Statistical Methods in Generative AI

Edgar Dobriban[1]

[1]Department of Statistics and Data Science, University of Pennsylvania, Philadelphia, PA, USA, 19104; email: dobriban@wharton.upenn.edu

Annu. Rev. Stat. Appl. 2025+. TBD:1–26

https://doi.org/10.1146/TBD

Keywords

Artificial Intelligence, generative AI, statistical methods, uncertainty quantification, AI evaluation, interventions and experiment design.

Abstract

Generative Artificial Intelligence is emerging as an important technology, promising to be transformative in many areas. At the same time, generative AI techniques are based on sampling from probabilistic models, and by default, they come with no guarantees about correctness, safety, fairness, or other properties. Statistical methods offer a promising potential approach to improve the reliability of generative AI techniques. In addition, statistical methods are also promising for improving the quality and efficiency of AI evaluation, as well as for designing interventions and experiments in AI. In this paper, we review some of the existing work on these topics, explaining both the general statistical techniques used, as well as their applications to generative AI. We also discuss limitations and potential future directions.

Contents

1. Introduction

Artificial Intelligence, and more specifically, Generative AI, is emerging as an important technology. Over the past few years a number of prominent generative AI technologies have been developed and have received widespread attention; ranging from text generation via large language models (ChatGPT, Claude, Llama, Gemini, DeepSeek, Qwen, etc), image generation via diffusion models (Dall-E, Stable Diffusion, etc), to scientific generative AI techniques used for protein generation (e.g., Watson et al. 2023, etc), DNA sequence editing (e.g., Ruffolo et al. 2025, etc).

Such methods have been quickly adopted by end users and institutions, both via direct usage, as well as integrated in other tools such as code assistants and web search agents. The scientific community has shown significant interest in using generative AI models, achieving a number of breakthrough results (see e.g., Davies et al. 2021, Hayes et al. 2025, etc), culminating in a 2024 Nobel Prize in Chemistry awarded in part for work with a significant component in protein structure design and generation (The Royal Swedish Academy of Sciences 2024).

Generative AI: The construction of probabilistic models over large semantic spaces (text, images, etc.) that allows sampling from these models, given certain inputs.

Yet, the adoption of generative AI (GenAI) methods more generally is hindered by their lack of reliability (see e.g., Farquhar et al. 2024, Strauss et al. 2025, Manduchi et al. 2025, etc). At their core, these methods rely on sampling from probability distributions over complex spaces that are learned from huge datasets. At the outset, GenAI does not provide any guarantees about correctness, safety, or any other desired criteria. While the performance and reliability of GenAI models is increasing steadily, so far, issues around reliability have not been successfully eliminated.

Statistical methods offer potential opportunities to improve the reliability of GenAI systems. In this paper, we review several examples, highlighting statistical methods with proven or potential applications in generative AI. We focus on four topics: improving and changing the behavior of systems, diagnostics and uncertainty quantification, AI evaluation, as well as interventions and experiment design. We highlight here that the approaches we discuss are as of now mainly in the research phase, and they are usually not yet deployed in mainstream generative AI products. Their eventual usefulness remains to be determined.

Table 1 Representative types of generative AI models and their input and output spaces.

Generative AI Model $\hat{p}$	Input Space $\mathcal{X}$	Output Space $\mathcal{Y}$
Language Models	Text	Text
Diffusion Models	Text	Images
Multimodal Language Models	Images, text	Images, text, sound, video
Protein Structure Generation	Amino acid sequence	3D structure

1.1. About This Review

Generative AI models are commonly studied separately, for each specific modality that they pertain to (text, images, video, etc), or based on the underlying technology (diffusion models, large language models or LLMs, etc). There are already a few reviews with significant coverage of statistics related to these topics individually, including Chen et al. (2024), Zhang et al. (2025) for diffusion models, Suh & Cheng (2024) for deep learning more generally, but also touching on generative models, and Ji et al. (2025) for language models.

Our focus is different, rendering our work largely non-overlapping with the above works. We focus on techniques that are applicable to all generative AI models, regardless of their modality. Moreover, with a few exceptions, do not focus on statistical methods that are applicable to generative AI only when the tasks of interest are essentially simple classification/regression tasks (e.g., multiple-choice question-answering with LLMs). For these, there are already numerous useful references.

We also focus specifically on statistical methodology for AI and omit discussion about statistical theory of generative AI, as well as statistics-adjacent methods that primarily leverage optimization or other techniques. Further, we omit certain topics, such as watermarking, which have already been discussed in detail in the above works. Due to space limitations, we mainly consider simple methods that have clear theoretical motivation and often provable guarantees. Moreover, we also omit discussion of how generative AI models can be used to improve statistical analysis, see e.g., Bashari et al. (2025) for a representative example.

Target audience. Our target audience includes statisticians eager to see how their expertise can drive impact in generative AI, AI researchers interested in how statistical methods can strengthen their tools, and scientists looking to better understand this emerging area. For this reason, our paper aims to be largely self-contained, with prerequisites that include knowledge of introductory undergraduate-level probability and statistics, and a basic familiarity with AI at the advanced undergraduate level.

1.2. What is Generative AI?

Generative AI usually refers to the use of generative models, which are learned probability distributions one can sample from. Concretely, consider an input space $\mathcal{X}$ (e.g., images, text, documents, their combinations, etc., represented in an appropriate way) and an output space $\mathcal{Y}$ (similarly, this could be images, text, audio, video, etc). See Table 1 for some examples.

Formally speaking, this includes as a special case standard statistical machine learning problems such as classification (when $\mathcal{Y}$ consists of the classes) and regression (when $\mathcal{Y} = \mathbb{R}$,

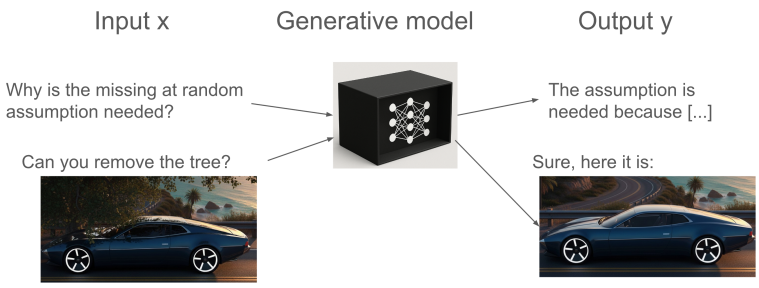

Figure 1

General workflow of a generative model: inputs (e.g., text prompts, images) are processed through a black-box model to produce outputs.

Generative Model: A generative model $\hat{p}$ provides a way to sample an output $Y \sim \hat{p}(\cdot \mid x)$ from the conditional distribution of $\hat{p}$ given any input $x \in \mathcal{X}$.

for instance). However, the cases of interest in generative AI are usually high-dimensional spaces $\mathcal{Y}$ representing objects that are semantically meaningful to humans, such as text—viewed as a sequence of symbols $(x_1, \ldots, x_k) \in V^k$ for a finite set V of symbols—or images, viewed as tensors representing pixels.

Generative AI models are often designed for interaction with humans. A simple protocol is as follows: The user inputs a specific $x \in \mathcal{X}$, for instance, a text prompt such as "How can I fix a broken lamp?". Then, the generative model $\hat{p}$ provides a way to draw a sample $Y \sim \hat{p}(\cdot \mid x)$ from the conditional distribution $\hat{p}$ given x; for instance, a textual response by the language model such as "To fix a broken lamp, you need to [...]". This is then returned to the user. See Figure 1 for an illustration. The interaction can also continue. For simplicity, we will mostly restrict our discussion to one round of interaction.

1.3. How is a Generative Model Learned?

The GenAI model $\hat{p}$ is usually obtained by empirical loss minimization, in a manner that is conceptually similar to that used in most standard statistical modeling and machine learning. This is performed by running an algorithm—often a stochastic gradient descent-based method or a variant—aiming to minimize a loss function over a large function class using a massive data set.

For instance, for language models, the training data consists of text represented as a collection of sequences $x = (x_1, x_2, \ldots, x_k)$, where for a finite set V usually referred to as a vocabulary, each $x_j \in V$, $j \leqslant k$. The length k of the strings can vary, up to a so-called context length L. Instead of viewing text as a sequence of letters, usually, text is encoded in tokens which are adjacent groups of letters that can offer more efficiency in the modeling process. For instance, "encoded" might consist of the tokens "en+code+d".

The loss used is often the negative log-likelihood $\theta \mapsto -\sum_{x \in \mathcal{D}} \log p_\theta(x)$. The function class $\theta \mapsto p_\theta$ usually consists of huge neural nets parametrized in very special ways, with up to hundreds of billions of parameters. The dataset used for training consists of text data crawled from the internet, enriched with high information content (Wikipedia, arXiv), and other sources such as books. Typical costs for training powerful Generative AI models can start from millions of US dollars, which means that only organizations with significant financial resources can perform the initial training.

1.4. Access Mode to the Generative Model

Black Box Access: An access model where we can only observe the output of a GenAI model, and not its internal workings.

An important consideration is the mode of access that we have to the generative model of interest. At the time of writing, the most powerful GenAI models are closed-source and run by commercial providers on their own cluster infrastructure, accessible only through querying. This leads to a *black box* mode of access, meaning that for any given input x, we can only observe the output Y, but not any internal components of the generation process $\hat{p}$. Sometimes some additional information is provided in a *gray box* access mode; for instance, the probability $\hat{p}(Y|x)$ may also be returned.

Open-source or open-weights GenAI models may be run on local machines depending on the available hardware.[1] In such cases, it is possible to inspect the internal workings of the models. However, since generative models tend to be highly complicated neural networks, using the internal information is challenging. Therefore, to maintain generality, we will usually focus on methods applicable to black box GenAI models. In a few cases, we will also discuss methods that require gray or white box access.

