Deep Neural Networks & Gaussian Processes

NNPDF, Varenna 2019

Parametric learning *vs* non-parametric learning

Problem

Given a training set (x_i, y_i) and a new $\bar{x}_a$ find $\bar{y}_a$

Ideally, find $P(\bar{y}_a | (x_i, y_i), \bar{x}_a)$

Parametric Learning

$$\bar{y}_a = f(\boldsymbol{w}, \bar{x}_a)$$

Fit a function
Functional bias

Non-Parametric Learning

$$\bar{y}_a = \sum_m \alpha_m \kappa(x_m, \bar{x}_a)$$

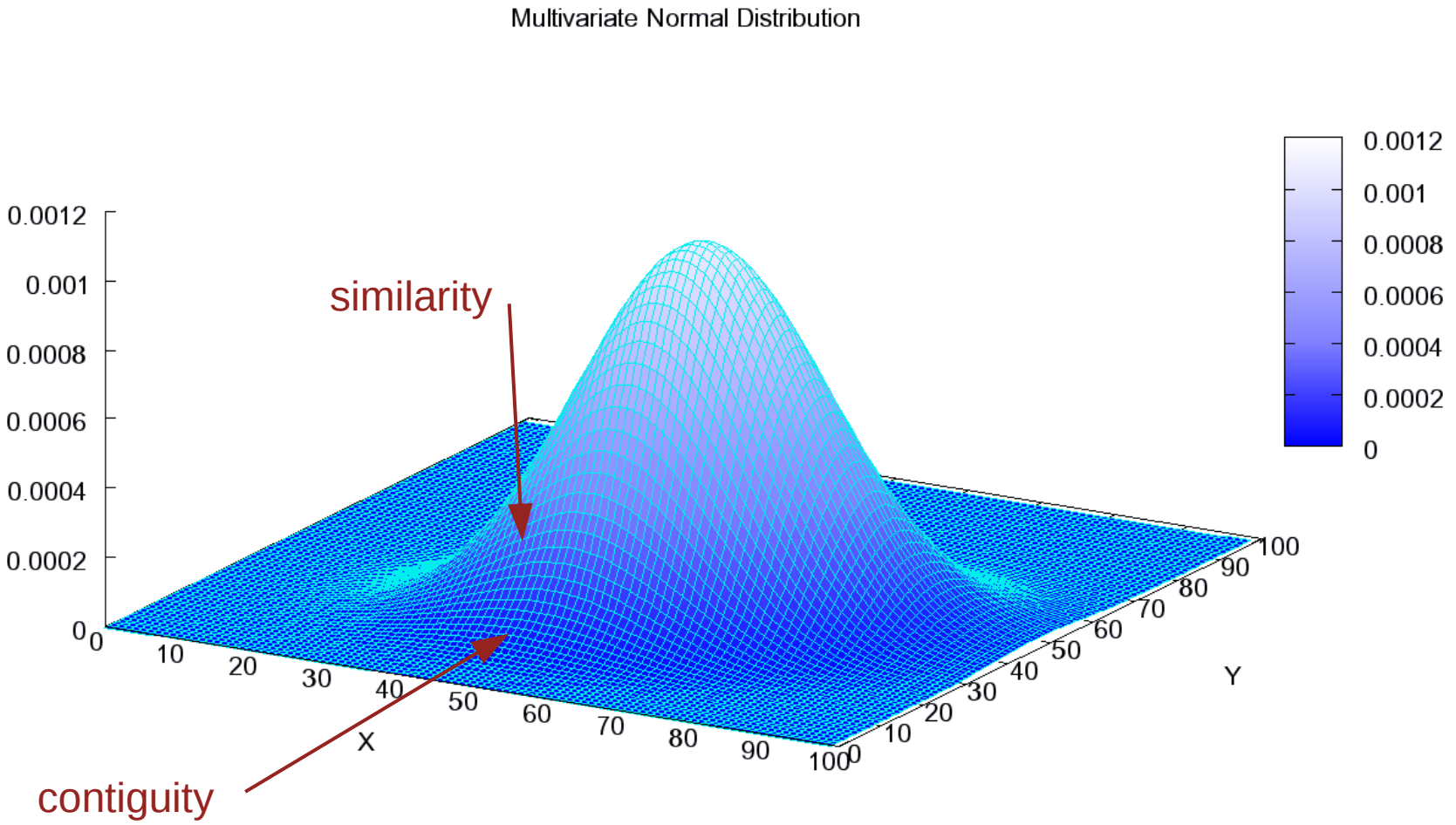
Guess by proximity
Computationally hard

Gaussian Process

Gaussian Process

Instead of dealing with a space of infinitely many functionalities

Gaussian Process assumes $(x_1, x_{2,..}, x_n) \rightarrow P(f(x_1), f(x_2), ..., f(x_n))$ is jointly gaussian



