Lecture 7: Performance Metrics for Classification[1]

Isabel Valera

Machine Learning Group
Department of Mathematics and Computer Science
Saarland University, Saarbrücken, Germany

13.05.2024

[1] Slides by Pablo Sanchez-Martin

Outline

1. Bibliograhy
2. Overview
3. Performance measures
4. Multi-class Classification
5. Summary

Main references

- https://neptune.ai/blog/evaluation-metrics-binary-classification
- Sorower, M. S. (2010). A literature survey on algorithms for multi-label learning. Oregon State University, Corvallis, 18, 1-25.

Outline

1. Bibliograhy
2. Overview
3. Performance measures
4. Multi-class Classification
5. Summary

Setting

Binary classification:

- We consider a binary classifier $\hat{y}_i = \mathrm{sign}(f(\mathbf{x}_i) + \theta)$, where we will call θ (classification) threshold. So far, we considered $\theta = 0$.
- We know that the optimal classifier is the Bayes classifier, as it minimizes the error probability (or equivalently, the average 0-1-loss).
- Last lecture, we saw two different ways to learn/train classifiers from a dataset by minimizing a a convex surrogate loss that upper-bounds the 0-1-loss, i.e., the logistic an the squared losses.
- We assume access to a **test dataset** (with i.i.d. samples) $D = \{\mathbf{x}_i, y_i\}_{i=1}^m$, with $\mathbf{x}_i \in \mathcal{R}^d$ and $\mathbf{x}_i = \{-1, 1\}$.

Performance evaluation

How do we evaluate the learned classifier?

- Even if we use a convex surrogate loss for training, we evaluate according to error probability, i.e., to 0-1-loss.
- The average 0-1-loss evaluated on data, counts the errors made in the dataset by the classifier. However, for some applications domains (e.g., when classes are highly imbalanced), the classification error may not be useful.
- **The performance of a learning method (both for classification or regression) has to be evaluated using independent data (a.k.a. test dataset) which has not been used during learning.** It makes no sense to use the training error because we optimize for it.

Summary of performance measures

- Confusion Matrix.
- True/False positive/negative rates
- Predictive values
- Accuracy.
- F-beta score.
- Receiver Operating Characteristic (ROC) curve: (area under the curve) AUC ROC score.
- Precision-Recall (PR) curve: AUC PR score a.k.a. Average precision.
- Kolmogorov-Smirnov statistic.
- Cohen Kappa metric κ.
- Matthews Correlation Coefficient MCC.

https://neptune.ai/blog/evaluation-metrics-binary-classification

Outline

1. Bibliograhy
2. Overview
3. Performance measures
4. Multi-class Classification
5. Summary

Confusion Matrix I

The confusion matrix is a common way of presenting true positive (tp), true negative (tn), false positive (fp) and false negative (fn) predictions. The Y-axis shows the true classes while the X-axis shows the predicted classes.

	$\hat{Y} = 1$	$\hat{Y} = -1$	total cases
$Y = 1$	true positives (tp)	false negatives (fn)	$P = tp + fn$
$Y = -1$	false positives (fp)	true negatives (tn)	$N = fp + tn$
total pred.	$\hat{P} = tp + fp$	$\hat{N} = fn + tn$	$m = N + P = \hat{P} + \hat{N}$

The confusion matrix for a binary classification problem. In the table m is the number of testing data points.

Confusion Matrix II

	$\hat{Y}=1$	$\hat{Y}=-1$	total cases
$Y=1$	true positives (tp)	false negatives (fn)	$P = tp + fn$
$Y=-1$	false positives (fp)	true negatives (tn)	$N = fp + tn$
total pred.	$\hat{P} = tp + fp$	$\hat{N} = fn + tn$	$m = N + P = \hat{P} + \hat{N}$

The confusion matrix for a binary classification problem.

$$tp = \sum_{i=1}^{m} \mathbb{1}_{\hat{Y}_i=1} \mathbb{1}_{Y_i=1}, \qquad fn = \sum_{i=1}^{m} \mathbb{1}_{\hat{Y}_i=-1} \mathbb{1}_{Y_i=1},$$

$$fp = \sum_{i=1}^{m} \mathbb{1}_{\hat{Y}_i=1} \mathbb{1}_{Y_i=-1}, \qquad tn = \sum_{i=1}^{m} \mathbb{1}_{\hat{Y}_i=-1} \mathbb{1}_{Y_i=-1}.$$

Confusion Matrix III

The Confusion Matrix is the basis of all performance measures:

- the confusion matrix all other performance measures can be derived from the confusion matrix;
- includes all information about class probabilities, which is important for cost-sensitive learning;
- in each specific setting there may exist a preferable performance measure, thus when reporting the confusion matrix since, one may decide which measure to focus on.

Accuracy and error

Accuracy measures the percentage of observations that are correctly classified, i.e.,

$$ACC = \frac{tp+tn}{tp+tn+fp+fn}.$$

Thus, it is an estimator of $P(\hat{Y} = Y)$. Alternatively, the **error** measures the percentage of observations that are wrongly classified classified, i.e.

$$Error = \frac{fp+fn}{tp+tn+fp+fn}.$$

Thus, it is an estimator of $P(\hat{Y} \neq Y)$.

Observation. $ACC = 1 - Error$ in the same way as $P(\hat{Y} = Y) = 1 - P(error) = P(\hat{Y} \neq Y)$.

Note. One shouldn't use accuracy/error on imbalanced problems, as it is easy to get a high accuracy score by simply classifying all observations as the majority class.

False positive/negative rates

Type I error, a.k.a False Positive Rate (FPR), measures how many observations out of all negative observations (i.e., $N = fp + tn$) have we classified as positive, i.e. it is an estimator of $P(\hat{Y} = 1 | Y = -1)$.

$$FPR = \frac{fp}{fp+tn}$$

Type II error, a.k.a False Negative Rate (FNR), measures how many observations out of all positive observations (i.e., $P = tp + fn$) have we classified as negative, i.e., it is an estmator of $P(\hat{Y} = -1 | Y = 1)$.

$$FNR = \frac{fn}{tp+fn}$$

Observations:

- They measure how often is the classifier wrong when the true class is respectively positive or negative.
- They measure ‘error’ conditioned on a specific value for the true class.

True positive/negative rates

Sensitivity/Recall, a.k.a True Positive Rate (TPR), measures how many observations out of all positive observations (i.e., $P = tp + fn$) have we classified as positive, i.e., it is an estimator of $P(\hat{Y} = 1 | Y = 1)$.

$$TPR = \frac{tp}{tp+fn}$$

Specificity, a.k.a True Negative Rate (TNR), measures how many observations out of all negative observations (i.e., $N = fp + tn$) have we classified as negative, i.e., it is an estimator of $P(\hat{Y} = -1 | Y = -1)$.

$$TNR = \frac{tn}{fp+tn}$$

Observations:

- They measure how often the classifier is correct when the true class is respectively positive or negative.
- They measure 'accuracy' conditioned on a specific value for the true class.

Predictive values

Negative Predictive Value (NPV) measures how many predictions out of all negative predictions (i.e., $\hat{N} = tn + fn$) are negative observations, i.e., it is an estimator of $P(Y = -1|\hat{Y} = -1)$.

$$NPV = \frac{tn}{tn+fn}$$

Precision, a.k.a Positive Predictive Value (PPV), measures how many predictions out of all positive predictions (i.e., $\hat{P} = tp + fp$) are positive observations, i.e., it is an estimator of $P(Y = 1|\hat{Y} = 1)$.

$$PPV = \frac{tp}{tp+fp}$$

Observations:

- They measure how often is the classifier correct when it predicts a certain class.
- Very useful when the class probabilities are highly unbalanced, e.g., in medical applications (diagnosis of diseases).

F-beta score

F_β**-score** is the weighted harmonic mean of precision (PPV) and recall (TPR),

$$F_\beta = (1+\beta^2)\frac{PPV * TPR}{\beta^2 * PPV + TPR}$$

Note. The parameter β controls the importance of recall over precision. In particular, β is chosen such that recall is considered β times as important as precision.

The (balanced) **F-score** denotes the case when $\beta = 1$, i.e.,

$$F_1 = 2\frac{PPV * TPR}{PPV + TPR} = \frac{2tp}{2to + fp + fn}$$

ROC/PR curves

- **Problem:** Different performance measures lead to different weighting of false positive and false negatives. Such differences are most prominent when data is unbalanced.
- **Solution:** assign cost to false positives and false negatives and then optimize this cost, i.e., solve a cost-sensitive classification problem.

In a ROC/PR curve, we can integrates all possible class probabilities in one plot!

The **Receiver Operating Characteristic (ROC) curve** plots the true positive rate (TPR) versus the false positive rate (FPR) by varying the discrimination threshold $\theta \in [0, 1]$.

The **Precision-Recall (PR) curve** plots precision (PPV) versus the recall (TPR) by varying the discrimination threshold $\theta \in [0, 1]$.

