Stochastic Application Domain Definition for Functional Trustworthiness Certification of AI Systems

Simon Schmid[1,3] Barbara Brune[2] Alexander Aufreiter[1] Lukas Gruber[3]
Kajetan Schweighofer[3] Xaver Stadlbauer[2] Thomas Doms[2]
Bernhard Nessler[1]

[1]Software Competence Center Hagenberg [2]TÜV Austria Data Intelligence GmbH
[3]Johannes Kepler Universität Linz

Abstract

As Artificial Intelligence (AI) systems are increasingly deployed in safety-critical and societally consequential contexts, the question of how to evaluate their performance in a trustworthy and interpretable manner becomes increasingly important. Within the European Union, this issue is reflected in the AI Act, which requires training, validation, and testing datasets to be relevant and sufficiently representative with respect to the system's intended purpose. This raises a fundamental technical question: representative of what population of situations?

From a statistical perspective, performance metrics such as error rates or expected losses are always defined with respect to a probability distribution. We refer to this distribution as the Application Domain (AD). In practice, however, the AD of real-world AI systems is rarely known in explicit mathematical form and must instead be characterized operationally through the procedures by which valid samples are generated or selected.

To address this problem, we introduce the Stochastic Application Domain Definition (SADD), a textual specification of the sampling protocol that induces the distribution under which performance claims are intended to hold. The SADD links intended use, dataset construction, and statistical testing by making the underlying data-generation assumptions explicit. We formalize the notion of protocol-induced distributions, discuss how SADDs guide feasible sampling procedures, contrast the approach with qualitative domain descriptions such as Operational Design Domains, and examine implications for the certification of AI systems.

1 Introduction

Artificial Intelligence (AI) has become an integral part of many technical and societal domains. Progress in machine learning (ML) and deep learning has been driven by large-scale datasets, increased computational power, and advances in model architectures, leading to major scientific and industrial breakthroughs in recent years (27; 7; 14; 24). At the same time, the deployment of ML systems in real-world settings introduces risks that differ from those of traditional software. ML systems are increasingly used in contexts affecting health, safety, economic participation, and fundamental rights, making the question of their trustworthiness a central concern in research, regulation, standardization, and certification (4; 22; 23; 29; 16; 8; 13).

Within the European Union, these concerns are reflected in the AI Act, which introduces obligations regarding the governance and quality of training, validation, and testing data. In particular, Article 10

The Third Austrian Symposium on AI, Robotics, and Vision (AIROV26).

requires datasets to be relevant and sufficiently representative with respect to the intended purpose of the system (1). This requirement raises a fundamental technical question: representative of what population of situations?

From a statistical perspective, performance metrics such as error rates or expected losses are always defined with respect to a probability distribution. Reported model performance is therefore meaningful only relative to a well-defined distribution of inputs and outputs. In this paper, we refer to this distribution as the Application Domain (AD) of the system. For real-world ML systems, however, the AD is rarely available in explicit mathematical form. Instead, it must be characterized operationally through the procedures by which valid samples are generated or selected.

To address this problem, we introduce the concept of a Stochastic Application Domain Definition (SADD). A SADD is a textual specification of the sampling protocol that induces the distribution under which performance claims are intended to hold. In this way, the SADD links intended use, dataset construction, and statistical evaluation, providing a basis for interpretable testing and independent verification.

A simple example illustrates the underlying issue. Consider a kitchen knife designed for cutting vegetables. Suppose the knife performs well in tests on broccoli, apples, and melons. Even so, this result does not yet specify under which conditions the claim should be trusted. Performance may differ between private and commercial kitchens, between types of produce, or across different usage conditions. The same principle applies to ML systems: a performance claim is meaningful only with respect to the population of situations under which the system is evaluated.

In ML, this issue is particularly important because testing relies on empirical estimates of expected performance. Such estimates are informative only if the test samples can be understood as draws from a well-defined domain of application (5). Different data-generation procedures may therefore lead to different performance estimates even for the same system and nominal use case. The central question is therefore not only which metric to report, but also how the relevant domain of evaluation is defined. This is precisely the role of the Application Domain and, in the framework proposed here, of the SADD.

1.1 Functional trustworthiness

For the purpose of functional trustworthiness assessment, three elements are essential (22):

(1) a clear **Stochastic Application Domain Definition**, which specifies the domain under which the system is intended to operate and under which its performance claims are to be understood;

(2) **risk-based minimum performance requirements**, which define which performance quantities matter and what levels are acceptable in view of the risks of the intended application and relevant foreseeable misuse scenarios; and

(3) **statistically valid testing based on independent random samples**, which enables performance estimation under the application domain defined by the SADD.

This paper focuses on the first of these elements. Our claim is that, without a sufficiently explicit definition of the application domain, even a well-designed statistical test cannot yield a well-interpretable guarantee.

2 Motivation and problem statement

The challenge of defining the application domain is closely linked to a fundamental difference between traditional software systems and ML systems. In traditional software engineering, the intended functionality is typically specified in formal or highly structured terms. Developers encode rules and procedures to solve tasks whose input–output relations are sufficiently well understood, such as sorting algorithms, database queries, or cryptographic methods. In such cases, the domain of valid inputs and the desired behaviour can often be described precisely.

ML systems differ in that the input–output relation is not explicitly programmed but learned from data. Model behaviour emerges from optimization on examples rather than from a fully specified

rule set, and the training objective may differ from the ultimate task-level performance criterion.[1] As a result, the operational domain under which the system is expected to perform is often left implicit, and performance claims may become detached from the conditions of actual use.