2. Statistical Methods in Generative AI

Our goal is to discuss a few emerging areas of research where statistical methods or ideas can be used in generative AI. A key starting point is that AI systems can be wrong. They can make any type of mistake, and they have no guarantees by default about correctness, content, logical consistency, safety, etc.[2] This stems intrinsically from their structure as sampling methods.[3]

While there are a variety of engineering approaches to improve reliability, such as endowing the AI models with external tools, such as calculators, web search, or access to a computer where they can run programs, the use of these tools is in turn orchestrated by a sampling-based generative AI model, which can still have reliability problems. Moreover, while there are constrained sampling methods that aim to ensure certain basic formatting and correctness criteria, their current scope is limited; for example, at the moment they cannot ensure logical correctness.

For these reasons, statistical methods that aim to improve the behavior of generative models—sometimes with provable guarantees—are particularly significant; we begin our discussion with this topic. Crucially, to have an impact in this area, statistical methods must directly align with AI practice and goals; endowing practically useful AI-enhancement methods with desirable guarantees. To put it another way, statistical methods act as simple tunable wrappers that can be calibrated to meet explicit error budgets with finite sample guarantees.

[1] They typically require powerful graphics processing units (GPUs) to be run with a reasonable speed.

[2] At the moment, it is only possible to rigorously understand and analyze individual components of GenAI models in isolation, see e.g., Noarov et al. (2025) for an example of analyzing the final decoding step in language models.

[3] It is often possible to ensure that the generation process is deterministic; for instance, in large language models, one can set the temperature parameter to zero. However, the resulting deterministic generative models still inherit the lack of intrinsic correctness due to the black box nature of the original model.

2.1. Improving and Changing Behavior

To improve the performance of a generative AI model, there are numerous of standard approaches relying on variants of standard training (e.g., supervised fine-tuning for LLMs). Once these have been exhausted, there is room for alternative techniques that change the behavior of the generative model in a non-standard way that can conceivably improve certain accuracy metrics, for instance, by returning a trimmed version of the input from which false claims have been deleted (see e.g., Mohri & Hashimoto 2024, etc). These techniques can be roughly categorized into changing (a) the output y, (b) the input x, or (c) the internal workings on the generative model $\hat{p}$.

Moreover, many of these techniques require a degree of hyperparameter tuning; for instance, determining how much to trim the outputs. This process of tuning can sometimes be endowed with statistical correctness guarantees (see Table 2), and so this is the first topic we review in this work.

2.1.1. An example: Controlling the probability of refusal/abstention. To get a sense of the types of problems that can be solved, as well as the types of statistical methods that are used, we will explain one specific example in some detail. We will consider the example of abstaining from generation when a risk score is high (see e.g., Farquhar et al. 2024, Yadkori et al. 2024, etc).

Refusal/Abstention: When a generative AI model does not return an output. Can be useful for improving safety.

Consider a given loss function[4] $\ell : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. This could measure the quality or safety of an input-output pair. There are many examples, including the negative log likelihood $\ell(x, y) = -\log \hat{p}(y|x)$ specified by the generative model itself, or the negative of a pre-trained reward function (measuring for instance safety), etc. The loss could depend on both x and y, or only on one of the two. If the loss only depends on the input x, it can capture either input ambiguity, or the dispersion in outputs generated by the model (Lin et al. 2024); or some combination thereof.

To improve user experience, a strategy is to refuse/refrain/abstain from answering when the loss is high. Specifically, we want to find a threshold τ such that when $\ell(x, Y) > \tau$ we should instead return a special message like ``Sorry I cannot answer.'', where $Y \sim \hat{p}(\cdot|x)$ is generated by the model $\hat{p}$. There is a trade-off: decreasing the threshold will ensure that only higher quality—lower loss—generations/answers are returned, but higher refusal also hampers utility to users.

Predictive Inference: The goal of endowing the outputs of predictive models—including GenAI—with statistical guarantees.

The threshold τ can be set by standard hyperparameter tuning, by checking the loss values and abstention rates on a dataset. However, there is also a statistical approach, which can provide provable guarantees on the behavior of the system under certain conditions. This approach is based on predictive inference/conformal prediction (Vovk et al. 2005), and the ideas date back to work on tolerance regions (e.g., Wilks 1941, Wald 1943, etc).

The statistical approach aims to guarantee generalization to a distribution D of prompts. The goal is then to control the abstention probability over the distribution D, which can be written as $\Pr_{X\sim D,\, Y\sim\hat{p}(\cdot|X)}\big(\ell(X, Y) > \tau\big)$. We do not fully know the distribution D, because it represents the behavior of future users. However, we assume that we have a *calibration dataset*—also referred to as a validation or hold-out dataset—$D_n = \{X_1, \ldots, X_n\}$ of prompts which we view as an i.i.d. sample from D. This is collected based on user inter-

[4] In the literature cited above, this is sometimes called a risk score, but we will not use that term, in order to avoid a conflict of terminology with the classical notion of risk—namely, expected loss—from statistical decision theory.

Table 2 Types of methods that change the behavior of generative AI systems; most of them endowed with statistical guarantees. Some methods belong to multiple categories.

Technique	Type	Examples
Change output	Additional output type	Highlight parts of output (Sun et al. 2022, Vasconcelos et al. 2025)
		Abstain from generation when a risk score is high (Farquhar et al. 2024, Yadkori et al. 2024)
		Add "Everything Else" as a possible answer (Noorani et al. 2025)
	Set of outputs	Construct prediction interval for each output coordinate (Horwitz & Hoshen 2022, Teneggi et al. 2023)
		Generate set of outputs (Quach et al. 2024, Gui et al. 2024, Nag et al. 2025)
	Trimmed output	Delete parts of output until correctness is achieved (Khakhar et al. 2023, Mohri & Hashimoto 2024)
		Find small parent set of possible outputs in a directed acyclic graph (Zhang et al. 2024)
	Regenerated output	Reformulate output until it is appropriately correct and specific (Jiang et al. 2025)
	Task-specific output	Train model to improve performance in downstream task (Band et al. 2024)
		Construct prediction intervals for latent variables of a generated output (Sankaranarayanan et al. 2022)
		Interactively ask questions that maximize the informativeness of the answers (Chan et al. 2025)
Change input	Set of inputs	Retrieve sets of documents in RAG (Li et al. 2024)
		Select prompts that control risk (Zollo et al. 2024)
Change other algorithm settings	—	Accelerate generation by early exit (Schuster et al. 2021, 2022, Jazbec et al. 2024)
		Reduce ambiguity by seeking additional input (Ren et al. 2023, 2024)
		Control a "size" component of the sampling mechanism (Ravfogel et al. 2023, Deutschmann et al. 2024, Ulmer et al. 2024)
		Switch between models when risk score is high (Overman & Bayati 2025)

actions that are representative of the distribution, and we assume that they have not been used for model training.

Then, we aim to construct an estimated threshold $\hat{\tau} = \hat{\tau}(D_n)$ using the calibration dataset D_n such that the abstention probability[5] is controlled at a user-specified level $\alpha > 0$, i.e., $\Pr_{X\sim D,\, Y\sim\hat{p}(\cdot|X),D_n}\big(\ell(X,Y) > \hat{\tau}(D_n)\big) \leqslant \alpha$.

The key observation is the following: suppose we generate responses $Y_i \sim \hat{p}(\cdot|X_i)$ for

Calibration dataset: Given a trained GenAI model, a separate dataset used to endow the model with various statistical properties.

[5]Notice that this probability now also includes the randomness over D_n.

Exchangeability: Informally, the property that a sequence of random variables is equally likely to be presented in any order.

each of our inputs $i = 1, \dots, n$ from the calibration dataset. Then the values $\ell_i := \ell(X_i, Y_i)$, $i = 1, \dots, n$ are i.i.d. random variables with the same distribution as the test loss $\ell(X, Y)$ where $X \sim D$ is a test data point and $Y \sim \hat{p}(\cdot|X)$ is a corresponding outcome sampled from the generative model. Of course, the distribution of these loss values is in general still unknown, because it depends on the unknown target distribution D.

Exchangeability. However, since the ℓ_i are i.i.d., conditional on the set (or multiset) of their values $S_{n+1} = \{\ell_1, \dots, \ell_n, \ell(X, Y)\}$, their ordering is uniform given S_{n+1}. This corresponds to exchangeability, and it is their only property used here.

Therefore, assuming for simplicity of exposition that there are no ties,[6] the rank of $\ell(X, Y)$ among $\ell_1, \dots, \ell_n, \ell(X, Y)$ is distributed uniformly over $\{1, \dots, n + 1\}$, conditional on S_{n+1}. Now, for any $\beta \in [0, 1]$, let Q_β be the β-th quantile of $\{\ell_1, \dots, \ell_n\}$, namely $Q_\beta = \inf\{t : \#\{i : \ell_i \leqslant t\} \geqslant \beta n\}$. We have that $\ell(X, Y) \geqslant Q_{(1-\alpha)(1+1/n)}$ if and only if the rank of $\ell(X, Y)$ among $\{\ell_1, \dots, \ell_n, \ell(X, Y)\}$ is at most $\lfloor \alpha(n + 1) \rfloor$.

Consequences of exchangeability. By exchangeability, this occurs with probability at most $\lfloor \alpha(n + 1) \rfloor / (n + 1) \leqslant \alpha$, conditional on S_{n+1}. Hence, if we choose $\hat{\tau}(D_n) := Q_{(1-\alpha)(1+1/n)}$, we find $\Pr\left(\ell(X, Y) \geqslant \hat{\tau}(D_n) \middle| S_{n+1}\right) \leqslant \alpha$, for any S_{n+1}. Since this holds for any set S_{n+1}, it also holds unconditionally, i.e., $\Pr\left(\ell(X, Y) \geqslant \hat{\tau}(D_n)\right) \leqslant \alpha$, as desired.

