PDEEC – Machine Learning 2018/19

Lecture 2 - Regression Notes

Jaime S. Cardoso
jaime.cardoso@inesctec.pt

INESC TEC and Faculdade Engenharia, Universidade do Porto

Oct. 04, 2018

Regression notes

Assumptions and notation

Consider the problem of finding a homogeneous real-valued linear function

$$g(\mathbf{x}) =< \mathbf{w}, \mathbf{x} >= \mathbf{w}^t\mathbf{x} = \sum_{i=1}^{D} w_i x_i$$

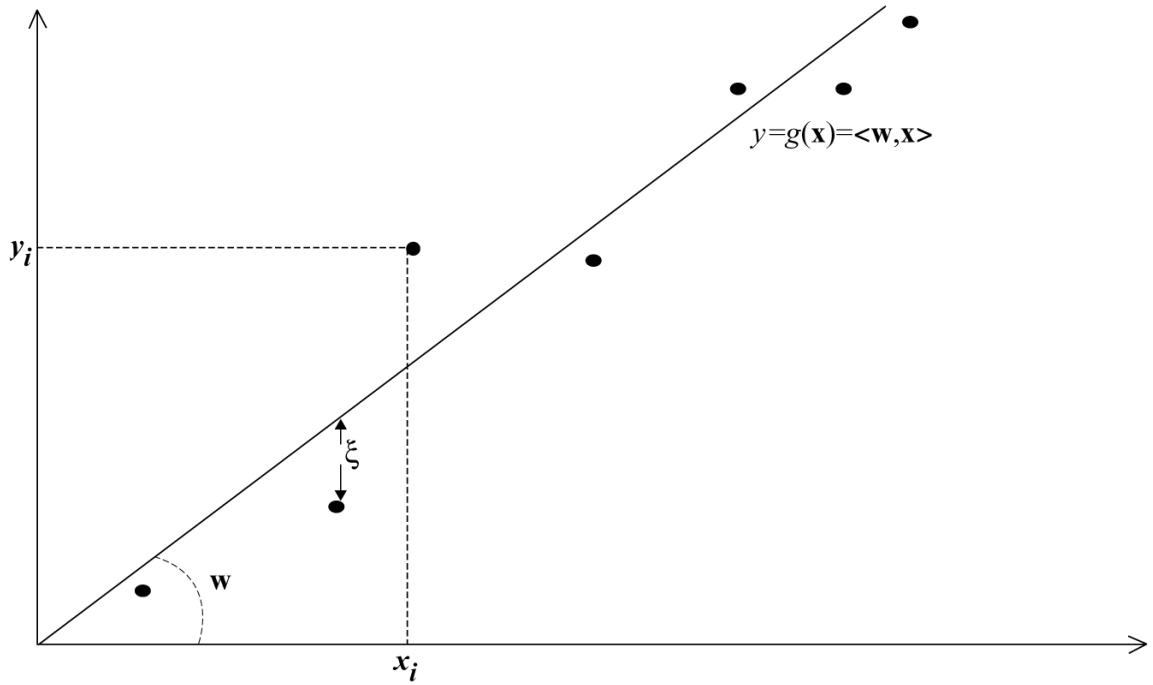
that best interpolates a given training set

$$S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)\}$$

Here, we use the notation $\mathbf{x} = (x_1, x_2, \ldots, x_D)$ for the D-dimensional input vectors, while $\mathbf{w}'$ denotes the transpose of the vector $\mathbf{w} \in \mathbf{R}^D$.
We will use $\mathbf{X}$ to denote the matrix whose rows are the row vectors $\mathbf{x}'_1, \mathbf{x}'_2, \ldots, \mathbf{x}'_N$ and $\mathbf{y}$ to denote the vector $(y_1, \ldots, y_N)$.

Regression notes

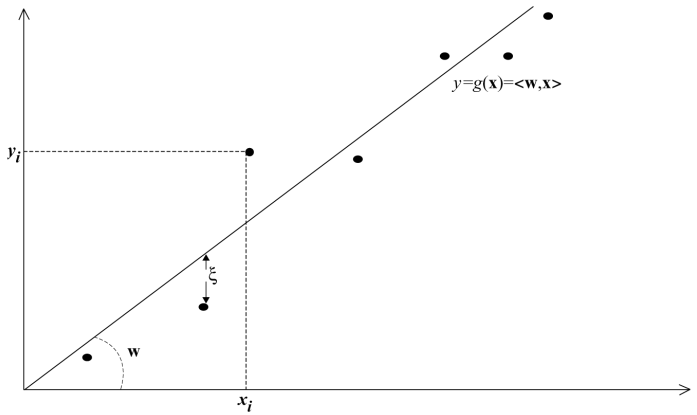
Assumptions and notation



Remark [Row versus column vectors] Note that our inputs are column vectors but they are stored in the matrix $\mathbf{X}$ as row vectors. We adopt this convention to be consistent with the typical representation of data in an input file and in our Matlab code, while preserving the standard vector representation.

Regression notes

Assumptions and notation



The distance shown as ξ in the figure is the error of the linear function on the particular training example, $\xi = (y - g(\mathbf{x}))$.
We would like to find a function for which all of these training errors are small. The sum of the squares of these errors is the most commonly chosen measure of the collective discrepancy between the training data and a particular function

$$\mathcal{L}(g, S) = \mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_i - g(\mathbf{x}_i))^2 = \frac{1}{2}\sum_{n=1}^{N}\xi_n^2$$

Regression notes

The Least-Mean-Square (LMS) method

The learning problem now becomes that of choosing the vector $\mathbf{w}$ that minimises the collective loss.