This issue can be expressed using the standard notion of statistical risk in machine learning. Let f denote a model, L a loss function, and let $(X, Y) \sim P$ denote the distribution of inputs and outputs under which the system is evaluated. The statistical risk is defined as

$$R(f) = \mathbb{E}_{(X,Y)\sim P}[L(Y, f(X))]. \tag{1}$$

Most learning algorithms estimate this quantity indirectly by minimizing the empirical risk, i.e., the average loss on a finite dataset.

The key observation is that the risk in (1) is defined only with respect to the distribution P. Reported model performance is therefore meaningful only relative to a specified distribution. If training, testing, and deployment occur under different distributions, performance guarantees may not reflect the behaviour of the system in its intended application setting.

In practice, however, the relevant distribution P is rarely known in explicit mathematical form. Real-world data-generating processes are complex and shaped by institutional, temporal, geographical, and technical constraints. Consequently, the application domain of an ML system typically cannot be specified through a complete probabilistic model. Instead, it must be characterized operationally through the procedure by which valid samples are generated or selected. In many empirical disciplines, the target of inference is defined in exactly this way: not through an explicit distribution, but through a documented sampling or study protocol (19; 11; 25; 21).

3 The Stochastic Application Domain Definition

3.1 Application domains as probability distributions

Performance guarantees for ML systems are statistical statements. Whether one reports an error rate, a sensitivity, a calibration quantity, or an expected loss, the reported number always refers—implicitly or explicitly—to a distribution over relevant situations. In this sense, the application domain of an ML system is the probability distribution under which its performance is to be interpreted.

This perspective is not optional. If the relevant distribution changes, then the meaning of the reported performance changes as well. A system can perform excellently under one distribution and poorly under another, even if the underlying task appears similar at a superficial level.

Instead of specifying this distribution analytically, we characterize it operationally through the procedure by which valid samples are generated or selected. This motivates the notion of a protocol-induced distribution.

3.2 Protocol-induced distributions

Let $\mathcal{X}$ denote the space of possible inputs. A practical sampling procedure can be understood as a rule that uses randomness, together with operational constraints, to generate samples from $\mathcal{X}$. Such a procedure induces a probability distribution on the samples that can occur under that protocol.

Formally, let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space representing the randomness involved in the sampling process, and let

$$g : \Omega \to \mathcal{X} \tag{2}$$

be a measurable mapping that transforms this randomness into an observable sample. The protocol then induces a probability distribution on $\mathcal{X}$ via the pushforward measure

$$P_{\Pi}(A) = \mathbb{P}\big(g^{-1}(A)\big) \tag{3}$$

[1] For example, cross-entropy loss may be optimized during training while classification error is used for evaluation; perplexity may be optimized although question answering quality is of ultimate interest.

for all measurable sets $A \subseteq \mathcal{X}$. We call P_Π the **protocol-induced distribution** and identify the application domain with such a distribution.

This perspective implies that the application domain is not simply "the real world" or a vague intended-use statement, but the distribution induced by a specified sampling protocol in a given operational setting. If different parties generate evaluation data using different protocols, they generally induce different distributions, even if they believe they are evaluating the same use case. Reported performance differences may therefore arise not from changes in the model, but from changes in the domain of evaluation.

For certification and independent verification, this is crucial. A performance claim becomes interpretable only if the underlying data-generation procedure is described clearly enough that another qualified party can reconstruct the relevant distribution to a practically sufficient extent.

3.3 Definition of the SADD

Since the relevant application domain is typically not available as an explicit mathematical object, it must be communicated in another way. We propose to do so through a textual, process-oriented description of the protocol that induces the domain.

We call this description the **Stochastic Application Domain Definition (SADD)**.

The SADD is therefore not only a statement of intended use, nor just a list of environmental conditions, nor a benchmark description. It is a **textual specification of the sampling protocol** that defines what counts as a valid draw from the application domain. Its purpose is to make explicit, in operational terms, the process through which samples relevant for testing are to be generated or selected.

A SADD should describe, as far as relevant for the intended use,

- the real-world process or objects that give rise to the data,
- inclusion and exclusion criteria,
- the geographical and temporal scope,
- the acquisition setting,
- relevant actors, devices, and procedural constraints, and
- the rules according to which valid samples are selected or generated.

Once the application domain is understood as a protocol-induced distribution P_Π, performance quantities can be defined in the standard statistical way. For a given evaluation functional φ, one may write

$$\theta = \mathbb{E}_{(X,Y)\sim P_\Pi}[\varphi(X, Y, f)].$$

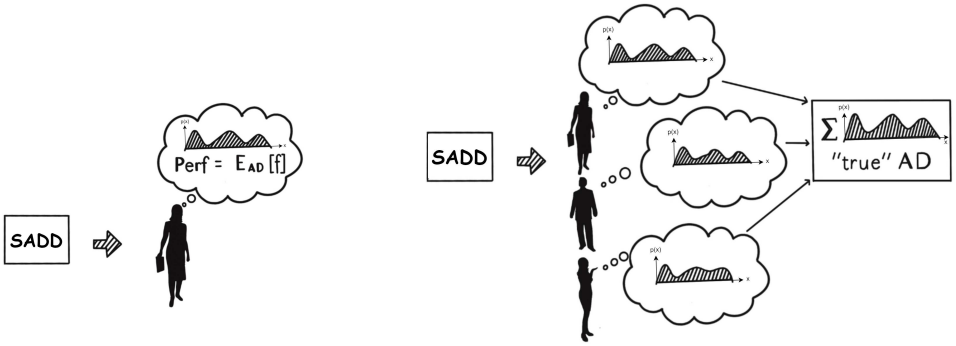
In practice, θ is estimated using samples drawn according to the sampling procedure described by the SADD. The interpretability of the resulting estimate therefore depends critically on the adequacy of that procedure.

