Gaussian Process for Big Data by James Hensman

Korawat Tanwisuth

UT Austin

December 6, 2018

Overview

1. Gaussian Process Overview
2. Computational Issues
3. Solution
   - Introducing pseudo inputs
   - Bound marginal likelihood
   - Stochastic optimization
4. Application

Gaussian Process Overview

Definition: A Gaussian process is a collection of random variables such that any finite number of which have a Gaussian distribution.

A Gaussian process is parametrized by:

$$m(x) \quad \text{(a mean function)}$$

$$k(x, x') \quad \text{(a covariance function)}$$

Gaussian Process Regression Overview

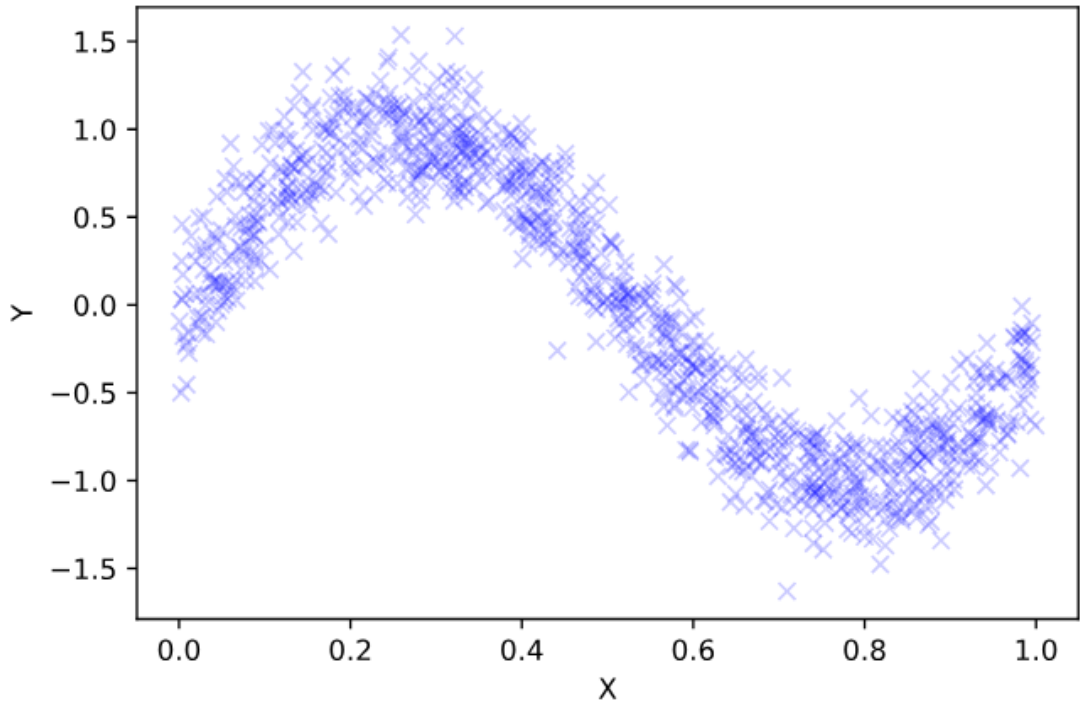
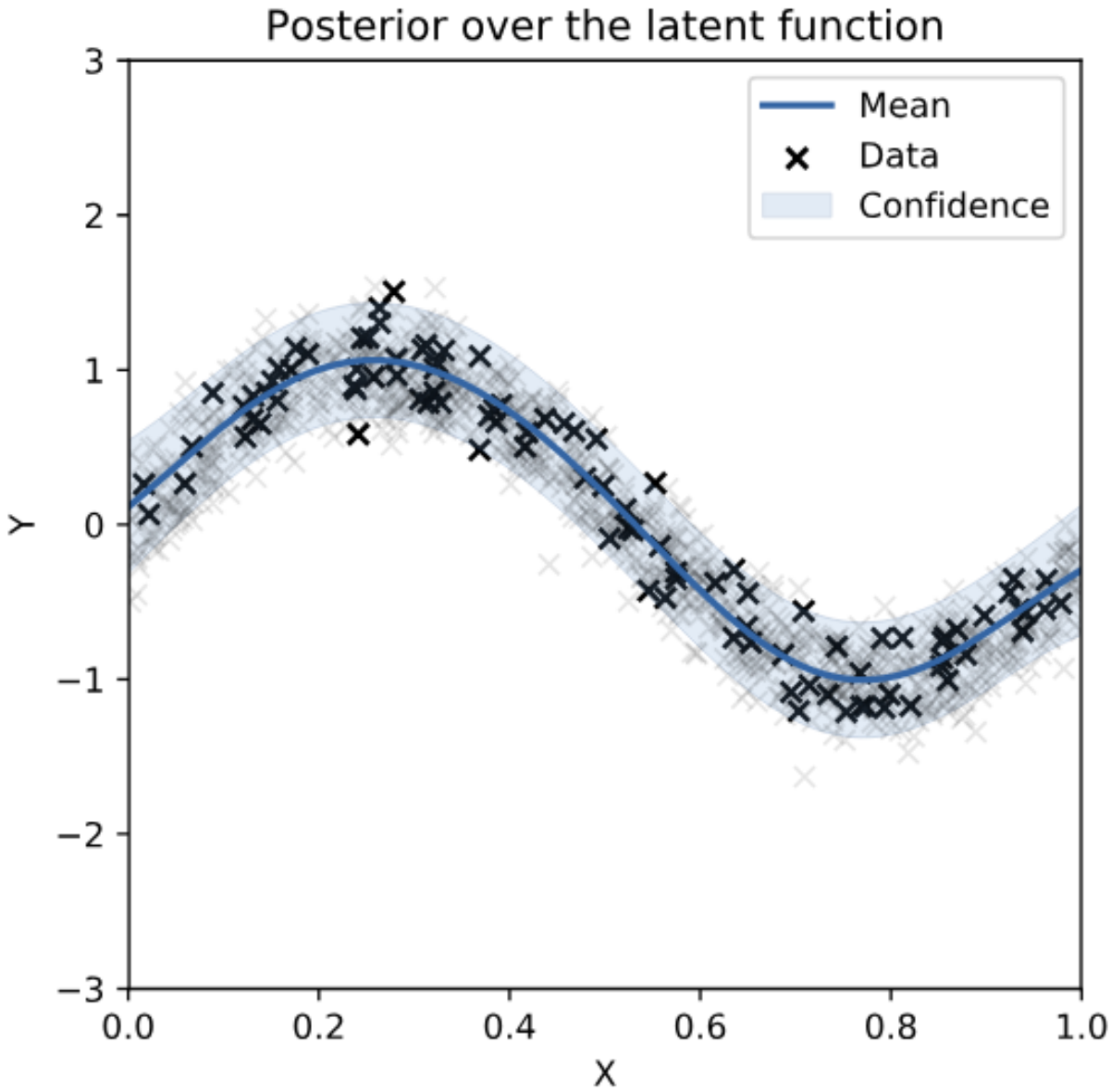


