Validation Reloaded –

Elevate Your Model Validation with Advanced Toolkits

MICCAI Educational Challenge 2024 Submission

Original Medium blog post can be found at https://medium.com/miccai-educational-initiative/validation-reloaded-elevate-your-model-validation-with-advanced-toolkits-2d67d07d2460

Please note:
The original Medium blog post was converted into a PDF for submission.

The original blog post contains both GIFs and videos, which cannot be displayed correctly in a PDF file. For the PDF, we have replaced the GIFs with screenshots and added links to the videos.

For better readability, we recommend reading the original blog post under the link above.

ANNIKA REINKE

A. EMRE KAVUR

German Cancer Research Center (DKFZ) and HI Helmholtz Imaging

Validation Reloaded — Elevate Your Model Validation with Advanced Toolkits

Annika Reinke · Follow
Published in MICCAI Educational Initiative
16 min read · 1 hour ago

Annika Reinke (a) and A. Emre Kavur (a, b)

(a) German Cancer Research Center (DKFZ) Heidelberg, Div. Intelligent Medical Systems and HI Helmholtz Imaging, Germany

(b) German Cancer Research Center (DKFZ) Heidelberg, Div. Medical Image Computing, Germany

Validating artificial intelligence (AI) algorithms is often seen as something boring — something that you do on the side while you focus on developing new models to improve your AI even more. But if you want to improve your model precisely, you need to measure that improvement. The only way to do that is through proper, problem-tailored validation. In current research, there is an overemphasis on developing new AI architectures, but a lack of rigorous comparisons and thorough validation. Validation may be underestimated, but is essential to prove that your algorithm does what it is supposed to do.

Imagine your goal is to find the winner of a marathon. The scoring would be quite simple: you measure the time it takes each runner to finish the marathon, sort the times in descending order and the best runner is the one with the shortest time. Now imagine that someone used shoe size as a measure instead of time. It wouldn't make sense, would it? Similarly, it is important to choose the right validation scheme for AI algorithms — especially since we cannot fully trust that our algorithms are doing the right things, and especially for critical applications such as medicine.

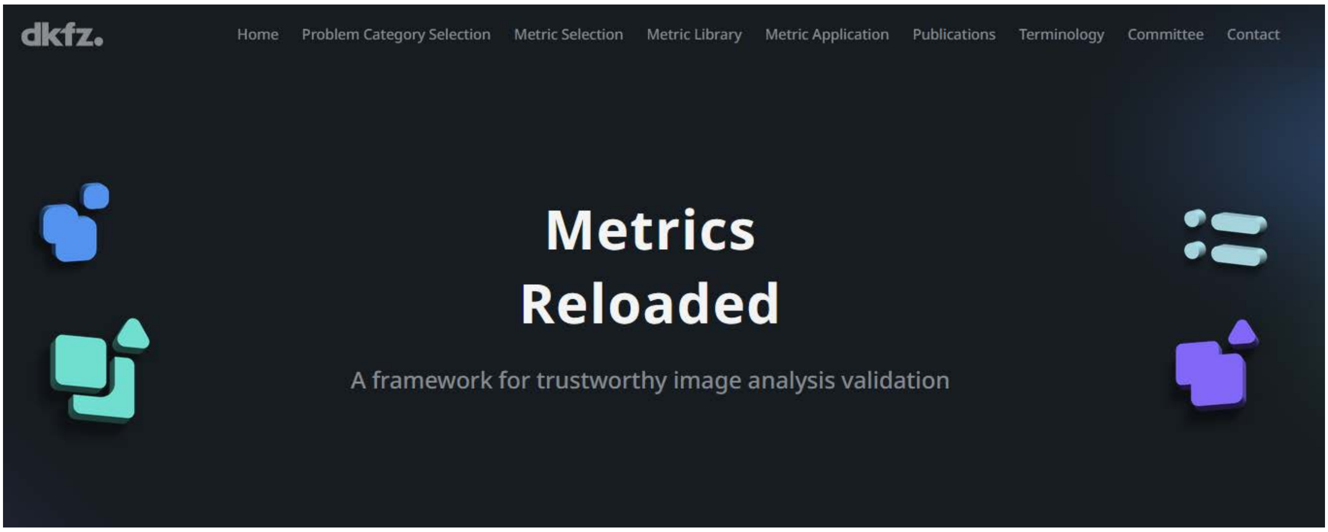
Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Imagine an AI that needs to recognize whether a patient has recovered from a brain tumor. If you don't choose the right validation metric, it may make incorrect predictions, for example missing small tumor lesions and telling the patient that (s)he has fully recovered — this would have dramatic consequences.

In recent years, theory has been developed to address these issues in validation. Large groups of researchers have worked on problem-aware metric recommendations [1, 2, 3] and advanced analysis and visualization tools for benchmarking [4] *(note: there are other outstanding publications along these lines, but we will focus on these two topics)*. While these were great achievements towards better research practices, the theory is not easy to understand. That's why we have been working on making the theory easy to use for the community by creating online toolkits that *are* easy to use. The good thing is that there is no need to worry or fully understand the theory — you are automatically guided through the process :) In this blog post, we will introduce you to two toolkits: **Metrics Reloaded** for selecting appropriate metrics for your research problem and **Rankings Reloaded** for robustly ranking your method against other models or hyperparameter settings.

. . .

Metrics Reloaded Toolkit



https://metrics-reloaded.dkfz.de/

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

We have shown that choosing wrong metrics can have serious consequences. If you are interested in an overview of metric pitfalls, we suggest to check out [1, 2, 5]. A large expert consortium has collected tons of metric-related problems. Based on these pitfalls, they came up with a list of properties that are important when choosing metrics. For example, some metrics cannot handle small structures, others focus only on overlap and ignore object shapes, while some metrics cannot deal with class imbalances. We call the selection of all these properties a *problem fingerprint*. Based on the fingerprint, the experts developed decision trees that help **to find the most appropriate set of metrics for a specific research problem** [4]. And thanks to this concept, the **problem-aware metric recommendations** are domain-agnostic.

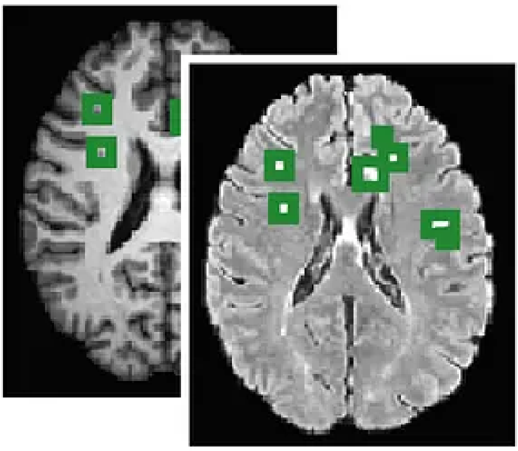
While the original paper [4] is very comprehensive and puts a lot of emphasis on understanding the theory, it is also very long (>200 pages with appendices!). It is a great resource to get a deep understanding of validation, but it also makes it a challenge!

The topic of choosing proper metrics is crucial, so we decided to make it as easy to use as possible — by implementing an online toolkit. **The beauty about the toolkit is that you don't have to worry about the theory, you don't have to read the whole paper — you just answer questions related to your dataset and research question and you'll get an appropriate set of metrics**! Let's guide you through an example to showcase how the toolkit works.

Note: The current metric recommendation framework (and toolkit) covers the most commonly addressed tasks semantic/instance segmentation, object detection, and image-level classification. Spoiler: We are already in the process of defining recommendations for other tasks such as image synthesis. Stay tuned!