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_n - g(\mathbf{x}_n))^2$$

- Consider a gradient descent algorithm:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \alpha \frac{\partial \mathcal{L}}{\partial \mathbf{w}}|_t$$

Regression notes

The Least-Mean-Square (LMS) method

- Now we have the following descent rule:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \alpha \sum_n (y_n - \mathbf{w}^t \mathbf{x}_n)\mathbf{x}_n$$

$$w_j^{(t+1)} = w_j^{(t)} + \alpha \sum_n (y_n - \mathbf{w}^t \mathbf{x}_n) x_{n,j} \quad j = 1, \ldots, D$$

- For a single training point $\mathbf{x}_i$, we have:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \alpha(y_i - \mathbf{w}^t \mathbf{x}_i)\mathbf{x}_i \tag{1}$$

$$w_j^{(t+1)} = w_j^{(t)} + \alpha(y_i - \mathbf{w}^t \mathbf{x}_i) x_{i,j} \quad j = 1, \ldots, D$$

  - This is known as the LMS update rule, or the Widrow-Hoff learning rule
  - This is actually a "stochastic" descent algorithm
  - This can be used as a **on-line** algorithm

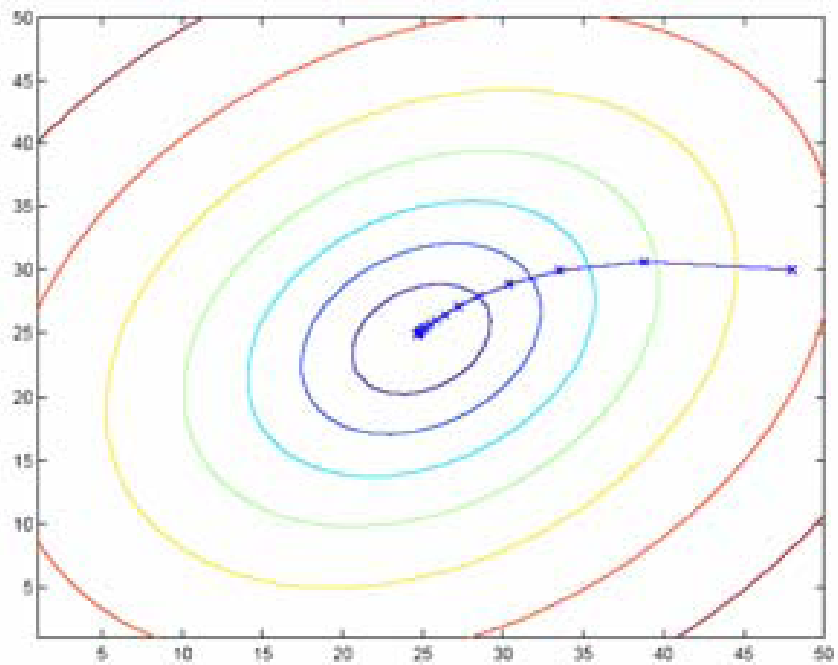
Regression notes

The Least-Mean-Square (LMS) method

- Steepest descent

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \alpha \sum_n (y_n - \mathbf{w}^t \mathbf{x}_n)\mathbf{x}_n \tag{2}$$

- This is as a batch gradient descent algorithm



Regression notes

The normal equations

- Write the cost function in matrix form:

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_i - g(\mathbf{x}_i))^2$$

$$= \frac{1}{2}(\mathbf{X}\mathbf{w}-\mathbf{y})^t(\mathbf{X}\mathbf{w}-\mathbf{y}) = \frac{1}{2}(\mathbf{w}^t\mathbf{X}^t\mathbf{X}\mathbf{w}-\mathbf{w}^t\mathbf{X}^t\mathbf{y}-\mathbf{y}^t\mathbf{X}\mathbf{w}+\mathbf{y}^t\mathbf{y})$$

- To minimize L($\mathbf{w}$, S), take derivative and set to zero

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = -\mathbf{X}^t\mathbf{y} + \mathbf{X}^t\mathbf{X}\mathbf{w} = 0$$

hence obtaining the so-called 'normal equations'

$$\mathbf{X}^t\mathbf{X}\mathbf{w} = \mathbf{X}^t\mathbf{y}$$

- If the inverse of $\mathbf{X}^t\mathbf{X}$ exists, the solution of the least squares problem can be expressed as

$$\mathbf{w} = (\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{y}$$

Regression notes

A recap:

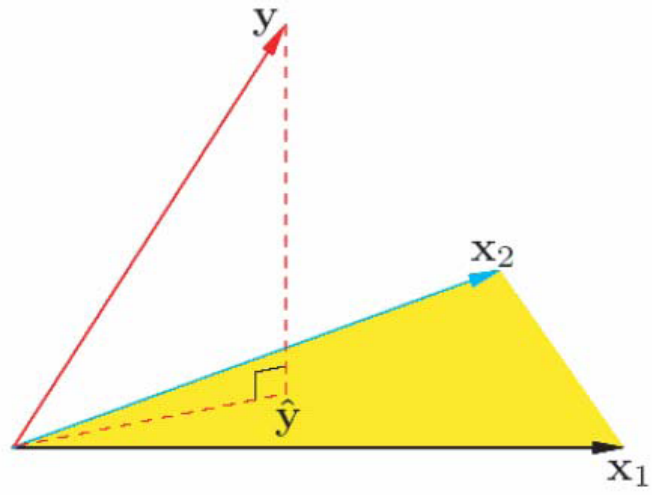
- LMS update rule
  - $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \alpha(y_i - \mathbf{w}^t \mathbf{x}_i)\mathbf{x}_i$
  - Pros: on-line, low per-step cost
  - Cons: coordinate, maybe slow-converging
- Steepest descent
  - $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \alpha \sum_n (y_n - \mathbf{w}^t \mathbf{x}_n)\mathbf{x}_n$
  - Pros: fast-converging, easy to implement
  - Cons: a batch,
- Normal equations
  - $\mathbf{w} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y}$
  - Pros: a single-shot algorithm! Easiest to implement.
  - Cons: need to compute pseudo-inverse $(\mathbf{X}^t \mathbf{X})^{-1}$, expensive, numerical issues (e.g., matrix is singular ..)

Regression notes

Geometric Interpretation of LMS

- The predictions on the training data are:

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = \mathbf{X}(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{y}$$



- Note that $\hat{\mathbf{y}} - \mathbf{y} = (\mathbf{X}(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t - \mathbf{I})\mathbf{y}$ and $\mathbf{X}^t(\hat{\mathbf{y}} - \mathbf{y}) = \mathbf{0}$

$\hat{\mathbf{y}}$ is the orthogonal projection of $\mathbf{y}$ into the space spanned by the columns of $\mathbf{X}$

Regression notes

Probabilistic Interpretation of LMS

- Let us assume that the target variable and the inputs are related by the equation: $y_n = \mathbf{w}^t \mathbf{x}_n + \epsilon_n$ where ϵ_n is an error term of unmodeled effects or random noise
- Now assume that ϵ_n follows a Gaussian $N(0, \sigma^2)$
  - That is to say that $y_n \sim N(\mathbf{w}^t \mathbf{x}_n, \sigma^2)$
- Then we have: $p(y_n|\mathbf{x}_n, \mathbf{w}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_n - \mathbf{w}^t \mathbf{x}_n)^2}{2\sigma^2}\right)$
- By independence assumption:

$$L(\mathbf{w}) =$$

$$= \prod_{n=1}^{N} p(y_n|\mathbf{x}_n, \mathbf{w}) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left(-\frac{\sum_{n=1}^{N}(y_n - \mathbf{w}^t \mathbf{x}_n)^2}{2\sigma^2}\right)$$

Regression notes

Probabilistic Interpretation of LMS

- Hence the log-likelihood is:

$$l(\mathbf{w}) = N \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{\sigma^2}\frac{1}{2}\sum_{n=0}^{N}(y_n - \mathbf{w}^t\mathbf{x}_n)^2$$

- Do you recognize the last term?
  Yes, it is $\mathcal{L}(\mathbf{w}) = \sum_{n=0}^{N}(y_n - \mathbf{w}^t\mathbf{x}_n)^2$
- Thus under independence assumption, LMS is equivalent to Maximum Likelihood Estimate of $\mathbf{w}$

Regression notes

Beyond basic LR

LR with non-linear basis functions

- LR does not mean we can only deal with linear relationships
- We are free to design (non-linear) features under LR

$$y = w_0 + \sum_{j=1}^{M} w_j h_j(\mathbf{x}) = \mathbf{w}^t \mathbf{h}(\mathbf{x})$$

  where the $h_j(\mathbf{x})$ are fixed basis functions (and we define $h_0(\mathbf{x}) = 1$).
- Example: polynomial regression: $\mathbf{h}(x) = [1, x, x^2, x^3]$
- We will be concerned with estimating (distributions over) the weights $\mathbf{w}$ and choosing the model order M.

Regression notes

Beyond basic LR

Basis functions

There are many basis functions, e.g.:

