Bayesian Approach for Inverse Problems

L Del Debbio

Higgs Centre for Theoretical Physics
University of Edinburgh

work in collaboration with

A Candido, A Chiefa, T Giani, A Lupo, M Panero, G Petrillo, N Tantalo, M Wilson

inverse problems

inverse problems are known to be ill-defined

$$\boxed{y_I = T_I[f]}$$

result depends on assumptions

Many questions asked at this workshop are more easily answered if we can build a common framework

Bayesian approach: framework to **understand and compare**

knowledge of f is encapsulated in the posterior distribution [talk by Aleksander]

$$\tilde{p}(f) = p(f|y) = \frac{p(y|f)p(f)}{p(y)}$$

likelihood & loss function

$$p(y|f) \propto \exp\left[-\mathcal{L}(y,f)\right]$$

$$\begin{aligned}
\mathcal{L}(y,f) &= \frac{1}{2}\sum_{I,J}(y_I - T_I)(C_Y^{-1})_{IJ}(y_J - T_J) \\
&= \frac{1}{2}(y-T)^T C_Y^{-1}(y-T) \\
&= \frac{1}{2}||y-T||_{C_Y}^2
\end{aligned}$$

all assumptions about the function f are in the **prior** $p(f)$

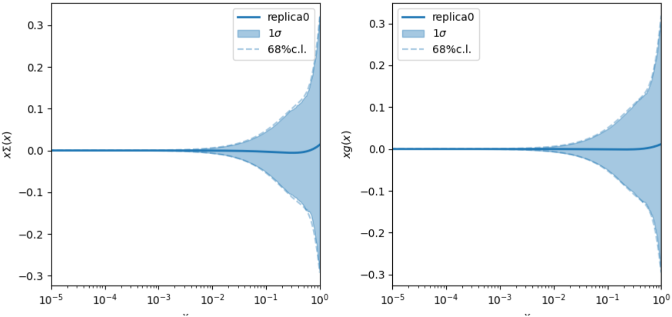
- for GP the prior covariance K dictates the fluctuations of f
- for parametrized functional forms $f(x;\theta)$

$$p(f) = \int d\theta p(\theta) \prod_x \delta\left(f(x) - f(x;\theta)\right)$$

  true for NN + fixed functional forms [talk by Aurore]

- the solution will depend on the prior [talk by Tommaso]
- NNPDF methodology: MC sampling of $\tilde{p}$ (and p) [talk by James]

NNPDF at initialization - distribution over replicas



(almost) Gaussian Process with covariance determined by the architecture of the NN [hyperoptimization in Juan's talk]

expected behaviour for large n (width of the network) how is $\tilde{p}(f)$ estimated

- GP: MonteCarlo sampling of the posterior
  [talks by Tommaso/Aleksander/James/Mark]

- fixed functional form

$$\mathcal{L} = \mathcal{L}_* + \frac{1}{2} H_{\mu\nu} \delta\theta_\mu \delta\theta_\nu$$

- NNPDF: flow towards minimum of $p(y|f)$, stop training after T epochs

$$\tilde{p}(f) = \int df' p(f') \delta\left(f - f_T\right)$$

gradient descent - for all parametrizations

$$\frac{d}{dt}\theta_\mu = -\nabla_\mu \mathcal{L}$$

$$\nabla_\mu \mathcal{L} = -(\nabla_\mu f_t)^T \left(\frac{\partial T}{\partial f}\right)_t^T C_Y^{-1} \epsilon_t\,, \quad \epsilon_t = y - T[f_t]$$

$$\frac{d}{dt} f_t = (\nabla_\mu f_t) \frac{d}{dt}\theta_\mu = \Theta_t \left(\frac{\partial T}{\partial f}\right)_t^T C_Y^{-1} \epsilon_t$$

where

$$\Theta_t = (\nabla_\mu f_t)(\nabla_\mu f_t)^T$$

is the Neural Tangent Kernel

for linear data:

$$y = (\mathrm{FK})f \quad \Longrightarrow \quad \left(\frac{\partial T}{\partial f}\right) = (\mathrm{FK})$$

for wide neural networks

$$\Theta_t = \Theta + O(1/n)$$

hence we get a linear equation for f_t

$$\begin{aligned}\frac{d}{dt} f_t &= \Theta(\mathrm{FK})^T C_Y^{-1} \left(y - (\mathrm{FK}) f_t\right) \\ &= -\Theta M f_t + b\end{aligned}$$

