

Unit 4 Regression Analysis

Dr. Latesh Malik

BE, MTech, PhD (Computer Science)

Associate Professor, Department of Computer Engineering

Govt Engineering College , Nagpur

Chapter 20 Correlation and Regression Analysis in R

- correlation and covariance
- correlation types and correlation coefficient
- regression analysis and its types
- build the simple and multiple regression models in R
- build the logistic regression model in R

Correlation Analysis

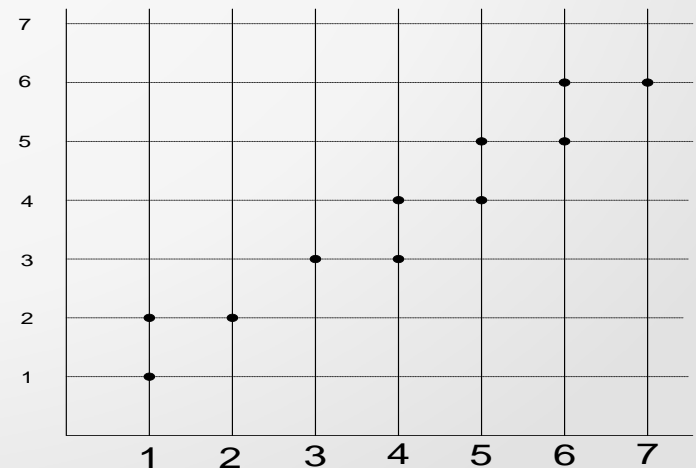
Association between variables is correlation.

For example, **weight is correlated to height.**

The correlation may be positive, negative or zero (scatter plot is shown below).

Positive correlation: If the value of the attribute A increases with an increase in the value of the attribute B, and vice-versa.

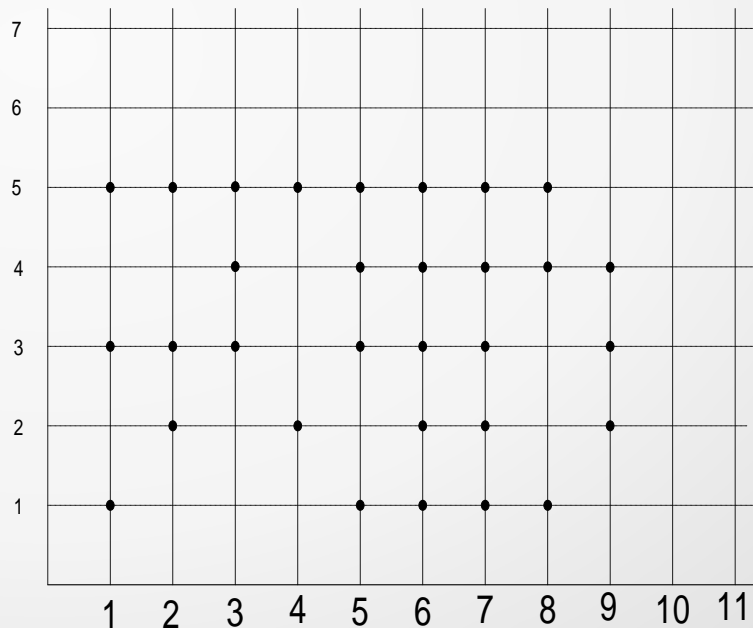
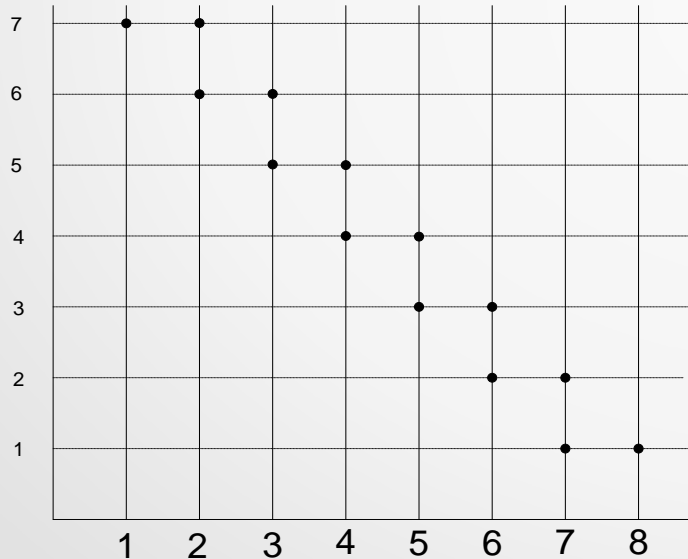
Positive correlation



Correlation Analysis

Negative correlation: If the value of the attribute A decreases with an increase in the value of the attribute B, and vice-versa.

Zero correlation: When the value of attribute A varies at random with attribute B, and vice-versa.



Correlation Coefficient

--The correlation coefficient is used to describe the linear relationship among quantitative variables.

--It measures the strength and direction of a relationship. The + and - signs indicate the direction of the relationship

--The magnitude indicates the strength of the relationship (ranging from 0 to 1 for a perfect predictable relationship).

Correlation Function in R

R has functions that can be used to produce covariance (cov()) and a variety of correlation (cor()) coefficients specified in the method parameter.

```
>cor(a,use=,method=)
```

where **a** is matrix or data frame;

use specifies the handling of missing data (default is “everything”);

method specifies the type of correlation (default is “pearson”, others can be spearman, kendall, partial, polychoric, polyserial, etc.).

Correlation Function in R

Example: Find the correlation and covariance of len and dose in the data set ToothGrowth. Observe if there is any linear relationship between the variables.

```
> head(ToothGrowth)
```

```
  len supp dose
1  4.2   VC  0.5
2 11.5   VC  0.5
3  7.3   VC  0.5
4  5.8   VC  0.5
5  6.4   VC  0.5
6 10.0   VC  0.5
```

```
> cor(ToothGrowth$len, ToothGrowth$dose)
```

```
[1] 0.8026913
```



```
> cov(ToothGrowth$len, ToothGrowth$dose)
```

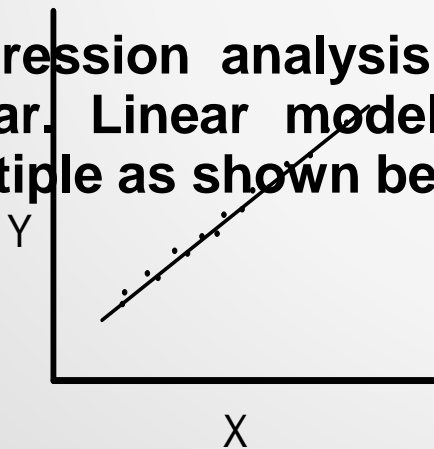
```
[1] 3.861299
```

The correlation coefficient of len and dose is 0.8026913. The variables are positively linearly related. The covariance is 3.86. It indicates a positive linear relationship between the two variables.

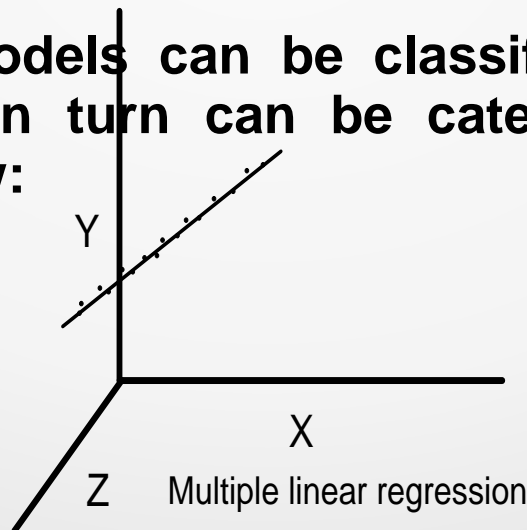
Regression Analysis

- It is a statistical modeling method used to formulate the mathematical model of the relationship among variables.
- The purpose of regression analysis is to estimate the value of the dependent variable, given the value/values of the independent variable.

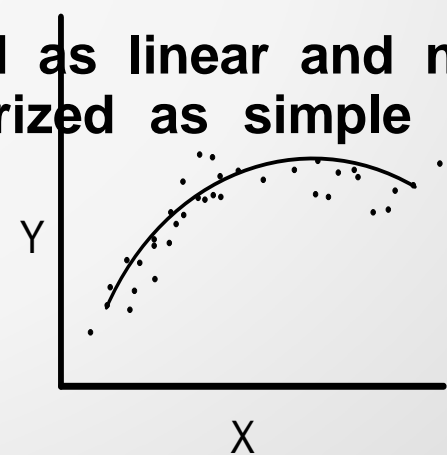
Regression analysis models can be classified as linear and non-linear. Linear models in turn can be categorized as simple and multiple as shown below:



Simple linear regression



Multiple linear regression



Non-linear regression

Prediction

- (Numerical) prediction is similar to classification
 - **construct a model**
 - **use model to predict value** for a given input
- Prediction is different from classification
 - Classification refers to predict categorical class label
 - Prediction models continuous-valued functions
- Major statistical method for prediction: regression
 - model **the relationship between** one or more *independent* or **predictor** variables and a *dependent* or **response** variable
- Regression analysis
 - Linear and multiple regression
 - Non-linear regression
 - Other regression methods: generalized linear model, Poisson regression, log-linear models, regression trees

