Last session (2015-10-29)

- Big Data Analysis Process
  - Model types
- Hands-on – Project 3

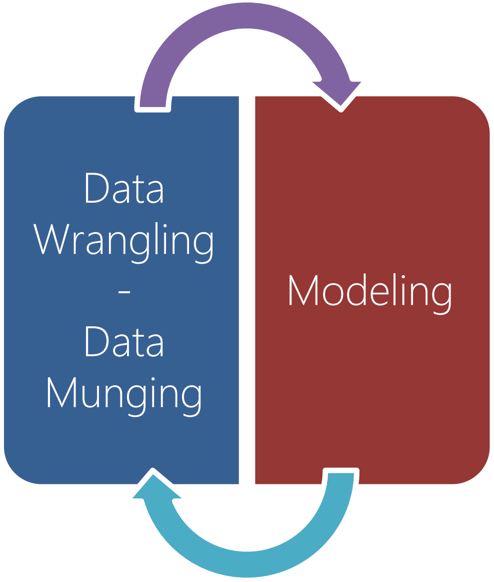
Today's session

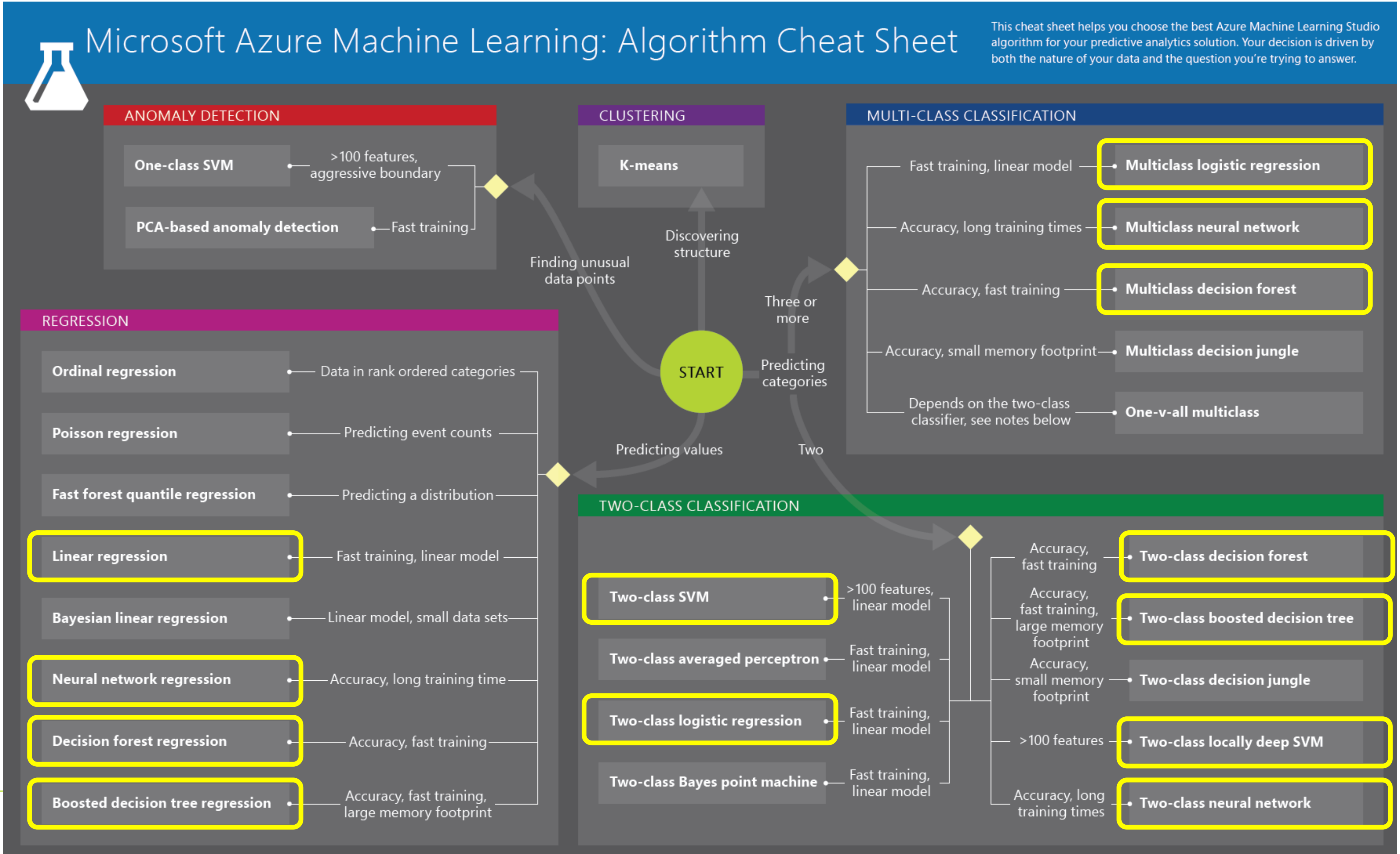
- Big Data Analysis Process
  - Model evaluation
- Hands-on

© www.esmartsystems.com

Big Data Analysis Process – Main Steps

- Data access
- Data pre-processing / cleaning
- Data transformation / manipulation
- Feature selection
- Feature extraction
- Feature engineering
- Model choice and training
- Model evaluation and tuning
- Model deployment




Microsoft Azure Machine Learning: Algorithm Cheat Sheet
This cheat sheet helps you choose the best Azure Machine Learning Studio algorithm for your predictive analytics solution. Your decision is driven by both the nature of your data and the question you're trying to answer.
ANOMALY DETECTION
One-class SVM
>100 features, aggressive boundary
PCA-based anomaly detection
Fast training
CLUSTERING
K-means
Discovering structure
Finding unusual data points
MULTI-CLASS CLASSIFICATION
Fast training, linear model
Multiclass logistic regression
Accuracy, long training times
Multiclass neural network
Accuracy, fast training
Multiclass decision forest
Accuracy, small memory footprint
Multiclass decision jungle
Depends on the two-class classifier, see notes below
One-v-all multiclass
Three or more
Predicting categories
START
REGRESSION
Ordinal regression
Data in rank ordered categories
Poisson regression
Predicting event counts
Fast forest quantile regression
Predicting a distribution
Linear regression
Fast training, linear model
Bayesian linear regression
Linear model, small data sets
Neural network regression
Accuracy, long training time
Decision forest regression
Accuracy, fast training
Boosted decision tree regression
Accuracy, fast training, large memory footprint
Predicting values
Two
TWO-CLASS CLASSIFICATION
Two-class SVM
>100 features, linear model
Two-class averaged perceptron
Fast training, linear model
Two-class logistic regression
Fast training, linear model
Two-class Bayes point machine
Fast training, linear model
Accuracy, fast training
Two-class decision forest
Accuracy, fast training, large memory footprint
Two-class boosted decision tree
Accuracy, small memory footprint
Two-class decision jungle
>100 features
Two-class locally deep SVM
Accuracy, long training times
Two-class neural network

Model Evaluation

Different performance measures for regression and classification

Regression

- MAE
- RMSE
- RAE
- RSE
- R^2

Classification

- Confusion Matrix
- Accuracy
- Precision
- Recall
- F-score
- ROC Curve
- Precision/Recall Curve

© www.esmartsystems.com

Regression Model Evaluation

Regression

- MAE
- RMSE
- RAE
- RSE
- R^2

Metrics

Mean Absolute Error	0.072338
Root Mean Squared Error	0.135749
Relative Absolute Error	0.176651
Relative Squared Error	0.085524
Coefficient of Determination	0.914476

© www.esmartsystems.com

Regression Model Evaluation

MAE – Mean Absolute Error

- Average of the absolute difference between predictions and true values

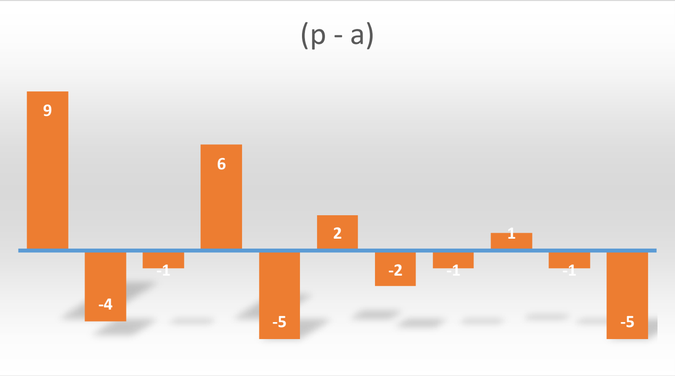
- Takes the absolute value so that positive and negative errors are treated equally
- If we just take the average difference between predictions and true values we will measure only the bias component of the error (which could be zero even if the model makes big errors if these are equally distributed between positive and negative errors)
- Scale dependent -> cannot be used to compare performance on different variables

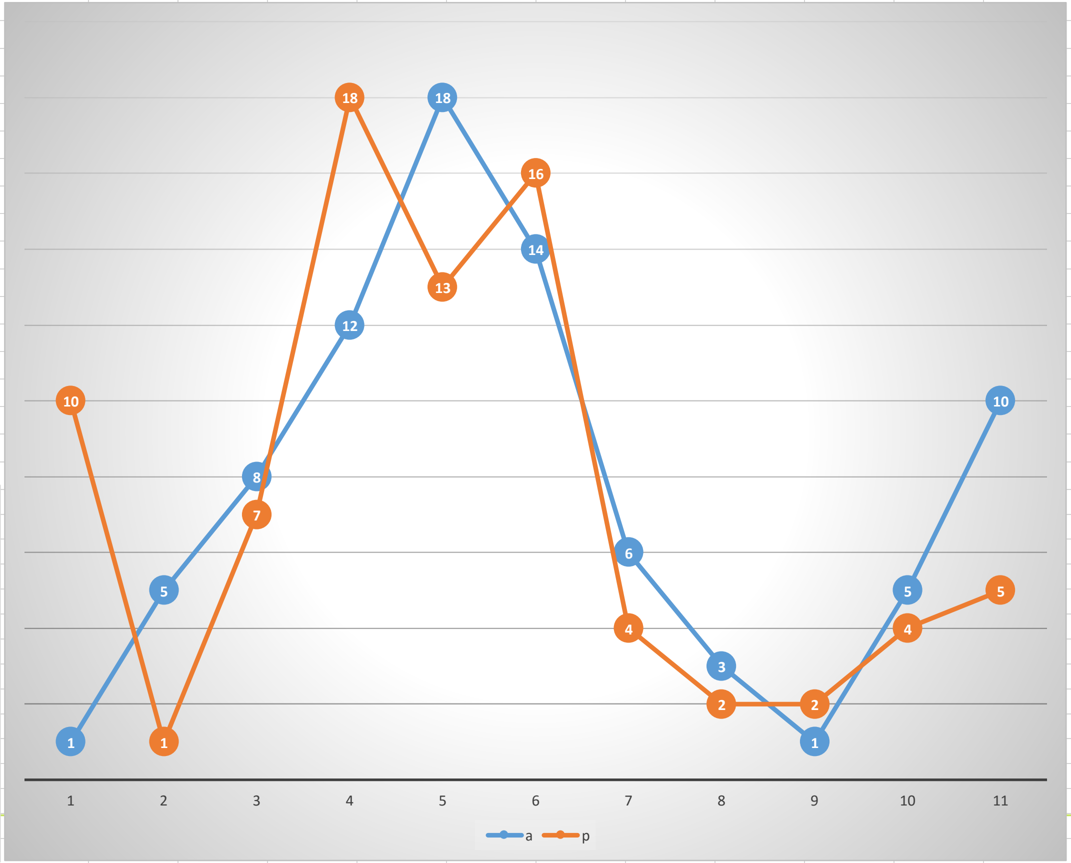
$$MAE = \frac{\sum_{i=1}^{n} |p_i - a_i|}{n}$$

Regression Model Evaluation - MAE

a	p	(p - a)	|p-a|
1	10	9	9
5	1	-4	4
8	7	-1	1
12	18	6	6
18	13	-5	5
14	16	2	2
6	4	-2	2
3	2	-1	1
1	2	1	1
5	4	-1	1
10	5	-5	5

