भारतीय प्रौद्योगिकी संस्थान जम्मू
INDIAN INSTITUTE OF TECHNOLOGY JAMMU
विद्याधनं सर्वधन प्रधानम्

Data Pre-processing and Evaluation metrics

Dr. Shaifu Gupta
shaifu.gupta@iitjammu.ac.in

Contents

- Identification of independent and dependent features
- Nominal features
- Missing values features
- Feature scaling
  - Min-max normalization
  - Standardization
- Evaluation metrics

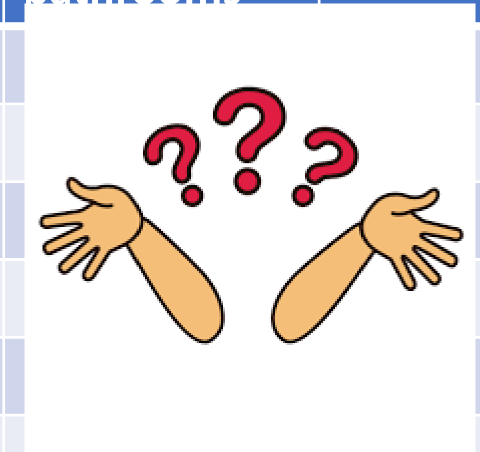
Identification of independent and dependent features

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price

Understand the problem at hand well!

Nominal features

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price
		Small	Yes			
		Medium	No			
		Large	Yes			
		Small	Yes			
		Small	No			
		Medium	No			
		Small	Yes			



Nominal features

Small -> 1
Medium -> 2
Large -> 3

Yes -> 1
No -> 2

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price
		Small -> 1	Yes -> 1			
		Medium -> 2	No -> 2			
		Large -> 3	Yes -> 1			
		Small -> 1	Yes -> 1			
		Small -> 1	No -> 2			
		Medium -> 2	No -> 2			
		Small -> 1	Yes -> 1			

Missing values features

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price
2000		Small	Yes			
5000		Medium	No			
300			Yes			
		Small	Yes			
4000			No			
		Medium	No			
600		Small	Yes			

Missing values features

1. Drop rows or columns
2. Replace by mean of entire feature
3. Replace by mean of consecutive values

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price
2000		Small	Yes			
5000		Medium	No			
300			Yes			
		Small	Yes			
4000			No			
		Medium	No			
600		Small	Yes			

Feature scaling

Area (sq. ft)	No. of bedrooms	Balcony	Terrace garden	Attached bathrooms	Airy kitchen	Price
2000	2	Small	Yes			1000000
5000	3	Medium	No			3000000
300	2		Yes			100000
	3	Small	Yes			2000000
4000	2		No			4000000
		Medium	No			1000000
600	1	Small	Yes			1000000

Different range of features!
But why is it a problem?

Feature scaling - issues

- Example:

$y = w0 + \beta1 \times w1 + \beta2 \times w2 + \ldots + \epsilon$

If β1 is in range [1..100], and β2 is in range [0.1..0.6], so on

What may the system learn? w1 > w2

Is it correct semantically?

Feature scaling techniques

- Min-Max normalization

$$x' = \frac{x - \min}{max - \min}$$

Value lies in range [0-1]

- Standardization

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

Value centered around mean with standard deviation 1
Preferred if data belongs to Gaussian distribution

Evaluation metrics

Classification and Regression

Classification / Regression

- Need to estimate accuracy and performance of classifier / regressor
- Focus on
  - Estimation strategy
  - Metrics for measuring accuracy
  - Metrics for measuring performance

Estimation Strategy

- Using some “training data”, building a classifier based on certain principle (called “learning a classifier”)
- After building a classifier and before using it for classification of unseen instance, we have to validate it using some “validation data”.
- Usually training data, validation data (and test data for sake of experimentation) are outsourced from a large pool of data already available.

Estimation Strategy – Holdout Method

- Basic concept of estimating a prediction.
  - Given a dataset, it is partitioned into three disjoint sets called training set, validation set and test set.
  - Classifier is learned based on the training set and get evaluated with validation set.
  - Proportion of training, validation and testing sets is at the discretion of analyst; typically 60:20:20

Holdout Method - Issue

- Over-presenting a class in one set thus under-presenting it in the other set and vice-versa.

