Interpretable meta-score for model performance

Alicja Gosiewska1✉, Katarzyna Woźnica2✉ and Przemysław Biecek2,3

Benchmarks are an integral part of machine learning development. However, the most common benchmarks share several limitations. For example, the difference in performance between two models has no probabilistic interpretation, it makes no sense to compare such differences between data sets and there is no reference point that indicates a significant performance improvement. Here we introduce an Elo-based predictive power meta-score that is built on other performance measures and allows for interpretable comparisons of models. Differences between this score have a probabilistic interpretation and can be compared directly between data sets. Furthermore, this meta-score allows for an assessment of ranking fitness. We prove the properties of the Elo-based predictive power meta-score and support them with empirical results on a large-scale benchmark of 30 classification data sets. Additionally, we propose a unified benchmark ontology that provides a uniform description of benchmarks.

The current rapid development of machine learning has resulted in a considerable increase in the number of new algorithms that need to be compared against state-of-the-art ones. This was accompanied by the need to establish procedures for systematic comparisons between algorithms. The current practice is to create benchmarks on predefined sets of tasks. Examples of such benchmarks for natural language processing models are General Language Understanding Evaluation (GLUE)1 and its successor SuperGlue2, which are collections of tasks used to evaluate algorithms. They include, among other things, sentiment analysis, semantic similarity and paraphrasing. Many areas of machine learning already have benchmarks, for example, the Visual Task Adaptation Benchmark (VTAB)3 to evaluate computer vision models. Benchmarks are mainly created in deep learning because the aim is to build unified algorithms with understanding beyond the shallow patterns in data, which is why it is crucial to compare algorithms on a wide variety of tasks from different domains. Another way to evaluate progress in algorithm development is to carry out large-scale assessments of models addressing a specific biological question. The majority of such initiatives belong to the category of critical assessments. The pioneering Critical Assessment of protein Structure Prediction4 revolutionized benchmarks of tools for the prediction of protein structure and helped to shape the landscape of the discipline into a ground fertile enough to stimulate the development of models as revolutionary as AlphaFold5. The Critical Assessment of protein Structure Prediction was quickly followed by similar competitions such as the Critical Assessment of protein Function Annotation algorithms6, focused on modelling the relationship between protein sequence and function, or the Critical Assessment of Prediction of Interactions7, where the goal is to computationally identify potential interactions between proteins. Through this, the performance of algorithms for predicting new structures, properties or interactions between proteins is regularly compared. Another popular approach for model comparison is storing and sharing the results of multiple algorithms on multiple data sets on platforms such as Kaggle, Papers With Code or OpenML8.

These days, it is difficult to imagine a high-quality article with a new algorithm without a comparison against state-of-the-art methods on at least one benchmark. However, there is no unified description of benchmarks to refer to when describing new ones. Moreover, performance measures of models currently used in benchmarks share many limitations, such as the inability to interpret differences in performance or the impossibility of comparing models between data sets.

Considering the shortcomings of existing benchmarks, the need for new approaches for comparing models and establishing new guidelines is being felt in the machine learning community7. This urgency is borne out by the fact that, in 2021, the organizers of the Thirty-Fifth Conference on Neural Information Processing Systems provided a new track dedicated to data sets and benchmarks https://nips.cc/Conferences/2021/CallForDatasetsBenchmarks. In this article, we propose an Elo-based predictive power (EPP) meta-score, which is a new way of aggregating model results and overcomes the most common problems with benchmarks and the performance scores they use. The main contributions of this paper are as follows:

- We propose a unified benchmark ontology that allows for the uniform description of different benchmarks.
- We identify and demonstrate the limitations of the most common measures of machine learning model performance, such as the lack of interpretation of differences and incomparability between data sets.
- In light of the highlighted limitations of the most common measures, we propose a new meta-score named EPP that is built as an aggregation of other measures and enriches them by providing interpretable comparisons of models, even between data sets.
- We apply EPP on a large-scale benchmark from the OpenML repository and VTAB. In both cases, we show how the use of EPP enriches the understanding of model performance.

The paper is organized as follows: Historical overview section provides an overview of the literature on the history of benchmarks and the Elo rating system. Unified benchmark ontology section presents the ontology for any machine learning benchmark. Computing EPP on the OpenML100 benchmark section presents

---

1Why R? Foundation, Warsaw, Poland. 2Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland. 3Faculty of Mathematics, Informatics, and Mechanics, University of Warsaw, Warsaw, Poland. ✉e-mail: alicjagosiewska@gmail.com; katarzyna.wozniaczok@pw.edu.pl; przemyslaw.biecek@pw.edu.pl
a use case of EPP applied to real-life OpenML benchmarks. This is followed by the Conclusions and future work section. Methods section describes the EPP meta-score, including sections on the Definition of the EPP score and how EPP handles the problems with common ML performance measures. Supplementary Section S1 includes results for the VTAB benchmark.

On a first reading, the reader can start with Methods section, which describes the EPP, and skip Unified benchmark ontology section, which shows the universality and comprehensiveness of the description of the benchmark but is not essential to understanding the later sections.

Historical overview. The problem of model assessment is even older than modern statistics. Its origins can be traced back to Laplace's work from 1796 on the nebular hypothesis, which describes the formation of the solar system from a slowly rotating nebula cloud of extremely hot gas. Since then, the increasing number of applications for models has led to an increase in the number of metrics describing their quality. The various measures of model performance differ in their properties and applications.[4,17]. The most common machine learning frameworks such as scikit-learn12, TensorFlow13 or mlr 14 rely on common measures such as accuracy, area under the receiver operating characteristic curve (AUC), recall, precision, F1 and cross-entropy for classification or the mean squared error (MSE), root mean square error (RMSE) and mean absolute error (MAE) for regression problems.

