Gaussian processes

Theory and applications in predictive modeling of spatiotemporal phenomena

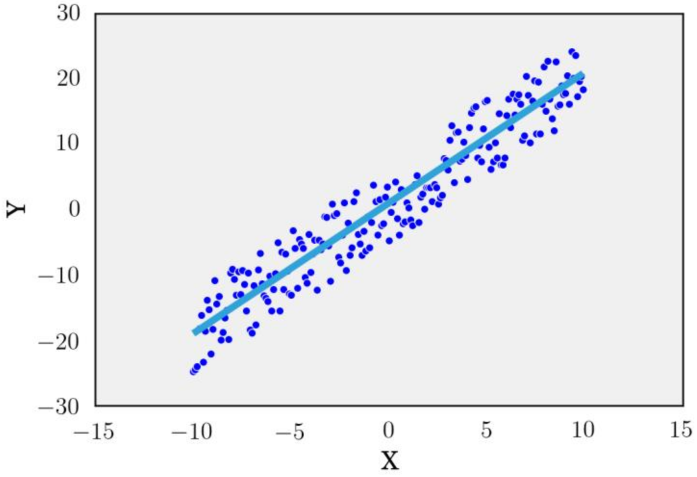
Martin Asenov

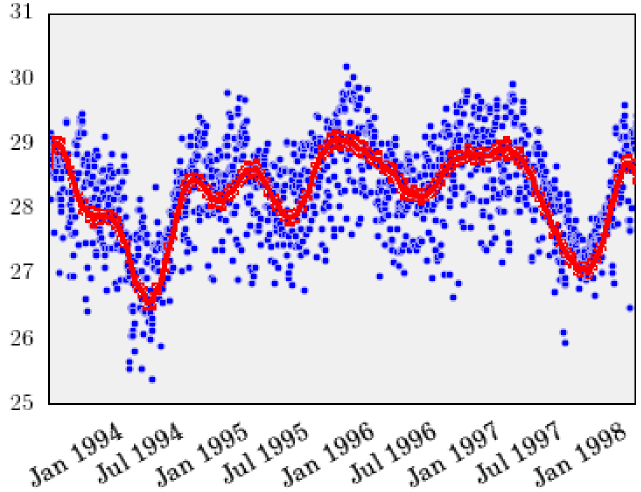
Supervised by Dr. Subramanian Ramamoorthy

Outline

- **Gaussian Processes for nonlinear regression**
  - Gaussian distribution – univariate and multivariate
  - Definition of Gaussian processes
  - Inference from data
  - Two-dimensional input space
- **Spatiotemporal phenomena**
  - Definition and applications
  - Modelling with Gaussian Processes
- **Summary and questions**

Regression



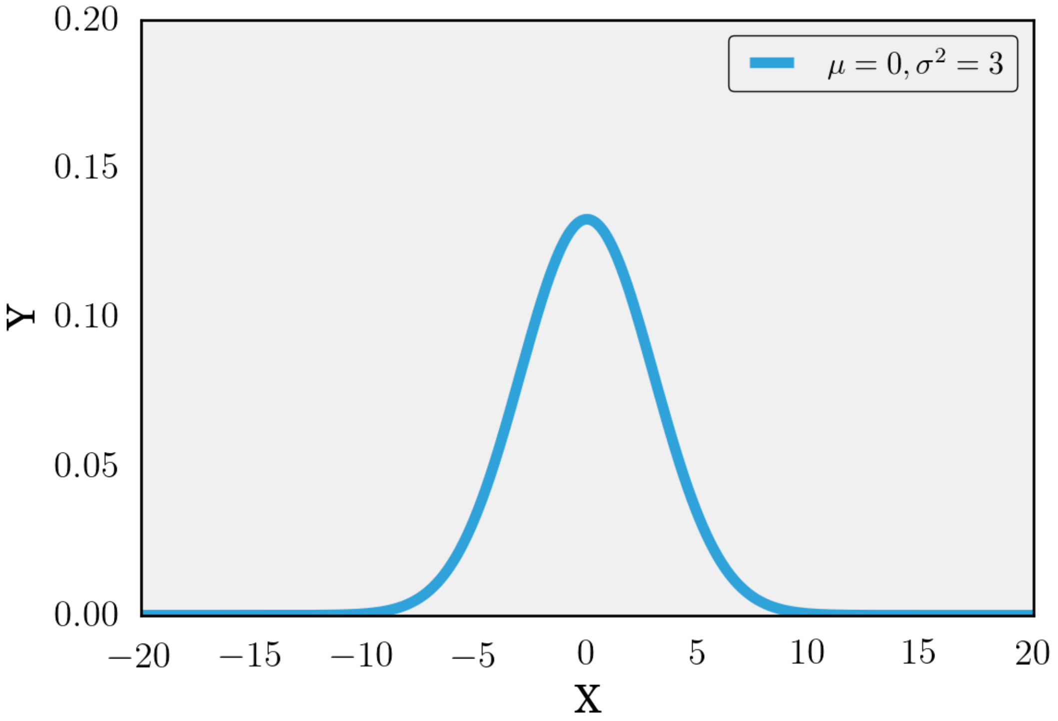


$$\{x_n, y_n\}_{n=1}^{N}$$

Can we do interesting machine learning problems (eg. Nonlinear regression) using only a Gaussian distribution?

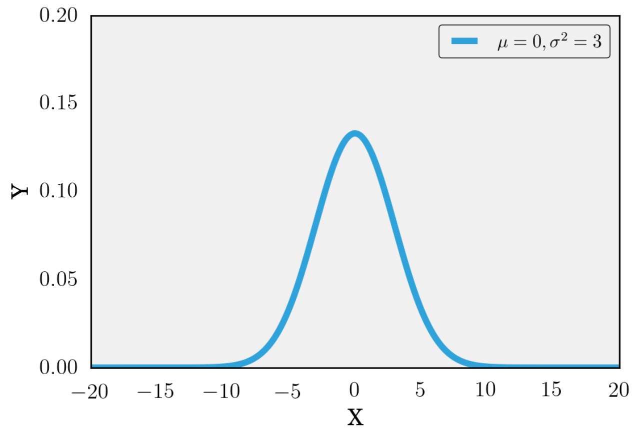
And what are the benefits?

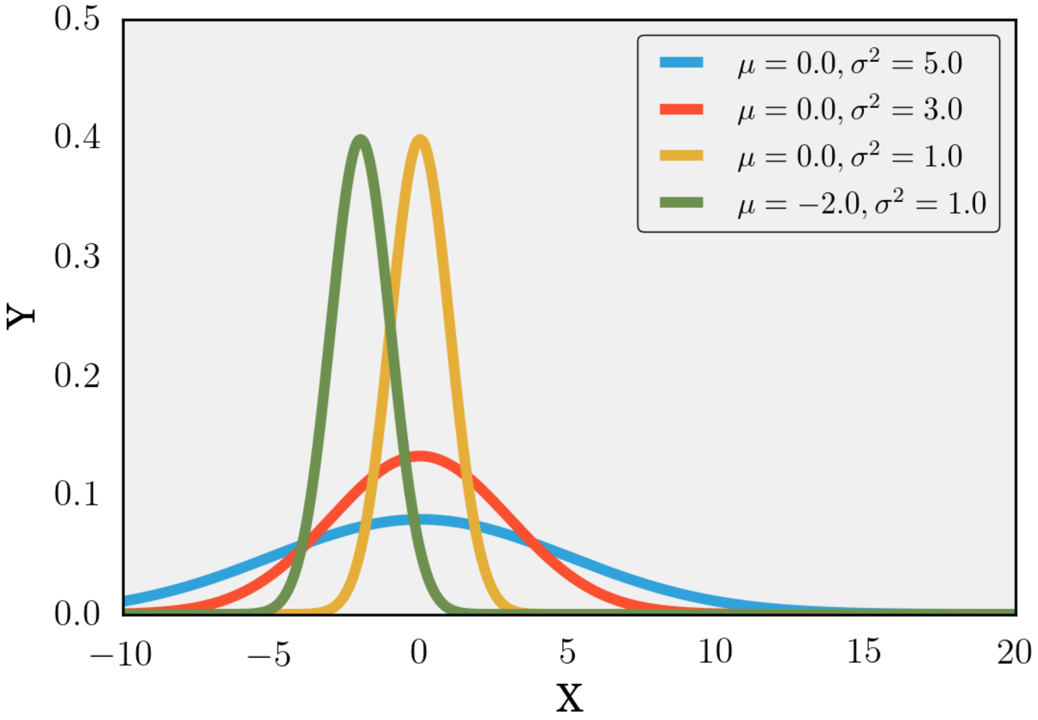
Gaussian distribution



$$N(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

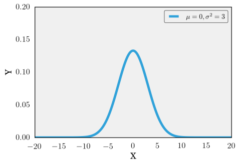
Gaussian distribution



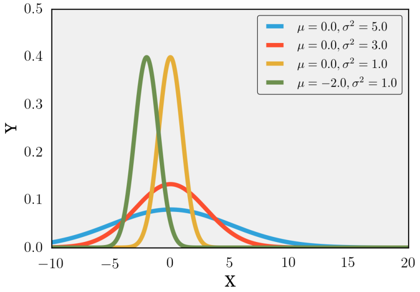


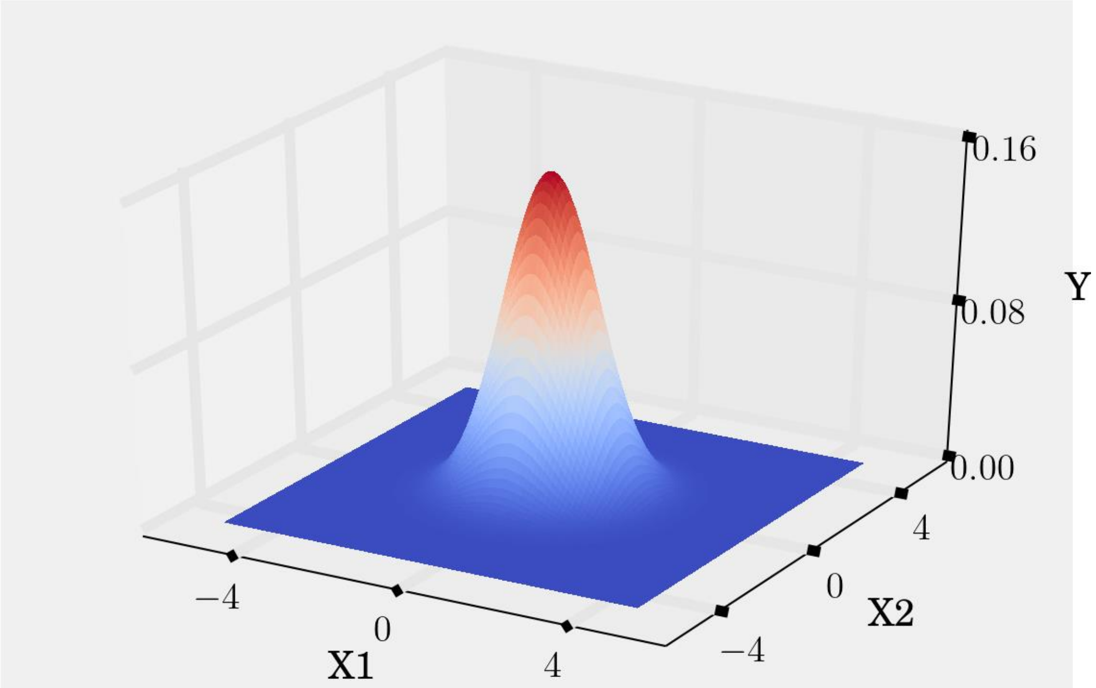
$$N(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Gaussian distribution



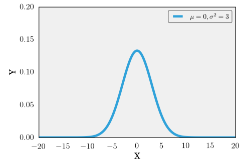
$$N(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



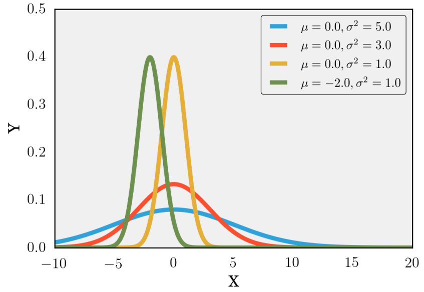


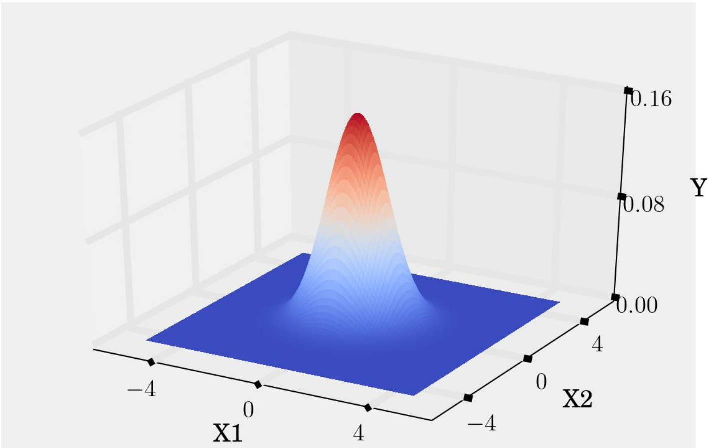
$$N(x|\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

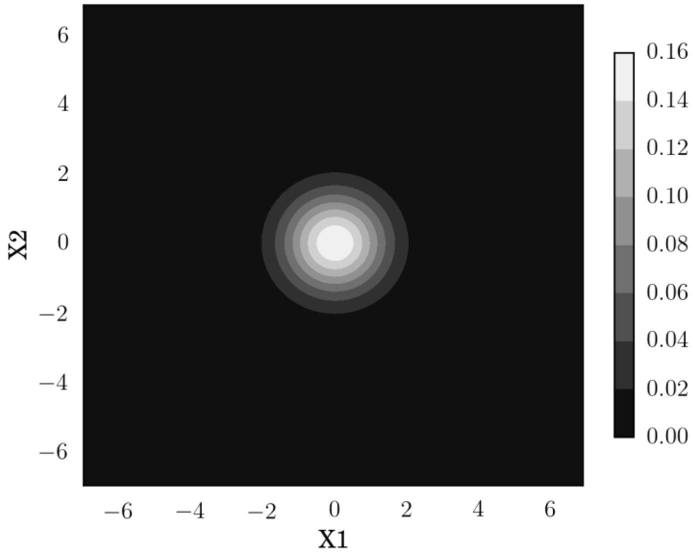
Gaussian distribution



$$N(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

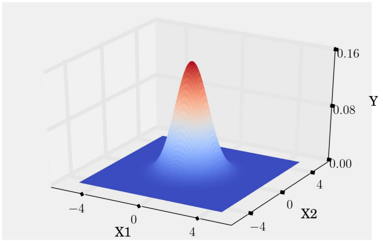


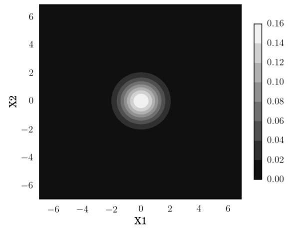




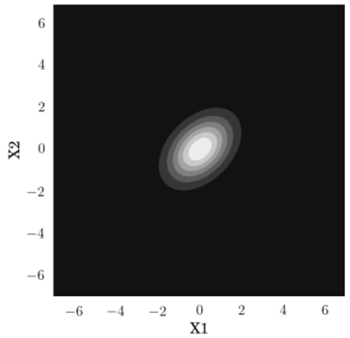
$$N(x|\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

Gaussian distribution

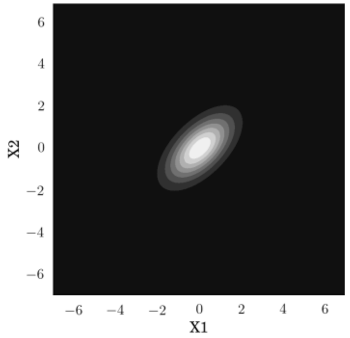




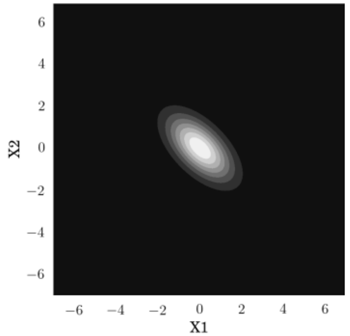
$$N(x|\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$



