TECHNICAL SPECIFICATION

ISO/IEC TS 4213

First edition
2022-10

Information technology — Artificial intelligence — Assessment of machine learning classification performance

Technologies de l'information — Intelligence artificielle — Evaluation des performances de classification de l'apprentissage machine

Reference number
ISO/IEC TS 4213:2022(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 42, *Artificial intelligence*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

As academic, commercial and governmental researchers continue to improve machine learning models, consistent approaches and methods should be applied to machine learning classification performance assessment.

Advances in machine learning are often reported in terms of improved performance relative to the state of the art or a reasonable baseline. The choice of an appropriate metric to assess machine learning model classification performance depends on the use case and domain constraints. Further, the chosen metric can differ from the metric used during training. Machine learning model classification performance can be represented through the following examples:

- A new model achieves 97,8 % classification accuracy on a dataset where the state-of-the-art model achieves just 96,2 % accuracy.
- A new model achieves classification accuracy equivalent to the state of the art but requires much less training data than state-of-the-art approaches.
- A new model generates inferences 100x faster than state-of-the-art models while maintaining equivalent accuracy.

To determine whether these assertions are meaningful, aspects of machine learning classification performance assessment including model implementation, dataset composition and results calculation are taken into consideration. This document describes approaches and methods to ensure the relevance, legitimacy and extensibility of machine learning classification performance assertions.

Various AI stakeholder roles as defined in ISO/IEC 22989:2022, 5.17 can take advantage of the approaches and methods described in this document. For example, AI developers can use the approaches and methods when evaluating ML models.

Methodological controls are put in place when assessing machine learning performance to ensure that results are fair and representative. Examples of these controls include establishing computational environments, selecting and preparing datasets, and limiting leakage that potentially leads to misleading classification results. Clause 5 addresses this topic.

Merely reporting performance in terms of accuracy can be inappropriate depending on the characteristics of training data and input data. If a classifier is susceptible to majority class classification, grossly unbalanced training data can overstate accuracy by representing the prior probabilities of the majority class. Additional measurements that reflect more subtle aspects of machine learning classification performance, such as macro-averaged metrics, are at times more appropriate. Further, different types of machine learning classification, such as binary, multi-class and multi-label, are associated with specific performance metrics. In addition to these metrics, aspects of classification performance such as computational complexity, latency, throughput and efficiency can be relevant. Clause 6 addresses these topics.

Complications can arise as a result of the distribution of training data. Statistical tests of significance are undertaken to establish the conditions under which machine learning classification performance differs meaningfully. Specific training, validation and test methodologies are used in machine learning model development to address the range of potential scenarios. Clause 7 addresses these topics.

Annex A illustrates calculation of multi-class classification performance, using examples of positive and negative classifications. Annex B illustrates a receiver operating characteristic (ROC) curve derived from example data in Annex A.

Annex C summarizes results from machine learning classification benchmark tests.

Annex D discusses a chance-corrected cause-specific mortality fraction, a machine learning classification use case. Apart from these, this document does not address any issues related to benchmarking, applications or use cases.

TECHNICAL SPECIFICATION **ISO/IEC TS 4213:2022(E)**

Information technology — Artificial intelligence — Assessment of machine learning classification performance

1 Scope

This document specifies methodologies for measuring classification performance of machine learning models, systems and algorithms.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 22989:2022, *Information technology — Artificial intelligence — Artificial intelligence concepts and terminology*

ISO/IEC 23053:2022, *Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML)*

3 Terms and definitions

For the purposes of this document, the terms and definitions in ISO/IEC 22989:2022, ISO/IEC 23053:2022, and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1 Classification and related terms

3.1.1
classification
method of structuring a defined type of item (objects or documents) into classes and subclasses in accordance with their characteristics

[SOURCE: ISO 7200:2004, 3.1]

3.1.2
classifier
trained model and its associated mechanism used to perform *classification* (3.1.1)

3.2 Metrics and related terms

3.2.1
evaluation
process of comparing the *classification* (3.1.1) predictions made by the model on data to the actual labels in the data

3.2.2

false negative

miss

type II error

F_N

sample wrongly classified as negative

3.2.3

false positive

false alarm

type I error

F_P

sample wrongly classified as positive

3.2.4

true positive

T_P

sample correctly classified as positive

3.2.5

true negative

T_N

sample correctly classified as negative

3.2.6

accuracy

number of correctly classified samples divided by all classified samples

Note 1 to entry: It is calculated as $a = (T_P + T_N)/(T_P + F_P + T_N + F_N)$.