Training set

Test set

It is a problem!
But why?

Estimation Strategy – Random Subsampling

- In this method, Holdout method is repeated ***k*** times, and in each time, **disjoint sets are chosen at random** with predefined sizes.

- Overall estimation is taken as the average of estimations obtained from each iteration.

Estimation Strategy – Cross-Validation

- Main drawback of Random subsampling is, it does not have control over the number of times each tuple is used for training and testing.

- Cross-validation is proposed to overcome this problem.

- There are two variations in the cross-validation method.

  - k-fold cross-validation

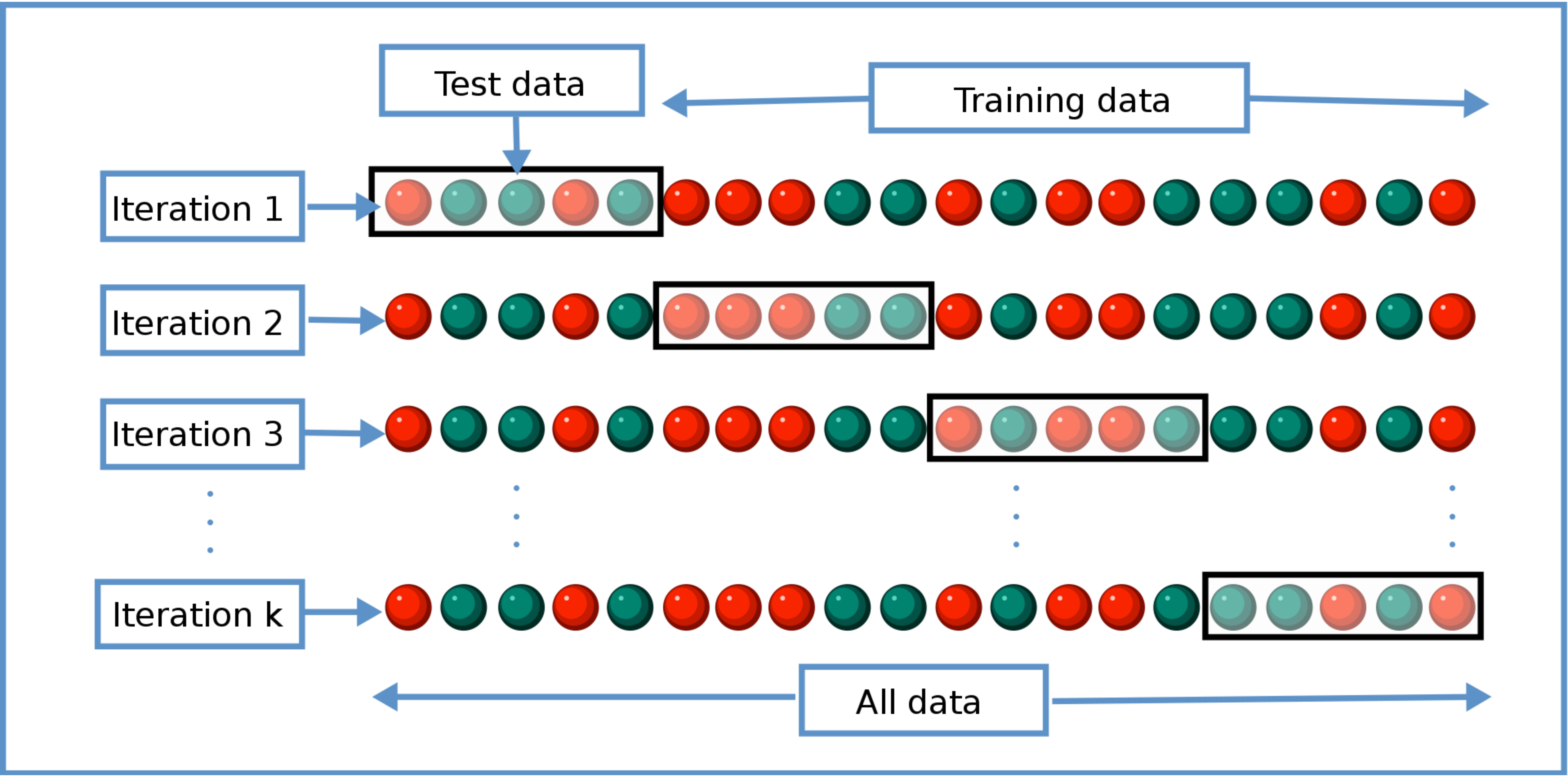
  - *N*-fold cross-validation

k-fold Cross-Validation

- Dataset consisting of *N* tuples is divided into *k* (usually, 5 or 10) equal, mutually exclusive parts or folds $(D_1, D_2, \ldots., D_k)$, and if *N* is not divisible by *k*, then the last part will have fewer tuples than other (*k*-1) parts.

- A series of *k* runs is carried out with this decomposition, and in *i*th iteration D_i is used as validation data and other folds as training data
  - Thus, each tuple is used same number of times for training and once for validation.

- Overall estimate is taken as the average of estimates obtained from each iteration.
- k-classifiers built


Test data
Training data
Iteration 1
Iteration 2
Iteration 3
Iteration k
All data

N-fold Cross-Validation

- *E*xtreme case of *k*-fold cross validation, often known as "Leave-one-out'' cross-validation.
- Here, dataset is divided into as many folds as there are instances; thus, all most each tuple forming a training set, building *N* classifiers.
- Overall estimation is then averaged out of the results of *N* classifiers.

N-fold Cross-Validation : Issue

- Computationally expensive, as here we have to repeat the run *N* times; this is particularly true when data set is large.

- High *variance* in estimates of model's error: single example used for testing in every iteration
  - Worse if outliers are present!

Accuracy Estimation

- There are mainly two things to be measured for a given classifier
  - Accuracy
  - Performance
- **Accuracy estimation**
  - If *N* is number of instances with which a classifier is tested and *p* is number of correctly classified instances, accuracy can be denoted as

