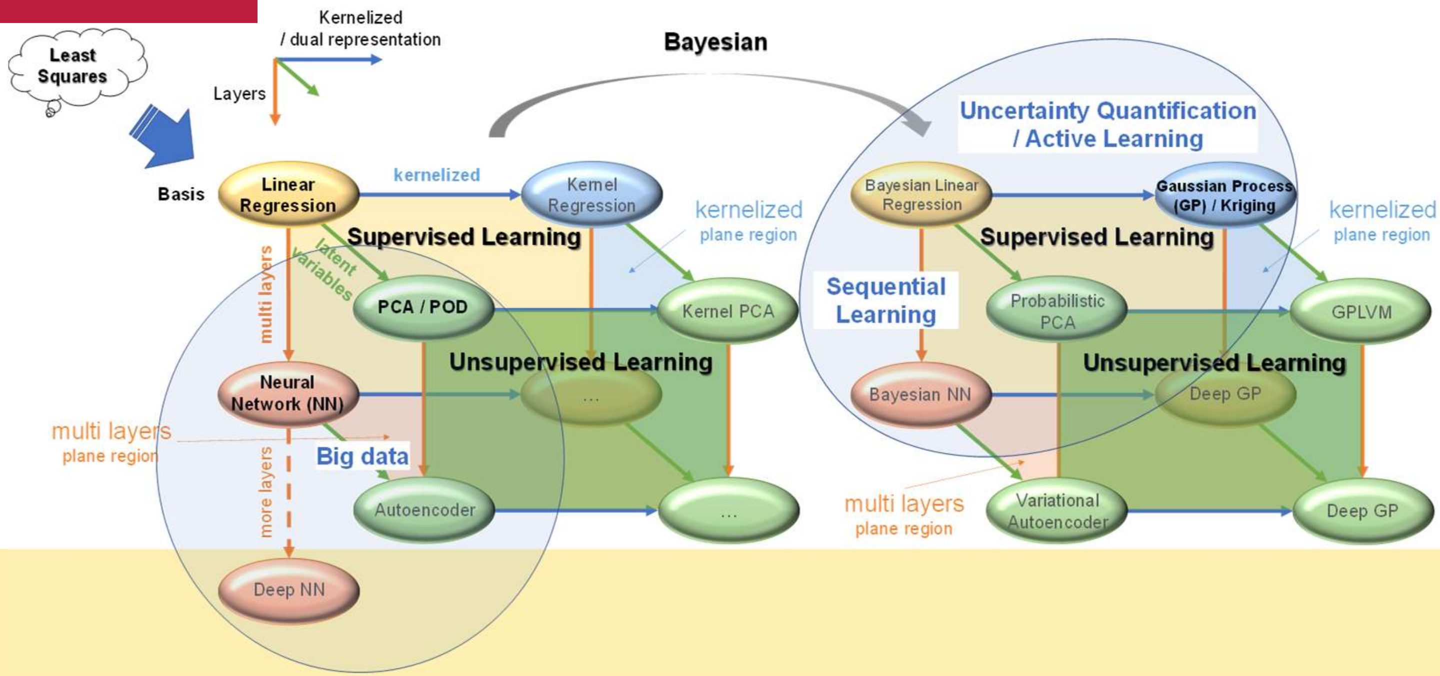


Scientific Machine Learning

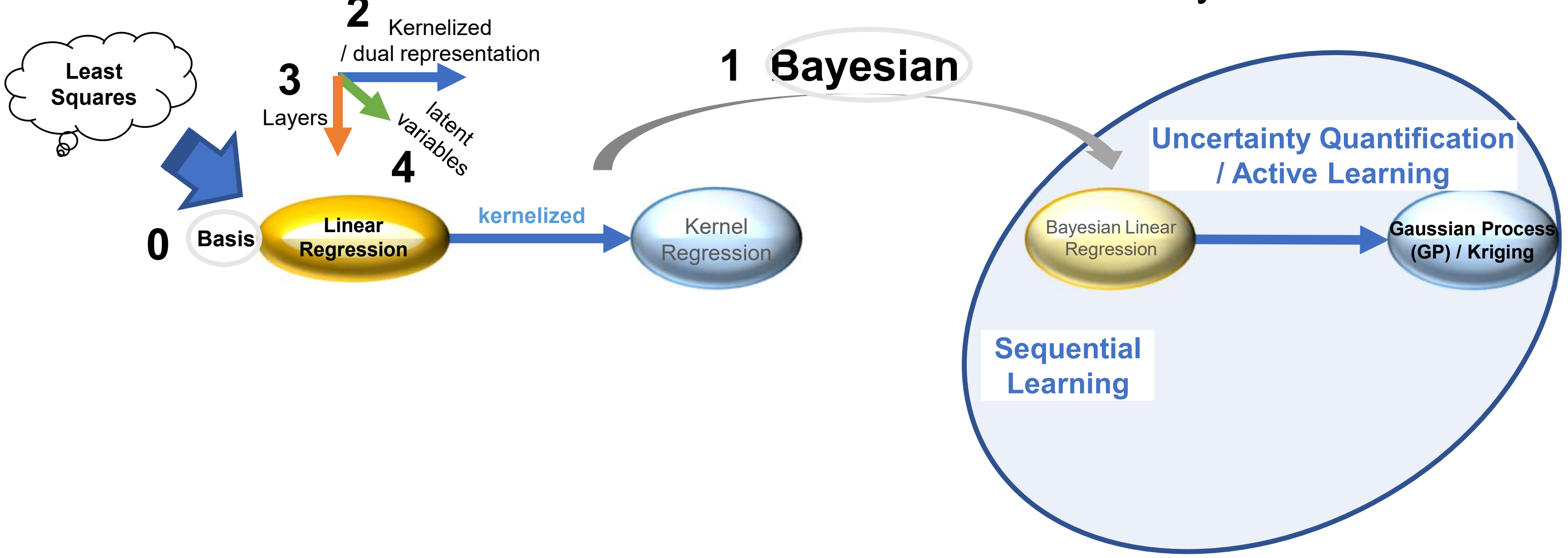
Lecture 7: Dual Representation (Kernel Methods)

Dr. Daigo Maruyama

Prof. Dr. Ali Elham

Key Components

the three axes + Bayesian axis = four axes



Introduction of Books (for Gaussian Processes)

Gaussian Process
(as perspectives from **machine learning**) – free pdf available officially

- C. E. Rasmussen and K. I. Williams, "Gaussian Processes for Machine Learning", the MIT Press, 2006.

