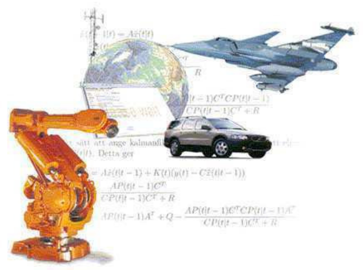
Machine Learning, Lecture 2
Linear Regression

"it is our firm belief that an understanding of linear models is essential for understanding nonlinear ones"



Thomas Schön

Division of Automatic Control
Linköping University
Linköping, Sweden.

Email: schon@isy.liu.se,
Phone: 013 - 281373,
Office: House B, Entrance 27.

Outline lecture 2

1. Summary of lecture 1
2. Linear basis function models
3. Maximum likelihood and least squares
4. Bias variance trade-off
5. Shrinkage methods
   - Ridge regression
   - LASSO
6. Bayesian linear regression
7. Motivation of kernel methods

Summary of lecture 1 (I/III)

The **exponential family** of distributions over x, parameterized by η,

$$p(x \mid \eta) = h(x)g(\eta)\exp\left(\eta^T u(x)\right)$$

One important member is the Gaussian density, which is commonly used as a building block in more sophisticated models. Important basic properties were provided.

The idea underlying **maximum likelihood** is that the parameters θ should be chosen in such a way that the measurements $\{x_i\}_{i=1}^N$ are as likely as possible, i.e.,

$$\widehat{\theta} = \arg\max_{\theta} p(x_1, \cdots, x_N \mid \theta).$$

Summary of lecture 1 (II/III)

The three basic steps of Bayesian modeling (where all variables are modeled as stochastic)

1. Assign **prior** distributions $p(\theta)$ to all unknown parameters θ.
2. Write down the **likelihood** $p(x_1, \ldots, x_N \mid \theta)$ of the data $x_1, \ldots, x_N$ given the parameters θ.
3. Determine the **posterior** distribution of the parameters given the data

$$p(\theta \mid x_1, \ldots, x_N) = \frac{p(x_1, \ldots, x_N \mid \theta)p(\theta)}{p(x_1, \ldots, x_N)} \propto p(x_1, \ldots, x_N \mid \theta)p(\theta)$$

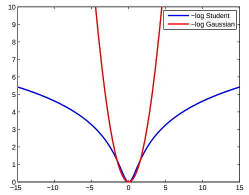
If the posterior $p(\theta \mid x_1, \ldots, x_N)$ and the prior $p(\theta)$ distributions are of the same functional form they are **conjugate distributions** and the prior is said to be a **conjugate prior** for the likelihood.

Summary of lecture 1 (III/III)

Modeling "heavy tails" using the Student's t-distribution

$$\mathrm{St}(x \mid \mu, \lambda, \nu) = \int \mathcal{N}\left(x \mid \mu, (\eta\lambda)^{-1}\right) \mathrm{Gam}\,(\eta \mid \nu/2, \nu/2)\, d\eta$$
$$= \frac{\Gamma(\nu/2 + 1/2)}{\Gamma(\nu/2)} \left(\frac{\lambda}{\pi\nu}\right)^{\frac{1}{2}} \left(1 + \frac{\lambda(x-\mu)^2}{\nu}\right)^{-\frac{\nu}{2}-\frac{1}{2}}$$

which according to the first expressions can be interpreted as an infinite mix of Gaussians with the same mean, but different variance.



Poor robustness is due to an unrealistic model, the ML estimator is inherently robust, provided we have the correct model.

Commonly used basis functions

In using nonlinear basis functions, $y(x, w)$ can be a nonlinear function in the input variable x (still linear in w).

