09.01.2020

Statistical Methods in AI (CSE/ECE 471)

Lecture-3: Intro to Performance Measures, Benchmarking

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

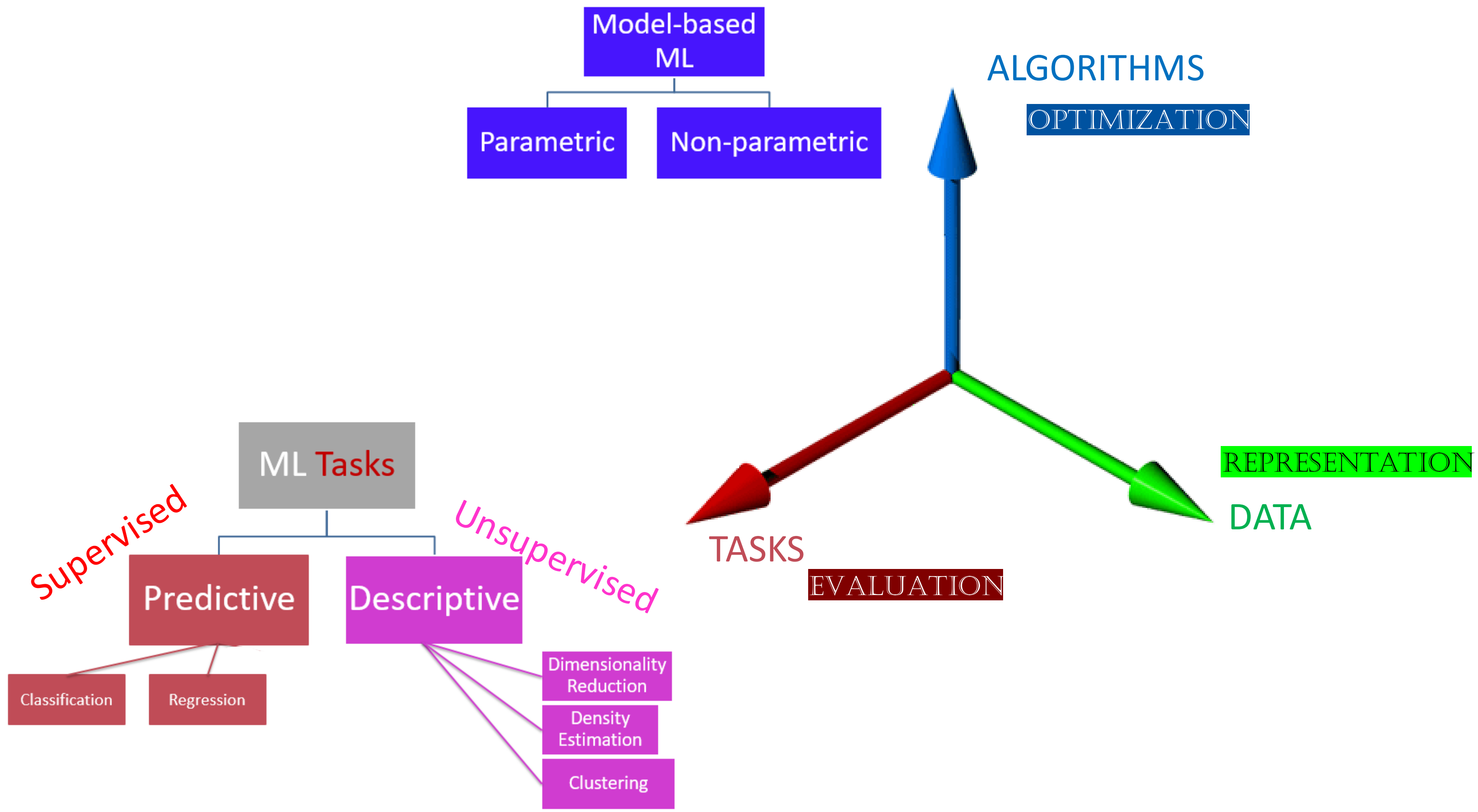
Vineet Gandhi (v.gandhi@iiit.ac.in)

Center for Visual Information Technology (CVIT)

IIIT Hyderabad

Machine Learning

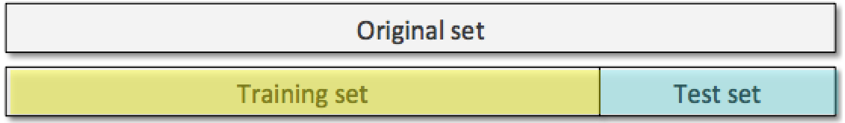
Study of Algorithmic methods that use data to improve their knowledge of a task


Model-based ML
Parametric
Non-parametric
ALGORITHMS
OPTIMIZATION
REPRESENTATION
DATA
TASKS
EVALUATION
ML Tasks
Supervised
Unsupervised
Predictive
Descriptive
Classification
Regression
Dimensionality Reduction
Density Estimation
Clustering

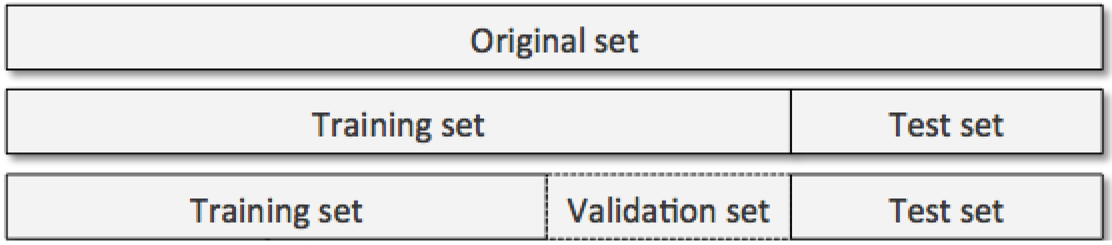
An interview analogy

1. Collect worked out problems (Q, S are both known)
2. Prepare on ALL the available problems.
3. Go for interview.

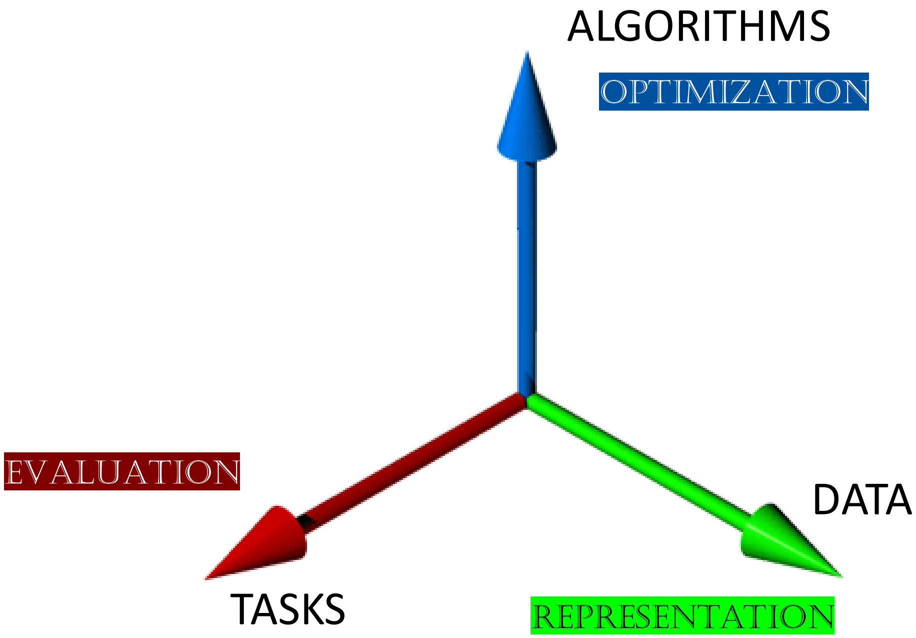
1. Collect worked out problems (Q,S are both known)
2. Randomly set aside a small number of problems.
3. Prepare on rest of the problems.
4. Take a mock interview containing all the 'set aside' problems.
5. Score answers and compare with solution.
6. Use mistakes to decide which topics to prepare better on.
7. Go for interview.