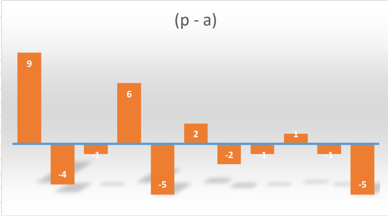
7,55	7,45	-0,09	3,36
Mean	Mean	ME	MAE

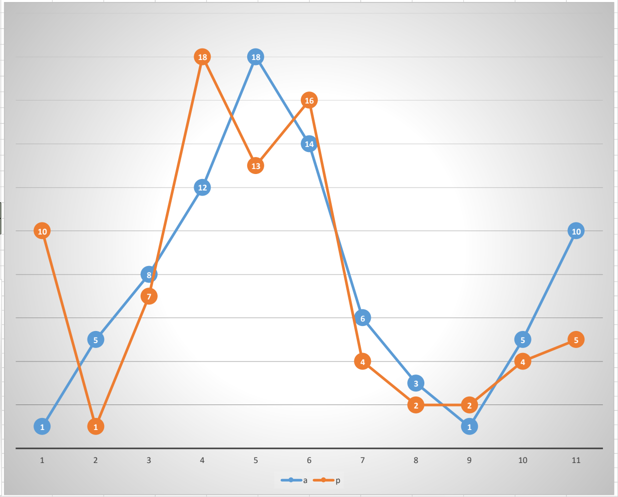


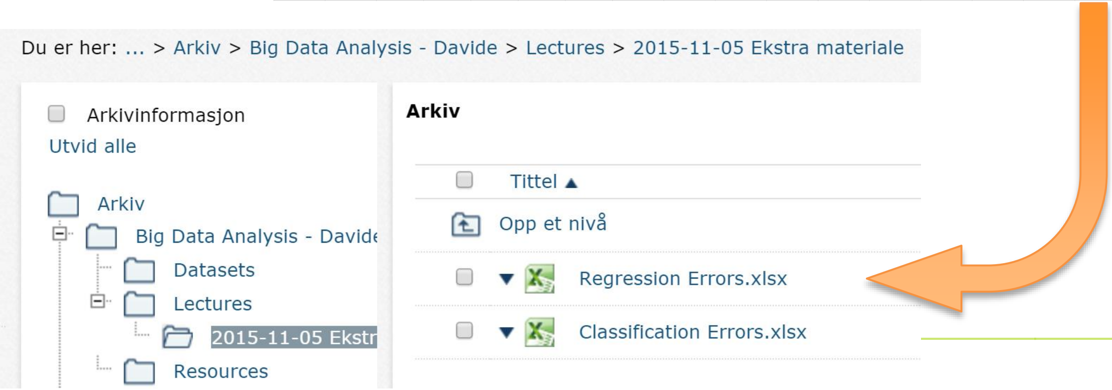


Regression Model Evaluation - MAE

a	p	(p - a)	|p-a|
1	10	9	9
5	1	-4	4
8	7	-1	1
12	18	6	6
18	13	-5	5
14	16	2	2
6	4	-2	2
3	2	-1	1
1	2	1	1
5	4	-1	1
10	5	-5	5
7,55	7,45	-0,09	3,36
Mean	Mean	ME	MAE







Regression Model Evaluation

RMSE – Root Mean Squared Error

- Measures the average of the squares of the errors, and then takes the root of that value
- Takes the squares of the errors so that positive and negative errors are treated equally
- Takes the root of that to "reverse" the squaring
- Compared to MAE it gives more weight to larger errors than smaller ones
- Scale dependent -> cannot be used to compare performance on different variables

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - a_i)^2}{n}}$$

$a = \text{actual target}$

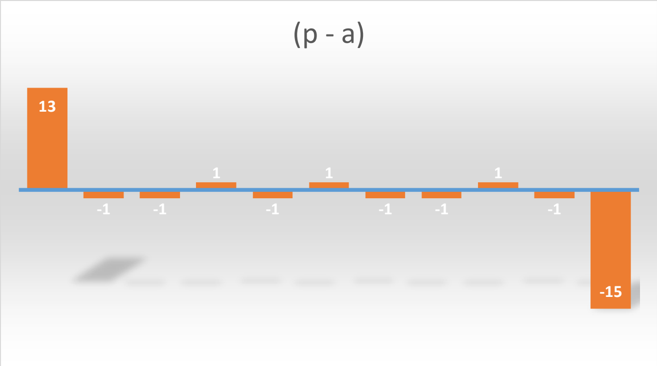
$p = \text{predicted target}$

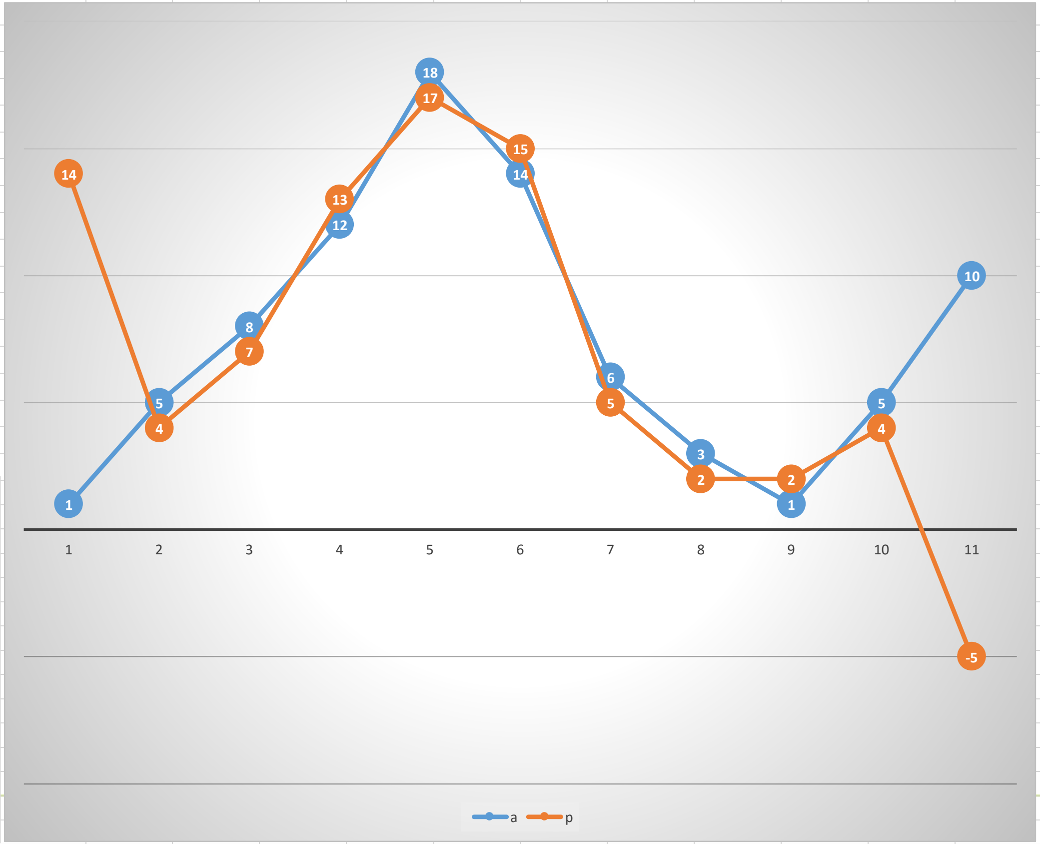
© www.esmartsystems.com

Regression Model Evaluation - RMSE

a	p	(p - a)	|p-a|	$(p-a)^2$
1	14	13	13	169
5	4	-1	1	1
8	7	-1	1	1
12	13	1	1	1
18	17	-1	1	1
14	15	1	1	1
6	5	-1	1	1
3	2	-1	1	1
1	2	1	1	1
5	4	-1	1	1
10	-5	-15	15	225

7,55	7,09	-0,45	3,36	6,05
Mean	Mean	ME	MAE	RMSE
			0%	18%





Regression Model Evaluation

RAE – Relative Absolute Error

- Takes the total absolute error and normalizes it by dividing by the total absolute deviation from the mean
- Whereas MAE measures how far the prediction is from the truth in absolute values, RAE represents the error as a fraction of the true value (or as a percentage if multiplied by 100)
- Scale independent -> can be used to compare performance on different variables

$$RAE = \frac{\sum_{i=1}^{n} |p_i - a_i|}{\sum_{i=1}^{n} |\bar{a} - a_i|}$$

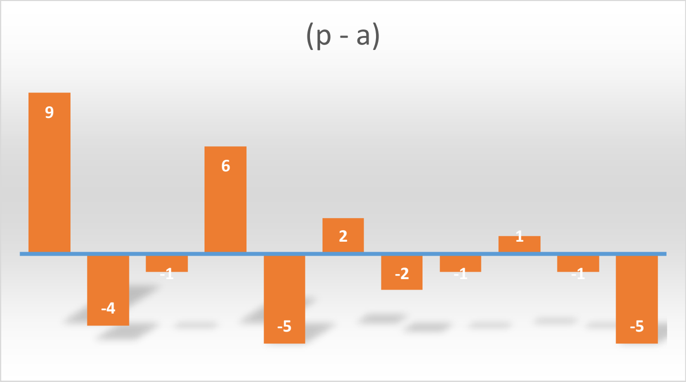
a = actual target

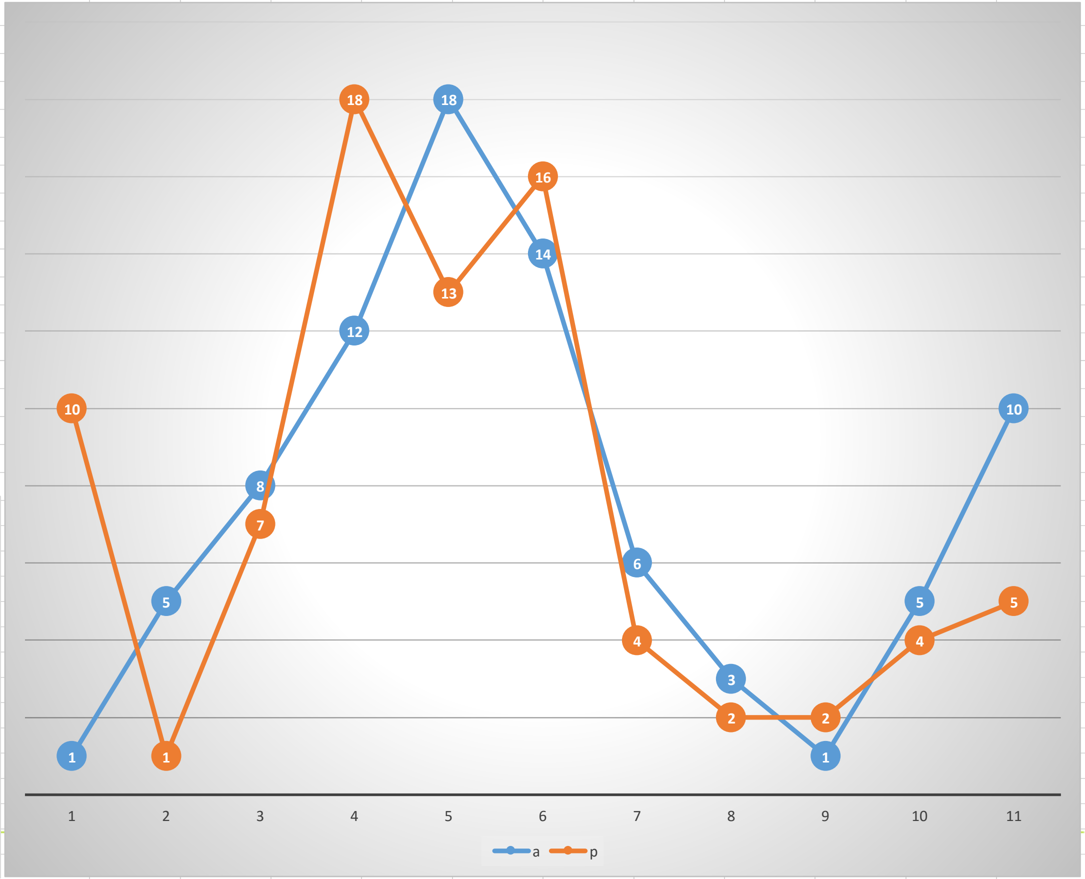
p = predicted target

Regression Model Evaluation - RAE

a	p	(p - a)	|p-a|	$(p-a)^2$	|ã-a|
1	10	9	9	81	6,55
5	1	-4	4	16	2,55
8	7	-1	1	1	0,45
12	18	6	6	36	4,45
18	13	-5	5	25	10,45
14	16	2	2	4	6,45
6	4	-2	2	4	1,55
3	2	-1	1	1	4,55
1	2	1	1	1	6,55
5	4	-1	1	1	2,55
10	5	-5	5	25	2,45