3.4 Interpretation of a SADD

A SADD is a textual document and therefore necessarily admits interpretation. Different stakeholders may read the same SADD against different backgrounds and with different practical assumptions. The goal of a good SADD is not to eliminate all interpretation—which would be unrealistic—but to reduce ambiguity sufficiently for communication, testing, and certification purposes.

Figure 1a illustrates the individual perspective: a reader forms a subjective understanding of the domain under which the system's performance claims are to be interpreted. This already has practical value, because the SADD informs developers, distributors, auditors, and users about the realm of intended application.

For certification purposes, however, a purely subjective reading is not sufficient. We therefore adopt a normative interpretation principle based on the notion of a **reasonably informed stakeholder**.



(a) Single user's interpretation (b) Normative interpretation

Figure 1: Interpreting the SADD. (a) A single informed user forms an individual understanding of the application domain described by the SADD. (b) For certification purposes, interpretation should be anchored in the understanding of a reasonably informed stakeholder community.

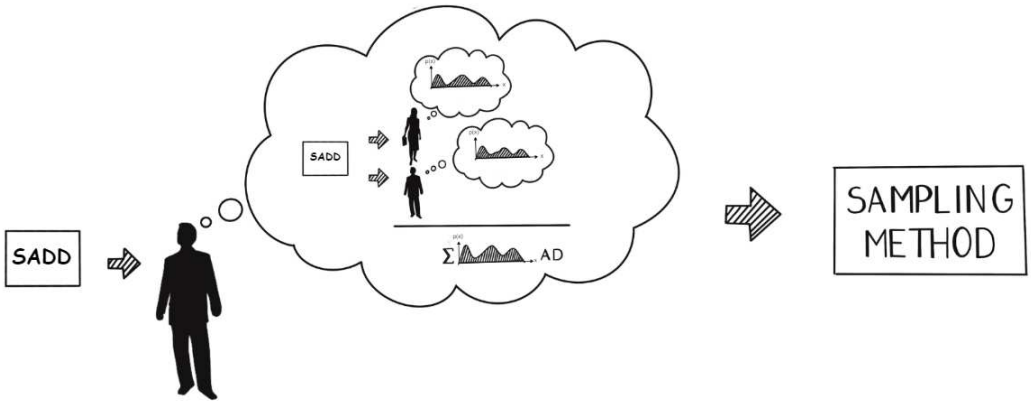


Figure 2: Deriving an operational sampling procedure from the SADD. An expert interprets the textual SADD while considering the reasonably expected understanding of qualified stakeholders and implements a feasible sampling procedure that approximates the intended application domain.

This notion is inspired by established approaches in legal and technical communication, where documents are interpreted not according to an arbitrary individual reading, but according to the understanding that can reasonably be expected from a qualified addressee in the relevant field.

In our context, the reasonably informed stakeholder is not a specific person but a reference construct. It represents the level of domain-specific understanding that can reasonably be expected from a competent actor in the intended field of application. The purpose of this construct is not to eliminate all variability in interpretation, but to provide a common reference point for communication and certification.

3.5 From SADD to sampling procedures

The practical purpose of a SADD is to describe how samples belonging to the application domain can arise. In doing so, it specifies which inputs are admissible members of the domain and outlines the conditions under which such samples occur in practice. In this sense, the SADD delineates a meaningful region within the space of all theoretically possible inputs to the AI system and describes a conceptual generative process for producing samples from this region.

In practice, however, the textual description of the SADD cannot usually be implemented literally. Real-world constraints such as limited data access, legal restrictions, costs, or geographic limitations

may prevent direct sampling from the idealized process described by the SADD. Therefore, the SADD must be translated into a feasible sampling procedure that approximates the protocol-induced distribution implied by the textual specification.

This translation is carried out by a domain expert who interprets the SADD while taking into account the reasonably expected understanding of the relevant stakeholder community (see Fig. 2). The resulting sampling procedure serves as an operational implementation of the SADD and provides the concrete mechanism for generating evaluation data used in statistical testing.

3.6 Sampling strategies

Once a SADD has specified the conditions under which valid samples arise, a sampling design is needed to generate evaluation data. Its role is to approximate the protocol-induced distribution defined by the SADD as closely as practical.

This is closely related to classical sampling theory (15; 19). Possible designs include simple random, cluster, and stratified sampling. Since different designs induce different effective distributions, they also lead to different interpretations of reported performance metrics. The sampling strategy is therefore part of the evaluation domain and should be explicitly documented and justified. Further details are given in Appendix A.

3.7 Relation to existing domain description practices

It is useful to distinguish the SADD from other concepts used to describe the intended operational setting of a system. A prominent example is the Operational Design Domain (ODD) in the automotive context, which specifies the conditions under which a system is intended to operate, such as weather, road type, traffic conditions, or time of day (17).

Such domain descriptions are valuable and often necessary. However, they do not by themselves define a sampling protocol and therefore do not uniquely determine a probability distribution. Two parties may fully agree on an ODD and nevertheless generate different evaluation datasets, thereby implicitly evaluating the system under different effective domains.

The SADD differs in exactly this respect. It does not only describe under which conditions a system is intended to operate, but also how valid samples from that domain are to be generated or selected. In this way, it turns an intended-use description into a stochastic object that supports statistically interpretable performance claims.

More broadly, several existing frameworks describe usage conditions, target populations, or contexts of use, but typically do not define the evaluation domain as a probability distribution induced by an explicit sampling protocol. The SADD is intended to complement these approaches by making this statistical reference domain explicit. A more detailed comparison is provided in Appendix B.

4 Case Study

To illustrate the practical role of a Stochastic Application Domain Definition, we consider a simple industrial inspection setting. The example is deliberately kept small in scope in order to isolate the statistical point. The purpose of the case study is not to present a realistic certification procedure in full detail, but to show how the meaning of a reported performance value depends on the sampling protocol through which the application domain is operationalized.

