# Gaussian Process 01 Introduction

Chen Gong

13 December 2019

本小节我们将进入 Gaussian Process 的学习。Gaussian 自然指的就是 Gaussian Distribution，而 Process 指的就是随机过程。在一维的 Gaussian Distribution 中我们可以令 $p(x) = \mathcal{N}(\mu, \sigma^2)$。如果对应到高维高斯分布的话，也就是 (Multivariate Gaussian Distribution) 也就是我们通常意义上说的 Gaussian Network，对于任意的 $x \in \mathbb{R}^p$，有 $p(x) = \mathcal{N}(\mu, \Sigma)$，且 $\Sigma$ 是一个 $p \times p$ 维的向量，$p < +\infty$。如果是一个无限维的 Gaussian Distribution，那么就是我们今天要讨论的 Gaussian Process 了。首先我们给出 Gaussian Process 的详细定义，Gaussian Process：定义在连续域上的无限多个高维随机变量所组成的随机过程。所谓连续域指的就是时间或者空间。下面我们来进行详细的解释。

## 1 Gaussian Process 解释

我们假设有一组随机变量 $\{\xi_t\}_{t\in T}$，$T$ 是一个连续域，如果 $\forall n \in N^+$，都有 $t_1, t_2, \cdots, t_n \in T$，并且存在一个约束条件 $s.t.$ $\{\xi_{t_1}, \xi_{t_2}, \cdots, \xi_{t_n}\} \triangleq \xi_{t_1 \sim t_n} \sim \mathcal{N}(\mu_{t_1\sim t_n}, \Sigma_{t_1\sim t_n})$。那么我们就称 $\{\xi_t\}_{t\in T}$ 是一个 Gaussian Process。那么，我们怎么来通俗的理解这个概念呢? 也就是说有一系列在时间上或空间上连续的点，他们之间分开看都是符合一个高斯分布的，而合起来看则是符合一个多维高斯分布的。也就是如下图所示的，在五个不同的时刻有 5 个不同的点，之上的随机变量为 $\{\xi_{t_1}, \xi_{t_2}, \xi_{t_3}, \xi_{t_4}, \xi_{t_5}\}$，他们分别都符合一个高斯分布。

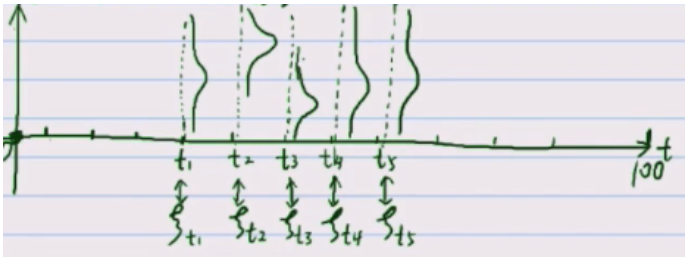


图 1: 多个点符合高斯分布的情况

## 2 Gaussian Process 举例说明

为了帮助大家更好的来理解高斯分布，我们在这里讲一个故事。假如一个人的一生可以活到 100 岁，横坐标就为时间 $t$，而纵坐标表示的为在这个时间点的表现值。(大概就这样将就的理解一下)。其中，$t \in [0, 100]$，每一个 $\xi_t \sim \mathcal{N}(\mu_t, \sigma_t)$。这是什么意思，也就是在每一个时刻这个人的表现值都符合一个独立的高斯分布，也就是在 $t$ 时刻他的表现为 $[0, 100]$。

现在，我们做一个假设，假设一个人，当 $t=0$ 的时刻，他的一生就已经确定了，也就是他的每一个时刻的表现值都会符合一个确定的 Gaussian Distribution，$\mu_t$ 和 $\sigma_t^2$ 都是确定的。假设人可以活很多次，每个点的表现值都是一个高斯分布，那么他每一生都将是不一样的，没过一生都是从高斯过程中的一次采样，如下图所示，