Beyond the selection of measures, there is an even more important problem related to evaluating whether differences between their values are statistically significant or come from noise in validation data sets. There have been many approaches to verify whether a newly proposed algorithm improves the performance compared with previous, state-of-the-art methods. The majority of these have been statistical testing procedures. Demšar15 reviewed commonly used practices and pointed out the huge number of problems with them. One of the earliest and most widely cited articles in this area is that by Dietterich16. He gave a broad description of the taxonomy of the different kinds of statistical questions that arise in machine learning. He also introduced a new procedure for testing which of two classifiers is more accurate, called the 5x2cv t-test, based on five iterations of two-fold cross-validation. In each replication, two algorithms are trained on each fold and tested on the other fold. The test statistic is then a modified statistic of a paired t-test, where the s.d. comes from the cross-validation. The 5x2cv test was improved later. Alpaydín17 introduced a robust 5x2cv F-test, while Bouckaert18 questioned the theoretical degrees of freedom and corrected them on the basis of the dependences between experiments. However, these methods do not fit into the current trends in machine learning algorithms, where new algorithms are tested against multiple, state-of-the-art models and over several data sets. For this purpose, more extensive methods began to be used. One of these is analysis of variance (ANOVA)29, for example, using Friedman's test30,31 for the comparison of multiple models, but this can only yield conclusions about whether there are differences in the performance of models. If such differences exist, post hoc tests are needed to determine which model performs better. One of the procedures used for such post hoc analysis is the Nemenyi test, which gives the statistical significance of the performance difference between a pair of models. The results of the Nemenyi test for many models can be aggregated as a critical difference plot that shows groups of models that are not different. However, these results are not transferable to new data sets (which were not used during testing) because the test results have no absolute meaning. Therefore, once a ranking is obtained, it cannot be extended without repeating the entire testing procedure. The first to use non-parametric tests for comparing models on multiple data sets was Hull22. Brazdil and Soares23 used ranks to compare classification algorithms but do not provide statistical tests.

![Table 1: Descriptions of the EPP benchmark components](image)

Demšar15 analysed papers from five years of the International Conference on Machine Learning (1999–2003) that compared at least two classification models. The conference papers included a wide range of approaches, from naïve average accuracy over all data sets, through counting the number of times a model performed better than the others, to assessing statistical significance by using pairwise t-tests. However, despite multiple hypothesis testing, only a few articles included Bonferroni correction, a method to adjust the P values of tests in case of multiple comparisons.

The conclusion of that analysis was that there is no well-established procedure for comparing algorithms over multiple data sets. Furthermore, there issues with common measures of model performance, such as the uninterpretability of differences between two values or the inability to compare these values between data sets (EPP handles the problems with common ML performance measures section). Therefore, there is an emerging need to develop better solutions for benchmarking of models. Here, we introduce a method for model comparison that is based on the Elo ranking system used in sports, for example, chess and football.

Elo ranking system. The rating introduced by Elo34 is a ranking system used for calculating the relative level of a player's skill. The difference between the Elo ratings of two players can be transferred into the probabilities of winning when they play against each other. Therefore, the difference in Elo scores is a predictor of the match result calculated on the basis of the history of the players' matches. The scores of the players are updated after each match they play, and a new rating is calculated on the basis of two components: the result of a match and the rating of the opponent. A player's level is not measured absolutely but is inferred from their wins, losses and draws against other players. After each match, the winner gains Elo points. The amount of points received is related to the strength of the opponent. If a player beats an opponent who has a higher Elo score, the victor will gain more points than if playing a weaker opponent. Conversely, a defeated player would lose more points than if losing a match against a player with a lower rating.

There are many variations of the Elo rating. One of them, popular in chess, is the algorithm of 400. It states that an average player has a rating of 1,500, and reaching a rating over 2,000 means that
the player is one of the best. Let us consider two players, Player_1 and Player_2. Player_1 has an expected score of \( E_1 \), which is their probability of winning plus half of the probability of drawing with Player_2, expressed as

\[
E_1 = \frac{1}{1 + 10^{(S_2 - S_1)/400}},
\]

where \( S_1 \) and \( S_2 \) are the ratings of Player_1 and Player_2. This formula shows an essential property of Elo scores: the possibility to interpret them in terms of the probability of winning. For example, a difference of 200 rating points means that a more skilled player has a probability of winning of \( \frac{1}{1 + 10^{(200)/400}} \approx 0.75 \).

In addition to this probabilistic interpretation, the Elo rating has one more advantage, that is, it is not necessary for every player to play against all others to provide a comparison of their skills. In the real world, it would be impossible to stage matches between all chess players. Therefore, the Elo rating is used to find an approximation to true skill. Of course, the more matches played, the better this approximation. However, not all players need to play against all others.

**Unified benchmark ontology**

In this section, we introduce the unified benchmark ontology for machine learning that fills the gap for a uniform description of benchmarks. Extended Data Fig. 1 contains a unified diagram for describing machine learning benchmarks. We use terms associated with sports tournaments, such as 'Player', 'Tournament', 'Round' and 'Leaderboard'. The detailed descriptions of all these components are provided in Table 1. In this article, whenever we refer to the components of a unified benchmark ontology, we indicate this with capitalization of the term. Each component may be assigned to a different machine learning element. The set of such assignments is a 'Scheme'. Examples of Schemes are provided in the next subsection.

**Example schemes.** Extended Data Table 1 presents example schemes, that is, mappings between components of a unified ontology benchmark and machine learning terms. The Model/CV Scheme is one of the most standard benchmarking settings where models are compared on different cross-validation splits. The Model/Task Scheme covers a situation when models are compared on several data sets where each is assigned to its own performance measure. The pair formed of a data set and a performance measure is a task. Examples of such benchmarks are SuperGlue and the VTAB. The third example Scheme is Data Set/Model. The aim here is to compare data sets and assess how high performance models can score on it. This can be useful when assessing how simple it is to train a good model on a particular data set.

Depending on the Scheme assumptions, the Scores for a particular Player may be independent (when Rounds are different data sets) or correlated (when Rounds are cross-validation splits) across Rounds. In both cases, we assume that, within a Round, the Players’ Scores are comparable and can be ranked against each other according to a given order relation.

**Real data examples**

In this section, we show that the EPP meta-score improves existing real data benchmarks for tabular data as well as computer vision and natural language processing problems.

In Computing EPP on the OpenML100 benchmark section, we consider the OpenML100 benchmark for table data\(^{19}\), which fits the Model/CV Scheme from Example schemes section. Every model (Player) is tested on a different train/test split of cross-validation (Round). The rankings are created per data set (Tournament).

To broaden the perspective, in Supplementary Section S1 we show the VTAB benchmark illustrating different Schemes. Similarly to the OpenML, the compared items are different neural architectures (Players), but the Rounds are determined by different independent tasks.