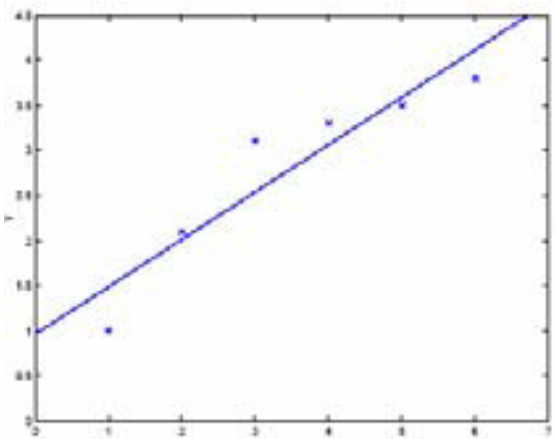
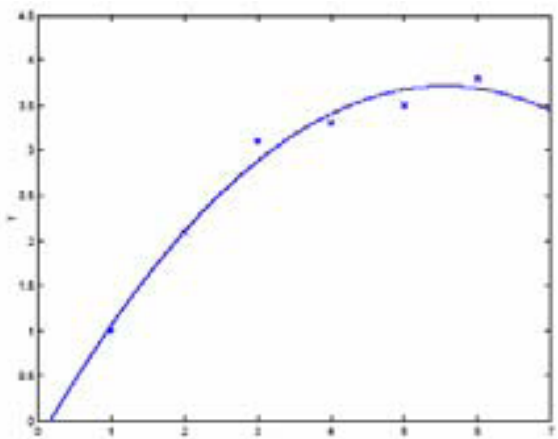
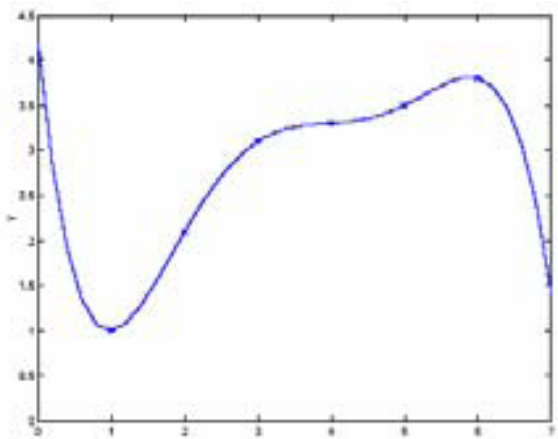
- Polynomial: $h_j(x) = x^{j-1}$
- Radial basis functions $h_j(x) = \exp -\frac{(x-\mu)^2}{2\sigma^2}$
- Sigmoidal
- Splines, Fourier, Wavelets, etc

Regression notes

Beyond basic LR

Regularization

- Overfitting and underfitting

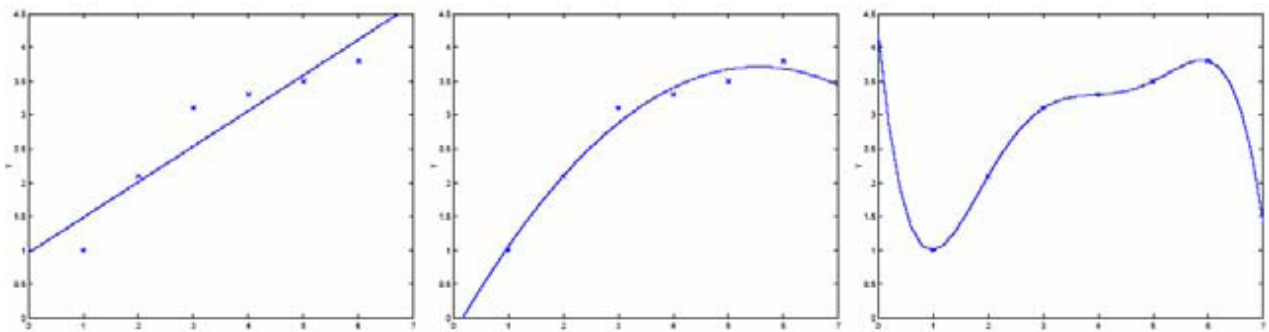
$y = w_0 + w_1 x$ $\quad$ $y = w_0 + w_1 x + w_2 x^2$ $\quad$ $y = \sum_{i=0}^{5} w_i x^i$

Regression notes

Beyond basic LR

Regularization

- we define the bias of a model to be the expected generalization error even if we were to train it to a very (say, infinitely) large training set.
- By fitting "spurious" patterns in the training set, we might again obtain a model with large generalization error. In this case, we say the model has large variance.



Regression notes

Beyond basic LR

Regularization

- there is not enough data to ensure that the matrix $\mathbf{X}^t\mathbf{X}$ is invertible, or
- there may be noise in the data making it unwise to try to match the target output exactly.
- ill-conditioned problem, since there is not enough information in the data to precisely specify the solution.
- In these situations an approach that is frequently adopted is to restrict the choice of functions in some way. Such a restriction or bias is referred to as regularisation.
  - introduce a regularisation term in the error function in order to control overfitting
  - one of the simplest forms of regularizer is given by the sum-of-squares of the weight vector elements: $\lambda \mathbf{w}^t\mathbf{w}$
  - The total error function takes the form

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^t\mathbf{x})^2 + \frac{1}{2}\lambda \mathbf{w}^t\mathbf{w}$$

Regression notes

Beyond basic LR

Regularization

- The total error function takes the form

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^t\mathbf{x})^2 + \frac{1}{2}\lambda\mathbf{w}^t\mathbf{w}$$

  corresponds to the Ridge Regression.
- λ is a positive real number that defines the relative tradeoff between norm and loss and hence controls the degree of regularization.
- The normal equation for ridge regression takes the form $(\mathbf{X}^t\mathbf{X} + \lambda I)\mathbf{w} = \mathbf{X}^t\mathbf{y}$.

