# Efficient Laplace kernel approximation with a Mondrian process

Matej Balog and Yee Whye Teh

University of Oxford

## Objectives

Kernel ridge regression on $N$ training points has $\Theta(N^3)$ complexity due to inverting an $N \times N$ kernel matrix. We investigate random feature approximations of the Laplace kernel that

- run in $\mathcal{O}(NC^2)$ time, where $C$ can be controlled to trade-off speed and approximation accuracy
- can be trained for several hyperparameter values at once efficiently, yielding a fast procedure for selecting the right model complexity

## Random kitchen sinks

Instead of inverting an $N \times N$ matrix, randomly find a feature map $z : \mathbb{R}^D \to \mathbb{R}^C$ ($C \ll N$) such that

$$k(\mathsf{x}, \mathsf{x}') \approx z(\mathsf{x})^T z(\mathsf{x}')$$

([2]). The resulting ridge regression problem has solution

$$\boldsymbol{\theta}^{\text{MAP}} = (\mathbf{Z}^T\mathbf{Z} + \delta^2\mathbf{I}_C)^{-1}\mathbf{Z}^T\mathbf{y}$$

where $\mathbf{Z} \in \mathbb{R}^{N \times C}$ is the feature matrix. We invert a $C \times C$ matrix, solving the problem in time $\mathcal{O}(C^3 + C^2N)$.

## Mondrian process

The Mondrian process [1] on an axis-aligned box $\Theta$ is a stochastic process taking values in **guillotine partitions** of $\Theta$. It starts with no cuts at time 0 and as time progresses, cuts randomly appear, hierarchically splitting $\Theta$ into more refined partitions.

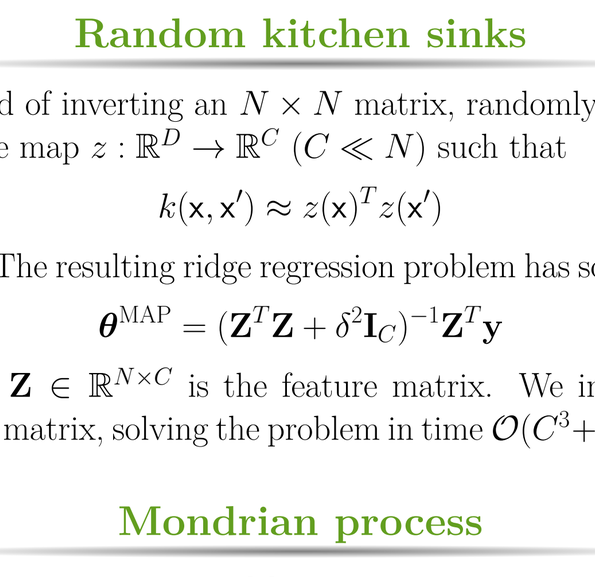


Figure 1: A sample from a 2D Mondrian process on $\Theta = [0, 1] \times [0, 1]$ with lifetime $\lambda = 1.5$.

The **Mondrian process** on $\Theta$ **with lifetime** $\lambda$ is the law of a Mondrian process on $\Theta$ stopped at time $\lambda$ (i.e., ignoring cuts after time $\lambda$). Useful property:

*A box $[a_1, b_1] \times \cdots \times [a_D, b_D]$ contains no cut up to time $\lambda$ with probability $\exp(-\lambda \sum_{d=1}^{D}(b_d - a_d))$.*

## Mondrian approximation

The **(symmetric) Laplace kernel** is

$$k(\mathsf{x}, \mathsf{x}') = \exp\left(-\lambda \|\mathsf{x} - \mathsf{x}'\|_1\right)$$

where $\lambda$ is the inverse lengthscale hyperparameter. We obtain our random feature mapping as follows:

- Sample from a Mondrian process with lifetime $\lambda$.
- Define $z(\mathsf{x})$ to be the indicator vector of the partition cell into which point $\mathsf{x} \in \mathbb{R}^D$ falls.

Inner products in this feature space are

$$z(\mathsf{x})^T z(\mathsf{x}') = \begin{cases} 1 & \text{if } \mathsf{x}, \mathsf{x}' \text{ are in the same cell} \\ 0 & \text{otherwise} \end{cases}$$

## Mondrian approximation

Points $\mathsf{x}$, $\mathsf{x}'$ fall into the same cell if and only if no cut of the Mondrian separates them. By properties of the Mondrian process:

$$\mathbb{P}(z(\mathsf{x})^T z(\mathsf{x}') = 1) = \exp\left(-\lambda \|\mathsf{x} - \mathsf{x}'\|_1\right) = k(\mathsf{x}, \mathsf{x}')$$

