SLAM 101

Lecture 00 Machine Learning1

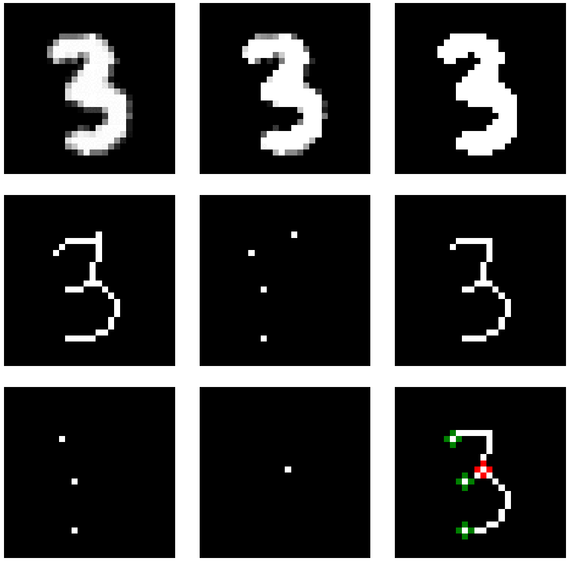
Ayoung Kim

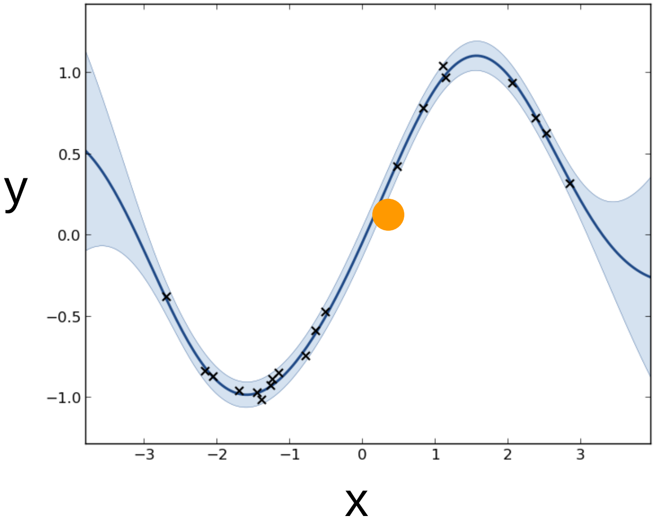
Overview

- Machine learning techniques in robotics
  - Regression
    - Gaussian process
    - Model fitting (RANSAC)
  - Clustering
    - K-means, dbscan

Introduction to Machine Learning

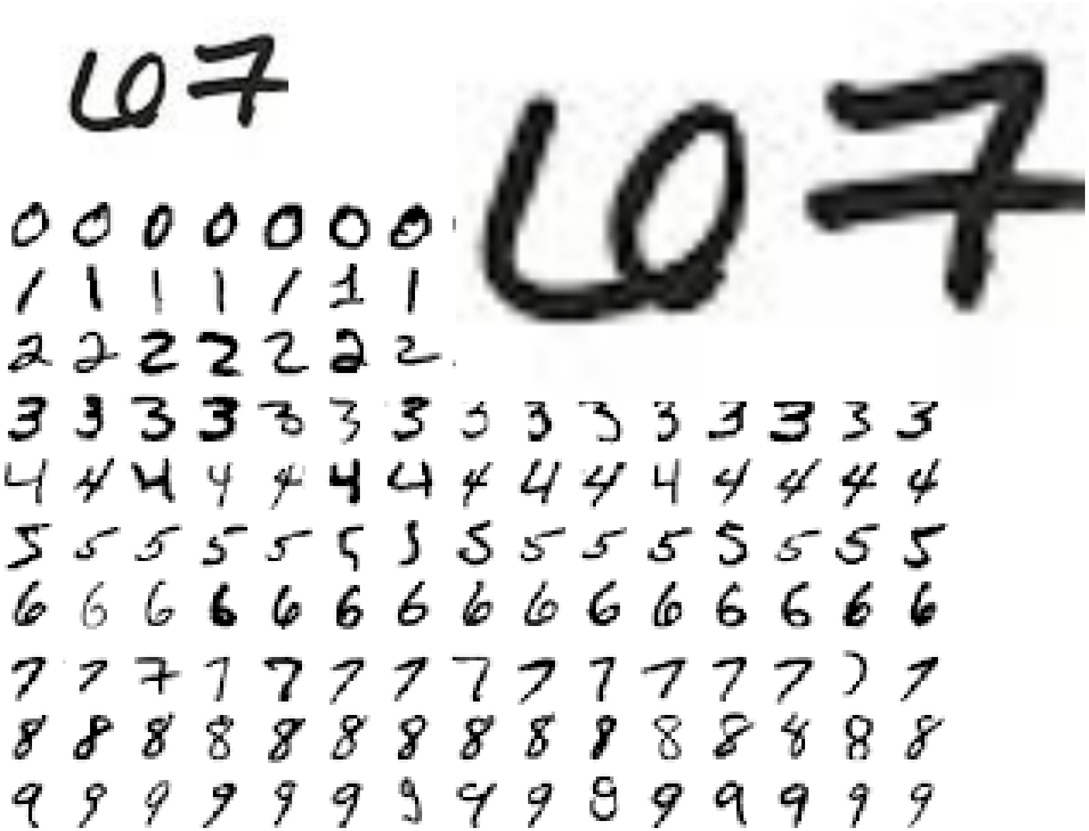
- Three types
  - Classification
  - Regression
  - Reinforcement learning