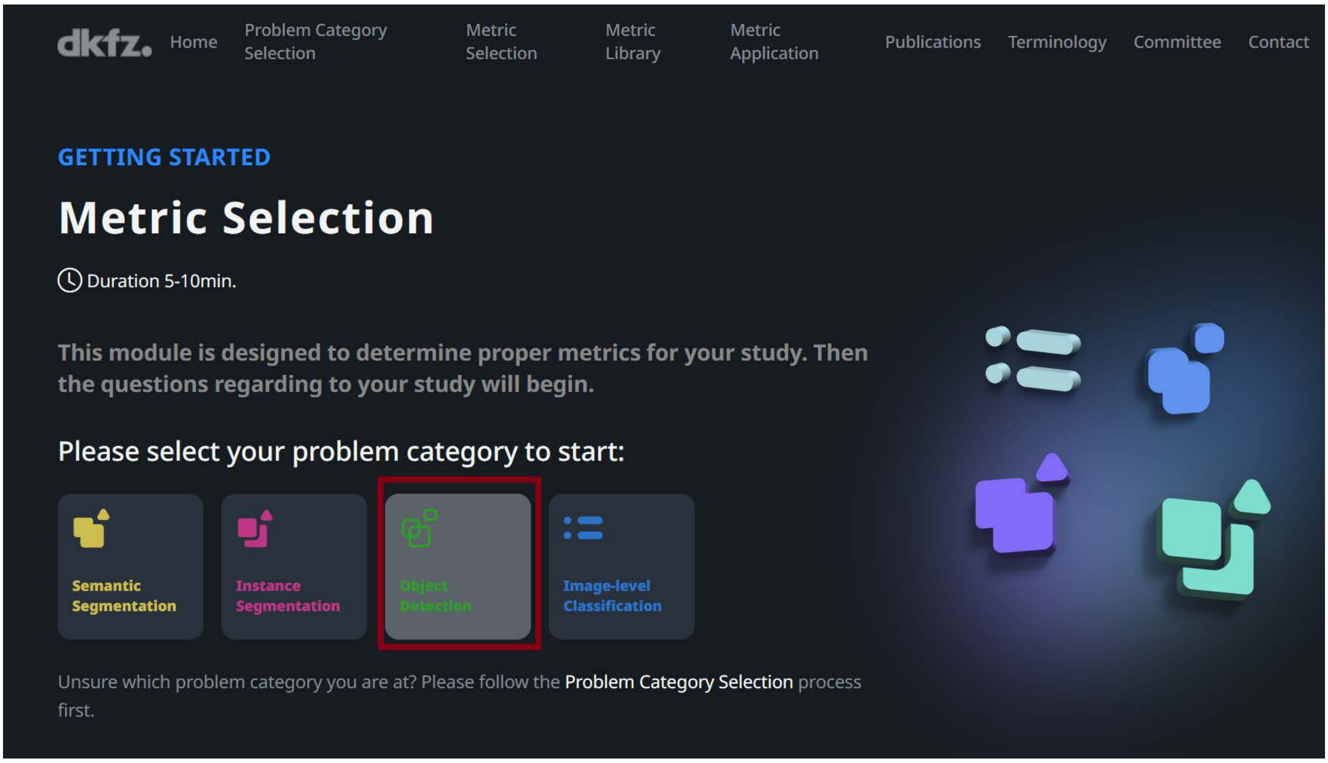
Example use case

Let's focus on a medical example: Detection of multiple sclerosis (MS) lesion in multimodal brain Magnetic Resonance Imaging (MRI) images. In this scenario, we want to localize small lesions from brain MRI scans, such as shown in the example below (from [3]).



Example image for MS lesion detection from multimodal brain images [3, 7, 8] (see use case ObD-2 in [3]).

The toolkit will guide you through various steps and metrics, so let's get started. First, click the ***Metric Selection*** link in the top menu to launch the toolkit, then select 'Object Detection'.



GETTING STARTED

Metric Selection

Duration 5-10min.

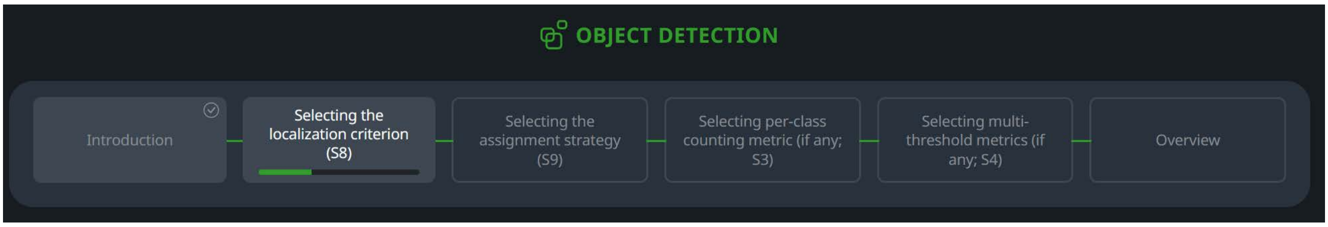
This module is designed to determine proper metrics for your study. Then the questions regarding to your study will begin.

Please select your problem category to start:

Unsure which problem category you are at? Please follow the Problem Category Selection process first.

An overview of the process is shown at the top of the toolkit webpage. The toolkit follows each of these steps and provides recommendations for each section (where applicable).

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF


OBJECT DETECTION
Introduction
Selecting the localization criterion (S8)
Selecting the assignment strategy (S9)
Selecting per-class counting metric (if any; S3)
Selecting multi-threshold metrics (if any; S4)
Overview

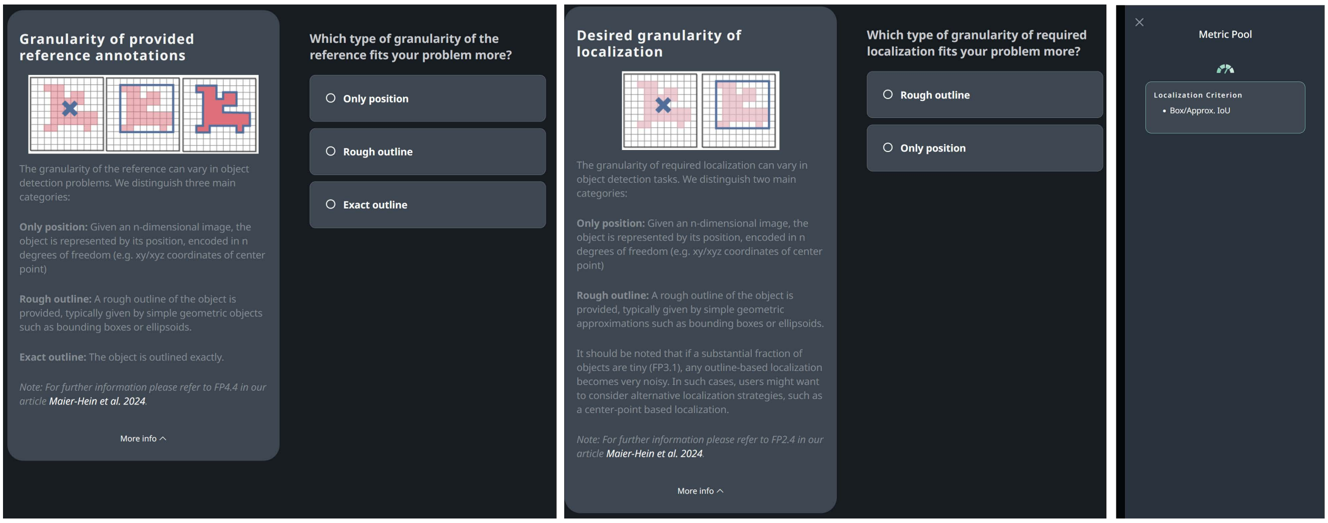
In object detection, the first step is to select a localization criterion, i.e., to decide whether a prediction matches a reference object. Let’s talk about this step in a bit more detail.

The first question asks about the granularity of the reference annotations provided, i.e., whether the reference is provided as an exact outline, a rough outline (e.g., bounding box), or just a position (e.g., center point). As you can see in the video below, each question is accompanied by a more detailed description. In our use case, references are provided as segmentation masks, so we select “exact outline”.

Next, we are asked about the desired granularity of the predicted localization, i.e., do we need something like a bounding box (“rough outline”) or is a center point sufficient to localize the objects (“only position”). In our use case, we need bounding boxes, so a rough outline.

That’s it for the first part! Based on our answers, the toolkit directly recommended the **Box/Approx. IoU**, i.e., the Intersection over Union of the bounding boxes. The toolkit also informs us on how to choose a localization threshold, so based on which threshold we count a prediction as True Positive (TP) or False Positive (FP). It provides pros and cons of lower and higher thresholds, so that we can decide based on our current situation. In our case, since we have mostly small structures, we should choose a small localization threshold such as 0.1 or 0.3.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



Granularity of provided reference annotations

The granularity of the reference can vary in object detection problems. We distinguish three main categories:

Only position: Given an n-dimensional image, the object is represented by its position, encoded in n degrees of freedom (e.g. xy/xyz coordinates of center point)

Rough outline: A rough outline of the object is provided, typically given by simple geometric objects such as bounding boxes or ellipsoids.

Exact outline: The object is outlined exactly.

Note: For further information please refer to FP4.4 in our article Maier-Hein et al. 2024.

Which type of granularity of the reference fits your problem more?

○ Only position

○ Rough outline

○ Exact outline

Desired granularity of localization

The granularity of required localization can vary in object detection tasks. We distinguish two main categories:

Only position: Given an n-dimensional image, the object is represented by its position, encoded in n degrees of freedom (e.g. xy/xyz coordinates of center point)

Rough outline: A rough outline of the object is provided, typically given by simple geometric approximations such as bounding boxes or ellipsoids.

It should be noted that if a substantial fraction of objects are tiny (FP3.1), any outline-based localization becomes very noisy. In such cases, users might want to consider alternative localization strategies, such as a center-point based localization.

Note: For further information please refer to FP2.4 in our article Maier-Hein et al. 2024.

Which type of granularity of required localization fits your problem more?

○ Rough outline

○ Only position

Metric Pool

Localization Criterion

- Box/Approx. IoU

Selecting the localization criterion for MS lesion detection in multimodal brain MRI images. After answering questions related to the granularity of the reference and the required granularity of the prediction, the toolkit provides you with the most appropriate localization criterion (here: Box/Approx. IoU).

As the localization does not necessarily lead to unambiguous matches, an assignment strategy must be chosen to resolve potential ambiguities. The recommended assignment strategy depends on the availability of continuous class scores, the possibility of overlapping predictions as well as on whether double assignments should be penakized, so the toolkit asks you exactly those questions. Based on our answers, the framework recommends the most appropriate strategy, which for our use case will be **Greedy (by Score) Matching**, i.e., ranking all predictions by their predicted class scores and iteratively assigning them to the reference object with the highest localization criterion for this prediction. It is also recommended to penalize double assignments as FPs.

