DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF COPENHAGEN

Linear Models For Regression, Part 2

Kim Steenstrup Pedersen

Plan for this lecture

- Presentation of the next assignment
- A snippet of theoretical foundation of regression
- Recap of linear models for regression
- Bayesian regression for linear models
- Bayesian sequential learning for regression
- Advanced topic: Full Bayesian approach by computing the predictive distribution by integration over all models.

Assignment 2: Basic learning algorithms

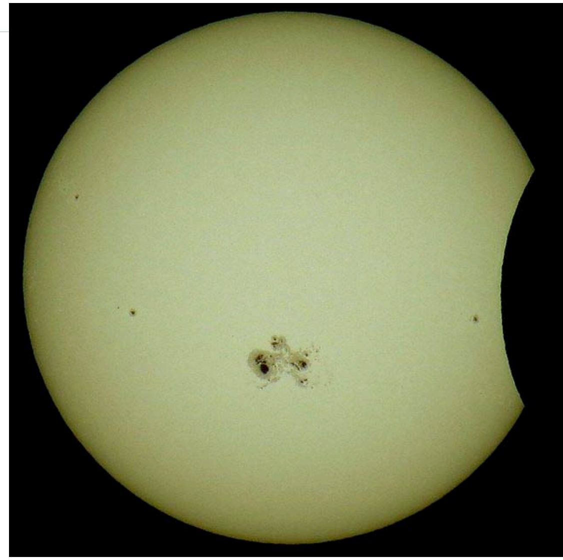
- Regression (this lecture):
  - Linear models for regression applied to a real data set in order to predict the number of sunspots from previous years sunspot numbers.
  - A theoretical question about weighted sum of squares

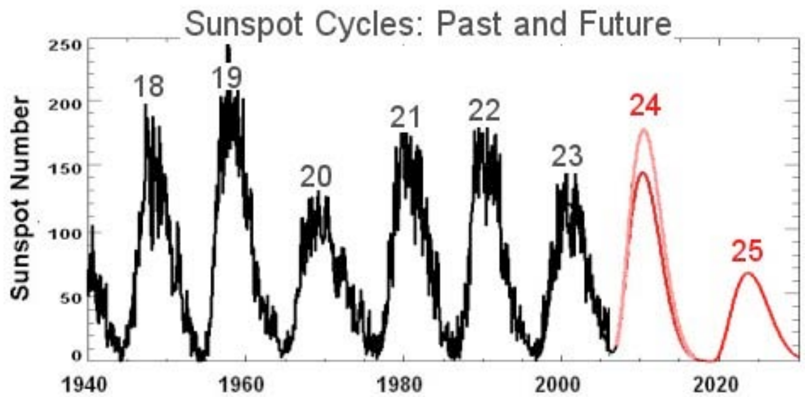
Example: Sunspots (Assignment 2)

- **Input variable:**
  - Number of sunspot in previous years
- **Target variable:**
  - Number of sunspots in following years
- **Your task:**
  - Learn a linear regression model

    $$t = \mathbf{y}(\mathbf{x})$$

    for predicting sunspot numbers
  - How to do this?
  - We learn today and Tuesday





http://en.wikipedia.org/wiki/Sunspot

Assignment 2: Basic learning algorithms

- Regression (this lecture):
  - Linear models for regression applied to a real data set in order to predict the number of sunspots from previous years sunspot numbers.
  - A theoretical question about weighted sum of squares
- Classification (next lecture):
  - Experiment with Linear Discriminant Analysis for classification on the Iris data set.
  - Theoretical question about the Bayes optimal classifier

Recall from last lecture

- The least squares solution is equivalent to maximum likelihood (ML) solution under Gaussian noise model. Both have tendency to overfit the data for M>=N (poor generalization).

$$\operatorname{argmin}_{\mathbf{w}} \tilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[y(x_n,\mathbf{w}) - t_n\right]^2 \quad \Leftrightarrow \quad \operatorname{argmax}_{\mathbf{w}} p(\mathbf{T} \mid \mathbf{X},\mathbf{w})$$

- Regularized least squares equivalent to maximum a posteriori (MAP) under Gaussian noise model and isotropic Gaussian prior on model parameters. Both behaves well for M>=N (or at least better than ML solution).

$$\operatorname{argmin}_{\mathbf{w}} \tilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[y(x_n,\mathbf{w}) - t_n\right]^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2 \quad \Leftrightarrow \quad \operatorname{argmax}_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{T},\mathbf{X})$$

Goals and approaches for regression

- Goal: Make prediction of unseen data.

- Approaches to making predictions in regression:

  $y(\mathrm{x})$ (Regression function)

  $p(t \mid x)$ (Predictive distribution)

  $p(t,x)$ (Joint distribution)

- Today we generalize the regression model and develop the full Bayesian approach to regression.

Decision theoretic interpretation of regression: Minimizing the risk (Recap)

- Assume we have 100 independent data sets $S=\{(t_1,x_1),\ldots,(t_N,x_N)\}_{i=1,\ldots,100}$
- We want to choose a model $y(\mathbf{x})$ that performs well on all these data sets.
- Choose the model that on average (over data sets) gives optimal performance.
- Optimal model? Optimality is defined through the loss function.

- Formally: Minimize the average *loss* $L(t,y(\mathbf{x}))$ (a.ka. the empirical risk) we incur by modeling data t with the model $y(\mathbf{x})$

$$R_S(y(\mathbf{x})) = \frac{1}{N}\sum_{n=1}^{N} L(t_n, y(\mathbf{x}_n))$$

- Or minimize the (theoretical) *risk*

$$R_p(y(\mathbf{x})) = E[L] = \iint L(t, y(\mathbf{x}))p(t,\mathbf{x})d\mathbf{x}dt$$

Decision theoretic interpretation of regression: Minimizing the risk (Recap)

- Common regression loss function: $L(t, y(\mathbf{x})) = (y(\mathbf{x}) - t)^2$

$$R_p(y(\mathbf{x})) = E[L] = \iint (y(\mathbf{x}) - t)^2 \, p(t, \mathbf{x}) \, d\mathbf{x} \, dt$$

- Minimization using calculus of variation (see Appendix D) leads to:

