Data Mining for Business

Predictive Performance and Classifier Performance

Shipra Maurya

Department of Management Studies

IIT (ISM) Dhanbad

Email: shipra@iitism.ac.in

Model Evaluation

- It is the process of testing the performance of the fitted model
- **A good model is the one which is generalizable on the future data**
- Future data may not be available at the time of the development of the model. Hence the data at hand (historical data) has to be partitioned into three i.e. **training set, validation set and holdout set**
- Most used dataset split percentages:
  - 60:20:20
  - 80:10:10
  - 70:15:15

Data Partitioning Methods

- Random split
- Temporal split
- Stratified split

Model Evaluation Measures

Classifier performance (categorical target variable)

1. Confusion Matrix
2. Precision-Recall Curve
3. Receiver Operating Characteristics (ROC)
4. LogLoss
5. Rank-Ordering
6. Lift curve
7. Cross-validation

Predictive performance (continuous target variable)

1. MAPE
2. SMAPE
3. RMSE
4. MAE
5. R^2
6. Adjusted R^2
7. Rank-ordering
8. Cross-validation

Model Evaluation – Confusion Matrix

	1 (Predicted)	0 (Predicted)
1 (Actual)	True Positive	False Negative Type 2 Error
0 (Actual)	False Positive Type 1 Error	True Negative

- Confusion matrix shows the summary of prediction results for a classification problem
- Threshold determination
- Following metrics can be derived from a confusion matrix:

1. Accuracy
2. Estimated misclassification rate
3. Recall / True Positive Rate (TPR)/Sensitivity
4. False Positive Rate (FPR)
5. True Negative Rate (TNR)/Specificity
6. False Negative Rate (FNR)
7. Precision
8. F-Score

Confusion Matrix Parameters (1/3)

	1 (Predicted)	0 (Predicted)
1 (Actual)	True Positive	False Negative Type 2 Error
0 (Actual)	False Positive Type 1 Error	True Negative

Accuracy: It determines the overall predicted accuracy of the model

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

Recall: indicates how many positive values, out of all the positive values, have been correctly predicted. Also known as **Recall or Sensitivity**

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

Estimated Misclassification Rate: indicates overall error rate

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

Confusion Matrix Parameters (2/3)

	1 (Predicted)	0 (Predicted)
1 (Actual)	True Positive	False Negative Type 2 Error
0 (Actual)	False Positive Type 1 Error	True Negative

FPR: indicates how many negative values, out of all the negative values, have been incorrectly predicted

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

TNR: indicates how many negative values, out of all the negative values, have been correctly predicted. It is also known as **Specificity**

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

Confusion Matrix Parameters (3/3)

	1 (Predicted)	0 (Predicted)
1 (Actual)	True Positive	False Negative Type 2 Error
0 (Actual)	False Positive Type 1 Error	True Negative

FNR: indicates how many positive values, out of all the positive values, have been incorrectly predicted

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

Precision: indicates how many values, out of all the predicted positive values, are actually positive

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

F-score: It is the harmonic mean of precision and recall. It lies between 0 and 1. Higher the value, better the model

$$\text{F-score} = 2 \ast \left(\frac{\text{Precision} \ast \text{Recall}}{\text{Precision}+\text{Recall}}\right)$$