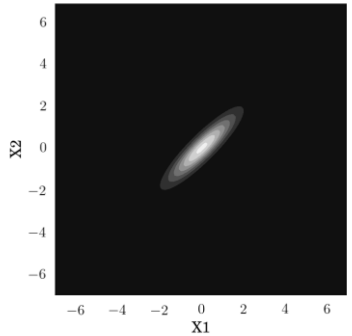
$$\Sigma = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 1 \end{bmatrix}$$



$$\Sigma = \begin{bmatrix} 1 & 0.6 \\ 0.6 & 1 \end{bmatrix}$$

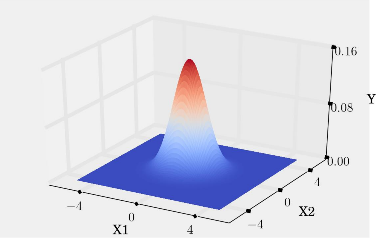


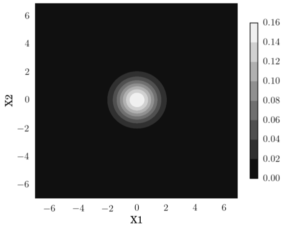
$$\Sigma = \begin{bmatrix} 1 & -0.6 \\ -0.6 & 1 \end{bmatrix}$$



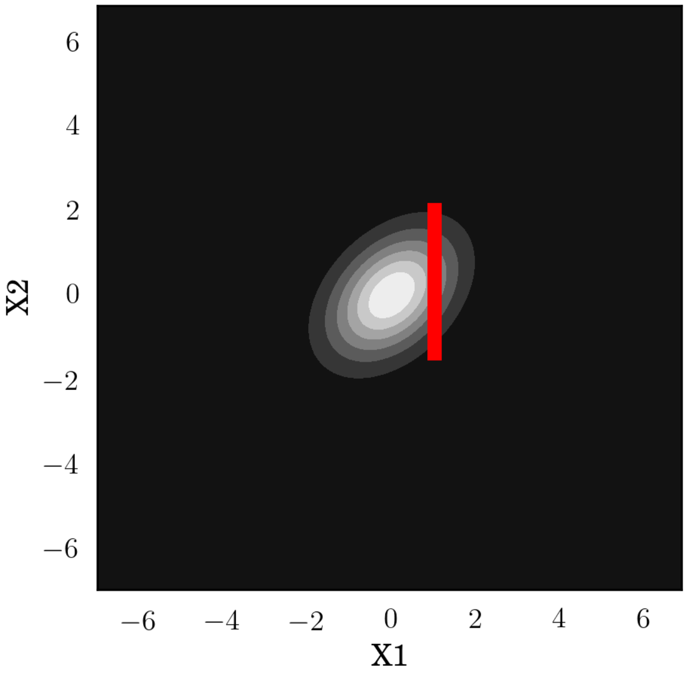
$$\Sigma = \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}$$

Gaussian distribution

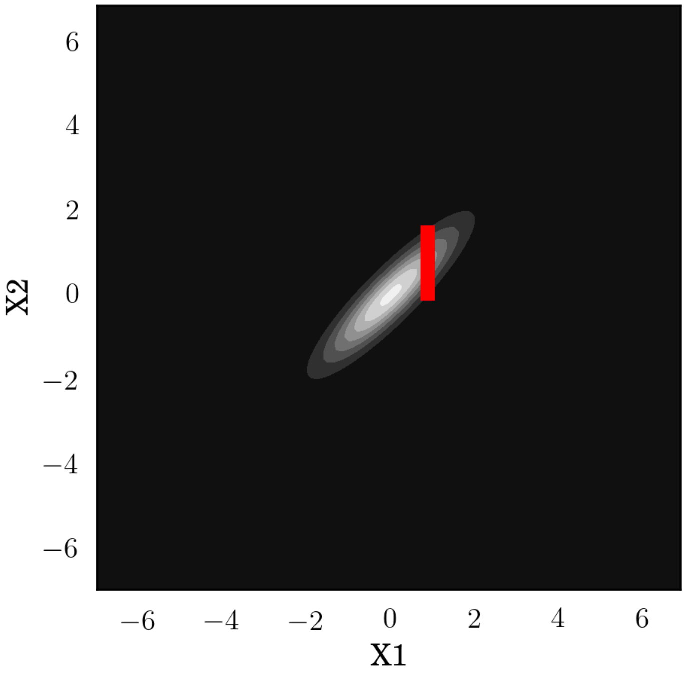




$$N(x|\mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

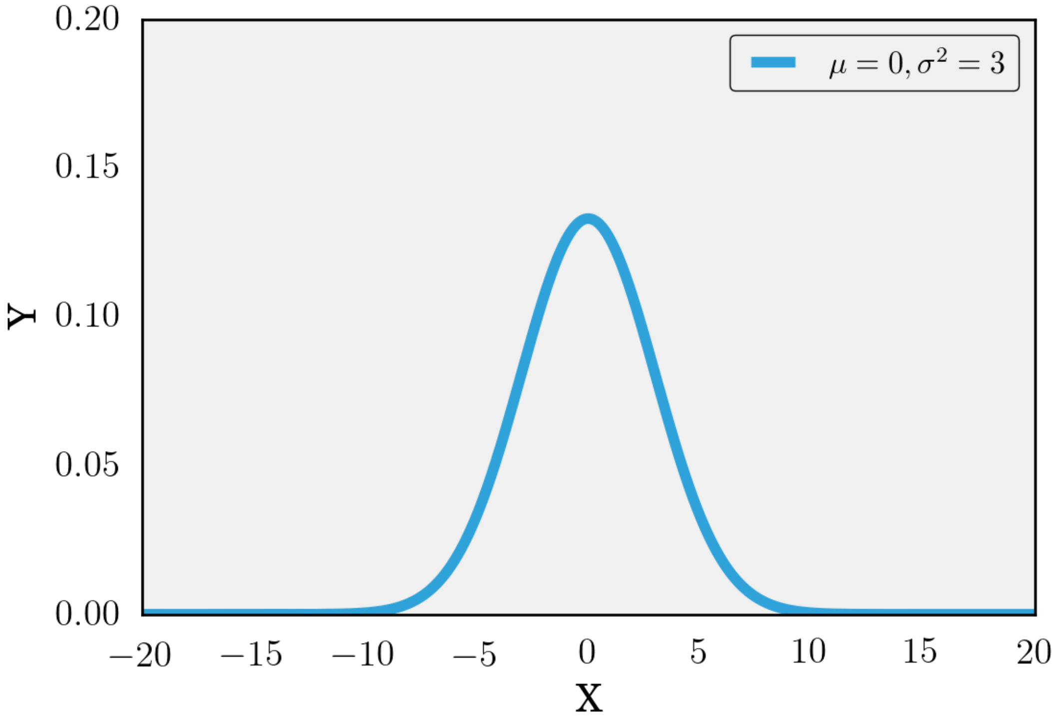


$$\Sigma = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 1 \end{bmatrix}$$