**Computing EPP on the OpenML100 benchmark.** Now, we demonstrate the advantages of the EPP meta-score in Leaderboards on a MementoML\(^{26}\) database, a large-scale benchmark on 30 binary classification data sets from the OpenML. We selected five machine learning algorithms: gradient boosting machines \((\text{gbm})\), a generalized linear model with regularization \((\text{glmnet})\), \(k\) nearest neighbours \((\text{kknn})\) and two implementations of random forest \((\text{RF} \text{ and } \text{ranger})\). Each algorithm was trained with 400 different, randomly chosen hyperparameter configurations. For each data set, models were tested on 20 random train/test splits with the AUC as a performance measure. This gave us an overall number of AUC values equal to \(30 \times 20 \times 5 \times 400 = 1,200,000\).

On the computed AUC scores, we calculated the EPP meta-score introduced in Methods section and Fig. 1. As a single ‘Round’, we consider a ‘train–test split’. A Match is a comparison of the performance of two models with specified hyperparameters on the same data set but not necessarily on the same ‘train–test split’. As a result, we obtained EPP values for each data–model–hyperparameter combination, yielding us \(30 \times 5 \times 400 = 60,000\) EPP meta-scores values.

The performance of models is highly variable depending on the data set, which can be seen in Fig. 2, where the distributions of the EPP meta-score values across models and data sets are shown. A longer boxplot means greater potential for model tuning. For example, one can see that tree-based models \((\text{gbm}, \text{RF} \text{ and } \text{ranger})\) perform better on the ‘madelon’ data set than the other two models do. Also, all the EPP values for random forest are positive, which means that, generally, its performance is above average. Due to the independent sampling of hyperparameters and an excessive \(L\) penalty in regularization, a part of the \text{glmnet} models achieved AUC scores equal to 0.5 or less. The models with AUC of 0.5 always lose.
against other models, which causes a huge range of values of EPP scores for glmnet.

In Extended Data Table 3 presents the AUC and EPP values for the four selected models for the 'ada_agnostic' data set from the experiments described above. To recall, in There is no interpretation of differences in performance section, in the example of Kaggle ranking, we postulate that the AUC score does not provide a probability interpretation. EPP addresses this issue, so we can assess the probability of one model's winning against another according to property 5.1. The descending order according to the averaged AUC is different from the EPP ranking. The lowest EPP value is found for the kknn model, although the lowest averaged AUC corresponds to glmnet. The difference between the AUC of the first and second model is 0.001497, while the difference between the AUC of the third and fourth score is 0.00395. Using the EPP, one can estimate the probability that gbm beats ranger as logit(1.27 – 1.08) ≈ 0.55. For the second pair of models (glmnet and kknn), despite the closeness to the averaged AUC, there is a 0.83 likelihood that glmnet will defeat the kknn model. These dissimilarities are not emphasized by the AUC score because the averaged cross-validation scores miss the variability of the metrics.

Fig. 2 | Boxplots of EPP scores split by different algorithms across data sets. The sample size for every boxplot is n = 400 points. Boxes represent quartiles: 25% quantile, median and 75% quantile. Horizontal bars indicate the largest (smallest) value within 1.5 times the interquartile range above (below) the 25% (25%) quantile. Each boxplot aggregates EPP scores of all models trained on all data sets.
With respect to property 5.4, the EPP meta-score enables an analysis of the performance between data sets. Because of the lack of interpretation of AUC differences, comparisons between model scores can be carried out in various ways, as described in detail in There is no way to compare scores between Tournaments section. Extended Data Tables 4 and 5 present the rankings for the best-in-class models for two data sets from our experiment: 'mozzilla4' and 'credit-g'. Even though the absolute differences of AUC between the first and second model in each ranking are around 0.002, the rankings have different levels of AUC scores (approximately 0.98 and 0.81 for 'mozzilla4' and 'credit-g', respectively), so distinct approaches provide dissimilar claims. The EPP overcomes this problem, so we can draw consistent conclusions regardless of the absolute value of a considered metric. Due to the differences in the EPP values, the 'mozzilla4' data set gbm model has a 0.77 probability of beating the best ranger model. In the 'credit-g' ranking, the RF model has a likelihood of only 0.53 of defeating the ranger model.

Recalling property 5.8, for every data set (EPP Leaderboard), we can use the deviance statistics to compare the quality of rankings. The two Leaderboards related to the lowest and highest divi 

described ontology for the Model/CV Scheme, where the single Round is a single data set split. Here, we consider as one round the comparison of scores on different train–test splits. This approach allows for more matches between players, and the EPP meta-score values are more reliable. This extension is valid because of the properties of cross-validation and the assumption of estimating the same value of the performance measure across train–test folds. Alongside this issue arises the question of the stability of EPP meta-score values and how many matches are needed to estimate EPP.

**Conclusions and future work**

Here, we introduce a new performance meta-score known as EPP. By introducing the unified benchmark ontology, we demonstrate how universal and applicable the EPP measure is across different machine learning domains. In addition, we highlight the most important objections regarding existing metrics and point out EPP properties that address these limitations.

The versatility combined with the statistical properties of EPP enhances the inference that comes from existing benchmarks, which is shown in the use case of OpenML. The most important is the possibility of transforming differently defined evaluation scores to the same scale, that is, the probability of winning against a competitive model. On the VTAB benchmark shown in Supplementary Section S1, we show that the EPP Leaderboard amplifies the original approach and provides a confidence interval for the EPP value, enabling an assessment of the significance of differences between algorithms. Based on the OpenML repository, we illustrate how EPP empowers the systematic benchmark with the well-defined space of machine learning models and hyperparameters. The EPP meta-score enables a comparison of the predictive power for different Tournaments, in this case data sets. The Tournaments are comparable in terms of the quality of the EPP Leaderboards.

Hence, EPP can be considered as competitive to commonly applied scores in rankings of machine learning challenges and as an alternative to existing approaches to aggregating scores. Moreover,
EPP extends the existing benchmarks and does not require them to be recomputed (see the VTAB use-case in Supplementary Section S1).

The EPP is a better alternative to the empirical probability, which can only be calculated for a pair of models. Moreover, the EPP meta-score has statistical foundations and should be a key aspect of further research. One remaining need is to examine how the interdependence of Rounds affects the EPP meta-score estimation and how many Rounds are required to obtain stable EPP values. In addition, it may be worth considering models other than logistic regression for EPP Score estimation, such as the probit model.

