Data Mining
Prediction

Jingpeng Li

What is Prediction?

- Predicting the identity of one thing based purely on the description of another related thing
- Not necessarily future events, just unknowns
- Based on the relationship between a thing that you can know and a thing you need to predict

Terms

Predictor => Predicted

- When building a predictive model, you have data covering both
- When using one, you have data describing the *predictor* and you want it to tell you the *predicted* value

How Does it Differ From Classification?

- A classification problem could be seen as a predictor of classes, but ….
- Predicted values are usually continuous whereas classifications are discreet.
- Predictions are often (but not always) about the future whereas classifications are about the present.
- Classification is more concerned with the input than the output

Usual Examples

- Predicting levels of sales that will result from a price change or advert.
- Predicting whether or not it will rain based on current humidity
- Predicting the colour of a pottery glaze based on a mixture of base pigments
- Predicting how far up the charts a single will go
- Predicting how much revenue a book will bring

Techniques

- Most prediction techniques are based on mathematical models:
  - Simple statistical models such as linear regression
  - Non-linear statistics such as power series
  - Neural networks, RBFs, etc
- All based on fitting a curve through the data, that is, finding a relationship from the predictors to the predicted

Simple Worked Example

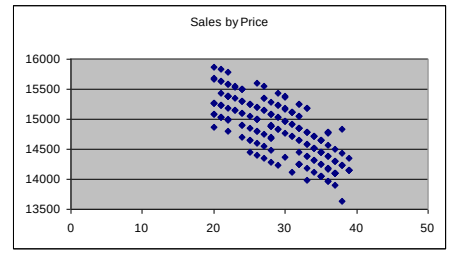
- Predicting sales levels for a national newspaper

Predictors	Predicted
– Price – Front cover story – Competitions – Advertising spend	Sales in Units

The Data

Price	Cover	Competition	Advert spend	Sales
22	Political	No	5900	15392
39	Political	No	5100	14350
28	Sport	No	5700	14491
25	Sport	No	5600	14849
38	Royal	No	5400	14029
22	Royal	No	5900	15192
21	Crime	No	5500	15433
20	Royal	No	5400	15273
31	Royal	High Val	5700	14914
26	Royal	Low Val	5300	14596
23	Sport	No	5100	14742
23	Sport	High Val	5900	15147
21	Royal	No	5400	15032
29	Crime	Low Val	5800	14635
25	Sport	Low Val	5500	14449
24	Sport	Low Val	5900	14500
32	Crime	No	5800	14852
27	Sport	No	5700	14546
30	Sport	High Val	5600	14774
31	Royal	No	5500	14713
29	Political	High Val	5900	15435
39	Sport	No	5600	13753
32	Political	No	5900	14852
23	Royal	High Val	5600	15345
31	Sport	No	5800	14315
27	Royal	No	5900	14947
31	Sport	No	5300	14511

Sales increase as price decreases but other factors play a part too.



Mathematical Model

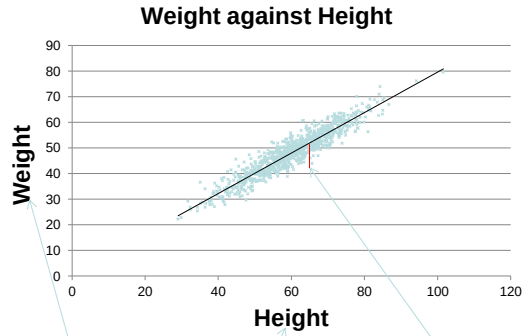
- Learns relationship between all predictors at once and the predicted outcome:

  Sales=f(Price, Cover, Adverts, Competition)
- Sales are a function of several variables.
- The job of a data mining algorithm is to find the function *f*

Regression

- In statistical modelling, regression analysis is a statistical process for estimating the relationships among variables.
- Regression models are built from data to predict the average you would expect one variable to have, given you know the value of one or more others.
- Simple linear regression maps one variable onto the mean value of another.

