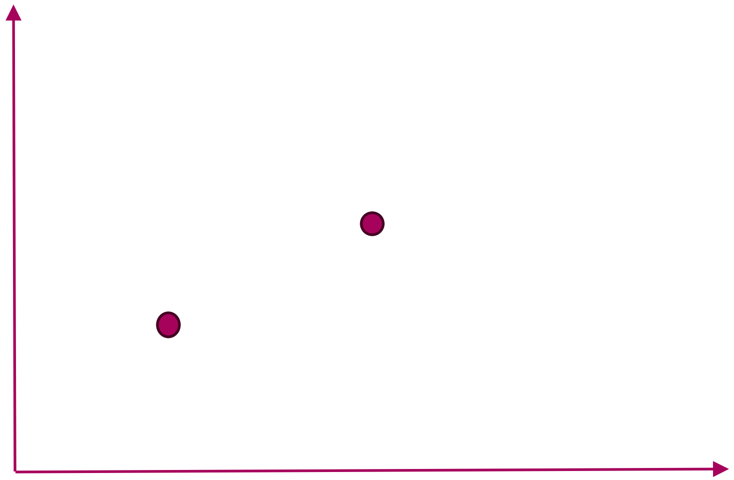
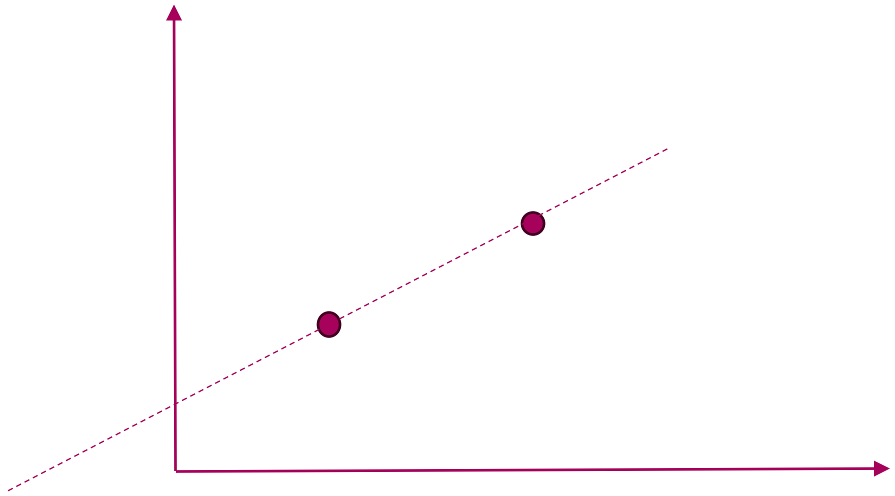
Gaussian Processes

