Multiresolution Kernel Approximation for Gaussian Process Regression

THE UNIVERSITY OF CHICAGO

Yi Ding, Risi Kondor, and Jonathan Eskreis-Winkler

NIPS 2017, Long Beach, CA, USA

Motivation

- Gaussian process regression generally does not scale to beyond a few thousands data points without applying some sort of kernel approximation method.
- Most approximations focus on the high eigenvalue part of the spectrum of the kernel matrix, K, which leads to bad performance when the length scale of the kernel is small.
- We introduce **Multiresolution Kernel Approximation (MKA)**, the first true broad bandwidth kernel approximation algorithm.
- MKA is memory efficient, and a direct method, which means that it also makes it easy to approximate K^{-1} and $\det(K)$.

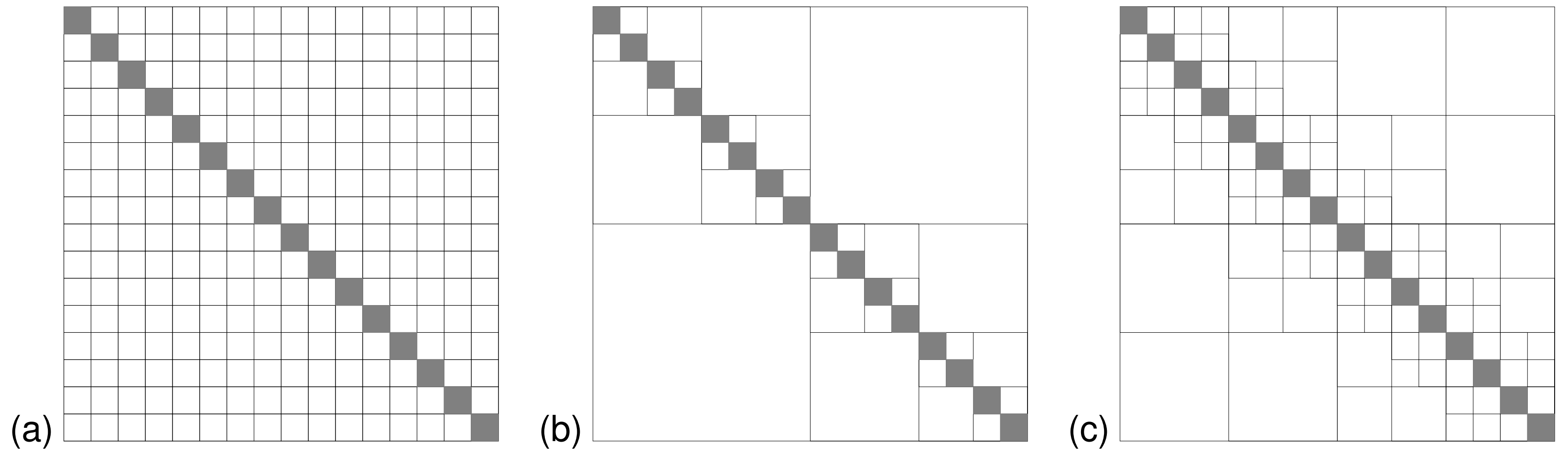
Gaussian Process Regression

- Gaussian Processes (GPs) are a generalization of multivariate Gaussian distributions to the case when the underlying variables form a continuum indexed by some set $\mathcal{X}$.
- A GP is fully specified by its mean function $\mu(x)$, and covariance function $k(x,x')$, where k can be any positive semi-definite kernel.
- Given training data $\{(x_1,y_1),\dots,(x_n,y_n)\}$, the model is $y_i = f(x_i)+\epsilon$, where $\epsilon \sim \mathcal{N}(0,\sigma^2)$ and σ^2 is a noise parameter. The posterior is also a GP with mean
$$\mu'(x) = \mu(x) + \boldsymbol{k}_x (K+\sigma^2 I)^{-1} \boldsymbol{y},$$
where $\boldsymbol{k}_x = (k(x,x_1),\dots,k(x,x_n))$, $\boldsymbol{y} = (y_1,\dots,y_n)$, K is the Gram matrix or kernel matrix with elements $K_{i,j} = k(x_i,x_j)$, and covariance
$$k'(x,x') = k(x,x') - \boldsymbol{k}_{x'}(K+\sigma^2 I)^{-1}\boldsymbol{k}_x.$$

Global Low Rank Methods

Mathematically,
$$k(x,x') \approx \sum_{s=1}^{m}\sum_{j=1}^{m} k(x,x_{i_s})\, c_{i_s,i_j}\, k(x_{i_j},x'),$$
Assuming that $\{x_{i_1},\dots,x_{i_m}\}$ is a subset of the original point set $\{x_1,\dots,x_n\}$, amounts to an approximation of the form $K \approx K_{*,I} C K_{*,I}^{\top}$, with $I = \{i_1,\dots,i_m\}$. The canonical choice for C is $C = W^{+}$, where $W = K_{I,I}$, and W^{+} denotes the Moore-Penrose pseudoinverse of W. The resulting approximation
$$K \approx K_{*,I} W^{+} K_{*,I}^{\top},$$
is known as the Nyström approximation.

