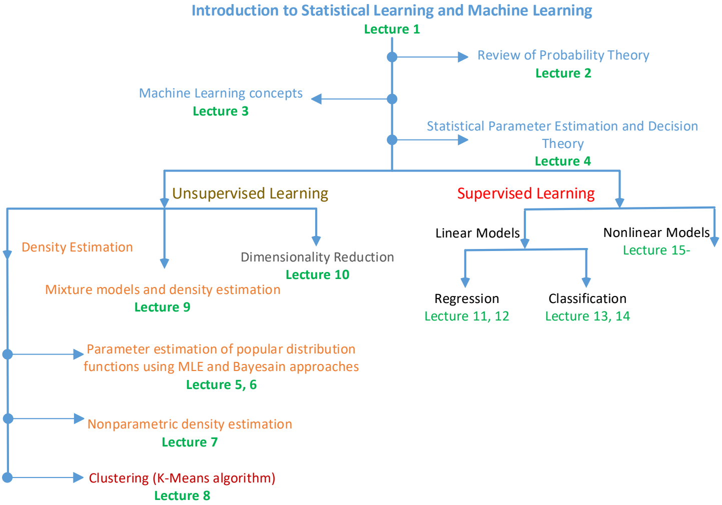
Statistical Learning and Machine Learning

Lecture 11 - Linear Models for Regression 1

October 3, 2021

Course overview and where do we stand

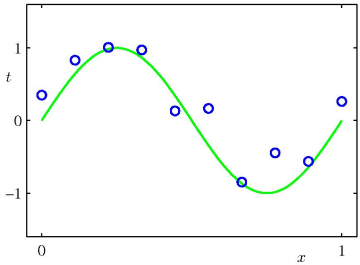


Objectives of the lecture

- Introduction to the linear models for regression
  - Maximum likelihood and least squares method
  - Geometry of least squares
  - regularized/weighted least squares

Goal of Regression

- The goal of regression is to predict the value of one or more continuous *target* variables t given the value of a D-dimensional vector $\boldsymbol{x}$ of *input* variables.
- Supervised learning: Training data consisting of N observations $\{\boldsymbol{x}_n\}$ for $n = 1, \ldots, N$ along with target values $\{t_n\}$ are available.
- Output: A function $y(\boldsymbol{x})$ whose values for new inputs $\boldsymbol{x}$ constitute the predictions t.



Linear Basis Function Models

Two ways to define linear models:

- Linear w.r.t. to both input $\boldsymbol{x}$ and parameters $\boldsymbol{w}$

$$y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + w_1 x_1 + \ldots, + w_D x_D = w_0 + \sum_{j=1}^{D} w_j x_j = \boldsymbol{w}^T \boldsymbol{x}$$

  where $\boldsymbol{x} = (x_0, x_1, \ldots, x_D)^T$ and $x_0 = 1$ in the final form.

- Non-linear functions $\phi_j(\cdot), j = 1, \ldots, M-1$ (basis functions) w.r.t. to the input, with $\phi_0(\boldsymbol{x}) = 1$

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

  where $\boldsymbol{\phi} = (\phi_0, \ldots, \phi_{M-1})^T$ and $\boldsymbol{w} = (w_0, \ldots, w_{M-1})^T$

Examples

Linear basis function in both $\boldsymbol{w}$ and $\boldsymbol{x}$

$$\phi_j(\boldsymbol{x}) = x_j, \text{ for } j = 1, \ldots, D$$

with $\phi_0(\boldsymbol{x}) = 1$.
In vector form, we get

$$\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_0, \ldots, \phi_{M-1})^T = \boldsymbol{x}$$

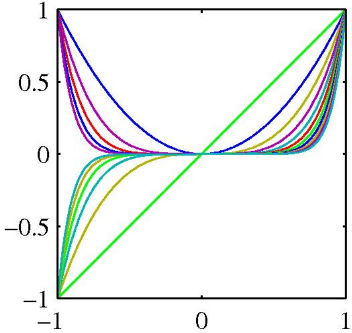
In this case, the output function becomes

$$y(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x}$$

Example basis functions

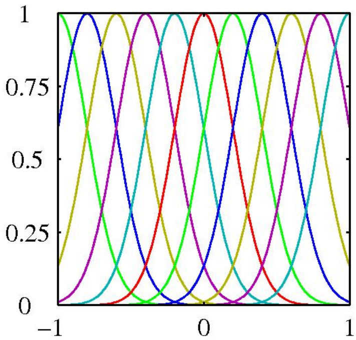
Polynomial basis function:

$$\phi_j(x) = x^j$$



Radial basis function:

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

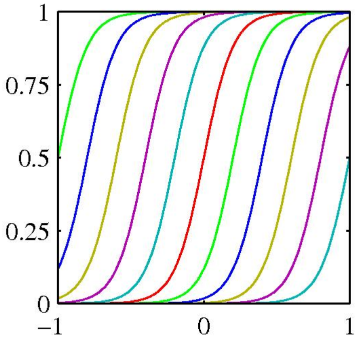


Example basis functions

Sigmoidal basis function:

$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$$

where $\sigma(a) = 1/(1 + exp(-a))$.



Least Squares

- Goal: Given a set of i.i.d. data points $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and the corresponding t_n, $n = 1, \ldots, N$, we want to estimate the parameters $\boldsymbol{w}$ of the regression model.
- Which model?

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

Least Squares

- Cost function to be minimized: Sum of the squares of the individual errors:

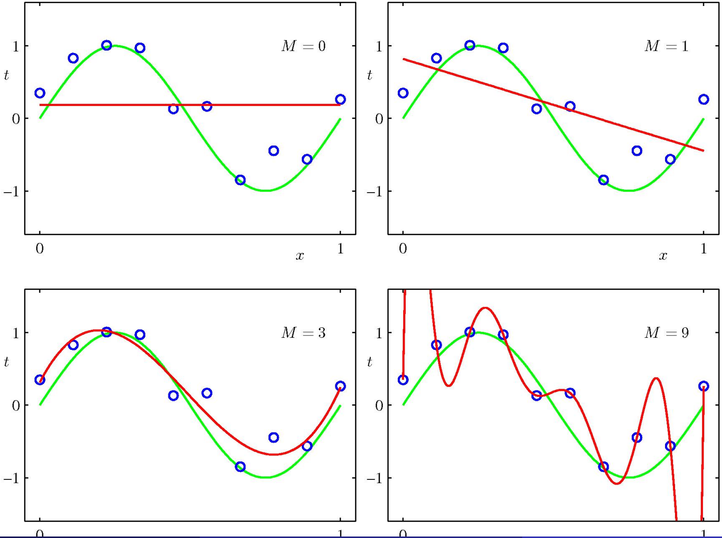
$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\Big(y(\boldsymbol{x}_n, \boldsymbol{w}) - t_n\Big)^2 = \frac{1}{2}\sum_{n=1}^{N}\Big(\boldsymbol{w}^T\phi(\boldsymbol{x}_n) - t_n\Big)^2$$

- Minimization: By setting $\frac{\partial E_D(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0$:

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

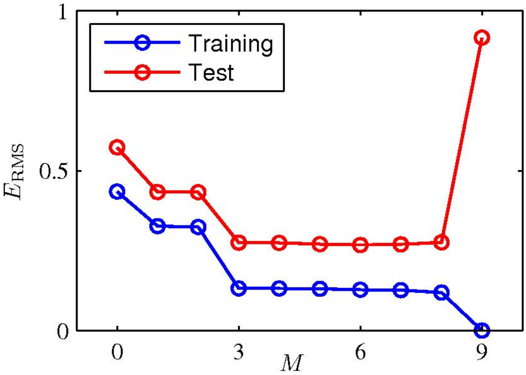
where $\Phi \in \mathbb{R}^{N\times M}$ is formed by using the vectors $\phi(\boldsymbol{x}_n)$ as rows and $\boldsymbol{t} = [t_1, \ldots, t_N]^T$.

Regression using polynomial basis function with varying order M.



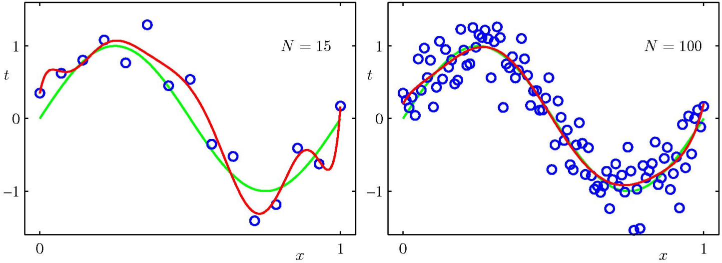
Examples

Error of regression using polynomial basis function with varying order M.