Example: weight-height relation



$$y_i = bx_i + a + \varepsilon_i$$

Simple Linear Regression

- To find the best values for a and b, simple linear regression uses a method known as ordinary least squares (OLS)
- Least squares means that the sum of the squared distance between each data point and its associated prediction is minimised
- That is, it minimises $\sum_{i=1}^{n} \varepsilon_i^2$

Finding a and b

- In the case of simple linear regression, a and b can be calculated as follows:

$$b = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$a = \bar{y} - b\bar{x}$$

Multiple Regression

- With multiple inputs, the general form of linear regression is

$$y_i = b_0 + x_{i1}b_1 + x_{i2}b_2 + x_{i3}b_3 + ... + \varepsilon_i$$

$$Y = Xb + \varepsilon$$

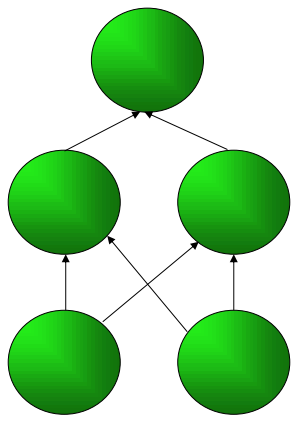
- The parameters in b are calculated as

$$b = (X^T X)^{-1} X^T Y$$

Neural Network Example

- A certain type of neural network, called a multi layer perceptron (MLP) can learn a function between our inputs (qualities of a newspaper) and the outcome (Sales)
- It works by building the function out of many small simple functions, joined by weighted connections

MLP Structure



Output Layer

Hidden Layer

Input Layer

Every unit does the same thing:

$$O_j = f(\sum_i w_{ij} \cdot O_i)$$

$$f(a) = \frac{1}{1+e^{-a}}$$

Neural Network Example

- A neural network uses the data to modify the weighted connections between all of its functions until it is able to predict the data accurately
- This process is referred to as *training* the neural network

Neural Network Training

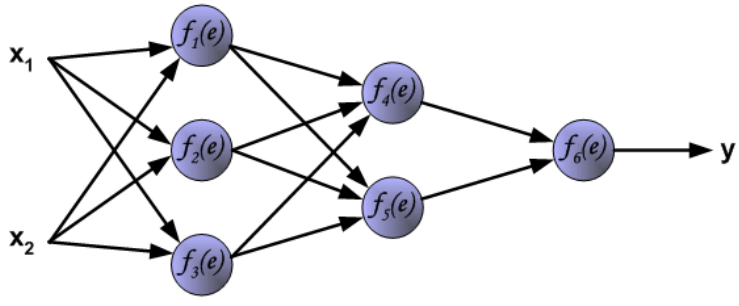
1. Prepare the data so that a file contains the predictors and the predicted variables with an example per row
2. Split the data into a test set and a training set
3. Read each row in turn into the neural network, presenting the predictors as inputs and the predicted value as the target output
4. Make a prediction and compare the value given by the neural network to the target value
5. Update the weights – see next slide
6. Present the next example in the file
7. Repeat until the error no longer reduces – ideally stop when the test error is at its lowest.

How are the Weights Changed?

- Training data has inputs and outputs, in this example, newspaper details and sales figures
- The MLP starts with random weights
- Each example in the training data is used as an input and the network generates an output
- The difference between that output and the value in the training data is known as the **error**

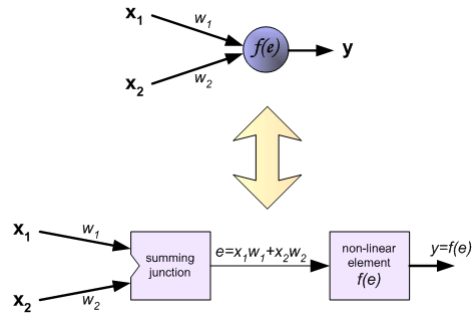
Backpropagation

- To illustrate this process a 3-layer neural network with 2 inputs and 1 output



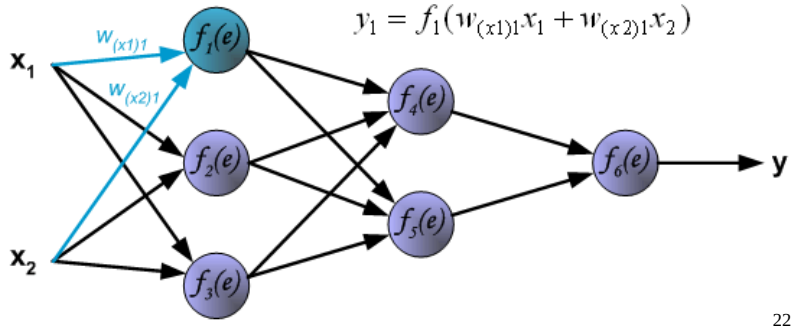
Backpropagation

- Each neuron is composed of two units
  1. The weighted sum of input signals.
  2. The realization of neuron activation function. Signal *e* is adder output signal, and *y* = *f(e)* is output signal of neuron.