Local and Hierarchical Low Rank Methods



(a) In a simple blocked low rank approximation the diagonal blocks are dense (gray), whereas the off-diagonal blocks are low rank. (b) In an HODLR matrix the low rank off-diagonal blocks form a hierarchical structure leading to a much more compact representation. (c) $\mathcal{H}^2$ matrices are a refinement of this idea.

Multiresolution Kernel Approximation (MKA)

MKA is a data adapted multiscale kernel matrix approximation method, which reflects the "distant clusters only interact in a low rank fashion" insight of the fast multipole method. We have the following two definitions:

- We say that a matrix H is **c–core-diagonal** if $H_{i,j} = 0$ unless either $i,j \le c$ or $i = j$.
- A **c–core-diagonal compression** of a symmetric matrix $A \in \mathbb{R}^{m\times m}$ is an approximation of the form
$$A \approx Q^{\top} H\, Q = \underbrace{q_1^{\top}\dots q_L^{\top}}_{Q^{\top}} H\, \underbrace{q_L \dots q_1}_{Q} = (\blacksquare)(\searrow)(\blacksquare),$$
where Q is orthogonal and H is c–core-diagonal.

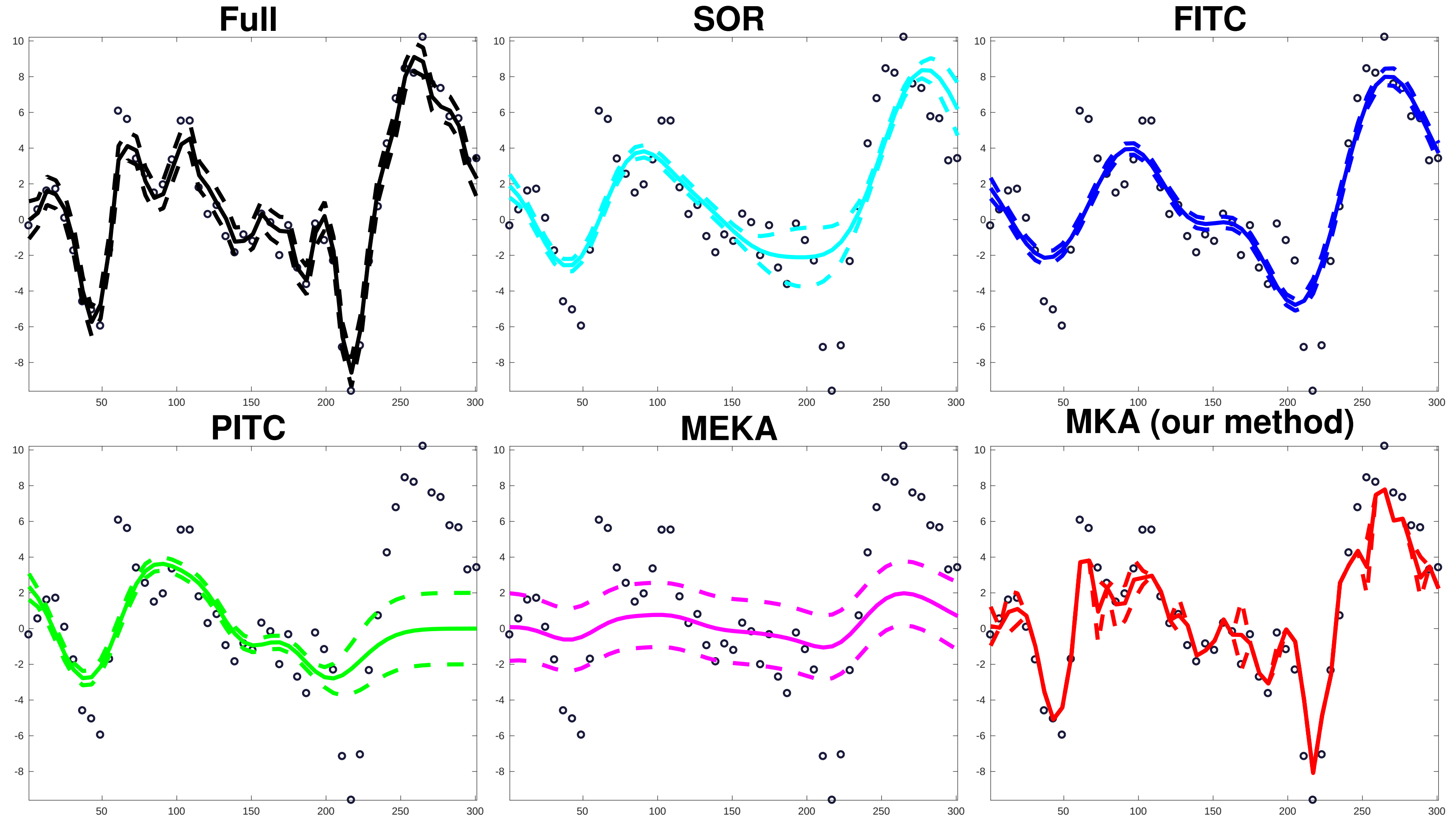
Application to GPs

The direct way of applying MKA to speed up GP regression is simply using it to approximate the augmented kernel matrix $K' = (K+\sigma^2 I)$ and then inverting this approximation. Note that the resulting $\tilde{K}'^{-1}$ never needs to be evaluated fully, in matrix form. Instead, the matrix-vector product $\tilde{K}'^{-1}\boldsymbol{y}$ can be computed in "matrix-free" form.