Similarly, we are guided through the remaining steps of the Metrics Reloaded framework. In the next step, we select metrics that assess the concrete detection performance. In object detection, we typically deal with multiple classes, so we recommend a pre-class assessment where each class of interest is assessed separately, preferably in a “one-versus-the-rest” fashion. The choice depends primarily on the cutoff (decision rule) strategy and the distribution of the classes. What is a decision rule? Well, modern algorithms output (continuous) predicted class scores. However, in order to classify cases in an actual application (i.e., to make actual decisions), it is necessary to apply a decision rule to the scores; this

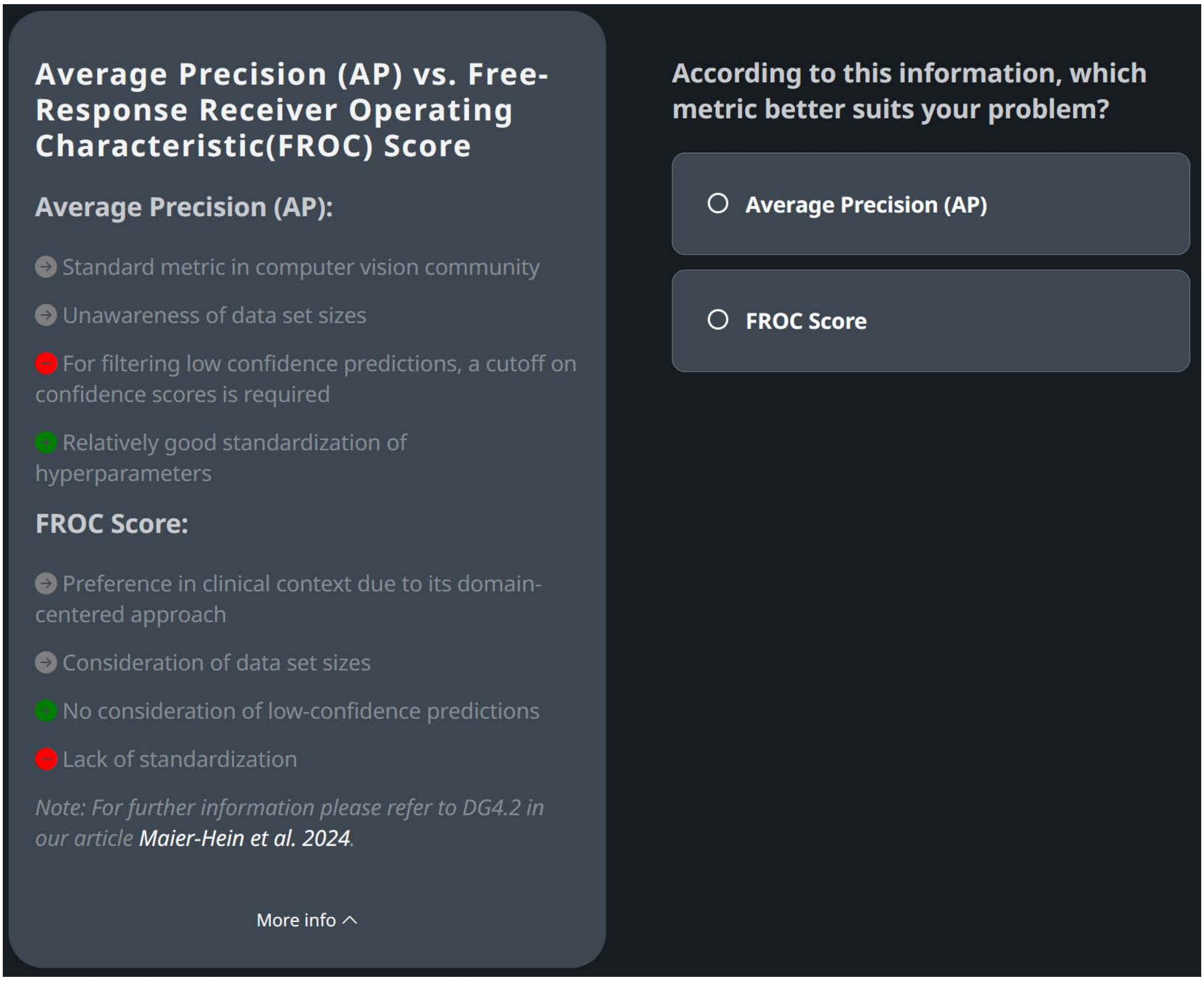
Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

amounts to setting a cutoff value in the case of binary classification. Based on this decision rule, we calculate the confusion matrix.

In our example, it is desirable to define a decision rule in such a way that a target metric value is achieved; more specifically, we aim for a Sensitivity score of 0.95. So the system recommends something rather strange: "**Metric@(TargetMetric = TargetValue)**". What does this mean? Well, exactly what we defined earlier! The decision rule is set according to our "target metric" (here: Sensitivity), which has reached a "target value" (here: 0.95). According to this decision rule, the standard confusion matrix is calculated and from this, the "metric" is calculated. In our case, we choose the **False Positives Per Image (FPPI)** metric, as it best suits the use case.

Thanks to the fact that we've already answered several questions, we don't actually need to answer another question to decide on a multi-threshold metric (= curve-based metric) — the toolkit has saved our previous answers and avoids asking the same question multiple times. And for our selection, this means that we don't even have to answer a single question for the multi-threshold metrics! However, we are presented with a metric decision guide… The reason for having the decision guides is quite simple: In the recommendation framework, we couldn't cover every single potential use case. Instead, we have developed decision guides that compare the pros and cons of the two or three most suitable metrics, in this case the Average Precision (AP) and the Free-Response Receiver Operating Characteristic (FROC) Score, based on your previous responses. This means that you can decide, based on your own research problem, which metric is the best fit and which problems or pitfalls are not as important. In our use case, we are in close contact with radiologists, so we prefer a metric that is well known among clinicians and we don't really care about the lack of standardization, so we choose the **FROC Score**.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



Average Precision (AP) vs. Free-Response Receiver Operating Characteristic(FROC) Score

Average Precision (AP):

➜ Standard metric in computer vision community

➜ Unawareness of data set sizes

🔴 For filtering low confidence predictions, a cutoff on confidence scores is required

🟢 Relatively good standardization of hyperparameters

FROC Score:

➜ Preference in clinical context due to its domain-centered approach

➜ Consideration of data set sizes

🟢 No consideration of low-confidence predictions

🔴 Lack of standardization

Note: For further information please refer to DG4.2 in our article Maier-Hein et al. 2024.

More info ^

According to this information, which metric better suits your problem?

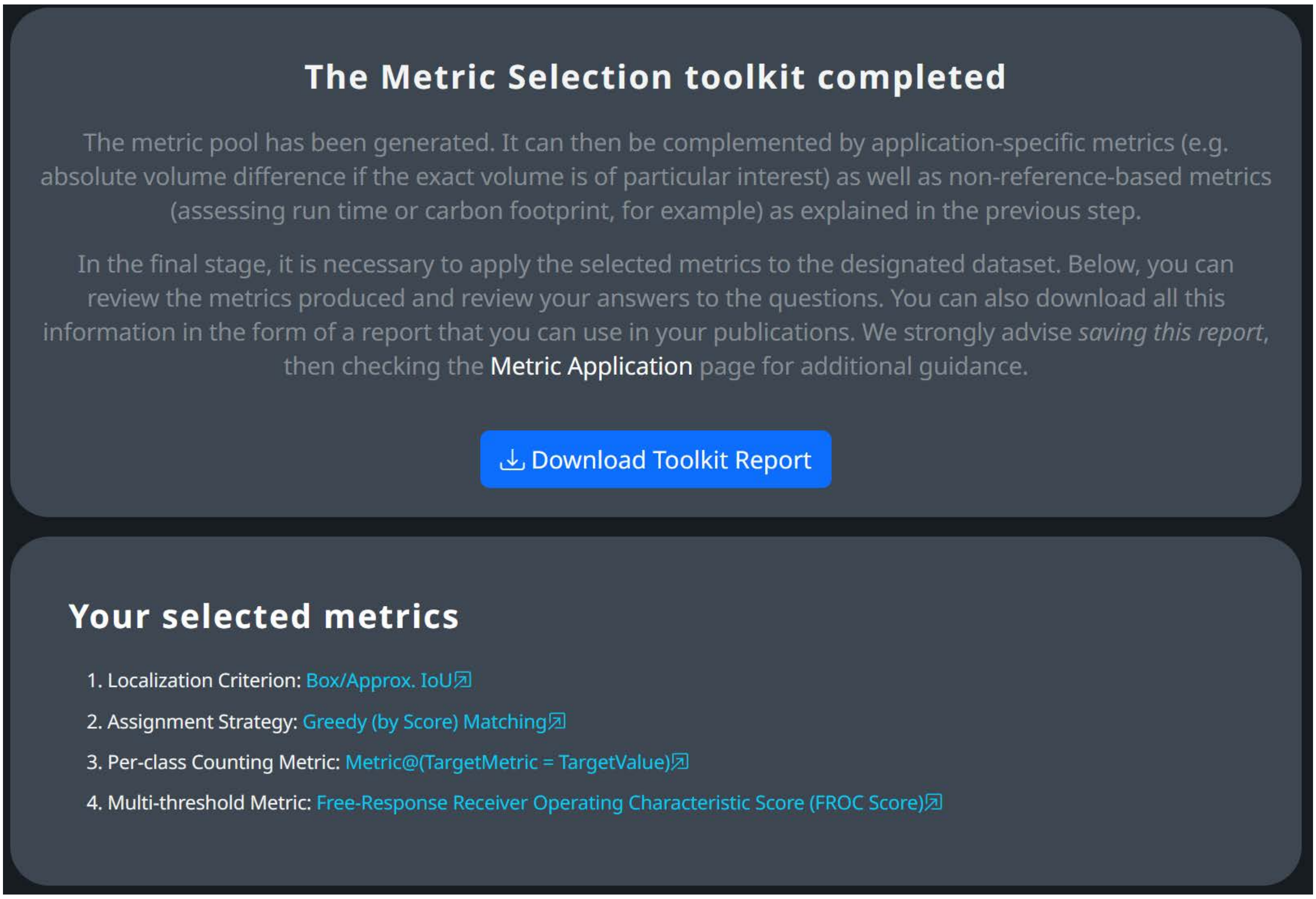
○ **Average Precision (AP)**

○ **FROC Score**

Decision guide for deciding between the metrics AP and FROC score. Based on the driving research question, choose the best metric for your use case. Decision guides are based on previous answers.