3.2.7

confusion matrix

matrix used to record the number of correct and incorrect *classifications* (3.1.1) of samples

3.2.8

F_1 score

F-score

F-measure

F_1-measure

harmonic mean of *precision* (3.2.9) and *recall* (3.2.10)

Note 1 to entry: It is calculated as $F_1 = 2T_P/(2T_P + F_P + F_N)$.

3.2.9

precision

positive predictive value

number of samples correctly classified as positive divided by all samples classified as positive

Note 1 to entry: It is calculated as $p = T_P/(T_P + F_P)$.

3.2.10

recall

true positive rate

sensitivity

hit rate

number of samples correctly classified as positive divided by all positive samples

Note 1 to entry: It is calculated as $r = T_P/(T_P + F_N)$.

3.2.11
specificity
selectivity
true negative rate
number of samples correctly classified as negative divided by all negative samples

Note 1 to entry: It is calculated as $s = T_N / (T_N + F_P)$.

3.2.12
false positive rate
fall-out
number of samples incorrectly classified as positive divided by all negative samples

Note 1 to entry: It is calculated as $F_{P,R} = F_P / (F_P + T_N)$.

3.2.13
cumulative response curve
gain chart
graphical method of displaying *true positive rates* (3.2.10) and percentage of positive prediction in the total data across multiple thresholds

3.2.14
lift curve
graphical method of displaying on the y-axis the ratio of *true positive rate* (3.2.10) between the model and a random classifier, and on the x-axis the percentage of positive predictions in the total data across multiple thresholds

3.2.15
precision recall curve
PRC
graphical method for displaying *recall* (3.2.10) and *precision* (3.2.9) across multiple thresholds

Note 1 to entry: A PRC is more suitable than a ROC (receiver operating characteristic) curve for showing performance with imbalanced data.

3.2.16
receiver operating characteristic curve
ROC curve
graphical method for displaying *true positive rate* (3.2.10) and *false positive rate* (3.2.12) across multiple thresholds

3.2.17
cross-validation
method to estimate the performance of a machine learning method using a single dataset

Note 1 to entry: Cross-validation is typically used for validating design choices before training the final model.

3.2.18
majority class
class with the most samples in a dataset

4 Abbreviated terms

AI	artificial intelligence
ANOVA	analysis of variance
AUPRC	area under the precision recall curve

AUROC	area under the receiver operating characteristic curve
CLT	central limit theorem
CPU	central processing unit
CRC	cumulative response curve
FC	fully connected
FDR	false discovery rate
IoU	intersection over union
GPU	graphics processing unit
ROC	receiver operating characteristic

5 General principles

5.1 Generalized process for machine learning classification performance assessment

A generalized process for machine learning classification performance assessment is shown in Figure 1.

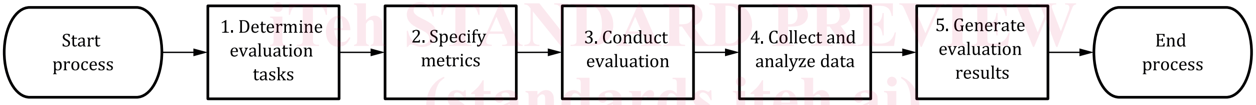
Start process
1. Determine evaluation tasks
2. Specify metrics
3. Conduct evaluation
4. Collect and analyze data
5. Generate evaluation results
End process

Figure 1 — Generalized process for machine learning classification performance assessment

Step 1: Determine evaluation tasks

Determine the appropriate classification task or tasks for the evaluation.

Step 2: Specify metrics

Based on the classification task, specify the required metric or metrics.

Step 3: Conduct evaluation

Create the evaluation plan, implement the evaluation environment including software and hardware, prepare datasets and process datasets.