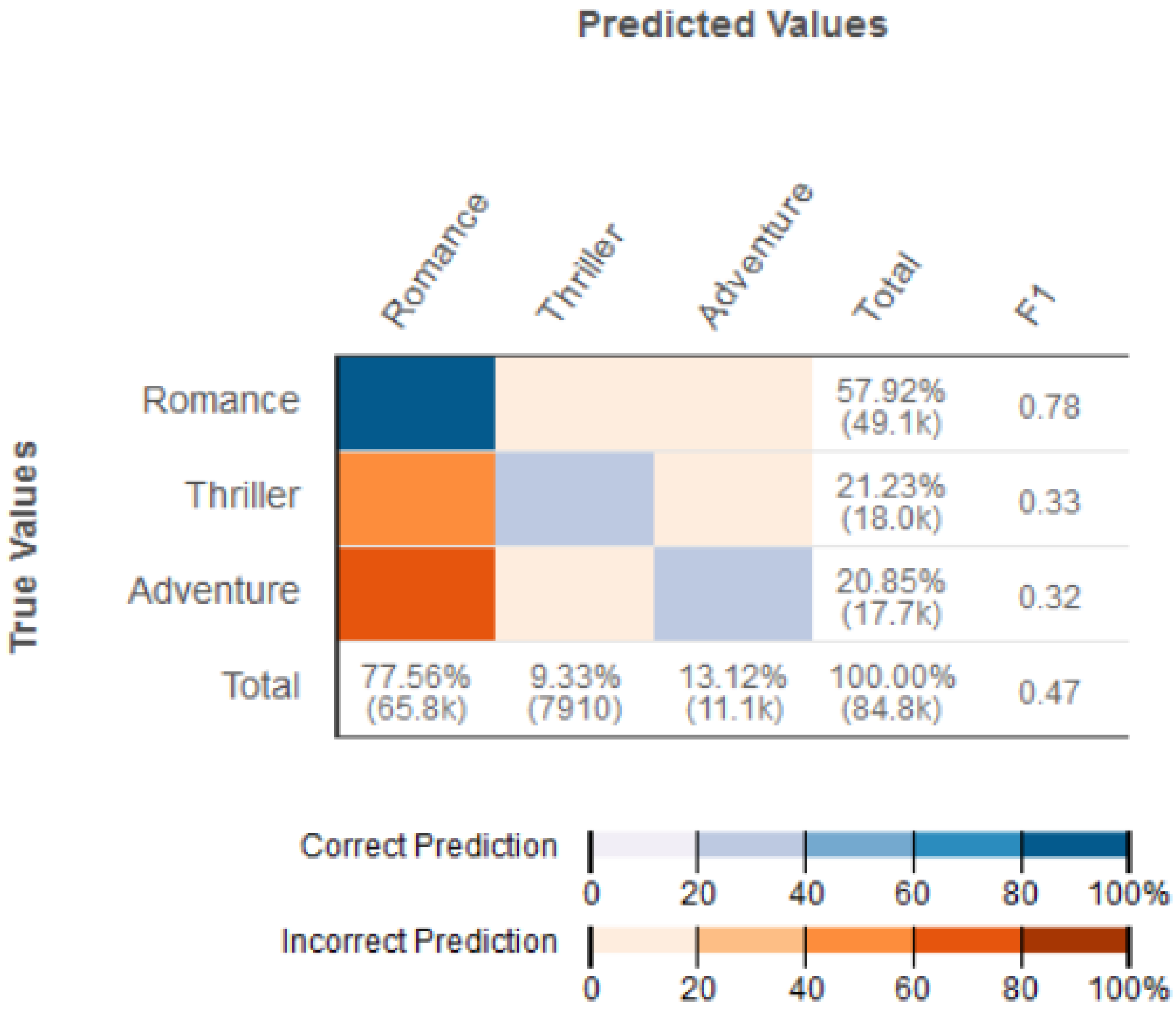
Confusion Matrix – Quick Exercise

		Predicted	
		Fraudulent Transaction (1)	Non-fraudulent Transaction (0)
Actual	Fraudulent Transaction (1)	TP - 45	FN - 20
	Non-fraudulent Transaction (0)	FP - 5	TN - 30

Calculate following with the help of given confusion matrix:

1. Accuracy
2. Recall/True Positive Rate (TPR)/Sensitivity
3. False Positive Rate (FPR)
4. True Negative Rate (TNR)/Specificity
5. False Negative Rate (FNR)
6. Precision
7. F-Score

Confusion matrix for Multi-class classification problem



- Micro F1 Score
- Macro F1 Score
- Weighted F1 Score

CLASSIFICATION REPORT:

	precision	recall	f1-score	support
Class 1	1.00	0.94	0.97	16
Class 2	0.85	0.81	0.83	21
Class 3	0.50	0.62	0.56	8
accuracy			0.82	45
macro avg	0.78	0.79	0.78	45
weighted avg	0.84	0.82	0.83	45