And that's it! The toolkit goes on to suggest that we could complement our metric pool with custom metrics to address application-specific complementary properties or with non-reference-based metrics to assess, for example, speed, memory consumption, or carbon footprint. Finally, the toolkit provides an overview of the selected metrics and the option to download a report which summarizes all your replies and selections.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



The Metric Selection toolkit completed

The metric pool has been generated. It can then be complemented by application-specific metrics (e.g. absolute volume difference if the exact volume is of particular interest) as well as non-reference-based metrics (assessing run time or carbon footprint, for example) as explained in the previous step.

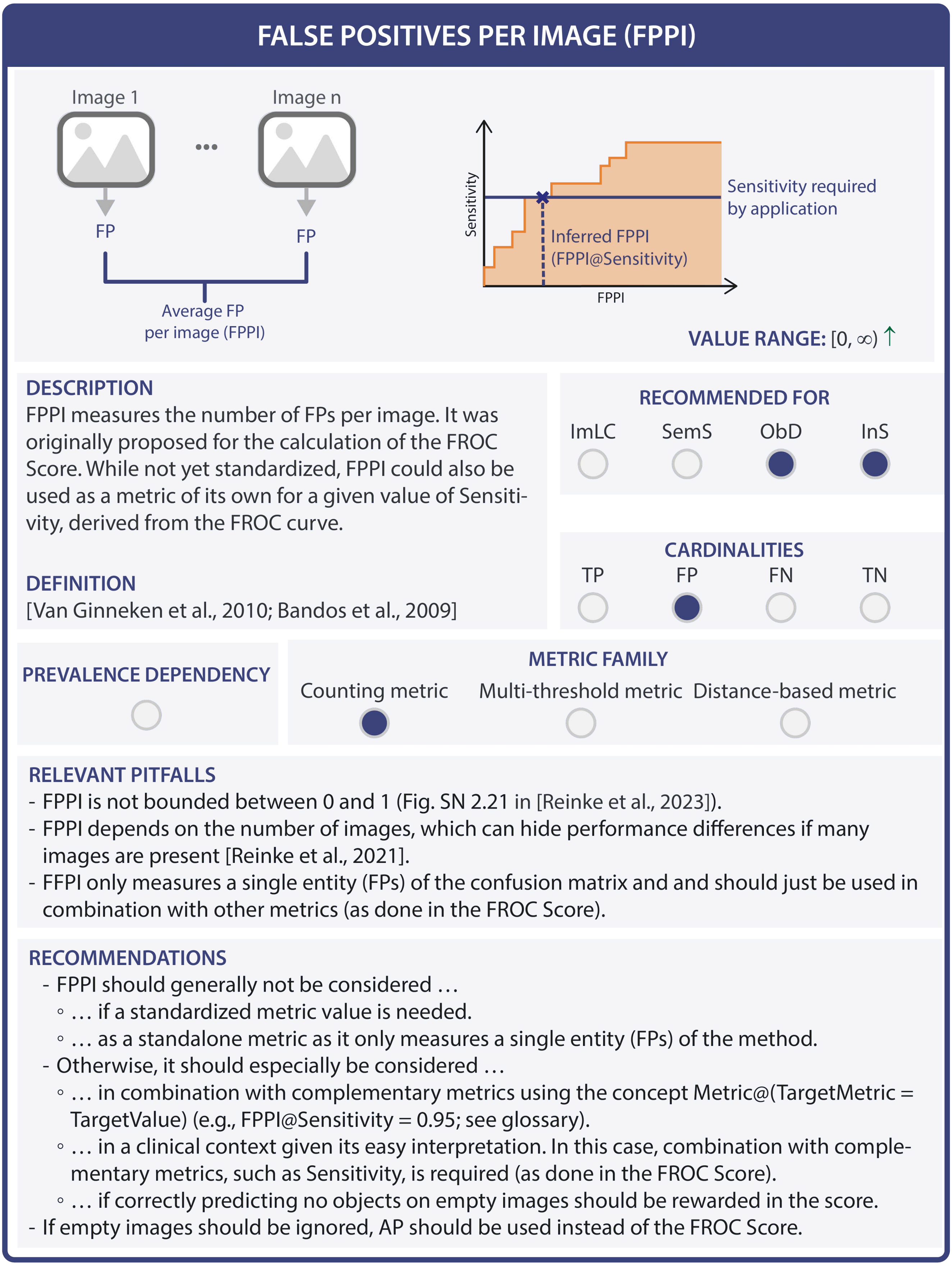
In the final stage, it is necessary to apply the selected metrics to the designated dataset. Below, you can review the metrics produced and review your answers to the questions. You can also download all this information in the form of a report that you can use in your publications. We strongly advise *saving this report*, then checking the Metric Application page for additional guidance.

Your selected metrics

1. Localization Criterion: Box/Approx. IoU
2. Assignment Strategy: Greedy (by Score) Matching
3. Per-class Counting Metric: Metric@(TargetMetric = TargetValue)
4. Multi-threshold Metric: Free-Response Receiver Operating Characteristic Score (FROC Score)

In addition, as you can see (and you may have already explored the links to the selected metrics in the text above), the system provides you with links to all the recommended metrics and these links take you to the ***Metrics Library***. This library provides an overview of all metrics in the framework and gives you the opportunity to explore and learn even more. For each metric, you will get a graphical explanation, a description, pitfalls, recommendations, and more. This is a great resource, a cheat sheet for each metric, helping you to have all relevant information for a metric at your fingertips!

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



FALSE POSITIVES PER IMAGE (FPPI)

Image 1
...
Image n
FP
FP
Average FP
per image (FPPI)
Sensitivity
FPPI
Sensitivity required
by application
Inferred FPPI
(FPPI@Sensitivity)

VALUE RANGE: [0, ∞) ↑

DESCRIPTION

FPPI measures the number of FPs per image. It was originally proposed for the calculation of the FROC Score. While not yet standardized, FPPI could also be used as a metric of its own for a given value of Sensitivity, derived from the FROC curve.

DEFINITION

[Van Ginneken et al., 2010; Bandos et al., 2009]

RECOMMENDED FOR

ImLC	SemS	ObD	InS
○	○	●	●

CARDINALITIES

TP	FP	FN	TN
○	●	○	○

PREVALENCE DEPENDENCY

○

METRIC FAMILY

Counting metric	Multi-threshold metric	Distance-based metric
●	○	○

RELEVANT PITFALLS

- FPPI is not bounded between 0 and 1 (Fig. SN 2.21 in [Reinke et al., 2023]).
- FPPI depends on the number of images, which can hide performance differences if many images are present [Reinke et al., 2021].
- FFPI only measures a single entity (FPs) of the confusion matrix and and should just be used in combination with other metrics (as done in the FROC Score).

RECOMMENDATIONS

- FPPI should generally not be considered …
  - … if a standardized metric value is needed.
  - … as a standalone metric as it only measures a single entity (FPs) of the method.
- Otherwise, it should especially be considered …
  - … in combination with complementary metrics using the concept Metric@(TargetMetric = TargetValue) (e.g., FPPI@Sensitivity = 0.95; see glossary).
  - … in a clinical context given its easy interpretation. In this case, combination with complementary metrics, such as Sensitivity, is required (as done in the FROC Score).
  - … if correctly predicting no objects on empty images should be rewarded in the score.
- If empty images should be ignored, AP should be used instead of the FROC Score.

Metric cheat sheet as provided by the Metrics Library of the toolkit.

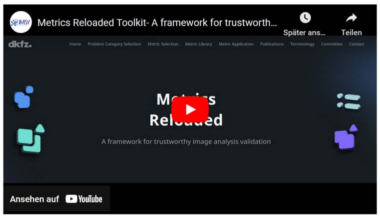
And now you have completed your metric selection. We hope that you have learned something about the process without having to read 200 pages! And if you

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

want to learn more, the toolkit always tells you where to go in the original paper!