Kriging
(as perspectives from **geostatistics**) – no free pdf available

- Alexander Forrester, Andras Sobester, Andy Keane, "Engineering Design via Surrogate Modelling: A Practical Guide", Wiley, 2008.
- Hans Wackernagel, ''Multivariate Geostatistics', Springer, 1995, 1998.

Please compare the description in these books from these two fields

Lecture content

- Review of Bayesian approach
- Introduction of kernel
- Dual representation (Introduction to Gaussian processes)

The lecture of this time partially follows the Chapter 6 and Section 2.3.3 and 3.3 of the book:
Christopher M. Bishop "Pattern Recognition And Machine Learning" Springer-Verlag (2006)
The name of this book is shown as “PRML” when it is referred in the slides.

The lecture slides contains many original contents in the context apart from the above sections in the book.

Lecture content

1. Bayesian approach (review)

Bayesian Approach

- Principally no overfitting
  - not because of prior information
  - because of considering all the possibilities of the parameter $\boldsymbol{w}$

→ $\boldsymbol{w}$ is not in the prediction anymore in the Bayesian approach

- **Deterministic (point estimate)**

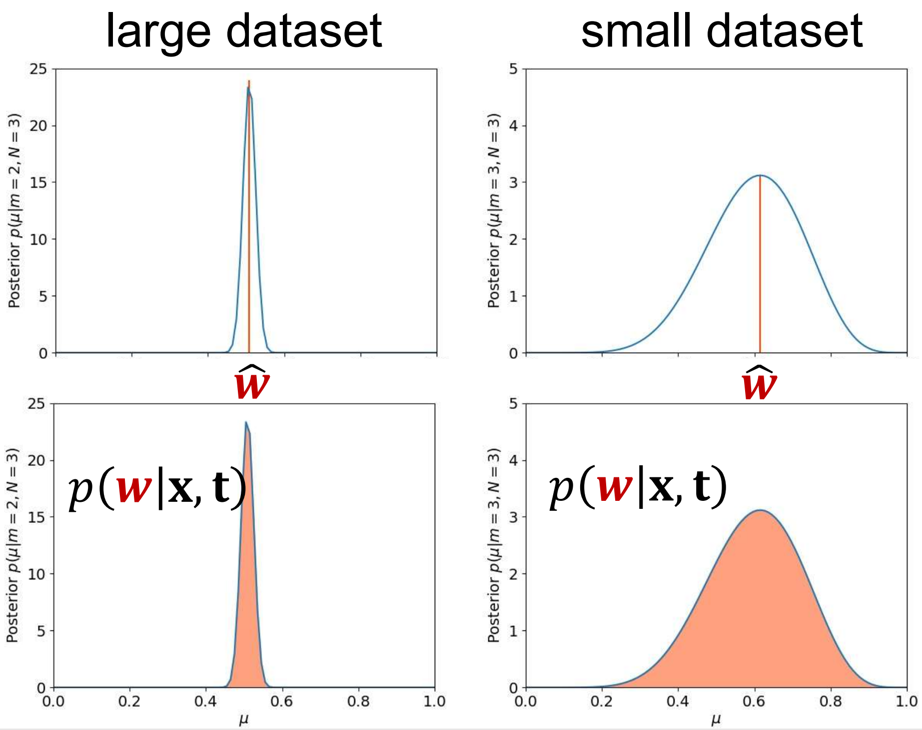
$p(t|x, \hat{\boldsymbol{w}})$

We believe that there is a true $\boldsymbol{w}$ (but learned by finite data in hand).

- **Stochastic (Bayesian)**

$$p(t|x, \mathbf{x}, \mathbf{t}) = \int p(t|x, \boldsymbol{w})p(\boldsymbol{w}|\mathbf{x}, \mathbf{t})\mathrm{d}\boldsymbol{w}$$

$\boldsymbol{w}$ is integrated out <u>by reflecting all the info of $\boldsymbol{w}$</u>.



Lecture content

2. Kernel approach

Feature Vector to Kernel Function

Design kernel function k itself

input → Feature Vector → Kernel Function

$\boldsymbol{x}$ $\qquad$ $\boldsymbol{\phi}(\boldsymbol{x})$ $\qquad$ $k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}')$

- Nonlinear function
- String
- Graph
- etc.