Image source - Internet

Cross-validation

- Cross-validation is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data.
- Used to detect overfitting, i.e., failing to generalize a pattern

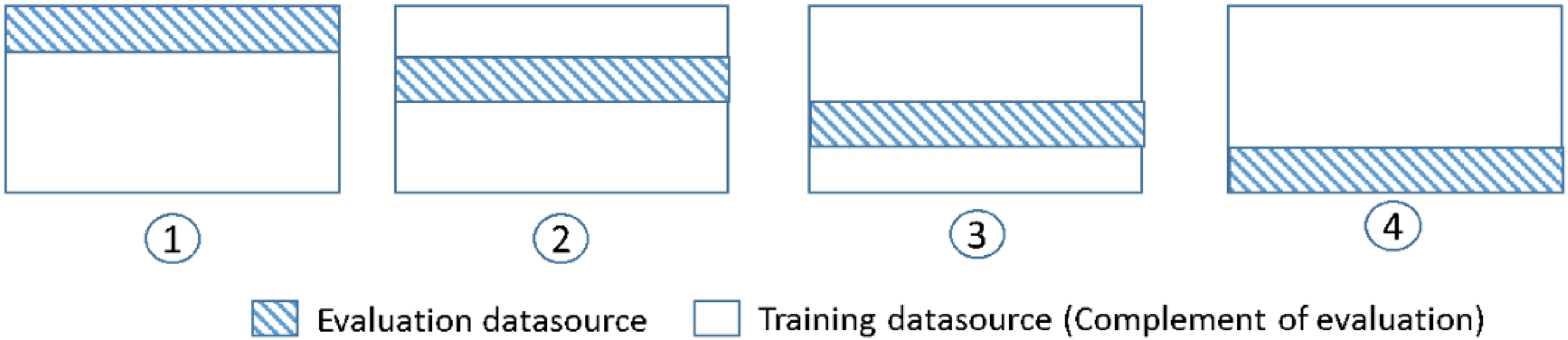
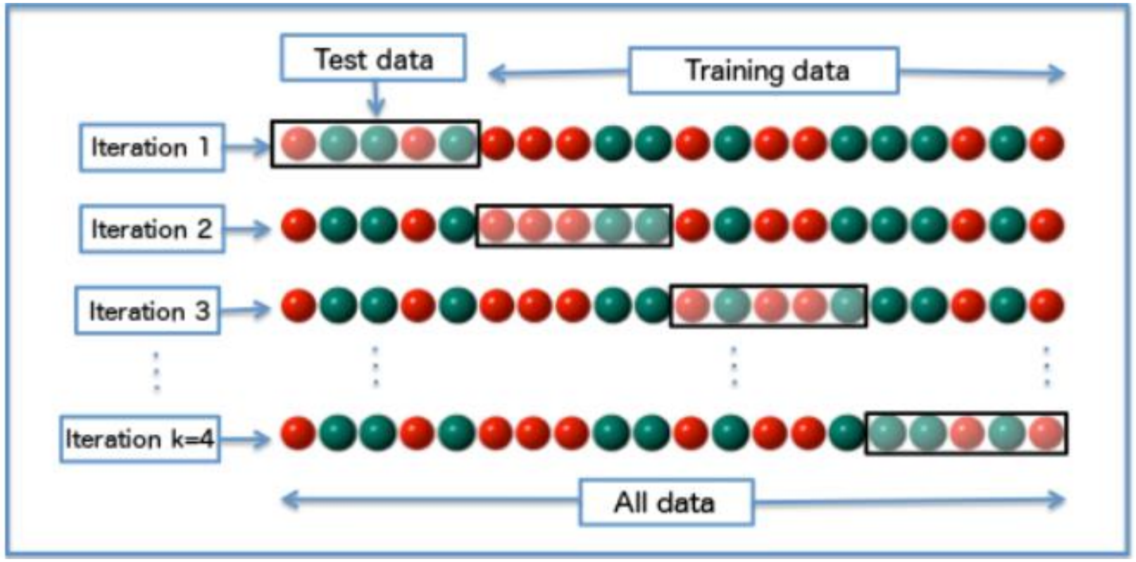