$$Accuracy = \frac{p}{N}$$

  - Also, error rate (i.e., misclassification rate) denoted by $\overline{\in}$ is denoted by

$$\overline{\in} = 1 - Accuracy$$

True and Predictive Accuracy

- **True accuracy** of classifier: accuracy when classifier is tested with all possible unseen instances in given classification space.
  - However, number of possible unseen instances is potentially very large (if it is not infinite)
  - For example, classifying a hand-written character
  - Hence, measuring true accuracy beyond the dispute is impractical.

- **Predictive accuracy** of classifier is an accuracy estimation for a given test data (which are mutually exclusive with training data).
  - Predictive accuracy varies with presented test data set

Predictive Accuracy

- Consider a classifier M^D developed with training set D using an algorithm *M*.
- Two predictive accuracies when M^D is estimated with two different training sets T_1 and T_2 are

  $(M^D)_{T1} = 95\%$

  $(M^D)_{T2} = 70\%$

- Further, assume size of T_1 and T_2 are

  $|T_1| = 100$ records

  $|T_2| = 5000$ records.

- Based on the above mentioned estimations, neither estimation is acceptable beyond doubt.

Statistical Estimation using Confidence Level

Experiment 1: When a coin is tossed, there is a probability that a head will occur. A simple experiment is that the coin is tossed many times and both numbers of heads and tails are recorded.

N=10		N=50		N=100		N=250		N=500		N=1000	
H	T	H	T	H	T	H	T	H	T	H	T
3	7	29	21	54	46	135	115	241	259	490	510
0.30	0.70	0.58	0.42	0.54	0.46	0.54	0.46	0.48	0.42	0.49	0.51

Thus, $p \to 0.5$ after a large number of trials in each experiment.

Statistical Estimation using Confidence Level

Experiment 2: To increase the accuracy of the result, we can repeat the experiment several times and take the average of the readings:

$$\bar{x} = \frac{\left(x_1 + x_2 + \ldots x_n\right)}{n}$$

Confidence Levels and Intervals

Experiment 3: Confidence Interval: Range of estimates for a value.

Refers to probability that parameter will fall between a set of values for a certain proportion of times.

Thus, if an estimate is generated with a 95% confidence interval of 9.50 - 10.50, it can be inferred that there is a 95% probability that the true value falls within that range.