Linear regression

dot product
= similarity between $\boldsymbol{x}$ and $\boldsymbol{x}'$

Input Vector

$y(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$ Linear regression (**simple**)

point estimate (MLE)

$$\widehat{\boldsymbol{w}} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{t}$$

$$\mathbf{X} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)^{\mathrm{T}} = \begin{pmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_N \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1D} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{ND} \end{pmatrix}$$

Please see Lecture 4

Prediction by “a line” (or hyperplane in general)

Feature Vector

$y(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x})$ Linear regression (**general**)

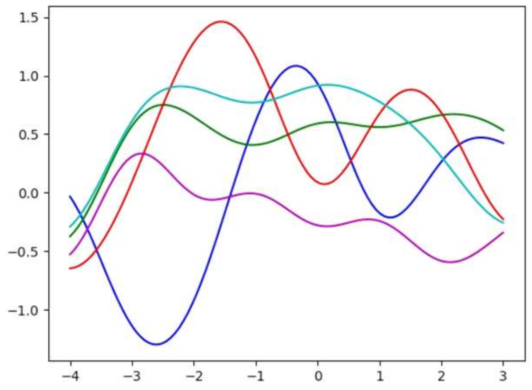
point estimate (MLE)

$$\widehat{\boldsymbol{w}} = \left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{t}$$

$$\boldsymbol{\Phi} = \begin{pmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1) \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}_N) \end{pmatrix} = \begin{pmatrix} \phi_1(\boldsymbol{x}_1) & \cdots & \phi_M(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_1(\boldsymbol{x}_N) & \cdots & \phi_M(\boldsymbol{x}_N) \end{pmatrix}$$

Please see Lecture 4

Prediction by “a curve” (curve fitting problem)



Kernel Function

$$y(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{k}(\boldsymbol{x}) = \sum_{n=1}^{N} w_n k(\boldsymbol{x}, \boldsymbol{x}_n)$$

point estimate (MLE)

$$\hat{\boldsymbol{w}} = \left(\mathbf{K}^{\mathrm{T}}\mathbf{K}\right)^{-1} \mathbf{K}^{\mathrm{T}} \mathbf{t}$$
$$= \boldsymbol{K}^{-1} \mathbf{t}$$

Linear regression

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

$\sigma = 1$

$$\mathbf{K} = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix}$$

Kernel regression

RBF (radial basis function) interpolation

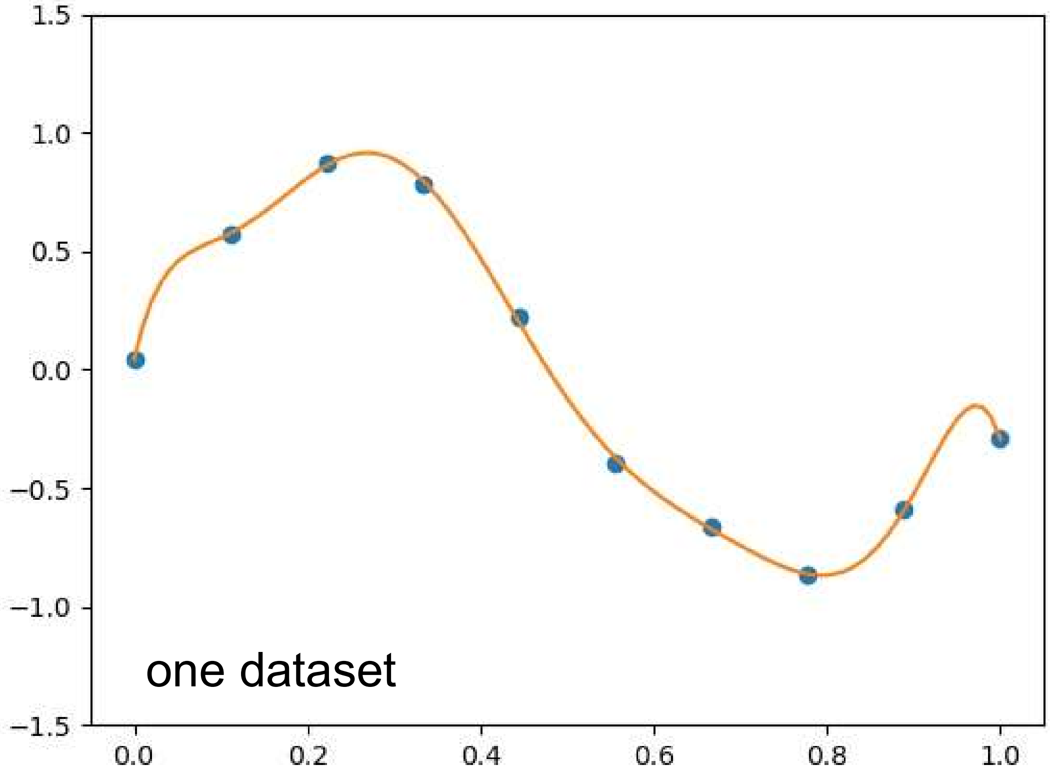
The dimensionality (degree of freedom) of $\boldsymbol{w}$ is always adjusted to the sample size N.

Guaranteed to pass through all the sample points.

