Logistic Regression - Day 2

Agenda

Logistic Regression - Day 2

In this session, you will explore the given topics:

1. Performance Measures: Logistic Regression
2. Python Demo

Measuring Model Performance

It's easy to think about performance metrics for a regression model; the notion of accuracy in a regression setting is straightforward.

- How can we talk about the notion of accuracy in a binary classification scenario?
- What output will a logistic regression model produce?
- How can the likelihood of an event be used to classify it?

Measuring Model Performance

Age	Good_Bad	Prediction
23	1	0.90
24	1	0.75
25	1	0.60
26	0	0.45
27	0	0.40
28	0	0.30
29	1	0.82

Threshold=0.5

Age	Good_Bad	Prediction	Pred_Label
23	1	0.90	1
24	1	0.75	1
25	1	0.60	1
26	0	0.45	0
27	0	0.40	0
28	0	0.30	0
29	1	0.82	1

Accuracy = 100%

Accuracy = # of Correctly predicted labels/N

Measuring Model Performance

Age	Good_Bad	Prediction
23	1	0.90
24	1	0.75
25	1	0.60
26	0	0.45
27	0	0.40
28	0	0.30
29	1	0.82

Threshold=0.5

Age	Good_Bad	Prediction	Pred_Label
23	1	0.90	1
24	1	0.75	1
25	1	0.48	0
26	0	0.52	1
27	0	0.53	1
28	0	0.30	0
29	1	0.82	1

	Actual_Class 1	Actual_Class 0
Predicted_Class 1	3	2
Predicted_Class 0	1	1

Measuring Model Performance

	Actual_Class 1	Actual_Class 0
Predicted_Class 1	True Positive	False Positive
Predicted_Class 0	False Negative	True Negative

Precision = tp/(tp+fp)

Recall = tp/(tp+fn)

Measuring Model Performance

- What is the peculiarity of the previous approach?
- What happens if p > 0.6 is used for classification?
- Will Precision-Recall change?

Measuring Model Performance

Age	Good_Bad	Prediction
23	1	0.90
24	1	0.75
25	1	0.60
26	0	0.45
27	0	0.40
28	0	0.30
29	1	0.82

Threshold=0.5

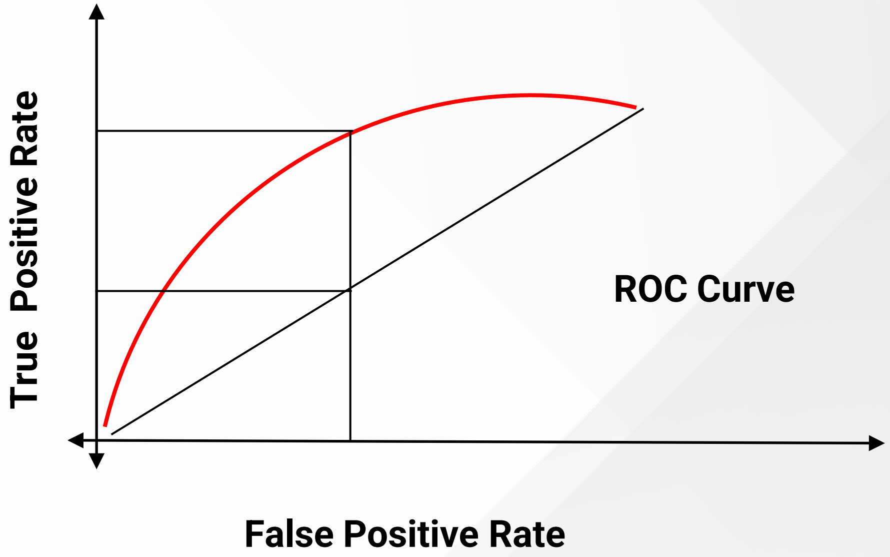
Age	Good_Bad	Prediction	Pred_Label
23	1	0.90	1
24	1	0.75	1
25	1	0.48	0
26	0	0.52	1
27	0	0.53	1
28	0	0.30	0
29	1	0.82	1

	Actual_Class 1	Actual_Class 0
Predicted_Class 1	4	2
Predicted_Class 0	0	1

Measuring Model Performance

	Actual_Class 1	Actual_Class 0
Predicted_Class 1	True Positive	False Positive
Predicted_Class 0	False Negative	True Negative