The above argument explains how, by choosing a threshold for abstention equal to a particular quantile of the calibration losses, we can control the abstention rate. All that is needed is that the test loss is exchangeable with the calibration losses.

2.1.2. An overview of applications and techniques. The above discussion is quite representative of a variety of methods designed to improve the behavior of generative AI models. Some of the common elements include: (1) introduction of a loss function; (2) introduction of a small number of tunable hyperparameters; (3) formulation of a desired goal in terms of expectations of the losses and probabilistic properties, and (4) using distribution-free or only weakly distributionally dependent probabilistic tools—such as the distribution of order statistics or concentration inequalities—to ensure the desired goal. We refer to the works cited in Table 2 for details.

For instance, one approach proposes to delete claims from the output Y of a large language model until correctness is reached (Khakhar et al. 2023, Mohri & Hashimoto 2024). This approach defines a deletion operator Δ, typically implemented by another large language model, and a sequence $Y^{(k)} = \Delta(Y^{(k-1)})$, $k \geqslant 1$, starting with $Y^{(0)} = Y$. The loss function ℓ is defined based on whether $Y^{(k)}$ has any claim that contradicts a ground truth answer y^* for the prompt x. This is also evaluated by another large language model. The tunable hyperparameter is the number k of deletions; and the goal is to ensure correctness with probability at least $1 - \alpha$. Then the required number of deletions can be determined based on a calibration dataset, similarly to above.

Many of the methods discussed above rely on some form of non-parametric statistics, distribution-free predictive inference, conformal prediction, and variants. The idea of distribution-free prediction sets dates back at least to the pioneering works of Wilks (1941), Wald (1943), etc. Distribution-free inference has been extensively studied in recent works (see, e.g., Saunders et al. 1999, Vovk et al. 1999, Papadopoulos et al. 2002, Vovk et al. 2005, Vovk 2012, Lei et al. 2013, Lei & Wasserman 2014, Lei et al. 2018, Guan 2023,

[6] The extension to the case of ties is not hard, but it can require some care; see for instance Vovk et al. (2005), Angelopoulos et al. (2023).

Table 3 Types and examples of uncertainty quantification methods for generative AI.

Approach	Type	Examples
Defining Uncertainty	Epistemic & Aleatoric Unc.	Define and estimate epistemic and aleatoric uncertainty through input clarification ensembling (Hou et al. 2024)
	Semantic Uncertainty	Cluster outputs to capture semantic uncertainty (Kuhn et al. 2023)
		Soft-cluster outputs with partially overlapping meaning from black-box models (Lin et al. 2024)
	Other measures	Estimate pseudo-entropy of a prompting-induced sampling distribution (Abbasi Yadkori et al. 2024)
		Approximate Bayesian posterior uncertainty by updating model (Yang et al. 2024, Wang et al. 2024)
Calibration		Re-calibrate probabilities in multiple choice/classification problems (Jiang et al. 2021)
		Calibrate uncertainty to predict performance (Huang et al. 2024, Liu et al. 2024)

Romano et al. 2020, Dobriban & Yu 2025, etc). Predictive inference methods have been developed under various assumptions (see, e.g., Geisser 2017, Bates et al. 2021, Park et al. 2022b,a, Sesia et al. 2023, Qiu et al. 2023, Li et al. 2022, Kaur et al. 2022, Si et al. 2024, Lee et al. 2024). Overviews of the field are provided by Vovk et al. (2005), Shafer & Vovk (2008), and Angelopoulos et al. (2023).

2.2. Diagnostics and Uncertainty Quantification

When AI systems encounter problems, one should of course aim to improve the behavior of the AI system. A crucial step toward this is to precisely diagnose the problem. A variety of approaches exist for this task, ranging from constructing unit tests to fine-tuning the model. There are also a number of methods based on computing certain specific diagnostic scores (e.g., Farquhar et al. 2024, Yadkori et al. 2024, Lin et al. 2024, etc). Such diagnostics are already used in many of the methods discussed in Section 2.1 to change or improve the generative model; for instance, if a safety score for the input is low, the model can refrain from generating an output.

In this section, we are specifically interested in diagnostics that aim to quantify uncertainty, as these have close connections to probability and statistics. There are a variety of interpretations of uncertainty quantification, see e.g., Baan et al. (2023), Shorinwa et al. (2024), Liu et al. (2025), Abbasli et al. (2025), Xia et al. (2025), He et al. (2025), Campos et al. (2024), Trivedi & Nord (2025). Due to space reasons, here we can only discuss a few specific approaches, see Table 3.

2.2.1. Epistemic and aleatoric uncertainty. We start by introducing the notions of epistemic and aleatoric uncertainty. To set the stage, we observe that given an input x, the output y is not always uniquely determined. For instance, the query $x =$ "Write a paragraph about an economist" has ambiguity, since it does not specify a particular economist. This is sometimes referred to as *epistemic uncertainty* (Der Kiureghian & Ditlevsen 2009, Hüllermeier & Waegeman 2021). It can be reduced by collecting more information. In particular, the AI

system could query "Which economist?", to which the answer, e.g., "Adam Smith", could greatly reduce the uncertainty of the answer to be generated.

In practice, there are usually many such sources of epistemic uncertainty for any given query. For instance, even after knowing which economist to consider, we still do not know the desired number of sentences, the target audience (children, general public, scientists, or some other group), etc. Some of these might be more important than others to the user, but either way they contribute to the uncertainty of the possible answers.

Epistemic and aleatoric uncertainty: Roughly speaking, uncertainty due to lack of knowledge, and due to random chance, respectively. Can be hard to define precisely.

We can contrast epistemic uncertainty with *aleatoric uncertainty*. For instance, in the query "Choose between A and B uniformly at random.", all information is perfectly well specified (so the epistemic uncertainty vanishes), yet there is still irreducible random uncertainty in the desired output, which is sometimes referred to as *aleatoric uncertainty* (Der Kiureghian & Ditlevsen 2009).

While multiple definitions exist (see, e.g., Schweighofer et al. (2025)), including approaches tailored to estimating them in generative AI models (see, e.g., Hou et al. (2024)), in many cases the definition of—say—aleatoric uncertainty reduces to specifying what we choose not to predict, rather than to something intrinsically fixed.

2.2.2. Uncertainty in model generations. While the discussion in Section 2.2.1 refers to uncertainty in ideal "ground truth" answers, in practice we need to take into account that we only have an empirical model $\hat{p}$, not the ground truth; and need to handle the uncertainty in the answers generated by $\hat{p}$. Equivalently, we should quantify to what extent the model is certain. There have been several approaches aimed to extract this form of uncertainty from generative AI models.

For language models, a special ability is that they might potentially be able to express uncertainty in words. However, this capability is not guaranteed to work well by default, and special fine-tuning techniques have been developed to induce this behavior in certain special cases (Lin et al. 2022).

Uncertainty and confidence scores: Numerical values computed based on the input, output, or other characteristics of the GenAI model, aiming to capture the level of uncertainty.

An approach that applies more generally to all generative models, regardless of their modality, is to compute some *uncertainty or confidence score*[7] based on the input x, the output y, and/or the model $\hat{p}$ (e.g., Farquhar et al. 2024, Yadkori et al. 2024, Lin et al. 2024, etc). For instance, one can consider the probability $\hat{p}(y|x)$, Which reflects how likely the generated output is according to the model; and thus can be viewed as a very basic form of a confidence score. Alternatively, for generations whose length can vary, such as for standard language models, one can consider a length-normalized version $\hat{p}(y|x)^{1/|y|}$, where $|y|$ is the length of y; aiming to correct for the effect that longer generations tend to have smaller probabilities.

However, it is not always straightforward to use and interpret such scores. There are multiple key challenges:

Challenge 1: Inability to recover "true" probabilities and lack of calibration. The probabilities $\hat{p}(y|x)$ represent only the model's internal beliefs about the likelihood of output y given input x; by default, they do not correspond to any notion of "true" probabilities. Because the input and output spaces are extremely high-dimensional, the probabilities produced by a generative AI model should not be expected to be consistent

[7] Note that Lin et al. (2024) define uncertainty scores to refer to the entire distribution $\hat{p}(\cdot|x)$ and confidence scores to refer to a specific input-output pair (x, y). Due to lack of space, we will not make this distinction.

for any "ground truth". However, we might hope to achieve weaker forms of correctness.

One such relaxation is *calibration*, which is a general property associated with probabilistic forecasts (Gneiting & Katzfuss 2014) that only asks for a restricted set of probabilities to reflect real probabilities. For instance, for a calibrated weather forecaster, if we predict "50% chance of rain tomorrow", then over all such days, we expect that it rains half the time (Lichtenstein et al. 1977, Van Calster & Vickers 2015, Van Calster et al. 2019). There are a variety of notions of calibration relevant to GenAI, and empirical work has found that model calibration is not guaranteed by default. Instead, it can depend strongly on model training, model size, etc; see e.g., Kadavath et al. (2022), Achiam et al. (2023).

Calibration: The property of a predicted probability that it reflects the empirical frequencies of a specific class of events.

A direct way to apply calibration to answers generated by an LM $\hat{p}$ is to construct an additional probability predictor $\hat{q}$ for the claim "The chance that my answer is right is $\hat{q}$." Such a probability predictor can be obtained via re-calibration on separate calibration data (Mincer & Zarnowitz 1969, Guo et al. 2017), but it might require a lot of calibration data.

If less calibration data is available, one may still be able to approximately satisfy a weaker form of calibration, e.g., that the average accuracy increases with the predicted probability of success, a behavior termed *rank-calibration* (Huang et al. 2024).

Rank-calibration: The property that the average accuracy increases with the predicted probability of success.

Challenge 2: Semantic multiplicity. Another key challenge is that there are often many equivalent answers. For instance, in text generation, answers such as "15 pages" and "fifteen pages" are semantically equivalent. We usually want to pool them together when determining the model's confidence.

