COSC343: Artificial Intelligence

Lecture 7 : Linear regression and Least Squares

Lech Szymanski

Dept. of Computer Science, University of Otago

In today's lecture

- Regression
- Linear vs. non-linear systems
- Linear regression
- Least squares
- Linear least squares

Regression

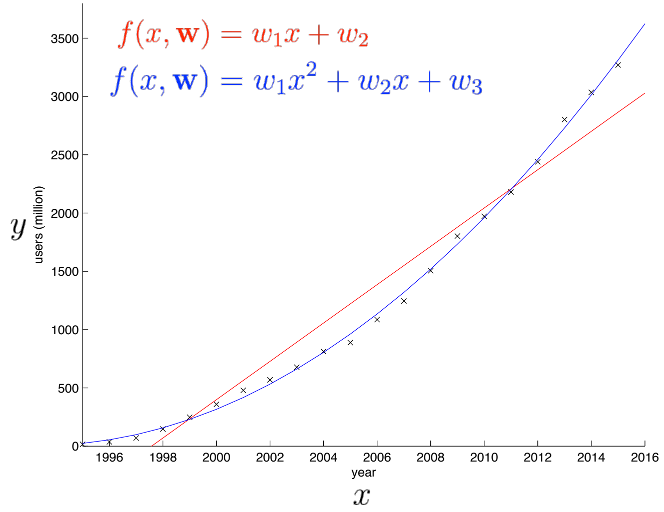
Regression is the problem of modelling a relationship between input variable of M dimensions, $\mathbf{x} = [x_1 \quad \ldots \quad x_M]^T$ and **continuous output** variable $y \in \mathbb{R}$,such that $y = f(\mathbf{x}, \mathbf{w})$.

- Attributes of the input can be discrete or continuous

Again, we're concerned with methods that learn a regression function from a set of *known input-output* examples:

- That is, the training data consist of N sample inputs - each a vector of dimension M for which *correct* output is known.

An example: future internet usage



Linear vs. non-linear systems

Linear w.r.t w

$$y = wf(x)$$

- Easy to solve for w

$$w = \frac{y}{f(x)}$$

- Derivative w.r.t. w doesn't depend on the value of w

$$\frac{dy}{dw} = f(x)$$

Non-linear w.r.t w

$$y = f(x, w)$$

- Not-easy or not possible to solve for w

$$w = g(x, f(x, w))$$

- Derivative w.r.t. w doesn't may depend on value of w

$$\frac{dy}{dw} = \frac{df(x, w)}{dw}$$

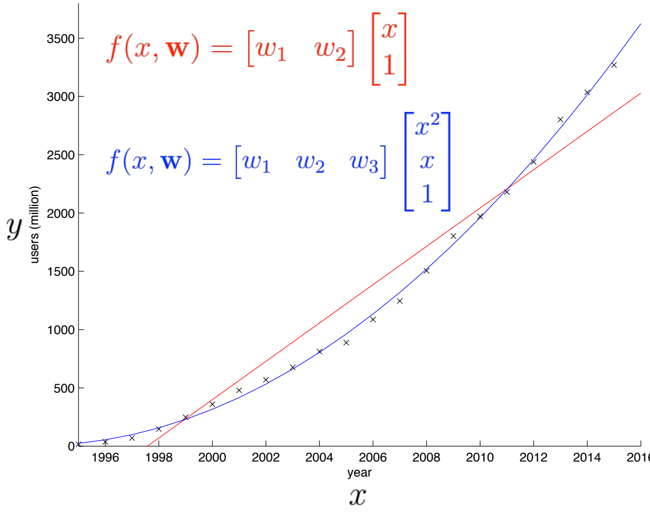
Linear regression

A regression problem modelled with a hypothesis function that is a weighted sum of a set of base functions is called **linear regression**.