AUC of ROC/PR curves

The ROC/PR curves can be summarized in one number computing the **Area Under the Curve** (AUC): the higher the ROC/PR AUC the better. Intuitively, it measures basically the quality of the ranking of a classifier:

- If all positive observations are ranked higher (i.e., get higher values of $f(X)$) than all negative ones, then $AUC = 1$.
- If all negative observations are ranked higher (i.e., get higher values of $f(X)$) than all positive ones, then $AUC = 0$.
- If the ordering is random, then $AUC = 0.5$.

Importantly, the Area Under the PR Curve is also known as **Average Precision**.

AUC of ROC/PR curves

When to use the AUC of the ROC curve?

- Do not use it heavily imbalanced datasets. Notice that the FPR for highly imbalanced datasets is pulled down due to a large number of true negatives.
- Use it when you care equally about the positive and negative classes.

When to use the AUC of the PR curve?

- The dataset is heavily imbalanced. Remember that it focuses mainly on the positive class (PPV and TPR) so it cares less about the frequent negative class.
- The positive class is more important than the negative class.

ROC-curve and AUC

ROC curve:

- change decision threshold for classifier $f : \mathcal{X} \to \mathbb{R}$,

$$\hat{y}_\theta(X) = \mathrm{sign}(f(X) + \theta),$$

- ROC-curve: plot the true positive rate versus the false positive rate by varying the discrimination threshold θ. thresholds.

Check Example with Logistic regression!

- A ROC curve can only be said to be better than another curve if the curve always lies above the second one.
- Alternatively, we can summarize the ROC-curve with a single number, using the **Area under the Curve (AUC)**.

Area under the curve

Intuitively, the AUC measures the **quality of the ranking** of a classifier.

- all positive samples. are ranked higher than all negative ones, then $\mathrm{AUC}= 1$,
- all negative samples. are ranked higher than all positive ones, then $\mathrm{AUC}= 0$,
- If the ordering is random, then the expectation of the AUC is 0.5.

Definition

Let us denote by $x_1^+, \ldots, x_P^+$ and $x_1^-, \ldots, x_N^-$ the set of points from positive and negative class in the test set (drawn i.i.d.) and a classification function $f : \mathcal{X} \to \mathbb{R}$. Then the **AUC** is defined as,

$$\mathrm{AUC} = \frac{1}{NP} \sum_{j=1}^{N} \sum_{i=1}^{P} \mathbb{1}_{f(x_i^+) > f(x_j^-)}.$$

AUC properties

Proposition

Let X^+ be distributed as $\mathrm{P}(X|Y=1)$ and X^- as $\mathrm{P}(X|Y=-1)$, then

$$\mathbb{E}[\,\mathrm{AUC}] = \mathrm{P}\big(f(X^+) > f(X^-)\big).$$

Proof: Decompose test sample X_{test} into the samples $X_1^+, \dots, X_p^+$ and $X_1^-, \dots, X_{m-p}^-$ from positive and negative class. Note, that

$$\mathbb{E}_{\{(X_1,Y_1),(X_2,Y_2),\dots,(X_m,Y_m)\}}[\,\mathrm{AUC}] = \mathbb{E}_p[\mathbb{E}_{\{(X_1^+,\dots,X_p^+),(X_1^-,\dots,X_{m-p}^-)\}}[\,\mathrm{AUC}\,|\,p]],$$

where $p \sim \mathrm{Bin}(m, \mathrm{P}(Y=1))$ and X_i^+ and X_j^- are i.i.d. samples from $\mathrm{P}(X|Y=1)$ and $\mathrm{P}(X|Y=-1)$. Thus,

$$\begin{aligned}\mathbb{E}[\,\mathrm{AUC}] &= \mathbb{E}_p[\frac{1}{(m-p)p}\sum_{i=1}^{p}\sum_{j=1}^{m-p}\mathbb{E}_{X_i^+,X_j^-}\mathbb{1}_{f(X_i^+)>f(X_j^-)}\,|\,p] \\ &= \mathbb{E}_p[\frac{1}{(m-p)p}(m-p)p\,\mathrm{P}\big(f(X^+) > f(X^-)\big)\,|\,p] = \mathrm{P}\big(f(X^+) > f(X^-)\big),\end{aligned}$$

where we have used that the pairs X_i^+ and X_j^- are i.i.d.