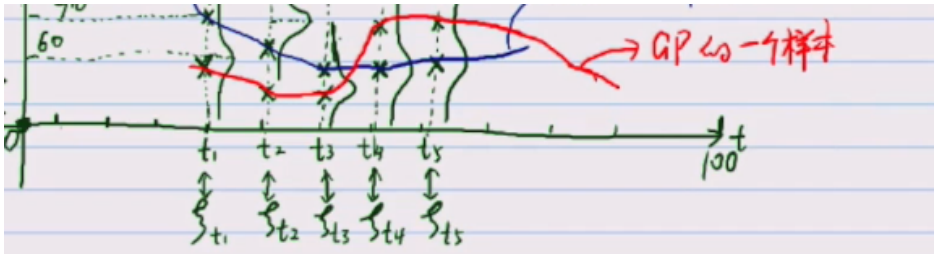


图 2: 高斯过程的样本

所以，$\{\xi_{t_1}, \xi_{t_2}, \cdots, \xi_{t_n}\}$，本身就是一个高斯分布，他们联合起来也是高斯分布，任何一个采样都属于高斯分布，我们可以看成是高斯过程的一个样本。用符号的语言描述就是 $GP(m(t), k(s,t))$。其中

$$m_t = \mathbb{E}[\xi_t] = mean\ function \tag{1}$$

$$k(s,t) = covariance function = \mathbb{E}\left[[\xi_s - m(s)][\xi_t - m(t)]^T\right] \tag{2}$$

也就是说，一个高斯过程属于一个多维高斯分布，服从分布的形式为 $\mathcal{N}(m_t, k(s,t))$。

# Gaussian Process 02 Weight Space

Chen Gong

14 December 2019

Gaussian Process 在这里我们主要讲解的是 Gaussian Process Regression。我们从 Bayesian Linear Regression 的角度来引出的 Gaussian Progress Regression。

## 1 Recall Bayesian Linear Regression

首先，我们需要回顾一下 Bayesian Linear Regression。

1. 首先对于一个，参数符合的分布，$p(w|Data) = \mathcal{N}(w|\mu_w, \Sigma_w)$。其中，$\mu_w = \sigma^{-2}A^{-1}X^TY$，$\Sigma_w = A^{-1}$，其中，$A = \sigma^{-2}X^TX + \Sigma_p^{-1}$。从这一步我们就成功的得到了在已知 Data 的情况下，未知参数的分布形式。

2. 在给定一个新的未知数向量 $X^*$ 的情况下，我们可以首先利用 noise-free 形式：$f(x) = w^Tx = x^Tw$，然后再求得 noise 形式：$y = f(x) + \epsilon$，而 $\epsilon \sim \mathcal{N}(0, \sigma^2)$。来获得我们想要的 prediction 值。这样，我们就可以得到：

$$p(f(x^*)|Data, x^*) \sim \mathcal{N}(x^{*T}\mu_w, x^{*T}\Sigma_w x^*) \tag{1}$$

$$p(y^*|Data, x^*) \sim \mathcal{N}(x^{*T}\mu_w, x^{*T}\Sigma_w x^* + \sigma^2) \tag{2}$$

但是，问题马上就来了，因为很多时候，我们不能仅仅使用线性分类的方法来解决问题。现实生活中有许多非线性的问题来待我们求解。而一种经常使用的方法，也就是将数据投影到高维空间中来将非线性问题，转换成一个高维空间中的线性可分问题。或者是使用 Bayesian Logistics Regression 来进行分类。如果，是将数据投影到高维空间中的话，我们很自然的就想到了 Kernel Bayesian Linear Regression。

那么这个非线性转换可以被我们写成：If $\phi : x \mapsto z$，$x \in \mathbb{R}^p$，$x \in \mathbb{R}^p$，$z \in \mathbb{R}^q$，$z = \phi(x)$。

## 2 非线性转换后的表达

数据集被我们描述为：$X = (x_1, x_2, \cdots, x_N)^T$，$Y = (y_1, y_2, \cdots, y_N)^T$。根据之前我们得到的 Bayesian Linear Regression 结果，我们代入可以得到：

$$p(f(x^*)|X, Y, x^*) \sim \mathcal{N}(x^{*T}(\sigma^2 A^{-1}X^TY), x^{*T}A^{-1}x^*) \tag{3}$$

而其中，$A = \sigma^{-2}X^TX + \Sigma_p^{-1}$，If $\phi : x \mapsto z$，$x \in \mathbb{R}^p$，$x \in \mathbb{R}^p$，$z \in \mathbb{R}^q$，$z = \phi(x)$(q>p)。这里的 $\phi$ 是一个非线性转换。我们定义：$\Phi = (\phi(x_1), \phi(x_2), \cdots, \phi(x_N))^T_{N\times q}$。

