Homework 2

Your Name

February 27, 2025

We will now start reflecting on the coding questions of Homework 2. The code base that can be used to replicate the result can be found in the following link: `https://github.com/TagoreZhao/STAT260/tree/main/HW2`

1 Problem 6, 7, and 8

The code needed for generating these three matrices is given below:

```
import numpy as np

def generate_covariance_matrix(d):
    indices = np.arange(d)
    Sigma = 2 * 0.5 ** np.abs(indices[:, None] - indices[None, :])
    return Sigma

def generate_gaussian_A(n, d, seed=1234):
    rng = np.random.default_rng(seed)
    Sigma = generate_covariance_matrix(d)
    mean = np.ones(d)
    A = rng.multivariate_normal(mean, Sigma, size=n)
    return A

def generate_t_distribution_A(n, d, df, seed=1234):
    rng = np.random.default_rng(seed)
    Sigma = generate_covariance_matrix(d)
    mean = np.ones(d)
    z = rng.multivariate_normal(mean, Sigma, size=n)
    chi2_samples = rng.chisquare(df, size=(n, 1))
    A = z / np.sqrt(chi2_samples / df)
    return A
```

Listing 1: Python code for generating matrices

Since numpy does not provide built in functions for generating t-distributed random variables, we have to generate the random variables ourselves. The Gaussian random variables are generated using the `multivariate_normal` function, while the t-distributed random variables are generated using the formula $A = Z/\sqrt{\chi^2/df}$, where Z is the Gaussian random variable, χ^2 is the chi-squared random variable, and df is the degrees of freedom.

Problem 9

We will first plot the norm based probability distribution for all three matrices that we generated using seed 1234.

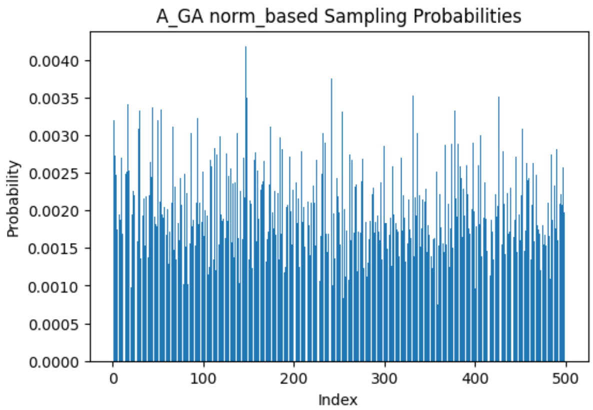

Figure 1: GA Norm based probability distribution

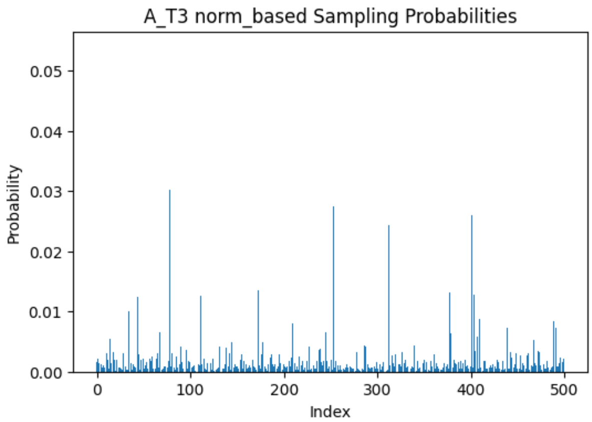

Figure 2: T3 Norm based probability distribution

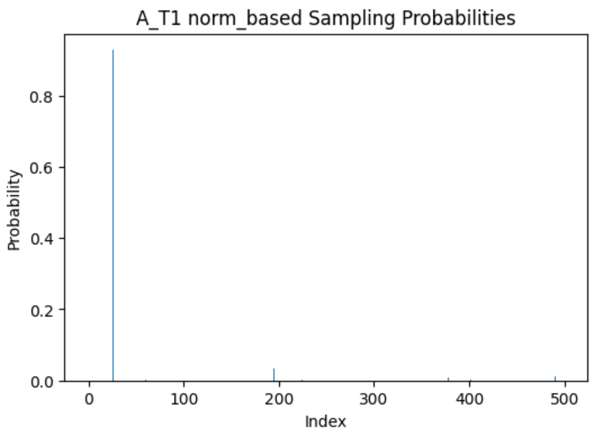

Figure 3: T1 Norm based probability distribution

We will now plot the Frobenius and spectral error for the approximations of the three matrix multiplications.

