Investigating RandNLA Methods for Schatten 2p-norm and Trace Estimation

Lucía Vilallonga

ABSTRACT

This report analyzes through simulation the trace and Schatten 2p-norm estimation methods outlined in Martinsson & Tropp (2021)[1], in particular sections 4.1, 4.2, 4.8 through 5.2, and 5.4. Code and proofs for each experiment can be found in the corresponding GitHub repositories[2]. The goals of each simulation are (1) to show that the sampling methods presented in the paper run faster than direct solutions, but still give reasonable results for the desired quantities, and (2) to determine whether there is a correlation between the size of the input matrix or the number of samples taken and the root mean square error (RMSE) of the estimates.

METHODS

To estimate the trace of a psd matrix $\mathbf{A} \in \mathbb{H}_n$, use the quantity $\overline{X}_k \overset{\text{def}}{=} \frac{1}{k}\sum_{i=1}^{k} X_i$, where X_i are iid samples and $X_i = \omega_i'(\mathbf{A}\omega_i)$ for isotropic (random) test vectors $\omega_i \in \mathbb{R}^n$ such that $\mathbb{E}[\omega_i \omega_i'] = \mathbf{I}$. Here k is the number of samples, X_i, to take. $\overline{X}_k$ is an unbiased estimate for $trace(\mathbf{A})$ (see proofs). If the matrix $\mathbf{A}$ were not stored in memory, for example, because it is too large, then the sampling method implemented in the simulation would be faster than indirectly calculating the trace with repeated matrix-vector multiplication. Without specifically creating a large matrix and testing this, the theory can be supported by inferring from how the performance of the sampling method scales as the size of $\mathbf{A}$ increases.

To estimate the Schatten 2p-norm of a matrix $\mathbf{B} \in \mathbb{R}^{mxn}$, use the quantity $V_p = C(k, p)^{-1}\, trace(\, T(\mathbf{X})^{p-1}\mathbf{X}\,)$, where $T : \mathbb{H}_k \to \mathbb{R}^{kxk}$ be the linear map that reports the strict upper triangle of a symmetric matrix, in this case $\mathbf{X} = (\mathbf{B\Omega})'(\mathbf{B\Omega}) \in \mathbb{R}^{kxk}$, where $\mathbf{\Omega} \in \mathbb{R}^{nxk}$ is a random test matrix with iid columns $\omega_i \in \mathbb{R}^n$ such that $\mathbb{E}[\omega_i \omega_i'] = \mathbf{I}$. Again, k is the number of samples to take. V_p is an unbiased estimate for $\| \mathbf{B} \|_{2p}^{2p}$ (see proofs). The direct method of finding the norm is to compute a singular value decomposition (SVD) and: $\sum_{i=1}^{min\{m,n\}} \sigma_i^{2p}$, where σ_i is the ith singular value of $\mathbf{B}$.

The R simulations for both trace and norm estimation proceed similarly. For a series of matrix sizes N ranging from 2 to 500 rows, and sample sizes K ranging from 2 to 1000, the simulation first computes the true trace, then the estimate using sampling methods, and records the runtime for the sampling method. For a matrix of fixed size 1000x2000, and a series of powers P ranging from 1 to 100 and sample sizes K from 100 to 1000, the simulation first computes the true Schatten 2p-norm, then the estimate using sampling methods, and records the runtime for both. I expect the RMSE of both sampling methods to decrease as the sample size increases, while the reverse should be true for the runtime.

RESULTS

As expected, the runtime for the trace sampling method grows with both sample size and the number of rows (Figure 1), and the effect is more pronounced for larger matrices (Figure A1). The expectation that RMSE would decrease with sample size and increase with matrix size was also correct (Figure 2).

---

[1] https://arxiv.org/abs/2002.01387v1, last accessed 4/30/21

[2] https://github.com/ghostpress/comp-stats-sims/tree/final-project/final-project

