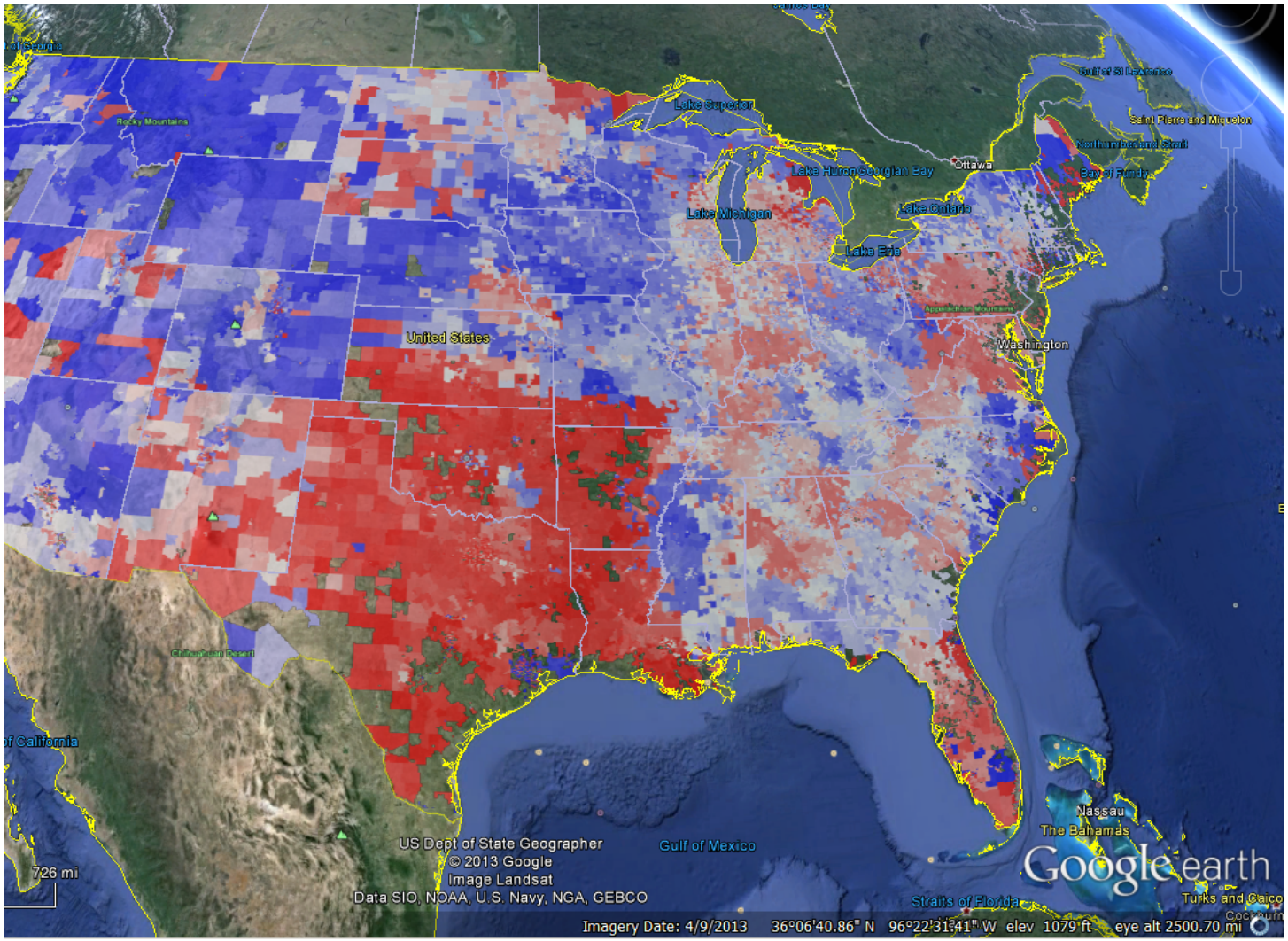
Data Science for Business
Discriminant Functions

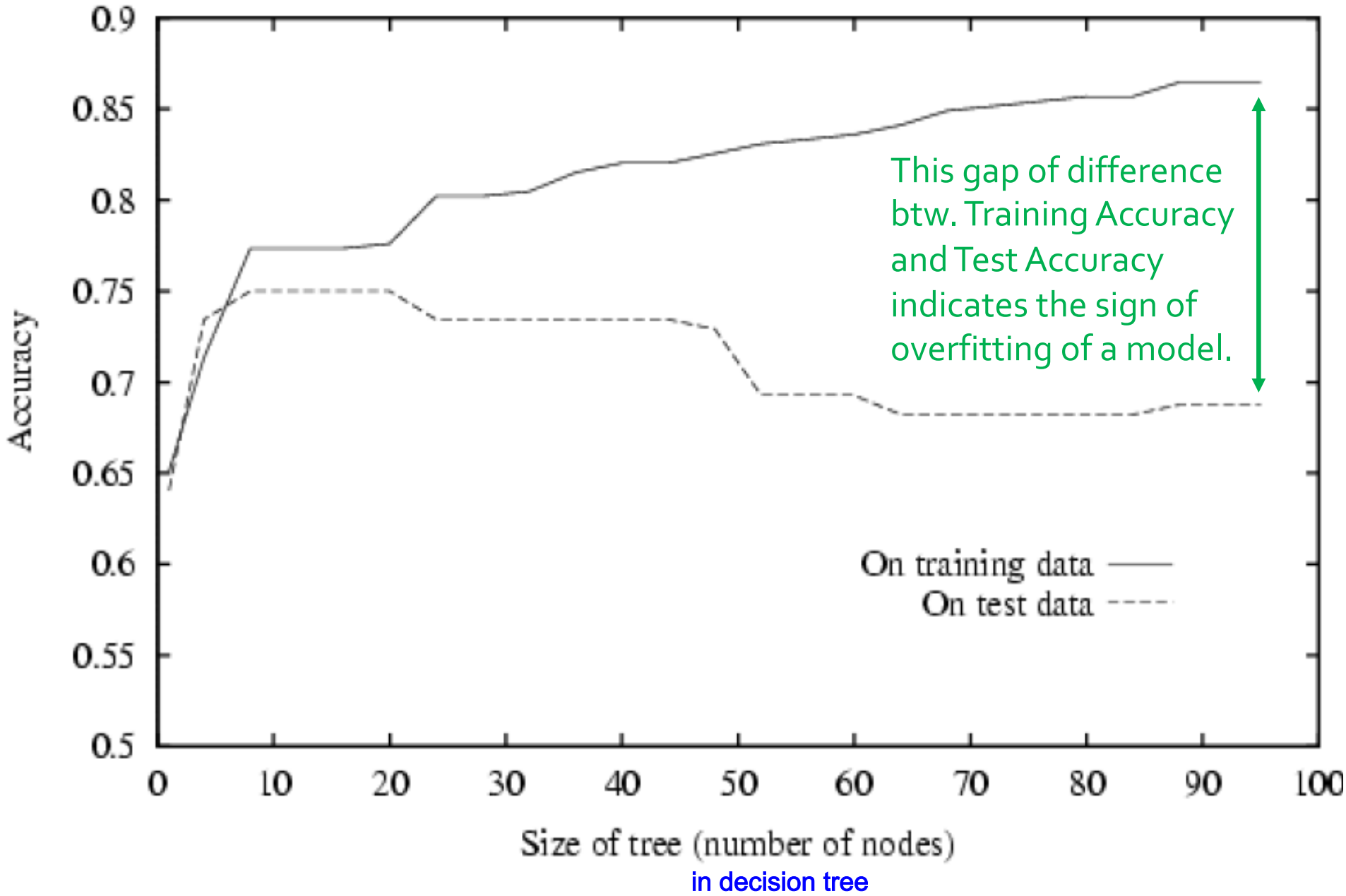
Asst. Prof. Teerapong Leelanupab (Ph.D.)
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang (KMITL)

Week 8.1

Heat map of XYZ Hotels geographic brand affinity

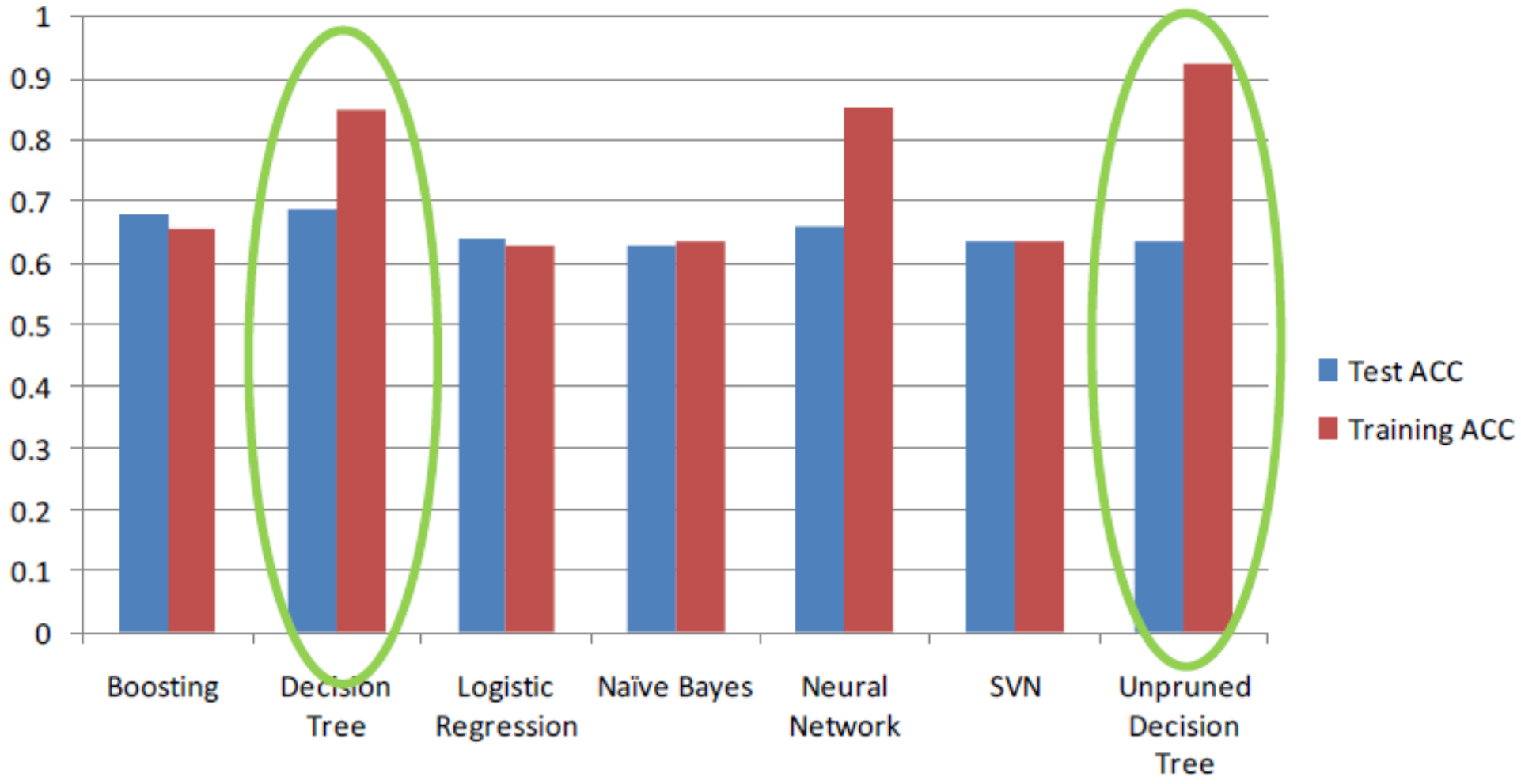


Tree Complexity and Overfitting



in decision tree

Trees on Churn

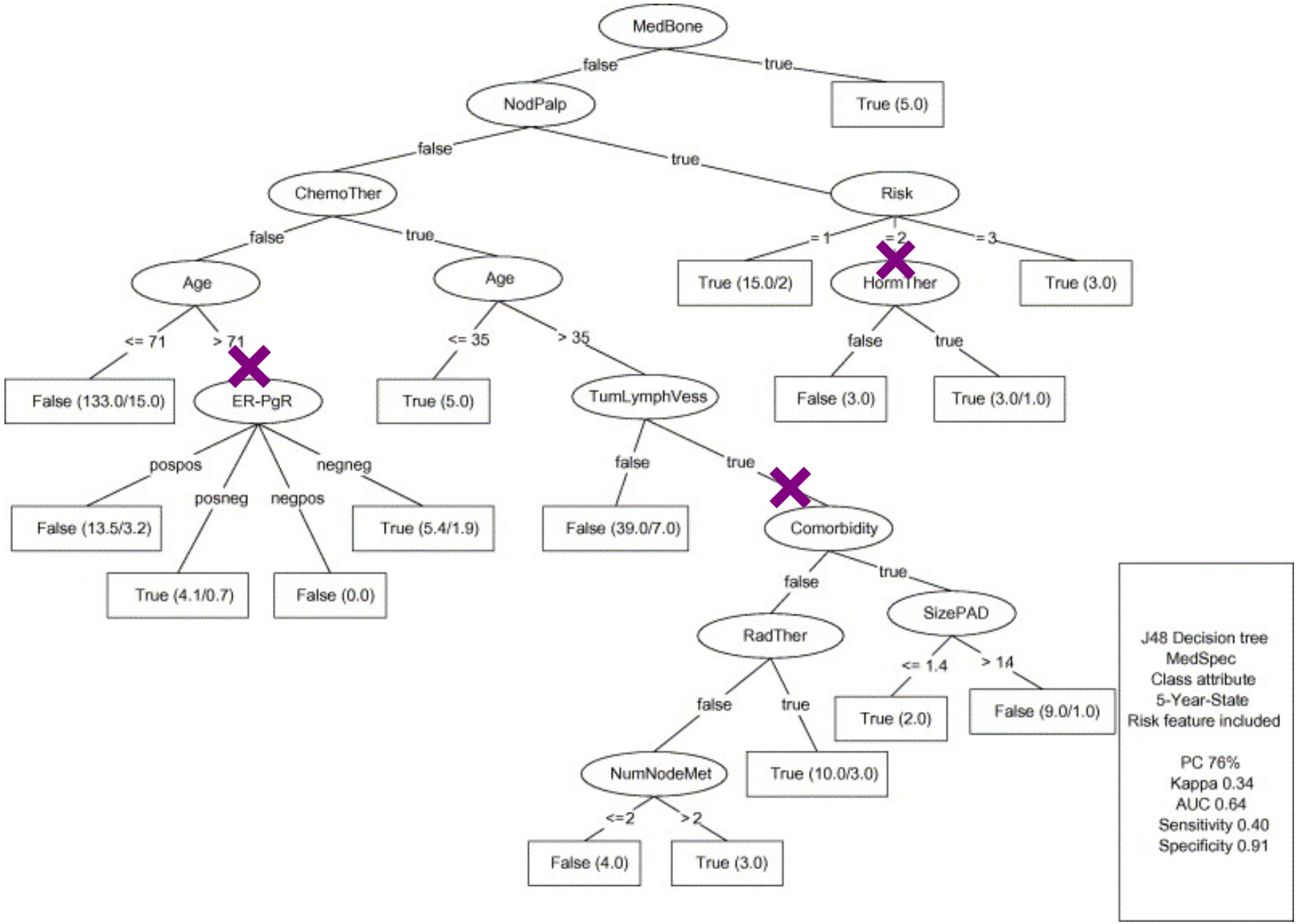


Pruning

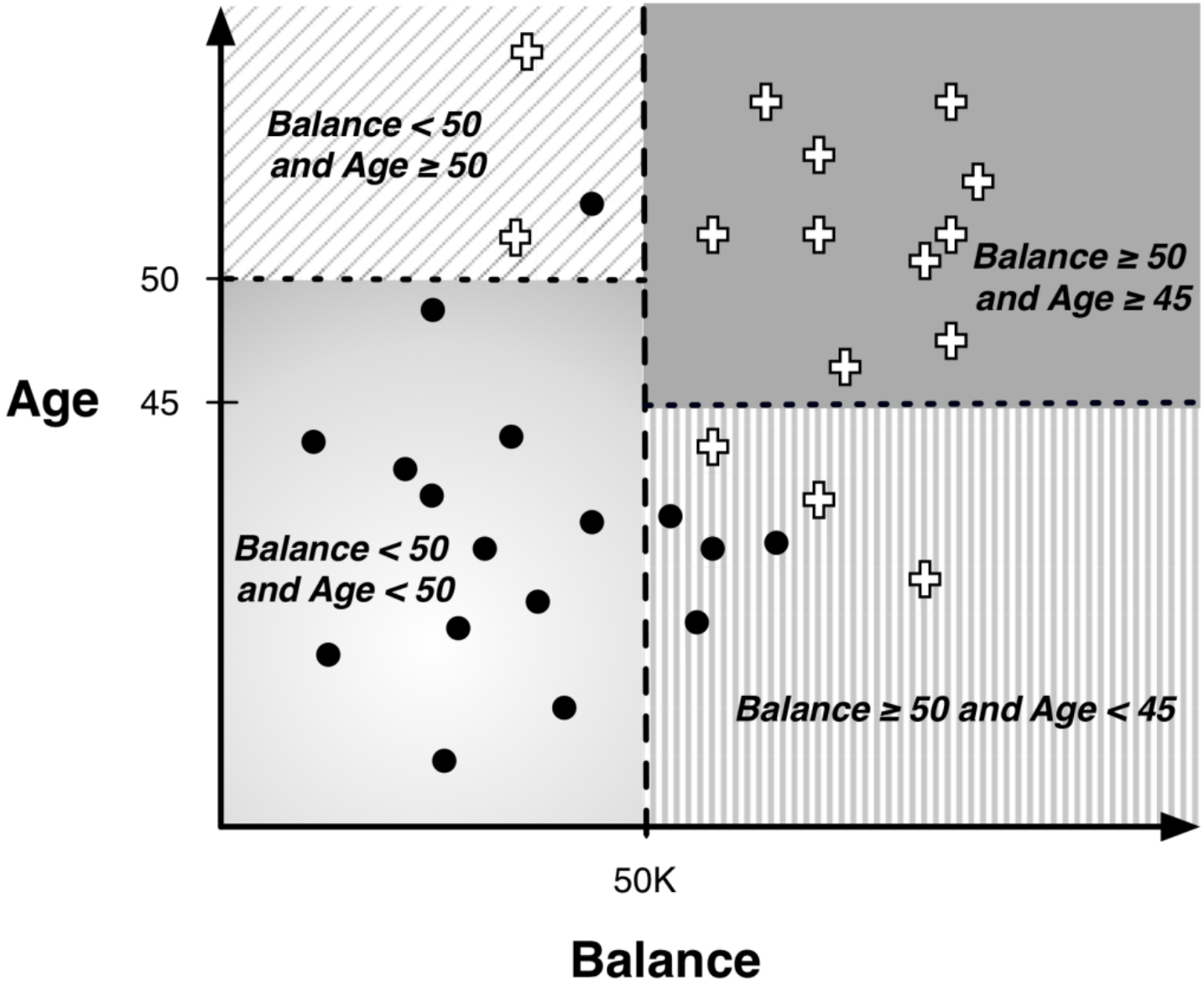
- Pruning simplifies a decision tree to prevent overfitting to noise in the data
- **Post-pruning**:
  - takes a fully-grown decision tree and discards unreliable parts
- **Pre-pruning**:
  - stops growing a branch when information becomes unreliable
- *Post-pruning* preferred in practice

