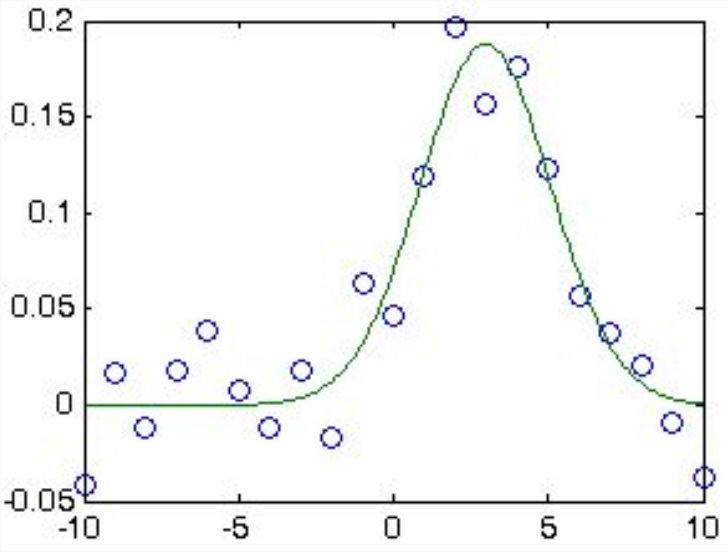
Gaussian Processes : Regression

UG Project

Submitted By :
Ishita Ankit
Roll no.13316
Dept. MTH

Generative models for Prediction



Task: Fitting a model for the given set of datapoints.

Types of models:

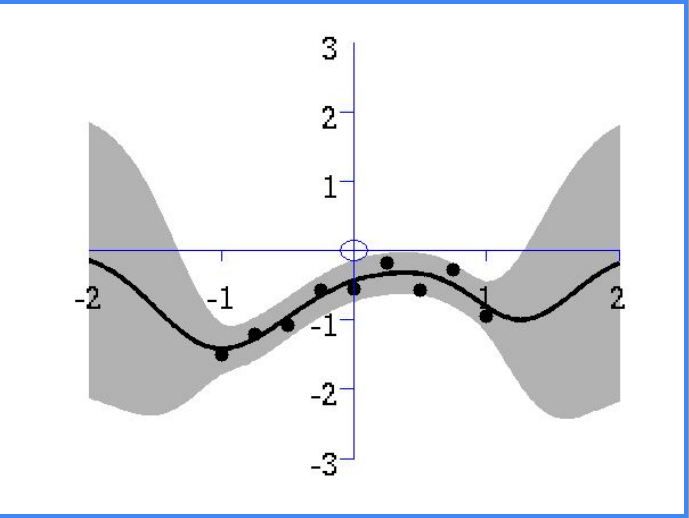
Discriminative:provides a model only for the target variable(s) conditional on the observed variables

Generative:a full probabilistic model of all variables.It is used in machine learning for either modeling data directly (i.e., modeling observations drawn from aprobability density function), or as an intermediate step to forming a conditional probability density function.

Advantages of Probabilistic Models

- ➔ Can get estimate of the the uncertainty in the parameter estimates via the posterior distribution
- ➔ Useful when we only have limited data for learning each parameter
- ➔ Can get estimate of the the uncertainty in the model's predictions
- ➔ Can handle missing and noisy data in a principled way
- ➔ Easy/more natural to do semi-supervised learning, active learning, etc.
- ➔ Can generate (synthesize) data by simulating from the data distribution
- ➔ Hyperparameters can be learned from data (need not be tuned)
- ➔ Simple models can be neatly combined to solve complex problems

Gaussian Process Models



Consider the problem of nonlinear regression:
You want to learn a function f with error bars from data
D = {X, y}

A Gaussian process defines a distribution over functions $p(f)$
which can be used for Bayesian regression:

$$p(f \mid \mathcal{D}) = \frac{p(f)p(\mathcal{D} \mid f)}{p(\mathcal{D})}$$

Gaussian Processes

Definition: A Gaussian process is a collection of random variables, any finite number of which have a consistent joint Gaussian distribution.(Rasmussen and Williams, 2006).