Linear Regression

- Linear regression: involves a response variable y and a single predictor variable x

$$y = w_0 + w_1 x$$

where w_0 (y-intercept) and w_1 (slope) are regression coefficients

- Method of least squares: estimates the best-fitting straight line

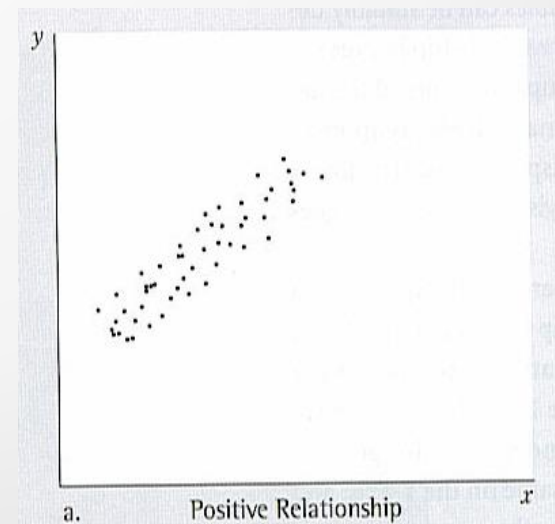
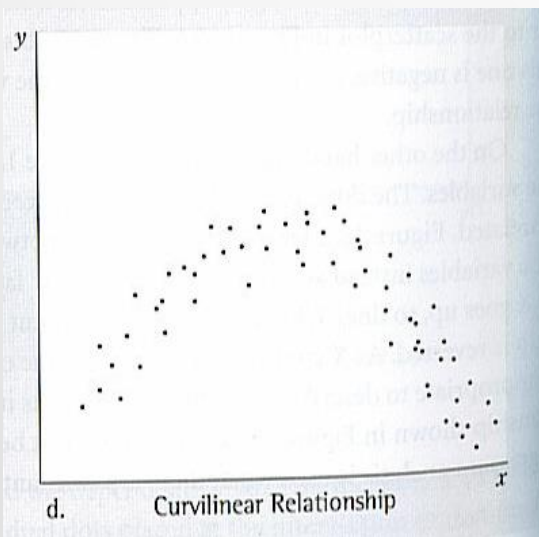
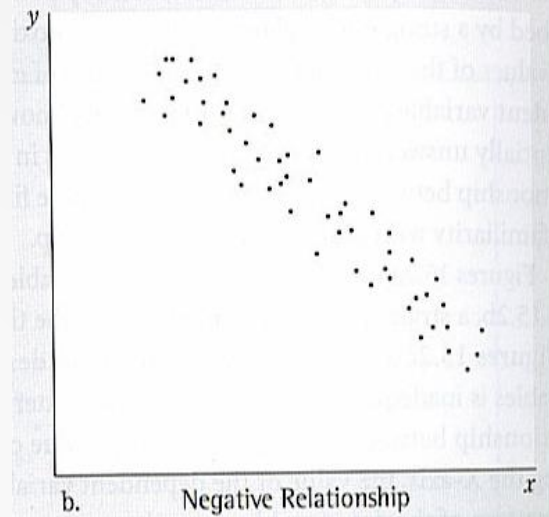
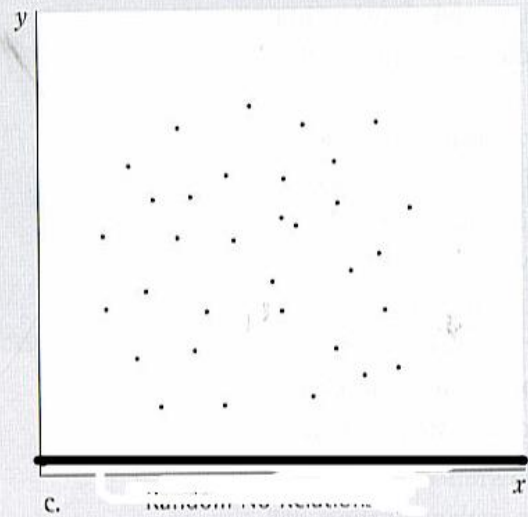
$$w_1 = \frac{\sum_{i=1}^{|D|} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|} (x_i - \bar{x})^2} \quad w_0 = \bar{y} - w_1 \bar{x}$$

- Multiple linear regression: involves more than one predictor variable
 - Training data is of the form $(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \dots, (\mathbf{X}_{|D|}, y_{|D|})$
 - Ex. For 2-D data, we may have: $y = w_0 + w_1 x_1 + w_2 x_2$

Scatter plots

- Regression analysis **requires interval and ratio-level data.**
- Regression analysis assumes a **linear relationship**. If you have a curvilinear relationship or no relationship, regression analysis is of little use.

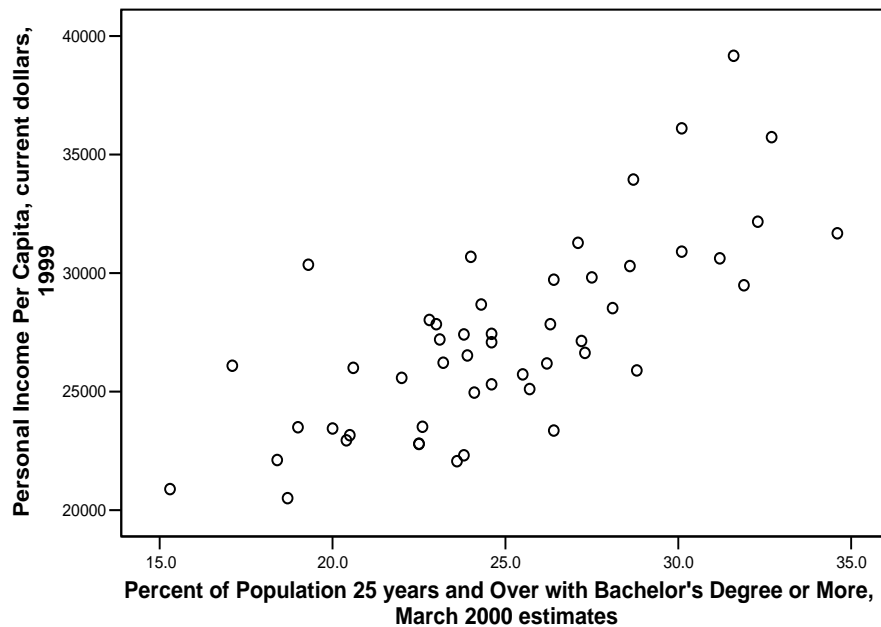
Types of Lines



Scatter plot

- This is a linear relationship

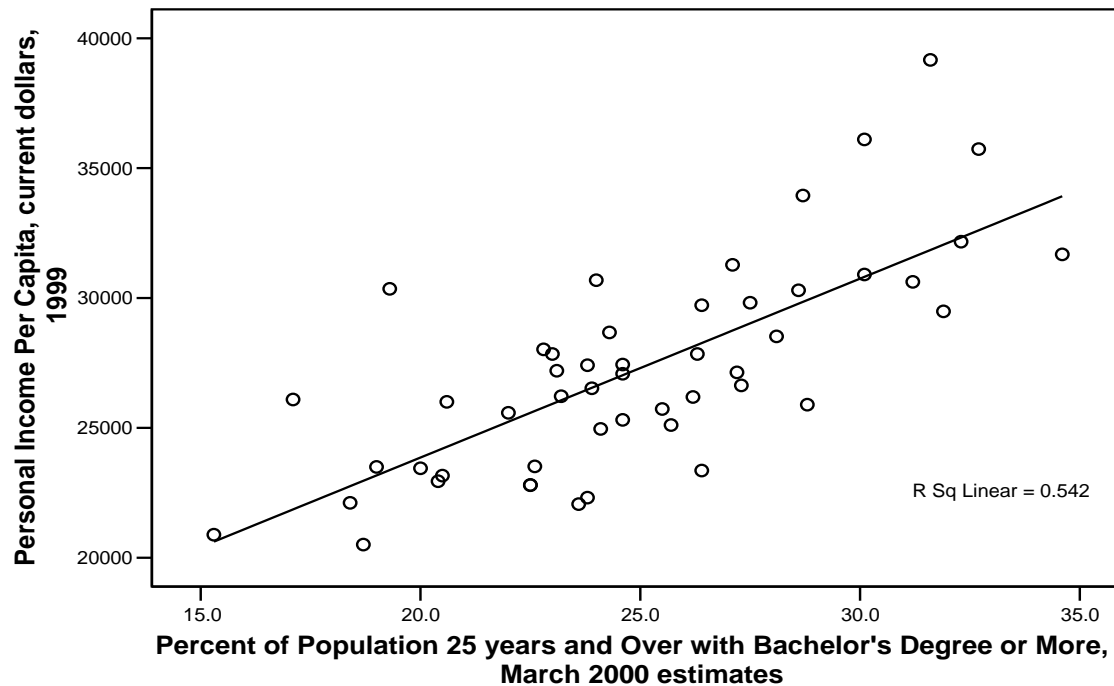
Percent of Population with Bachelor's Degree by Personal Income Per Capita



Regression Line

- Regression line is the best straight line

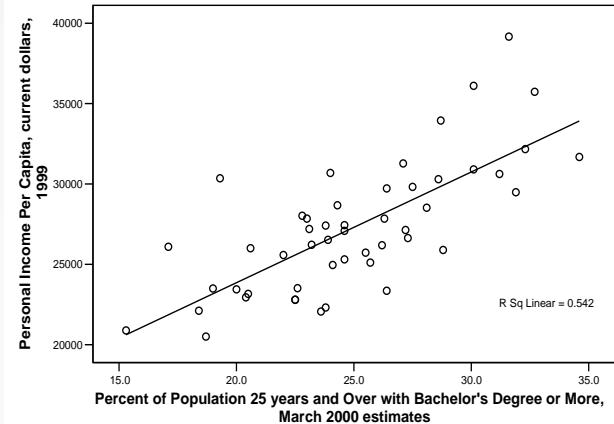
Percent of Population with Bachelor's Degree by Personal Income Per Capita



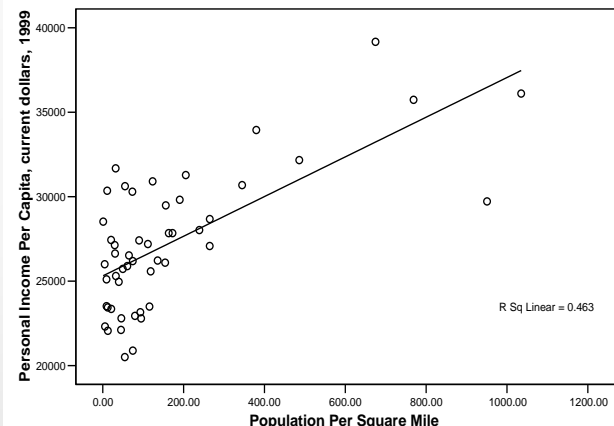
Regression Table

- The regression coefficient is not a good indicator for the strength of the relationship.
- Two scatter plots with very different dispersions could produce the same regression line.

