08.08.2023

Statistical Methods in AI (CS7.403)

Lecture-3: Data Visualization, Intro to Performance Measures, Benchmarking

Ravi Kiran (ravi.kiran@iiit.ac.in)

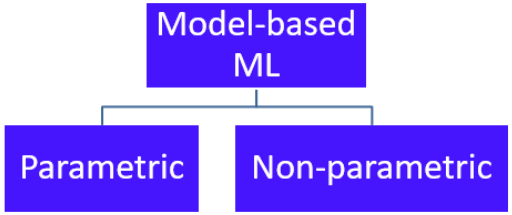
https://ravika.github.io

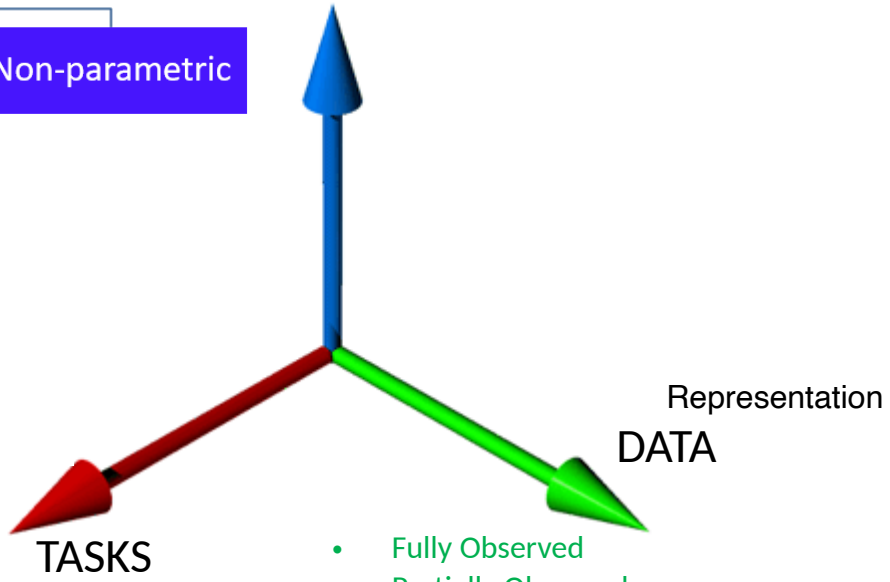
@vikataravi

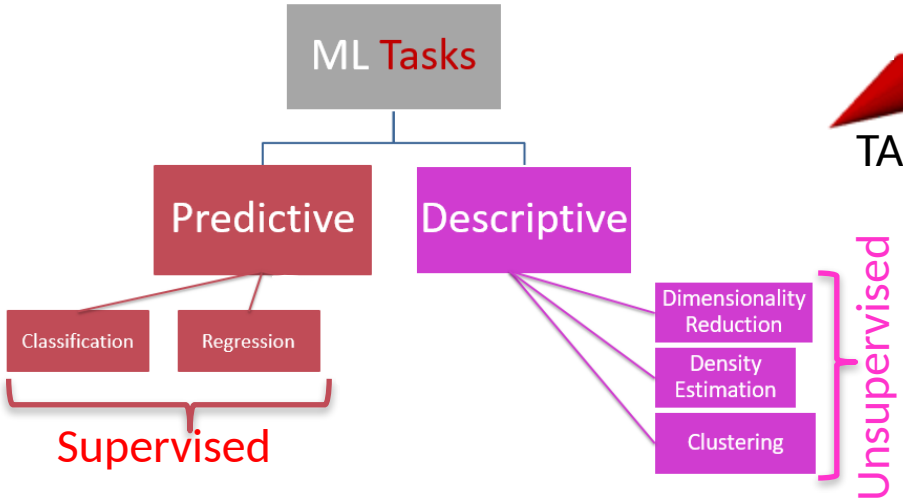
Center for Visual Information Technology (CVIT)

IIIT Hyderabad

Recap

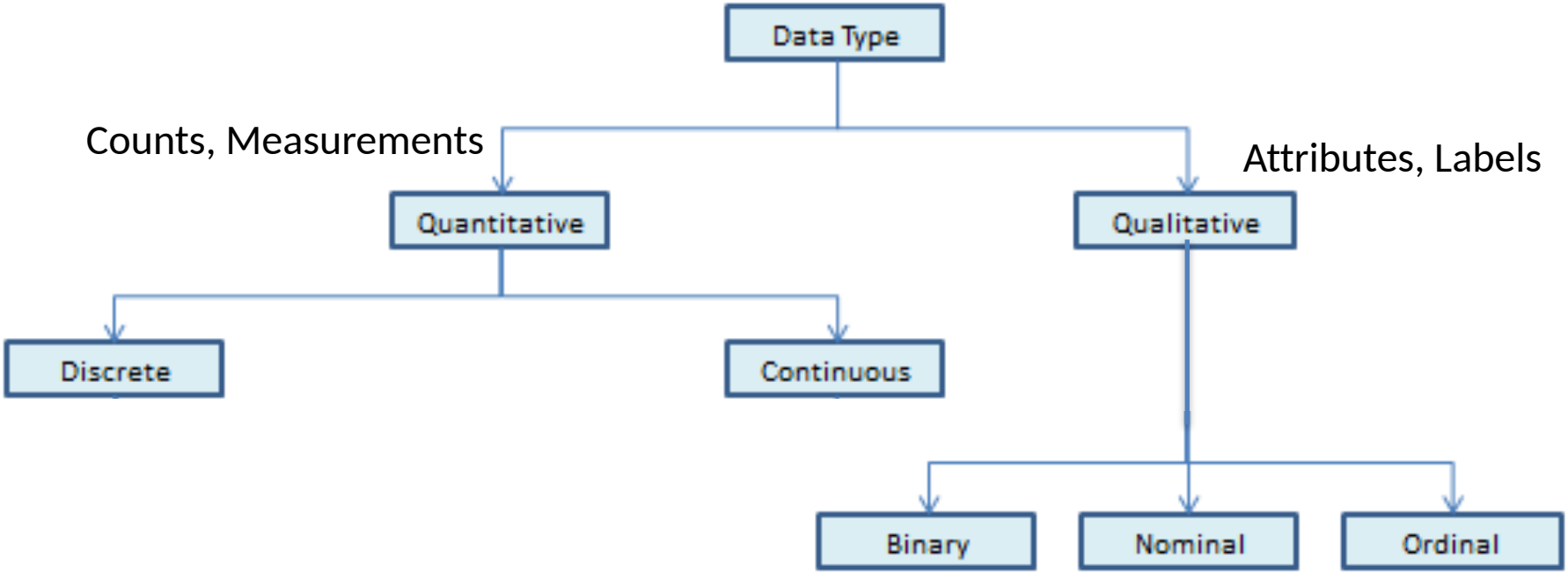




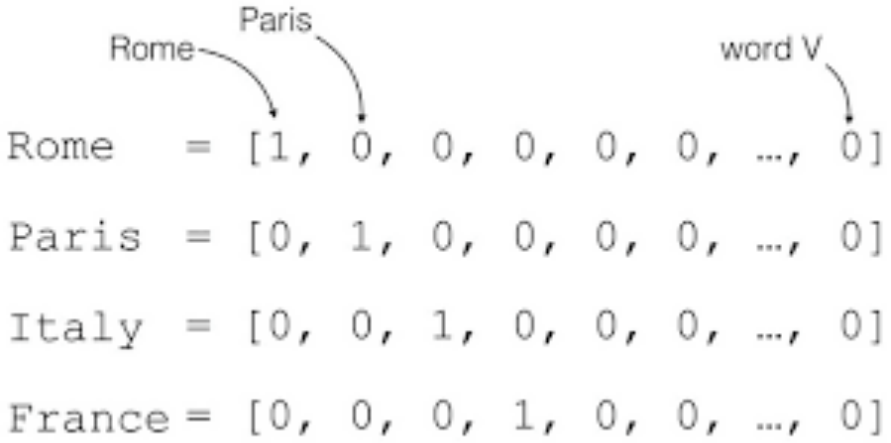


- Fully Observed
- Partially Observed
  - Some variables systematically not observed (e.g. 'topic' of a document)
  - Some variables missing some of the time (e.g. 'faulty sensor' readings)

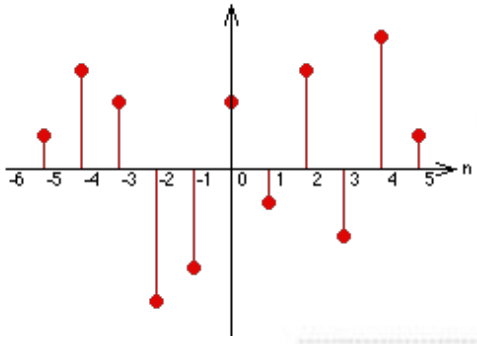
Taxonomy of data



Numerical encoding of categorical variables



Data Sample Representations

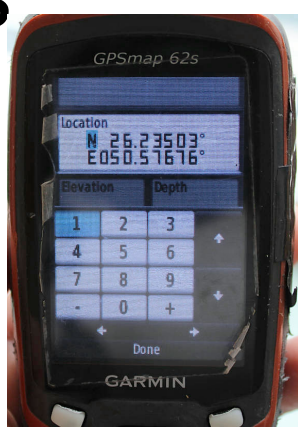


Scalars

$$x$$

Vectors

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$



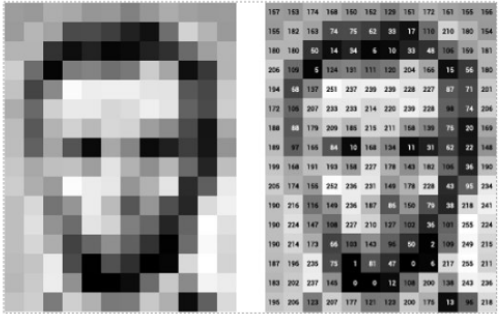
Matrix

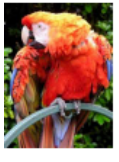
$$\mathbf{X} = \begin{bmatrix} \mathbf{x} & \cdots & \mathbf{x}_N \end{bmatrix} = \begin{bmatrix} x_{1,1} & & x_{N,1} \\ \vdots & \cdots & \vdots \\ x_{1,M} & & x_{N,M} \end{bmatrix}$$