Owing to the interpretation of differences and the comparability of EPP across diverse data sets, new measures provide the opportunity to research and verify state-of-the-art AutoML benchmarks in a new light. So far, researchers have had to make assumptions to simplify finding the optimal configuration of algorithm settings across multiple data sets. The EPP meta-score does not require the same scale of score and additionally provides an interpretation for comparing Leaderboards. The second major opportunity is to use EPP for navigated hyperparameter tuning. EPP score can be used to assess the probability that the performance can be improved by continuing to search the hyperparameter space.

Methods

In this section, we introduce the EPP benchmark that fits into the nomenclature introduced in Unified benchmark ontology section. In Idea of EPP meta-score section, we show key concepts of the EPP score, while in EPP handles the problems with common ML performance measures section, we show the common problems with state-of-the-art benchmarking methods and derive the properties of the EPP score that overcome such issues.

Figure 1 presents the EPP benchmark with the nomenclature from the universal benchmark ontology. The thick arrow from Extended Data Fig. 1 is broken down here into additional components such as Opponents and Matches. The detailed descriptions of components specific to the EPP benchmark are presented in Extended Data Table 2.

Idea of EPP meta-score. The EPP meta-score is used for establishing the Players' Leaderboard according to a single Round of the experiment. The Players are ranked according to their Score values in a single Round. However, the order of Players on the Leaderboard consistent with all Rounds may be impossible to determine. A single order might not have the property of connectivity with all Rounds and therefore be non-linear. This is why the common procedure is to aggregate, for example, as an average, the Scores over Rounds and then obtain the Leaderboard. However, the mean is sensitive to outlier observations, thus models with strongly varying results will distort the aggregated ranking. In this paper, we introduce an alternative approach, that is, EPP. The idea is not to aggregate values of Scores but rather to compare the relative performances of Players. We ignore the absolute values of Players' Scores, and the winner is the one whose Score is better (in terms of a given order relation). Every Round, consists of Match, in which Player, competes with another Player, (Opponent,). In consequence, for every pair of Players, we get the sequence of win–lose results for every Round and can use this table for calculating the relative EPP Scores of Players' performances. This relativity of the Players' performance makes EPP very similar to Elo, in particular that both methods give a probabilistic interpretation of differences in score values.

Elo ranking was dedicated to sports that suffer from practical limitations that do not arise in the field of machine learning. In classical Elo ranking, every Player does not face every other. Indeed, 100 Players would have to play 4,950 Matches, which might be impossible for logistical or time reasons. Therefore, it is often hard to use all possible results of Matches. In the case of machine learning models, the cost of calculating EPP meta-scores is not as time-consuming as human Matches. It is worth noting that a Match result is a comparison of Players' performances in one Round, and the performance for a specific Score is the same, regardless of the Opponent. Therefore, for 100 Players, we can obtain the results of all Matches by calculating only 100 values (the performance of each Player in each Round). It is worth noting that it is not always possible to get the Score value for each model in every Round, for example, owing to missing data, which some only models can deal with. However, EPP can still be calculated in the presence of missing Players' Scores.

In classical Elo ranking, the scores are updated after consecutive matches. Therefore, there is a natural order of updates. As the number of Elo points given to the winning player depends on both their and their opponent's Elo score, the order in which the matches are played may affect the final Elo scores. However, note that the need to calculate Elo sequentially is due to the aforementioned weakness, that is, the inability to play matches between all the players at once. Therefore, Elo ranking is sensitive to selective pairing. Two players with the same results can end up with different Elo Scores depending on the order of the Matches. In Elo ranking, when a new Player is added to the Tournament, this Player cannot compete with all the other Players at once. Therefore, the new Player can play only with a subset of Tournament Players. Depending on this subset of Tournament Players, the new Player's Elo Score would vary. We propose an EPP model scoring method that does not require the sequential calculation of Match results and preserves the desired Elo properties, that is, the possibility of interpretation on an interval scale.

It is worth noting that Elo introduced a solution to the problem of how to measure the skill of all players with only partial information about the outcome of matches. The EPP score applies the Bradley–Terry model to machine learning models and therefore provides a direct way to calculate the values approximated with Elo. For this reason, the order of model comparisons is irrelevant for EPP.

Definition of EPP score. Now, we formally define the EPP meta-score in the terminology of the unified EPP benchmark. Let \( \mathcal{M} = \{M_1, M_2, \ldots, M_m\} \) be a set of \( m \) Players. For a selected single Tournament \( T' \), we specify a set of R Rounds \( \mathcal{R} = \{R_1, R_2, \ldots, R_r\} \) and Score.

Denote the result of a single Match in a Round \( R_k \) between Players \( M_i \) and \( M_j \) as \( \gamma_{ij} \), where \( \gamma_{ij} = 1 \) if Player \( M_i \) has a better Score than Player \( M_j \), in Round \( R_k \), \( \gamma_{ij} = 0.5 \) if both Players have the same Score value, and \( \gamma_{ij} = 0 \) otherwise.

Elo29 proposed a solution to the problem of how to measure the skill of all players with only partial information about the outcome of matches. The EPP score applies the Bradley–Terry model to machine learning models and therefore provides a direct way to calculate the values approximated with Elo.

Definition 1. The odds(\( \gamma_{ij} \)) (i,j) are the odds that Player \( M_i \) has a better Score than Player \( M_j \), and are expressed as

\[
\text{odds}(i,j) = \frac{p_{ij}}{1 - p_{ij}}
\]

where \( p_{ij} \) is the probability that Player \( M_i \) has a better Score than Player \( M_j \) in a random R.

Definition 2. The \( \beta_{M_i} \) and \( \beta_{M_j} \) are the EPP meta-scores for Players \( M_i, M_j \in \mathcal{M} \), respectively, if they satisfy the property

\[
\log \frac{p_{ij}}{1 - p_{ij}} = \beta_{M_i} - \beta_{M_j},
\]

where \( p_{ij} \) can be estimated as \( \hat{p}_{ij} \) in two exploratory variables logistic regression of the form

\[
\log \frac{\hat{p}_{ij}}{1 - \hat{p}_{ij}} = \beta_{M_i} + \beta_{M_j} x_{ij},
\]

where \( x_{ij} = 1 \) and \( x_{ij} = -1 \).
Note that the EPP meta-scores $\hat{\beta}_M$ and $\hat{\beta}_M$ are determined by logistic regression using the Newton–Raphson algorithm, although the algorithm will fail to converge in some cases and we will not get values that satisfy equation (1), or infinite values of coefficients. The latter issue appears in extreme cases, for example, where there are two separable groups of players, that is, very good players and very poor ones. EPP is estimated by logistic regression with an iterative reweighted least-squares algorithm of computational complexity $O(q^2nq^3)$, where $n$ is the number of observations and $q$ is the number of features. In our setting, $q = m$, where $m$ is the number of players, so the estimation of EPP has complexity $O(m^2)$.