Confidence Levels and Intervals

- Use confidence intervals to understand statistical significance of predictions.
  - For example, a researcher selects different samples randomly from the same population and computes a confidence interval for each sample to see how it may represent the true value of the population variable

- Confidence interval displays probability that a parameter will fall between a pair of values around the mean.

Calculating a Confidence Interval

- **Mean :** $\mu = \frac{1+2+3+...+n}{n}$ **say** 240

- **Standard deviation:** $\sigma = \sqrt{\sum \frac{(x_i-\mu)^2}{n}}$ **say** 25

Calculating a Confidence Interval

- **Find the Z value for the selected confidence interval**

80%	1.282
85%	1.440
90%	1.645
95%	1.960
99%	2.576
99.5%	2.807
99.9%	3.291

Calculating a Confidence Interval

$$\mu \pm t\left(\frac{\sigma}{\sqrt{n}}\right)$$

where,
μ = mean
t = chosen Z-value from the table
σ = the standard deviation
n = number of observations

Value lies between lower limit and upper limit

Performance Estimation of a Classifier

- Predictive accuracy works fine, when the classes are balanced
  - That is, every class in the data set are equally important
- In fact, data sets with imbalanced class distributions are quite common in many real life applications
- When the classifier classified a test data set with imbalanced class distributions then, predictive accuracy on its own is not a reliable indicator of a classifier's effectiveness.

Performance Estimation of a Classifier

Example : Effectiveness of Predictive Accuracy

- Given a data set of stock markets, need to classify them as “good” and “worst”. Suppose, in the data set, out of 100 entries, 98 belong to “good” class and only 2 are in “worst” class.
  - With this data set, if classifier’s predictive accuracy is 0.98, may consider it good because of very high value but,
    - There is a high chance that 2 “worst” stock markets may incorrectly be classified as “good”
  - On the other hand, if the predictive accuracy is 0.02, then none of the stock markets may be classified as “good”

Confusion Matrix

- A confusion matrix for a two classes (+, -) is shown below.

	C_1	C_2
C_1	True positive	False negative
C_2	False positive	True negative

True Positive : Number of instances that were positive and correctly classified as positive.
False Negative : Number of instances that were positive and incorrectly classified as negative. It is also known as **Type 2 Error**.
False Positive : Number of instances that were negative (-) and incorrectly classified as (+). This also known as **Type 1 Error**.
True Negative: The number of instances that were negative (-) and correctly classified as (-).

Confusion Matrix

- Having m classes, confusion matrix is a table of size $m \times m$

Class	Good	Worst
Good	6954	46
Worst	412	2588

Performance Evaluation Metrics

True Positive Rate (TPR): fraction of the positive examples predicted correctly by the classifier.

$$TPR = \frac{TP}{P} = \frac{TP}{TP+FN}$$

- This metric is also known as ***Recall***, ***Sensitivity*** or ***Hit rate***.

False Positive Rate (FPR): fraction of negative examples classified as positive class by the classifier.

$$FPR = \frac{FP}{N} = \frac{FP}{FP+TN}$$

- This metric is also known as ***False Alarm Rate***.

Performance Evaluation Metrics

False Negative Rate (FNR): fraction of positive examples classified as a negative class by the classifier.

$$FNR = \frac{FN}{P} = \frac{FN}{TP + FN}$$

True Negative Rate (TNR): fraction of negative examples classified correctly by the classifier

$$TNR = \frac{TN}{N} = \frac{TN}{TN + FP}$$

This metric is also known as ***Specificity***.

Performance Evaluation Metrics

- **Precision** : fraction of the positive examples classified as positive that are really positive

$$Precision = \frac{TP}{TP + FP}$$

- **F_1 Score** :

$$F_1 = \frac{2r.p}{r + p} = \frac{2TP}{2TP + FP + FN}$$

Note

- F_1 represents the harmonic mean between recall and precision
- High value of F_1 score ensures that both Precision and Recall are reasonably high.