转换之后为：$f(x) = \phi(x)^T w$。那么，

$$p(f(x^*)|X, Y, x^*) \sim \mathcal{N}(\sigma^{-2}\phi(x^*)^T(A^{-1}\Phi(X)^T Y), \phi(x^*)^T A^{-1}\phi(x^*)) \tag{4}$$

而其中，$A = \sigma^{-2}\Phi(X)^T\Phi(X) + \Sigma_p^{-1}$。但是，很快我们又将面临一个新的问题，也就是 $A^{-1}$ 应该如何计算呢？这里我们需要使用到一个公式为，Woodbury Formula 公式：

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1} \tag{5}$$

所以，

$$\begin{aligned} A &= \sigma^{-2}\Phi(X)^T\Phi(X) + \Sigma_p^{-1} \\ A\Sigma_p &= \sigma^{-2}\Phi(X)^T\Phi(X)\Sigma_p + I \\ A\Sigma_p\Phi(X)^T &= \sigma^{-2}\Phi(X)^T\Phi(X)\Sigma_p\Phi(X)^T + \Phi(X)^T = \sigma^{-2}\Phi(X)^T(K + \sigma^2 I) \\ \sigma^{-2}A^{-1}\Phi(X)^T &= \Sigma_p\Phi(X)^T(K + \sigma^2 I)^{-1} \end{aligned} \tag{6}$$

然后，两边同乘一个 $\phi(x^*)$ 和 $Y$ 就可以得到：

$$\sigma^{-2}\phi(x^*)A^{-1}\Phi(X)^T Y = \phi(x^*)\Sigma_p\Phi(X)^T(K + \sigma^2 I)^{-1}Y \tag{7}$$

而这个 $\sigma^{-2}\phi(x^*)A^{-1}\Phi(X)^T Y$ 正好就是 $p(f(x^*)|X, Y, x^*)$'s Expectation。而这里的 $\Sigma_p = p(w)$ 是一个先验 $\sim \mathcal{N}(0, \Sigma_p)$，而 $\sigma^2$ 为先验分布的噪声，$X^*$ 是一个 new input，而$K = \Phi\Sigma_p\Phi^T$。所以，使用类似的方法我们可以得到，$p(f(x^*)|X, Y, x^*)$'s Covarience 为：$\phi(x^*)^T\Sigma_p\phi(x^*) - \phi(x^*)^T\Sigma_p\Phi(X)^T(K + \sigma^2 I)^{-1}\Phi(X)\Sigma_p\phi(x^*)$。所以：

$$p(f(x^*)|X, Y, x^*) \sim \mathcal{N}(\phi(x^*)\Sigma_p\Phi(X)^T(K+\sigma^2 I)^{-1}Y, \phi(x^*)^T\Sigma_p\phi(x^*) - \phi(x^*)^T\Sigma_p\Phi(X)^T(K+\sigma^2 I)^{-1}\Phi(X)\Sigma_p\phi(x^*)) \tag{8}$$

而大家注意观察一下，下面几个等式：

$$\phi(x^*)^T\Sigma_p\Phi^T \qquad \phi(x^*)^T\Sigma_p\phi(x^*) \qquad \Phi\Sigma_p\Phi^T \qquad \Phi\Sigma_p\phi(x^*) \tag{9}$$

我们再来谢谢这里的这个 $\Phi$ 是个什么东西？

$$\Phi_{N\times q} = (\phi(x_1), \phi(x_2), \cdots, \phi(x_N))^T_{N\times q} \tag{10}$$

所以大家想一想就知道了，公式 (9) 中的四个公式实际上是一个东西，而 $\Phi(X)$ 只不过是将多个向量拼接在了一起而已。而 $K(x, x') = \phi(x)^T\Sigma_p\phi(x')$，$x, x'$ 是两个不一样的样本，矩阵展开以后，形式都是一样的。那么下一个问题就是 $K(x, x')$ 是否可以表达为一个 Kernel Function 的形式？那么，相关的探究就变得有趣了。