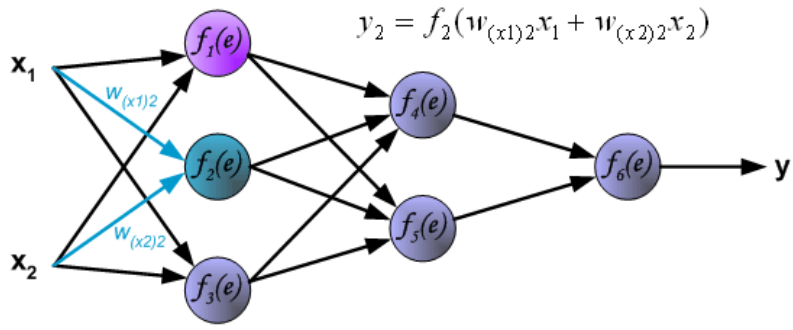


Backpropagation

- $w_{(xm)n}$ -- weights of connections between network input x_m and neuron n in input layer.
- Y_n -- output signal of neuron n.

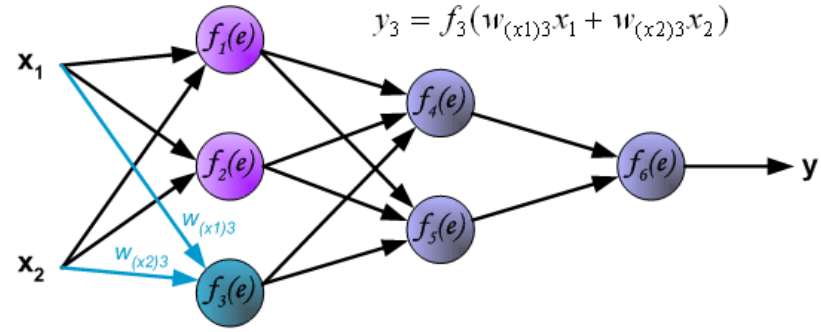


Backpropagation



23

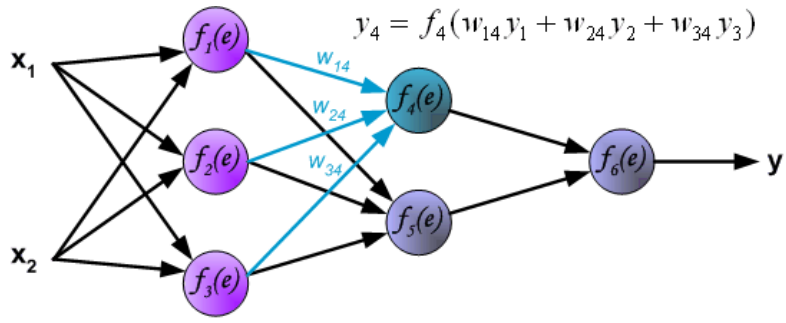
Backpropagation



24

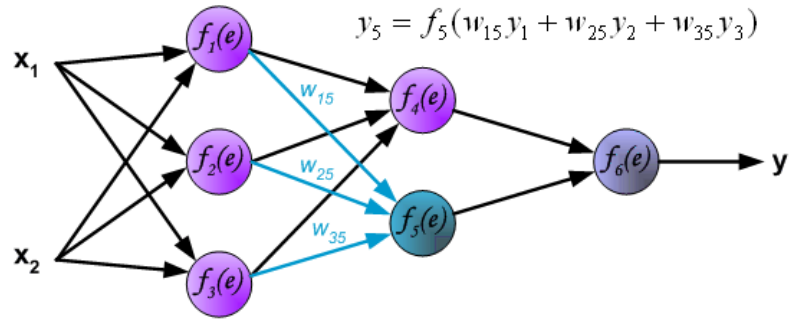
Backpropagation

- Propagation of signals through the hidden layer
- W_{mn} -- weights of connections between output of neuron *m* and input of neuron *n* in the next layer