Percent of Population with Bachelor's Degree by Personal Income Per Capita



Percent of Population with Bachelor's Degree by Personal Income Per Capita



Regression coefficient

- The regression coefficient is the **slope of the regression line** and tells you what the nature of the relationship between the variables is.
- The larger the regression coefficient the more change.

Nonlinear Regression

- Some **nonlinear models** can be modeled by a **polynomial function**
- A polynomial regression model can be transformed into linear regression model. For example,

$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

convertible to linear with new variables: $x_2 = x^2$, $x_3 = x^3$

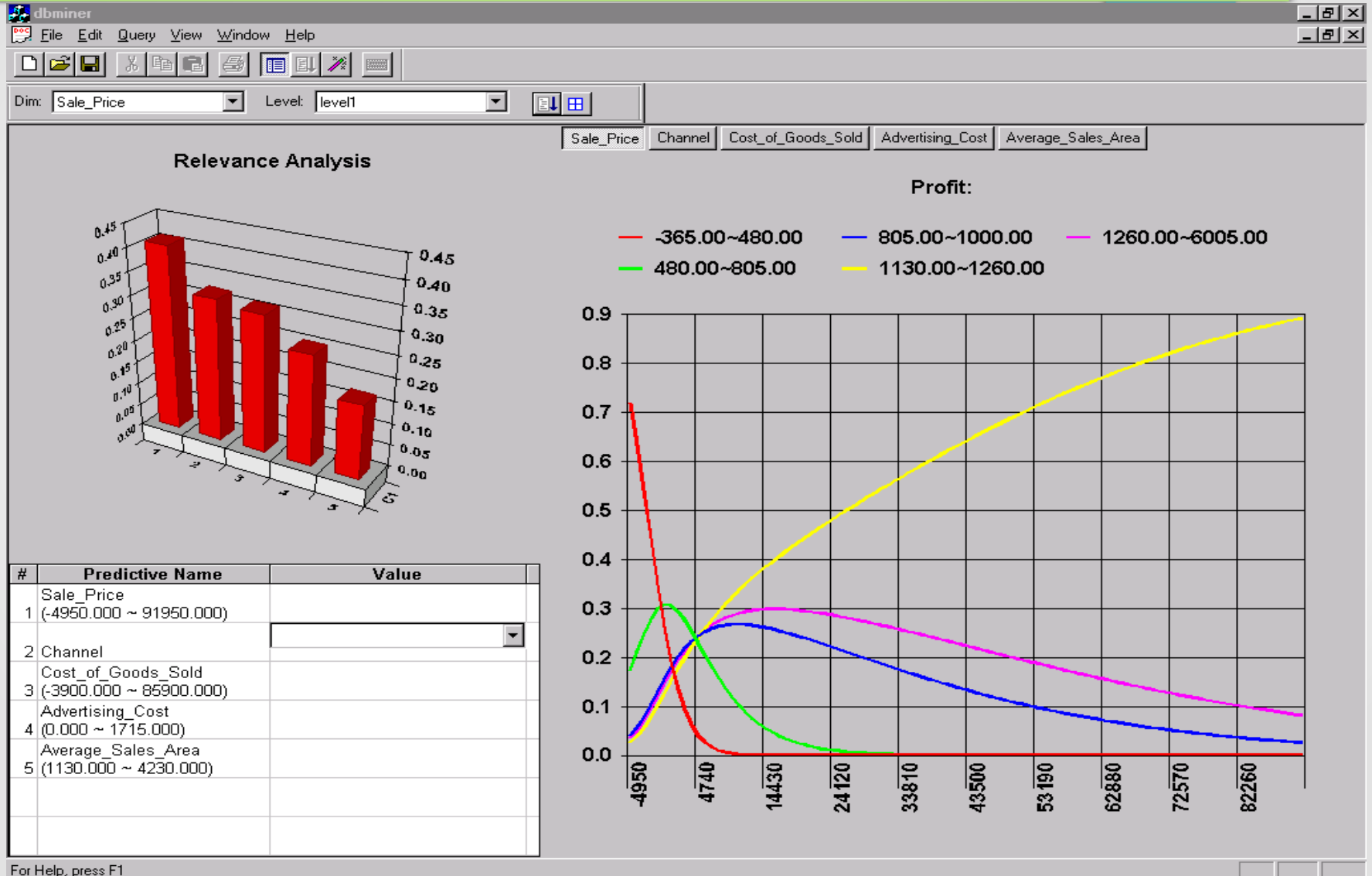
$$y = w_0 + w_1 x + w_2 x_2 + w_3 x_3$$

- Other functions, such as power function, can also be transformed to linear model

Regression Trees and Model Trees

- Regression trees and model trees
 - Trees to predict continuous values rather than class labels
- Regression tree: proposed in CART system (Breiman et al. 1984)
 - CART: Classification And Regression Trees
 - Each leaf stores a *continuous-valued prediction*
 - It is the *average value of the training tuples that reach the leaf*
- Model tree: proposed by Quinlan (1992)
 - Each leaf holds a regression
 - A more general case than regression tree

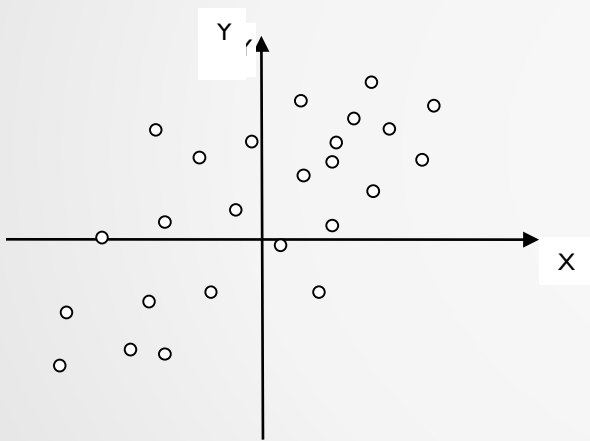
Prediction: Numerical Data



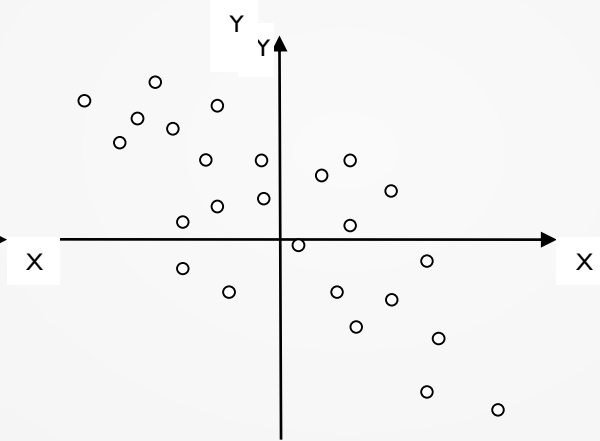
The relationship between x and y

- Correlation: is there a relationship between 2 variables?
- Regression: how well a certain independent variable predict dependent variable?

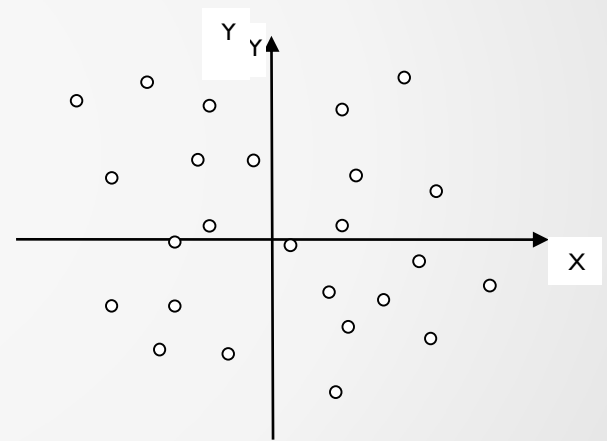
Scattergrams



Positive correlation



Negative correlation



No correlation

Variance vs Covariance

- Do two variables change together?

Variance:

- Gives information on variability of a single variable.

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

Covariance:

- Gives information on the degree to which two variables vary together.
- Note how similar the covariance is to variance: the equation simply multiplies x's error scores by y's error scores as opposed to squaring x's error scores.

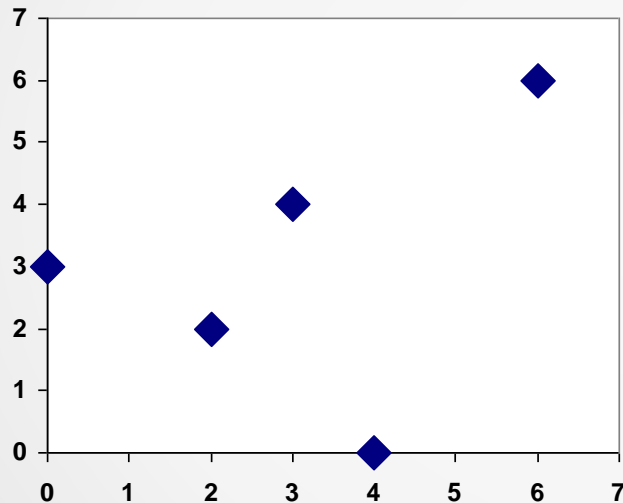
$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

Covariance

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

- When $X \uparrow$ and $Y \uparrow$: $\text{cov}(x, y) = \text{pos.}$
- When $X \downarrow$ and $Y \uparrow$: $\text{cov}(x, y) = \text{neg.}$
- When no constant relationship: $\text{cov}(x, y) = 0$

Example Covariance



x	y	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$
0	3	-3	0	0
2	2	-1	-1	1
3	4	0	1	0
4	0	1	-3	-3
6	6	3	3	9
$\bar{x} = 3$	$\bar{y} = 3$			$\Sigma = 7$

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1} = \frac{7}{4} = 1.75$$

What does this number tell us?