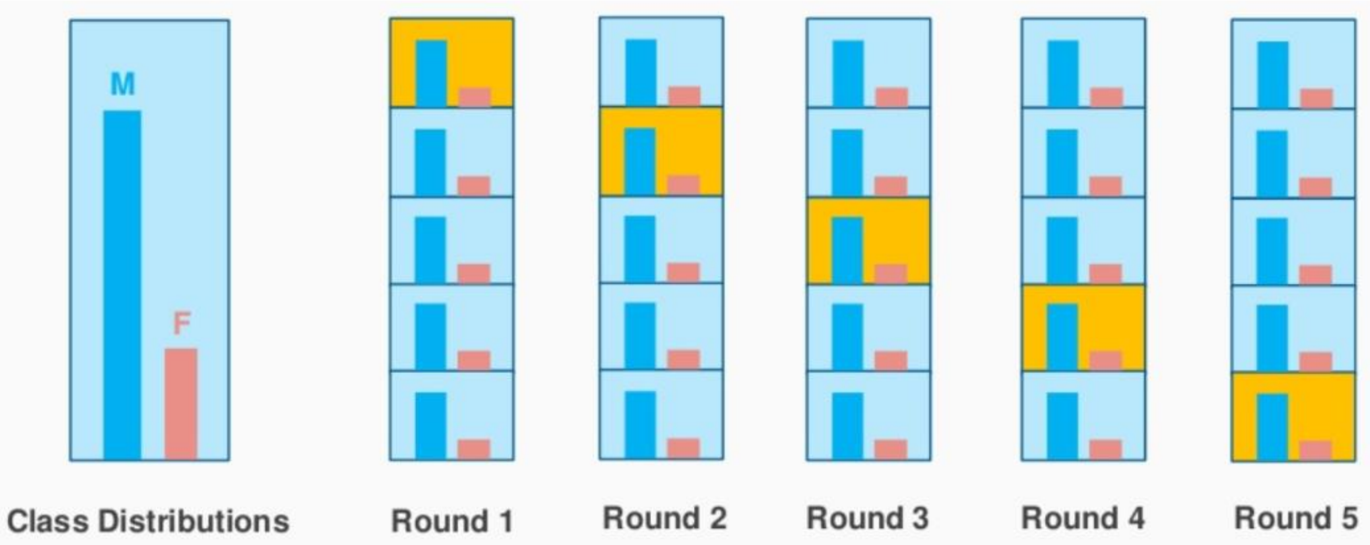
Image source - Internet

Approaches to Cross-validation

- Validation set
- k-fold cross-validation
- Stratified k-fold cross-validation



k-fold cross-validation



Stratified k-fold cross-validation

Image source - Internet

Rank Ordering

Decile		Default=1	Default=0	Default Rate
Rank_group	# of Customers	Defaulting customers	Non defaulting customers	Default Rate
1	1000	325	675	32.5%
2	1000	295	705	29.5%
3	1000	263	737	26.3%
4	1000	210	790	21.0%
5	1000	194	806	19.4%
6	1000	157	843	15.7%
7	1000	126	874	12.6%
8	1000	69	931	6.9%
9	1000	34	966	3.4%
10	1000	16	984	1.6%
Total	**10000**	**1689**	**8311**	**16.9%**

Note : The probability scores are sorted from highest to lowest. The top decile has the highest probability scores

Decile		Default=1	Default=0	Default Rate
Rank_group	# of Customers	Defaulting customers	Non defaulting customers	Default Rate
1	1000	325	675	32.5%
2	1000	295	705	29.5%
3	1000	263	737	26.3%
4	1000	270	730	27.0%
5	1000	194	806	19.4%
6	1000	157	843	15.7%
7	1000	180	820	18.0%
8	1000	69	931	6.9%
9	1000	34	966	3.4%
10	1000	16	984	1.6%
Total	**10000**	**1803**	**8197**	**18.0%**