$$y(\mathbf{x}) = \int t \, p(t \mid \mathbf{x}) dt = E_t[t \mid \mathbf{x}]$$

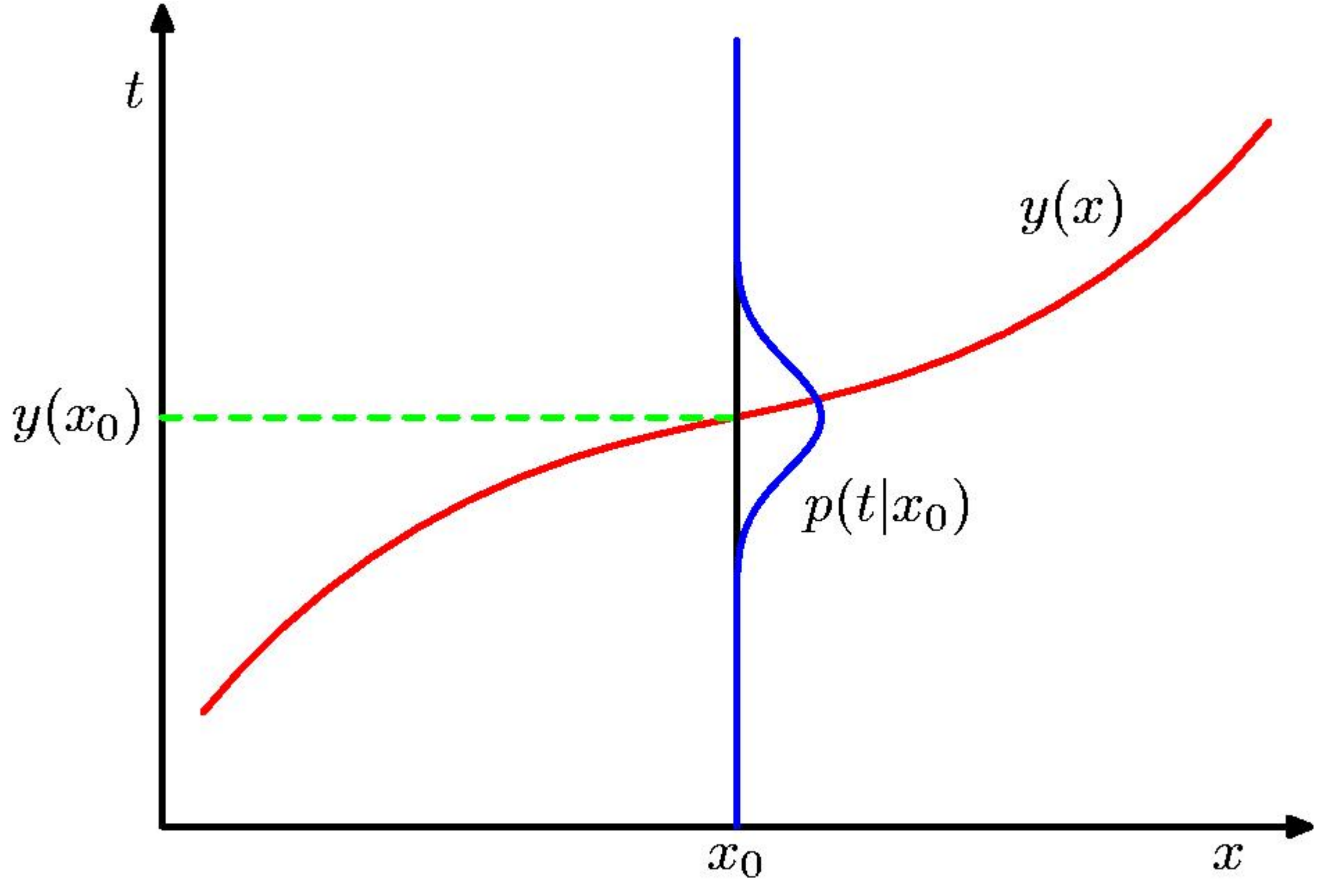
- That is, the optimal solution under squared loss is given by the conditional mean of *t* given **x** with respect to the predictive distribution.
- Or said in another way: The solution is given by the mean of the predictive distribution

Optimal regression function under quadratic loss

The noise model:

$$t(\mathbf{x}) = y(\mathbf{x},\mathbf{w}) + \varepsilon$$

$$\varepsilon \sim \mathcal{N}(t \mid 0, \beta^{-1})$$



Noise model leads to:

$$p(t \mid \mathbf{x},\mathbf{w},\beta) = \mathcal{N}\left(t \mid y(\mathbf{x},\mathbf{w}),\beta^{-1}\right) \Rightarrow E_t[t \mid \mathbf{x}] = \int t\, p(t \mid \mathbf{x},\mathbf{w},\beta)dt = y(\mathbf{x},\mathbf{w})$$

Linear basis function models (Recap)

- Training data set: $X = \{x_1, \ldots, x_N\}$

$$T = \{t_1, \ldots, t_N\}$$

- The (M-1)'th order polynomial model is linear in the *M* model parameters:

$$y(x,\mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_{M-1} x^{M-1} = w_0 + \sum_{j=1}^{M-1} w_j x^j$$

- Generalize this model using (non-linear) basis functions:

$$y(x,\mathbf{w}) = w_0 + w_1 \phi_1(x) + w_2 \phi_2(x) + \cdots + w_{M-1} \phi_{M-1}(x) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(x)$$

- In vector notation using $\phi_0(x) = 1$:

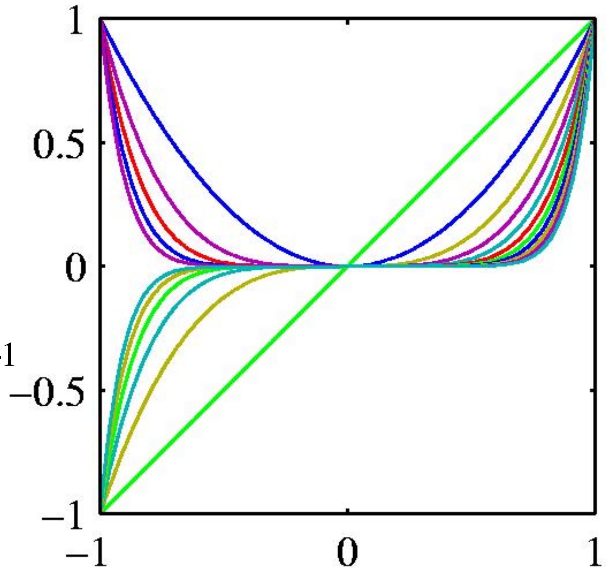
$$y(x,\mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(x) = \mathbf{w}^T \overline{\phi}(x)$$

$$\mathbf{w} = \left(w_0, \ldots, w_{M-1}\right)^T \ , \ \overline{\phi}(x) = \left(\phi_0(x), \ldots, \phi_{M-1}(x)\right)^T$$

Examples of basis functions (Recap)

- Simple D-dim. linear model: Assume $\mathbf{x} = \left(x_1,\ldots,x_D\right)^T$
  Basis functions:
  $$\phi_j(\mathbf{x}) = x_j \quad , \quad \overline{\phi}(\mathbf{x}) = (1, x_1, \ldots, x_D)^T$$
  Regression model:
  $$y(\mathbf{x},\mathbf{w}) = \mathbf{w}^T \overline{\phi}(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_D x_D$$

- Polynomial model (monomial basis):
  Basis functions:
  $$\phi_j(x) = x^j \quad , \quad \overline{\phi}(x) = (1, x, x^2, \ldots, x^{M-1})^T$$
  Regression model:
  $$y(x,\mathbf{w}) = \mathbf{w}^T \overline{\phi}(\mathrm{x}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_{M-1} x^{M-1}$$

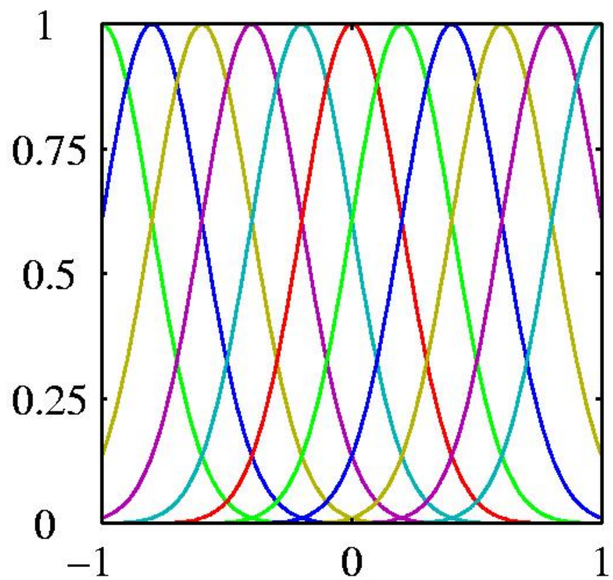


Examples of basis functions

- Gaussian basis function:
  Basis functions:
  $$\phi_j(x) = \exp\left[-\frac{(x-\mu_j)^2}{2s^2}\right]$$
  Regression model:

$$y(x,\mathbf{w}) = w_0 + w_1 \exp\left[-\frac{(x-\mu_1)^2}{2s^2}\right] + \cdots + w_{M-1} \exp\left[-\frac{(x-\mu_{M-1})^2}{2s^2}\right]$$