One final note: the toolkit also provides you with a *Problem Category Mapping* to help you choose the right problem category. If you are unsure, just try it out to make sure you are really working on the right problem! For example, in our use case, we are mostly dealing with small structures. While the problem is often treated as a segmentation problem, we recommend phrasing it as a detection problem, as many segmentation metrics cannot handle small structures properly.

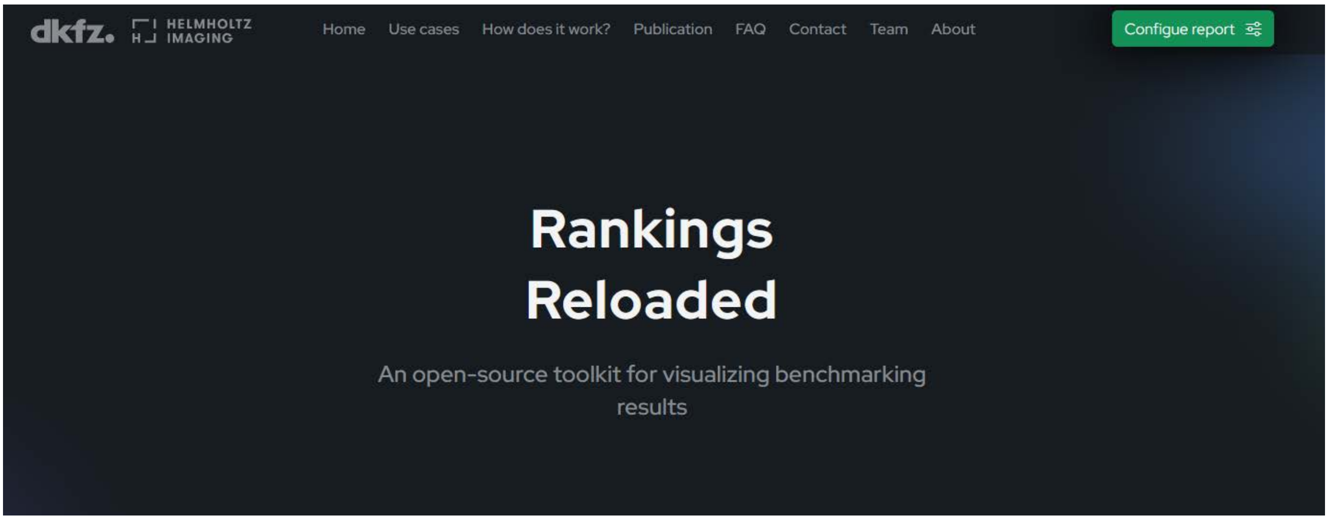
Below, you'll see the whole selection process in a video. As you can see, it doesn't take long and the more often you try it for different use cases, the more familiar you get with the different questions and the faster you get!


Metrics Reloaded Toolkit- A framework for trustworth...
Später ans...
Teilen
dkfz.
Metrics Reloaded
A framework for trustworthy image analysis validation
Ansehen auf YouTube

Full selection process of the use case of MS lesion detection in multimodal brain MRI images.
Video available at https://youtu.be/qMfrgqlOaag.

• • •

Rankings Reloaded Toolkit



Rankings Reloaded

An open-source toolkit for visualizing benchmarking results

https://www.rankings-reloaded.de/

Now that we know how to choose the best metrics, the question is how to apply them. When publishing a new method, most journals and conferences require us to show a comparison with other baseline methods, which means that benchmarking is becoming increasingly important, both on a small and large scale, e.g., to organize your very own challenge. But how do you do that?

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Staying with our marathon example, it is quite simple: sort the times taken by each runner and create a ranking based on the order. Now let's say we are comparing three models for a segmentation task. In this case, we have one (or more) metric score per image per model. To rank the models, we could for example aggregate the metric scores per model with a mean or median. Or we could calculate a ranking per image and aggregate the ranks. Or we could perform pairwise statistical tests and rank by the number of significant superiorities. All options are possible, and as previous research [6] has shown, rankings are extremely sensitive to the parameters chosen! It is therefore very important to analyze and visualize the results adequately. Imagine a ranking table in a paper without any visualization — would you be able to see if rank 1 is really better than rank 2? No, you would not! **Visualization and ranking robustness analysis is extremely important for interpretability — how else can you claim to outperform others?**

This is where our Rankings Reloaded toolkit comes in. In the theoretical part of the toolkit [4], the authors presented several advanced analysis and visualization methods which we implemented in the toolkit. **The toolkit provides robust and accurate uncertainty analysis and visualization of algorithm performance, enabling researchers to conduct fair benchmarking by revealing the true strengths and weaknesses of each.** So what are the use cases for the toolkit?

Compare different algorithms: The tool robustly ranks your algorithm of interest against baselines and provides advanced visualizations of the results. It helps you assess ranking robustness to see if your algorithm is really superior to others, even if the dataset changes slightly.

Hyperparameter optimization: Want to know which hyperparameter configuration works best? The tool helps by providing descriptive statistics and visualization, showing ranking stability across all images, and can help you decide whether a new approach has a positive impact compared to others.

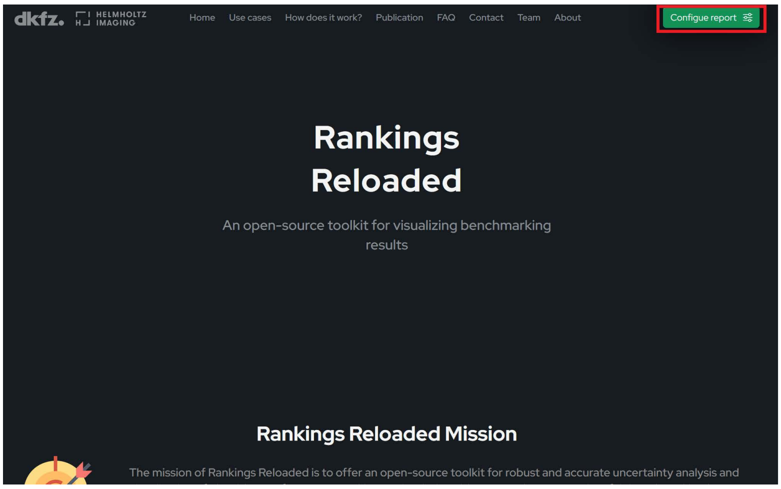
Benchmarking (e.g., organizing a challenge): The tool provides comparisons of different ranking methods and visualizes ranking stability by using bootstrapping or statistical tests. It allows to take comprehensive snapshots of challenge results, rankings, and outcomes.

Example use case

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Let’s stay with the segmentation example. For simplicity, we will use only one metric (but you can extend to multiple metrics), the Dice Similarity Score (DSC). The toolkit is divided into four main steps.

1. Upload your data

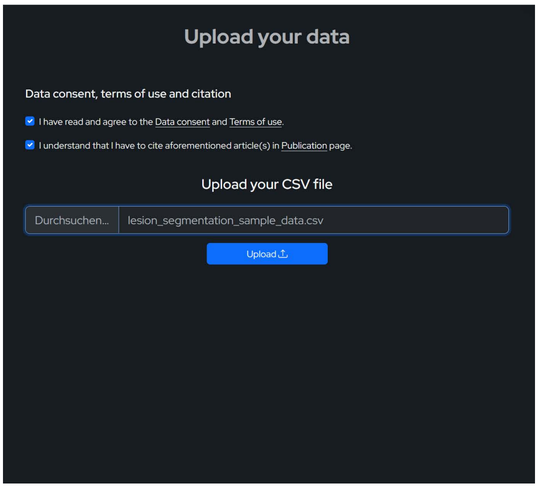


Rankings Reloaded

An open-source toolkit for visualizing benchmarking results

Rankings Reloaded Mission

The mission of Rankings Reloaded is to offer an open-source toolkit for robust and accurate uncertainty analysis and



Upload your data

Data consent, terms of use and citation

☑ I have read and agree to the Data consent and Terms of use.

☑ I understand that I have to cite aforementioned article(s) in Publication page.