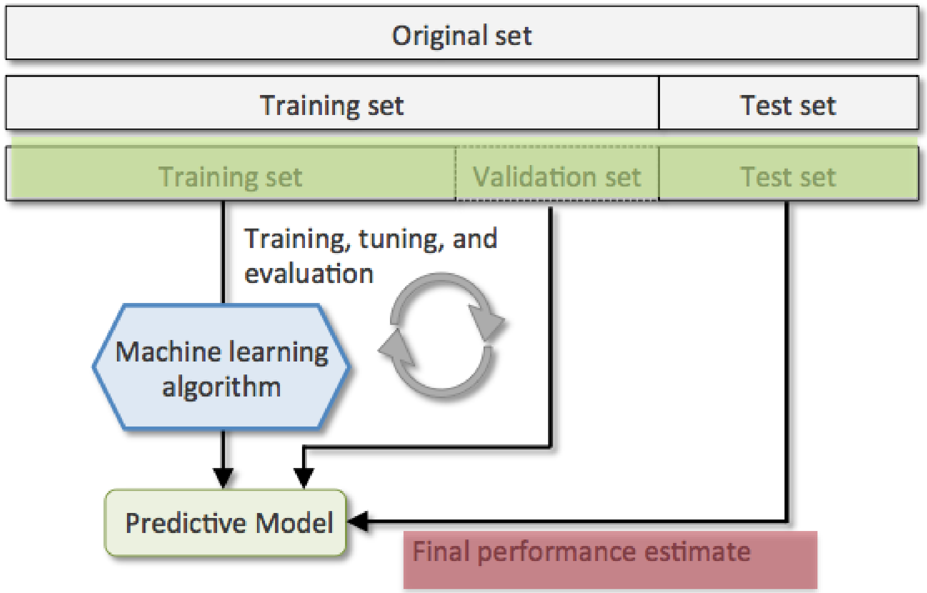


The Train-Validation-Test paradigm

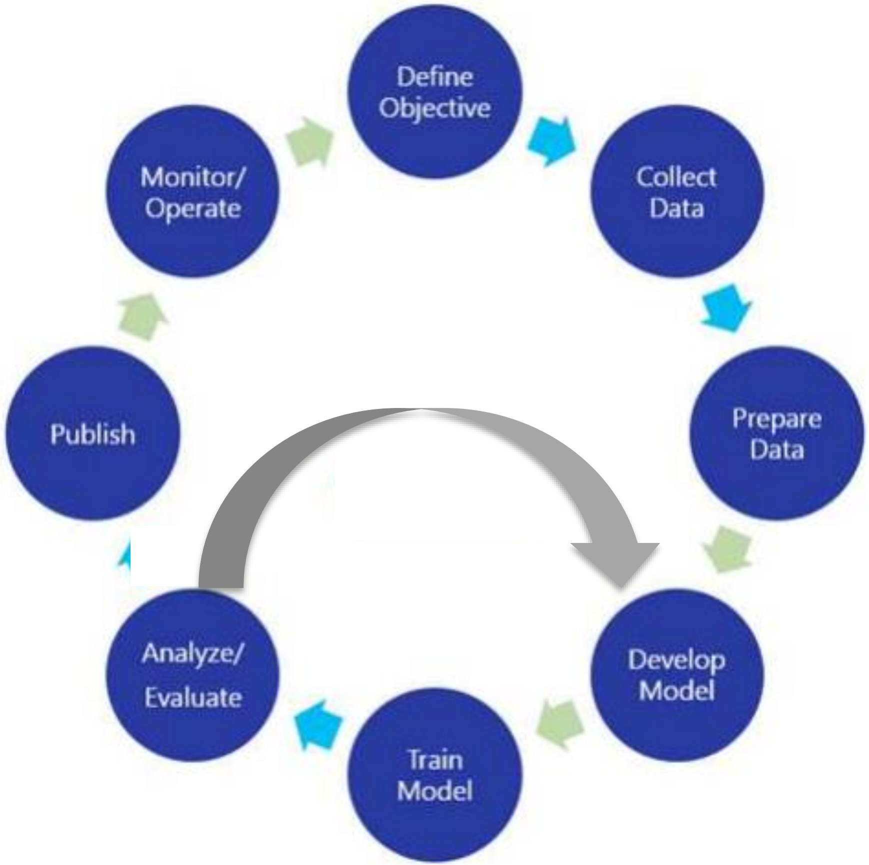


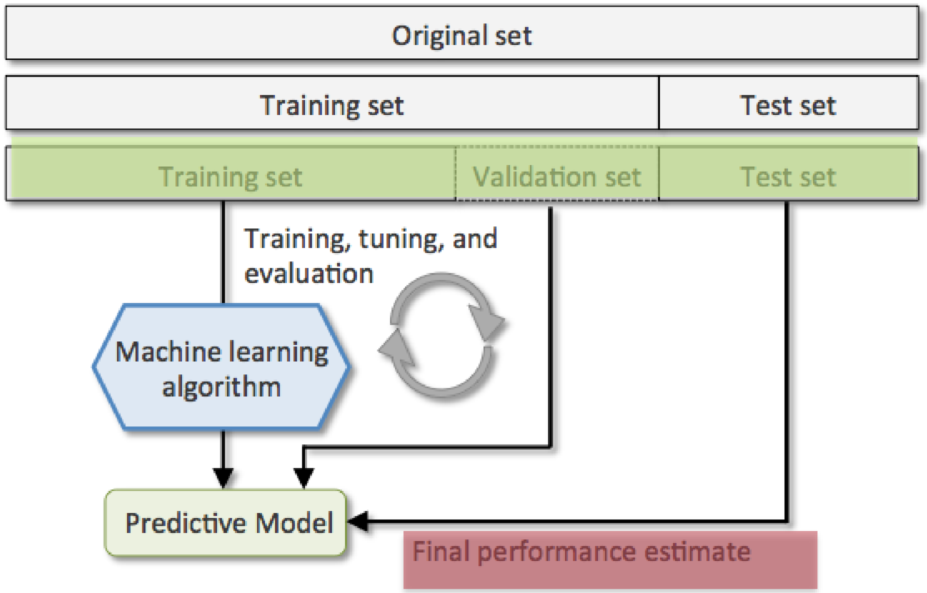
The Train-Validation-Test paradigm

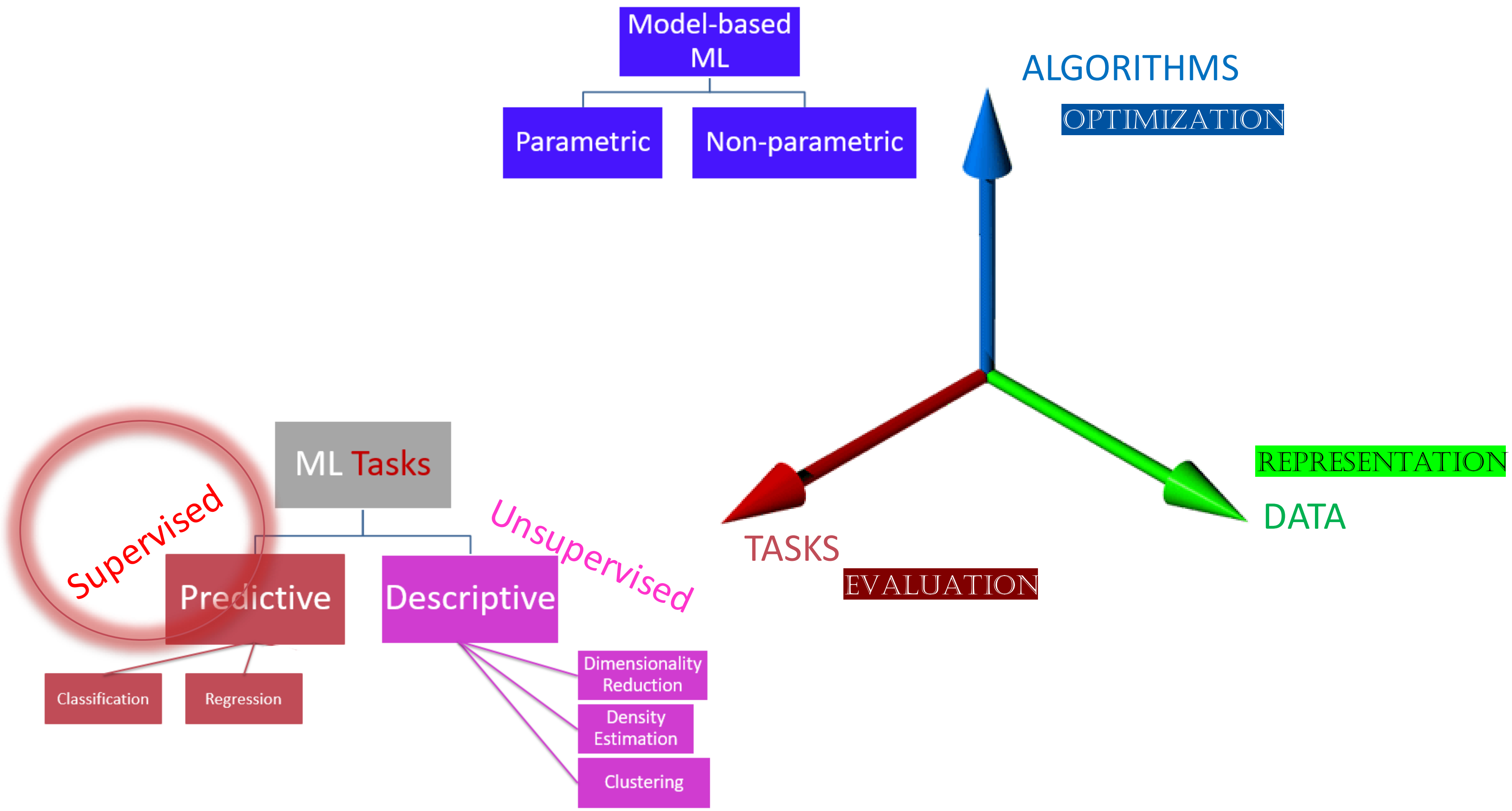


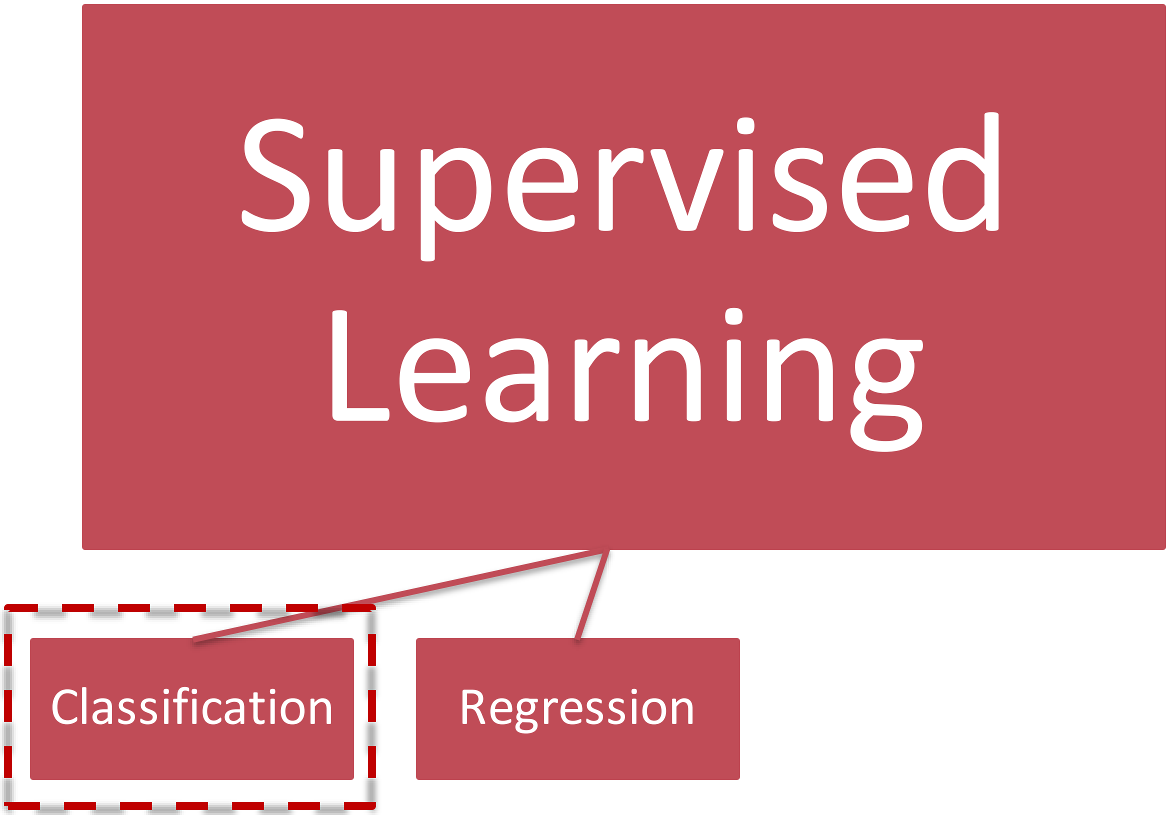


The Train-Validation-Test paradigm





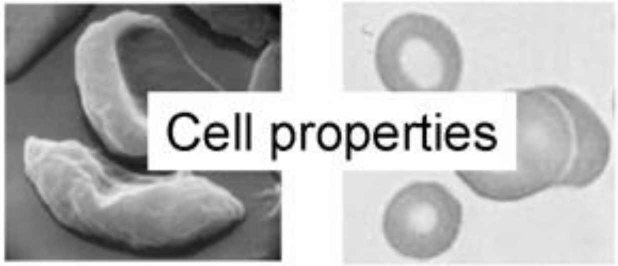

Model-based ML
Parametric
Non-parametric
ALGORITHMS
OPTIMIZATION
ML Tasks
Supervised
Unsupervised
Predictive
Descriptive
Classification
Regression
Dimensionality Reduction
Density Estimation
Clustering
TASKS
EVALUATION
REPRESENTATION
DATA


Supervised Learning
Classification
Regression

ML::Tasks → Predictive → Classification

Feature Space $\mathcal{X}$ **Label** Space $\mathcal{Y}$

"Sports"
"News"
"Science"
...