## 3 Kernel Trick

因为 $\Sigma_p$ 是一个 positive define matrix，并且它也是 symmetry 的。所以，令 $\Sigma_p = (\Sigma_p^{\frac{1}{2}})^2$。那么，我们可以做如下的推导：

$$\begin{aligned} K(x, x') =& \phi(x)^T\Sigma_p^{\frac{1}{2}}\Sigma_p^{\frac{1}{2}}\phi(x') \\ =& (\Sigma_p^{\frac{1}{2}}\phi(x))^T \cdot \Sigma_p^{\frac{1}{2}}\phi(x) \\ =& < \varphi(x), \varphi(x') > \end{aligned} \tag{11}$$

其中，$\varphi(x) = \Sigma_p^{\frac{1}{2}}\phi(x)$。那么，我们利用 Kernel Trick 可以有效的避免求 $\phi(X)$，而是直接通过 $K(x, x')$ 中包含的高维空间的转化。而Bayesian Linear Regression + Kernel Trick 中就蕴含了一个 Non-Linear Transformation inner product。我们就可以将这个转换定义到一个核空间中，避免了直接来求这个复杂的转化。这也就是 Kernel Trick。

看到了这里，大家很容易会产生一个疑惑，那就是，好像这里的 GPR 并没有和 GP 有一毛钱的关系。而实际上这里的 GPR 有两种不同的思考角度，也就是两种 View，而这两种 View 可以得到 equal result：

1. Weight-Space view，也就是我们这一小节所讲的东西。指的就是那两个等式，$f(x) = \phi(x)^T w$ 和 $y = f(x) + \epsilon$。在这里我们的研究对象就是 $w$，假设 $w$ 的先验，需要求得 $w$ 的后验，所以是从 Weight-Space 的角度分析的。

2. Function-Space view，我们将 $f(x)$ 看成是一个随机变量，$f(x) \sim GP(m(x), K(x, x'))$。这个我们会在后面的小节中进行详细的描述，大家就可以看到 GP 的思想在其中的运用了。

而有一句话对 GPR 的总结，非常的有意思，Gaussian Process Regress is the extension of Bayesian Linear Regression with kernel trick. 仔细想一想就知道了，我们把逻辑思路理一下，我们想用贝叶斯练习回归来解决非线性的问题，所以我们需要把输入空间投射到一个高维空间中，低维空间中的线性不可分问题将可以转化为高维空间中的线性可分问题。那么，我们就需要一个转换函数来完成这个工作，但是这个转换函数怎么求？有可能会很难求，而且维度很高。那么，我们就不求了，直接使用核技巧，也就是两个向量的内积等于一个核函数的值就可以了。这大概就是本节中 Weight-Space View 的一个主线的思路。

# Gaussian Process 03 Function View

Chen Gong

15 December 2019

在上一小节中，我们从 Weight-Space View 来看 Gaussian Process Regression，好像和 Gaussian Process 并没有什么关系。但是这一小节，我们从函数的角度来看就可以看到了。

## 1 Recall Gaussian Process

对于一组随机变量 $\{\xi_t\}_{t\in T}, T:$ continuous space or time。If:$\forall n \in N^+ \ (n \geq 1)$, Index:$\{t_1, t_2, \cdots, t_n\} \longrightarrow$ random variable: $\{\xi_{t_1}, \xi_{t_2}, \cdots, \xi_{t_n}\}$。令 $\xi_{1:n} = \{\xi_{t_1}, \xi_{t_2}, \cdots, \xi_{t_n}\}^T$。If $\xi_{1:n} \sim \mathcal{N}(\mu_{1:n}, \Sigma_{1:n})$，那么我们称 $\{\xi_t\}_{t\in T}$ is a Gaussian Distribution。并且，$\xi_t \sim GP(m(t), k(t,s))$，$m(t)$ 为 mean function，$k(t,s)$ 为 covariance function。下面我们回到 Weight-Space View 中。

## 2 Weight-Space view to Function-Space view

在这里 $w$ 是一个先验分布，$f(x)$ 是一个随机变量。$f(x) = \phi(x)^T w$，$y = f(x) + \epsilon$，$\epsilon \sim \mathcal{N}(0, \sigma^2)$。在 Bayesian 的方法中，对于给定的先验信息 (prior)：$w \sim \mathcal{N}(0, \Sigma_p)$。因为，$f(x) = \phi(x)^T w$，所以可以得到：

$$\mathbb{E}_w[f(x)] = \mathbb{E}_w[\phi(x)^T w] = \phi(x)^T \mathbb{E}_w[w] = 0 \tag{1}$$

