Scalable Log Determinants for Gaussian Process Kernel Learning

Team 7: K. Pirov, F. Loginov, A. Bauman, K. Nazirkhanova, A. Savinov

December 23, 2018

1 Introduction

The problem of estimating the trace of matrix functions appears frequently in applications of machine learning, signal processing, scientific computing, statistics, computational biology and computational physics. An important instance of the trace estimation problem is that of approximating $\log(\det(A))$. Log-determinants of covariance matrices play an important role in Gaussian processes(GPs) [1], [2]. The goal of our project is to reduce the complexity of exact GP $\mathcal{O}\left(n^3\right)$. Log-determinant computations also appear in applications such as kernel learning [13], discrete probabilistic models, Bayesian Learning, spatial statistics.

2 Problem statement and methods

Our goal is to estimate, for a symmetric positive definite matrix $\widetilde{K}$,

$$\log|\widetilde{K}| = \operatorname{tr}(\log(\widetilde{K})) \quad \text{and} \quad \frac{\partial}{\partial\theta_i}[\log|\widetilde{K}|] = \operatorname{tr}\left(\widetilde{K}^{-1}\left(\frac{\partial K}{\partial\theta_i}\right)\right).$$

- If $\widetilde{K} = QTQ^T$, where Q is orthonormal, then because $\widetilde{K}$ and T have the same eigenvalues:

  $$\log|\widetilde{K}| = \operatorname{tr}(\log T) = E\left[\mathbf{z}_i^T(\log T)\mathbf{z}_i\right] \approx \sum_{i=1}^{t}\mathbf{z}_i^T(\log T)\mathbf{z}_i$$

  So we need to obtain a decomposition $\widetilde{K} = QTQ^T$, it can be done using the **Lanczos tridiagonalization algorithm** [4]. This algorithm takes the matrix $\widetilde{K}$ and a probe vector $\mathbf{z}$ with independent entries with mean zero and variance one and outputs the decomposition QTQ^T ($\mathbf{z}$ is the first column of Q).
  The second method is to use **Chebyshev approximation** [5]. We can approximate log $|\widetilde{K}|$ using the Chebyshev interpolant of $\log(1+\alpha x)$:

  $$\log|\widetilde{K}| - n\log\beta = \log|I+\alpha B| \approx \sum_{j=0}^{m} c_j \operatorname{tr}\left(T_j(B)\right),$$

where T_j - Chebyshev polynomials, c_j - coefficients of Chebyshev decomposition and $B = (\widetilde{K}/\beta - 1)/\alpha$ has eigenvalues $\lambda_i \in (-1, 1)$
We compute for all probes of vector $\mathbf{z}$:

$$w_0 = \mathbf{z}, w_1 = B\mathbf{z}, w_{j+1} = 2Bw_j - w_{j-1} \text{ for } j \geq 1$$

$$\log |\widetilde{K}| \approx E\left[\sum_{j=0}^{m} c_j \mathbf{z}^T w_j\right]$$

- Estimating of derivatives can be implemented also using Lanczos or Chebyshev algorithm[5].
  Using Chebyshev method we need to compute derivatives of vector w_i and matrix B with respect to parametres.

$$\frac{\partial w_0}{\partial \theta_i} = 0, \quad \frac{\partial w_1}{\partial \theta_i} = \frac{\partial B}{\partial \theta_i}\mathbf{z}, \quad \frac{\partial w_{j+1}}{\partial \theta_i} = 2\left(\frac{\partial B}{\partial \theta_i} w_j + B\frac{\partial w_j}{\partial \theta_i}\right) - \frac{\partial w_{j-1}}{\partial \theta_i} \text{ for } j \geq 1$$

$$\frac{\partial}{\partial \theta_i} \log |\widetilde{K}| \approx E\left[\sum_{j=0}^{m} c_j \mathbf{z}^T \frac{\partial w_j}{\partial \theta_i}\right]$$

  Applying Lanczos decomposition it is possible to approximate derivatives of the log determinant at minimal cost.

$$\hat{g} = Q_m(T^{-1} e_1 \|z\|) \approx \widetilde{K}^{-1} z$$

$$\mathrm{tr}(\widetilde{K}^{-1}(\frac{\partial \widetilde{K}}{\partial \theta_i})) = E\left[(\widetilde{K}^{-1}\mathbf{z})^T \frac{\partial \widetilde{K}}{\partial \theta_i}\mathbf{z}\right]$$

The quality of measurements

As error metrics we used Euclidean norm for vectors and spectral norm for matrices such as:

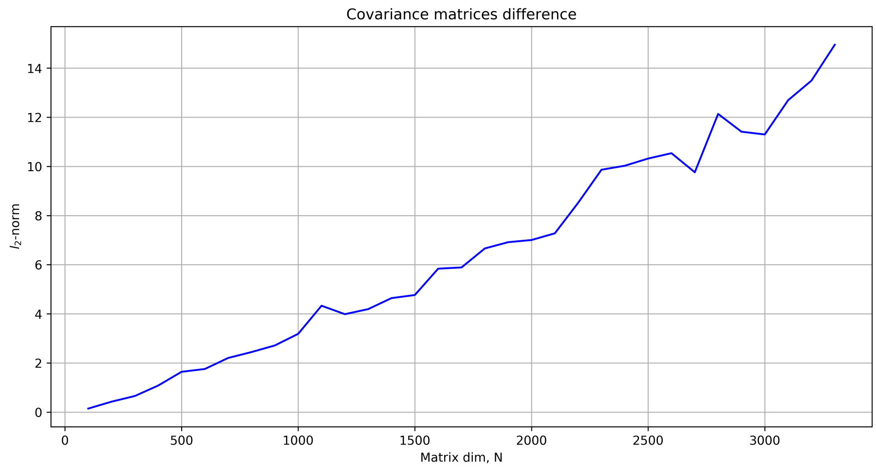
- accuracy:
$$\|y_{pred} - y_{true}\|_2,$$
  where y_{pred} is the answer of our model and y_{true} is the right answer taken from the dataset,
- the loss of orthogonality:
$$\|Q^T Q - I\|_2,$$
- difference between matrices A and B is:
$$\|A - B\|_2,$$
  it uses for covariance matrices difference,
- relative error:
$$\frac{|a - b|}{|b|},$$
  where a is $\log\det(A)$ calculated using our methods implementation and b is $\log\det(A)$ using embedded numpy function.

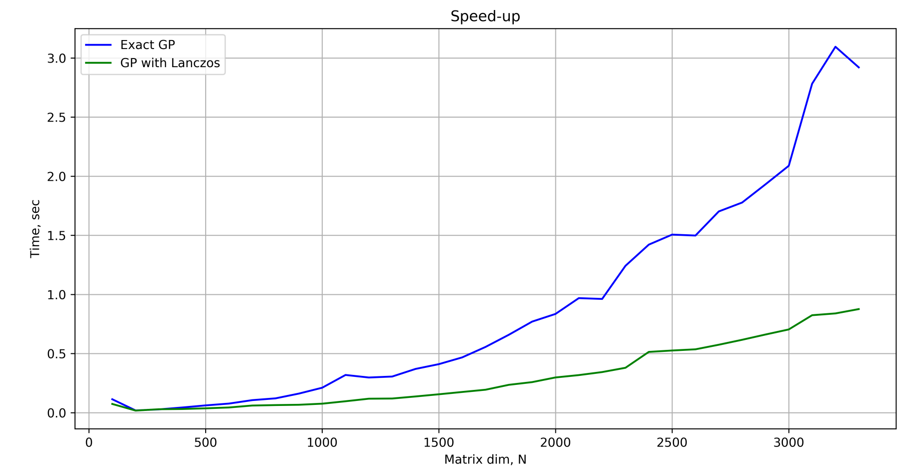
3 Data and applications

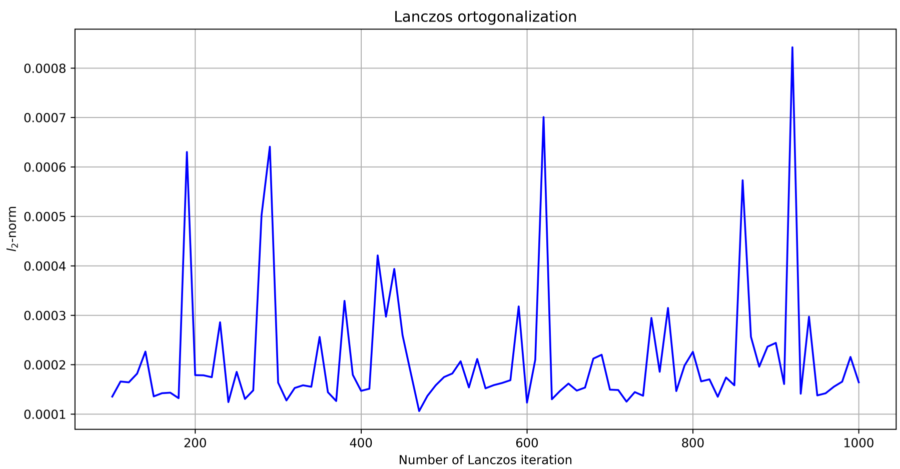
In our project we used Skillcraft dataset from the UCI dataset repository [3], which contains next information: game ID; league index; player's age; hours per week spent playing; total hours playing; action per minute; number of units or buildings selections by hotkeys; number of units or buildings assigned to hotkeys; number of unique hotkeys; number of attack actions on minimap; number of right-clicks on minimap; number of PACs; mean duration in between PACs; mean latency from the onset of a PACs to their first action; mean number of actions within each PAC; the number of map viewed by the player; number of SCVs, drones, and probes trained; unique unites, number of ghosts, infestors, and high templars trained; number of used abilities requiring specific targeting instructions. We solved simple Gaussian Process regression problem of action per minute predicting.