4.1 Industrial inspection example and sampling interpretation

Suppose an AI system has been developed to detect scratches in the paint of metal parts produced by a specific machine type operated at several locations in Austria. The system classifies each part as either faulty or intact based on images taken during production (28). A company considering deployment on its own machine cannot determine the suitability of the system solely from an overall accuracy value reported by the developer. Instead, the assessment must be made relative to the intended application domain defined by the SADD and the statistical testing procedure used to estimate performance.

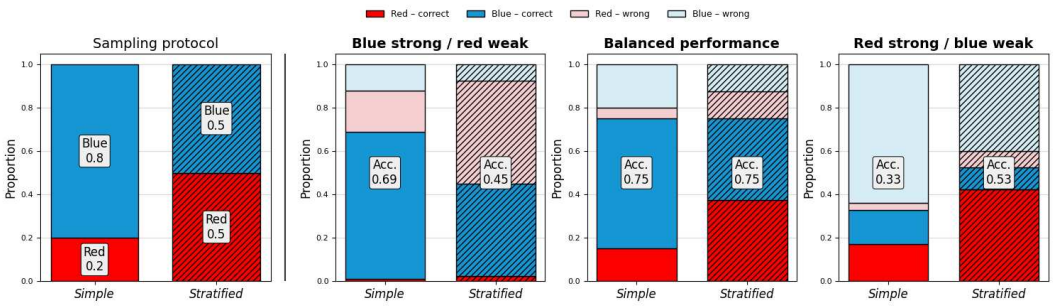


Figure 3: Influence of the sampling protocol on the interpretation of classification accuracy in the metal-part inspection example. The leftmost panel shows the evaluation dataset induced by two sampling strategies: simple random sampling from production (20% red, 80% blue) and stratified sampling by color (50% red, 50% blue). The three panels on the right show different class-wise performance profiles and the resulting aggregate accuracies under the two sampling strategies. Dark segments denote correctly classified parts and light segments incorrectly classified parts; hatched bars indicate the stratified design. The figure shows that aggregate accuracy is not self-interpreting but depends on the sampling protocol that defines the reference distribution (28).

To illustrate this, assume that the metal parts occur in only two colors, red and blue, and that accuracy is used as the performance metric. In actual production, suppose that 80% of parts are blue and 20% are red. Consider the two sampling strategies shown in Figure 3. Under **Strategy A**, the test set is obtained by simple random sampling from production, so that the evaluation sample reflects the natural 80/20 distribution and the reported accuracy estimates the probability that a randomly selected production part is classified correctly. Under **Strategy B**, the test set is stratified by color and both colors are sampled in equal proportions, so the resulting accuracy refers to a distribution that assigns equal weight to both colors.

The difference becomes particularly visible when class-wise accuracies differ. If the system performs well on blue parts but poorly on red parts, overall accuracy under simple random sampling may still appear high because blue parts dominate production. A stratified evaluation would make the weakness on red parts more visible by giving both colors equal weight. The key point is that performance metrics are not self-interpreting: an accuracy value is always an expectation with respect to some distribution. Different sampling strategies therefore lead to different interpretations of the same reported number.

4.2 Relation to the SADD

The case study illustrates the practical function of the SADD. A sufficiently explicit SADD does not merely state that the system is intended for "painted metal parts produced by machine A in Austria," but also clarifies in operational terms what counts as a valid sample for testing and how relevant variation within the intended domain should be represented.

For example, the SADD may imply that the intended domain corresponds to the actual production stream of eligible parts under ordinary operating conditions. In that case, simple random sampling from production would naturally approximate the induced distribution. Alternatively, the intended claim may concern balanced performance across relevant subdomains such as color categories, in which case a stratified design may be justified. In this way, the SADD connects the informal intended-use statement to a concrete sampling design and determines which distribution the evaluation aims to approximate, making the resulting performance measure interpretable and open to independent scrutiny.

4.3 Lessons from the example

Even in this simplified setting, the example illustrates several general points. First, the claim that a dataset is "representative" is incomplete unless the sampling protocol or induced distribution with respect to which representativeness is asserted is made explicit. Second, different sampling strategies

may be appropriate for different evaluation goals, but they should not be treated as interchangeable, since they correspond to different application domains in the statistical sense developed in this paper.

Third, independent testing and certification require more than benchmark results or a generic intended-use statement. They require a sufficiently explicit description of the domain under which the performance claim is supposed to hold, which is precisely the role of the SADD. While the example is intentionally simple, the same logic applies in more complex settings such as medical diagnosis, industrial quality assurance across multiple production sites, or multimodal systems deployed under heterogeneous environmental conditions.

5 Limitations

The proposed framework has important limitations.

Residual ambiguity of textual definitions. Even a carefully written SADD remains a textual document. Different qualified readers may still interpret certain terms or operational details differently. The framework does not eliminate this issue; rather, it makes it explicit and seeks to reduce it through better specification and normative interpretation. In practice, the expert deriving a sampling procedure from the SADD may still introduce bias through their own assumptions about how others would interpret the text.

Feasibility constraints in sampling. In many real-world settings, the ideal protocol suggested by a SADD cannot be implemented exactly. Access restrictions, costs, legal constraints, and organizational barriers may make direct sampling impossible. The resulting sampling design is then an approximation of the intended domain. This does not invalidate the framework, but it means that the relationship between the SADD and the actual sampling procedure must remain transparent and justified.

Rare events and outliers. The framework is primarily designed to support statistically meaningful evaluation on the intended application domain. It is not, by itself, a method for guaranteeing performance on rare cases, extreme outliers, or adversarial situations that are only weakly represented under the induced distribution. Additional robustness analyses and targeted stress tests may therefore still be necessary.