**Definition 3.** The EPP meta-score Leaderboard for Tournament $T$ is the set of EPP meta-score values for the set of $m$ Players $\mathcal{M} = \{M_1, M_2, \ldots, M_m\}$:

$$L_T^m = \{\hat{\beta}_M, \ldots, \hat{\beta}_M\}.$$  

The properties of the EPP meta-score are described in the next subsection.

**EPP handles the problems with common ML performance measures.** In this section, we identify problems with the most common performance measures used in ML benchmarks and show that the EPP meta-score handles these issues. The attributes of the EPP meta-score can be described with three aspects:

- **EPP is a meta-approach based on values of other performance measures.** EPP broadens the possibilities of comparing Players because of its unique properties introduced below.
- **EPP is an alternative approach to aggregating results, such as mean scores for repeated measurements, that is, Rounds.** EPP gives the statistical possibility to assess the stability of Scores (Mean aggregation may be misleading: the variance for Round and The mean aggregation may be misleading: the variance for Player sections).
- **Unlike the methods used so far, EPP gives the possibility to compare benchmarks.** It allows the assessment of the quality of Leaderboards across Tournaments (You cannot assess the quality of a Leaderboard section).

It is worth noting that one objectively best method for model performance aggregation does not exist. This issue can be compared to the problem of evaluating the performance of two binary classification models. Either accuracy or the F1 score can be used to assess their quality. The choice of one of those measures depends on the user’s needs. Of course, it can be shown that certain properties of one measure are more favourable in a given case, yet this does not ensure objectively that one measure is better than the other. Similarly, EPP is an alternative to the mean as an aggregation method for a performance model.

The following sections are structured as follows: First, we discuss a problem and a new method for solving it. Second, we discuss a problem with existing methods and the problem at hand. We describe such properties in terms of the EPP benchmark. This distinction helps to better separate the examples from the theoretical part.

There is no interpretation of differences in performance. Use case. Extended Data Table 6 presents an example of Kaggle ranking. The difference between the AUC value of the first and the second model is 0.00018. However, this absolute difference provides no additional information. The AUC is useful for ordering models, but its differences have no interpretation. It does not provide any quantitative comparison of the performance of models. There is no single accepted way to compare the power of enhancement of performance measures. Some say we should equate absolute differences regardless of the absolute values of the score, while others would suggest analysing the relative improvement. However, these two approaches may lead to opposite conclusions depending on the absolute value of a performance measure.

**Solution.** This ranking fits for the EPP benchmark Scheme Model/Task with just one Round. The EPP score provides a direct interpretation in terms of probability.

**Property 1.** The difference between the EPP scores for Players $M_i$ and $M_j$ is the logit of the probability that $M_i$ achieves better performance than $M_j$. Indeed, from definition 2, we have that

$$\log \left( \hat{p}_{ij} \right) = \log \left( \frac{\hat{\beta}_j - \hat{\beta}_i}{1 + \hat{\beta}_i - \hat{\beta}_j} \right) = \beta_{M_j} - \beta_{M_i}.$$  

After reformulating equation (2), we obtain a direct formula for the probability that Player $M_i$ achieves better performance than Player $M_j$:

$$\hat{p}_{ij} = \text{invlogit} \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_j} \right) = \frac{\exp \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_j} \right)}{1 + \exp \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_j} \right)}.$$  

There is no method to assess the significance of differences. Use case. Extended Data Table 7 presents the results of an IEEE-CIS Fraud Detection Kaggle Competition. The AUC values of all the models in Extended Data Table 7 differ in the third decimal place. There is no reference point to indicate whether this difference represents a significant improvement in prediction or not. Significance in the statistical sense means that these differences are not on the noise level. The scoring of models in this example is deployed in a Scheme Model/Task with just one Tournament and one Round. A similar situation with minor differences appears in many state-of-the-art benchmarks. When a Player gains improvement by a decimal place, it would be desirable to distinguish between a real improvement and an apparent improvement owing to the noise coming from different Round settings, for example, splitting into train and test data. Currently, there are no many formal methods to assess the significance of differences. One way is to use a Kruskal–Wallis test for the equality of medians. Nevertheless, results from statistical tests are not transitive. We can compare two Players, but we would not get an overall Leaderboard for all of them.

**Solution.** The EPP score enables the assessment of the significance of a score value, which provides an intuition regarding whether a difference in performance is due to noise or not.

**Property 2.** EPP meta-score values are coefficients of a logistic regression model with intercept $\hat{\beta}_0 = 0$.

Equation (2) can be generalized to Note that, if $i \neq j$, then $x_{M_i} = 0$ and $\hat{\beta}_M$ is not considered, so this equation is in line with equation (2).

The EPP scores are the estimates of the unknown $\hat{\beta}_M$ coefficients from the multiple exploratory variables logistic regression where $x_{M_i}$ indicates whether the Player is computed. Because the EPP meta-score values are calculated by logistic regression, a logit of probabilities gives an additional benefit in the form of providing a significance of EPP scores. This is an advantage over raw empirical probabilities.

**Property 3.** The statistical significance of the difference between EPP for two Players $M_i$ and $M_j$ may be tested as the null hypothesis that

$$\hat{\beta}_{M_i} = \hat{\beta}_{M_j}.$$  

If Round performances are independent and the sample size is sufficiently, this hypothesis may be tested with a Wald test or likelihood ratio test.

However, even when assumptions about the independence of splits are violated, and observations appear in different bootstrap samples, one can rely on test results as they are robust. Another way is to use approximately unbiased bootstrap resampling (15).

There is no way to compare Scores between Tournaments. Use case. In Extended Data Tables 6 and 7, the differences between the second and third best models for each data set are around 0.00008. The question that then arises is whether these differences are comparable between data sets. Does 0.00008 on Springer Marketing data mean the same increase in model quality on IEEE-CIS fraud data? There are at least three points of view. One is that the gaps are almost the same for both data sets because the differences in AUC values are almost the same. The second is that the gap in the IEEE-CIS Fraud Competition is larger as the AUC value is close to 1. The relative improvement for the fraud detection data ($\log \frac{0.967637}{0.967722} \approx 0.00026$) is larger than the relative improvement for the Springer Marketing data ($\log \frac{0.999081}{0.999049} \approx 0.0004)$.

