2 arrays from outcome 2, where n_1 and n_2 are defined above. This will be the training data for batch B.
3. Preprocess the training data for batch A and batch B separately using the preprocessing algorithm of your choosing.
4. Build a predictive model for each of the batches, thus creating model A and model B.
5. Create test data for batch A by randomly choosing n_1 arrays from outcome 1 (that were not chosen in the training data) and n_2 arrays from outcome 2. Note that if batch A has an even number of arrays for each outcome, then the testing data will just be the remaining arrays that were not in the training set.
6. Create a test data for batch B by randomly choosing n_1 arrays from outcome 1 (that were not chosen in the training data) and n_2 arrays from outcome 2.
7. *For within-batch cross-validation:* Predict the outcomes for test data A using model A, and for test data B using model B.
8. *For between-batch cross-validation:* Predict the outcomes for test data A using model B, and for test data B using model A.
9. Repeat 100 times to obtain robust accuracy rates.

Pooled batch cross-validation

1. Do not subdivide data into batches. Randomly choose half of the arrays from outcome 1 (N_1) and half of the arrays from outcome 2 (N_2). This will be the training data.
2. Preprocess the training data.
3. Build a predictive model using the training data.
4. Create test data by randomly choosing N_1 arrays from outcome 1 and N_2 arrays from outcome 2. Note that if the data has an even number of arrays for each outcome, then the testing data will just be the remaining arrays that were not in the training set.

5. Predict the outcomes for the test data using the model built on the training data.
6. Repeat 100 times to obtain robust accuracy rates.

(a) Wang et al. (2005)



(b) Minn et al. (2005)

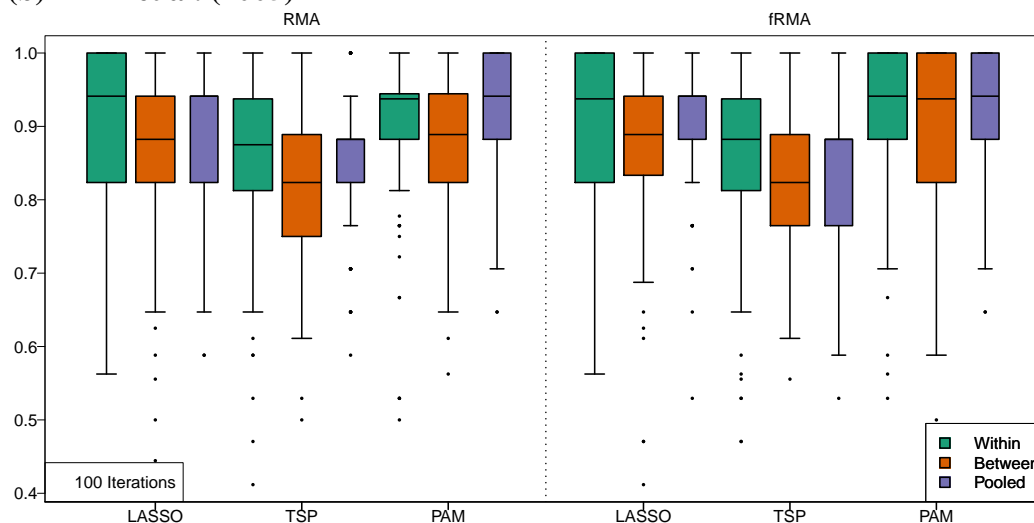


Figure 9: **Predicting between batches increases variance and decreases accuracy.** Boxplots of 100 cross-validated prediction accuracy rates, as found using the cross-validation design described in the paper are shown for a) Wang et al. (2005) and b) Minn et al. (2005) datasets. Prediction accuracy was measured within batches, between batches, and pooling batches, in order to assess the role that batch plays in prediction. Data were preprocessed with two commonly-used preprocessing methods - RMA and fRMA - in order to see the affect of preprocessing on batch. We see that in general, the variance of accuracy rates increases when building the model on one batch and testing it on another (between-batch).

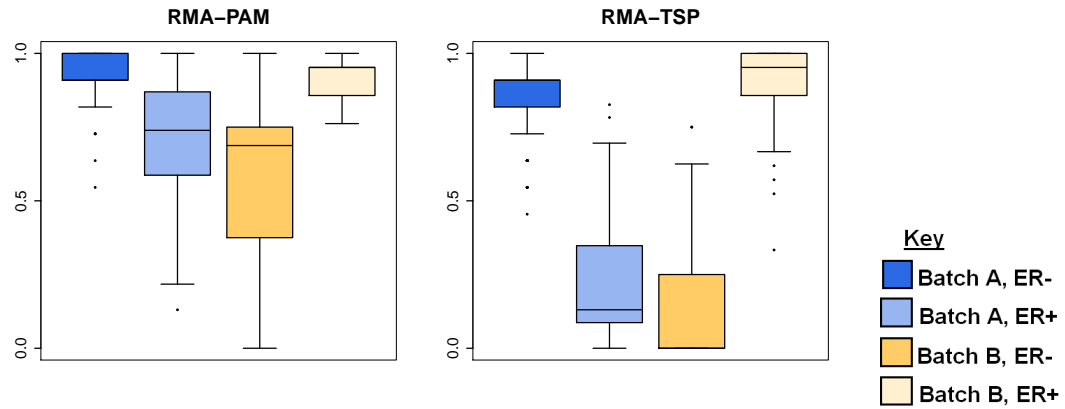


Figure 10: **Prediction accuracy rates for perfect confounding simulated design show that fRMA-PAM combination is optimal.** The study design is presented above (Figure 3), and prediction accuracy rates are shown as boxplots for the accuracy measurements taken from the 100 iterations. Results are shown only for RMA-preprocessing here. fRMA-preprocessing results are shown in Figure 4.