Kernel Function

Kernel regression: examples

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

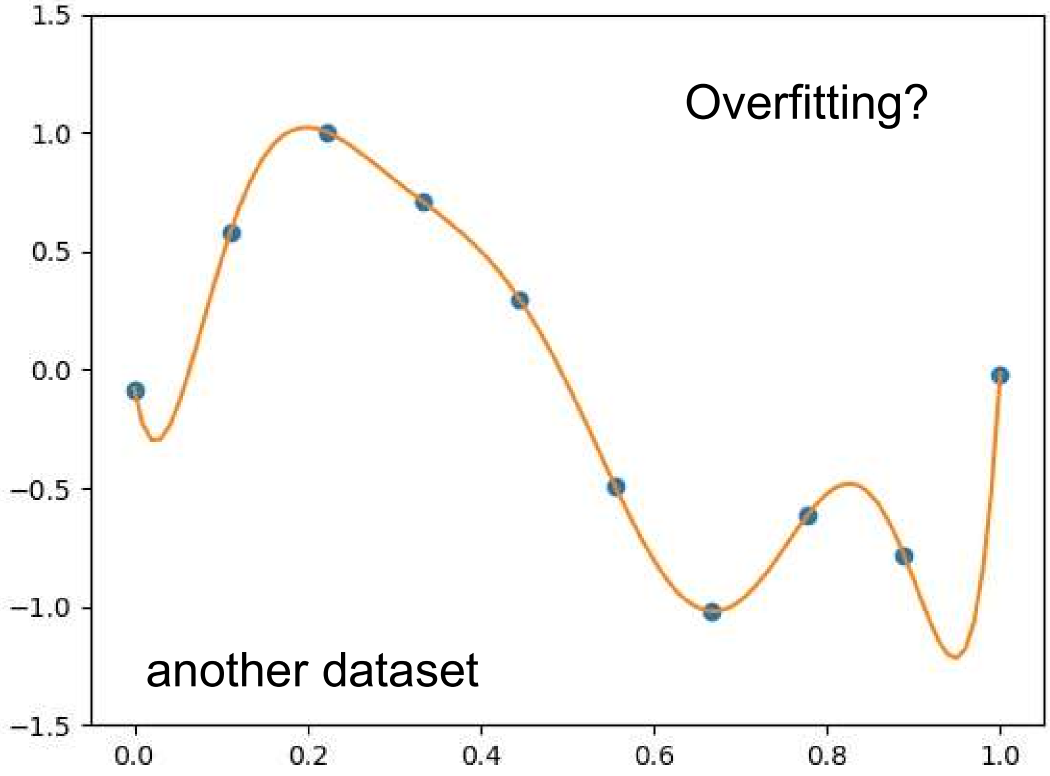


$\sigma = 1$

Kernel Function

Kernel regression: examples

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right)$$



$\sigma = 1$

$\sigma = 1$ and it cannot be tuned by MLE.

<u>dual representation</u> can solve this.

Design of Kernel Function

Necessary and sufficient conditions for a function $k(\boldsymbol{x}, \boldsymbol{x}')$

Gram matrix $\mathbf{K}$ to be positive (semi)definite

the same property as the covariance matrix $\boldsymbol{\Sigma}$ of multiple Gaussian distributions $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

$$\mathbf{Y} = \big(\boldsymbol{y}(\boldsymbol{x}_1), \cdots, \boldsymbol{y}(\boldsymbol{x}_N)\big)^{\mathrm{T}}$$

$$\mathbf{Y}_C = \mathbf{Y} - E[\mathbf{Y}] \qquad \text{centered}$$

Definition of covariance

$$cov[\boldsymbol{y}, \boldsymbol{y}'] = E\big[(\boldsymbol{y} - E[\boldsymbol{y}])(\boldsymbol{y}' - E[\boldsymbol{y}'])^{\mathrm{T}}\big]$$

$$\boldsymbol{\Sigma}[\mathbf{Y}, \mathbf{Y}] = E\big[(\mathbf{Y} - E[\mathbf{Y}])(\mathbf{Y} - E[\mathbf{Y}])^{\mathrm{T}}\big]$$

$$\boldsymbol{\Sigma} = \frac{1}{N}\boxed{\mathbf{Y}_C {\mathbf{Y}_C}^{\mathrm{T}}} \quad \text{dot product}$$

$$= \begin{pmatrix} cov[\boldsymbol{y}(\boldsymbol{x}_1), \boldsymbol{y}(\boldsymbol{x}_1)] & \cdots & cov[\boldsymbol{y}(\boldsymbol{x}_1), \boldsymbol{y}(\boldsymbol{x}_N)] \\ \vdots & \ddots & \vdots \\ cov[\boldsymbol{y}(\boldsymbol{x}_N), \boldsymbol{y}(\boldsymbol{x}_1)] & \cdots & cov[\boldsymbol{y}(\boldsymbol{x}_N), \boldsymbol{y}(\boldsymbol{x}_N)] \end{pmatrix} = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix}$$

Arbitrary covariance matrix $\boldsymbol{\Sigma}$ can be a Gram matrix $\mathbf{K}$.

Design of Kernel Function

Kernel trick

Gram matrix $\mathbf{K}$ to be positive (semi)definite

As far as this is satisfied, any kernel functions $k(\boldsymbol{x}, \boldsymbol{x}')$ can be designed.

$$\mathbf{K} = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix}$$

Design of Kernel Function

An example of kernel trick

$$k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

Gaussian kernel

We did not have to explicitly specify what the feacture vector $\boldsymbol{\phi}(\boldsymbol{x})$ is.

$\boldsymbol{\phi}(\boldsymbol{x})$ in this case is originally a nonlinear function of <u>infinity dimension</u>.

An exercise to get familiar with kernel functions

see PRML, 294-295

$$k(\boldsymbol{x}, \boldsymbol{z}) = \left(\boldsymbol{x}^{\mathrm{T}} \boldsymbol{z}\right)^2$$

What is $\boldsymbol{\phi}(\boldsymbol{x})$?

$$\begin{aligned} k(\boldsymbol{x}, \boldsymbol{z}) &= \left(\boldsymbol{x}^{\mathrm{T}} \boldsymbol{z}\right)^2 \\ &= \cdots \\ &= \left({x_1}^2, \sqrt{2} x_1 x_2, {x_2}^2\right)\left({z_1}^2, \sqrt{2} z_1 z_2, {z_2}^2\right)^{\mathrm{T}} \end{aligned}$$

Design of Kernel Function

Once valid kernels are created:

Sum of kernel = kernel
Product of kernel = kernel
…

$$k(\boldsymbol{x}, \boldsymbol{x}') = ck_1(\boldsymbol{x}, \boldsymbol{x}')$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}')$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}')k_2(\boldsymbol{x}, \boldsymbol{x}')$$

…

Practical design of kernel functions: Existing kernel functions which are already guaranteed that the covariance matrix composed by them are positive (semi)definite.

Read books and papers, then free to compose new kernels based on them.

This can be naturally understood by considering some covariance matrices:

$$\mathbf{K} = \begin{pmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix} + \hat{\sigma}^2 \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \begin{cases} 1 \; if \; \boldsymbol{x} = \boldsymbol{x}' \\ 0 \; else \end{cases}$$

Lecture content

3. Kernels from dual representation

- A bridge between the Bayesian linear regression and Gaussian processes
- Introduction to Gaussian processes in the next lectures

Bayesian Linear Regression (REVIEW)

Let's try to apply this concept to **the curve fitting problem**.