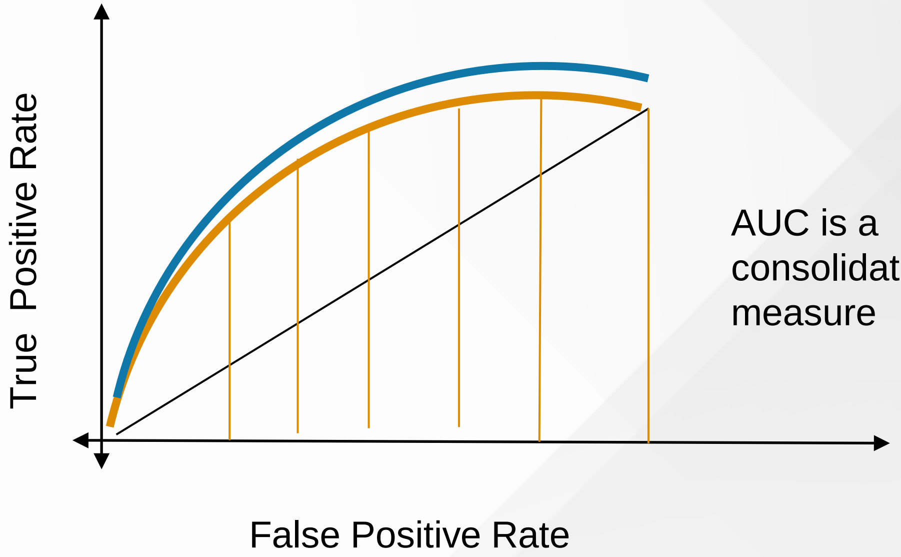
Cutoff	TPR=TP/N	FPR=FP/N
P1	TPR1	FPR1
P2	TPR2	FPR2
..	..	..
Pn	TPRn	FPRn



Measuring Model Performance

	Actual_Class 1	Actual_Class 0
Predicted_Class 1	True Positive	False Positive
Predicted_Class 0	False Negative	True Negative

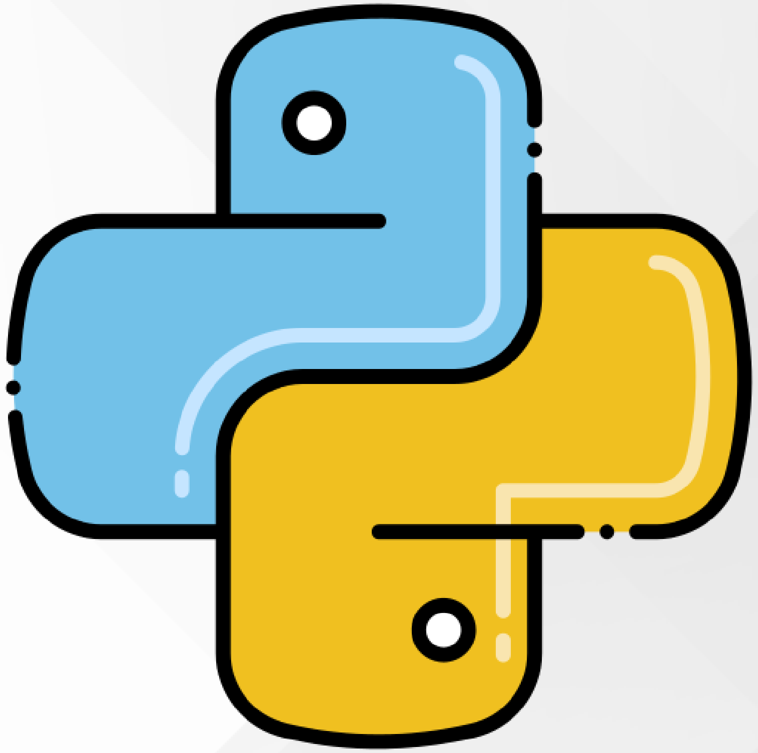
Cutoff	TPR=TP/N	FPR=FP/N
P1	TPR1	FPR1
P2	TPR2	FPR2
..	..	..
Pn	TPRn	FPRn



AUC is a consolidated measure

Measuring Performance: Python Demo

02.ipynb: Measuring Model Performance



Python Demo: Building a Model End-to-End

03.ipynb

Thank You!