CZ1115 Topic ONE:

Data Analytic Thinking

# RAW DATA -> ACTIONABLE INTELLIGENCE

## Real-life problems translated in data

* || Practical motivation: can the real-life problem be related to data?
* //Problem formulation: what is the data science problem that you can formulate and solve?
* || Sample collection: can you collect relevant and representative data?
* //Data preparation: is the data clean enough to analyse?

## Descriptive/inferential analytics

* || Statistical description: how to describe, summarise the data?
* || Exploratory analysis: how to explore, mine and gain basic insights from data?
* //Pattern recognition: how to discover unknown traits?
* //Analytic visualisation: how to represent and convey the data to humans?
* \\Machine learning: how to formulate and automate the “learning”?
* \\Algorithmic optimisation: how to make the above “learning” optimal?

## Effective communication and decision

* || Statistical inference: how to confidently infer from the data?
* || Information presentation: how to present and communicate the analysis?
* //Intelligent decision: link back to practical motivation; how to solve real-life problem via data?
* //Ethical consideration

# ALGORITHMIC SOLUTION |

# Five Primary Questions

## How much? How many?

### Prediction: Numeric

### REGRESSION:

* + Find the relationship between the independent and dependent variables
* Model: Total = f(variables)
  + Total is the response, variables are the predictors
  + Linear regression models (univariate/multivariate)
  + Tree models for regression (decision trees/random forest)
  + Neural network for regression

## Is it type A or type B?

### Prediction: class (Boolean)

### CLASSIFICATION

* + Find the probability of something being true
* Model: P(event) = f(variables)
  + Logistic regression models
  + Tree models for classification
  + Neural network for classification
* *Note: distinguish this from a decision problem. A decision problem entails two choices to be done by an agent (e.g. you) which has consequences. A classification problem is simply determining a “characteristic” of an entity, such as gender, race, whether it is recyclable etc.*

## How is this organised?

### Detection: structure

### CLUSTERING

* + Find groups of data points that are close to each other but far apart from other groups
  + Close-far depends on “distance”
* Model:
  + Distance: Euclidean, Jaccard etc
  + Nearest Neighbour
  + k-Means algorithm for clustering
  + Hierarchial model for clustering

## Is it a weird behaviour?

### Detection: anomaly (Boolean)

### ANOMALY DETECTION

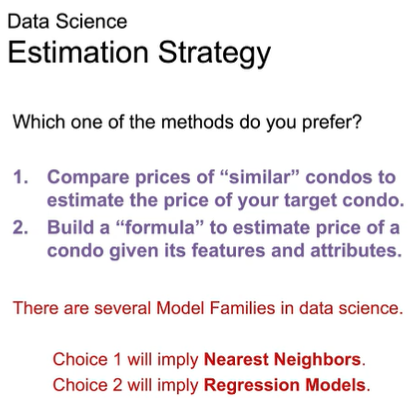
* + Find deviations of the data compared to the regular pattern observed through the data model
* Model:
  + Cluster-analysis based detection
  + Nearest neighbour detection model
  + Support vector based detection

## What should be done next?

### Decision: action

### ADAPTIVE LEARNING

* + Model a profit/loss function depending on any given state, and try to maximise/minimise them respectively
* Model: optimise f(state, variables)
  + Reinforcement learning approach
  + Monte-Carlo, state-action-reward
  + Q-learning
  + Deep reinforcement



# DATA ACQUISITION/PREPARATION |

# Two Primary Datatypes

## Structured data

* Highly organised data
* Clearly defined variables
* Easy to mine and analyse
* e.g. numeric, factor/categorical
  + Numeric continuous variables, or factor/level/class variables (can also be and, i.e. numeric + factor. This is called mixed data)
  + Source: Spreadsheets (Excel, CSV), SQL databases, sensors and devices

Note: cannot assume some variable is numeric if their values look like numbers! E.g. if 0 and 1, it could refer to a Boolean (male/female for example)

* e.g. time series
  + Numeric with timestamps
  + Source: Stock/equity markets, weather data over time, prices and promotions
* e.g. network data
  + Nodes and connections
  + Source: Social networks and web, transport networks (MRT), financial transactions

## Unstructured data

* Highly unorganised
* Non-obvious variables
* Highly context-sensitive
* e.g. text
  + Words, phrases, emoticons
  + Source: social networks, text messages, books, wikis, documents
* e.g. image
  + Pixels and objects, how to identify?
  + Source: social networks, blogs, wikis, documents
* e.g. videos
  + Images, frames, objects in an even more unorganised manner (moving)
  + Source: Youtube, video calls/messages
* e.g. voice
  + Voice signals and waves
  + Source: songs, social media, recordings, announcements

CZ1115 Topic TWO:

Data Analytic Thinking

# Numeric Uni-Variate Data

## **Central Tendency:**

## Mean u

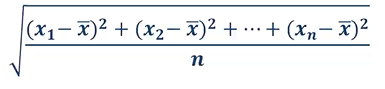
* Sum/count

## Median

* Marker that divides the data 50:50, i.e. P(x<median) = P(x>median) = 0.5

## **Spread:**

## Standard deviation Ꮾ

* Squared deviation of each data point from the mean, xbar
* Apply square root to go from variance to std: 

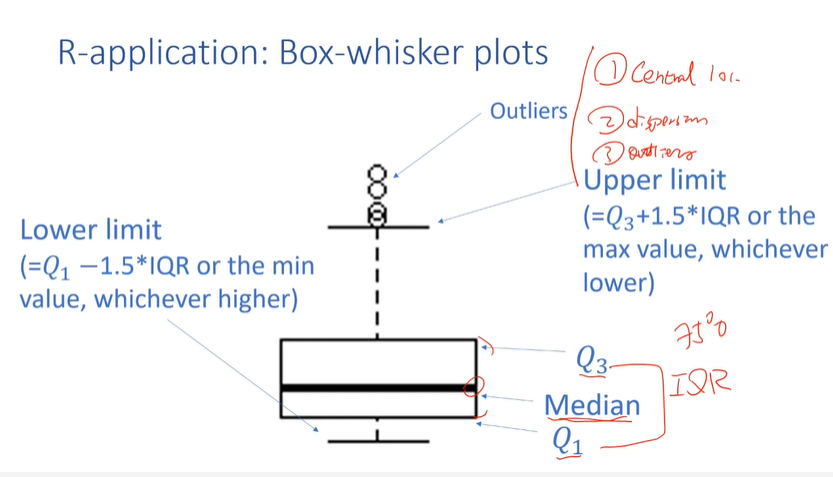
## Quartiles Q1/Q3

* Markers that divide the data 25:50:25. P(x<Q1) = 0.25, P(x>Q3) = 0.25, P(Q1<x<Q3) = 0.5

