PS5

ShubeiWang

10/15/2018

1.

Denote $\Lambda_{ii} = \lambda_i, 1 \leqq i \leqq n (\lambda_i$ is the i^{th} eigenvalue), so $|\Lambda| = \Pi_i^n \lambda_i$. And because Γ is an orthogonal matrix of eigenvectors, $|\Gamma| = |\Gamma^T| = 1$. So $A = \Gamma\Lambda\Gamma^T \to |A| = |\Gamma||\Lambda||\Gamma^T| \to |A| = 1 * \Pi_i^n \lambda_i * 1 = \Pi_i^n \lambda_i$, i.e. $|A|$ is the product of the eigenvalues.

2.

Because when z is large the expression will cause overflow. The $exp(z)$ term will be *Inf* in R and the *expit* function will be *NaN*. To avoid this we can re-express it as $expit(z) = \frac{1}{1+exp(-z)}$. This function will return 1 when z takes very large values.

3.

Because we can only accurately present about 16 digits, x will lost accuracy after 4 decimal places. Thus their variance won't be the same and will be accurate to about 4 places. In this case they agree to 5 digits.

4.

(a)

Because in this case each task takes little time and the communication overhead of starting and stopping the tasks will reduce efficiency. So it's better to separate the computation into p tasks.

(b)

Assume each element in X, Y takes up a bytes.

A:

(1) There will be p copies of X and p blocks of Y which take up $(pn^2 + n^2)a$ bytes. And the result takes n^2a bytes. So the memory used=$(p + 2)n^2a$ bytes.

(2) In total we need to transfer X p times, each of the p blocks of Y once and the result. So the communication cost= $n^2p + 2n^2$.

B:

(1) There will be p copies of one block of X and p blocks of Y at a single moment which take up $(n^2p/p + n^2)a = 2n^2a$ bytes. And the result in a single moment takes up n^2a/p bytes. So the memory used=$(2n^2 + n^2/p)a$ bytes.

(2) In total we need to transfer each of the p blocks of X p times, each of the p blocks of Y p times and the result.The communication cost=$2n^2p + n^2$.

We can conclude that B is better for minimizing memory use and A is for minimizing communication.