

Topic \Rightarrow Gaussian Process Regression Model.

Multivariate Gaussian Distribution \Rightarrow Multivariate Gaussian Distribution (~~MVND~~) (MVG D) is what Gaussian Processes are based upon. It is defined by mean vector and Covariance matrix.

$$x \sim \mathcal{N}(\mu, \Sigma)$$

It possesses two properties:

i) Marginalisation \Rightarrow With Joint Gaussian distribution, it can be written as

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \right)$$

marginalised y with resulting x can be written as,

$$p(x) = \mathcal{N}(\mu_x, \Sigma_{xx})$$

ii) Conditioning \Rightarrow

$$p(x|y) = \mathcal{N}(\mu_x', \Sigma_{xx}')$$

$$\text{where } \mu_x' = \mu_x + \Sigma_{xy} \Sigma_{yy}^{-1} (y - \mu_y)$$

$$\Sigma_{xx}' = \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx}$$

Now for Gaussian Process, in the same way we define GP as

$$f(x) = \text{GP}(\mu(x), k(x, x'))$$

where $\mu(x)$ = mean function.

$k(x, x')$ = Kernel function. (discussed later)

x = vector

$f(x)$ = real-valued function.

kernels : These are functions that are used to generate covariance matrix.

In this case our kernel function has 2 parameters :

σ : scale of variance

l : influence of pts to neighbouring points.

$$k(x_i, x_j) = \sigma^2 \cdot \exp\left(\frac{(x_i - x_j)^2}{2l^2}\right)$$

*** Function taken from Prof. Nipun Batta's lecture Notes (ES-654).

For Given test data x' , we have to predict y'
Hence,

$$\begin{pmatrix} y \\ y' \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_y \\ \mu_{y'} \end{bmatrix}; \begin{pmatrix} K & K' \\ K'^T & K'' \end{pmatrix}\right)$$

$$K = \text{kernel}(Y, Y)$$

$$K' = \text{kernel}(Y, Y')$$

$$K'' = \text{kernel}(Y', Y')$$

So,

$$p(y' | x', x, Y) \propto \mathcal{N}(y', \Sigma')$$

where $\mu' = \mu' + K'^T K^{-1} (x - \mu)$

$$\Sigma' = K'' - K'^T K^{-1} K'$$

Since, we have to perform matrix inversion while trying to find updated covariance. In practical, matrix inversions are avoided due to reason:

i) Numerically Unstable.

ii) Computationally heavy \Rightarrow Calculating inverse takes $O(N^3)$ time complexity.

Since K is semi-positive definite matrix, it can be decomposed ~~into~~ using cholesky decomposition.

$$K = LL^T$$

where $L \equiv$ Lower-Triangular Matrix with entries $\in \mathbb{R} > 0$.

Now, we know that

$$\mu_0 = \mu' + K'^T K' (x - \mu)$$

$$\text{let } \alpha = K' (x - \mu)$$

$$\alpha = (LL^T)^{-1} (x - \mu)$$

$$\alpha = L^{-T} L^{-1} (x - \mu)$$

$$\text{let } \gamma = L^{-1} (x - \mu)$$

$$L\gamma = (x - \mu)$$

$$\gamma = L^{-1} (x - \mu)$$

then

$$\alpha = L^{-T} \gamma$$

$$\alpha = L^{-T} (L^{-1} (x - \mu))$$

So

$$\mu_0 = \mu' + K'^T \alpha$$

Similarly for Σ_0

$$\text{let } V = L^{-1} K'$$

$$V = L^{-1} K'$$

$$V^T V = K'^T L^{-T} L^{-1} K'$$

Hence

$$\Sigma_0 = K'' - V^T V$$