- Global (in the sense that a small change in x affects all basis functions) basis function
  1. Polynomial (see illustrative example in Section 1.1) (ex. identity $\phi(x) = x$)
- Local (in the sense that a small change in x only affects the nearby basis functions) basis function
  1. Gaussian
  2. Sigmoidal

Linear regression model on matrix form

It is commonly convenient to write the linear regression model

$$t_n = w^T \phi(x_n) + \epsilon_n, \qquad n = 1, \dots, N,$$

where $w = \begin{pmatrix} w_0 & w_1 & \dots & w_{M-1} \end{pmatrix}^T$ and $\phi = \begin{pmatrix} 1 & \phi_1(x_n) & \dots & \phi_{M-1}(x_n) \end{pmatrix}^T$ on matrix form

$$T = \Phi w + E,$$

where

$$T = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix} \quad \Phi = \begin{pmatrix} \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_{M-1}(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \dots & \phi_{M-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \dots & \phi_{M-1}(x_N) \end{pmatrix} \quad E = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_N \end{pmatrix}$$

Maximum likelihood and least squares (I/IV)

In our linear regression model,

$$t_n = w^T \phi(x_n) + \epsilon_n,$$

assume that $\epsilon_n \sim \mathcal{N}(0, \beta^{-1})$ (i.i.d.). This results in the following likelihood function

$$p(t_n \mid w, \beta) = \mathcal{N}(w^T \phi(x_n), \beta^{-1})$$

Note that this is a slight abuse of notation, $p_{w,\beta}(t_n)$ or $p(t_n; w, \beta)$ would have been better, since w and β are both considered deterministic parameters in ML.

Maximum likelihood and least squares (II/IV)

The available training data consisting of N input variables $X = \{x_i\}_{i=1}^N$ and the corresponding target variables $T = \{t_i\}_{i=1}^N$.

According to our assumption on the noise, the likelihood function is given by

$$p(T \mid w, \beta) = \prod_{n=1}^{N} p(t_n \mid w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \mid w^T \phi(x_n), \beta^{-1})$$

which results in the following log-likelihood function

$$L(w,\beta) \triangleq \ln p(t_1, \dots, t_n \mid w, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n \mid w^T \phi(x_n), \beta^{-1})$$

$$= \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta \sum_{n=1}^{N} (t_n - w^T \phi(x_n))^2$$

Maximum likelihood and least squares (III/IV)

The maximum likelihood problem now amounts to solving

$$\arg\max_{w,\beta} L(w, \beta)$$

Setting the derivative $\frac{\partial L}{\partial w} = 2\beta \sum_{n=1}^{N} (t_n - w^T \phi(x_n))\phi(x_n)^T$ equal to 0 gives the following ML estimate for w

$$\widehat{w}^{\text{ML}} = \underbrace{(\Phi^T \Phi)^{-1} \Phi^T}_{\Phi^\dagger} T,$$

$$\Phi = \begin{pmatrix} \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_{M-1}(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \dots & \phi_{M-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \dots & \phi_{M-1}(x_N) \end{pmatrix}$$

Note that if $\Phi^T \Phi$ is singular (or close to) we can fix this by adding λI, i.e.,

$$\widehat{w}^{\text{RR}} = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T T,$$

Maximum likelihood and least squares (IV/IV)

Maximizing the log-likelihood function $L(w, \beta)$ w.r.t. β results in the following estimate for β

$$\frac{1}{\widehat{\beta}^{\text{ML}}} = \frac{1}{N} \sum_{n=1}^{N} \left(t_n - \widehat{w}^{\text{ML}} \phi(x_n)\right)^2$$

Finally, note that if we are only interested in w, the log-likelihood function is proportional to

$$\sum_{n=1}^{N} (t_n - w^T \phi(x_n))^2,$$

which clearly shows that assuming a Gaussian noise model and making use of Maximum Likelihood (ML) corresponds to a Least Squares (LS) problem.

Interpretation of the Gauss-Markov theorem

The least squares estimator has the smallest mean square error (MSE) of all linear estimators with no bias, **BUT** there may exist a biased estimator with lower MSE.

"the restriction to unbiased estimates is not necessarily a wise one."