## Min/Max

# Visualisations

## Box Plot

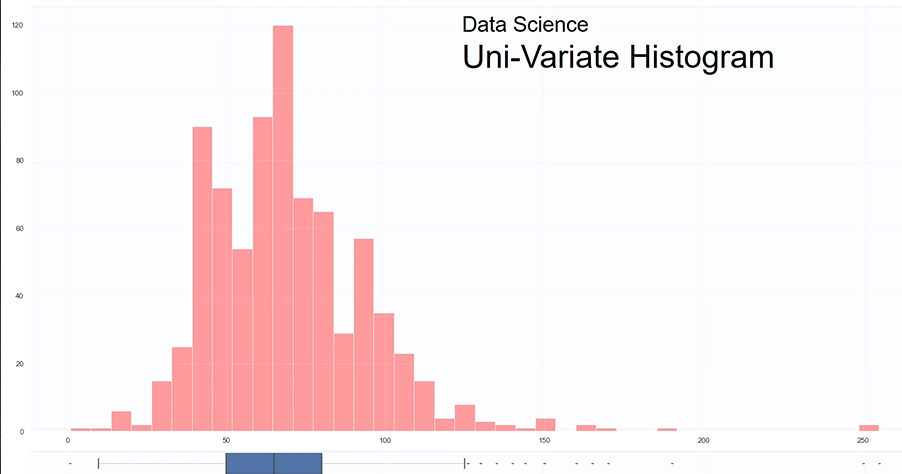


## Histogram (255 bins)

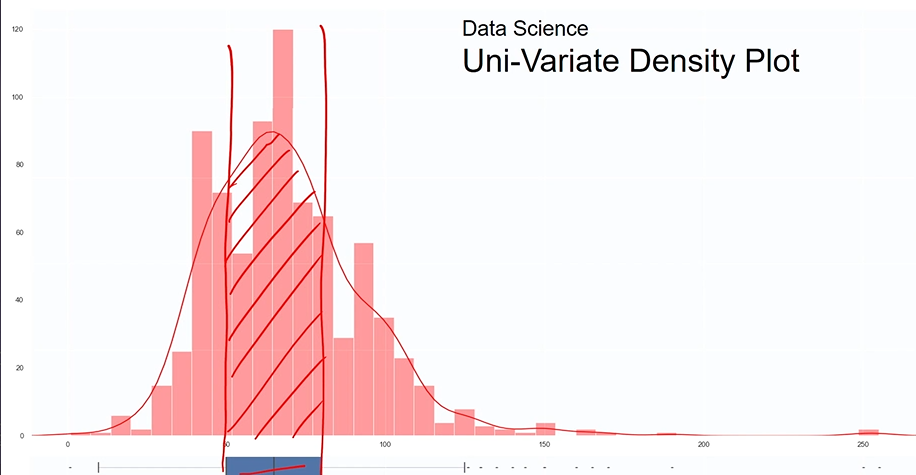
* Bins = categories/chunks



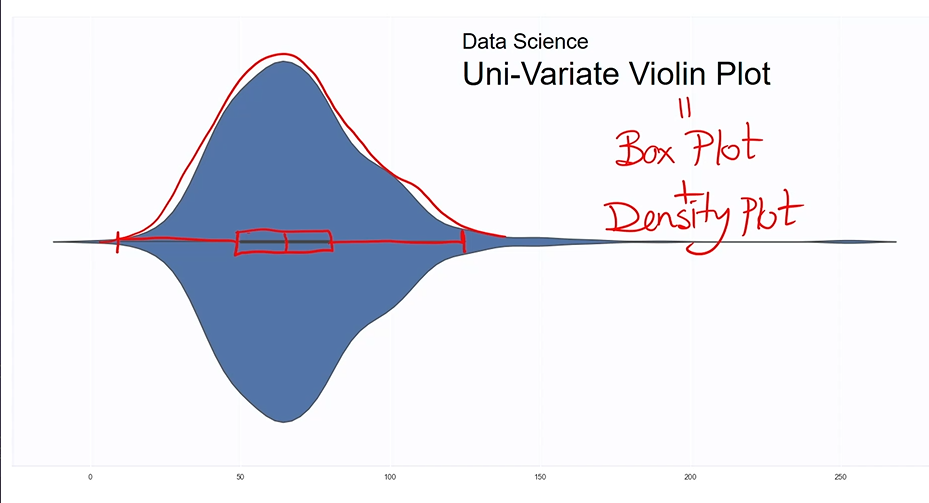
## Histogram (w/less bins)



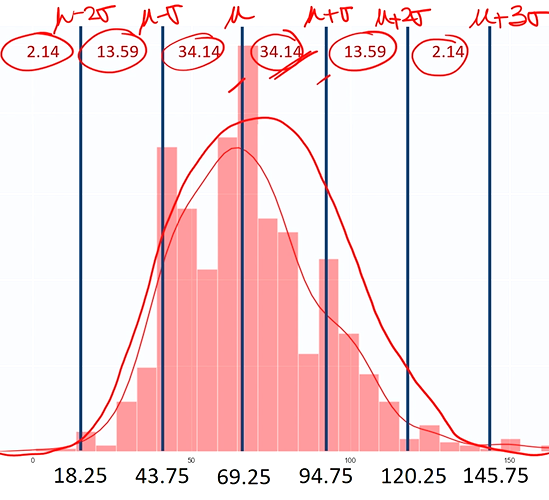
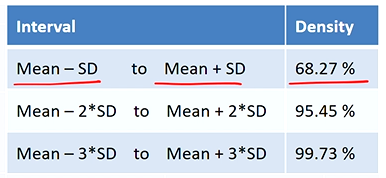
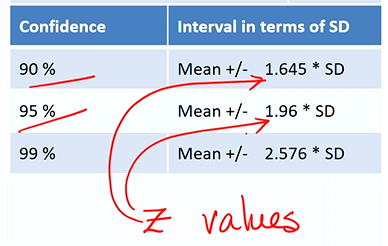
## KDE / Density Plot (the continuous line)



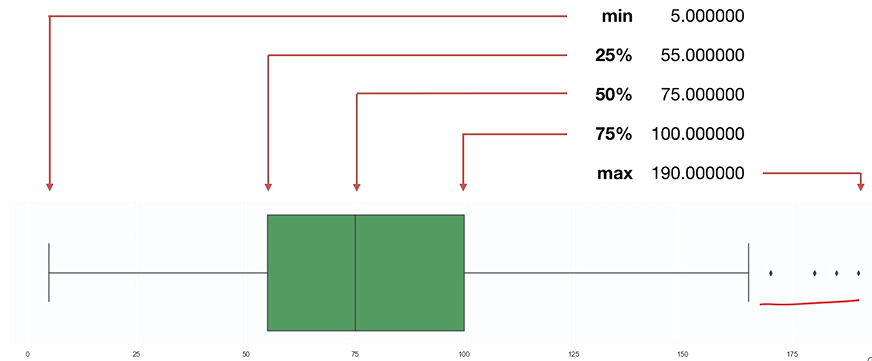
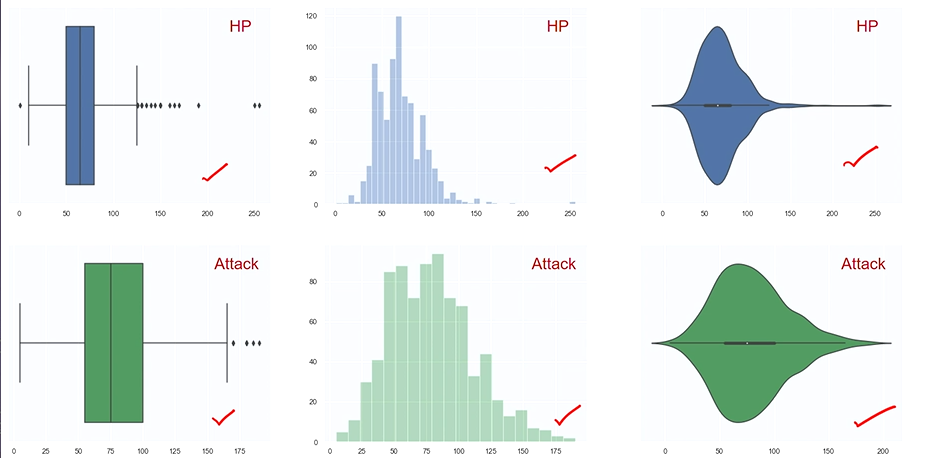
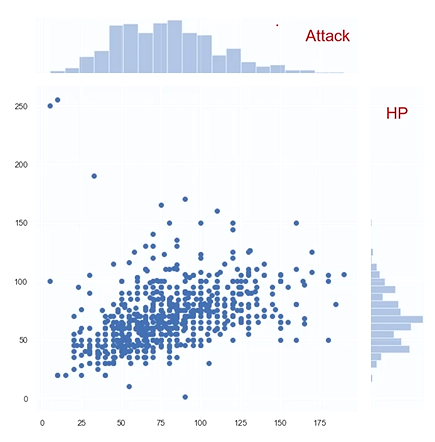
## Violin Plot (histogram + density plot)