General-purpose AI. General-purpose AI (GPAI) models pose a particular challenge for the framework proposed here. Under the EU AI Act, general-purpose AI models are characterised by their generality and their capability to competently perform a wide range of distinct tasks, rather than being tied to one narrowly specified purpose (1). Accordingly, at the foundation-model level, there may be no single application domain in the sense discussed in this paper. Assessment at that level may therefore focus on training procedures, benchmark results, documentation duties, or model-level risk properties. However, once a GPAI model is integrated into or adapted for a specific downstream task, it becomes meaningful again to define a task-specific SADD. At that stage, the same logic proposed in this paper applies: functional trustworthiness claims should be interpreted with respect to an explicitly defined application domain.

6 Conclusion and outlook

This paper introduced a statistical perspective on the intended use of ML systems. We argued that performance guarantees are meaningful only with respect to a probability distribution, which we refer to as the Application Domain (AD). Because this distribution is typically unknown for real-world systems, we proposed the Stochastic Application Domain Definition (SADD), a textual specification of the sampling protocol that induces the distribution under which performance claims are intended to hold.

The SADD links intended use, representative data generation, and statistically interpretable testing. It supports communication along the value chain by clarifying the scope under which performance claims apply and provides the basis for deriving sampling procedures for statistically valid testing and independent verification. The framework does not assume that application domains can be specified perfectly; rather, it makes explicit that practical testing procedures inevitably rely on approximations and therefore require transparent documentation.

From a certification perspective, this transparency is essential. Process and documentation quality alone cannot replace a meaningful assessment of the functional trustworthiness of the deployed ML system. Future work should therefore investigate how SADDs can be standardized across sectors, how their quality can be assessed, how disagreements in interpretation can be documented, and how derived sampling procedures can be validated against the intended domain.

References

[1] Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence and amending regulations (ec) no 300/2008, (eu) no 167/2013, (eu) no 168/2013, (eu) 2018/858, (eu) 2018/1139 and (eu) 2019/2144 and directives 2014/90/eu, (eu) 2016/797 and (eu) 2020/1828 (artificial intelligence act) (text with eea relevance), Jul 2024.

[2] Matthew Arnold, Rachel K. E. Bellamy, Michael Hind, et al. Factsheets: Increasing trust in ai services through supplier's declarations of conformity. IBM Journal of Research and Development, 63(4/5):6:1–6:13, 2019.

[3] Emily M. Bender and Batya Friedman. Data statements for natural language processing: Toward mitigating system bias and enabling better science. In Transactions of the ACL, volume 6, pages 587–604, 2018.

[4] Yoshua Bengio, Geoffrey Hinton, Andrew Yao, Dawn Song, Pieter Abbeel, Yuval Noah Harari, Ya-Qin Zhang, Lan Xue, Shai Shalev-Shwartz, Gillian Hadfield, et al. Managing ai risks in an era of rapid progress. arXiv preprint arXiv:2310.17688, 2023.

[5] Christopher M Bishop and Nasser M Nasrabadi. Pattern recognition and machine learning, volume 4. Springer, 2006.

[6] Gary S. Collins, Karel G. M. Moons, et al. Tripod+ai: Reporting guideline for studies developing, validating, or updating a prediction model that uses artificial intelligence. BMJ, 2024.

[7] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. arXiv preprint arXiv:2407.21783, 2024.

[8] Carlos Galán. The certification as a mechanism for control of artificial intelligence in europe, 2019.

[9] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, et al. Datasheets for datasets. In Communications of the ACM, volume 64, pages 86–92, 2021.

[10] Delaram Golpayegani, Isabelle Hupont, Cecilia Panigutti, Harshvardhan J Pandit, Sven Schade, Declan O'Sullivan, and Dave Lewis. Ai cards: towards an applied framework for machine-readable ai and risk documentation inspired by the eu ai act. In Annual Privacy Forum, pages 48–72. Springer, 2024.

[11] Robert M Groves and Lars Lyberg. Total survey error: Past, present, and future. Public opinion quarterly, 74(5):849–879, 2010.

[12] Isabelle Hupont, David Fernández-Llorca, Sandra Baldassarri, and Emilia Gómez. Use case cards: a use case reporting framework inspired by the european ai act. Ethics and Information Technology, 26(2):19, 2024.

[13] Information technology — artificial intelligence — overview of trustworthiness in artificial intelligence. ISO/IEC TR 24028:2020, 2020. International Organization for Standardization.

[14] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. arXiv preprint arXiv:2401.04088, 2024.

[15] Göran Kauermann and Helmut Küchenhoff. Stichproben: Methoden und praktische Umsetzung mit R. Springer-Lehrbuch. Springer Berlin Heidelberg.

[16] Johann Laux, Sandra Wachter, and Brent Mittelstadt. Trustworthy artificial intelligence and the european union ai act: On the conflation of trustworthiness and acceptability of risk. Regulation & Governance, 18(1):3–32, 2024.

[17] Chung Won Lee, Nasif Nayeer, Danson Evan Garcia, Ankur Agrawal, and Bingbing Liu. Identifying the operational design domain for an automated driving system through assessed risk. In 2020 IEEE Intelligent Vehicles Symposium (IV), pages 1317–1322, 2020. doi: 10.1109/IV47402.2020.9304552.

[18] Chung Won Lee, Nasif Nayeer, Danson Evan Garcia, Ankur Agrawal, and Bingbing Liu. Identifying the operational design domain for an automated driving system through assessed risk. In IEEE Intelligent Vehicles Symposium, pages 1317–1322, 2020.