Examples

- Read the following

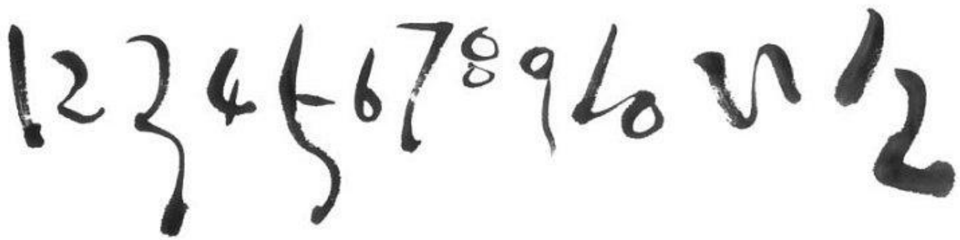


Training data

New data

Randomness

- Generalization is the ability to categorize correctly new examples that differ from those used in the training set
  - In most application, there is some inevitable *randomness* or *uncertainty*



  - The training data x (or the set X={x1,x2,…,xn}) can be viewed as a *random variable*
  - Probabilistic approaches

Supervised Learning

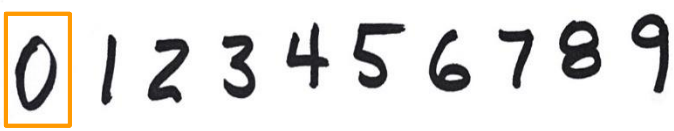
- Classification

  $y \in \{1, \cdots, M\}$

  - Example: handwritten numbers

    $y \in \{0, 1, \cdots, 9\}$

  - Train with labels



Label = 0

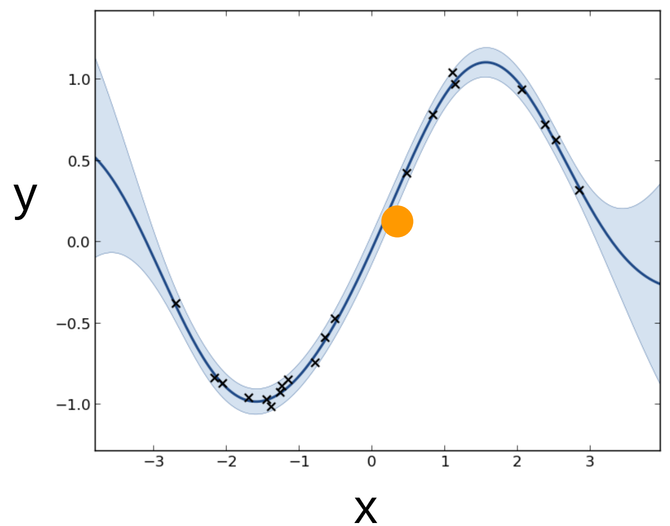
  - Predict the label of a new data

Label = ?