Problem with Covariance:

- The value obtained by covariance is dependent on the **size of the data's standard deviations**: if large, the value will be greater than if small... *even if the relationship between x and y is exactly the same in the large versus small standard deviation datasets.*

Example of how covariance value relies on variance

	High variance data				Low variance data		
Subject	x	y	x error * y error		x	y	X error * y error
1	101	100	2500		54	53	9
2	81	80	900		53	52	4
3	61	60	100		52	51	1
4	51	50	0		51	50	0
5	41	40	100		50	49	1
6	21	20	900		49	48	4
7	1	0	2500		48	47	9
Mean	51	50			51	50	
Sum of x error * y error :			7000		Sum of x error * y error :		28
Covariance:			1166.67		Covariance:		4.67

Solution: Pearson's r

- Covariance does not really tell us anything

» *Solution: standardise this measure*

- Pearson's R: standardises the covariance value.
- Divides the covariance by the multiplied standard deviations of X and Y:

$$r_{xy} = \frac{\text{COV}(x, y)}{s_x s_y}$$

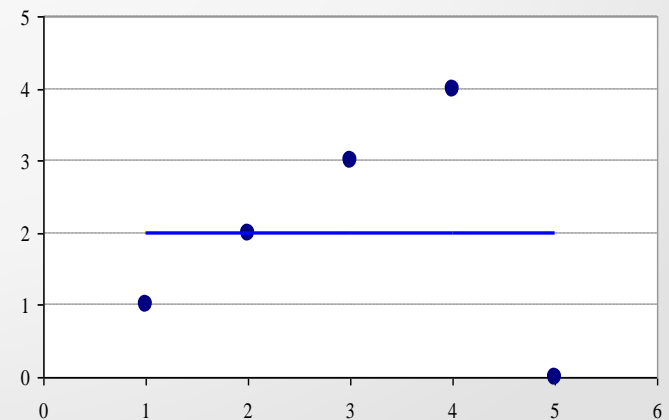
Pearson's R continued

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1} \quad \rightarrow \quad r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}$$

$$r_{xy} = \frac{\sum_{i=1}^n Z_{x_i} * Z_{y_i}}{n-1}$$

Limitations of r

- When $r = 1$ or $r = -1$:
 - We can predict y from x with certainty
 - all data points are on a straight line: $y = ax + b$
- r is actually \hat{r}
 - r = true \hat{r} of whole population
 - \hat{r} = estimate of r based on data
- r is very sensitive to extreme values:

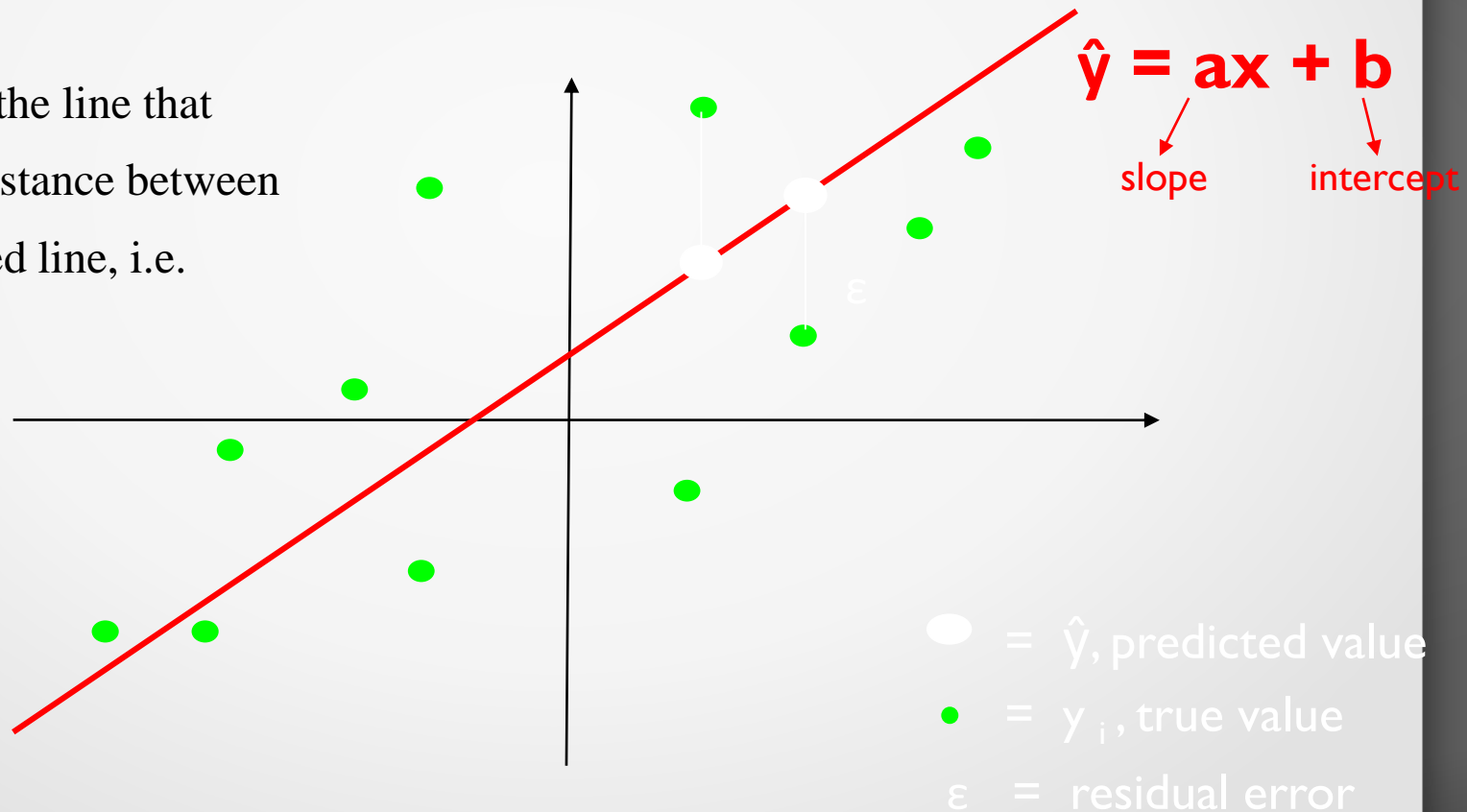


Regression

- **Correlation** tells you if there is an association between x and y but it **doesn't** describe the relationship or **allow you to predict one variable from the other.**
- To do this we need REGRESSION!

Best-fit Line

- Aim of linear regression is to fit a straight line, $\hat{y} = ax + b$, to data that gives best prediction of y for any value of x
- This will be the line that minimises distance between data and fitted line, i.e. the residuals



Least Squares Regression

- To find the best line we must minimise the sum of the squares of the residuals (the vertical distances from the data points to our line)