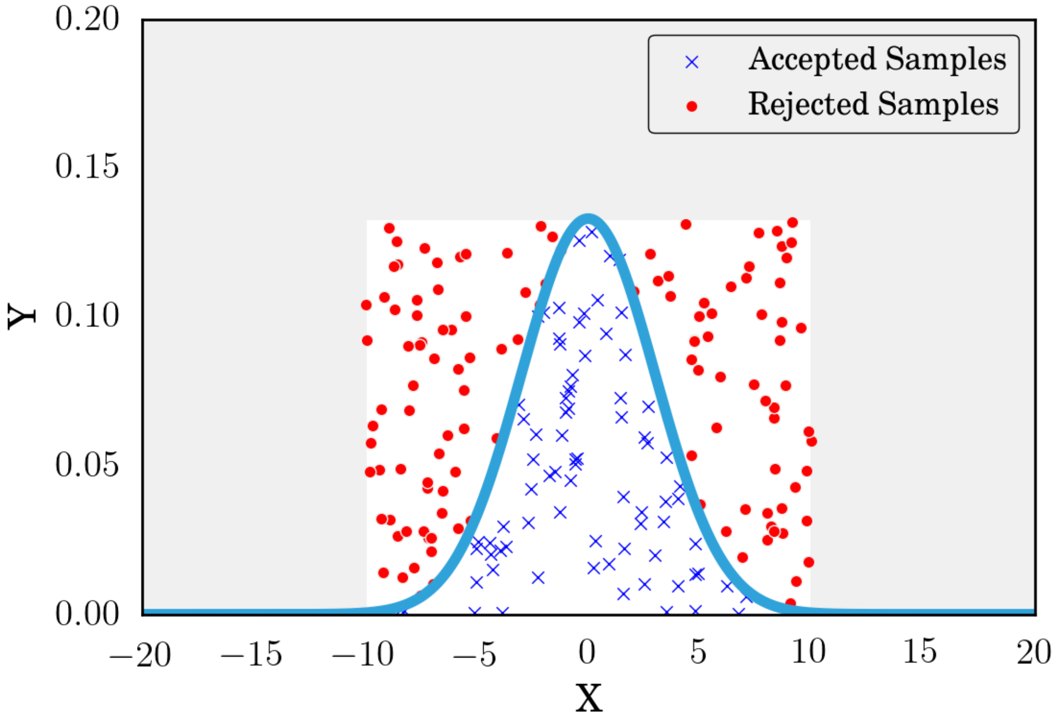


$$\Sigma = \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}$$

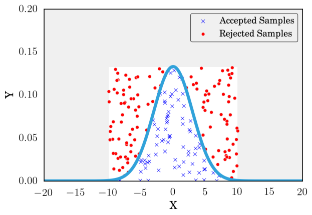
Sampling

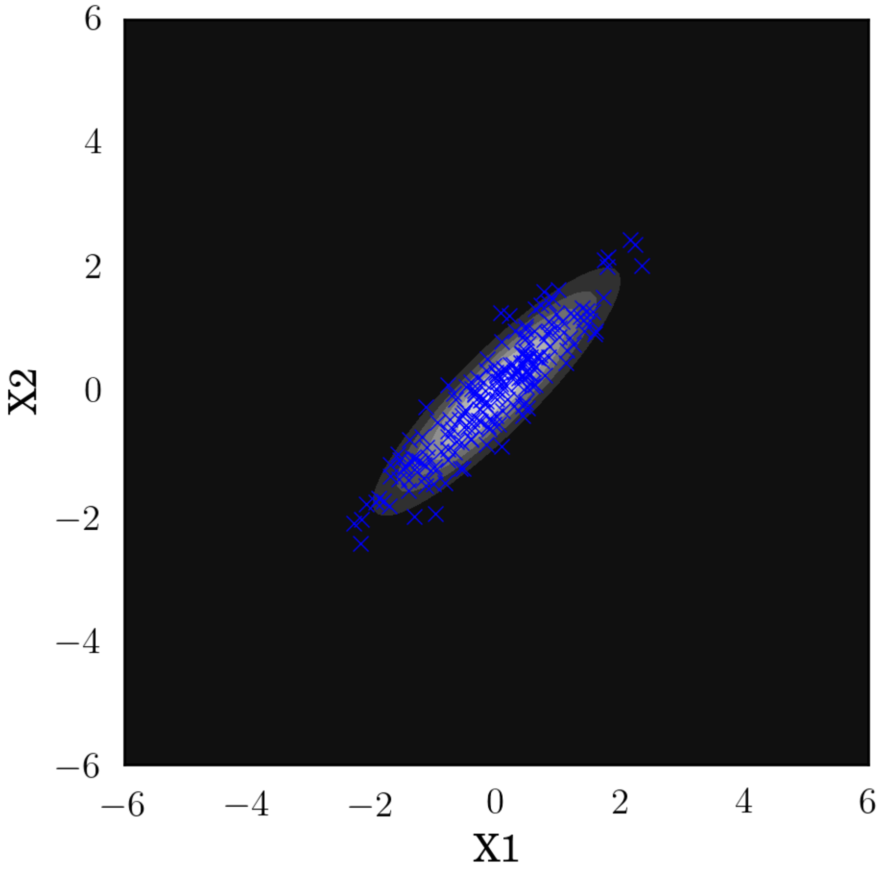


Rejection Sampling

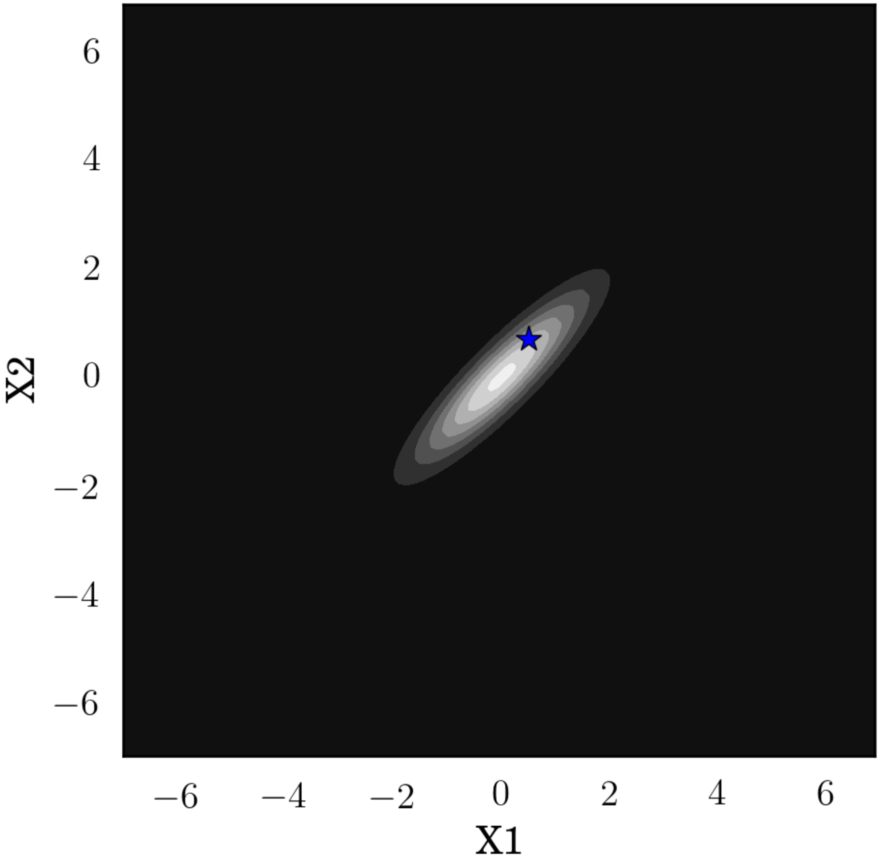


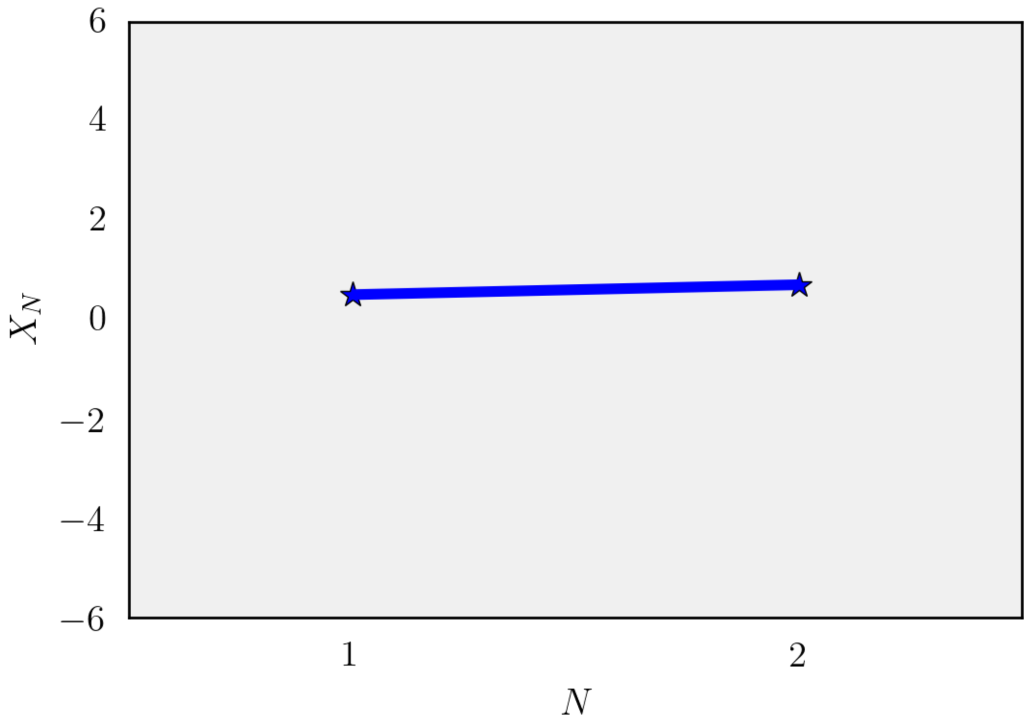
Samples from a bivariate Gaussian



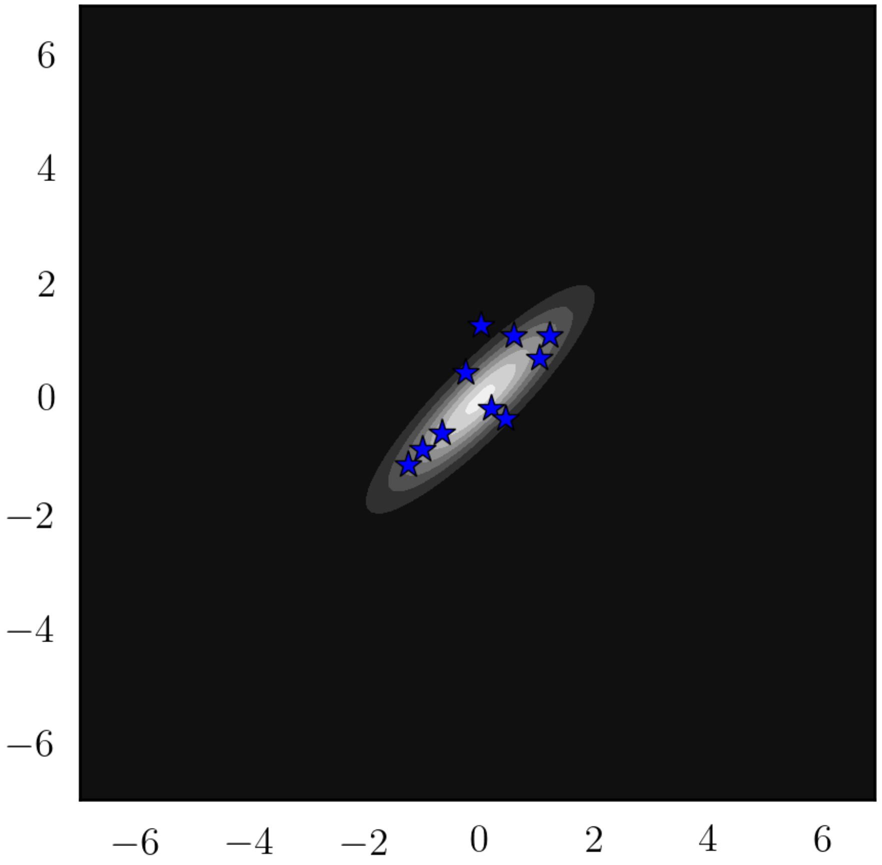


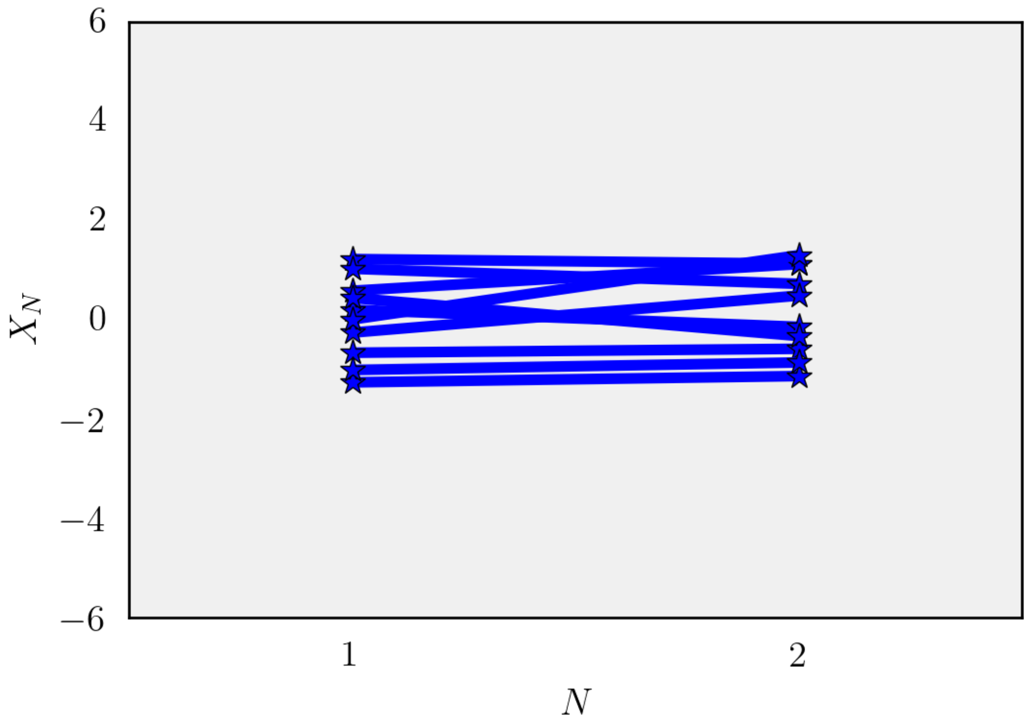
Alternative visualizing of samples





Alternative visualizing of samples





Visualizing higher dimensional Gaussian

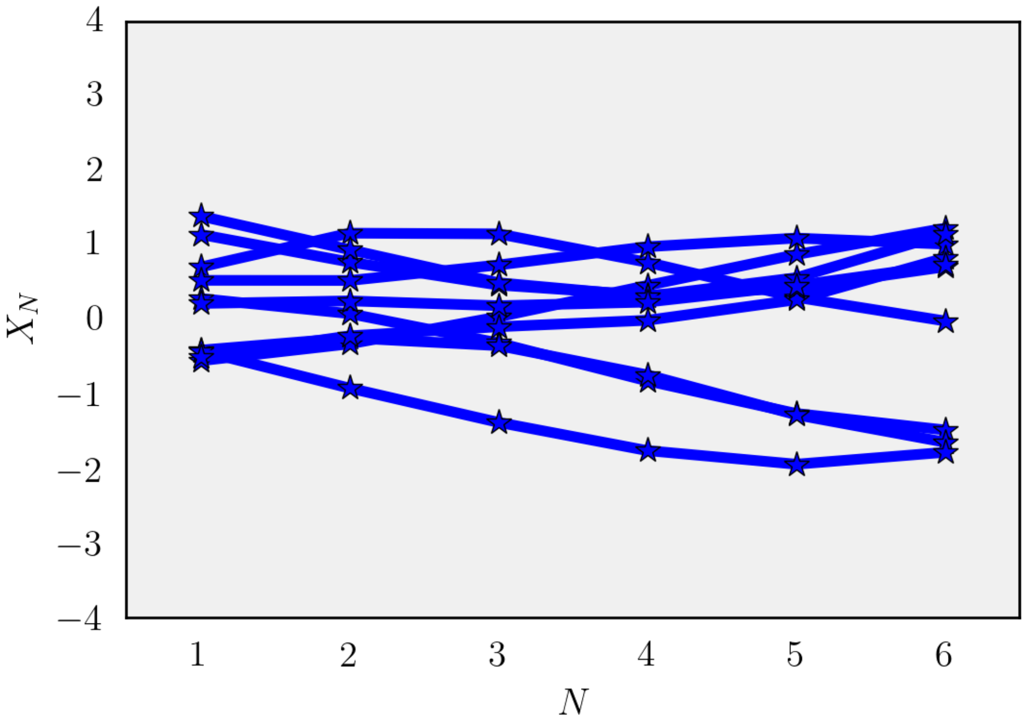
$$\mu = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 1 & 0.95 & 0.8 & 0.6 & 0.41 & 0.25 \\ 0.95 & 1 & 0.95 & 0.8 & 0.6 & 0.41 \\ 0.8 & 0.95 & 1 & 0.95 & 0.8 & 0.6 \\ 0.6 & 0.8 & 0.95 & 1 & 0.95 & 0.8 \\ 0.41 & 0.6 & 0.8 & 0.95 & 1 & 0.95 \\ 0.25 & 0.41 & 0.6 & 0.8 & 0.95 & 1 \end{bmatrix}$$

Visualizing higher dimensional gaussians

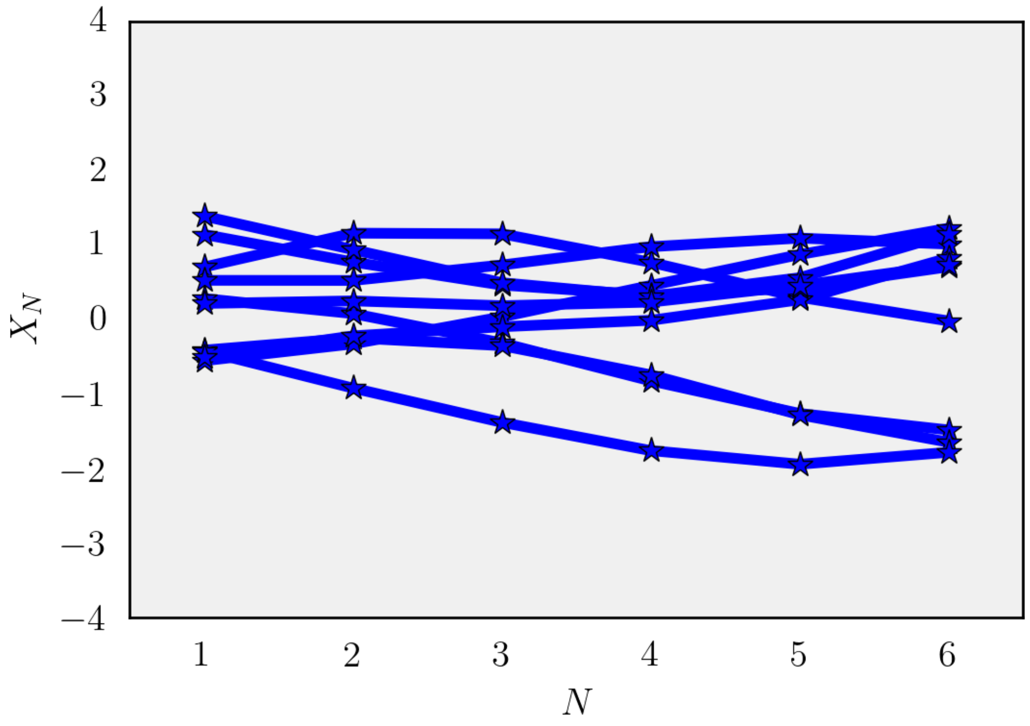
$$\mu = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 1 & 0.95 & 0.8 & 0.6 & 0.41 & 0.25 \\ 0.95 & 1 & 0.95 & 0.8 & 0.6 & 0.41 \\ 0.8 & 0.95 & 1 & 0.95 & 0.8 & 0.6 \\ 0.6 & 0.8 & 0.95 & 1 & 0.95 & 0.8 \\ 0.41 & 0.6 & 0.8 & 0.95 & 1 & 0.95 \\ 0.25 & 0.41 & 0.6 & 0.8 & 0.95 & 1 \end{bmatrix}$$



Each line is one sample from a 6D Gaussian

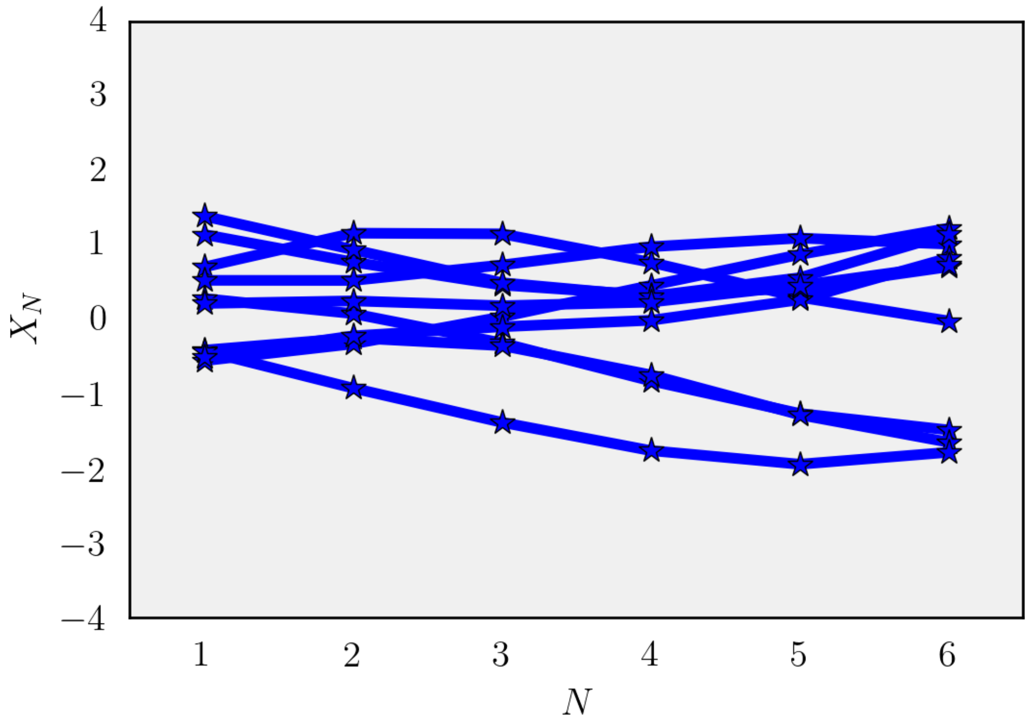
Visualizing higher dimensional gaussians



- The lines resemble nonlinear regression

Each line is one sample from a 6D Gaussian

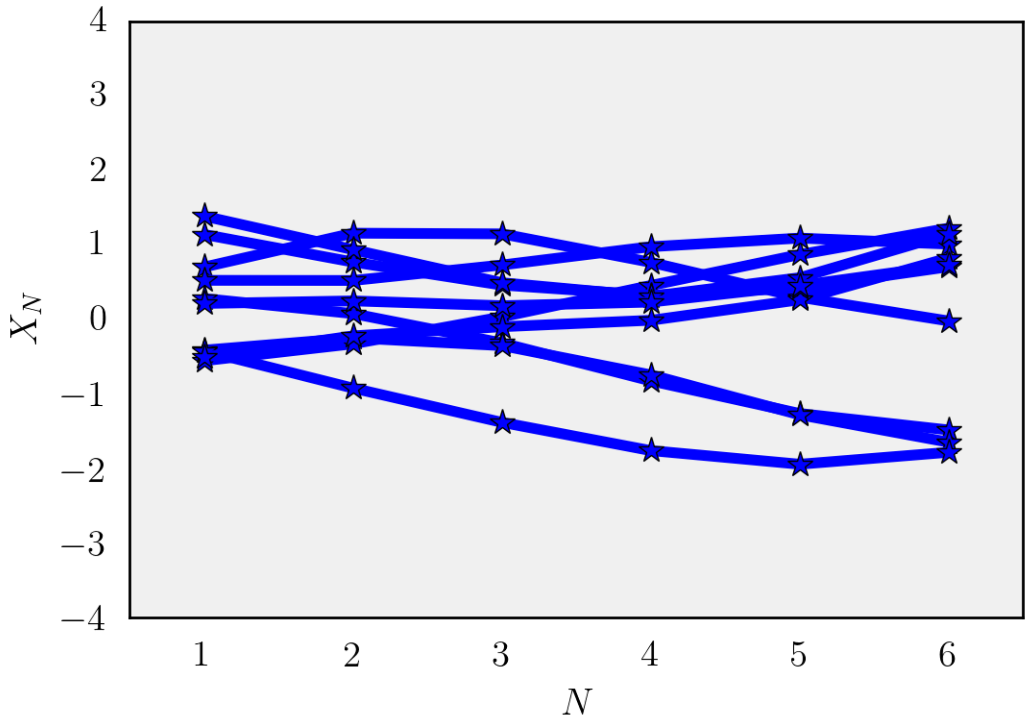
Visualizing higher dimensional gaussians