那么对于 $\forall x, x' \in \mathbb{R}^p$，

$$\begin{aligned} cov(f(x), f(x')) =& \mathbb{E}[(f(x) - \mathbb{E}[f(x)])(f(x') - \mathbb{E}[f(x')])] \\ =& \mathbb{E}[f(x)f(x')] \\ =& \mathbb{E}[\phi(x)^T w \phi(x')^T w] \\ =& \mathbb{E}[\phi(x)^T w w^T \phi(x')] \end{aligned} \tag{2}$$

因为 $\phi(x')^T w$ 的结果是一个实数，所以它的转置就等于它自己。又因为 $w \sim \mathcal{N}(0, \Sigma_p)$，均值为 0，协方差为 $\Sigma_p$。并且有 $\mathbb{E}[ww^T] = \mathbb{E}[(w-0)(w^T-0)]$，这个东西不就是协方差矩阵 $cov(w) = \Sigma_p$。

而 $\phi(x)^T \Sigma_p \phi(x')$ 是一个 kernel function，前面我们已经证明过了，$\varphi(x) = \Sigma_p^{\frac{1}{2}}$。而 $\phi(x)\Sigma_p\phi(x') =< \varphi(x), \varphi(x') >= K(x, x')$。

推导进行到了这里，我们就知道了 $f(x)$ 的期望为 0，协方差矩阵为一个核函数 $K(x, x')$。那么我们是不是惊奇的发现，这个和我们高斯过程的定义：$\xi_t \sim GP(m(t), K(t,s))$，是多么惊人的相似呀。所以，这里可以启发我们：$f(x)$ 的组成是否可以看成一个 GP，而 $\{f(x)\}_{x\in\mathbb{R}^p}$。那么，首先 $f(x)$ 是一个

function，而且 $f(x)$ 还是一个服从高斯分布的随机变量，$m(t)$ 是一个 mean function，$K(t,s)$ 是一个 covariance function。为了加深大家的理解，我们做进一步清晰的对比：

$$\begin{cases} t \longrightarrow \xi_t,\ \{\xi_t\}_{t\in T} \sim GP \\ x \longrightarrow f(x),\ \{f(x)\}_{x\in\mathbb{R}^p} \sim GP \end{cases} \tag{3}$$

其实，我这样一对比，就非常的清晰了。在 GPR 的算法中，

1. Weight-Space view 中关注的是 $w$，即为：

$$x^* \longrightarrow y^* \quad p(y^*|Data, x^*) = \int p(y^*|Data, x^*, w)p(w)dw \tag{4}$$

又因为 $w$ 本身就是从 Data 中，推导得到的，所以 $p(y^*|Data, x^*, w) = p(y^*|x^*, w)$。

2. Function-Space view 中关注的是 $f(x)$，即为：

$$p(y^*|Data, x^*) = \int p(y^*|f(x), x^*)p(f(x))df(x) \tag{5}$$

写到了这里，不知道大家有没有一定感觉了，这里就是把 $f(x)$ 当成了一个随机变量来看的。这里也就是通过 $f(x)$ 来直接推导出 $y^*$。在 Weight-Space View 中，我们没有明确的提到 GP，但是在 Weight-Space view 中，$f(x)$ 是符合 GP 的，只不过是没有显性的表示出来而已。我们可以用一个不是很恰当的例子来表述一个，Weight-Space view 就是两个情侣之间，什么都有了，孩子都有了，但是就是没有领结婚证，那么他们两个之间的关系就会比较复杂。而 Function-Space view 就是两个情侣之间先领结婚证，在有了孩子，按部就班的来进行，所以他们之间的关系就会比较简单。

# 3 Function-Space View

上一小节中，我们从 Weight-Space View 过渡到了 Function-Space View，而 Weight 指的就是参数。

$$\begin{aligned} &\{f(x)\}_{x\in\mathbb{R}^p} \sim GP(m(x), K(x,x')) \\ &m(x) = \mathbb{E}[f(x)] \quad K(x,x') = \mathbb{E}[(f(x)-m(x))(f(x')-m(x'))] \end{aligned} \tag{6}$$