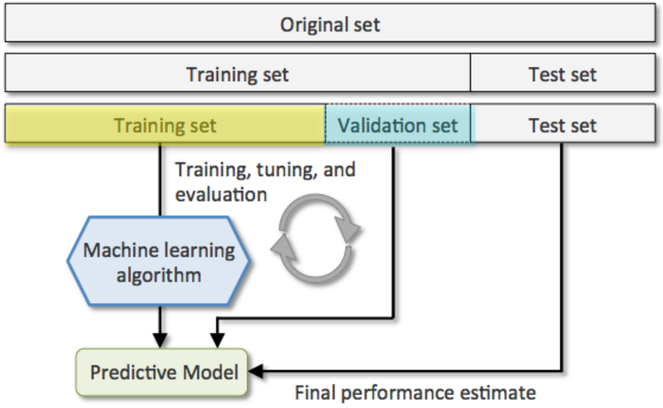


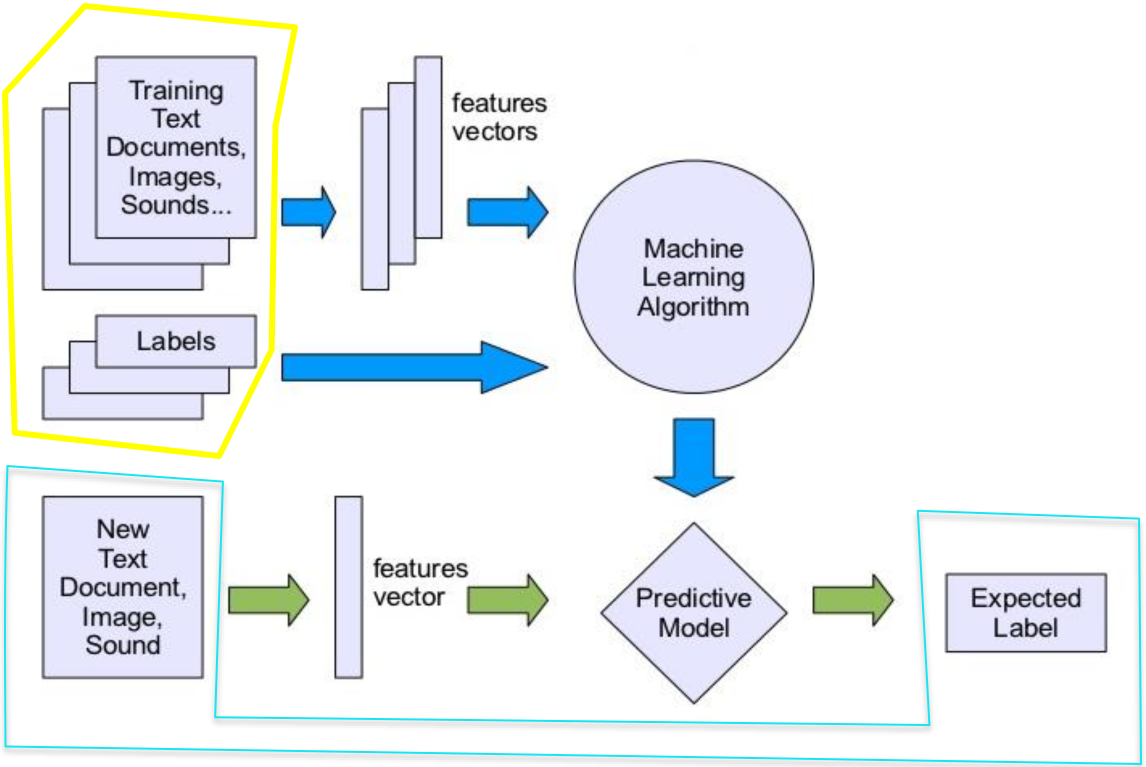
"Anemic cell"
"Healthy cell"

Task: Given $X \in \mathcal{X}$, predict $Y \in \mathcal{Y}$.

Discrete Labels

The Train-Validation-Test paradigm

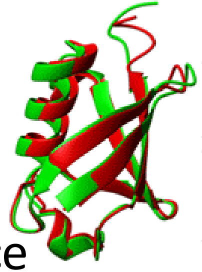




x → Supervised Learning → y

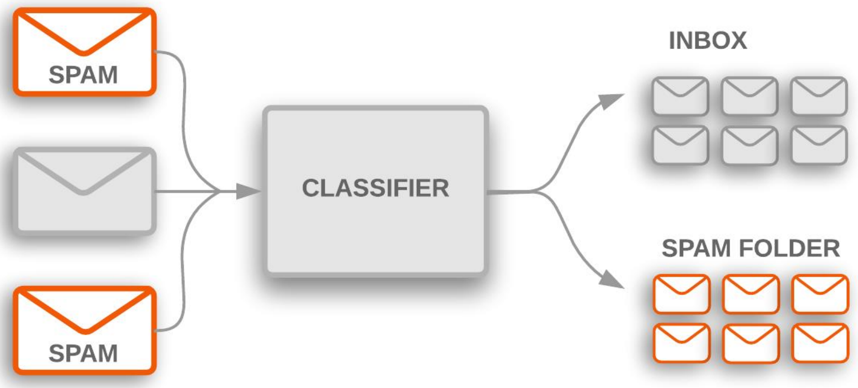
Classification

- Binary: {0,1}
- Multi-class: 1-of-K
- Multi-label: n-of-K
- Structure:



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

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

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

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

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

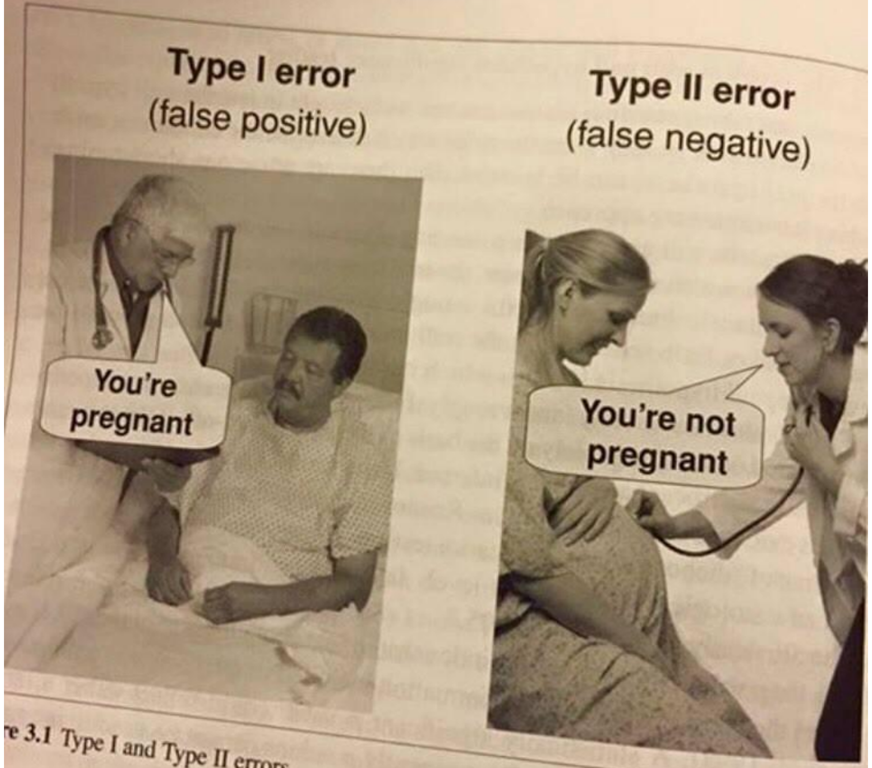
$$FalseNegativeRate(FN) = \frac{(5)}{105} = 0.048$$

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

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

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

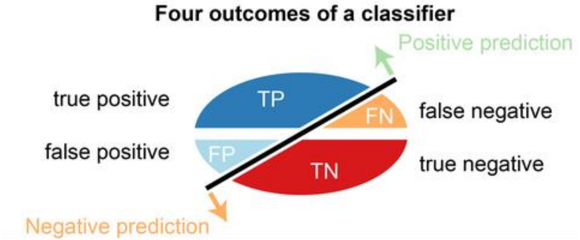
n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