μ_j position of basis function and s scale



- Other basis functions:
  - Sigmoid
  - Fourier
  - Wavelets
  - Splines (piecewise polynomial), …

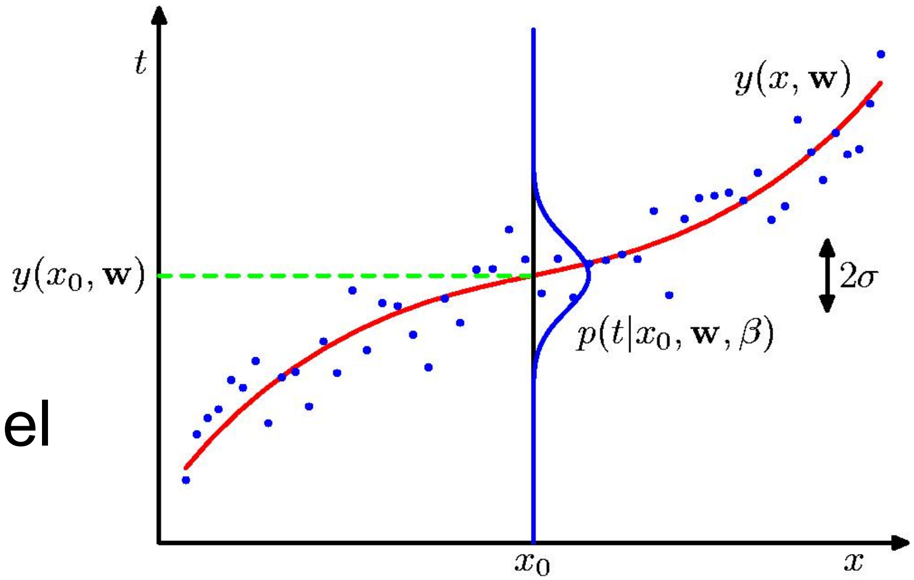
Likelihood under general linear model

Observations (i.i.d.):

$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$

$$\mathbf{T} = (t_1, \ldots, t_N)^T$$

$$p(t \mid \mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}\left(t \mid y(\mathbf{x}, \mathbf{w}), \beta^{-1}\right)$$



Under the Gaussian noise model we have the likelihood:

$$p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n \mid \mathbf{w}^T \overline{\phi}(\mathbf{x}_n), \beta^{-1}\right)$$

$$= \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left[-\frac{\beta}{2} \sum_{n=1}^{N} \left(t_n - \mathbf{w}^T \overline{\phi}(\mathbf{x}_n)\right)^2\right]$$

Maximum Likelihood (ML) solution for the general linear model

Maximize the log-likelihood with respect to **w**:

$$\frac{\partial}{\partial w_j}\log p(\mathbf{T}\,|\,\mathbf{X},\mathbf{w},\beta)=\frac{\partial}{\partial w_j}\left[-\frac{\beta}{2}\sum_{n=1}^{N}\left(t_n-\mathbf{w}^T\overline{\phi}(\mathbf{x}_n)\right)^2\right]=0\ \text{ for all } j$$

$$\Downarrow$$

$$\mathbf{w}=\left(\left[\sum_{n=1}^{N}\overline{\phi}(\mathbf{x}_n)\overline{\phi}^T(\mathbf{x}_n)\right]^{-1}\right)^T\left(\sum_{n=1}^{N}t_n\overline{\phi}^T(\mathbf{x}_n)\right)^T$$

Voila, we get the ML solution – but what an ugly expression!

The design matrix

- Introduce design matrix notation: $\Phi_{nj} = \phi_j(\mathbf{x}_n)$

$$\Phi = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{bmatrix} \in \mathbb{R}^{N\times M}$$

- Each row contains the outcome of evaluating the *M* basis functions in the *n*'th data point.

Examples of design matrices

- 1-dim. (M-1)'th order polynomial model:

$$y(x,\mathbf{w}) = w_0 + w_1 x + \cdots + w_{M-1} x^{M-1}$$

$$\Phi = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{M-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{M-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & x_N^2 & \cdots & x_N^{M-1} \end{bmatrix}$$

- 1-dim. linear model: $y(x,\mathbf{w}) = w_0 + w_1 x$

$$\Phi = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix}$$

The design matrix specifies the likelihood

- We can rewrite the likelihood using the design matrix:

$$p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n \mid \mathbf{w}^T \overline{\phi}(\mathbf{x}_n), \beta^{-1}\right) = \mathcal{N}\left(\mathbf{T} \mid \Phi\mathbf{w}, \beta^{-1}\mathbf{I}\right)$$

- The design matrix and the target vector **T** represents the training data in the likelihood.

- With unspecified **T** the likelihood is a N-dim. multivariate Gaussian with isotropic covariance.

ML solution for general linear model

- Maximize with respect to parameters **w**:

$$\mathbf{w}_{\mathrm{ML}} = \left(\left[\sum_{n=1}^{N} \bar{\phi}(\mathbf{x}_n) \bar{\phi}^T(\mathbf{x}_n) \right]^{-1} \right)^T \left(\sum_{n=1}^{N} t_n \bar{\phi}^T(\mathbf{x}_n) \right)^T = \left(\Phi^T \Phi \right)^{-1} \Phi^T \mathbf{T}$$

- Maximize with respect to precision β:

$$\frac{\partial}{\partial \beta} \log p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}_{\mathrm{ML}}, \beta) = \frac{N}{2} \frac{1}{\beta} - \frac{1}{2} \sum_{n=1}^{N} \left(t_n - \mathbf{w}_{\mathrm{ML}}^T \bar{\phi}(\mathbf{x}_n) \right)^2 = 0 \Rightarrow$$

$$\frac{1}{\beta_{\mathrm{ML}}} = \frac{1}{N} \sum_{n=1}^{N} \left(t_n - \mathbf{w}_{\mathrm{ML}}^T \bar{\phi}(\mathbf{x}_n) \right)^2$$

Example: Prediction of body fat percentage

- **The problem:**
  - Measuring percentage of body fat accurately is inconvenient/ costly and requires weighing the body underwater and in air and applying the so-called Siri's equation.
  - How accurately can we predict the percentage of body fat from measurements of circumferences of selected body parts?
- **Data set (measurements from *N*=252 men):**
  - Density determined from underwater weighing
  - Percentage body fat from Siri's (1956) equation
  - Age (years)
  - Weight (lbs)
  - Height (inches)
  - Circumferences (cm): Neck, Chest, Abdomen 2, Hip, Thigh, Knee, Ankle, Biceps (extended), Forearm, Wrist

Example: Prediction of body fat percentage

- Consider a 1-dim. linear model of a subset of the percentage body fat data set by selecting column 8 (Abdomen 2) as the x variable:

t [%]	x [cm]
14.7	83.3
17.8	88.2
16.9	90.3
32.6	113.4
5.7	84.5
32.6	108.1
15.2	98.8
25.3	108.8

Design matrix

$$\Phi = \begin{bmatrix} 1 & 83.3 \\ 1 & 88.2 \\ 1 & 90.3 \\ 1 & 113.4 \\ 1 & 84.5 \\ 1 & 108.1 \\ 1 & 98.8 \\ 1 & 108.8 \end{bmatrix}$$