Probabilistic model

$$p(t|x,\boldsymbol{w},\sigma) = \mathcal{N}(t|y(x,\boldsymbol{w}),\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{\{t-y(x,\boldsymbol{w})\}^2}{2\sigma^2}\right\}$$

Likelihood function

$$p(\mathbf{t}|\mathbf{x},\boldsymbol{w},\sigma) = \prod_{i=1}^{N}\mathcal{N}(t_i|y(x_i,\boldsymbol{w}),\sigma^2) = \prod_{i=1}^{N}\frac{1}{\sqrt{2\pi\sigma^2}}exp\left[-\frac{\{t_i-y(x_i,\boldsymbol{w})\}^2}{2\sigma^2}\right]$$

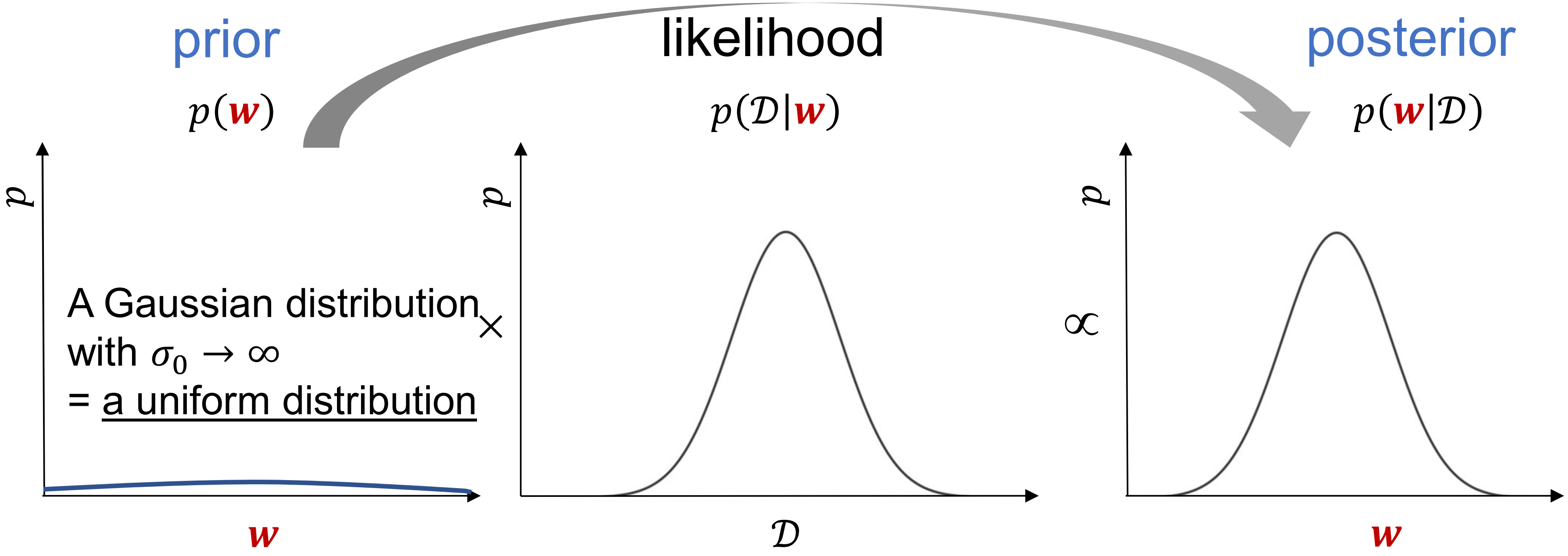
Posterior distribution $p(\boldsymbol{w},\sigma|\mathbf{x},\mathbf{t}) = \boldsymbol{complicated}$

This is a Gaussian distribution wrt t_i,
but NOT a Gaussian distribution wrt $\boldsymbol{w},\sigma$.

Predictive distribution

$$p(t|x,\mathbf{x},\mathbf{t}) = \iint \underbrace{p(t|x,\boldsymbol{w},\sigma)p(\boldsymbol{w},\sigma|\mathbf{x},\mathbf{t})}_{\textbf{probabilistic model} \times \textbf{posterior}}\mathrm{d}\boldsymbol{w}\mathrm{d}\sigma = \boldsymbol{complicated}$$

Conjugate Prior + Linear Regression (REVIEW)



$$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_0, \mathbf{S}_0)$$

$$\boldsymbol{m}_0 = \mathbf{0}$$

$$\mathbf{S}_0 = {\sigma_0}^2\mathbf{I}$$

$$p(\boldsymbol{w}|\mathcal{D}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_N, \mathbf{S}_N)$$

Conjugate prior
+ Linear regression

Analytical solutions of
- Posterior distribution
- Predictive distribution

Bayesian Linear Regression

Let's try to apply this concept to **the curve fitting problem**.

Probabilistic model

$$p(t|x,\boldsymbol{w}) = \mathcal{N}(t|y(x,\boldsymbol{w}),\hat{\sigma}^2) = \boldsymbol{Gaussian}$$

Likelihood function

$$p(\mathbf{t}|\mathbf{x},\boldsymbol{w}) = \boldsymbol{Gaussian}$$

If **Prior distribution** $p(\boldsymbol{w}) = \boldsymbol{Gaussian}$

$\Rightarrow$ **Posterior distribution** $p(\boldsymbol{w}|\mathbf{x},\mathbf{t}) = \boldsymbol{Gaussian}$

Predictive distribution

$\boldsymbol{w}$ is already integrated out.

$$p(t|x,\mathbf{x},\mathbf{t}) = \int p(t|x,\boldsymbol{w})p(\boldsymbol{w}|\mathbf{x},\mathbf{t})\mathrm{d}\boldsymbol{w} = \boldsymbol{Gaussian}$$

Directly design the **predictive distribution**

please read 2.3.3 in PRML to follow the formulations.
But the concept described here is important.