Model line: $\hat{y} = ax + b$ $a = \text{slope}, b = \text{intercept}$

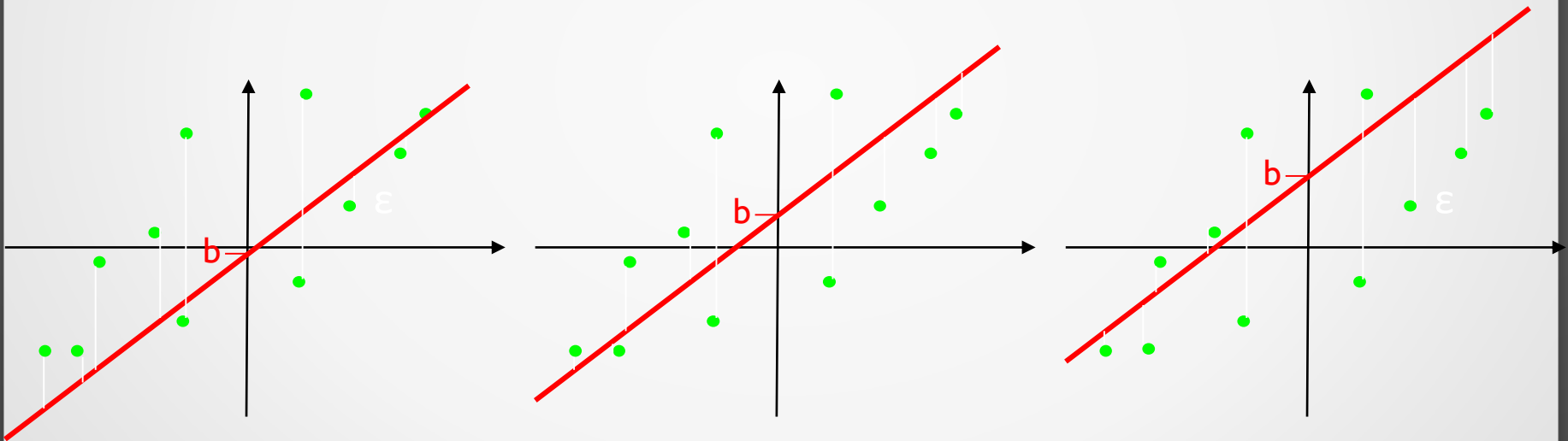
Residual (ε) = $y - \hat{y}$

Sum of squares of residuals = $\sum (y - \hat{y})^2$

- we must find values of a and b that minimise
 $\sum (y - \hat{y})^2$

Finding b

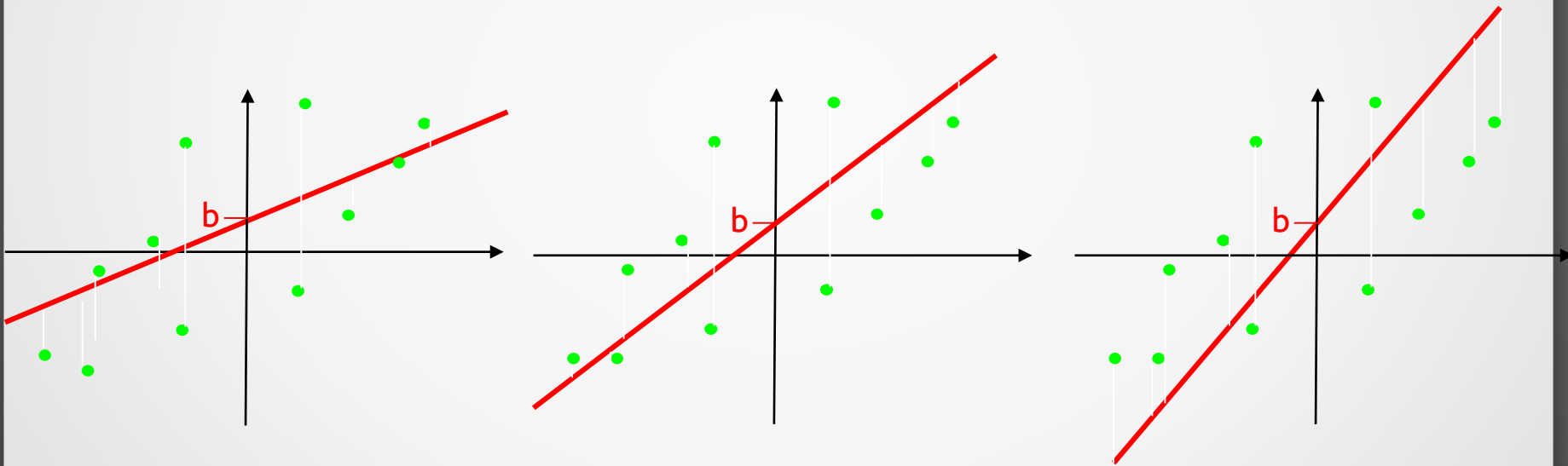
- First we find the value of b that gives the min sum of squares



- Trying different values of b is equivalent to shifting the line up and down the scatter plot

Finding a

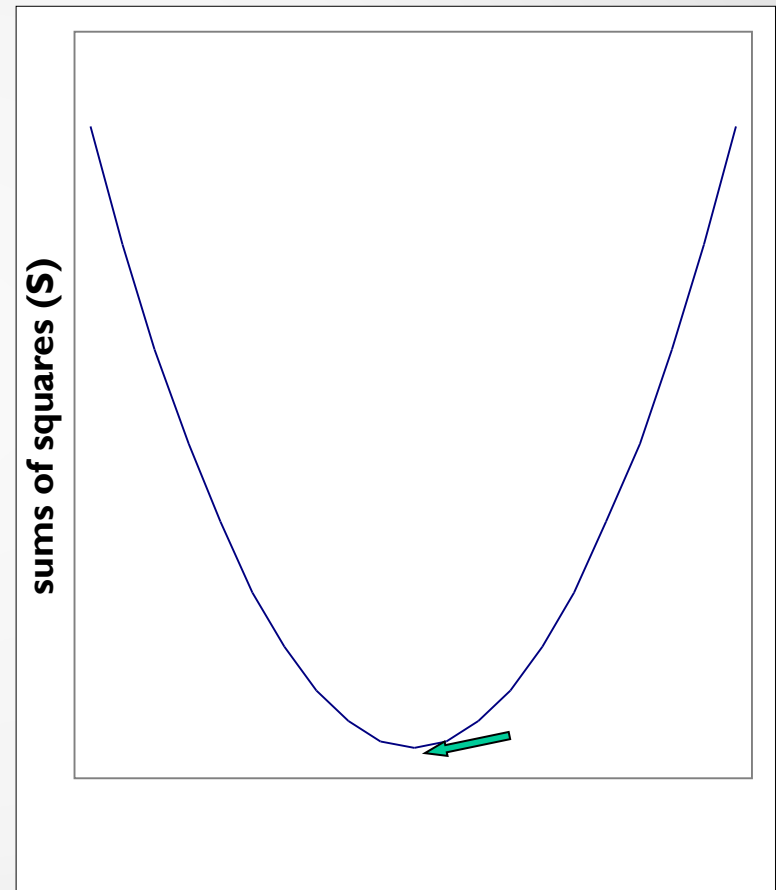
- Now we find the value of a that gives the min sum of squares



- Trying out different values of a is equivalent to changing the slope of the line, while b stays constant

Minimising sums of squares

- Need to minimise $\Sigma(y - \hat{y})^2$
- $\hat{y} = ax + b$
- so need to minimise:
$$\Sigma(y - ax - b)^2$$
- If we plot the sums of squares for all different values of a and b we get a parabola, because it is a squared term
- So the min sum of squares is at the bottom of the curve, where the gradient is zero.



The maths bit

- The min sum of squares is at the bottom of the curve where the gradient = 0
- So we can find a and b that give min sum of squares by taking partial derivatives of $\Sigma(y - ax - b)^2$ with respect to x and y separately
- Then we solve these for 0 to give us the values of a and b that give the min sum of squares

The solution

- Doing this gives the following equations for a and b:

$$a = \frac{r s_y}{s_x}$$

r = correlation coefficient of x and y

s_y = standard deviation of y

s_x = standard deviation of x

■ From you can see that:

- A low correlation coefficient gives a flatter slope (small value of a)
- Large spread of y , i.e. high standard deviation, results in a steeper slope (high value of a)
- Large spread of x , i.e. high standard deviation, results in a flatter slope (low value of a)

The solution cont.

- Our model equation is $\hat{y} = ax + b$
- This line must pass through the mean so:

$$\bar{y} = a\bar{x} + b \quad \longrightarrow \quad b = \bar{y} - a\bar{x}$$

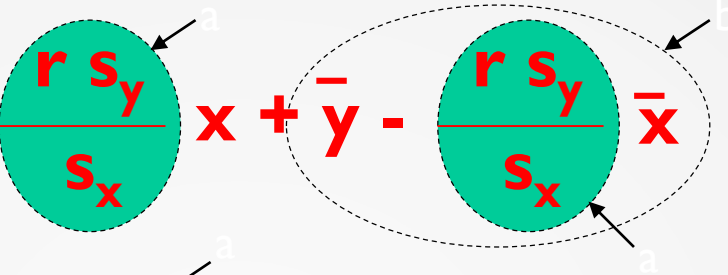
- We can put our equation for a into this giving:

$$b = \bar{y} - \frac{r s_y}{s_x} \bar{x}$$

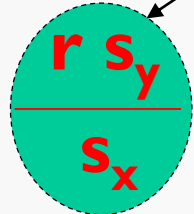
r = correlation coefficient of x and y
 s_y = standard deviation of y
 s_x = standard deviation of x

- The smaller the correlation, the closer the intercept is to the mean of y

Back to the model

$$\hat{y} = ax + b = \frac{r s_y}{s_x} x + \bar{y} - \frac{r s_y}{s_x} \bar{x}$$


Rearranges to:

$$\hat{y} = \frac{r s_y}{s_x} (x - \bar{x}) + \bar{y}$$


- If the correlation is zero, we will simply predict the mean of y for every value of x, and our regression line is just a flat straight line crossing the x-axis at y
- But this isn't very useful.
- We can calculate the regression line for any data, but the important question is how well does this line fit the data, or how good is it at predicting y from x

How good is our model?

- Total variance of y:

$$s_y^2 = \frac{\sum (y - \bar{y})^2}{n - 1} = \frac{SS_y}{df_y}$$

- Variance of predicted y values (\hat{y}):

$$s_{\hat{y}}^2 = \frac{\sum (\hat{y} - \bar{y})^2}{n - 1} = \frac{SS_{\text{pred}}}{df_{\hat{y}}}$$

This is the variance explained by our regression model

- Error variance:

$$s_{\text{error}}^2 = \frac{\sum (y - \hat{y})^2}{n - 2} = \frac{SS_{\text{er}}}{df_{\text{er}}}$$

This is the variance of the error between our predicted y values and the actual y values, and thus is the variance in y that is NOT explained by the regression model

How good is our model cont.

- Total variance = predicted variance + error variance

$$s_y^2 = s_{\hat{y}}^2 + s_{er}^2$$

- Conveniently, via some complicated rearranging

$$s_{\hat{y}}^2 = r^2 s_y^2$$



$$r^2 = s_{\hat{y}}^2 / s_y^2$$

- so r^2 is the proportion of the variance in y that is explained by our regression model

How good is our model cont.

- Insert $r^2 s_y^2$ into $s_y^2 = s_{\hat{y}}^2 + s_{er}^2$ and rearrange to get:

—

$$\begin{aligned} s_{er}^2 &= s_y^2 - r^2 s_y^2 \\ &= s_y^2 (1 - r^2) \end{aligned}$$

- From this we can see that the greater the correlation the smaller the error variance, so the better our prediction

Is the model significant?

- i.e. do we get a significantly better prediction of y from our regression equation than by just predicting the mean?

