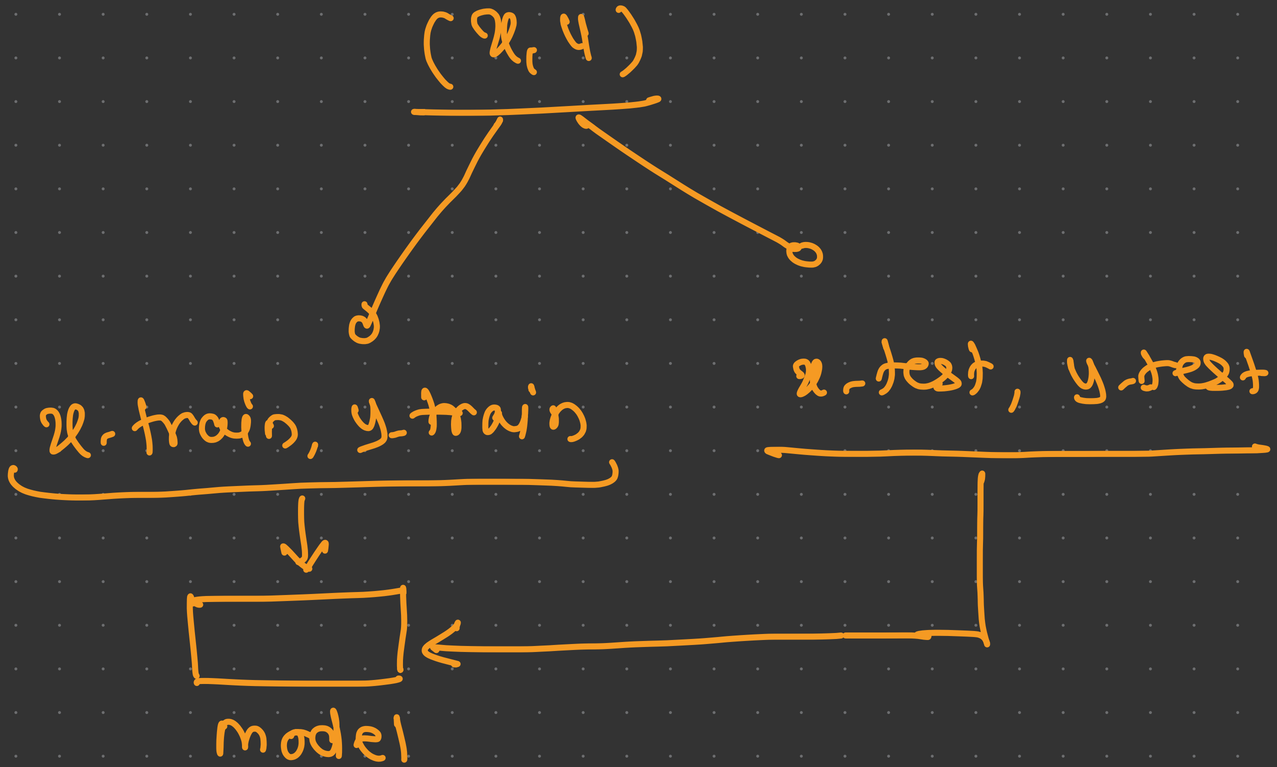
Model Evaluation

Regression

Regression Model Evaluation Metrics

- For evaluation of regression model, following metrics are used
  - MAE
  - MSE
  - RMSE
  - R2
  - Adjusted R2


(x, y)
x_train, y_train
x_test, y_test
model

Mean Absolute Error (MAE)

Error = $(y - \hat{y})$, MAE = $\frac{\sum |y - \hat{y}|}{n}$

- The MAE measures the average magnitude of the errors in a set of forecasts, without considering their direction → absolute value
- It measures accuracy for continuous variables
- The MAE is the average over the verification sample of the absolute values of the differences between forecast and the corresponding observation
- The MAE is a linear score which means that all the individual differences are weighted equally in the average

Error = calculate − observed

Mean Squared Error (MSE)

$\text{Error} = (y - \hat{y})$ $\quad MSE = \frac{\Sigma(y-\hat{y})^2}{n}$

- In statistics, the mean squared error (MSE) or mean squared deviation (MSD) of an estimator (of a procedure for estimating an unobserved quantity) measures the average of the squares of the error
- That is, the average squared difference between the estimated values and the actual value
- MSE is a risk function, corresponding to the expected value of the squared error loss
- The fact that MSE is almost always strictly positive (and not zero) is because of randomness or because the estimator does not account for information that could produce a more accurate estimate
- The MSE is a measure of the quality of an estimator
- As it is derived from the square of Euclidean distance, it is always a positive value with the error decreasing as the error approaches zero