Regression notes

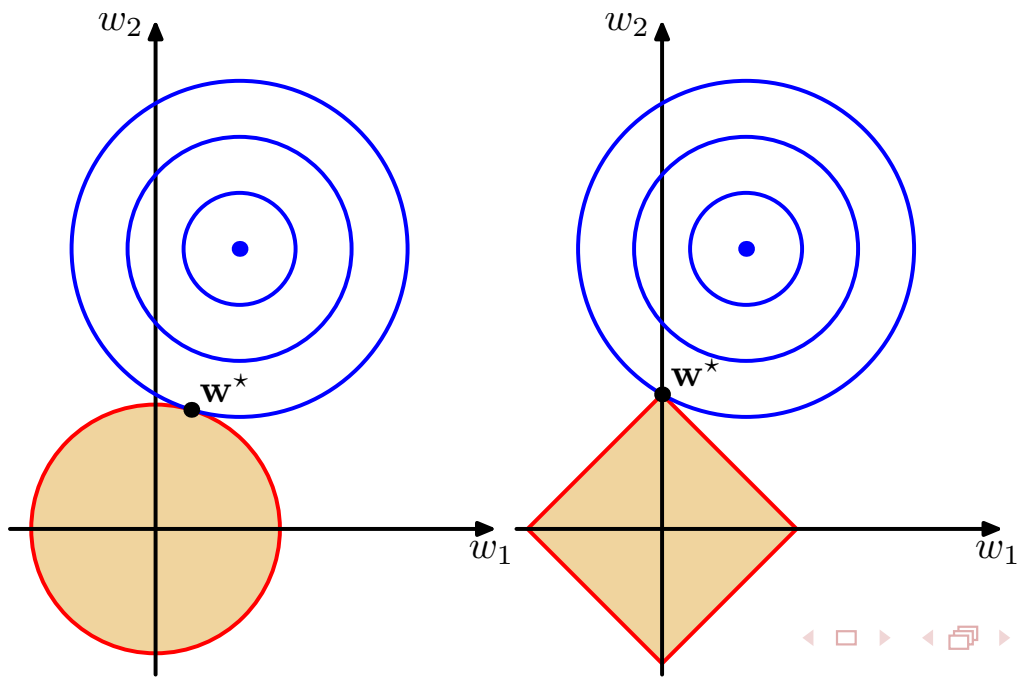
Beyond basic LR

Regularization

- A more general regularizer is sometimes used, for which the regularized error takes the form

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^t\mathbf{x})^2 + \frac{1}{2}\lambda\sum_{i=1}^{D}|w_i|^q$$

- The case $q = 1$ is known as the lasso regression



Regression notes

Probabilistic Interpretation of Ridge Regression

- Let us assume a prior distribution over the coefficients $\mathbf{w}$:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$$

- Using Bayes' theorem, the posterior distribution for $\mathbf{w}$ is proportional to the product of the prior distribution and the likelihood function

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w})$$

- We can now determine $\mathbf{w}$ by finding the most probable value of $\mathbf{w}$ given the data, in other words by maximizing the posterior distribution. This technique is called maximum posterior, or simply MAP.
- Taking the negative logarithm of the previous expression, we find that the maximization of the posterior corresponds to the minimization of the error given by Rigde Regression

Regression notes

Maximum A Posteriori Probability Estimation

- In Maximum Likelihood (ML) method, $\mathbf{w}$ was considered as a parameter
- Here we shall look at $\mathbf{w}$ as a random vector described by a pdf $p(\mathbf{w})$, assumed to be known
- Given

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^t \\ \mathbf{x}_2^t \\ \dots \\ \mathbf{x}_N^t \end{bmatrix}$$

  Compute the maximum of $p(\mathbf{w}|\mathbf{X})$
- From Bayes theorem $p(\mathbf{w}|\mathbf{X}) = \frac{p(\mathbf{X}|\mathbf{w})p(\mathbf{w})}{p(\mathbf{X})}$

Regression notes

Maximum A Posteriori Probability Estimation

- We can now determine $\mathbf{w}$ by finding the most probable value of $\mathbf{w}$ given the data, in other words by maximizing the posterior distribution. This technique is called maximum posterior, or simply MAP.
- $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(\mathbf{w}|\mathbf{X})$
- Taking the negative logarithm of the previous expression, we find that the maximization of the posterior corresponds to the minimization of the error given by Rigde Regression

Bayesian Linear Regression

Bayesian Linear Regression

- Better than estimating the a single value for $\mathbf{w}$ we can estimate the probability density for $\mathbf{w}$ after seeing the data: the posterior probability
- A common choice for the prior distribution over the coefficients is $p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \rho^2 \boldsymbol{I})$ (in Bishop $\alpha = 1/\rho^2$)
- Using Bayes' theorem, the posterior distribution for $\mathbf{w}$ is proportional to the product of the prior distribution and the likelihood function

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{X}, \mathbf{w}) p(\mathbf{w})$$

-

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \propto \prod_{n=1}^{N} p(y_n|\mathbf{x}_n, \mathbf{w}) p(\mathbf{w}) =$$
$$\left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left(-\frac{\sum_{n=0}^{N}(y_n - \mathbf{w}^t\mathbf{x}_n)^2}{2\sigma^2}\right) \frac{1}{\sqrt{2\pi}\rho} \exp\left(\frac{-\mathbf{w}^t\mathbf{w}}{2\rho^2}\right)$$

Bayesian Linear Regression

Bayesian Linear Regression

- $$p(\mathbf{w}|\mathbf{X},\mathbf{y}) \propto \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left(-\frac{(\mathbf{y}-\mathbf{X}\mathbf{w})^t(\mathbf{y}-\mathbf{X}\mathbf{w}))}{2\sigma^2}\right)\frac{1}{\sqrt{2\pi}\rho}\exp\left(\frac{-\mathbf{w}^t\mathbf{w}}{2\rho^2}\right)$$
- It is possible to show that