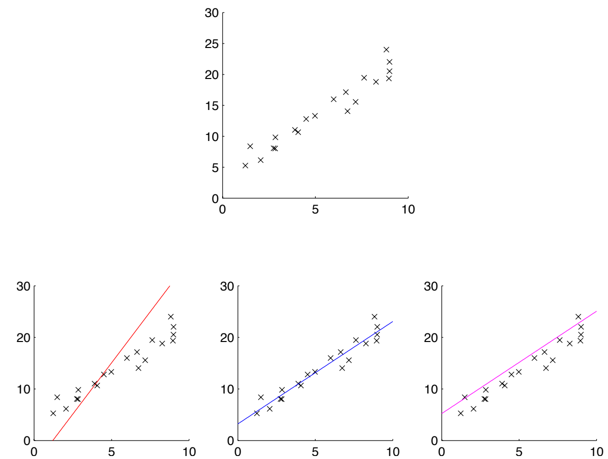
- The weight coefficients are the parameters of the model
- The model is linear in parameters
- The model can be non-linear in input

$$f(\mathbf{x}, \mathbf{w}) = \sum_{u=1}^{U} w_u f_u(\mathbf{x}) = \begin{bmatrix} w_1 & \ldots & w_U \end{bmatrix} \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_U(\mathbf{x}) \end{bmatrix}$$

An example: future internet usage

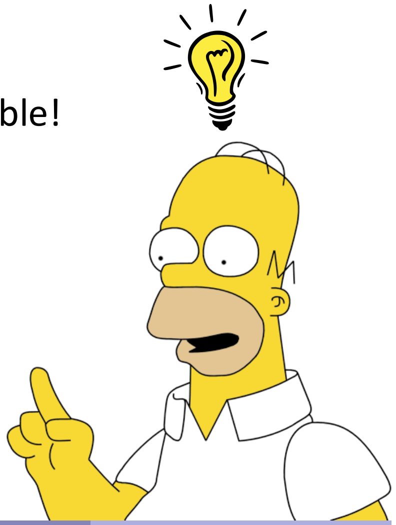


An example: Which fit is best?



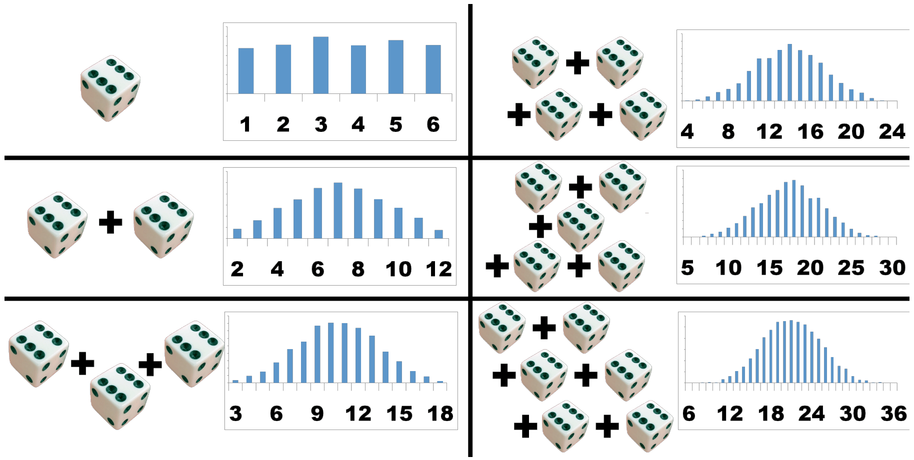
How to describe the best fit mathematically?

- We want to fit data well!
- We don’t want to overfit!
- We want the best fit possible!



The residual error should be normally distributed!!!

Recall the Central Limit Theorem (CLT)



CLT, Information Theory and the Least Squares

- The greater number of random events contributing to a result, the more Gaussian is the probability distribution of that result.
- Out of all real-valued distributions of a fixed variance, normal distribution has the maximum entropy – it’s most random.
- Given a hypothesis, fit it to data so that residual error is zero on average, and otherwise as random as possible – that is, normally distributed with zero mean.
  - Least squares doesn’t tell you how to pick a good hypothesis, just how to fit it to data