Examples

Regression using polynomial basis function of order $M = 9$ and varying number of data points N.



Overfitting explained

Insight into **Over-fitting** phenomenon for large values of M.

- For $M = 9$, the values of calculated parameters w are very large
- Those large values lead to massive oscillations that are undesirable.

	$M=0$	$M=1$	$M=6$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

Regularized Least Squares

It is possible to *augment* the error function with a regularization term $E_W(\boldsymbol{w})$, which allows to:

- overcome over-fitting to the training data
- avoid problems related to the inversion of singular matrices.

Generic form of a regularization term:

$$E_W(\boldsymbol{w}) = \sum_{j=1}^{M} |w_j|^q$$

Then, the error function takes the form:

$$E(\boldsymbol{w}) = E_D(\boldsymbol{w}) + \lambda E_W(\boldsymbol{w})$$

λ is the regularization coefficient.

Regularized Least Squares

We usually use the l_2 regularization term (which corresponds to $q = 2$):

$$E_W(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}.$$

Then, using $\lambda \geq 0$ the error function becomes:

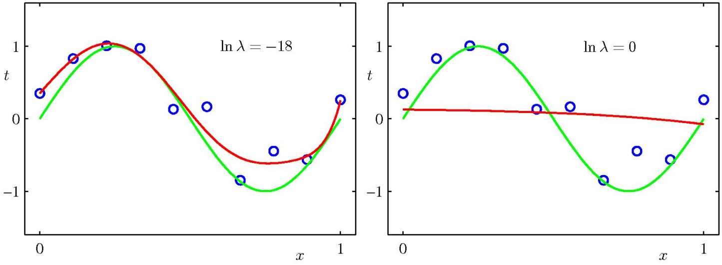
$$E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\boldsymbol{w}^T\phi(\boldsymbol{x}_n) - t_n\right)^2 + \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w}$$

Minimization: Setting $\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0$ and rearranging yields

$$\boldsymbol{w} = \left(\Phi^T\Phi + \lambda \boldsymbol{I}\right)^{-1}\Phi^T\boldsymbol{t}$$

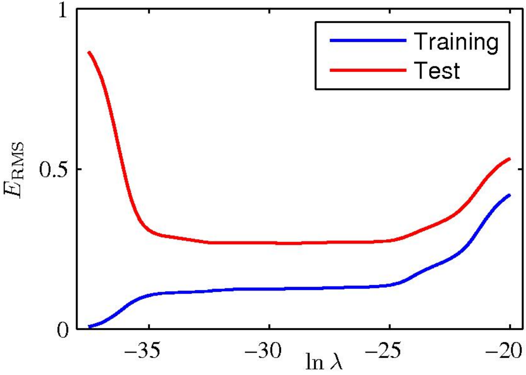
Examples

Regularized regression using polynomial basis function of order $M = 9$, number of data points $N = 10$ and different regularization parameter values λ.



Examples

Error of regularized regression using polynomial basis function of order $M = 9$, number of data points $N = 10$ and varying regularization parameter values λ.



Maximum Likelihood

We assume that the target variable t takes the following form

$$t = y(\boldsymbol{x}, \boldsymbol{w}) + \epsilon$$

where ϵ expresses a noise factor.

Gaussian Noise Assumption: We model $\epsilon \in \mathcal{N}(0, \beta^{-1})$. Then predictive distribution of t given the observation $\boldsymbol{x}$ and $\boldsymbol{w}$ becomes

$$p(t|\boldsymbol{x}, \boldsymbol{w}, \beta) = \mathcal{N}(t|y(\boldsymbol{x}, \boldsymbol{w}), \beta^{-1}).$$

Can you justify the above predictive distribution?