An approach to this problem—termed *semantic uncertainty*—was proposed by Kuhn et al. (2023), who suggested generating multiple outputs $Y_1, \ldots, Y_K \sim \hat{p}(\cdot|x)$ i.i.d., clustering them based on their semantics (via another LLM), and then estimating uncertainty based on the resulting distribution induced over the clusters.

Semantic uncertainty: Uncertainty after semantic equivalences have been accounted for.

2.3. AI Evaluation

Evaluating generative AI models is important in order to properly understand the capabilities that these models possess. However, model evaluation can be surprisingly challenging, and in particular, it can bring novel challenges compared to the evaluation of more standard machine learning models, see e.g., Burden et al. (2025) for a review.

A typical current workflow for evaluating a GenAI model—in particular, a large language model—is as follows. Suppose we want to measure reasoning ability in mathematical problems. To evaluate this ability, we collect test data consisting of such problems. Then we evaluate the accuracy of the model on these problems and report the results.

This simple workflow is mired with a number of challenges. First of all, the specific data required for evaluation (say mathematical problems), can be quite complex, and finding genuinely new test problems that the model has not seen during training is hard. Indeed, information leakage from standard public test datasets into the model training sets is a genuine concern (see e.g., Matton et al. 2024, etc). This leads to potential biases in model performance evaluation, where the models score higher because they have already seen the problems during training.

A potential approach is to have private test data sets that are not released to the public. Another potential approach is to use dynamically generated AI evaluation environments, such as based on debates (Moniri et al. 2025). Due to these reasons, and as collecting large, high quality, and genuinely new evaluation datasets can be expensive, high quality

test datasets sometimes have relatively small sample sizes.[8]

Second, checking correctness can be non-trivial and ambiguous outside of simple problems with clear, well-defined answers. For instance, in a mathematical problem, it can be straightforward to evaluate the correctness of a numerical answer, but it can be much harder to evaluate the correctness of a reasoning process. For this reason, often heuristics such as other LLMs are used for checking answers, which in turn raises questions about reliability.

Third, evaluating the largest models can be expensive, which further poses a limit on the sample sizes that we can collect for evaluation depending on the available budget.

Due to these reasons, evaluation can involve dealing with small sample sizes and various biases. Thus, statistical methods and thinking can be valuable for reliable and efficient evaluation.

2.3.1. A basic statistical formulation of model evaluation. We consider a basic setting of model evaluation, in which we have some inputs x for which we wish to evaluate the performance of a GenAI model. For mathematical reasoning, this could correspond to the problem statement, and may also include instructions to the model. The problem has a ground truth answer y^*, which can be an entire solution/reasoning path or just the final result. Then, we sample a candidate answer $Y \sim \hat{p}(\cdot|x)$. Again, this may include intermediate steps, and a final answer is extracted at the end.

As in Section 2.1.1, the quality of the answer is evaluated via a loss function ℓ, such that $\ell(x, y^*, y)$ measures the (negative) utility of answer y for input x with ground truth y^*. In some cases, designing loss functions is straightforward. For instance, for an integer answer y^*, we may use the binary loss $I(y \neq y^*)$. However, for more elaborate problems, designing a loss function can be non-trivial. For instance, for a reasoning problem, we want to make sure that all valid and concise reasoning paths receive low loss, not just the reference path.

Tasks in AI evaluation. Given these components, there are several possible tasks of interest. For a distribution D of inputs, we may want to estimate or perform statistical inference (confidence intervals, tests) for the task performance $\theta = \mathbb{E}_{(X,Y^*)\sim D,\, Y\sim\hat{p}(\cdot|X)}\, \ell(X, Y^*, Y)$.

Given a dataset $D_n = \{(X_i, Y_i^*) : i = 1, \ldots, n\}$ of question–answer pairs sampled i.i.d. from D, we can generate outputs $Y_i \sim \hat{p}(\cdot \mid X_i)$ independently, and compute the loss values $\ell_i = \ell(X_i, Y_i^*, Y_i)$, $i = 1, \ldots, n$; as in Section 2.1.1.

Then, these loss values $\ell_1, \ldots, \ell_n$ are sampled i.i.d. from a distribution whose population mean is the unknown true task performance θ. Thus, this problem becomes that of inference for a population mean, for which many statistical methods exist (Casella & Berger 2024, Lehmann & Romano 2005).

Notably, in many important examples we are interested in (A) binary losses, leading to inference for a Binomial parameter; (B) or bounded losses (for which concentration inequalities such as Hoeffding's inequality can be used); (C) or given a large sample size (so that an asymptotic normal approximation works well).

[8] One of the most reliable approaches at the moment is to use new test datasets that are initially designed for humans; for instance, it is common to test mathematical reasoning on the problems of mathematical competitions, such as the International Mathematical Olympiad (IMO), as soon as the new problems are released. The thought process is that those problems have been filtered by the problem selection committee to be new for humans, and thus this reduces the chances of contamination from the training set. However, this again leads to relatively small sample sizes, for instance, the International Mathematical Olympiad has six problems every year.

AI Evaluation and Statistical Inference. AI evaluation with limited data has a very close link to statistical inference.

Table 4 Types and examples of statistical evaluations of generative AI models.

Technique	Type	Examples
Inference on Performance	Confidence intervals	Review of standard large-sample methods (Miller 2024)
		Construct CIs with improved finite-sample coverage on model accuracy under i.i.d. and clustered data settings (Bowyer et al. 2025)
		Develop asymptotically valid CIs for comparing the KL divergence to the true distribution of two models (Gao & Sun 2025)
		Construct uniform upper bound on the CDF of a performance metric (Vincent et al. 2024)
		Construct confidence interval for probability of biased answers on counterfactual prompts (Chaudhary et al. 2025)
	Hypothesis testing	Test hypothesis about which policy achieves higher reward, choosing number of trials adaptively (Snyder et al. 2025)
Small-data Evaluation	Small-sample performance	Estimate model accuracy on multiple questions and models leveraging item response theory (Polo et al. 2024)
	Synthetic + human labels	Combine synthetic and human labels for unbiased performance estimates and CIs (Boyeau et al. 2024, Fisch et al. 2024, Oosterhuis et al. 2024)
		Rank models with hybrid label sets (Chatzi et al. 2024)
Multi-task Evaluation	Active testing	Actively sample and evaluate in multitask settings (Anwar et al. 2025)

An important observation here is that AI evaluation with limited data has a very close link to statistical inference. Beyond this core setting, there are a variety of important additional scenarios. For instance, we may be interested in comparing the performance of two models $\hat{p}_1, \hat{p}_2$. If we can query both models on the same inputs $X_i, i = 1, \ldots, n$, this can be formulated as statistical inference for the parameter $\Delta = \mathbb{E}_{(X,Y^*)\sim D,\, Y_1\sim\hat{p}_1(\cdot|X),\, Y_2\sim\hat{p}_2(\cdot|X)}\left[\ell(X, Y^*, Y_1) - \ell(X, Y^*, Y_2)\right]$. Considerations and methods similar to the ones above apply.

Standard methods for the above two problems have been reviewed in Miller (2024); where other considerations, such as power analysis and clustered data arising from repeated generations for the same input, are also considered. However, this work focuses on a signal-plus-noise model for the observed losses, which may need to be relaxed.

2.3.2. Additional methods. There are a variety of works addressing other settings in AI evaluation, see Table 4 for examples. A few of them are discussed in more detail below. However, a comprehensive and unified statistical methodology that addresses most of the common evaluation problems with a unified terminology and set of methods remains to be developed.

1. Bowyer et al. (2025) study methods for producing confidence intervals on model performance, focusing on inference for Bernoulli parameters of model accuracy. They include single-model performance (for i.i.d. and clustered data), two-model comparison (both independent data and paired samples). They conclude that the most straightforward asymptotic normality-based confidence intervals can be inaccurate for small datasets at most $n = 100$ datapoints. They argue for using Bayesian credible intervals, which they argue have adequate frequentist coverage when one can specify appropriate prior distributions.
2. Gao & Sun (2025) develop methods for comparing the Kullback-Leibler (KL) divergence of two generative methods for which the probabilities $\hat{p}$ can be computed. They show how to construct an asymptotically valid confidence interval for the difference of KL divergences.
3. Polo et al. (2024) develop methods for estimating accuracy using a small number of datapoints, leveraging methods item response theory. They consider settings where the performance of a model $\hat{p}$ on an example x is captured by (unknown) model-specific and example-specific latent variables $\theta_{\hat{p}}$ and γ_x. For instance, we may model the probability $Q(\hat{p}, x)$ of a correct answer by $\hat{p}$ on the input x via a logistic model $\text{logit}(Q(\hat{p}, x)) = \theta_{\hat{p}}^\top \gamma_x + \beta_x$. Then, these parameters are estimated on a small dataset, and the correctness probability predictions they induced are used on new test examples to extrapolate correctness; leading to significant savings in the number of test examples needed. See also Zhou et al. (2025), Gignac & Ilić (2025), Kipnis et al. (2025) for other uses of item response theory and related methods.
4. Boyeau et al. (2024), Fisch et al. (2024), Oosterhuis et al. (2024) develop methods to use a large set of synthetically generated labels along with a small set of human labels for unbiased model evaluation, including confidence intervals for model performance. See Chatzi et al. (2024) for ranking.
5. Anwar et al. (2025) develop methods for multi-task evaluation of (robot) policies with active testing, where they pool information on performance of several policies across several tasks, prioritizing tasks with high information gain leveraging Bayesian active learning (Houlsby et al. 2011).
6. Chowdhury et al. (2025) develop a variational lower bound on the expected loss incurred by a language model, and use it to find prompts that elicit problematic behavior. Concretely, let ℓ be a loss, $\hat{p}$ be the target LLM. Our goal is to find prompts x to make $\ell(x, Y)$ large when $Y \sim \hat{p}(\cdot|x)$. Formally, we aim to make $S(x) = \log \mathbb{E}_{Y\sim\hat{p}(\cdot|x)} \exp(S(x, Y))$ large. To find such x, we rely on an auxiliary LLM $\hat{q}$ for which the loss tends to be larger for all x. Due to Jensen's inequality, we have the variational lower bound $S(x) \geqslant \mathbb{E}_{Y\sim\hat{q}(\cdot|x)}[\log \hat{q}(Y|x) - \log \hat{p}(Y|x) + S(x, Y)]$. This lower bound is estimated by sampling $Y \sim \hat{q}(\cdot|x)$ repeatedly, which can be more efficient than estimating $S(x)$ directly.