- F-statistic:

$$F_{(df_{\hat{y}}, df_{er})} = \frac{s_{\hat{y}}^2}{s_{er}^2} \overset{\substack{\text{complicated} \\ \text{rearranging}}}{= \dots =} \frac{r^2 (n - 2)^2}{1 - r^2}$$

■ And it follows that:

(because $F = t^2$)
$$t_{(n-2)} = \frac{r (n - 2)}{\sqrt{1 - r^2}}$$

So all we need to know are r and n

General Linear Model

- Linear regression is actually a form of the General Linear Model where the parameters are a , the slope of the line, and b , the intercept.

$$y = ax + b + \varepsilon$$

- A General Linear Model is just any model that describes the data in terms of a straight line

Multiple regression

- Multiple regression is used to determine the effect of a number of independent variables, x_1 , x_2 , x_3 etc, on a single dependent variable, y
- The different x variables are combined in a linear way and each has its own regression coefficient:

$$y = a_1x_1 + a_2x_2 + \dots + a_nx_n + b + \varepsilon$$

- The a parameters reflect the independent contribution of each independent variable, x , to the value of the dependent variable, y .
- i.e. the amount of variance in y that is accounted for by each x variable after all the other x variables have been accounted for

SPM

- Linear regression is a GLM that models the effect of one independent variable, x , on ONE dependent variable, y
- Multiple Regression models the effect of several independent variables, x_1, x_2 etc, on ONE dependent variable, y
- Both are types of General Linear Model
- GLM can also allow you to analyse the effects of several independent x variables on several dependent variables, y_1, y_2, y_3 etc, in a linear combination

Regression in R

R has the `lm()` and `glm()` functions to build a linear regression model, which is used for predicting continuous data.

> `lm(formula, data)`

where `data` represents the variable that contains the data set and `formula` describes the model. The formula should be given in a specific format; that is, `varY~varX` in simple linear regression, where `varY` is the dependent or predicted variable and `varX` is the independent or predictor variable.

>`glm(formula, family=familytype, data=)`

where `data` is the variable that contains the data set; `formula` describes the model and `family` takes the values “binomial”, “gaussian”, “gamma”, “poisson”, “quasi”, “quasipoisson”, “quasibinomial” and “inverse gaussian” to perform logistic and other regressions/classifications.

Regression in R

Example:- Assume that we have the height (x) data in inches of 40 adults. We can use the `rnorm()` function to generate a vector of heights with a mean of 80 (roughly the population mean), and a standard deviation of 5 inches. Similarly, weight (y) can be generated using `rnorm()` with a mean of 70 and standard deviation of 6 inches.

```
> x <- sort(rnorm(40, 80, 5))
```

```
> y <- sort(rnorm(40, 70, 6))
```



> x

**[1] 72.29567 72.66903 73.09784 73.50450 73.64954
73.76746 73.86345 74.34988 74.71307**

>y

**[1] 58.22440 58.51788 59.53612 62.04827 62.71820
64.31005 64.34964 64.37748 64.89418**

Regression in R

```
>cor(x,y)
```

```
[1] 0.9757246
```

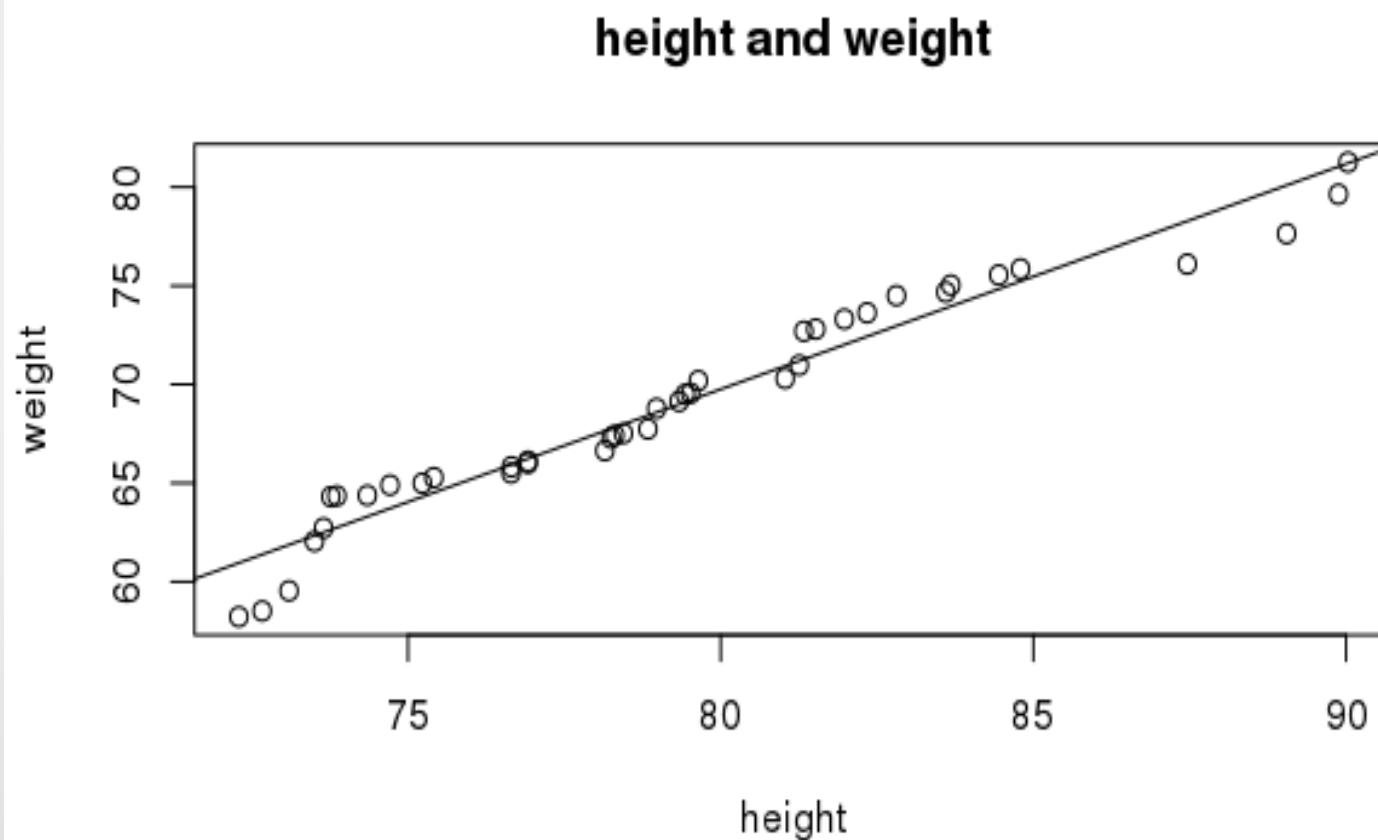
The correlation of x and y shows that there is a strong positive linear relation between height and weight.

```
>plot(x, y, xlab = "height", ylab = "weight", main = "height and weight")
```

```
>abline(lm(y ~ x))
```

lm() function is used to build linear regression model. Here, y (weight) is the dependent variable and x (height) is the independent variable.

Regression in R



Regression in R

Using the `summary()` function for the model, we can obtain information such as the formula called, the error in our model, the values of the coefficients and their significance. We can also access information on the overall performance of the model with the adjusted R-squared (.9876 in our case) that represent the amount of variation in *y* explained by *x*; so 98% of the variation in 'weight (*y*)' can be explain by the variable 'height (*x*)'.

```
> m1<-lm(y~x)
```

```
> summary(m1)
```

Call:

```
lm(formula = y ~ x)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.2938	-0.3970	0.2273	0.3721	0.8682

Coefficients:

Regression in R

The prediction of value m1 can be done using function predict().

```
> predict(m1)
```

1	2	3	4	5	6	7	8	9
59.43206	60.90593	63.35430	63.56232	63.82316	63.84543	64.27267		
64.67051	64.70872							

10	11	12	13	14	15	16	17	18
64.83711	65.13128	65.22970	65.23072	65.32096	67.76135	68.32733		
68.58346	68.62780							

19	20	21	22	23	24	25	26	27
68.65912	68.84693	68.97648	69.49723	70.01205	70.03058	70.04389		
70.15092	70.15788							

28	29	30	31	32	33	34	35	36

37	38	39	40	41	42	43	44	45
70.92202	71.32028	72.18010	72.85014	74.17816	74.95200	75.25782		

Logistic Regression in R

Linear regression is used to model data when Y is a continuous variable. However, we cannot use it when the output variable is binary categorical.

Logistic regression can be used to model applications where the output is only 0 or 1. Such problems are called binary classification problems. For example, to identify an email as spam or not, review comments about a movie (good or bad), if a credit card transaction is fraudulent or not, whether the given tissue sample is malignant or not, and so on.

Logistic Regression in R

Logistic regression can be built using the `glm()` function, the family argument set to "binomial".

Implementation in R:

Step 1: Build the logistic regression model.

```
>logMod <- glm(Y ~ X1 + X2, family="binomial", data =  
  trainingData)
```

Step 2: Predict Y.

```
>predY <- predict(logMod, testData, type="response")
```


Logistic Regression in R

Sample code using the breast cancer dataset:

```
>install.packages("mlbench")
>data(BreastCancer, package="mlbench")
>bc <- BreastCancer[complete.cases(BreastCancer), ] # create copy
>head(bc)
  >trainData <- bc[100, ]
  >testData <- bc[-100, ]
  >logitmod <- glm(Class ~ Cl.thickness + Cell.size + Cell.shape, family =
  "binomial", data=bc)
  >summary(logitmod)
  >pred <- predict(logitmod, newdata = testData, type = "response")
  >y_pred_num <- ifelse(pred > 0.5, 1, 0)
  >y_pred <- factor(y_pred_num, levels=c(0, 1))
  >y_act <- testData$Class
```

Logistic Regression in R

The output of some actual and predicted values are as follows:

```
>y_act
```

```
[1] benign  benign  benign  benign  benign  malignant benign  
benign
```

```
 [9] benign  benign  benign  benign  malignant benign  
malignant malignant
```

```
[17] benign  benign  malignant benign  malignant malignant  
benign  benign...
```

```
[57] malignant malignant malignant benign  malignant malignant  
benign  malignant
```

```
Levels: benign malignant
```

```
> y_pred
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
0	0	0	1	0	1	0	0	0	0	0	0	1	0	1	1	0	0	1	0	1
22	23	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	42		

Polynomial Regression

- Polynomial Linear Regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y ,
- Polynomial regression fits a nonlinear model to the data, as a statistical estimation problem it is linear, in the sense that the regression function is linear in the unknown parameters that are estimated from the data.
- Polynomial regression is considered to be a special case of multiple linear regression.