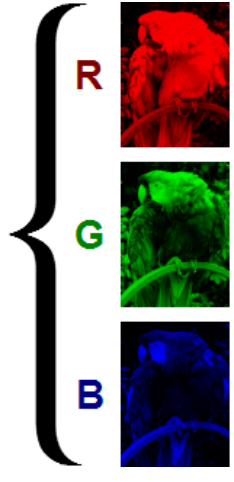
Tensor

$$\mathbf{X} = \{\mathbf{X}_1, \ldots \mathbf{X}_K\} = \begin{bmatrix} x_{1,1,1} & & x_{N,1,1} \\ \vdots & \cdots & \vdots \\ x_{1,M,1} & & x_{N,M,1} \end{bmatrix} \begin{bmatrix} \ddots & & \\ & & \\ & & \end{bmatrix} \begin{bmatrix} x_{1,1,K} & & x_{N,1,K} \\ \vdots & \cdots & \vdots \\ x_{1,M,K} & & x_{N,M,K} \end{bmatrix}$$

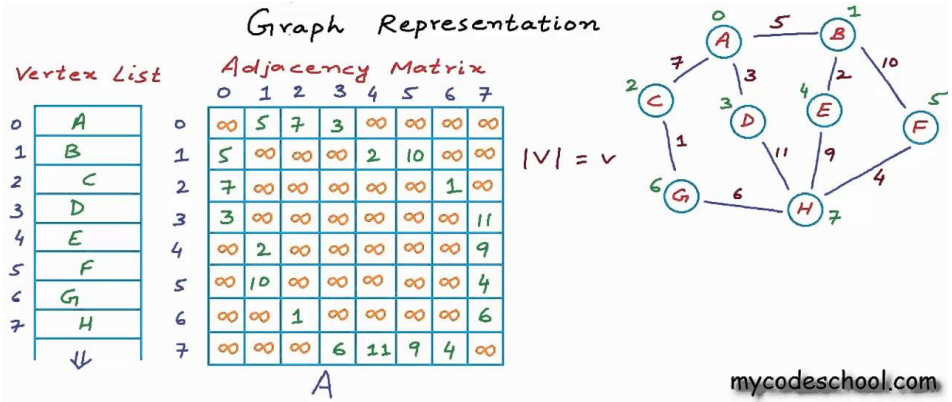
2-d image





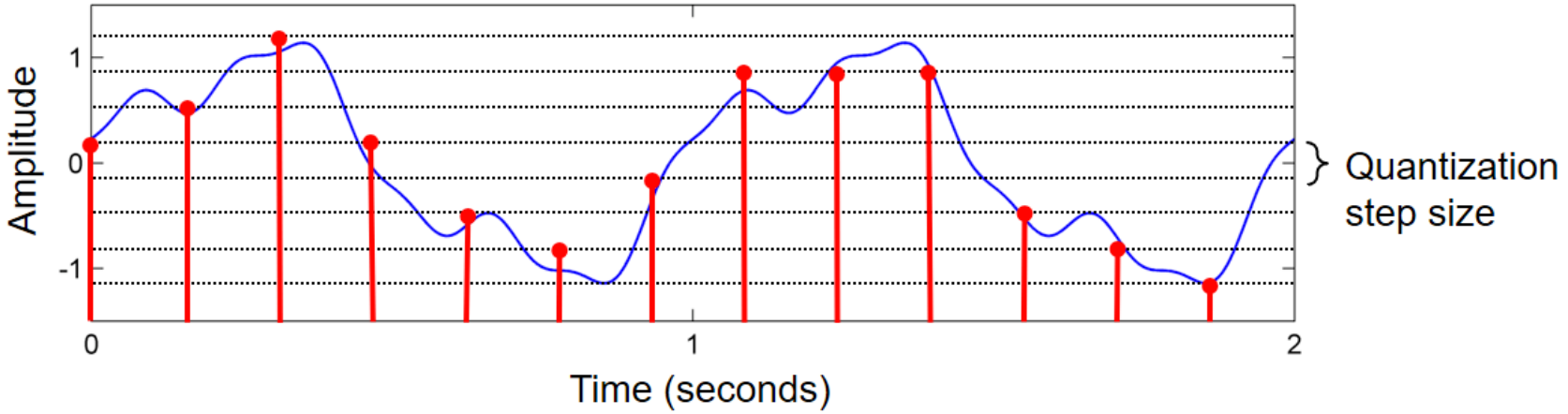


Data Sample Representations



Quantization

1. Continuous → Discrete (‘Rounding off’)



2. Binary Quantization (‘Thresholding’)

Data Normalization (applied to each feature)

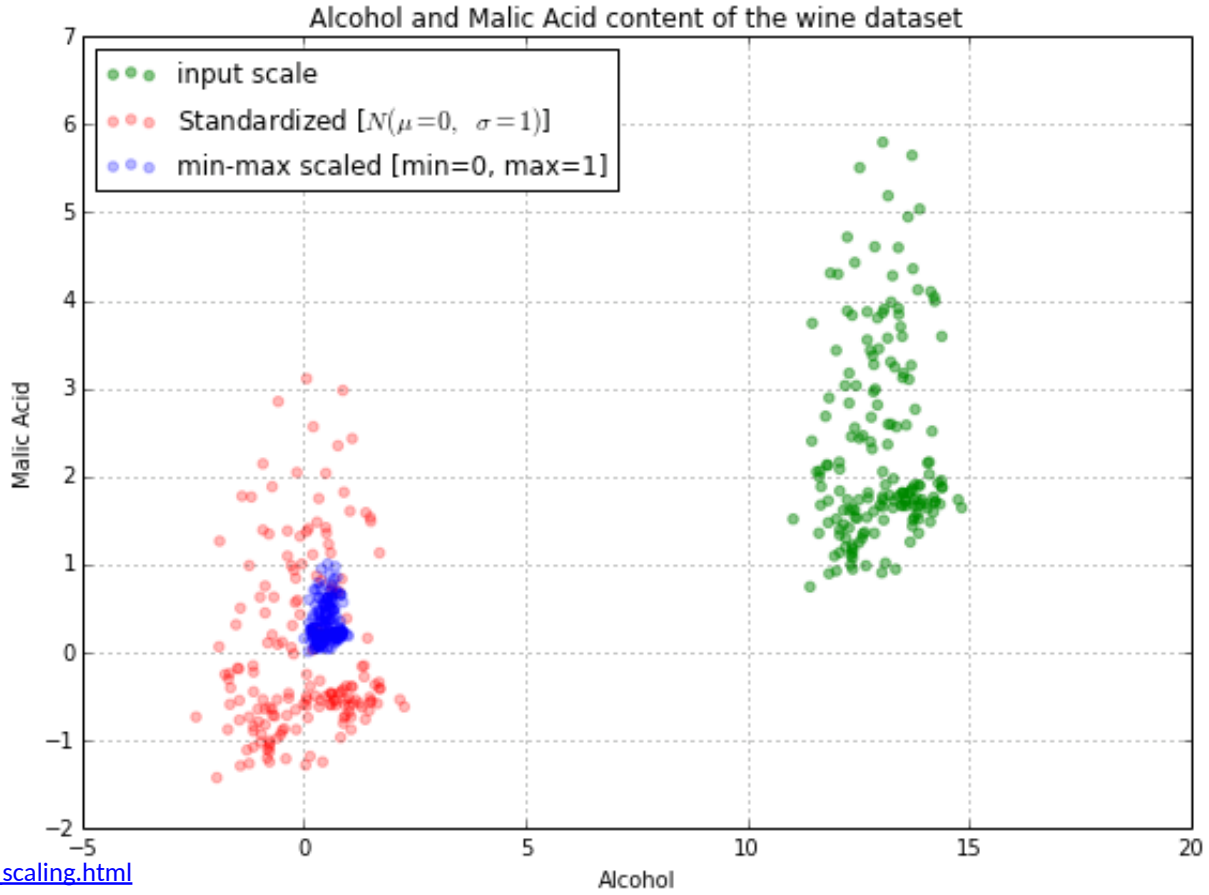
	Class label	Alcohol	Malic acid
0	1	14.23	1.71
1	1	13.20	1.78
2	1	13.16	2.36
3	1	14.37	1.95
4	1	13.24	2.59

- $$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

  MinMax Scaling

- Standardization
  (Unit Normal Scaling)

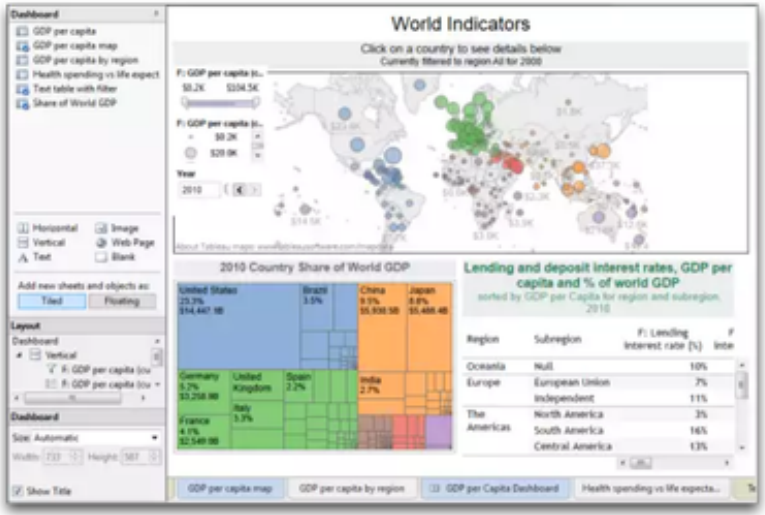
  $$z = \frac{x - \mu}{\sigma}$$



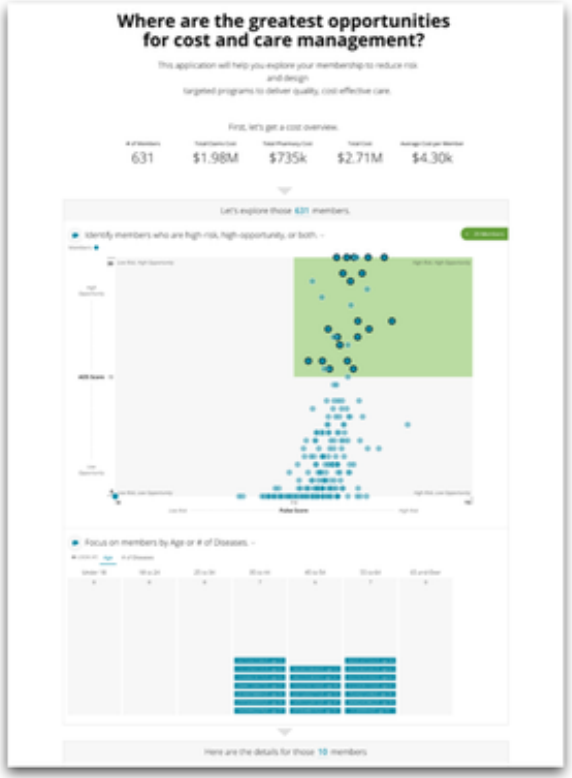
https://sebastianraschka.com/Articles/2014_about_feature_scaling.html