2.4. Interventions and Experiment Design

Interventions refer to systematically modifying or perturbing the inputs of an AI system, to gain understanding or control of its behavior. This approach has become one of the most widely used and most powerful tools in a variety of AI research directions, including interpretability, robustness, and fairness (e.g., Zhao et al. 2018, Rudinger et al. 2018, Belinkov 2022, Kotek et al. 2023, etc). The ideas underlying interventions are closely connected to

statistical causality and experiment design; see also Pearl (2001), Soumm (2024).

2.4.1. Basic setting for interventions. In a basic setting for interventions, we have a generative model $\hat{p}$ to which we can provide an input x (e.g., a query to an LLM). In contrast to the other parts covered in this review, for interventions, it is often the case that the intermediate computations are of crucial importance. The reason is that, empirically, certain internal mechanisms can sometimes be responsible for specific behaviors, such as biases and harmful outputs (see e.g., Mikolov et al. 2013b, Turner et al. 2023, Rimsky et al. 2024, Zou et al. 2023, etc).

Therefore, in this section, we will sometimes also assume that we have access to intermediate computations $e(x)$ (e.g., representations, intermediate/chain of thought tokens) of the model. Most often, vector-valued intermediates $e(x)$ are considered. Finally, we also consider the output layer $o(x)$ of the model (e.g., last-layer predicted probabilities or log-probabilities), as well as the final model output y. These quantities can be either deterministic or random.

Interventions: Perturbing components of the model (input or intermediate computations) to achieve a desired effect, such as reducing biases.

We want to understand or control a certain components of the behavior of the AI system. We consider components measured through the input, intermediate computation, or output. For instance, which components of an LLM (activations, neurons) contribute to gender bias? How can we intervene to reduce such biases? How does an LLM behave internally when it is non-truthful, and does this differ from truthful behavior? Are there specific components that are activated when the LLM generates harmful output, and can we intervene to suppress this behavior?

To do this, we find a way to intervene by perturbing the input x to induce the condition of interest. For example, to understand how harmfulness is propagated, we can change part of a harmful input to a harmless concept: e.g., $x =$ "how to build a bomb?" $\to x' =$ "how to build a chair?". We can also intervene on an intermediate computation in the AI system. Then, we track the change in either the intermediate stage or the final output, depending on what we are interested in.

Example 2.1. Contextual concept vectors *measure the difference in embeddings that a change in a concept leads to, in the form* $C_{x\to x'} := e(x') - e(x)$*, where* x *is an input and* x' *is the corresponding input with the concept changed, e.g., for the concept of gender,* x*="king",* x'*="queen";* x*="actor",* x'*="actress", etc. Early work investigating related questions dates back at least to Mikolov et al. (2013a,b), Pennington et al. (2014) for word embeddings, and more recently has studied human biases (Bolukbasi et al. 2016), developed steering vectors (Turner et al. 2023, Rimsky et al. 2024) and introduced representation engineering (Zou et al. 2023).*

Contextual concept vector: The effect of changing a concept in the input on some vector in the intermediate computation of the genAI model.

To obtain a more stable and generalizable picture about the effect of the intervention, it is common to consider a distribution D of interest, and the associated mean $\mathbb{E}_{X\sim D}[C_{X\to X'}]$ or top principal component of the covariance matrix $\mathrm{Cov}_{X\sim D}(C_{X\to X'})$ (Zou et al. 2023). These are typically estimated using the standard plug-in estimators. Let $\hat{c}$ be such an estimated concept vector.

Steering vector: A quantity that used in—typically added to—the intermediate representations of a model to make a desired behavior more likely.

Steering vectors. These estimates can be used as steering vectors (Turner et al. 2023, Rimsky et al. 2024) to make certain behaviors more likely. A common approach is to take any input x, compute its intermediate representation $e(x)$, and add a scaled version $\lambda \cdot \hat{c}$ for some $\lambda > 0$ to obtain a new intermediate representation $e' = e(x) + \lambda \cdot \hat{c}$. The computation then continues identically to obtain the final output. Here λ is a hyperparameter that

requires careful tuning. This operation approximates a shift of the representation of the original input towards the representation of a changed input $e(x')$. For instance, in the above example, the goal would to approximately remove the harmful concept. Empirically, it has been observed that the resulting final output can sometimes indeed correspond to the desired concept change (Turner et al. 2023, Rimsky et al. 2024); which however comes with caveats (Tan et al. 2024).

Assessing biases. Analogously, to assess biases[9] (e.g., gender bias), one can choose a representative output variable $o(x)$, such as the probability of a gendered word, and then repeat the above analysis. For instance, to study gender bias, Kotek et al. (2023) intervene to modify gender in an input such as $x =$ "The doctor called the nurse because he was late. Who was late?" They change this to $x' =$ "The doctor called the nurse because she was late. Who was late?"

Then, they evaluate its effect on an output o which they choose as a measure of the probability of the output "nurse". Specifically, they compute $O_{x \to x'} = o(x') - o(x)$, which measures how much more likely the model is to output "nurse" solely due to the change "he" $\to$ "she", and thus it can be interpreted as a form of gender bias. Kotek et al. (2023) also design an improved version that also permutes "doctor" and "nurse", aiming to control for the effect of syntactic position.

Probing: Training models based on intermediate features to see if they contain information about a specific concept.

Probing. A related concept is that of probing (see e.g., Alain & Bengio 2016, Belinkov 2022, etc). To understand if a feature e captures a concept $x \mapsto x'$, in probing one trains a classifier of datapoints $X \sim D$ versus their transformed counterparts X', using a simple function—often linear—of the features e. If this classifier has a high accuracy, then it is concluded that the feature captures the concept. This approach has been leveraged in generative AI, e.g., to understand where models store spatial information about the input (Gurnee & Tegmark 2024).

See Table 5 for some examples of related methods. A few examples are discussed below:

1. There is work aiming to identify sub-networks (not just representations) responsible for specific tasks, by pruning to the networks and checking if they can still perform the computation (Nanda et al. 2023). Further, Zhang & Nanda (2024) systematized activation patching methods to localize causal computations in LLMs, providing best practices for intermediate-stage interventions.
2. Greenblatt et al. (2023) used intervention-based prompts to elicit deceptive behavior from an LLM, finding that LMs may internally simulate misaligned objectives while faking alignment.
3. There has been work to design perturbations of standard mathematical datasets to evaluate LLM reasoning robustness (Shi et al. 2023, Mirzadeh et al. 2025).

2.4.2. Causal mediation analysis. Causal mediation analysis (Pearl 2001) is a more advanced technique from statistical causality, which can be used to identify the precise effects of intermediate components of generative AI models (e.g., Vig et al. 2020, etc.). In a basic setting for causal mediation analysis, we consider an input x, and a changed input x', Where we intervene via an intervention that we would like to study, for instance changing the sentiment of a review x from positive to negative.

[9] The term bias is used with a variety of meanings in AI, which are moreover usually different from the standard statistical meaning of bias in estimation. In our example, bias refers to a behavior that is different from a desired one (equal frequency of genders output).

Table 5 Types and examples of interventions and experiment design in generative AI.

Technique	Type	Examples
Understand Behavior via Intervention	Learn bias or association	Learn gender bias in output by modifying input (Bolukbasi et al. 2016, Zhao et al. 2018, Rudinger et al. 2018)
		Identify internal/intermediate component associated with bias or factual association via causal mediation analysis (Vig et al. 2020, Meng et al. 2022, Dai et al. 2022)
		Learn effect of circuits (sub-networks) by pruning to the circuit and observing behavior (Nanda et al. 2023)
		Learn effect of thoughts (intermediate outputs) by modifying them (Bogdan et al. 2025)
	Learn concept	Learn concept or steering vector by inducing concept modifying input (Mikolov et al. 2013a,b, Pennington et al. 2014, Turner et al. 2023, Rimsky et al. 2024, Zou et al. 2023)
	Evaluate performance	Perform ablation study: change algorithm setting and test behavior
		Design perturbed dataset to evaluate LLM reasoning robustness (Wu et al. 2024, Shi et al. 2023, Mirzadeh et al. 2025)
	Evaluate alignment	Design prompt eliciting behavior that would modify AI system and observe behavior (Greenblatt et al. 2023)
Understand Behavior via Probing		Identify neurons associated with sentiment (Radford et al. 2017) or neurons that represent world state (Li et al. 2023a)
		Identify sparse linear combinations of neurons that represent features (Gurnee et al. 2023)
Change Behavior via Intervention		Add gradient of concept classifier (Dathathri et al. 2020) or steering vector (Subramani et al. 2022, Turner et al. 2023, Zou et al. 2023, Li et al. 2023b) to elicit behavior
		Patch activations from one input into the activations of another input (Meng et al. 2022, Zhang & Nanda 2024)

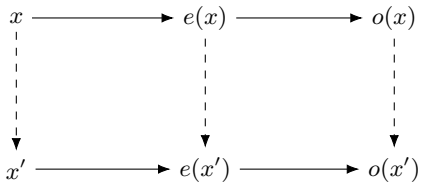

Figure 2

Diagram to represent computational flow and interventions, for use with causal mediation analysis Solid arrows denote standard computational flows; dashed arrows denote interventions or their effects.