[HTF, page 51]

Two classes of potentially biased estimators, 1. Subset selection methods and 2. Shrinkage methods.

This is intimately connected to the bias-variance trade-off

- We will give a system identification example related to ridge regression to illustrate the bias-variance trade-off.
- See Section 3.2 for a slightly more abstract (but very informative) account of the bias-variance trade-off. (this is a perfect topic for discussions during the exercise sessions!)

Interpretation of RR using the SVD of Φ

By studying the SVD of Φ it can be shown that ridge regression projects the measurements onto the principal components of Φ and then shrinks the coefficients of low-variance components more than the coefficients of high-variance components.

(See Section 3.4.1. in HTF for details.)

Bias-variance tradeoff – example (I/IV)

(Ex. 2.3 in Henrik Ohlsson's PhD thesis) Consider a SISO system

$$y_t = \sum_{k=1}^{n} g_k^0 u_{t-k} + e_t, \tag{1}$$

where u_t denotes the input, y_t denotes the output, e_t denotes white noise ($\mathbf{E}\,(e) = 0$ and $\mathbf{E}\,(e_t e_s) = \sigma^2 \delta(t-s)$) and $\{g_k^0\}_{k=1}^n$ denote the impulse response of the system.

Recall that the *impulse response* is the output y_t when $u_t = \delta(t)$ is used in (1), which results in

$$y_t = \begin{cases} g_t^0 + e_t & t = 1, \dots, n, \\ e_t & t > n. \end{cases}$$

Bias-variance tradeoff – example (II/IV)

The task is now to estimate the impulse response using an n^{th} order FIR model,

$$y_t = w^T \phi_t + e_t,$$

where

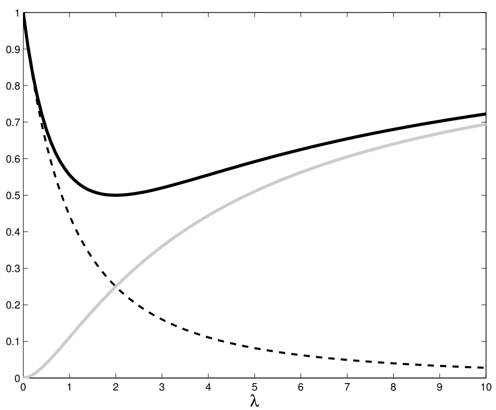
$$\phi_t = \begin{pmatrix} u_{t-1} & \dots & u_{t-n} \end{pmatrix}^T, \qquad w \in \mathbb{R}^n$$

Let us use Ridge Regression (RR),

$$\widehat{w}^{\text{RR}} = \arg\min_w \|Y - \Phi w\|_2^2 + \lambda w^T w.$$

to find the parameters w.

Bias-variance tradeoff – example (III/IV)



Squared bias (gray line)

$$\left(\mathbf{E}_{\widehat{w}}\left(\widehat{w}^T \phi_*\right) - w_0^T \phi_*\right)^2$$

Variance (dashed line)

$$\mathbf{E}_{\widehat{w}}\left(\left(\mathbf{E}_{\widehat{w}}\left(\widehat{w}^T \phi_*\right) - \widehat{w}^T \phi_*\right)^2\right)$$

MSE (black line)

MSE = (bias)^2 + variance

Bias-variance tradeoff – example (IV/IV)

"Flexible" models will have a low bias and high variance and more "restricted" models will have high bias and low variance.

The model with the best predictive capabilities is the one which strikes the best tradeoff between bias and variance.

Recent contributions on impulse response identification using regularization, see

- Gianluigi Pillonetto and Giuseppe De Nicolao. **A new kernel-based approach for linear system identification**. *Automatica*, 46(1):81–93, January 2010.
- Tianshi Chen, Henrik Ohlsson and Lennart Ljung. **On the estimation of transfer functions, regularizations and Gaussian processes – Revisited**. *Automatica*, 48(8): 1525–1535, August 2012.