สร้างก่อนแล้วค่อยตัดทิ้ง ดีกว่าตัดทิ้งแล้วสร้างเพิ่ม

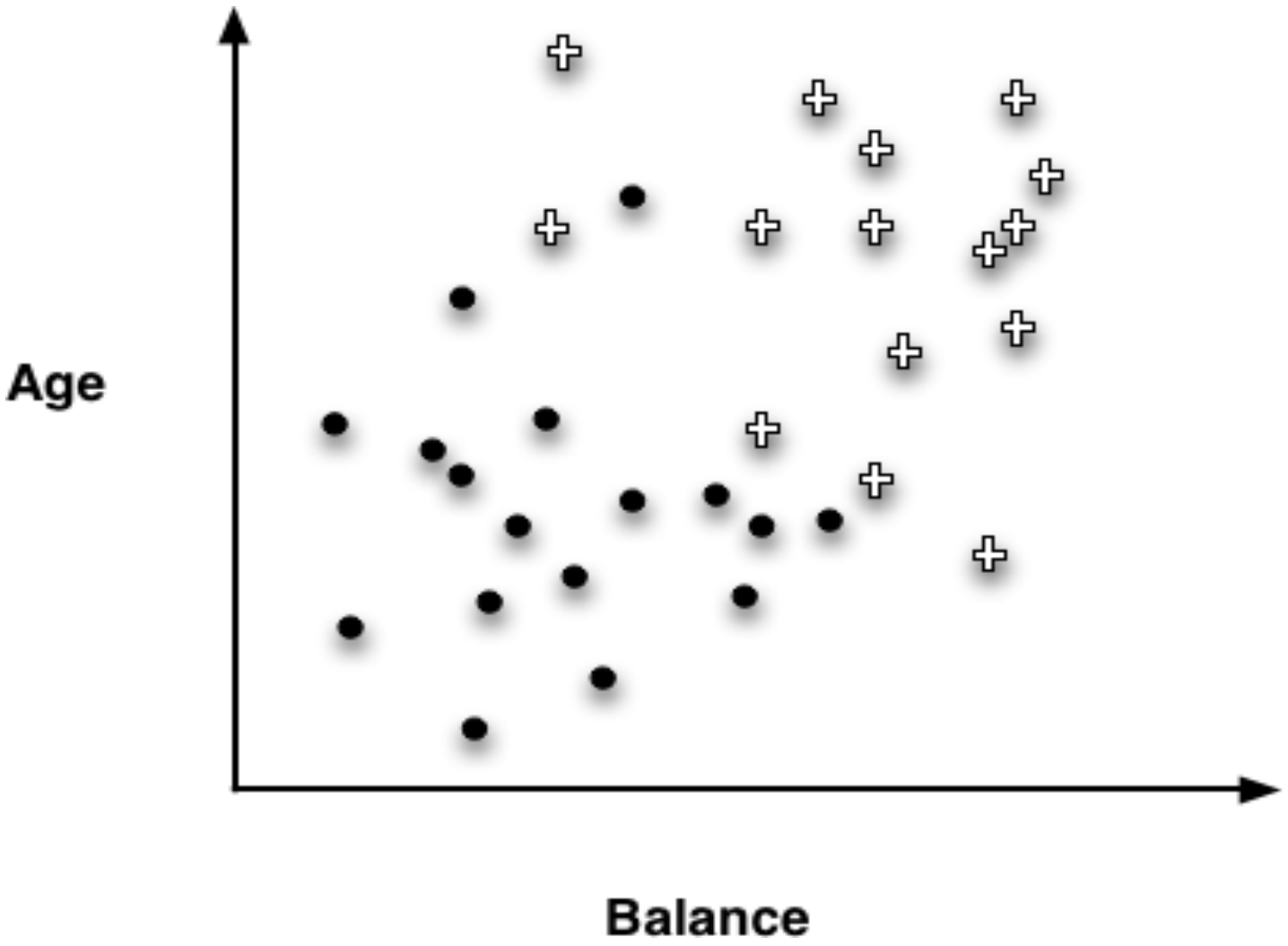
Post-pruning a tree



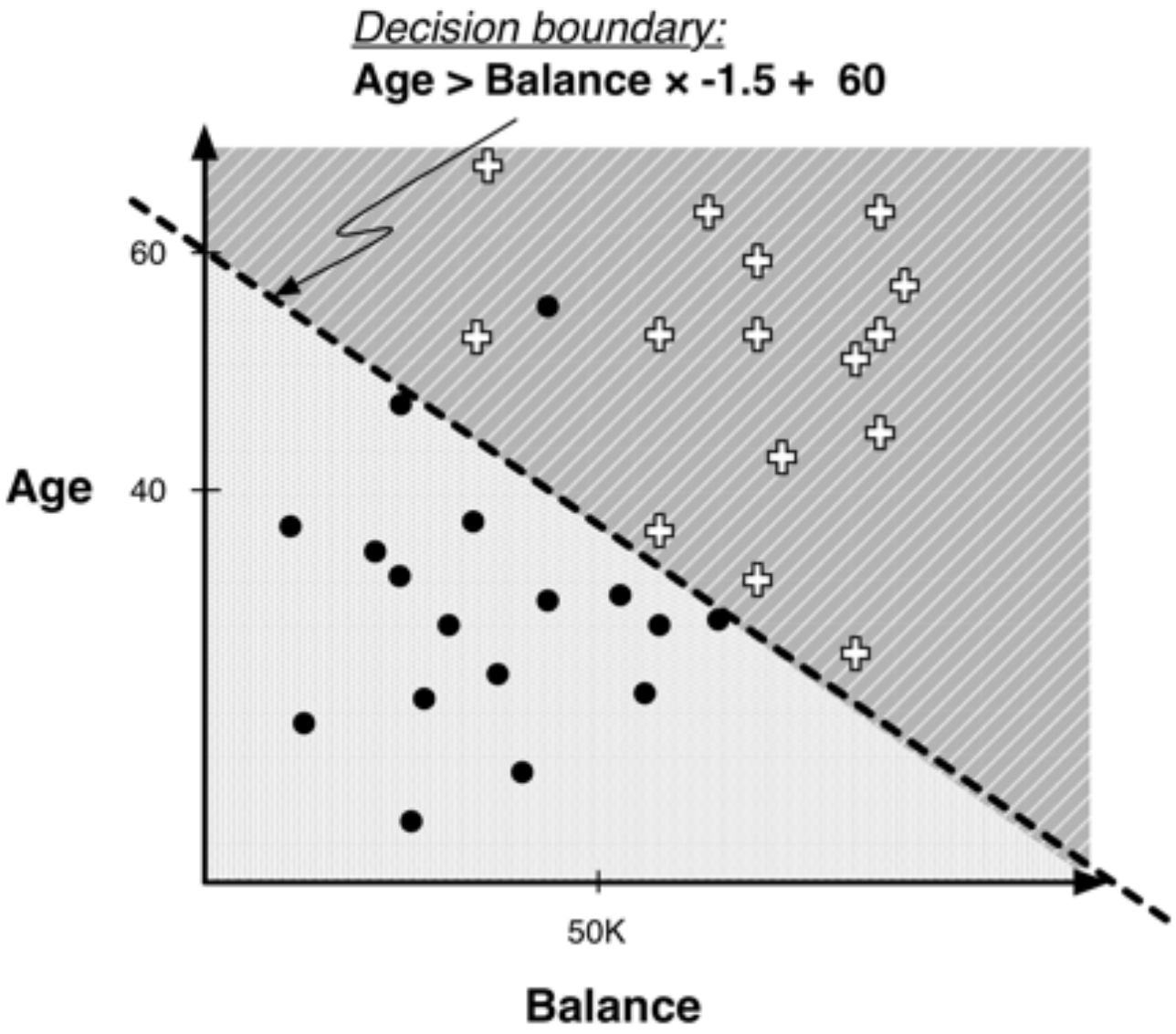
Decision Boundaries



Instance Space



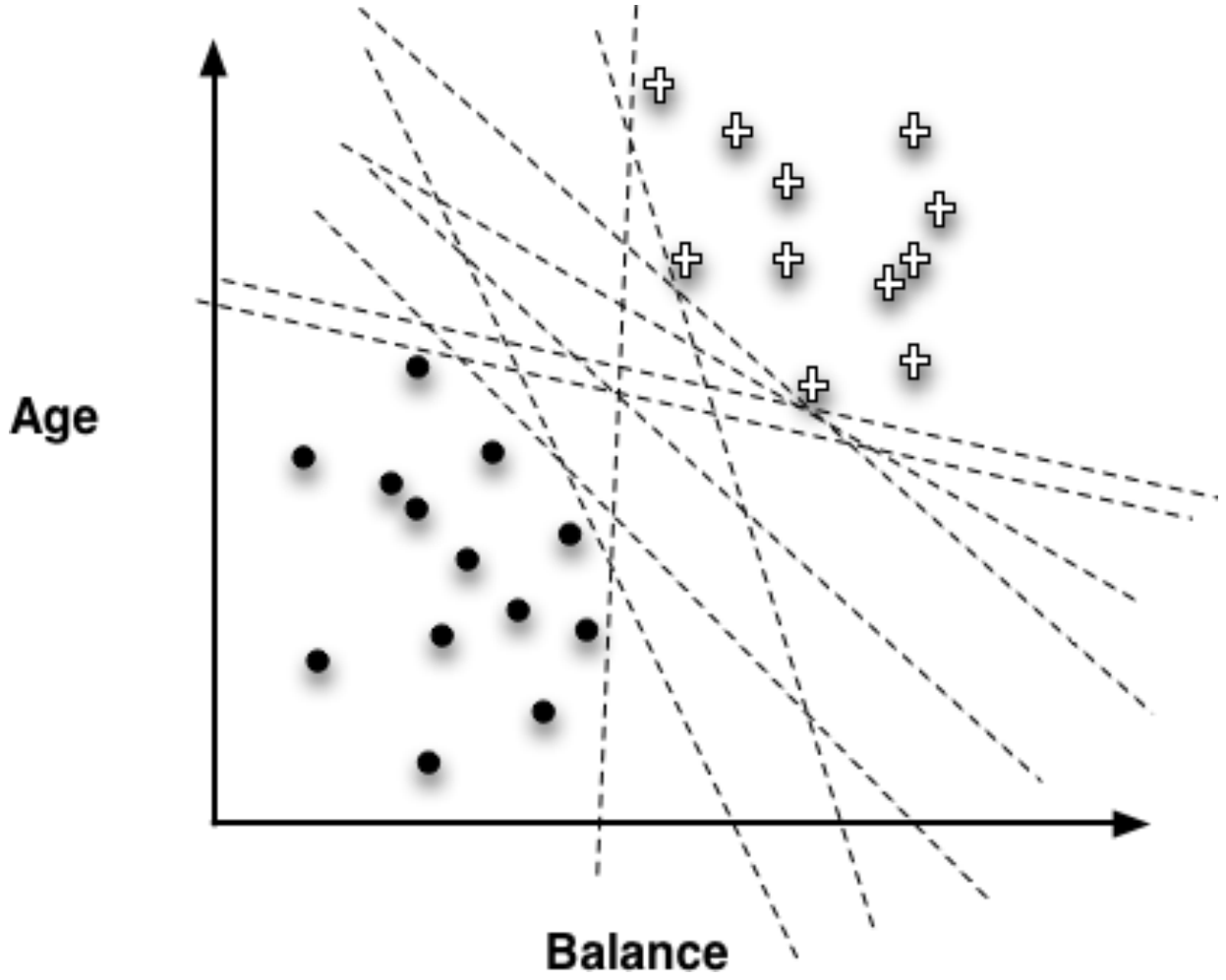
Linear Classifier



Example of Classification Function

- Linear discriminant:
- $class(x) = \begin{cases} + \text{ if } 1.0 \times Age - 1.5 \times Balance + 60 > 0 \\ \bullet \text{ if } 1.0 \times Age - 1.5 \times Balance + 60 \leq 0 \end{cases}$
- We now have a **parameterized model**: the weights of the linear function are the parameters
- The weights are often *loosely* interpreted as **importance indicators** of the features
- A different sort of multivariate supervised segmentation
    - The difference from DTs is that the method for taking multiple attributes into account is to create a mathematical function of them

Choosing the "best" line



Objective Functions

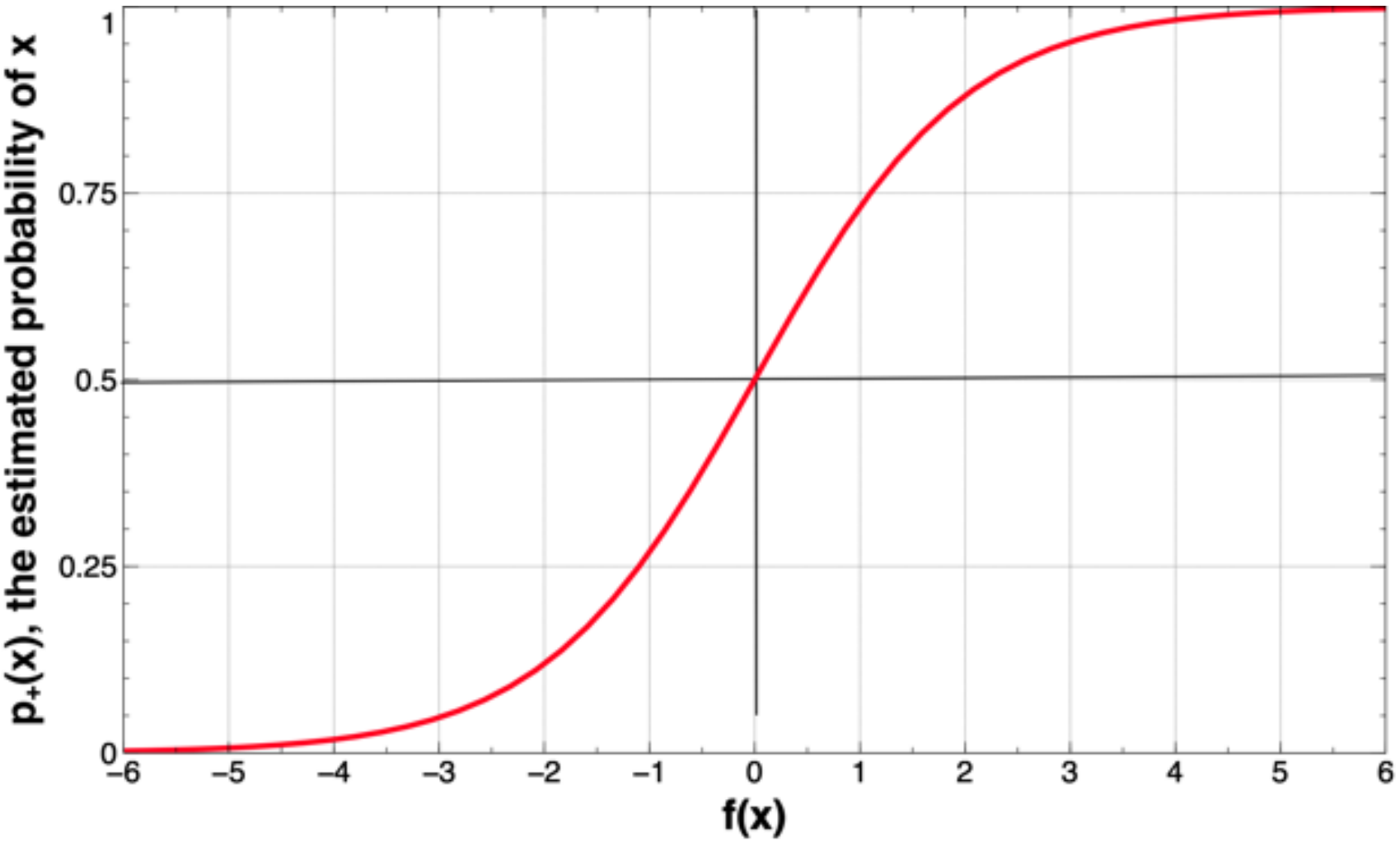
- "Best" line depends on the **objective (loss) function**
  - Objective function should represent our goal
- A loss function determines how much penalty should be assigned to an instance based on the error in the model's predicted value
- Examples of objective (or loss) functions:
  - $\lambda(y; x) = |y - f(x)|$
  - $\lambda(y; x) = \big(y - f(x)\big)^2$ [convenient mathematically – linear regression]
  - $\lambda(y; x) = I(y \neq f(x))$
- **Linear regression**, **logistic regression**, and **support vector machines (SVM)** are all very similar instances of our basic fundamental technique:
  - The key difference is that each uses **a different objective function**

ตระกูลเดียวกัน แต่การปรับหา loss function ต่างกัน

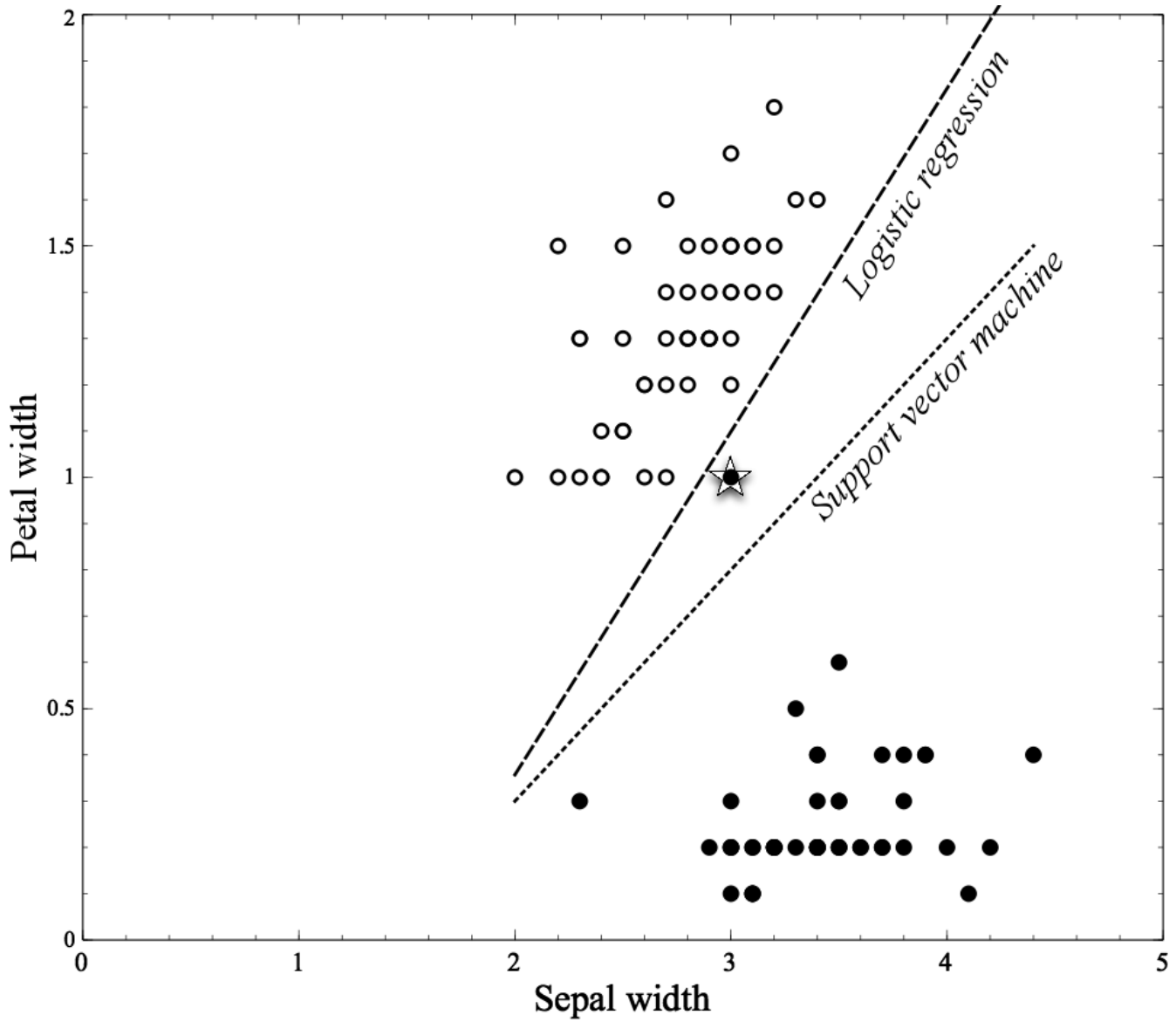
Logistic regression is a misnomer

- The distinction between classification and regression is whether the value for the **target variable is categorical or numeric**
- For logistic regression, the model produces a numeric estimate
- However, **the values of the target variable in the data are categorical**
- Logistic regression is estimating the probability of class membership (a numeric quantity) over a **categorical class**
- Logistic regression is a **class probability estimation model** and not a regression model

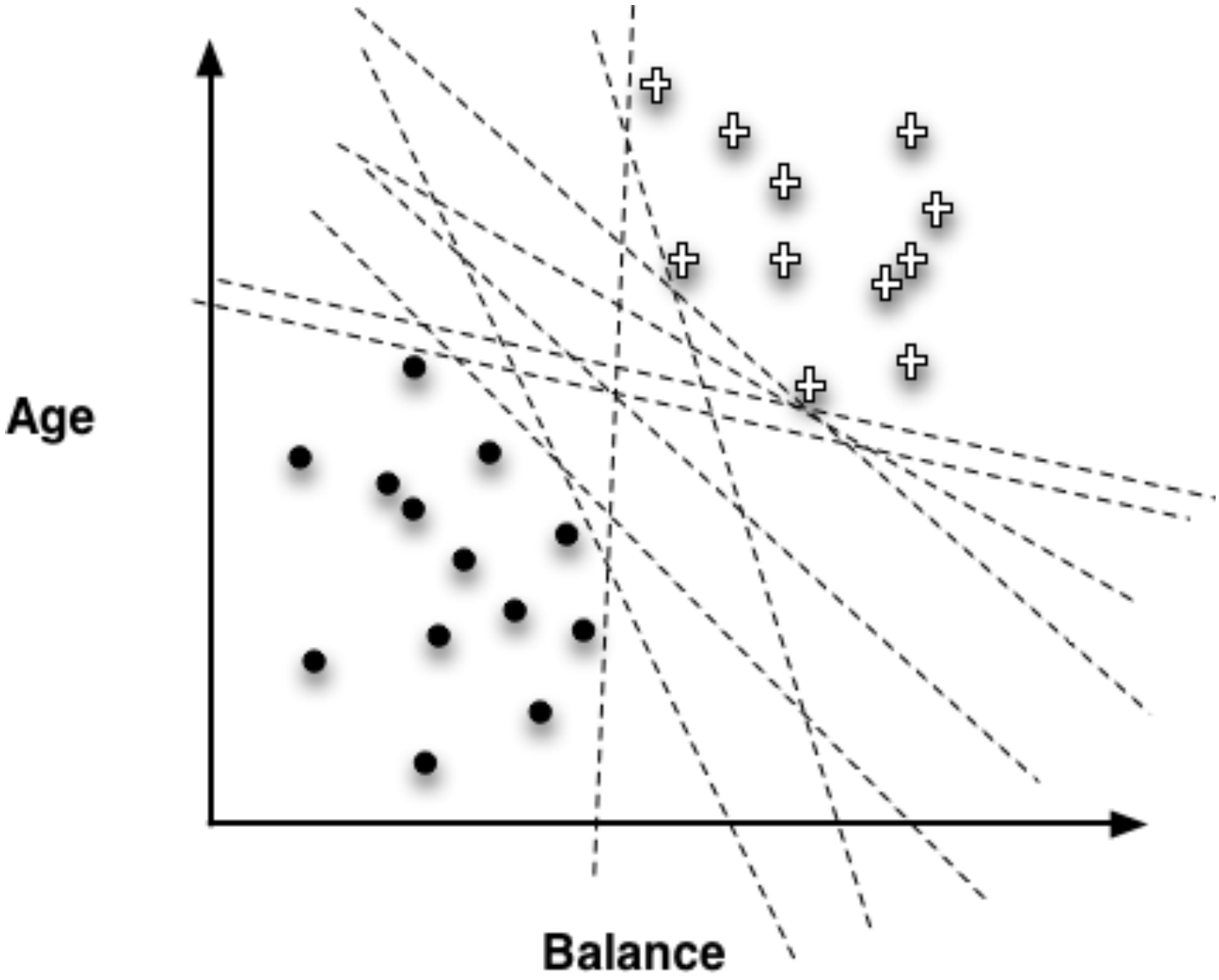
Logistic regression ("sigmoid") curve



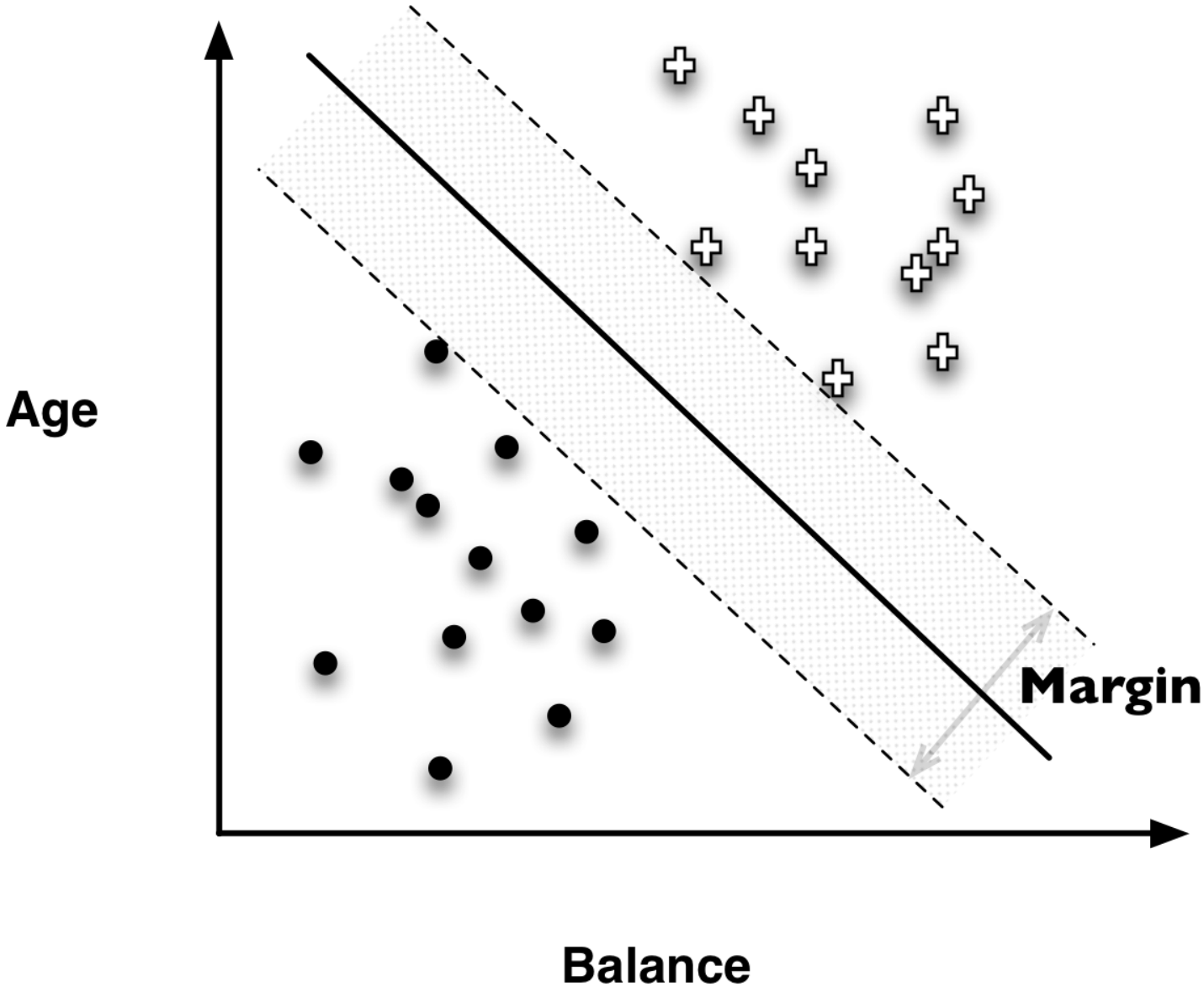
Classifying Flowers



Choosing the "best" line



Support Vector Machines (SVMs)



Support Vector Machines (SVMs)

ก็คือเปลี่ยนจากหา loss function (minimize loss) ไปทำการให้ maximum margin

- Linear Discriminants
- Effective
- Use “hinge loss”
- Also, non-linear SVMs

Hinge Loss functions

- Support vector machines use **hinge loss**
- Hinge loss incurs no penalty for an example that is not on the wrong side of the margin
- The hinge loss only becomes positive when an example is on the wrong side of the boundary and beyond the margin
    - Loss then increases linearly with the example's distance from the margin
    - Penalizes points more the farther they are from the separating boundary

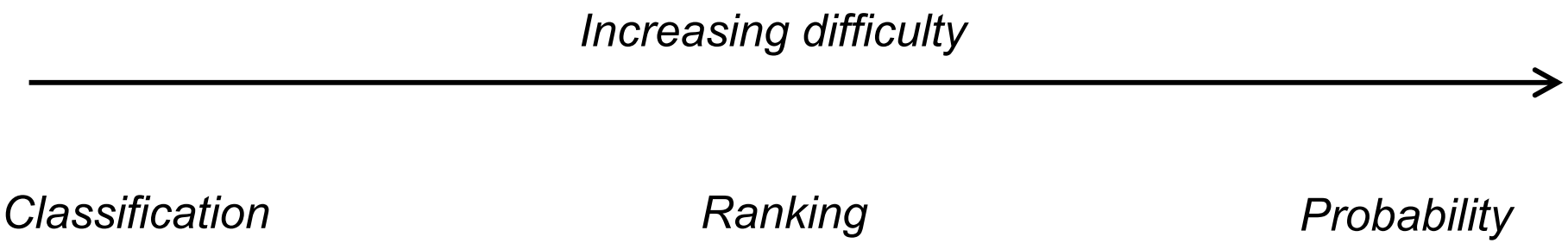
Loss Functions

- **Zero-one loss** assigns a loss of zero for a correct decision and one for an incorrect decision
- **Squared error** specifies a loss proportional to the square of the distance from the boundary
  - Squared error loss usually is used for numeric value prediction (regression), rather than classification
  - The squaring of the error has the effect of greatly penalizing predictions that are grossly wrong

Ranking Instances and Probability Class Estimation

- In many applications, we don't simply want a yes or no prediction of whether an instance belongs to the class, but we want some notion of **which examples are more or less likely to belong to the class**
    - Which consumers are most likely to respond to this offer?
    - Which customers are most likely to leave when their contracts expire?
- Ranking
    - Tree induction
    - Linear discriminant functions (e.g., linear regressions, logistic regressions, SVMs)
        - Ranking is free
- Class Probability Estimation
    - Tree induction
    - Logistic regression

The many faces of classification: Classification / Probability Estimation / Ranking



- Ranking:
  - Business context determines the number of actions ("how far down the list")
- Probability:
  - You can always rank / classify if you have probabilities!