[19] Sharon L. Lohr. Sampling: Design and Analysis. Brooks/Cole, 2nd ed edition.

[20] Margaret Mitchell, Simone Wu, Andrew Zaldivar, et al. Model cards for model reporting. In Proceedings of the ACM Conference on Fairness, Accountability, and Transparency, pages 220–229, 2019.

[21] John D. Musa. Operational profiles in software-reliability engineering. IEEE Software, 10(2):14–32, 1993.

[22] Bernhard Nessler, Thomas Doms, and Sepp Hochreiter. Functional trustworthiness of ai systems by statistically valid testing. arXiv, 2310.02727, 2023.

[23] Maximilian Poretschkin, Anna Schmitz, Maram Akila, Linara Adilova, Daniel Becker, Armin B Cremers, Dirk Hecker, Sebastian Houben, Michael Mock, Julia Rosenzweig, et al. Guideline for designing trustworthy artificial intelligence. 2023.

[24] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10684–10695, 2022.

[25] Peter M Rothwell. External validity of randomised controlled trials:"to whom do the results of this trial apply?". The Lancet, 365(9453):82–93, 2005.

[26] SAE International. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles, 2018.

[27] John Schulman, Barret Zoph, and Christina Kim. Introducing chatgpt, Nov 2022. Online, accessed 2023-05-17. URL: https://openai.com/blog/chatgpt.

[28] Kajetan Schweighofer, Barbara Brune, Lukas Gruber, Simon Schmid, Alexander Aufreiter, Andreas Gruber, Thomas Doms, Sebastian Eder, Florian Mayer, Xaver-Paul Stadlbauer, Christoph Schwald, Werner Zellinger, Bernhard Nessler, and Sepp Hochreiter. Safe and certifiable ai systems: Concepts, challenges, and lessons learned, 2025. URL: https://arxiv.org/abs/2509.08852.

[29] George Sharkov, Christina Todorova, and Pavel Varbanov. Strategies, policies, and standards in the eu towards a roadmap for robust and trustworthy ai certification. Information & Security, 50(1):11–22, 2021.

[30] U.S. Food and Drug Administration. Considerations for the use of artificial intelligence to support regulatory decision-making for drug and biological products, 2025. Draft Guidance.

[31] Robert F. Wolff, Karel G. M. Moons, Richard D. Riley, et al. Probast: A tool to assess risk of bias and applicability of prediction model studies. Annals of Internal Medicine, 170(1):51–58, 2019.

A Common sampling strategies

The challenge of obtaining representative samples under practical constraints is well known in statistical sampling theory. There, one aims to draw samples from a target population by means of a documented sampling design. We adapt these ideas here to the setting in which the target distribution is defined not by a complete population list, but by a textual domain specification (15; 19).

A.1 Simple random sampling

A simple random sample is the conceptual gold standard for representative sampling, because every eligible element has the same probability of being selected. If such sampling is possible, it provides the cleanest basis for statistical inference.

In the context of a SADD, however, simple random sampling is often not feasible. It would require access to a sufficiently complete sampling frame corresponding to the intended application domain. For many real-world ML applications, no such frame exists in explicit form. Even when a partial frame exists, legal, economic, and logistical constraints often prevent direct random access to all eligible units. For that reason, simple random sampling is usually best understood as an ideal benchmark rather than a practically available design.

A.2 Cluster sampling

Cluster sampling addresses this difficulty by using naturally occurring groups, such as hospitals, schools, factories, devices, or regions, as intermediate sampling units. Instead of sampling directly from the entire target domain, one first selects clusters and then samples within them.

In a SADD-based setting, this is often a natural way of operationalizing the intended domain. The sampling protocol may follow a hierarchical structure: for example, choose a country, then a clinic, then a department, then a device, and finally an observation recorded with that device. Each stage corresponds to an explicit design choice in the operational sampling protocol.

For cluster sampling to approximate the intended application domain well, the clusters should be chosen in a way that does not systematically distort the target domain. In practice, cluster sampling is often more feasible and cost-effective than direct random sampling, but it usually increases sampling variance and requires explicit treatment of intra-cluster dependence in the statistical analysis (15; 19).

A.3 Stratified sampling

Stratified sampling divides the domain into relevant subgroups, or strata, and samples separately within each of them. In contrast to cluster sampling, strata are defined to be internally homogeneous with respect to selected characteristics, while the full set of strata captures domain variability.

In a SADD-based framework, stratification is appropriate when certain variables are known to be relevant for the intended application domain, such as country, device type, age group, clinical subgroup, or environmental condition. The stratum definitions and the allocation of samples across strata must then be made explicit.

Stratified sampling can improve precision and ensure that important subdomains are adequately represented. However, it requires that eligible units can be assigned to strata in a meaningful way and that the weighting scheme used for aggregate performance estimation is taken into account when interpreting overall results (15; 19).

B Positioning with Respect to Existing Approaches

The problem of specifying the conditions under which the performance of an AI system should be interpreted arises in multiple research communities, including software reliability engineering, autonomous systems safety, machine learning documentation, and clinical AI evaluation. Existing approaches operationalize the application domain in different ways, typically emphasizing either usage distributions, operational conditions, target populations, or structured documentation of intended use. In the following, we review the main strands of work relevant to the specification of the application domain.

B.1 Operational profiles in software reliability engineering

One of the earliest formal approaches to operationalizing the domain under which software reliability claims should be interpreted is the operational profile introduced by Musa (21). An operational profile is a quantitative characterization of system usage, defined as a probability distribution over the operations or input classes that occur during actual use. The operational profile is used to guide testing and reliability estimation by ensuring that test cases are sampled according to the expected usage distribution.