Figure: Noisy observations $y_i = \sin(6x_i) + 0.2\epsilon_i$ where $\epsilon_i \sim \mathcal{N}(0, 1)$

Given a set of input out put $\{x_i, y_i\}_{i=1}^n$, we are interested in finding the posterior $p(\mathbf{f}|\mathbf{y}, \mathbf{X})$.

Gaussian Process Regression Overview



Gaussian Process Regression Overview

To perform inference, we are interested in the quantity:

$$p(\mathbf{f}|\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})}{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})d\mathbf{f}}$$

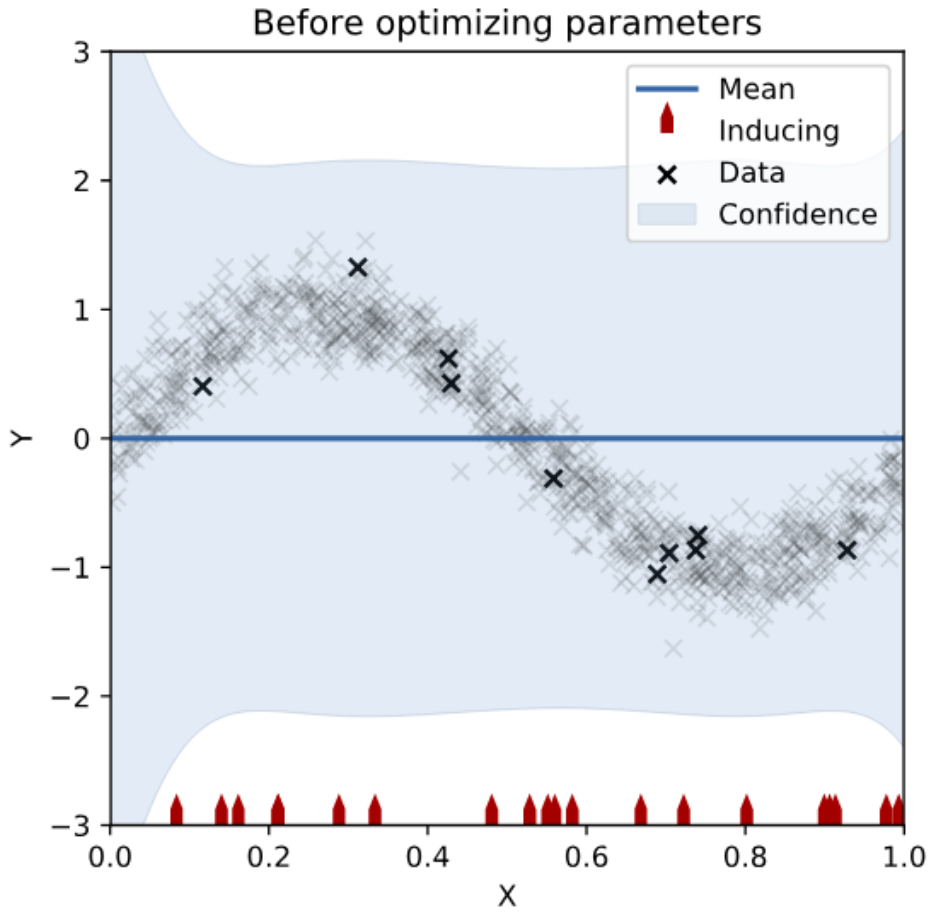
$$\begin{aligned} p(\mathbf{f}|\mathbf{X}) &= \mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K}_{nn}) \\ p(\mathbf{y}|\mathbf{f}) &= \mathcal{N}(\mathbf{y}|\mathbf{f},\sigma^2\mathbf{I}) \\ p(\mathbf{y}|\mathbf{X}) &= \mathcal{N}(\mathbf{y}|\mathbf{0},\mathbf{K}_{nn}+\sigma^2\mathbf{I}) \end{aligned} \tag{1}$$

where $\mathbf{K}_{nn}$ denotes the covariance between our data points. This can grow very large as we obtain more samples.

Computational Issues

Inference with the model involves inverting $\mathbf{K}_{nn}$.
Time complexity: $\mathcal{O}(n^3)$
Storage: $\mathcal{O}(n^2)$

Introducing Pseudo Inputs $\mathbf{Z}, \mathbf{u}$



We now introduce m inducing points $\mathbf{Z}$, which lives in the same space as $\mathbf{X}$. Denote $\mathbf{u}$ the evaluation of f at $\mathbf{Z} = \{z_i\}_{i=1}^m$.