Analysis with Performance Metrics

- Based on the various performance metrics, we can characterize a classifier.
- We do it in terms of TPR, FPR, Precision, Recall and Accuracy

Case 1: Perfect Classifier

When every instance is correctly classified, it is called perfect classifier. In this case, $TP = P$, $TN = N$ and Confusion Matrix is

		Predicted Class	
		+	-
Actual class	+	P	0
	-	0	N

Analysis with Performance Metrics

Case 2: Worst Classifier

When every instance is wrongly classified, it is called worst classifier. In this case, $TP = 0$, $TN = 0$ and the Confusion Matrix is

		Predicted Class	
		+	-
Actual class	+	0	P
	-	N	0

Regression

Mean Absolute Error (MAE):

$$\frac{\sum_{i=1}^{N} |y_i - y_i{}'|}{N}$$

Mean Squared Error(MSE):

$$\frac{\sum_{i=1}^{N} (y_i - y_i{}')^2}{N}$$

In addition, a relative error measurement is also known.

In this measure, the error is measured relative to mean value $\tilde{y}$ calculated as mean of y_i (i = 1, 2, …, N) of the training data say D.

Two measures are

Relative Absolute Error (RAE):

$$\frac{\sum_{i=1}^{N} |y_i - y_i{'}|}{\sum_{i=1}^{N} |y_i - \tilde{y}\ |}$$

Relative Squared Error (RSE):

$$\frac{\sum_{i=1}^{N} (y_i - y_i{'})^2}{\sum_{i=1}^{N} (y_i - \tilde{y}\ )^{\ 2}}$$