Bayesian Linear Regression

This has to be a linear regression model.

1. We have (defined):
   - A probabilistic model (of $\mathbf{t}$)

     $$p(\mathbf{t}|\boldsymbol{w}) = \mathcal{N}\left(\mathbf{t}\middle|\boxed{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Phi}}, \hat{\boldsymbol{\Sigma}}\right)$$

   - A prior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_0, \mathbf{S}_0)$$

a familiar probabilistic model (of t)

$$p(t|\boldsymbol{w}) = \mathcal{N}\left(t\middle|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}), \hat{\sigma}^2\right)$$

$$p(\mathbf{t}|\boldsymbol{w}) = \mathcal{N}\left(\begin{pmatrix} t_1 \\ t_2 \\ \vdots \end{pmatrix}\middle| \begin{pmatrix} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_1) \\ \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_2) \\ \vdots \end{pmatrix}, \begin{pmatrix} \hat{\sigma}^2 & 0 & \cdots \\ 0 & \hat{\sigma}^2 & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}\right)$$

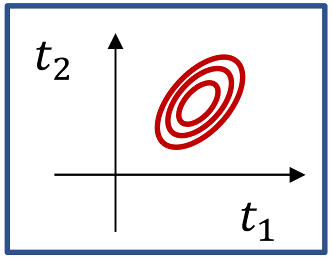
$$\hat{\boldsymbol{\Sigma}} = \hat{\sigma}^2\mathbf{I}$$

2. Then we obtain:
   - A posterior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}|\mathbf{t}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_N, \mathbf{S}_N)$$

   - A predictive distribution (of $\mathbf{t}$)

     $$p(\mathbf{t}) = \mathcal{N}\left(\mathbf{t}\middle|\boldsymbol{m}_N{}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\boldsymbol{\Sigma}} + \boldsymbol{\Phi}\mathbf{S}_N\boldsymbol{\Phi}^{\mathrm{T}}\right)$$

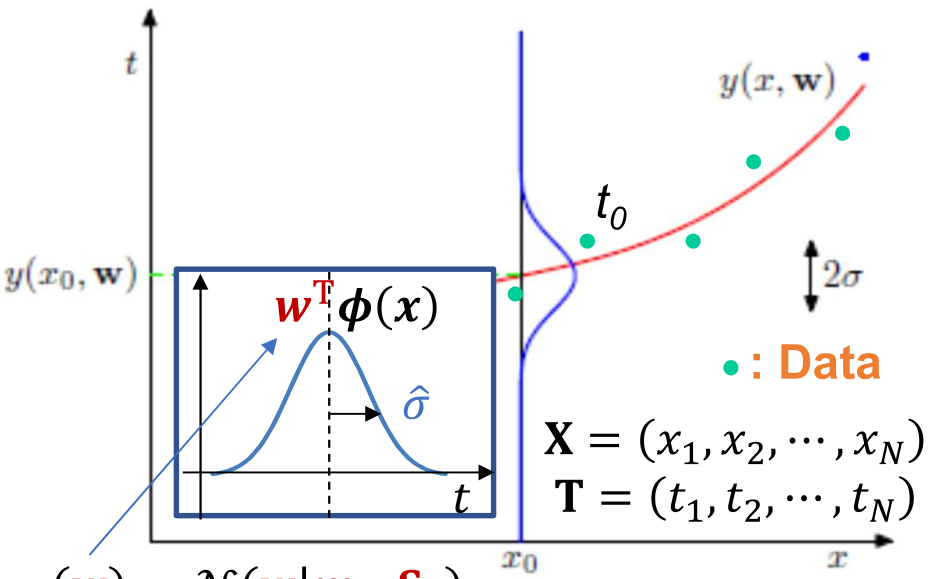


A multivariate Gaussian distribution in general

Dual Representation

Probabilistic model:
An isotropic Gaussian distribution

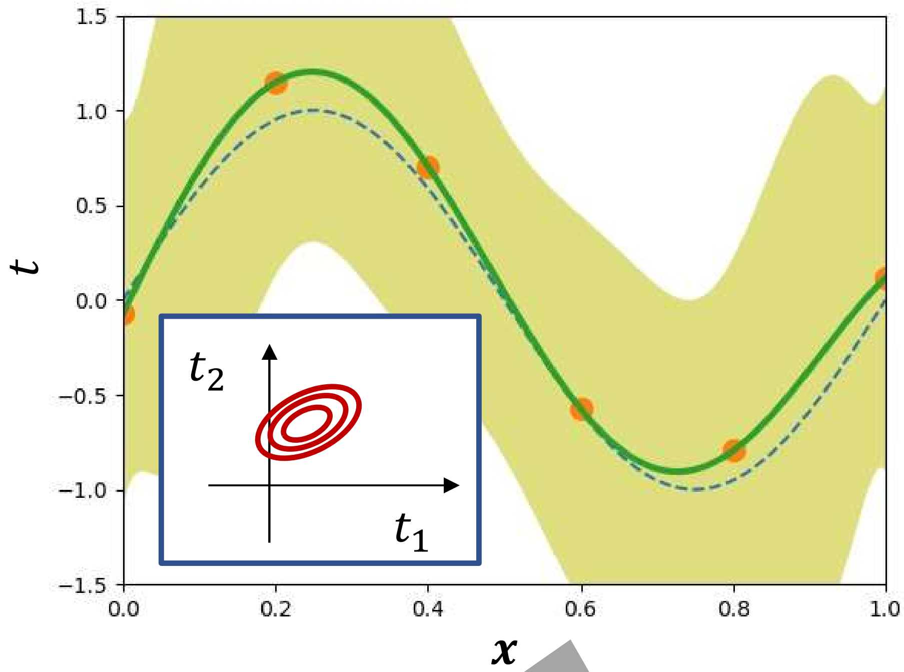
$$p(t|x,\boldsymbol{w}) = \mathcal{N}\left(t \middle| \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}), \hat{\sigma}^2\right)$$