Lasso

The Lasso was introduced during lecture 1 as the MAP estimate when a **Laplacian prior** is assigned to the parameters. Alternatively we can motivate the Lasso as the solution to

$$\begin{aligned} \min_{w} \quad & \sum_{n=1}^{N} \left(t_n - w^T\phi(x_n)\right)^2 \\ \text{s.t.} \quad & \sum_{j=0}^{M-1} |w_j| \leq \eta \end{aligned}$$

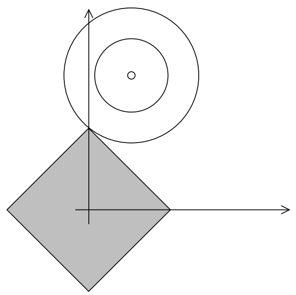
which using a Lagrange multiplier λ can be stated

$$\min_{w} \sum_{n=1}^{N} \left(t_n - w^T\phi(x_n)\right)^2 + \lambda \sum_{j=0}^{M-1} |w_j|$$

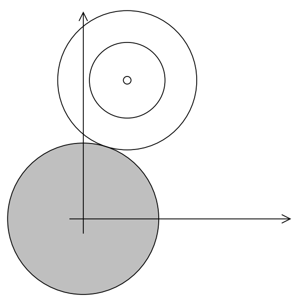
The difference to ridge regression is simply that Lasso make use of the ℓ_1-norm $\sum_{j=0}^{M-1} |w_j|$, rather than the ℓ_2-norm $\sum_{j=0}^{M-1} w_j^2$ used in ridge regression in shrinking the parameters.

Graphical illustration of Lasso and RR

Lasso



Ridge Regression (RR)



The circles are contours of the least squares cost function (LS estimate in the middle). The constraint regions are shown in gray $|w_0| + |w_1| \leq \eta$ (Lasso) and $w_0^2 + w_1^2 \leq \eta$ (RR). The shape of the constraints motivates why Lasso often leads to **sparseness**.

Implementing Lasso

The ℓ_1-regularized least squares problem (lasso)

$$\min_{w} \|T - \Phi w\|_2^2 + \lambda \|w\|_1 \qquad (2)$$

YALMIP code solving (2). Download: http://users.isy.liu.se/johanl/yalmip/

```
w=sdpvar(M,1);
ops=sdpsettings('verbose',0);
solvesdp([],(T-Phi*w)'*(T-Phi*w) + lambda*norm(w,1),ops)
```

CVX code solving (2). Download: http://cvxr.com/cvx/

```
cvx_begin
variable w(M)
minimize((T-Phi*w)'*(y-Phi*w) + lambda*norm(w,1))
cvx_end
```

A MATLAB package dedicated to ℓ_1-regularized least squares problems is `l1_ls`. Download: http://www.stanford.edu/~boyd/l1_ls/

Bayesian linear regression – example (I/VI)

Consider the problem of fitting a straight line to noisy measurements. Let the model be ($t \in \mathcal{R}, x_n \in \mathcal{R}$)

$$t_n = \underbrace{w_0 + w_1 x_n}_{y(x,w)} + \epsilon_n, \qquad n = 1, \dots, N. \tag{3}$$

where

$$\epsilon_n \sim \mathcal{N}(0, 0.2^2), \qquad \beta = \frac{1}{0.2^2} = 25.$$

According to (3), the following identity basis function is used

$$\phi_0(x_n) = 1, \qquad \phi_1(x_n) = x_n.$$

The example lives in two dimensions, allowing us to plot the distributions in illustrating the inference.

Bayesian linear regression – example (II/VI)

Let the true values for w be $w^\star = \begin{pmatrix} -0.3 & 0.5 \end{pmatrix}^T$ (plotted using a white circle below).