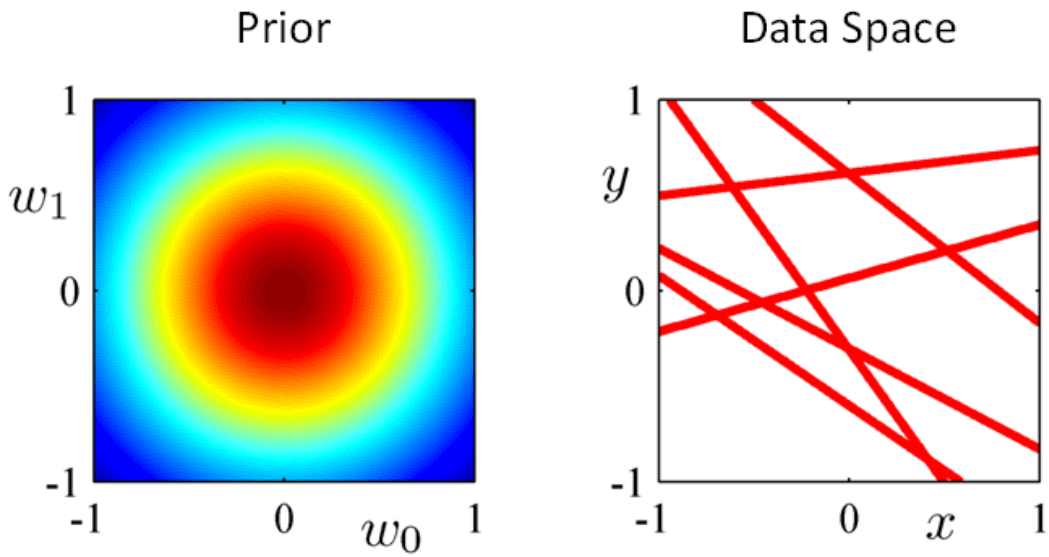
$$p(\mathbf{w}|\mathbf{X},\mathbf{y}) \sim N(\boldsymbol{\mu},\boldsymbol{\Sigma})$$

$$\boldsymbol{\Sigma}^{-1} = \frac{\boldsymbol{I}}{\rho^2} + \frac{\mathbf{X}^t\mathbf{X}}{\sigma^2}$$

$$\boldsymbol{\mu} = \frac{\boldsymbol{\Sigma}\mathbf{X}^t\mathbf{y}}{\sigma^2}$$

Online Bayesian Linear Regression

0 data points observed

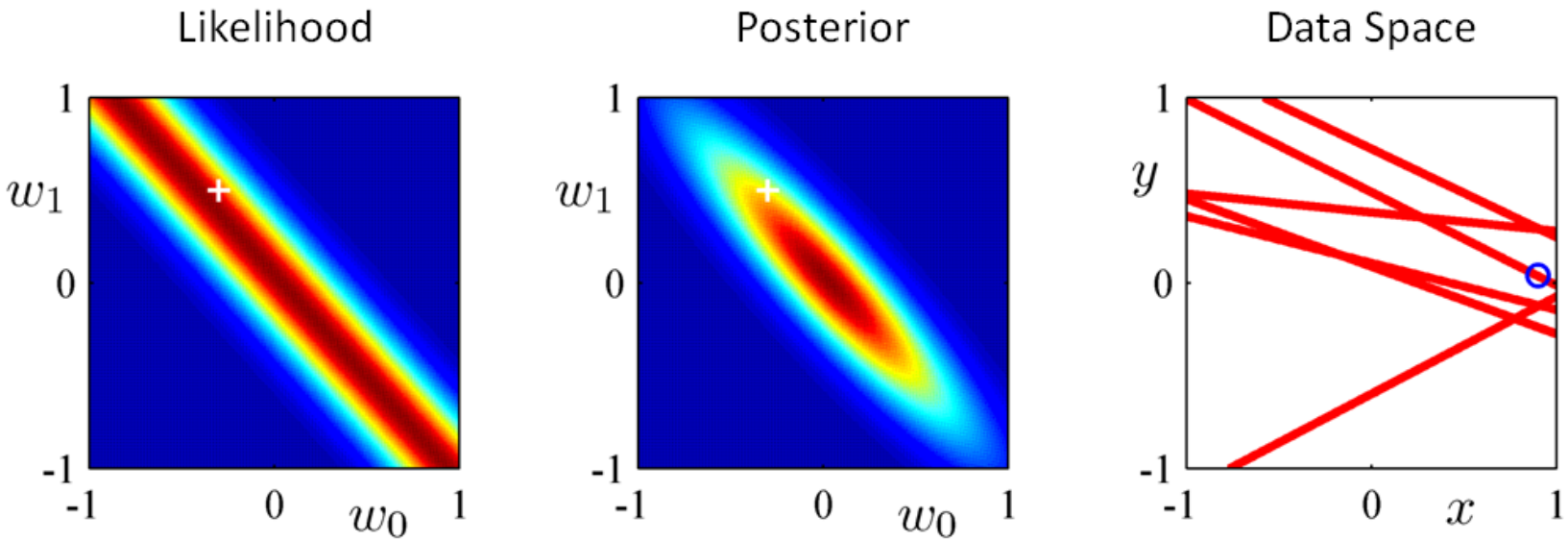


$$\boldsymbol{\Sigma}_0^{-1} = \frac{\boldsymbol{I}}{\rho^2}$$
$$\boldsymbol{\mu}_0 = 0$$