$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_0, \mathbf{S}_0)$
Prior of $\boldsymbol{w}$

Predictive distribution:
A multivariate Gaussian distribution

$$p(\mathbf{t}|\mathbf{x},\mathbf{X},\mathbf{T}) = \mathcal{N}\left(\mathbf{t} \middle| \boldsymbol{m}_N{}^{\mathrm{T}}\mathbf{\Phi}, \hat{\sigma}^2\mathbf{I} + \mathbf{\Phi}\mathbf{S}_N\mathbf{\Phi}^{\mathrm{T}}\right)$$



Start (traditional)

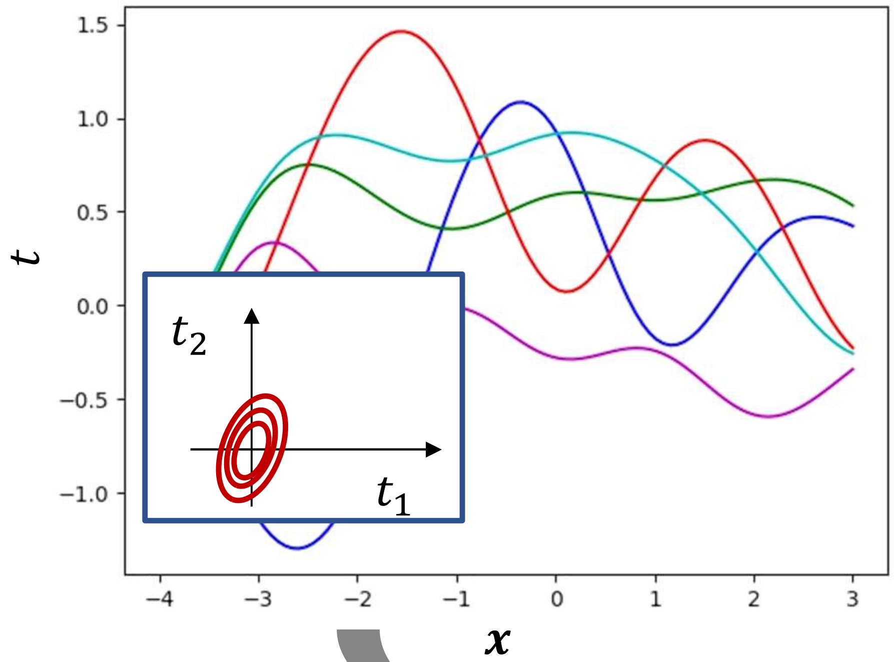
Arbitrary data: $\mathbf{X}, \mathbf{T}$

Goal (traditional)

Dual Representation

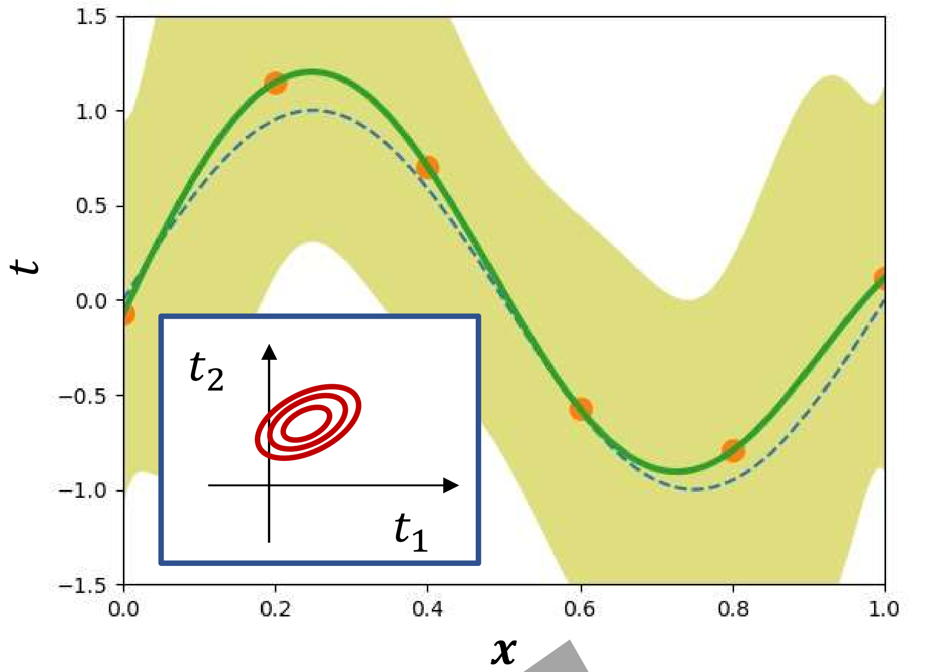
Predictive distribution:
A multivariate Gaussian distribution

$$p(\mathbf{t}|\mathbf{x}) = \mathcal{N}\left(\mathbf{t}\middle|\boldsymbol{m}_0{}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\sigma}^2\mathbf{I} + \boldsymbol{\Phi}\boldsymbol{S}_0\boldsymbol{\Phi}^{\mathrm{T}}\right)$$



Predictive distribution:
A multivariate Gaussian distribution

$$p(\mathbf{t}|\mathbf{x}, \mathbf{X}, \mathbf{T}) = \mathcal{N}\left(\mathbf{t}\middle|\boldsymbol{m}_N{}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\sigma}^2\mathbf{I} + \boldsymbol{\Phi}\boldsymbol{S}_N\boldsymbol{\Phi}^{\mathrm{T}}\right)$$