7,55	7,45	-0,09	3,36	4,21	0,76
Mean	Mean	ME	MAE	RMSE	RAE





Regression Model Evaluation

RSE – Relative Squared Error

- Takes the total squared error and normalizes it by dividing by the total squared deviation from the mean (variance)
- Compared to RAE it gives more weight to larger errors than smaller ones
- Scale independent -> can be used to compare performance on different variables

$$RSE = \frac{\sum_{i=1}^{n}(p_i - a_i)^2}{\sum_{i=1}^{n}(\bar{a} - a_i)^2}$$

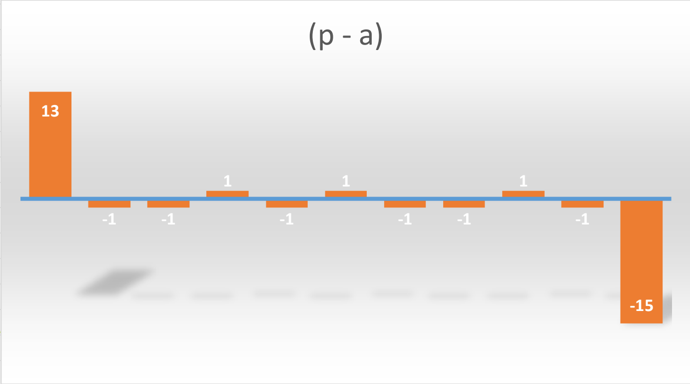
a = actual target

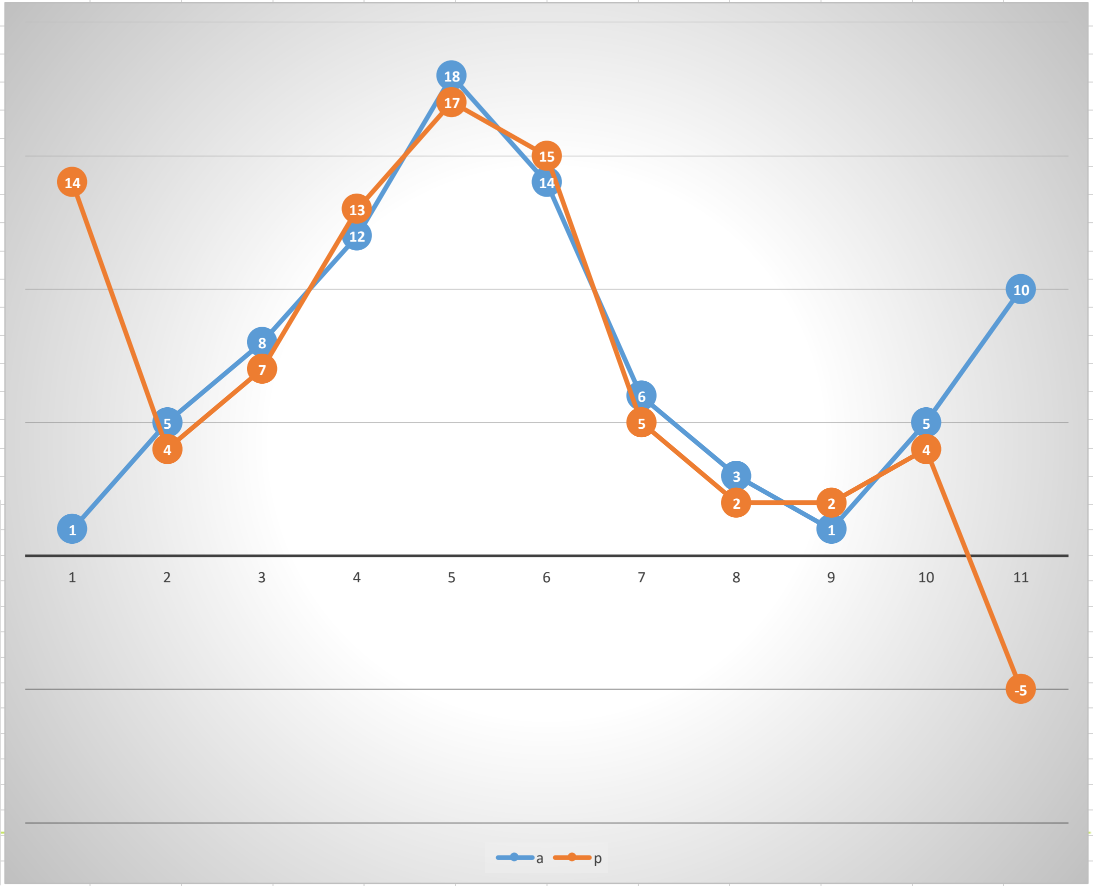
p = predicted target

© www.esmartsystems.com

Regression Model Evaluation - RSE

a	p	(p - a)	|p-a|	$(p-a)^2$	|ã-a|	$(ã-a)^2$
1	14	13	13	169	6,55	42,84
5	4	-1	1	1	2,55	6,48
8	7	-1	1	1	0,45	0,21
12	13	1	1	1	4,45	19,84
18	17	-1	1	1	10,45	109,30
14	15	1	1	1	6,45	41,66
6	5	-1	1	1	1,55	2,39
3	2	-1	1	1	4,55	20,66
1	2	1	1	1	6,55	42,84
5	4	-1	1	1	2,55	6,48
10	-5	-15	15	225	2,45	6,02
7,55	7,09	-0,45	3,36	6,05	0,76	1,35
Mean	Mean	ME	MAE	RMSE	RAE	RSE
					0%	35%





Regression Model Evaluation

R^2 – Coefficient of Determination

- R^2 describes the proportion of variance of the target variable explained by the regression model

    If the regression model is "perfect", SSE is zero, and R^2 is 1.

    If the regression model is a total failure, SSE is equal to SST, no variance is explained by regression, and R^2 is zero

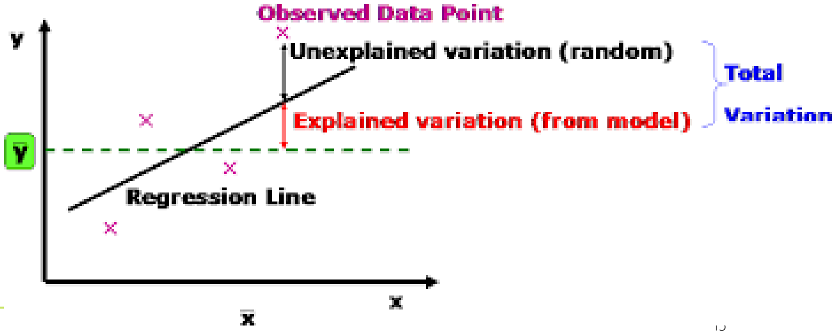
- Scale independent -> can be used to compare performance on different variables

Coefficient of Determination → $$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

Sum of Squares Total → $$SST = \sum (y - \bar{y})^2$$

Sum of Squares Regression → $$SSR = \sum (y' - \bar{y'})^2$$

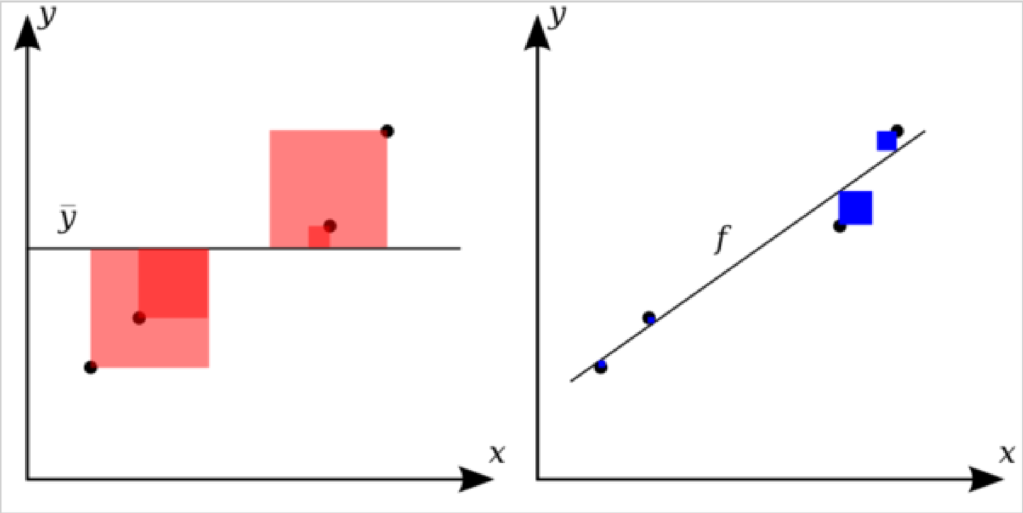
Sum of Squares Error → $$SSE = \sum (y - y')^2$$



Regression Model Evaluation

R^2 – Coefficient of Determination

Measure of goodness of fit

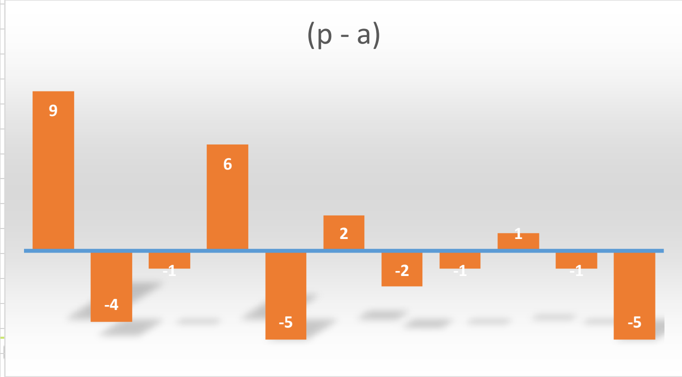


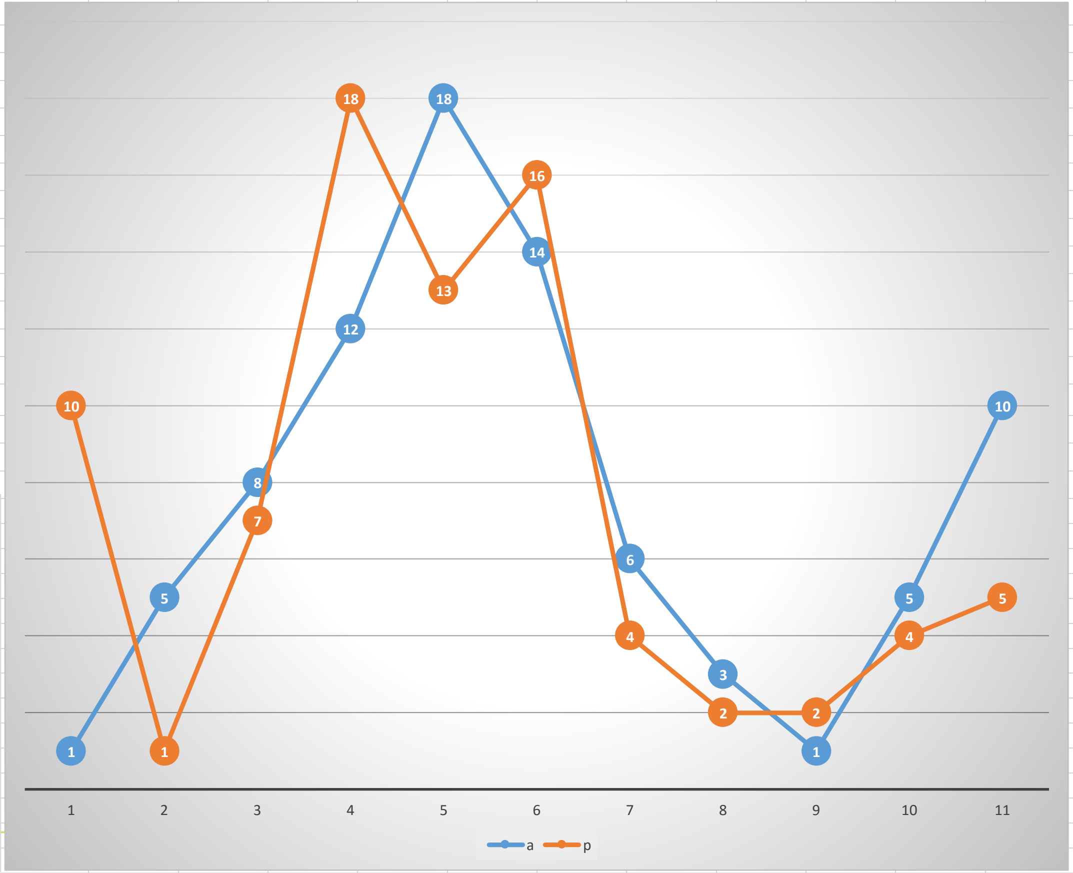
$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

The better the linear regression (on the right) fits the data in comparison to the simple average (on the left graph), the closer the value of R^2 is to 1. The areas of the blue squares represent the squared residuals with respect to the linear regression. The areas of the red squares represent the squared residuals with respect to the average value.