Given a set of i.i.d. data points $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$ and the corresponding $t_n,\ n = 1, \ldots, N$, joint predictive distribution of $\boldsymbol{t}$ is

$$p(\boldsymbol{t}|\boldsymbol{X}, \boldsymbol{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\boldsymbol{w}^T \phi(\boldsymbol{x}_n), \beta^{-1}).$$

Maximum Likelihood

The log-likelihood function is:

$$\begin{aligned} \ln p(\boldsymbol{t}|\boldsymbol{X}, \boldsymbol{w}, \beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\boldsymbol{w}^T\phi(\boldsymbol{x}_n), \beta^{-1}) \\ &= \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi) - \beta E_D(\boldsymbol{w}) \end{aligned}$$

where:

$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\boldsymbol{w}^T\phi(\boldsymbol{x}_n) - t_n\right)^2.$$

Minimizing $E_D(\boldsymbol{w})$ w.r.t $\boldsymbol{w}$ corresponds to the ML solution, assuming Gaussian distribution for ϵ. How does this link with the least squares solution?

Maximum Likelihood

Setting $\frac{\partial \ln p(\boldsymbol{t}|\boldsymbol{X},\boldsymbol{w},\beta)}{\partial \boldsymbol{w}} = 0$, we get

$$\boldsymbol{w}_{ML} = \left(\Phi^T \Phi\right)^{-1} \Phi^T \boldsymbol{t} = \Phi^\dagger \boldsymbol{t}$$

where $\Phi^\dagger = (\Phi^T \Phi)^{-1} \Phi^T$ is the *pseudo-inverse* of the matrix Φ.

There exist two versions of $\Phi^\dagger$:

- $\Phi^\dagger = (\Phi^T \Phi)^{-1} \Phi^T$ requiring the inversion of $(\Phi^T \Phi) \in \mathbb{R}^{M \times M}$
- $\Phi^\dagger = \Phi(\Phi \Phi^T)^{-1}$ requiring the inversion of $(\Phi \Phi^T) \in \mathbb{R}^{N \times N}$

We can choose one of them, depending on the values of N and M.

Maximum Likelihood

Interpretation of w_0 parameter:

$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(t_n - w_0 - \sum_{j=1}^{M-1} w_j\phi_j(\boldsymbol{x}_n)\right)^2$$

Setting $\frac{\partial E_D(\boldsymbol{w})}{\partial w_0} = 0$:

$$w_0 = \bar{t} - \sum_{j=1}^{M-1} w_j\bar{\phi}_j$$

where:

$$\bar{t} = \frac{1}{N}\sum_{n=1}^{N} t_n \qquad \text{and} \qquad \bar{\phi}_j = \frac{1}{N}\sum_{n=1}^{N}\phi_j(\boldsymbol{x}_n).$$

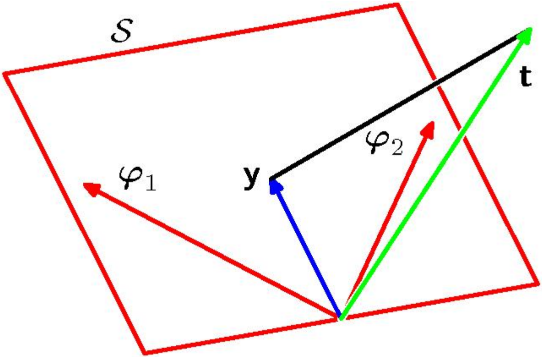
Maximum Likelihood

Interpretation of β parameter:

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

The inverse of the noise precision expresses the residual variance of the target values around the regression function.

Least Squares: Geometric interpretation



Linear Regression: Sequential updates

We obtain a sequential (or *on-line*) learning algorithm for updating $\boldsymbol{w}$ by applying **stochastic gradient descent (SGD)**:

- If the error function has the form $E(\boldsymbol{w}) = \sum_n E_n(\boldsymbol{w})$ then:

$$\boldsymbol{w}^{(\tau+1)} = \boldsymbol{w}^{(\tau)} - \eta \nabla E_n$$

  where τ denotes the iteration number, η is a learning rate parameter and ∇ is the gradient operator.

- For the least-squares error case:

$$\boldsymbol{w}^{(\tau+1)} = \boldsymbol{w}^{(\tau)} + \eta \left(t_n - \boldsymbol{w}^{(\tau)T} \phi(\boldsymbol{x}_n) \right) \phi(\boldsymbol{x}_n).$$

  The value of η needs to be chosen appropriately to ensure convergence of the algorithm.