Computer vision in the new era of Artificial Intelligence and Deep Learning

Visión por computador en la nueva era de la Inteligencia Artificial y el Deep Learning

Rubén Usamentiaga*, Alberto Fernández°
*** University of Oviedo**
° TSK

Gijón (Spain)
5 – 16 April 2021

https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course

Scikit-learn

Introduction to metrics in scikit-learn

- metrics_for_classification_with_scikit_learn.ipynb

Open in Colab

- metrics_for_classification_with_scikit_learn.ipynb

https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course

Metrics for classification

Study evaluating a test that screens people for a disease. The test outcome can be:

- **Positive**: classifying the person as having the disease
- **Negative**: classifying the person as not having the disease

- **True positive (TP)**: Sick people (actual = 1) correctly identified as sick (predicted = 1)
- **False positive (FP)**: Healthy people (actual = 0) incorrectly identified as sick (predicted = 1)
- **True negative (TN)**: Healthy people (actual = 0) correctly identified as healthy (predicted = 0)
- **False negative (FN)**: Sick people (actual = 1) incorrectly identified as healthy (predicted = 0)

Confusion Matrix		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Metrics for classification

Study evaluating a test that screens people for a disease. The test outcome can be:

- **Positive**: classifying the person as having the disease
- **Negative**: classifying the person as not having the disease

- **True positive (TP)**: Sick people (actual = 1) correctly identified as sick (predicted = 1)
- **False positive (FP)**: Healthy people (actual = 0) incorrectly identified as sick (predicted = 1)
- **True negative (TN)**: Healthy people (actual = 0) correctly identified as healthy (predicted = 0)
- **False negative (FN)**: Sick people (actual = 1) incorrectly identified as healthy (predicted = 0)

Confusion Matrix		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

All the samples that are negative

All the samples that are positive

All the samples predicted as negative

All the samples predicted as positive

Metrics for classification

Using TP, FP, FN and TN, we can calculate some metrics: sensitivity (or recall), specificity, and precision.

- **Sensitivity (recall)** is a measure of how well a test can identify true positives
- **Specificity** is a measure of how well a test can identify true negatives.
- **Precision** (positive predictive values (PPV)). It is the proportions of positive results that are true positive

Confusion Matrix		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Specificity

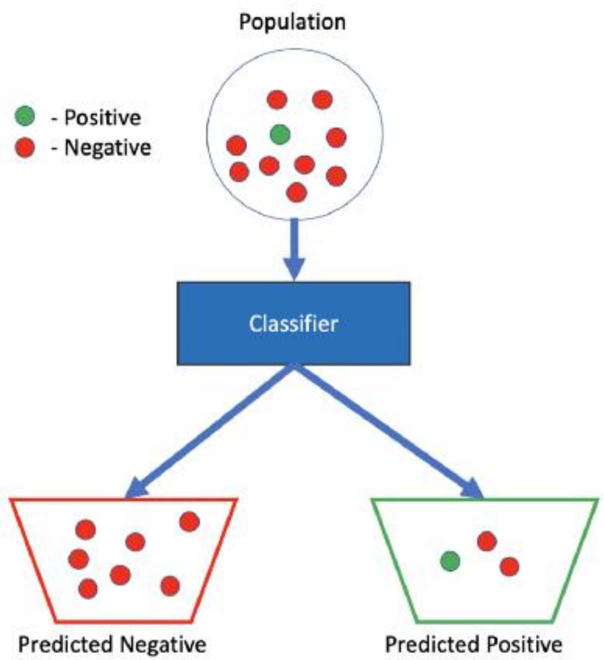
Sensitivity (recall)

Precision

how sure you are of your true positives

Metrics for classification

- Sensitivity (recall) is a measure of how well a test can identify true positives
- Specificity is a measure of how well a test can identify true negatives.
- Precision (positive predictive values (PPV)). It is the proportions of positive results that are true positive (how sure you are of your true positives).