Regression Model Evaluation – R^2

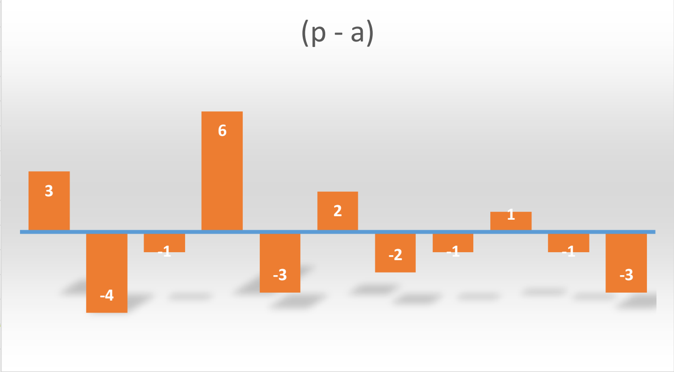
a	p	(p - a)	|p-a|	$(p-a)^2$	|ã-a|	$(ã-a)^2$	
1	10	9	9	81	6,55	42,84	
5	1	-4	4	16	2,55	6,48	
8	7	-1	1	1	0,45	0,21	
12	18	6	6	36	4,45	19,84	
18	13	-5	5	25	10,45	109,30	
14	16	2	2	4	6,45	41,66	
6	4	-2	2	4	1,55	2,39	
3	2	-1	1	1	4,55	20,66	
1	2	1	1	1	6,55	42,84	
5	4	-1	1	1	2,55	6,48	
10	5	-5	5	25	2,45	6,02	
7,55	7,45	-0,09	3,36	4,21	0,76	0,65	0,35
Mean	Mean	ME	MAE	RMSE	RAE	RSE	R^2
				195		298,73	
				SSE		SST	

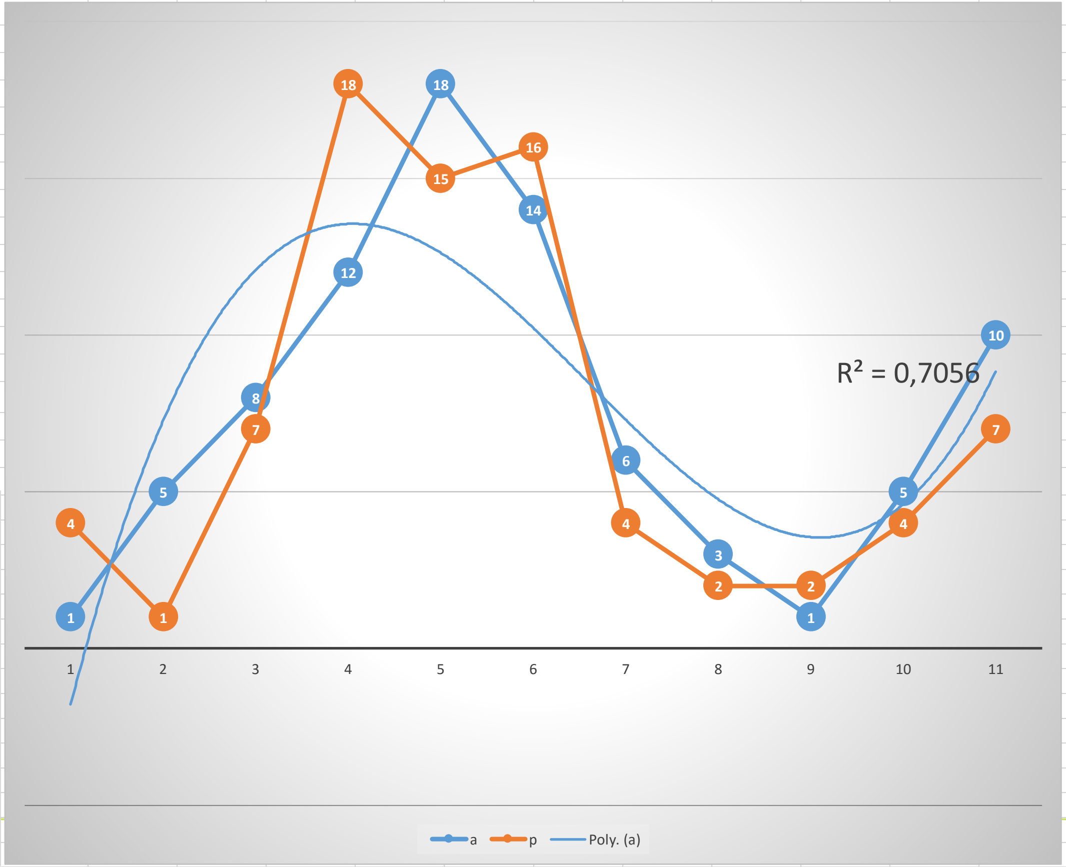




Regression Model Evaluation – R^2

a	p	(p - a)	|p-a|	$(p-a)^2$	|ã-a|	$(ã-a)^2$	
1	4	3	3	9	6,55	42,84	
5	1	-4	4	16	2,55	6,48	
8	7	-1	1	1	0,45	0,21	
12	18	6	6	36	4,45	19,84	
18	15	-3	3	9	10,45	109,30	
14	16	2	2	4	6,45	41,66	
6	4	-2	2	4	1,55	2,39	
3	2	-1	1	1	4,55	20,66	
1	2	1	1	1	6,55	42,84	
5	4	-1	1	1	2,55	6,48	
10	7	-3	3	9	2,45	6,02	
7,55	7,27	-0,27	2,45	2,88	0,56	0,30	0,70
Mean	Mean	ME	MAE	RMSE	RAE	RSE	R^2
				91		298,73	
				SSE		SST	





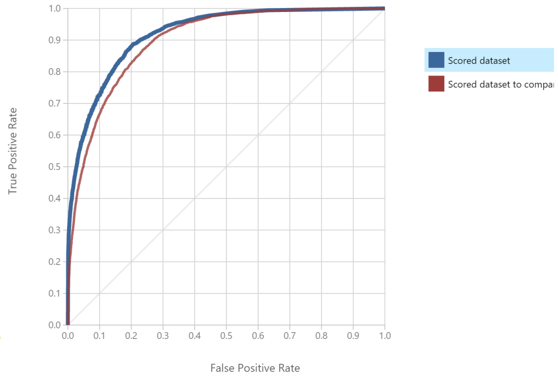
Classification Model Evaluation

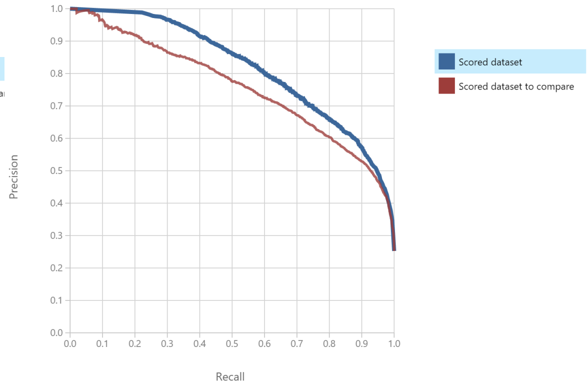
Classification

- Confusion Matrix
- Accuracy
- Precision
- Recall
- F-score
- ROC
- Precision/Recall
- Lift

	A	B	C	D	E	F	G
A	85.1%						0.5%
B		73.1%		3.1%	0.3%	0.5%	0.3%
C			75.5%		2.7%		5.5%
D		3.9%		79.4%		0.3%	
E		4.0%	0.2%	0.2%	42.2%	2.5%	8.1%
F		4.4%		3.9%	0.5%	49.3%	1.7%
G	1.6%	3.4%	20.6%		1.3%	0.8%	38.5%

True Positive	False Negative	Accuracy	Precision
2606	1203	0.861	0.745
False Positive	**True Negative**	**Recall**	**F1 Score**
890	10382	0.684	0.713
Positive Label	**Negative Label**		
>50K	<=50K		





Classification Model Evaluation

Confusion Matrix

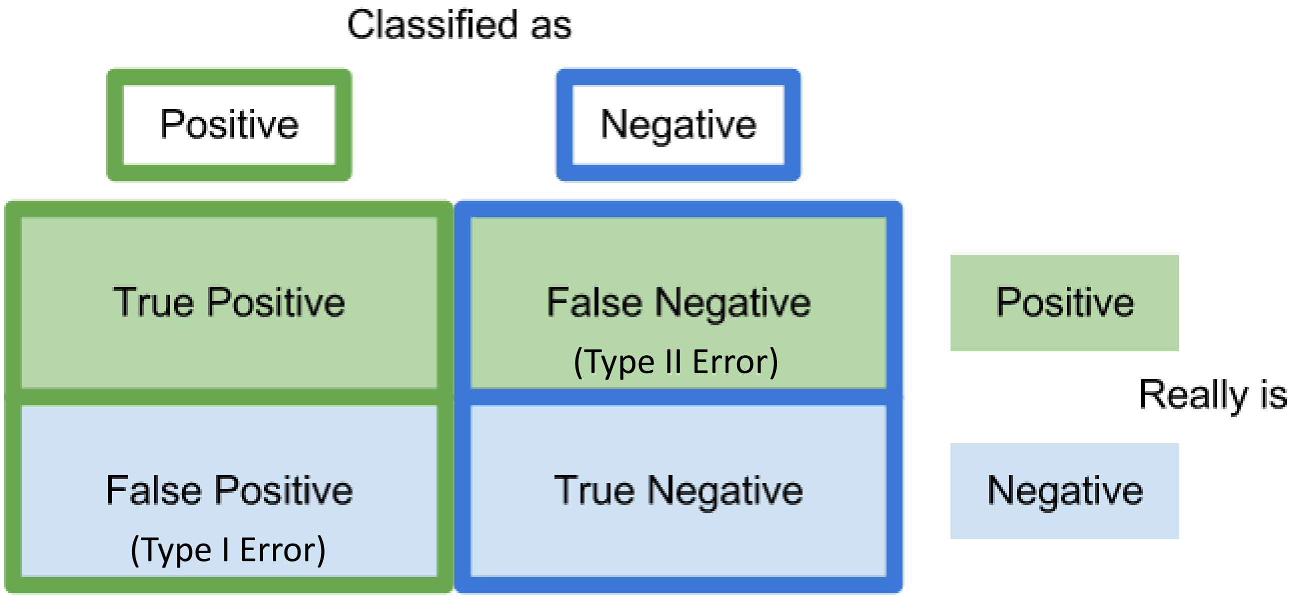
- A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known

		Predicted		
		Cat	Dog	Rabbit
Actual class	Cat	5	3	0
	Dog	2	3	1
	Rabbit	0	2	11

Two-Class Classification Model Evaluation

Confusion Matrix

- True Positive (TP), True Negative (TN), False Positive (FP), False Negatives (FN)



Two-Class Classification Model Evaluation

Accuracy = (TP + TN) / Total

- Overall, how often is the classifier correct

Precision = TP / (TP + FP) = TP / Predicted positive

- When it predicts yes, how often is it correct?
- "focus" on False Positives

Recall = TP / (TP + FN) = TP / Actual positive

- When it's actually yes, how often does it predict yes?
- "focus" on False Negatives

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

- The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst at 0

Two-Class Classification Model Evaluation

Confusion Matrix

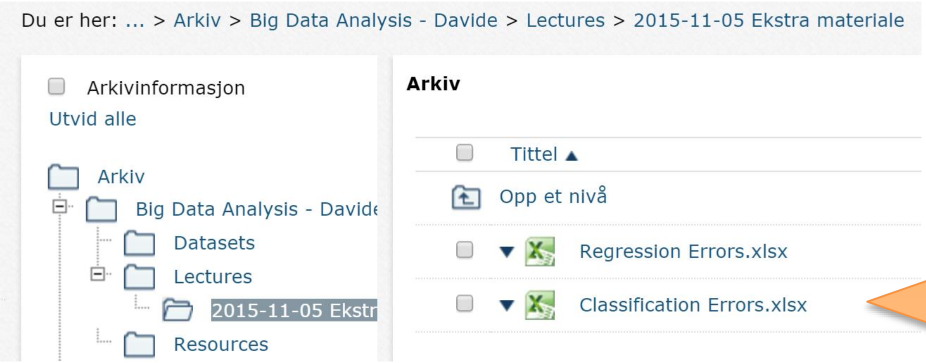
- In Azure ML

True Positive	False Negative	Accuracy	Precision
2606	1203	0.861	0.745
False Positive	True Negative	Recall	F1 Score
890	10382	0.684	0.713
Positive Label	Negative Label		
>50K	<=50K		

http://blogs.msdn.com/b/andreasderuiter/archive/2015/02/09/performance-measures-in-azure-ml-accuracy-precision-recall-and-f1-score.aspx

Two-Class Classification Model Evaluation

		Predicted								
		<=50K	>50K	tot						
Actual	<=50K	2606	1203	*3809*	Precision	**0,745**	0,821	F1	**0,713**	0,811
	>50K	890	10382	*11272*	Precision	0,896		F1	0,908	
	tot	*3496*	*11585*	*15081*						
		Recall	Recall	Accuracy	**0,861**					
		0,684	0,921							
		0,803								



Two-Class Classification Model Evaluation

Confusion Matrix

- Example from earlier lesson when we talked about class imbalance

Imagine we have

```
9900 examples of class A
 100 examples of class B
```

A model that always predicts A will be 99% accurate

This is obviously not what we want!