Generate synthetic measurements by

$$t_n = w_0^\star + w_1^\star x_n + \epsilon_n, \qquad \epsilon_n \sim \mathcal{N}(0, 0.2^2),$$

where $x_n \sim \mathcal{U}(-1, 1)$.

Furthermore, let the prior be

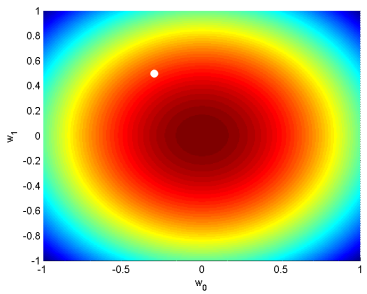
$$p(w) = \mathcal{N}\left(w \mid \begin{pmatrix} 0 & 0 \end{pmatrix}^T, \alpha^{-1} I\right),$$

where

$$\alpha = 2.$$

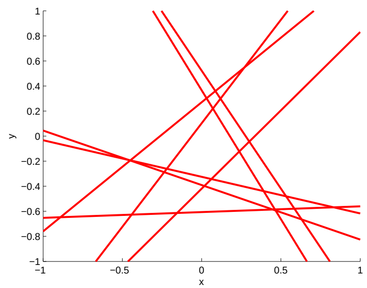
Bayesian linear regression – example (III/VI)

Plot of the situation before any data arrives.



Prior,

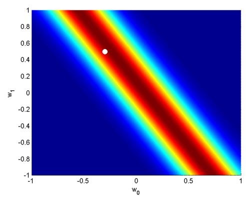
$$p(w) = \mathcal{N}\left(w \mid \begin{pmatrix} 0 & 0 \end{pmatrix}^T, \frac{1}{2} I\right)$$



Example of a few realizations from the posterior.

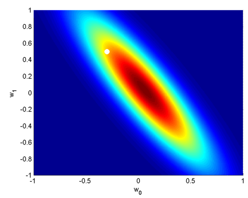
Bayesian linear regression – example (IV/VI)

Plot of the situation after one measurement has arrived.



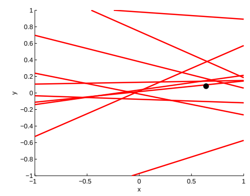
Likelihood (plotted as a function of w)

$$p(t_1 \mid w) = \mathcal{N}(t_1 \mid w_0 + w_1 x_1, \beta^{-1})$$



Posterior/prior,

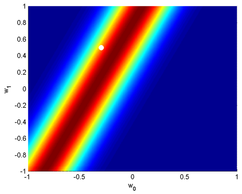
$$p(w \mid t_1) = \mathcal{N}(w \mid m_1, S_1),$$
$$m_1 = \beta S_1 \Phi^T t_1,$$
$$S_1 = (\alpha I + \beta \Phi^T \Phi)^{-1}.$$



Example of a few realizations from the posterior and the first measurement (black circle).

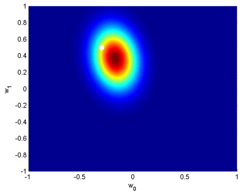
Bayesian Linear Regression - Example (V/VI)

Plot of the situation after two measurements have arrived.



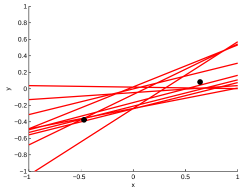
Likelihood (plotted as a function of w)

$$p(t_2 \mid w) = \mathcal{N}(t_2 \mid w_0 + w_1 x_2, \beta^{-1})$$



Posterior/prior,

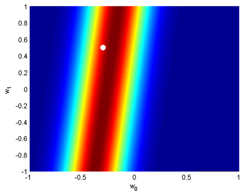
$$\begin{aligned} p(w \mid T) &= \mathcal{N}(w \mid m_2, S_2), \\ m_2 &= \beta S_2 \Phi^T T, \\ S_2 &= (\alpha I + \beta \Phi^T \Phi)^{-1}. \end{aligned}$$



Example of a few realizations from the posterior and the measurements (black circles).