Regression notes

Online Bayesian Linear Regression

1 data point observed



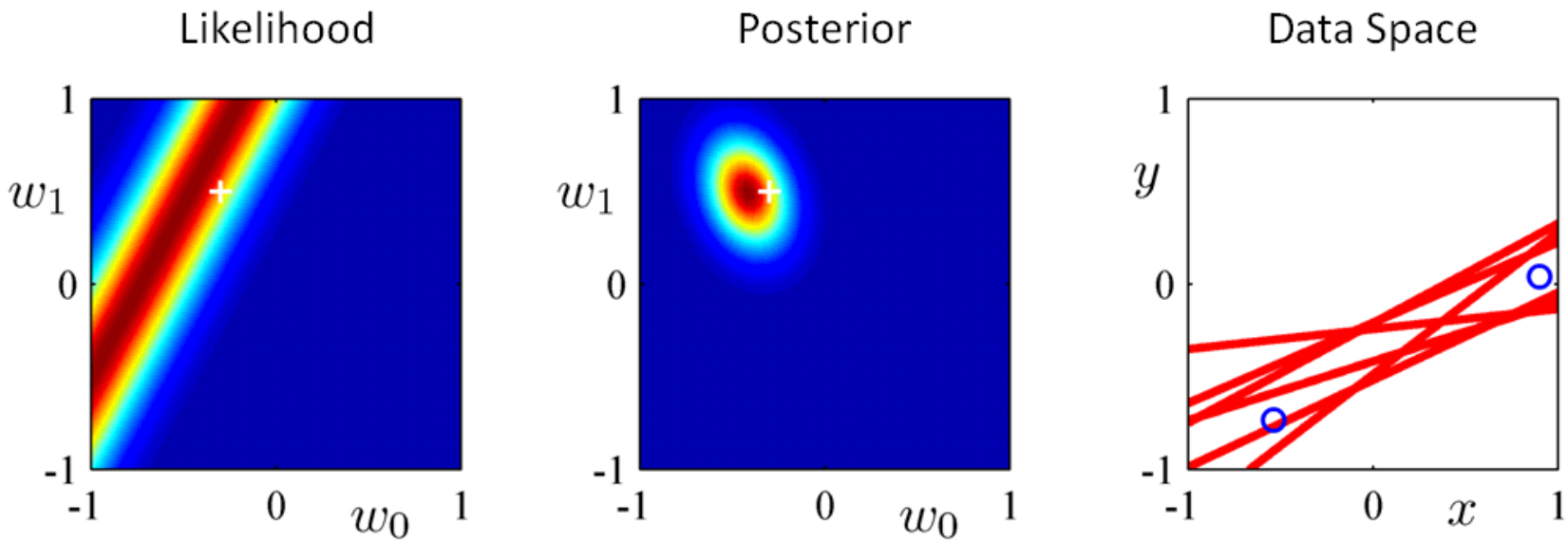
$$\boldsymbol{\Sigma}_1^{-1} = \boldsymbol{\Sigma}_0^{-1} + \frac{\mathbf{x}_1 \mathbf{x}_1^t}{\sigma^2}$$

$$\boldsymbol{\mu}_1 = \Sigma_1 \left(\Sigma_0^{-1} \boldsymbol{\mu}_0 + \frac{\mathbf{x}_1 y_1}{\sigma^2} \right)$$