We aim to study a generative model of interest. We consider an intermediate representation/activation e whose effect we aim to study; in causal mediation analysis e is known as the *mediator*. The final output representation o of the generative model depends on the intermediate representation e, as well as on other model components, which together we denote by $e^{\perp}$. Algebraically, we write the output representation in the functional form

$o(x) = g(e(x), e^{\perp}(x))$ for all $x \in \mathcal{X}$, for some set of computations denoted by g. See Figure 2 for a diagram representing this setting.

Then, $o(x') - o(x)$ represents the overall effect of the intervention $x \to x'$. Typically, we are interested not just in the particular query x, but rather about the average behavior over a distribution of interest. The total average effect of $x \to x'$ is $\mathbb{E}\left[o(X') - o(X)\right]$. This can be decomposed into a the sum of natural direct and indirect effects.

Natural Direct Effect: The effect of an input on an output that happens through pathways other than the mediator under study.

Natural direct effect. The natural direct effect of $x \to x'$ on o is the effect that happens through pathways other than the mediator e. This expression keeps e fixed:

$$\mathbb{E}\left[o\left(e(X), e^{\perp}(X')\right) - o(X)\right] = \mathbb{E}\left[o\left(e(X), e^{\perp}(X')\right) - o\left(e(X), e^{\perp}(X)\right)\right]$$

If the direct effect is small, this can be interpreted as the mediator e capturing most of the effect of x' on o. When the direct effect is small, we can view the mediator as having an important role in enacting the effect $x \mapsto x'$, making it a promising target for interventions if we aim to mitigate this effect.

Natural Indirect Effect: The effect of an input on an output that happens through the mediator under study.

Natural indirect effect. To complement this, the natural indirect effect of $x \to x'$ on o captures the remaining part of the total effect, which goes through the mediator $e(x) \to e(x')$:

$$\mathbb{E}\left[o(X') - o\left(e(X), e^{\perp}(X')\right)\right] = \mathbb{E}\left[o\left(e(X'), e^{\perp}(X')\right) - o\left(e(X), e^{\perp}(X')\right)\right].$$

This decomposition of effects into direct and indirect ones has been used, among others, to identify components responsible for gender bias (Vig et al. 2020) as well as other factual associations (Meng et al. 2022, Dai et al. 2022) in LLMs. In some cases, x' can correspond to "adding noise" to tokens that contain specific information, e.g., "The Space Needle is in" →"**[i.i.d. Gaussian activations]** is in"; by acting at the levels of token embeddings of x. This allows capturing the effect of deleting information from the input. However, fully rigorous and well-justified methods for interventions on the identified mediators have not yet been developed.

3. Discussion

We have presented overviews of some applications of statistical ideas to generative AI, focusing on topics such as improving and changing the behavior of GenAI models, diagnostics and uncertainty quantification, evaluation, as well as interventions and experiment design. These leverage ideas from classical statistical inference, distribution-free predictive inference, forecasting and calibration, as well as causality.

At the moment, generative AI models are exceedingly complex, and are usually best viewed as black boxes. To ensure usefulness in GenAI, one needs to develop methods that are light on assumptions. Moreover, in order to to maximize impact, the methods need to be illustrated on current GenAI models, which requires both a familiarity with ongoing developments in AI, and adequately large computational resources. For statisticians, collaboration with AI researchers can help ensure that these requirements are met.

SUMMARY POINTS

1. **GenAI lacks guarantees.** Generative AI models are stochastic black boxes: as probability distributions over large semantic spaces (text, images) from which we

can sample. While showing promising performance in a variety of areas, they do not have any guarantees about correctness, safety, etc., by default.

2. **Statistical methods for GenAI need to handle black box models.** In order to be applicable to black-box generative AI models, statistical methods need to be light on assumptions and able to handle structured semantic input and output spaces.
3. **The flexibility of statistical "wrappers".** There are a variety of approaches to change the behavior of AI models, both in terms of their inputs and their outputs. Statistical "wrappers" can be used in order to precisely control the performance of these approaches.
4. **Uncertainty quantification must be calibrated and handle semantics.** Quantifying the uncertainty of a GenAI model could be a promising way to make it more reliable; however, the issues of semantic multiplicity and lack of calibration need to be handled.
5. **Evaluation is statistical inference.** AI evaluation, especially with small datasets, presents opportunities for leveraging statistical inference methods.
6. **The power of interventions.** Interventions on generative AI systems, building on ideas from causal inference, have the potential to identify components responsible for specific capabilities and to induce desired behaviors.
7. **The promise of dataset and experiment design.** Calibration, evaluation, and intervention all hinge on carefully collected, held-out calibration sets and targeted perturbations, which offer opportunities for statistical thinking.

FUTURE ISSUES

1. Statistical methods aimed at improving AI models need to be developed by taking into account the black-box nature of AI, where often only the inputs and outputs of the models are available, and the intermediate computations are unknown.
2. A comprehensive statistical framework for the evaluation of generative AI systems is yet to be developed.
3. Well-justified methods for interventions on mediators identified in generative AI models remain to be introduced.

DISCLOSURE STATEMENT

The author is not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

This work was supported in part by the US NSF, ARO, AFOSR, ONR, the Simons Foundation and the Sloan Foundation. The opinions expressed in this document are solely those of the author and do not represent the views of the above institutions.

LITERATURE CITED

Abbasi Yadkori Y, Kuzborskij I, György A, Szepesvari C. 2024. To believe or not to believe your llm: Iterative prompting for estimating epistemic uncertainty. *Advances in Neural Information Processing Systems* 37:58077–58117

Abbasli T, Toyoda K, Wang Y, Witt L, Ali MA, et al. 2025. Comparing uncertainty measurement and mitigation methods for large language models: A systematic review. *arXiv preprint arXiv:2504.18346*

Achiam J, Adler S, Agarwal S, Ahmad L, Akkaya I, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*

Alain G, Bengio Y. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*

Angelopoulos AN, Bates S, et al. 2023. Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning* 16(4):494–591

Anwar A, Gupta R, Merchant Z, Ghosh S, Neiswanger W, Thomason J. 2025. Efficient evaluation of multi-task robot policies with active experiment selection. *arXiv preprint arXiv:2502.09829*

Baan J, Daheim N, Ilia E, Ulmer D, Li HS, et al. 2023. Uncertainty in natural language generation: From theory to applications. *arXiv preprint arXiv:2307.15703*

Band N, Li X, Ma T, Hashimoto T. 2024. Linguistic calibration of long-form generations, In *Proceedings of the 41st International Conference on Machine Learning*, pp. 2732–2778

Bashari M, Lotan RM, Lee Y, Dobriban E, Romano Y. 2025. Synthetic-powered predictive inference. *arXiv preprint arXiv:2505.13432*

Bates S, Angelopoulos A, Lei L, Malik J, Jordan M. 2021. Distribution-free, risk-controlling prediction sets. *Journal of the ACM (JACM)* 68(6):1–34

Belinkov Y. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics* 48(1):207–219

Bogdan PC, Macar U, Nanda N, Conmy A. 2025. Thought anchors: Which llm reasoning steps matter? *arXiv preprint arXiv:2506.19143*

Bolukbasi T, Chang KW, Zou JY, Saligrama V, Kalai AT. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings, In *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29

Bowyer S, Aitchison L, Ivanova DR. 2025. Position: Don't use the clt in llm evals with fewer than a few hundred datapoints. *ICML (Spotlight Position Paper)*

Boyeau P, Angelopoulos AN, Yosef N, Malik J, Jordan MI. 2024. Autoeval done right: Using synthetic data for model evaluation. *arXiv preprint arXiv:2403.07008*

Burden J, Tešić M, Pacchiardi L, Hernández-Orallo J. 2025. Paradigms of ai evaluation: Mapping goals, methodologies and culture. *arXiv preprint arXiv:2502.15620*

Campos M, Farinhas A, Zerva C, Figueiredo MA, Martins AF. 2024. Conformal prediction for natural language processing: A survey. *Transactions of the Association for Computational Linguistics* 12:1497–1516

Casella G, Berger R. 2024. Statistical inference. Chapman and Hall/CRC

Chan KHR, Ge Y, Dobriban E, Hassani H, Vidal R. 2025. Conformal information pursuit for interactively guiding large language models. *arXiv preprint arXiv:2507.03279*

Chatzi I, Straitouri E, Thejaswi S, Rodriguez M. 2024. Prediction-powered ranking of large language models. *Advances in Neural Information Processing Systems* 37:113096–113133

Chaudhary I, Hu Q, Kumar M, Ziyadi M, Gupta R, Singh G. 2025. Certifying counterfactual bias in LLMs, In *The Thirteenth International Conference on Learning Representations*

Chen M, Mei S, Fan J, Wang M. 2024. An overview of diffusion models: Applications, guided generation, statistical rates and optimization

Chowdhury N, Schwettmann S, Steinhardt J, Johnson DD. 2025. Surfacing pathological behaviors in language models. `https://transluce.org/pathological-behaviors`

Dai D, Dong L, Hao Y, Sui Z, Chang B, Wei F. 2022. Knowledge neurons in pretrained transformers,

In *Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 8493–8502

Dathathri S, Madotto A, Lan J, Hung J, Frank E, et al. 2020. Plug and play language models: A simple approach to controlled text generation, In *Proc. of the International Conference on Learning Representations (ICLR)*

Davies A, Veličković P, Buesing L, Blackwell S, Zheng D, et al. 2021. Advancing mathematics by guiding human intuition with ai. *Nature* 600(7887):70–74

Der Kiureghian A, Ditlevsen O. 2009. Aleatory or epistemic? does it matter? *Structural safety* 31(2):105–112

Deutschmann N, Alberts M, Martinez MR. 2024. Conformal autoregressive generation: Beam search with coverage guarantees, In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 11775–11783

Dobriban E, Yu M. 2025. Symmpi: predictive inference for data with group symmetries. *Journal of the Royal Statistical Society Series B: Statistical Methodology* :qkaf022