Alternative Posterior

Original posterior:

$$p(\mathbf{f}|\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})}{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})d\mathbf{f}} \tag{2}$$

New posterior:

$$p(\mathbf{u}|\mathbf{y},\mathbf{Z}) = \frac{p(\mathbf{y}|\mathbf{u})p(\mathbf{u}|\mathbf{Z})}{\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u}|\mathbf{Z})d\mathbf{u}} \tag{3}$$

We now turn our attention to this quantity since m, the number of inducing points, is much smaller than n, the number of observations. Still, $p(\mathbf{y}|\mathbf{u})$ involves inverting $\mathbf{K}_{nn}$.

$$p(\mathbf{y}|\mathbf{u}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})}{p(\mathbf{f}|\mathbf{y},\mathbf{u})}$$

Variational Bayes

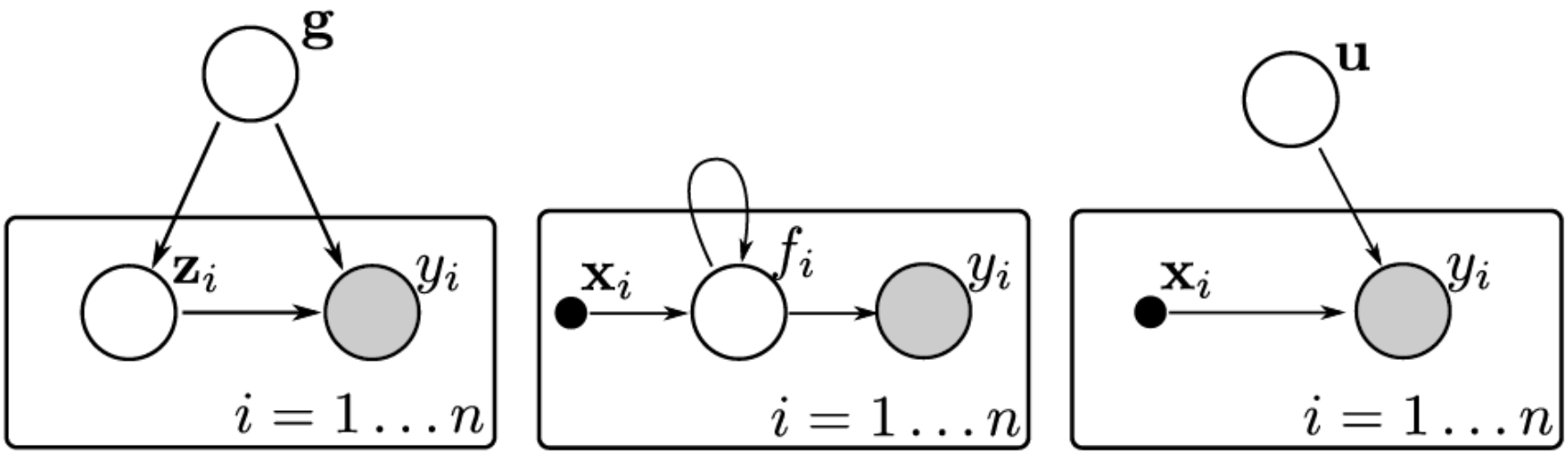
$$\begin{aligned}\ln p(\mathbf{y}|\mathbf{u}) &= \mathbb{E}_{p(\mathbf{f}|\mathbf{u})}[\ln p(\mathbf{y}|\mathbf{f})] + \mathbb{E}_{p(\mathbf{f}|\mathbf{u})}[\ln \frac{p(\mathbf{f}|\mathbf{u})}{p(\mathbf{f}|\mathbf{y},\mathbf{u})}] \\ &= \ln \tilde{p}(\mathbf{y}|\mathbf{u}) + \mathcal{KL}[p(\mathbf{f}|\mathbf{u})||p(\mathbf{f}|\mathbf{y},\mathbf{u})]\end{aligned} \tag{4}$$

With this lower bound $\tilde{p}(\mathbf{y}|\mathbf{u})$, we do not need to invert $\mathbf{K}_{nn}$.

$$\tilde{p}(\mathbf{y}|\mathbf{u}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{y}_i|\mathbf{k}_{mn}^T\mathbf{K}_{mm}^{-1}\mathbf{u}, \sigma^2) \exp\{-\frac{1}{2\sigma^2}(\mathbf{k}_{nn} - \mathbf{k}^T\mathbf{K}_{mm}^{-1}\mathbf{k}_{mn})\}$$

Lower bound on marginal likelihood

With this new lower bound, if we use $\ln \tilde{p}(\mathbf{y}|\mathbf{u})$ to obtain a lower bound for $\log p(\mathbf{y}|\mathbf{X})$ by marginalizing $\mathbf{u}$. We get a complexity $\mathcal{O}(nm^2)$. This still depends on n. Instead, we will treat $\mathbf{u}$ as variational parameter.