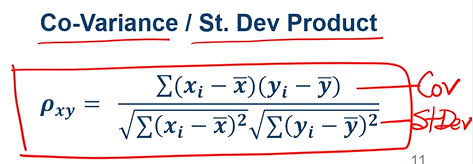
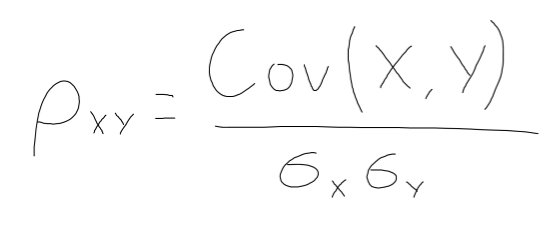
# Normal / Gaussian / Bell Distribution

* In a perfect distribution, mean == median. In the boxplot this is visualised by the centre line (median line) dividing the box perfectly.
* In a good distribution, almost all the points except outliers are within the mean (u) +- 3 standard deviations (Ꮾ), in the following percentages:
  + 
  + 
  + 
  + Z values: sd in a standard normal distribution Z~N(0,1)

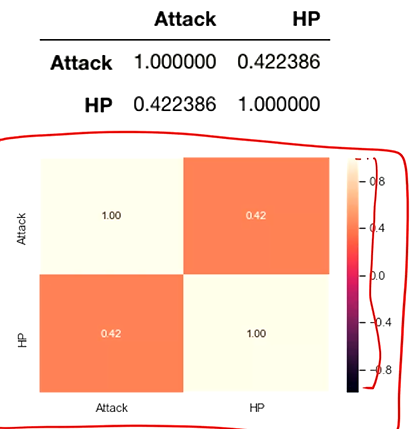
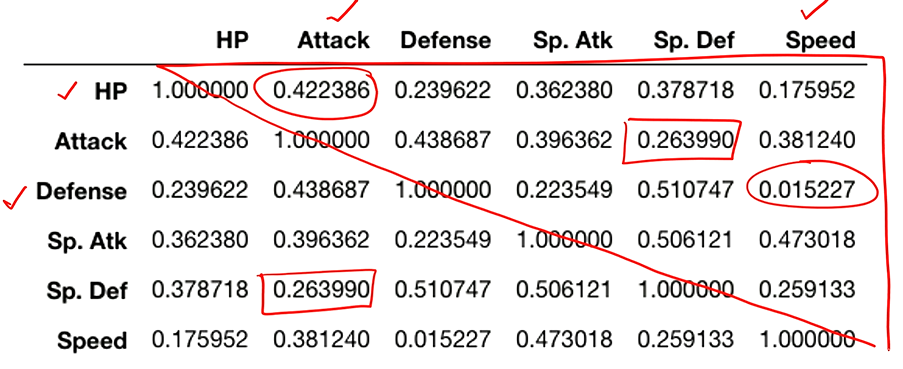
# Bi-/Multi-Variate Analysis

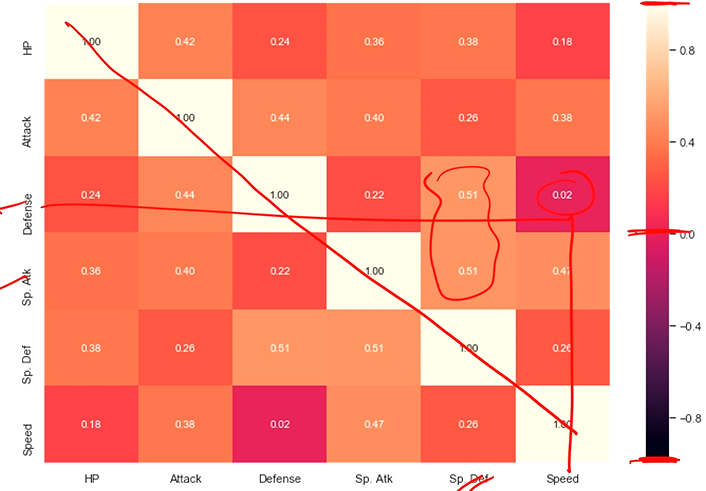
1. Obtain statistical summary of the individual variables
   1. Spread: sd, quartiles, min/max
   2. Central tendency: mean, median
   3. +count
2. Plot the statistical summary in a box plot, and in other plots:
   1. 
   2. 
   3. Insights on spread and central tendency, e.g. HP is less than attack, HP has smaller spread than attack.
3. Ask the statistical questions:
   1. Is there a mutual dependence?
   2. What is the mutual relationship?
4. They can be answered using a joint plot:
   1. 
   2. x-axis = attack, y-axis = HP. Coordinates (x,y) describe the points.
5. Pattern recognition: HP increases as attack increases on average. Dependence is moderately strong

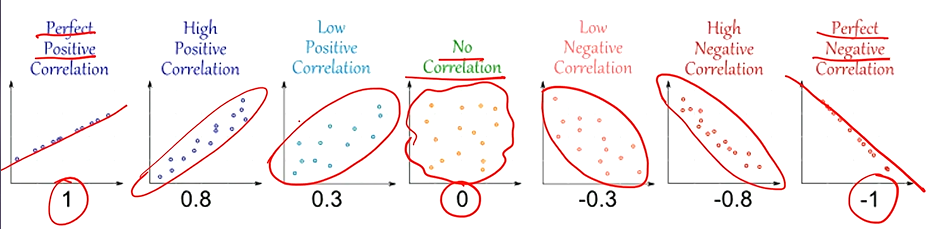
### Pearson’s Correlation coefficient

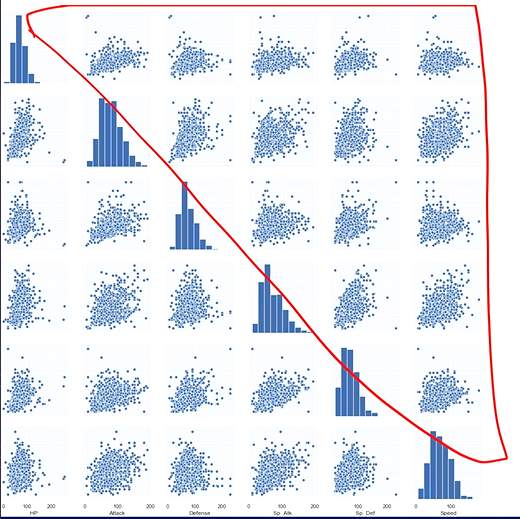
* 1. Dependence of two variables
  2. 
  3. 
  4. Takes on value between –1 and 1. 0 = no dependence. +1 = perfect +ve correlation.

### Correlation matrix and plot

* 1. 
  2. 



* 1. Closer the magnitude of correlation coefficient to 1, the tighter the spread of dots on the joint plot, and the more you can confidently infer a linear relationship.
  2. 

1. Multi-variate distributions also can do multi-variate pair-plots
   1. 

Note: correlation is not causation. Can only predict outcome with variables, not say that the variables caused a certain outcome

CZ1115 Topic THREE:

MACHINE LEARNING

Are variables mutually dependent? How to find relation between them? How to predict one using another?

## How to optimally learn from the data?

# Supervised learning

## Application:

### Prediction: Numeric (REGRESSION)

* + Find the relationship between the independent and dependent variables
  + Model: Total = f(variables)
    - Total is the response, variables are the predictors

1. Split the dataset into training and testing data.
   1. The train set should be representative of the main dataset and hence should be chosen carefully via uniform random selection from the main labelled dataset.
   2. Testing and training set should have no overlap.
2. Guess the initial values of the “Parameters” (a, gradient and b, y-intercept) for the hypothesised linear model
3. Predict the values of the response variable for all observations in the training data
4. Compute the errors in train data, compared to actual values of the response.
5. Choose a specific Cost Function (e.g. sum square of errors) for optimisation
6. Reassign or tune the “Parameters” of the model to optimise the cost function

# Hypothesise a Linear Model

example: Total = a x HP + b + ∈

**Response = a x Predictor + b**

∈ = error (uncertainty in prediction)

a,b = parameters

## Cost function

* Is a function of the parameters a and b, and we have to optimise (minimise) the bi-variate cost function J(a,b)

### Total Sum of Squares (TSS)

TSS = n x Var



* Comes from the benchmark
* MSE = variance for the benchmark that takes the mean