- Regression

  $y \in \mathcal{R}$

  - Example: curve fitting



  - Predict the y value given a new x

x = 0.234 **y = ?**

Unsupervised Learning

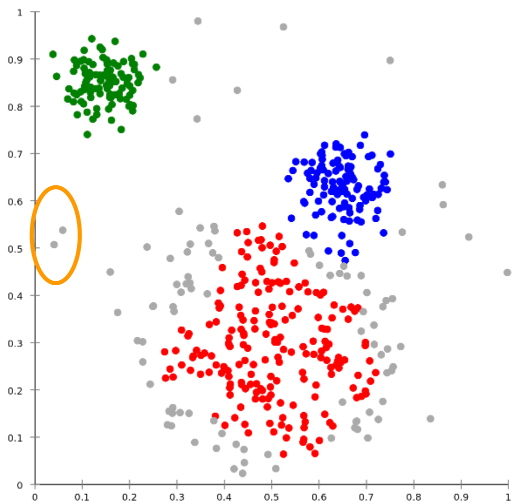
- The input data is without label

  $$X = \{\mathrm{x}_1, \cdots, \mathrm{x}_N\}$$

- The goal of unsupervised learning is to understand the structure in the data sample. No input-output relation is of interest.
- The primary unsupervised learning problems:
  - Clustering
  - Density estimation
  - Dimensionality reduction

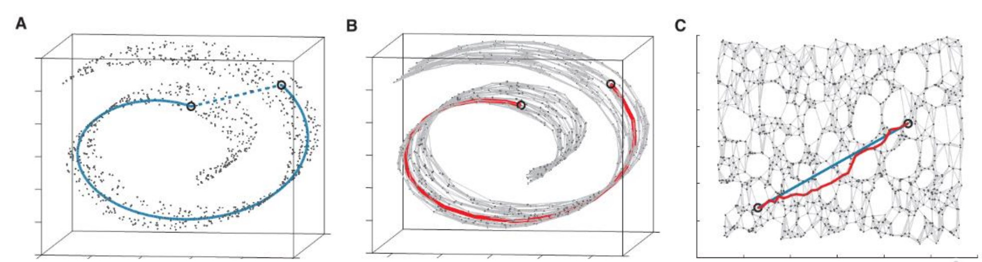
Unsupervised Learning

- Clustering
  - Group input data
  - Distance function
  - Example: 2D clustering



- Density estimation

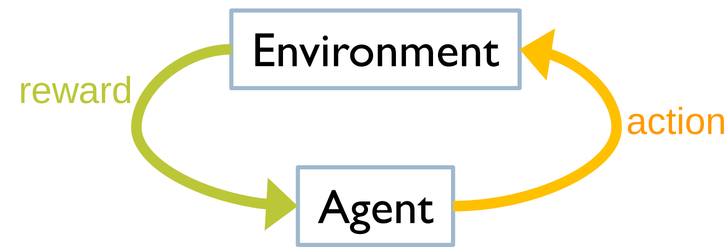
- Dimensionality reduction
  - Project higher dim data into lower dim
  - Example: Swiss roll dimension reduction



Reinforcement Learning (RL)

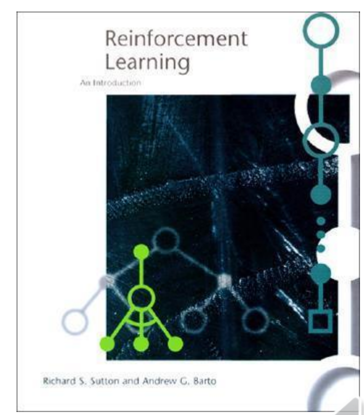
- The patterns are observed sequentially

$$X = \{\mathrm{x}_1, \cdots, \mathrm{x}_N\}$$



  - After each observation the learner takes an action
  - After each action it received a reward