Image Source : Internet

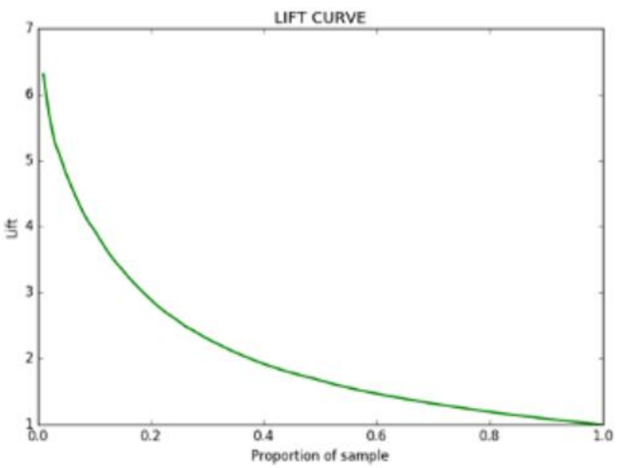
Lift Curve

Decile		Default=1	Default=0		
Rank_group	# of Customers	Defaulting customers	Non defaulting customers	Default Rate	Lift
1	1000	325	675	32.5%	1.92
2	1000	295	705	29.5%	1.75
3	1000	263	737	26.3%	1.56
4	1000	210	790	21.0%	1.24
5	1000	194	806	19.4%	1.15
6	1000	157	843	15.7%	0.93
7	1000	126	874	12.6%	0.75
8	1000	69	931	6.9%	0.41
9	1000	34	966	3.4%	0.20
10	1000	16	984	1.6%	0.09
Total	**10000**	**1689**	**8311**	**16.9%**	

Note : The probability scores are sorted from highest to lowest. The top decile has the highest probability scores

$$Lift = \frac{Predicted\ Rate}{Average\ Rate}$$

Curve 1



Curve 2

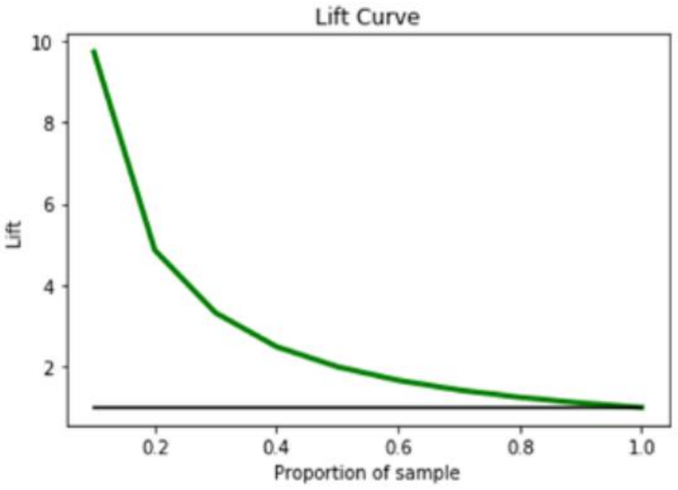


Image Source : Internet

Comparison of Two Lift curves

Precision Recall Curve (1/2)