Root Mean Squared Error (RMSE)

- RMSE is the most popular evaluation metric used in regression problems
- It follows an assumption that error are unbiased and follow a normal distribution
- Here are the key points to consider on RMSE:
  - The power of 'square root' empowers this metric to show large number deviations
  - The 'squared' nature of this metric helps to deliver more robust results which prevents cancelling the positive and negative error values
- It avoids the use of absolute error values which is highly undesirable in mathematical calculations
- When we have more samples, reconstructing the error distribution using RMSE is considered to be more reliable
- RMSE is highly affected by outlier values. Hence, make sure you've removed outliers from your data set prior to using this metric.
- As compared to mean absolute error, RMSE gives higher weightage and punishes large errors

R-Squared (R2)

- We learned that when the RMSE decreases, the model's performance will improve
- But these values alone are not intuitive
- When we talk about the RMSE metrics, we do not have a benchmark to compare
- This is where we can use R-Squared metric

using mean ($\bar{y}$)

- In other words how good our regression model as compared to a very simple model that just predicts the mean value of target from the train set as predictions

[0 , +1]

Adjusted R-Squared

- A model performing equal to baseline would give R-Squared as 0
- Better the model, higher the r2 value
- The best model with all correct predictions would give R-Squared as 1
- However, on adding new features to the model, the R-Squared value either increases or remains the same

  ↳ independent
- R-Squared does not penalize for adding features that add no value to the model
- So an improved version over the R-Squared is the adjusted R-Squared

$$\bar{R}^2 = 1-\left(1-R^2\right)\left[\frac{n-1}{n-\left(k+1\right)}\right]$$

- k: number of features
- n: number of samples

Model Evaluation
Classification

Classification Model Evaluation Metrics

- For evaluation of classification model, following metrics are used
  - Confusion Matrix
  - F1 Score
  - AuC-Roc

precision

recall

Confusion Matrix

- A confusion matrix is an N X N matrix, where N is the number of classes being predicted
- The confusion matrix provides more insight into not only the performance of a predictive model, but also which classes are being predicted correctly, which incorrectly, and what type of errors are being made

		Predicted Class		
		Positive	**Negative**	
Actual Class	**Positive**	True Positive (TP)	False Negative (FN) **Type II Error**	**Sensitivity** $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) **Type I Error**	True Negative (TN)	**Specificity** $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

x	y
23	1
25	0
28	1
30	0

$\rightarrow$ model $\rightarrow$ prediction

x	y (observed)	$\hat{y}$ (calculated)	
23	1 →	1	✓
25	0 →	1	✗
28	1 →	0	✗
30	0 →	0	✓

$$\text{accuracy} = \frac{\text{correct answers}}{\text{all answers}} = \frac{2}{4} = 0.5 = 50\%$$

predicted

observed / actual	1	0
1	(0,0) 1 TP	(0,1) 1 FN
0	(1,0) 1 FP	(1,1) 1 TN 1

correct = [0][0] + [1][1]

incorrect = [0][1] + [1][0]

total = [0][0] + [0][1] + [1][0] + [1][1]

	observed	predicted	
True positive (TP)	1	1	✓
True Negative (TN)	0	0	✓
False Negative (FN) (Type II error)	1	0	✗
False Positive (FP) (type I error)	0	1	✗

TP vs FP vs TN vs FN

positive = image is a cat

1	2	3	4	5	6	7	8	9
N	P	N	P	N	P	N	P	P

P	P	P	P	N	N	N	P	P
Cat	Cat	Cat	Cat	No Cat	No Cat	No Cat	Cat	Cat
FP	TP	FP	TP	TN	FN	TN	TP	TP

TP = 4

TN = 2

FN = 1

FP = 2

prediction

Observed	1	0
1	(0,0) TP 4	(0,1) FN 1
0	(1,0) FP 2	(1,1) TN 2

6

total = TP + TN + FN + FP = 9

	0	1
0	TN	FP
1	FN	TP

Accuracy

	TP		TP	TN		TN	TP	TP
								
Cat	Cat	Cat	Cat	No Cat	No Cat	No Cat	Cat	Cat
✗	✓	✗	✓	✓	✗	✓	✓	✓

$$\text{accuracy} = \frac{\#\ \text{correct}}{\#\ \text{total}} = \frac{6}{9}$$

$$= \frac{TP + TN}{TP + TN + FN + FP}$$

How many we got right ?