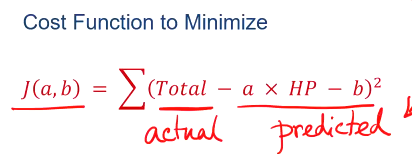
### Residual Sum of Squares (RSS)

RSS = n x MSE



* Response and predictor = available as fixed values from the training set and plugged into the cost function before optimisation process starts.
* a,b = parameters to estimate, i.e. the variables!!
* Is the new sum of squares for this new model. Should be compared to TSS (benchmark) to evaluate performance (this is done for R^2)

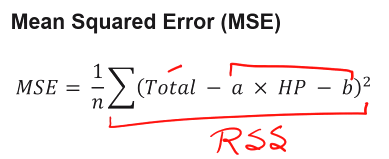
Example:



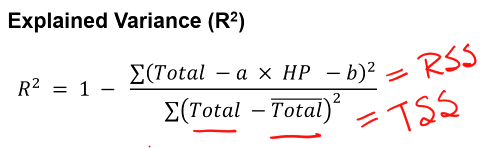
### Residual Absolute Sum of Errors



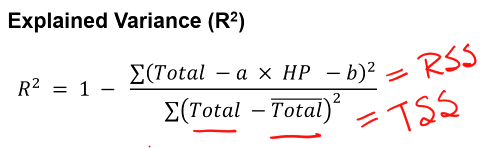
## Goodness of Fit



* Total, HP = test data
* a,b = model
* n = number of test data
* The lower the MSE the better.



R2 = 1 – RSS / TSS = 1 – MSE / VAR





* R^2 measures how better your final model is compared to the benchmark.
* 0 <= R^2 <= 1. The higher the R^2 the better. R^2 can be negative on test data, but this means you are doing worse than your benchmark (wrong direction)
* Higher R^2 --> points are clustered closer to the model’s line.
* Is equal the square of correlation.

Good fit on training data =/= good fit on test data. No guarentee that high R^2 means the model best predicts the test set.

# Unsupervised learning

1. Given a set of data
2. Find the optimal setting from the data
3. Justify interpretation of the groups

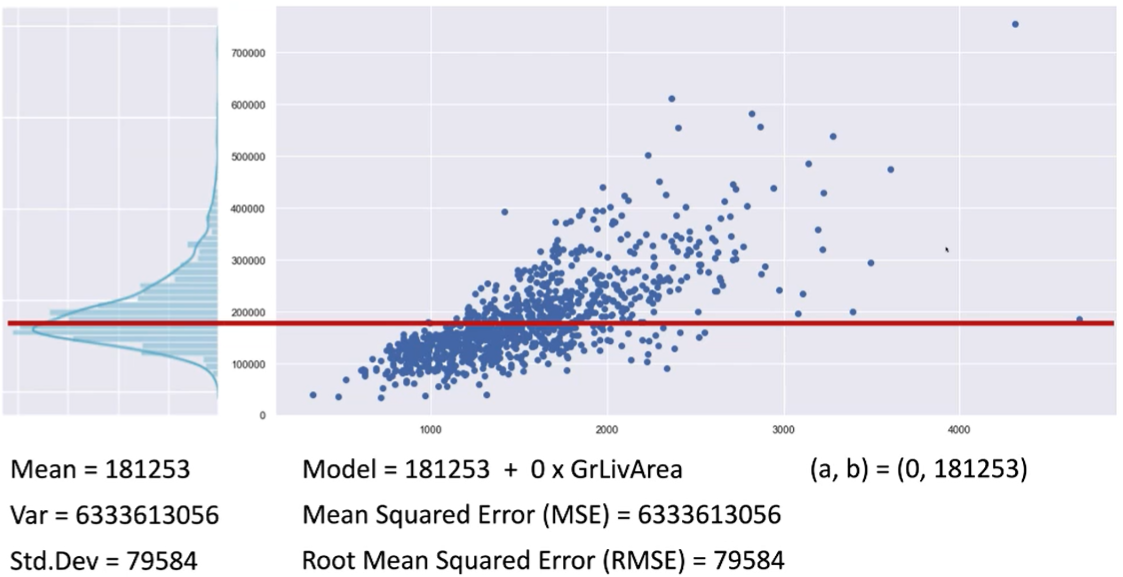
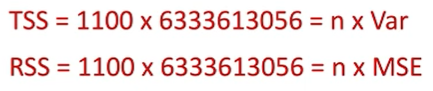
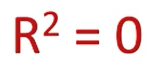
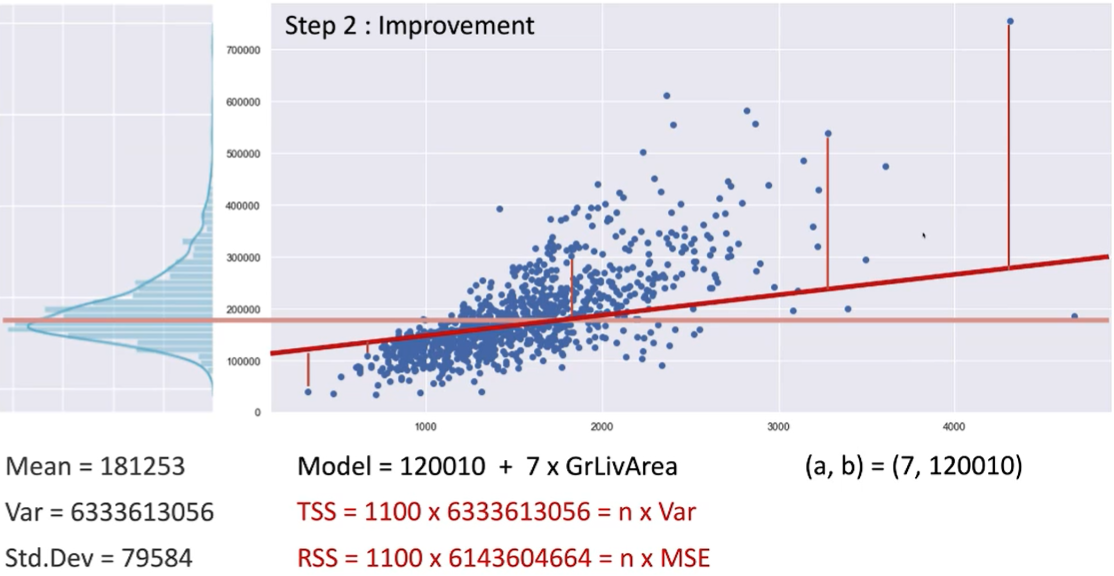
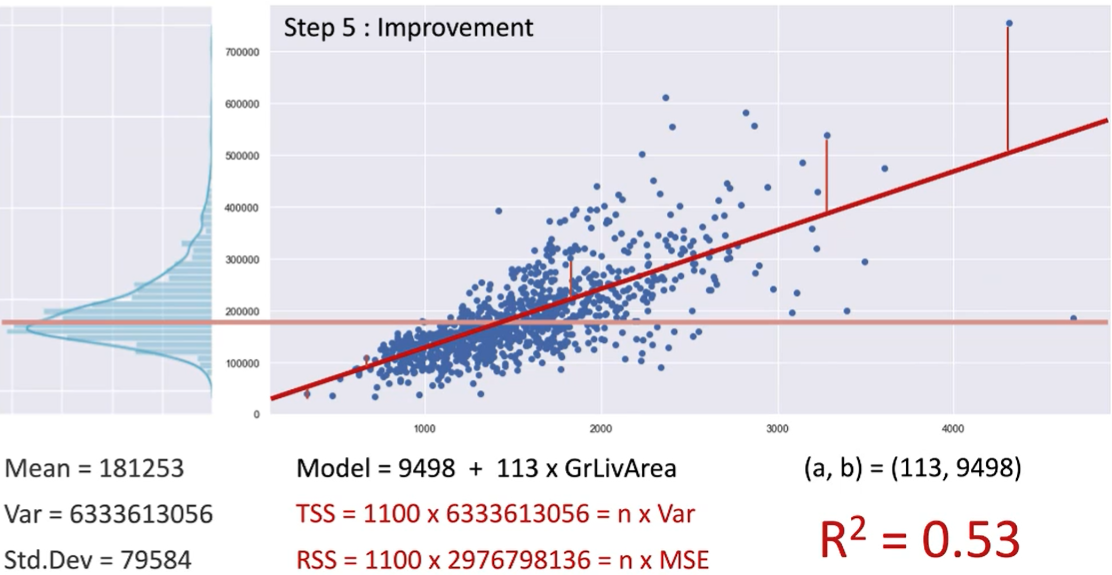
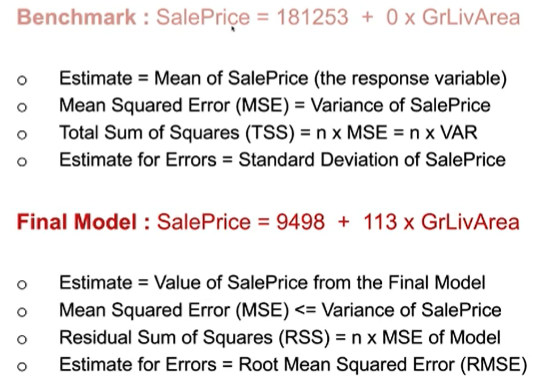
## Application:

### Detection: structure (CLUSTERING)

* + Find groups of data points that are close to each other but far apart from other groups
  + Close-far depends on “distance”
  + Model:
    - Distance: Euclidean, Jaccard etc
    - Nearest Neighbour
    - k-Means algorithm for clustering
    - Hierarchial model for clustering

### Detection: anomaly (Boolean) (ANOMALY DETECTION)

* + Find deviations of the data compared to the regular pattern observed through the data model
  + Model:
    - Cluster-analysis based detection
    - Nearest neighbour detection model
    - Support vector based detection

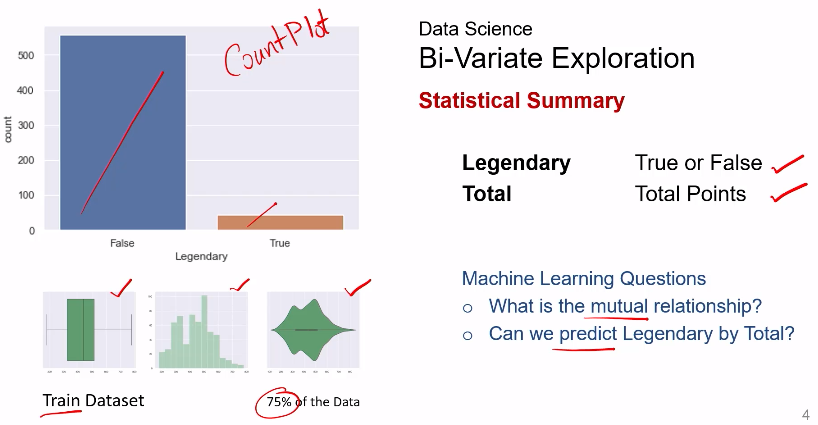
1. We get only sale price.
2. We set the benchmark as:
   1. Estimate = mean
   2. Mean Squared Error (MSE) = variance
   3. We do not want to perform any worse than this.
   4. 
   5. 
   6. 
3. Improvement
   1. 
   2. 
4. Repeat step 4 till you get the final model.
   1. Estimate = value based on the final model
   2. Mean Squared Error (MSE) <= variance (hoped, because we should do better than the benchmark)
   3. 
5. 
6. Now use the model on the test data.
   1. R^2 and MSE of test and train should be as close as possible

CZ1115 Topic FOUR:

BINARY CLASSIFICATION

## Binary classification: response variable that only has two classes.

E.g. does total points (predictor) tell us whether a pokemon is legendary or not (response)?



## Boxplots are distinctly different for true and false --> total points may be an important variable

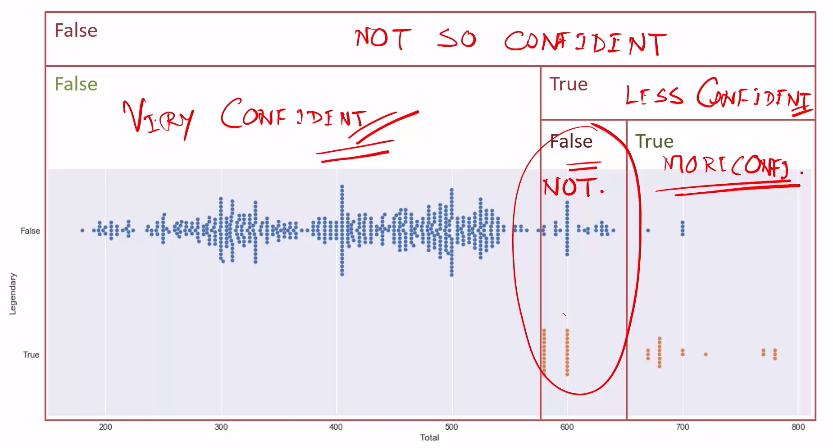
# Supervised learning

## Application:

### Prediction: class (CLASSIFICATION)

* + Find the probability of something being true
  + Model: P(event) = f(variables)

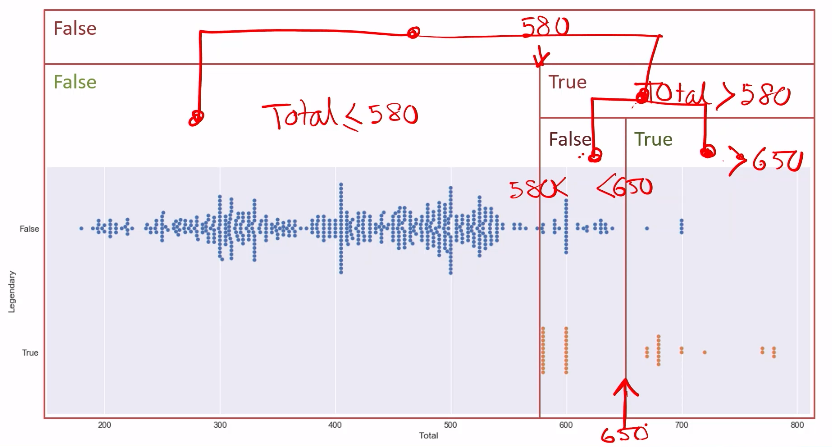
## Partitioning:



|

|

|/



# Decision Tree

## Paritions made in the data space methodically represented using consecutive binary decisions

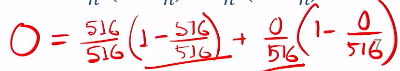
## value = [x, y]

## (Non-Legendary : Legendary)

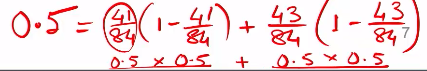
## Confidence of classification: Gini Index (tells you chance of misclassification)

## = 1 – (x/n)^2 – (y/n)^2 = 2xy/n^2

e.g. for leftmost orange node: Gini =



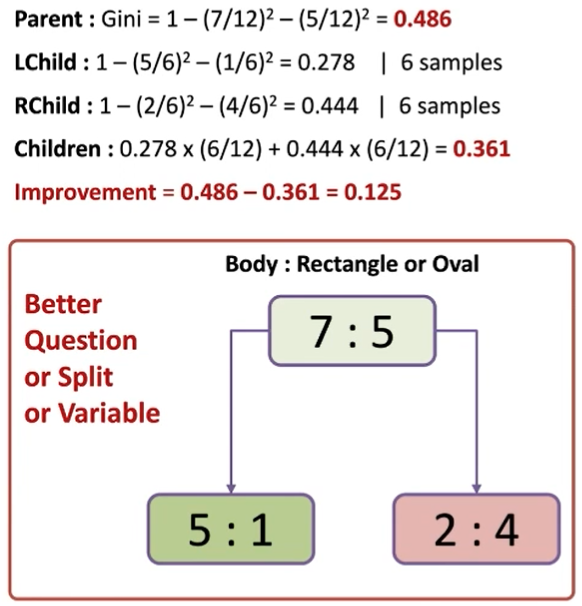
e.g. for white node: Gini =



* Max depth of decision tree is set beforehand – this is to prevent overfitting of the training data
* Lower the Gini, the better (less chance of misclassification).
* Only Ginis of the final nodes are concerning – white node has Gini of 0.5 but it doesn’t matter since there are child nodes under it
* 0.5 is the highest possible Gini, since anything above is the same as 1 – that number, e.g. Gini of 0.6 basically means a Gini of 0.4