Target vector

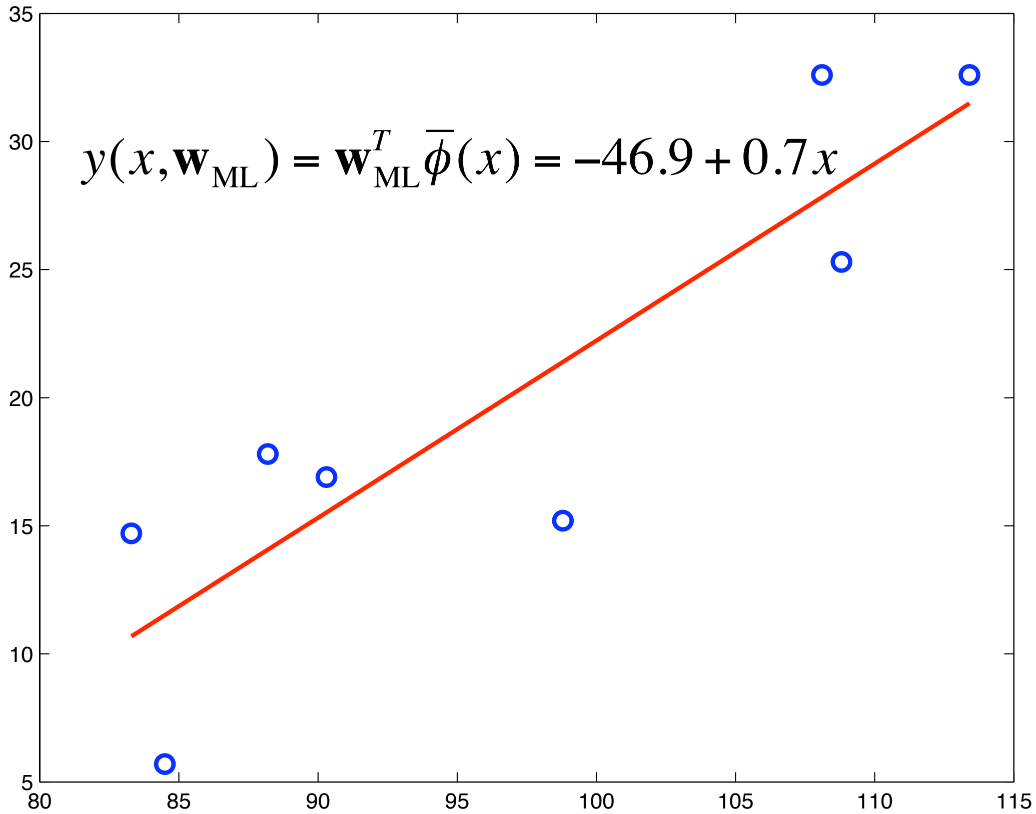
$$\mathbf{T} = \begin{bmatrix} 14.7 \\ 17.8 \\ 16.9 \\ 32.6 \\ 5.7 \\ 32.6 \\ 15.2 \\ 25.3 \end{bmatrix}$$

Example: Prediction of body fat percentage

Design matrix Target vector

$$\Phi = \begin{bmatrix} 1 & 83.3 \\ 1 & 88.2 \\ 1 & 90.3 \\ 1 & 113.4 \\ 1 & 84.5 \\ 1 & 108.1 \\ 1 & 98.8 \\ 1 & 108.8 \end{bmatrix} \qquad \mathbf{T} = \begin{bmatrix} 14.7 \\ 17.8 \\ 16.9 \\ 32.6 \\ 5.7 \\ 32.6 \\ 15.2 \\ 25.3 \end{bmatrix}$$

$$\mathbf{w}_{\mathrm{ML}} = \left(\Phi^T \Phi\right)^{-1} \Phi^T \mathbf{T} = \begin{bmatrix} -46.9 \\ 0.7 \end{bmatrix}$$



Summary: Maximum likelihood (ML) regression

- Learn the model parameters **w** on the training set:

$$X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$

$$\mathbf{T} = (t_1, \ldots, t_N)^T$$

$$\Phi = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{bmatrix}$$

$$\mathbf{w}_{\mathrm{ML}} = \left(\Phi^T \Phi\right)^{-1} \Phi^T \mathbf{T} \qquad \beta_{\mathrm{ML}}^{-1} = \frac{1}{N} \sum_{n=1}^{N} \left(t_n - \mathbf{w}_{\mathrm{ML}}^T \overline{\phi}(\mathbf{x}_n)\right)^2$$

- Apply the model to new data **x**:

$$y(\mathbf{x}, \mathbf{w}_{\mathrm{ML}}) = \mathbf{w}_{\mathrm{ML}}^T \overline{\phi}(\mathbf{x})$$

$$p(t \mid \mathbf{x}, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}^{-1}) = \mathcal{N}\left(t \mid y(\mathbf{x}, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$

Summary: Maximum likelihood (ML) regression

- Apply the model to the test set

$$\tilde{X} = \left\{ \tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{\tilde{N}} \right\}$$

$$\tilde{\mathbf{T}} = (\tilde{t}_1, \ldots, \tilde{t}_{\tilde{N}})^T$$

- and compute root-mean-square error:

$$\mathrm{RMS} = \sqrt{\frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} \left(\tilde{t}_n - y(\tilde{\mathbf{x}}_n, \mathbf{w}_{\mathrm{ML}}) \right)^2}$$

Example: Prediction of body fat percentage

Root mean square error on the training set:

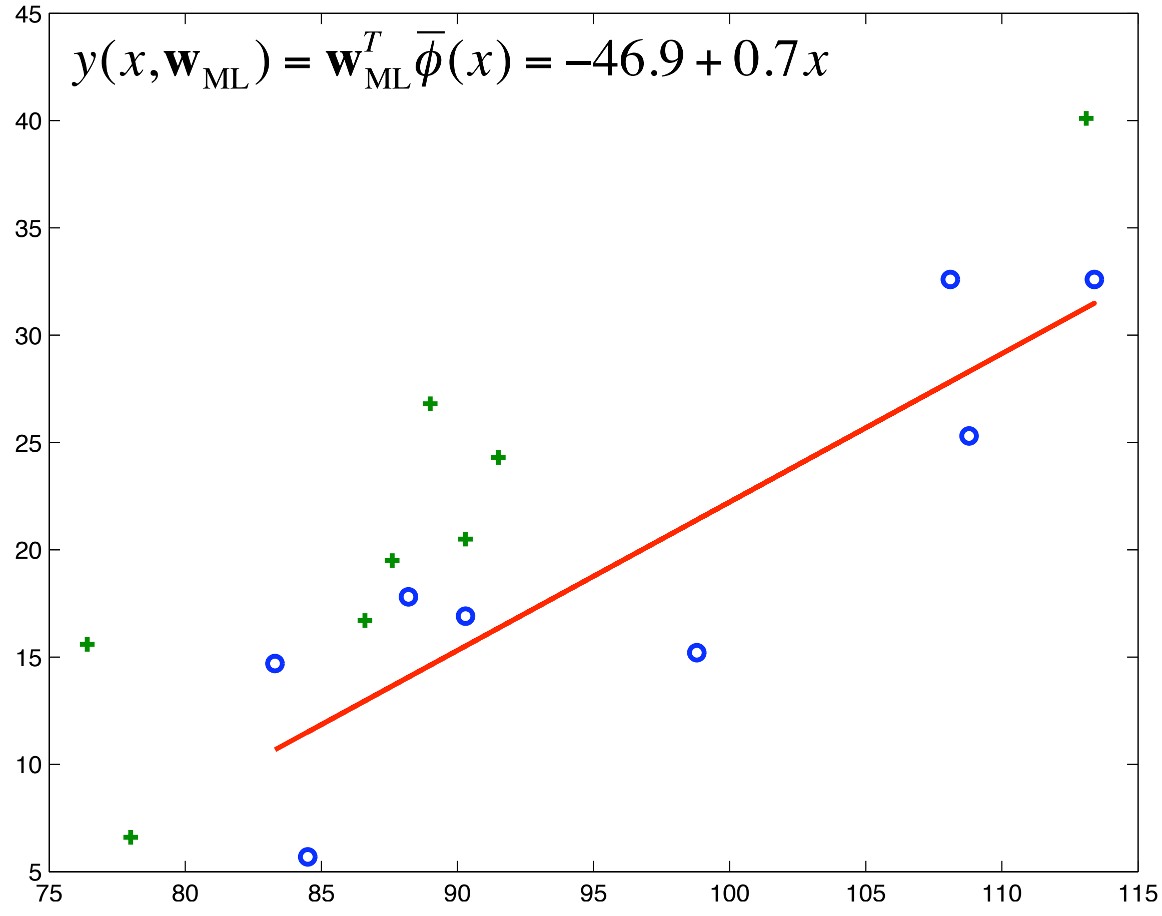
$$\text{RMS} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(t_n - y(\mathbf{x}_n, \mathbf{w}_{\text{ML}})\right)^2}$$