Example - SVM separable test data

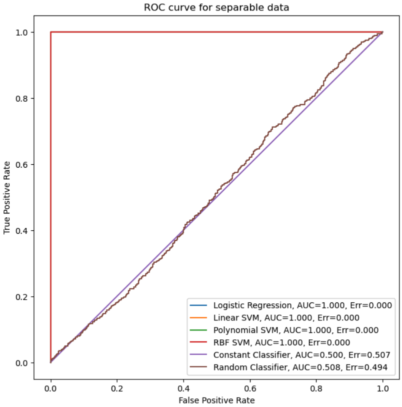

Figure: When the problem is separable one can easily find classifiers with $AUC = 1$. Classifiers 4 corresponds to a constant classifier and classifier 5 correspond to a random classifier

Example - SVM non-separable test data

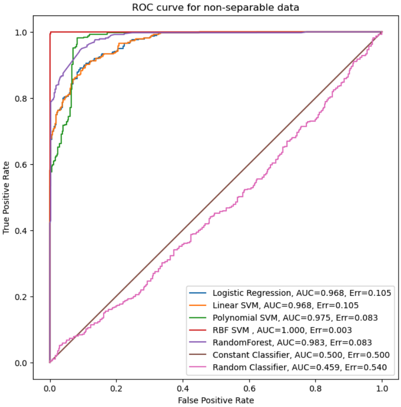

Figure: When the problem is not separable anymore - ROC-curves come closer to the random baseline.

Outline

1. Bibliograhy
2. Overview
3. Performance measures
4. Multi-class Classification
5. Summary

Multi-class Setting

Multi-class classification: We consider the multi-class classifier $\hat{Y}(X) \in 1 \ldots, K$. Additionally, we consider a test dataset $D = \{x_i, y_i\}_{i=1}^{m}$.

- Generalization of binary measures to the multi-class case can be difficult.
- **Class balanced error:** Let x_j^k denote the samples from class k and let N_k be the cardinality of class k in the test set, then

$$\mathrm{Error}_{\mathrm{balanced}} = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{N_k} \sum_{j=1}^{N_k} \mathbb{1}_{\hat{Y}(x_j^k) \neq k}.$$

Confusion matrix

Now the confusion matrix is a $K \times K$ matrix:

	prediction class 1	$\dots$	prediction class K	total cases
pred. class 1	$\lvert\{\hat{y}_i = 1, y_i = 1\}\rvert$	$\dots$	$\lvert\{\hat{y}_i = K, y_i = 1\}\rvert$	$\lvert\{y_i = 1\}\rvert$
$\vdots$	$\vdots$		$\vdots$	$\vdots$
pred. class K	$\lvert\{\hat{y}_i = 1, y_i = K\}\rvert$	$\dots$	$\lvert\{\hat{y}_i = K, y_i = K\}\rvert$	$\lvert\{y_i = K\}\rvert$
total pred.	$\lvert\{\hat{y}_i = 1\}\rvert$	$\dots$	$\lvert\{\hat{y}_i = K\}\rvert$	

Table: The confusion matrix for a multi-class problem.

Alternatively, we can compute a confusion matrix for each class $k \in \{1, \dots, K\}$ such that the k-th confusion matrix considers the class k as the positive class and all the other classes, i.e., $j \neq k$, as the negative class.

Macro/micro averaging

Macro averaging. Compute the metric independently for each class k and then take the average. Hence, macro averaging treats all classes with equal importance.
Micro averaging. Compute the metric globally over all observations and classes. Micro averaging is preferable with imbalance datasets.

Comparison using Precision (a.k.a Positive Predictive Value):

$$PPV_{macro} = \frac{1}{K} \sum_{j=1}^{K} \frac{\sum_{i=1}^{m} \mathbb{1}_{y_i=j} \mathbb{1}_{\hat{y}_i=j}}{\sum_{i=1}^{n} \mathbb{1}_{\hat{y}_i=j}}$$

$$PPV_{micro} = \frac{\sum_{j=1}^{K} \sum_{i=1}^{n} \mathbb{1}_{y_i=j} \mathbb{1}_{\hat{y}_i=j}}{\sum_{j=1}^{K} \sum_{i=1}^{n} \mathbb{1}_{\hat{y}_i=j}},$$

If you are interested in knowing more....

- Evaluating partitions.
  - Example-based: exact match ratio (MR), accuracy (A), precision (P), recall (R), F-beta measure (F_β), and hamming loss (HL).
  - Label-based: macro/micro averaged measures and α-evaluation.
- Evaluating ranking.
  - One error (0), ranking loss (RL), and average precision (AP).
- Label hierarchy.
  - Extended hamming loss.

Sorower, M. S. (2010). A literature survey on algorithms for multi-label learning. Oregon State University, Corvallis, 18, 1-25.

Outline

1 Bibliograhy

2 Overview

3 Performance measures

4 Multi-class Classification

5 Summary

Golden rule for your machine learning problem

Carefully think about what is the correct performance measure/cost matrix for your problem and then optimize that and not something else!