The third point of view is that the gap between the first and second place for Springer (0.00018) is smaller than the same difference for IEEE-CIS fraud detection (0.000415). Therefore, the relative gain from the difference between second and third place for Springer is higher.

**Solution.** From the definition of EPP score, the probability of winning against an average Player (equivalent to an intercept $\hat{\beta}_0$) has the same meaning, regardless of the Tournament. The EPP scores are absolute values with a mean of zero. Therefore, comparing EPP values between Tournaments is possible by comparing a probability of winning against an average Player.

**Property 4.** The probability that Player $M_i$ would win against an average Player $M_{avg}$ is

$$\hat{p}_{i,avg} = \frac{\exp \left( \hat{\beta}_{M_i} \right)}{1 + \exp \left( \hat{\beta}_{M_j} \right)}.$$  

From property 5.2, we have that the intercept is $\hat{\beta}_0 = 0$. In the logistic regression, the intercept relates to the mean, therefore $\hat{\beta}_{M_{avg}} = 0$ and

$$\hat{p}_{i,avg} = \text{invlogit} \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_{avg}} \right) = \frac{\exp \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_{avg}} \right)}{1 + \exp \left( \hat{\beta}_{M_i} - \hat{\beta}_{M_{avg}} \right)} = \frac{\exp \left( \hat{\beta}_{M_i} \right)}{1 + \exp \left( \hat{\beta}_{M_i} \right)}.$$  

Mean aggregation may be misleading: the variance for Round. Use case. Figure 4 shows four selected models from the VTAB. Every small point indicates the top-1 marks. It allows the assessment of the quality of Leaderboards across Tournament.
We analyse the results of these models within pairs. The first pair is the Sup-Rotation-100% model and the Sup-Exemplar-100% model. The second pair is the Uncond-BigGAN model and VAE model. Averaged top-1 accuracies across data sets are close to models in these pairs. The Sup-Rotation-100% and Sup-Exemplar-100% models have evidently higher predictive power than the Uncond-BigGAN and VAE models for most tasks.

The top-1 accuracy scores of the Sup-Rotation-100% and Sup-Exemplar-100% models are very close to each other within specified data sets. This is represented by parallel lines connecting dots for two models in the top part of the plot. In the second pair, the relationship between Uncond-BigGAN and VAE is more ambiguous. Comparing average across data sets shows that the Uncond-BigGAN model is comparable to VAE. However, when considering the green path in Fig. 4, all four models show similar performance. On the other hand, when considering the black path, VAE significantly defeats Uncond-BigGAN and performs comparably to Sup-Rotation-100% and Sup-Exemplar-100%. The probable reason for these two scenarios is the different tunability of data sets, which appears through a diverse variance of top-1 accuracy for different models.

Solution. The VТАВ ranking exemplifies the Scheme Model/Task of the EPP benchmark with just 19 Rounds determined with data sets. Averaging the Score values neglects the information about the distribution of Score values within Rounds. This is especially relevant in cases where we compare the Scores of different definitions and the range of values for a sequence of Rounds. EPP also ignores the dispersion of Score values for a specified Round, but this simplification comes, by design, from the definition of EPP computing.

Property 5. The EPP score is an aggregate over all rounds. By fitting the logistic regression model from equation (3) as dependent variables of observations, we use the results of Matches, whether one Player beats another.

The mean aggregation may be misleading: the variance for Player captures the relationship between the models' performance better than the Leaderboard captures the relationship between the models' performance better than the Leaderboard. Therefore, the model aggregation method may be misleading: the variance for Player cannot be calculated using the deviance statistic. If $D(P_{M}^{T}, P_{M}^{T}) ≤ D(P_{M}^{T}, P_{M}^{T})$, the Leaderboard $L_{M}^{T}$ for Tournament $T$ is better fitted to the actual probabilities than the Leaderboard $L_{M}^{T}$ for Tournament $T$.

The number of degrees of freedom of the deviance statistic is on the order of $m$ Therefore, deviance statistics can take very large values if the number of Players is on the order of 100 or higher. From the $x'$ distribution properties, the skewness of this asymmetric distribution decreases with the number of Players $m$. If the number of Players is sufficiently high, $D(P_{M}^{T}, P_{M}^{T})$ converges to a distribution with a normal distribution, $\mathcal{N}(m(m-2), 2m(m-2))$. The deviance statistics can be scaled and shifted depending on the number of Players.

Property 8. Given the set of $M = \{M_1, M_2, ..., M_n\}$ Players and two Tournaments $T_1$ and $T_2$, the quality of the EPP meta-score Leaderboards $L_{M_1}^{T_1}$ and $L_{M_2}^{T_2}$ can be compared using the deviance statistic. If $D(P_{M_1}^{T_1}, P_{M_1}^{T_1}) ≤ D(P_{M_2}^{T_2}, P_{M_2}^{T_2})$, the Leaderboard $L_{M_1}^{T_1}$ for Tournament $T_1$ is better fitted to the actual probabilities than the Leaderboard $L_{M_2}^{T_2}$ for Tournament $T_2$. This means that Leaderboard $L_{M_1}^{T_1}$ captures the relationship between the models' performance better than Leaderboard $L_{M_2}^{T_2}$.

An alternative approach to comparing deviance statistics is to compare $P$ values corresponding to the chi-square distribution (property 5.8) or for the standard normal distribution (property 5.10) for deviance statistics. A higher $P$ value corresponding to deviance statistics indicates that the Leaderboard fits more accurately. However, with an increasing number of Players, one may observe the discretization of $P$ values related to the deviation for tournaments.

Data availability

The data sets generated during the current study are available in the EPP meta-score GitHub repository available at https://github.com/agosewks/ EPP-meta-score. Source data are provided with this paper.
Received: 17 August 2021; Accepted: 5 August 2022; Published online: 22 September 2022

References

1. Wang, A. et al. GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: analyzing and interpreting neural networks for NLP (eds. Linzen, T., Chrupała, G. & Alishahi, A.), 353–355 (Association for Computational Linguistics, 2018).

2. Wang, A. et al. SuperGLUE benchmark for general-purpose language understanding systems. Adv. Neural Inform. Process. Syst. 3261–3275 (2019).