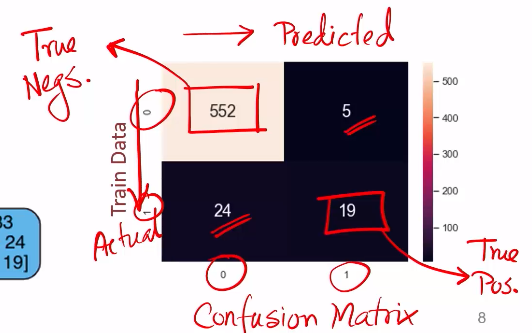
# Children Gini

## LChild’s Gini \* (proportion in LChild) + RChild’s Gini \* (proportion in RChild)



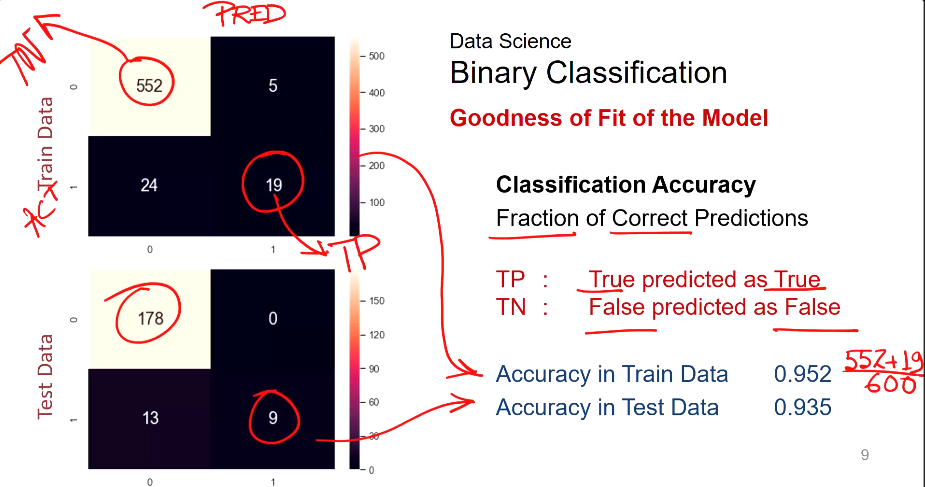
# Decision Tree – On the test data

## Confusion Matrix



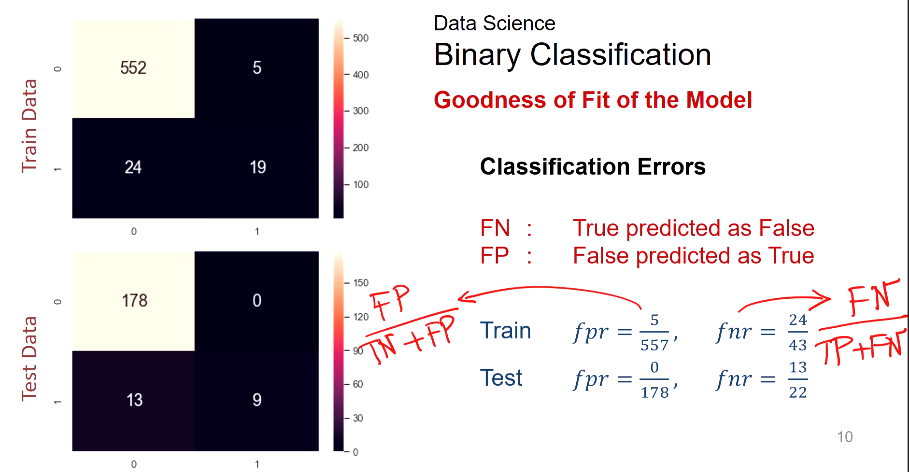
## Goodness of Fit - Do on both test and train dataset

|  |  |
| --- | --- |
| **TN** | **FP** |
| **FN** | **TP** |



## **Classification Accuracy**: fraction of correct predictions

## **= (TP+TN) / Total amount**



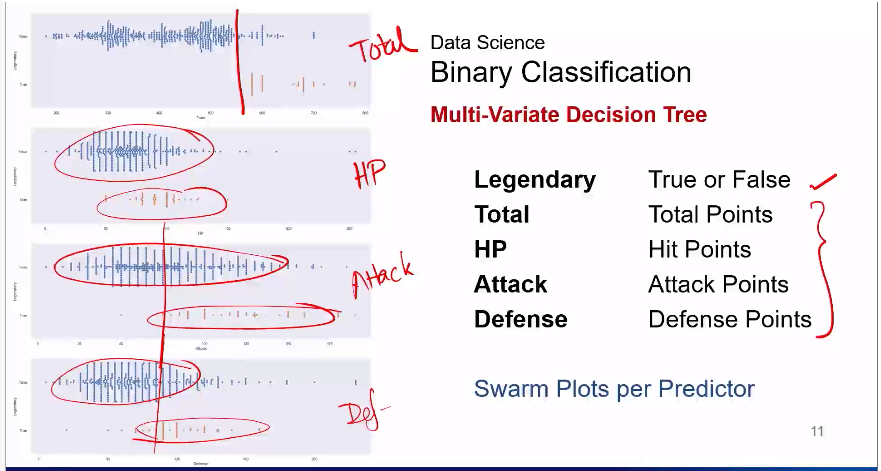
## **Classification Errors:**

## fpr = FP / total number of actual negatives

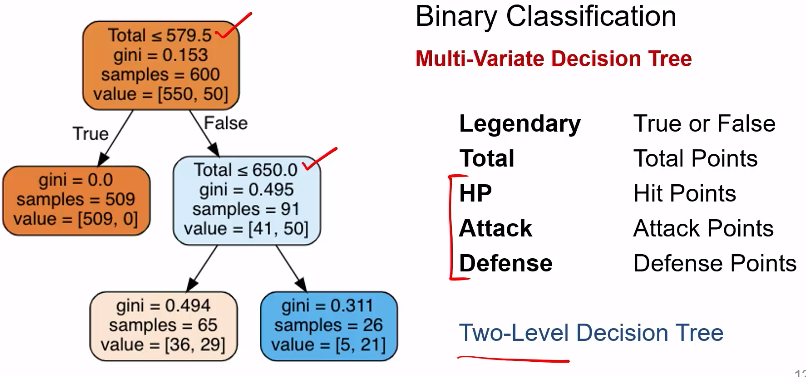
## fnr = FN / total number of actual positives

How to reduce classification errors? Normally, can only reduce one. So which one to minus depends on the purpose of your experiment, and hence whether false positives or false negatives are more problematic.