- The goal of the RL is to determine a policy to maximize the long term reward



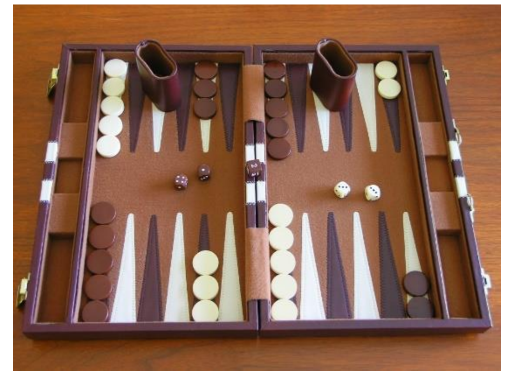
- RL is important in Robotics, Economics
  - Navigation, path planning

Reinforcement Learning (RL)

- Learning rules via *trial and error*
  - Backgammon (Tesauro 1994)
    - Neural network learned the game

Chess

Tic-tac-toe



Backgammon

- Policy: balancing between exploration and exploitation
  - Take a action you already know vs. adventure

1 2 3 4 5 6 7 8 9 10

₩ 5,000 per game

₩ 4,000 ₩ 1,000 ₩ 50,000 ₩ 6,000 ?

State

Action

Reward

Regression

Data Analysis

- Today topics
  - Regression
  - Clustering
- Objectives
  - Just to have an idea and know you can use “A” in your research

Least Square
Linear Regression

Least Square Linear Regression

- Regression vs. classification
  - y is continuous function vs. from a discrete set
  - Regression is supervised learning

- In regression problem, the training data is given

$$(\mathrm{x}_1, \mathrm{y}_1), \cdots, (\mathrm{x}_n, \mathrm{y}_n)$$

$$x \in \mathcal{R}^d$$
$$y \in \mathcal{R}$$

  - Here $(\mathrm{x}_i, \mathrm{y}_i)$ is realization of a random pair (X, Y)
  - Goal of regression is to predict the response y associated to a new input x

Least Square Linear Regression

- Regression model

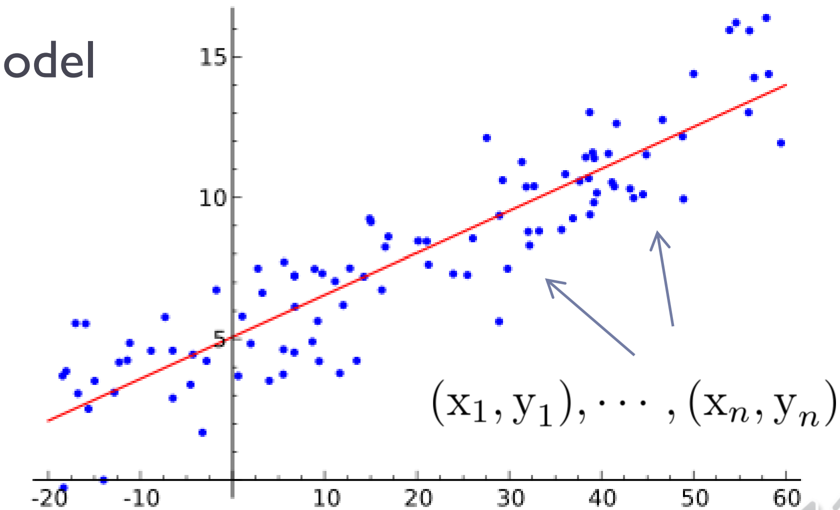
$$y = f(x) + \epsilon$$

ϵ is noise
$f \in$ some class of function

- Linear regression model
  - Assume the form of f is a linear model

$$f(x) = \beta^\top x + \beta_0$$

Estimate β, β_0



Least Square Linear Regression

- # of unknowns < # equations

$$\beta, \beta_0 \qquad\qquad (\mathrm{x}_1, \mathrm{y}_1), \cdots, (\mathrm{x}_n, \mathrm{y}_n)$$

  - No exact solution →Optimal solution
  - Objective function = minimize what?