Ex:GP, dictated by kernel (correlation)

$$\bar{y}_a = \sum_m \alpha_m \kappa(x_m, \bar{x}_a)$$

Similarity between new data and training set

Fix α_m as $min_\alpha(\sum_m (\bar{y}_m - y_m)^2 + \lambda \sum_m \alpha_m^2)$

Exact solution: $\alpha = (K + \lambda I)^{-1} y$ $K_{mn} = \kappa(x_m, x_n)$

Prediction: $\bar{y} = y^T (K + \lambda I)^{-1} \kappa(x, \bar{x})$

The choice of the kernel $\kappa(x_m, x_n)$ is a prior and hides a bias from hyperparameters

e.g. $\kappa(x_m, x_n) = \sigma^2 e^{\frac{-(x_m - x_n)^2}{2l^2}}$

GP prediction: any new data must comply with a multivariate normal

$$(x_1, x_{2,..}, x_n) \rightarrow P(f(x_1), f(x_2), \ldots, f(x_n))$$ jointly normal

$$(x_1, x_{2,..}, x_n, x^*) \rightarrow P(f(x_1), f(x_2), \ldots, f(x_n), f(x^*))$$ assumption: also jointly normal

$$\begin{pmatrix} K_{D,D} + \sigma^2 I & K_{D,*} \\ K_{D,*}^T & K_{**} \end{pmatrix} \qquad K = \langle (f(x_n) - \mu_n)(f(x_m) - \mu_m) \rangle$$

Exact Bayesian update (Bayesian Regression O'Hagan 1978): max marginal likelihood

$$P(y|x) = -\frac{1}{2} y^T (K + \sigma^2 I)^{-1} y - \frac{1}{2} \log det (K + \sigma^2 I) - \frac{n}{2} \log (2\pi)$$