Concatenating feature vectors $z_1(\mathsf{x}), \ldots, z_M(\mathsf{x})$ from $M$ independent Mondrian samples we obtain a feature space in which inner products are Monte Carlo estimates of the target kernel:

$$\left(\frac{z(\mathsf{x})}{\sqrt{M}}\right)^T \left(\frac{z(\mathsf{x}')}{\sqrt{M}}\right) = \frac{1}{M}\sum_{m=1}^{M} z_m(\mathsf{x})^T z_m(\mathsf{x}') \to k(\mathsf{x}, \mathsf{x}')$$

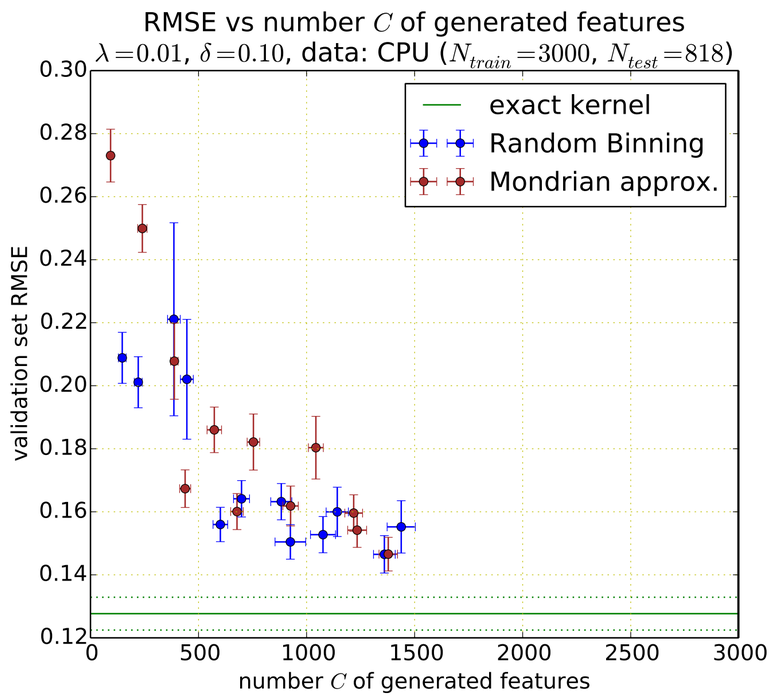


Figure 2: Convergence (in terms of validation set RMSE) to exact kernel regression as number $C$ of generated features increases. Random Binning is one of the random feature kernel approximation schemes proposed in [2].

## Entire regularization path

Instead of recomputing the approximation for several lifetimes (inverse lengthscales) $\lambda$ from scratch, we note

- **decrease** $\lambda \equiv$ remove cuts from Mondrian samples $\equiv$ merge partition cells together $\equiv$ sum features (columns of $\mathbf{Z}$) together
- **increase** $\lambda \equiv$ add cuts to Mondrian samples $\equiv$ split partition cells into two $\equiv$ replace a feature (column of $\mathbf{Z}$) with two new ones

Under these operations, the inverse $(\mathbf{Z}^T\mathbf{Z} + \delta^2\mathbf{I}_C)^{-1}$ can be updated in $\mathcal{O}(C^2)$ time and the resulting regression model evaluated in time $\mathcal{O}(CN)$.

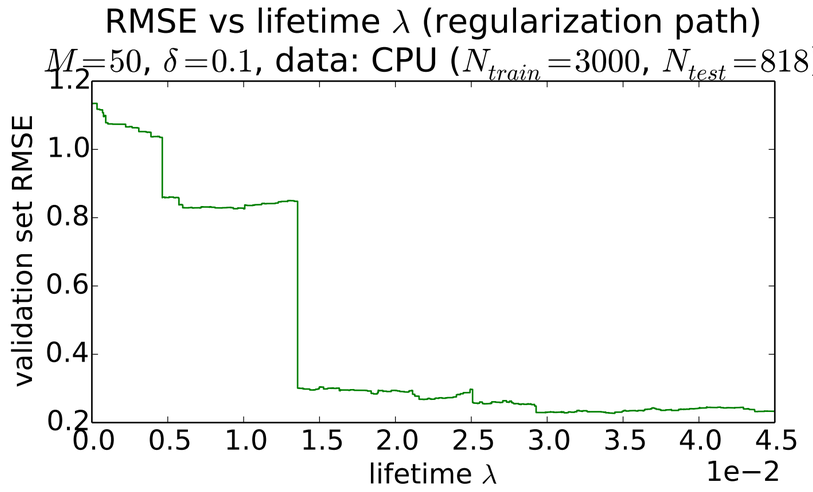


Figure 3: Entire regularization path of Laplace kernel approximation.

Our Laplace kernel approximation can be trained and evaluated for all inverse lengthscale hyperparameter values $\lambda \in [0, \Lambda]$ at essentially the same cost as for just the single value $\Lambda$.