Ranking: Examples

- Search engines
    - Whether a document is relevant to a topic / query

Class Probability Estimation: Examples

- MegaTelCo
  - Ranking vs. Class Probability Estimation
- Identify accounts or transactions as likely to have been defrauded
  - The director of the fraud control operation may want the analysts to focus not simply on the cases most likely to be fraud, but on accounts where the **expected monetary loss** is higher
    - We need to estimate the actual probability of fraud

Application of Logistic Regression

- The Wisconsin Breast Cancer Dataset

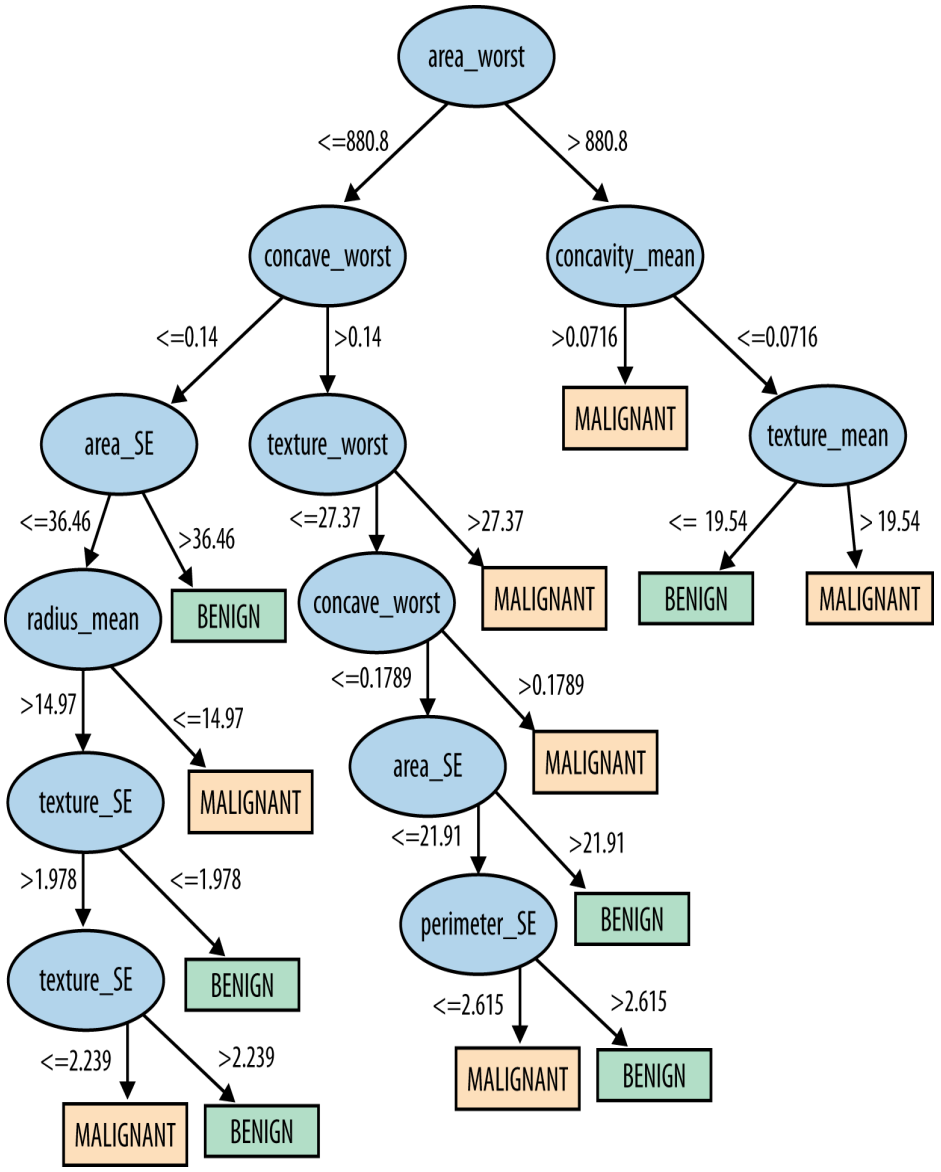
Wisconsin Breast Cancer dataset

Attribute name	Description
RADIUS	*Mean of distances from center to points on the perimeter*
TEXTURE	*Standard deviation of grayscale values*
PERIMETER	*Perimeter of the mass*
AREA	*Area of the mass*
SMOOTHNESS	*Local variation in radius lengths*
COMPACTNESS	*Computed as: perimeter^2/area – 1.0*
CONCAVITY	*Severity of concave portions of the contour*
CONCAVE POINTS	*Number of concave portions of the contour*
SYMMETRY	*A measure of the symmetry of the nucleii*
FRACTAL DIMENSION	*'Coastline approximation'* – 1.0
DIAGNOSIS (Target)	*Diagnosis of cell sample: malignant or benign*

- From each of these basic characteristics, three values were computed: the mean (_mean), standard error (_SE), and "worst" or largest

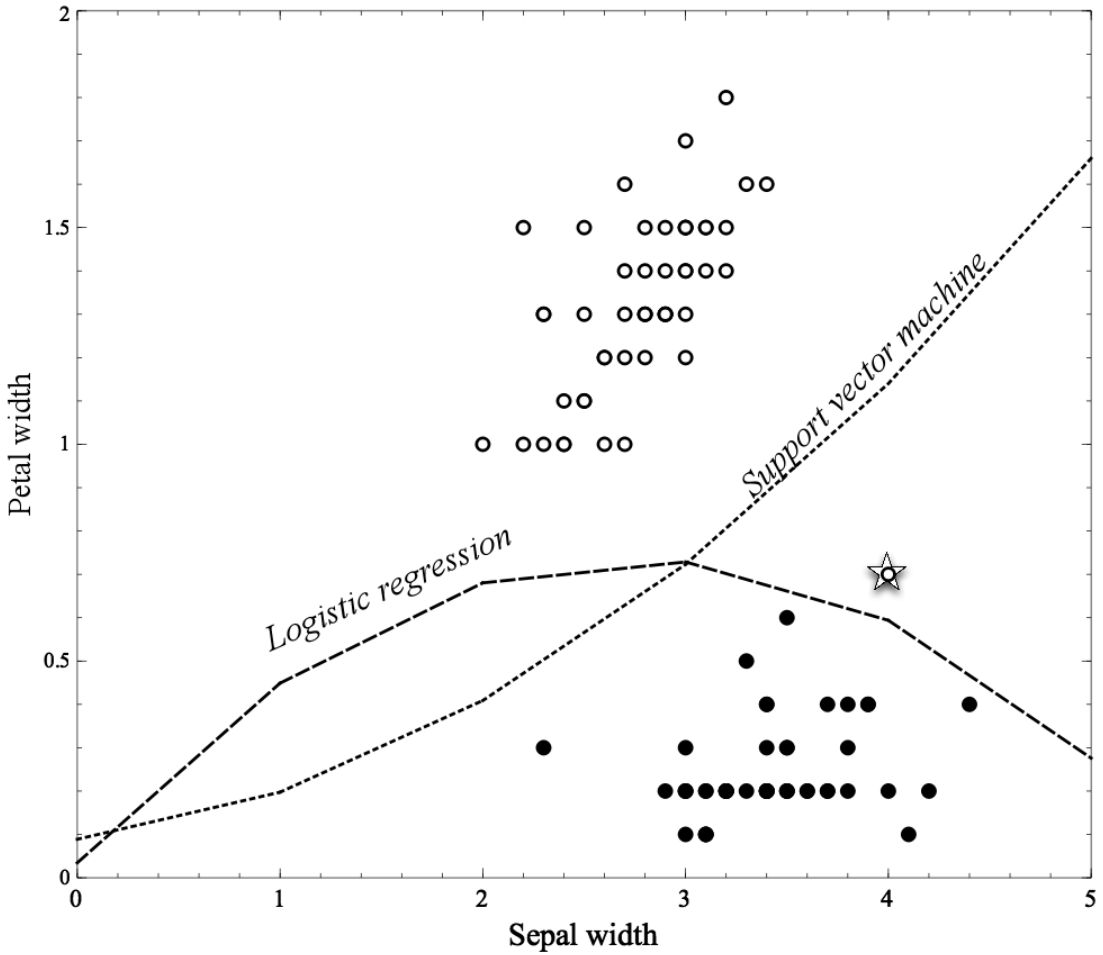
Wisconsin Breast Cancer dataset

Attribute	Weight (learned parameter)
SMOOTHNESS_worst	22.3
CONCAVE_mean	19.47
CONCAVE_worst	11.68
SYMMETRY_worst	4.99
CONCAVITY_worst	2.86
CONCAVITY_mean	2.34
RADIUS_worst	0.25
TEXTURE_worst	0.13
AREA_SE	0.06
TEXTURE_mean	0.03
TEXTURE_SE	–0.29
COMPACTNESS_mean	–7.1
COMPACTNESS_SE	–27.87
w_0 *(intercept)*	–17.7



Non-linear Functions

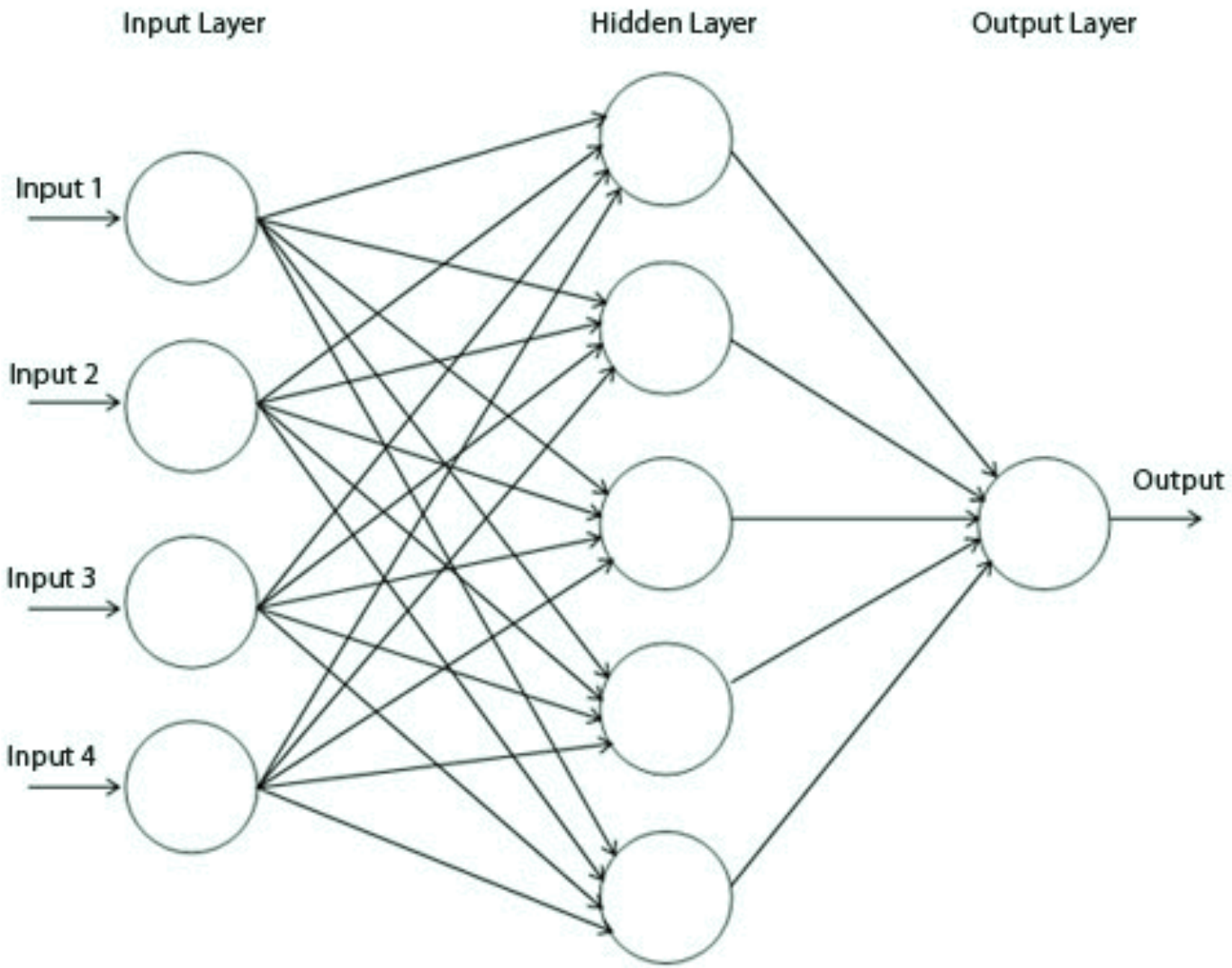
- **Linear functions can actually represent nonlinear models,** if we include more complex features in the functions



Non-linear Functions

- Using "higher order" features is just a "trick"
- Common techniques based on fitting the parameters of complex, nonlinear functions:
    - Non-linear support vector machines and neural networks
- **Nonlinear support vector machine** with a "polynomial kernel" consider "higher-order" combinations of the original features
    - Squared features, products of features, etc.
- Think of a **neural network** as a "stack" of models
    - On the bottom of the stack are the original features
    - Each layer in the stack applies a simple model to the outputs of the previous layer
- Might fit data *too* well (..to be continued)

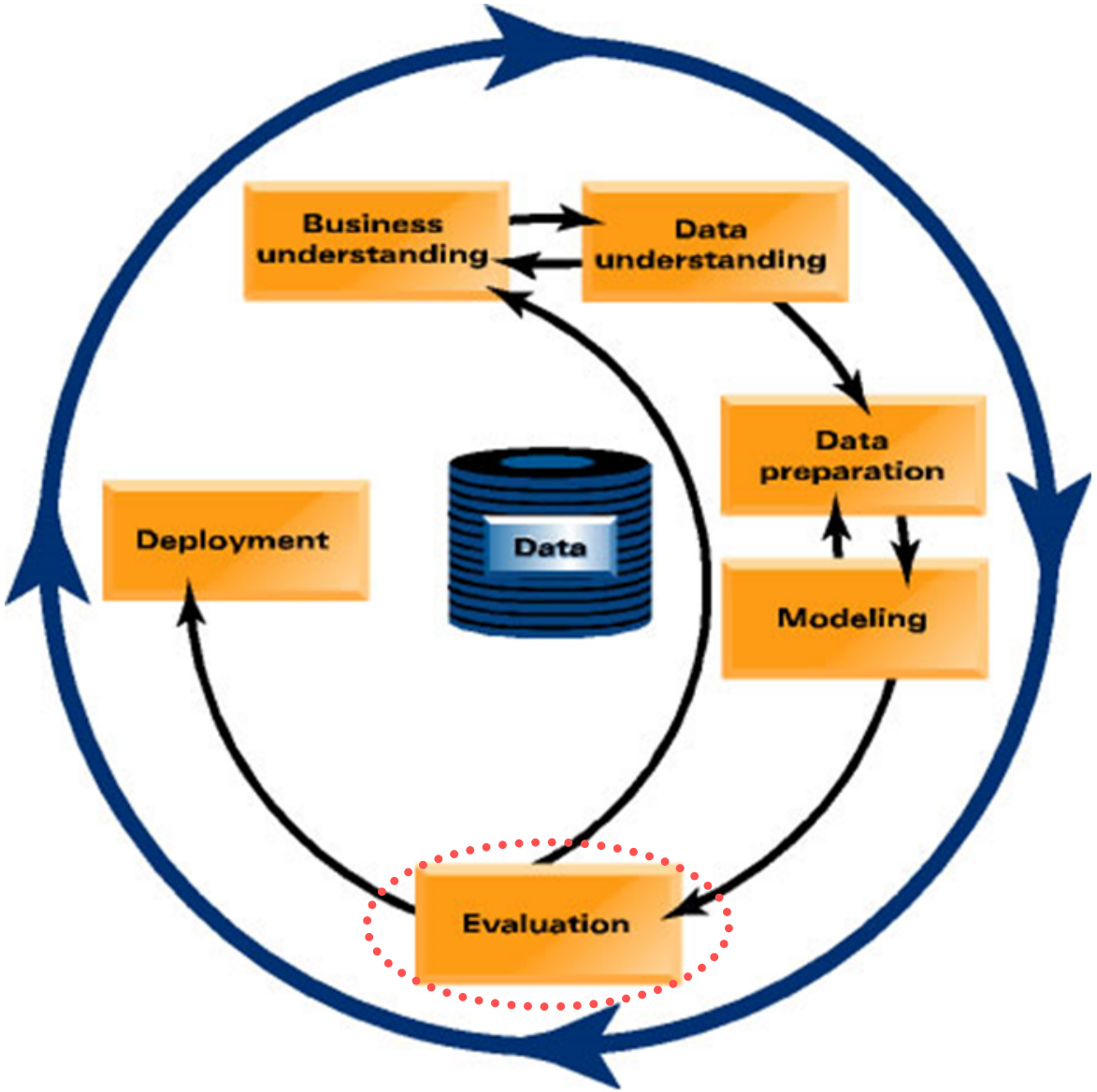
Simple Neural Network



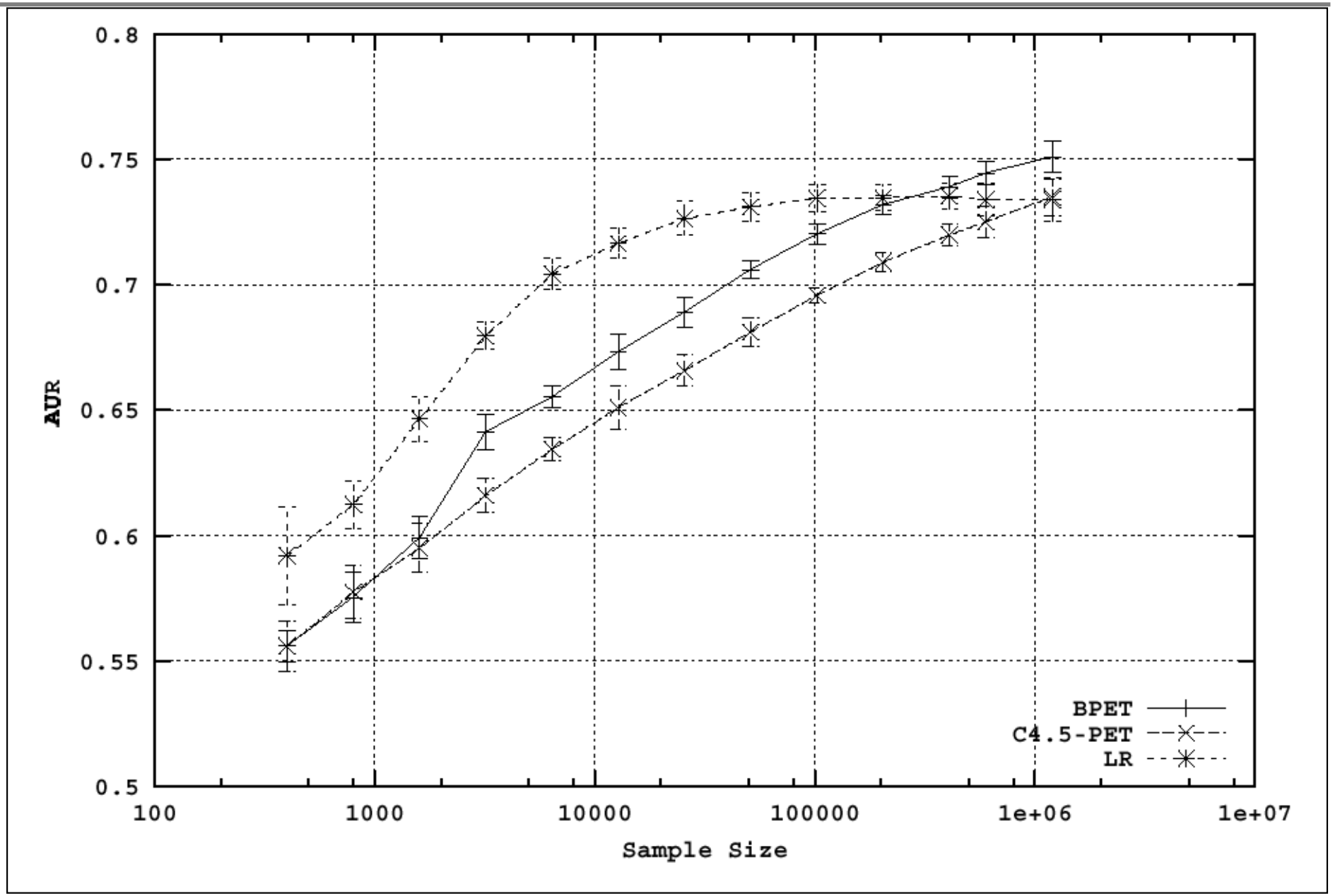
Linear Models versus Tree Induction

- What is more comprehensible to the stakeholders?
  - Rules or a numeric function?
- How "**smooth**" is the underlying phenomenon being modeled?
  - Trees need a lot of data to approximate curved boundaries
- How "**non-linear**" is the underlying phenomenon being modeled?
  - If very, much "data engineering" needed to apply linear models
- **How much data do you have?!**
  - There is a key tradeoff between the complexity that can be modeled and the amount of training data available
- What are the characteristics of the data: missing values, types of variables, relationships between them, how many are irrelevant, etc.
  - Trees fairly robust to these complications

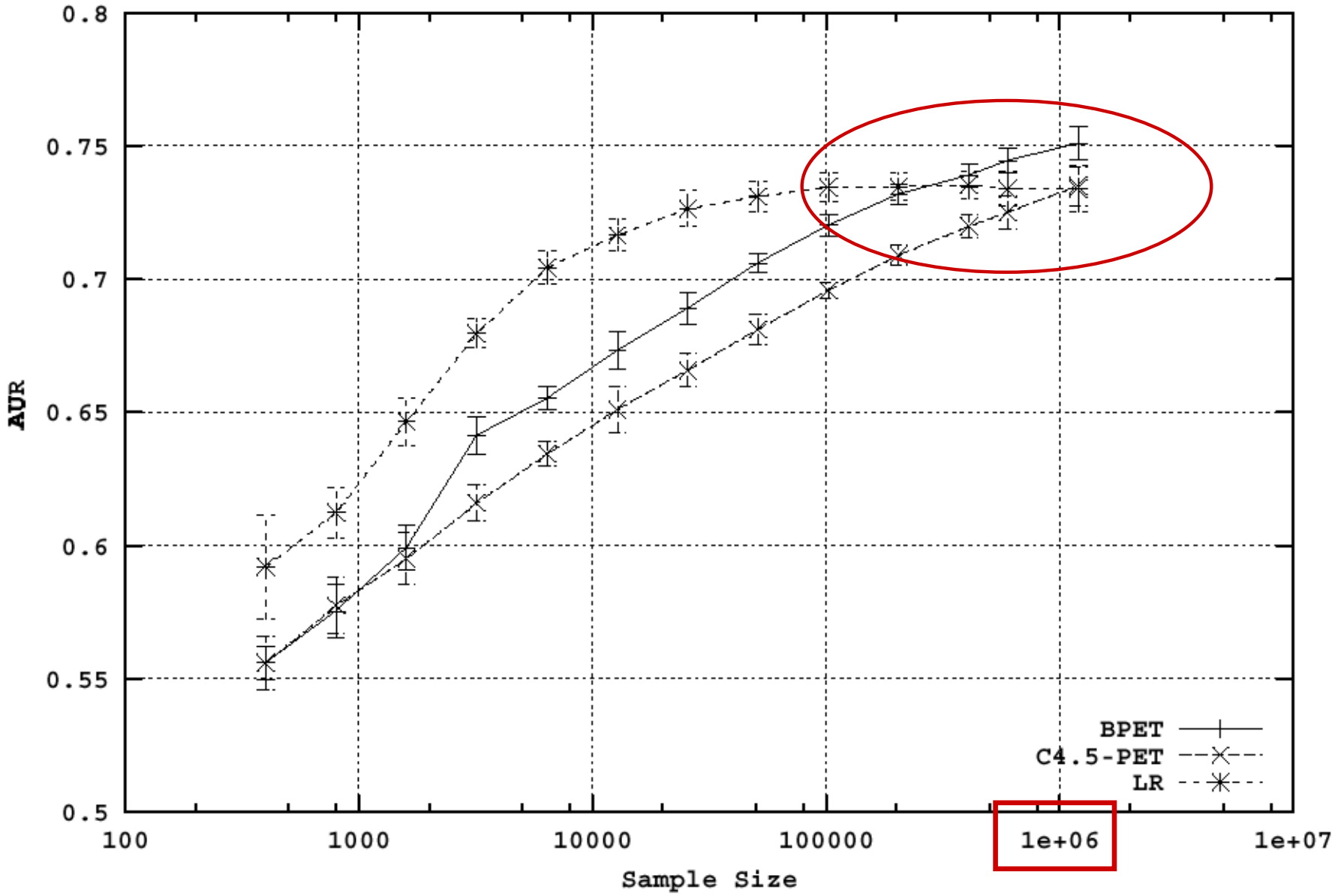
Data Mining Process



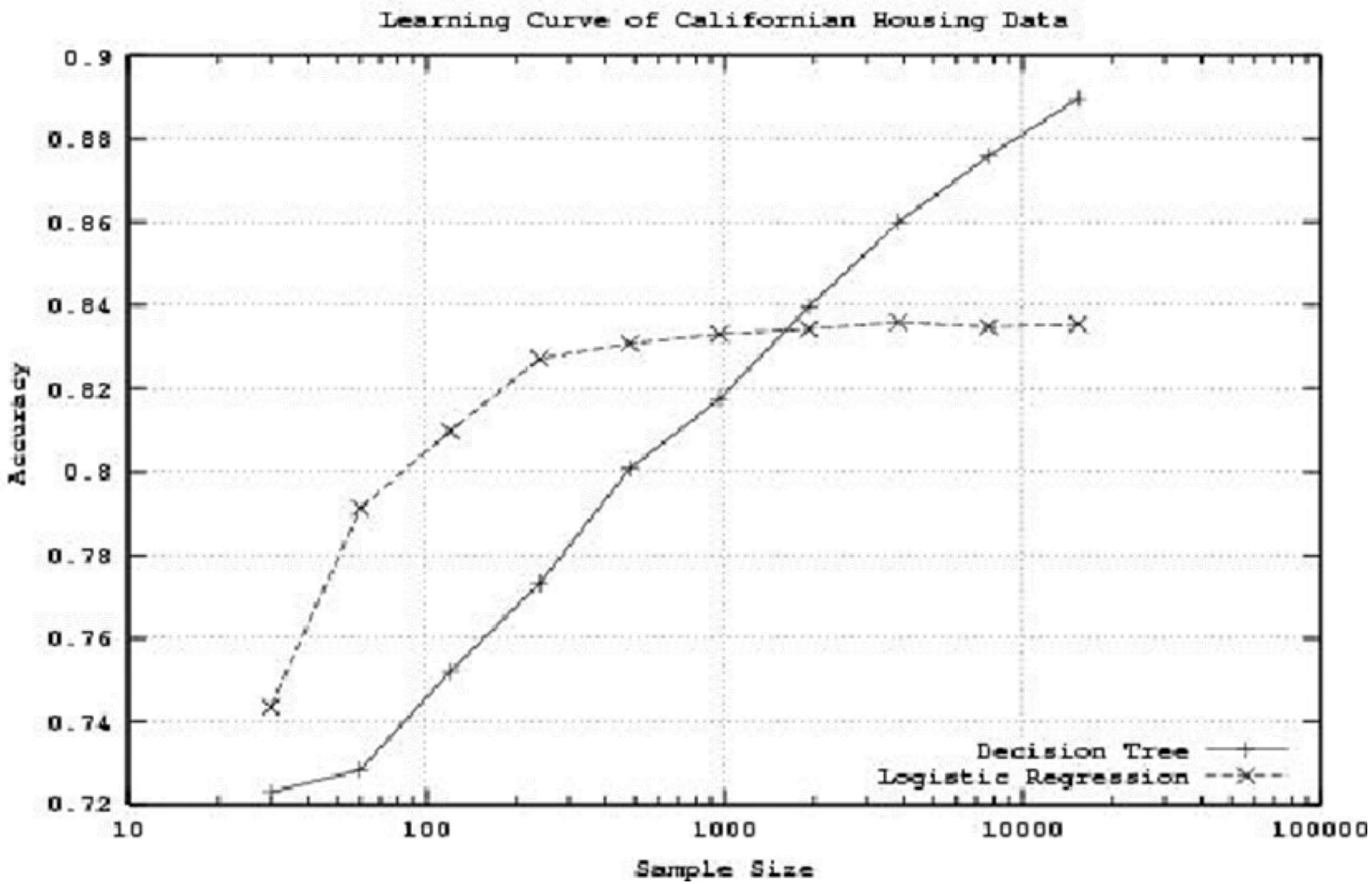
Comparing learning curves is essential



Comparing learning curves is essential



Choice of algorithm is not trivial!