Similar to a Gaussian distribution, which is fully specified by a mean vector and a covariance matrix, a GP is fully specified by a mean function $m_h(\cdot)$ **and a** *covariance function.*

We define a distribution over functions, $p(h)$, where h is a function mapping some input space X to

$$h : X \rightarrow \mathfrak{R}.$$

Using the definition $h= (h(x_1), \ldots, h(x_n))$ is a finite collection of GP hence has a joint Gaussian Distribution.

Gaussian Processes

- Training Data : $\mathcal{D}$: $\{x_n, y_n\}$ $x_n \in \mathbb{R}^d$ $y_n \in \mathbb{R}$.
- Assume the labels to be noisy function of the inputs

$$y_i = h(x_i) + \epsilon_i$$

$\epsilon_i = N(0, \sigma^2)$: Gaussian measurement Noise

- Assume a gaussian prior over the function *h* $p(h) = \mathcal{N}(h \mid 0, K)$
- Thus the likelihood model

$$p(y_n \mid h_n) = \mathcal{N}(y_n \mid h_n, \sigma^2)$$

- For N i.i.d. responses, the joint likelihood can be written as

$$p(Y \mid h) = \mathcal{N}(Y \mid h, \sigma^2)$$

- Use Bayes rule to get the posterior on *h*

$$p(h \mid y) = \frac{p(h)p(y \mid h)}{p(y)}$$

Reference: http://www.gaussianprocess.org

Gaussian Processes

- Prior over the function *h* $p(h) = \mathcal{N}(h \mid 0, K)$
- The likelihood model

$$p(y \mid h) = \mathcal{N}(y \mid h, \sigma^2)$$

- The marginal distribution of the training data responses y

$$p(y) = \int p(y \mid h) p(h) dh = \mathcal{N}(y \mid 0, K + \sigma^2 I_{\mathcal{N}}) = \mathcal{N}(y \mid 0, C_{\mathcal{N}})$$

Reference: http://www.gaussianprocess.org

Gaussian Prior

- In the GP model, we have to specify the prior mean function and the prior covariance function.
- we consider a prior mean function $m_h \equiv 0$ and use the squared exponential (SE)covariance function with automatic relevance determination

$$k_{SE}(x_p, x_q) := \alpha^2 \exp(-\tfrac{1}{2}(x_p - x_q)\Lambda^{-1}(x_p - x_q)) \quad x_p, x_q \in \mathbb{R}^D,$$

- $\Lambda = \text{diag}([\ell_1^2, \ldots, \ell_D^2])$ is a diagonal matrix of squared characteristic length-scales and α is the signal variance of the latent function h. These come under the hyperparameters of the function h.
- With the SE covariance function in the above equation, we assume that the latent function h is smooth and stationary.

Reference: http://www.gaussianprocess.org

Posterior

- After having observed function values y with $y_i = h(x_i) + \varepsilon_i$, $i = 1, \ldots, n$, for a set of input vectors X, Bayes' theorem yields

$$p(h \mid X, y, \theta) = \frac{p(h \mid \theta) p(y \mid h, X, \theta)}{p(y \mid X, \theta)}$$

- We assume that the observations y_i are conditionally independent given X.
- Therefore, the likelihood of h factors is:

$$p(y \mid h, X, \theta) = \prod p(y_i \mid h(x_i), \theta) = \prod \mathcal{N}(y_i \mid h(x_i), \sigma_\varepsilon^2) = \mathcal{N}(y \mid h(X), \sigma_\varepsilon^2 I).$$

- Given a Gaussian prior, hyperparameters and a gaussian likelihood, the posterior is also a GP with mean and Covariance function given by :

$$\square_h[h(x') \mid X, y, \theta] = k_h(x', X)(k_h(X, X) + \sigma_\varepsilon^2 I)^{-1} y,$$

$$cov_h[h(x'), h(x) \mid X, y, \theta] = k_h(x, x) - k_h(x, X)(k_h(X, X) + \sigma_\varepsilon^2 I)^{-1} k_h(X, x)$$

Reference: http://www.gaussianprocess.org

Evidence Maximization

- The flat prior on the hyper-parameters has computational advantages: It makes the posterior distribution over θ proportional to the marginal likelihood in equation,

  that is, $p(\theta|X, y) \propto p(y|X, \theta)$.