Gazing at Data: Data visualization

data exploration

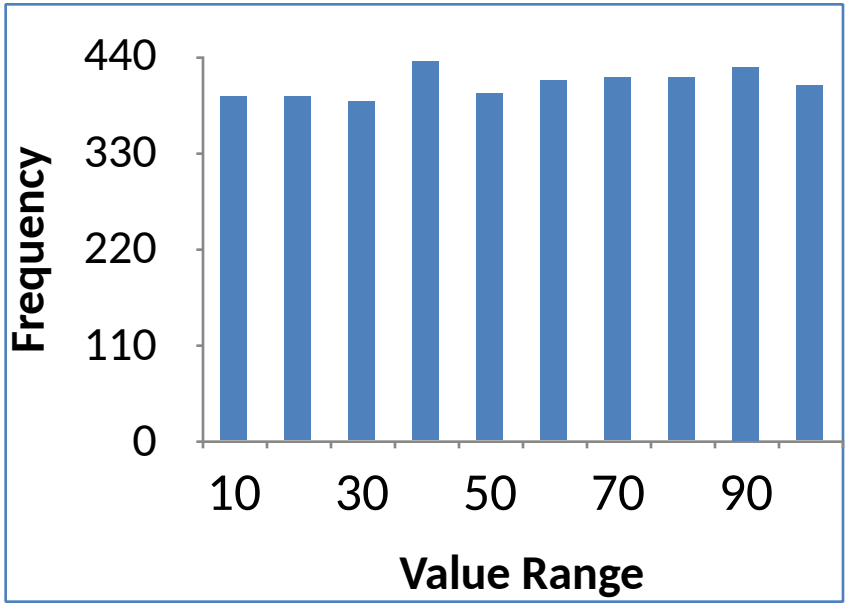


data presentation

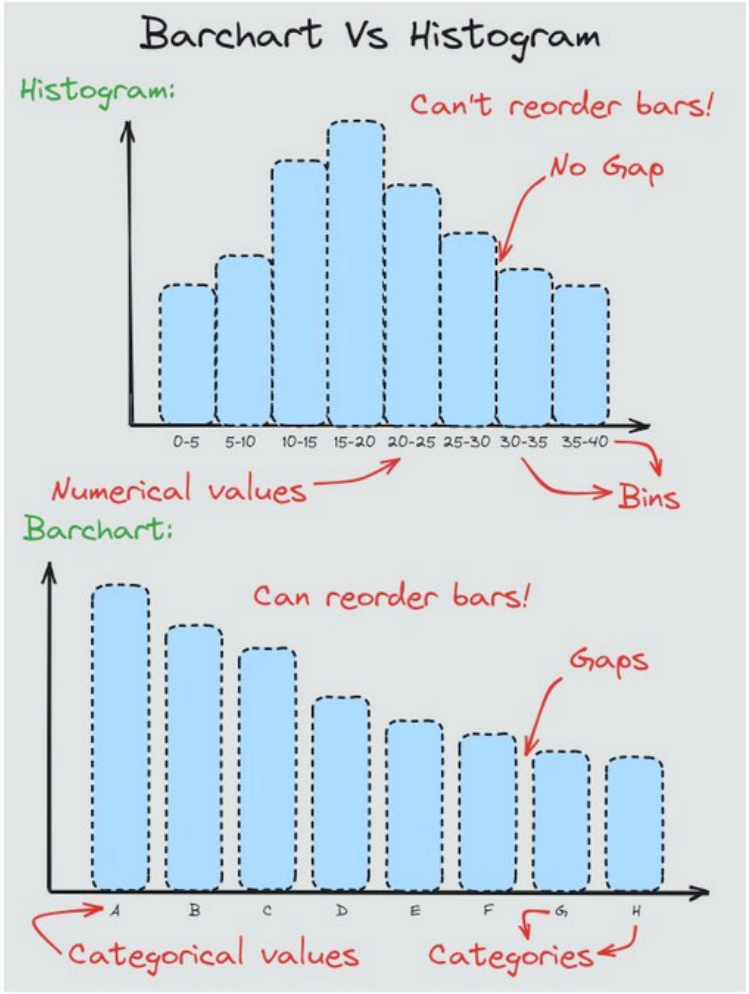


Why Plot Data?

22.65	42.12	67.24	59.13	81.49
23.03	53.42	40.54	89.97	21.85
12.07	93.43	51.93	49.30	43.76
47.68	51.91	13.12	73.88	60.29
86.20	41.28	66.24	62.15	46.87
20.02	92.09	26.50	83.53	70.99
48.38	46.21	10.85	29.61	62.15
55.23	84.90	15.37	35.00	83.23
65.30	26.56	5.78	72.59	12.47
75.71	93.15	3.67	49.80	43.05
69.73	53.77	82.80	43.59	32.35
77.95	14.94	63.71	9.30	1.31
58.90	42.53	62.74	99.91	53.17
6.45	46.29	67.34	32.65	23.94
32.39	57.39	10.61	54.07	53.28
74.35	60.10	2.25	77.55	12.05
82.87	17.02	80.73	29.60	9.96
47.05	97.01	19.84	76.59	45.90
34.26	86.80	19.11	4.80	1.24
12.54	30.40	67.94	55.53	58.25
			73.13	0.23



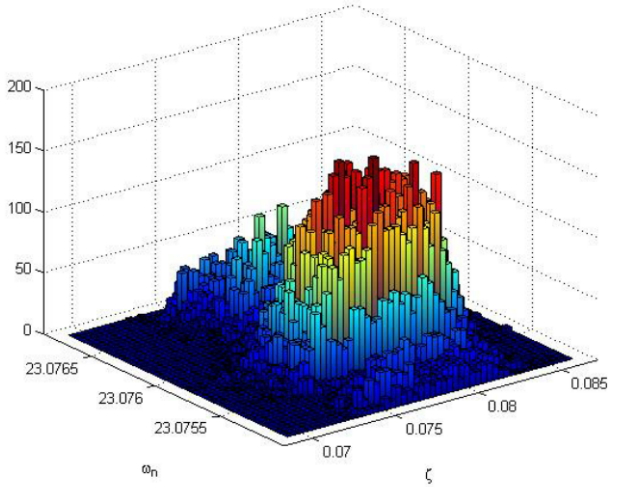
- Visualization of data provides specific insights into the nature of the data.
- Depending on the plot, we gain different insights

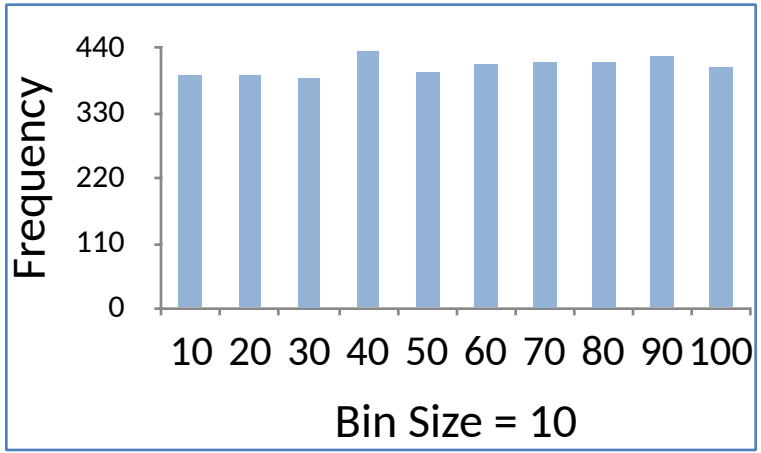


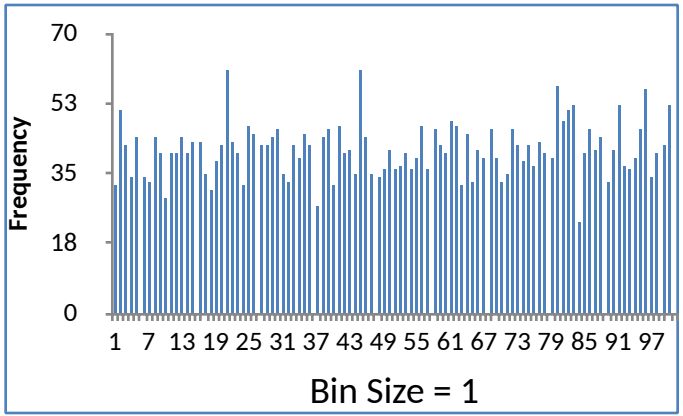
https://pbs.twimg.com/media/F1ej_z8X0AEiTvf?format=jpg&name=small

Histogram

- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution
- Visualize one feature at a time
- Possible to extend to two
- Dependency on bins ($\sqrt{n}$; $2\sqrt[3]{n}$)

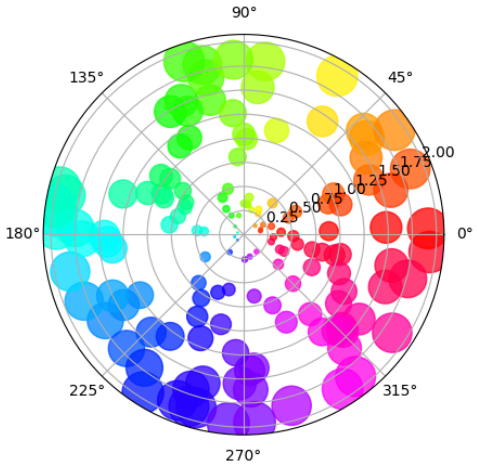


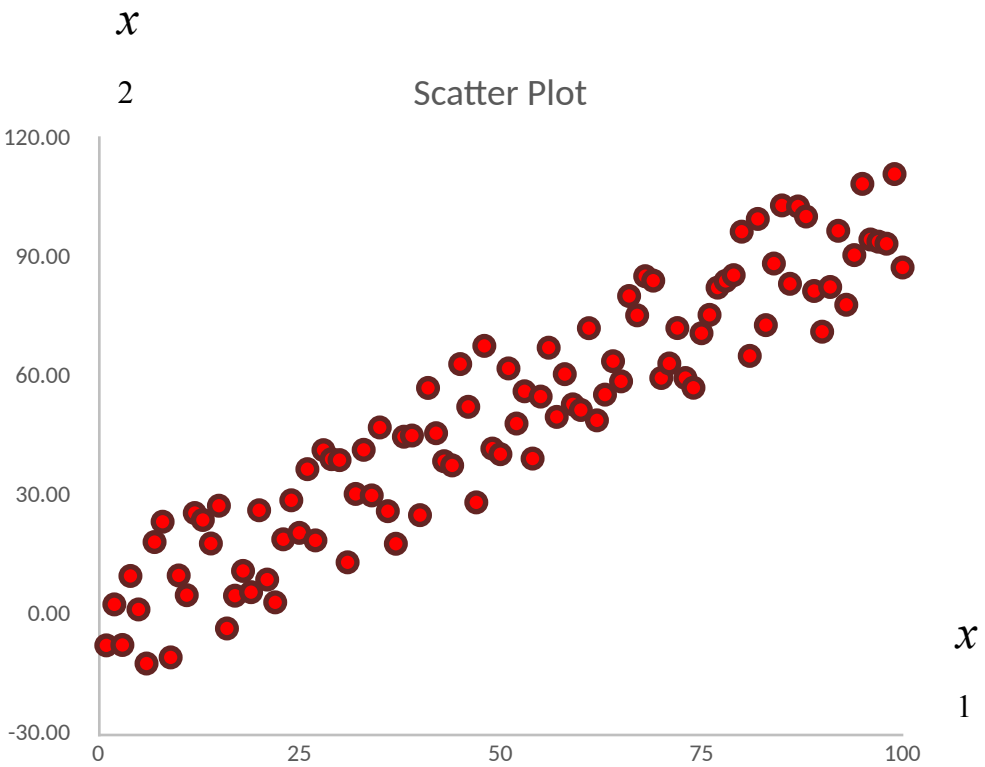




Scatter Diagram

- Plots two features at a time
- Captures the correlation between the two
- Other formats possible