Parsa Hariri

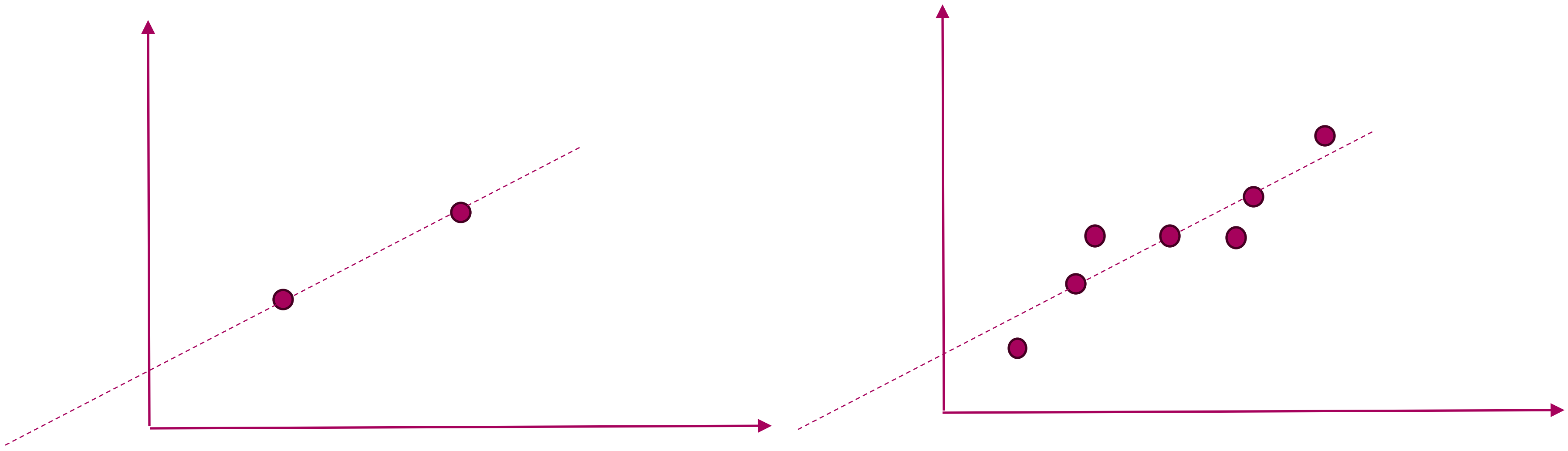
linear regression



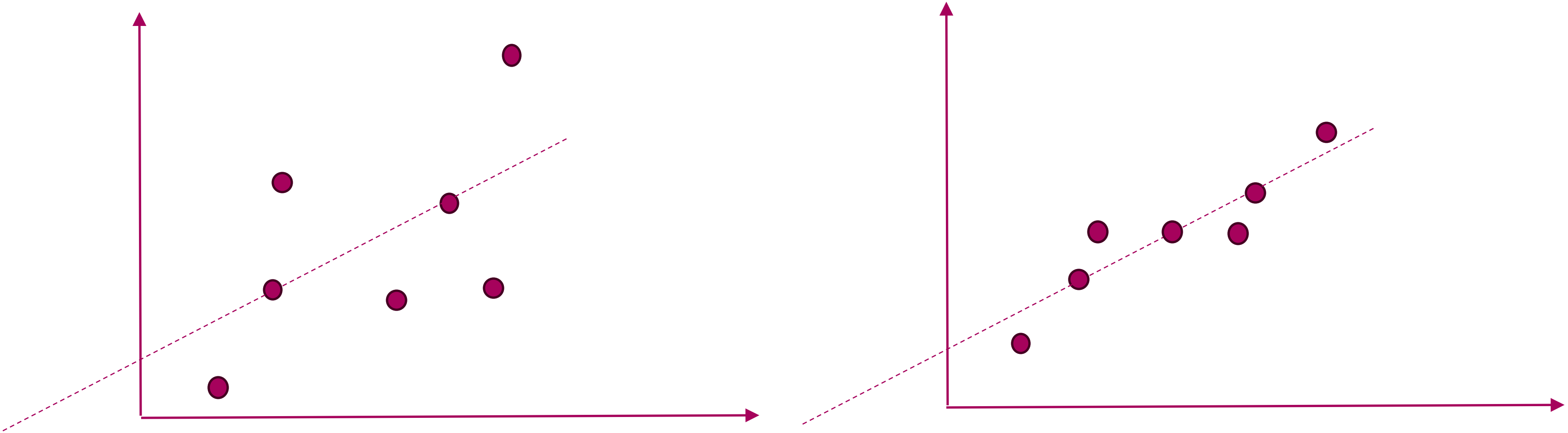
linear regression



linear regression



linear regression



Do you see any Problems?

Definition & Intuition

$$P(y(\mathbf{x}) \mid \mathbf{t}_N, \mathbf{X}_N) = \frac{P(\mathbf{t}_N \mid y(\mathbf{x}), \mathbf{X}_N) P(y(\mathbf{x}))}{P(\mathbf{t}_N \mid \mathbf{X}_N)}$$

Definition & Intuition

$$P(y(\mathbf{x}) \mid \mathbf{t}_N, \mathbf{X}_N) = \frac{P(\mathbf{t}_N \mid y(\mathbf{x}), \mathbf{X}_N) P(y(\mathbf{x}))}{P(\mathbf{t}_N \mid \mathbf{X}_N)}$$

$y(\mathbf{x})$

Find the parametrics function and then calculte the probbablity

Definition & Intuition

$$P(y(\mathbf{x}) \mid \mathbf{t}_N, \mathbf{X}_N) = \frac{P(\mathbf{t}_N \mid y(\mathbf{x}), \mathbf{X}_N) P(y(\mathbf{x}))}{P(\mathbf{t}_N \mid \mathbf{X}_N)}$$

$$P(y(\mathbf{x}))$$

Sum over finite number of functions, then no need for singe function probability !