3. Zhai, X. et al. A large-scale study of representation learning with the Visual Task Adaptation Benchmark. Preprint at https://arxiv.org/abs/1910.04867 (2020).

4. Kryuxtapovsky, A., Schwed, T., Topf, M., Fidelis, K. & Moult, J. Critical assessment of methods of protein structure prediction (CASP) - round XIII. Protins 87, 1011–1020 (2019).

5. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. Nature 596, 583–589 (2021).

6. Zhou, N. et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. Genome Biol. 20, 1–23 (2019).

7. Lensink, M. F., Nadzirin, N., Velankar, S. & Wodak, S. J. Modeling protein–protein, protein–peptide, and protein–oligosaccharide complexes: CAPRI 7th edition. Proteins: Structure, Function, and Bioinformatics 88, 916–938 (2020).

8. Vanschoren, J., van Rijn, J. N., Bischl, B. & Torgo, L. OpenML: networked science in machine learning. SIGKDD Explor. Newsl. 15, 49–60 (2014).

9. Martinez-Plumed, F., Barredo, P., Hughearkh, S. Ó. & Hernandez-Orallo, J. Research community dynamics behind popular AI benchmarks. Nat. Mach. Intell. 3, 581–589 (2021).

10. Powers, D. Evaluation: from precision, recall and F-factor to ROC, informedness, markedness & correlation. J. Mach. Learn. Technol. 2, 37–63 (2008).

11. Sokolova, M. & Lapalme, G. A systematic analysis of performance measures for classification tasks. Inform. Process. Manage. 45, 427–437 (2009).

12. Pedregosa, F. et al. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830 (2011).

13. Abadi, M. et al. TensorFlow: a system for large-scale machine learning. In 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI’16), 265–283 (USENIX Association, 2016).

14. Bischl, B. et al. mlr: machine learning in R. J. Mach. Learn. Res. 17, 1–5 (2016).

15. Demšar, J. Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30 (2006).

16. Dietterich, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput. 10, 1895–1923 (1998).

17. Alpaydin, E. Combined 5 × 2 cv F test for comparing supervised classification learning algorithms. Neural Comput. 11, 1885–1892 (1999).

18. Bouckaert, R. R. Choosing between two learning algorithms based on calibrated tests. In Proceedings of the Twentieth International Conference on International Conference on Machine Learning (eds. Fix, W., Tumer, K., et al.), ICML03, 51–58 (AAAI Press, 2003).

19. Salzberg, S. L. On comparing classifiers: pitfalls to avoid and a recommended approach. Data Min. Knowl. Discov. 1, 317–328 (1997).

20. Guerrero Vázquez, E., Yañez Escalona, A., Galindo Riaño, P. & Pizarro Junquera, J. in Bio-Inspired Applications of Connectionism (eds. Mira, J., Prieto, A.), pp. 88–95 (Springer, 2001).

21. Pizarro, J., Guerrero, E. & Galindo, P. L. Multiple comparison procedures applied to model selection. Neurocomputing 48, 155–173 (2002).

22. Hull, D. Information Retrieval Using Statistical Classification. PhD thesis, Stanford Univ. (1994).

23. Brazdil, P. B. & Soares, C. A comparison of ranking methods for classification algorithm selection. In Machine Learning: ECML 2000 (eds. López de Mántaras, R., Plaza, E.), 63–75 (Springer, 2000).

24. Elo, A. & Sloan, S. The Rating of Chess Players, Past and Present (Ishi, 2008).

25. Bischl, B. et al. OpenML benchmarking suites. In Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (eds. Ranzato, M., Beygelzimer, A., Dauphin, X., Liang, R.s. & Wortman Vaughan, J.), vol. 1 (Curran Associates, Inc., 2021).

26. Kretowicz, W. & Biecek, P. MementoML: performance of selected machine learning algorithm configurations on OpenML100 datasets. Preprint at https://arxiv.org/abs/2008.13162 (2020).

27. Probst, P., Boulesteix, A.-L. & Bischl, B. Tunability: importance of hyperparameters of machine learning algorithms. J. Mach. Learn. Res. 20, 1–32 (2019).

28. Bradley, R. A. & Terry, M. E. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrics 39, 324–345 (1952).

29. Clark, A. P., Howard, K. L., Woods, A. T., Penton-Voak, I. S. & Neumann, C. Why rate when you could compare? Using the “EloChoice” package to assess pairwise comparisons of perceived physical strength. PLOS ONE 13, 1–16 (2018).

30. Agresti, A. In Categorical Data Analysis, vol. 482, chap. 6 (Wiley, 2003).

31. Shimodaira, H. An approximately unbiased test of phylogenetic tree selection. Syst. Biol. 50, 492–508 (2002).

32. Shimodaira, H. Approximately unbiased tests of regions using multistep-multiscale bootstrap resampling. Ann. Stat. 32, 2616–2641 (2004).

33. Suzuki, R. & Shimodaira, H. Pvlust: an R package for assessing the uncertainty in hierarchical clustering. Bioinformatics 22, 1540–1542 (2006).

34. Agresti, A. In Categorical Data Analysis, vol. 482, chap. 4 (Wiley, 2003).

35. Gosiewska, A., Bakala, M., Woznicza, K., Zwozniak, M. & Biecek, P. EPP: interpretable score of model predictive power. Preprint at https://arxiv.org/abs/1908.09213 (2019).

36. Gosiewska, A. & Woznicza, K. agosiewska/EPP-meta-score: EPP paper. Zenodo https://doi.org/10.5281/zenodo.6949519 (2022).

Acknowledgements

Work on this project is financially supported by NCN Opus grant 2017/27/BST6/01307. We thank L. Bakala and D. Rafacz for inspiring ideas, W. Kretowicz and M. Zwozniak for preliminary work and P. Tenseyre, E. Stenkiewicz, H. Baniecki and B. Rychalska for useful comments.

Author contributions

A.G. and K.W. designed and implemented the EPP method as an R package, as well as, studied and described theoretical properties of the EPP. A.G. performed the EPP Leaderboard on the VTAB benchmark and developed the unified benchmark ontology. K.W. performed the EPP Leaderboard on the OpenML benchmark and designed and performed simulations in the Supplementary Materials. P.B. supervised the project, provided technical advice, helped design the method and analysed the experiments. All authors participated in the conceptualization and preparation of the paper.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-022-00531-2.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-022-00531-2.