Start (Dual)

Arbitrary data: $\mathbf{X}, \mathbf{T}$

Goal (Dual)

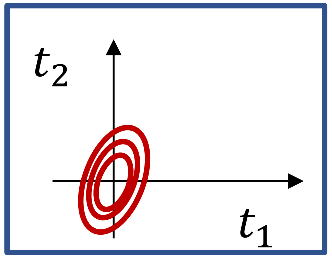
Dual Representation

Direct definition of a predictive distribution:
A multivariate Gaussian distribution
(represented by kernels)

1. We have (defined):
   - A predictive distribution (of $\mathbf{t}$)

     $$p(\mathbf{t}) = \mathcal{N}\left(\mathbf{t} \middle| \boldsymbol{m}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\boldsymbol{\Sigma}} + \boldsymbol{\Phi}\boldsymbol{S}\boldsymbol{\Phi}^{\mathrm{T}}\right)$$

     Since we know the form.



   - A prior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_0, \boldsymbol{S}_0)$$

     $$p(\mathbf{t}|\mathbf{x}) = \mathcal{N}\left(\mathbf{t} \middle| \overset{=\mathbf{0}}{\boldsymbol{m}_0{}^{\mathrm{T}}}\boldsymbol{\Phi}, \overbrace{\hat{\boldsymbol{\Sigma}} + \boldsymbol{\Phi}\boldsymbol{S}_0\boldsymbol{\Phi}^{\mathrm{T}}}^{\equiv \mathbf{K}}\right)$$

     Prior

Data $\mathbf{X}, \mathbf{T}$ is given.

2. Then we obtain:
   - A posterior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}|\mathbf{t}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_N, \boldsymbol{S}_N)$$

     $$p(\mathbf{t}|\mathbf{x}, \mathbf{X}, \mathbf{T}) = \mathcal{N}\left(\mathbf{t} \middle| \boldsymbol{m}_N{}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\boldsymbol{\Sigma}} + \boldsymbol{\Phi}\boldsymbol{S}_N\boldsymbol{\Phi}^{\mathrm{T}}\right)$$

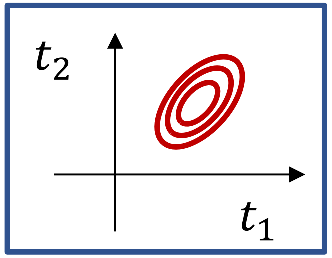
     Posterior

$\mathbf{x}, \mathbf{t}$: prediction

$\mathbf{X}, \mathbf{T}$: data

$$\boldsymbol{m}_N = \boldsymbol{m}_N(\mathbf{X}, \mathbf{T})$$

$$\boldsymbol{S}_N = \boldsymbol{S}_N(\mathbf{X})$$



Dual Representation

Direct definition of a predictive distribution:
A multivariate Gaussian distribution
(represented by kernels)

1. We have (defined):
   - A predictive distribution (of $\mathbf{t}$)

     $$p(\mathbf{t}) = \mathcal{N}\left(\mathbf{t}\middle|\boldsymbol{m}^{\mathrm{T}}\boldsymbol{\Phi}, \hat{\Sigma} + \boldsymbol{\Phi}\mathbf{S}\boldsymbol{\Phi}^{\mathrm{T}}\right)$$

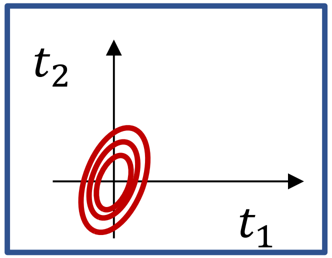
     Since we know the form.

   - A prior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_0, \mathbf{S}_0)$$

     $$\boxed{p(\mathbf{t}|\mathbf{x}) = \mathcal{N}(\mathbf{t}|\mathbf{0}, \mathbf{K}_0)}$$

Prior

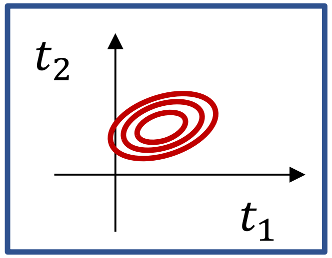


Data $\mathbf{X}, \mathbf{T}$ is given.

2. Then we obtain:
   - A posterior distribution (of $\boldsymbol{w}$)

     $$p(\boldsymbol{w}|\mathbf{t}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_N, \mathbf{S}_N)$$

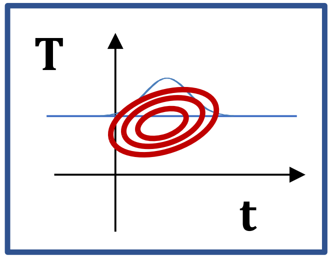
     $$\boxed{p(\mathbf{t}|\mathbf{x}) = \mathcal{N}(\mathbf{t}|\mathbf{0}, \mathbf{K}_N)}$$



$\mathbf{K}_N \neq \mathbf{K}_N(\mathbf{X}, \mathbf{T})$

Please remind you of **the conditional Gaussian distributions** in general introduced in Lecture 4.

Posterior



$$\boxed{p(\mathbf{t}|\mathbf{x}, \mathbf{X}, \mathbf{T}) = \mathcal{N}\left(\mathbf{t}\middle|\boldsymbol{k}_N(\mathbf{x})^{\mathrm{T}}\mathbf{K}_N{}^{-1}\boldsymbol{t}, \mathbf{K}_N(\mathbf{x}, \mathbf{x}) - \boldsymbol{k}_N(\mathbf{x})^{\mathrm{T}}\mathbf{K}_N{}^{-1}\boldsymbol{k}_N(\mathbf{x})\right)}$$