- Least square linear regression
  - Select β, β_0 that minimize the sum of squared error

$$\begin{aligned} \mathrm{y} &= f(\mathrm{x}) + \epsilon \\ f(\mathrm{x}) &= \beta^\top \mathrm{x} + \beta_0 \end{aligned}$$

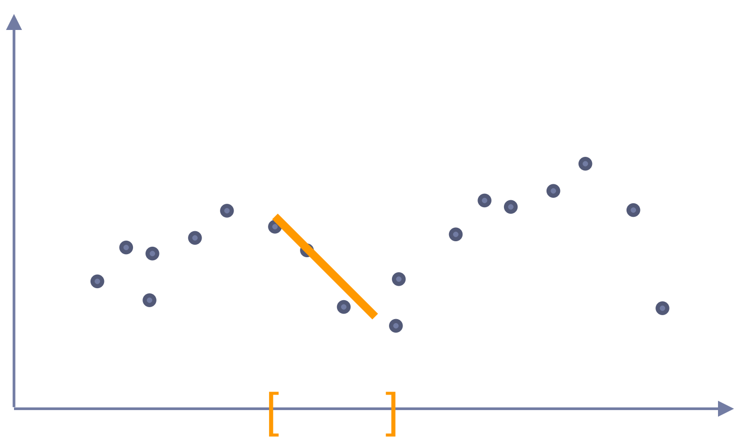
$$\begin{aligned} \epsilon &= \mathrm{y} - f(\mathrm{x}) \\ &= \mathrm{y} - \beta^\top \mathrm{x} - \beta_0 \end{aligned}$$

$$E = \sum_{i=1}^{n} \left(\mathrm{y}_i - \beta^\top \mathrm{x}_i - \beta_0\right)^2$$

Locally Linear Regression

- When the data is nonlinear
  - So far we only know linear regression
  - Given $(\mathrm{x}_1, \mathrm{y}_1), \cdots, (\mathrm{x}_n, \mathrm{y}_n)$

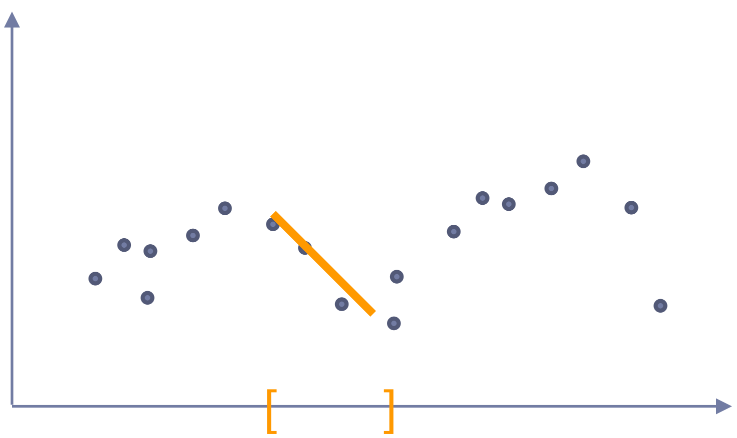
$$x \in \mathcal{R}^d$$
$$y \in \mathcal{R}$$



Locally Linear Regression

- Easy two game plans
  - Locally averaging

$$\begin{aligned}\hat{f}(\mathrm{x}) &= \text{avg. } y_i \text{ among } \mathrm{x}_i \\ &\quad \text{such that } |\mathrm{x}-\mathrm{x}_i| < \delta \\ &= \frac{1}{||i : |\mathrm{x}-\mathrm{x}_i| < \delta||}\sum y_i\end{aligned}$$



Gaussian Process

Gaussian Process

- The prediction using kernel

$$
\begin{aligned}
p(f_*|x_*, X, \mathrm{y}) &= \mathcal{N}(\phi_* \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathrm{y} \\
&, \quad \phi_*^\top \Sigma_p \phi_* - \phi_*^\top \Sigma_p \Phi (\sigma_n^2 I + K)^{-1} \Phi^\top \Sigma_p \phi_*)
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{mean}(f_*) &= \phi_* \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathrm{y} \\
\mathrm{cov}(f_*) &= \phi_*^\top \Sigma_p \phi_* - \phi_*^\top \Sigma_p \Phi (\sigma_n^2 I + K)^{-1} \Phi^\top \Sigma_p \phi_*
\end{aligned}
$$