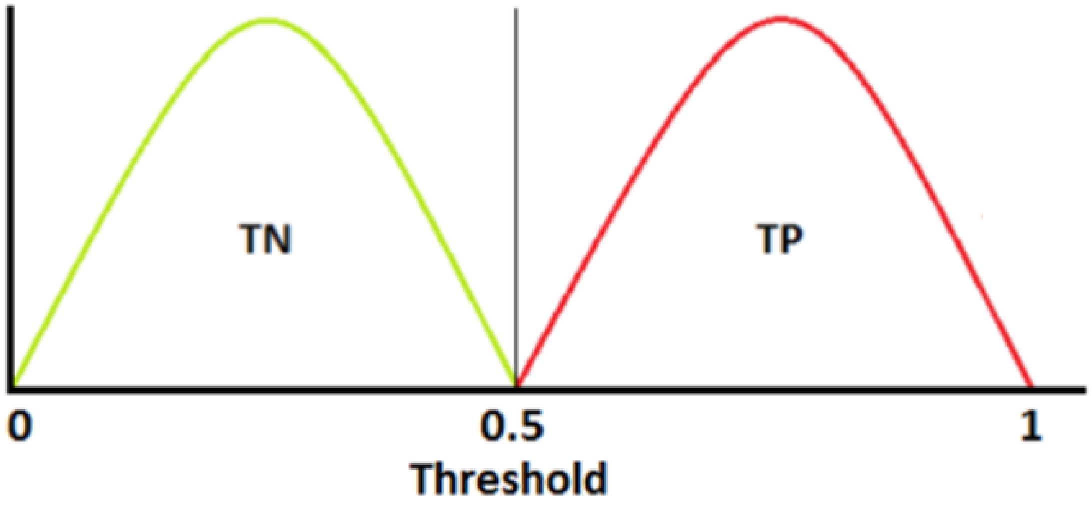
Precision-Recall

$$\text{Precision (p)} = \frac{a}{a+c} = \frac{TP}{TP+FP}$$

$$\text{Recall (r)} = \frac{a}{a+b} = \frac{TP}{TP+FN}$$

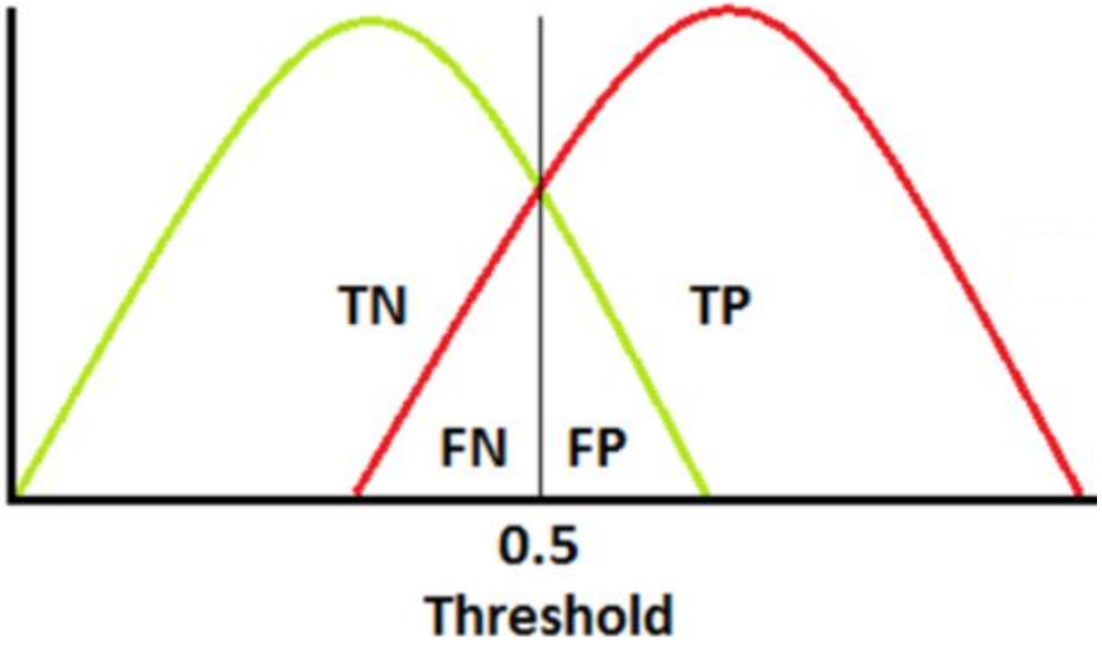
Count	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	a	b
ACTUAL CLASS Class=No	c	d

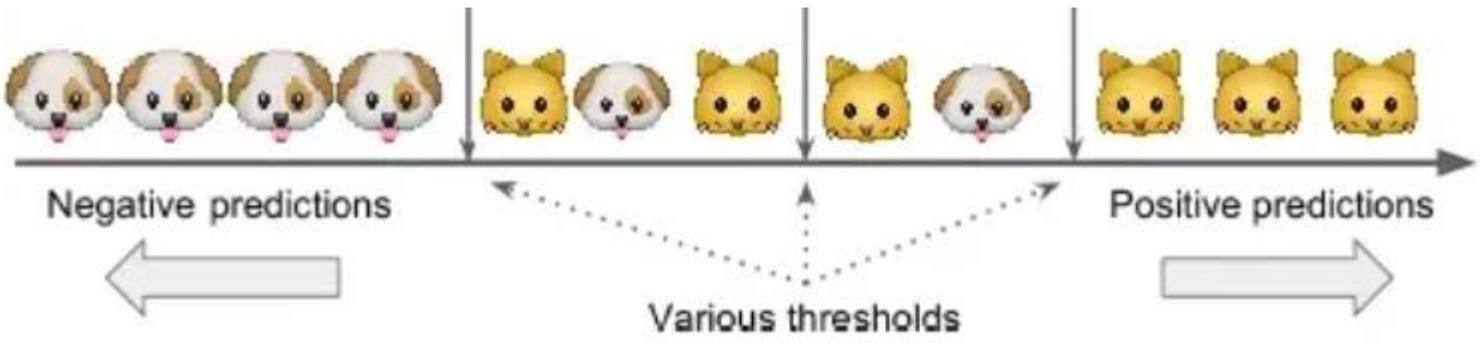
- Precision is biased towards **C(Yes|Yes) & C(Yes|No)**
- Recall is biased towards **C(Yes|Yes) & C(No|Yes)**