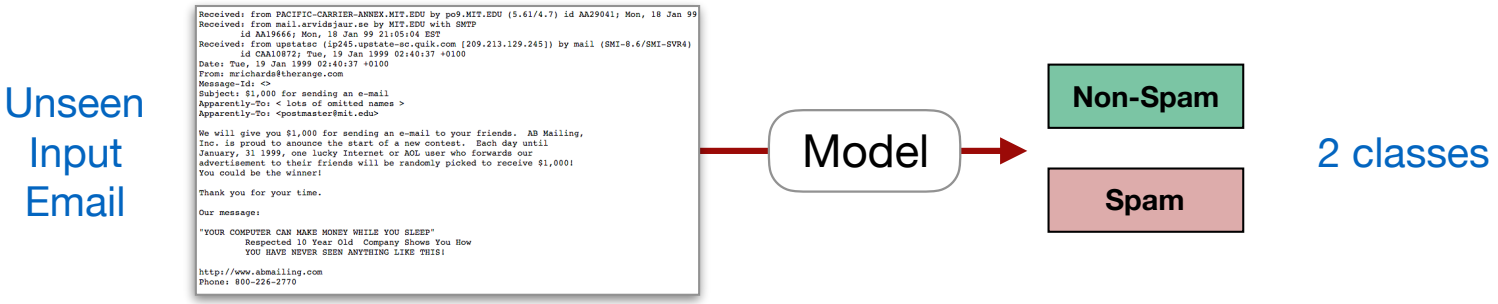
Data Science for Business
Linear Regression

Asst. Prof. Teerapong Leelanupab (Ph.D.)
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang (KMITL)

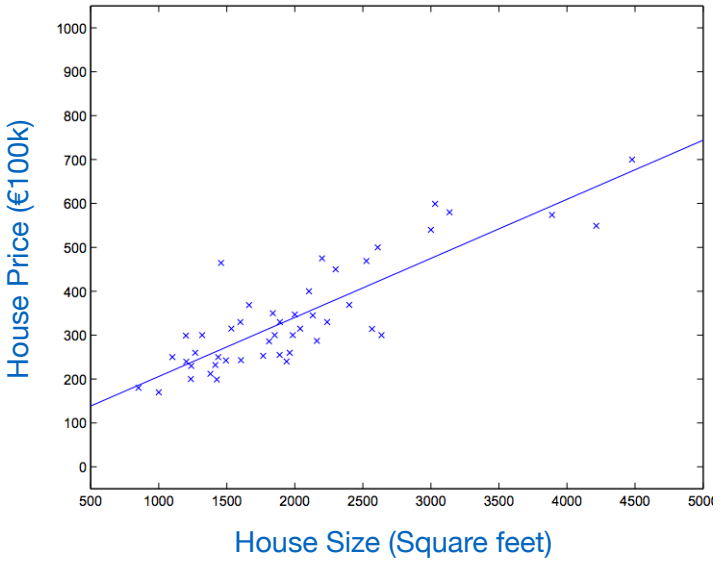
Week 8.2

Supervised Learning

- **Classification:**
  Learn from a labelled training set to make a prediction to assign a new "unseen" example to one of a fixed number of classes.

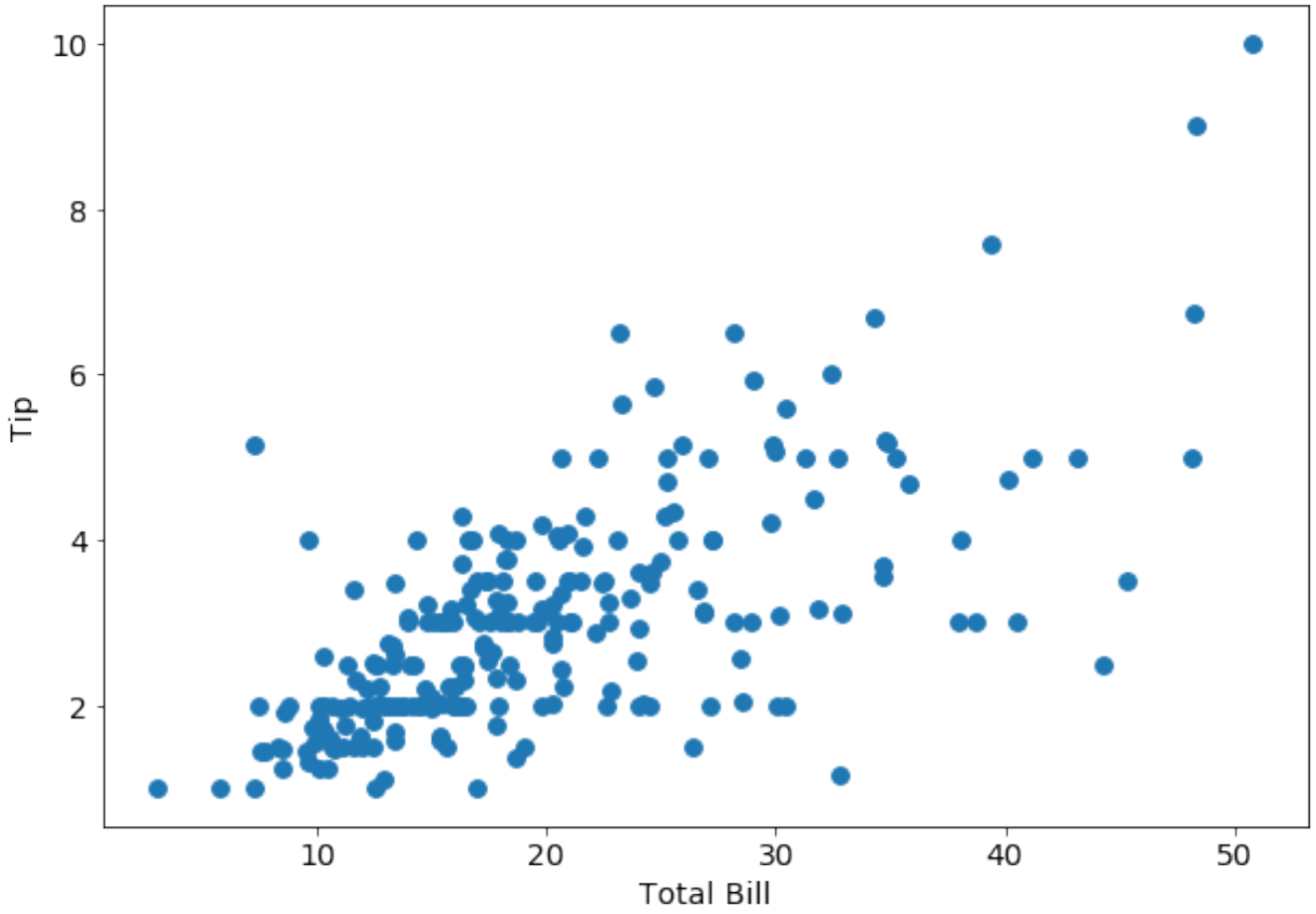


- **Regression:**
  Learn from a labelled training set to decide the value of a continuous output variable (i.e. the output is a number).



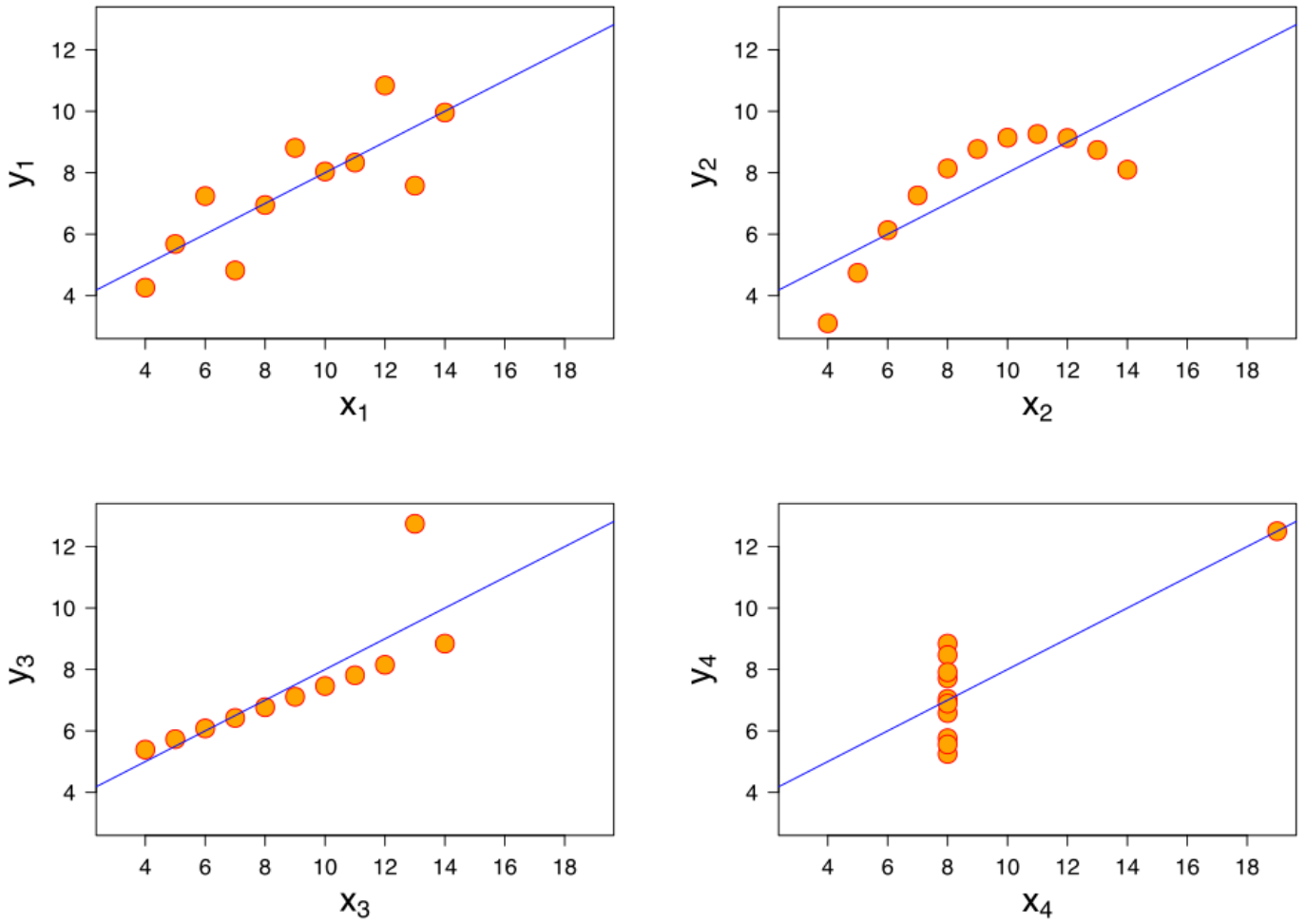
Finding Patterns in Data

- When analysing a new dataset, as a first step we might visualise the data to identify any obvious patterns.
- **Example:** Dataset of 244 meals, with details of total meal bill and tip amount. What can we say about the data?



Anscombe's Quartet

- 4 datasets with nearly identical statistical properties. Yet, each expresses quite different relationships between *Y* and *X*. Important to look at the data visually before building a model!

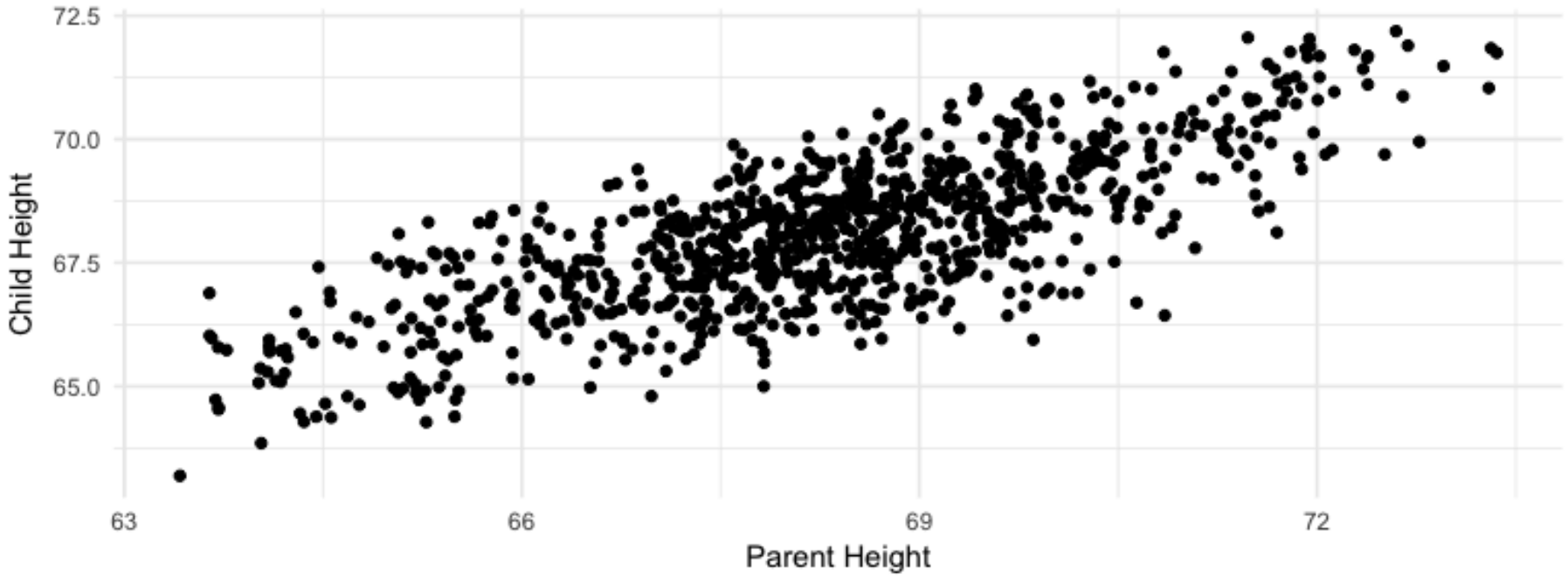


https://en.wikipedia.org/wiki/Anscombe%27s_quartet

Wikipedia

Regression

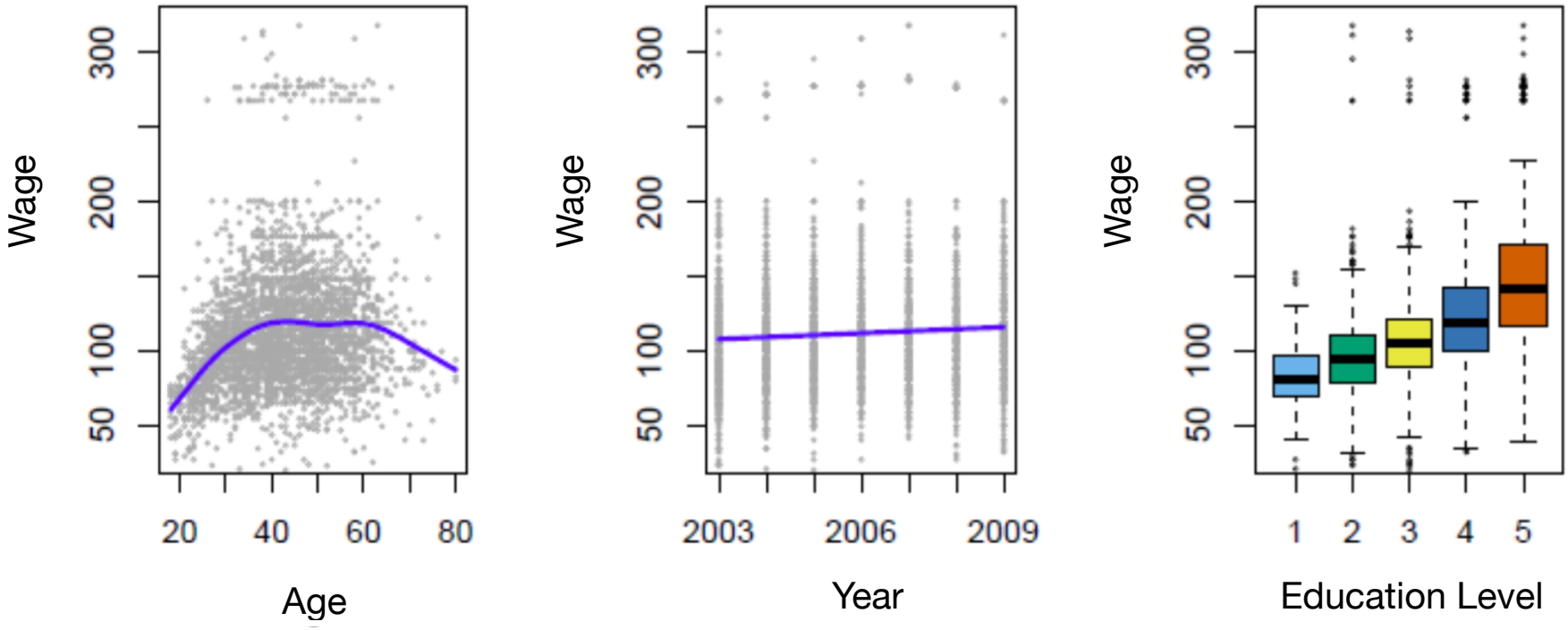
- Regression analysis: A common statistical process for estimating the relationships between variables. This can allow us to make numeric predictions based on past data.
- **Simple example**: Can we predict a child's height, based on their parent's height?



https://web.stanford.edu

Regression

- Regression analysis: A common statistical process for estimating the relationships between variables. This can allow us to make numeric predictions based on past data.
- **Example**: Can we establish a relationship between salary level and demographic variables in population survey data?



(Statistical Learning - Hastie & Tibshirani)

Regression

- Linear dependence: constant rate of increase of one variable with respect to another (as opposed to, e.g., diminishing returns).
- A regression analysis describes the relationship between two or more variables.

- We will try to understand:
  - how to build the linear model
  - make predictions
  - test the significance of the results

Examples:

How does earned income relate to educational level?

What is the connections between demand for electricity and the weather

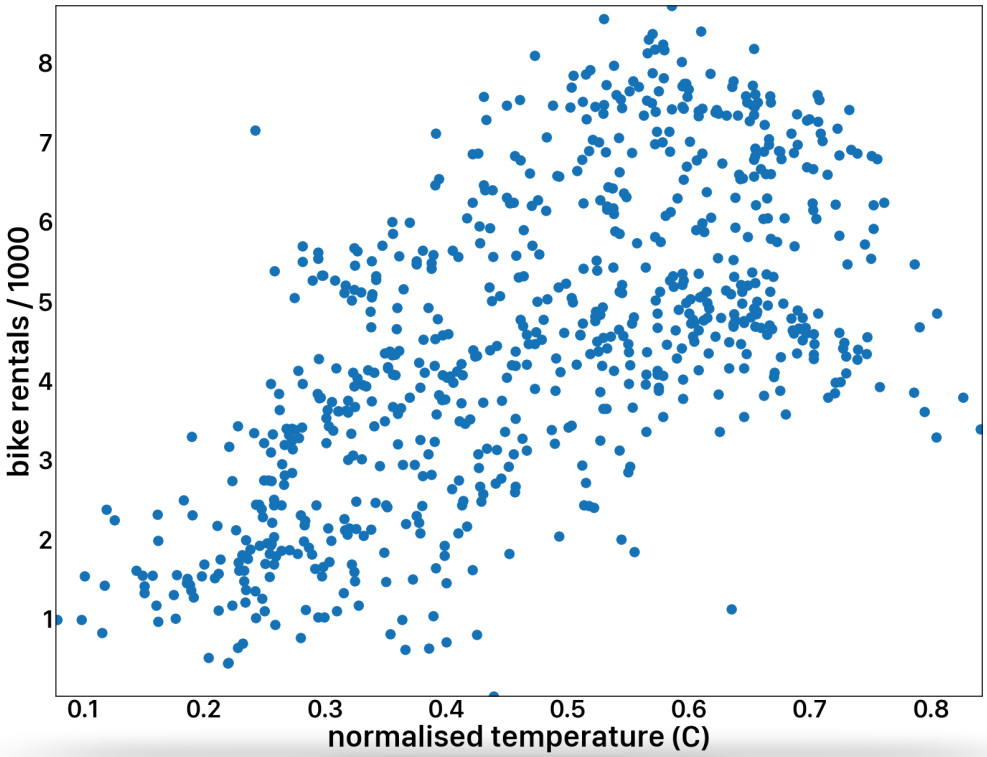
How do house sales depend on interest rates?

Regression: Terms

- Regression has a long history and there are many overlapping terms and names in common use
- Linear regression is a linear model: we assume the output variables *Y* model a linear relationship between the input variables *X*
- Simple Linear Regression: there is just one input variable *X* (changes is *Y* are caused by changes in *X*)
- Multiple Linear Regression: there is more than one input variable (we will only consider simple linear regression)
- One of the simplest ways to train the model is with Ordinary Least Squares, commonly called Least Squares Regression

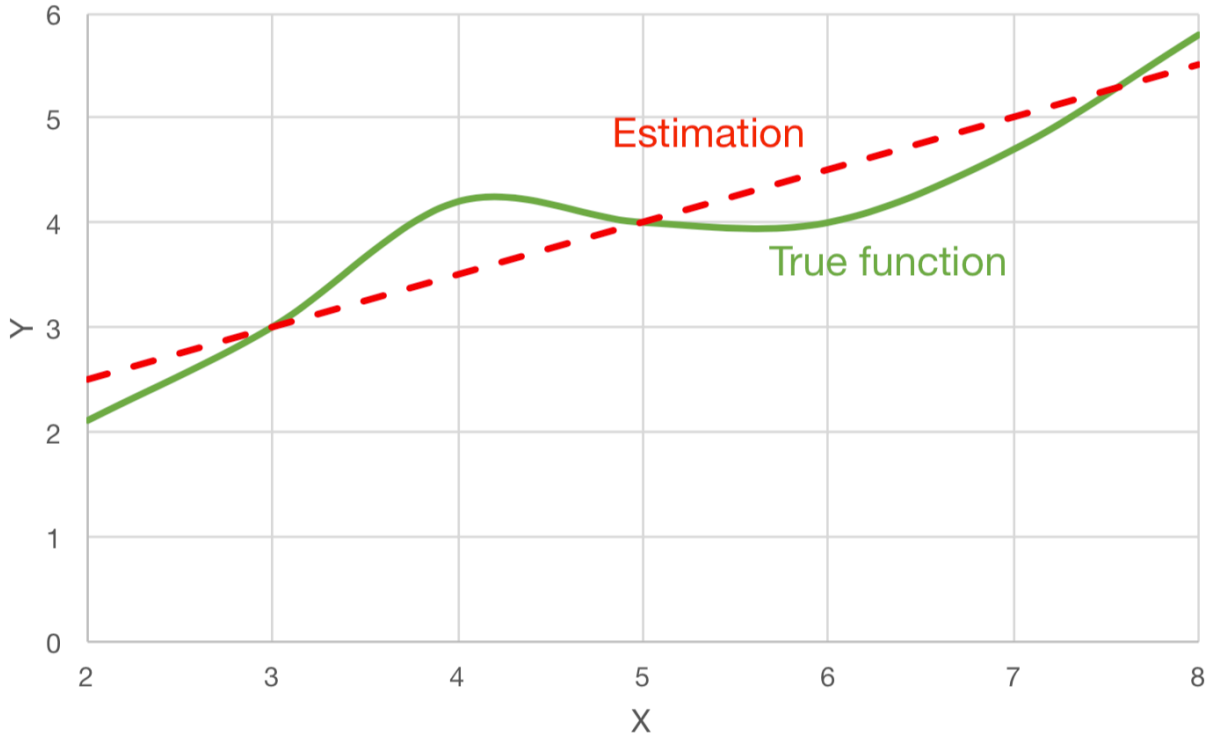
Regression Example

- Bike Sharing Data
- We would like to know how the number of bike rentals per day depends on the the temperature
- From the data it looks like there are more rentals on warmer days (perhaps expected?)
- But what is the relationship between the temperature and rentals?



Linear Regression

- Linear Regression: a simple approach to predictive modelling. It assumes that the dependence of a response (dependent) variable Y on input (independent) variables $X_1, X_2, \ldots$ is linear.
- While true regression functions are never linear, simple linear regression is still often useful.