Definition & Intuition

$$P(y(\mathbf{x}) \mid \mathbf{t}_N, \mathbf{X}_N) = \frac{P(\mathbf{t}_N \mid y(\mathbf{x}), \mathbf{X}_N) P(y(\mathbf{x}))}{P(\mathbf{t}_N \mid \mathbf{X}_N)}$$

$$P(y(\mathbf{x}))$$

One of the mode simple functions we can use is Gaussian process model

$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$

mean function

$$m(x) = \mathbb{E}[f(x)]$$

Definition & Intuition

$$P(y(\mathbf{x}) \mid \mathbf{t}_N, \mathbf{X}_N) = \frac{P(\mathbf{t}_N \mid y(\mathbf{x}), \mathbf{X}_N) P(y(\mathbf{x}))}{P(\mathbf{t}_N \mid \mathbf{X}_N)}$$

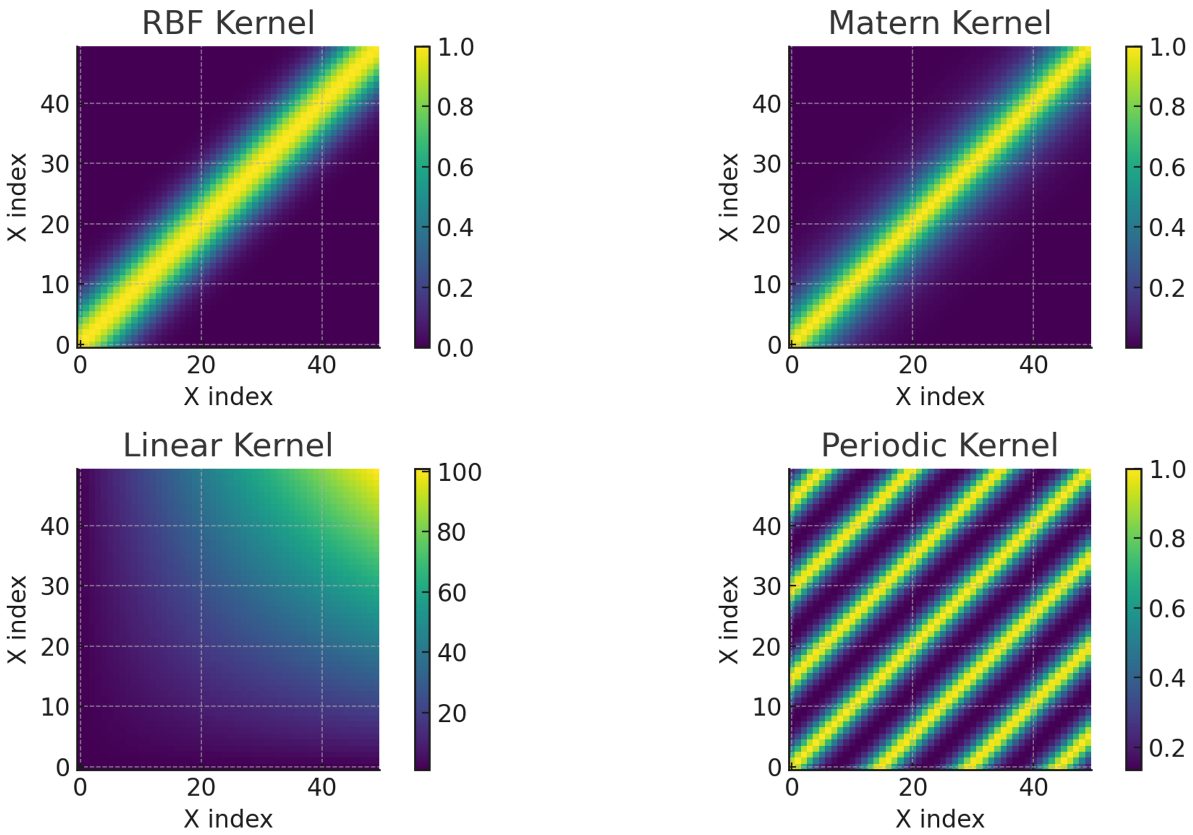
$$P(y(\mathbf{x}))$$

One of the mode simple functions we can use is Gaussian process model

$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$

covariance function (kernel)

$$k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))]$$



Constant : $K_{\mathrm{C}}(x, x') = C$

Linear: $K_{\mathrm{L}}(x, x') = x^{\mathsf{T}} x'$

white Gaussian noise: $K_{\mathrm{GN}}(x, x') = \sigma^2 \delta_{x,x'}$

Squared exponential: $K_{\mathrm{SE}}(x, x') = \exp\left(-\frac{d^2}{2\ell^2}\right)$