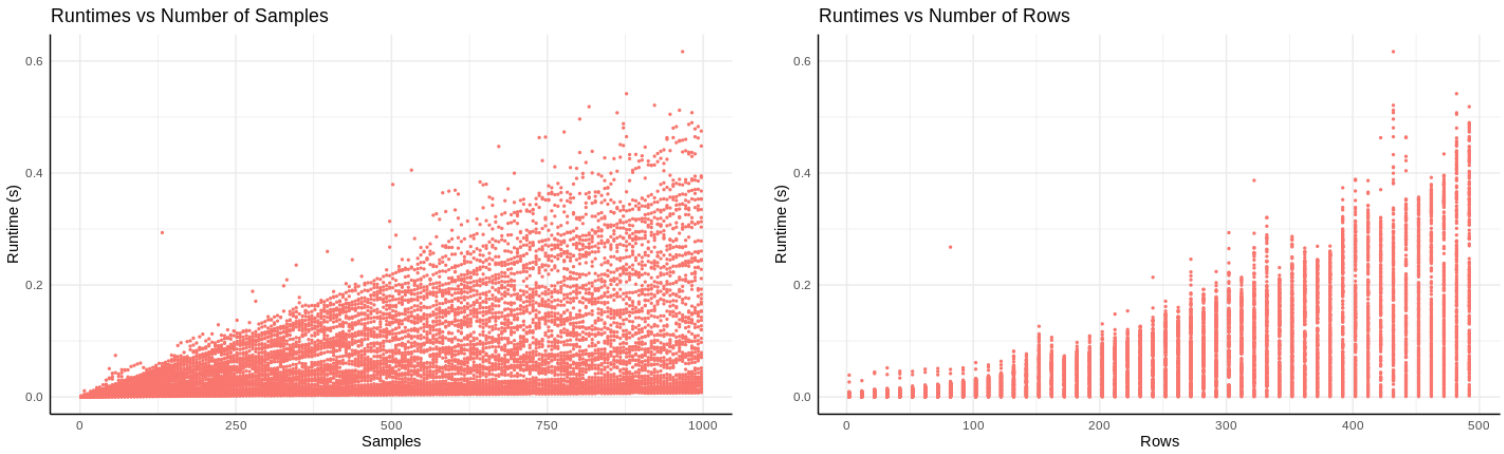

Figure 1: Plot of runtimes vs number of samples (left) and number of rows (right) for the trace estimation method

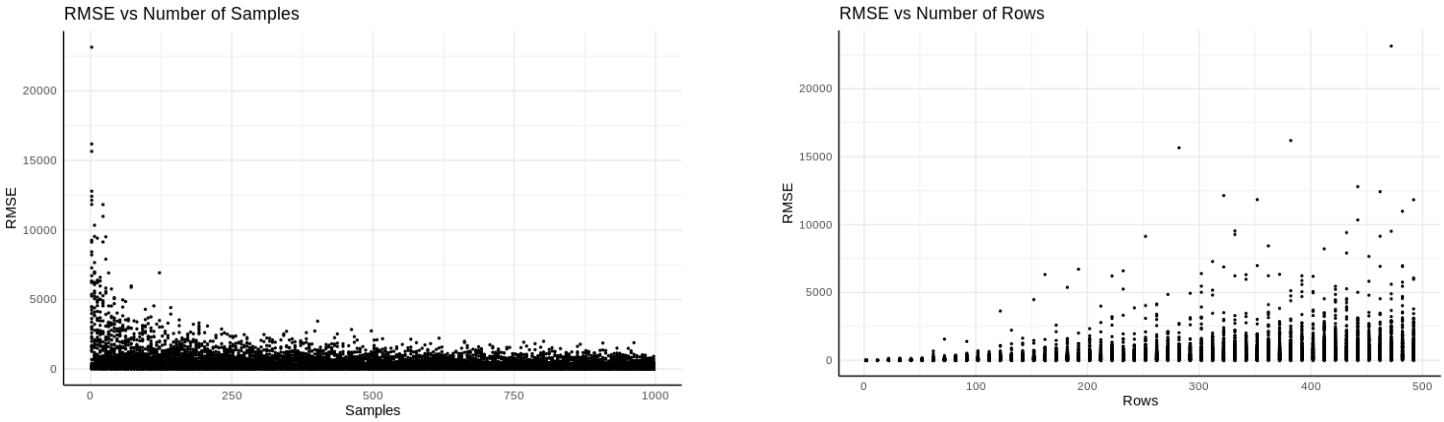

Figure 2: Plot of RMSE vs number of samples (left) and number of rows (right) for the trace estimation method

The RMSE for the trace estimation method followed the distribution:

Quantile	Value
Min	0.000106
25th	50.47337
Mean	195.6933
75th	524.0395
Max	23142.95

Plotting the trace estimate vs the number of samples suggests that as the sample size increases, the estimate converges to the true value of the trace (Figure 3 and Figure A2).