Correspondence and requests for materials should be addressed to Alicja Gosiewska, Krzysztof Woznicza or Przemyslaw Biecek.

Peer review information Nature Machine Intelligence thanks Philipp Probst and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© The Author(s), under exclusive licence to Springer Nature Limited 2022.
Extended Data Fig. 1 | A unified Ontology of ML Benchmarks. The violet dashed rectangle shows a minimal setup for any benchmark.
## Extended Data Table 1 | Example Schemes for EPP Benchmark

| Component   | Scheme Model/CV                  | Scheme Model/Task                                      | Scheme Data Set/Model       |
|-------------|----------------------------------|-------------------------------------------------------|-----------------------------|
| **Player**  | Model                            | Model                                                 | Data set                    |
| **Score**   | Performance measure, for example, AUC | Score defined separately for each data set           | Performance measure, for example, AUC |
| **Round**   | Cross-Validation split           | Data sets with one train/test split each              | Model                       |
| **Tournament** | Data set                     | One data set                                         | Set of models               |
| **Leaderboard** | Separate rankings of models for each data set | One ranking of all models                            | One ranking of all data sets |
Extended Data Table 2 | The descriptions of the EPP Benchmark components that extend the Unified Benchmark Ontology

| Component   | Description |
|-------------|-------------|
| Opponent \(_{i,j}\) | \(Player_j\) whose \(Score\) values are compared to the \(Scores\) values of the \(Player_i\). |
| Match \(_{i,j,k}\) | A single comparison of the \(Score\) values of a pair of \(Players\), i.e. \(Player_i\) and \(Opponent_{i,j}\) in \(Round_k\). |
Extended Data Table 3 | EPP of selected models for ada_agnostic data set. AUC values are averaged. The numbers of models are IDs from the MementoML benchmark

| Model          | AUC  | EPP  |
|----------------|------|------|
| gbm1305        | 0.890| 1.27 |
| ranger1088     | 0.888| 1.08 |
| kknn1396       | 0.816| -7.52|
| glmnet1242     | 0.812| -5.91|
Extended Data Table 4 | The best models in algorithm class for mozilla4 data set. AUC values are averaged. The numbers of models are IDs from the MementoML benchmark.

| Model           | AUC  | EPP    |
|-----------------|------|--------|
| gbm1184         | 0.986| 7.49   |
| ranger1106      | 0.984| 6.25   |
| RF1106          | 0.984| 6.22   |
| kknn1016        | 0.942| -6.78  |
| glmnet1011      | 0.922| -11.24 |
**Extended Data Table 5** | The best models in algorithm class for credit-g data set. AUC values are averaged. The numbers of models are IDs from the MementoML benchmark

| Model       | AUC  | EPP   |
|-------------|------|-------|
| RF1155      | 0.809| 1.29  |
| ranger1212  | 0.807| 1.16  |
| gbm1136     | 0.807| 1.16  |
| glmnet1379  | 0.802| 0.97  |
| kknn1038    | 0.769| -0.54 |
**Extended Data Table 6 | Springleaf Marketing Response Kaggle Competition.**

[https://www.kaggle.com/c/springleaf-marketing-response](https://www.kaggle.com/c/springleaf-marketing-response)

| Team Name       | AUC   |
|-----------------|-------|
| Asian Ensemble  | 0.80925 |
| ARG eMMSamble .baGGaj. | 0.80907 |
| .baGGaj.        | 0.80899 |
Extended Data Table 7 | IEEE-CIS Fraud Detection Kaggle Competition. [https://www.kaggle.com/c/ieee-fraud-detection](https://www.kaggle.com/c/ieee-fraud-detection)

| Team Name       | AUC          |
|-----------------|--------------|
| AlKo            | 0.968137     |
| FraudSquad      | 0.967722     |
| Young for you   | 0.967637     |
Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a Confirmed

- The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- The statistical test(s) used AND whether they are one- or two-sided
  *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
- A description of all covariates tested
- A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- For null hypothesis testing, the test statistic (e.g. F, t, r) with confidence intervals, effect sizes, degrees of freedom and P value noted. Give P values as exact values whenever possible.
- For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- Estimates of effect sizes (e.g. Cohen’s d, Pearson’s r), indicating how they were calculated

Software and code

Policy information about: availability of computer code

Data collection
An implementation of the EPP score is available at [url](https://github.com/ModelOriented/EloML). The codes generated during the current study are available in the EPP-meta-score GitHub repository available at [url](https://github.com/agosiewska/EPP-meta-score) at citegosiewska2022epreprepo).

Data analysis
An implementation of the EPP score is available at [url](https://github.com/ModelOriented/EloML). The codes generated during the current study are available in the EPP-meta-score GitHub repository available at [url](https://github.com/agosiewska/EPP-meta-score) at citegosiewska2022epreprepo).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about: availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The datasets generated during the current study are available in the EPP-meta-score GitHub repository available at [url](https://github.com/agosiewska/EPP-meta-score) at citegosiewska2022epreprepo).
Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

- Life sciences
- Behavioural & social sciences
- Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

| Sample size | That is a large-scale benchmark on 30 binary classification data sets from the OpenML.
|             | We selected 5 machine learning algorithms: gradient boosting machines, a generalized linear model with regularization, k-nearest neighbours, and two implementations of random forest. Each algorithm was trained with 400 different, randomly chosen hyperparameter configurations. For each data set, models were tested on 20 random train/test splits with AUC as a performance measure. This gave us an overall number of AUC values equal to 30 * 20 * 5 * 400 = 1 200 000.

| Data exclusions | No data are excluded.

| Replication | To assess the stability of AUC measure we 10-fold cross-validation. To assess the results stability in simulations, computational experiments are replicated 12 or 20 times.

| Randomization | Randomization is not relevant in this study. There are experimental evaluation on the OpenML benchmark - often used by machine learning community.

| Blinding | Blinding is not relevant in this study since there is no subjective assessment - models are evaluated with AUC performance and then aggregated with EPP. Accuracy of EPP ranking may be validated with deviance.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

| Materials & experimental systems | n/a
| - Involved in the study
| - Antibodies
| - Eukaryotic cell lines
| - Palaeontology and archaeology
| - Animals and other organisms
| - Human research participants
| - Clinical data
| - Dual use research of concern

| Methods | n/a
| - Involved in the study
| - ChIP-seq
| - Flow cytometry
| - MRI-based neuroimaging