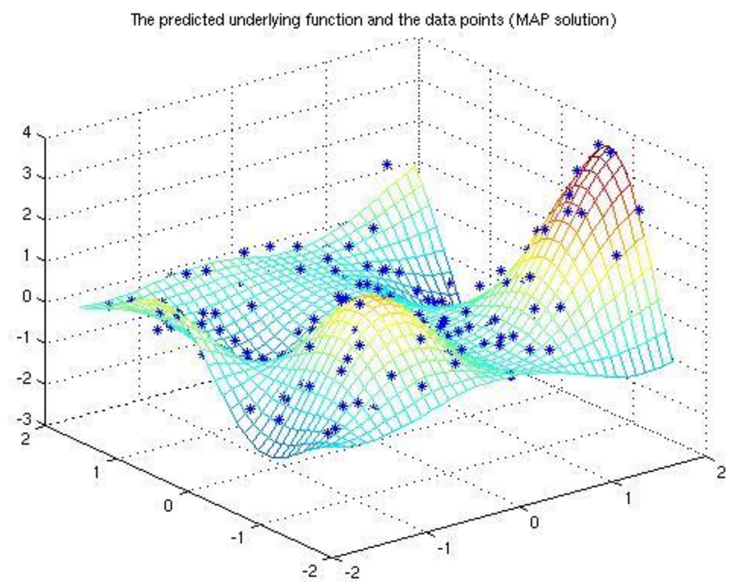
$$
\phi_*^\top \Sigma_p \phi_* = k(\mathrm{x}_*, \mathrm{x}_*),\ \phi_*^\top \Sigma_p \Phi = \mathrm{k}_*,\ \Phi^\top \Sigma_p \Phi = K
$$

$$
\boxed{
\begin{aligned}
\mathrm{mean}(f_*) &= \mathrm{k}_* (K + \sigma_n^2 I)^{-1} \mathrm{y} \\
\mathrm{cov}(f_*) &= k(\mathrm{x}_*, \mathrm{x}_*) - \mathrm{k}_*^\top (\sigma_n^2 I + K)^{-1} \mathrm{k}_*
\end{aligned}
}
$$

Gaussian Process

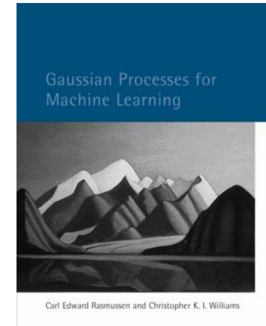
- GP in a nutshell

$$\begin{aligned}\mathrm{mean}(f_*) &= \mathrm{k}_*(K+\sigma_n^2 I)^{-1}\mathrm{y} \\ \mathrm{cov}(f_*) &= k(\mathrm{x}_*,\mathrm{x}_*) - \mathrm{k}_*^\top(\sigma_n^2 I + K)^{-1}\mathrm{k}_*\end{aligned}$$



Gaussian Process – Algorithm

- Gaussian Processes for Machine Learning (GPML)
  - http://www.gaussianprocess.org/gpml/



input: X (inputs), $\mathbf{y}$ (targets), k (covariance function), σ_n^2 (noise level), $\mathbf{x}_*$ (test input)

$$L := \text{cholesky}(K + \sigma_n^2 I)$$

$$\left.\begin{array}{l} \boldsymbol{\alpha} := L^\top \backslash (L \backslash \mathbf{y}) \\ \bar{f}_* := \mathbf{k}_*^\top \boldsymbol{\alpha} \end{array}\right\} \text{ predictive mean eq. (2.25)}$$

$$\left.\begin{array}{l} \mathbf{v} := L \backslash \mathbf{k}_* \\ \mathbb{V}[f_*] := k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{v}^\top \mathbf{v} \end{array}\right\} \text{ predictive variance eq. (2.26)}$$

$$\log p(\mathbf{y}|X) := -\tfrac{1}{2}\mathbf{y}^\top \boldsymbol{\alpha} - \textstyle\sum_i \log L_{ii} - \tfrac{n}{2} \log 2\pi \qquad \text{eq. (2.30)}$$

return: $\bar{f}_*$ (mean), $\mathbb{V}[f_*]$ (variance), $\log p(\mathbf{y}|X)$ log marginal likelihood