- To find a vector of “good” hyper-parameters, we therefore maximize the marginal likelihood in equation with respect to the hyper-parameters as recommended by MacKay (1999).

- In particular, the log-marginal likelihood (log-evidence) is

$$\log p(y|X,\theta) = \log \int p(y|h,X,\theta)\, p(h|\theta)\, dh = -\tfrac{1}{2}\, y\, (K_\theta + \sigma_\varepsilon^2 I)^{-1} y - \tfrac{1}{2} \log |K_\theta + \sigma_\varepsilon^2 I| - (D/2) \log(2\pi).$$

- Hence Maximizing the above equation we get the hyper-parameter vector($\hat{\theta}$)

$$\in \arg\max \log p(y|X, \theta)$$

Reference: http://www.gaussianprocess.org

Predicting Data

- Infer t_{N+1} given $\mathbf{t}_N$: $P(t_{N+1}|\mathbf{t}_N) = \frac{P(t_{N+1}, \mathbf{t}_N)}{P(\mathbf{t}_N)}$

Simple, because conditional distribution is also a Gaussian

- $$P(t_{N+1}|\mathbf{t}_N) \propto \exp\left[-\frac{1}{2}\begin{bmatrix} \mathbf{t}_N & t_{N+1} \end{bmatrix} \mathbf{C}_{N+1}^{-1} \begin{bmatrix} \mathbf{t}_N \\ t_{N+1} \end{bmatrix}\right]$$
- Use incremental form of $\mathbf{C}_{N+1}$

$$\mathbf{C}_{N+1} \equiv \begin{bmatrix} \begin{bmatrix} \mathbf{C}_N \end{bmatrix} \begin{bmatrix} \mathbf{k} \end{bmatrix} \\ \begin{bmatrix} \mathbf{k}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \kappa \end{bmatrix} \end{bmatrix}$$

Reference: http://www.gaussianprocess.org

Predicting Data

- We can rewrite this equation
  - Use partitioned inverse equations to get from $\mathbf{C}_N^{-1}$ $\mathbf{C}_{N+1}^{-1}$
  - $$P(t_{N+1}|\mathbf{t}_N) = \frac{1}{Z}\exp\left[-\frac{(t_{N+1}-\hat{t}_{N+1})^2}{2\sigma^2_{\hat{t}_{N+1}}}\right]$$
  -

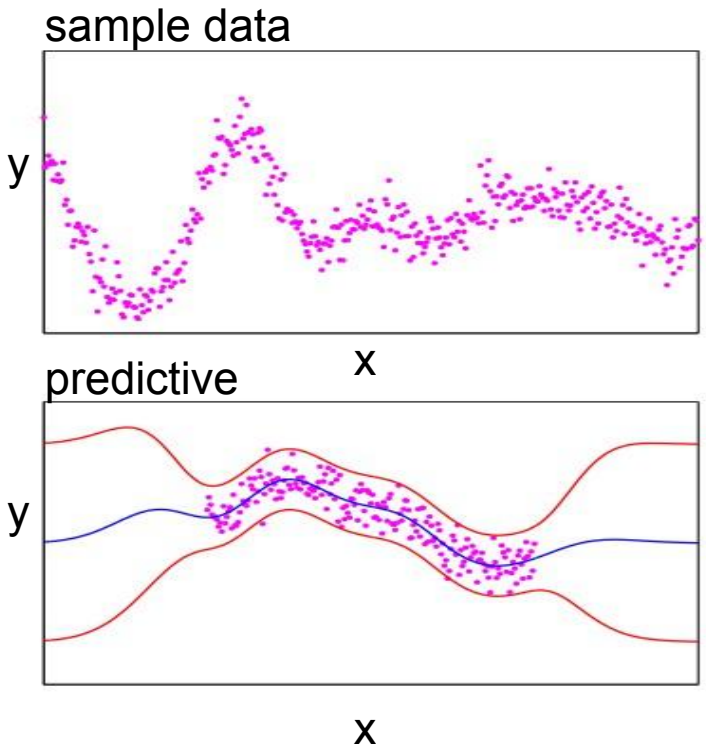
Predictive mean: $\hat{t}_{N+1} = \mathbf{k}^T\mathbf{C}_N^{-1}\mathbf{t}_N$