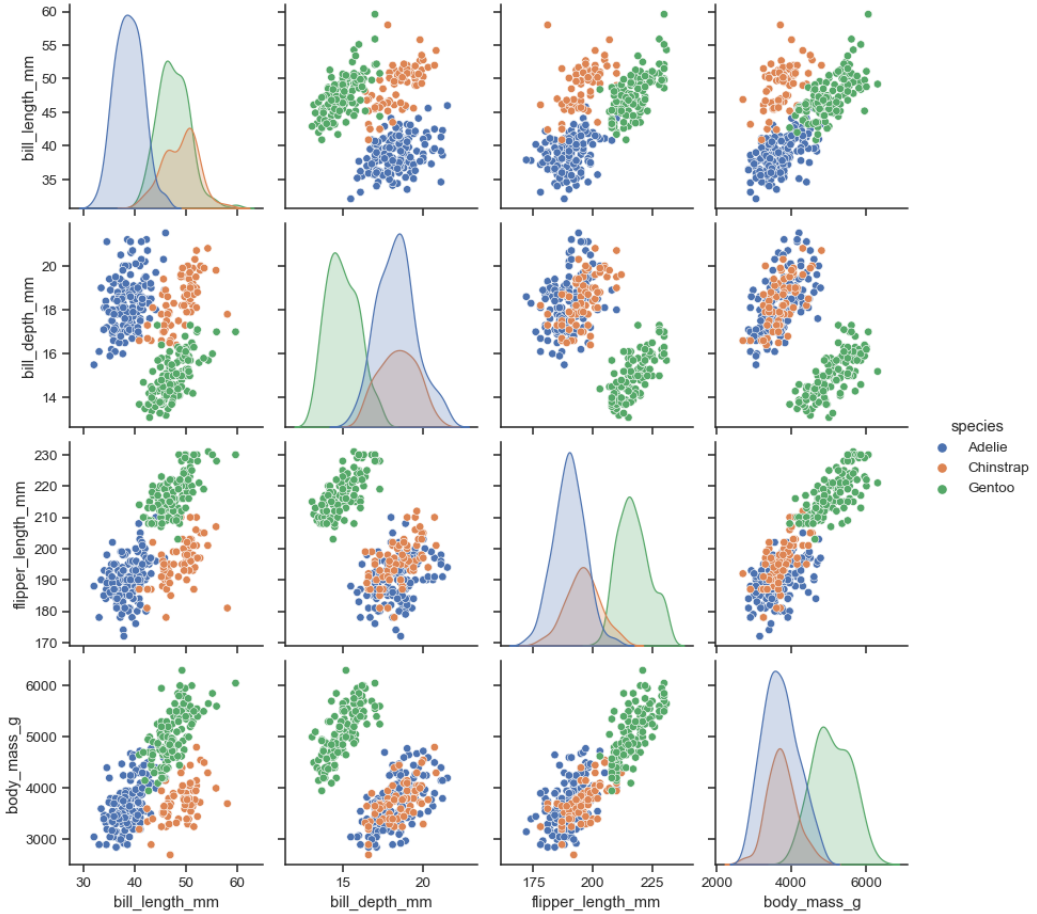




Polar Plot Courtesy: Scatterplot Documentation [matplotlib.org]

Pair Plot

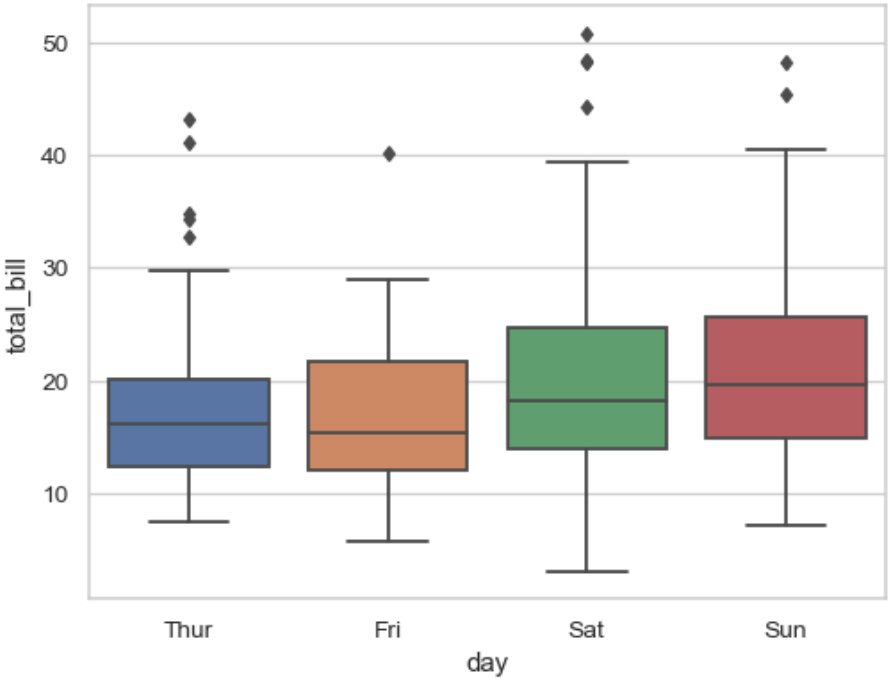
- Plot each pair of features as a matrix
  - Diagonal entries are histograms (densities)
  - Off-diagonal entries are scatter plots
- Can use other plots at each cell.



Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

Box Plot

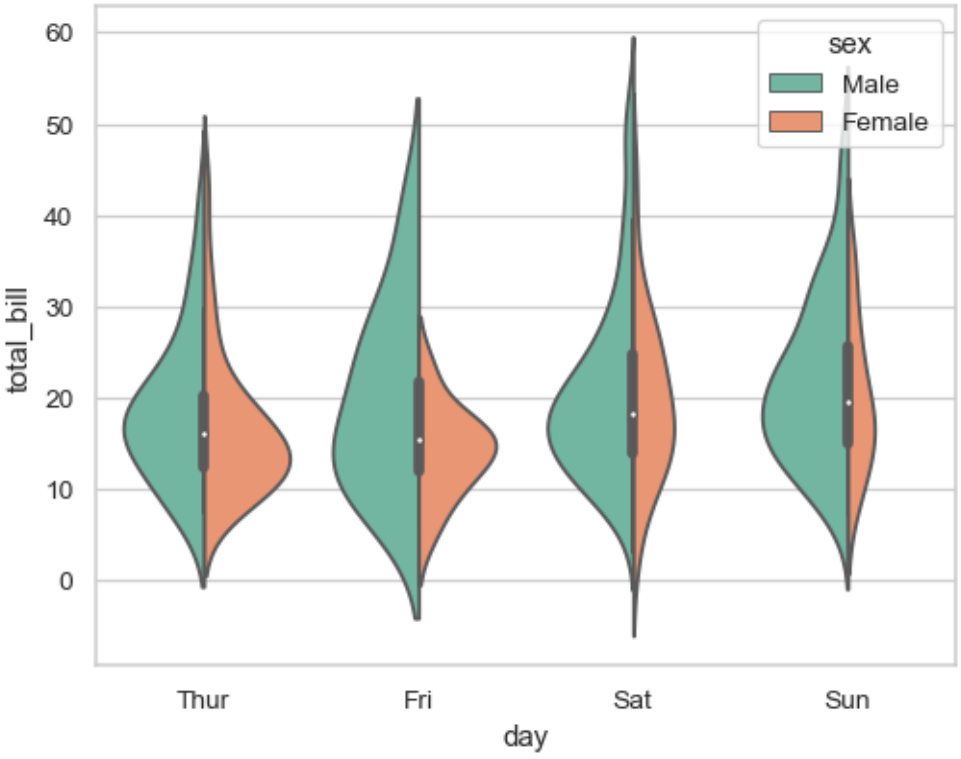
- Show median and quartiles of each feature
  - Outliers are removed
  - Box-and-whisker plot
- Whiskers can represent other percentiles/data
- Simpler than histograms of each feature



Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

Violin Plot

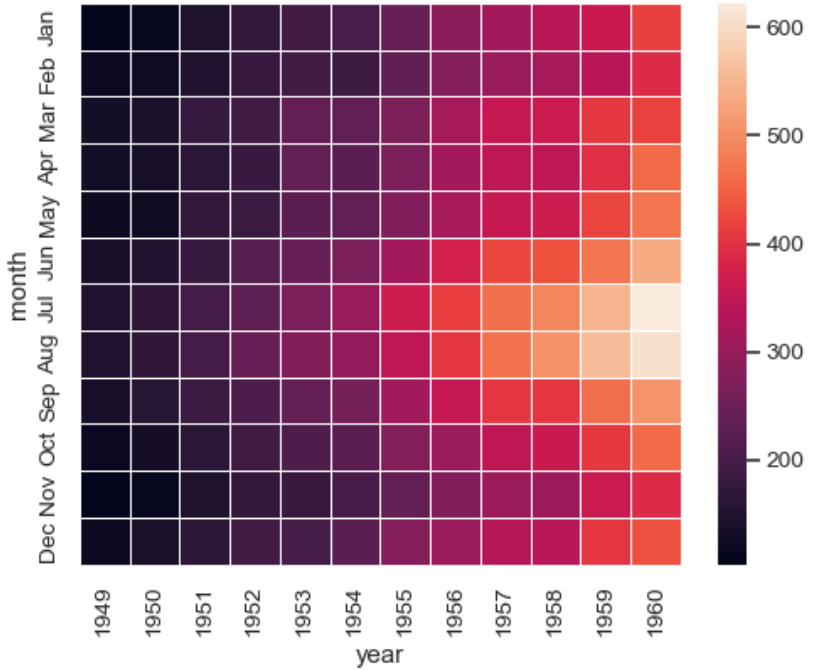
- Shows the density plot of each feature
  - Similar to Box Plot
- Either side can represent different densities
- Densities are smoothened estimates from data



Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

Heat Map

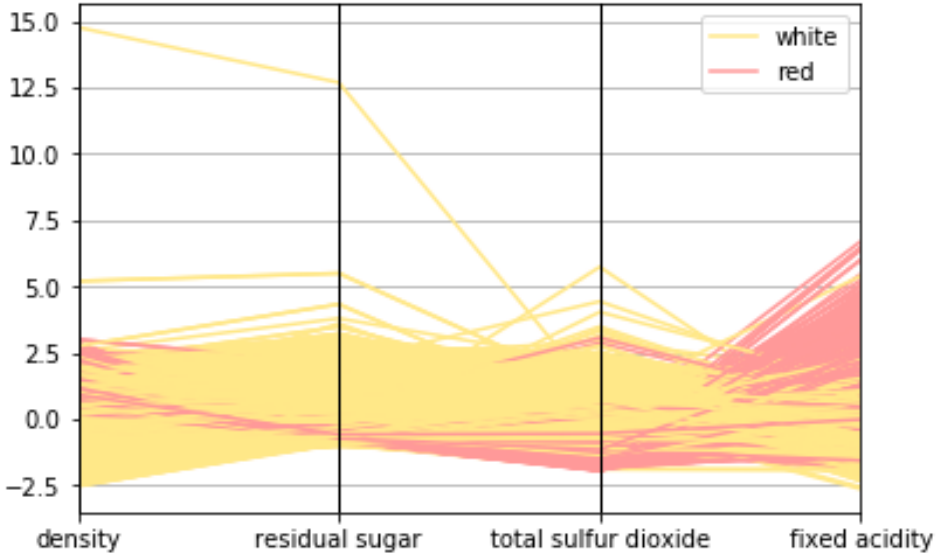
- A color-coded representation of 2D data
- Can be raw data, 2D histogram or any other function of 2 variables
- A color map accompanies the heat map
- We will learn other metrics in future that may be visualized as a heat map



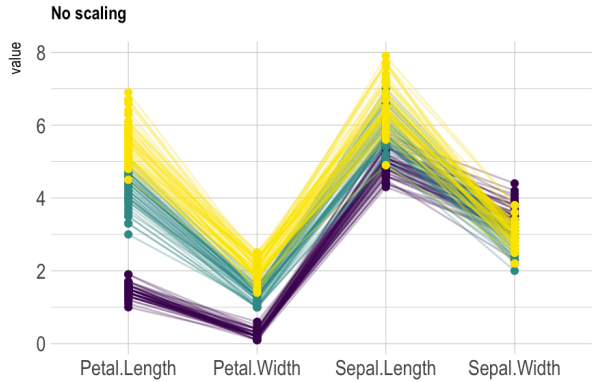
Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

Direct Visualization

- Parallel Co-ordinates
  - Each vertical line is a dimension
  - A data item is connected by line segments
  - Large number of samples clutters the visualization

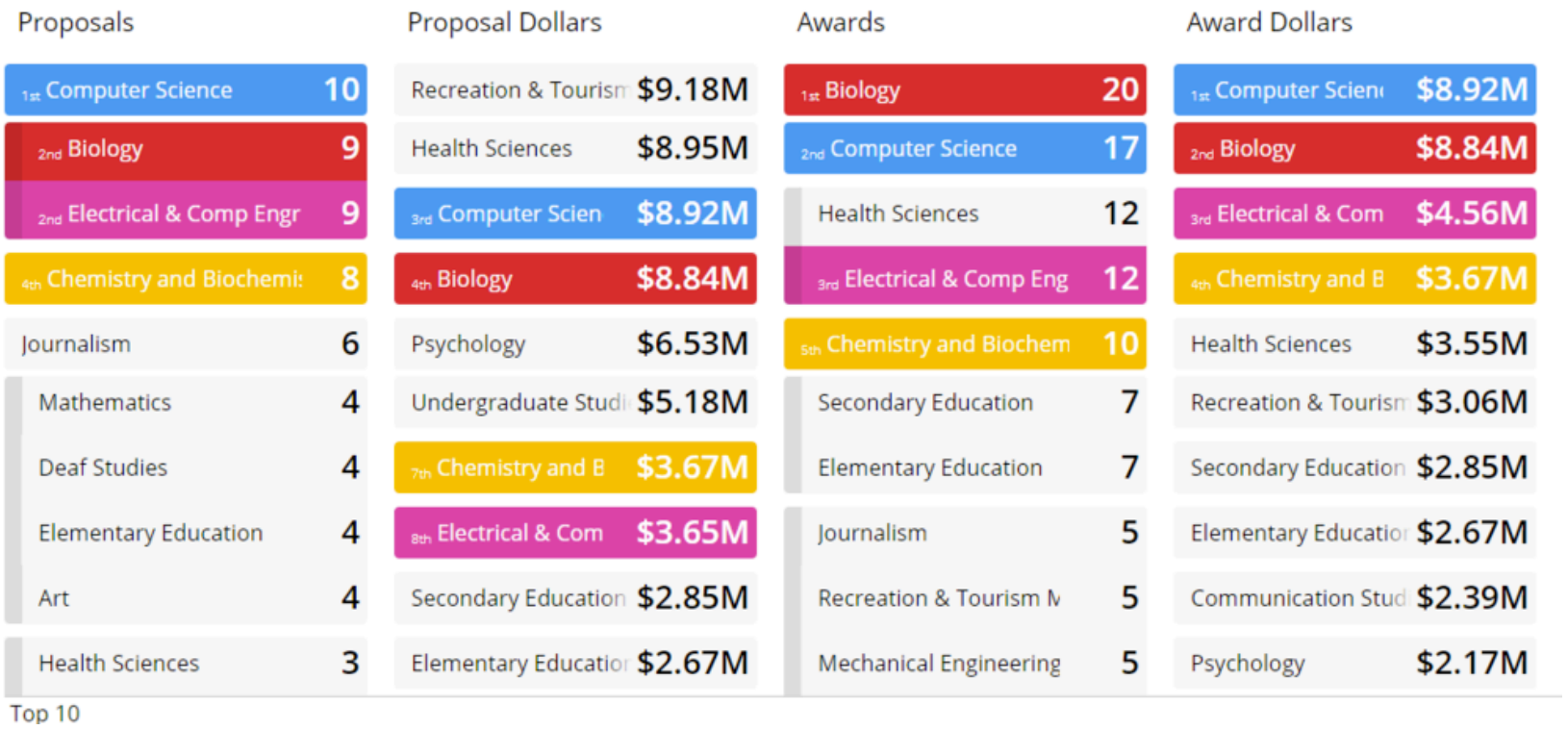


Parallel Coordinates

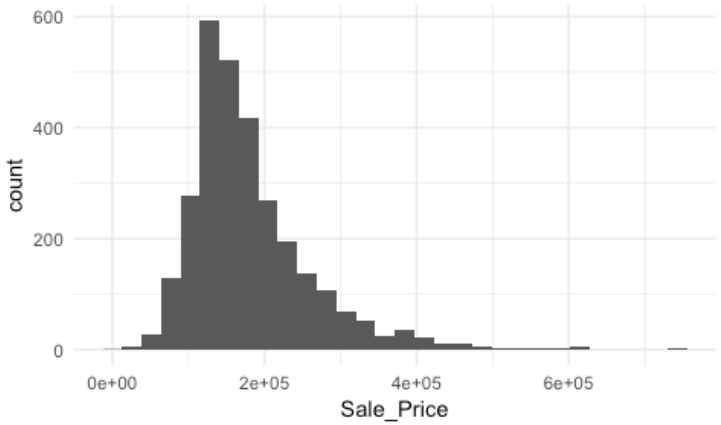


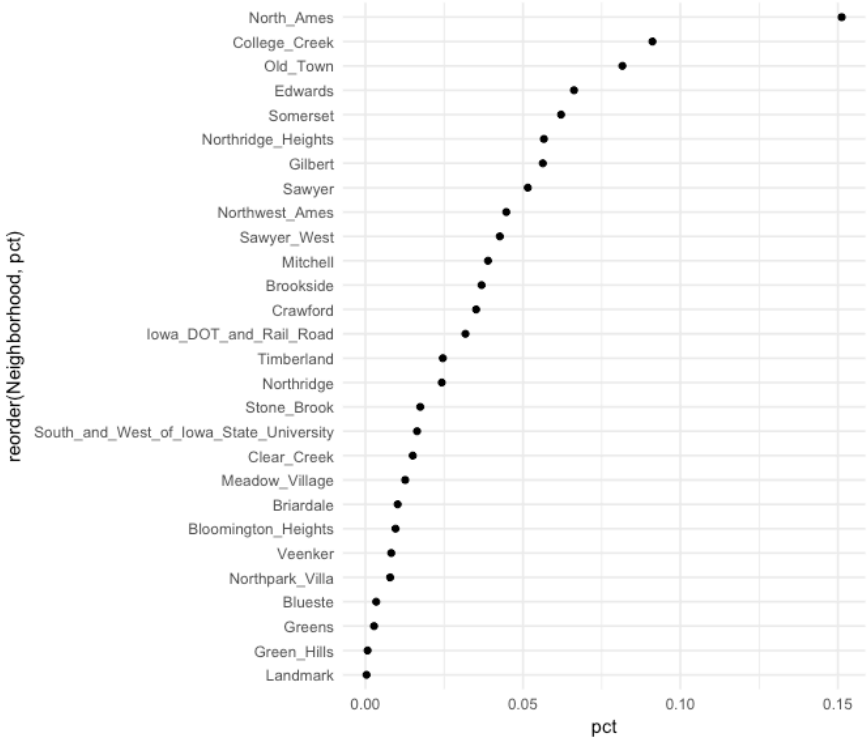
https://www.data-to-viz.com/graph/parallel.html