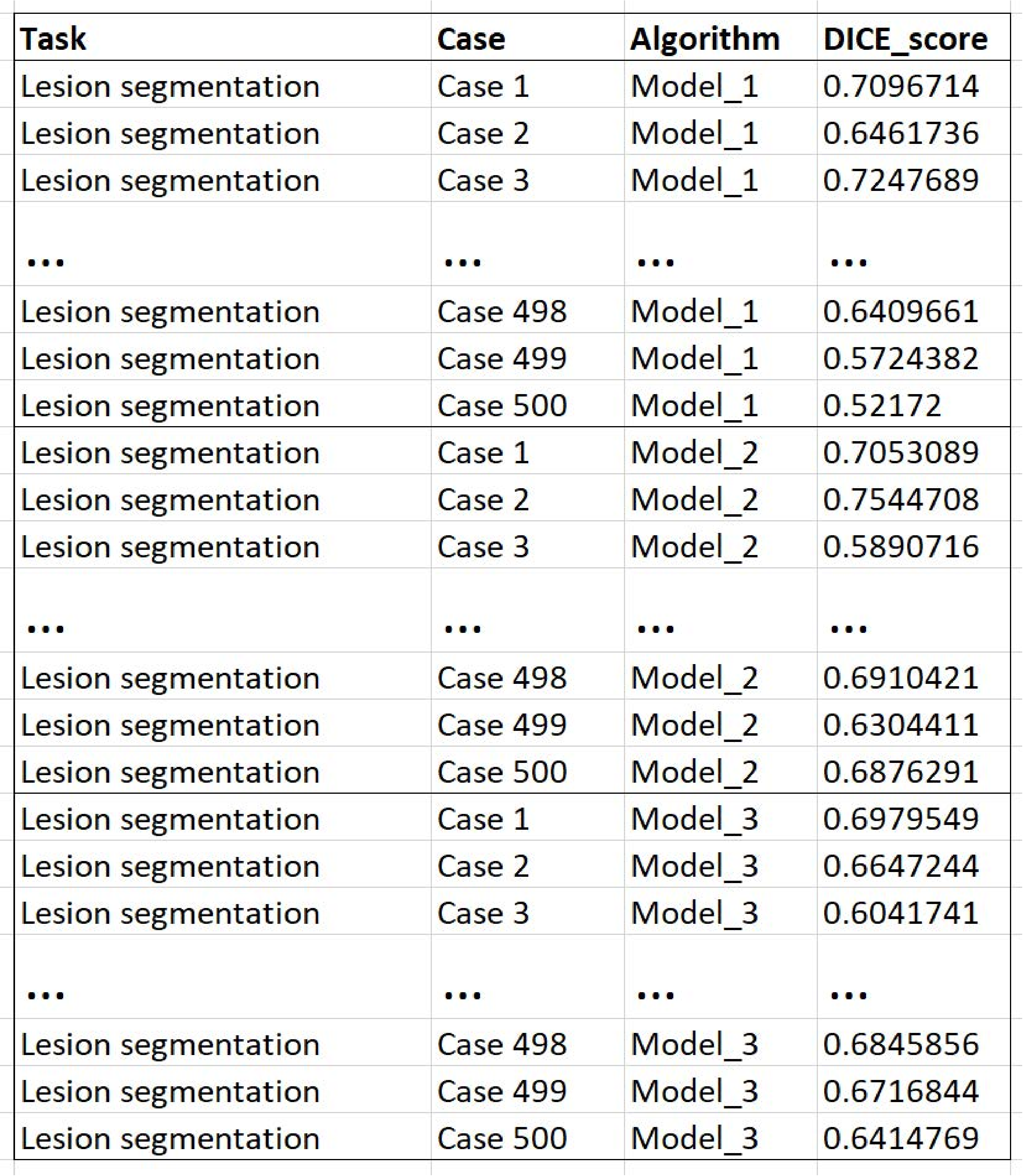
Upload your CSV file

Durchsuchen... lesion_segmentation_sample_data.csv

The first step is to prepare our data in the correct format, i.e., we need it as a *.csv* file similar to the example in the screenshot below.

- A **task** identifier is needed when different tasks are performed. A task may, for example, refer to “lesion segmentation” but may also refer to different metrics.
- A **case** identifier should contain all cases (images) that are used in the benchmarking experiment. Ensure that each case only appears once for the same algorithm and task.
- An **algorithm** identifier should contain all algorithms/methods used in the comparison. For each case, the algorithm should appear once.
- The calculated **metric values** should appear in a separate column (“DICE_score” in the screenshot below). For a single metric, one value should appear for each algorithm for each case. In case of missing metric values, a missing observation must be provided (either as a blank field or “NA”), otherwise the system cannot generate the report.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



Task	Case	Algorithm	DICE_score
Lesion segmentation	Case 1	Model_1	0.7096714
Lesion segmentation	Case 2	Model_1	0.6461736
Lesion segmentation	Case 3	Model_1	0.7247689
...	...	...	...
Lesion segmentation	Case 498	Model_1	0.6409661
Lesion segmentation	Case 499	Model_1	0.5724382
Lesion segmentation	Case 500	Model_1	0.52172
Lesion segmentation	Case 1	Model_2	0.7053089
Lesion segmentation	Case 2	Model_2	0.7544708
Lesion segmentation	Case 3	Model_2	0.5890716
...	...	...	...
Lesion segmentation	Case 498	Model_2	0.6910421
Lesion segmentation	Case 499	Model_2	0.6304411
Lesion segmentation	Case 500	Model_2	0.6876291
Lesion segmentation	Case 1	Model_3	0.6979549
Lesion segmentation	Case 2	Model_3	0.6647244
Lesion segmentation	Case 3	Model_3	0.6041741
...	...	...	...
Lesion segmentation	Case 498	Model_3	0.6845856
Lesion segmentation	Case 499	Model_3	0.6716844
Lesion segmentation	Case 500	Model_3	0.6414769

Example data. The .csv file should include a task identifier (here: ‘Task’), a case identifier (here: ‘Case’), an algorithm identifier (here: ‘Algorithm’), and a metric value column (here: ‘DICE_score’).

Note: The toolkit also provides a sample .csv file for download!

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Once the data is in the correct format, upload it to the toolkit. It will give you a preview of the data which should be used as a sanity check. The tool will ask you about which column refers to which identifier (so no need to name them specifically, you can just call them as “foo bar” ;)). To make sure that the ranking is calculated correctly, the toolkit will ask you how to order the results (ascending or descending order).

If there are NaNs in the data, choose how to deal with them. In the case of many algorithms, some plots may be overloaded, so you can also choose to show only the top X algorithms.

2. Configure ranking

The second step is to decide which ranking method to use. There are currently three main aggregation methods:

- **Metric-based aggregation (aggregate-then-rank):** The most commonly used ranking method starts by aggregating metric values across all test cases (e.g., with mean, median, or other quartile) for each algorithm. This aggregate is then used to calculate a rank for each algorithm.
- **Case-based aggregation (rank-then-aggregate):** The case-based ranking method starts by calculating a rank for each test case for each algorithm (“rank first”). The final rank is based on the aggregated test case ranks.
- **Significance ranking (test-based):** In a complementary approach, statistical tests are computed for each possible pair of algorithms to assess differences in metric values between the algorithms. Ranking is performed according to the resulting relations or according to the number of significant one-sided test results per algorithm.

Based on the chosen ranking strategy, you also have the option to define the aggregation operator (e.g., mean vs. median), alpha level and p-value adjustment for multiple testing for significance rankings, or how to handle ties.

To help you further, the tool provides you with more detailed explanations of each strategy.

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

3. Configure uncertainty analysis

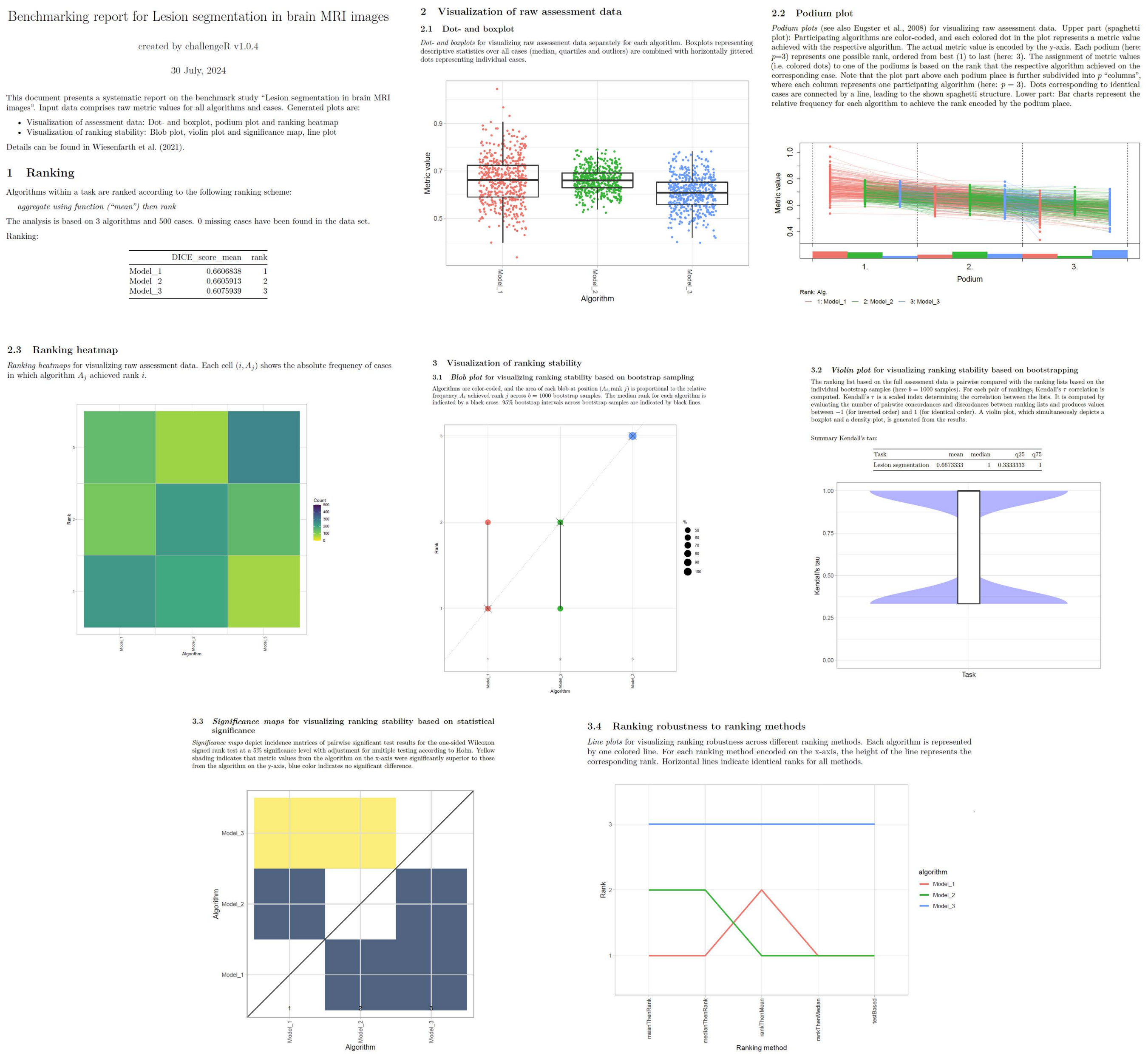
Uncertainty analysis is important to check whether the ranking is stable or whether it changes with small perturbations. The toolkit offers the possibility to perform bootstrapping. This involves creating new bootstrap datasets based on sampling with replacement. This means that you generate, for example, 1000 slightly different datasets, in which some cases occur several times and others don't occur at all, to simulate small perturbations to the data. For each bootstrap dataset, you calculate your ranking and check whether it changed compared to the original. If not — great, your ranking is very stable! If it does, your ranking may not be very stable. The results of the bootstrapped rankings are shown in the form of graphs.