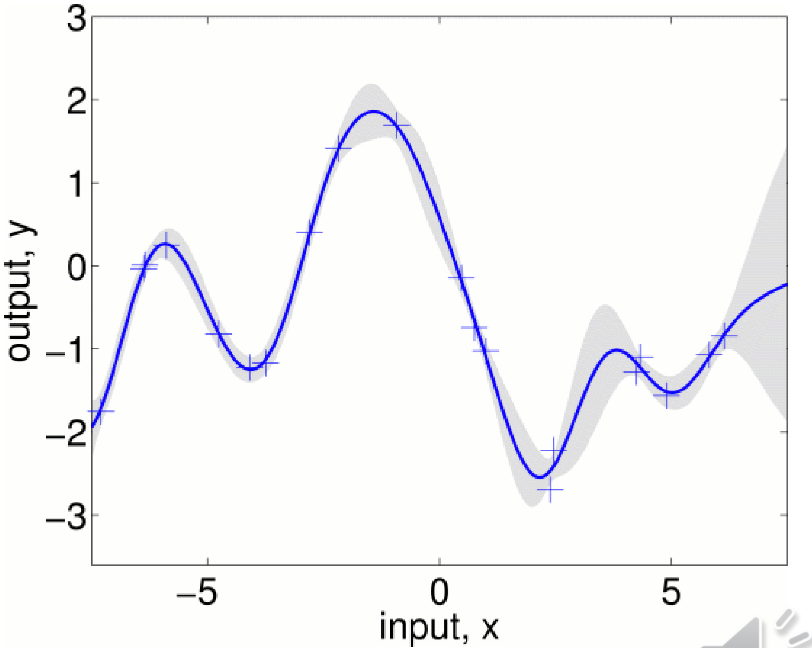
GP – Covariance Function

- Selection of covariance function
  - Covariance function choice
  - Related hyperparameters

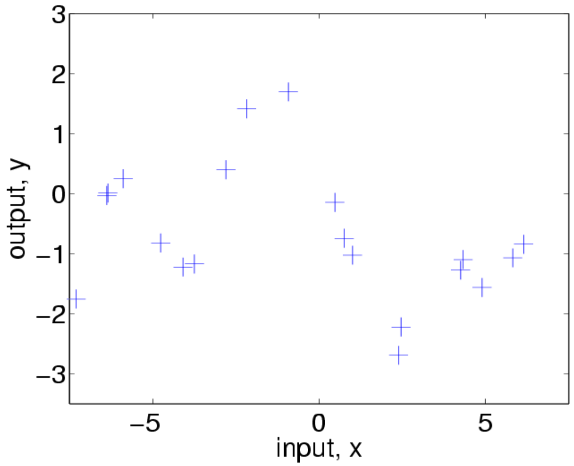
- Example: SE (Squared exponential) function with three params

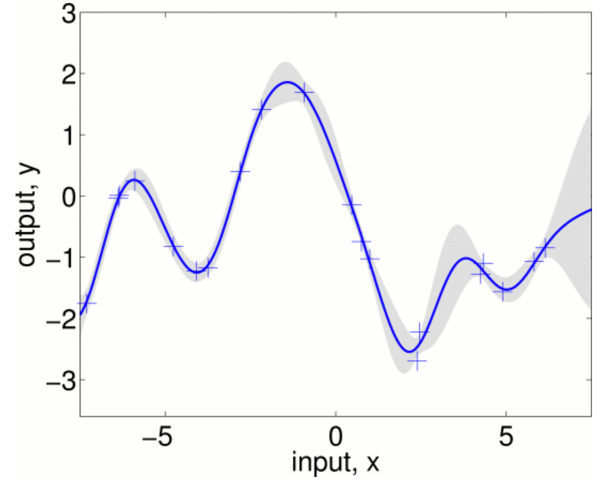
$$k(\mathrm{x}_p, \mathrm{x}_q) \quad = \quad \sigma_f^2 \exp(-\frac{1}{2\ell^2}(\mathrm{x}_p - \mathrm{x}_q)^2) + \sigma_n^2 \delta_{pq}$$