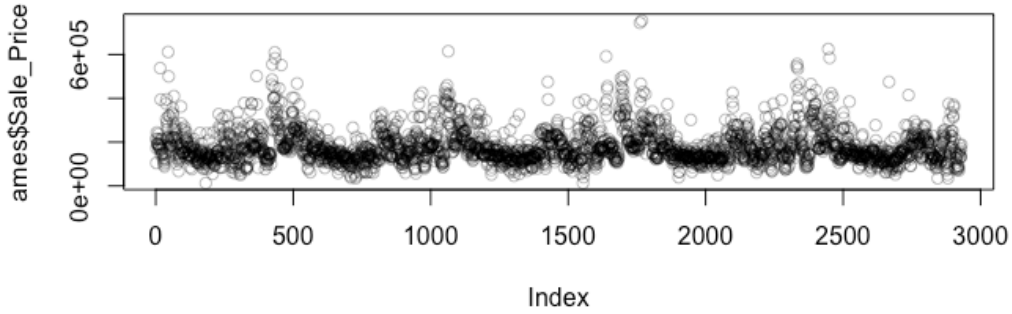
Leaderboard

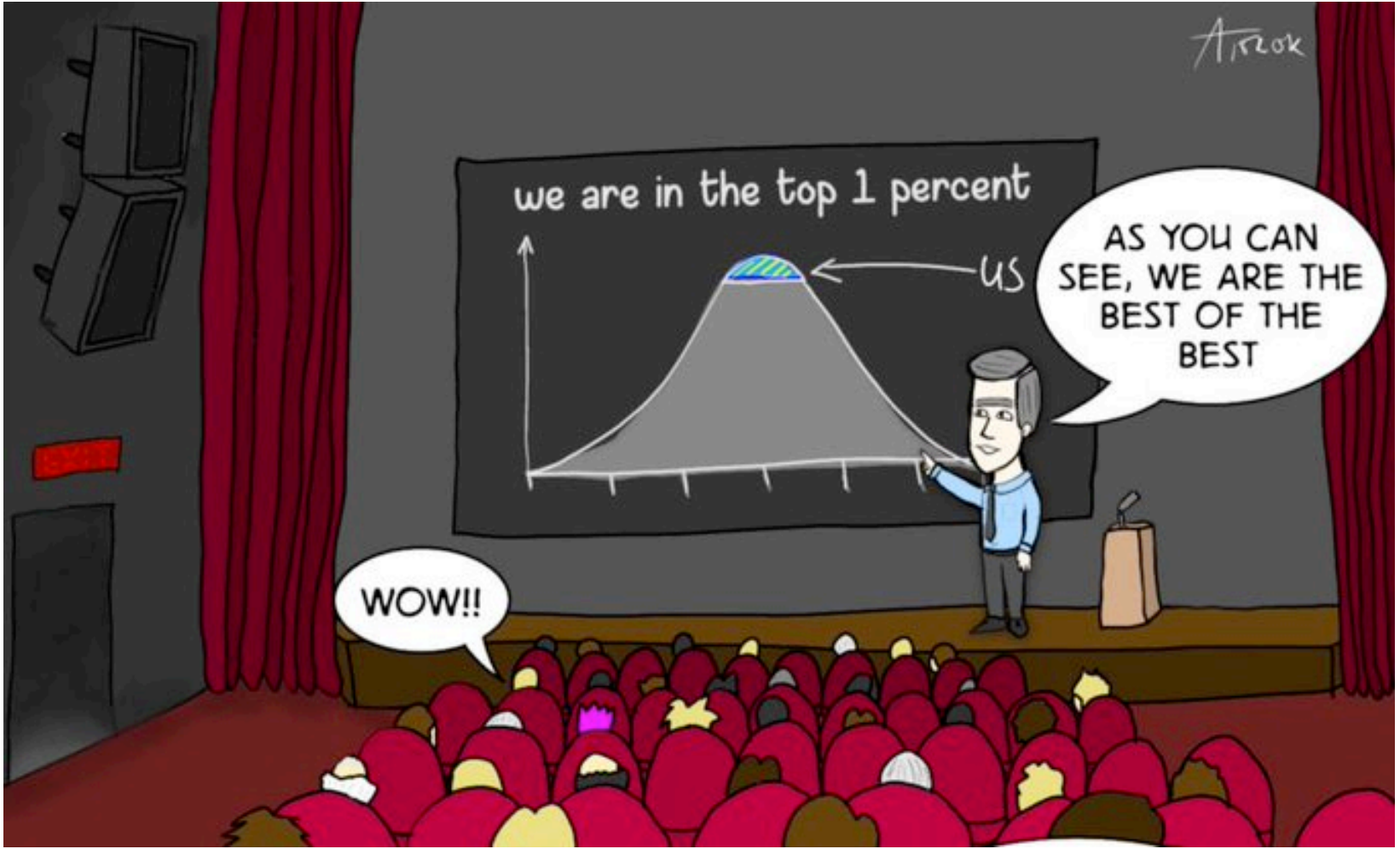


https://www.juiceanalytics.com/writing/writing/parallel-coordinates


600
400
200
0
count
0e+00
2e+05
4e+05
6e+05
Sale_Price


North_Ames
College_Creek
Old_Town
Edwards
Somerset
Northridge_Heights
Gilbert
Sawyer
Northwest_Ames
Sawyer_West
Mitchell
Brookside
Crawford
Iowa_DOT_and_Rail_Road
Timberland
Northridge
Stone_Brook
South_and_West_of_Iowa_State_University
Clear_Creek
Meadow_Village
Briardale
Bloomington_Heights
Veenker
Northpark_Villa
Blueste
Greens
Green_Hills
Landmark
reorder(Neighborhood, pct)
0.00
0.05
0.10
0.15
pct


ames$Sale_Price
0e+00
6e+05
0
500
1000
1500
2000
2500
3000
Index



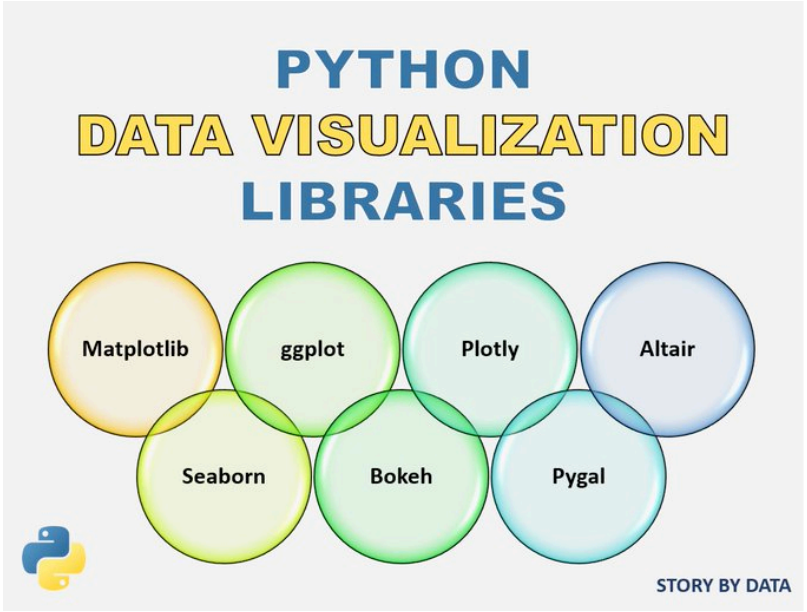
https://pbs.twimg.com/media/F23DL9KWwAAIE0-

"In good information visualization, there are no rules, no guidelines, no templates, no standard technologies, no stylebooks ... You must simply do whatever it takes."

—Edward Tufte

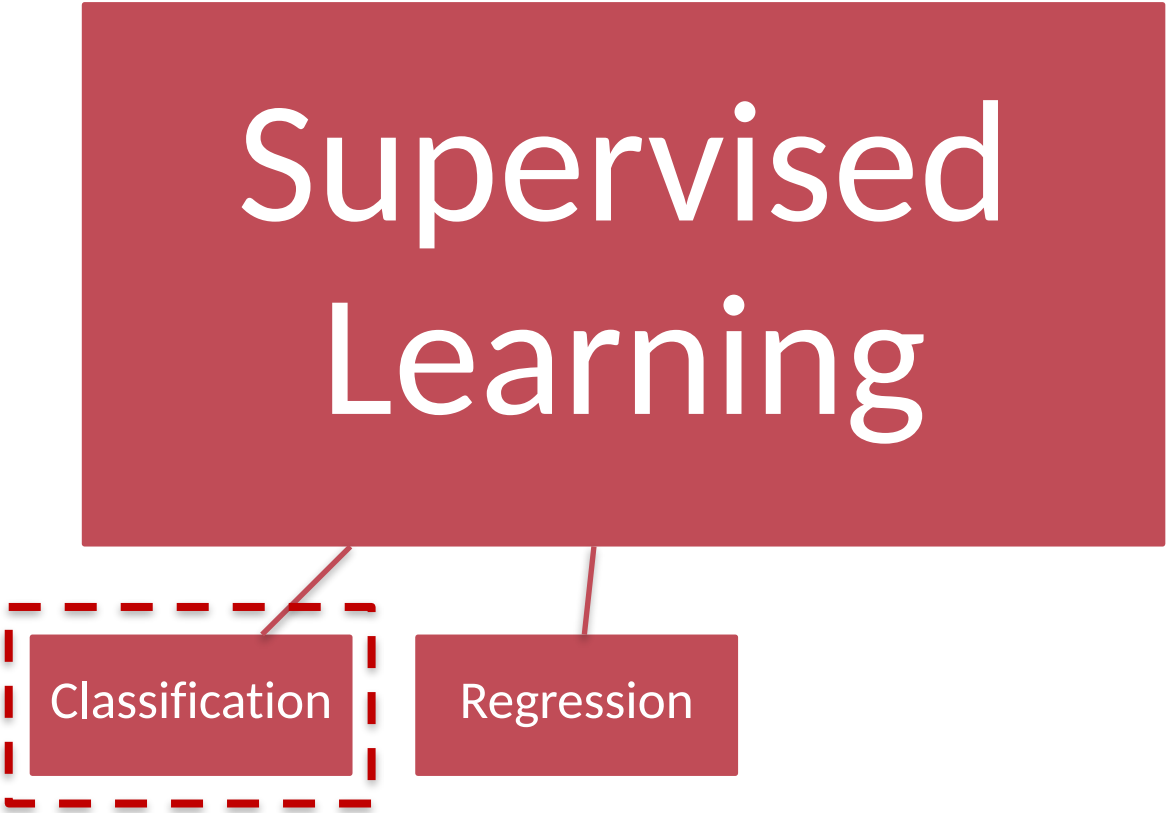
Resources

- https://towardsdatascience.com/5-quick-and-easy-data-visualizations-in-python-with-code-a2284bae952f