Assuming that the testing set $\{x_1,\dots,x_p\}$ is known at training time, however, we can take an alternative approach, whereby instead of approximating K or K', we compute the MKA approximation of the joint train/test kernel matrix
$$\mathcal{K} = \left(\begin{array}{c|c} K & K_* \\ \hline K_*^{\top} & K_{\text{test}} \end{array}\right) \quad \text{where} \begin{array}{l} K_{i,j} = k(x_i,x_j)+\sigma^2 \\ [K_*]_{i,j} = k(x_i,x'_j) \\ [K_{\text{test}}]_{i,j} = k(x'_i,x'_j). \end{array}$$
Writing $\mathcal{K}^{-1}$ in blocked form
$$\tilde{\mathcal{K}}^{-1} = \left(\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right),$$
and taking the Schur complement of D now recovers an alternative approximation $K^{-1} := A - BD^{-1}C$ to K^{-1} which is consistent with the off-diagonal block K^{*} leading to our final MKA–GP formula $\widehat{f} = K_*^{\top} K^{-1} y$, where $\widehat{\boldsymbol{f}} = (\widehat{f}(x'_1),\dots,\widehat{f}(x'_p))^{\top}$. While conceptuall this is somewhat more involved than naively estimating K', assuming $p \ll n$, the cost of inverting D is negligible, and the overall serial complexity of the algorithm remains $(n+p)^2$.

Simulation: 1D Toy Data

Figure 1: Snelson's 1D example: ground truth (black circles); prediction mean (solid line curves); one standard deviation in prediction uncertainty (dashed line curves).