# Multi-Variate Decision Tree

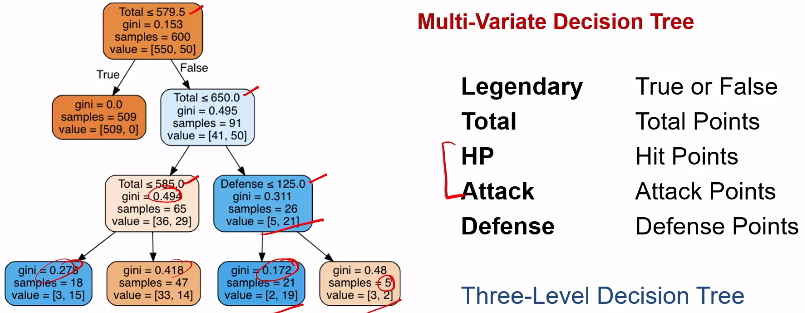


## For a two-level decision tree, the following is generated:

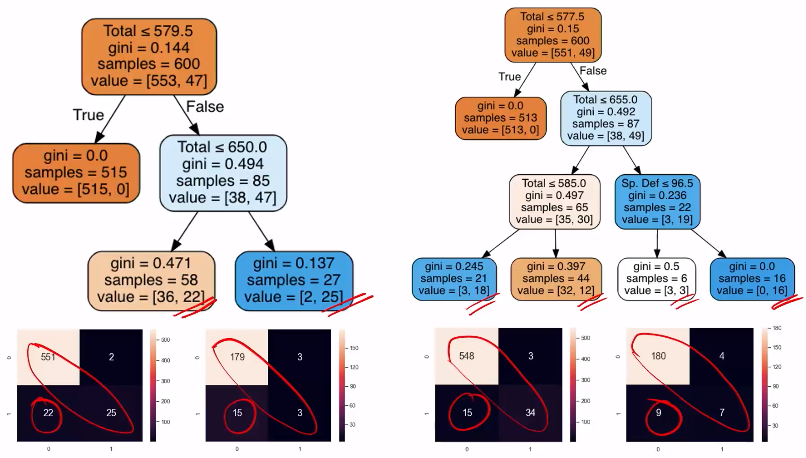


## Only total was used. Conclusion: total is the most important variable.

## For a three-level decision tree:

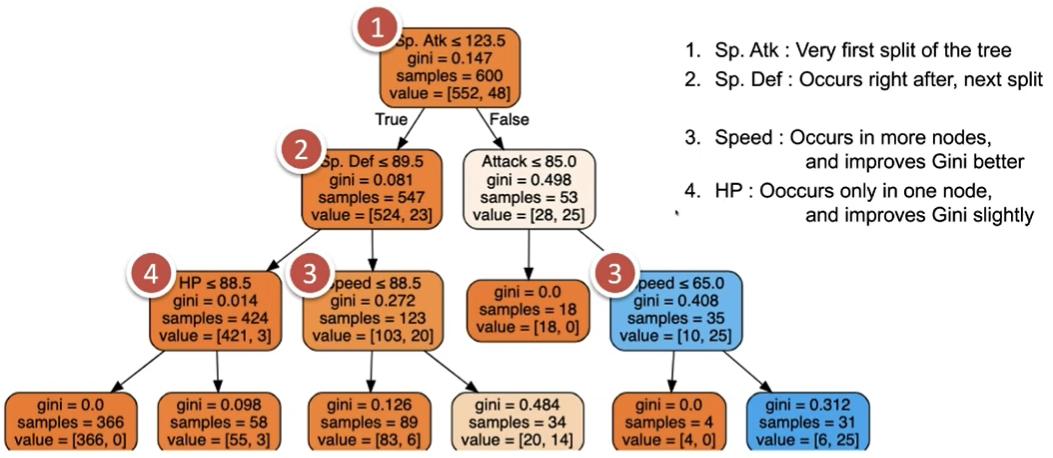


## From a two-level to three-level tree: goodness of fit increases.



## Importance of a variable in trees

* Is more if it appears at the top of the tree, that is, if it affects the initial decisions.
* Is also more if it appears as a deciding variable in a lot of nodes in the tree.



CZ1115 Topic FIVE

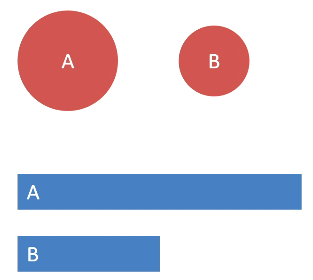
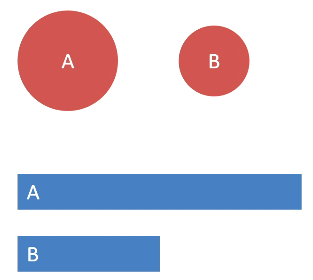
VISUALISATION

# Convey your story

* “Story” hidden in data; using visuals as information; telling “story” effectively.

## Effectiveness

* More effective if info is more readily perceived.

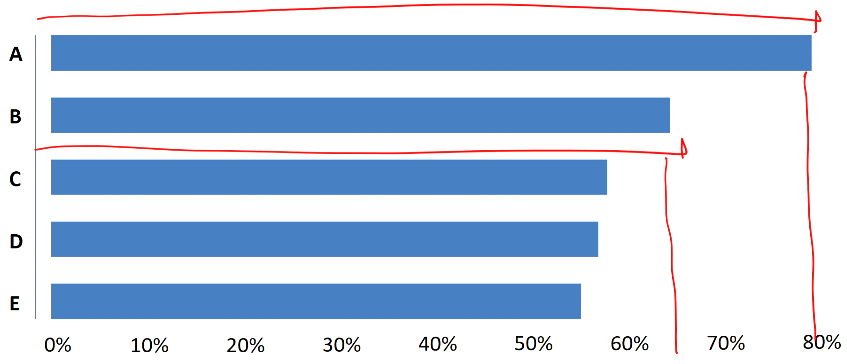
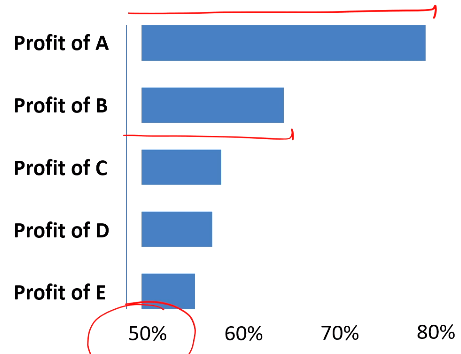


no; yes

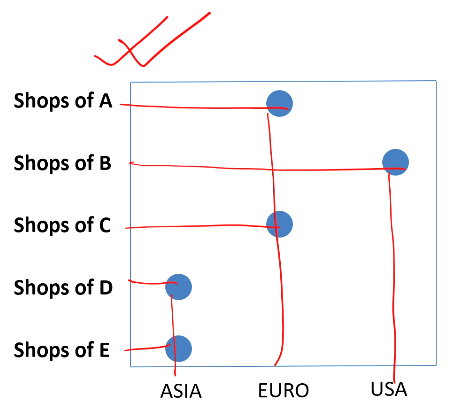
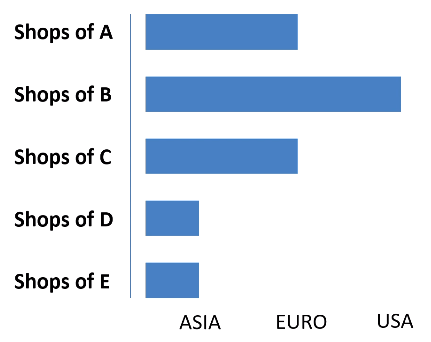
# Establish credibility

## Expressible

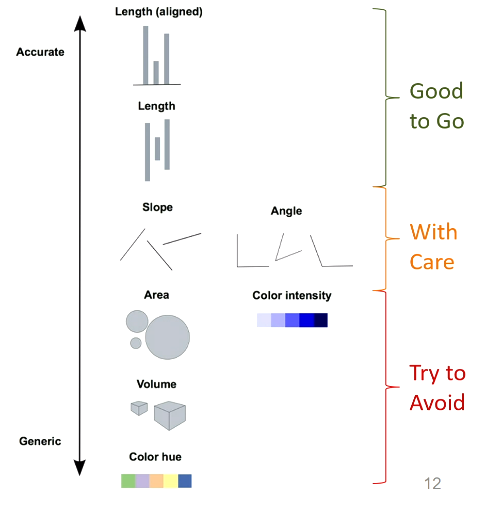
* Express all the facts and only the facts



no; yes

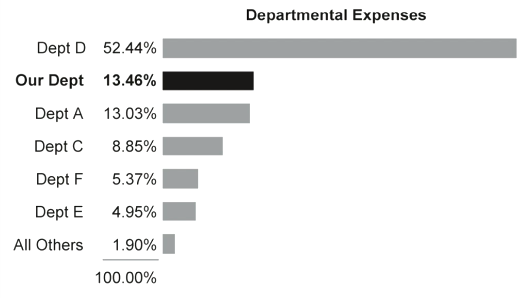
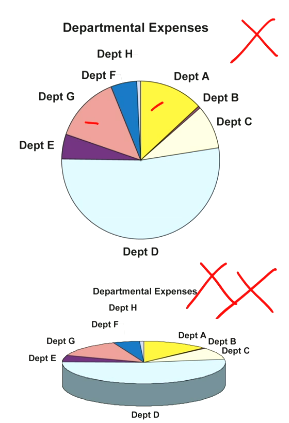


no; yes



# Show me the Numbers

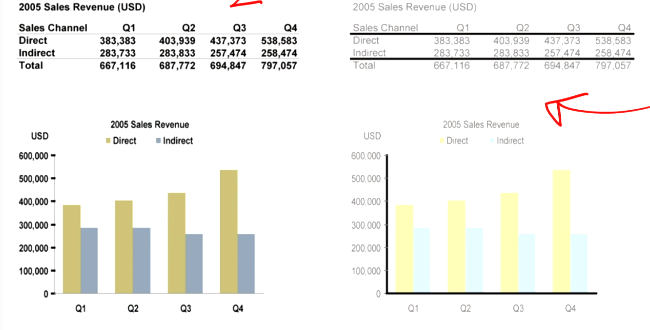
## Pie Chart Bad. 3D Bad. Stacked bar chart > treemap > pie chart