Precision

- Precision talks about how precise/accurate your model is out of those predicted positive, how many of them are actual positive
- Precision is a good measure to determine, when the costs of False Positive is high
- For instance, in email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam). The email user might lose important emails if the precision is not high for the spam detection model.

$$\text{Precision} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}}$$

$$= \frac{\textit{True Positive}}{\textit{Total Predicted Positive}}$$

Precision

①  Cat

②  Cat

③  Cat

④  Cat

①  No Cat TN

②  No Cat FN

③  No Cat TN

⑤  Cat

⑥  Cat

$$+ve\ \text{precision} = \frac{TP}{\text{total } +ve} = \frac{TP}{TP+FP} = \frac{4}{6}$$

$$-ve\ \text{precision} = \frac{TN}{\text{total } -ve} = \frac{TN}{TN+FN} = \frac{2}{3}$$

Out of all Cat predictions how many we got right ?

Recall

- Recall actually calculates how many of the Actual Positives our model capture through labelling it as Positive (True Positive)
- Applying the same understanding, we know that Recall shall be the model metric we use to select our best model when there is a high cost associated with False Negative
- For instance, in fraud detection or sick patient detection, if a fraudulent transaction (Actual Positive) is predicted as non-fraudulent (Predicted Negative), the consequence can be very bad for the bank
- Similarly, in sick patient detection, if a sick patient (Actual Positive) goes through the test and predicted as not sick (Predicted Negative), the cost associated with False Negative will be extremely high if the sickness is contagious

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$

Recall

Cat | Cat | Cat | Cat | No Cat | No Cat | No Cat | Cat | Cat

$$\text{Recall} = \frac{TP}{TP + FN}$$

Out of all Cat truth how many we got right ?

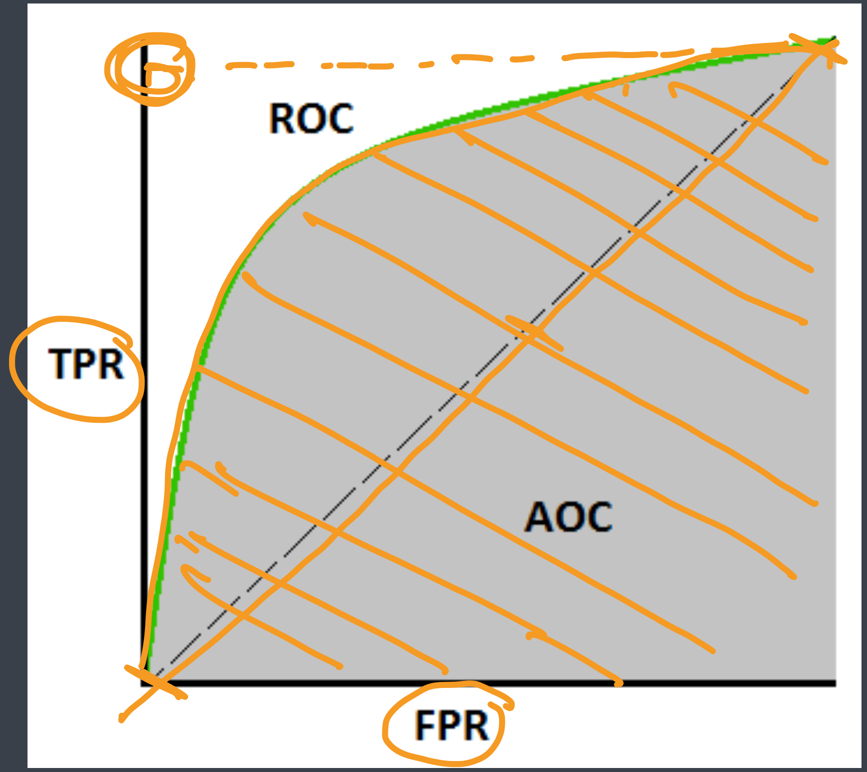
F1 Score

- The F1 score is the harmonic mean of the precision and recall
- The highest possible value of an F-score is 1.0, indicating perfect precision and recall, and the lowest possible value is 0, if either the precision or the recall is zero
- The F1 score is also known as the Sørensen–Dice coefficient or Dice similarity coefficient (DSC)

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

Receiver Operating Characteristic (ROC)

- ROC curve is a metric that assesses the model ability to distinguish between binary classes
- It is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings
- The TPR is also known as sensitivity, recall or probability of detection in machine learning
- The FPR is also known as the probability of false alarm and can be calculated as 1– specificity
- Points above the diagonal line represent good classification (better than random)
- The model performance improves if it becomes skewed towards the upper left corner



Receiver Operating Characteristic (ROC)

TPR (True Positive Rate) / Recall /Sensitivity

$$\text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

Image 3

Specificity

$$\text{Specificity} = \frac{TN}{TN + FP}$$

Image 4

FPR

$$\text{FPR} = 1 - \text{Specificity}$$

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