re 3.1 Type I and Type II errors

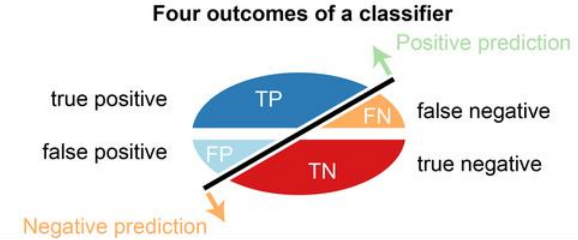
levels to .01 or even .001

Summary of Measures

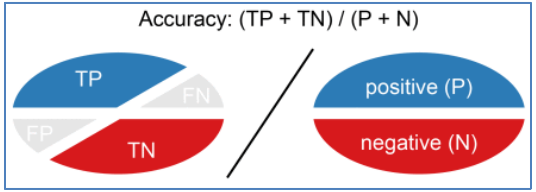


n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Summary of Measures

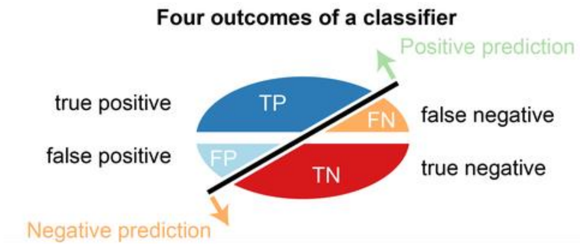


n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

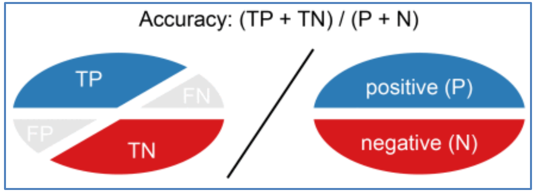


% of correct predictions

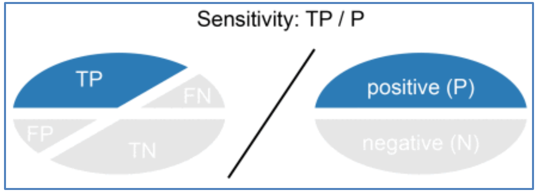
Summary of Measures



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

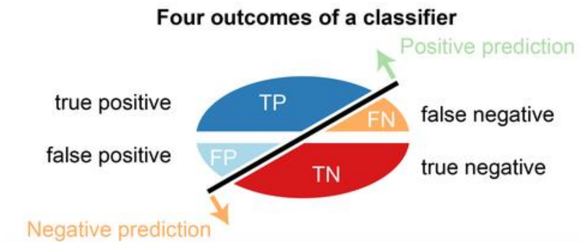


% of correct predictions

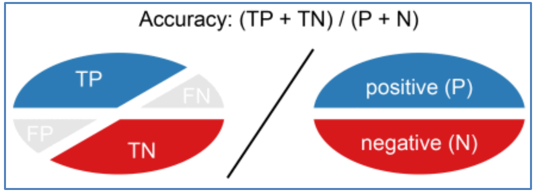


% of + class correctly predicted
[aka Recall / TPR]

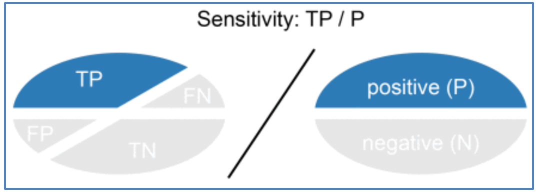
Summary of Measures



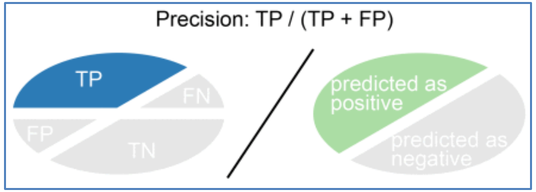
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



% of correct predictions

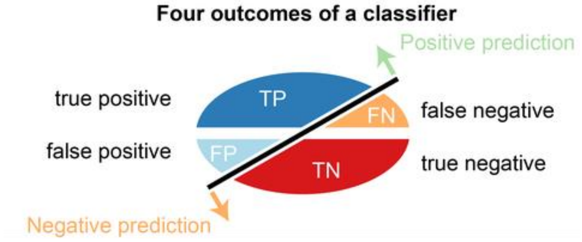


% of + class correctly predicted
[aka Recall / TPR]

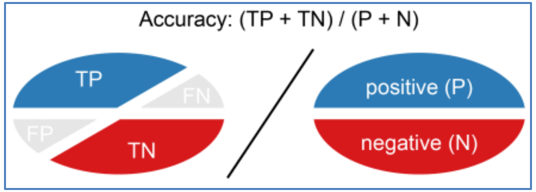


correct prediction of + class
[aka Precision]

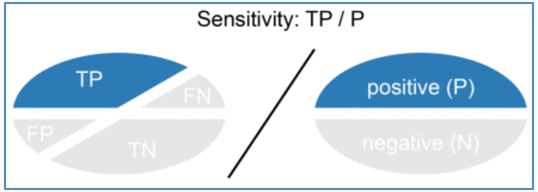
Summary of Measures



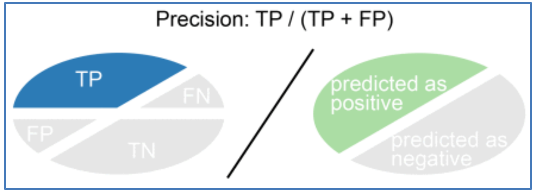
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



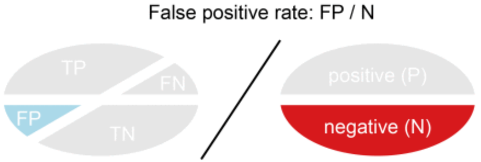
% of correct predictions



% 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%

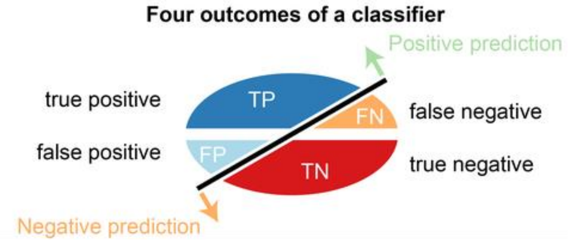
Precision and Recall – examples

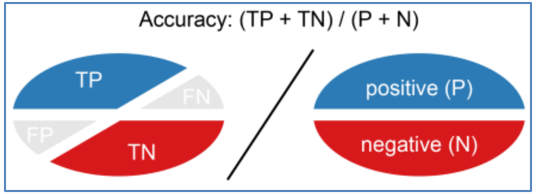
- A system which needs to launch a missile at a terrorist hideout located in a dense urban area.
- Precision not 100% ➔ civilian casualties

- A system which needs to identify cancer-risk patients
- Recall not 100% ➔ some patients will die of cancer

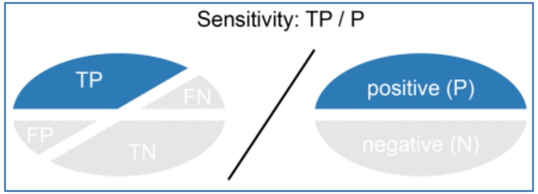
Accuracy vs Precision vs Recall

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

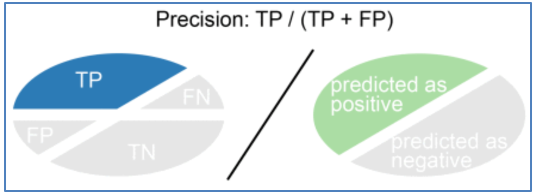




% of correct predictions



% 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* 😎
- *(rapid cuts of time passing, you collecting data, referring to SMAI slides, coding ; dramatic music in background)*

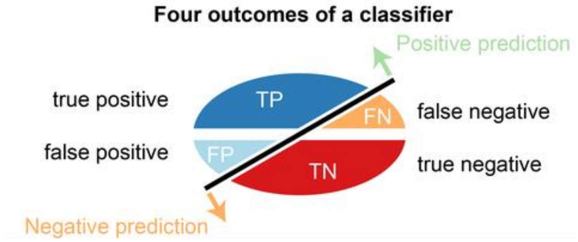
Farmer Shri MoneyBags and ML-FruitPicker

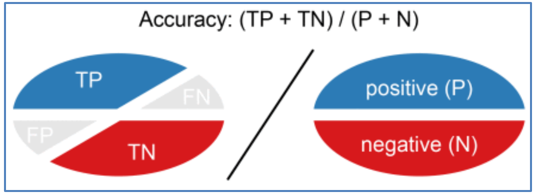
After 6 months …

- 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.