$$P(f^*|D, x^*) \sim N(\bar{\mu}, \bar{K})$$

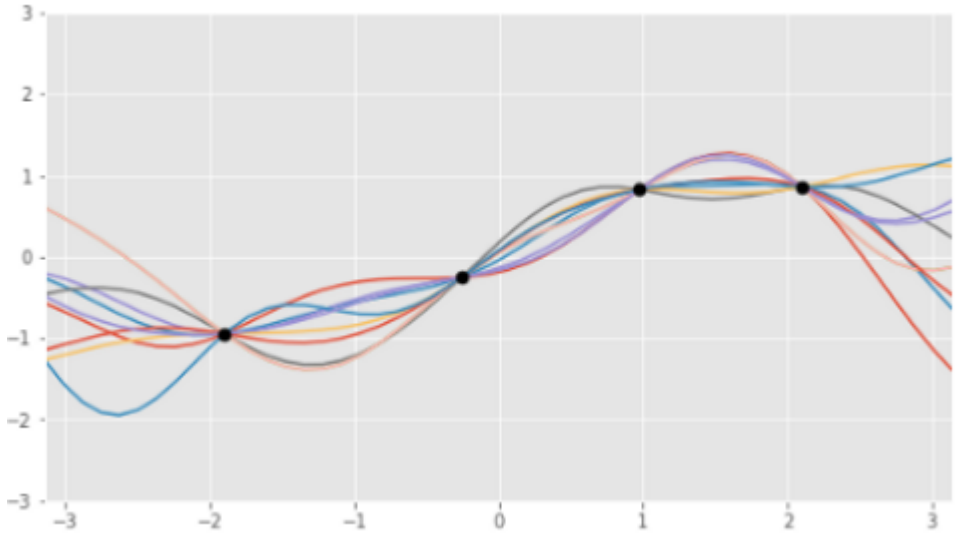
$$\bar{\mu} = K_{D,*} K_{D,D}^{-1} f$$

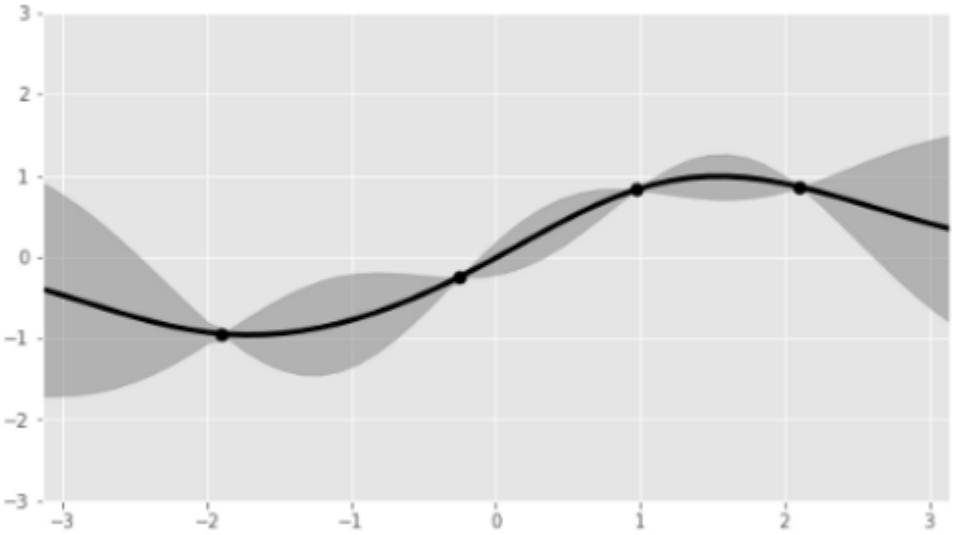
$$\bar{K} = K_{**} - K_{D,*} (K_{D,D} + \sigma^2 I)^{-1} K_{D,*}^T$$

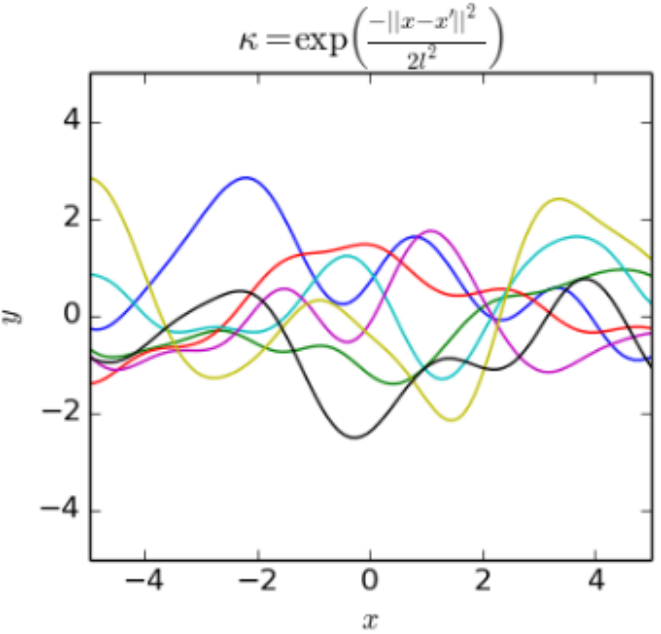
$$\bar{f}^* = \mu^* + K_{*x} (K_{D,D} + \sigma^2 I)$$

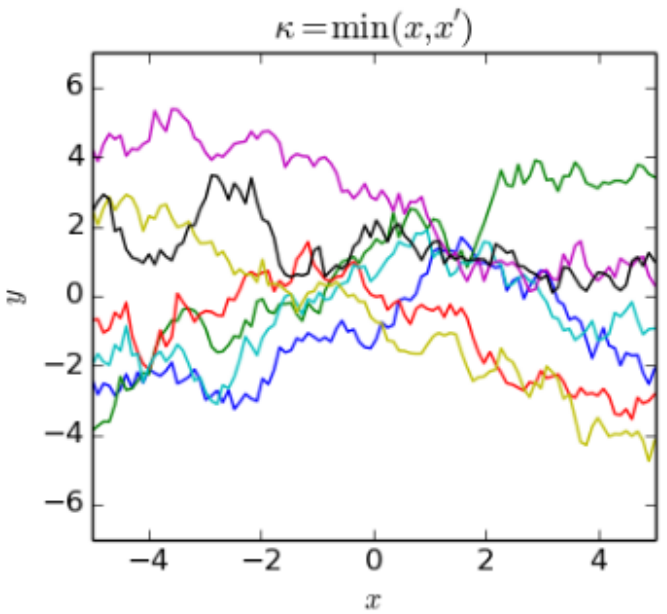
Major drawback: hard computation, scaling as N^3 for N data points