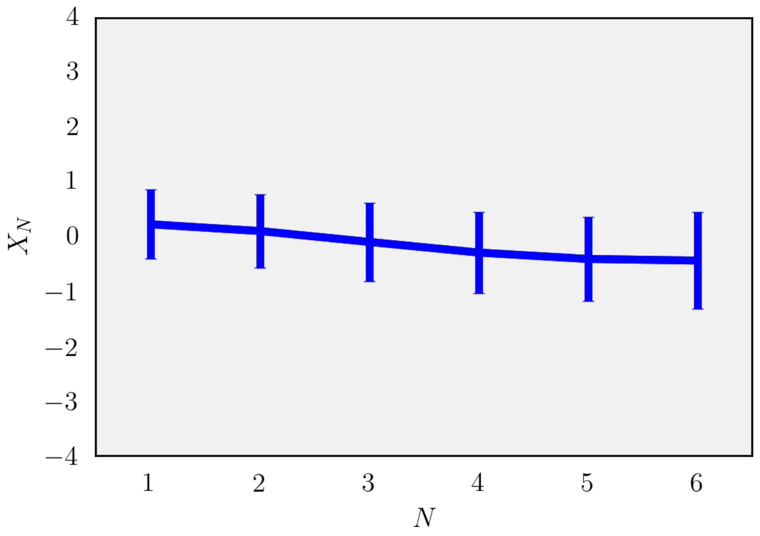
- The lines resemble nonlinear regression
- Close points seem to be correlated to each other

Each line is one sample from a 6D Gaussian

Visualizing higher dimensional gaussians

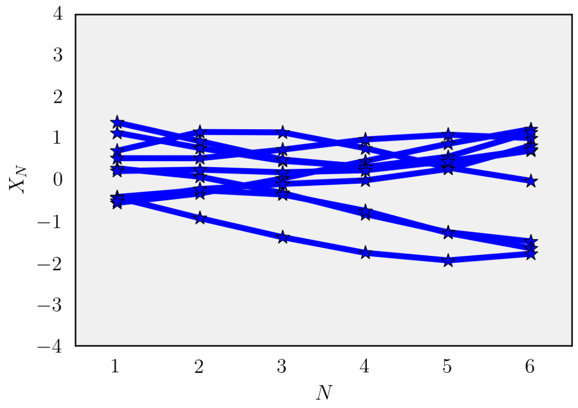


- The lines resemble nonlinear regression
- Close points seem to be correlated to each other
- We can measure the variance at each point



Each line is one sample from a 6D Gaussian

Generating the covariance matrix



$$\Sigma = \begin{bmatrix} 1 & 0.95 & 0.8 & 0.6 & 0.41 & 0.25 \\ 0.95 & 1 & 0.95 & 0.8 & 0.6 & 0.41 \\ 0.8 & 0.95 & 1 & 0.95 & 0.8 & 0.6 \\ 0.6 & 0.8 & 0.95 & 1 & 0.95 & 0.8 \\ 0.41 & 0.6 & 0.8 & 0.95 & 1 & 0.95 \\ 0.25 & 0.41 & 0.6 & 0.8 & 0.95 & 1 \end{bmatrix}$$

The **kernel** specifies how the entries in the covariance matrix are generated.

'Squared exponential' kernel

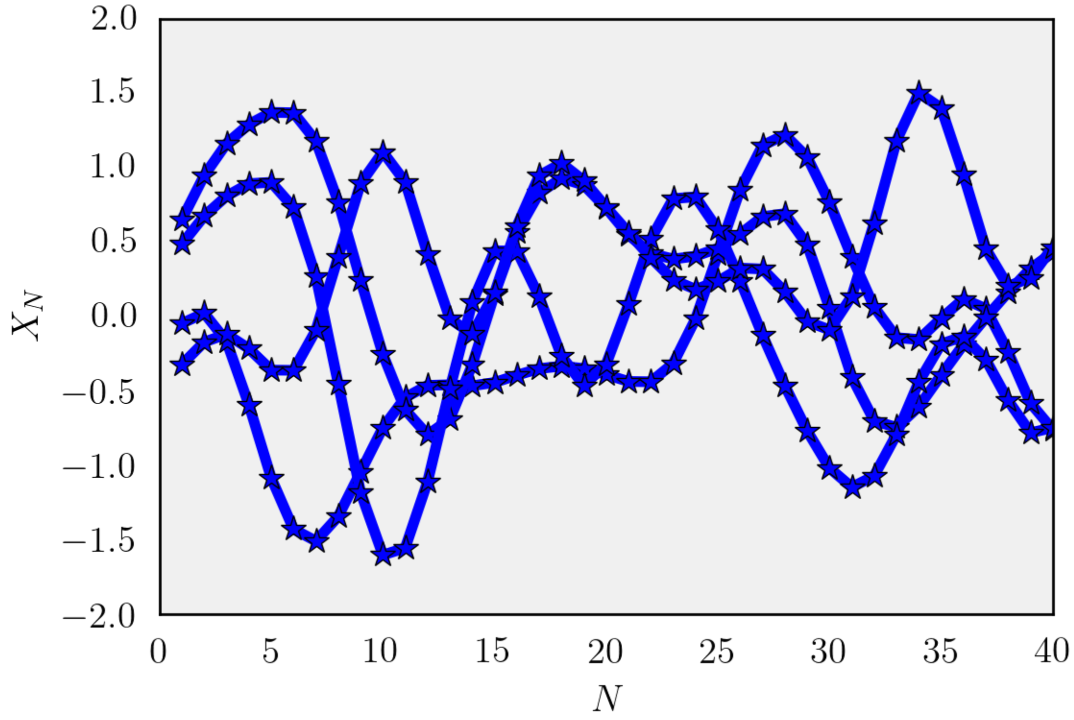
$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

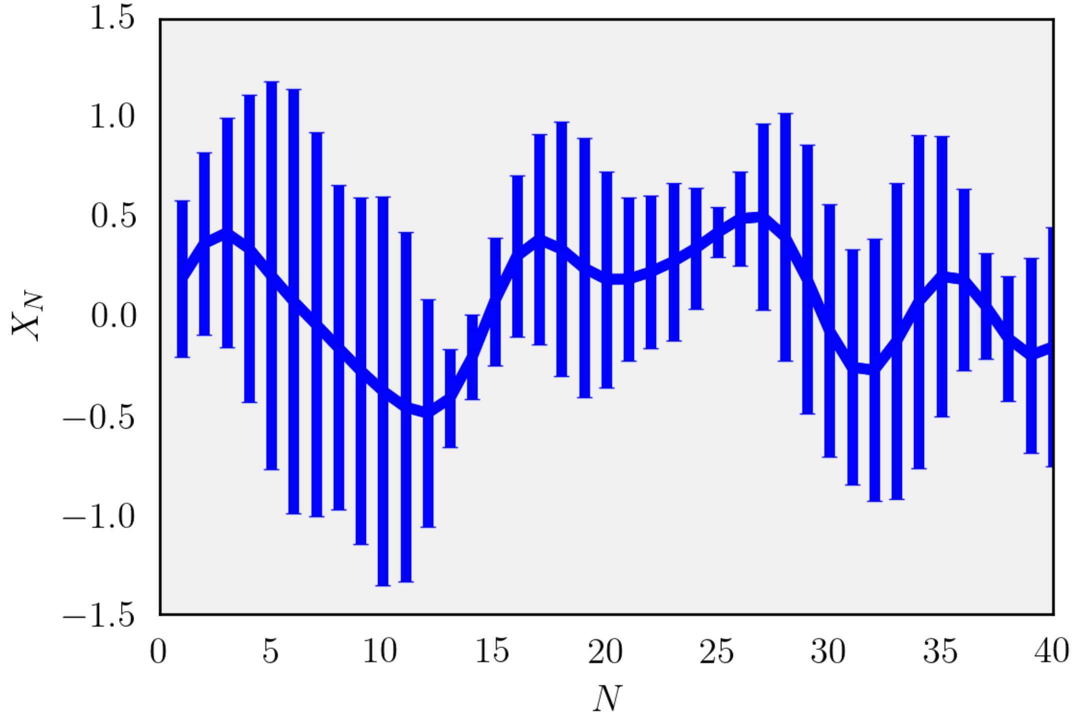
$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

Gaussian process (GP) is fully specified by a mean and covariace function.

Formal definition: A Gaussian process is a collection of random variables with the property that the joint distribution of any finite subset is a Gaussian

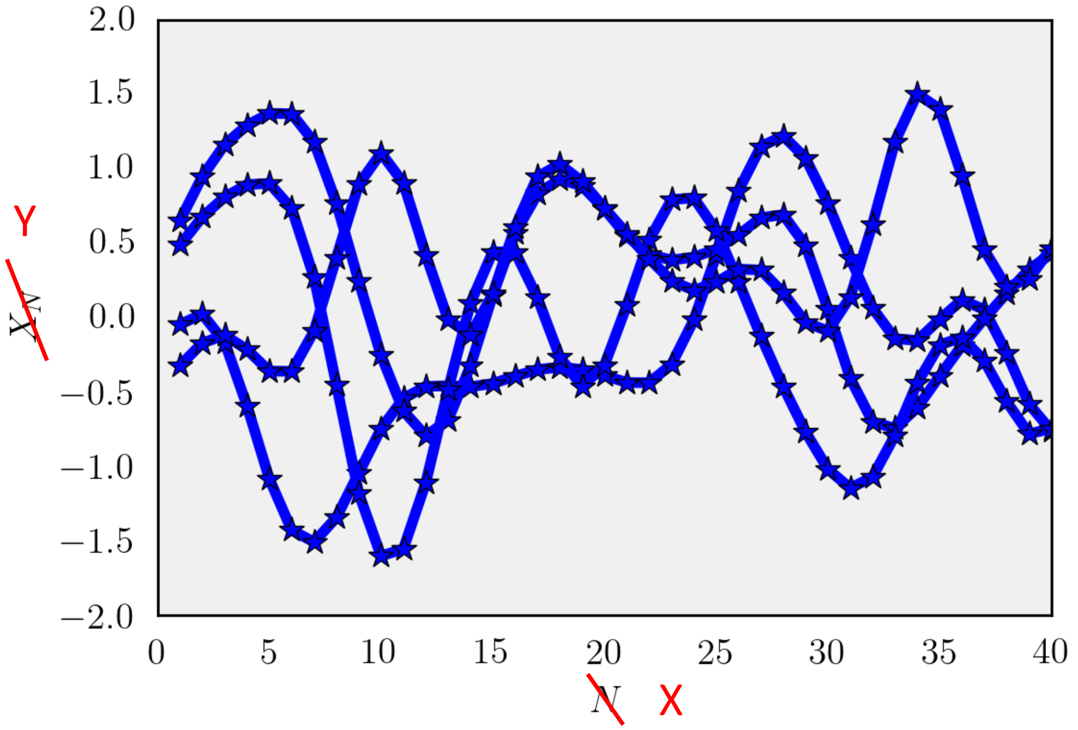
Extending to more dimensions





Each line is one sample from a 40D Gaussian

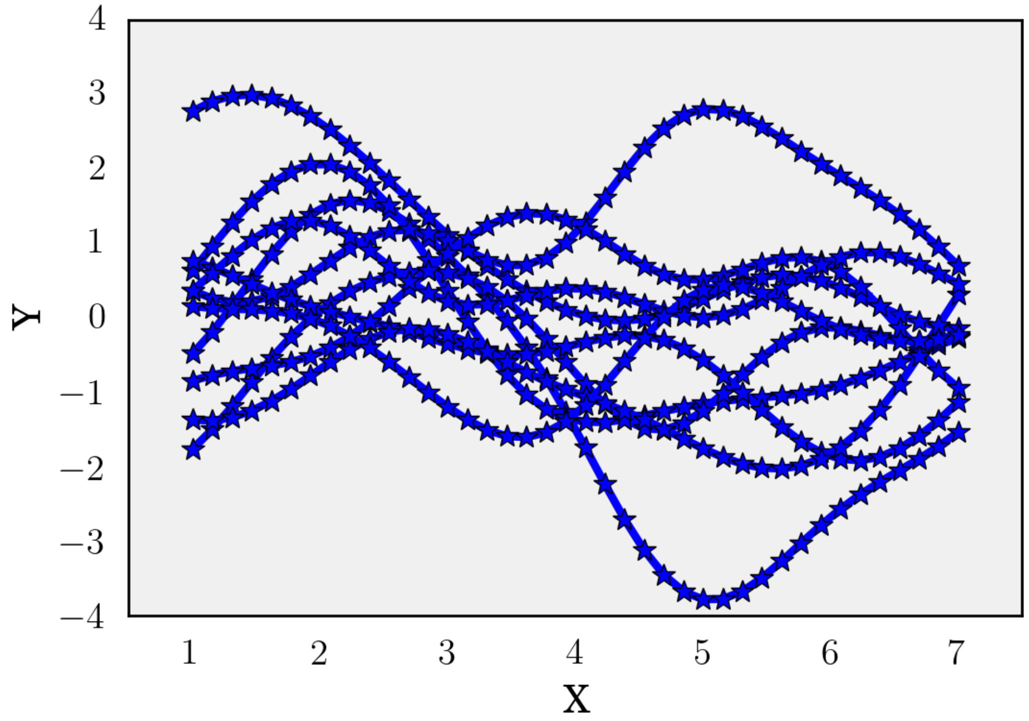
Another change in notation



- Remap axis

Each line is one sample from a 40D Gaussian

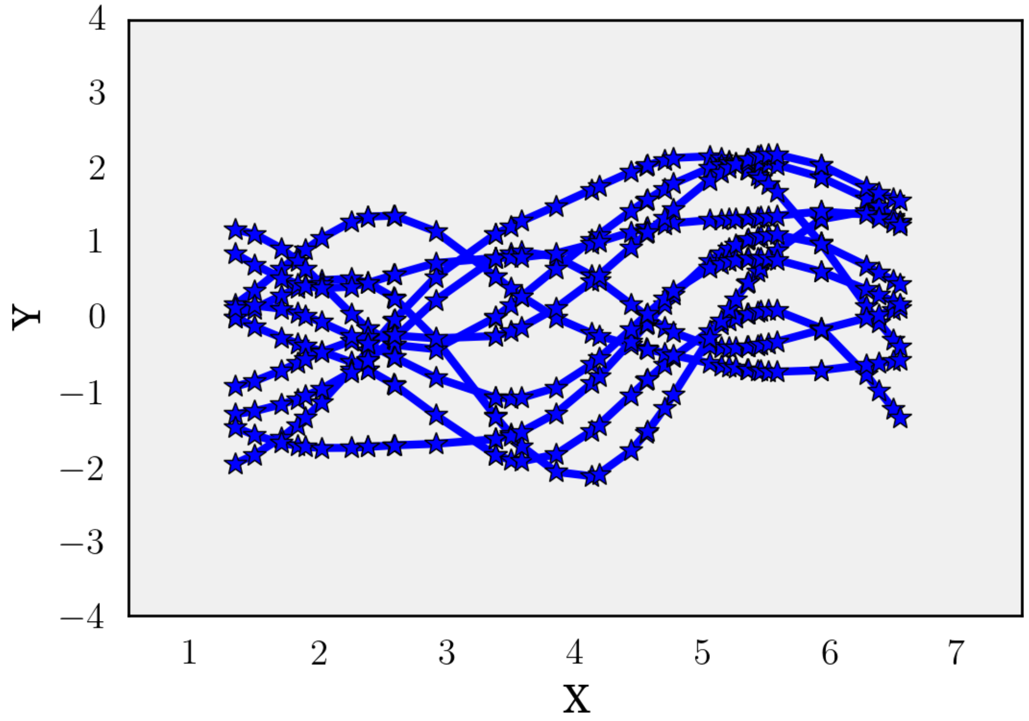
Another change in notation



- Remap axis
- We don’t have to increase the dimension of the X axis with the dimension of the Gaussian