$$= 4.1$$

Root mean square error on the test set:

$$\text{RMS} = \sqrt{\frac{1}{\tilde{N}}\sum_{n=1}^{\tilde{N}}\left(\tilde{t}_n - y(\tilde{\mathbf{x}}_n, \mathbf{w}_{\text{ML}})\right)^2}$$

$$= 7.6$$



Bayesian linear regression (MAP)

- The Gaussian likelihood from before (Assume known noise precision *β*):

$$p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}, \beta) = \mathcal{N}\left(\mathbf{T} \mid \Phi\mathbf{w}, \beta^{-1}\mathbf{I}\right)$$

- Add a prior to regularize the solution, thereby reducing the risk of *overfitting* to the training.
- Pick the (conjugated) Gaussian prior for the parameters

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

- Posterior:

$$p(\mathbf{w} \mid \mathbf{T}, \mathbf{X}, \beta) = \frac{p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}, \beta)\, p(\mathbf{w})}{p(\mathbf{T})}$$

Bayesian linear regression (MAP)

- The posterior is a Gaussian:

$$p(\mathbf{w} \mid \mathbf{T}, \mathbf{X}, \beta) = \mathcal{N}\left(\mathbf{T} \mid \Phi\mathbf{w}, \beta^{-1}\mathbf{I}\right)\mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, \mathbf{S}_0) / p(\mathbf{T})$$

$$= \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N)$$

- Posterior covariance and mean (by completing the square):

$$p(\mathbf{w} \mid \mathbf{T}, \mathbf{X}, \beta) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N)$$

$$\mathbf{S}_N = \left(\mathbf{S}_0^{-1} + \beta\Phi^T\Phi\right)^{-1} \in \mathbb{R}^{M \times M}$$

$$\mathbf{m}_N = \mathbf{S}_N\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\Phi^T\mathbf{T}\right) \in \mathbb{R}^M$$

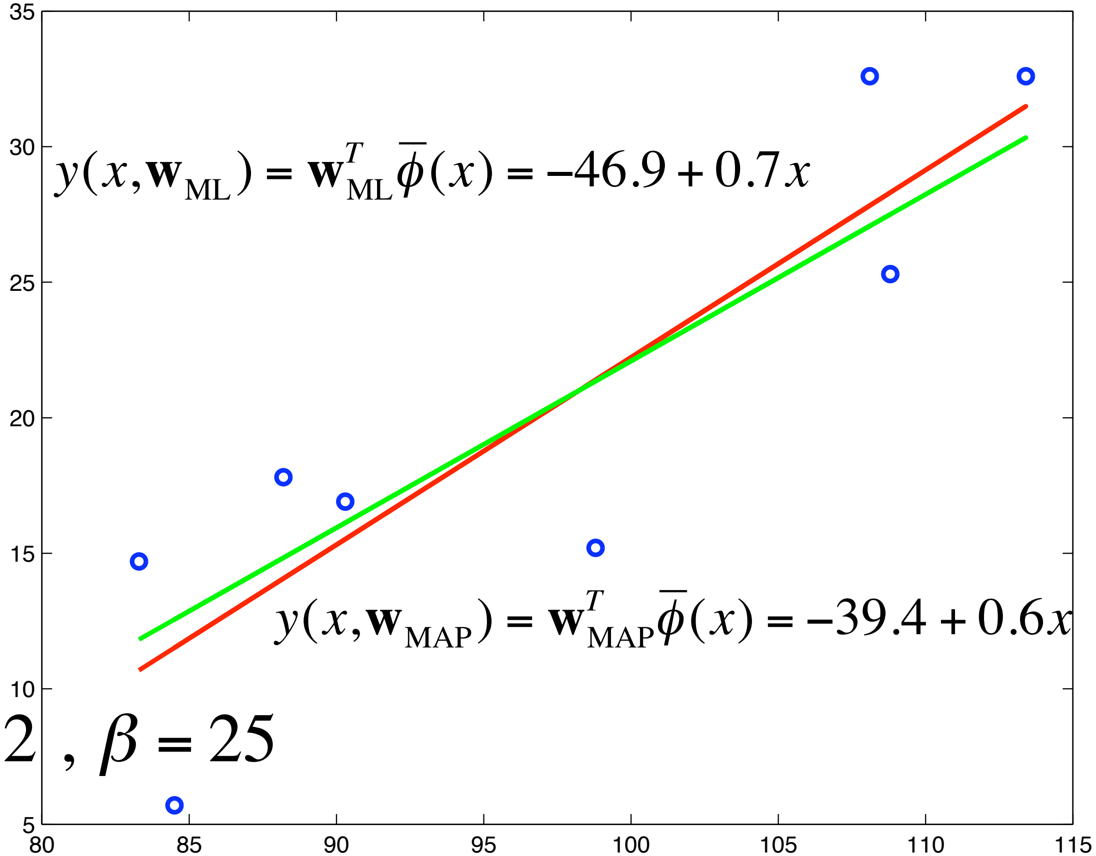
- MAP solution: $\mathbf{w}_{\mathrm{MAP}} = \mathbf{m}_N$

Example: Prediction of body fat percentage
Regression using $\mathbf{S}_0 = \alpha^{-1}\mathbf{I}$, $\mathbf{m}_0 = \mathbf{0}$

Design matrix Target vector

$$\mathbf{\Phi} = \begin{bmatrix} 1 & 83.3 \\ 1 & 88.2 \\ 1 & 90.3 \\ 1 & 113.4 \\ 1 & 84.5 \\ 1 & 108.1 \\ 1 & 98.8 \\ 1 & 108.8 \end{bmatrix} \qquad \mathbf{T} = \begin{bmatrix} 14.7 \\ 17.8 \\ 16.9 \\ 32.6 \\ 5.7 \\ 32.6 \\ 15.2 \\ 25.3 \end{bmatrix}$$

$$\mathbf{w}_{\mathrm{ML}} = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\mathbf{T} = \begin{bmatrix} -46.9 \\ 0.7 \end{bmatrix}$$



$$\mathbf{w}_{\mathrm{MAP}} = \mathbf{m}_N = \begin{bmatrix} -39.4 \\ 0.6 \end{bmatrix}, \ \alpha = 1/2 \ , \ \beta = 25$$