Ornstein–Uhlenbeck: $K_{\mathrm{OU}}(x, x') = \exp\left(-\frac{d}{\ell}\right)$

Matérn: $K_{\mathrm{Matern}}(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}d}{\ell}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}d}{\ell}\right)$

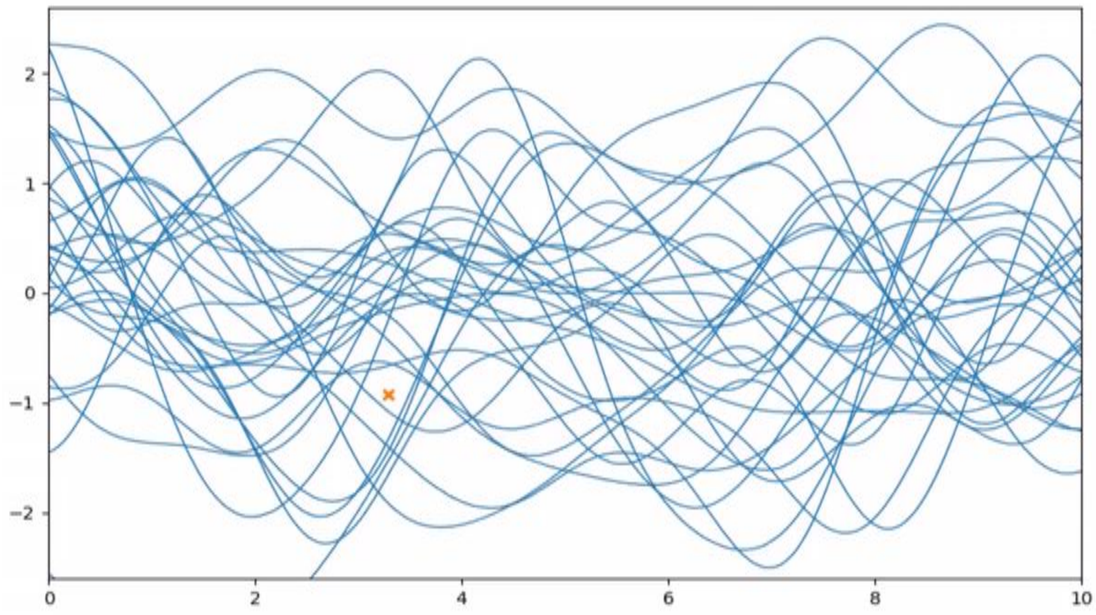
Periodic: $K_{\mathrm{P}}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(d/2)\right)$

Rational quadratic: $K_{\mathrm{RQ}}(x, x') = \left(1 + d^2\right)^{-\alpha}, \quad \alpha \geq 0$

How to use it in Python?

```
>>> from sklearn.datasets import make_friedman2
>>> from sklearn.gaussian_process import GaussianProcessRegressor
>>> from sklearn.gaussian_process.kernels import DotProduct, WhiteKernel
>>> X, y = make_friedman2(n_samples=500, noise=0, random_state=0)
>>> kernel = DotProduct() + WhiteKernel()
>>> gpr = GaussianProcessRegressor(kernel=kernel,
...         random_state=0).fit(X, y)
>>> gpr.score(X, y)
0.3680...
>>> gpr.predict(X[:2,:], return_std=True)
(array([653.0, 592.1]), array([316.6, 316.6]))
```

Thank you for your attiontion!



Key Papers & Further Reading

- Brown (1827) observed jittering pollen grains
- Einstein & Smoluchowski (1905–1906) explained Brownian motion
- Perrin (1908–1909) validated atomic hypothesis
- Wiener & Kolmogorov (1920s–1940s) formalised GPs
- Mandelbrot & Van Ness (1968) introduced fractional Brownian motion
- Rasmussen & Williams (2006) popularised GPs in ML
- Metzler & Klafter (2000) reviewed anomalous diffusion
- Randomly scaled GPs model sub/superdiffusion
- Biophysical studies reveal anomalous transport
- Sohl-Dickstein et al. (2015) introduced diffusion models
- Ho et al. (2020) proposed denoising diffusion probabilistic models
- Song et al. (2021) developed score-based models

[13] [14]