Farquhar S, Kossen J, Kuhn L, Gal Y. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature* 630(8017):625–630

Fisch A, Maynez J, Hofer RA, Dhingra B, Globerson A, Cohen WW. 2024. Stratified prediction-powered inference for hybrid language model evaluation. *arXiv preprint arXiv:2406.04291*

Gao Z, Sun Y. 2025. Statistical inference for generative model comparison

Geisser S. 2017. Predictive inference: an introduction. Chapman and Hall/CRC

Gignac GE, Ilić D. 2025. Psychometrically derived 60-question benchmarks: Substantial efficiencies and the possibility of human-ai comparisons. *Intelligence* 110:101922

Gneiting T, Katzfuss M. 2014. Probabilistic forecasting. *Annual Review of Statistics and Its Application* 1(1):125–151

Greenblatt R, Denison C, Wright B, Roger F, MacDiarmid M, et al. 2023. Alignment faking in large language models. *arXiv preprint arXiv:2412.14093*

Guan L. 2023. Localized conformal prediction: A generalized inference framework for conformal prediction. *Biometrika* 110(1):33–50

Gui Y, Jin Y, Ren Z. 2024. Conformal alignment: Knowing when to trust foundation models with guarantees, In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*

Guo C, Pleiss G, Sun Y, Weinberger KQ. 2017. On calibration of modern neural networks, In *International conference on machine learning*, pp. 1321–1330, PMLR

Gurnee W, Nanda N, Pauly M, Harvey K, Troitskii D, Bertsimas D. 2023. Finding neurons in a haystack: Case studies with sparse probing. *Transactions on Machine Learning Research*

Gurnee W, Tegmark M. 2024. Language models represent space and time, In *The Twelfth International Conference on Learning Representations*

Hayes T, Rao R, Akin H, Sofroniew NJ, Oktay D, et al. 2025. Simulating 500 million years of evolution with a language model. *Science* 387(6736):850–858

He J, Yu L, Li C, Yang R, Chen F, et al. 2025. Survey of Uncertainty Estimation in Large Language Models -Sources, Methods, Applications, and Challenge. Working paper or preprint

Horwitz E, Hoshen Y. 2022. Conffusion: Confidence intervals for diffusion models. *arXiv preprint arXiv:2211.09795*

Hou B, Liu Y, Qian K, Andreas J, Chang S, Zhang Y. 2024. Decomposing uncertainty for large language models through input clarification ensembling, In *Proceedings of the 41st International Conference on Machine Learning*, pp. 19023–19042

Houlsby N, Huszár F, Ghahramani Z, Lengyel M. 2011. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*

Huang X, Li S, Yu M, Sesia M, Hassani H, et al. 2024. Uncertainty in language models: Assessment through rank-calibration, In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, eds. Y Al-Onaizan, M Bansal, YN Chen

Hüllermeier E, Waegeman W. 2021. Aleatoric and epistemic uncertainty in machine learning: An

introduction to concepts and methods. *Machine learning* 110(3):457–506

Jazbec M, Timans A, Hadži Veljković T, Sakmann K, Zhang D, et al. 2024. Fast yet safe: Early-exiting with risk control. *Advances in Neural Information Processing Systems* 37:129825–129854

Ji W, Yuan W, Getzen E, Cho K, Jordan MI, et al. 2025. An overview of large language models for statisticians. *arXiv preprint arXiv:2502.17814*

Jiang Z, Araki J, Ding H, Neubig G. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics* 9:962–977

Jiang Z, Liu A, Van Durme B. 2025. Conformal linguistic calibration: Trading-off between factuality and specificity. *arXiv preprint arXiv:2502.19110*

Kadavath S, Conerly T, Askell A, Henighan T, Drain D, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*

Kaur R, Jha S, Roy A, Park S, Dobriban E, et al. 2022. idecode: In-distribution equivariance for conformal out-of-distribution detection, In *Proceedings of the AAAI Conference on Artificial Intelligence*

Khakhar A, Mell S, Bastani O. 2023. Pac prediction sets for large language models of code, In *International Conference on Machine Learning*, pp. 16237–16249, PMLR

Kipnis A, Voudouris K, Buschoff LMS, Schulz E. 2025. metabench - a sparse benchmark of reasoning and knowledge in large language models, In *The Thirteenth International Conference on Learning Representations*

Kotek H, Dockum R, Sun D. 2023. Gender bias and stereotypes in large language models, In *Proceedings of the ACM collective intelligence conference*, pp. 12–24

Kuhn L, Gal Y, Farquhar S. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, In *The Eleventh International Conference on Learning Representations*

Lee Y, Dobriban E, Tchetgen ET. 2024. Conditional predictive inference for missing outcomes. *arXiv preprint arXiv:2403.04613*

Lehmann EL, Romano JP. 2005. Testing statistical hypotheses. Springer Science & Business Media

Lei J, G'Sell M, Rinaldo A, Tibshirani R, Wasserman L. 2018. Distribution-free predictive inference for regression. *Journal of the American Statistical Association* 113(523):1094–1111

Lei J, Robins J, Wasserman L. 2013. Distribution-free prediction sets. *Journal of the American Statistical Association* 108(501):278–287

Lei J, Wasserman L. 2014. Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76(1):71–96

Li K, Hopkins AK, Bau D, Viégas F, Pfister H, Wattenberg M. 2023a. Emergent world representations: Exploring a sequence model trained on a synthetic task, In *The Eleventh International Conference on Learning Representations*

Li K, Patel O, Viégas F, Pfister H, Wattenberg M. 2023b. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems* 36:41451–41530

Li S, Ji X, Dobriban E, Sokolsky O, Lee I. 2022. Pac-wrap: Semi-supervised pac anomaly detection, In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*

Li S, Park S, Lee I, Bastani O. 2024. Traq: Trustworthy retrieval augmented question answering via conformal prediction, In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3799–3821

Lichtenstein S, Fischhoff B, Phillips LD. 1977. Calibration of probabilities: The state of the art, In *Decision Making and Change in Human Affairs: Proceedings of the Fifth Research Conference on Subjective Probability, Utility, and Decision Making, Darmstadt, 1–4 September, 1975*, pp. 275–324, Springer

Lin S, Hilton J, Evans O. 2022. Teaching models to express their uncertainty in words. *Transactions*

on Machine Learning Research

Lin Z, Trivedi S, Sun J. 2024. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*

Liu H, Dou ZY, Wang Y, Peng N, Yue Y. 2024. Uncertainty calibration for tool-using language agents, In *Findings of the Association for Computational Linguistics: EMNLP 2024*, eds. Y Al-Onaizan, M Bansal, YN Chen. Miami, Florida, USA: Association for Computational Linguistics

Liu X, Chen T, Da L, Chen C, Lin Z, Wei H. 2025. Uncertainty quantification and confidence calibration in large language models: A survey. *arXiv preprint arXiv:2503.15850*

Manduchi L, Meister C, Pandey K, Bamler R, Cotterell R, et al. 2025. On the challenges and opportunities in generative AI. *Transactions on Machine Learning Research* Survey Certification

Matton A, Sherborne T, Aumiller D, Tommasone E, Alizadeh M, et al. 2024. On leakage of code generation evaluation datasets. *arXiv preprint arXiv:2407.07565*

Meng K, Bau D, Andonian A, Belinkov Y. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems* 35:17359–17372

Mikolov T, Chen K, Corrado G, Dean J. 2013a. Efficient estimation of word representations in vector space

Mikolov T, Yih Wt, Zweig G. 2013b. Linguistic regularities in continuous space word representations, In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pp. 746–751

Miller E. 2024. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*

Mincer JA, Zarnowitz V. 1969. The evaluation of economic forecasts. In *Economic forecasts and expectations: Analysis of forecasting behavior and performance*. NBER, 3–46

Mirzadeh SI, Alizadeh K, Shahrokhi H, Tuzel O, Bengio S, Farajtabar M. 2025. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models, In *The Thirteenth International Conference on Learning Representations*

Mohri C, Hashimoto T. 2024. Language models with conformal factuality guarantees, In *Forty-first International Conference on Machine Learning*

Moniri B, Hassani H, Dobriban E. 2025. Evaluating the performance of large language models via debates, In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*

Nag S, Ghosh U, Ta CK, Bose S, Li J, Roy-Chowdhury AK. 2025. Conformal prediction and mllm aided uncertainty quantification in scene graph generation, In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11676–11686

Nanda N, Rajamanoharan S, Kramár J, Shah R. 2023. Fact-finding: Attempting to reverse-engineer factual recall

Noarov G, Mallick S, Wang T, Joshi S, Sun Y, et al. 2025. Foundations of top-k decoding for language models. *arXiv preprint arXiv:2505.19371*

Noorani S, Kiyani S, Pappas G, Hassani H. 2025. Conformal prediction beyond the seen: A missing mass perspective for uncertainty quantification in generative models. *arXiv preprint arXiv:2506.05497*

Oosterhuis H, Jagerman R, Qin Z, Wang X, Bendersky M. 2024. Reliable confidence intervals for information retrieval evaluation using generative ai, In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2307–2317

Overman W, Bayati M. 2025. Conformal arbitrage: Risk-controlled balancing of competing objectives in language models. *arXiv preprint arXiv:2506.00911*

Papadopoulos H, Proedrou K, Vovk V, Gammerman A. 2002. Inductive confidence machines for regression, In *European Conference on Machine Learning*, pp. 345–356, Springer

Park S, Dobriban E, Lee I, Bastani O. 2022a. PAC prediction sets for meta-learning, In *Advances in Neural Information Processing Systems*

Park S, Dobriban E, Lee I, Bastani O. 2022b. PAC prediction sets under covariate shift, In *Inter-*

national Conference on Learning Representations

Pearl J. 2001. Direct and indirect effects. *Probabilistic and Causal Inference: The Works of Judea Pearl* :373