		Predicted				
		A	B	tot		
Actual	A	9900	0	9900	Precision	0,990
	B	100	0	100	Precision	#DIV/0!
	tot	10000	0	10000		
		Recall	Recall	Accuracy	0,990	
		1,000	0,000			
		0,500				

Two-Class Classification Model Evaluation

eSmart SYSTEMS

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
2606	1203	0.861	0.745	0.5	0.920
False Positive	True Negative	Recall	F1 Score		
890	10382	0.684	0.713		
Positive Label	Negative Label				
>50K	<=50K				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall
(0.900,1.000]	1683	199	0.125	0.846	0.591	0.894	0.442	0.839	0.982
(0.800,0.900]	370	179	0.161	0.858	0.658	0.845	0.539	0.861	0.966
(0.700,0.800]	220	175	0.187	0.861	0.685	0.804	0.597	0.875	0.951
(0.600,0.700]	176	155	0.209	0.863	0.703	0.776	0.643	0.886	0.937
(0.500,0.600]	157	179	0.232	0.861	0.714	0.746	0.684	0.896	0.921
(0.400,0.500]	158	218	0.257	0.857	0.720	0.714	0.726	0.907	0.902
(0.300,0.400]	172	282	0.287	0.850	0.722	0.679	0.771	0.919	0.877
(0.200,0.300]	188	331	0.321	0.841	0.722	0.645	0.820	0.933	0.848
(0.100,0.200]	216	509	0.369	0.821	0.712	0.600	0.877	0.951	0.802
(0.000,0.100]	469	9045	1.000	0.253	0.403	0.253	1.000	1.000	0.000

Two-Class Classification Model Evaluation

		Predicted								
		<=50K	>50K	tot						
Actual	<=50K	2606	1203	*3809*	Precision	**0,745**	0,821	F1	**0,713**	0,811
	>50K	890	10382	*11272*	Precision	0,896		F1	0,908	
	tot	*3496*	*11585*	*15081*						
		Recall	Recall	Accuracy	**0,861**					
		0,684	0,921							
		0,803								

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall
(0.900,1.000]	1683	199	0.125	0.846	0.591	0.894	0.442	0.839	0.982
(0.800,0.900]	370	179	0.161	0.858	0.658	0.845	0.539	0.861	0.966
(0.700,0.800]	220	175	0.187	0.861	0.685	0.804	0.597	0.875	0.951
(0.600,0.700]	176	155	0.209	0.863	0.703	0.776	0.643	0.886	0.937
(0.500,0.600]	157	179	0.232	0.861	0.714	0.746	0.684	0.896	0.921

© www.esmartsystems.com

Two-Class Classification Model Evaluation

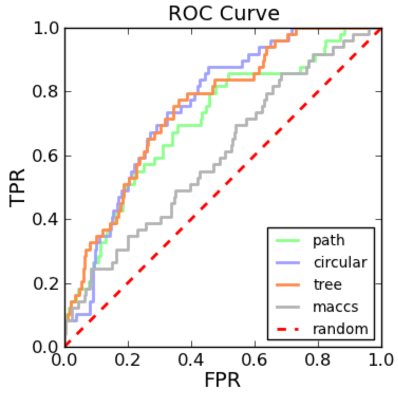
		True condition			
	Total population	Condition positive	Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
Predicted condition	Predicted condition positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$
	Predicted condition negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$
	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR−}}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$	

https://en.wikipedia.org/wiki/Confusion_matrix

Two-Class Classification Model Evaluation

ROC Curve - Receiver Operating Characteristic curve

- Graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied
- The curve is created by plotting the true positive rate (TPR, or Recall) against the false positive rate (FPR, or Fall-out) at various threshold settings



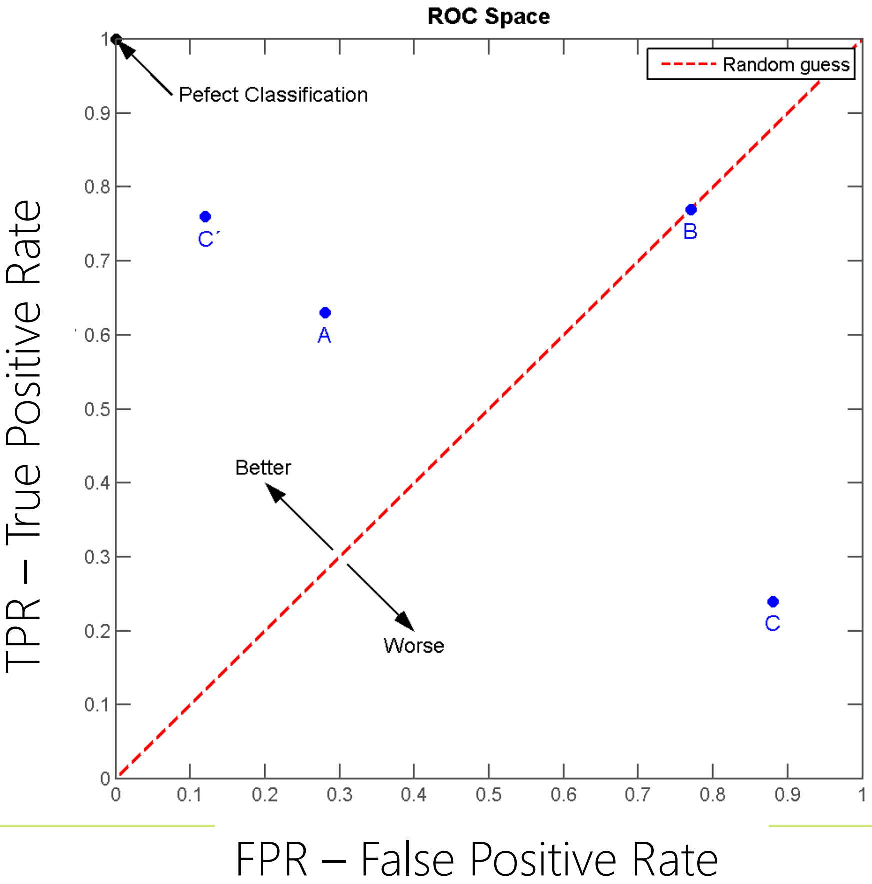
  - The ROC curve was first developed by electrical engineers and radar engineers during World War II for detecting enemy objects in battlefields

Find balance between True Positives and False Positives

© www.esmartsystems.com

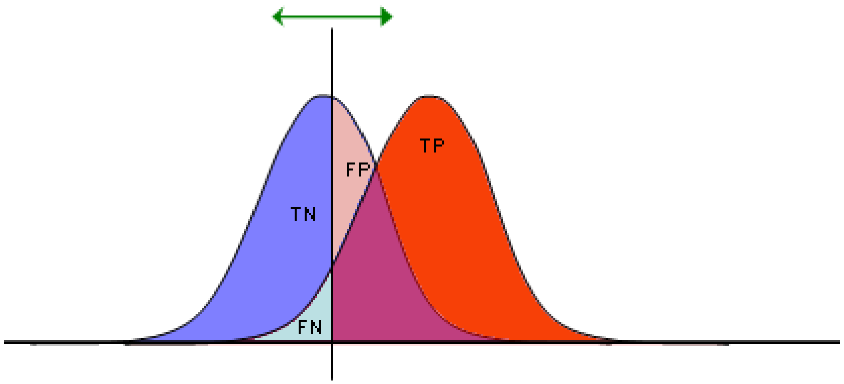
Two-Class Classification Model Evaluation