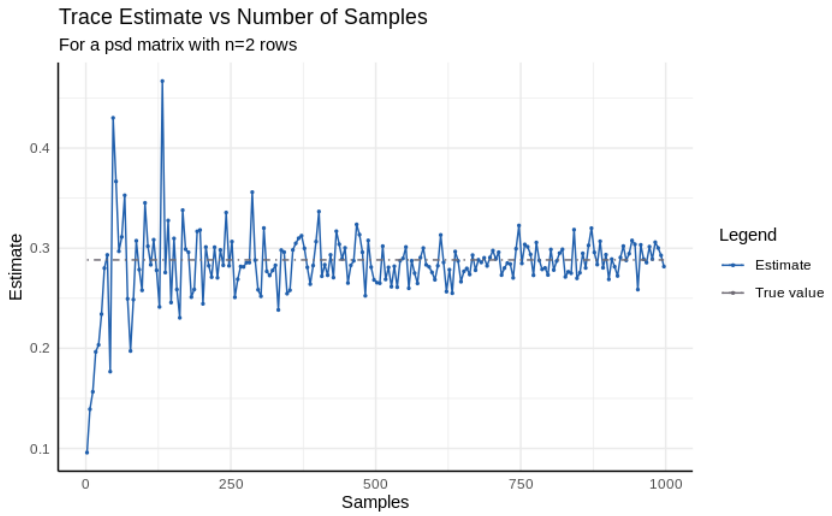

Figure 3: Plot of trace estimate vs number of samples for n=2, with a dashed horizontal line for the true value in gray

At relatively higher values of p (eg. 54, 55, 60, etc.), the norm estimation method's output was Inf or -Inf. An output of NaN was also more common, and starting at $p = 68$ it was also giving 0 values for the estimate. The reason for the 0 values are somewhat obvious: for high values of k compared to p, $C(k, p)^{-1} \rightarrow 0$, and thus the output might have been rounded to 0. Similarly, perhaps after several squares of the strict upper triangular, the value of $trace(\ T(\mathbf{X})^{p-1}\mathbf{X}\ )$ was approaching 0 and multiplying by $C(k, p)^{-1}$ yielded 0/0, yielding NaN. Finally, outputs of +/-Inf might have resulted from overflow, for example from computing $C(1000, 100)$. Before analyzing the results, all of these values needed to be scrubbed from the data. Similarly, for $p > 37$, the RMSE recorded was Inf. This may have also resulted from overflow, and is no doubt an indirect consequence of the high variance of V_p.

As expected, runtime increased as both number of samples and norm power increased (Figure 4). Moreover, a greater sample size was generally correlated with longer runtime and a greater increase in runtime, independent of the powers of the norm (Figure A1).

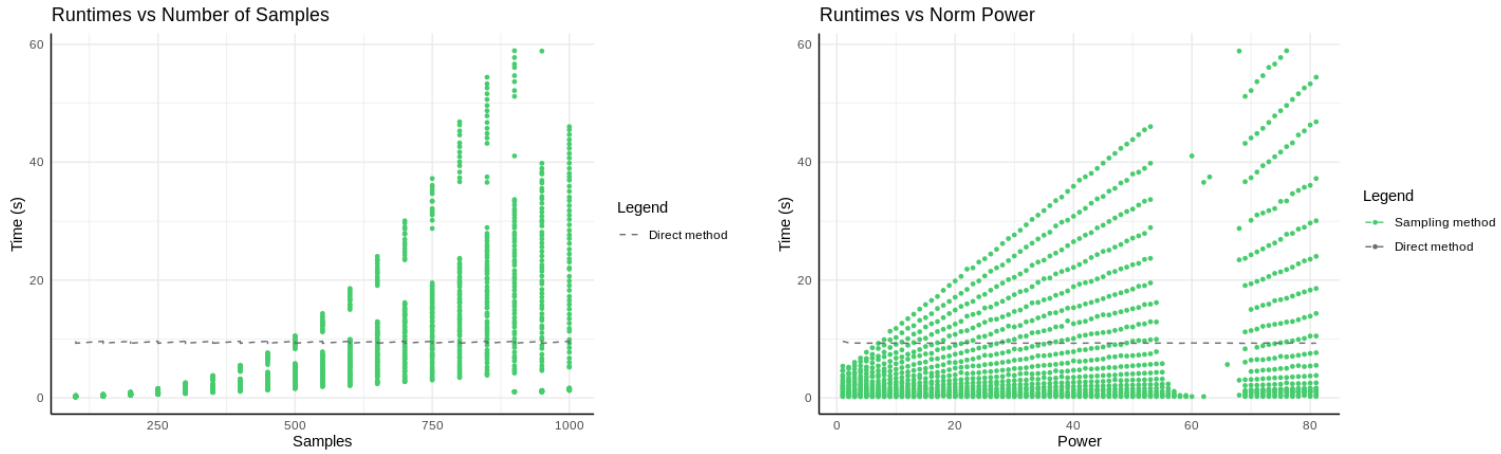

Figure 4: Plot of runtimes vs number of samples (left) and norm power (right), with a dashed horizontal line for the runtime of the direct method in gray for the norm estimation method