Each line is one sample from a 40D Gaussian

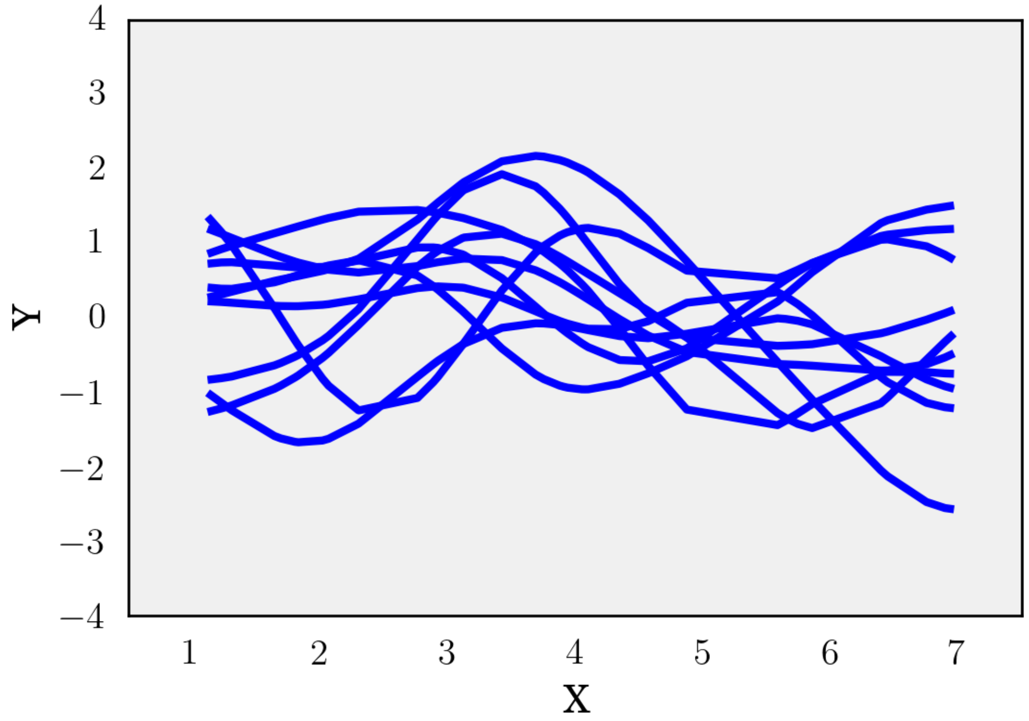
Another change in notation



- Remap axis
- We don’t have to increase the dimension of the X axis with the dimension of the Gaussian
- We don’t have to take points equally spaced to each other

Each line is one sample from a 40D Gaussian

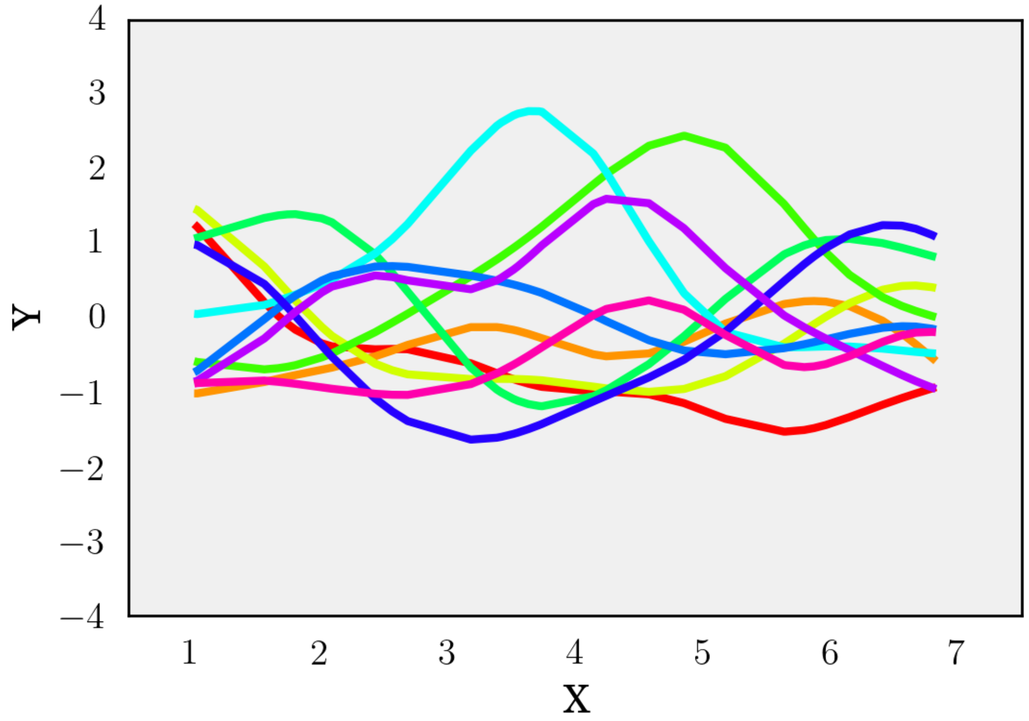
Another change in notation



- Remap axis
- We don’t have to increase the dimension of the X axis with the dimension of the Gaussian
- We don’t have to take points equally spaced to each other
- We can remove the points, just for clarity

Each line is one sample from a 40D Gaussian

Another change in notation



- Remap axis
- We don't have to increase the dimension of the X axis with the dimension of the Gaussian
- We don't have to take points equally spaced to each other
- We can remove the points, just for clarity
- Use colors for the different samples, again for clarity

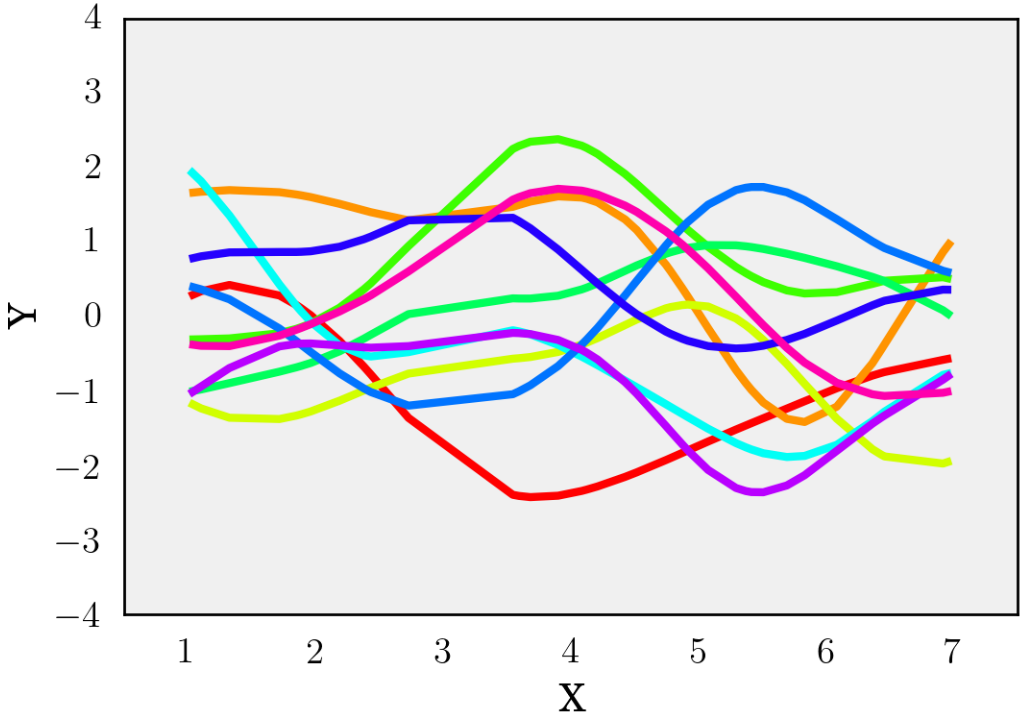
Each line is one sample from a 40D Gaussian

Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

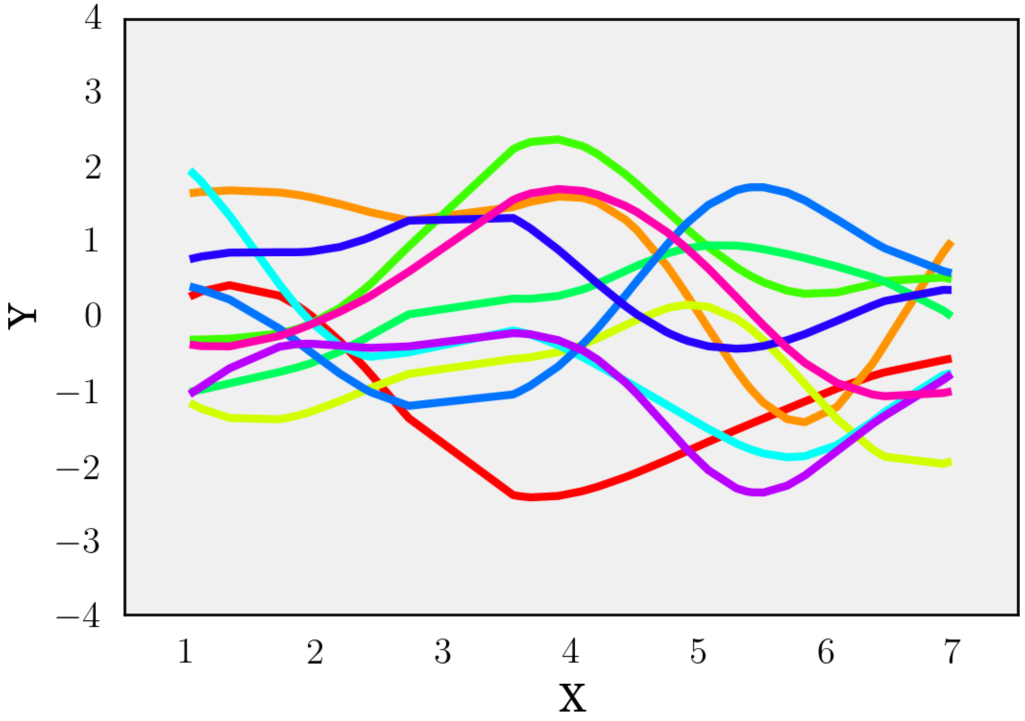


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

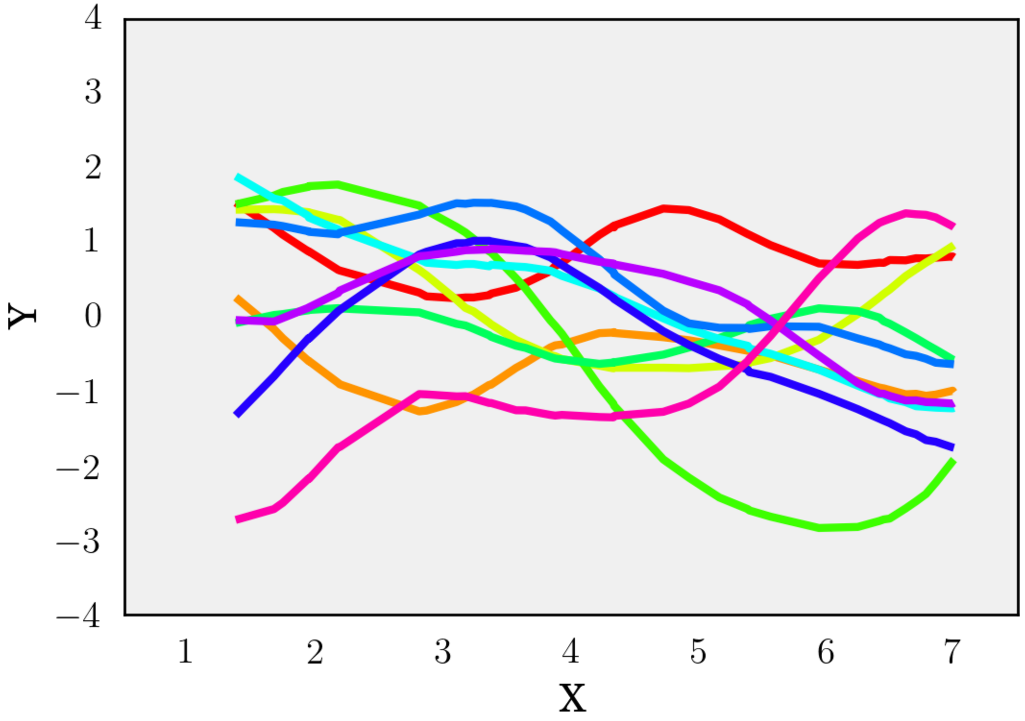


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0.01
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

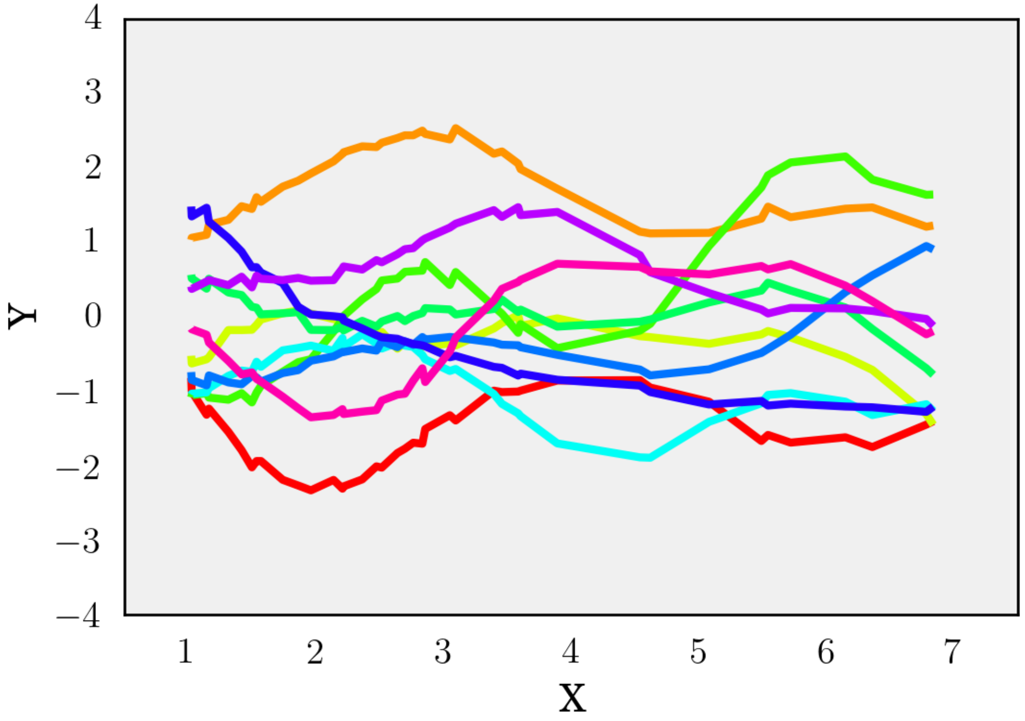


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0.05
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

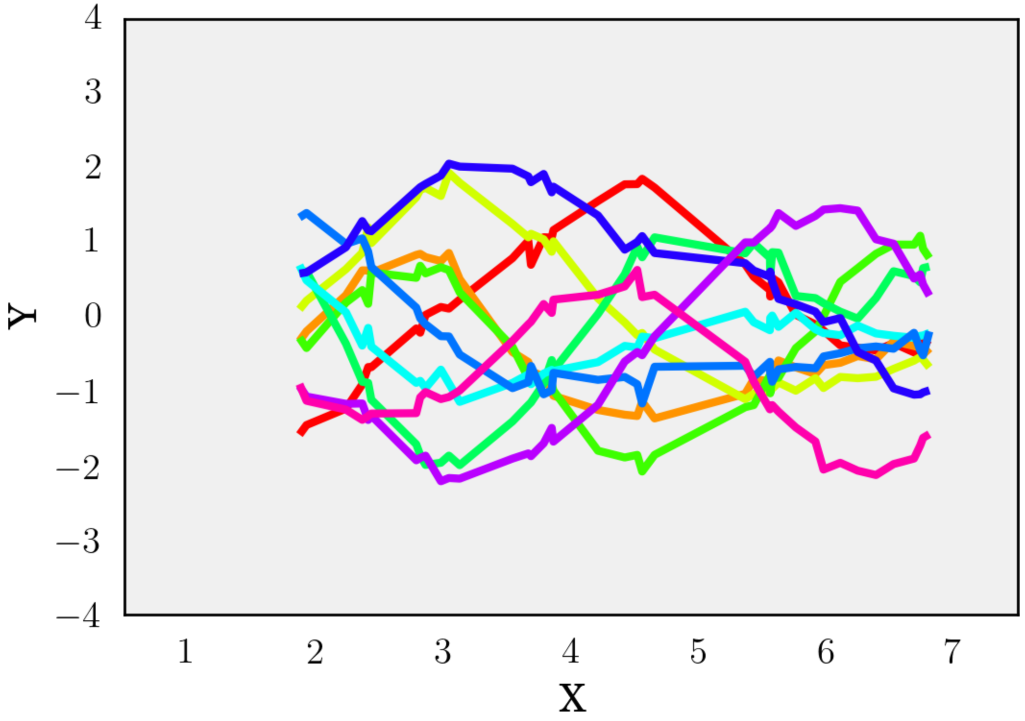


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0.1
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

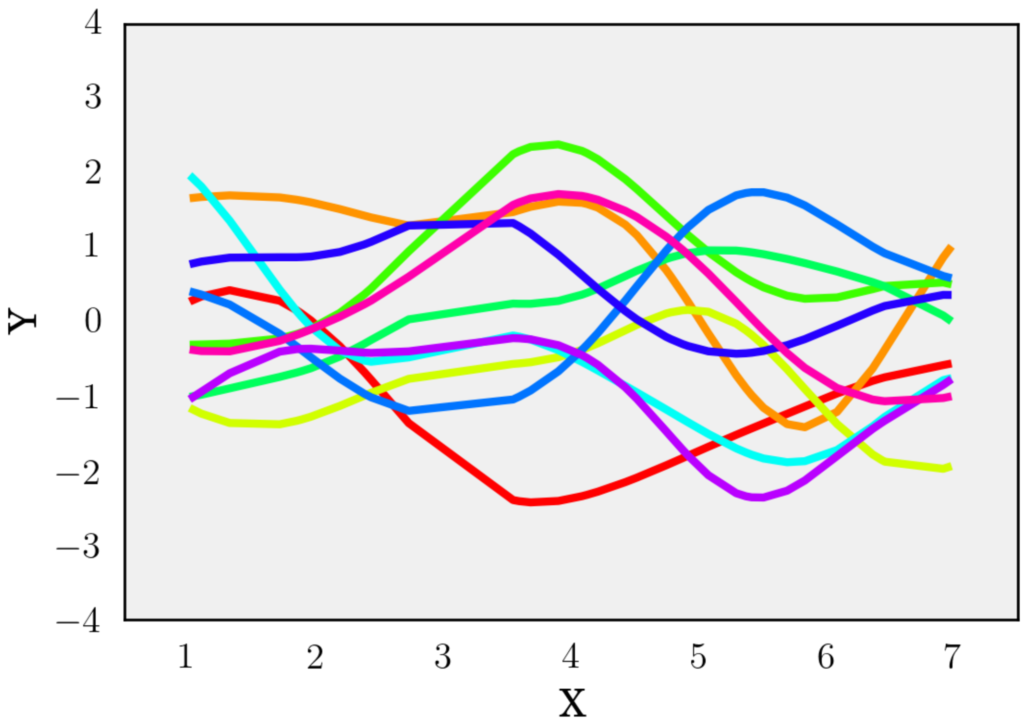


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

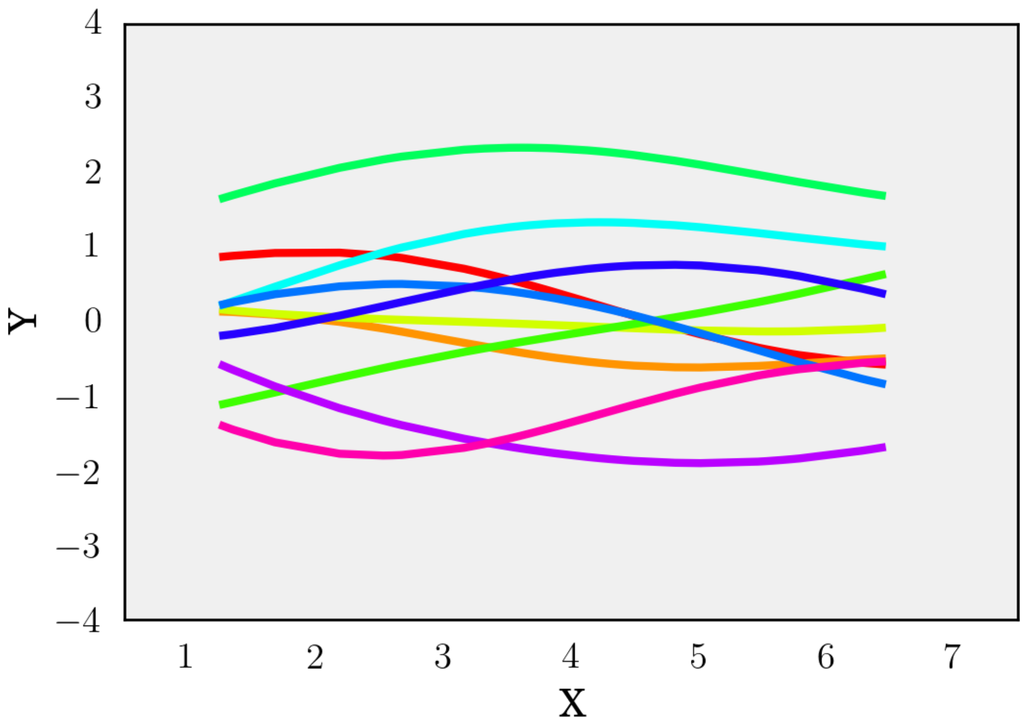


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	3
σ_f	Vertical lengthscale	1

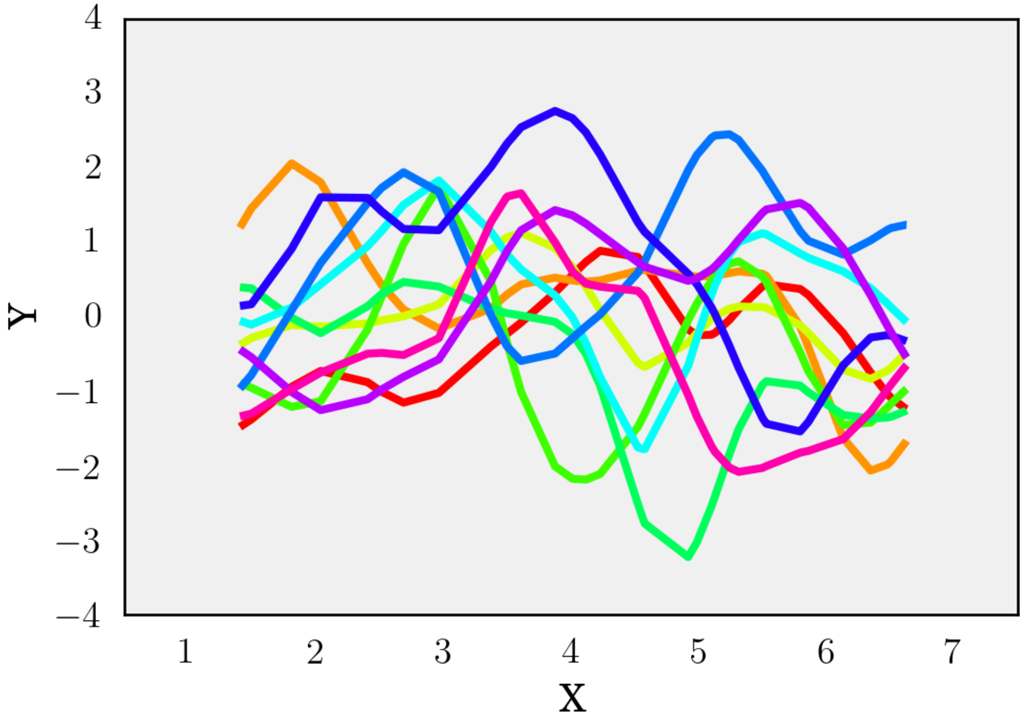


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	0.5
σ_f	Vertical lengthscale	1

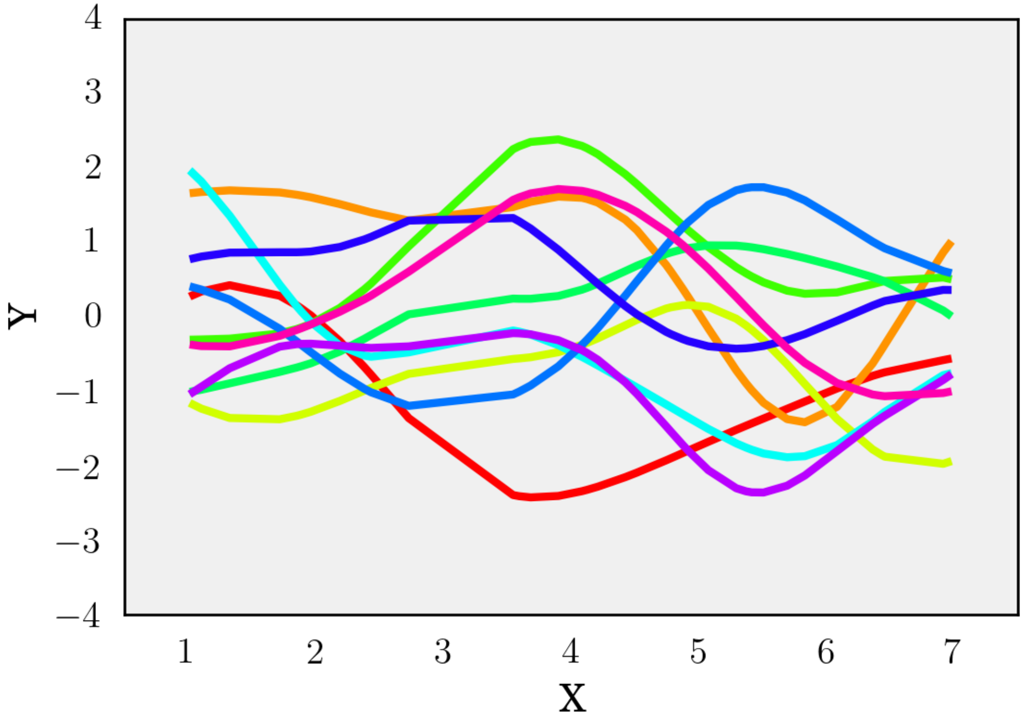


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

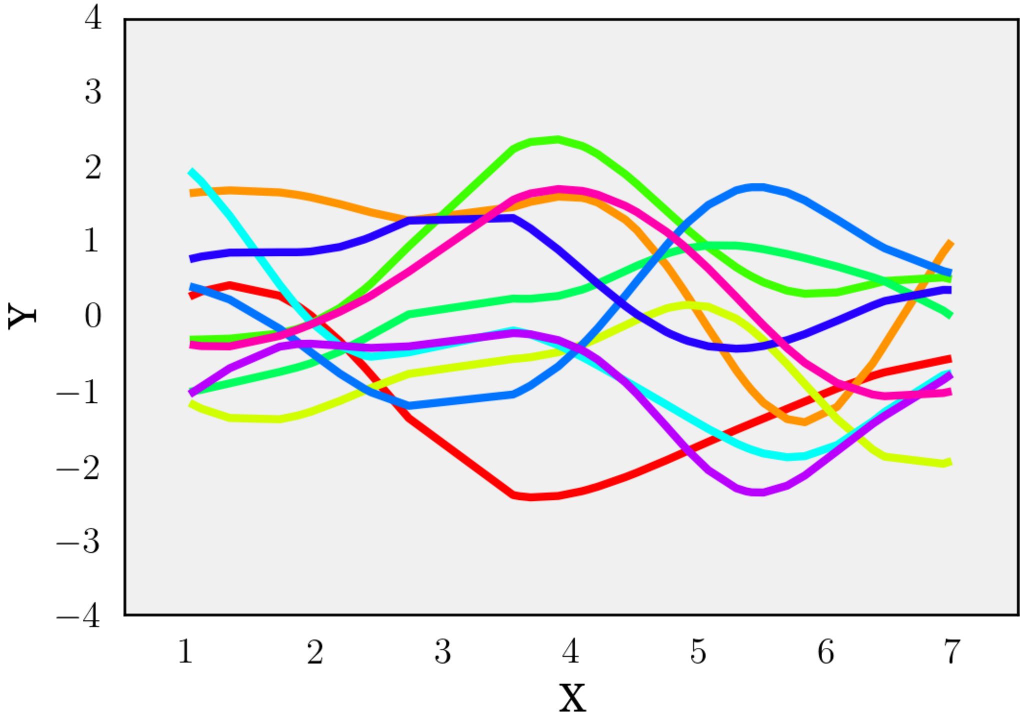


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	1

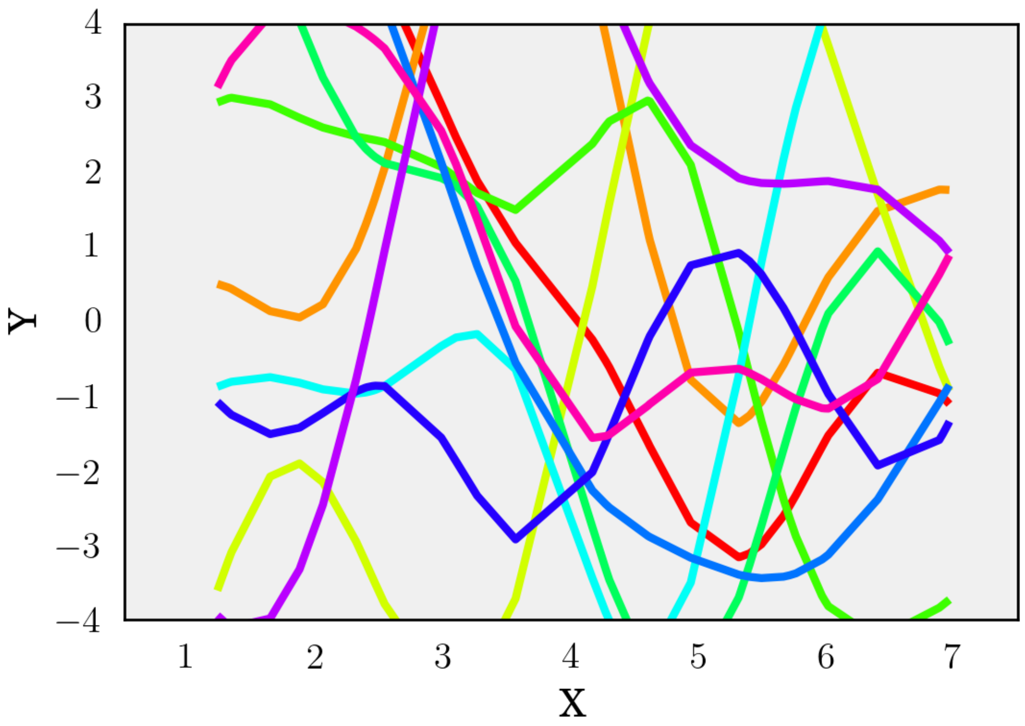


Varying hyperparameters of the kernel

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	**Vertical lengthscale**	**3**

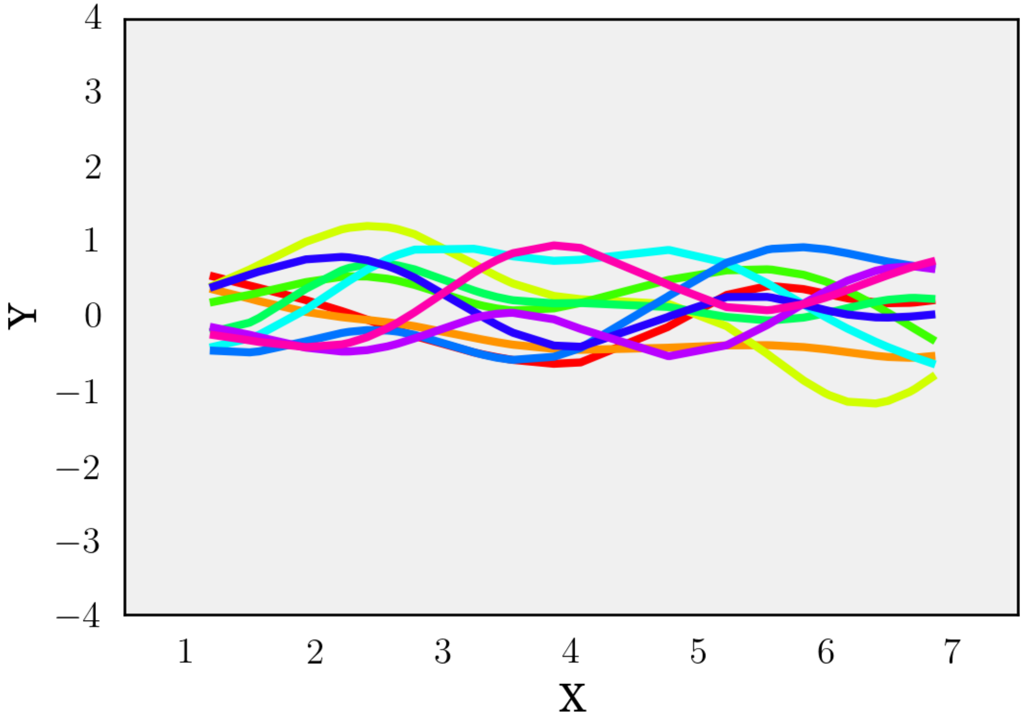


Varying hyperparameters of the kernel

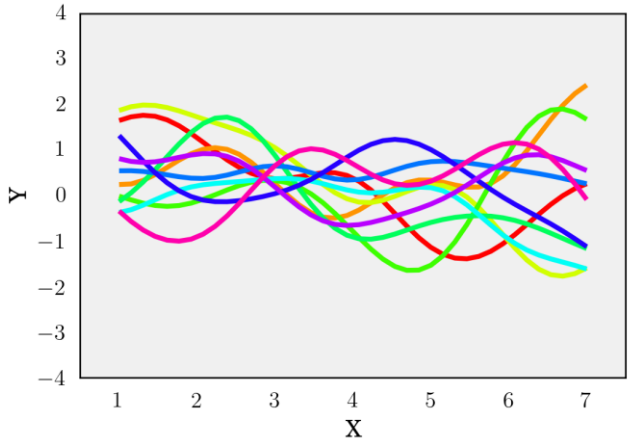
$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(-\frac{1}{2l^2}(x_n - x_{n'})^2\right)$$