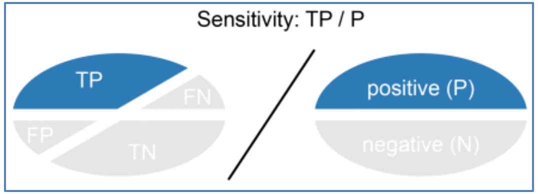
Accuracy vs Precision vs Recall

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

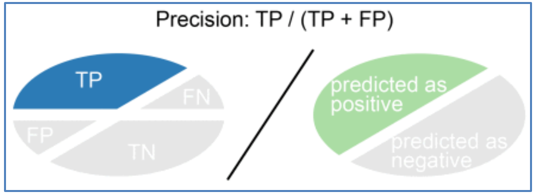




% of correct predictions



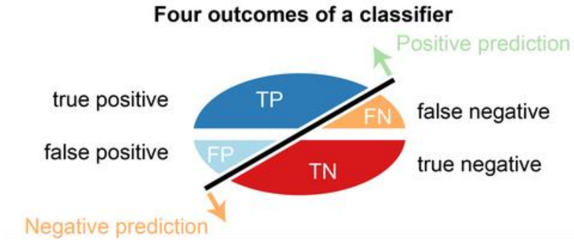
% of + class correctly predicted
[aka Recall / TPR]

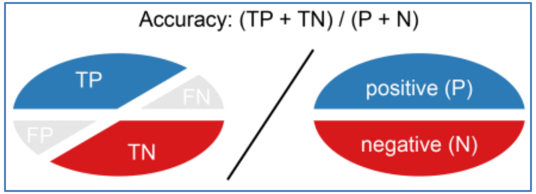


correct prediction of + class
[aka Precision]

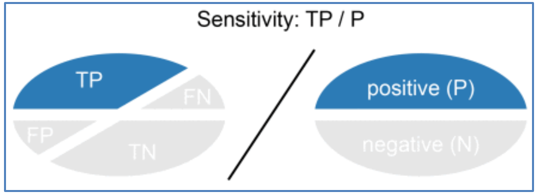
Accuracy vs Precision vs Recall

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

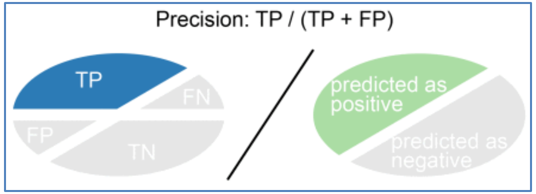




% of correct predictions

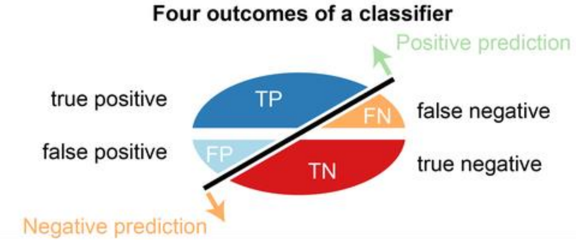


% of + class correctly predicted
[aka Recall / TPR]

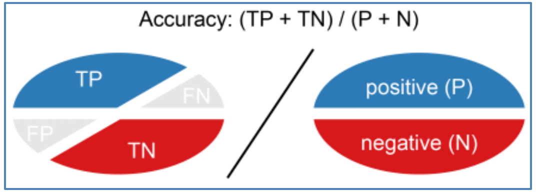


correct prediction of + class
[aka Precision]

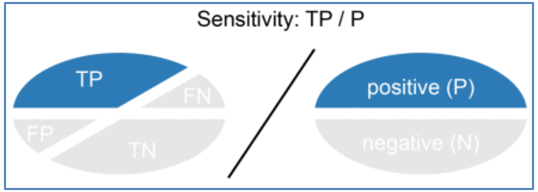
Summary of Measures



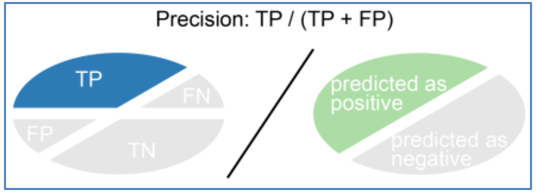
n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	



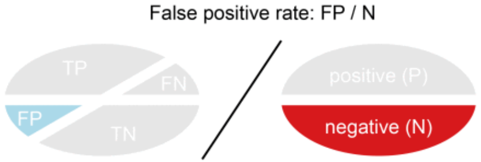
% of correct predictions



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



correct prediction of + class



% of – class incorrectly predicted

F1-score: A unified measure

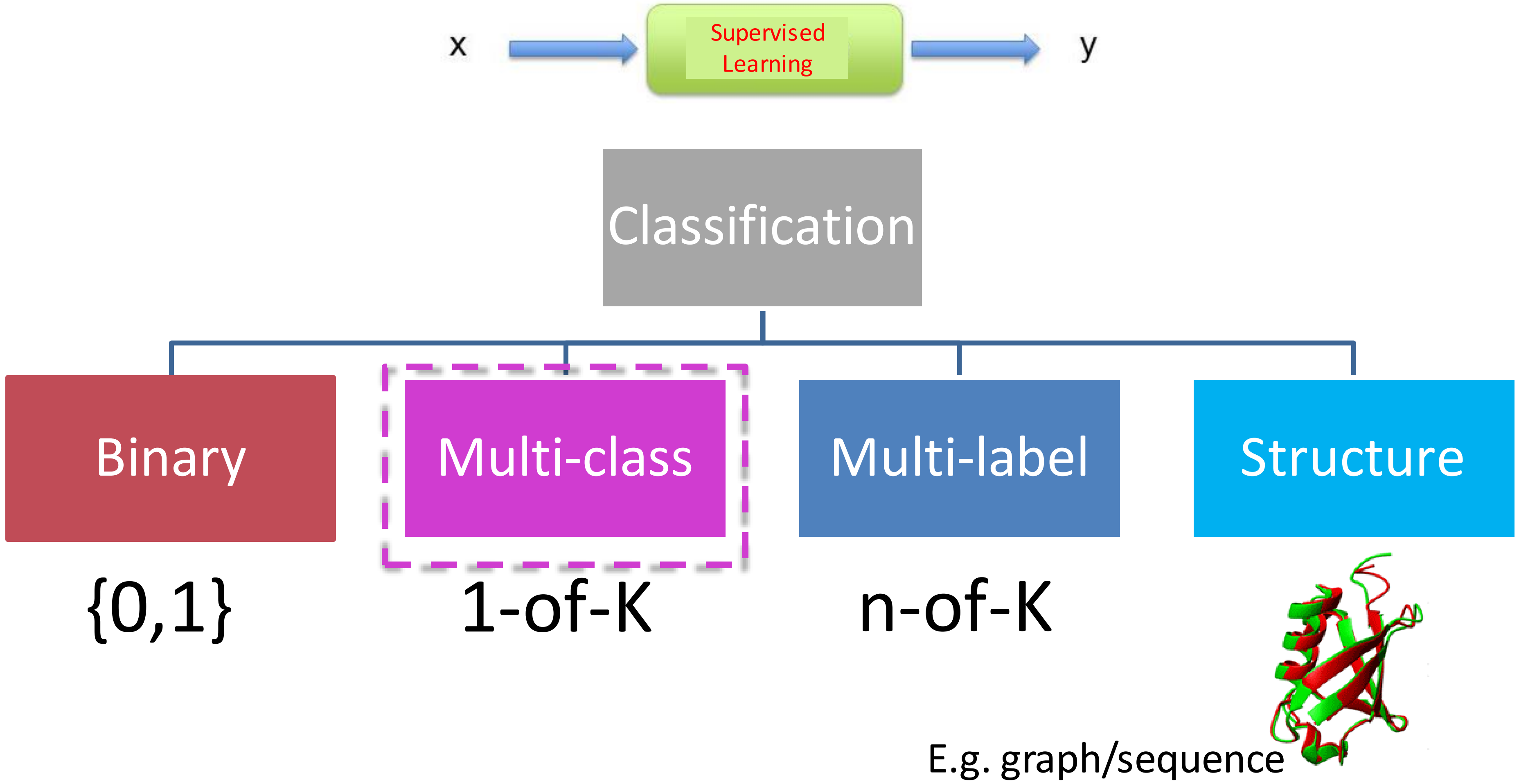
- What to do when one classifier has better precision but worse Recall, while other classifier behaves exactly opposite?
  - F-measure (Information Retrieval)
    - $F_1 = \dfrac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$