(a) Requirements for SVI (b) Gaussian Process regression (c) Variational GP regression

Final objective function

$$\log p(\mathbf{y}|\mathbf{X}) \geq \mathbb{E}_{q(\mathbf{u})}[\log \tilde{p}(\mathbf{y}|\mathbf{u})] - \mathcal{KL}(q(\mathbf{u})||p(\mathbf{u})) = \mathcal{L}$$

Now, we are interested in finding the variational distribution $q(\mathbf{u})$ where $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$.

$\mathcal{L}$ depends on:

- parameters of the variational distribution $q(\mathbf{u})$
- location of inducing inputs $\mathbf{Z}$
- parameters of the covariance function

Optimizing $\mathbf{Z}$ can be computationally expensive so we suggest performing $\mathcal{K}$-means clustering on $\mathbf{X}$ and use the centroids as $\mathbf{Z}$.

Stochastic Optimization

Algorithm:

- Take a mini-batch of data to compute noisy estimate of the gradient
- Move in the direction of the gradient where step size is controlled by learning rate
- Stop when convergence criteria is met (ie. number of iterations or change in objective function)

Natural Gradient

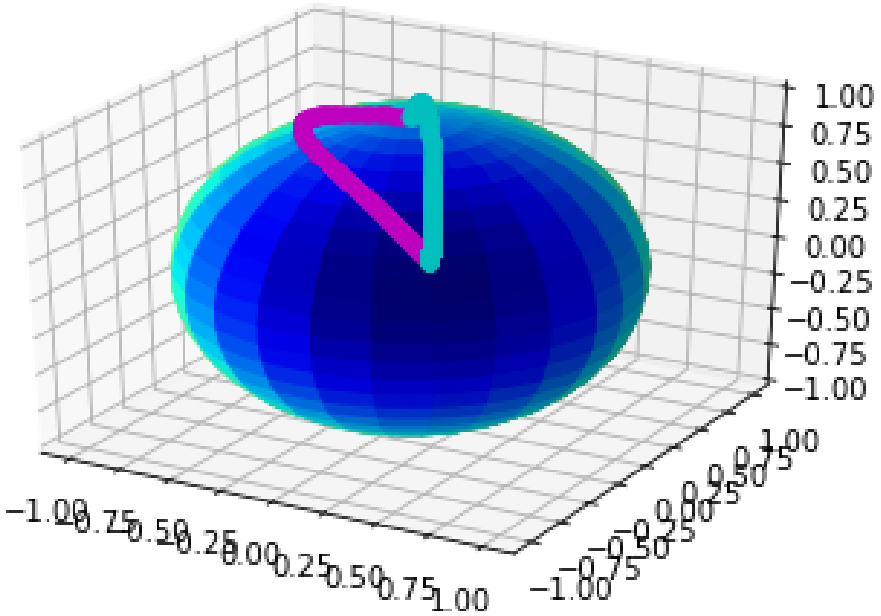


Figure: The blue line shows the path using a natural gradient and the purple line depicts the path using a regular gradient.

$$\tilde{\mathrm{g}}(\boldsymbol{\theta}) = G(\boldsymbol{\theta})^{-1} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\eta}}$$

where $\boldsymbol{\theta}$ is the canonical parameter and $\boldsymbol{\eta}$ is the expectation parameter of exponential family.