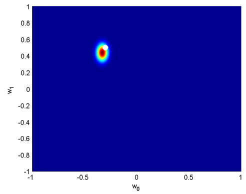
Bayesian linear regression – example (VI/VI)

Plot of the situation after 30 measurements have arrived.



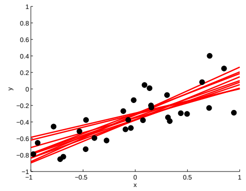
Likelihood (plotted as a function of w)

$$p(t_{30} \mid w) = \mathcal{N}(t_{30} \mid w_0 + w_1 x_{30}, \beta^{-1})$$



Posterior/prior,

$$\begin{aligned} p(w \mid T) &= \mathcal{N}(w \mid m_{30}, S_{30}), \\ m_{30} &= \beta S_{30} \Phi^T T, \\ S_{30} &= (\alpha I + \beta \Phi^T \Phi)^{-1}. \end{aligned}$$



Example of a few realizations from the posterior and the measurements (black circles).

Empirical Bayes

Important question: How do we decide on the suitable values for hyperparameters η?

Idea: Estimate the hyperparameters from the data by selecting them such that they maximize the marginal likelihood function,

$$p(T \mid \eta) = \int p(T \mid w, \eta) p(w \mid \eta) \mathrm{d}w,$$

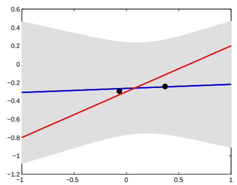
where η denotes the hyperparameters to be estimated.

Travels under many names, besides empirical Bayes, this is also referred to as *type 2 maximum likelihood*, *generalized maximum likelihood*, and *evidence approximation*.

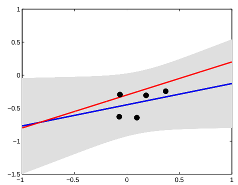
Empirical Bayes **combines** the two statistical philosophies; frequentistic ideas are used to estimate the hyperparameters that are then used within the Bayesian inference.

Predictive distribution – example

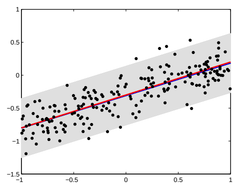
Investigating the predictive distribution for the example above



$N = 2$ observations



$N = 5$ observations



$N = 200$ observations

- True system ($y(x) = -0.3 + 0.5x$) generating the data (red line)
- Mean of the predictive distribution (blue line)
- One standard deviation of the predictive distribution (gray shaded area) Note that this is the *point-wise predictive standard deviation* as a function of x.
- Observations (black circles)

Posterior distribution

Recall that the posterior distribution is given by

$$p(w \mid T) = \mathcal{N}(w \mid m_N, S_N),$$

where

$$m_N = \beta S_N \Phi^T T,$$
$$S_N = (\alpha I + \beta \Phi^T \Phi)^{-1}.$$

Let us now investigate the posterior mean solution m_N, which has an interpretation that directly leads to the kernel methods (lecture 5), including Gaussian processes.

A few concepts to summarize lecture 2

Linear regression: Models the relationship between a continuous target variable t and a possibly nonlinear function $\phi(x)$ of the input variables.

Hyperparameter: A parameter of the prior distribution that controls the distribution of the parameters of the model.

Maximum a Posteriori (MAP): A point estimate obtained by maximizing the posterior distribution. Corresponds to a mode of the posterior distribution.

Gauss Markov theorem: States that in a linear regression model, the best (in the sense of minimum MSE) linear unbiased estimate (BLUE) is given by the least squares estimate.

Ridge regression: An ℓ_2-regularized least squares problem used to solve the linear regression problem resulting in potentially biased estimates. A.k.a. Tikhonov regularization.

Lasso: An ℓ_1-regularized least squares problem used to solve the linear regression problem resulting in potentially biased estimates. The Lasso typically produce sparse estimates.