Regression 问题被我们描述为：

Data：$\{(x_i, y_i)\}_{i=1}^N$，$x = (x_1, x_2, \cdots, x_N)^T_{N\times p}$，$Y = (y_1, y_2, \cdots, y_N)^T_{N\times 1}$。又因为 $f(x)$ 符合一个 GP，所以，$f(x) \sim \mathcal{N}(\mu(x), K(x,x'))$。且 $Y = f(x) + \epsilon$，所以 $Y \sim \mathcal{N}(\mu(x), K(x,x') + \sigma^2 I)$。那么，给定 new input：$X^* = (x_1^*, x_2^*, \cdots, x_N^*)$，我们想要的 Prediction output 为 $Y^* = f(x^*) + \epsilon$。那么，我们可以得到 $Y$ 和 $f(x^*)$ 的联合概率密度分布为：

$$\begin{bmatrix} Y \\ f(x^*) \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} \mu(x) \\ \mu(x^*) \end{bmatrix}, \begin{bmatrix} K(X,X) + \sigma^2 I & K(X,X^*) \\ K(X^*,X) & K(X^*,X^*) \end{bmatrix} \right) \tag{7}$$

在这里，我必须要首先列举一下，前面我们曾经提到的更加联合概率密度求边缘概率密度的方法。已知，$X \sim \mathcal{N}(\mu, \Sigma)$，

$$x = \begin{bmatrix} x_a \\ x_b \end{bmatrix} \qquad \mu = \begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix} \qquad \Sigma = \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix} \tag{8}$$

而我们可以得到：

$$
\begin{aligned}
&p(x_b|x_a) \sim \mathcal{N}(\mu_{b|a}, \Sigma_{b|a}) \\
&\mu_{b|a} = \Sigma_{ba}\Sigma_{aa}^{-1}(x_a - \mu_a) + \mu_b \\
&\Sigma_{b|a} = \Sigma_{bb} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ab}
\end{aligned} \tag{9}
$$

我们要求的概率为 $p(f(x^*)|Y, X, x^*)$，就是一个我们要求的条件概率，有的同学可能就会有疑惑了，不应该是 $p(f(x*)|Y)$ 吗？为什么这里可以把 $X, x^*$ 给忽略掉了？因为 $X$ 和 $Y$ 相关，因为 $Y = \phi(X)^T w + \epsilon$。而 $X^*$ 涵盖在了 $f(x^*)$ 中，因为 $f(x^*) = \phi(X^*)^T w$。

所以，我们的目标也就是求 $p(f(x*)|Y$，也就是已知联合概率分布的情况下求条件概率分布。

我们对比公式 (8) 和公式 (9) 就可以发现，$Y \to x_a, f(x^*) \to x_b, K(X, X)+\sigma^2 I \to \Sigma_{aa}, K(X, X^*) \to \Sigma_{ba}, K(X*, X*) \to \Sigma_{bb}$。那么，我们可以令 $p(f(x^*)|Y, X, x^*) \sim \mathcal{N}(\mu^*, \Sigma^*)$，代入之前获得的公式的结果我们就可以得到：

$$
\begin{aligned}
\mu^* =& K(X^*, X)(K(X, X) + \sigma^2 I)^{-1}(Y - \mu(X)) + \mu(X^*) \\
\Sigma^* =& K(X^*, X^*) - K(X^*, X)(K(X, X) + \sigma^2 I)^{-1}
\end{aligned} \tag{10}
$$

并且，$y* = f(X^*) + \epsilon$。那么 noise-free 的形式可以被我们写完：$p(f(x^*)|Y, X, x^*) = \mathcal{N}(\mu^*, \Sigma^*)$。而 $p(f(y^*)|Y, X, x^*) = \mathcal{N}(\mu_y^*, \Sigma_y^*)$，而 $y^* = f(X^*) + \epsilon$，而 $\mu_y^* = \mu^*$，$\Sigma_y^* = \Sigma^* + \sigma^2 I$。

在 Function-Space View 中，$f(x)$ 本身是符合 GP 的，那么我们可以直接写出 *Prediction* 矩阵，并将其转化为已知联合概率密度分布求条件概率密度的问题。Function-Space View 和 Weight-Space View 得到的结果是一样的，但是更加的简单。