Preprocessed Data

- **ML Algorithms**
  - Accuracy
  - Precision
  - Recall
  - F1-score
- **Deep Learning Algorithms (LSTM)**
  - RMSE
  - MAE
  - Epochs
  - Batch size
- **Exploratory Data Analysis**
  - Crime rate
  - Crime count
  - Crime type
  - Crime regions
- **ARIMA MODEL**
  - Future crime trend
  - High crime density areas.

Accuracy:

=> Accuracy is a metric for classification model that measures the no. of predictions that are correct as a percentage of the total no. of predictions that are made.

$$\text{Accuracy} = \frac{\text{no. of correct predictions}}{\text{no. of total predictions}}$$

Imbalanced Data :-

99% of website visitors don't ~~buy~~ buy and that only 1% of visitors buy something in that website, so we are building a classification model to predict which website visitors are buyers and which are just lookers.

Now imagine a model that doesn't work very well. It predicts that 100% of your visitors are just lookers and that 0% of your visitors are buyers. It is clearly a very wrong & useless model.

⇒ So Accuracy is not a good metric to use when ~~you~~ we have class imbalance.

Solving imbalance data through metrics:-

Way to solve class imbalance is to use better accuracy metrics like the F1-score, which take into account not only the no. of prediction errors that your model make, but that also look at the type of errors that are made.

Precision :-

⇒ First part of the F1-score.

⇒ It can also be used as an individual ML. metric.

$$\text{Precision} = \frac{\text{no. of true positives}}{\text{no. of true positives} + \text{no. of false positives}}$$

Within everything that has been predicted as a positive, precision counts the percentage that is correct:

* A not precise model:-

may find a lot of positives, but its selection method is noisy; it also wrongly detects many positive that aren't actually positives.

* A precise model:-

It is very "pure". maybe it does not find all the positives, but the ones that the model does class as positive are very likely to be correct.

Recall:-

⇒ second part of F1-score.

$$\text{Recall} = \frac{\text{no. of true positives}}{\text{no. of true positives} + \text{no. of false negatives.}}$$

Within everything that actually is positive, how many did the model succeed to find:

* High recall:

A model with high recall succeed well in finding ~~the~~ all the positive cases in the data, even though they may also

Scanned by TapScanner

wrongly identify some negative cases as positive cases.

* Low recall:

A model wid low recall is not able to find all (or a large part) of the positive cases in the data.

Precision - Recall Trade off:

⇒ Ideally, we would want both: a model that identifies only positive cases It is so called precision - Recall Trade - Off.

⇒ The precision - Recall Trade - off represents the fact that in many cases, we can tweak a model to increase precision at a cost of lower recall, or on the other hand increase recall at the cost of lower precision.

F1 score: combining precision & Recall:

⇒ Precision & Recall are 2 building blocks of F1 score.

⇒ The goal of F1 - score is to combine the precision & Recall metrics into a single metric. At the same time, the F1 score has been designed to work well on imbalanced data.

⇒ F1 score is defined as harmonic mean of precision & recall.

↓

* reciprocal of arithmetic mean.

* useful when computing average rate.

⇒ In F1-score, we compute the average of Precision and recall. They are both rates, which makes it a logical choice to use the Harmonic mean.

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

F1-score is an average of precision and Recall, it means that the F1-score gives equal weight to precision & Recall.

* High F1-score:

A model will obtain a high F1 score, if both precision & Recall are high.

* Low F1-score:

A model will obtain a low F1-score, if both Precision & Recall are low.

* Medium F1-score:

A model will obtain a medium F1-score, if one of Precision & Recall is low & the other is high.

Accuracy Vs precision & Recall:-

⇒ Simplest classification metric is accuracy.

⇒ It simply measures the percentage of correct predictions that a ml model has made.

⇒ But accuracy is a bad metric in the case of imbalanced data, because it cannot distinguish between specific types of errors. (false positive & false negative)

⇒ Precision & Recall are performance metrics that are more suitable when having imbalanced data because they allow taking into account the type of errors (false +ve & false -ve) that our model makes.

F1-score Vs Precision & Recall:-

⇒ F1-score combines precision & Recall into a single metric.

⇒ In many situation it is much more convenient to have only one performance metric rather than multiple.

⇒ F1-score is used when we need to compare 2 or more ml algorithms for the same data. We opt for the algorithm whose f1 score is higher.

Eg:

truth	dog	car	dog	dog	dog	mbl	home	dog	dog	Bird
Prediction	dog	dog	dog	No dog	dog	No dog	dog	No dog	dog	dog
+ve Prediction	dog ✓	dog X	dog ✓		dog ✓		dog X		dog ✓	dog X

True positive = 4   False positive = 3

(True → reality, positive → class)

No dog Prediction	No dog X	No dog ✓	No dog X

True Negative = 1   False Negative = 2

Truth	dog	car	dog	dog	dog	mbl	home	dog	dog	bird
Prediction	dog	dog	dog	No dog	dog	No dog	dog	No dog	dog	dog
	✓	X	✓	X	✓	✓	X	X	✓	X

Scanned by TapScanner

How may we got right → 5

∴ Accuracy = 5/10 ⇒ 0.5

Positive Prediction:

$$\text{precision} = \frac{\text{no. of true positives}}{\text{no. of true positive} + \text{no. of false positives}}$$

$= 4/7 = 0.57$

Recall:

Total dog truth samples $\}$ = 6 True positive = 4.

$$\text{Recall} = \frac{TP}{TP + FN} = 4/6 = 0.67$$

* For precision think about predictions as our base.

* For recall think about truth as our base

Negative prediction

precision = 1/3 = 0.33

Recall = 1/4 = 0.25

F1 - Score:

For dog class:

$$F1 = 2 * \frac{(0.57 * 0.67)}{(0.57 + 0.67)}$$

$= 0.615967$

For No dog class

$$F1 = 2 * \frac{(0.33 * 0.25)}{(0.33 + 0.25)}$$

$= 0.2844827$