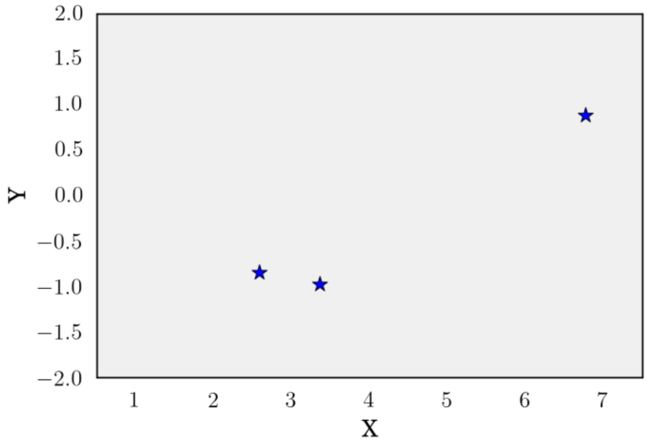
σ_v	Noise level	0
l	Horizontal lengthscale	1
σ_f	Vertical lengthscale	0.5



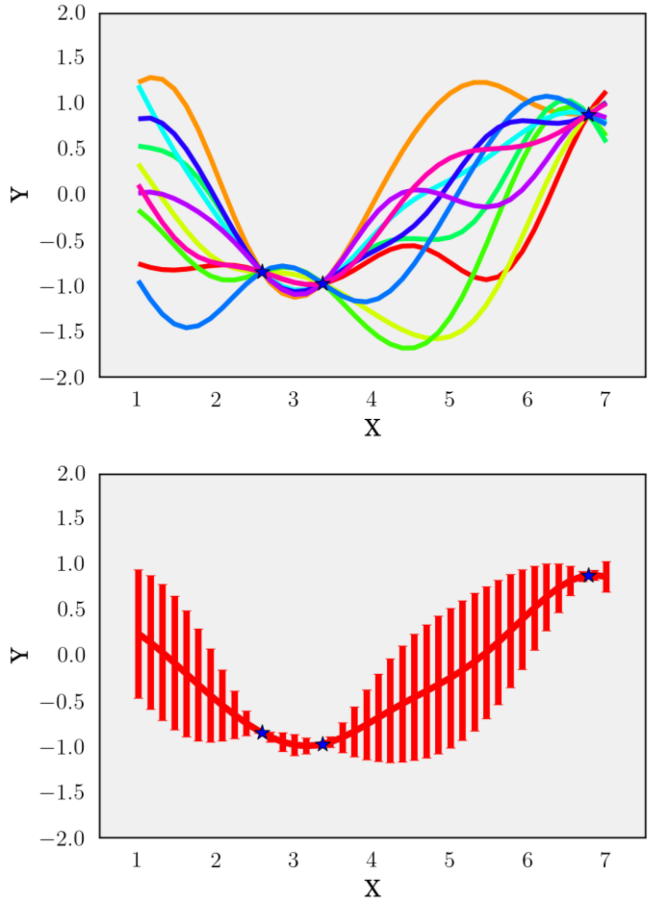
Infererence from data



$$f_* \sim N(0, \Sigma(X_*, X_*))$$



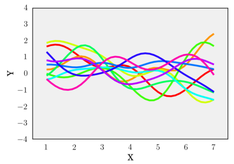
$$\{X, f\}$$



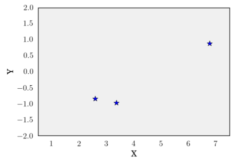
$$f_* | X_*, X, f \sim N(\mu_{f_*}, \Sigma_{f_*})$$

Infererence from data

$$f_* \sim N(0, \Sigma(X_*, X_*))$$

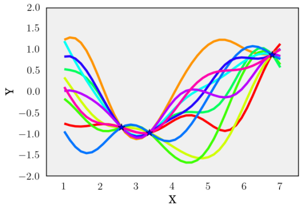


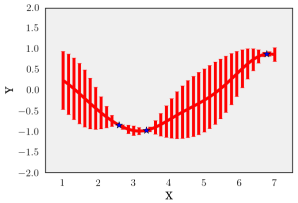
$$\{x_n, y_n\}_{n=1}^N \qquad \{X, f\}$$



$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} \Sigma(X, X) & \Sigma(X, X_*) \\ \Sigma(X_*, X) & \Sigma(X_*, X_*) \end{bmatrix}\right)$$

$$f_* | X_*, X, f \sim N(\mu_{f_*}, \Sigma_{f_*})$$





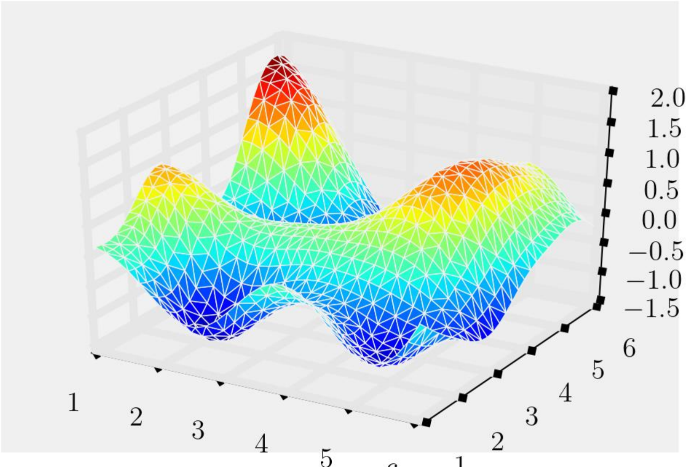
$$\Sigma_{f_*} = \Sigma(X_*, X_*) - \Sigma(X_*, X)\Sigma(X, X)^{-1}\Sigma(X, X_*)$$

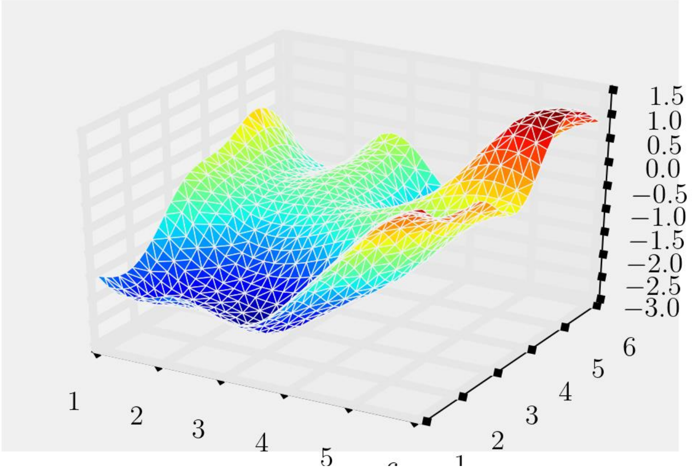
$$\mu_{f_*} = \Sigma(X_*, X)\Sigma(X, X)^{-1} f$$

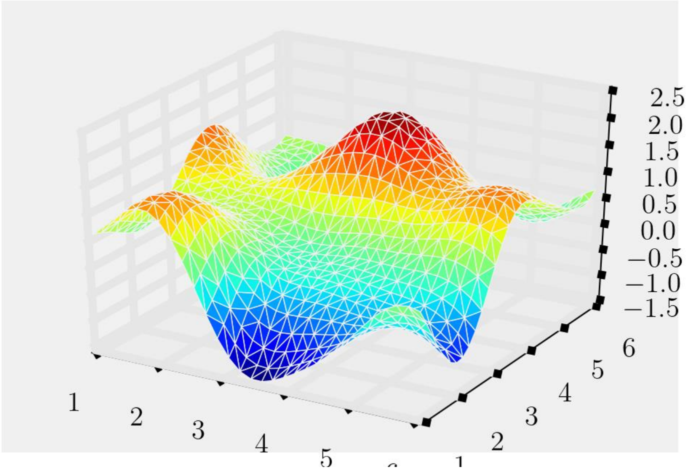
Two-dimensional input space

$$cov(y_n, y_{n'}) = k(x_n, x_{n'}) + \sigma_v^2 \delta_{nn'}$$

$$k(x_n, x_{n'}) = \sigma_f^2 exp\left(- \sum_{d=1}^{D} \frac{1}{2l^2} (x_{dn} - x_{dn'})^2 \right)$$







Spatiotemporal phenomena

Def. ***Event depended and changing with respect to time and space***

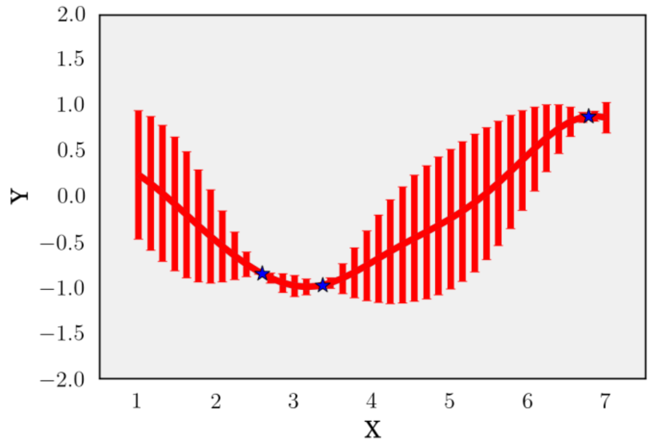
Examples:
- Weather temperature
- Wind speed
- Plankton densities in sea

Modelling with Gaussian Processes

Let's say we want to make a **prediction of the weather in Scotland**

Limitations:
We can take only **limited number of measurements**

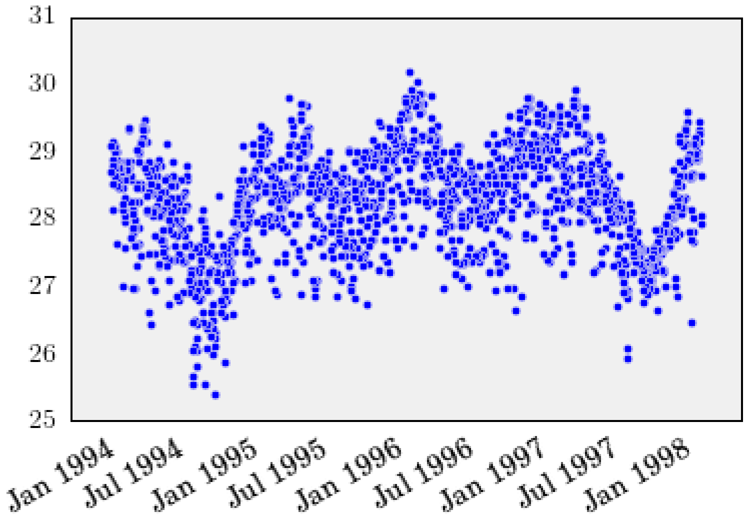
Where do we collect measurements to have the most accurate predictions?
Once we have the measurements, how do we do the prediction?
How certain are we of our prediction?
How can we exploit the structure of the problem?



Gaussian processes have the following desirable properties:

- **Useful priors** from our kernel function (close locations have similar temperature)
- **Measuring the uncertainty** of the field (how accurate our prediction is in a given place)
- **Picking next point to predict** (knowing where to place an extra sensor to improve our prediction the most)
- **Lazy evaluation** (measure the weather in one place, update our prediction, pick the most uncertain place, measure the weather in that place, and so on)

Modelling with Gaussian Processes (cont.)

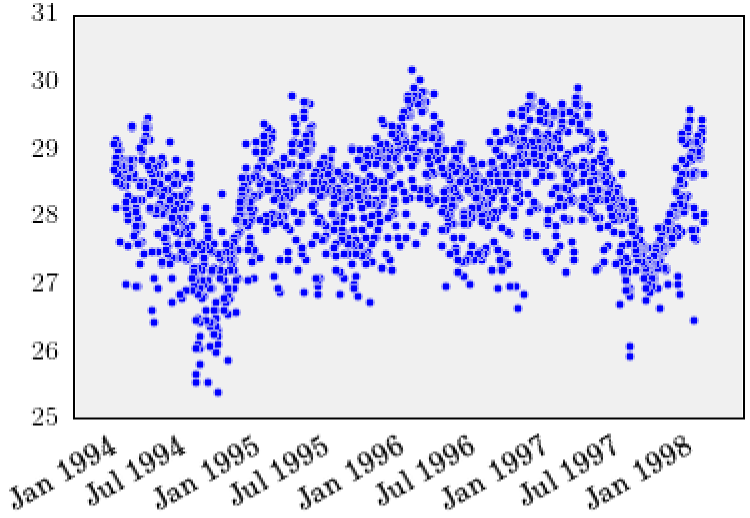


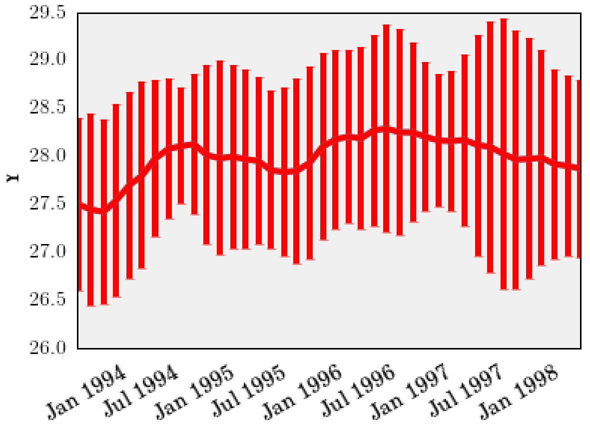
El Nino Data Set

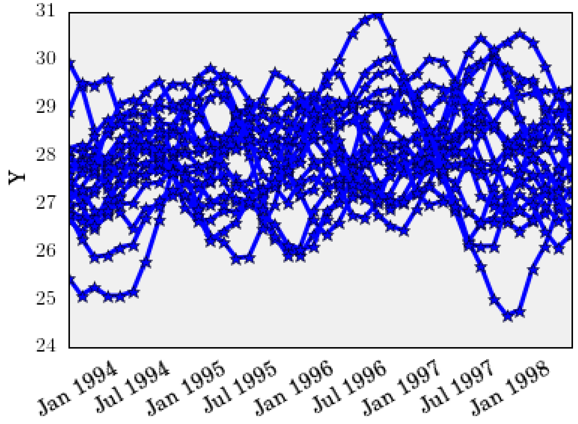
- Measured meteorological variables across the Pacific - air temperature, relative humidity, surface winds, sea surface temperatures, etc.
- 178080 instances
- We are going to use air temperature, for a fixed location (156 longitude, -6 latitude) over the span of 5 years

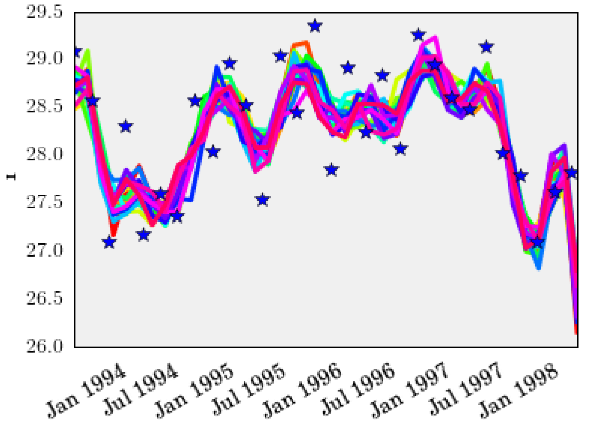
We are trying to find underlying structure in the data

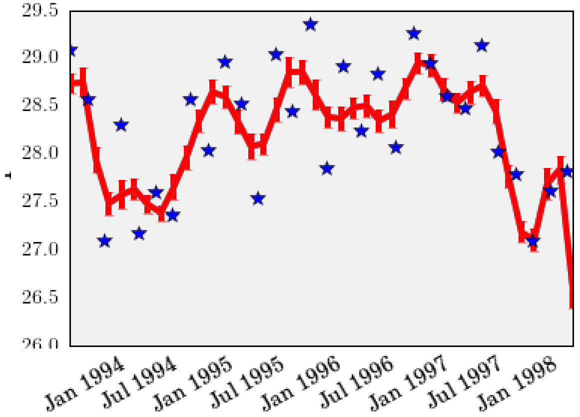
Modelling with Gaussian Processes (cont.)