Contrary to my expectations, there does not seem to be a strong correlation between RMSE and either the number of samples or the norm power, although higher values of RMSE were seen at higher values of both number of samples and norm power (Figure 5)[3]. All plots involving the errors do not include the Inf values. Other outliers may be attributed to the high variance of V_p.

[3] Note: these plots are misleading because of the effect of outliers on the scale of the y-axis, but zooming in also revealed no strong correlation

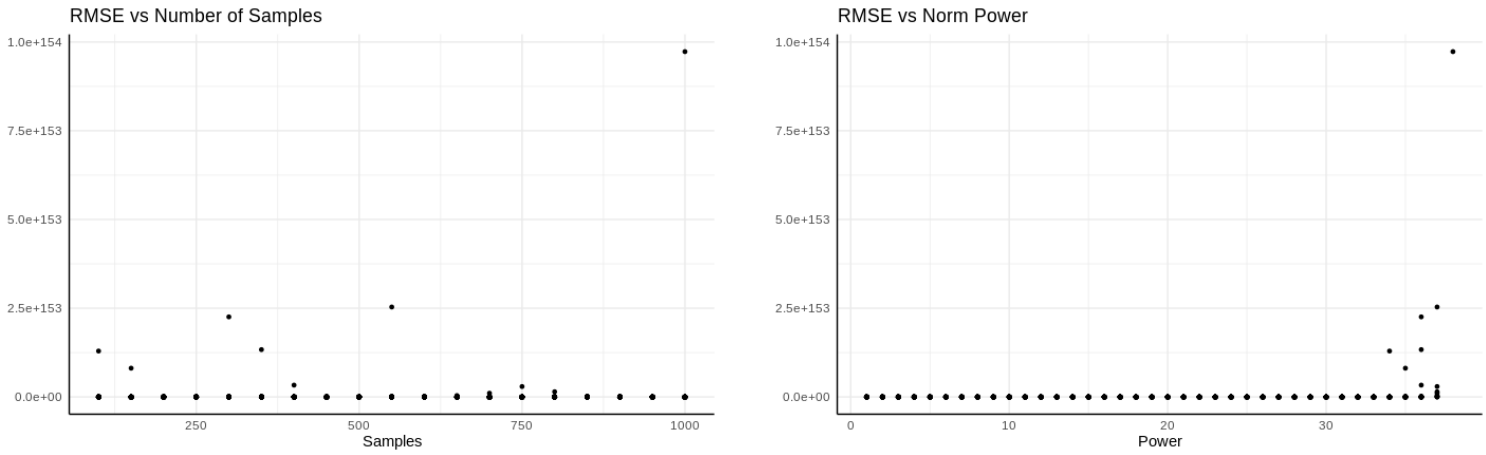

Figure 5: Plot of RMSE vs number of samples (left) and norm power (right) for the norm estimation method

The (finite) RMSE for the norm estimation method were large, and followed:

Quantile	Value
Min	381.21
25th	4.41e38
Mean	2.35e75
75th	3.96e114
Max	9.73e153

For $p = 1$, the estimated value was generally closer for a greater sample size but still exhibited high variance: it is unclear whether this method converged during the simulation (Figure 6). Moreover, the distribution of the estimates for this norm is unsatisfactory, and although it seems symmetric around the true value the mean is not the true value and there is a skew to the left.