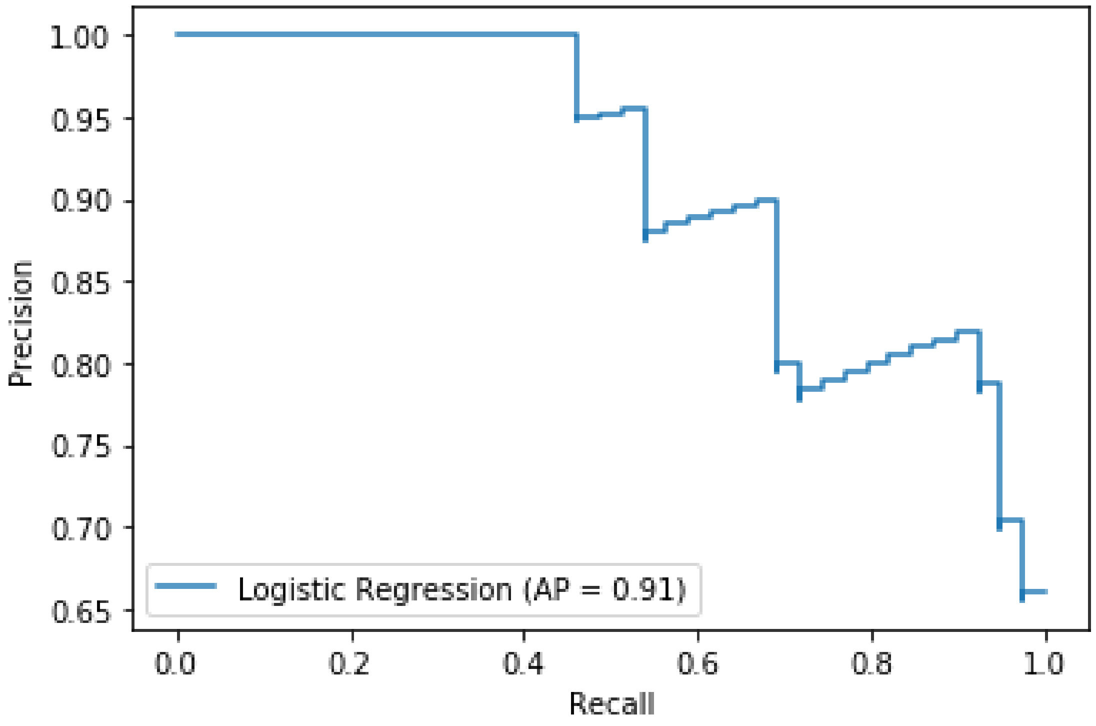
- Precision-recall curves provide a graphical representation of a classifier's performance across many thresholds, rather than a single value

- Generally, classifier predicts 1 if the predicted probability is greater than or equal to 0.5 and predicts 0 if the predicted probability is less than 0.5. Here threshold value = 0.5

- But when we deal with some sensitive situations, we would want to be more sure about False positives and False negatives. Hence, we can increase the threshold or cut-off point from 0.5 to 0.8 or to 0.9.

- **A precision-recall curve helps to visualize how the choice of threshold affects classifier performance, and can even help us select the best threshold for a specific problem**

Precision Recall Curve (2/2)