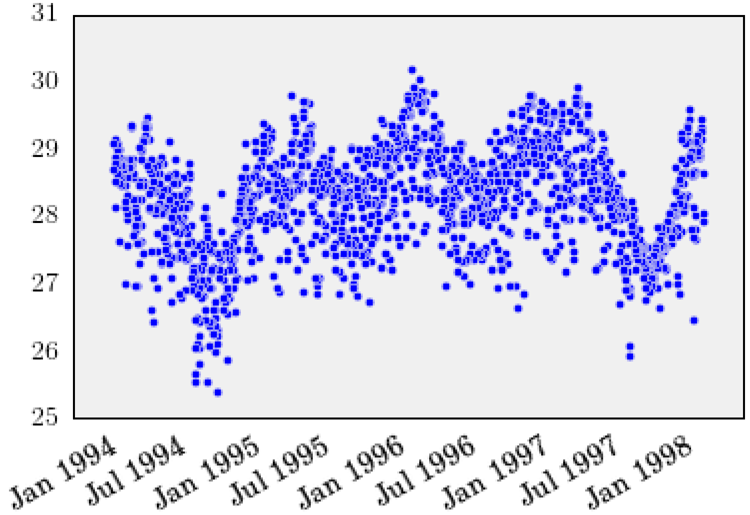


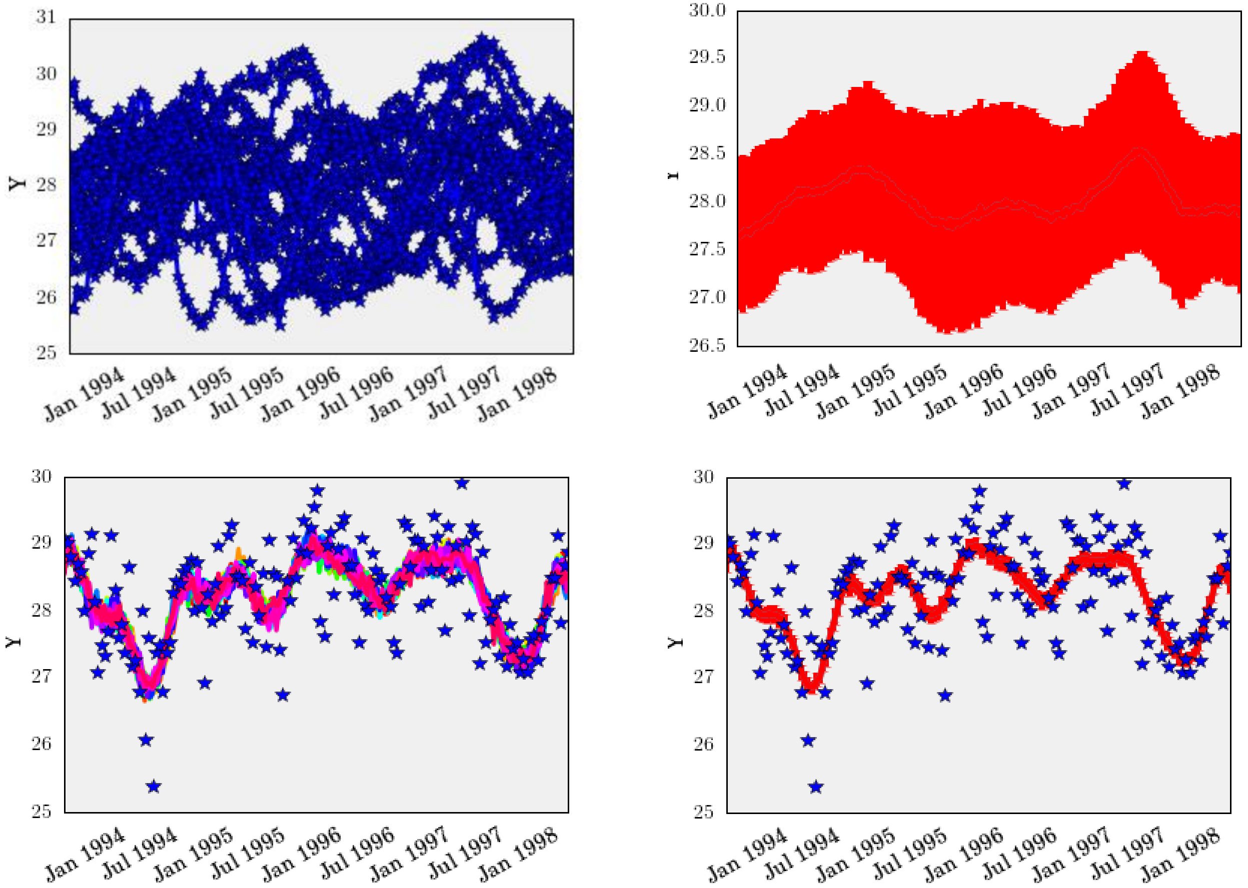




30 data points

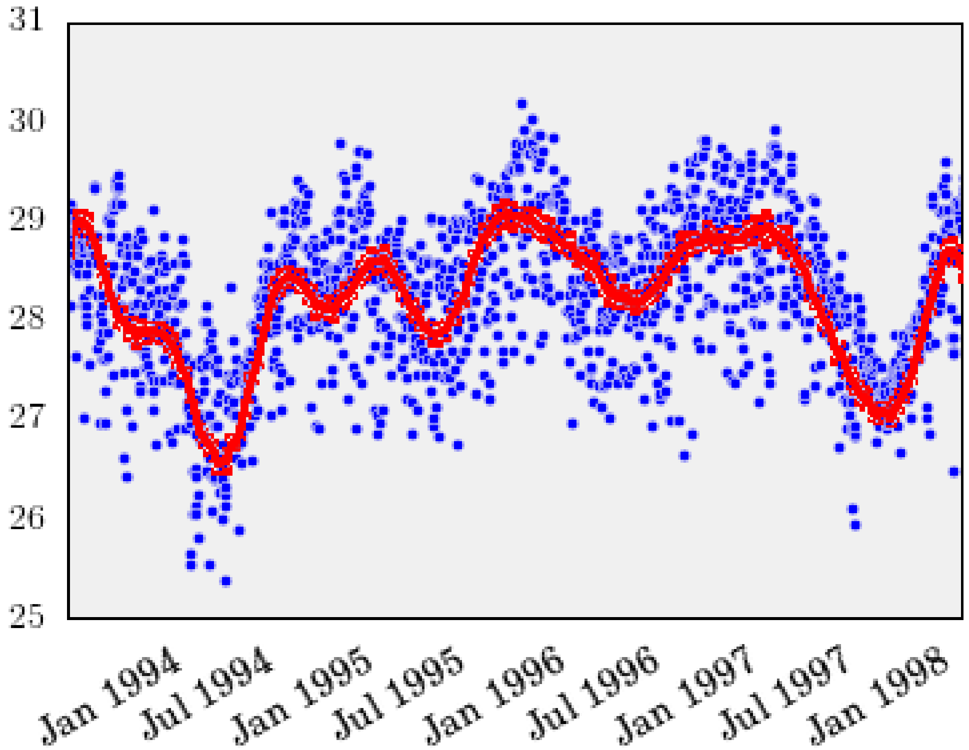
Modelling with Gaussian Processes (cont.)





100 data points

Modelling with Gaussian Processes (cont.)



Applications

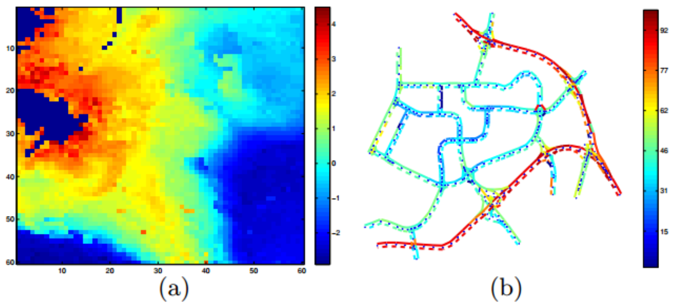
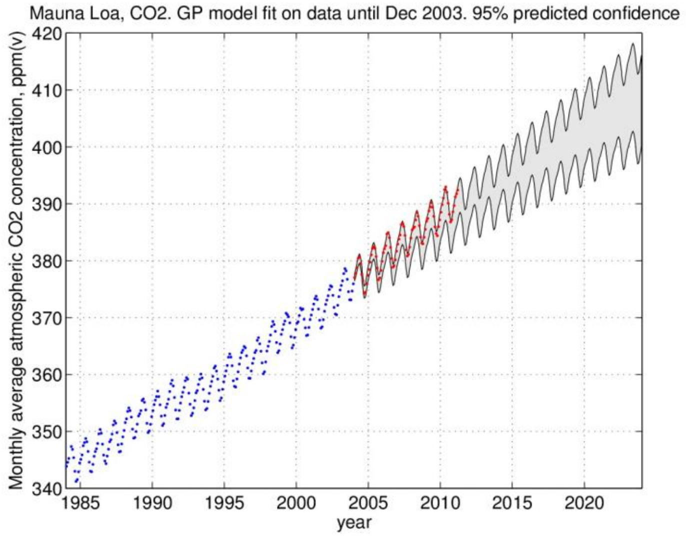


Figure 1: Real-world non-stationary environmental phenomena: (a) Plankton density (chl-a) phenomenon (measured in $\mathrm{mg/m^3}$) in log-scale in Gulf of Mexico, and (b) traffic (road speeds) phenomenon (measured in $\mathrm{km/h}$) over an urban road network.



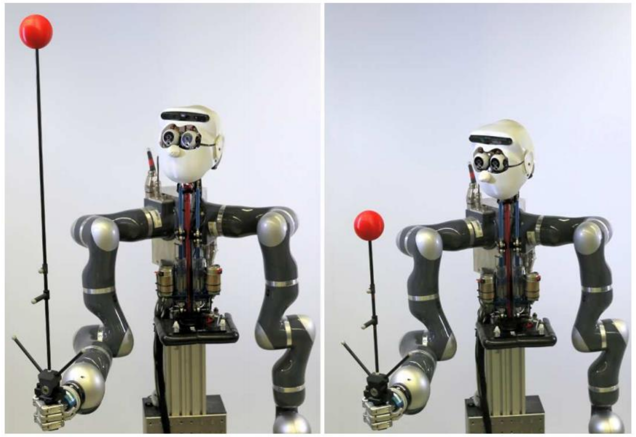
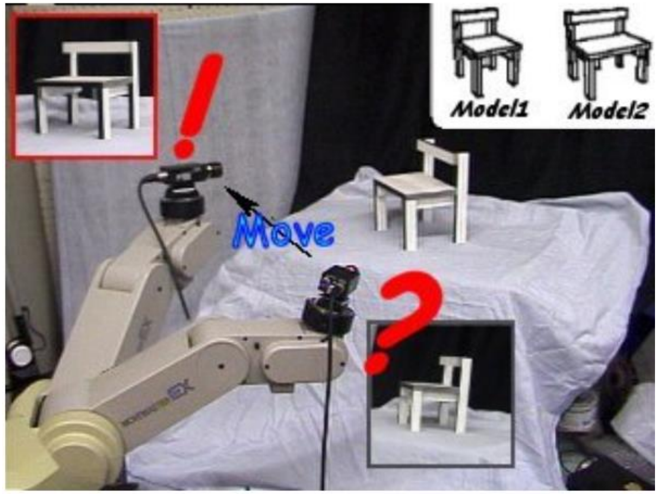


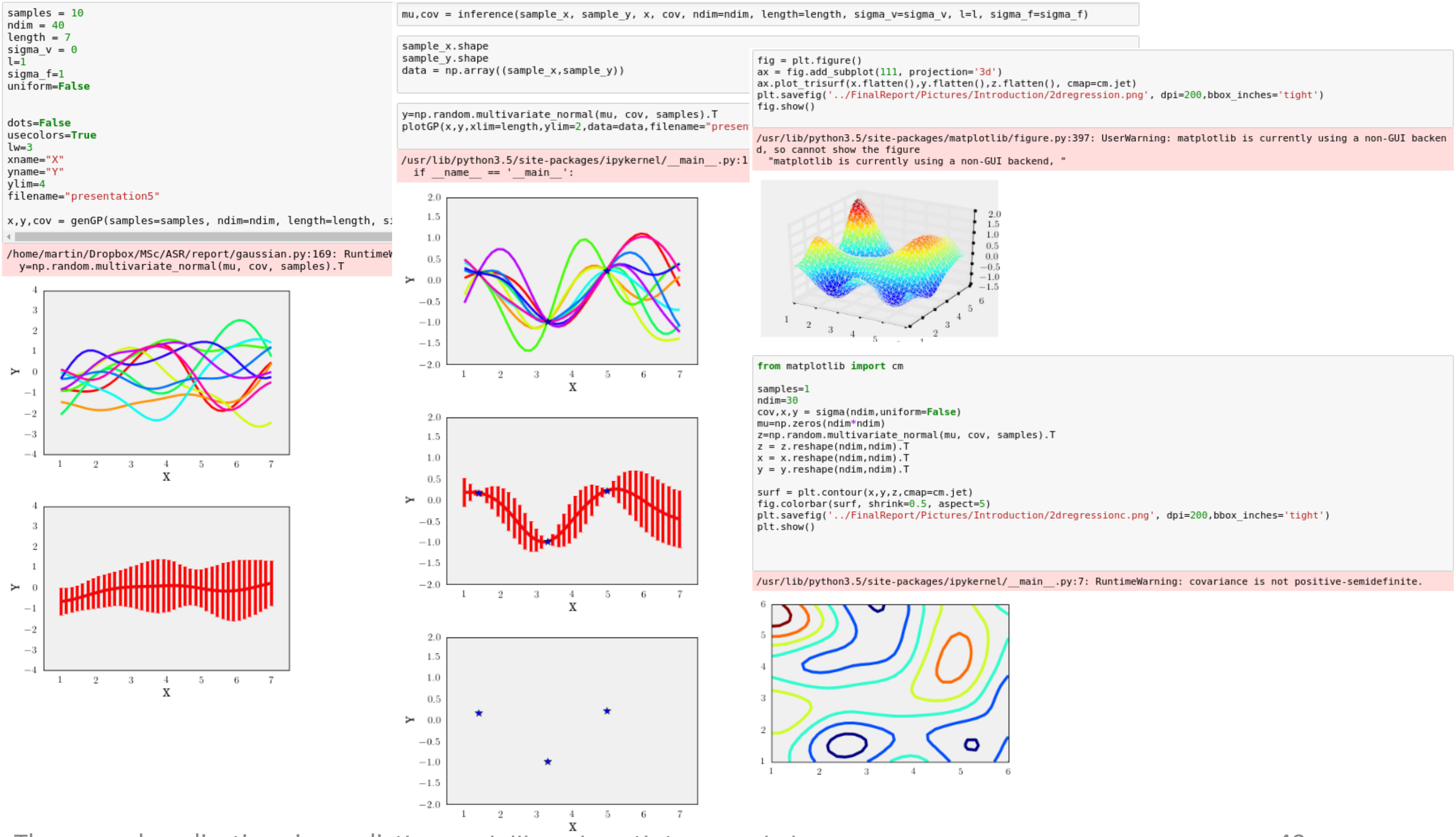
FIGURE 1.1: Robot Apollo balancing two inverted poles. These experimental platforms are used as a demonstrators of the automatic tuning framework.



The code is available on Github

- Different visualizations
- Inference from data
- 1D and 2D regression

https://github.com/masenov/GP_Intro



Not enough time to cover…

- Using GPs for classification and reinforcement learning
- Connections with neural networks
- Different kernel functions
- Efficient calculations of the covariance matrix

Thank you for your attention

Any questions?