This idea closely parallels the statistical perspective adopted in this paper: reliability or performance metrics are meaningful only relative to a distribution of use cases. However, operational profiles are typically defined over software operations or usage events rather than over real-world situations that generate inputs to a machine learning system. As a result, they are well suited to traditional software systems but less directly applicable to modern ML systems whose inputs arise from complex real-world processes.

B.2 Operational design domains in autonomous systems

In the domain of automated driving, the most widely used concept for describing the scope of system operation is the Operational Design Domain (ODD). The ODD specifies the conditions under which an automated driving system is designed to function safely, including environmental conditions, road types, traffic situations, and geographical or temporal constraints (26; 18).

The ODD provides a structured way of describing the operational context of a system and is widely used in safety cases and certification processes for automated driving. Recent standardization efforts such as ASAM OpenODD further aim to formalize ODD descriptions and provide machine-readable representations of operational domains.

While ODDs provide an explicit specification of admissible operating conditions, they typically do not define a sampling procedure or probability distribution over situations within the domain. Consequently, different parties may evaluate the same system within the same ODD using different datasets and thereby implicitly evaluate different effective distributions. In contrast, the framework proposed in this paper explicitly links the domain specification to a stochastic sampling protocol.

B.3 Target population and intended setting in clinical AI

In clinical prediction modeling and medical AI, a related problem is addressed through the specification of the target population, intended setting, and intended use of a model. Reporting guidelines such as TRIPOD-AI require authors to document the healthcare setting, target population, intended purpose, and intended users of a predictive model (6).

Closely related concepts include targeted validation and risk-of-bias assessment frameworks such as PROBAST, which evaluate whether a study population and setting are representative of the intended use of the model (31). These approaches emphasize that model performance must be interpreted relative to the population and setting in which the system is intended to operate.

Compared with the framework proposed here, these approaches primarily specify the population and setting for which a model is intended, but typically do not formalize the statistical domain as a distribution induced by an explicit sampling protocol.

B.4 Context-of-use frameworks in regulatory evaluation

Regulatory frameworks for AI and computational models increasingly emphasize the notion of context of use. For example, recent FDA guidance on AI models used in regulatory decision-making defines the context of use as the specific role and scope of the model within a decision process (30). The context of use describes how model outputs will be used, the question being addressed, and the consequences of model errors.

Such frameworks operationalize the domain of application by situating the model within a specific decision workflow. While this provides important information about the purpose and risk implications of the model, it does not directly specify the distribution of situations under which model performance should be evaluated.

B.5 Documentation frameworks for AI systems and datasets

A growing body of work proposes structured documentation frameworks to describe the intended use and limitations of AI systems and datasets. Model Cards (20) aim to communicate key information about machine learning models, including intended uses, evaluation conditions, and potential limitations. Similarly, Datasheets for Datasets (9) and Data Statements for NLP datasets (3) document dataset provenance, composition, and intended uses in order to improve transparency and enable more responsible deployment.

Related proposals such as FactSheets (2), Use Case Cards (12), and AI Cards (10) extend this idea to broader AI system documentation, including usage scenarios, system capabilities, and potential risks. These approaches help clarify the contexts in which a system is expected to be used and discourage deployment outside the intended domain.

However, these frameworks primarily focus on documentation and communication rather than on defining a statistical reference distribution for evaluation. As a result, they provide valuable contextual information but do not by themselves specify how representative evaluation datasets should be generated.

B.6 Position of this work

Taken together, existing approaches operationalize the application domain of AI systems in several complementary ways: through usage distributions (operational profiles), operational conditions (ODDs), target populations and settings (clinical evaluation frameworks), decision contexts (context-of-use models), and structured documentation of intended use.

The framework proposed in this paper builds on these ideas but emphasizes the statistical interpretation of performance claims. Specifically, we model the application domain as a probability distribution induced by a sampling protocol and introduce the Stochastic Application Domain Definition (SADD) as a textual specification of that protocol. This perspective explicitly links intended use descriptions with the sampling procedures required for statistically interpretable evaluation.

C Operationalization of Dataset Representativeness through the Stochastic Application Domain Definition (SADD)

C.1 Scope

This section specifies requirements and procedures for operationalizing the requirement of dataset representativeness as stated in Article 10(3) of Regulation (EU) 2024/1689 (Artificial Intelligence Act). It defines the concept of the Stochastic Application Domain Definition (SADD) and establishes requirements for its use in the specification, construction, and evaluation of testing datasets for AI systems.

The provisions of this section apply to the development, evaluation, and certification of AI systems for which performance claims are made with respect to an intended application domain. The objective is to ensure that reported performance metrics are interpretable and statistically valid with respect to the conditions under which the AI system is intended to operate.

Approach	Domain representation	Operationalization	Dist.?	Use context
Operational Profile (21)	Usage distribution	Probability distribution over software operations or usage events	Yes	Software reliability engineering
Operational Design Domain (ODD) (26; 18)	Operational conditions	Admissible environmental and operational constraints	No	Automated driving safety and certification
Target Population / Intended Setting (6; 31)	Population and setting	Specification of target population, intended setting, users, and purpose	Partial	Clinical prediction and medical AI
Context of Use (COU) (30)	Decision context	Role and scope of the model in a decision workflow	No	Regulatory evaluation
Model Cards (20)	Intended use and evaluation conditions	Structured documentation of intended uses, results, and limitations	No	Model transparency and documentation
Datasheets / Data Statements (9; 3)	Dataset provenance and population	Documentation of composition, collection, demographics, and intended uses	Indirect	Dataset documentation and bias analysis
FactSheets / AI documentation frameworks (2; 12; 10)	Usage scenarios and governance	Structured documentation of use cases, risks, and operational scope	No	AI governance and certification documentation
SADD (this work)	Stochastic application domain	Textual specification of a sampling protocol inducing an evaluation distribution	Yes	Statistical evaluation and certification

Table 1: Comparison of existing approaches for specifying the application domain of AI systems. Existing approaches typically operationalize the domain through conditions, populations, workflows, or documentation frameworks. In contrast, the Stochastic Application Domain Definition (SADD) explicitly links the intended application domain to a sampling protocol that induces the probability distribution under which performance claims are interpreted.