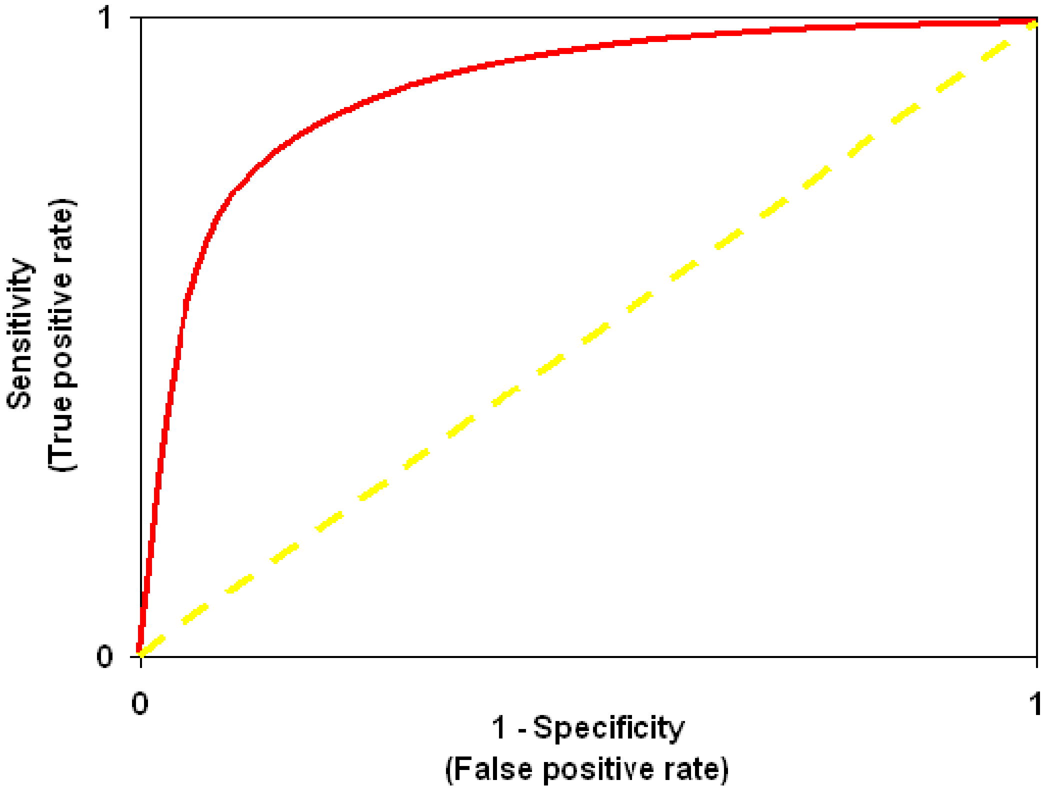
$$AP = \sum_n (R_n - R_{n-1}) P_n$$

- When the precision and recall both are high, that is an indication that the algorithm is doing very well
- AP – Average Precision - weighted mean of precision achieved at each threshold, with the increase in recall from the previous threshold used as the weight
- Higher AP value indicates better classifier performance

Image Source : Internet

ROC Curve

- ROC - Receiver Operating Characteristics Curve
- ROC determines the accuracy of a classification model at a user defined threshold value
- It determines the model's accuracy using Area Under Curve (AUC)
- ROC is plotted between True Positive Rate (Y axis) and False Positive Rate (X Axis)



- The yellow line represents the ROC curve at 0.5 threshold
- The objective is to push the red curve (in the chart) toward 1 (left corner – y-axis) and maximize the area under curve
- Higher the area, better the model

Image Source : Internet

Log Loss

- Log Loss measures the inaccuracy of predicted probabilities
- Log-loss increases when predicted probabilities diverges away from the actual label
- A perfect model would have a log loss of 0

Log-loss= -(ylog(p) + (1 - y)log(1 - p))

Where p is the predicted value of y

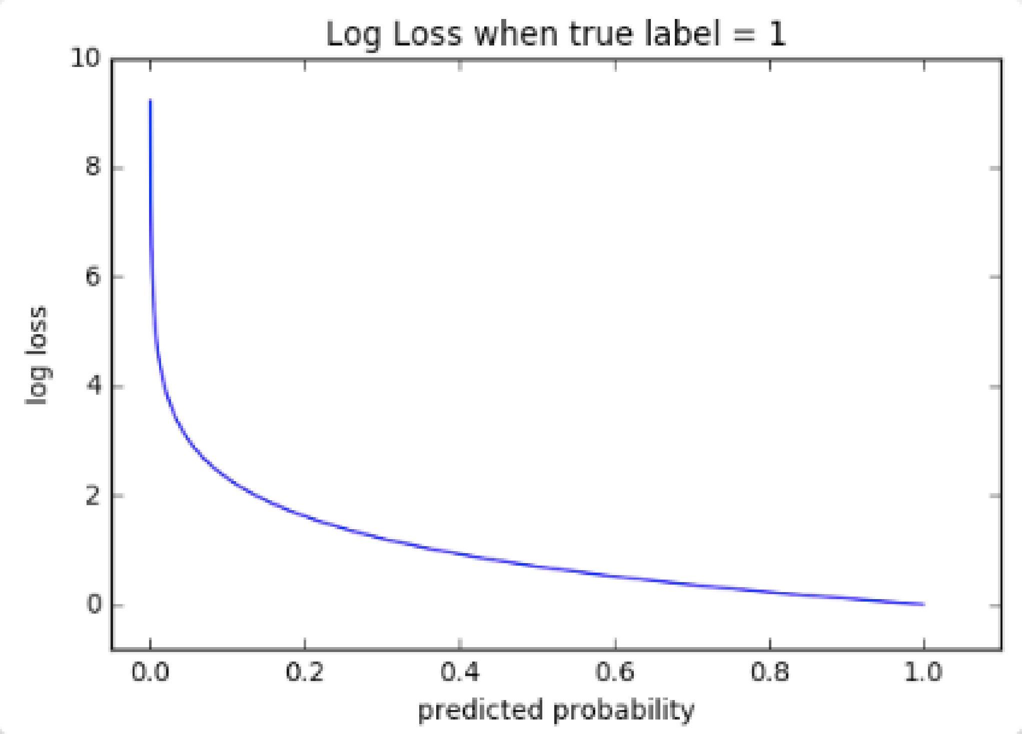


Image Source : Internet

Predictive Performance Measure (1/2)