25

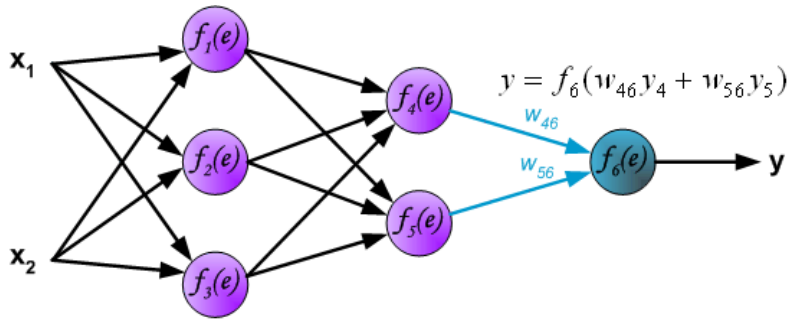
Backpropagation



26

Backpropagation

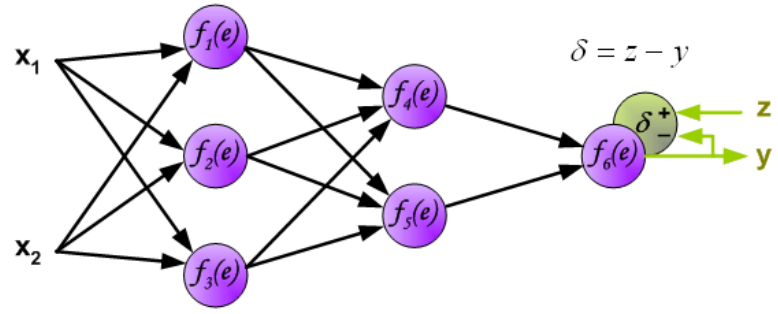
- Propagation of signals through the output layer



27

Backpropagation

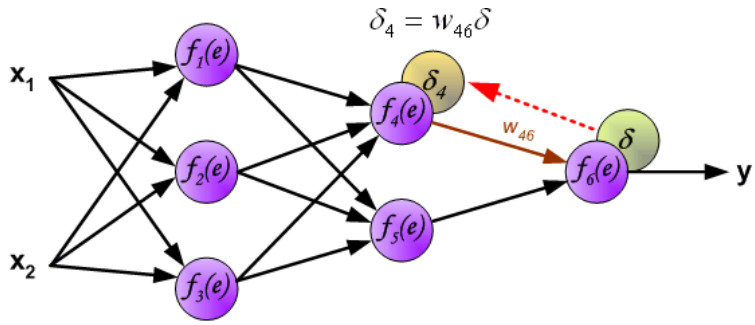
- The output *y* is compared with the desired output value (the target) z, which is found in training data set.
- The difference is called error signal *d* of output layer neuron



28

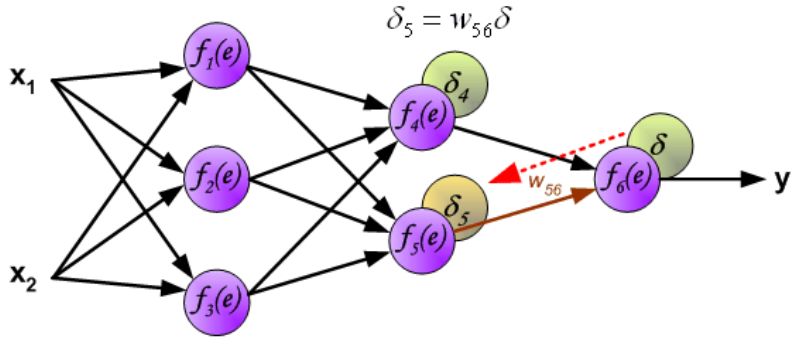
Backpropagation

- The idea is to propagate error signal *d* back to all neurons, which output signals were input for discussed neuron.