And Covariance $\sigma^2_{\hat{t}_{N+1}} = \kappa - \mathbf{k}^T\mathbf{C}_N^{-1}\mathbf{k}$

Reference: http://www.gaussianprocess.org

Gaussian Regression

Gaussian observation noise: $y_n = f_n + \epsilon_n$, where $\epsilon_n \sim N(0, \sigma^2)$



marginal likelihood

$$p(y|X) = N(0, K_N + \sigma^2 I)$$

predictive distribution

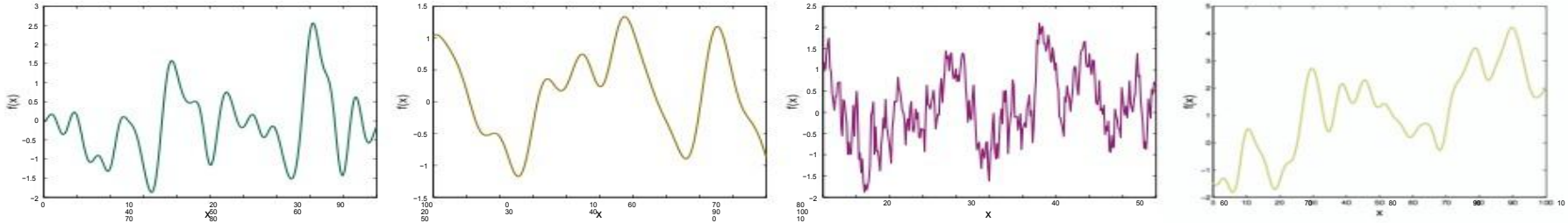
$$p(y_*|x_*, X, y) = N(\mu_*, \sigma^2_*)$$

$$\mu_* = K_{*N}(K_N + \sigma^2 I)^{-1} y$$

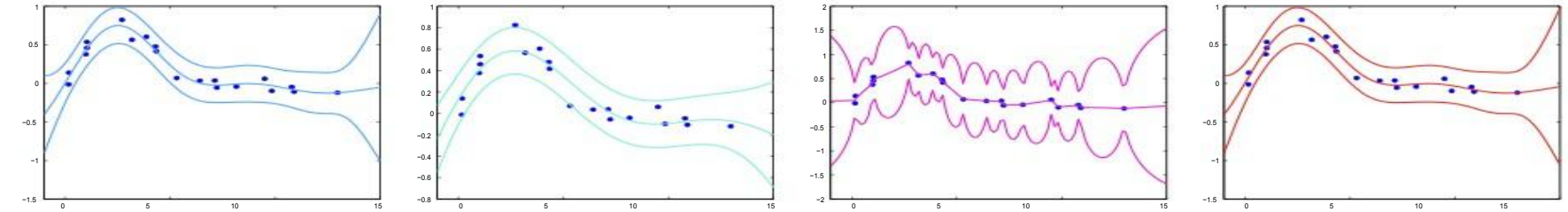
$$\sigma^2_* = K_{**} - K_{*N}(K_N + \sigma^2 I)^{-1} K_{N*} + \sigma^2$$

Predicting using different K(x,x')

A sample from the prior for each covariance function:



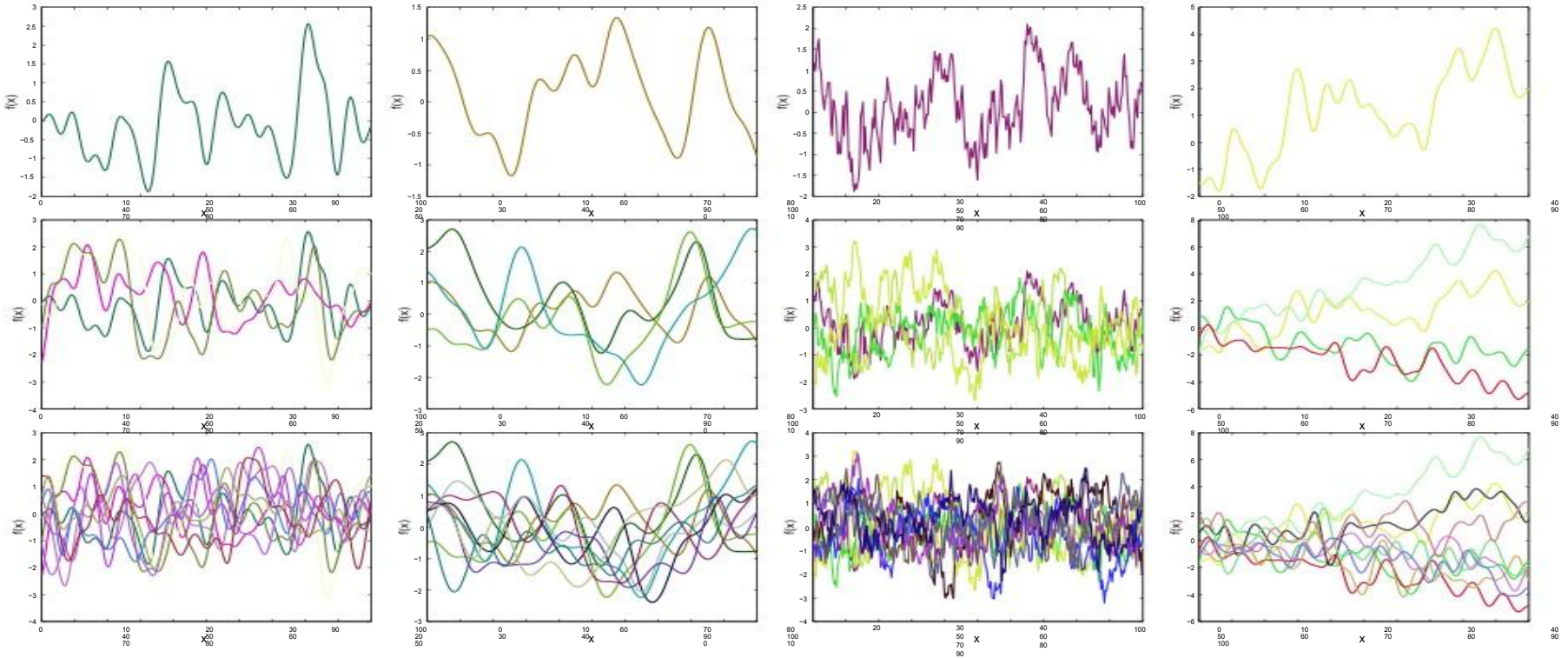
Corresponding predictions:



Why Gaussian ?

- Marginals of Gaussians are Gaussian
- Conditionals of Gaussians are Gaussian
- Variety of covariance functions can be used
- Non- Parametric models

Varying Covariance Functions



Conclusions

- One of the recent application of Gaussian Process in Reinforcement Learning is being used in training the robots to make them autonomous with fewer trials.
- Gaussian are self conjugates which makes them strong tools for Bayesian Learning.
- Gaussian Processes are strong, flexible and simple models and have wide range of applications and the years to come it shall be a strong tool to solving sturdy problems.

Thank you