Least squares from Maximum Likelihood

- Given a hypothesis function $y = f(\mathbf{x}, \mathbf{w})$ and a training set $\{(\mathbf{x}_1, \tilde{y}_1), \ldots, (\mathbf{x}_N, \tilde{y}_N)\}$, define residual error as the difference between the model output and the true (target) output: $e = y - \tilde{y}$
- Assuming errors from sample to sample are independent and identically distributed (i.i.d) with distribution $p(e)$:
  - Find parameters that maximise the overall probability ... $\prod_i p(e_i)$
  - ...which is the same as minimizing .... $-\sum_i \ln p(e_i)$
  - ...which for $p(e_i) = \frac{1}{\sqrt{2\pi}\sigma}\exp^{-\frac{e_i^2}{2\sigma^2}}$ is the same as minimizing:

$$J = \frac{1}{2}\sum_i e_i^2$$

cost

sum of squared errors

Least squares solution for linear regression

The least squares parameters that minimise $J = \frac{1}{2}\sum_i e_i^2$, where $e_i = y_i - \tilde{y}_i$, can be found by solving:

$$\frac{dJ}{d\mathbf{w}} = \sum_i \frac{de_i}{d\mathbf{w}} e_i = 0$$

For models that are linear in parameters, $y = f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_U(\mathbf{x}) \end{bmatrix}$, where

$\mathbf{w}^T = \begin{bmatrix} w_1 & \dots & w_U \end{bmatrix}$, there is a closed form solution:

$$\mathbf{w} = \left(\mathbf{F}\mathbf{F}^T\right)^{-1} \mathbf{F}\tilde{\mathbf{y}}^T,$$ where

$$\mathbf{F} = \begin{bmatrix} f_1(\mathbf{x}_1) & \dots & f_1(\mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ f_U(\mathbf{x}_1) & \dots & f_U(\mathbf{x}_N) \end{bmatrix}$$ and $$\tilde{\mathbf{y}} = \begin{bmatrix} \tilde{y}_1 & \dots & \tilde{y}_N \end{bmatrix}$$

An example: future internet usage

x	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
$\tilde{y}$	16	36	70	147	248	359	479	569	677	812	888	1086	1245	1504	1802	1971	2180	2439	2802	3035

Hypothesis: $y = f(x, \mathbf{w}) = w_1 x + w_2$

$\tilde{\mathbf{y}} = [16 \;\; 36 \;\; 70 \;\; 147 \;\; 248 \;\; 359 \;\; 479 \;\; 569 \;\; 677 \;\; 812 \;\; 888 \;\; 1086 \;\; 1245 \;\; 1504 \;\; 1802 \;\; 1971 \;\; 2180 \;\; 2439 \;\; 2802 \;\; 3035]$

$$\mathbf{F} = \begin{bmatrix} x_1 & \dots & x_{20} \\ 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} 1995 & 1996 & 1997 & 1998 & 1999 & 2000 & 2001 & 2002 & 2003 & 2004 & 2005 & 2006 & 2007 & 2008 & 2009 & 2010 & 2011 & 2012 & 2013 & 2014 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

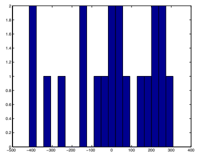
$\mathbf{w}^T = [w_1 \;\; w_2]$

Solve: $\mathbf{w} = \left(\mathbf{F}\mathbf{F}^T\right)^{-1} \mathbf{F}\tilde{\mathbf{y}}^T$

$$\mathbf{w} = \begin{bmatrix} 158.00 \\ -315600.29 \end{bmatrix}$$

Root mean square (RMS) error:

$\sqrt{E[e^2]} = 218.3$



$e = y - \tilde{y}$

An example: future internet usage

x	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
$\tilde{y}$	16	36	70	147	248	359	479	569	677	812	888	1086	1245	1504	1802	1971	2180	2439	2802	3035

