Data Science

معالجة اللغات الطبيعية
Natural Language Processing(NLP)

تطبيق عملي
إختبار و تقييم النموذج –(خوارزمية تعلم الاله)
Model Evaluation(2)
Model Evaluation Metrics
SMS SPAM Filtering

AHMAD SHHADEH

Model Evaluation Metrics

- **Evaluation metrics** explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results.
- Model evaluation metrics are required to **quantify model performance**.
- Ideally, the estimated performance of a model **tells us how well it performs on unseen data**.
- It’s important to **understand the context** before choosing a metric because each machine learning model tries to solve a problem with a different objective using a different dataset(**Different evaluation metrics are used for different kinds of problems)**
- The choice of evaluation metrics **depends on a given machine learning task** (such as classification, regression, ranking, clustering, topic modeling..etc
- There are several evaluation metrics, like **confusion matrix, Log Loss, Gini Coefficient AUC-ROC curve, etc.** in our model will talk only about :
  - **Classification Accuracy**
  - **Confusion matrix**

Classification Accuracy

- **Classification accuracy** is simply the rate of correct classifications,(ratio of **correct predictions** to **total predictions made**)
- Classification accuracy = **correct predictions / total predictions * 100**
- The main problem with classification accuracy is that it **hides the detail you need to better understand** the performance of your classification model. There are two examples where you are most likely to encounter this problem:
  1. When the data does not have an even number of classes.(**Imbalanced data set**)
  2. When your data **has more than 2 classes**.
- Classification accuracy is a **great place to start**, but often encounters problems in practice.

Metrics Beyond Accuracy -Confusion Matrix

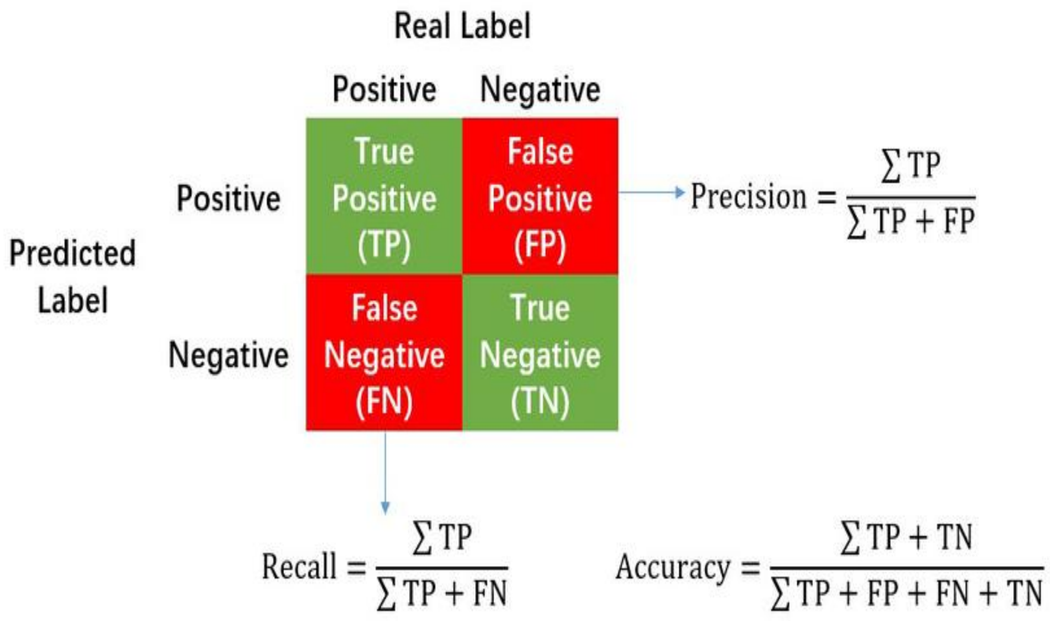
- Classification accuracy alone **can be misleading** if you have an **unequal number** of observations in each class or if you have more than two classes in your dataset.
- A confusion matrix provides **a more detailed breakdown of correct and incorrect** classifications for each class.
- Calculating a confusion matrix can give you a better **ideal/ insightful picture** of what your classification model **is getting right and what types of errors it is making,(**which classes are being predicted correctly and incorrectly**)**

		Actual Value	
		Positive	Negative
Predicted Value	Positive	TP (True Positive)	FP (False Positive)
	Negative	FN (False Negative)	TN (True Negative)

- True Positive (TP) : Observation is positive, and is predicted to be positive.
- False Negative (FN) : Observation is positive, but is predicted negative.
- True Negative (TN) : Observation is negative, and is predicted to be negative.
- False Positive (FP) : Observation is negative, but is predicted positive.