https://twitter.com/storybydata/status/1166337648341991424

Supervised Learning


Supervised Learning
Classification
Regression

Classification

Binary

Multi-class

Multi-label

Structure

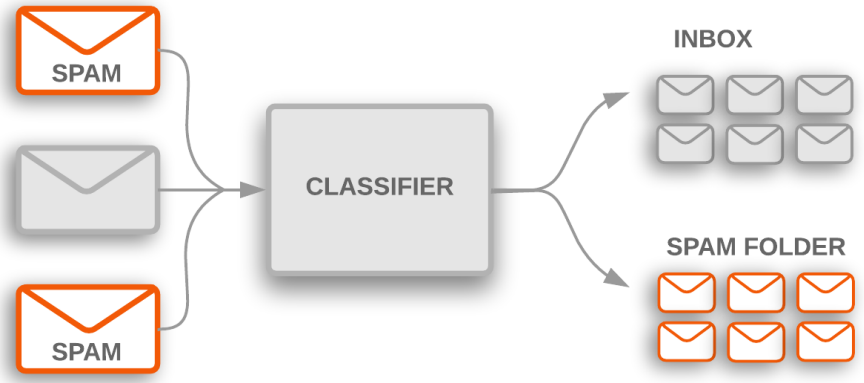
{0,1}

1-of-K

n-of-K

E.g. graph/sequence

Binary Classification



Performance Measures - Accuracy

$$Accuracy = \frac{(100 + 50)}{165} = 0.91$$

$$Misclassification = \frac{(10 + 5)}{165} = 0.09$$

n=165	Predicted: NO	Predicted: YES
Actual: NO	50	10
Actual: YES	5	100

- **Pool of 100 patients' data used for validation of a cancer prediction ML model**
- Prediction:
  - 3 have cancer
  - Rest (100-3=97) are healthy.
- Reality:
  - 1 of the 3 did not actually have cancer !
  - 3 from 97 predicted healthy actually have cancer
- Accuracy =

n=	Predicted: NO	Predicted: YES
Actual: NO		
Actual: YES		

- **Pool of 100 patients' data used for validation of a cancer prediction ML model**
- Prediction:
  - 3 have cancer
  - Rest (100-3=97) are healthy.
- Reality:
  - 1 of the 3 did not actually have cancer !
  - 3 from 97 predicted healthy actually have cancer
- Accuracy = (100 - 4) / 100 = 96% !

n=	Predicted: NO	Predicted: YES
Actual: NO		
Actual: YES		

- **Pool of 100 patients' data used for validation of a cancer prediction ML model**
- Prediction:
  - 3 have cancer → selected for chemotherapy
  - Rest (100-3=97) are healthy.
- Reality:
  - 1 of the 3 did not actually have cancer !
  - 3 from 97 predicted healthy actually have cancer → should have been selected for chemotherapy
- Accuracy = (100 - 4) / 100 = 96% !

n=	Predicted: NO	Predicted: YES
Actual: NO		
Actual: YES		

Performance Measures - Accuracy

$$Accuracy = \frac{(100 + 50)}{165} = 0.91$$

$$Misclassification = \frac{(10 + 5)}{165} = 0.09$$

n=165	Predicted: NO	Predicted: YES
Actual: NO	50	10
Actual: YES	5	100

Performance Measures – Accuracy, TPR, FPR

$$Accuracy = \frac{(100 + 50)}{165} = 0.91$$

$$Misclassification = \frac{(10 + 5)}{165} = 0.09$$

$$FalsePositiveRate(FP) = \frac{(10)}{60} = 0.17$$

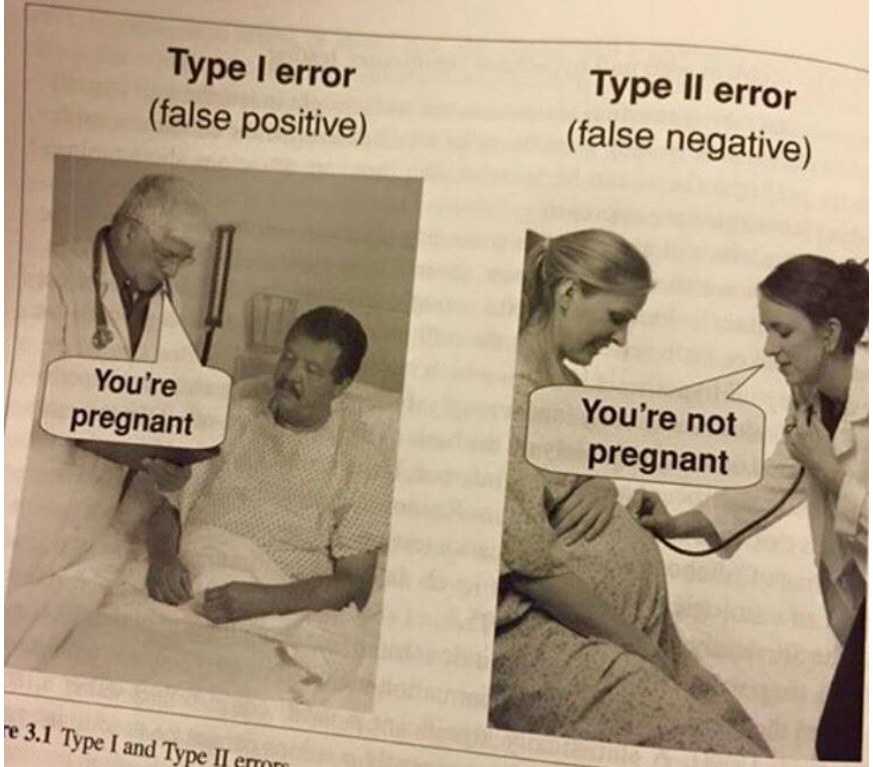
$$FalseNegativeRate(FN) = \frac{(5)}{105} = 0.048$$

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

$$TrueNegativeRate(TN) = \frac{(50)}{60} = 0.833$$

$$TruePositiveRate(TP) = \frac{(100)}{105} = 0.95$$

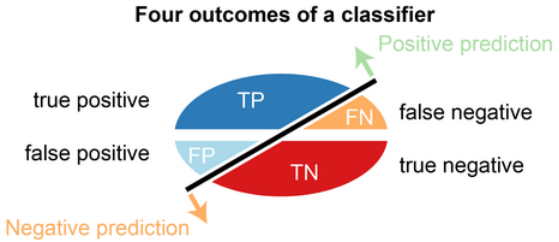
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



re 3.1 Type I and Type II errors

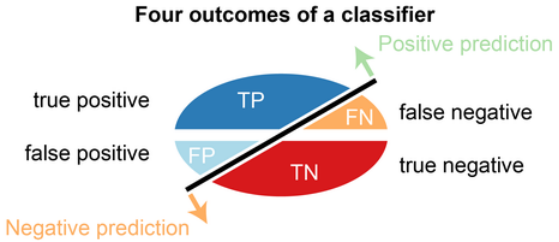
levels to .01 or even .001

Summary of Measures

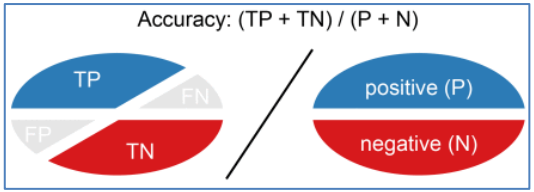


n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Summary of Measures

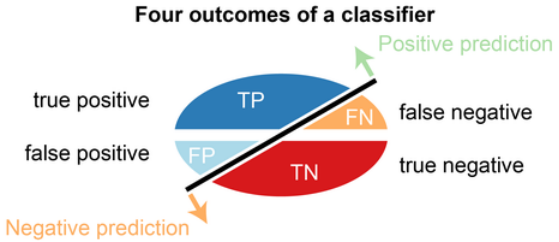


n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

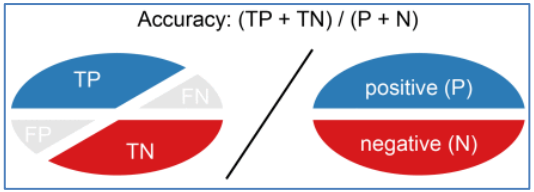


% of correct predictions

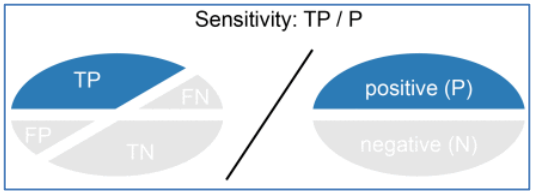
Summary of Measures



n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

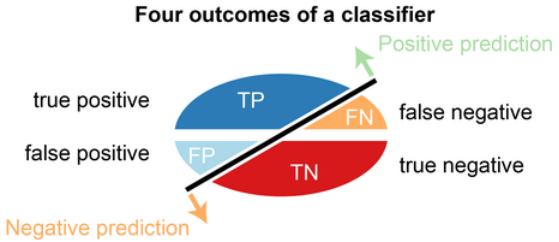


% of correct predictions

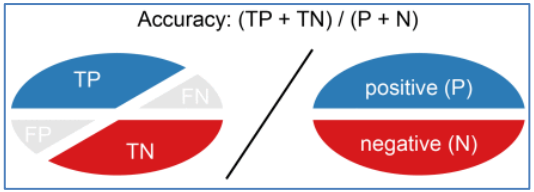


% of + class correctly predicted
[aka Recall / TPR]

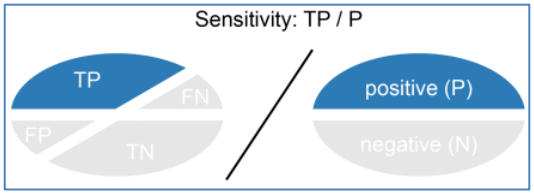
Summary of Measures



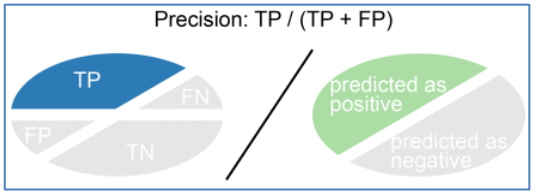
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



% of correct predictions

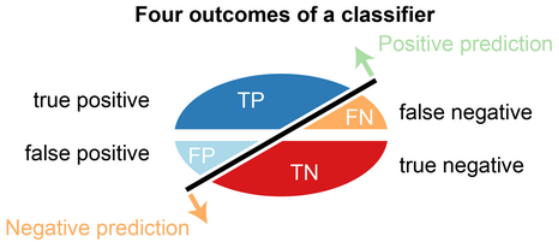


% of + class correctly predicted
[aka Recall / TPR]