## Mondrian Grid

The (general) **Laplace kernel** is given by

$$k(\mathsf{x}, \mathsf{x}') = e^{-\sum_{d=1}^{D} \lambda_d |x_d - x'_d|}$$

where $\lambda_1, \ldots, \lambda_D$ are inverse lengthscales of the kernel. A Mondrian grid is a collection of $D$ independent one-dimensional Mondrian processes, each running on one of the coordinate axes of $\mathbb{R}^D$.

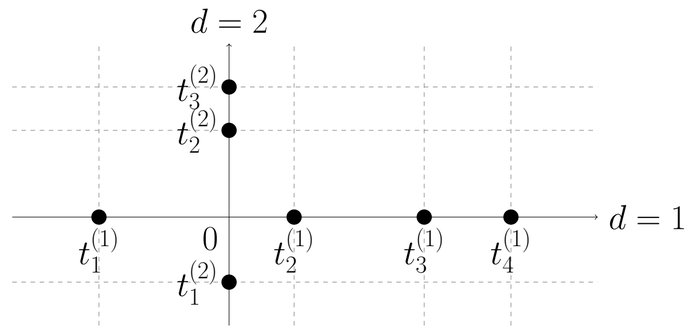


Figure 4: Mondrian grid sample in $2D$. Each cut is associated with a time. Cuts in dimension $d$ have times $t_i^{(d)} \leq \lambda_d$, where $\lambda_d$ is the lifetime of the Mondrian running on the $d$-th coordinate axis.

We use the same feature mapping as before (indicators of partition cells), so that $z(\mathsf{x})^T z(\mathsf{x}') = 1$ if and only if $\mathsf{x}$, $\mathsf{x}'$ fall into the same cell. By independence

$$\mathbb{P}(z(\mathsf{x})^T z(\mathsf{x}') = 1) = \prod_{d=1}^{D} e^{-\lambda_d |x_d - x'_d|} = k(\mathsf{x}, \mathsf{x}')$$

Again we concatenate feature vectors from $M$ independent Mondrian grids to obtain Monte Carlo estimates.

## Lengthscale configuration exploration

Adjust the inverse lengthscale of the approximated kernel in each dimension independently by changing the lifetime of the 1D Mondrian(s) on the corresponding coordinate axis. Amounts to adding or removing cuts from the grid, which translates to adding and removing features from the generated feature space. The resulting regression model can be updated in time $\mathcal{O}(C^2 + CN)$.

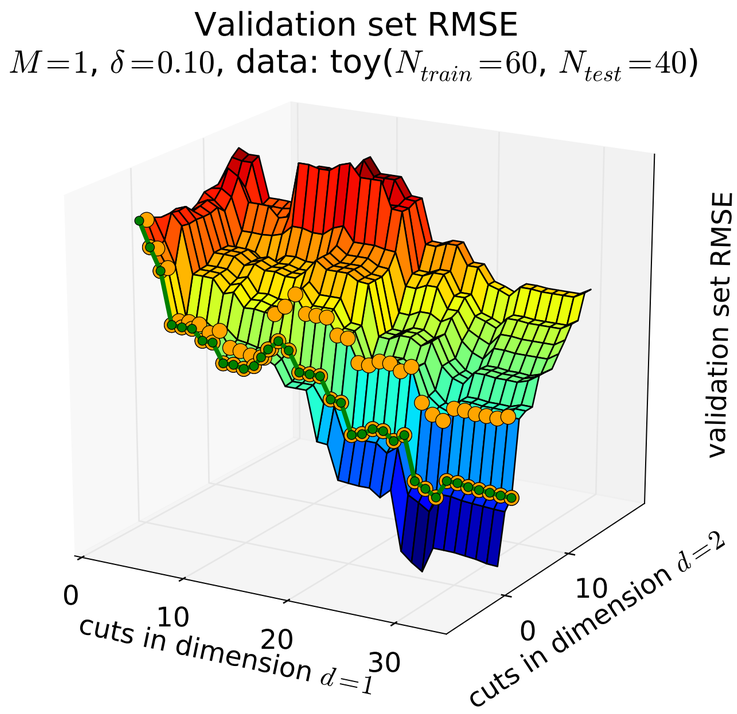


Figure 5: Lengthscale configuration exploration where the second dimension $d = 2$ is irrelevant. This exploration starts with lifetime 0 in all dimensions and at each step, the dimension in which the lifetime is increased is chosen greedily.

## References

[1] Daniel M Roy and Yee Whye Teh.
The mondrian process.
*Adv. in Neural Inform. Processing Syst*, 21:27, 2009.

[2] Ali Rahimi and Benjamin Recht.
Random features for large-scale kernel machines.
In *Advances in neural information processing systems*, pages 1177–1184, 2007.

## Contact

matej.balog@gmail.com