ROC Space

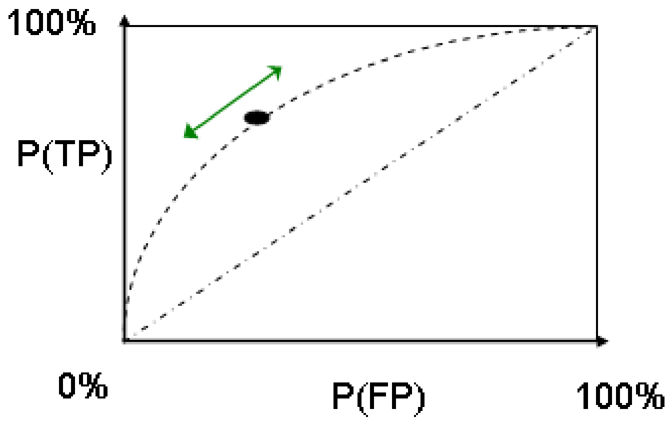


Two-Class Classification Model Evaluation

ROC Curve

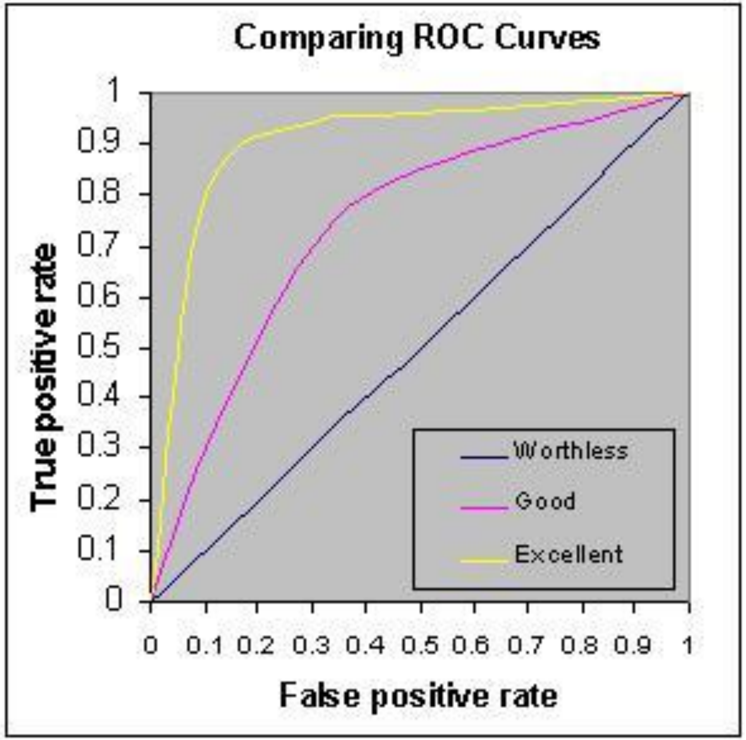


TP	FP
FN	TN
1	1



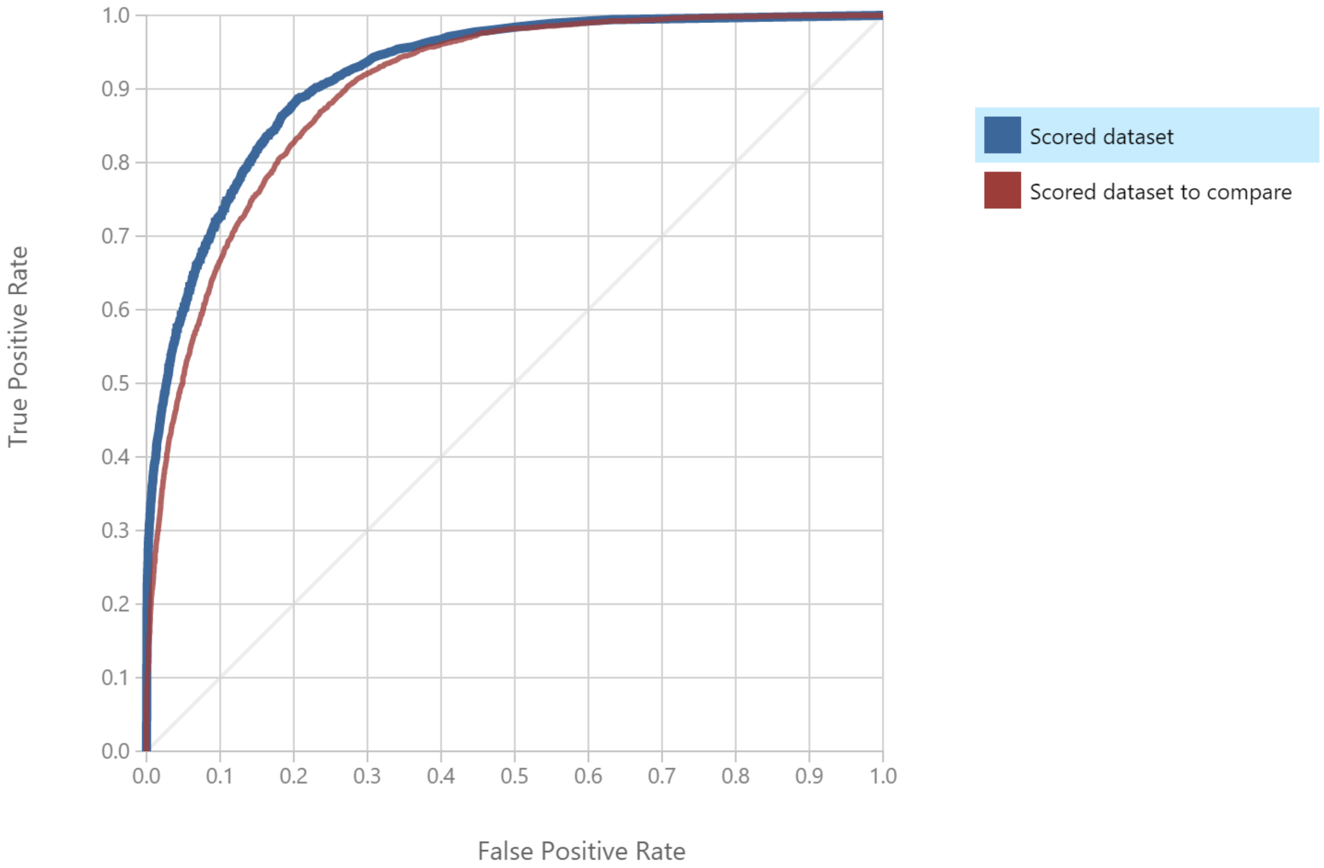
Two-Class Classification Model Evaluation

ROC Curve



Two-Class Classification Model Evaluation

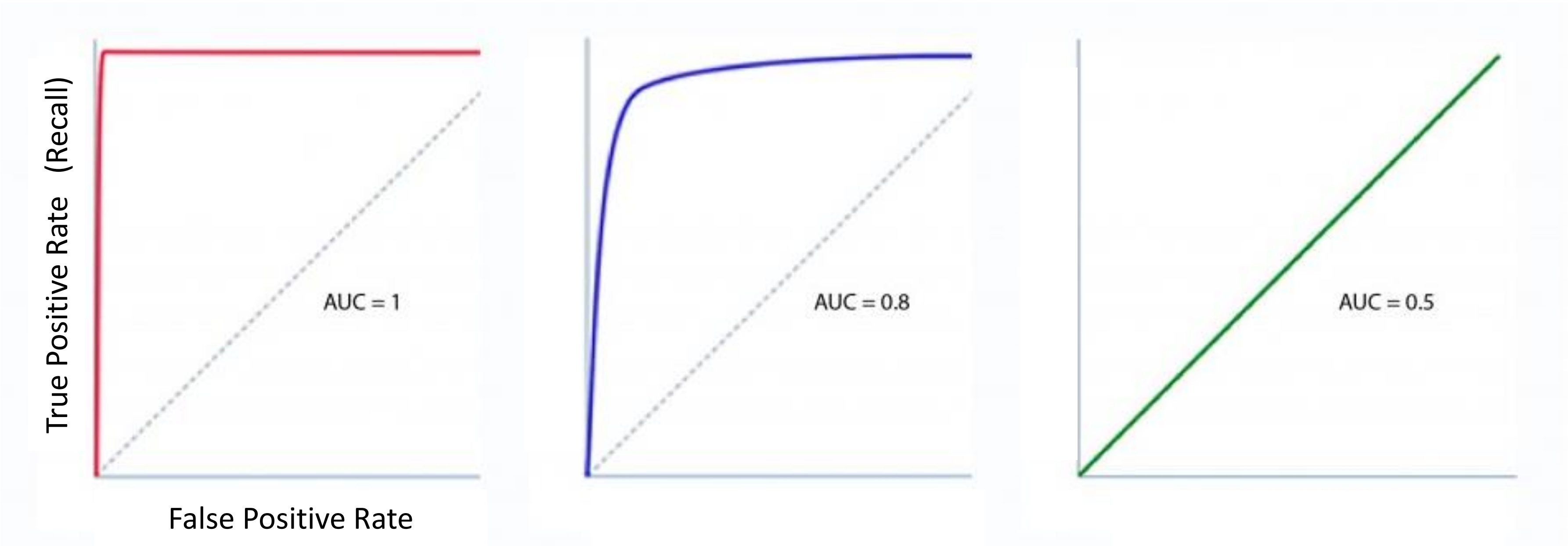
ROC Curve



POWERED BY VALUES

Two-Class Classification Model Evaluation

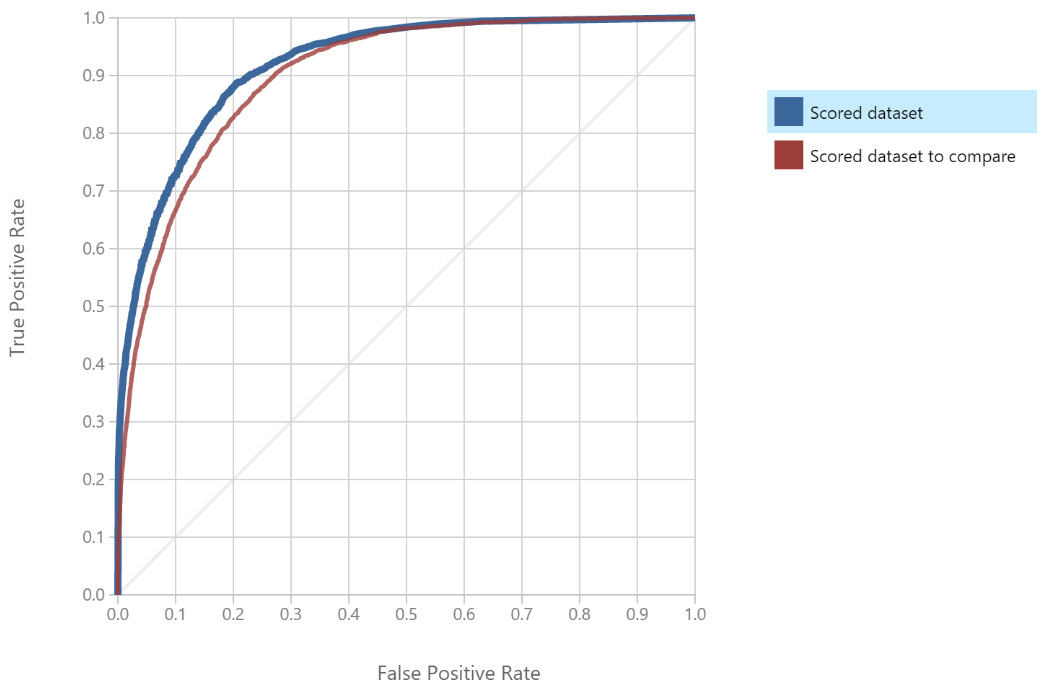
ROC AUC – Area Under the Curve



© www.esmartsystems.com

Two-Class Classification Model Evaluation

ROC AOC



Threshold

0.5

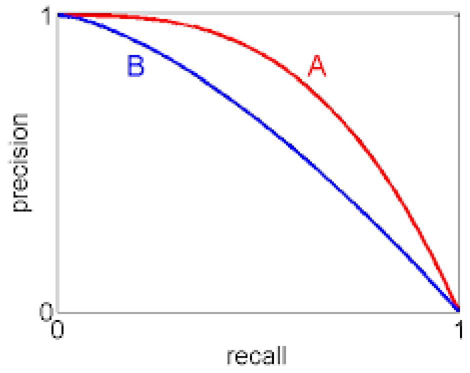
AUC

0.920

Two-Class Classification Model Evaluation

Precision/Recall Curve

- Graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied
- The curve is created by plotting the Precision against the Recall at various threshold settings



Precision = TP / (TP + FP) = TP / Predicted positive

- When it predicts yes, how often is it correct?
- "focus" on False Positives

Recall = TP / (TP + FN) = TP / Actual positive

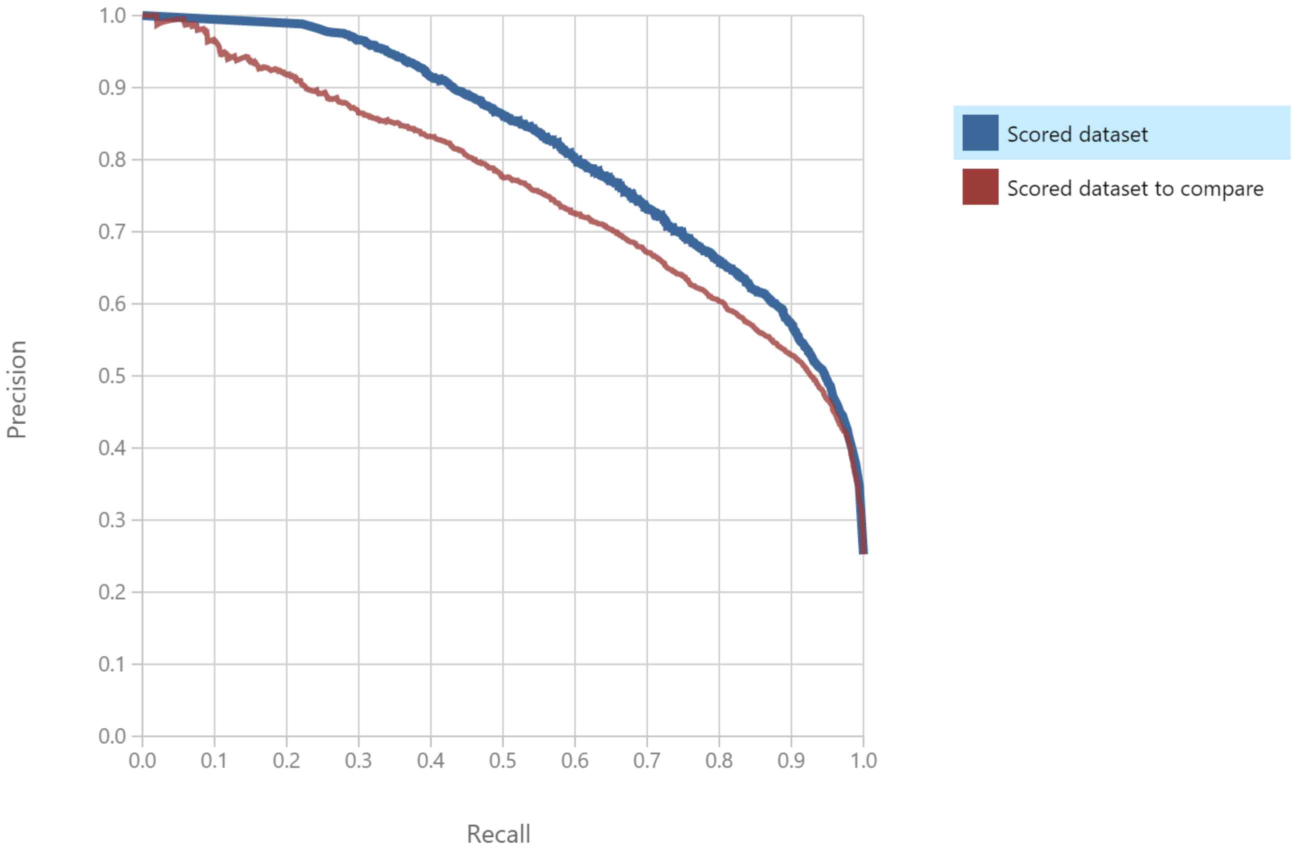
- When it's actually yes, how often does it predict yes?
- "focus" on False Negatives