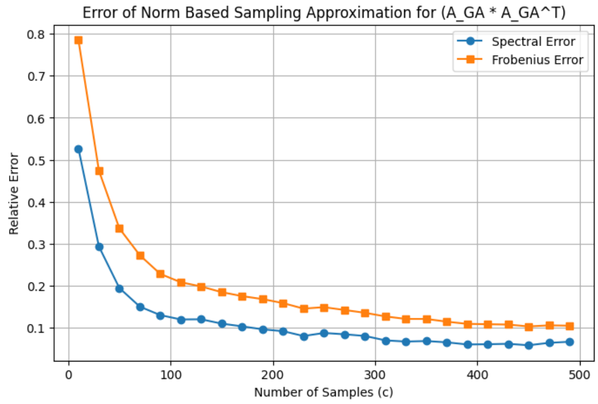

Figure 4: Error of Norm Based Sampling Approximation for $(A_{GA}^{\top} A_{GA})$

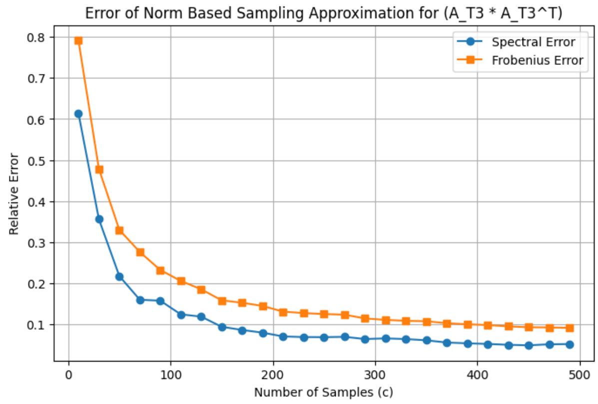

Figure 5: Error of Norm Based Sampling Approximation for $(A_{T3}^{\top} A_{T3})$

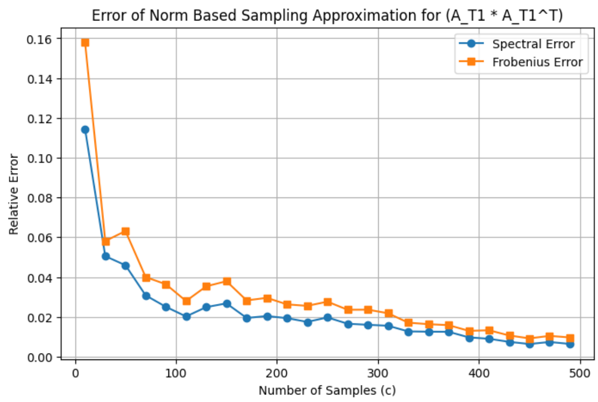

Figure 6: Error of Norm Based Sampling Approximation for $(A_{T1}^{\top} A_{T1})$

Problem 10

We will now plot the Frobenius and spectral error for the approximations of the three left singular matrices multiplication.

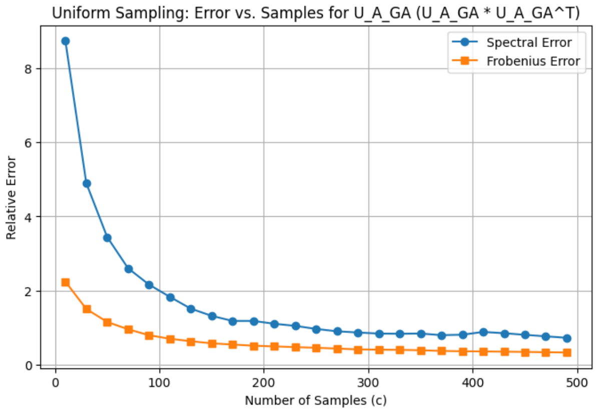

Figure 7: Error of Uniform Based Sampling Approximation for $(U_{GA}^{\top} U_{GA})$

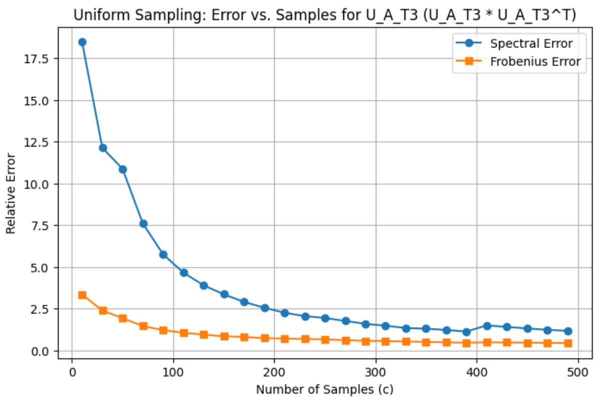

Figure 8: Error of Uniform Based Sampling Approximation for $(U_{T3}^{\top} U_{T3})$