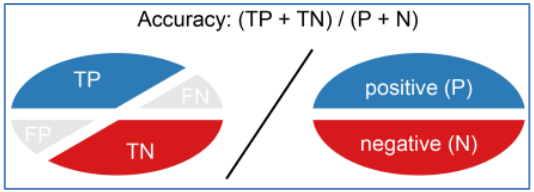


correct prediction of + class
[aka Precision]

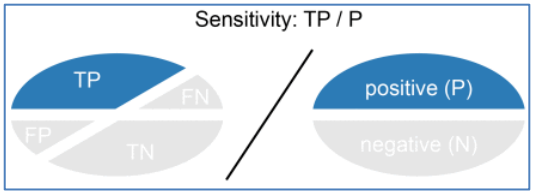
Summary of Measures



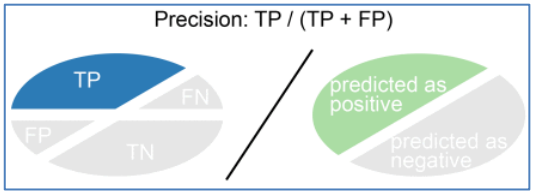
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



% of correct predictions



% of + class correctly predicted
[aka Recall / TPR]



correct prediction of + class
[aka Precision]

% of – class incorrectly predicted

- **Cancer-Prediction System**
- Precision =
- Recall =
- Accuracy =

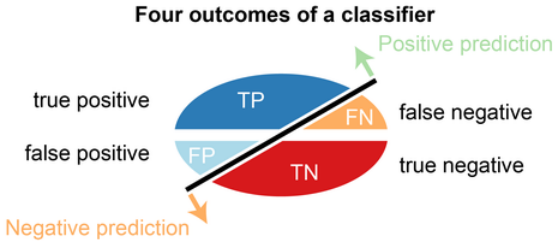
- **Cancer-Prediction System**
- Precision = 2/(2+1) = 67%
- Recall = 2/(2+3) = 40%
- Accuracy = (94+2)/100 = 96%

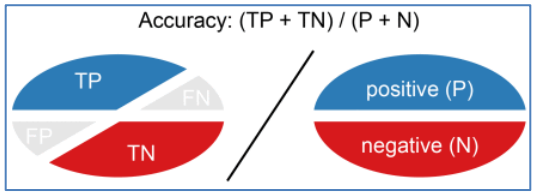
What measure are we optimizing for ?

- Screening for a terminal disease
- Do not want to miss anyone: Maximize Recall
- Classification between apples and oranges
  - Both types of errors are equally imp.: Maximize Accuracy
- Automatic bombing on detecting a target from a drone
  - Should not hurt civilians: Zero False Positives
- Giving access to a secure installation
  - No access to unauthorized personnel: Low False Positives

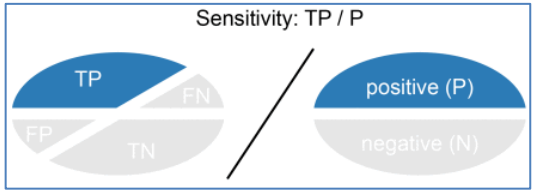
Accuracy vs Precision vs Recall

- Accuracy : Performance w.r.t both classes
- Recall : Performance w.r.t '+' class
- Precision : Reliability of predictions w.r.t '+' class

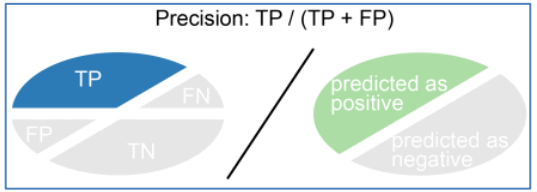




% of correct predictions



% of + class correctly predicted
[aka Recall / TPR]



correct prediction of + class
[aka Precision]

Utility and Cost

- Sometimes, there is a cost for each error
  - E.g. Earthquake prediction
    - False positive: Cost of preventive measures
    - False negative: Cost of recovery

- Detection Cost (Event detection)
  - Cost = C_{FP} * FP + C_{FN} * FN

Farmer Shri MoneyBags and ML-FruitPicker

- MB : I want an automated fruit picker and packer. I will pay an unholy amount for it.
- You (having just finished this lecture) : Sure
- *You (Thinking): I love unholy amounts of money* 😎

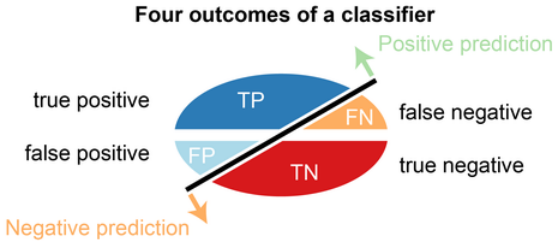
Farmer Shri MoneyBags and ML-FruitPicker

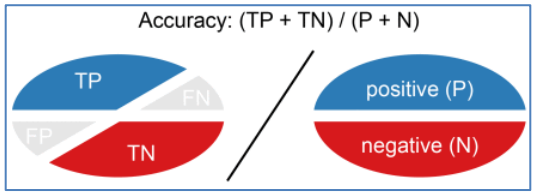
After 6 months …

- MB : Well ?
- You : I have a High Precision ML-FruitPicker. But its Recall is 20% !
- MB : (confused) Precision ? Recall ?
- *You : (thinking) Should I go over first 3 lectures of SMAI with MB ? He'll probably run away !*
- You : It rejects 80% of good, pickable fruit, but whatever it picks, those fruits are good !
- MB : I'll take your system. How do I transfer unholy amount of money to you ?
- You : 😯
- MB (seeing your shocked face) : See, in a batch of 100 fruits, 10 fruits are usually bad. Among the 90 good ones, your system will select 18 of them on average. But from any given selection, I pack only 8.

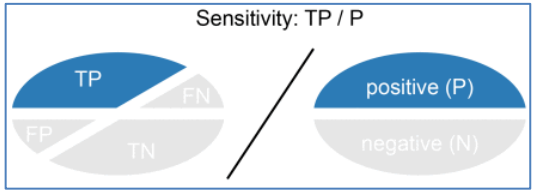
Accuracy vs Precision vs Recall

- Monitor **Precision** if a false positive carries higher cost.
- Monitor **Recall** if a false negative carries higher cost.

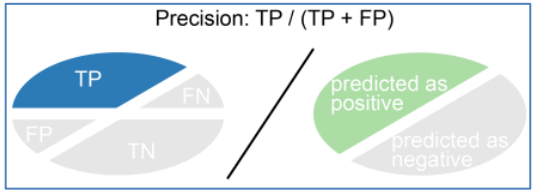




% of correct predictions



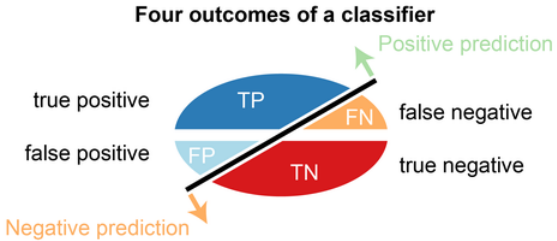
% of + class correctly predicted
[aka Recall / TPR]

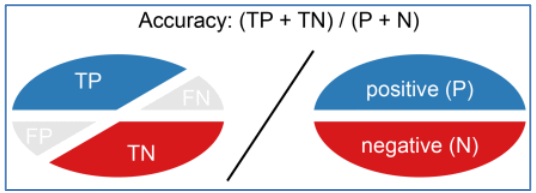


correct prediction of + class
[aka Precision]

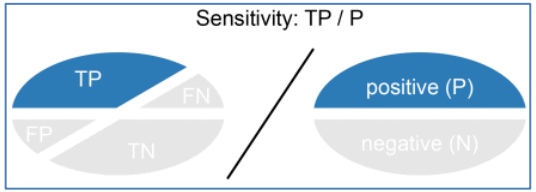
Accuracy vs Precision vs Recall

- **Precision →** Cost of inclusion
- **Recall →** Cost of exclusion

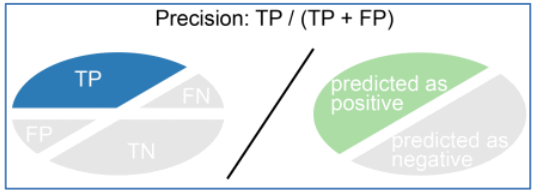




% of correct predictions

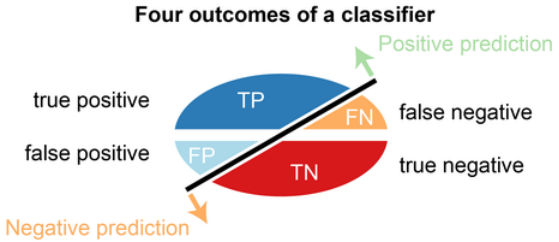


% of + class correctly predicted
[aka Recall / TPR]

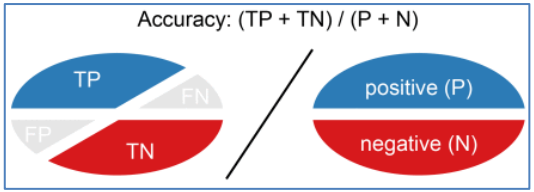


correct prediction of + class
[aka Precision]

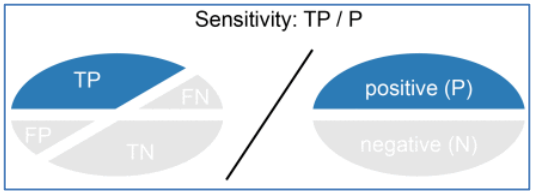
Summary of Measures



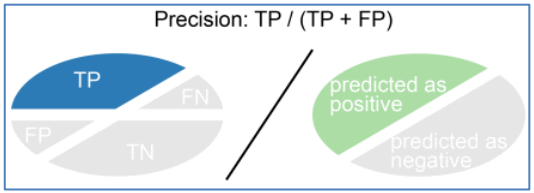
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



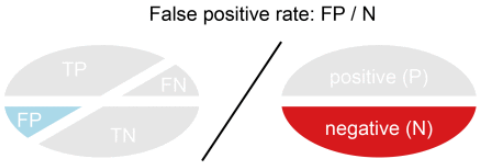
% of correct predictions



% of + class correctly predicted
[aka Recall / TPR]



correct prediction of + class



% of – class incorrectly predicted

F1-score: A unified measure

- What to do when one classifier has better precision but worse Recall, while other classifier behaves exactly opposite?
  - F-measure (Information Retrieval)
    - $F_1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$

Utility and Cost

- What to do when one classifier has better Precision but worse Recall, while other classifier behaves exactly opposite?
  - F-measure (Information Retrieval)
    - $F_1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$

→ **F1 measure punishes extreme values more !**

→ **Definition of Recall and Precision have same numerator, different denominators. A sensible way to combine them is harmonic mean.**