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)
    - $F_1 = \dfrac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$

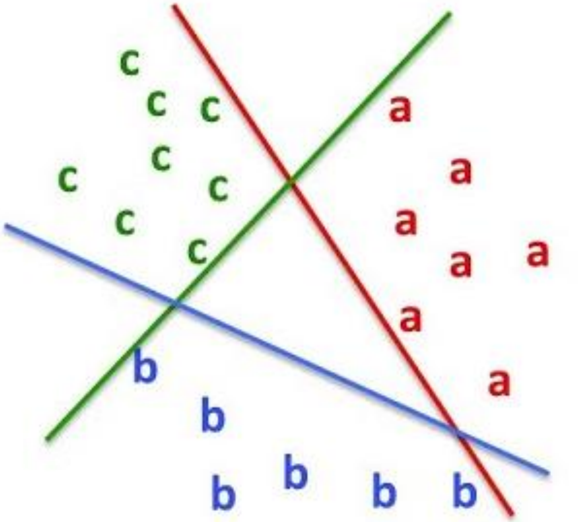
→ **F1 measure punishes extreme values more !**

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


x
Supervised Learning
y
Classification
Binary
Multi-class
Multi-label
Structure
{0,1}
1-of-K
n-of-K
E.g. graph/sequence

How to use 2-class measures for multi-class ?

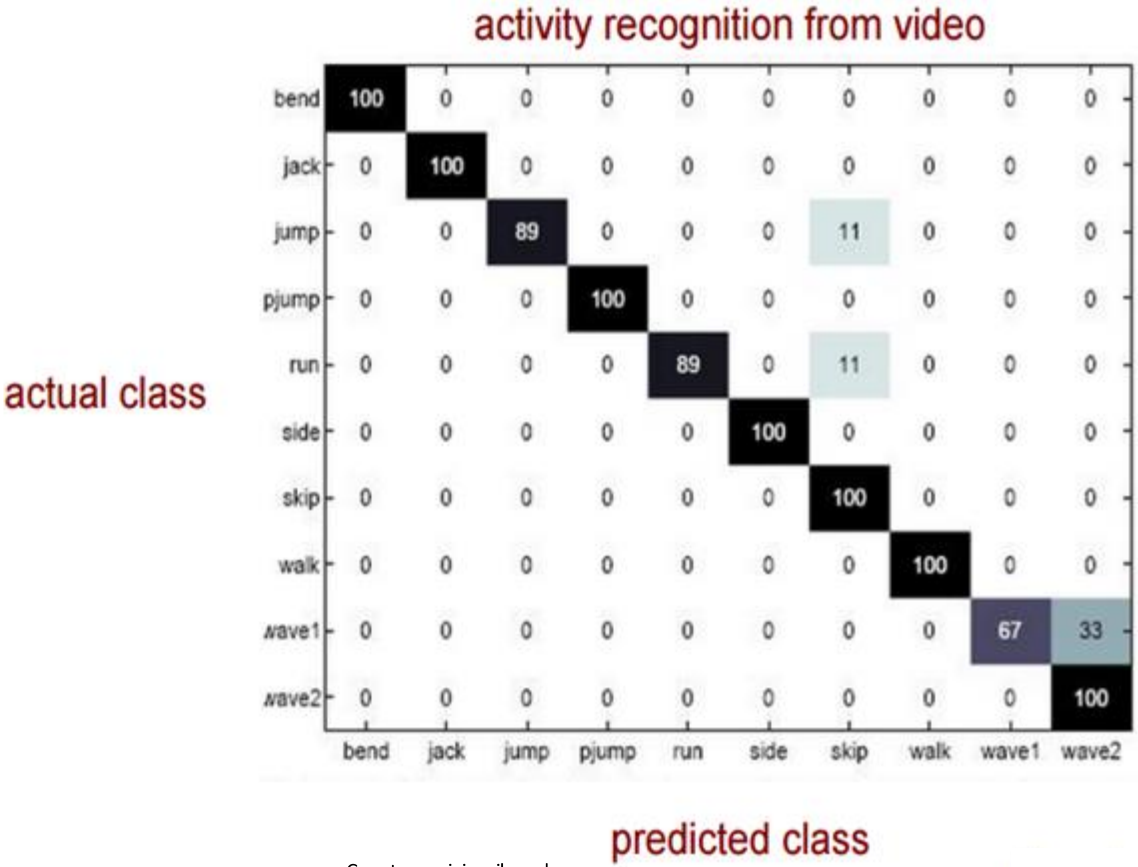
- Convert into 2-class problems !



- Average Precision, Recall etc.

Avg. accuracy may not be very meaningful with imbalanced class label distribution

Multi-class problems - Confusion matrix



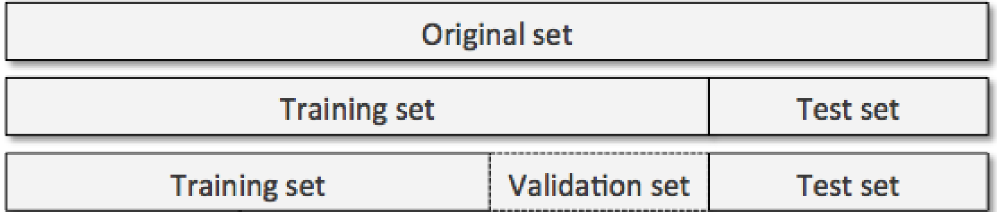
Courtesy: vision.jhu.edu

Multi-class Classification: Measures

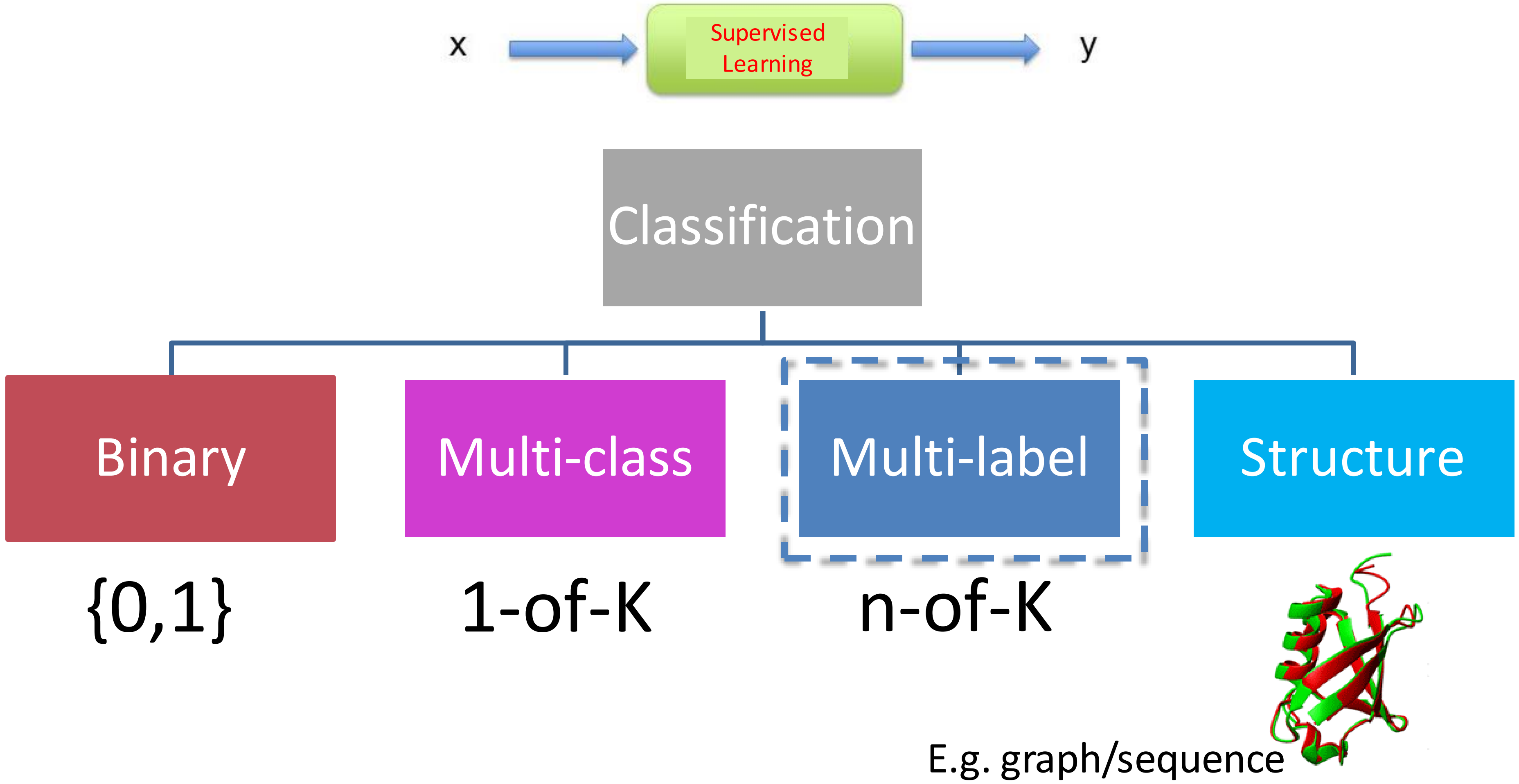
- Mean <measure> +- standard deviation
- Median <measure> +- median absolute deviation