29

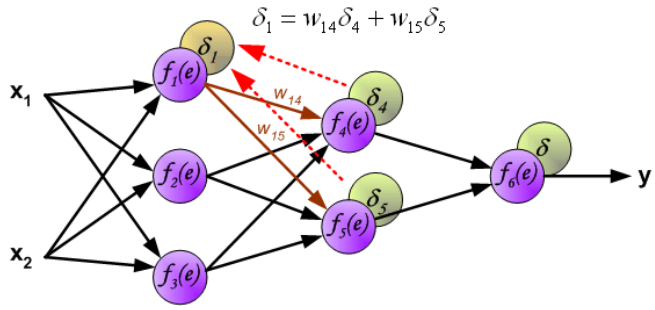
Backpropagation



30

Backpropagation

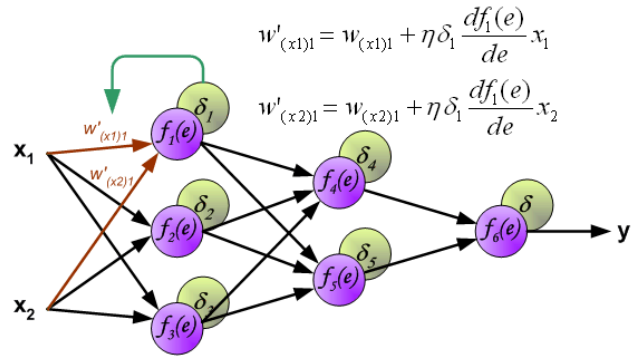
- w_{mn} used to propagate errors back are equal to this used during computing output value.
- This technique is used for all layers. If propagated errors came from few neurons they are added.



31

Backpropagation

- When the error signal for each neuron is computed, the weights of each input node are modified
- *df(e)/de* represents derivative of activation function



32

Derivative of Sigmoid function

Let's denote the sigmoid function as $\sigma(x) = \frac{1}{1+e^{-x}}$.

The derivative of the sigmoid is $\frac{d}{dx}\sigma(x) = \sigma(x)(1-\sigma(x))$.

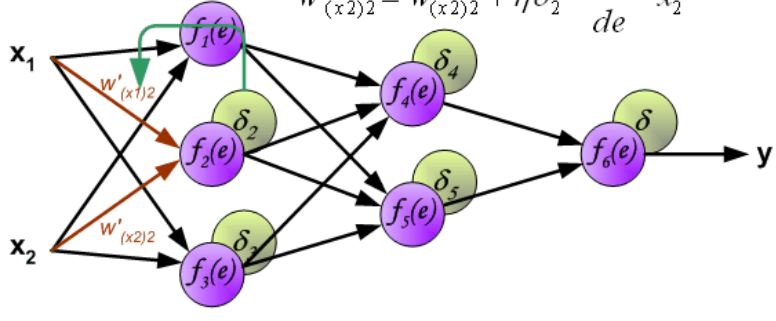
Here's a detailed derivation:

$$\begin{aligned}
\frac{d}{dx}\sigma(x) &= \frac{d}{dx}\left[\frac{1}{1+e^{-x}}\right] \\
&= \frac{d}{dx}\left(1+e^{-x}\right)^{-1} \\
&= -(1+e^{-x})^{-2}(-e^{-x}) \\
&= \frac{e^{-x}}{(1+e^{-x})^2} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{(1+e^{-x})-1}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \left(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right) \\
&= \frac{1}{1+e^{-x}} \cdot \left(1 - \frac{1}{1+e^{-x}}\right) \\
&= \sigma(x) \cdot (1-\sigma(x))
\end{aligned}$$

Backpropagation

$$w'_{(x1)2} = w_{(x1)2} + \eta\delta_2 \frac{df_2(e)}{de} x_1$$

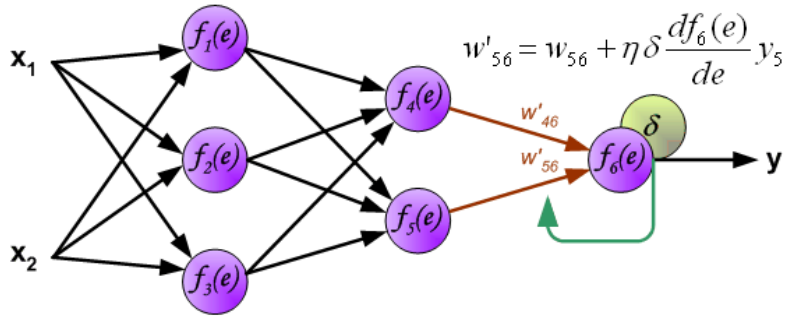
$$w'_{(x2)2} = w_{(x2)2} + \eta\delta_2 \frac{df_2(e)}{de} x_2$$