Regression Example

- Output variable *Y* is the number of bike rentals
- The input variable *X* is the temperature
- Our linear model is:

$$\text{bike rentals} = 7501.8339 \times temperature + 945.824$$

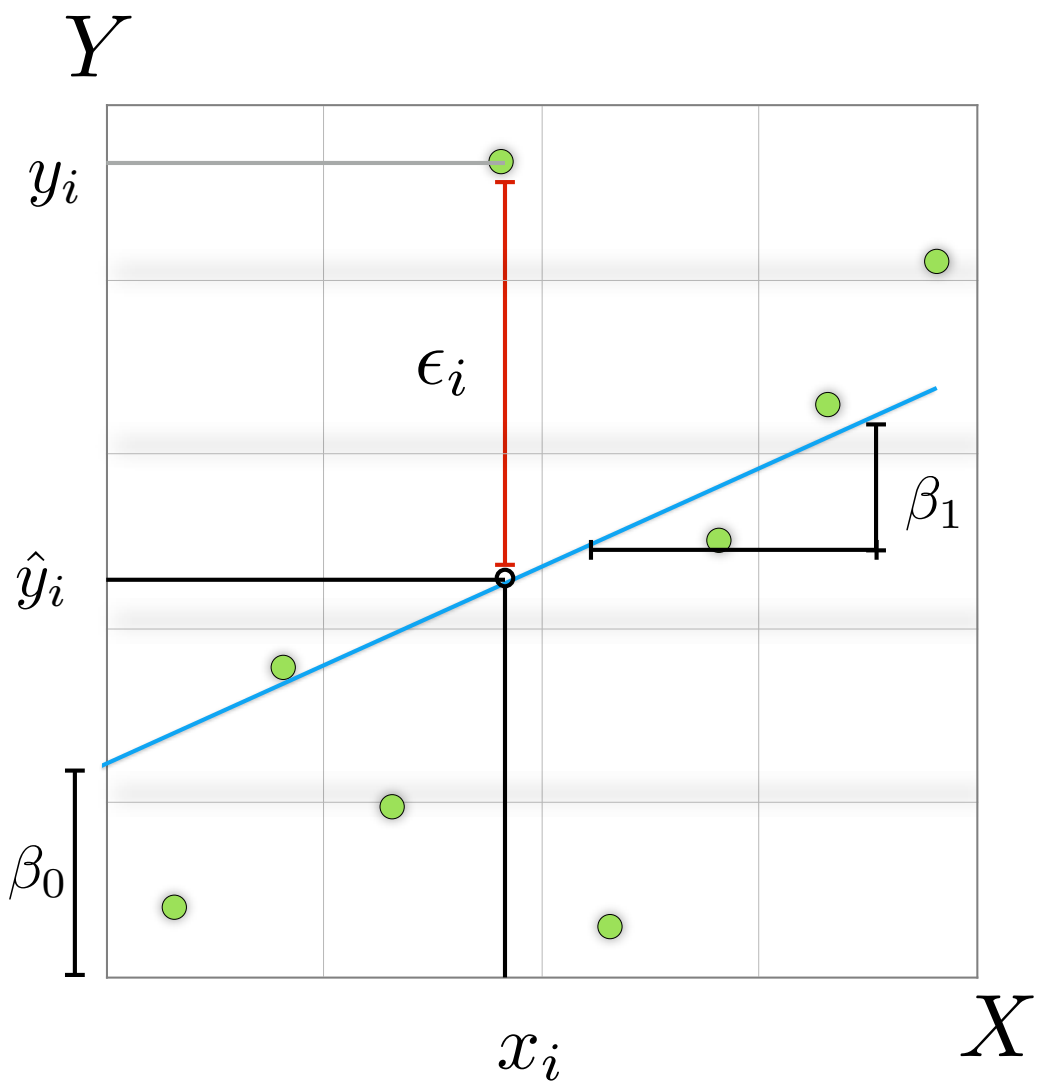
General form

$$Y = \beta_0 + \beta_1 \times X$$

for every 1degree increase in the temperature, we would expect the bike rentals to increase by 7501

Terminology

X	independent variables
Y	dependent variables
y_i	observed value of Y
$\hat{y}_i$	predicted value of Y
β_0	intercept ตัดแกน y ที่ไหนก็เป็นจุดเริ่มต้น
β_1	slope
ϵ_i	error



Simple Linear Regression

- Simple Linear Regression: Method for predicting a numeric response using a single input variable (feature).
- The model is: $y = \beta_0 + \beta_1 x$

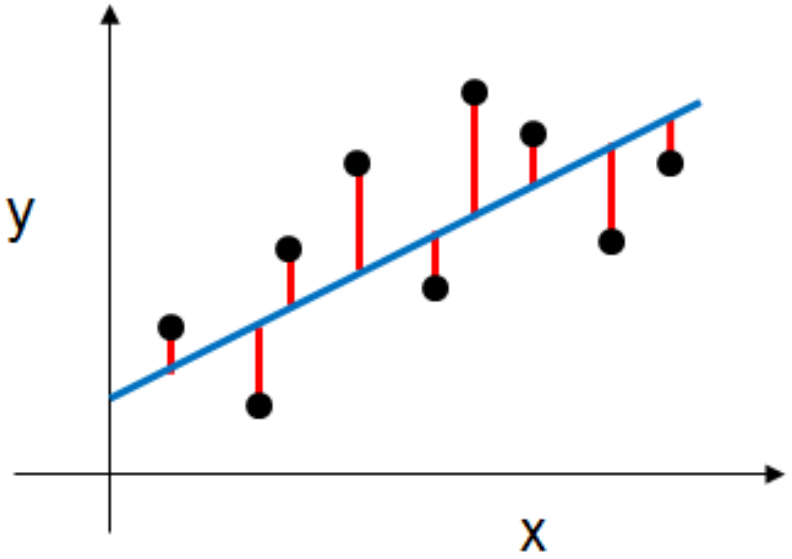
y : the response variable

x : the input feature

β_0, β_1 : the model coefficients (intercept and slope)

Goal is to learn the model coefficients from existing data.

Once we have learned the model, we can make future predictions.



We learn the model by finding the best line (coefficients) which minimises the squared distance between our examples and the line.

Simple Linear Regression

- We assume that *Y* (the dependent variable) is a linear function of *X*
- The parameters of the model are β_0, β_1:
  - β_0: estimated average value of *Y* when *X* is 0
  - β_1: estimated change in average value of *Y* if we make a unit change in *X*

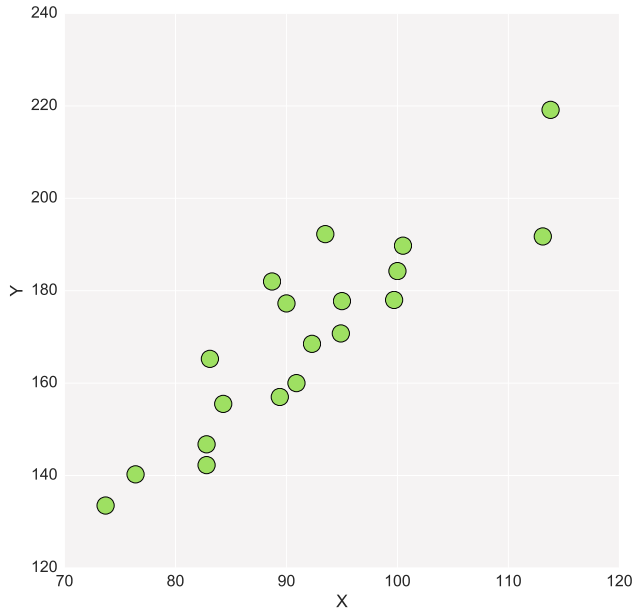
simple linear regression model

$$Y = \beta_0 + \beta_1 \times X$$

Simple Linear Regression

- We want to estimate the parameters β_0, β_1
- We want to find the best line which represents the data
- But how do we find the best possible representation?

draw a line through the data and use a measure which gives us information on how well that line represents the data



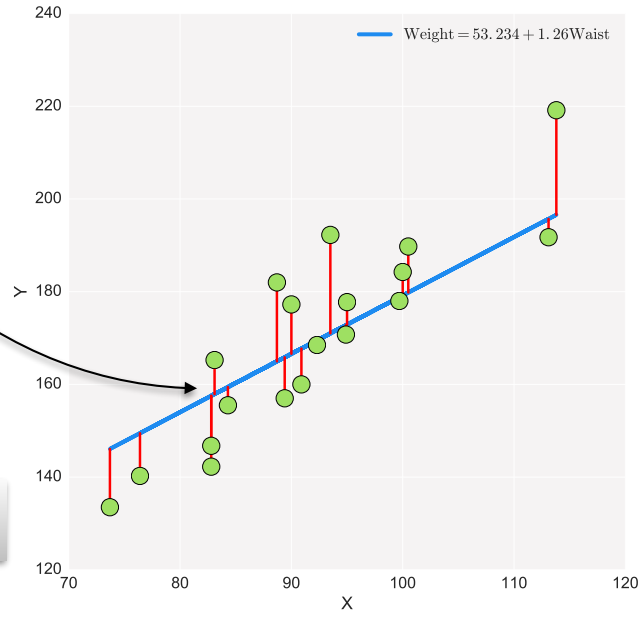
Simple Linear Regression

- We want to estimate the parameters β_0, β_1
- We want to find the best fit or representation of the points
- But how do we find the best possible representation?

draw a line through the data and use a measure which gives us information on how well that line represents the data

error

residuals



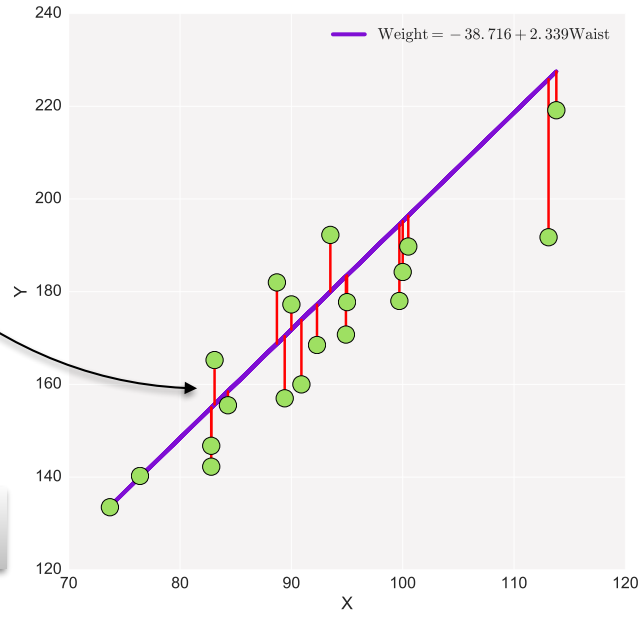
sum(squared error) = 2333.0325

Simple Linear Regression

- We want to estimate the parameters β_0, β_1
- We want to find the best fit or representation of the points
- But how do we find the best possible representation?

some lines are a better fit than others
the difference between the data and the line is the ***residual***

residuals



sum(squared error) = 2995.6991

sigma(e^2)

- Use the least squares method to minimise the error (residuals)

Least Squares (วิธีกำลังสองน้อยที่สุด)

- The best fit values for the parameters β are found by minimising the sum of squared errors:

$$\frac{\partial}{\partial \beta} \sum_{i=1}^{n} \epsilon_i^2 = 0$$

With a little bit calculus we find the the values of β_0 and β_1 which minimise the squared residuals. These are **Ordinary Least Square** (OLS) estimates of β_0 and β_1

$$\hat{\beta}_1 = \frac{Cov(X,Y)}{s_X^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

covariance
(ความแปรปรวนร่วมเกี่ยว)

variance
(ความแปรปรวน)

การหาความแตกต่างของค่าเฉลี่ย
ระหว่างกลุ่มตัวอย่างตั้งแต่ 2 กลุ่มขึ้นไป

- where:

standard deviation of X

$$s_X = \sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2}$$

covariance of X,Y

$$Cov(X,Y) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{n-1}$$

The regression slope, β_1, is the covariance between *X* and *Y* divided by the variance of *X* or the square of SD of *X*.

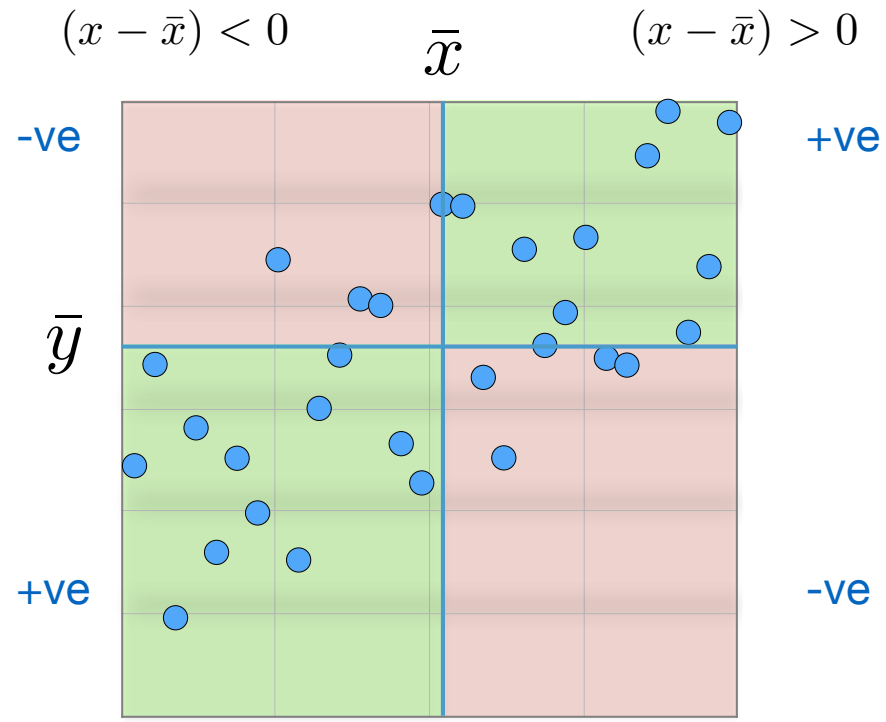
Covariance (ความแปรปรวมร่วมเกี่ยว)

- If *Y* increases as *X* increases the relationship is positive (there will be more points in the *+ve* quadrants) and vice versa.
- The *covariance* between *Y* and *X* indicates the direction of the relationship
- But, it doesn't tell us about the strength of the relationship

$$Cov(Y,X) = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})(X_i - \bar{X})}{n-1}$$

$Cov(Y,X) > 0$ positive relationship

$Cov(Y,X) < 0$ negative relationship



$$Cor(X,Y) = \frac{Cov(X,Y)}{s_X s_Y}$$

Pearson Correlation: covariance scaled by the standard deviations

The Pearson correlation, Cor(*X*,*Y*), is the covariance divided by the product of the standard deviations (SD).

Linear Regression

- Now that we have the least squares parameters we can construct the equation of the line:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$$

- Taking our previous bike sharing dataset we find:

$$\hat{\beta}_1 = \frac{Cov(X,Y)}{s_X^2}$$
$$= \frac{199.221}{0.0266}$$
$$= 7501.834$$

$$\hat{\beta}_0 = \bar{Y} - \beta_1 \bar{X}$$
$$= 4504.348 - 7501.833 \times 0.4744$$
$$= 945.824$$

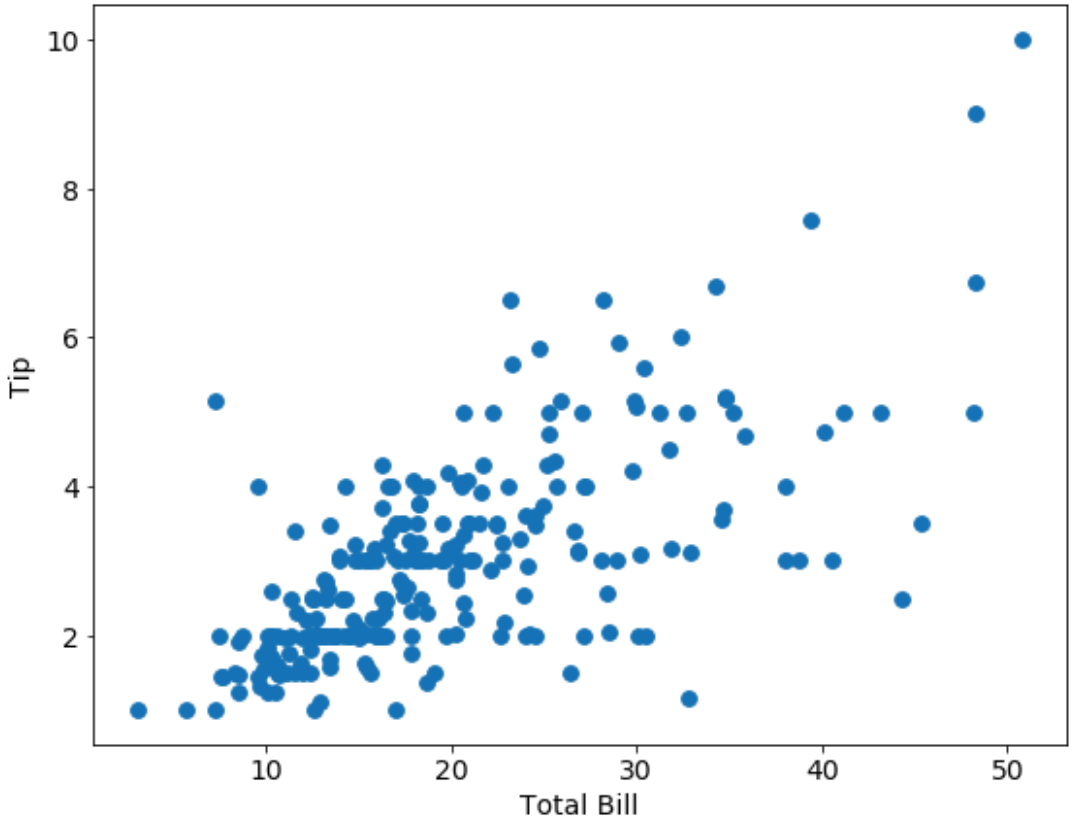
regression model:

$$\text{bike rentals} = 7501.8339 \times temperature + 945.824$$

Example: Linear Regression

Q. Can we predict the tip amount, from the total bill?

- Independent variable: the total bill amount X
- Dependent variable: the tip amount Y ("response" variable)



Use our historic dataset of 244 meals as the training data to build a new regression model which can then be used to make predictions.

Correlation in Python

- Often useful to know the strength of relationship between Y and X, but independent of the units of measurement.
- The Correlation between Y and X is a statical measure of how strongly two variables are related. It is dimensionless, i.e. a unit-free measure of the relationship between variables.
- Takes a value in [-1,+1], where 1 is total positive correlation, 0 is no correlation, –1 is total negative correlation.

$$Cor(Y,X) = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{y_i-\bar{y}}{s_y}\right)\left(\frac{x_i-\bar{x}}{s_x}\right) \quad \text{where} \quad s_y^2 = \frac{\sum_i^n (y_i-\bar{y})^2}{n-1}$$

```
x = np.array([0.1, 0.3, 0.4, 0.8, 0.9])
y = np.array([3.2, 2.4, 2.4, 0.1, 5.5])
np.corrcoef(x,y)
```

```
array([[ 1.        , -0.95363007],
       [-0.95363007,  1.        ]])
```

Calling the NumPy `corrcoef(x,y)` function will create a 2x2 Pearson correlation coefficient matrix.

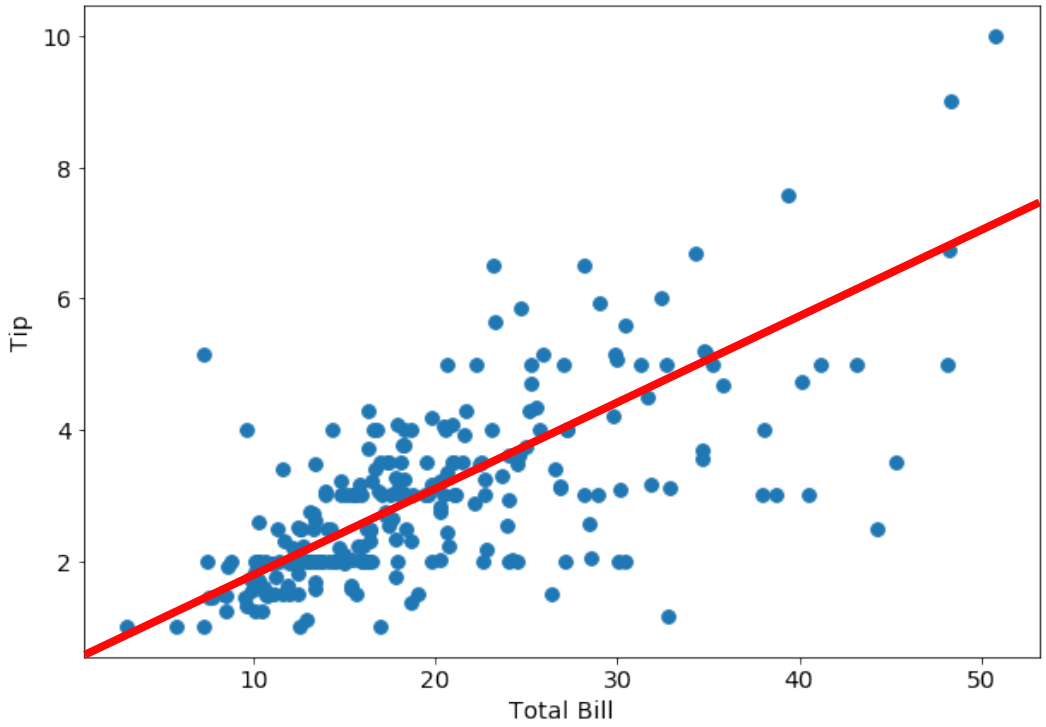
The off-diagonals indicate strength of relationships.

Correlation in Python

- If our data is stored in a Pandas DataFrame, we can also use the `df.corr()` function.

```
df.corr()
```

	total_bill	tip
total_bill	1.000000	0.675734
tip	0.675734	1.000000



The off-diagonals indicate strength of relationships.

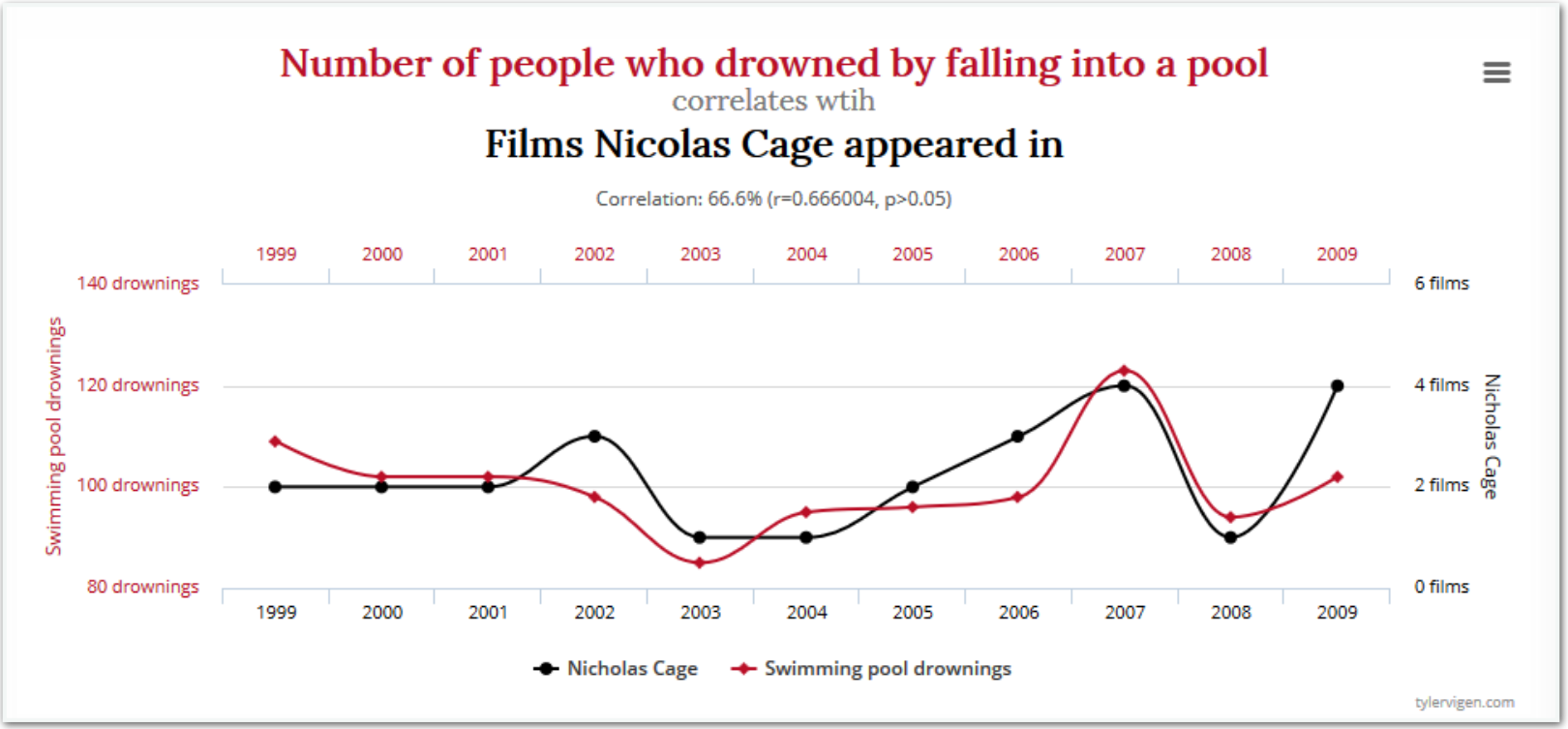
Remember, 1 is total positive correlation, 0 is no correlation, –1 is total negative correlation.

	total_bill	tip
0	16.99	1.01
1	10.34	1.66
2	21.01	3.50
3	23.68	3.31
4	24.59	3.61

การเป็นสาเหตุ

Correlation vs Causation

- Causation: indicates that one event is the result of the occurrence of the other event - i.e. there is a causal relationship between the two events.
- But a correlation between variables does not automatically mean that the change in one variable is the cause of the change in the values of the other variable.



https://amplitude.com/blog/2017/01/19/causation-correlation

Example: Linear Regression

- Scikit-Learn provides functions to apply linear regression to NumPy arrays. To build a model for input *x* and response *y*:

```
from sklearn.linear_model import LinearRegression
```

Create and fit the model based on the training data

```
model = LinearRegression()
model.fit(x, y)
```

Get the intercept coefficient

```
model.intercept_
```

```
0.92026961
```

B0

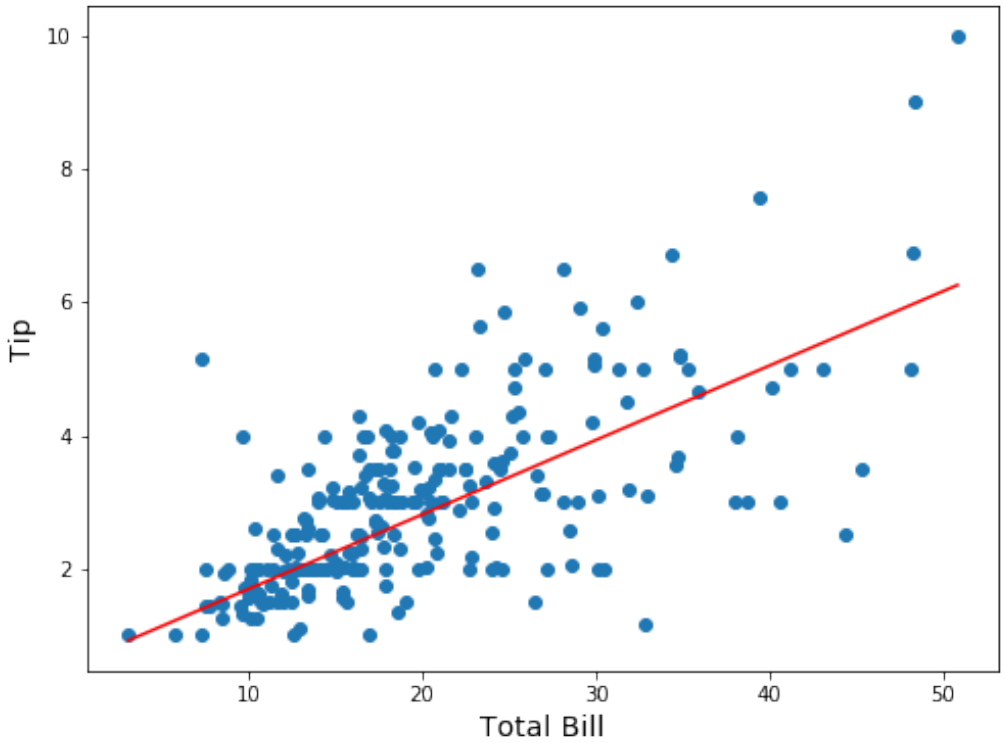
Get the slope coefficient

```
model.coef_[0]
```

```
0.10502452
```

B1

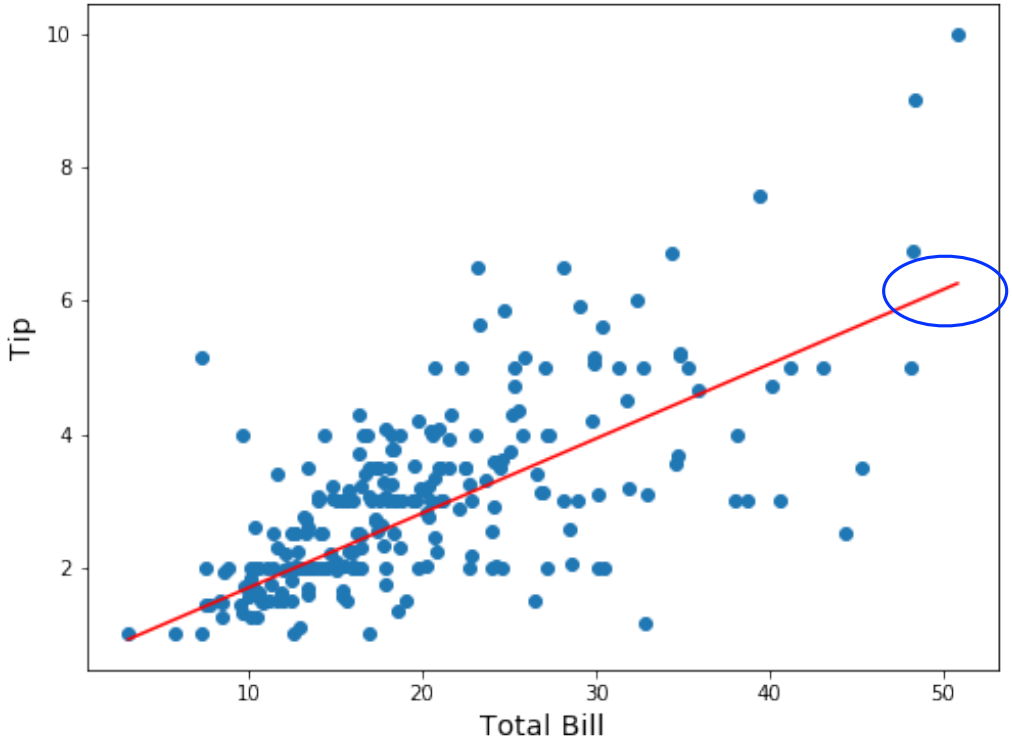
y = 0.92 + 0.105(xi)



Example: Linear Regression

- Now that we have a built our regression model, we can use it to make predictions - i.e. predict the tip amount, based on some specified amount for the total bill.