C.2 Normative references

The following documents are referred to in this section and are indispensable for its application.

- Regulation (EU) 2024/1689 of the European Parliament and of the Council (Artificial Intelligence Act)
- ISO/IEC TR 24028:2020, Information technology — Artificial intelligence — Overview of trustworthiness in artificial intelligence

C.3 Terms and definitions

For the purposes of this section, the following terms and definitions apply.

Application domain (AD). Probability distribution over relevant input–output situations under which the performance of an AI system is intended to be interpreted.

Stochastic Application Domain Definition (SADD). Textual specification of the sampling protocol that induces the probability distribution corresponding to the application domain.

Protocol-induced distribution. Probability distribution over admissible samples resulting from the stochastic procedure specified by the sampling protocol described in the SADD.

Evaluation dataset. Dataset constructed for the purpose of estimating the performance of an AI system with respect to the application domain defined by the SADD.

Sampling procedure. Operational implementation of the sampling protocol defined in the SADD used to generate or select evaluation samples.

C.4 General principle of representativeness

For the purposes of Article 10(3) of the Artificial Intelligence Act, a dataset shall be considered representative if it is generated according to a sampling procedure that approximates the protocol-induced distribution defined by the SADD.

Representativeness shall therefore be interpreted with respect to the application domain defined by the SADD and not solely with respect to superficial similarity to real-world data or previously used datasets.

Performance metrics reported for an AI system shall be interpreted as estimates of statistical quantities defined with respect to the protocol-induced distribution associated with the SADD.

C.5 Requirements for the Stochastic Application Domain Definition

A SADD shall be documented for each AI system for which performance claims are made.

The SADD shall include the following elements, where applicable:

1. description of the real-world process, environment, or objects that generate the inputs to the AI system;
2. specification of inclusion and exclusion criteria defining admissible samples;
3. description of the geographical and temporal scope of the intended application;
4. description of relevant actors, devices, sensors, and data acquisition procedures;
5. description of relevant operational constraints or conditions;
6. description of the mechanism through which valid samples arise in the real-world process;
7. specification of any relevant stratification variables or subdomains;
8. description of foreseeable sources of variation within the application domain.

The SADD shall be documented in a manner that enables a qualified independent party to reconstruct a sampling procedure that approximates the intended application domain.

C.6 Derivation of sampling procedures

A sampling procedure shall be derived from the SADD for the purpose of constructing evaluation datasets.

The derivation of the sampling procedure shall satisfy the following requirements:

1. The procedure shall approximate the protocol-induced distribution defined by the SADD to the extent that is reasonably achievable in practice.
2. The procedure shall be documented in sufficient detail to allow independent reproduction of the sampling process.
3. Any deviations from the idealized sampling protocol implied by the SADD shall be documented and justified.

The derivation of the sampling procedure shall be carried out by a qualified domain expert or by a team possessing expertise in the application domain and statistical sampling methodology.

C.7 Sampling design

Evaluation datasets shall be generated using a documented sampling design consistent with the SADD.

The sampling design may include one or more of the following strategies:

- simple random sampling,
- stratified sampling,
- cluster sampling,
- multi-stage sampling.

The choice of sampling design shall be justified with respect to the characteristics of the application domain and the feasibility constraints of data collection.

Where stratified sampling is used, the definition of strata and the weighting scheme used for aggregate performance estimation shall be documented.

C.8 Construction of evaluation datasets

Evaluation datasets used for testing AI systems shall satisfy the following requirements:

1. The dataset shall be constructed using the sampling procedure derived from the SADD.
2. The sampling process shall be documented and auditable.
3. The dataset shall contain sufficient information to enable verification of compliance with the SADD.
4. The dataset shall be independent of the training and validation datasets used in model development, unless otherwise justified.

The dataset size shall be determined based on statistical considerations, including the desired precision of performance estimates.

C.9 Documentation requirements

The following documentation shall be maintained and made available to relevant stakeholders or conformity assessment bodies:

- the Stochastic Application Domain Definition;
- the derived sampling procedure;
- the sampling design and its justification;
- the dataset construction process;
- any deviations from the intended sampling protocol;
- statistical methods used for performance estimation.

The documentation shall allow an independent assessor to evaluate whether the evaluation dataset reasonably approximates the intended application domain.

C.10 Interpretation of performance metrics

Performance metrics reported for the AI system shall be interpreted as statistical estimates with respect to the protocol-induced distribution defined by the SADD.

Reported performance values shall therefore be accompanied by:

- a reference to the SADD under which the evaluation was conducted;
- the sampling design used to construct the evaluation dataset;
- the statistical uncertainty associated with the performance estimate.

Where evaluation datasets are generated using different sampling procedures, the resulting performance estimates shall be interpreted as referring to different effective application domains.

C.11 Limitations

The SADD framework does not eliminate all sources of ambiguity in the specification of application domains. Interpretation of a SADD shall be guided by the understanding that can reasonably be expected from qualified stakeholders in the relevant application domain.

Where practical constraints prevent exact implementation of the sampling protocol described in the SADD, the resulting sampling procedure shall be documented as an approximation of the intended domain.

Additional robustness testing, stress testing, or targeted evaluation may be required to assess performance in rare, extreme, or adversarial situations that are not adequately represented under the protocol-induced distribution.