- the rate at which features are learned is dictated by the eigenvalues/eigenvectors of Θ
- there is a strong hierarchy in the eigenvalues (spectral bias)
- consider for simplicity the case

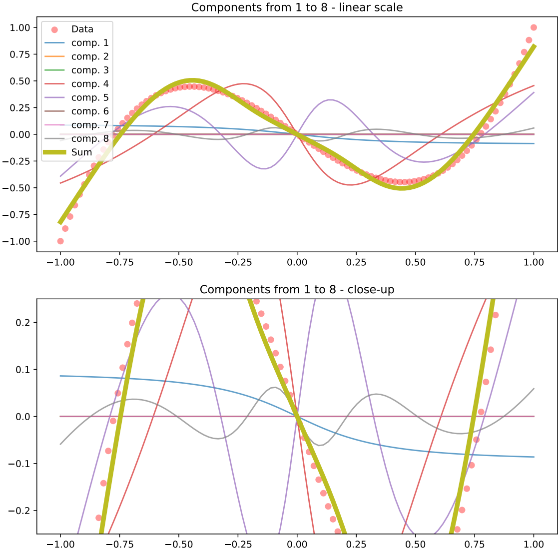
$$C_Y = 0\,, \quad (\mathrm{FK}) = 1$$

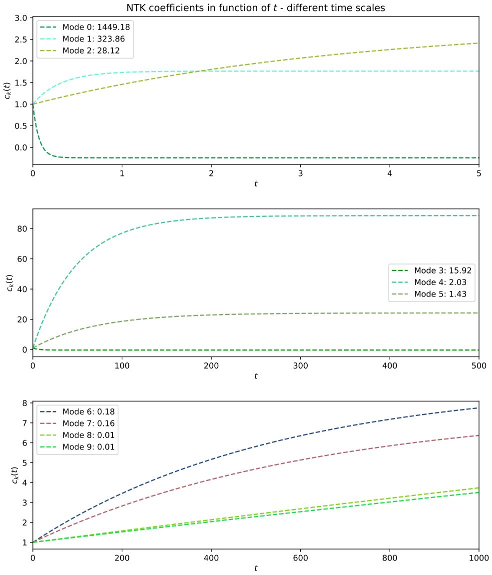
then

$$f_{t,\mathbf{x}^*} = f_{0,\mathbf{x}^*} + \Theta_{\mathbf{x}^*\mathbf{x}} \Theta_{\mathbf{x}\mathbf{x}}^{-1} \left(1 - e^{-\Theta t}\right) \left(y - f_{0,\mathbf{x}}\right)$$

coincides with GP posterior if $\Theta = K$ and $t \to \infty$

- however $t \to \infty$ is what we would call an overfitted solution, $\epsilon = 0$


Components from 1 to 8 - linear scale
Data
comp. 1
comp. 2
comp. 3
comp. 4
comp. 5
comp. 6
comp. 7
comp. 8
Sum
Components from 1 to 8 - close-up


NTK coefficients in function of t - different time scales
Mode 0: 1449.18
Mode 1: 323.86
Mode 2: 28.12
Mode 3: 15.92
Mode 4: 2.03
Mode 5: 1.43
Mode 6: 0.18
Mode 7: 0.16
Mode 8: 0.01
Mode 9: 0.01
$c_k(t)$
t

a few more thoughts

- fixed form parametrization do not have a t-independent NTK

- a puzzling property
$$\mathrm{tr}\,\Theta = \mathrm{tr}\,H$$

- length of training: validation set, analytical control?
model dependence $\ll$ uncertainty

- correlations between PDF fits
$$p(f_{F1}, f_{F2}|y) = \frac{p(y|f_{F1}, f_{F2})\, p(f_{F1}, f_{F2})}{p(y)}$$

conclusions

- bayesian analysis offers an independent tool to look at inverse problems
- all hypotheses are explicitly spelled out in the prior
- for linear data, we get analytical results useful to build intuition
- understand the training better
- comparison with other methods
- robust errors