Application

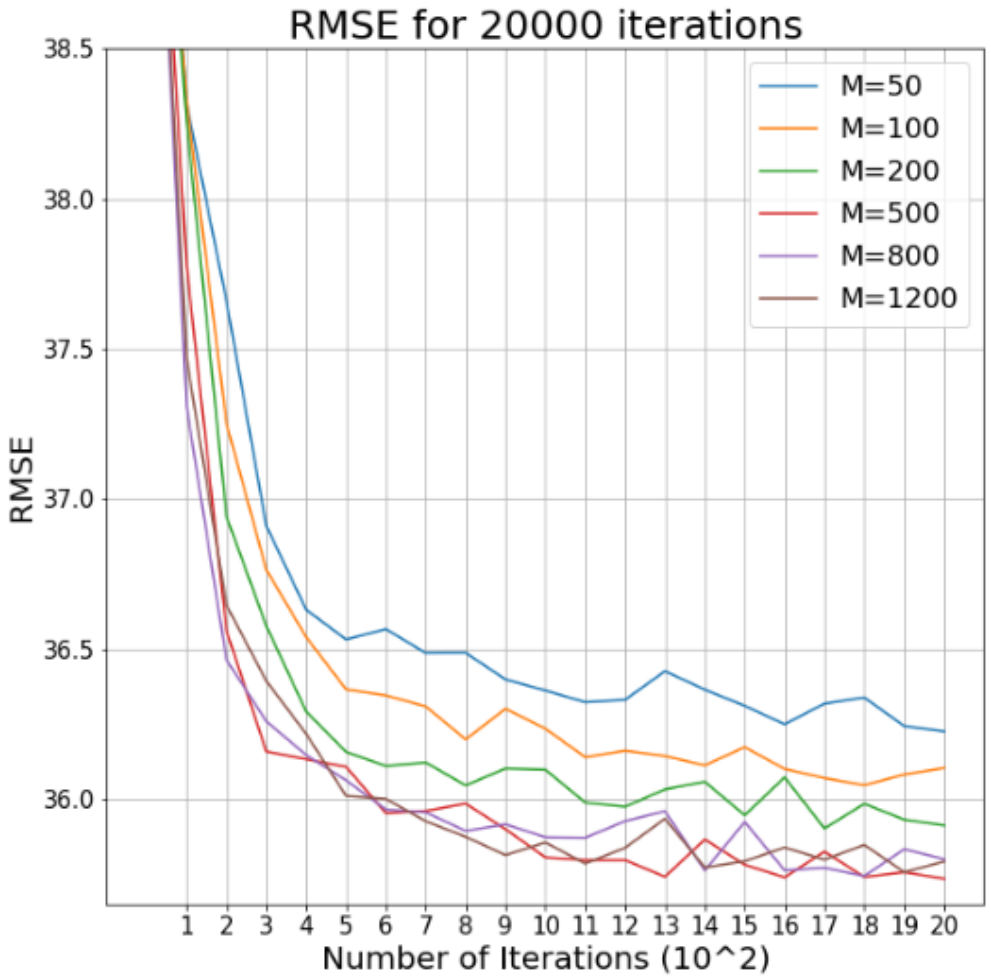
Data:

- Flight arrivals and departures in 2015 from DOT's monthly Air Travel and Consumer Report
- 5,714,008 rows, 31 columns
- Target variable (Flight delays)