Step 4: Collect and analyse data

According to the specified metrics, collect model outputs such as classification predictions for each sample.

Step 5: Generate evaluation results

Generate evaluation results based on specified metrics and other relevant information.

5.2 Purpose of machine learning classification performance assessment

The purpose of the assessment and its baseline requirements can vary greatly depending on whether it applies to the "design and development" or "verification and validation” stage.

The purpose of assessment during the “design and development” stage is to optimize hyperparameters to achieve the best classification performance. The purpose of assessment during the "verification and validation" stage is to estimate the trained model performance.

Performance assessment can be applied for several purposes, including:

- model assessment, to know how good the model is, how reliable the model’s predictions are, or the expected frequency and size of errors;
- model comparison, to compare two or more models in order to choose between them;
- out-of-sample and out-of-time comparisons, to check that performance has not degraded with new production data.

5.3 Control criteria in machine learning classification performance assessment

5.3.1 General

When assessing machine learning classification performance, consistent approaches and methods should be applied to demonstrate relevance, legitimacy and extensibility. Special care should be taken in comparative assessments of multiple machine learning classification models, algorithms or systems to ensure that no approach is favoured over another.

5.3.2 Data representativeness and bias

Except when done for specific goal-relevant reasons, the training and test data should be as free of sampling bias as possible. That is, the distribution of features and classes in the training data should be matched to their distribution in the real world to the extent possible. The training data does not need to match the eventual use case exactly. For example, in the case of self-driving cars, it can be acceptable to assess the classification performance of machine learning models trained on closed-circuit tracks rather than on open roads for prototype systems. The data used to test a machine learning model should be representative of the intended use of the system.

Data can be skewed, incomplete, outdated, disproportionate or have embedded historical biases. Such unwanted biases can propagate biases present in the training data and are detrimental to model training. If the machine learning operating environment is complex and nuanced, limited training data will not necessarily reflect the full range of input data. Moreover, training data for a particular task is not necessarily extensible to different tasks. Extra care should be taken when splitting unbalanced data into training and test to ensure that similar distributions are maintained between training data, validation data and test data.

Data capture bias can be based on both the collection device and the collector’s preferences. Label biases can occur if categories are poorly defined (e.g. similar images can be annotated with different labels while, due to in-class variability, the same labels can be assigned to visually different images). For more information on bias in AI systems, see ISO/IEC TR 24027[1].

5.3.3 Preprocessing

Special care should be taken in preprocessing and its impact on performance assessment, especially in the case of comparative assessment. Depending on the purpose of the evaluation, inconsistent preprocessing can lead to biased interpretation of the results. In particular, when preprocessing favours one model over another, their performance gap should not be attributed to the downstream algorithms. Examples of preprocessing include removal of outliers, resolving incomplete data or filtering out noise.

5.3.4 Training data

Special care should be taken in the choice of training and validation data and how the choice impacts performance assessment, especially in the case of comparative assessment. Depending on the purpose of the evaluation, the use of different training data can lead to a biased interpretation of the results. In particular, in such cases any performance gap should be attributed to the combination of the algorithm and training data, rather than to just the algorithm.

In the context of model comparison, the training data used to build the respective models can differ. One can take two models, trained on different training data, and evaluate them against each other on the same test data.

5.3.5 Test and validation data

The data used to test a machine learning model shall be the same for all machine learning models being compared. The test and validation data shall contain no samples that overlap with training data.

5.3.6 Cross-validation

Cross-validation is a method to estimate the performance of a machine learning method using a single dataset.

The dataset is divided into k segments, where one segment is used for test while the rest is used for training. This process is repeated k times, each time using another segment as the test set. When k is equal to N, the size as the dataset, this is called leave-one-out cross-validation. When k is smaller than N, this is called k-fold cross-validation.

It can be of interest to compare the performance of different cross-validation techniques when all other variables are controlled. However, models whose performance is being compared should not use different cross-validation techniques (e.g. it is not appropriate to compare Model A k-fold cross-validation results against the mean of Model B single train-test split results).