Find balance between False Positives and False Negatives

Two-Class Classification Model Evaluation

eSmart SYSTEMS

Precision/Recall Curve

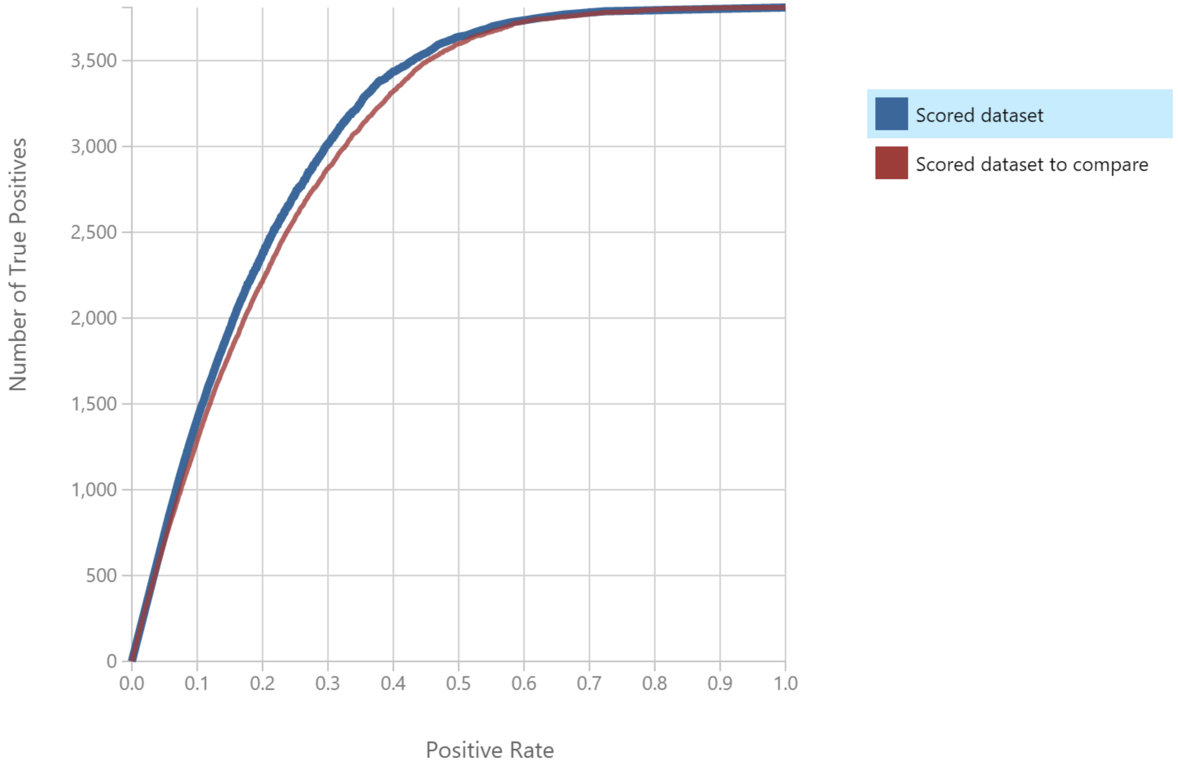


POWERED BY VALUES

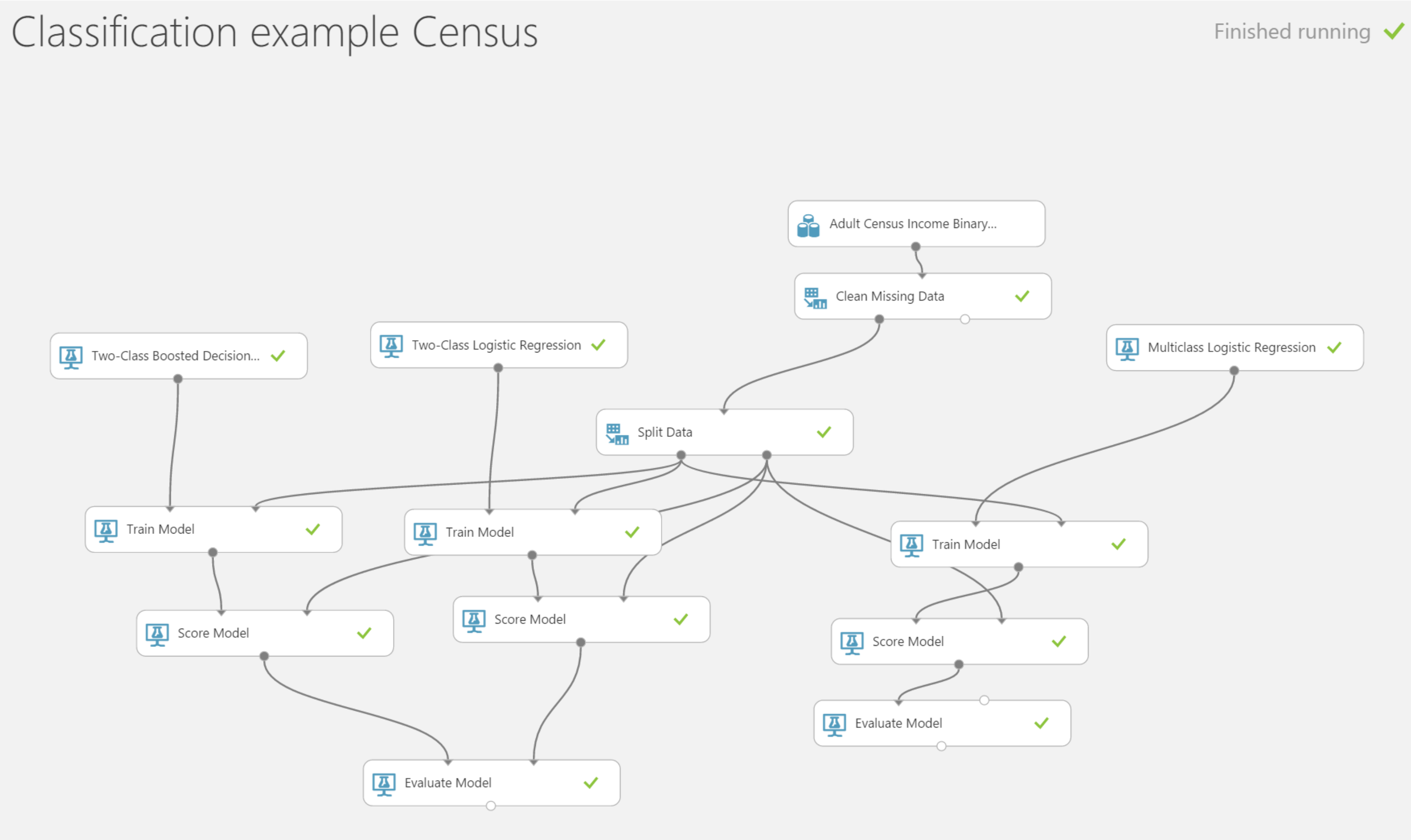
Two-Class Classification Model Evaluation

Lift

"not documented"



Two-Class Classification Model Evaluation

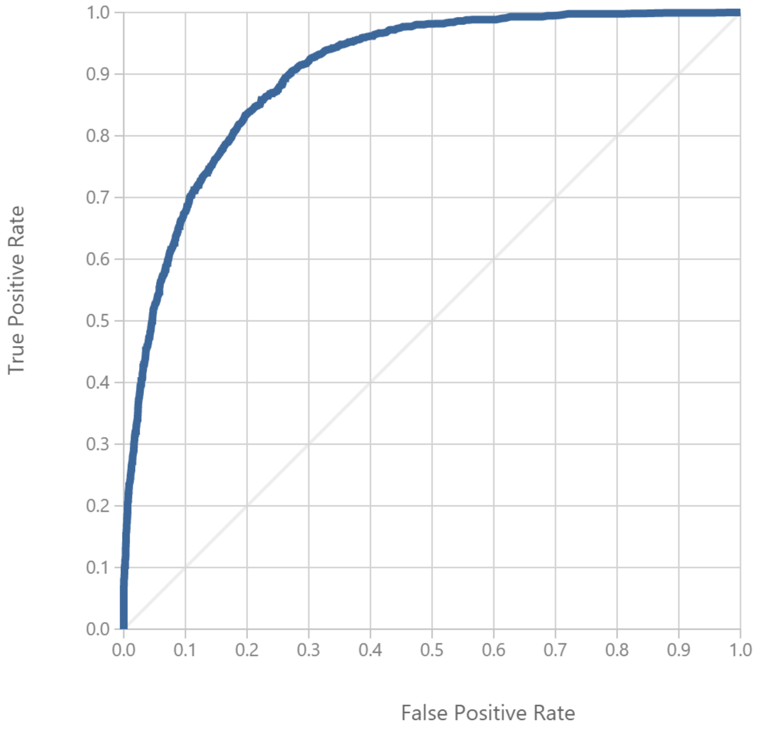


POWERED

Two-Class Classification Model Evaluation

ROC Curve in Azure ML

http://blogs.msdn.com/b/andreasderuiter/archive/2015/02/10/using-roc-plots-and-the-auc-measure-in-azure-ml.aspx



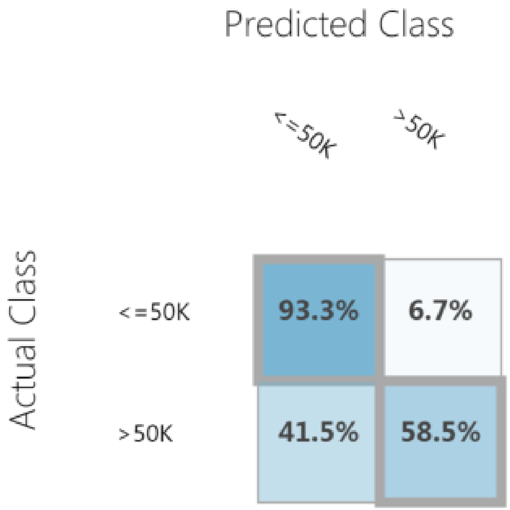
© www.esmartsystems.com

Multiclass Classification Model Evaluation

Confusion Matrix

- In Azure ML

Multiclass



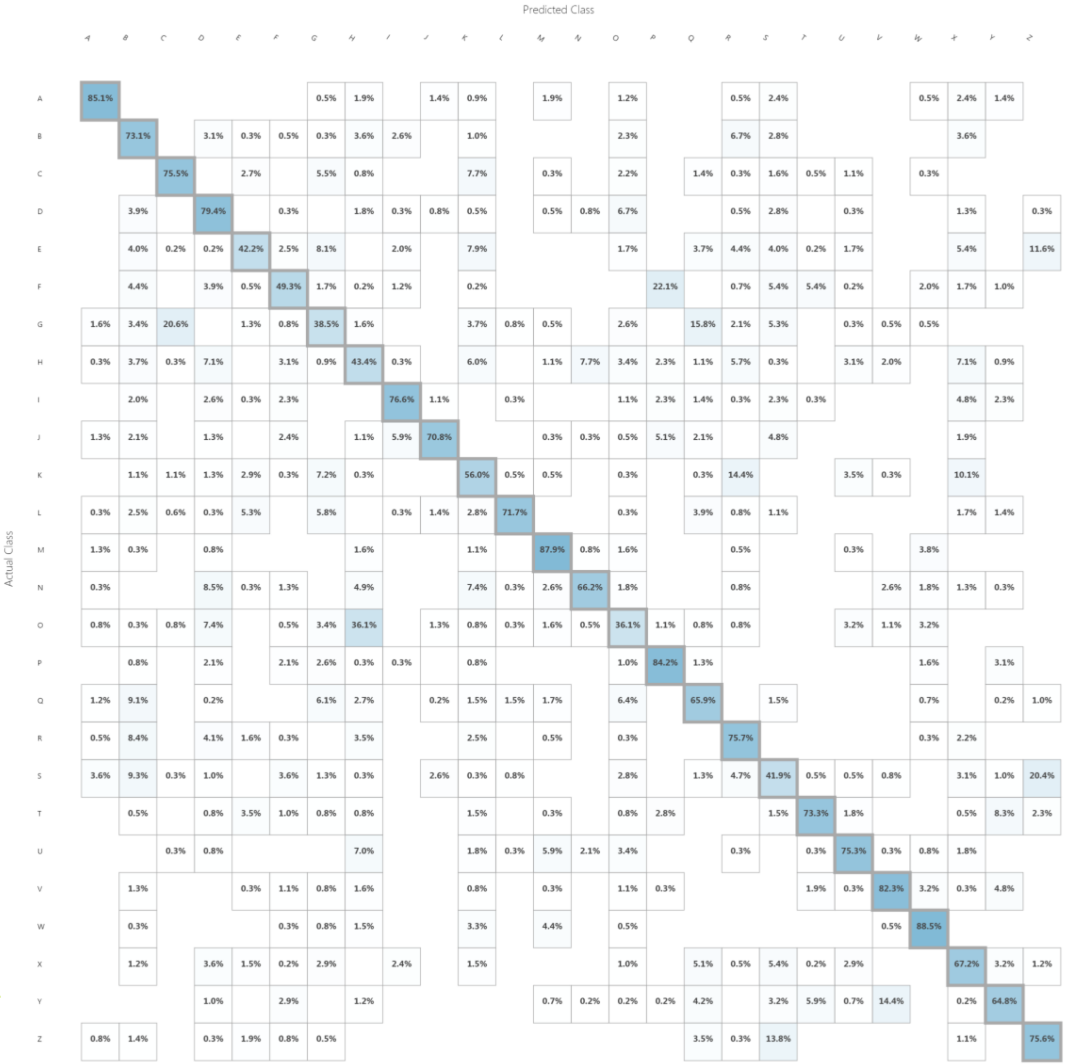
Two-Class

True Positive	False Negative
1104	786
False Positive	True Negative
385	5265
Positive Label	Negative Label
>50K	<=50K