- MAPE

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|A_i - F_i|}{A_i} * 100$$

- Symmetric MAPE

$$SMAPE = \frac{100}{n}\sum_{i=1}^{n}\frac{|Forecast_i - Actual_i|}{(|Actual_i| + |Forecast_i|)/2}$$

- RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

Predictive Performance Measure (2/2)

- MAE

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

- R^2

$$r^2 = 1 - \frac{\sum \hat{u}_i^2}{\sum (Y_i - \bar{Y})^2} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

- Adjusted R^2

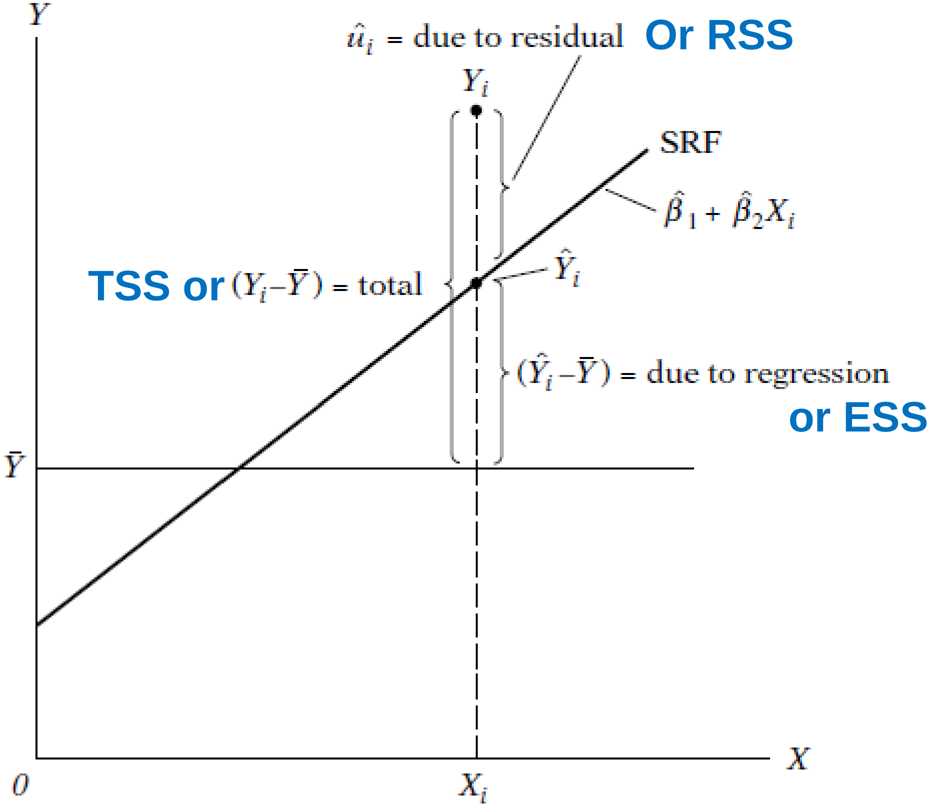
$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(N-1)}{N-p-1}$$

Where

R^2 Sample R-Squared

N Total Sample Size

p Number of independent variable



TSS – Total Sum of Squares

ESS – Explained Sum of Squares

RSS – Residual Sum of Squares

Image source - Internet

How to improve the Model Accuracy?

- **Collect more data** – increase the size of the training data

- **Feature Engineering** – Add more features which contribute to explain the target variable and generate new features from the existing features

- **Model parameter tuning** - Consider alternate values for the training parameters used by your learning algorithm

- Check for target leakage

- Model Calibration

Thank you!