Hyperparameters: σ_f, ℓ, σ_n

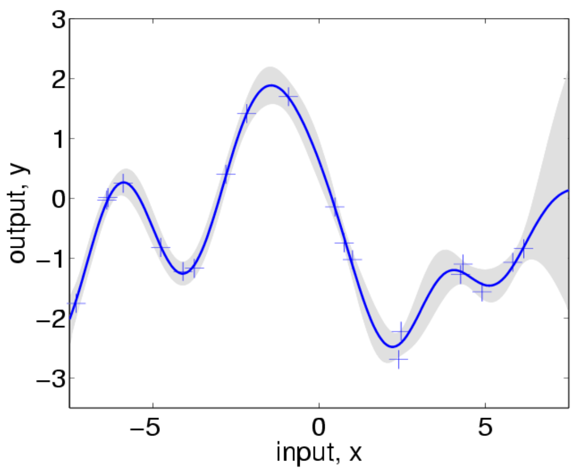


GP – Covariance Function

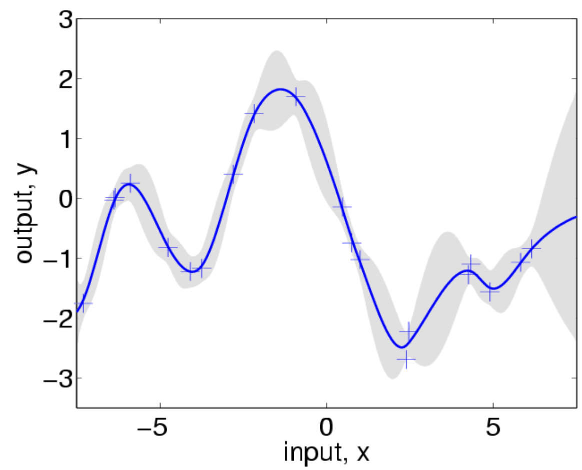




SE1



SE2



Matern

GP – Covariance Function

$$f(x,x') = \exp\left(\frac{|x-x'|}{l}\right)^2$$

$$f(x,x') = \frac{2}{\pi}\sin^{-1}\left(\frac{2\tilde{x}^\top\Sigma\tilde{x}}{\sqrt{(1+2\tilde{x}^\top\Sigma\tilde{x})(1+2\tilde{x}^\top\Sigma\tilde{x})}}\right) \quad \tilde{x} = (x_1, x_2, \cdots, x_d)^\top$$

$$f(x,x') = \begin{cases} \sigma_o\left(\frac{2+cos(2\pi d/l)}{3}(1-d/l) + \frac{\sin(2\pi d/l)}{2\pi}\right), & d < l \\ 0, & \text{o.w.} \end{cases}$$

$$f(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}|x-x'|}{l}\right)^\nu K_v\left(\frac{\sqrt{2\nu}|x-x'|}{l}\right)$$

$$K_v = \exp\left(-\frac{\sqrt{2\nu}|x-x'|}{l}\right)\frac{\Gamma(p+1)}{\Gamma(p+2)} \cdot \sum_{i=0}^{p}\frac{(p+i)!}{(p-i)!}\left(\frac{\sqrt{8\nu}|x-x'|}{l}\right)^{(p-i)}$$

$$f(x,x') = \exp\frac{|x-x'|}{l} \qquad f(x,x') = \exp\left(\frac{|x-x'|}{l}\right)^\gamma$$

Next Lecture

- RANSAC
- Clustering

Thank you very much !!