Backpropagation

$$w'_{46} = w_{46} + \eta\delta\frac{df_6(e)}{de}y_4$$

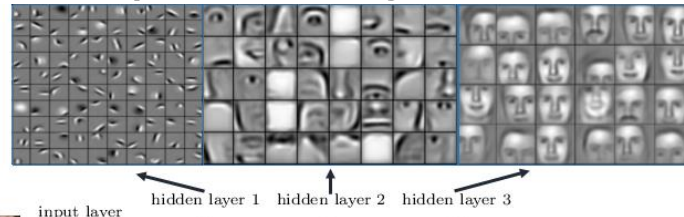
$$w'_{56} = w_{56} + \eta\delta\frac{df_6(e)}{de}y_5$$

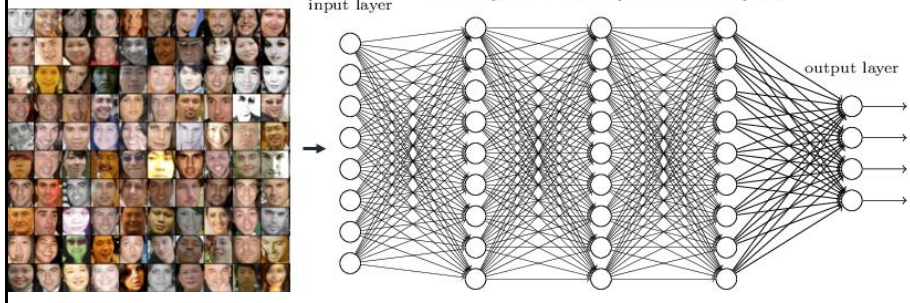


35

Deep Learning

Deep neural networks learn hierarchical feature representations





- A host of statistical machine learning techniques
- Enables the automatic learning of feature hierarchies
- Usually based on artificial neural networks

36

Qualities of a Predictor

- Which ever technique you use, it should have the following qualities:
  - Ability to make correct predictions on data that is not in the original training data
  - Ability to provide a certainty measure with its predictions
- How well a solution performs depends on both the data and the person who built it

Important Concepts

- Over Fitting
  - A data mining predictor can capture the structure of the data so well that irrelevant details are picked up and used when they are not generally true
- Data Quantity and Quality
  - Insufficient data or data that does not capture the relationship between predictors and predicted can produce a very poor solution

Important Concepts

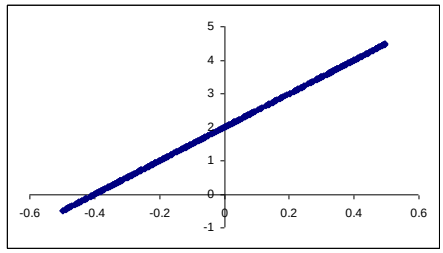
- Multiple solutions
  - It is possible (easy, in fact) to build more than one correct (or equally accurate) predictor from the same data set
  - Several such predictors should be built and compared
  - A winner might be chosen, or several could be used as a 'panel of experts'

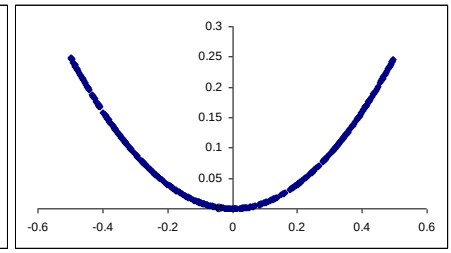
Non-linear?

- Curvy! Or to be more specific:

  "If *x* predicts *y* then they have a non-linear relationship if the effect on *y* of a small change in *x* depends on the current value of *x*."

Non-linear?





Where ever you are along the line on the linear plot above, moving one unit to the right will move you up 5 units. The 1/5 ratio is constant so the relationship is linear

Here, moving a unit to the right on the line above will carry you up a different amount, depending on where you are: non-linear

41

Non-Linear

- Note that if you have more than one predictor, non-linearity can occur as two or more predictors combine
- E.g. Putting the price up 1p will cause you to sell 1000 fewer newspapers when there is a political story on the front cover, but only 500 fewer with sport on the cover

Advantages of Neural Networks

- Very powerful predictors – almost always better than any rule based system a human expert could design
- Can cope with non-linear relationships, multiple numeric and discreet variables
- Able to generalise to data that it has not seen before

Disadvantages

- How predictions are gained can be hard to understand by a human user
- Not easy to ask why an answer was given (though some help is possible)
- No rules to look at
- Can make big errors if not trained properly
- Requires a certain degree of faith!