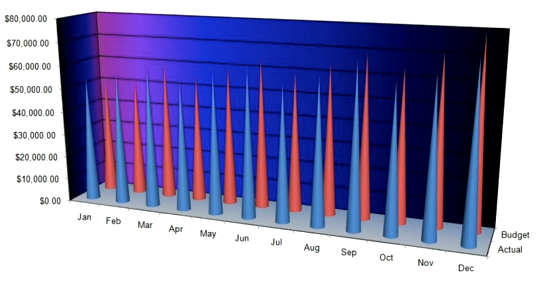
no; yes

## Data Ink vs Non-data Ink

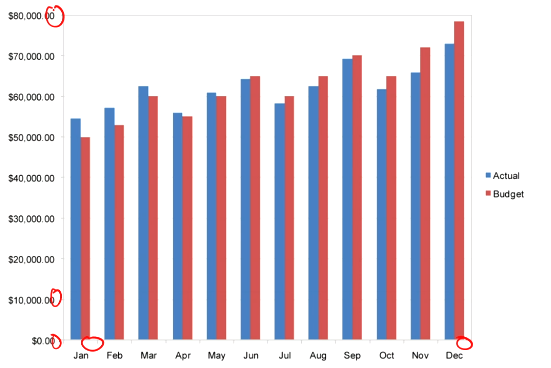
Data-ink ratio must be as high as possible.



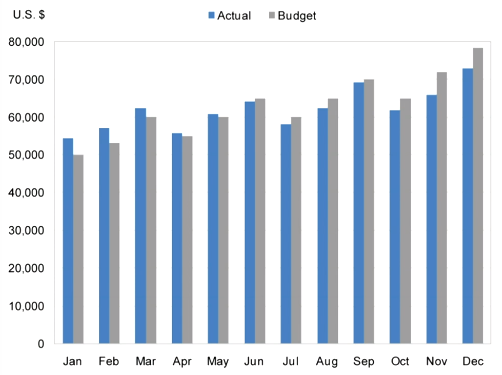
yes; no



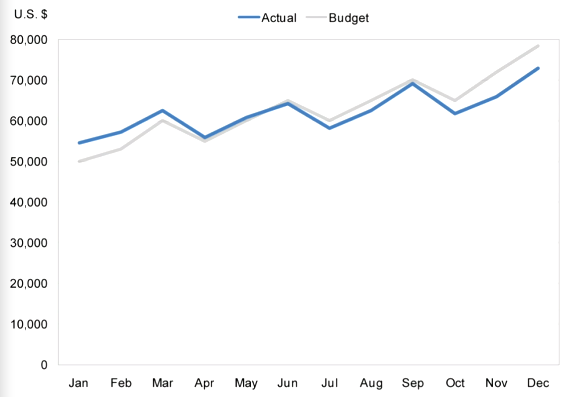
no 3D, no fancy pointy shape



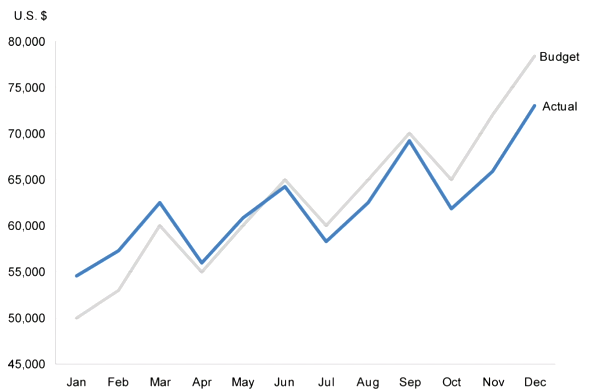
no appendages, no decimal points, show axis title and legend at the top, change colours to primary and secondary (don’t use high contrast)



more concerned about trend over months. use time series

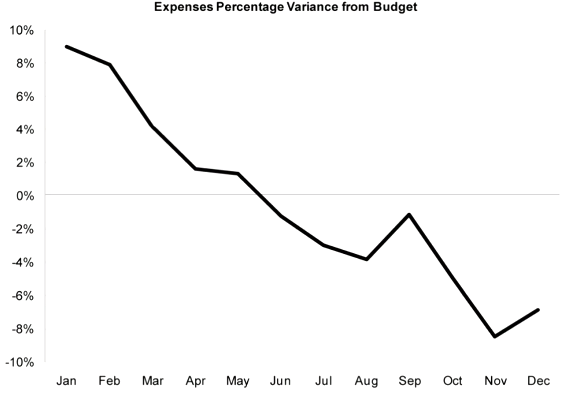


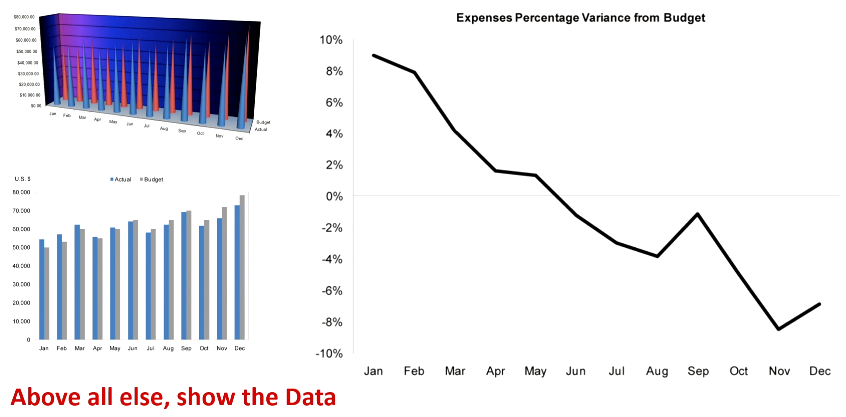
more interested in trend. truncate axis. also move legend down to the line



## Note: For bar graph, comparison is important, so cannot truncate axis. For line graph, more interested in trend, so can truncate axis.

purpose is to compare budget vs actual, percentage wise. change whole graph.

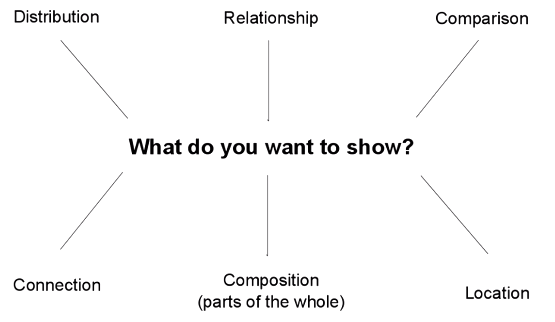




# Show what?

<https://www.data-to-viz.com/>

<https://raw.githubusercontent.com/ft-interactive/chart-doctor/master/visual-vocabulary/poster.png>



Distribution: uni-variate. Histogram.

Relationship: multi-variate. Joint plot. Regression line.

Comparison: categorical variable. Boxplot/violinplot.

Connection: classification. Decision tree.

Composition: bar chart.   
Location: map data.