Hypothesis: $y = f(x, \mathbf{w}) = w_1 x^2 + w_2 x + w_3$

$\tilde{\mathbf{y}} = [16 \;\; 36 \;\; 70 \;\; 147 \;\; 248 \;\; 359 \;\; 479 \;\; 569 \;\; 677 \;\; 812 \;\; 888 \;\; 1086 \;\; 1245 \;\; 1504 \;\; 1802 \;\; 1971 \;\; 2180 \;\; 2439 \;\; 2802 \;\; 3035]$

$$\mathbf{F} = \begin{bmatrix} x_1^2 & \dots & x_{20}^2 \\ x_1 & \dots & x_{20} \\ 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} 3980025 & 3984016 & 3988009 & 3992004 & 3996001 & 4000000 & 4004001 & 4008004 & 4012009 & 4016016 & 4020025 & 4024036 & 4028049 & 4032064 & 4036081 & 4040100 & 4044121 & 4048144 & 4052169 & 4056196 \\ 1995 & 1996 & 1997 & 1998 & 1999 & 2000 & 2001 & 2002 & 2003 & 2004 & 2005 & 2006 & 2007 & 2008 & 2009 & 2010 & 2011 & 2012 & 2013 & 2014 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

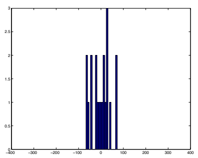
$\mathbf{w}^T = [w_1 \;\; w_2 \;\; w_3]$

Solve: $\mathbf{w} = \left(\mathbf{F}\mathbf{F}^T\right)^{-1} \mathbf{F}\tilde{\mathbf{y}}^T$

$$\mathbf{w} = \begin{bmatrix} 7.24 \\ -28868.93 \\ 28776401.68 \end{bmatrix}$$

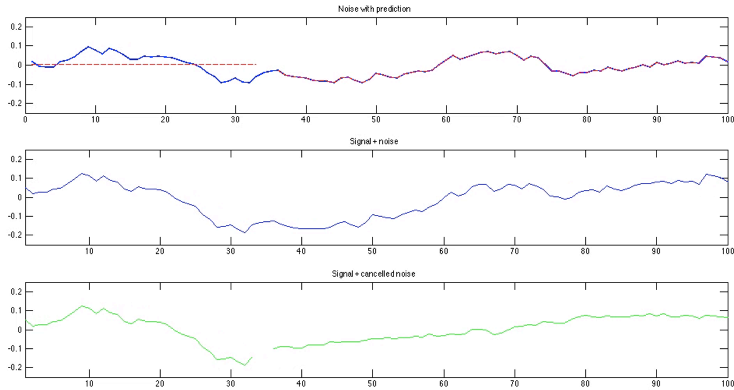
Root mean square (RMS) error:

$\sqrt{E[e^2]} = 40.6$



$e = y - \tilde{y}$

An example: active noise control



An example: active noise control

Sample number

- Outside noise: $x[n]$

Uncorrupted signal

- Signal + noise: $s[n] = \tilde{s}[n] + x[n]$

Create a model, where the output is a linear combination of past M samples:

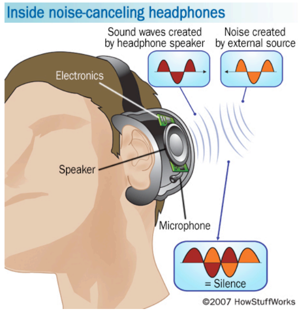
$$y[n] = \sum_{m=1}^{M} w_m x[n-m]$$

Compute parameters that predict next sample from previous M samples using saved batch of N samples:

$$e[n] = y[n] - x[n]$$

Use these parameters to predict the next sample:

$$y[n+1] = w_1 x[n] + \ldots + w_M x[n-M]$$



Send out inverted predicted signal in time to meet the next sample of the coming noise:

$$s[n+1] - y[n+1] \approx \tilde{s}[n+1]$$

if $y[n+1] \approx x[n+1]$

recovered signal

Summary and reading

- Regression = model output is continuous
- Linear regression – model is linear in parameters (not necessarily in input)
- Least squares fit – best fit that makes the error normally distributed
- Linear least squares – closed form solution for parameters

Reading for the lecture: AIMA Chapter 18 Section 6

Reading for next lecture: AIMA Chapter 18 Section 4