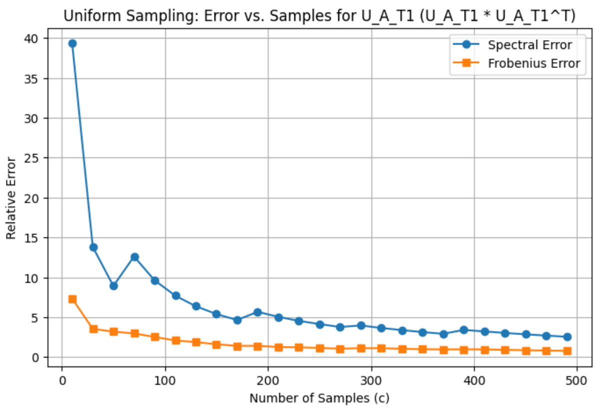

Figure 9: Error of Uniform Based Sampling Approximation for $(U_{T1}^{\top} U_{T1})$

Problem 11

We will now plot the Frobenius and spectral error for the approximations of $A^\top A$ using leverage score sampling.

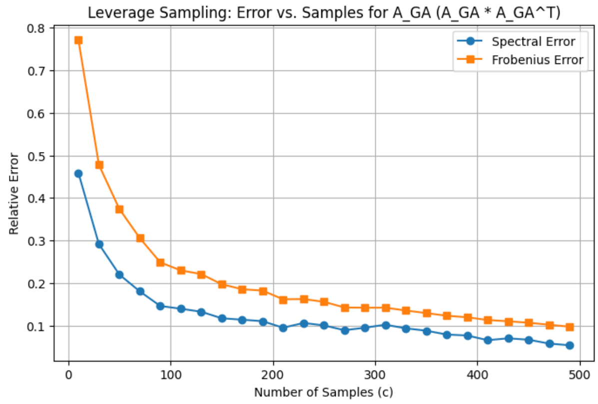

Figure 10: Error of Leverage Based Sampling Approximation for $(A_{GA}^\top A_{GA})$

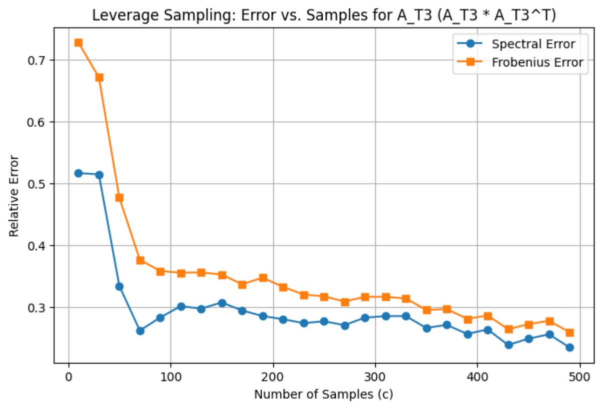

Figure 11: Error of Leverage Based Sampling Approximation for $(A_{T3}^\top A_{T3})$

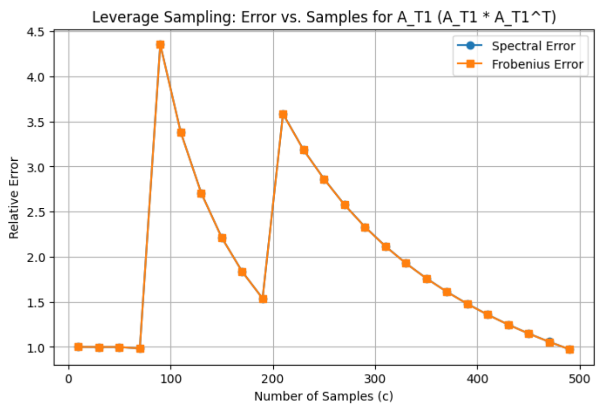

Figure 12: Error of Leverage Based Sampling Approximation for $(A_{T1}^\top A_{T1})$

The results looks similar for A_{GA} and A_{T3}, but the error for A_{T1} is significantly higher than the other two matrices. This means that the leverage score sampling is not as effective for A_{T1} as it is for the other two matrices.

Problem 12