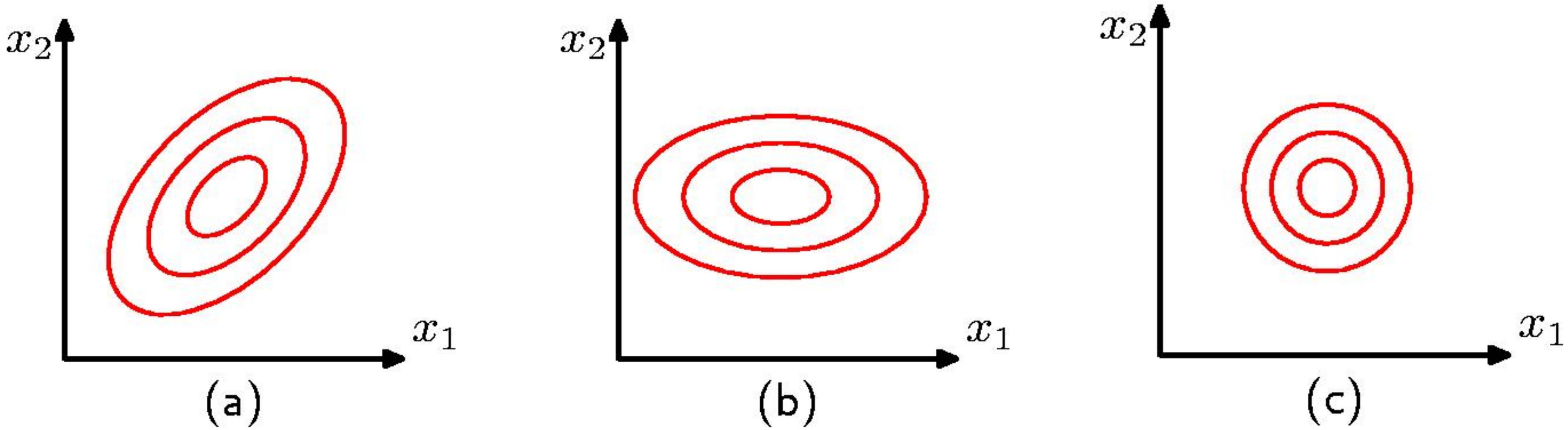
Example: Prediction of body fat percentage

- Root mean square error on the test set:

$$\mathrm{RMS(ML)} = \sqrt{\frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} \left(\tilde{t}_n - y(\tilde{\mathbf{x}}_n, \mathbf{w}_{\mathrm{ML}}) \right)^2} = 7.6$$

$$\mathrm{RMS(MAP)} = \sqrt{\frac{1}{\tilde{N}} \sum_{n=1}^{\tilde{N}} \left(\tilde{t}_n - y(\tilde{\mathbf{x}}_n, \mathbf{w}_{\mathrm{MAP}}) \right)^2} = 7.1$$

Simplified Gaussian Prior Models



$\mathbf{S}_0$ $\quad$ $\mathbf{S}_0 = \mathrm{diag}(\alpha_1^{-1}, \ldots, \alpha_D^{-1})$ $\quad$ $\mathbf{S}_0 = \alpha^{-1}\mathbf{I}$

Choose an appropriate prior, but consider:

- (a) In general the covariance matrix consists of $D(D+1)/2$ free parameters.
- Reduce the amount of parameters to D in the diagonal model (b) and 1 in the isotropic model (c).

Bayesian linear regression: Effect of the prior

- In order to simplify, lets assume an isotropic prior: $\mathbf{S}_0 = \alpha^{-1}\mathbf{I}$

- Uniform prior: $\alpha^{-1} \to \infty \quad , \quad \alpha \to 0: \qquad \mathbf{S}_0^{-1} = \alpha\mathbf{I} \to \underline{\underline{\mathbf{0}}}$

$$\mathbf{S}_N = \left(\mathbf{S}_0^{-1} + \beta\Phi^T\Phi\right)^{-1} \to \left(\beta\Phi^T\Phi\right)^{-1}$$

$$\mathbf{m}_N = \mathbf{S}_N\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\Phi^T\mathbf{T}\right) \to$$

$$\mathbf{S}_N\beta\Phi^T\mathbf{T} = \beta^{-1}\left(\Phi^T\Phi\right)^{-1}\beta\Phi^T\mathbf{T} = \left(\Phi^T\Phi\right)^{-1}\Phi^T\mathbf{T} = \mathbf{w}_{\mathrm{ML}}$$

- No data: $N = 0$

$$\mathbf{S}_N = \mathbf{S}_0$$

$$\mathbf{m}_N = \mathbf{S}_N\mathbf{S}_0^{-1}\mathbf{m}_0 = \mathbf{S}_0\mathbf{S}_0^{-1}\mathbf{m}_0 = \mathbf{m}_0 \quad (\text{posterior} \to \text{prior})$$

Bayesian sequential learning for regression
The online learning version

Posterior for N-1 acts as prior for parameter at *N*

$$p(\mathbf{w} \mid \mathbf{T}, \mathbf{X}, \beta) = \frac{p(\mathbf{T} \mid \mathbf{X}, \mathbf{w}, \beta) p(\mathbf{w})}{p(\mathbf{T})} \propto$$

$$\underbrace{p(t_N \mid \mathbf{x}_N, \mathbf{w}, \beta)}_{\text{likelihood for } N} \underbrace{p(\mathbf{w}) \prod_{n=1}^{N-1} p(t_n \mid \mathbf{x}_n, \mathbf{w}, \beta)}_{\text{posterior for } N\text{-1}} =$$

$$\prod_{n=1}^{N} \mathcal{N}\left(t_n \mid \mathbf{w}^T \bar{\phi}(\mathbf{x}_n), \beta^{-1}\right) \mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, \mathbf{S}_0) =$$

$$\mathcal{N}\left(t_N \mid \mathbf{w}^T \bar{\phi}(\mathbf{x}_N), \beta^{-1}\right) \prod_{n=1}^{N-1} \mathcal{N}\left(t_n \mid \mathbf{w}^T \bar{\phi}(\mathbf{x}_n), \beta^{-1}\right) \mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, \mathbf{S}_0)$$

Example: Line regression

- Synthetic data set: $f(x,\mathbf{a}) = a_0 + a_1 x \quad , \quad a_0 = -0.3 \ , \ a_1 = 0.5$

  $t_n = f(x_n,\mathbf{a}) + \varepsilon \quad , \quad \varepsilon \sim \mathcal{N}(\varepsilon \mid 0, 0.2^2) \ , \ x_n \sim \mathcal{U}(x \mid -1, 1)$
- Regression model: $y(x,\mathbf{w}) = w_0 + w_1 x$
- Lets assume the isotropic prior: $\mathbf{S}_0 = \alpha^{-1}\mathbf{I} \quad , \quad \mathbf{m}_0 = \mathbf{0}$