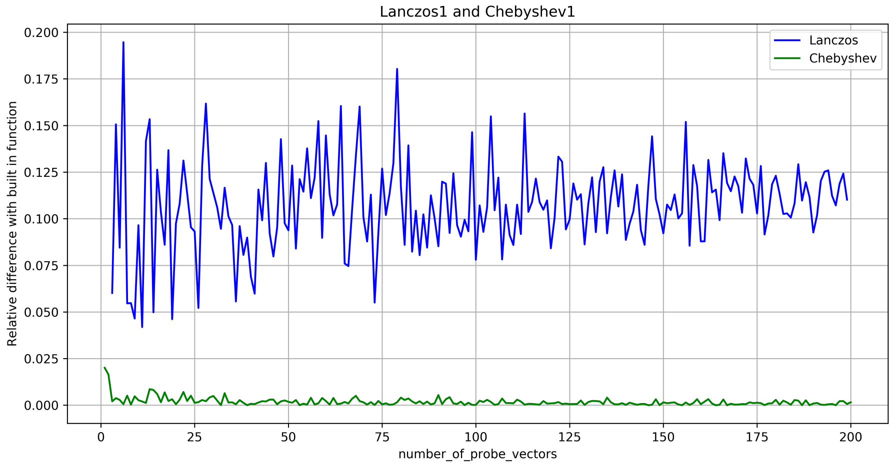
4 Results

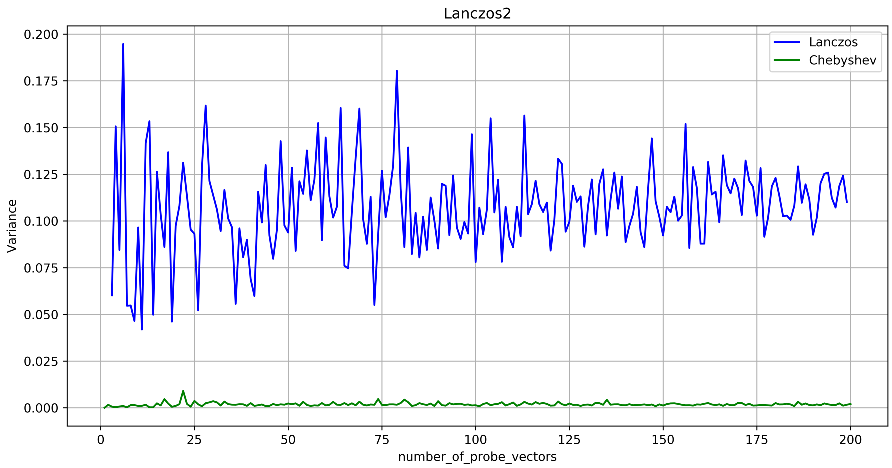
Full results are available at
`https://github.com/KhurramPirov/Log-Determinants-Estimator`




Speed-up
Exact GP
GP with Lanczos
Time, sec
Matrix dim, N
0
500
1000
1500
2000
2500
3000
0.0
0.5
1.0
1.5
2.0
2.5
3.0


Lanczos ortogonalization
l_2-norm
Number of Lanczos iteration
200
400
600
800
1000
0.0001
0.0002
0.0003
0.0004
0.0005
0.0006
0.0007
0.0008





4.1 Summary

According to our results, Chebyshev algorithm is more stable than Lanczos one, because during Chebyshev has no problem with orthogonalization. Also in Chebyshev iterations we approximated $\log |I + \alpha B|$, but it is the same for Bayessian problem implementation.

Important comments For Chebyshev iterations it was neccesary to generate random vectors from uniform distribution $U((-1,1)^n)$. We tried to use normal distribution, but iterations was failed. Lanczos iterations required random vector distribution with mean value is equal to zero along each axis.

5 Contribution of every team member

Khurram was responsible for report, ideas, theory, plots, Lanczos and Chebyshev implementation, Fedor and Alexey were helping with Lanczos implementation, final results presentation and report, Kamilla and Andrey have done full data analysis with speed-up, Lanczos stability, solution accuracy and report, also Andrey took part and gave useful ideas for Chebyshev algorithm.

References

[1] C. Paige. Practical use of the symmetric Lanczos process with re-orthogonalization.

[2] B. N. Parlett. A new look at the Lanczos algorithm for solving symmetric systems of linear equations. Linear algebra and its applications

[3] A. Asuncion and D. Newman. UCI machine learning repository. `http://archive.ics.uci.edu/ml/datasets/skillcraft1+master+table+dataset`

[4] Jacob R. Gardner, Geoff Pleiss, David Bindel, Kilian Q. Weinberger, Andrew Gordon Wilson. GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration

[5] Kun Dong, David Eriksson, Hannes Nickisch, David Bindel, Andrew Gordon Wilson. Scalable Log Determinants for Gaussian Process Kernel Learning