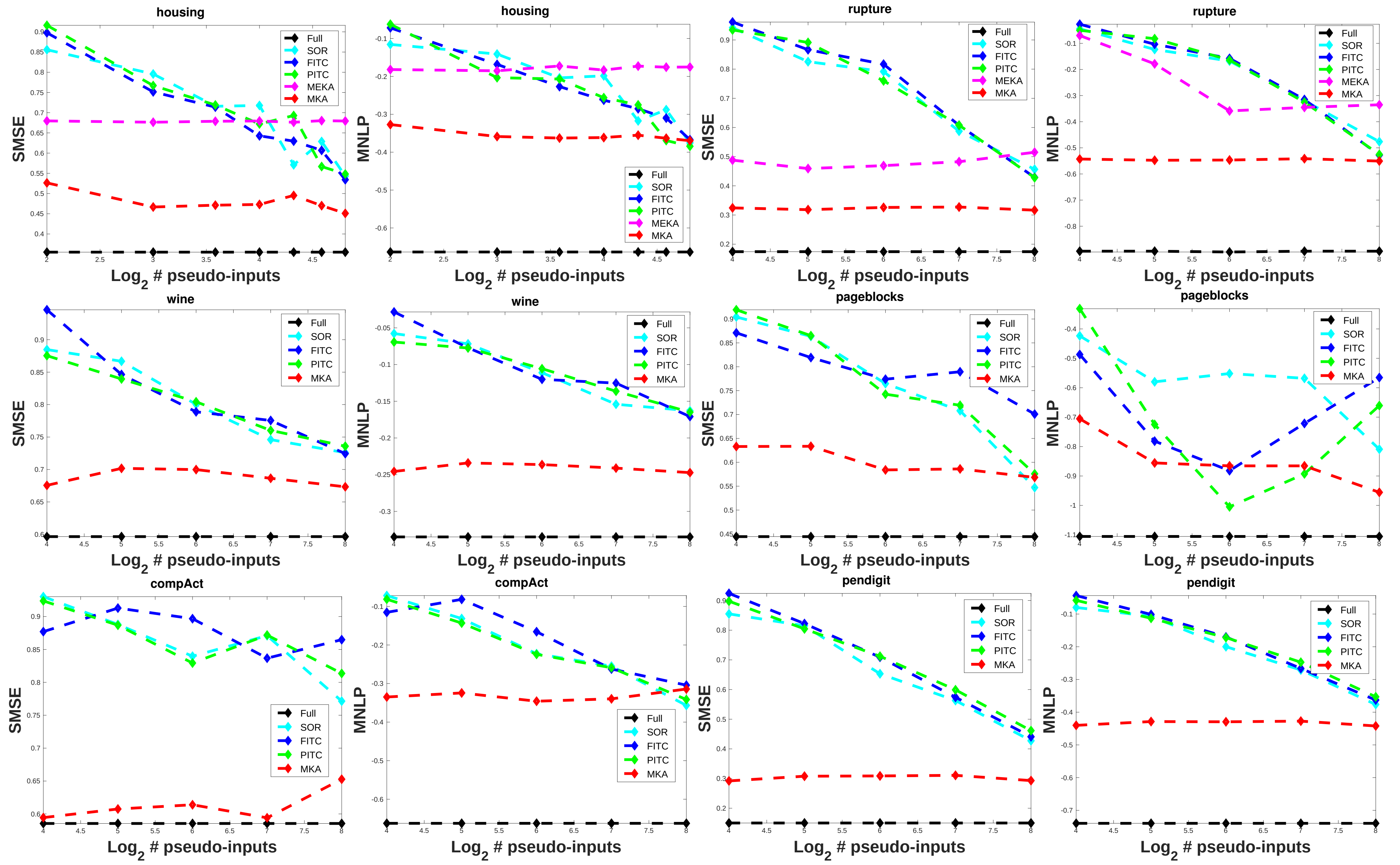


Experiments on Real Data

Table 1: Regression Results with k to be # pseudo-inputs/d_{core} : SMSE(MNLP)

Method	k	Full	SOR	FITC	PITC	MEKA	MKA
housing	16	0.36(−0.32)	0.93(−0.03)	0.91(−0.04)	0.96(−0.02)	0.85(−0.08)	**0.52(−0.32)**
rupture	16	0.17(−0.89)	0.94(−0.04)	0.96(−0.04)	0.93(−0.05)	0.46(−0.18)	**0.32(−0.54)**
wine	32	0.59(−0.33)	0.86(−0.07)	0.84(−0.03)	0.87(−0.07)	0.97(−0.12)	**0.70(−0.23)**
pageblocks	32	0.44(−1.10)	0.86(−0.57)	0.81(−0.78)	0.86(−0.72)	0.96(−0.10)	**0.63(−0.85)**
compAct	32	0.58(−0.66)	0.88(−0.13)	0.91(−0.08)	0.88(−0.14)	0.75(−0.21)	**0.60(−0.32)**
pendigit	64	0.15(−0.73)	0.65(−0.19)	0.70(−0.17)	0.71(−0.17)	0.53(−0.29)	**0.30(−0.42)**

Figure 2: SMSE and MNLP as a function of the number of pseudo-inputs/d_{core} on two datasets. In the given range MKA clearly outperforms the other methods in both error measures.



Across the range of pseudo-inputs/d_{core} size considered, MKA's performance is robust to d_{core}, while low-rank based methods' performance changes rapidly, which shows MKA's ability to achieve good regression results even with a crucial compression level.

Conclusions

- Whether a learning problem is low rank or not depends on the nature of the data rather than just the spectral properties of the kernel matrix.
- MKA allows fast direct calculations of the inverse of the kernel matrix and its determinant, which are almost always the computational bottlenecks in GP problems.

Contacts: Yi Ding (dingy@uchicago.edu) Risi Kondor (risi@uchicago.edu) Jonathan Eskreis-Winkler (eskreiswinkler@uchicago.edu)