Descriptor	Spectral bands	
	RGB	PCA RGB
Gist	74.14 ± 1.93	77.76 ± 2.62
MSIFT	88.92 ± 1.39	90.97 ± 1.81
MBoW	88.60 ± 1.70	88.31 ± 1.38
cSIFT	88.17 ± 1.17	88.76 ± 1.74
rgSIFT	88.24 ± 1.89	87.71 ± 1.33
BoWV [8]	71.86	N/A
SPMK [12]	74.00	N/A
SPCK++ [8]	76.05	N/A
Dense SIFT [2]	81.67 ± 1.23	N/A

Exam analogy: Did you prepare at least a little ?



- Compute <Performance Measure> (e.g. Accuracy) for TRAINING SET
- Verify it is “decent”


x
Supervised Learning
y
Classification
Binary
Multi-class
Multi-label
Structure
{0,1}
1-of-K
n-of-K
E.g. graph/sequence

Example-based

- n is the number of examples.
- Y_i is the ground truth label assignment of the i^{th} example..
- x_i is the i^{th} example.
- $h(x_i)$ is the predicted labels for the i^{th} example.

Precision = $\frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap h(x_i)|}{|h(x_i)|}$

What fraction of labels are predicted correctly ?

Recall = $\frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap h(x_i)|}{|Y_i|}$

What % of correct labels were predicted ?

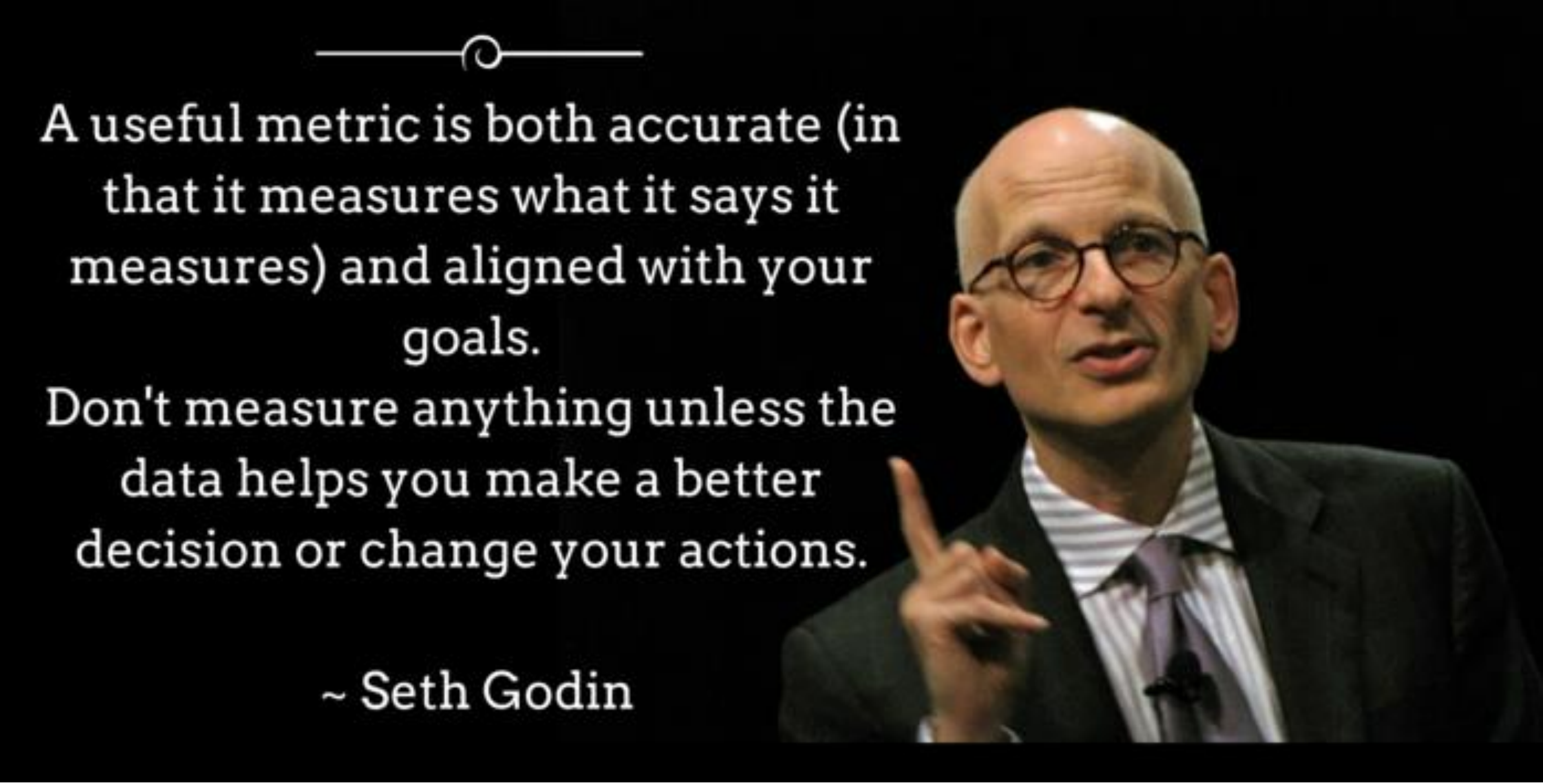
Accuracy = Fraction of samples predicted correctly

Baselines

- 0 cost-to-build classifiers
- Binary
  - Equal # of samples / class → Random Guessing (50% accuracy)
  - Class imbalance
    - → Guess according to class proportion (Accuracy =)
    - 0-Rule: Majority class (Accuracy =) [slightly stronger baseline]

Summary

- Many metrics:
  - Accuracy, TP, FP, Precision, Recall, AP/mAP
  - Class imbalance and decision-cost imbalance must be taken into account
- Confusion Matrix: Important to analyze and refine solution.


A useful metric is both accurate (in that it measures what it says it measures) and aligned with your goals.
Don't measure anything unless the data helps you make a better decision or change your actions.
~ Seth Godin

References and Reading

- Code
  - https://scikit-learn.org/stable/modules/model_evaluation.html#classification-metrics