THE UNIVERSITY OF MELBOURNE

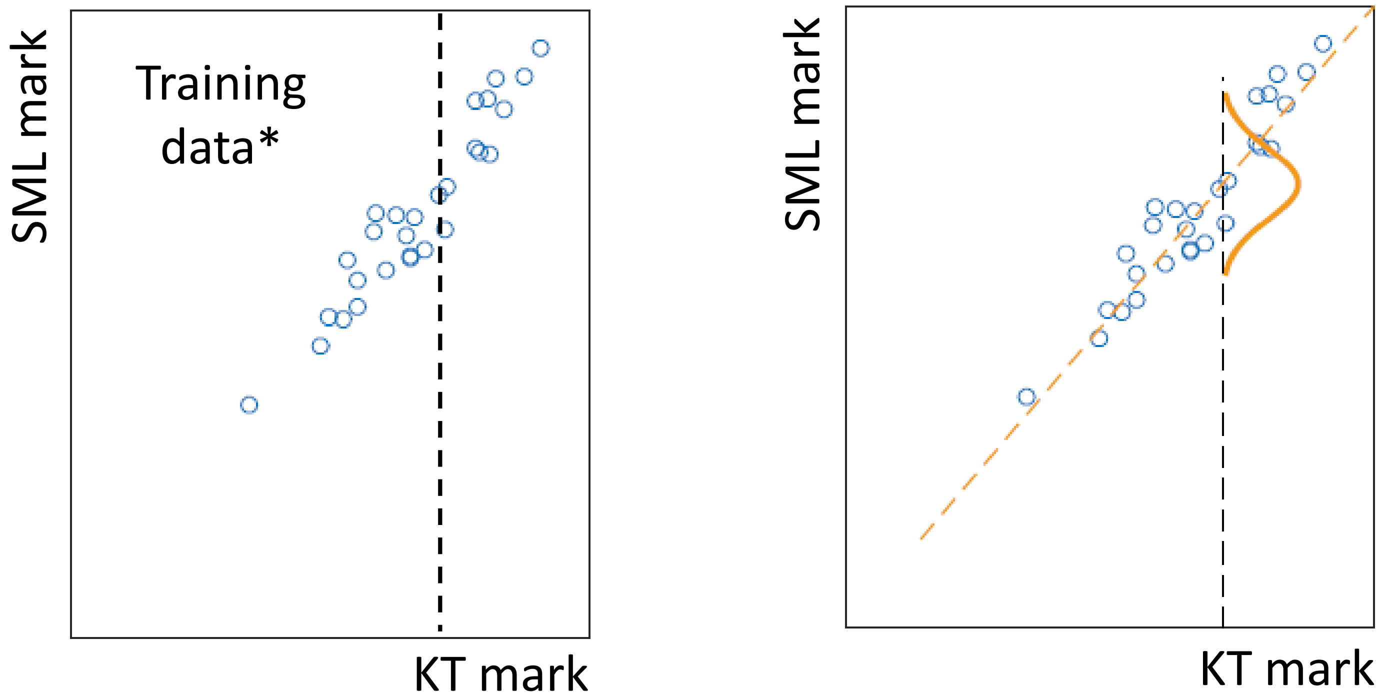
Workshop 3

COMP90051 Statistical Machine Learning

Semester 2, 2019

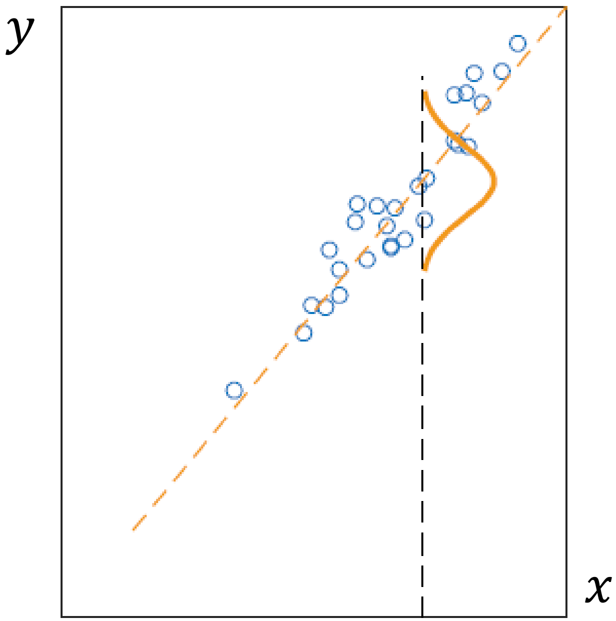
Data is noisy!

Example: predict mark for Statistical Machine Learning (SML) from mark for Knowledge Technologies (KT)



* synthetic data :)

Regression as a probabilistic model



- Assume a probabilistic model: $Y = \boldsymbol{X}'\boldsymbol{w} + \varepsilon$
  - Here $\boldsymbol{X}, Y$ and ε are r.v.'s
  - Variable ε encodes noise
- Next, assume Gaussian noise (indep. of $\boldsymbol{X}$): $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

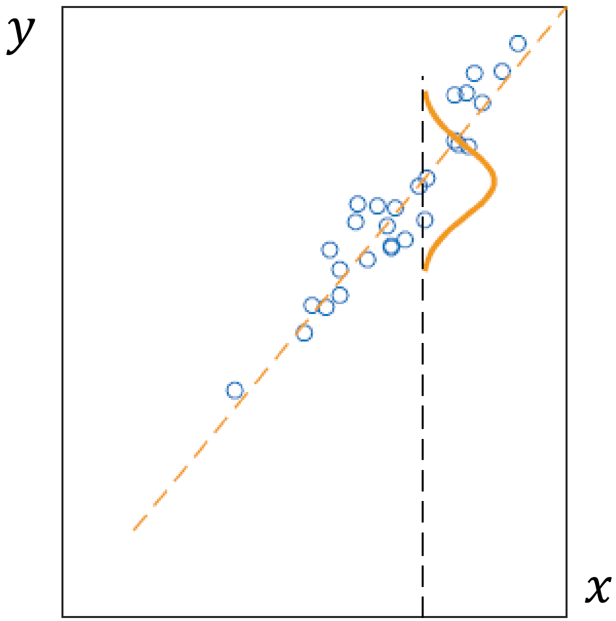
$Y \sim \mathcal{N}(w'X, \sigma^2)$

this is a squared error!