Pennington J, Socher R, Manning CD. 2014. Glove: Global vectors for word representation, In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543

Polo FM, Weber L, Choshen L, Sun Y, Xu G, Yurochkin M. 2024. tinybenchmarks: evaluating llms with fewer examples, In *International Conference on Machine Learning*, pp. 34303–34326, PMLR

Qiu H, Dobriban E, Tchetgen Tchetgen E. 2023. Prediction sets adaptive to unknown covariate shift. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 85(5):1680–1705

Quach V, Fisch A, Schuster T, Yala A, Sohn JH, et al. 2024. Conformal language modeling, In *The Twelfth International Conference on Learning Representations*

Radford A, Jozefowicz R, Sutskever I. 2017. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*

Ravfogel S, Goldberg Y, Goldberger J. 2023. Conformal nucleus sampling, In *Findings of the Association for Computational Linguistics: ACL 2023*, eds. A Rogers, J Boyd-Graber, N Okazaki. Toronto, Canada: Association for Computational Linguistics

Ren AZ, Clark J, Dixit A, Itkina M, Majumdar A, Sadigh D. 2024. Explore until confident: Efficient exploration for embodied question answering, In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*

Ren AZ, Dixit A, Bodrova A, Singh S, Tu S, et al. 2023. Robots that ask for help: Uncertainty alignment for large language model planners, In *Conference on Robot Learning*, pp. 661–682, PMLR

Rimsky N, Gabrieli N, Schulz J, Tong M, Hubinger E, Turner A. 2024. Steering llama 2 via contrastive activation addition, In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522

Romano Y, Sesia M, Candes E. 2020. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems* 33:3581–3591

Rudinger R, Naradowsky J, Leonard B, Van Durme B. 2018. Gender bias in coreference resolution, In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 8–14

Ruffolo JA, Nayfach S, Gallagher J, Bhatnagar A, Beazer J, et al. 2025. Design of highly functional genome editors by modelling crispr–cas sequences. *Nature* :1–8

Sankaranarayanan S, Angelopoulos A, Bates S, Romano Y, Isola P. 2022. Semantic uncertainty intervals for disentangled latent spaces., In *NeurIPS*

Saunders C, Gammerman A, Vovk V. 1999. Transduction with confidence and credibility, In *IJCAI*

Schuster T, Fisch A, Gupta J, Dehghani M, Bahri D, et al. 2022. Confident adaptive language modeling, In *Advances in neural information processing systems*, eds. S Koyejo, S Mohamed, A Agarwal, D Belgrave, K Cho, A Oh, vol. 35, p. 17456–17472, Curran Associates, Inc.

Schuster T, Fisch A, Jaakkola T, Barzilay R. 2021. Consistent accelerated inference via confident adaptive transformers, In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, eds. MF Moens, X Huang, L Specia, SWt Yih. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics

Schweighofer K, Aichberger L, Ielanskyi M, Hochreiter S. 2025. On information-theoretic measures of predictive uncertainty, In *The 41st Conference on Uncertainty in Artificial Intelligence*

Sesia M, Favaro S, Dobriban E. 2023. Conformal frequency estimation using discrete sketched data with coverage for distinct queries. *Journal of Machine Learning Research* 24(348):1–80

Shafer G, Vovk V. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research* 9(Mar):371–421

Shi F, Chen X, Misra K, Scales N, Dohan D, et al. 2023. Large language models can be easily

distracted by irrelevant context, In *International Conference on Machine Learning*, pp. 31210–31227, PMLR

Shorinwa O, Mei Z, Lidard J, Ren AZ, Majumdar A. 2024. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *arXiv preprint arXiv:2412.05563*

Si W, Park S, Lee I, Dobriban E, Bastani O. 2024. PAC prediction sets under label shift. *International Conference on Learning Representations*

Snyder D, Hancock AJ, Badithela A, Dixon E, Miller P, et al. 2025. Is your imitation learning policy better than mine? policy comparison with near-optimal stopping. *arXiv preprint arXiv:2503.10966*

Soumm M. 2024. Causal inference tools for a better evaluation of machine learning. *arXiv preprint arXiv:2410.01392*

Strauss I, Moure I, O'Reilly T, Rosenblat S. 2025. Real-world gaps in ai governance research. *arXiv preprint arXiv:2505.00174*

Subramani N, Suresh N, Peters ME. 2022. Extracting latent steering vectors from pretrained language models, In *Findings of the Association for Computational Linguistics (ACL Findings)*

Suh N, Cheng G. 2024. A survey on statistical theory of deep learning: Approximation, training dynamics, and generative models. *Annual Review of Statistics and Its Application* 12

Sun J, Liao QV, Muller M, Agarwal M, Houde S, et al. 2022. Investigating explainability of generative ai for code through scenario-based design, In *Proceedings of the 27th International Conference on Intelligent User Interfaces*, pp. 212–228

Tan D, Chanin D, Lynch A, Paige B, Kanoulas D, et al. 2024. Analysing the generalisation and reliability of steering vectors. *Advances in Neural Information Processing Systems* 37:139179–139212

Teneggi J, Tivnan M, Stayman W, Sulam J. 2023. How to trust your diffusion model: A convex optimization approach to conformal risk control, In *International Conference on Machine Learning*, pp. 33940–33960, PMLR

The Royal Swedish Academy of Sciences. 2024. Press release: The nobel prize in chemistry 2024. `https://www.nobelprize.org/prizes/chemistry/2024/press-release/`. Accessed: 2 September 2025

Trivedi S, Nord BD. 2025. On the need to align intent and implementation in uncertainty quantification for machine learning. *arXiv preprint arXiv:2506.03037*

Turner AM, Thiergart L, Leech G, Udell D, Vazquez JJ, et al. 2023. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*

Ulmer D, Zerva C, Martins A. 2024. Non-exchangeable conformal language generation with nearest neighbors, In *Findings of the Association for Computational Linguistics: EACL 2024*, eds. Y Graham, M Purver. St. Julian's, Malta: Association for Computational Linguistics

Van Calster B, McLernon DJ, Van Smeden M, Wynants L, Steyerberg EW, et al. 2019. Calibration: the achilles heel of predictive analytics. *BMC medicine* 17(1):230

Van Calster B, Vickers AJ. 2015. Calibration of risk prediction models: impact on decision-analytic performance. *Medical decision making* 35(2):162–169

Vasconcelos H, Bansal G, Fourney A, Liao QV, Wortman Vaughan J. 2025. Generation probabilities are not enough: Uncertainty highlighting in ai code completions. *ACM Trans. Comput.-Hum. Interact.* 32(1)

Vig J, Gehrmann S, Belinkov Y, Qian S, Nevo D, et al. 2020. Investigating gender bias in language models using causal mediation analysis. *Advances in neural information processing systems* 33:12388–12401

Vincent JA, Nishimura H, Itkina M, Shah P, Schwager M, Kollar T. 2024. How generalizable is my behavior cloning policy? a statistical approach to trustworthy performance evaluation. *IEEE Robotics and Automation Letters*

Vovk V. 2012. Conditional validity of inductive conformal predictors, In *Asian conference on ma-*

chine learning, pp. 475–490, PMLR

Vovk V, Gammerman A, Saunders C. 1999. Machine-learning applications of algorithmic randomness, In *International Conference on Machine Learning*

Vovk V, Gammerman A, Shafer G. 2005. Algorithmic learning in a random world. Springer Science & Business Media

Wald A. 1943. An extension of wilks' method for setting tolerance limits. *The Annals of Mathematical Statistics* 14(1):45–55

Wang Y, Shi H, Han L, Metaxas D, Wang H. 2024. Blob: Bayesian low-rank adaptation by backpropagation for large language models. *Advances in Neural Information Processing Systems* 37:67758–67794

Watson JL, Juergens D, Bennett NR, Trippe BL, Yim J, et al. 2023. De novo design of protein structure and function with rfdiffusion. *Nature* 620(7976):1089–1100

Wilks SS. 1941. Determination of sample sizes for setting tolerance limits. *The Annals of Mathematical Statistics* 12(1):91–96

Wu Z, Qiu L, Ross A, Akyürek E, Chen B, et al. 2024. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks, In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, eds. K Duh, H Gomez, S Bethard. Mexico City, Mexico: Association for Computational Linguistics

Xia Z, Xu J, Zhang Y, Liu H. 2025. A survey of uncertainty estimation methods on large language models. *arXiv preprint arXiv:2503.00172*

Yadkori YA, Kuzborskij I, Stutz D, György A, Fisch A, et al. 2024. Mitigating llm hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*

Yang AX, Robeyns M, Wang X, Aitchison L. 2024. Bayesian low-rank adaptation for large language models, In *The Twelfth International Conference on Learning Representations*

Zhang B, Li S, Bastani O. 2024. Conformal structured prediction. *arXiv preprint arXiv:2410.06296*

Zhang F, Nanda N. 2024. Towards best practices of activation patching in language models: Metrics and methods, In *Proc. of the International Conference on Learning Representations (ICLR)*

Zhang H, Wang P, Chen S, Zhang Z, Qu Q. 2025. Generalization of diffusion models: Principles, theory, and implications. *SIAM News* `https://www.siam.org/publications/siam-news/articles/generalization-of-diffusion-models-principles-theory-and-implications/`

Zhao J, Wang T, Yatskar M, Ordonez V, Chang KW. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods, In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*

Zhou H, Huang H, Zhao Z, Han L, Wang H, et al. 2025. Lost in benchmarks? rethinking large language model benchmarking with item response theory. *arXiv preprint arXiv:2505.15055*

Zollo TP, Morrill T, Deng Z, Snell J, Pitassi T, Zemel R. 2024. Prompt risk control: A rigorous framework for responsible deployment of large language models, In *The Twelfth International Conference on Learning Representations*

Zou A, Phan L, Chen S, Campbell J, Guo P, et al. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*