Actual

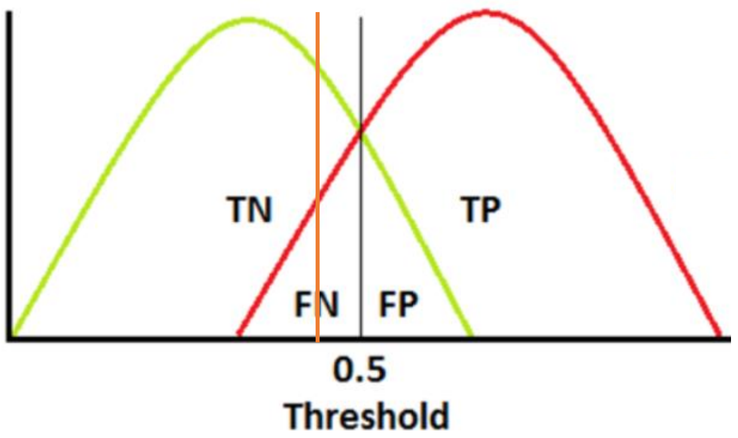
Prediction (TP and TN reduced to give rise to FP and FN)

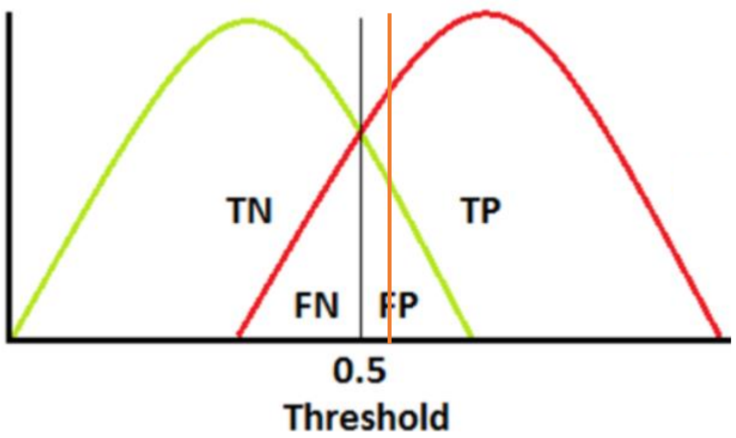




Higher FP, Lower FN -> Low Precision
-> High Recall

Higher FN, Lower FP -> Low Recall
-> High Precision





Practice Question

Previous Grade	Branch	Prediction score	Prediction (33 % cutoff)	Actual Result
7.8	CS	0.40	Pass	Fail
8.9	CS	0.67	Pass	Pass
4.5	CS	0.32	Fail	Pass
5.8	EE	0.56	Pass	Pass
8.9	EE	0.78	Pass	Pass
8.9	ME	0.65	Pass	Pass
2.4	ME	0.31	Fail	Pass
4.4	MT	0.67	Pass	Fail
7.7	MT	0.20	Fail	Pass
2.6	CE	0.80	Pass	Fail

Previous Grade	Branch	Prediction score	Prediction (33 % cutoff)	Actual Result	Accuracy type
7.8	CS	0.40	Pass	Fail	FP 1.
8.9	CS	0.67	Pass	Pass	TP 1.
4.5	CS	0.32	Fail	Pass	FN
5.8	EE	0.56	Pass	Pass	TP 2.
8.9	EE	0.78	Pass	Pass	TP 3.
8.9	ME	0.65	Pass	Pass	TP 4.
2.4	ME	0.31	Fail	Pass	FN
4.4	MT	0.67	Pass	Fail	FP 2.
1.7	MT	0.20	Fail	Fail	TN
7.6	CE	0.80	Pass	Fail	FP 3.

$$\text{Precision (p)} = \frac{4}{4+3} = 0.57$$

$$\text{Recall (r)} = \frac{4}{4+2} = 0.66$$

Increasing the cut-off to 60%

Prediction score	Prediction (60 % cutoff)	Actual Result	Accuracy type
0.40	Fail	Fail	TN 1.
0.67	Pass	Pass	TP 1.
0.32	Fail	Pass	FN
0.56	Fail	Pass	FN
0.78	Pass	Pass	TP 2.
0.65	Pass	Pass	TP 3.
0.31	Fail	Pass	FN
0.67	Pass	Fail	FP 1.
0.20	Fail	Fail	TN
0.80	Pass	Fail	FP 2.

Threshold moved right:
Higher FN: 2 -> 3
Lower FP: 3 -> 2

$$\text{Precision (p)} = \frac{3}{3+2} = 0.6$$

$$\text{Recall (r)} = \frac{3}{3+3} = 0.5$$

Precision: Increased
Recall: Reduced