- Recall that $\mathcal{N}(x; \mu, \sigma^2) \equiv \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$
- Therefore

$$p_{\boldsymbol{w},\sigma^2}(y|\boldsymbol{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\boxed{(y - \boldsymbol{x}'\boldsymbol{w})^2}}{2\sigma^2}\right)$$

Parametric probabilistic model



- Using simplified notation, discriminative model is:

$$p(y|\boldsymbol{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y - \boldsymbol{x}'\boldsymbol{w})^2}{2\sigma^2}\right)$$

- Unknown parameters: $\boldsymbol{w}, \cancel{\sigma^2}$

- Given observed data $\{(\boldsymbol{X}_1, Y_1), \dots, (\boldsymbol{X}_n, Y_n)\}$, we want to find parameter values that "best" explain the data
- Maximum likelihood estimation: choose parameter values that maximise the probability of observed data

Maximum likelihood estimation

- Assuming independence of data points, the probability of data is

$$p(y_1, \dots, y_n | \boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \prod_{i=1}^{n} p(y_i | \boldsymbol{x_i})$$

- For $p(y_i|\boldsymbol{x}_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \boldsymbol{x}_i{}'\boldsymbol{w})^2}{2\sigma^2}\right)$

- "Log trick": Instead of maximising this quantity, we can maximise its logarithm (why?)

$$\sum_{i=1}^{n} \log p(y_i|\boldsymbol{x_i}) = -\frac{1}{2\sigma^2} \boxed{\sum_{i=1}^{n} (y_i - \boldsymbol{x}_i{}'\boldsymbol{w})^2} + C$$

the sum of squared errors!

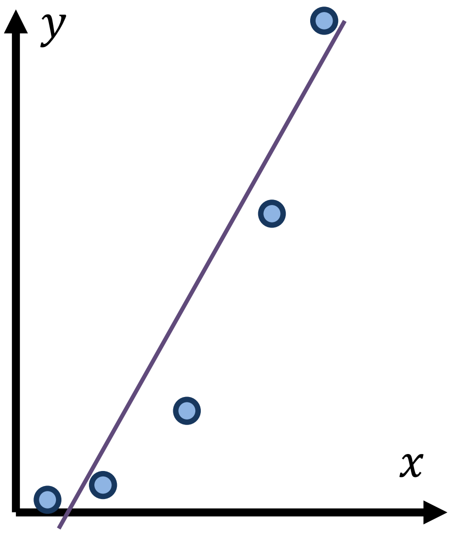
here C doesn't depend on $\boldsymbol{w}$ (it's a constant)

- Under this model, maximising log-likelihood as a function of $\boldsymbol{w}$ is equivalent to minimising the sum of squared errors

Basis expansion for linear regression

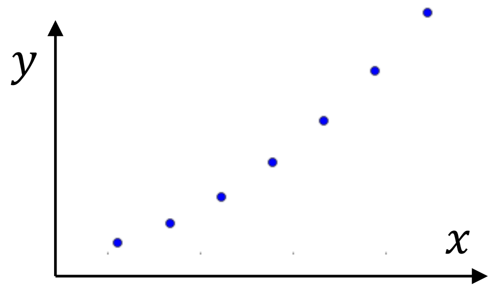
- Let’s take a step back. Back to linear regression and least squares
- Real data is likely to be non-linear
- What if we still wanted to use a linear regression?
  - It’s simple, easier to understand, computationally efficient, etc.
- How to marry non-linear data to a linear method?

If you can’t beat’em, join’em



Transform the data

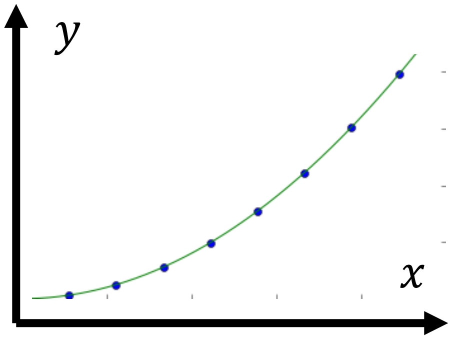
- The trick is to transform the data: Map data onto another features space, s.t. data is linear in that space
- Denote this transformation $\varphi: \mathbb{R}^m \to \mathbb{R}^k$. If $\boldsymbol{x}$ is the original set of features, $\varphi(\boldsymbol{x})$ denotes new feature set
- Example: suppose there is just one feature x, and the data is scattered around a parabola rather than a straight line



Example: Polynomial regression

- No worries, mate: define

$$\varphi_1(x) = x$$
$$\varphi_2(x) = x^2$$



- Next, apply linear regression to φ_1, φ_2

$$y = w_0 + w_1\varphi_1(x) + w_2\varphi_2(x) = w_0 + w_1 x + w_2 x^2$$

and here you have **quadratic regression**

- More generally, obtain **polynomial regression** if the new set of attributes are powers of x

Basis expansion

- Data transformation, also known as basis expansion, is a general technique
  - We'll see more examples throughout the course
- It can be applied for both regression and classification
- There are many possible choices of φ