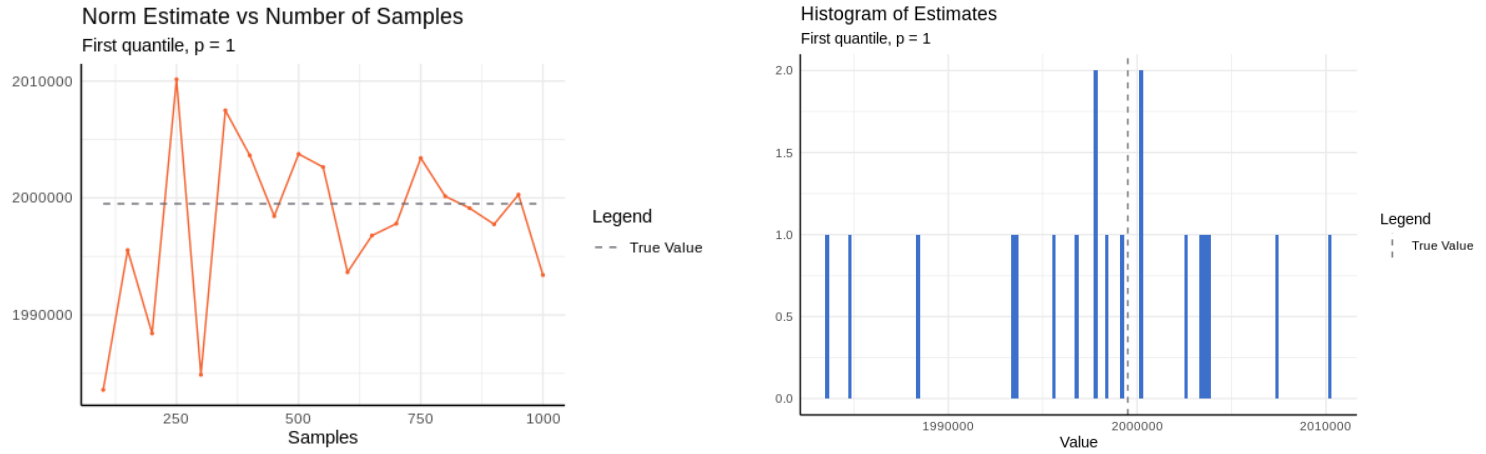

Figure 6: Plot of norm estimate vs number of samples and histogram for p=1, with a dashed line for the true value in gray for the norm estimation method

Similar plots for each quantile of p would have been useful for determining more concretely whether the method is breaking down, but after the first quantile the extreme range of the estimates made the axes unworkable.

CONCLUSION

The trace estimation method was more successful in simulation and better supported the theory than the norm estimation method. For the norm estimation method, improvements might be made by using resampling techniques after initially calculating V_p. Analysis of how well it performs would likely also have been easier with either a smaller input matrix or repeating the experiment with smaller values of p.

APPENDIX

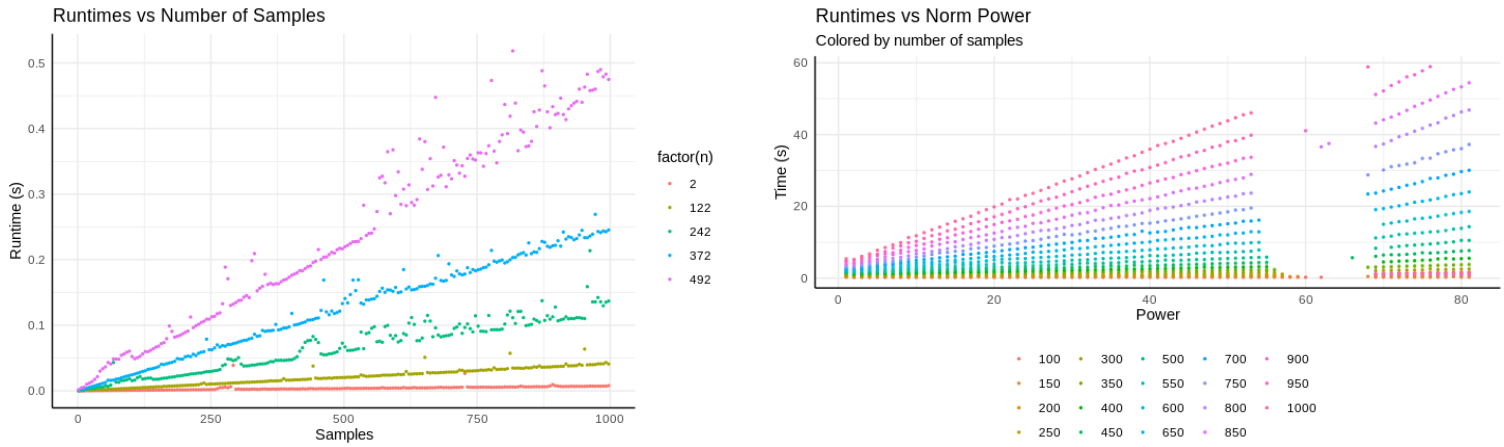

Figure A1: Plot of runtimes vs number of samples for trace estimate, colored by number of rows (left) and runtimes vs norm power for the norm estimation method, colored by number of samples (right)