Note that bootstrapping can take some time — you may also decide to reduce the number of bootstraps or to skip this section.

4. Generate the report

The final step is to generate the report. Just enter a report title and be a bit patient — your report will soon appear with several different types of visualizations. Each one is described and explained, so don't worry!

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



Benchmarking report for Lesion segmentation in brain MRI images

created by challengeR v1.0.4

30 July, 2024

This document presents a systematic report on the benchmark study "Lesion segmentation in brain MRI images". Input data comprises raw metric values for all algorithms and cases. Generated plots are:

- Visualization of assessment data: Dot- and boxplot, podium plot and ranking heatmap
- Visualization of ranking stability: Blob plot, violin plot and significance map, line plot

Details can be found in Wiesenfarth et al. (2021).

1 Ranking

Algorithms within a task are ranked according to the following ranking scheme:

aggregate using function ("mean") then rank

The analysis is based on 3 algorithms and 500 cases. 0 missing cases have been found in the data set.

Ranking:

	DICE_score_mean	rank
Model_1	0.6606838	1
Model_2	0.6605913	2
Model_3	0.6075939	3

2 Visualization of raw assessment data

2.1 Dot- and boxplot

Dot- and boxplots for visualizing raw assessment data separately for each algorithm. Boxplots representing descriptive statistics over all cases (median, quartiles and outliers) are combined with horizontally jittered dots representing individual cases.

2.2 Podium plot

Podium plots (see also Eugster et al., 2008) for visualizing raw assessment data. Upper part (spaghetti plot): Participating algorithms are color-coded, and each colored dot in the plot represents a metric value achieved with the respective algorithm. The actual metric value is encoded by the y-axis. Each podium (here: p=3) represents one possible rank, ordered from best (1) to last (here: 3). The assignment of metric values (i.e. colored dots) to one of the podiums is based on the rank that the respective algorithm achieved on the corresponding case. Note that the plot part above each podium place is further subdivided into p "columns", where each column represents one participating algorithm (here: $p = 3$). Dots corresponding to identical cases are connected by a line, leading to the shown spaghetti structure. Lower part: Bar charts represent the relative frequency for each algorithm to achieve the rank encoded by the podium place.

2.3 Ranking heatmap

Ranking heatmaps for visualizing raw assessment data. Each cell (i, A_j) shows the absolute frequency of cases in which algorithm A_j achieved rank i.

3 Visualization of ranking stability

3.1 *Blob plot* for visualizing ranking stability based on bootstrap sampling

Algorithms are color-coded, and the area of each blob at position $(A_i, \text{rank } j)$ is proportional to the relative frequency A_i achieved rank j across $b = 1000$ bootstrap samples. The median rank for each algorithm is indicated by a black cross. 95% bootstrap intervals across bootstrap samples are indicated by black lines.

3.2 *Violin plot* for visualizing ranking stability based on bootstrapping

The ranking list based on the full assessment data is pairwise compared with the ranking lists based on the individual bootstrap samples (here $b = 1000$ samples). For each pair of rankings, Kendall's τ correlation is computed. Kendall's τ is a scaled index determining the correlation between the lists. It is computed by evaluating the number of pairwise concordances and discordances between ranking lists and produces values between -1 (for inverted order) and 1 (for identical order). A violin plot, which simultaneously depicts a boxplot and a density plot, is generated from the results.

Summary Kendall's tau:

Task	mean	median	q25	q75
Lesion segmentation	0.6673333	1	0.3333333	1

3.3 *Significance maps* for visualizing ranking stability based on statistical significance

Significance maps depict incidence matrices of pairwise significant test results for the one-sided Wilcoxon signed rank test at a 5% significance level with adjustment for multiple testing according to Holm. Yellow shading indicates that metric values from the algorithm on the x-axis were significantly superior to those from the algorithm on the y-axis, blue color indicates no significant difference.

3.4 Ranking robustness to ranking methods

Line plots for visualizing ranking robustness across different ranking methods. Each algorithm is represented by one colored line. For each ranking method encoded on the x-axis, the height of the line represents the corresponding rank. Horizontal lines indicate identical ranks for all methods.

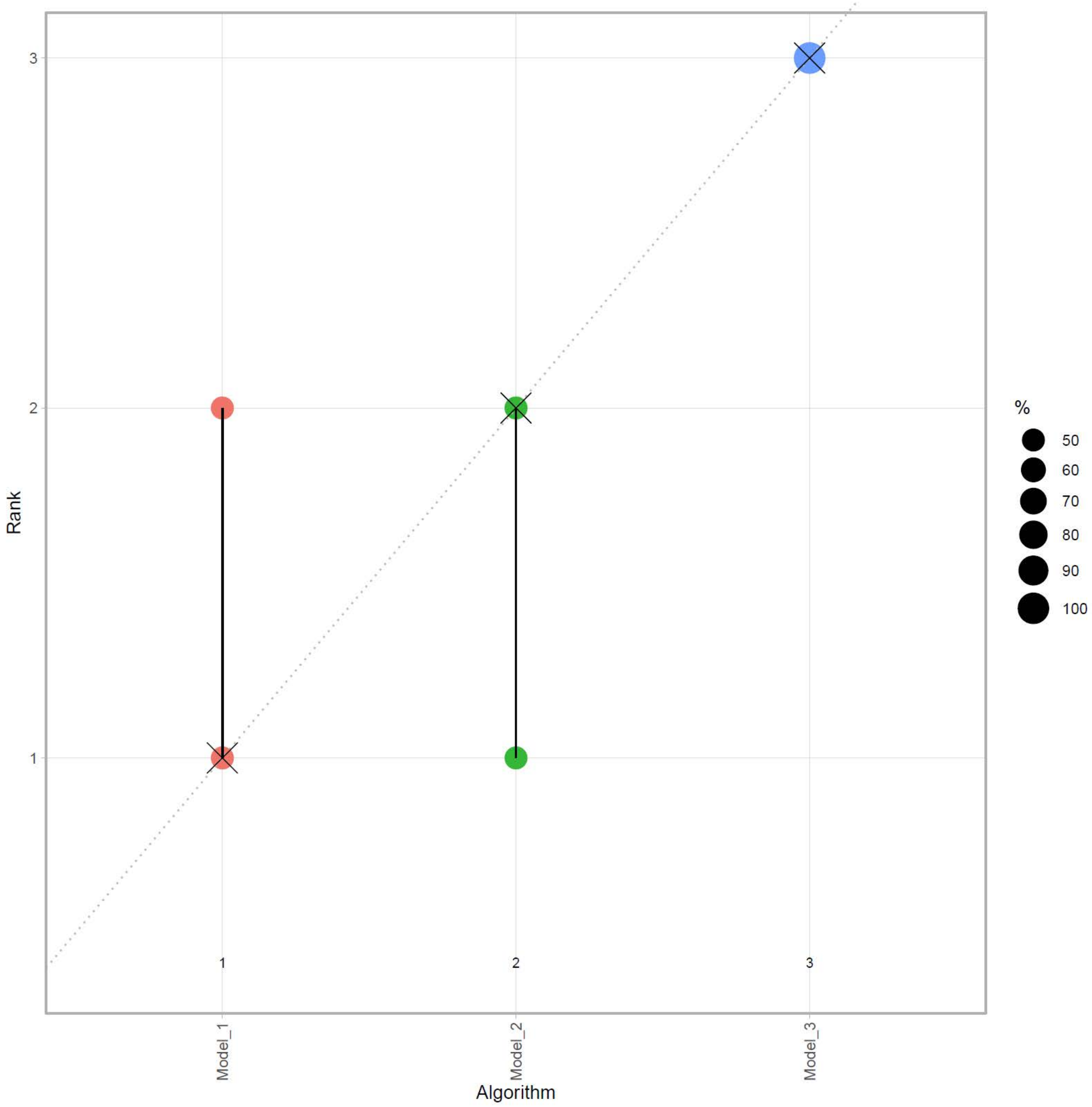
Generated report for a single segmentation task for three models including various plots and descriptions for advanced analysis and visualization of the ranking.