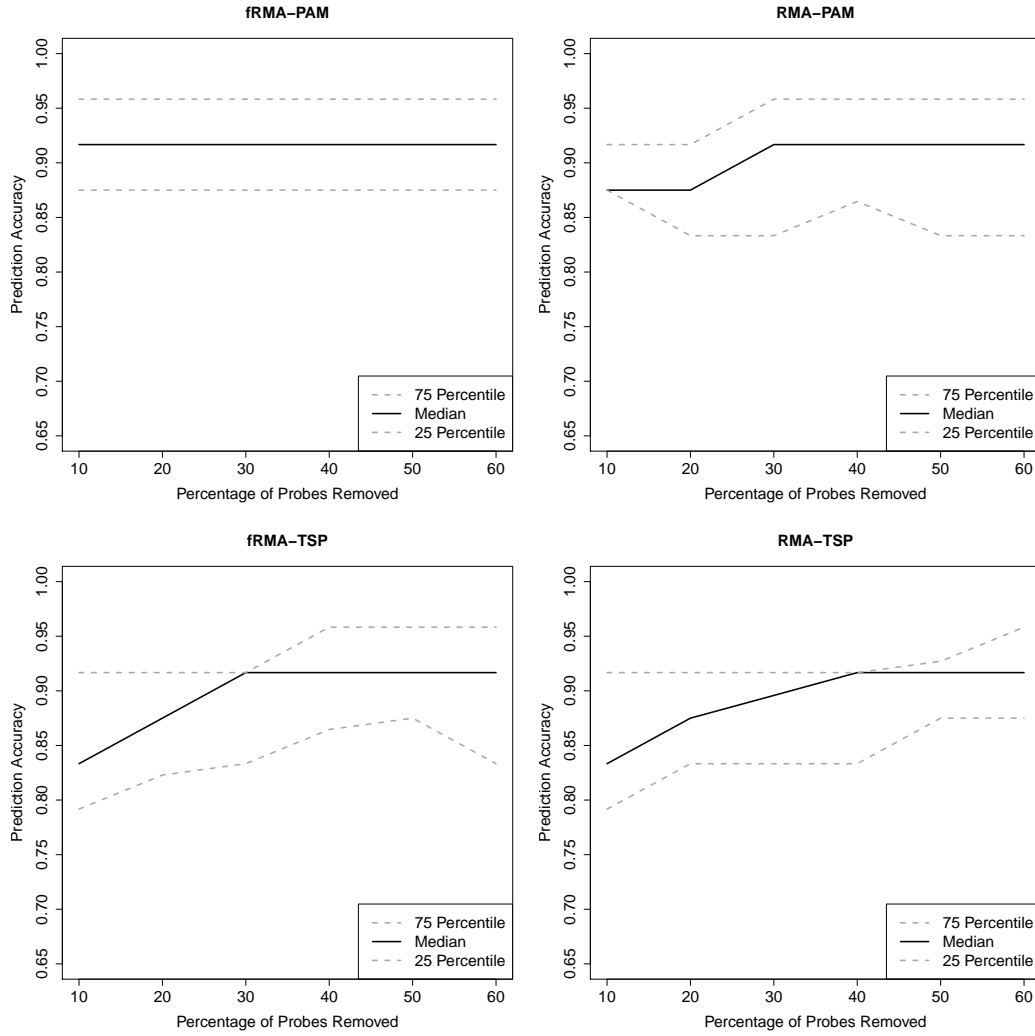


Figure 11: **Prediction accuracy only somewhat improves as batch-affected probes are removed.** Batch-affected probes in the Minn et al. (2005) dataset were determined by fitting model 1 and selecting probes with the most significant β_1 estimates.

Wang et al. (2005)

Percent Batch-Affected	RMA	fRMA
10%	0.08 (0.08, 0.08)	0.08 (0.08, 0.08)
20%	0.16 (0.16, 0.17)	0.17 (0.16, 0.17)
30%	0.25 (0.25, 0.26)	0.25 (0.25, 0.25)
40%	0.34 (0.34, 0.35)	0.34 (0.33, 0.34)
50%	0.43 (0.43, 0.44)	0.43 (0.42, 0.43)
60%	0.53 (0.52, 0.53)	0.52 (0.52, 0.53)

Minn et al. (2005)

10%	0.07 (0.07, 0.08)	0.07 (0.07, 0.07)
20%	0.15 (0.15, 0.15)	0.15 (0.15, 0.15)
30%	0.23 (0.23, 0.24)	0.23 (0.23, 0.23)
40%	0.31 (0.31, 0.32)	0.31 (0.31, 0.32)
50%	0.40 (0.39, 0.40)	0.40 (0.39, 0.40)
60%	0.48 (0.48, 0.49)	0.49 (0.48, 0.49)

Table 3: **Proportion overlap of batch-affected probes in pairwise comparisons of the 100 iterations shows little consistency.** 100 sets of batch-affected probes were generated from model 1, based on the simulated design described in figure 3. We then made pairwise comparisons of the proportion overlap of batch-affected probes, and report the median (25th percentile, 75th percentile).