Bill	*Predicted Tip*
€10.00	€1.97
€15.00	€2.50
€20.00	€3.02
€25.00	€3.55
€30.00	€4.07
€35.00	€4.60
€40.00	€5.12
€45.00	€5.65
€50.00	€6.17
€55.00	€6.70
€60.00	€7.22
€65.00	€7.75

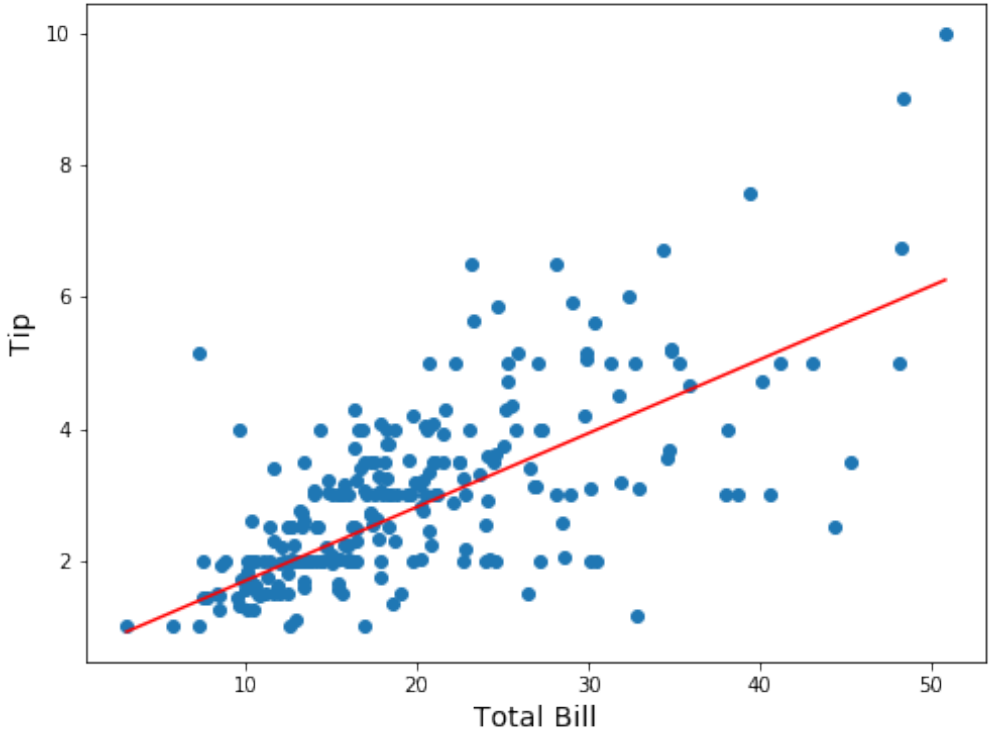


e.g. Predicted tip for €50 meal is €6.17

Example: Linear Regression

- We can also compare the output of our model with the original data to see if it agrees. When we look at the first 10 rows of the training data, we see there are some errors. Our regression model does not fit the data perfectly.

Bill	*Predicted Tip*	*Actual Tip*
€16.99	€2.70	€1.01
€10.34	€2.01	€1.66
€21.01	€3.13	€3.50
€23.68	€3.41	€3.31
€24.59	€3.50	€3.61
€25.29	€3.58	€4.71
€8.77	€1.84	€2.00
€26.88	€3.74	€3.12
€15.04	€2.50	€1.96
€14.78	€2.47	€3.23



Example: Linear Regression

Example: What is the relationship between budget spent on different advertising media and product sales?

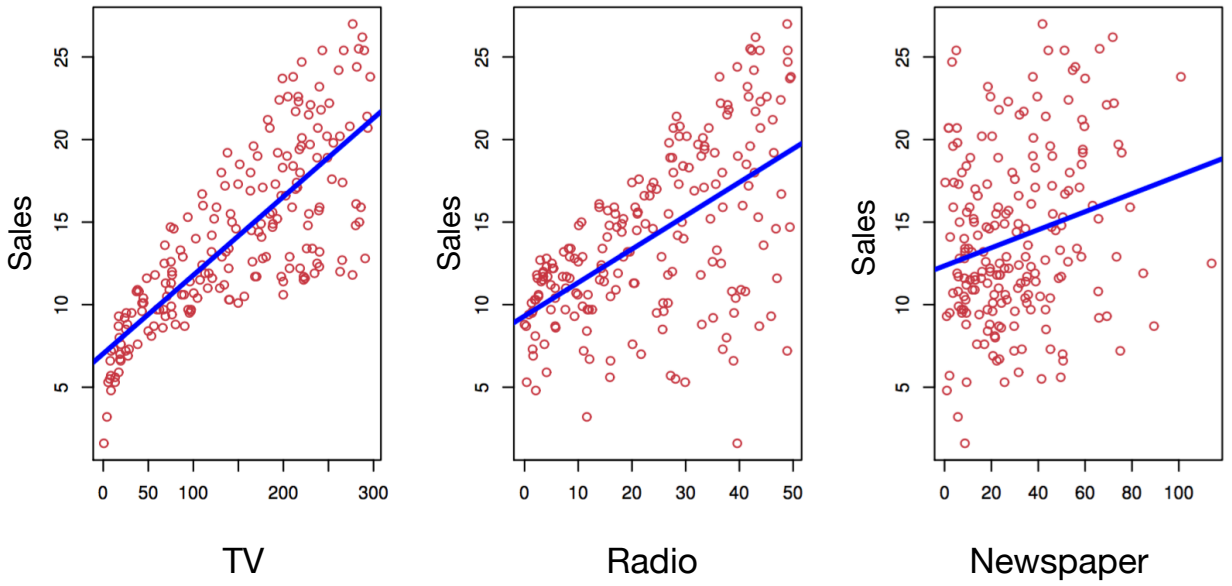
TV	Radio	Newspaper	Sales
230.10	37.80	69.20	22.10
44.50	39.30	45.10	10.40
17.20	45.90	69.30	9.30
151.50	41.30	58.50	18.50
180.80	10.80	58.40	12.90

Input Features:

1. TV: Budget (€) spent on TV advertising
2. Radio: Budget (€) spent on Radio advertising
3. Newspaper: Budget (€) spent on print advertising

Response:

- Sales: Overall product sales (€)



Which advertising medium has the greatest impact on product sales?

Can we predict future sales?

We could examine scatter plots of each feature vs sales from our historic data

(Hastie & Tibshirani)

Example: Linear Regression

- **Example:** What is the relationship between budget spent on different advertising media and product sales?

```
from sklearn.linear_model import LinearRegression
df = pd.read_csv("advertising.csv", index_col=0)
x = df[["TV"]]
```

Let's examine the relationship between TV budget and sales

Create and fit the model

```
model = LinearRegression()
model.fit(x, df["Sales"])
```

Get the intercept coefficient

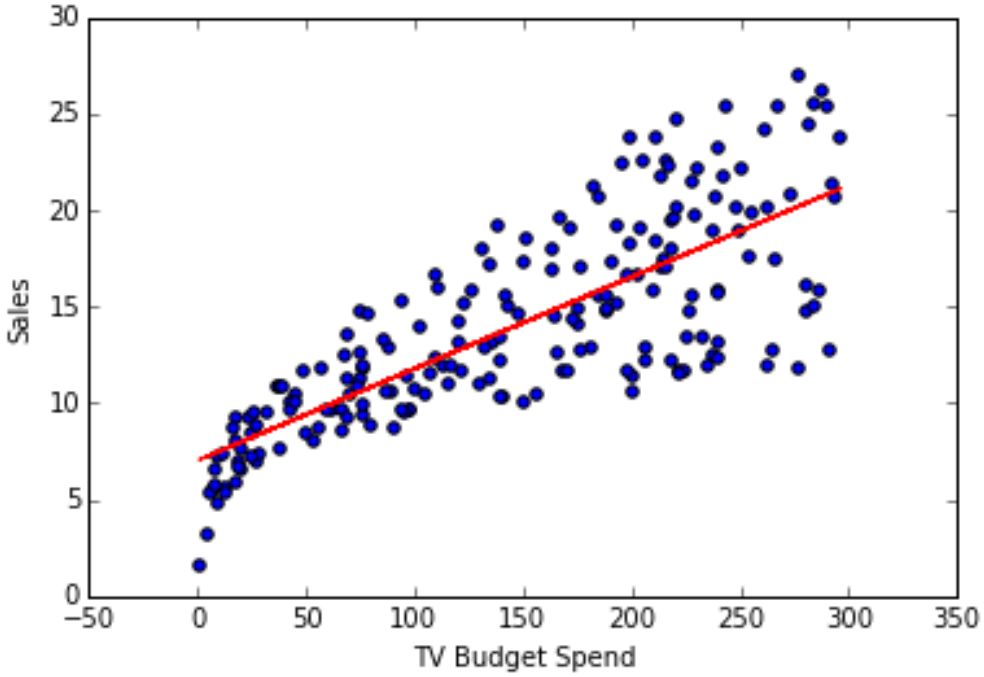
```
model.intercept_
```

```
7.03259354913
```

Get the slope coefficient

```
model.coef_[0]
```

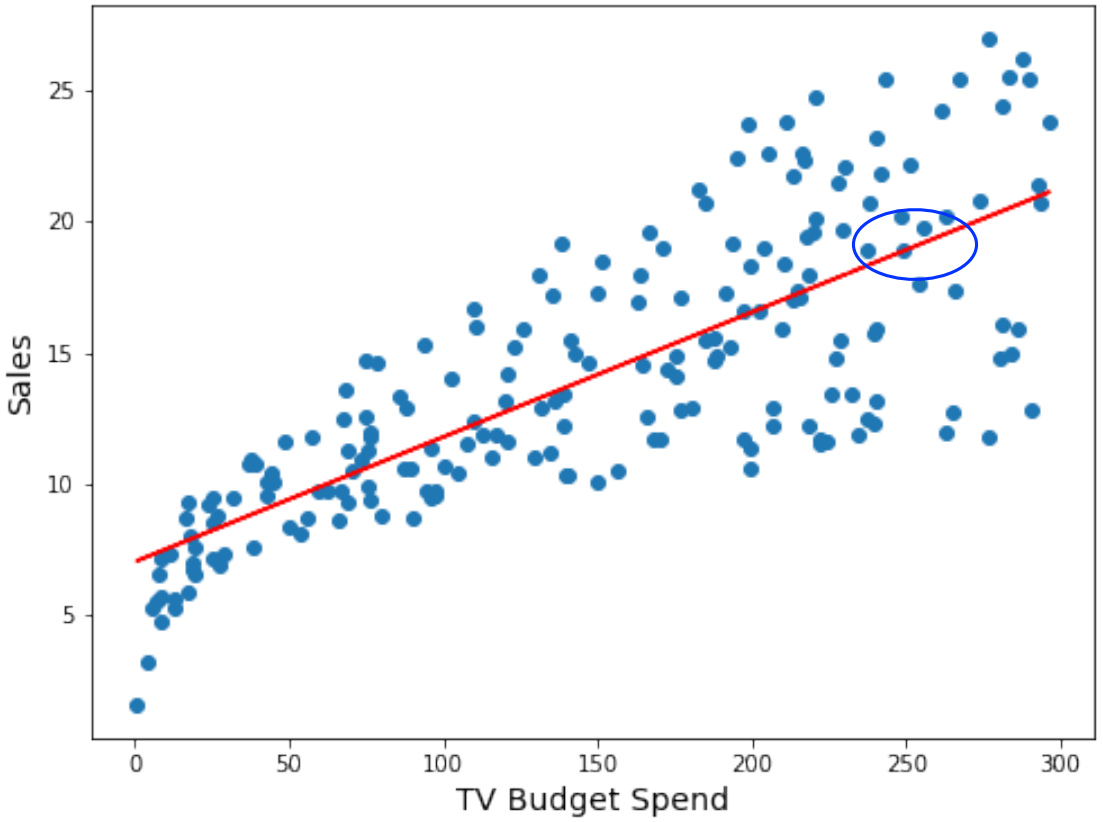
```
0.047536640433
```



Example: Linear Regression

- Now that we have a built our regression model, we can use it to make predictions - i.e. predict sales revenue, based on TV advertising budget spend.

Budget	*Predicted Sales*
€0.00	€7.03
€50.00	€9.41
€100.00	€11.79
€150.00	€14.16
€200.00	€16.54
€250.00	€18.92
€300.00	€21.29
€350.00	€23.67



e.g. Predicted sales for €250 budget is €18.92

Errors

- When we want to measure the strength of the linear relationship in the data
- First, consider the sources of error in regression

$$SST = \sum_{i=1}^{n} (Y_i - \bar{Y})^2$$

Sum of Squares Total

total sum of squared deviations in Y (observed dependent variable) from its mean

$$SSR = \sum_{i=1}^{n} (\hat{Y}_i - \bar{Y})^2$$

Sum of Squares Regression

total sum of squared deviation of the best fit (value predicted by regression) from the mean

predict

$$SSE = \sum_{i=1}^{n} (\hat{Y}_i - Y_i)^2$$

Sum of Squares Error

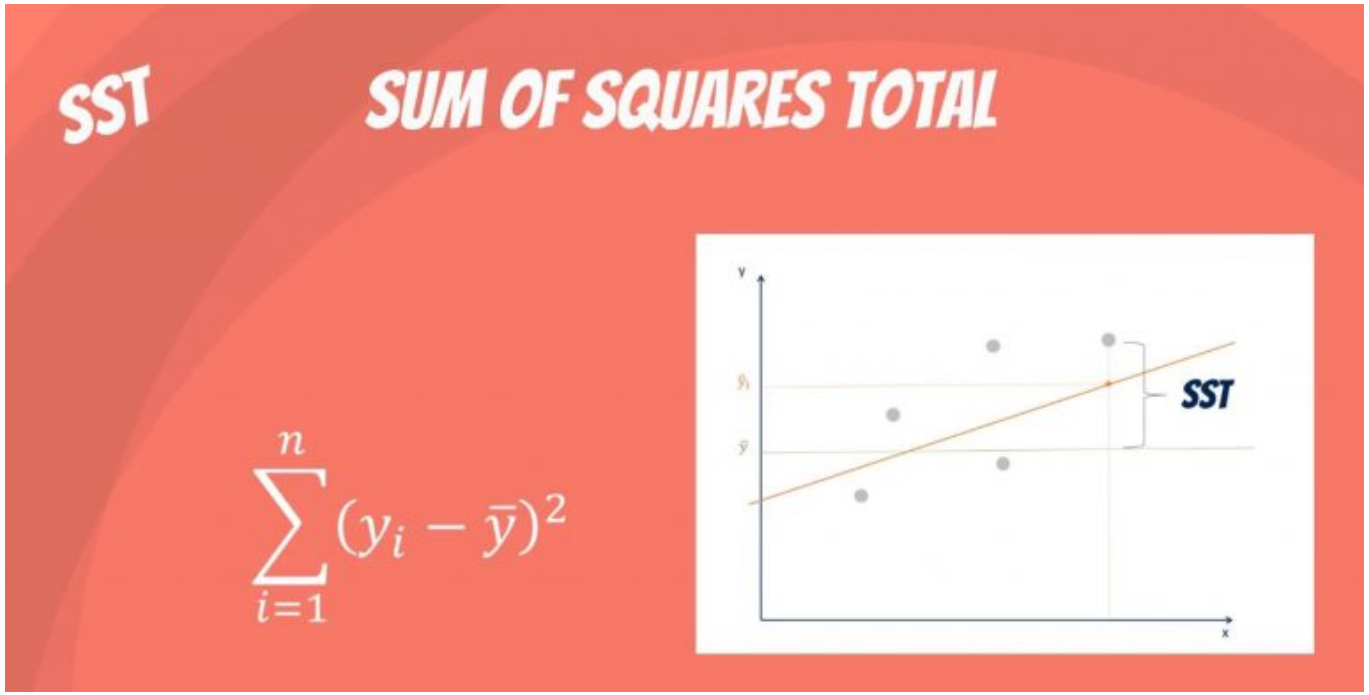
total sum of squared residuals (difference between fit and the observed data)

SSR: Variation in *Y* explained by the regression line.

SSE: Variation in *Y* that is left unexplained.

What is the Sum of Squares Total (SST)?

The **sum of squares total**, denoted **SST**, is the squared differences between the observed *dependent variable* and its **mean**. You can think of this as the dispersion of the observed variables around the **mean** – much like the **variance** in descriptive statistics.



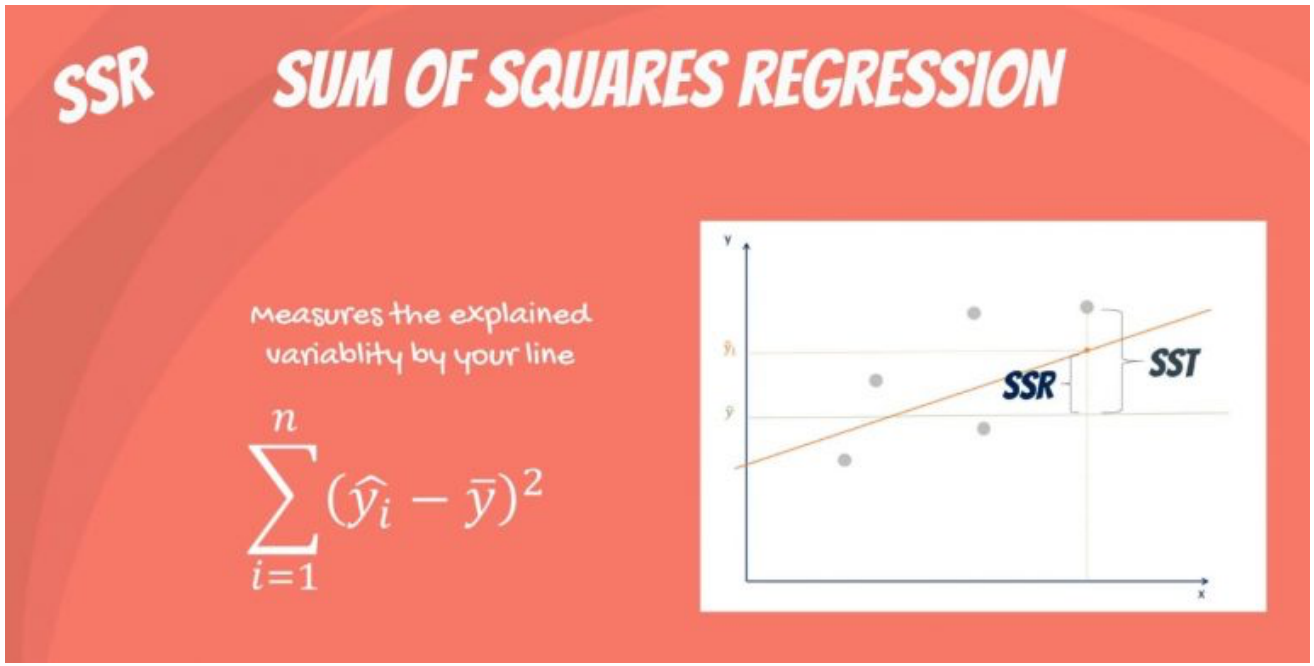
It is a measure of the total variability of the dataset.

There is another notation for the **SST**. It is **TSS** or ***total sum of squares***.

https://365datascience.com/sum-squares/

What is the Sum of Squares Regression (SSR)?

The second term is the **sum of squares due to regression**, or **SSR**. It is the sum of the differences between the *predicted* value and the **mean** of the *dependent variable*. Think of it as a measure that describes how well our line fits the data.

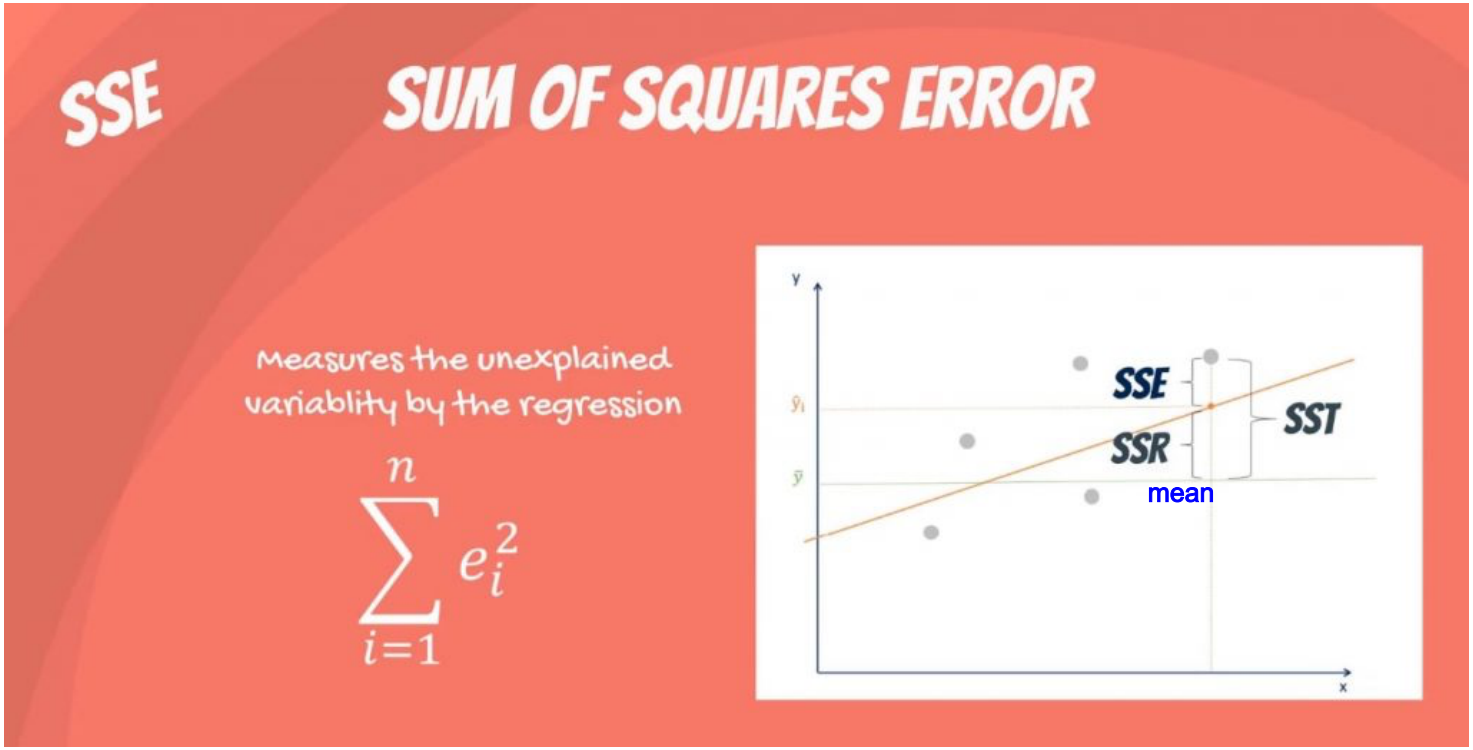


If this value of **SSR** is equal to the **sum of squares total**, it means our **regression model** captures all the observed variability and is perfect. Once again, we have to mention that another common notation is **ESS** or ***explained sum of squares***.

https://365datascience.com/sum-squares/

What is the Sum of Squares Error (SSE)?

The last term is the **sum of squares error**, or **SSE**. The error is the difference between the *observed* value and the *predicted* value.