Metrics Beyond Accuracy -Confusion Matrix

1. **Accuracy** : the ratio of **correct** predictions to *total* predictions made.
   - Accuracy=(TP+TN)/(TP+TN+FN+FP)
2. **Precision:** the ratio between the **True Positives** and **all the Positives.**
   - For our problem statement, *If the spam filter says this email is spam, what's the probability that it's spam*
   - **Precision = TP/(TP+FP)**
3. **Recall:** the measure of our model **correctly identifying True Positives.**
   - The all SMSs are **Spam** ,recall tells us how many we correctly identified as **Spam**
   - **Recall =** TP / (TP + FN)



Confusion Matrix

- Number of test SMSs : ', TP+TN+FP+FN)= 10+77+8+5= 100
- Number of actual SPAM SMSs : ', TP+FN)=10+8= 18
- Number of actual HAM SMSs: : ', TN+FP)= 8+77= 82
- Number of predicted SMSs as SPAM: ', TP+FP) =10+5= 15
- Number of predicted SMSs as HAM : ', FN+TN) =85
- The number along the diagonal Green Box tell us how many of the samples **were correctly classified**
- The number Not on the diagonal Red Box tell us how many of the samples **were messed up**
- Accuracy=(TP+TN)/(TP+TN+FN+FP)= 87=100= 0.87
- Precision = TP / (TP + FP) =10/18= 10/15= 0.667
- Recall =TP / (TP + FN) = 10/18= 0.556

Sample 100		Actual	
		Spam	Ham
Predicted	Spam	TP (10)	FP (5)
	Ham	FN(8)	TN (77)

- **10 Spam SMSs** ➔ model successfully classified it as spam
- **77 Ham SMSs**➔ model successfully classified it as spam
- **8 Spam SMSs** ➔ model misclassified it by saying those are ham and should send them to the inbox
- **5 Ham SMSs**➔ model misclassified it by saying those are Spam and should send them to the Junk

Confusion Matrix

- **Accuracy**: Overall, how often does the classifier predict right?
  (TP+TN)/Total = (10+77)/100 = **87%**
- **Misclassification Rate (Error Rate):** Overall, how often is it wrong?
  (FP+FN)/Total = (5+8)/100 = **13%**
  This is the equivalent of 1 minus Accuracy.
- **Prevalence**: How often does the "yes" event occur in the sample?
  Actual Yes/Total = 18/100 = **18%**
- **True Positive Rate**: Events that were correctly predicted by the model as "occurred = Yes."
  TP/ actual yes = 10/18 = **56 %** ("**Recall**" or "**Sensitivity**")

Sample 100		Actual	
		Spam	Ham
Predicted	Spam	TP (10)	FP (5)
	Ham	FN(8)	TN (77)

- **10 Spam SMSs** ➔ model successfully classified it as spam
- **77 Ham SMSs**➔ model successfully classified it as spam
- **8 Spam SMSs** ➔ model misclassified it by saying those are ham and should send them to the inbox
- **5 Ham SMSs**➔ model misclassified it by saying those are Spam and should send them to the Junk

Confusion Matrix

- **Specificity**: When an event was actually predicated as "no," and it was actually a "no."
  TN/actual no = 77/82= **94%**
- **False Positive Rate**: Events that were predicted as "occurred = Yes," but in reality, it was "not occurred = No."
  FP/actual no = 5/82 = **6%**
- This is the equivalent of 1 minus **Specificity**.
- **Precision**: When an event is predicted "yes," how often it is correct?
  TP/predicted yes = 10/15 = **67%**

Sample 100		Actual	
		Spam	Ham
Predicted	Spam	TP (10)	FP (5)
	Ham	FN(8)	TN (77)

- **10 Spam SMSs** ➔ model successfully classified it as spam
- **77 Ham SMSs**➔ model successfully classified it as spam
- **8 Spam SMSs** ➔ model misclassified it by saying those are ham and should send them to the inbox
- **5 Ham SMSs**➔ model misclassified it by saying those are Spam and should send them to the Junk

Thank You