Bayesian Linear Regression

Bayesian Linear Regression

2 data points observed

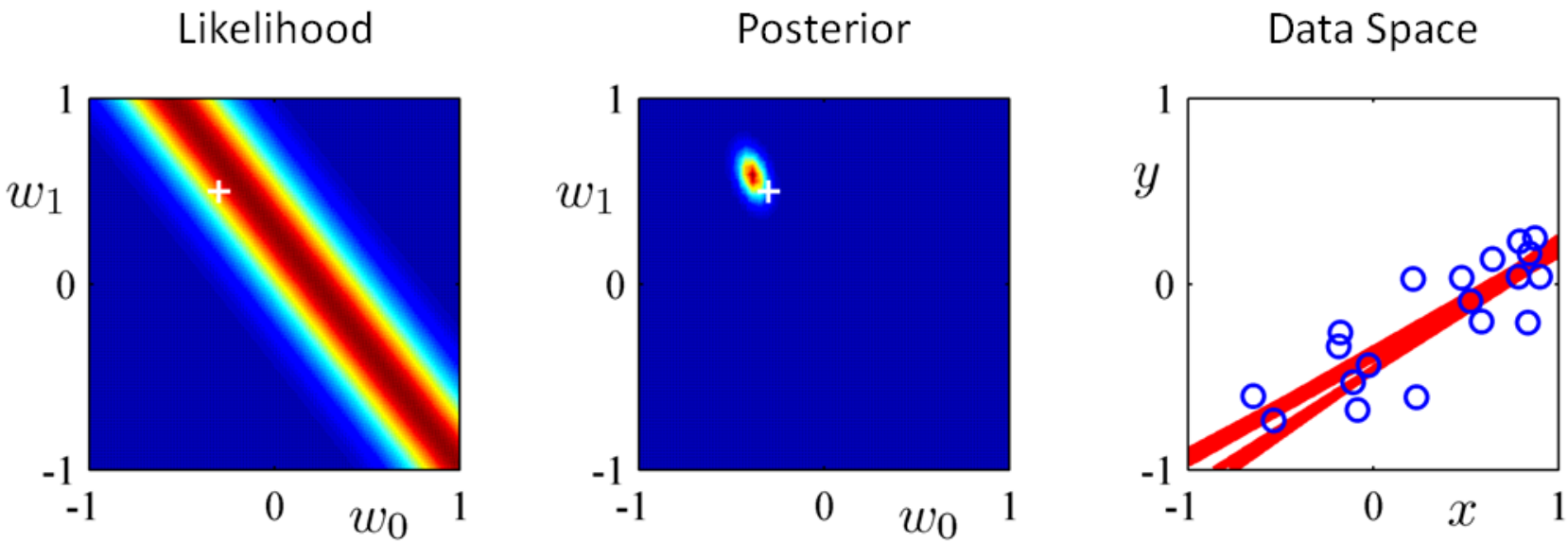


$$\boldsymbol{\Sigma}_2^{-1} = \boldsymbol{\Sigma}_1^{-1} + \frac{\mathbf{x}_2 \mathbf{x}_2^t}{\sigma^2}$$
$$\boldsymbol{\mu}_2 = \Sigma_2 \left(\Sigma_1^{-1} \boldsymbol{\mu}_1 + \frac{\mathbf{x}_2 y_2}{\sigma^2} \right)$$

Bayesian Linear Regression

Bayesian Linear Regression

20 data points observed



Bayesian Linear Regression

Predictive Distribution

- Predict y for new values of $\mathbf{x}$ by integrating over $\mathbf{w}$: $p(y) = \int p(y, \mathbf{w}) d\mathbf{w} = \int p(y|\mathbf{w}) p(\mathbf{w}) d\mathbf{w}$
- under the normality conditions assumed so far, this is also a normal distribution

Regression notes

Beyond basic LR

Dual Variables in Regression

- $(\mathbf{X}^t\mathbf{X} + \lambda I)\mathbf{w} = \mathbf{X}^t\mathbf{y}$
- $\mathbf{w} = \lambda^{-1}\mathbf{X}^t(\mathbf{y} - \mathbf{X}\mathbf{w}) = \mathbf{X}^t\boldsymbol{\alpha}$
- showing that $\mathbf{w}$ can be written as a linear combination of the training points, $\mathbf{w} = \sum_{n=1}^{N} = \alpha_n\mathbf{x}_n$, with $\boldsymbol{\alpha} = \lambda^{-1}(\mathbf{y} - \mathbf{X}\mathbf{w})$
- The elements of $\boldsymbol{\alpha}$ are called the dual variables.
- $$\begin{aligned} \boldsymbol{\alpha} &= \lambda^{-1}(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &\Rightarrow \lambda\boldsymbol{\alpha} = \mathbf{y} - \mathbf{X}\mathbf{X}^t\boldsymbol{\alpha} \\ &\Rightarrow (\mathbf{X}\mathbf{X}^t + \lambda I)\boldsymbol{\alpha} = \mathbf{y} \\ &\Rightarrow \boldsymbol{\alpha} = (\mathbf{G} + \lambda I)^{-1}\mathbf{y} \end{aligned}$$
- $\mathbf{G} = \mathbf{X}\mathbf{X}^t$ or, component-wise, $G_{ij} =< \mathbf{x}_i, \mathbf{x}_j >$.
- The resulting prediction function is given by $g(\mathbf{x}) =< \mathbf{w}, \mathbf{x} >= \left\langle \sum_{n=1}^{N} \alpha_n\mathbf{x}_n, \mathbf{x} \right\rangle = \sum_n \alpha_n < \mathbf{x}_n, \mathbf{x} >$

Regression notes

Take home message

- Gradient descent
  - On-line
  - Batch
- Normal equations
- Equivalence of LMS and MLE
- LR does not mean fitting linear relations, but linear combination or basis functions (that can be non-linear)
- The dual formulation opens the door to the introduction of kernel regression methods

Regression notes

Questions

- In some contexts it is interesting to introduce different costs per example in the error function

$$\mathcal{L}(\mathbf{w}, S) = \frac{1}{2}\sum_{n=1}^{N} c_n(y_n - \mathbf{w}^t\mathbf{x}_n)^2 + \frac{1}{2}\lambda\mathbf{w}^t\mathbf{w}$$

- Deduce in matrix notation
  - the LMS update rule
  - the steepest descent rule
  - the normal equation
  - the dual formulation

Limitations of Fixed Basis Functions

- comment the conjugate priors
- comment the Bias-Variance Decomposition
- Limitations of Fixed Basis Functions
  - M basis function along each dimension of a D-dimensional input space requires M^D basis functions: the curse of dimensionality.
  - In the next lecture, we shall see how we can get away with fewer basis functions, by choosing these using the training data.

References


Christopher M. Bishop
Pattern recognition and machine learning,
Springer, 2006 .


Eric Xing' Homepage
http://www.cs.cmu.edu/~epxing/
http://www.cs.cmu.edu/~epxing/Class/10701-08s/


Mário A. T. Figueiredo
Lecture Notes on Linear Regression
http://www.lx.it.pt/~mtf/linear_regression.pdf