The report starts by presenting the final ranking table, followed by plots visualizing raw data, i.e., the metric values. Several different plots are provided, for example dots- and boxplots, so that you can choose the one that best suits your problem or which provides the best interpretability.

In the following section, ranking stability is assessed using the bootstrap method. One way of visualizing ranking uncertainty is to use *blob plots*, where the area of each blob at position (Model_i, rank j) is proportional to the relative frequency Model_i achieving rank j across the bootstrap datasets. The median rank for each

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

algorithm is indicated by a black cross. The 95% bootstrap intervals across the bootstrap samples are indicated by black lines. In our example, we see that Models 1 and 2 are very close to each other and we cannot generally say that Model 1 outperforms Model 2.

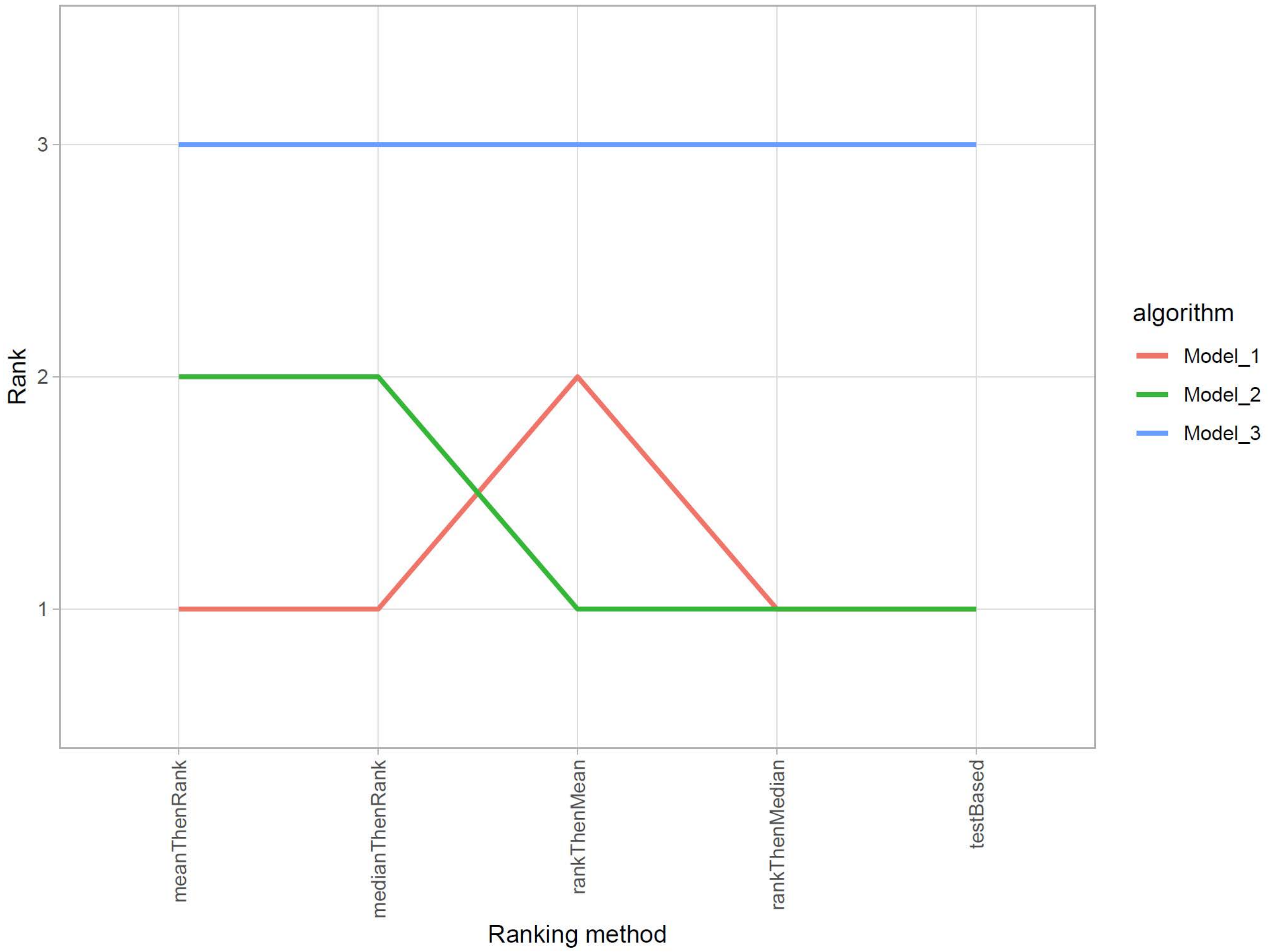


Blob plot for visualizing ranking stability based on bootstrap sampling [4].

Another option for uncertainty analysis would be to calculate the correlation between the original and the bootstrap rankings and plot the correlation coefficient values in violin plots or plot incidence matrices of pairwise significant test results in a so-called *significance map*.

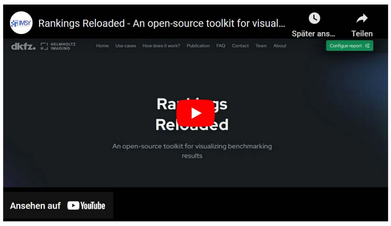
Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Finally, the report compares different ranking methods and shows how ranks would change if you had chosen a different ranking method. We can see that Models 1 and 2 sometimes exchange their ranks.



Line plots for visualizing ranking robustness across different ranking methods [4].

If your use case involves multiple tasks, the report will also include results per task and plots for cross-task insights.


Rankings Reloaded - An open-source toolkit for visual...
Später ans...
Teilen
Rankings Reloaded
An open-source toolkit for visualizing benchmarking results
Ansehen auf YouTube

Full walkthrough of the Rankings Reloaded toolkit, including data upload, ranking and uncertainty analysis configuration, and report generation.
Video available at https://youtu.be/9zpupko2V5Y.

• • •

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Conclusion

Putting theory into practice is an important step. We hope that you will find these two example toolkits helpful in ensuring that best practice can be followed in an easy way. We are always open to further suggestions and will try to incorporate them. Feel free to drop us a line (metrics-reloaded(at)dkfz.de or rankings-reloaded(at)dkfz.de)!

. . .

References

[1] Reinke/Tizabi et al. (2024). Understanding metric-related pitfalls in image analysis validation. Nature Methods.

[2] Reinke/Tizabi et al. (2021). Common limitations of image processing metrics: A picture story. arXiv preprint.

[3] Maier-Hein/Reinke et al (2024). Metrics reloaded: recommendations for image analysis validation. Nature Methods.

[4] Wiesenfarth et al. (2021). Methods and open-source toolkit for analyzing and visualizing challenge results. Scientific Reports.

[5] Reinke, Sudre, Tizabi. A discovery dive into the world of evaluation — Do's, don'ts and other considerations (2021). Medium Blogpost.

[6] Maier-Hein/Eisenmann (2018). Why rankings of biomedical image analysis competitions should be interpreted with care. Nature Communications.

[7] Commowick et al. (2018). Objective evaluation of multiple sclerosis lesion segmentation using a data management and processing infrastructure. Scientific reports.

[8] Kofler et al. (2023). Blob loss: Instance imbalance aware loss functions for semantic segmentation. International Conference on Information Processing in Medical Imaging.

. . .

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

Acknowledgements

Part of this work was funded by Helmholtz Imaging (HI), a platform of the Helmholtz Incubator on Information and Data Science.

HELMHOLTZ IMAGING

dkfz. GERMAN CANCER RESEARCH CENTER IN THE HELMHOLTZ ASSOCIATION

Research for a Life without Cancer

Special thanks to the co-authors of the theoretical works:

L. Maier-Hein, P.F. Jäger, P. Godau, M.D. Tizabi, F. Buettner, E. Christodoulou, B. Glocker, F. Isensee, J. Kleesiek, M. Kozubek, M. Reyes, M.A. Riegler, M. Wiesenfarth, C.H. Sudre, M. Baumgartner, M. Eisenmann, D. Heckmann-Nötzel, T. Rädsch, L. Acion, L. Aguilera Saiz, M. Antonelli, T. Arbel, S. Bakas, A. Benis, M.B. Blaschko, M. J. Cardoso, J. Chen, V. Cheplygina, B.A. Cimini, G.S. Collins, K. Farahani, L. Ferrer, A. Galdran, B. van Ginneken, R. Haase, D.A. Hashimoto, M.M. Hoffman, M. Huisman, P. Jannin, C.E. Kahn, D. Kainmueller, B. Kainz, A. Karargyris, A. Karthikesalingam, F. Kofler, T. Kooi, A. Kopp-Schneider, A. Kreshuk, T. Kurc, B.A. Landman, G. Litjens, A. Madani, K. Maier-Hein, A.L. Martel, P. Mattson, E. Meijering, B. Menze, K.G.M. Moons, H. Müller, B. Nichyporuk, F. Nickel, J. Petersen, S.M. Rafelski, N. Rajpoot, N. Rieke, J. Saez-Rodriguez, C.I. Sánchez, S. Shetty, M. van Smeden, R.M. Summers, A.A. Taha, A. Tiulpin, S.A. Tsaftaris, B. Van Calster, G. Varoquaux, Z.R. Yaniv

Metrics Rankings Validation AI Toolkit

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF