Special designs and competition protocols

Wei-Wei Tu TUWEIWEI@4PARADIGM.COM
The 4th Paradigm
China

Adrien Pavão ADRIEN.PAVAO@GMAIL.COM
LISN, CNRS
Université Paris-Saclay
France

Reviewed on OpenReview: *No*

Abstract

With the development of AI technology, many novel machine learning frameworks have been raised and applied in AI academic and industry research and business application. Organizing competitions in these areas can greatly help the research and development of related algorithms and technology. In this chapter, we explore the design of competitions in various kinds of machine learning field: supervised learning, automated machine learning, metalearning, time series analysis, reinforcement learning, adversarial learning, and using confidential data. For each of these specific competition protocol, we discuss the framework and design of the competition process. We believe this chapter can make great help to both the organizers and the participants, therefore accelerate the development of AI industry and research.

Keywords: competition, design, supervised learning, automated machine learning, metalearning, time series analysis, reinforcement learning, adversarial learning, confidential data

1 Introduction

Machine learning is an expansive field offering a rich diversity of algorithms, each developed to solve specific tasks. These algorithms are commonly grouped into three main categories: supervised learning, unsupervised learning, and reinforcement learning algorithms. Beyond the algorithms themselves, the possibilities are further augmented by the diversity of data and domains of applications. Depending on the nature of the data, its source, shape, quantity and patterns, different approaches are required. The applications of machine learning are virtually limitless, covering medicine, physics, natural language processing, economics, and more. To be able to capture this complexity and diversity in competitions and benchmarks, innovative experimental design is required.

In this chapter, we analyse the features and special designs about the challenges and benchmarks of supervised learning (Section 2), automated machine learning (Section 3), metalearning (Section 4), time series analysis (Section 5), reinforcement learning (Section 6), adversarial learning (Section 7), and using confidential data (Section 8). We also give some tips about how to perform well in these competitions as participants.

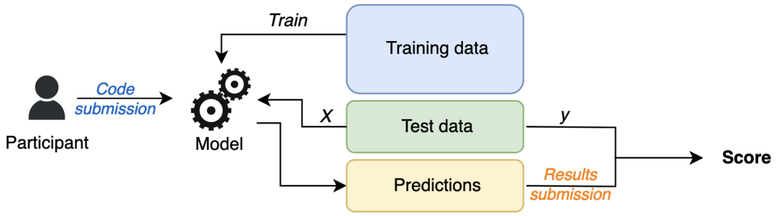


Figure 1: Supervised learning evaluation workflow. Models are trained and subsequently evaluated on the withheld test set. The two possible protocols, *code submission* and *results submission*, are illustrated. X represents the features, and y the ground truth, of the test set.

2 Supervised learning

Supervised learning is a foundational paradigm in machine learning where models are trained using labeled data. In this paradigm, for each input instance in the dataset, there is an associated correct output, commonly referred to as *label* or *ground truth*. The primary goal of supervised learning is to construct a model capable of making accurate predictions for unseen instances based on this training.

In a classic supervised learning competition, participants evaluate their models on a given task using a dataset split into training and test sets. Typically, as depicted in Figure 1, participants are provided with a training dataset to develop their models, and the evaluation is conducted on the withheld test set. Each competition phase must feature a different test set to prevent overfitting. Importantly, participants should not have access to the labels of the test set.

The choice of evaluation metrics in supervised learning challenges typically depends on the nature of the task — be it regression, classification, or others. The underlying goal is to objectively measure the performance of submitted models in terms of *accuracy*, *precision*, or other relevant metrics. Further insights into evaluation metrics are provided in the chapter 4.

3 Automated machine learning

Automated machine learning (AutoML) is a field of study that focuses on developing methods and systems that can automate the process of building machine learning models. The goal of AutoML is to make it easier to build accurate and effective machine learning models without requiring extensive human intervention. By nature, AutoML methods are built to be able to solve a wide variety of tasks. Examples of such competitions include the AutoML Challenge Series (Guyon et al., 2019), the AutoDL Challenge Series (Liu et al., 2021), the AutoML Decathlon (Roberts et al., 2022) and the AutoML Cup (Roberts et al., 2023) based on the NAS-Bench-360 benchmark (Tu et al., 2023).

The general competition protocol consists in evaluating the candidate algorithms on a set of m tasks. For each of these tasks, the model is trained from scratch and evaluated on a hold-out test set, as demonstrated by the diagram in Figure 2. The m scores that result from evaluating the algorithm across the tasks are subsequently fed into to a master scoring and ranking process. Although the scores obtained on various tasks could simply be averaged, we suggest computing the average of the ranks achieved by comparing all candidates across the given tasks. This approach ensures a more

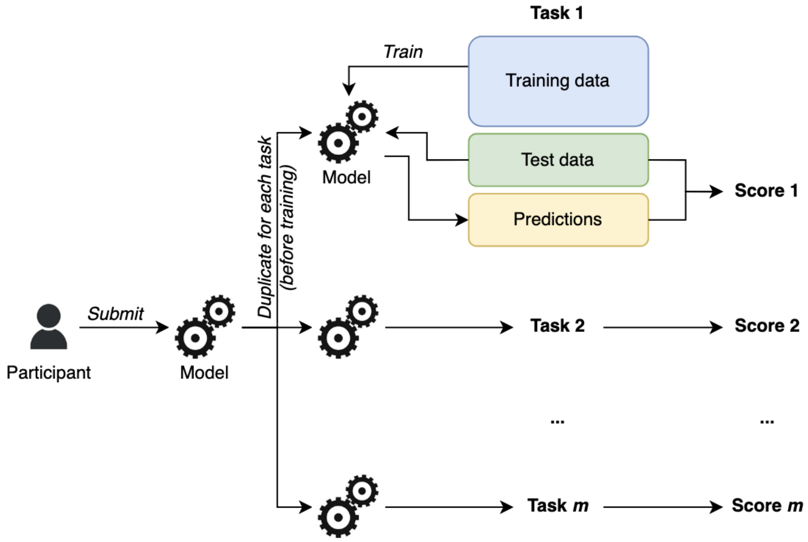


Figure 2: Automated machine learning evaluation workflow. The submitted model is trained and tested from scratch on a set of independent tasks.

robust ranking that accurately reflects the aim of a competition in automated machine learning. This point is explained in details in the chapter 4.

A key aspect of the experimental design of automated machine learning competitions and benchmarks is the **blind testing**. To accurately assess a model's capability to solve diverse and unrelated tasks, participants must not have access to the test data. While some example training datasets can be made available to help participants in developing their models, the feedback and final evaluation stages must be conducted blindly, typically through the submission of code rather than direct interaction with the test data.

The selection of datasets and evaluation metrics is flexible and intrinsically tied to the specific objectives of the challenge. The main principle is that greater diversity in datasets is likely to yield a winning solution with more general applicability. Reciprocally, using similar datasets and metrics is more likely to produce an algorithm specialized in a particular domain or task. Having a large number of different tasks, while computationally expensive, enhances the overall diversity of the model's capabilities. This topic is discussed in the chapter 4.

While most AutoML challenges focus on supervised learning tasks, classification and regression, this experimental design can be used to organize crowd-sourced competitions or benchmarks on the automation of other machine learning tasks, such as data processing, clustering, content recommendation and more. The pre-requisite is to have a scoring metric defining the objective of the problem.

4 Meta-learning

Meta-learning is a sub-problem of AutoML. In its general definition, AutoML is a process of automating the machine learning process, including tasks such as data preprocessing, feature engineering, model selection, and hyperparameter tuning. AutoML techniques use algorithms to search

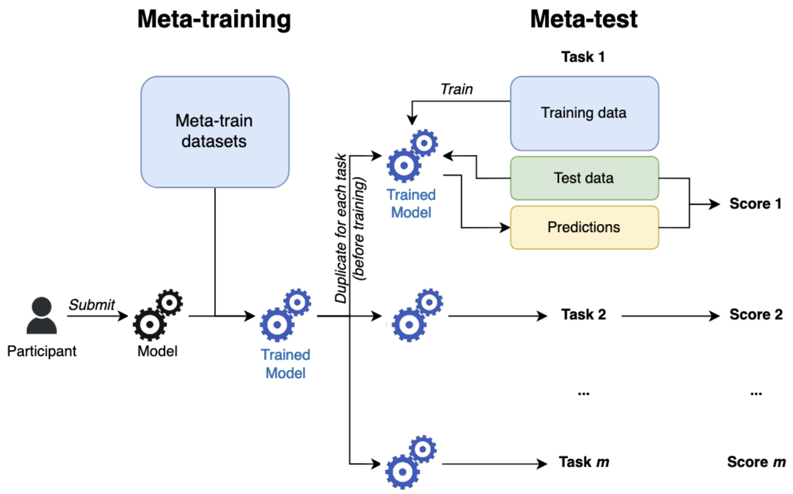


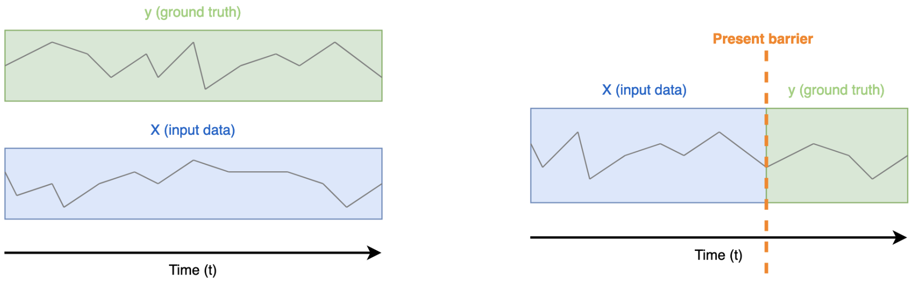
Figure 3: Meta-learning evaluation workflow. The submitted model is trained on meta-train datasets, and then it is tested on a set of meta-test tasks.

for the best machine learning pipeline automatically. On the other hand, meta-learning is focused on learning how to learn. Meta-learning algorithms learn from experience to adapt their learning strategies for different tasks and domains (Brazdil et al., 2022). In essence, while AutoML automates the process of finding the best machine learning pipeline for a specific task, meta-learning takes a step further and automates the process of improving the learning algorithm's generalization capability across multiple tasks.

In the meta-learning challenge protocol proposed by El Baz et al. (2021); Baz et al. (2021) for the Cross-domain MetaDL Challenge, the evaluation of the candidate algorithms is divided in two sequential phases: the meta-training and the meta-test. During the meta-training, the submitted algorithm is trained on a set of datasets. The trained model is then forwarded to the meta-test, where it will be trained and evaluated separately on a new set of tasks. The whole process is illustrated by the diagram in Figure 3. The set of scores produced is then used to compare the model with other candidate models. As for the AutoML Challenge, the entire process is conducted blindly, preventing the participants from adapting their approaches to the specific datasets used. The main difference with the AutoML protocol (Section 3) is the use of a controlled meta-training phase, which implies that all candidate algorithms are pre-trained on the same data.

5 Time series analysis

Time series analysis includes a wide variety of tasks, such as anomaly detection, sequence-to-sequence problems, or survival analysis, each presenting unique specificity. In the this section, our discussion centers around two central time series tasks: *time series regression* and *time series forecasting* (or prediction). While time series regression (Section 5.1) involves modeling the relationship between a dependent time-indexed variable and one or more independent variables, aiming to understand or predict the dependent variable's variations over time, time series forecasting (Section 5.2), on the other hand, is primarily concerned with predicting future values of a series based



(a) Time series regression

(b) Time series forecasting

Figure 4: Schematic view of time series regression (left) and time series forecasting (right). In regression, predictors can use past, present, or future values, while in forecasting, the task is to predict future values based on historical data and trends.

on its own past values and inherent patterns. This distinction is highlighted by Figure 4. The key distinction lies in the fact that regression models are more general and can be applied to predict values at any point in time, not strictly in the future, whereas forecasting is explicitly future-oriented, leveraging the temporal order of data to make predictions. Non sequential meta-data may also be available.

5.1 Time series regression

Time series regression is essentially a supervised learning task with a temporal dimension, where the goal is to predict a continuous target variable based on historical data. While it shares similarities with classical regression in terms of learning from input-output pairs and minimizing prediction error, the time component introduces dependencies between observations, necessitating consideration of the order and timing of data points in the modeling process. Time series regression can be multivariate, meaning that multiple variables must be predicted, as it is the case in the *Paris Region AI Challenge 2020* (PRAIC) (Pavao et al., 2021). In this case, the performance can be measured using an average score (weighted or not) across the output variables, or using any ranking function. Another example of time series regression competition is the *AutoSeries Challenge* (Xu et al., 2021), which happens to be also an AutoML competition. This competition confirmed the efficiency of Gradient-Boosting Machines (GMB) to tackle time series regression tasks, as well as random search hyper-parameter tuning to tackle the AutoML part of the problem.

5.2 Time series forecasting

"*A Brief History of Time Series Forecasting Competitions*" by Hyndman (2023) traces the transformative impact of forecasting competitions from the *Makridakis Competitions* series, organized by Spyros Makridakis and spanning from 1980 to today (Makridakis et al., 1982; Makridakis and Hibon, 2000; Makridakis et al., 2018), highlighting their role in shaping forecasting methodologies across diverse data types. The paper emphasizes the consistent success of combination forecasts, encouraging the use of ensemble methods, and points out the balance needed between automated forecasting and domain-specific expertise. Other exemplary time series prediction competitions include the competitions organized at the Santa Fe Institute (Weigend and Gershenfeld, 1993) which contributed to our understanding of time series prediction in a variety of contexts.

Time series forecasting competitions can be designed in both interactive or non-interactive settings, depending on the objectives and constraints of the challenge. In a **non-interactive** format, participants are provided with a complete dataset up to a certain point in time, and they are required to make predictions for future data points. The models are then evaluated based on their accuracy in predicting these unseen data points. This format is straightforward but may not fully capture the dynamic nature of real-world time series forecasting, where new data continuously become available, and models need to be updated accordingly. On the other hand, **interactive competitions** aim to mimic these real-world conditions by releasing data in stages. Participants make predictions based on available data, and as the competition progresses, new data are released, which can be used to update and improve the models. This format encourages the development of adaptive models that can respond to changes in data patterns over time. This design was typically found in the *COVID-19 Global Forecasting*[1] challenge on Kaggle, where participants were tasked with predicting the spread of COVID-19 disease. Subsequently, the initially unknown ground truth was revealed and added to the training data on a weekly basis.

6 Reinforcement learning

Reinforcement Learning (RL) is a subset of machine learning where an agent learns to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties, guiding it to optimize its behavior to maximize cumulative rewards over time, as illustrated by Figure 5. RL has been successfully applied in various domains, including robotics, game playing, and autonomous vehicles. Organizers of such challenges must choose suitable problem simulations, balance environmental complexity with computational demands, and set objective evaluation criteria that consider efficiency, adaptability, and robustness of the agent's performance.

Designing challenges for RL is an inherently complex task. One of the primary difficulties is the requirement for a simulated or real-world environment where participants' algorithms can interact, learn, and be evaluated. Ensuring the stability, reliability, and realism of these environments is crucial, as inconsistencies or inaccuracies can lead to misleading results and prevent the learning process. Furthermore, RL algorithms typically require a substantial amount of interactions with the environment to learn effectively, making the computational cost a significant consideration. Additionally, there is no one-size-fits-all metric for assessing the performance of RL algorithms across various tasks and environments, necessitating the careful selection and design of evaluation criteria that accurately reflect the objectives of the specific competition.

RL challenges can be designed following different protocols, primarily distinguished by the availability of pre-collected data. In challenges **without** pre-collected data, the algorithms proposed by participants engage directly with the environment, allowing data acquisition and learning concurrently. On the other hand, challenges **with** pre-collected data enable participants to refine and train their algorithms in an offline manner. The setting without pre-collected data is often referred to as **online learning**, and typically occurs in *OpenAI Gym Competitions* such as the *Retro Contest* (Nichol et al., 2018) where agents interacts with video games environment without prior knowledge. This approached is opposed to **offline learning**, which uses pre-collected data, such as the Atari Grand Challenge dataset (Bellemare et al., 2013). This particular dataset comprises a col-

[1] `https://www.kaggle.com/c/covid19-global-forecasting-week-1`

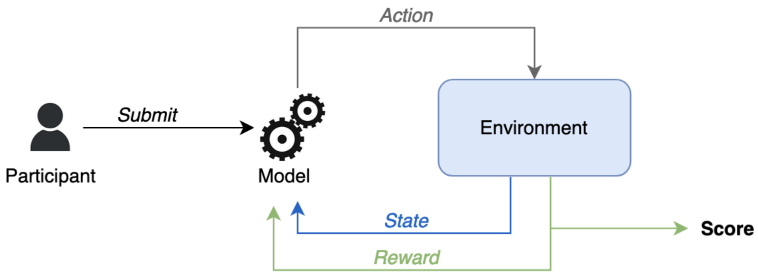


Figure 5: Reinforcement learning competition workflow. The submitted agent interacts with the environment, with the objective of maximizing the reward over time.

lection of human demonstrations across various of Atari games. In offline learning scenarios, the algorithm learns exclusively from this existing data, without the opportunity for real-time interaction or data acquisition, as seen in online learning settings. Regardless of whether algorithms are trained through online or offline learning protocols, their performance must ultimately be evaluated by interacting to live environments.

It is common to use a cumulative reward over time as a primary metric for determining the final score of participants' models. In such settings, the manner in which time is quantified plays a crucial role in the evaluation process, introducing a potential challenge in ensuring equitable and unbiased benchmarking. To mitigate inconsistencies that may arise from hardware disparities, it is advisable to standardize the measurement of total time in terms of the **number of environmental steps** taken, rather than relying on real-time duration measured in seconds. Adopting this approach ensures a consistent and equitable evaluation framework, as it remains invariant across different hardware configurations, thereby enhancing the fairness and reliability of the competition results.

Moreover, in RL challenges, having well-defined and expert-crafted metrics is crucial for evaluating the performance of participating algorithms accurately and fairly. These metrics need to capture not just the immediate rewards but also the long-term impact of decisions made by the RL agents. The design of these metrics requires a deep understanding of the specific domain, the goals of the RL task, and the potential trade-offs between different objectives.

Another interesting distinctions in RL competition design is **interacting with the environment** versus **interacting with other agents**. In the "interaction with environment" setting, the agents submitted by participants are evaluated by interact with the given environment. The related scenes includes single-player games, auto-driving, robot controlling, etc. In the "interaction between agents" setting, the agents are ranked by the performance of competition with other agents. The related scenes includes for instances multi-player games and stock market.

Figure 6 shows the process about the interacting with environment setting and the Figure 7 shows the process about the interacting with other agents setting reinforcement learning competition. The most important issue in the design of reinforcement learning competition is how to evaluate the ranking of submissions. In the setting of interacting with environment, the performance of submissions can be measured by the cumulative reward the submission gained by interacting with the environment. So it is a very important challenge for competition organizers to construct a good simulating environment. The quality of environment determines the quality of the whole competition. In the setting of interacting with other agents, the most popular way is to model the ranking

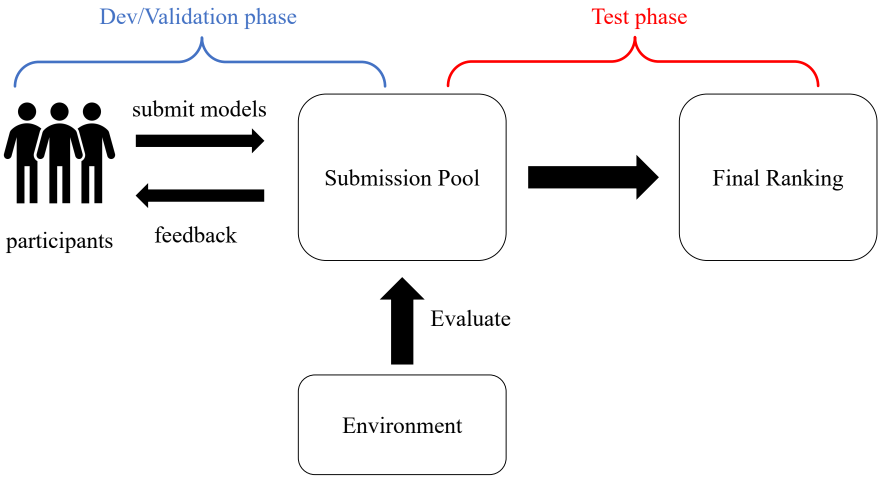


Figure 6: Process of reinforcement learning competitions of interacting with environment.

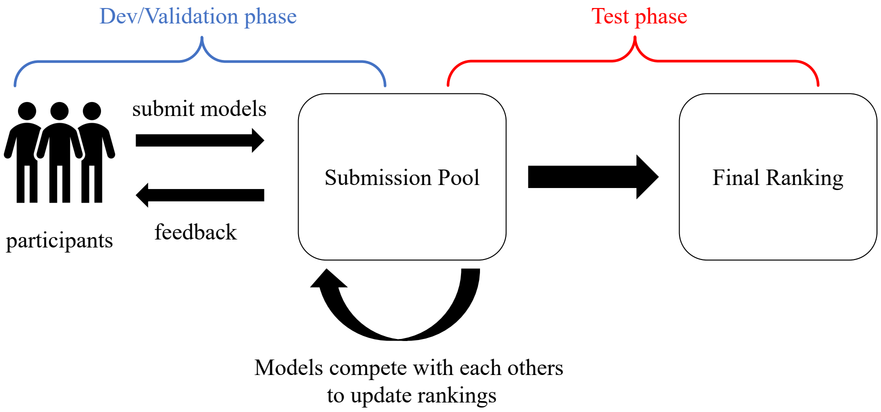


Figure 7: Process of reinforcement learning competitions of interacting with other agents.

score by a Gaussian distribution $N(\mu,\sigma^2)$. Submissions with similar skill rating would be picked to finish a match. The winner's μ will be increased while the loser's μ will be decreased. If there is a draw, the μ of two teams will be moved closer to their mean. Here the key issue is how pick submissions with similar ranking score.

Instances of RL competitions include those in the domain of biomechanics. Notable examples in this category are the *Learning to Run* challenge (Kidzinski et al., 2018) and the *AI for Prosthetics* competition (Łukasz Kidziński et al., 2018). These events enabled progress in simulating and understanding complex biomechanical processes. In addition to biomechanics, RL competitions also extend to video game environments, offering unique challenges that require agents to navigate and interact within virtual worlds. The *MineRL Competition* (Guss et al., 2019) and the *Procgen Competition* (Mohanty et al., 2021) typically test the ability of RL algorithms to adapt and perform across procedurally generated environments. Furthermore, competitions such as *Metalearning from Learning Curves* (Nguyen et al., 2022) explore general aspects of machine learning. This challenge study the meta-analysis of learning processes, encouraging the development of algorithms that can learn effectively from existing learning trajectories.

Two notably interesting examples of past reinforcement learning competitions are the *NetHack 2021 NeurIPS Challenge* and the *Google Research Football with Manchester City F.C*.

Held by Meta and DeepMind, the **NetHack 2021 NeurIPS Challenge** required participants to design an agent to play the game NetHack automatically. NetHack is a single-player video game in which the player is required to navigate the procedurally generated, ascii dungeons to find the amulet. Although it is a very complex game, it can be simulated efficiently by the NetHack Learning Environment (NLE) which is presented at NeurIPS 2020. This competition was split into a development and test phase. During the development phase, participants were able to submit their agents to the leaderboard once a day and 512 evaluation runs would be performed to calculate a preliminary place on the dev-phase leaderboard. The top 15 participants for each track were taken from the dev-phase leaderboard and invited to join the test phase. In test phase, participants were able to submit their best agents 3 to the test-phase leaderboard and 4096 evaluation runs would be performed to calculate the final ranking Net (2020). 42 teams joined this competition and submitted 632 submissions to compete a total prize of 20,000 dollars.

The **Google Research Football with Manchester City F.C** competition was held on Kaggle in 2019 by Google Research and Manchester City F.C. foo (2020). In this competition, each team was required to create AI agents to control a 11-player football team, and them compete with other teams in the simulation environment Kurach et al. (2020). To simplify this challenge, at each time step, the team only need to control one player by choosing an action from a given set of 19 actions. Each submission had an estimated skill rating modeled by a Gaussian distribution $N(\mu,\sigma^2)$, and the rating was updated by the procedure mentioned above. 1,138 teams participated this competitions to compete a total prize of 6,000 dollars.

There are some tips for participants who wants to get good performance in reinforcement learning competitions. The first tip is to focus on the design of reward function, especially in some scenes the reward function is very sparse. The second tip is to design the feature processing model and reinforcement model structure carefully, because they are the key issues to speed up the training progress and improve the model performance. The third tip is to combine with some typical reinforcement learning algorithms such as MCTS and on-policy algorithms.

In conclusion, designing challenges for RL competitions is a nuanced task that requires careful consideration of the learning environment, computational resources, and evaluation metrics.

7 Adversarial learning

Recent research shows that many machine learning classifiers, especially deep learning models, are highly vulnerable to adversarial examples Biggio et al. (2013); Szegedy et al. (2014). An adversarial example is a sample of input data which has been slightly modified to mislead the classifiers while human observers can not notice the modification at all. The existence of adversarial samples raises a huge challenge to the security of machine learning and AI systems. Adversarial learning competition is an important way to examine the adversarial attack and defense algorithms, thus plays an important role in adversarial learning researches.

The adversarial learning competitions includes two aspects: attack and defense. In the attack competition, participants are required to attack a given model. There are three types of attack settings categorized by the revealed information of victim model.

- **White-box attack setting**. In this setting, participants have the full information about the victim model, including the model structure, the value of parameters and hyper-parameters.

- **Black-box attack setting**. In this setting, participants can not directly know the details about the victim model. Instead, participants can query the victim model certain inputs and observe the prediction results.

- **Universal attack setting**. In this setting, participants can not know anything about the victim model or query the victim model. It requires to construct the universal adversarial samples which can mislead most machine learning classifiers.

The attack setting can also be divided into targeted attack and non-targeted attack. In non-targeted attack the adversary only need to cheat the victim model to give a wrong prediction, while in the target attack the adversary is required to mislead the victim model to output a special given label as prediction.

In the defense competition, the participants are required to submit classifiers trained on the given dataset. Then the accuracy of submissions are measured on the adversarial samples constructed by certain adversarial attack algorithm.

Similar with the reinforcement learning competition design, one of the most important issue in adversarial learning competition design is how to evaluate and rank the performance of submissions. For the evasion attack, there are two dimensions in the measure about the attack performance: disturb norm and the attack success rate. For simplicity, we only discuss the case of non-targeted attack, the measure of targeted attack can be designed with similar method. The performance of submissions can be evaluated by the attack success rate with the restricted disturb norm. For example, the score of submitted attack model on test sample x can be represented by

$$\text{score}(g,x) = \begin{cases} 1, \text{ if} ||x - g(x)|| \leq \varepsilon \,\text{and} f(x) \neq f(g(x)); \\ 0, \text{ otherwise} \end{cases} . \tag{1}$$

Here g represents the submission, $g(x)$ represents the adversarial sample produced by the submission model, f represents the victim model, and the ε represents the threshold of disturb norm. The performance of submissions can also be evaluated by the disturb norm of adversarial samples which attack the victim model successfully. The loss can be designed as follows:

$$\ell(g,x) = \begin{cases} ||x - g(x)||, \text{ if} f(x) \neq f(g(x)); \\ A, \text{ if} f(x) = f(g(x)) \end{cases}, \tag{2}$$

in which A represents the penalty for adversarial samples failed to mislead the victim model. A must be larger than all the possible disturb norms, i.e. $A \geq \max_{x,x'} ||x - x'||$.

In the evaluation of defense model, the influence of disturb norm should be considered, too. Similar with the evasion attack case, there are two ways to measure the performance of defense model. The first way is to test the defense models with adversarial samples with restricted disturbed norm, then compare the prediction accuracy of defense models on adversarial samples. The second way is to evaluate by the disturb norm of adversarial samples the submissions can distinguish successfully. For example, the score function of the second evaluation method can be designed as follows:

$$\text{score}(f,x') = \begin{cases} 0, \text{if} f(x') \neq y; \\ ||x - x'||, \text{if} f(x') = y \end{cases}, \tag{3}$$

in which (x,y) represents the initial samples and the ground-truth label, and the x' represents the adversarial sample on x.

Interestingly, adversarial learning competitions can also be organized as interactive benchmarks, where particiants' models can attack and defend against each other. Two designs are possible: the **sequential design** where the competition unfolds in distinct stages or phases, and the **simultaneous design** where challenges run both phases concurrently. Examples of sequential adversarial challenges include the ASVSpoof Challenge (Yamagishi et al., 2021; Liu et al., 2023) and the Data Anonymization and Re-identification Challenge (DARC) (Boutet et al., 2020). Examples of sequential adversarial challenges include the Hide-and-Seek Privacy Challenge (Jordon et al., 2020) and the Privacy Workshop Cup (Murakami et al., 2023).

One of the most famous adversarial learning competition is NeurIPS 2017 Adversarial Attacks and Defences Competition organized by Google Brain. This competition is consisted with 3 tracks: 1) non-targeted black-box attack; 2) targeted black-box attack; 3) defense against adversarial attacks. In each track, the participants submitted their models, then the submitted model was given a set of images (and target classes in case of targeted attack) as an input, and had to produce either an adversarial image (for attack submission) or classification label (for defense submission) for each input image Kurakin et al. (2018). The performance of attack models were measured by the average accuracy of victim models, and the performance of defense models were measured by their average accuracy against attack models. 91 teams participated the track 1), 65 teams participated the track 2), and 107 teams participated the track 3).

In IJCAI 2019, Alibaba Group organized an adversarial learning competition including 3 tracks: targeted attack track, non-targeted attack track and the defense track ijc (2019). In this competition, 110,000 pictures of goods from 110 commodity categories are published as training and test sets. In the attack tracks, the submissions were required to attack 5 defense models, then the average disturb norms on these 5 models were used to evaluate the performance of submissions. In the defense track, the submissions were also tested by 5 different attack models, then the average disturb norms of adversarial samples were disputed as the score of submissions. 2519 teams participated this competition to compete a total prize of 39,000 dollars. The teams from USTC won the championship of defense track, the teams from Southeast University won the championship of target attack track, the teams from Guangzhou University won the championship of non-target attack track.

In KDD 2020, biendata and zhipu.AI organized a competition about the adversarial learning on graph data bie (2020). In particular, this competition is focus on the evasion attack and defense on the citation network de Solla Price (1965). The citation network is a kind of academic graph where academic papers are the nodes and citations are the directed edge. This graph is an important tool which can help researchers to analyse the cite relation of each paper and evaluate the impact of papers. Preventing the attack against the citation network (for example, manipulating citations Chawla (2019)) This competition included 2 phases, in which 543,486 nodes were training set and 50,000 nodes were test set. In the first phase, organizers provided a graph with 593,486 nodes and 100 features on each node. The participants were required to submit a black-box attack model to mislead the organizer's classifier by adding no more than 500 nodes. The performance of attack model was evaluated by the decrease on accuracy of organizer's classifier. In the second phase, each team submitted an attack model and a defense model trained on a similar but different dataset with the one of first phase. Then, the organizer matched all attack models and all defense models. The score of defense model was disputed by the average accuracy on each match, and the score of attack model was disputed by the average error rate on each match. The final score of each team was the average score of its attack model and defense model. 608 partcipants from 511 teams joined this competition to compete a total prize of 20,000 dollars.

Here we provide some tips for participants who wants to get good performance in adversarial learning competitions. For the evasion attack competitions, it is a good idea to combine the attack strategies with the domain knowledge. It is also important to have a well-designed adversarial loss function. For the defense competitions, there are two aspects in which the defense model can be improved. In the feature aspect, feature processing technology such as feature denoising Xie et al. (2019) and feature transformation Song et al. (2020) can help to improve the adversarial robustness of submissions. Some novel models such as topology adaptive model Du et al. (2017) can also be used to construct the adversarial robust model.

8 Use of confidential data

Confidential data may include sensitive information such as personal data, financial data, or trade secrets, and it is important to ensure that this data is handled in a secure and ethical manner. There are several concerns associated with using confidential data in machine learning competitions, including the need to protect the data from unauthorized access, and the need to comply with relevant laws and regulations. Confidential data holds a great importance in many applications, both in scientific and industrial contexts. Some examples include finance, healthcare, and human resources. Using confidential data in crowd-sourced benchmarks is a challenge in itself, but can be highly beneficial by enabling innovation in critical fields.

In this section, we present two different protocols for handling confidential data: **replacing the data by synthetic data**, and **running the participants' models blindly on the real data**. These two protocols, with their advantages and drawbacks, make it possible to crowd-source research on private data without compromising confidentiality.

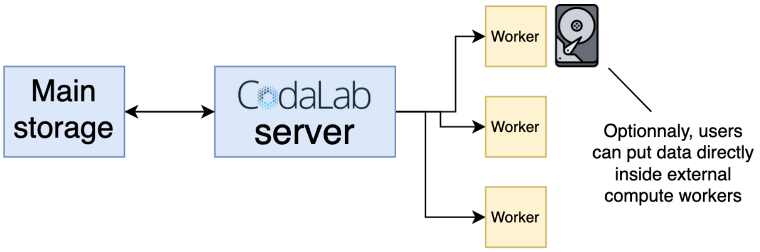

Main storage
CodaLab server
Worker
Worker
Worker
Optionnaly, users can put data directly inside external compute workers

Figure 8: Confidential data can be put directly inside organizers' compute workers, externally from the main servers of the platform.

8.1 Synthetic data

In order to propose a task based on confidential data to the participants without exposing the private data, one approach is to train a generative model to replicate the dataset. Synthetic data can then be generated from the model, and used to simulate the task without disclosing the actual dataset. This approach raises two antagonistic issues: in one hand, the synthetic data must resemble the original data to ensure the problem remains relevant and connected to the real world; on the other hand, the generative model must not leak any real data points. We have developed metrics to evaluate generators *utility* and *privacy* (Yale et al., 2019, 2020) (presented in the chapter 4).

The limitation of using synthetic data is the potential trade-off between *privacy* and *utility*. The *utility* of artificial data can be evaluated by deploying it in real-world scenarios and verifying that model outcomes are consistent with those achieved using real data.

We applied this concept in "To be or not to be", referenced in Pavao et al. (2019), a challenge designed to instruct health students. The task is to predict the survival or decease of patients in intensive healthcare units, based on tabular medical records. The source of the data is the MIMIC-III dataset, which consists in both numerical and categorical variables describing thousands of patients, such as age and blood pressure. Given the inherent confidential and sensitive nature of this data, it is subject to access restrictions. We generated a synthetic dataset using a Wasserstein Generative Adversarial Network (WGAN) (Goodfellow et al., 2014; Arjovsky et al., 2017) model. The resultant challenge continues to be used in Rensselaer Polytechnique Institute to train health students[2].

8.2 Blind access to the data

The second approach for utilizing private data is to blindly execute participants' models on the real data. Two mechanisms are in play to benchmark the participants' solutions despite the private nature of the data: **code submission**, and **storing the data inside the compute workers**, as exposed in Figure 8. This way, only the uploaded models can read the data, it remains completely hidden from the participants. We implemented this feature to *CodaLab Competitions*. This is particularly interesting since, as shown in the chapter 11, organizers can link their own machines to the platform as external compute workers, ensuring a complete control over the data security. We employed this approach in the Paris Region AI Challenge 2020 (Pavao et al., 2021)

[2] `https://codalab.lisn.upsaclay.fr/competitions/3073`

Protocol	**Data**	**Multiple tasks**	**Code submission**	**Interactive design**
Supervised learning	✓			
AutoML	✓	✓	✓	
Metalearning	✓	✓	✓	
Time series	✓			*depends*
Reinforcement Learning	*depends*		✓	✓
Confidential data	✓		*depends*	
Adversarial challenges	*depends*		*depends*	✓

Table 1: Characterization of the challenge protocols presented in the chapter, indicating the specific criteria that are mandatory (✓), and highlighting those that are possible depending on the design of the challenge (*depends*).

A sample of artificial data, as well as documentation and baseline methods, should be provided to help the participants building their methods despite the constraints associated with not being able to access the dataset directly. Having extensive output logs can also helps the participants to navigate through the problem despite of the blind testing. However, it is advised to limit the size of the output logs to avoid the leakage of the sensitive data. The security of external workers is ensured because the computer workers are owned by the organizers, and *CodaLab Competitions* platform cannot read them. The main limitation of this approach is that it is harder for the participants to work without direct access to the data, making it more difficult to reach the same performance level.

9 Conclusion

In this chapter we analysed the features and special designs of various type of machine learning competitions (adversarial learning, automated machine learning, etc.). We believe that the analysis in this chapter can help both organizers and participants, and also offers reference and inspiration about competitions of novel machine learning paradigms in the future.

In this chapter, we focused on examining the design specificity inherent in competitions and benchmarks in machine learning. We illustrated various experimental designs: supervised learning, AutoML and metalearning, time series analysis, reinforcement learning, the use of confidential data and adversarial challenges. The main characteristics and differences between these designs are outlined in Table 1. A common thread of most of these protocols is the necessity for participants to submit their model's code to the platform for evaluation. This resonates with the recommendation to use code submissions, both allowing complex evaluation procedures and improving the reproducibility and the validity of the evaluation. Interactive designs are at play in reinforcement learning, where algorithms interact with a dynamic environment; in adversarial challenges, where competing algorithms engage with one another; and occasionally in time series prediction tasks, where datasets are regularly augmented with new observations, allowing previously used testing data to become part of the training set for future iterations.

It is also interesting to note that artificial data holds potential utility in certain challenge designs. For tasks where the ground truth is almost exclusively artificial data, or when emulating real data that is confidential, synthetic datasets are beneficial. The latter can also be addressed with real data, by employing blind-testing methods, ensuring participants cannot access confidential datasets. More generally, synthesizing artificial datasets can be beneficial for tasks lacking a ground truth, such as in unsupervised learning, since the synthesis rules can be precisely known by the organizers. Indeed, synthetic data in machine learning competitions can offer a controlled environment to evaluate algorithms, with the advantage of generating diverse and challenging scenarios that resemble complex real-world data distributions. Moreover, using artificial data, organizers can control the task difficulty, generate large datasets, address data imbalance, and reduce data collection cost. Additionally, in scenarios like reinforcement learning, where agents must learn from interaction within an environment, synthetic data provides an endless landscape of tasks for testing the robustness and adaptability of algorithms, as seen in competitions such as the *AI Driving Olympics* (Zilly et al., 2019). The main drawback of this approach is the potential *reality gap* between artificial and real data.

Adversarial learning, focusing on attack and defense algorithms, allows to explore the boundaries of the strengths and weaknesses of existing models. Adversarial learning competition can either focus on attack, on defense, or on both using an interactive design.

Given the diverse and rapidly evolving nature of the field of machine learning, a comprehensive enumeration of all possible design features and evaluation criteria is impossible. For instance, competitions centered on one-shot learning might evaluate the ability of models to generalize from minimal data, while those focusing on fairness could prioritize unbiased predictions across diverse demographic groups. In the domain of real-time processing, the emphasis might shift to algorithmic speed and responsiveness. Tasks involving multi-modal learning demand the integration of information from varied data sources like text, images, and audio. Meanwhile, resource-constrained competitions challenge participants to optimize the model performance with tight computational or memory budgets. However, the methodologies and approaches outlined here can serve as references for future competitions, particularly those in emerging paradigms such as automated machine learning or adversarial challenges.

References

Ijcai-19 alibaba adversarial ai challenge, 2019. https://tianchi.aliyun.com/markets/tianchi/ijcai19_en, Last accessed on 2022-07-15.

Nethack 2020 neurips competition, 2020. https://www.aicrowd.com/challenges/neurips-2021-the-nethack-challenge#challenge-motivation/, Last accessed on 2022-07-15.

Kdd cup 2020: Graph adversarial attack and defense, 2020. https://www.biendata.xyz/competition/kddcup_2020/, Last accessed on 2022-07-15.

Google research football with manchester city f.c., 2020. https://www.kaggle.com/c/google-football, Last accessed on 2022-07-15.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

Adrian El Baz, Isabelle Guyon, Zhengying Liu, Jan N. van Rijn, Sébastien Treguer, and Joaquin Vanschoren. Advances in metadl: AAAI 2021 challenge and workshop. In Isabelle Guyon, Jan N. van Rijn, Sébastien Treguer, and Joaquin Vanschoren, editors, *AAAI Workshop on Meta-Learning and MetaDL Challenge, MetaDL@AAAI 2021, virtual, February 9, 2021*, volume 140 of *Proceedings of Machine Learning Research*, pages 1–16. PMLR, 2021. URL https://proceedings.mlr.press/v140/el-baz21a.html.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 387–402, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-40994-3.

Antoine Boutet, Mathieu Cunche, Sébastien Gambs, Benjamin Nguyen, and Antoine Laurent. DARC : Data Anonymization and Re-identification Challenge. In *RESSI 2020 - Rendez-vous de la Recherche et de l'Enseignement de la Sécurité des Systèmes d'Information*, Nouan-le-Fuzelier, France, December 2020. URL `https://inria.hal.science/hal-02512677`.

Pavel Brazdil, Jan N. van Rijn, Carlos Soares, and Joaquin Vanschoren. Metalearning: Applications to automated machine learning and data mining. 2022.

Dalmeet Singh Chawla. Elsevier investigates hundreds of peer reviewers for manipulating citations. *Nature*, 573(7773):174, 2019.

Derek J. de Solla Price. Networks of scientific papers. *Science*, 149(3683):510–515, 1965. doi: 10.1126/science.149.3683.510. URL `https://www.science.org/doi/abs/10.1126/science.149.3683.510`.

Jian Du, Shanghang Zhang, Guanhang Wu, José M. F. Moura, and Soummya Kar. Topology adaptive graph convolutional networks. *CoRR*, abs/1710.10370, 2017. URL `http://arxiv.org/abs/1710.10370`.

Adrian El Baz, Ihsan Ullah, Edesio Alcobaça, André C. P. L. F. Carvalho, Hong Chen, Fabio Ferreira, Henry Gouk, Chaoyu Guan, Isabelle Guyon, Timothy Hospedales, Shell Hu, Mike Huisman, Frank Hutter, Zhengying Liu, Felix Mohr, Ekrem Öztürk, Jan N van Rijn, Haozhe Sun, Xin Wang, and Wenwu Zhu. Lessons learned from the NeurIPS 2021 MetaDL challenge: Backbone fine-tuning without episodic meta-learning dominates for few-shot learning image classification. In *NeurIPS 2021 Competition and Demonstration Track*, On-line, United States, December 2021. URL `https://hal.science/hal-03688638`.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

William H. Guss, Cayden Codel, Katja Hofmann, Brandon Houghton, Noburu Kuno, Stephanie Milani, Sharada P. Mohanty, Diego Perez Liebana, Ruslan Salakhutdinov, Nicholay Topin, Manuela Veloso, and Phillip Wang. The minerl competition on sample efficient reinforcement learning using human priors. *CoRR*, abs/1904.10079, 2019. URL `http://arxiv.org/abs/1904.10079`.

Isabelle Guyon, Lisheng Sun-Hosoya, Marc Boullé, Hugo Jair Escalante, Sergio Escalera, Zhengying Liu, Damir Jajetic, Bisakha Ray, Mehreen Saeed, Michèle Sebag, Alexander R. Statnikov, Wei-Wei Tu, and Evelyne Viegas. Analysis of the automl challenge series 2015-2018. In Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren, editors, *Automated Machine Learning - Methods, Systems, Challenges*, The Springer Series on Challenges in Machine Learning, pages 177–219. Springer, 2019. doi: 10.1007/978-3-030-05318-5_10. URL https://doi.org/10.1007/978-3-030-05318-5_10.

Rob J Hyndman. Forecasting competitions, 2023. URL https://robjhyndman.com/hyndsight/forecasting-competitions/.

James Jordon, Daniel Jarrett, Evgeny Saveliev, Jinsung Yoon, Paul W. G. Elbers, Patrick Thoral, Ari Ercole, Cheng Zhang, Danielle Belgrave, and Mihaela van der Schaar. Hide-and-seek privacy challenge: Synthetic data generation vs. patient re-identification. In Hugo Jair Escalante and Katja Hofmann, editors, *NeurIPS 2020 Competition and Demonstration Track, 6-12 December 2020, Virtual Event / Vancouver, BC, Canada*, volume 133 of *Proceedings of Machine Learning Research*, pages 206–215. PMLR, 2020. URL http://proceedings.mlr.press/v133/jordon21a.html.

Lukasz Kidzinski, Sharada Prasanna Mohanty, Carmichael F. Ong, Zhewei Huang, Shuchang Zhou, Anton Pechenko, Adam Stelmaszczyk, Piotr Jarosik, Mikhail Pavlov, Sergey Kolesnikov, Sergey M. Plis, Zhibo Chen, Zhizheng Zhang, Jiale Chen, Jun Shi, Zhuobin Zheng, Chun Yuan, Zhihui Lin, Henryk Michalewski, Piotr Milos, Blazej Osinski, Andrew Melnik, Malte Schilling, Helge J. Ritter, Sean F. Carroll, Jennifer L. Hicks, Sergey Levine, Marcel Salathé, and Scott L. Delp. Learning to run challenge solutions: Adapting reinforcement learning methods for neuromusculoskeletal environments. *CoRR*, abs/1804.00361, 2018. URL http://arxiv.org/abs/1804.00361.

Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. Google research football: A novel reinforcement learning environment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):4501–4510, Apr. 2020. doi: 10.1609/aaai.v34i04.5878. URL https://ojs.aaai.org/index.php/AAAI/article/view/5878.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, et al. Adversarial attacks and defences competition. In *The NIPS'17 Competition: Building Intelligent Systems*, pages 195–231. Springer, 2018.

Xuechen Liu, Xin Wang, Md. Sahidullah, Jose Patino, Héctor Delgado, Tomi Kinnunen, Massimiliano Todisco, Junichi Yamagishi, Nicholas W. D. Evans, Andreas Nautsch, and Kong Aik Lee. Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild. *IEEE ACM Trans. Audio Speech Lang. Process.*, 31:2507–2522, 2023. doi: 10.1109/TASLP.2023.3285283. URL https://doi.org/10.1109/TASLP.2023.3285283.

Zhengying Liu, Adrien Pavao, Zhen Xu, Sergio Escalera, Fabio Ferreira, Isabelle Guyon, Sirui Hong, Frank Hutter, Rongrong Ji, Julio C. S. Jacques Junior, Ge Li, Marius Lindauer, Zhipeng Luo, Meysam Madadi, Thomas Nierhoff, Kangning Niu, Chunguang Pan, Danny Stoll, Sébastien Treguer, Jin Wang, Peng Wang, Chenglin Wu, Youcheng Xiong, Arber Zela, and Yang Zhang. Winning solutions and post-challenge analyses of the chalearn autodl challenge 2019. *IEEE*

Trans. Pattern Anal. Mach. Intell., 43(9):3108–3125, 2021. doi: 10.1109/TPAMI.2021.3075372. URL https://doi.org/10.1109/TPAMI.2021.3075372.

Spyros Makridakis and Michèle Hibon. The m3-competition: results, conclusions and implications. *International Journal of Forecasting*, 16(4):451–476, Oct 2000. doi: 10.1016/S0169-2070(00)00057-1.

Spyros Makridakis, A Andersen, R Carbone, R Fildes, Michèle Hibon, R Lewandowski, J Newton, E Parzen, and R Winkler. The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting*, 1(2):111–153, Apr 1982. doi: 10.1002/for.3980010202.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: Results, findings, conclusion and way forward. *International Journal of Forecasting*, 34(4):802–808, 2018.

Sharada P. Mohanty, Jyotish Poonganam, Adrien Gaidon, Andrey Kolobov, Blake Wulfe, Dipam Chakraborty, Grazvydas Semetulskis, João Schapke, Jonas Kubilius, Jurgis Pasukonis, Linas Klimas, Matthew J. Hausknecht, Patrick MacAlpine, Quang Nhat Tran, Thomas Tumiel, Xiaocheng Tang, Xinwei Chen, Christopher Hesse, Jacob Hilton, William Hebgen Guss, Sahika Genc, John Schulman, and Karl Cobbe. Measuring sample efficiency and generalization in reinforcement learning benchmarks: Neurips 2020 procgen benchmark. *CoRR*, abs/2103.15332, 2021. URL https://arxiv.org/abs/2103.15332.

Takao Murakami, Hiromi Arai, Koki Hamada, Takuma Hatano, Makoto Iguchi, Hiroaki Kikuchi, Atsushi Kuromasa, Hiroshi Nakagawa, Yuichi Nakamura, Kenshiro Nishiyama, Ryo Nojima, Hidenobu Oguri, Chiemi Watanabe, Akira Yamada, Takayasu Yamaguchi, and Yuji Yamaoka. Designing a location trace anonymization contest. *Proc. Priv. Enhancing Technol.*, 2023(1):225–243, 2023. doi: 10.56553/popets-2023-0014. URL https://doi.org/10.56553/popets-2023-0014.

Manh Hung Nguyen, Lisheng Sun, Nathan Grinsztajn, and Isabelle Guyon. Meta-learning from Learning Curves Challenge: Lessons learned from the First Round and Design of the Second Round. working paper or preprint, August 2022. URL https://hal.science/hal-03725313.

Alex Nichol, Vicki Pfau, Christopher Hesse, Oleg Klimov, and John Schulman. Gotta learn fast: A new benchmark for generalization in RL. *CoRR*, abs/1804.03720, 2018. URL http://arxiv.org/abs/1804.03720.

A. Pavao, D. Kalainathan, L. Sun-Hosoya, K. Bennett, and I. Guyon. Design and Analysis of Experiments: A Challenge Approach in Teaching. *NeurIPS*, December 2019. URL http://ciml.chalearn.org/ciml2019/accepted/Pavao.pdf?attredirects=0&d=1.

Adrien Pavao et al. Airplane numerical twin: A time series regression competition. *International Conference on Machine Learning and Applications (ICMLA)*, 2021.

Nicholas Roberts, Samuel Guo, Cong Xu, Ameet Talwalkar, David Lander, Lvfang Tao, Linhang Cai, Shuaicheng Niu, Jianyu Heng, Hongyang Qin, Minwen Deng, Johannes Hog, Alexander Pfefferle, Sushil Ammanaghatta Shivakumar, Arjun Krishnakumar, Yubo Wang, Rhea Sukthanker, Frank Hutter, Euxhen Hasanaj, Tien-Dung Le, Mikhail Khodak, Yuriy Nevmyvaka, Kashif Rasul, Frederic Sala, Anderson Schneider, Junhong Shen, and Evan Sparks. Automl decathlon: Diverse tasks, modern methods, and efficiency at scale. In Marco Ciccone, Gustavo Stolovitzky, and Jacob Albrecht, editors, *Proceedings of the NeurIPS 2022 Competitions Track*, volume 220 of *Proceedings of Machine Learning Research*, pages 151–170. PMLR, 28 Nov–09 Dec 2022. URL `https://proceedings.mlr.press/v220/roberts22a.html`.

Nicholas Roberts, Spencer Schoenberg, Tzu-Heng Huang, Dyah Adila, Changho Shin, Jeffrey Li, Sonia Cromp, Cong Xu, Samuel Guo, Adrien Pavao, Ameet Talwalkar, and Frederic Sala. Toward data-centric automl. In *Competition, Poster*. AutoML Conference 2023, 2023.

Chuanbiao Song, Kun He, Jiadong Lin, Liwei Wang, and John E. Hopcroft. Robust local features for improving the generalization of adversarial training. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=H1lZJpVFvr`.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Renbo Tu, Nicholas Roberts, Mikhail Khodak, Junhong Shen, Frederic Sala, and Ameet Talwalkar. Nas-bench-360: Benchmarking neural architecture search on diverse tasks, 2023.

A.S. Weigend and N.A. Gershenfeld. Results of the time series prediction competition at the santa fe institute. In *IEEE International Conference on Neural Networks*, pages 1786–1793 vol.3, 1993. doi: 10.1109/ICNN.1993.298828.

Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Zhen Xu, Wei-Wei Tu, and Isabelle Guyon. Automl meets time series regression design and analysis of the autoseries challenge. *CoRR*, abs/2107.13186, 2021.

Andrew Yale, Saloni Dash, Ritik Dutta, Isabelle Guyon, Adrien Pavao, and Kristin P. Bennett. Privacy preserving synthetic health data. *European Symposium on Artificial Neural Networks (ESANN)*, 2019.

Andrew Yale, Saloni Dash, Ritik Dutta, Isabelle Guyon, Adrien Pavao, and Kristin P. Bennett. Generation and evaluation of privacy preserving synthetic health data. *Neurocomputing*, 416: 244–255, 2020. doi: 10.1016/j.neucom.2019.12.136. URL `https://doi.org/10.1016/j.neucom.2019.12.136`.

Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md. Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas W. D. Evans, and Héctor Delgado. Asvspoof 2021: accelerating progress in spoofed and deepfake speech detection. *CoRR*, abs/2109.00537, 2021. URL `https://arxiv.org/abs/2109.00537`.

Julian G. Zilly, Jacopo Tani, Breandan Considine, Bhairav Mehta, Andrea F. Daniele, Manfred Diaz, Gianmarco Bernasconi, Claudio Ruch, Jan Hakenberg, Florian Golemo, A. Kirsten Bowser, Matthew R. Walter, Ruslan Hristov, Sunil Mallya, Emilio Frazzoli, Andrea Censi, and Liam Paull. The AI driving olympics at neurips 2018. *CoRR*, abs/1903.02503, 2019. URL `http://arxiv.org/abs/1903.02503`.

Łukasz Kidziński et al. Ai for prosthetics challenge, 2018. URL `https://www.crowdai.org/challenges/neurips-2018-ai-for-prosthetics-challenge`.