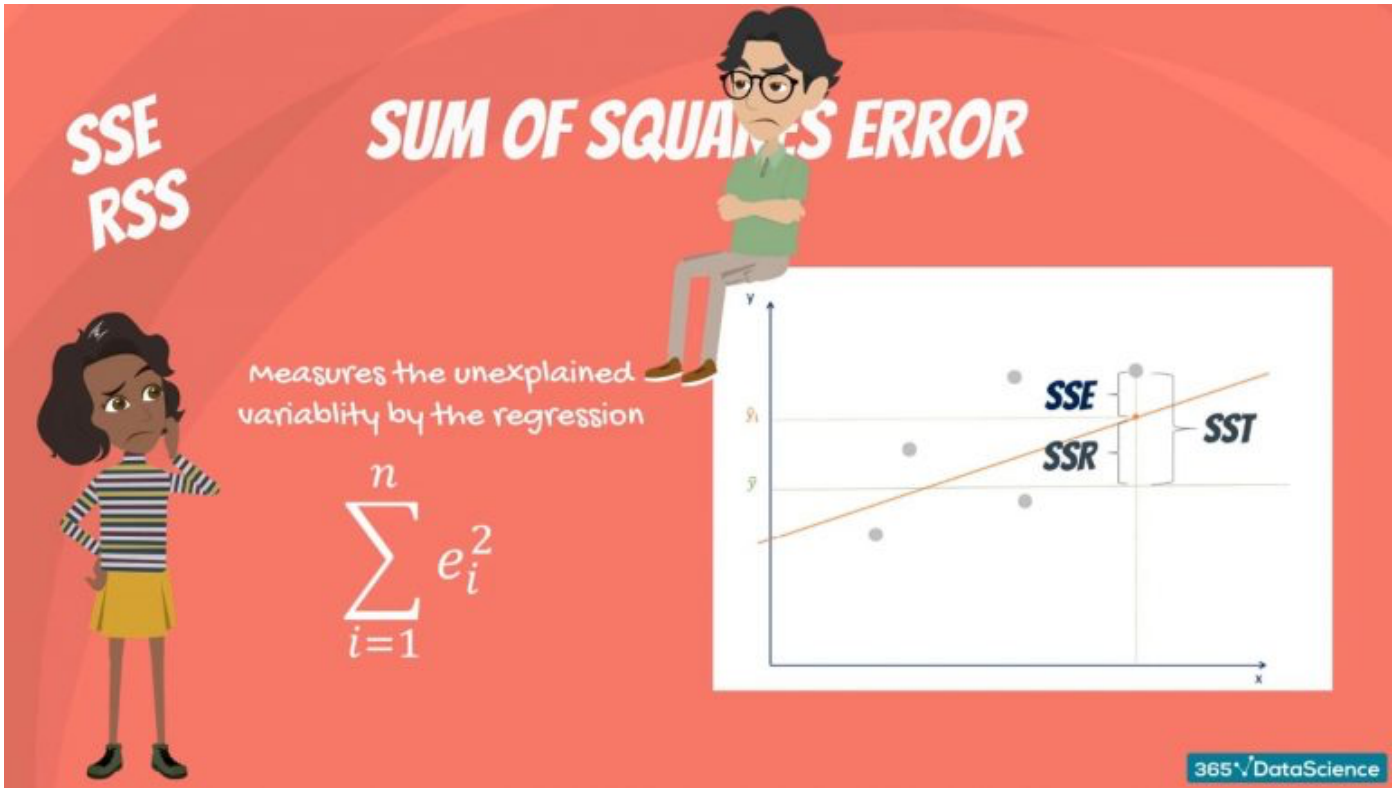


We usually want to minimize the error. The smaller the error, the better the estimation power of the **regression**. Finally, I should add that it is also known as **RSS** or ***residual sum of squares***. Residual as in: remaining or unexplained.

https://365datascience.com/sum-squares/

The Confusion between the Different Abbreviations (1)

It becomes really confusing because some people denote it as **SSR**. This makes it unclear whether we are talking about the **sum of squares due to regression** or **sum of squared residuals**.

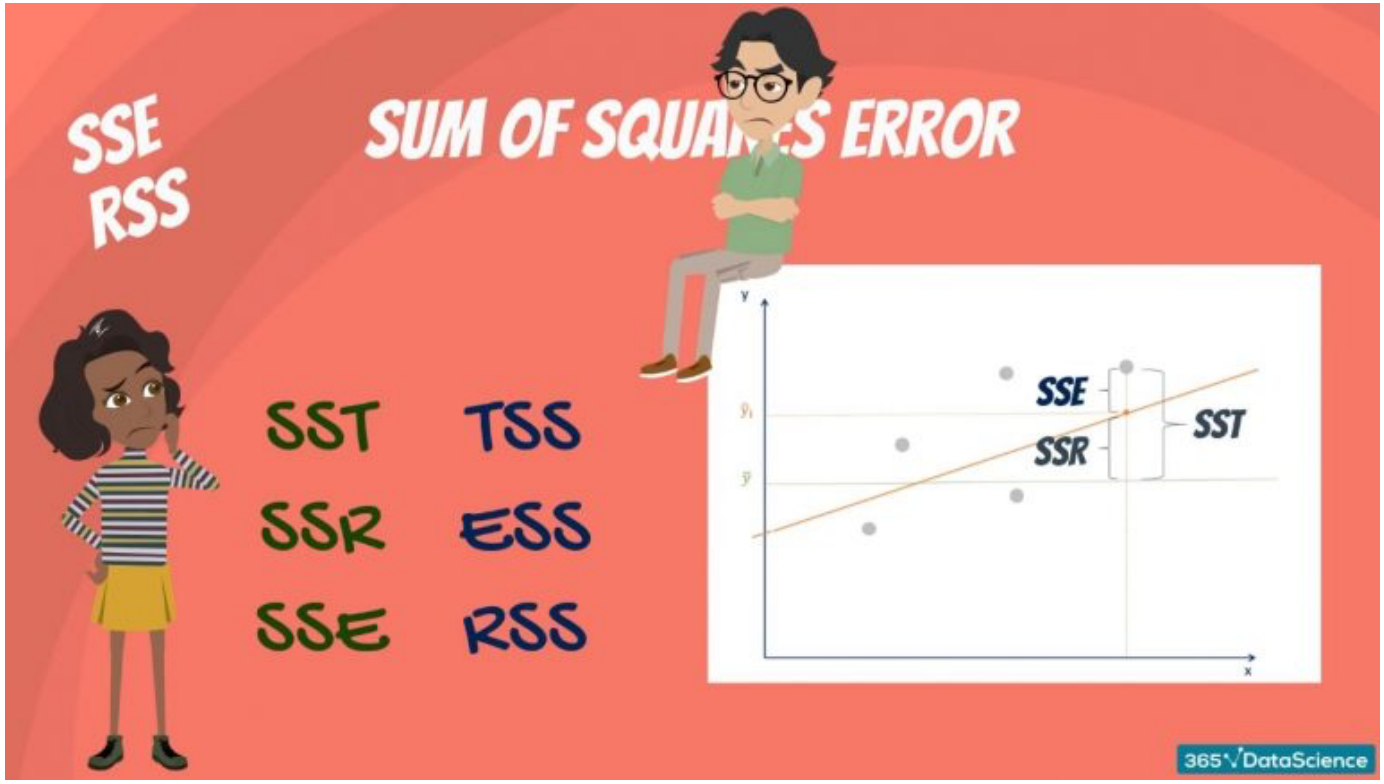


In any case, neither of these are universally adopted, so the confusion remains and we'll have to live with it.

https://365datascience.com/sum-squares/

The Confusion between the Different Abbreviations (2)

Simply remember that the two sets of notations are {**SST**, **SSR**, **SSE**} or {**TSS**, **ESS**, **RSS**}

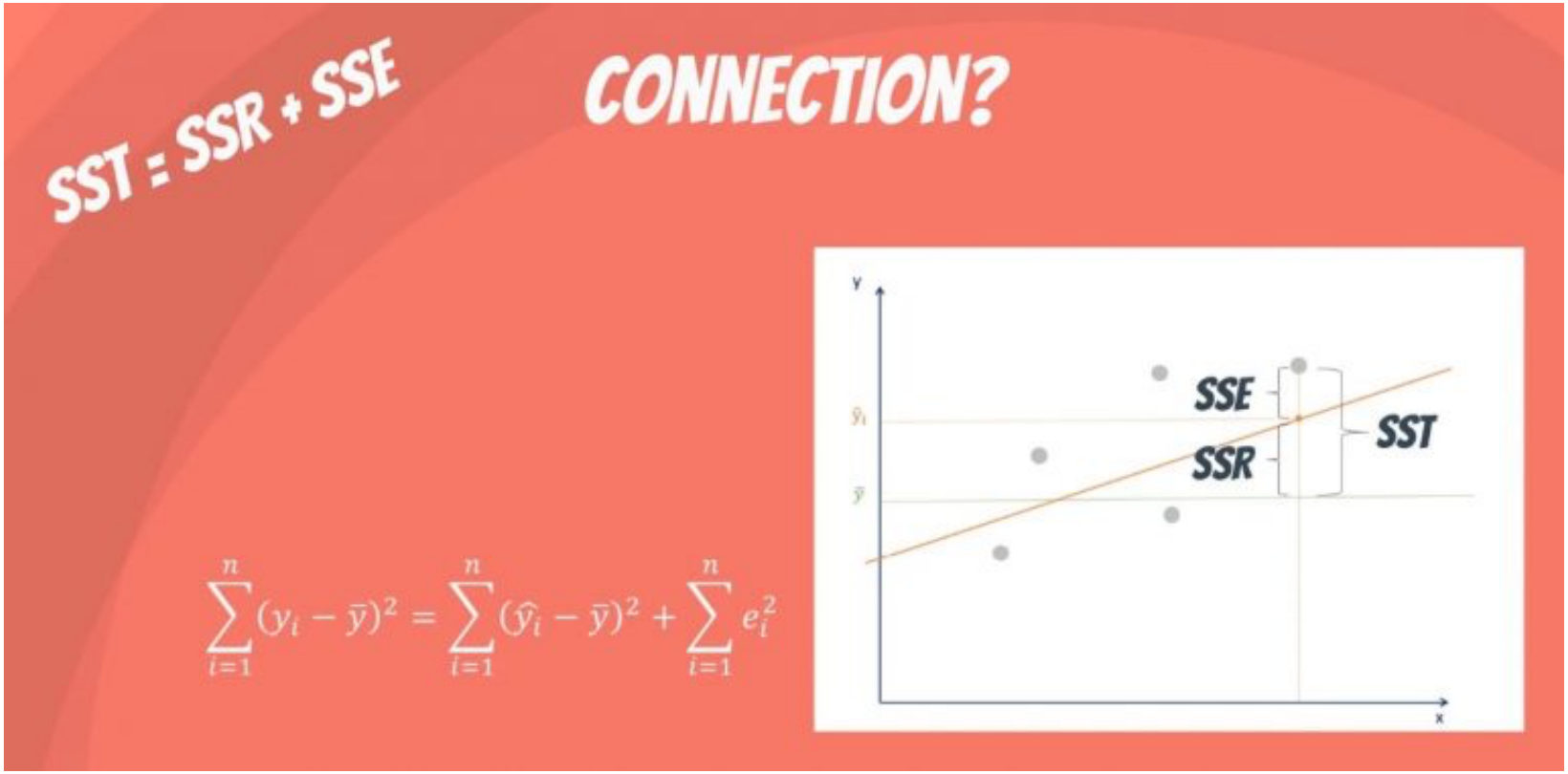


There's a conflict regarding the abbreviations, but not about the concept and its application. So, let's focus on that.

https://365datascience.com/sum-squares/

How Are They (SST, SSR and SSE) Related? (1)

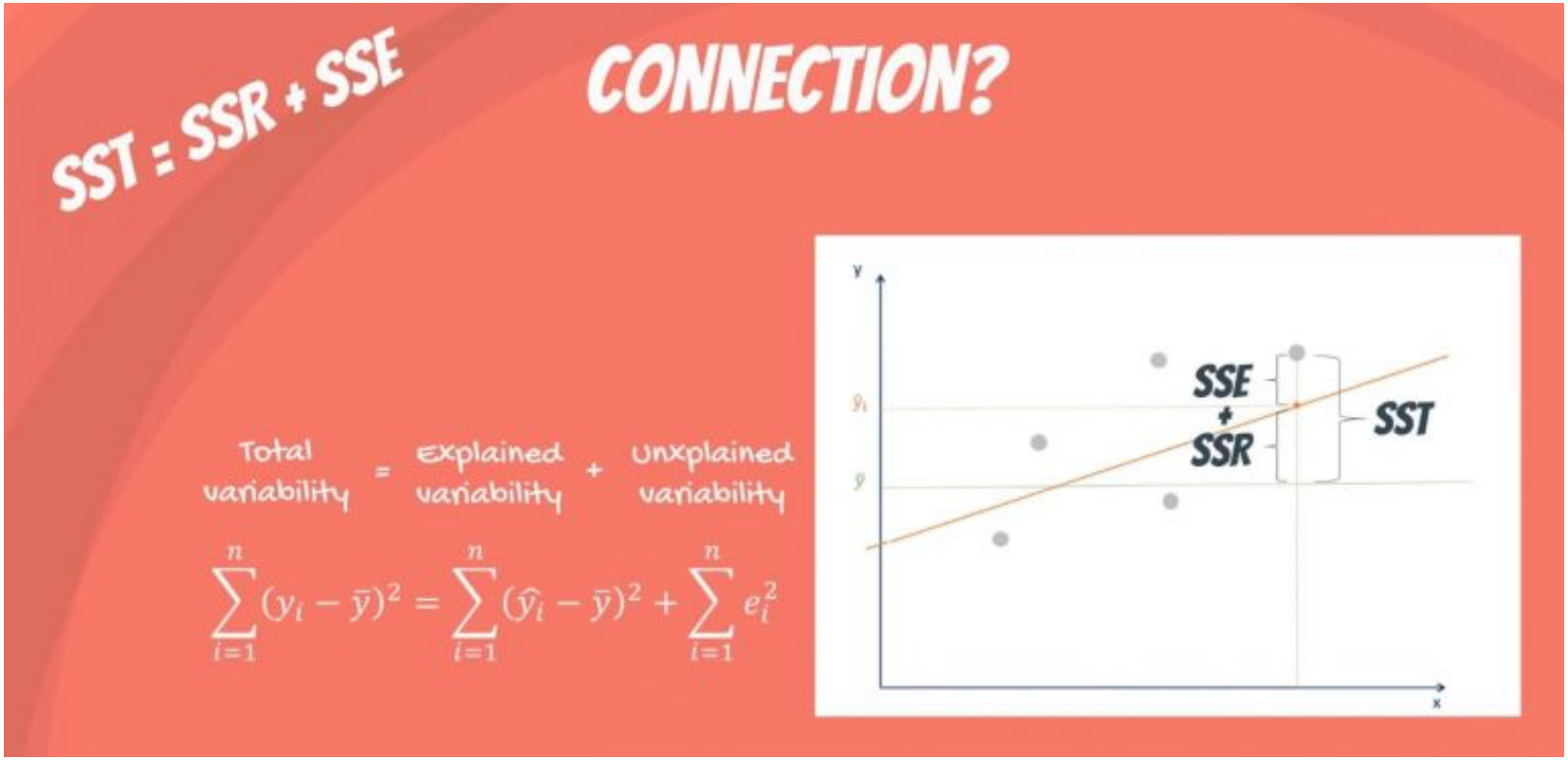
Mathematically, **SST** = **SSR** + **SSE**.



The rationale is the following: the total variability of the data set is equal to the variability explained by the **regression line** plus the unexplained variability, known as error.

https://365datascience.com/sum-squares/

How Are They (SST, SSR and SSE) Related? (2)



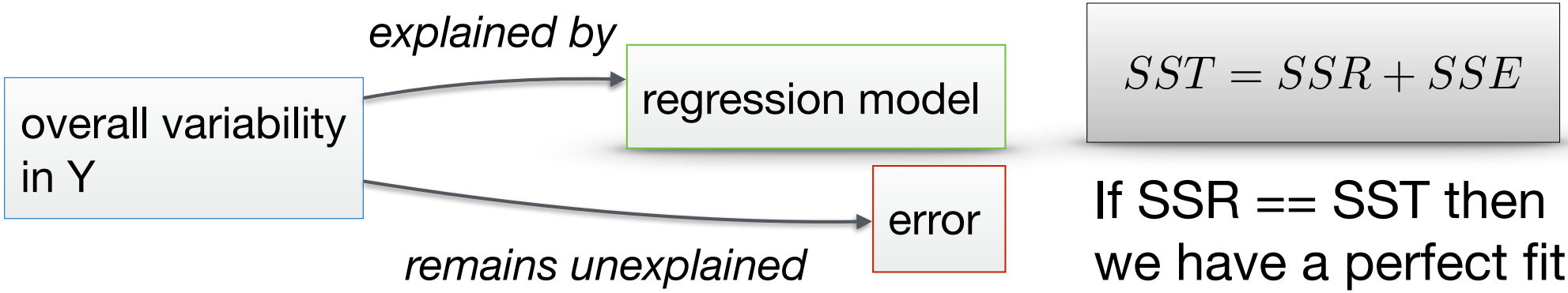
Given a constant total variability, a lower error will cause a better **regression**. Conversely, a higher error will cause a less powerful **regression**. And that's what you must remember, no matter the notation.

https://365datascience.com/sum-squares/

Errors

- When we want to measure the strength of the linear relationship in the data
- First, consider the sources of error in regression

$SST = \sum_{i=1}^{n}(Y_i - \bar{Y})^2$	total sum of squared deviations in Y from its mean
$SSR = \sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2$	total sum of squared deviation of the best fit from the mean
$SSE = \sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2$	total sum of squared residuals (difference between fit and the observed data)

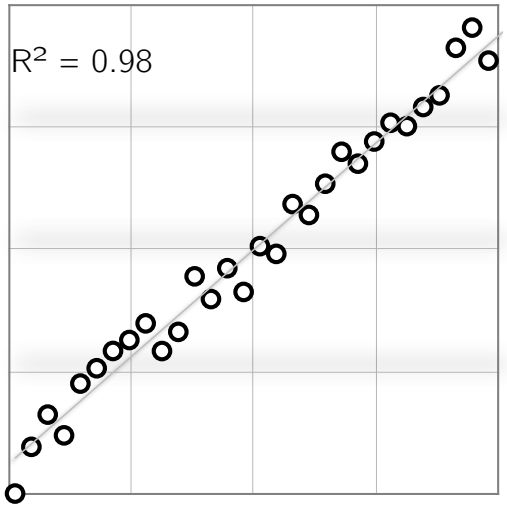


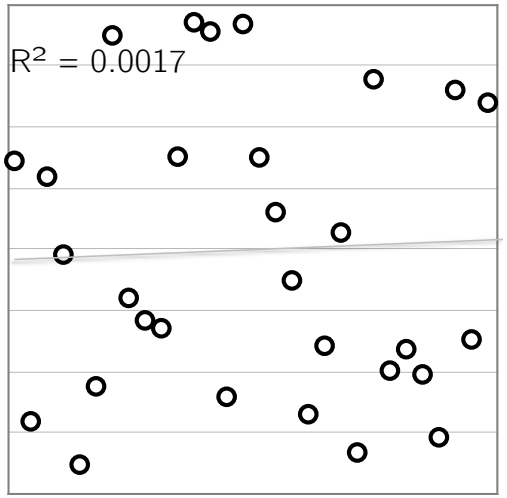
How well did we predict?

- If there is linear relationship the slope (β_1) will be non-zero.
- start with hypothesis that $\beta_1 = 0$ (this implies there is no linear relationship between the variables *Y* and *X*)
- we would like to test this hypothesis
- this will give us a statistical measure of how certain we can be there is a linear relationship in the data.

strong *+ve* relationship

β_1 is > 0





no relationship

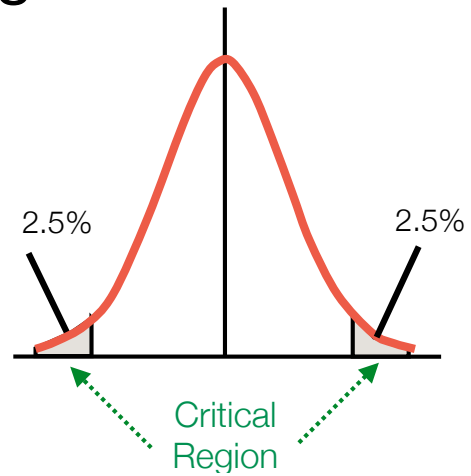
β_1 is 0

t-test

- Having determined the slope of our best fit line, we would like to make an inference about the slope by testing:

$$H_0: \quad \hat{\beta}_1 = 0 \quad \text{null hypothesis}$$

$$H_1: \quad \beta_1^0 \neq 0$$



- The appropriate test is a two-sided t-statistic with (n-2) degrees of freedom

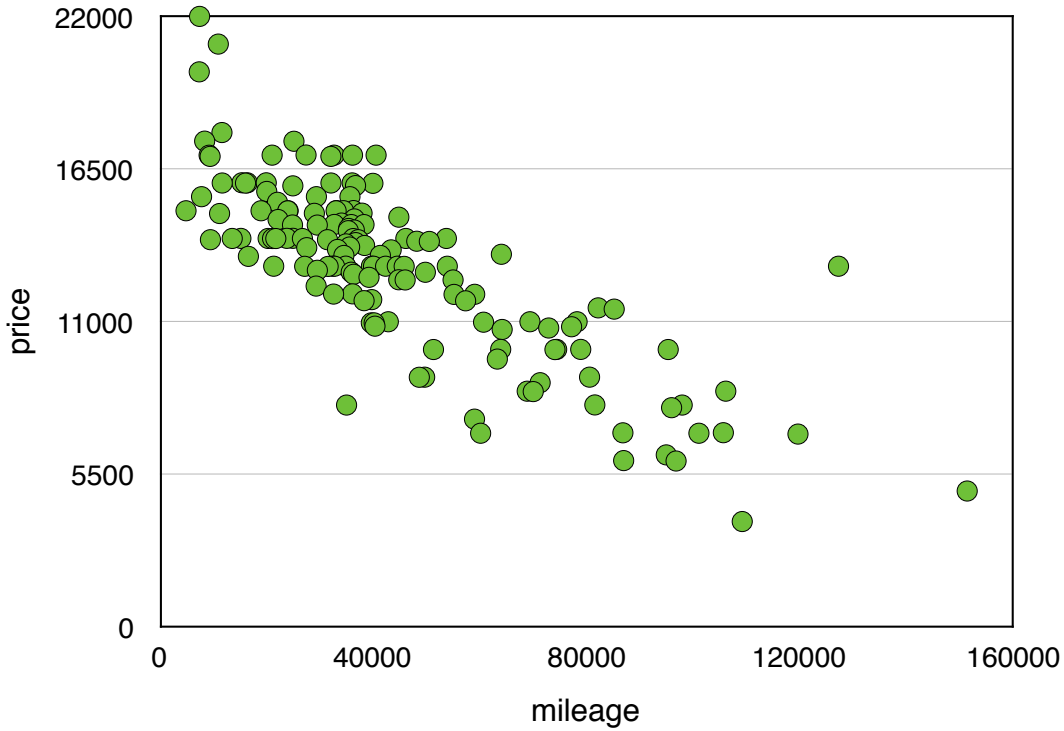
$$s.e(\hat{\beta}_1) = \sqrt{\frac{1}{n-2} \frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n} (X_i - \bar{X})^2}}$$

$$t = \frac{\hat{\beta}_1 - \beta_1^0}{s.e(\hat{\beta}_1)}$$

https://www.analystforum.com/forums/cfa-forums/cfa-level-iii-forum/9852893

Example t-test

- used cars are bought at auction and the dealer would like to be able to predict the sale price given the mileage on the odometer.
- the best fit linear model is:

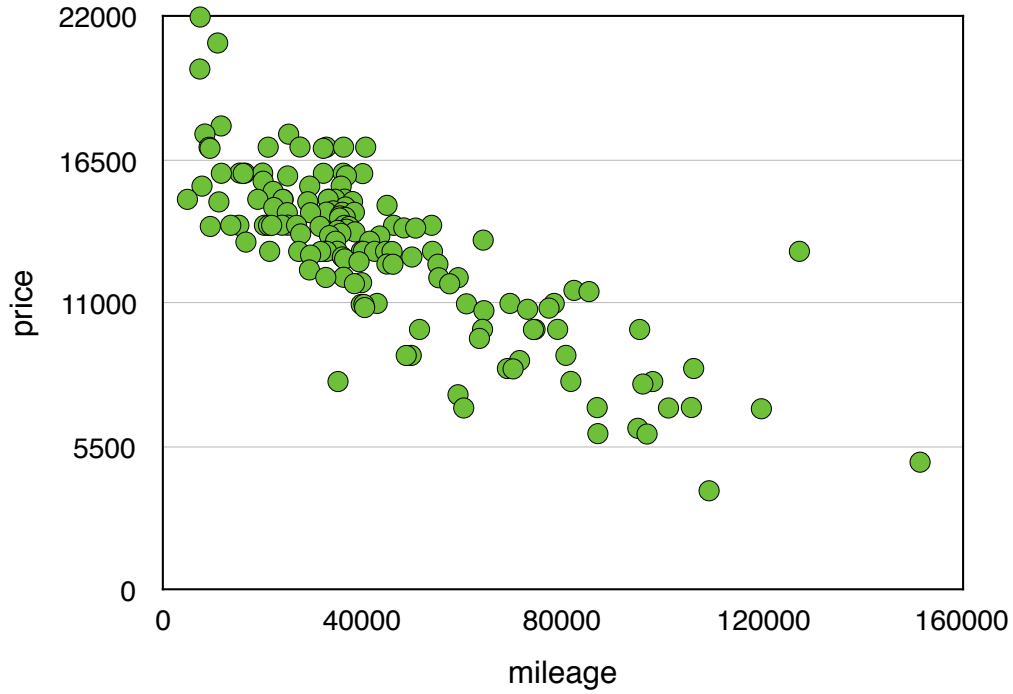


price = -0.093 * mileage + 17091.05

- Is a linear model a good fit to the data (t-test)?
- How closely does the line fit (Cor)?

Example t-test

- compute the sum of the squared error, which tells us how close the fit is to the actual data



$$s.e(\hat{\beta}_1) = \sqrt{\frac{1}{n-2}\frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}$$

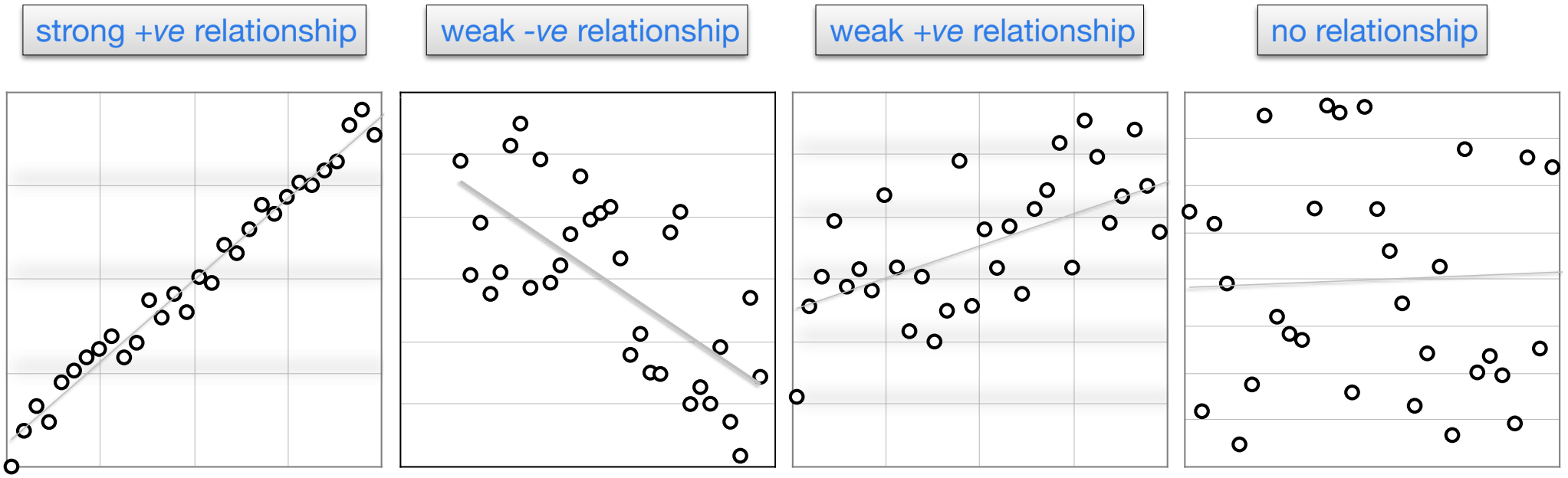
$$t = \frac{\hat{\beta}_1 - 0}{s.e(\hat{\beta}_1)}$$
$$= \frac{-0.093}{0.00563} = -16.657$$

- is this significant? Ans: Yes
- we can accept H_0 with 95% confidence if:

$$-t_{\alpha/2,n-2} < t < t_{\alpha/2,n-2}$$
$$-1.976 < t < 1.976 \quad (\alpha = 5\%)$$

Measuring Performance of Regression by Correlation (1)

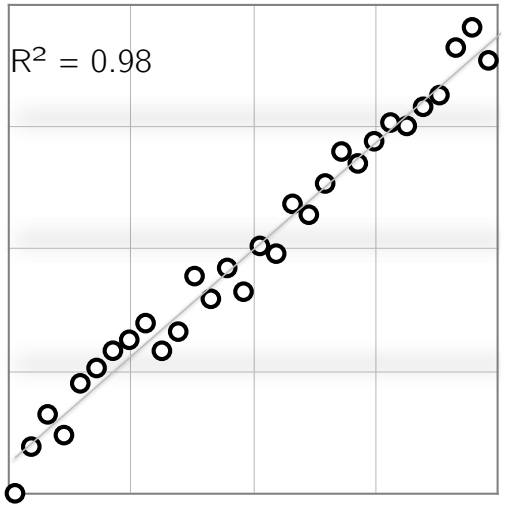
- Visualising data using scatter plots can provide us with a sense of the relationship between two variables. Relationships can have different directions (+ve, -ve) and different strengths.
- How do we quantify this numerically?



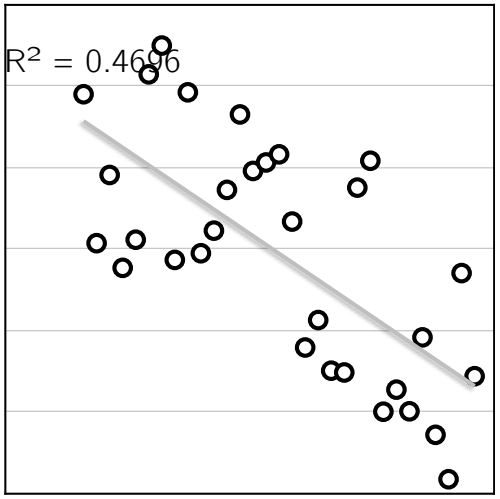
Measuring Performance of Regression by Correlation (2)

- Quantify the relationships between two variables, i.e., our regression model and test data, by R^2 correlation coefficients

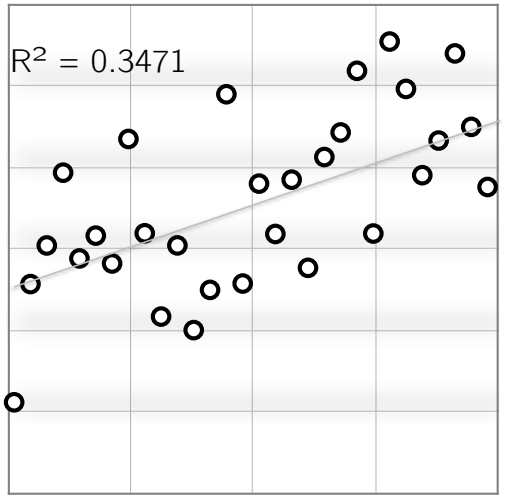
strong *+ve* relationship



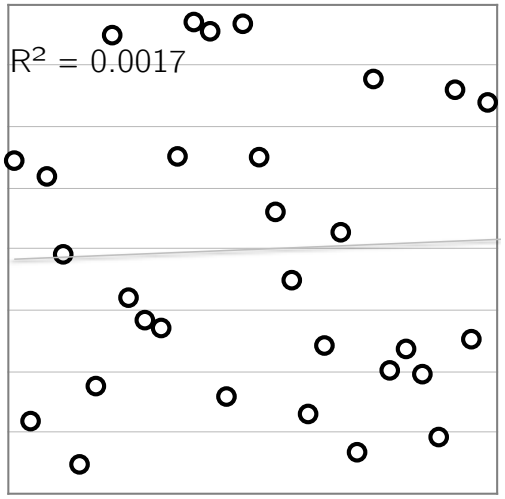
weak *-ve* relationship



weak *+ve* relationship



no relationship



R^2 Correlation

- When we want to measure the strength of the linear relationship, we use the coefficient of determination R^2.
- R^2 is the fraction of the variation which is explained by the linear regression model.
- It measures the explanatory power of the model
- tells us how well our regression line matches the real data. The closer R^2 is to 1 the better the fit.

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

correlation coefficient R^2

Does not tell us if *X* is the cause of changes in *Y*

http://blog.uwgb.edu/bansalg/statistics-data-analytics/linear-regression/what-is-the-difference-between-coefficient-of-determination-and-coefficient-of-correlation/

R^2 Correlation

- R^2 is a *goodness-of-fit* index
- $0 \leq R^2 \leq 1$ because SSR≤SST.
- If R^2 is near 1 then it means that *X* accounts for a large part of the variation in *Y*.
- it gives us an idea of how the predictor variable *X* accounts for, or determines the response variable *Y*.
- known as R^2 or r^2, also known as the ***square*** *of* Pearson coefficient, or coefficient of determination

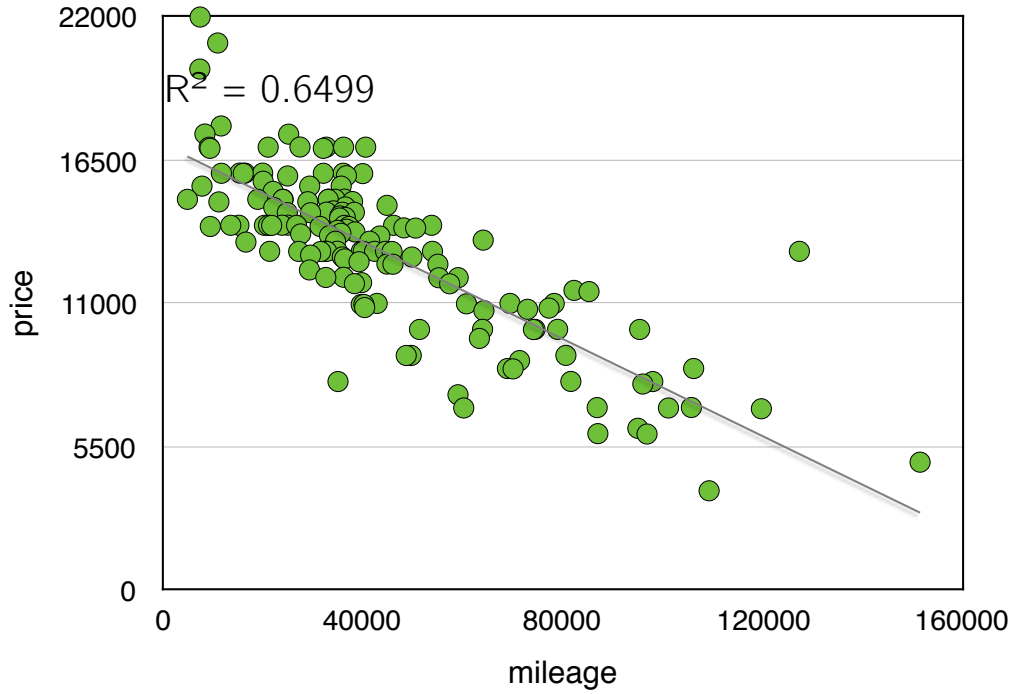
$$R^2 = [Cor(Y, X)]^2$$

simply related to the Cor(*Y*, *X*)

Example R^2

Cor(*Y*, *X*) = -0.8062

$$R^2 = [Cor(Y, X)]^2$$
$$= (0.8062)^2 = 0.650$$



- Car price at auction
- 65% of the total variability in the price is accounted for by the mileage of the car
- This indicates a reasonably strong linear relationship between the price at auction and the mileage on the odometer

Multiple Regression

- Multiple linear regression: Simple linear regression can easily be extended to include multiple features, where we try to learn a model: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots$
- Each feature x_i has its own coefficient β_i

TV	Radio	Newspaper	Sales
230.10	37.80	69.20	22.10
44.50	39.30	45.10	10.40
17.20	45.90	69.30	9.30
151.50	41.30	58.50	18.50
180.80	10.80	58.40	12.90

e.g. Predict sales based on 3 features

$$y = \beta_0 + \beta_1 \times \text{TV} + \beta_2 \times \text{Radio} + \beta_3 \times \text{Newspaper}$$

Remove column we want to predict

```
x = df.drop("Sales",axis=1)
```

Fit the model based on all 3 features

```
model = LinearRegression()
model.fit(x,df["Sales"])
```

We can now use the model to make sales predictions for unseen values of *(TV, Radio, Newspaper)*

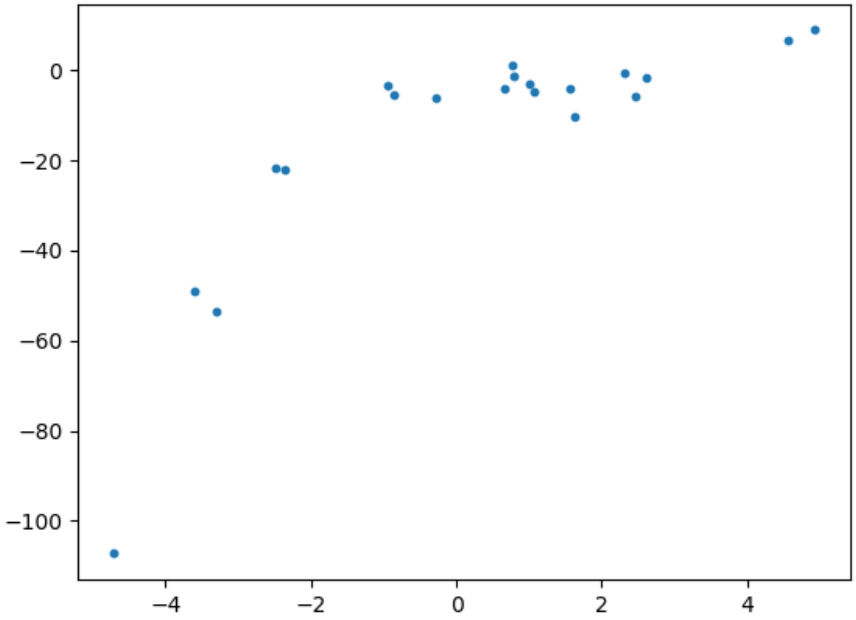
```
model.predict(test_x)
```

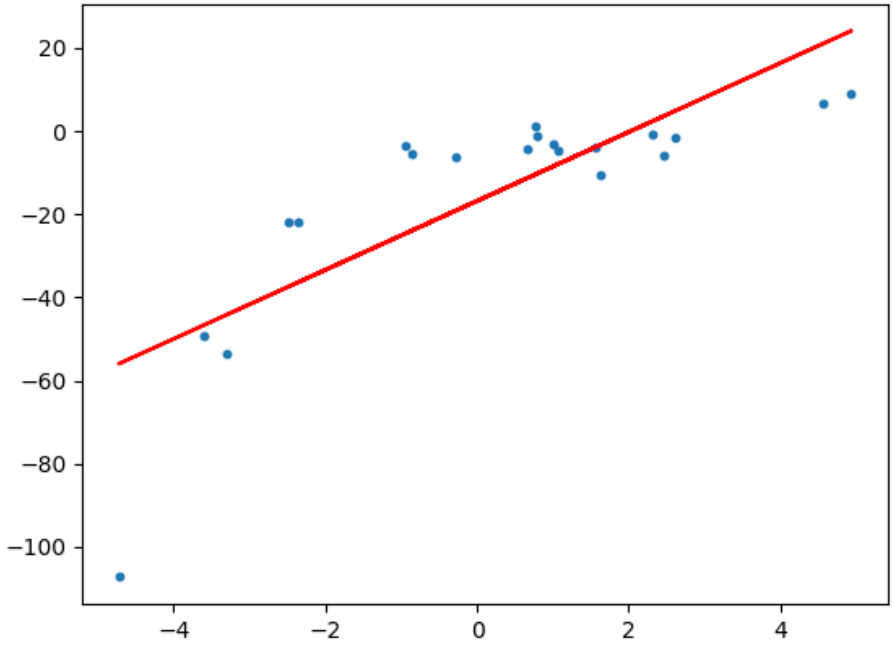
```
20.52
```

Polynomial Regression (1)

Why Polynomial Regression?

Because sometime the straight line is unable to capture the patterns in the data. This is an example of *underfitting*.

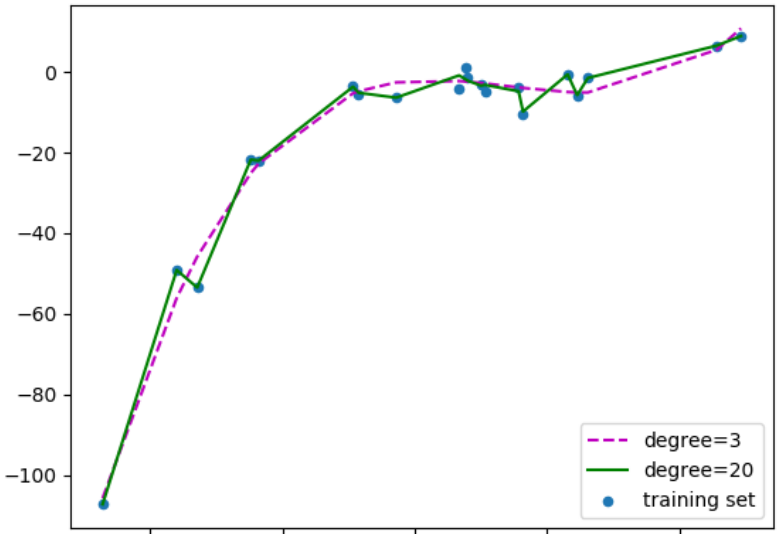


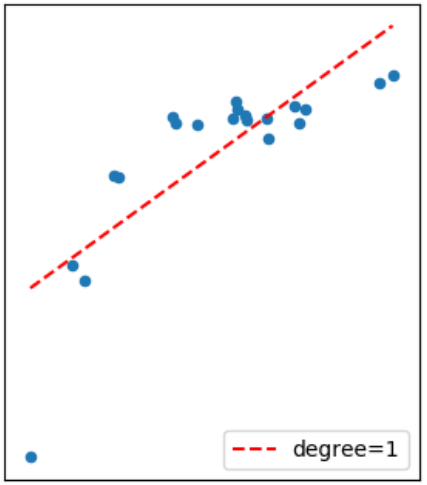


Further reading

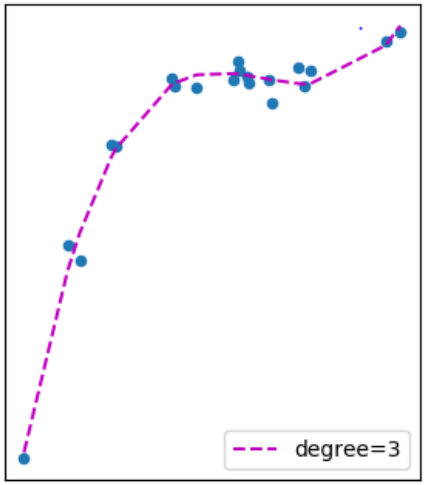
https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

Polynomial Regression (2)

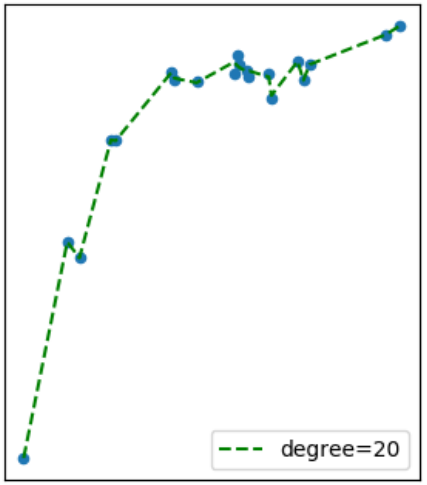




Underfit
High Bias
Low Variance



Correct Fit
Low Bias
Low Variance

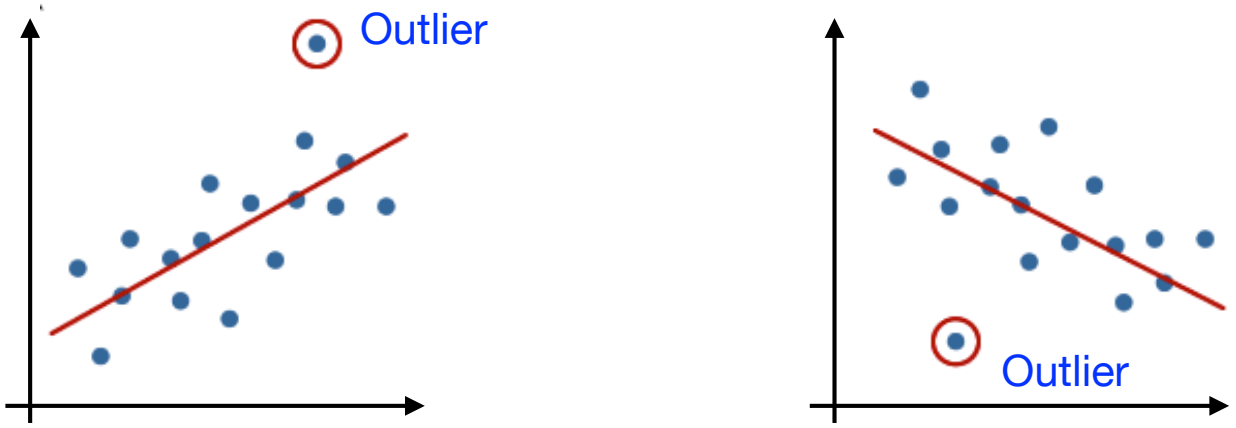


Overfit
Low Bias
High Variance

https://towardsdatascience.com/polynomial-regression-bbe8b9d97491

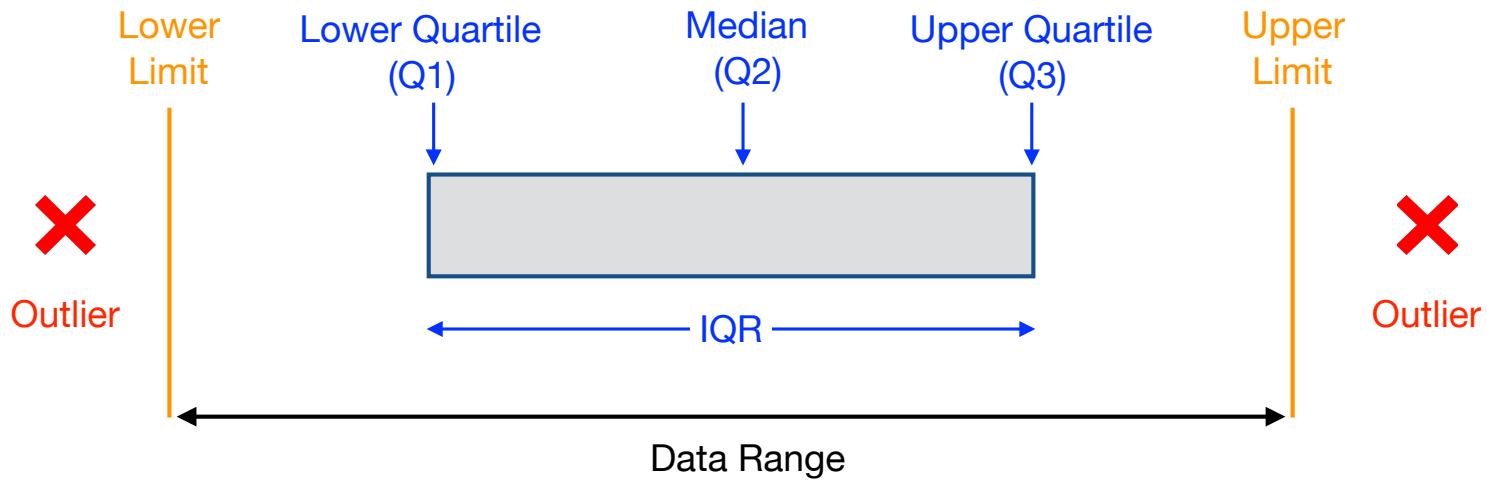
Outliers in Regression

- Outlier: An observed data point that has a dependent variable value which is very different to the value predicted by the regression equation.
- Linear regression models assume there should be no significant outliers, as they can lead to a very poor regression fit.
- In some cases we might drop the outlier and recalculate the regression model. But it is always important to investigate the nature of the outlier before deciding whether to drop.



Finding Outliers

- Box plot diagram: a graphical method which helps in defining the upper limit and lower limits beyond which any data points lying will be considered as outliers.
- Median (Q2): the middle value of the dataset.
- Lower quartile (Q1): the median of the lower half of the dataset.
- Upper quartile (Q3): the median of the upper half of the dataset.
- Interquartile range (IQR): the spread of the middle 50% of the data values = Q3 - Q1



Boxplots in Python

- We can create boxplots in Python to visualise the distributions of values for different variables, and look for any potential outliers.

Create a boxplot with Matplotlib:

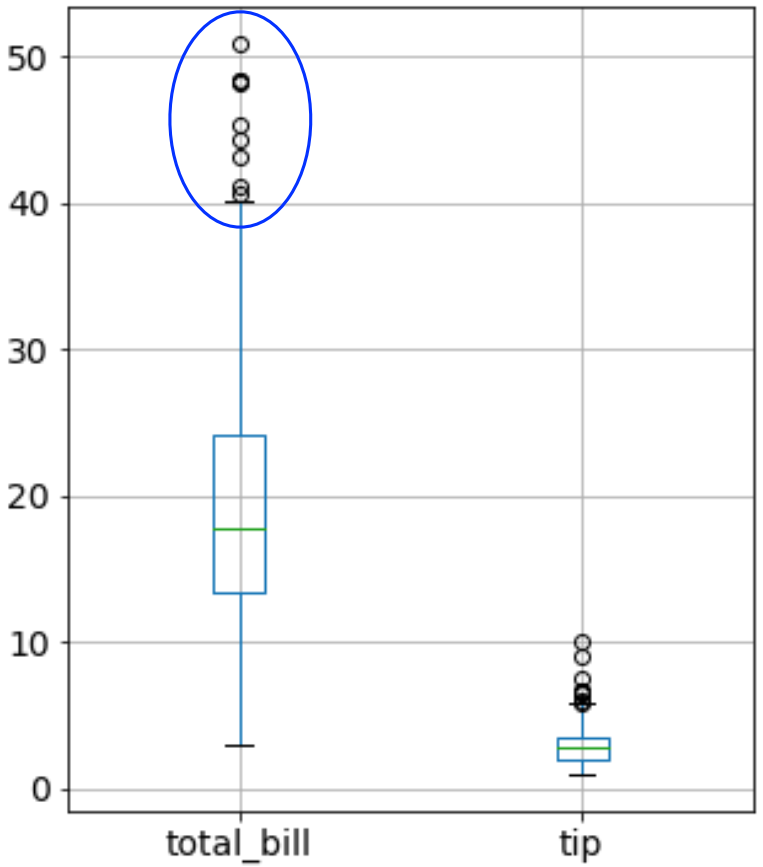
```
plt.figure()
plt.boxplot(data)
```

Create a boxplot from all of the columns in a Pandas Data Frame:

```
df.boxplot()
```

- Remember, it is always important to investigate the nature of an outlier before deciding whether to drop.

Possible outliers?



Summary

- We looked at the overview of different discriminant functions, i.e. linear regression, logistic regression, and support vector machine.
- Pruning a decision tree can somewhat prevent the tree model from overfitting to noise in data. And, pruning can be applied before or after tree induction, called "Pre-pruning" or "Post-pruning". The latter is preferred in practice.
- We looked at least squares fit and interpretation.
- We looked at simple and multiple linear regression models.
- We looked at how to measure the strength of the linear relationship in the data, using the measures of errors, i.e. SST, SSR and SSE.
- We also looked at how to quantify the relationship between two variables using the coefficient of determination R^2.