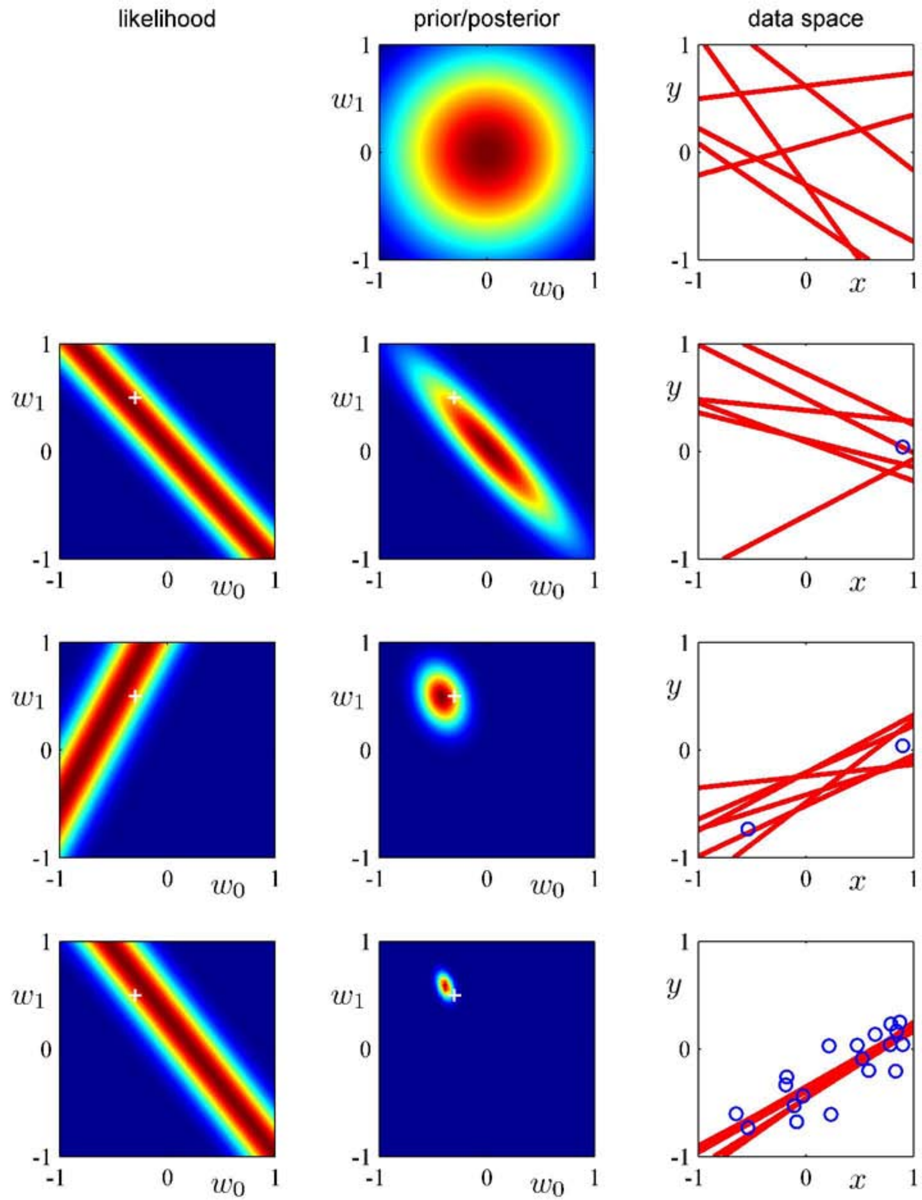
  $p(\mathbf{w} \mid \alpha) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, \mathbf{S}_0)$
- Then posterior mean and covariance becomes:

$$\mathbf{S}_N = \left(\mathbf{S}_0^{-1} + \beta \Phi^T \Phi\right)^{-1} = \left(\alpha \mathbf{I} + \beta \Phi^T \Phi\right)^{-1}$$

$$\mathbf{m}_N = \mathbf{S}_N\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta \Phi^T \mathbf{T}\right) = \beta \mathbf{S}_N \Phi^T \mathbf{T}$$

- Assume noise precision known $\beta=(1/0.2^2)=25$ and set prior precision to $\alpha=2$ and the MAP estimate of parameters is then $\mathbf{w}_{MAP}=\mathbf{m}_N$.

Example: Line regression



Full Bayesian Approach (Advanced): Model independent predictive distribution

- In general, we don't care about the specific choice of parameter, but want to make predictions of new unseen data:

$$p(t \mid x) \qquad \text{(Predictive distribution)}$$

- Including the observations (training set), the *model independent predictive distribution* is given by marginalization over all models:

$$p(t \mid \mathbf{x},\mathbf{T},\mathbf{X},\alpha,\beta) = \int \overbrace{\underbrace{p(t \mid \mathbf{x},\mathbf{w},\beta)}_{\text{Noise model}}\underbrace{p(\mathbf{w} \mid \mathbf{T},\mathbf{X},\alpha,\beta)}_{\text{Posterior}}}^{p(t,\mathbf{w} \mid \mathbf{x},\mathbf{T},\mathbf{X},\alpha,\beta)} d\mathbf{w}$$

Gaussian predictive distribution

- Consider the case of Gaussian noise model, prior and posterior:

$$p(t \mid \mathbf{x},\mathbf{w},\beta) = \mathcal{N}\left(t \mid y(\mathbf{x},\mathbf{w}),\beta^{-1}\right) \quad \text{(Noise model)}$$

$$p(\mathbf{w} \mid \mathbf{T},\mathbf{X},\alpha,\beta) = \mathcal{N}\left(\mathbf{w} \mid \mathbf{m}_N,\mathbf{S}_N\right) \quad \text{(Posterior)}$$

- Predictive distribution:

$$p(t \mid \mathbf{x},\mathbf{T},\mathbf{X},\alpha,\beta) = \int \underbrace{p(t \mid \mathbf{x},\mathbf{w},\beta)}_{\text{Noise model}}\underbrace{p(\mathbf{w} \mid \mathbf{T},\mathbf{X},\alpha,\beta)}_{\text{Posterior}}d\mathbf{w}$$

$$= \int \mathcal{N}\left(t \mid y(\mathbf{x},\mathbf{w}),\beta^{-1}\right)\mathcal{N}\left(\mathbf{w} \mid \mathbf{m}_N,\mathbf{S}_N\right)d\mathbf{w}$$

$$= \int \mathcal{N}\left(t,\mathbf{w} \mid \mathbf{x},\beta^{-1},\mathbf{m}_N,\mathbf{S}_N\right)d\mathbf{w} = \mathcal{N}\left(t \mid y(\mathbf{x},\mathbf{m}_N),\sigma_N^2(\mathbf{x})\right)$$

$$\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \overline{\phi}^T(\mathbf{x})\mathbf{S}_N\overline{\phi}(\mathbf{x}) \qquad \text{(Predictive variance)}$$

Example: Sinusoidal data set

- Synthetic sinusoidal data set: $X = (x_1, \ldots, x_N)^T$

  $t(x) = Sin(2\pi x) + \chi \ , \ \chi \sim N(\chi \mid 0, 0.3^2)$ $\quad T = (t_1, \ldots, t_N)^T$

- Linear regression with 9 Gaussian basis functions:

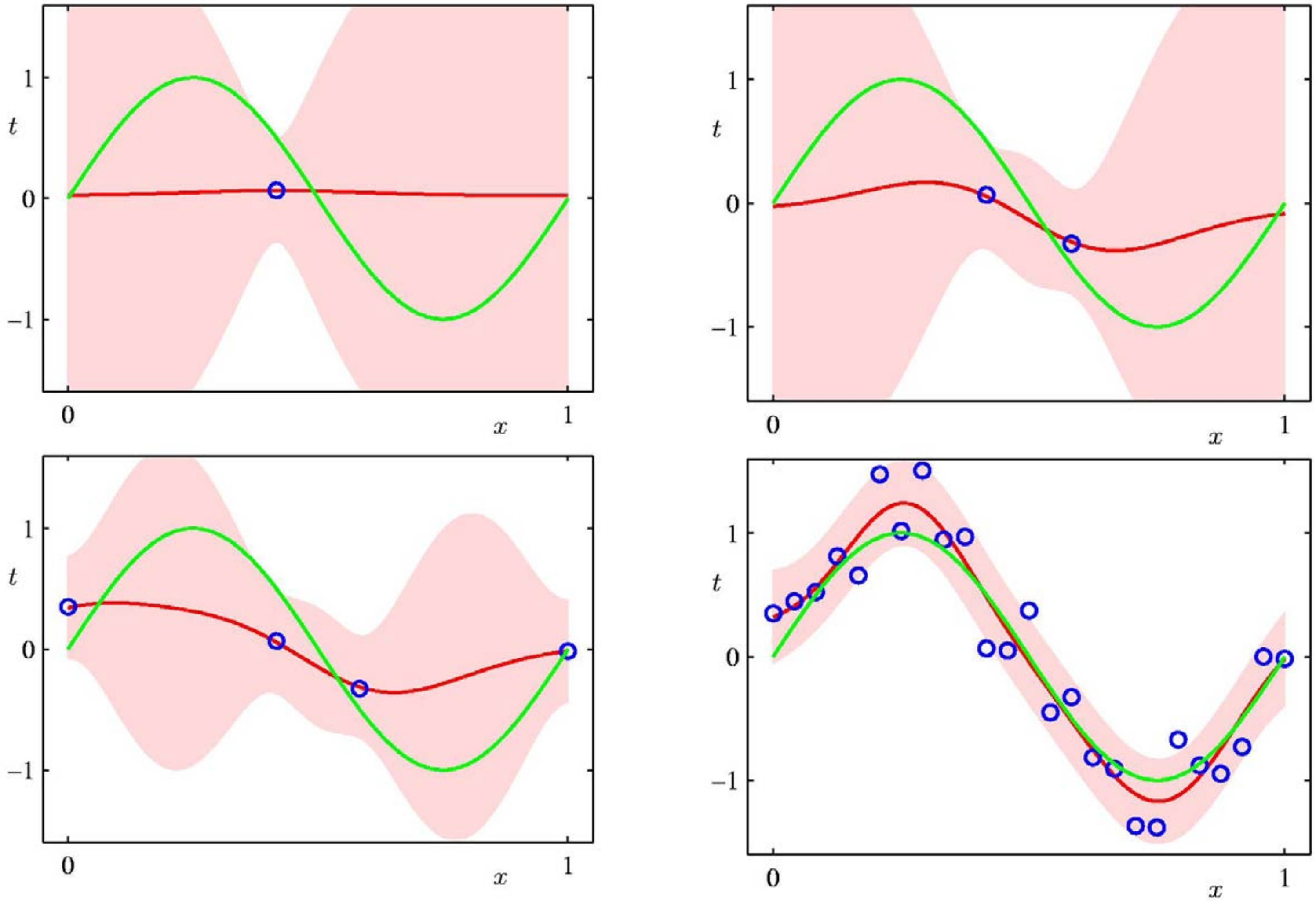
$$\phi_j(x) = \exp\left[-\frac{(x-\mu_j)^2}{2s^2}\right]$$

- Lets plot the predictive mean curve:

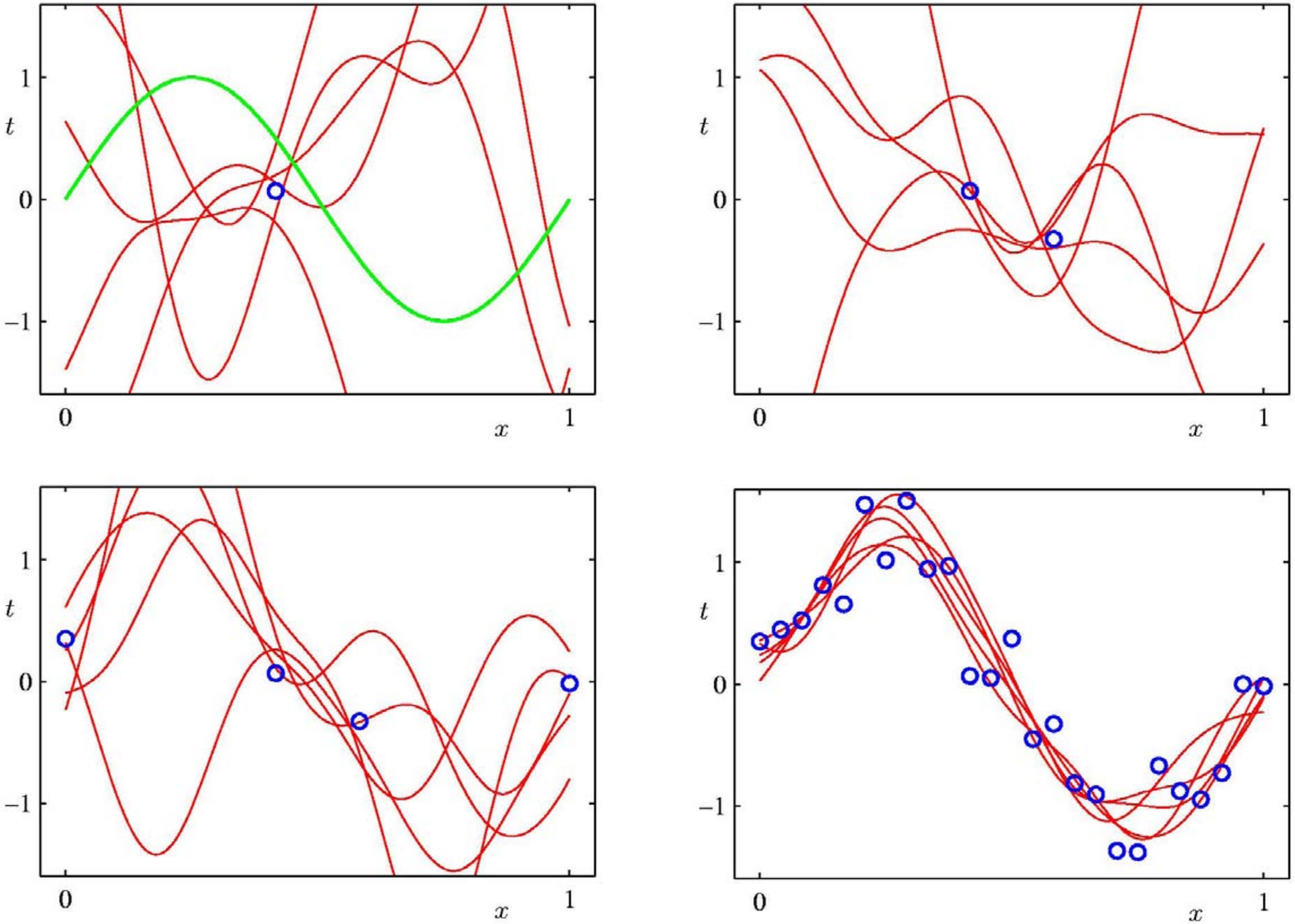
$$\int tp(t \mid \mathbf{x}, \mathbf{T}, \mathbf{X}, \alpha, \beta)dt = \int t\mathcal{N}\left(t \mid y(\mathbf{x}, \mathbf{m}_N), \sigma_N^2(\mathbf{x})\right)dt = y(\mathbf{x}, \mathbf{m}_N)$$

- And the predictive standard deviation curve: $\sigma_N(\mathbf{x})$

Examples of Predictive Distribution



Samples From Posterior Distribution



Summary

- Linear models for regression
- Bayesian regression for linear models
- Bayesian sequential learning for regression
- Full Bayesian approach – the model independent predictive distribution
- Advanced: It is difficult to simultaneously estimate **w**, α, β analytically, however approximations exist (not for this course).

Literature

- Linear models for regression: Sec. 3.1
- Loss function for regression: Sec. 1.5.5
- Bayesian Linear Regression: Sec. 3.3 – 3.3.2
- Limitations on fixed basis functions: Sec. (3.6)