The primary use of cross-validation is for validating design choices such as hyperparameter values, by comparing their overall effect on various models. That is why it is typical to retrain a model on the full dataset after that validation, using the hyperparameters that performed best on average. However, cross-validation does not provide a performance assessment of that final model, and extrapolating performance from the output of cross-validation is a rough approximation with no guarantee of faithfulness.

Another use of cross-validation is for comparative evaluation of machine learning algorithms, without subsequently training a final model. An algorithm is considered to outperform another if on average its resulting models perform best.

5.3.7 Limiting information leakage

Information leakage occurs when a machine learning algorithm uses information not in the training data to create a machine learning model.

Information leakage is often caused when training data includes information not available during production. In an evaluation, information leakage can result in a machine learning model's classification accuracy being overstated. A model trained under these conditions will typically not generalize well.

Evaluations should be designed to prevent information leakage between training and test data.

EXAMPLE A machine learning model can be designed to classify between native and non-native Spanish speakers, using multiple audio samples from each subject. Some observation features, such as vowel enunciation, are potentially useful for this type of speaker classification. However, such features can also be used to identify the specific speaker. The model can use identity-based information to accurately classify test data, even though this information would not be available in production systems. The solution would be to not include the same subject in both training and test data, even if the training and test samples differ.

5.3.8 Limiting channel effects

A channel effect is a characteristic of data that reflects how data were collected as opposed to what data were collected. Channel effects can cause machine learning classification algorithms to learn irrelevant characteristics from training data as opposed to relevant content, which in turn can lead to poor machine learning classification performance.

Channel effects can be caused by the mechanism used to acquire data, preprocessing applied to data, the identity of the individual obtaining data, and environmental conditions under which data were acquired, among other factors.

The data should be as free of channel effects as possible. Controlling channel effects in training data contributes to better performance. Controlling channel effects in test data enables higher-quality assessments.

NOTE One method of reducing channel effects is to balance channel distributions for each class in the data.

Reporting should describe known channel effects introduced to the training data. Channel effects should be accounted for during statistical significance testing (see Clause 7).

EXAMPLE A vision-based system can be designed to distinguish between images of cats and dogs. However, if all “cat” images are high-resolution, and all “dog” images are low-resolution, a machine learning classifier can learn to classify images based on resolution as opposed to content.

5.3.9 Ground truth

Ground truth is the value of the target variable for a particular item of labelled input data. Cleanliness in ground truth can affect classification performance measurement. When assessing classification performance, a strong generalizable ground truth should be established.

General agreement on an aggregated ground truth can be quantified using measurements of agreement such as Cohen's kappa coefficient.

In some domains (e.g. medical), inter-annotator variation can be significant, especially in tasks where team-based consensus is involved.

5.3.10 Machine learning algorithms, hyperparameters and parameters

Most machine learning algorithms have characteristics that affect their learning processes, known as hyperparameters. Machine learning algorithms use hyperparameters and training data to establish internal parameters. The manner in which these parameters are computed can vary. For example, generative algorithms can optimize parameters such that the probability of the available training data is maximized, whereas discriminative algorithms can optimize parameters to maximize classification accuracy.

Hyperparameter types should be reported for all machine learning algorithms in an assessment, as well as hyperparameter values for each machine learning model.

Hyperparameter selection bias should be taken into account when machine learning models are compared. Different machine learning algorithms can have different numbers of hyperparameters with different adjustment capabilities. The degree of overfitting in the training process can then differ across machine learning algorithms.

This is especially pronounced in deep learning with its many combinations of architectures, activation functions, learning rates and regularization parameters. No information from the test set shall be used when adjusting hyperparameters, as this typically leads to over-optimistic performance estimation. When label information is needed for such tuning, it is typically drawn from a separate set of data, called the validation set, which is disjoint from the test set.

This challenge can be addressed through approaches such as nested cross-validation. In this training process, an outer loop measures prediction performance while an inner loop adjusts the hyperparameters of the individual models. In this fashion, methods can choose optimal settings for building predictive models in the outer loop.

See Annex C for summary information on selected machine learning classification benchmark tests, including model parameters and values associated with performance against various datasets.