How to Handle Non-Linear Effects

Power transformations can make simple monotone relationships more linear (Fig A). Polynomial regression (or other transformations, e.g. logit) is often needed for more complex relationships (Figs, B & C)

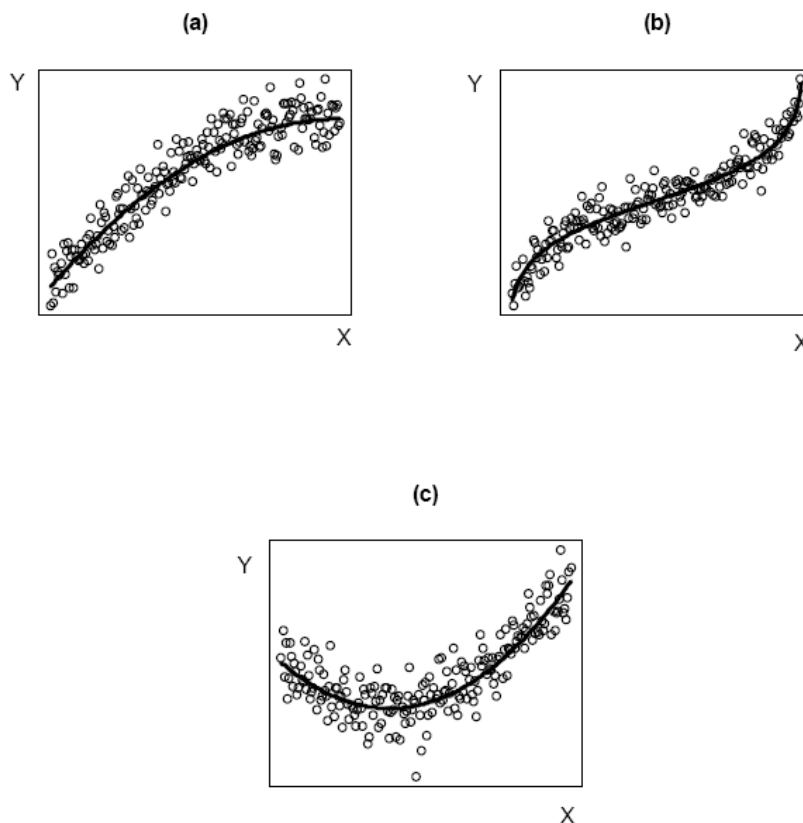


Figure 17. (a) A simple monotone relationship. (b) A monotone relationship that is not simple. (c) A simple nonmonotone relationship.

How to Handle Non-Linear Effects

- Simple monotonic relationships: Power transformations.
- Polynomial Regression- Quantitative Variables
- [Polynomial Regression- Categorical Variables]
- Generalized Linear Models (e.g., logistic regression)

Non-Linear Effects in MR/GLM

Multiple regression/GLM is “*linear in the regressors*”

The predicted score is a linear combination of the regressors (X's) in the model

Each regressor is multiplied by its coefficient and added together (+ intercept/constant)

$$Y' = b_1X_1 + b_2X_2 + \dots + b_0$$

Linear vs. Logistic Regression

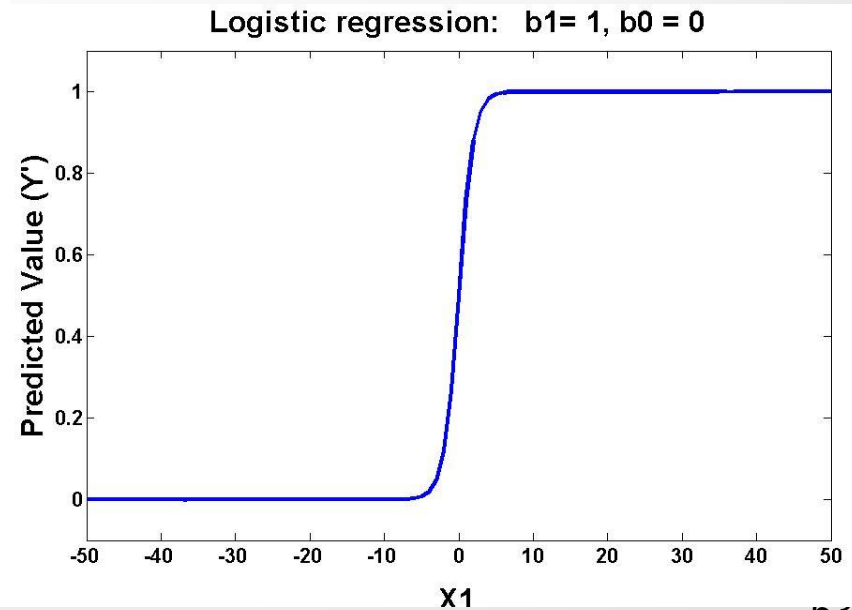
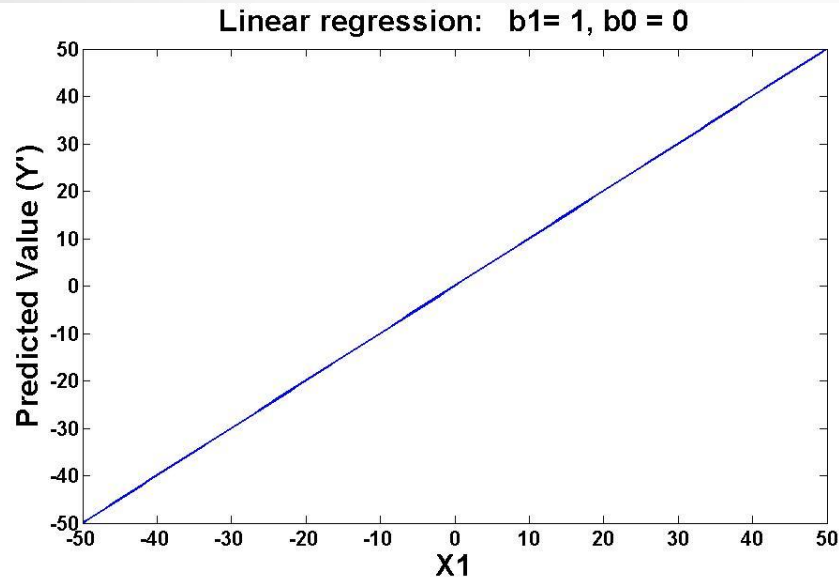
Multiple Regression

$$Y' = b_0 + b_1 X_1$$

Logistic Regression

$$Y' = \frac{e^{b_0 + b_1 X_1}}{(1 + e^{b_0 + b_1 X_1})}$$

Predicted values graphed for $X_1 = -50$ to 50



Polynomial Regression

$$Y' = A + BX + CX^2 + DX^3 + \dots QX^{N-1}$$

It is important to distinguish between regressors in the model vs. variables of interest.

In this example there is only one variable of interest. The powers of X act as a structural set of regressors to allow for non-linear relationships between this variable of interest & Y .

However, the model is still linear in the regressors. I.E. linear combination of the regressors multiplied by their parameter estimates.

Polynomial Regression Order

$$Y' = A + BX + CX^2 + DX^3 + \dots QX^{N-1}$$

If you include (N-1) regressors based on X, you will perfectly fit the data.

The order of the equation is the highest power: (N-1) in this example.

$X^{(N-1)}$ is the highest order predictor. All other regressors are lower order.

The highest order regressor determines the overall shape of the relationship within the range of $-\infty$ to ∞

Polynomial Regression Shape

$$Y' = A + BX + CX^2 + DX^3 + \dots QX^{N-1}$$

The highest order regressor determines the overall shape of the relationship within the range of $-\infty$ to ∞

Linear

$$Y' = A + BX$$

Zero bends

Quadratic

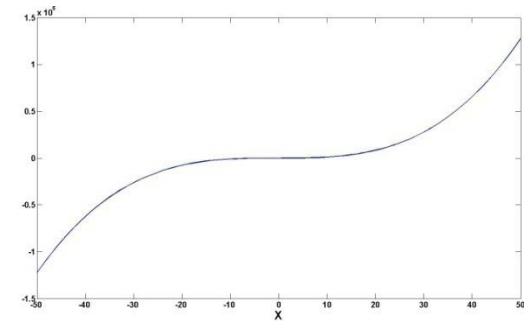
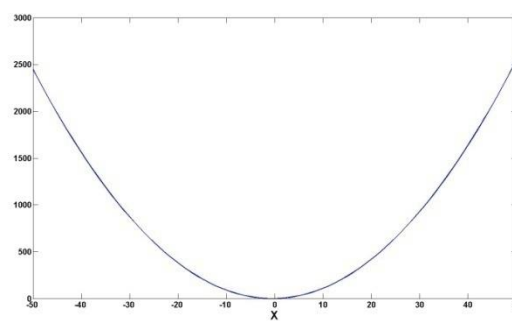
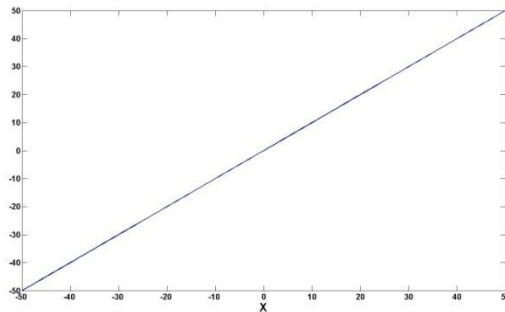
$$Y' = A + BX + CX^2$$

One bend

Cubic

$$Y' = A + BX + CX^2 + DX^3$$

Two bends



Shape and Coefficient Sign

The sign of the coefficient for the highest order regressor determines the direction of the curvature

Linear

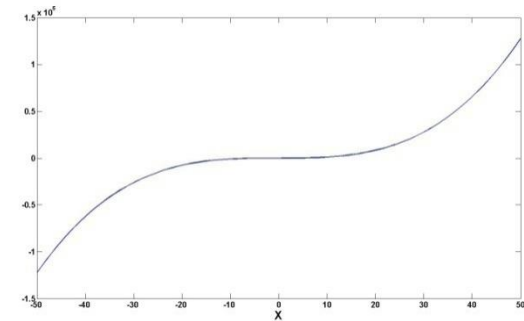
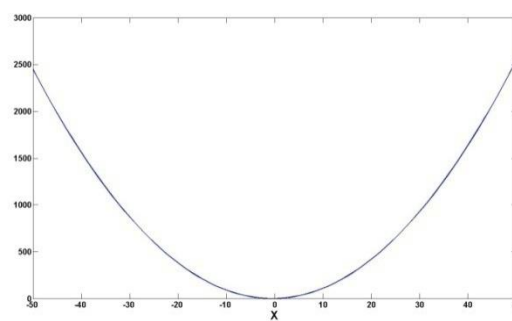
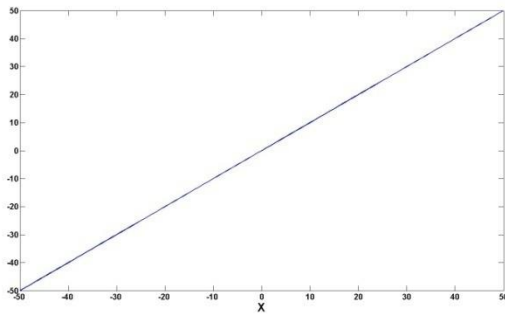
$$Y' = 0 + 1X$$

Quadratic

$$Y' = 0 + 1X + 1X^2$$

Cubic

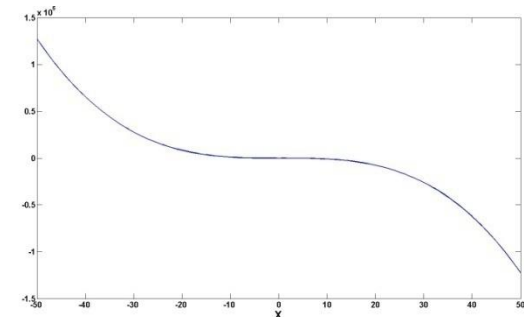
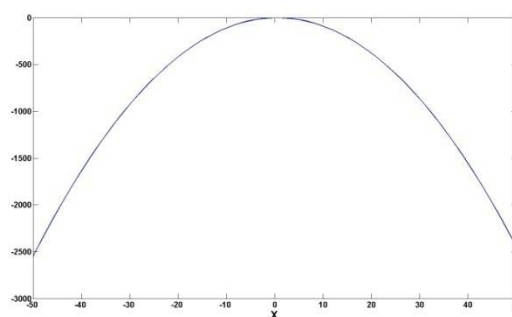
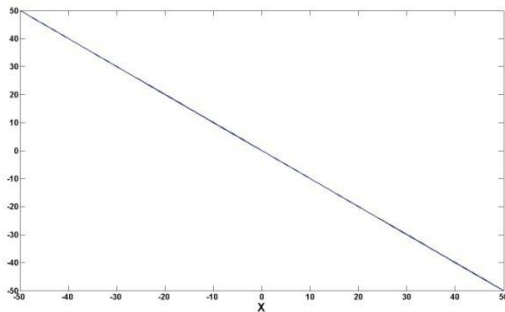
$$Y' = 0 + 1X + 1X^2 + 1X^3$$



$$Y' = 0 + -1X$$

$$Y' = 0 + 1X + -1X^2$$

$$Y' = 0 + 1X + 1X^2 + -1X^3$$



How to Determine Order

Can fit order up to $N-1$ but wont

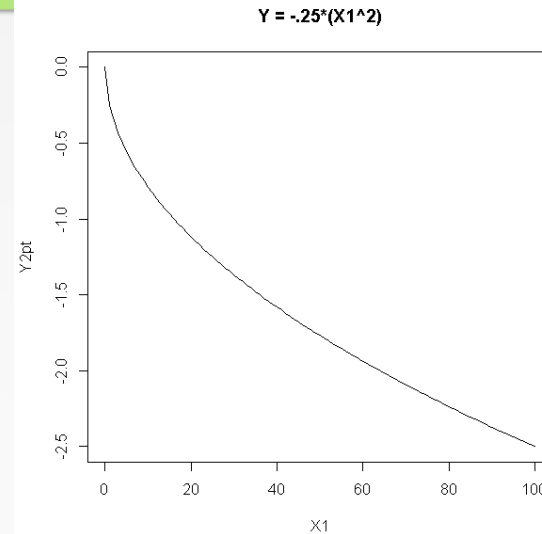
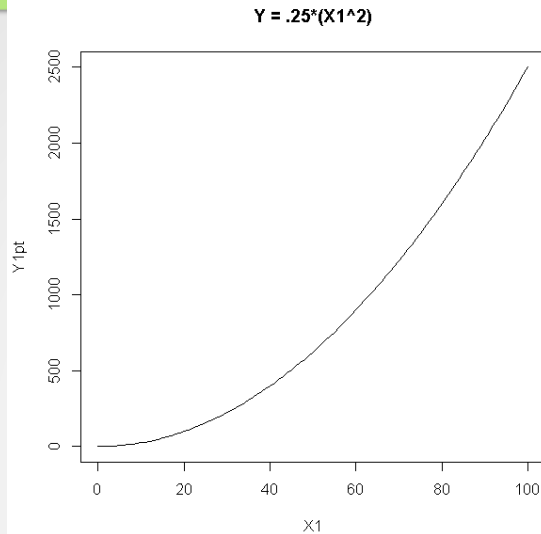
Theory should generally guide order

Social science theory rarely predicts higher than cubic (and typically not higher than quadratic)

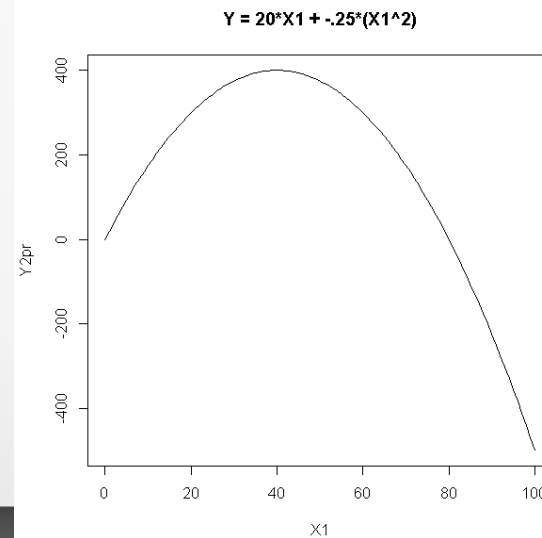
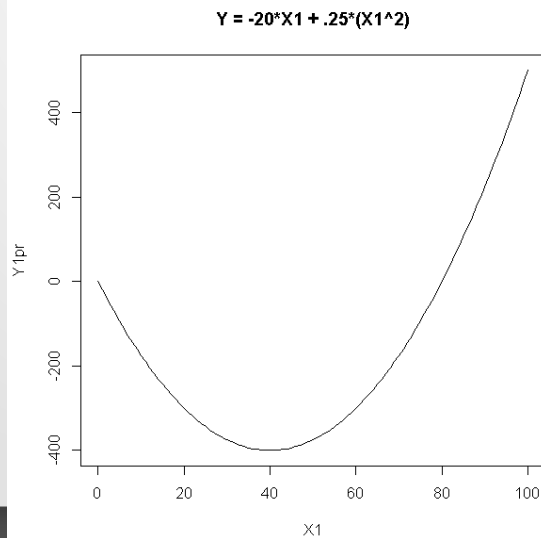
Sometimes higher order models (quadratic, cubic) are implicated by the distribution of residuals.

Polynomial vs. Power Transformation of X

Power Transformations of X



Polynomial Regression



Piecewise Polynomials/Splines

- Polynomials are good locally not globally.
- Piecewise polynomials use this to model data locally in many regions (governed by knots) to approx. global fit.
- Two main types of Splines:
 - 1) Regression Splines
 - 2) Smoothing Splines

Piecewise Polynomials/Splines

- **Regression Splines** = (# of knots) < (# of data points)
- No regularization, fit by LS, nice linear smoother properties

Cubic spline function with K knots:

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \sum_{k=1}^K b_k (x - \xi_k)_+^3$$

- But what order do we choose (linear, quadratic, cubic?)
- How many knots and where to place them?

Piecewise Polynomials/Splines

- **Smoothing Splines:**

$$\sum_{i=1}^N \{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 dt,$$

- Minimizing above quantity (for any function $f(x)$), leads to $f(x)$ having a functional form of a NATURAL cubic spline w/ knots at every data point

$$f(x) = \sum_{j=1}^N N_j(x) \theta_j,$$

- **Natural cubic splines:** Cubic regression splines (as talked about before + imposing linearity beyond the leftmost/rightmost knots).
- The N_j basis functions are derived from the cubic regression spline basis functions.