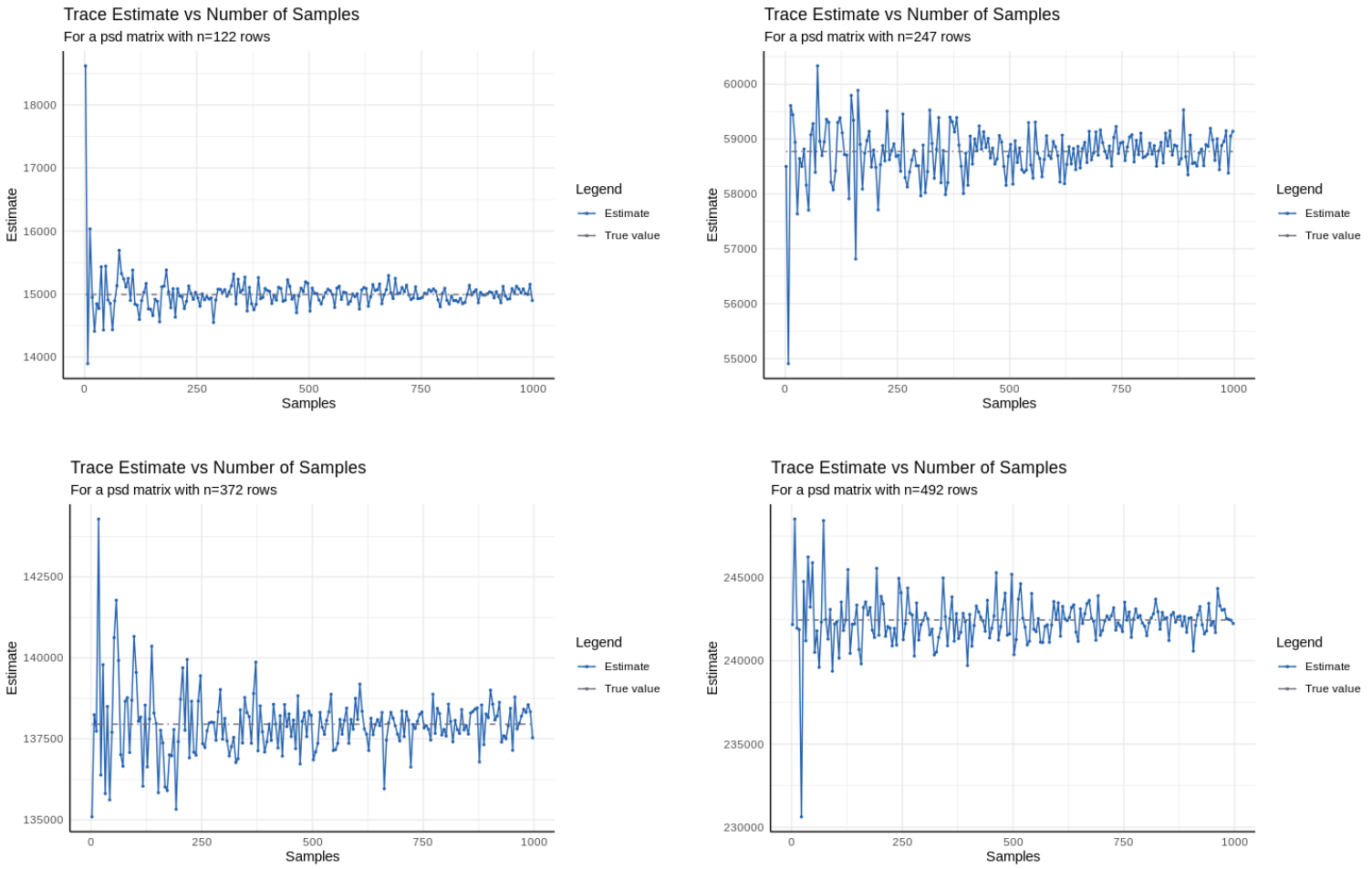

Figure A2: Plots of trace estimate vs number of samples for n = 122, 247, 372, and 492 rows, with a horizontal dashed line for the true value in gray