© www.esmartsystems.com

Multicalss Classification Model Evaluation

Confusion Matrix



POWERED BY VALUES

Multicclss Classification Model Evaluation

Metrics

Metrics

Overall accuracy	0.8068
Average accuracy	0.985138
Micro-averaged precision	0.8068
Macro-averaged precision	0.813606
Micro-averaged recall	0.8068
Macro-averaged recall	0.806804

Multicclss Classification Model Evaluation

Metrics

Metrics

Overall accuracy	0.8068
Average accuracy	0.985138
Micro-averaged precision	0.8068
Macro-averaged precision	0.813606
Micro-averaged recall	0.8068
Macro-averaged recall	0.806804

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Overall Accuracy vs Average Accuracy

3 classes A, B, and C

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

$$overall\ accuracy = \frac{TP(A) + TP(B) + TP(C)}{total} = \frac{31 + 40 + 23}{127} = 0.74$$

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Overall Accuracy vs **Average Accuracy**

3 classes A, B, and C

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

$$average\ accuracy = \frac{accuracy(A) + accuracy(B) + accuracy(C)}{3}$$

$$accuracy(A) = \frac{TP(A) + TN(A)}{total}$$

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Overall Accuracy vs **Average Accuracy**

Accuracy of A against B & C

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

$$accuracy(A) = \frac{TP(A) + TN(A)}{total}$$

$$accuracy(A) = \frac{31 + (40 + 1 + 11 + 23)}{127}$$

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Overall Accuracy vs **Average Accuracy**

		Predicted			
		A	B	C	*tot*
Actual	A	31	4	6	*41*
	B	3	40	1	*44*
	C	8	11	23	*42*
	tot	*42*	*55*	*30*	*127*

$$average\ accuracy = \frac{accuracy(A) + accuracy(B) + accuracy(C)}{3}$$

$$= \frac{\frac{TP(A) + TN(A)}{total} + \frac{TP(B) + TN(B)}{total} + \frac{TP(C) + TN(C)}{total}}{3}$$

$$= \frac{\frac{31 + (41 + 1 + 11 + 23)}{127} + \frac{40 + (31 + 6 + 8 + 23)}{127} + \frac{23 + (31 + 4 + 3 + 40)}{127}}{3} = 0.83$$

© www.esmartsystems.com

Multicclss Classification Model Evaluation

Metrics

Metrics

Overall accuracy	0.8068
Average accuracy	0.985138
Micro-averaged precision	0.8068
Macro-averaged precision	0.813606
Micro-averaged recall	0.8068
Macro-averaged recall	0.806804

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Micro vs Macro

The macro is the average of the precision/recall taken separately for each class. Therefore it is an average over classes.

The micro average on the contrary is an average over instances.

Therefore classes which have many instances are given more importance.

© www.esmartsystems.com

Multiclass Classification Model Evaluation

Micro vs **Macro**

$$precision(X) = \frac{TP(X)}{TP(X) + FP(X)} = \frac{TP(X)}{Total\ predicted\ as\ X}$$

$$recall(X) = \frac{TP(X)}{TP(X) + FN(X)} = \frac{TP(X)}{Total\ X}$$

3 classes A, B, and C

$$macro\ averaged\ precision = \frac{precision(A) + precision(B) + precision(C)}{3}$$

$$macro\ averaged\ recall = \frac{recall(A) + recall(B) + recall(C)}{3}$$

All classes are treated equally

© www.esmartsystems.com

Multicalss Classification Model Evaluation

Micro vs Macro

$$precision(X) = \frac{TP(X)}{TP(X) + FP(X)} = \frac{TP(X)}{Total\ predicted\ as\ X}$$

$$recall(X) = \frac{TP(X)}{TP(X) + FN(X)} = \frac{TP(X)}{Total\ X}$$

3 classes A, B, and C

$$micro\ averaged\ precision = \frac{TP(A,B,C)}{TP(A,B,C) + FP(A,B,C)} = \frac{TP(A,B,C)}{Total}$$

$$micro\ averaged\ recall = \frac{TP(A,B,C)}{TP(A,B,C) + FN(A,B,C)} = \frac{TP(A,B,C)}{Total}$$

Classes which have many instances are given more importance

$$micro\ averaged\ precision = micro\ averaged\ recall\ = overall\ accuracy$$

© www.esmartsystems.com

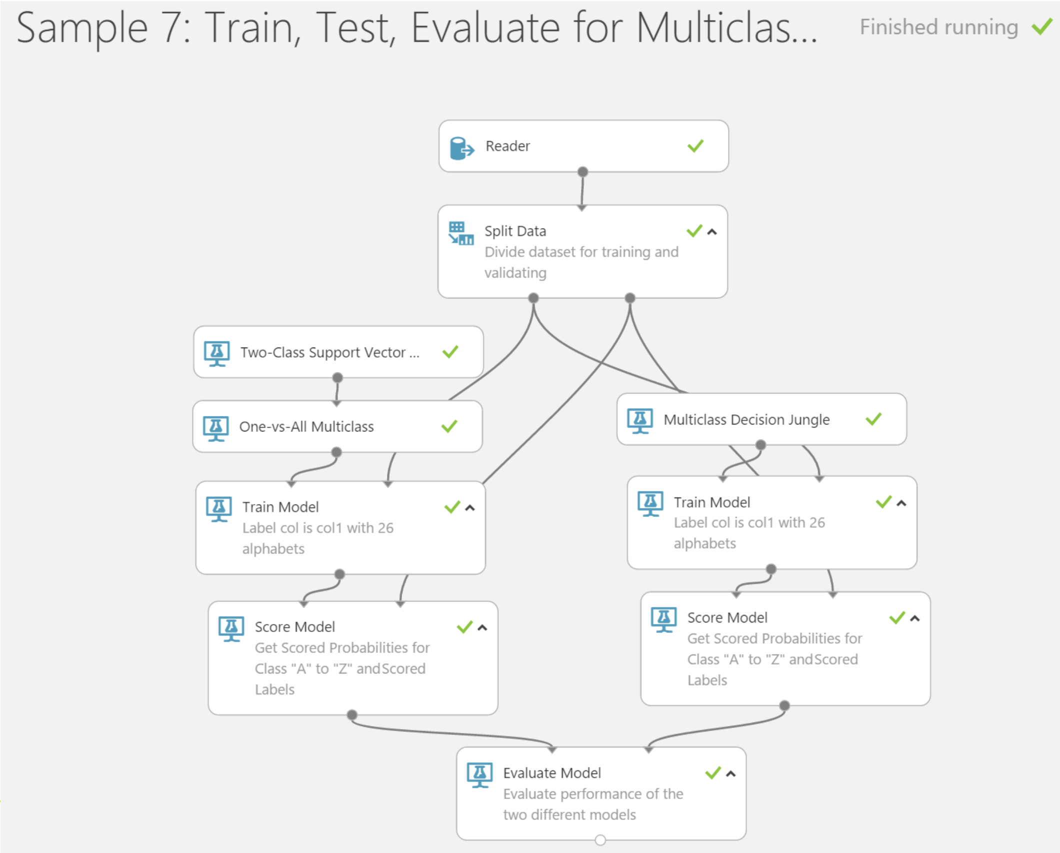
Multiclass Classification Model Evaluation

Metrics

Metrics

Overall accuracy	$\boxed{0.8068}$
Average accuracy	0.985138
Micro-averaged precision	$\boxed{0.8068}$
Macro-averaged precision	0.813606
Micro-averaged recall	$\boxed{0.8068}$
Macro-averaged recall	0.806804

Multiclass Classification Model Evaluation



Author: Swami Chandrasekaran
1. Fundamentals
Matrices & Linear Algebra Fundamentals
Hash Functions, Binary Tree, O(n)
Relational Algebra, DB Basics
Inner, Outer, Cross, Theta Join
CAP Theorem
Tabular Data
Entropy
Data Frames & Series
Sharding
OLAP
Multidimensional Data Model
ETL
Reporting Vs BI Vs Analytics
JSON & XML
NoSQL
Regex
Vendor Landscape
Env Setup
5%
2. Statistics
Pick a Dataset (UCI Repo)
Descriptive Statistics (mean, median, range, SD, Var)
Exploratory Data Analysis
Histograms
Percentiles & Outliers
Probability Theory
Bayes Theorem
Random Variables
Cumul Dist Fn (CDF)
Continuos Distributions (Normal, Poisson, Gaussian)
Skewness
ANOVA
Prob Den Fn (PDF)
Central Limit Theorem
Monte Carlo Method
Hypothesis Testing
p-Value
Chi² Test
Estimation
Confid Int (CI)
MLE
Kernel Density Estimate
Regression
Covariance
Correlation
Pearson Coeff
Causation
Least² Fit
Euclidean Distance
3. Programming
Python Basics
Working in Excel
R Setup R Studio
R Basics
Expressions
Variables
IBM SPSS
Rapid Miner
Vectors
Matrices
Arrays
Factors
Lists
Data Frames
Reading CSV Data
Reading Raw Data
Subsetting Data
Manipulate Data Frames
Functions
Install Pkgs
Factor Analysis
15%
4. Machine Learning
What is ML?
30%
Numerical Var
Categorical Var
Supervised Learning
Unsupervised Learning
Concepts, Inputs & Attributes
Training & Test Data
Classifier
Prediction
Lift
Overfitting
Bias & Variance
Trees & Classification
Classification Rate
Decision Trees
Boosting
Naive Bayes Classifiers
K-Nearest Neighbor
Classification
Logistic Regression
Ranking
Linear Regression
Regression
Perceptron
Hierarchical Clustering
K-means Clustering
Clustering
Neural Networks
Sentiment Analysis
Collaborative Filtering
Tagging
50%
5. Text Mining / NLP
Corpus
Named Entity Recognition
Text Analysis
UIMA
Term Document Matrix
Term Frequency & Weight
Support Vector Machines
Association Rules
Market Based Analysis
Feature Extraction
Using Mahout
Using Weka
Using NLTK
Classify Text
Vocabulary Mapping
6. Visualization
Data Exploration in R (Hist, Boxplot etc)
Uni, Bi & Multivariate Viz
ggplot2
Histogram & Pie (Uni)
Tree & Tree Map
Scatter Plot (Bi)
Line Charts (Bi)
Spatial Charts
Survey Plot
Timeline
Decision Tree
D3.js
InfoVis
IBM ManyEyes
Tableau
40%
7. Big Data
Map Reduce Fundamentals
Hadoop Components
HDFS
Data Replication Principles
Setup Hadoop (IBM / Cloudera / HortonWorks)
Name & Data Nodes
Job & Task Tracker
M/R Programming
Sqoop: Loading Data in HDFS
Flume, Scribe: For Unstruct Data
SQL with Pig
DWH with Hive
Scribe, Chukwa For Weblog
Using Mahout
Zookeeper Avro
Storm: Hadoop Realtime
Rhadoop, RHIPE
rmr
Cassandra
MongoDB, Neo4j
60%
8. Data Ingestion
Summary of Data Formats
Data Discovery
Data Sources & Acquisition
Data Integration
Data Fusion
Transformation & Enrichment
Data Survey
Google OpenRefine
How much Data?
Using ETL
80%
9. Data Munging
Dimensionality & Numerosity Reduction
Normalization
Data Scrubbing
Handling Missing Values
Unbiased Estimators
Binning Sparse Values
Feature Extraction
Denoising
Sampling
Stratified Sampling
Principal Component Analysis
10. Toolbox
MS Excel w/ Analysis ToolPak
Java, Python
R, R-Studio, Rattle
Weka, Knime, RapidMiner
Hadoop Dist of Choice
Spark, Storm
Flume, Scibe, Chukwa
Nutch, Talend, Scraperwiki
Webscraper, Flume, Sqoop
tm, RWeka, NLTK
RHIPE
D3.js, ggplot2, Shiny
IBM Languageware
Cassandra, MongoDB
100%