		Real	
		Positive	Negative
Predicted	Positive	1	2
	Negative	0	7

$$precision = \frac{tp}{tp+fp} = \frac{1}{3} = 33\%$$
$$recall = \frac{tp}{tp+fn} = \frac{1}{1} = 100\%$$
$$specificity = \frac{tn}{tn+fp} = \frac{7}{9} = 78\%$$
$$sensitivity = recall = 100\%$$

Accuracy: 80.00% (8/10)
F1 score (2 * (precision * recall) / (precision + recall)): 50.00%:

Metrics for classification in scikit-learn

```
from sklearn.metrics import accuracy_score

from sklearn.metrics import balanced_accuracy_score

from sklearn.metrics import confusion_matrix

from sklearn.metrics import precision_score

from sklearn.metrics import recall_score

from sklearn.metrics import f1_score

from sklearn.metrics import classification_report
```

Metrics for classification in scikit-learn

Receiver Operating Characteristic (ROC) curve

- from sklearn.metrics import roc_curve

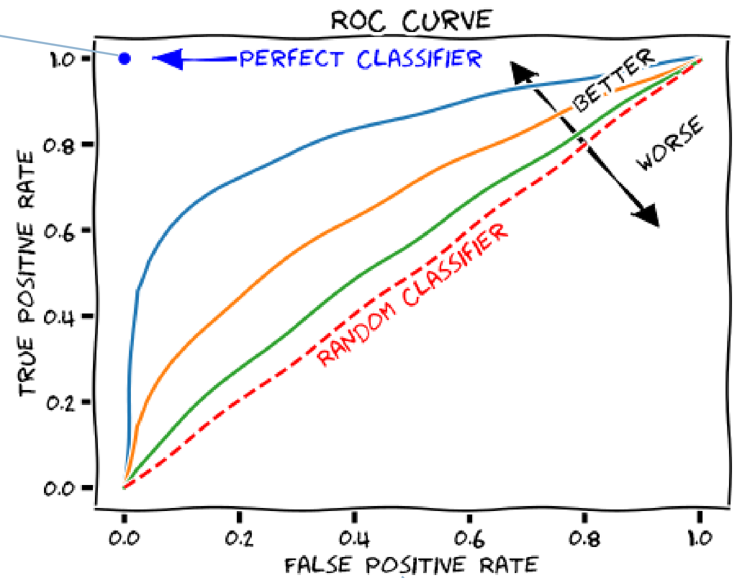
Compute Area Under the Receiver Operating Characteristic Curve (ROC AUC)

- from sklearn.metrics import roc_auc_score

Trade-offs between true positive (benefits) and false positive (costs)

100% sensitivity
(no false negatives)
100% specificity
(no false positives)

sensitivity



1 - specificity

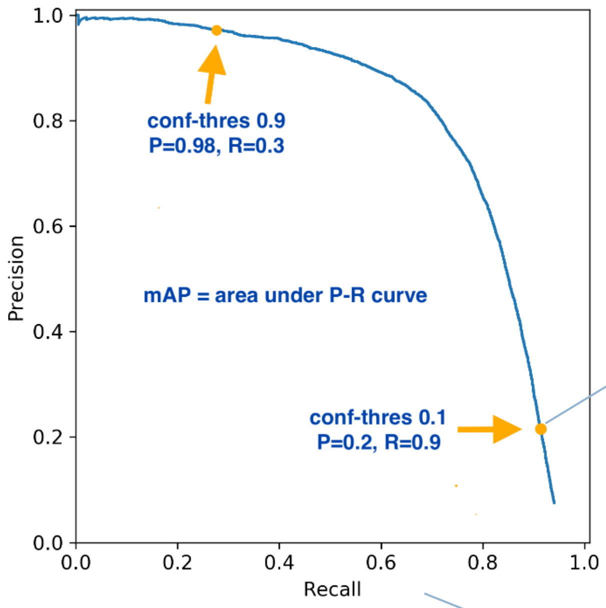
	actual_label	model_RF	model_LR
0	1	0.639816	0.531904
1	0	0.490993	0.414496
2	1	0.623815	0.569883
3	1	0.506616	0.443674
4	0	0.418302	0.369532

As you decrease the threshold you get higher TPR at the cost of a higher FPR

Metrics for classification in scikit-learn

from sklearn.metrics import precision_recall_curve

How sure you are about your positives



As you decrease the threshold you get higher recall at the cost of a lower precision (I will be not sure about my positives)

Recall = Sensitivity
how well we can identify true positives

Scikit-learn

Introduction to metrics in scikit-learn

https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course