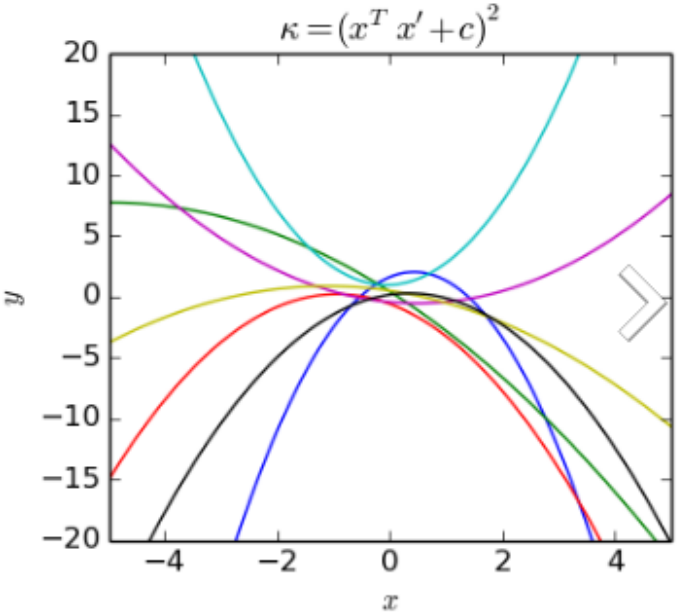
GP delivers a gaussian spread for every point











Wiggles are related the choice of kernel,
that is, to the correlation length dictated by the kernel (covariance matrix)

Kernel, inherits the concept of contiguity

$$\kappa(x_m, x_n) = \kappa(d(x_n, x_m))$$

The kernel translates the sense of proximity
from the original space to the target space

In the extrapolation distance,
the loss of contiguity produces an increase of spread

Neural Networks as GP

Why Neural Networks work so well?

Theorem (1989): **A NN with an infinite hidden layer is a universal approximant**

$$x_k$$

$$x_j^1(x) = g\left(\sum_k w_{jk}^0 x_k\right)$$

$$z_i(x) = \sum_j w_{ij}^1 x_j^1(x)$$

Harmonic Analysis: Carleson Theorem (1966)
Adapted to NN: Cybenko 1989, Funahashi 1989, Hornik, Stinchcombe, and White 1989

$$z_i(x) = \sum_j \alpha_j \phi(w_{ji} x_i)$$

The activation function needs to be nonlinear and bounded

Convergence to any function point-by-point

Why Neural Networks work so well?

Theorem (RM Neal, 1994): **A NN with a infinite random hidden layer is a GP**

NN

x_k

$x_j^1(x) = g\left(\sum_k w_{jk}^0 x_k\right)$

$z_i(x) = \sum_j w_{ij}^1 x_j^1(x)$

w_{jk}^0, w_{ij}^1 i.i.d.

Central limit theorem

$z_i \sim GP\left(\mu^1, K^1\right)$

$\mu^1 = 0$

$K^1 = E\left(z(x), z(x')\right)$ average over all w_{jk}^0, w_{ij}^1

Theorem (2018): **A Deep NN is a GP**

Stack of GPs. Limiting *K* is constant.

Conclusion

(Rasmussen & Williams, *Gaussian Process for Machine Learning, 2006*)

A Deep NN approaches a GP, which are non-parametric, most functionality unbiased

Training of NN provide a refinement of w, that is a better definition of contiguity

Predictions of NN comply with “bayesian” philosophy

NN are easier to train than using straight GP for very large data sets

NNPDF should not be afraid of large NN

A penalty term for weights might be revisited (less wiggles)

Lack of spreading in the extrapolation region = excessive contguity ?

On hierarchical models

posterior likelihood prior

$$P(w|D,M)=\frac{P(D|w,M)P(w|M)}{P(D|M)}$$

M model
D data
w parameters

marginal likelihood

$$P(D|M)=\int P(D|w,M)\,P(M|w)\,dw$$

Hyperprior $P(\alpha|M)$

Prior $P(w|\alpha,M)$

Likelihood $P(D|w,M)$

$$P(w|D.M\ \alpha)=\frac{P(D|w,M)P(w|\alpha,M)}{P(D|\alpha,M)}$$ Infer parameters

$$P(\alpha|D,M)=\frac{P(D|\alpha,M)P(\alpha|M)}{P(D|M)}$$ Infer hyperparameters

$$P(M|D)\sim P(D|M)P(M)$$ Infer model

Parametric

$$P(y|D) = P(y|w)P(w|D)$$

Non-parametric

$$P(y|\alpha, D)$$