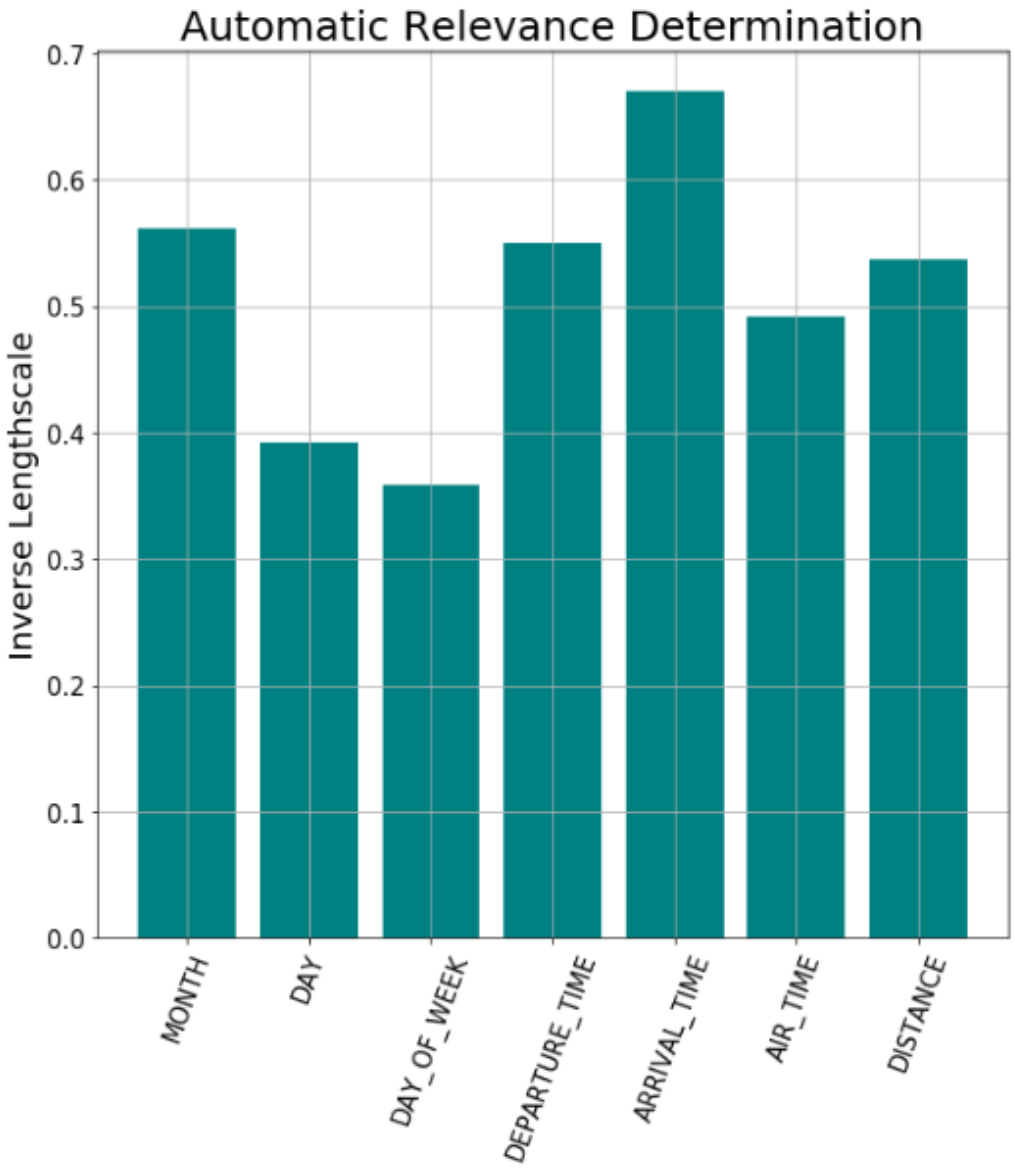
Model:

- 7 predictors (Month, Day of the month, Day of the week, Airtime, Arrival time, Departure time, Distance that needs to be covered)
- 800,000 (700,000 for training and 100,000 for testing)
- m = 50, 100, 200, 500 , 800, 1200 inducing points
- Momentum = 0.9, Batch-size = 100, Learning rate = 0.1

Result


RMSE for 20000 iterations
RMSE
38.5
38.0
37.5
37.0
36.5
36.0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Number of Iterations (10^2)
M=50
M=100
M=200
M=500
M=800
M=1200

Automatic Relevance Determination



Conclusion

- We introduce inducing points to help approximate the posterior of the latent function.
- We treat $\mathbf{u}$ as variational parameters to find the variational distribution $q(\mathbf{u})$ that minimizes the lower bound to the log marginal.
- We use stochastic optimization to find the optimal parameters.

Result: Complexity reduces to $\mathcal{O}(m^3)$, which is independent of n.