Statistical Significance of Feature Importance Rankings

Jeremy Goldwasser[1] **Giles Hooker**[2]

[1]Department of Statistics, University of California, Berkeley
[2]Department of Statistics and Data Science, University of Pennsylvania

Abstract

Feature importance scores are ubiquitous tools for understanding the predictions of machine learning models. However, many popular attribution methods suffer from high instability due to Monte Carlo sampling. Leveraging novel ideas from hypothesis testing, we devise techniques that ensure the most important features are correct with high-probability guarantees. These are capable of assessing both the set of K top-ranked features as well as the order of its elements. Given local or global importance scores, we demonstrate how to retrospectively verify the stability of the highest ranks. We then introduce two efficient sampling algorithms that identify the K most important features, perhaps in order, with bounded error rate. The theoretical justification for these procedures is validated empirically on SHAP and LIME.

1 INTRODUCTION

Many machine learning (ML) algorithms have impressive predictive power but poor interpretability relative to simpler alternatives like decision trees and linear models. This trade-off has motivated a wide body of work seeking to explain how black-box models make predictions [Belle and Papantonis, 2021]. Such work is essential for building trust in ML systems in areas like finance, healthcare, and criminal justice, in which the consequences of model misbehavior may be severe [Dubey and Chandani, 2022, Ferdous et al., 2020, Mandalapu et al., 2023, Berk, 2019]. Interpretable ML methods can also help develop understanding of complex processes, augmenting domain knowledge with new hypotheses.

To that end, feature importance scores quantify how much the features contribute to the model's predictions. These may explain model behavior at the resolution of an individual input (*local*) or in aggregate (*global*). A wide range of methods have been proposed, detailed in Section 2.

The particular value of a feature's importance score may be of less practical interest than its ranking: which features are the most important, and the order of these highlighted features. A scientist using a machine learning model to predict disease risk from a patient's genetic profile will focus on the genes with the highest importance scores for further study, prioritizing the ranking of the metric over its specific values. Similarly, explanations often report only a small number of features in order not to overwhelm the user. LIME [Ribeiro et al., 2016], for example, explicitly regularizes to report a fixed number of features.

Unfortunately, many standard attribution methods suffer from instability induced by random sampling. Rerunning the same procedure could yield different explanations for which features are the most important. Local methods such as SHAP and LIME require Monte Carlo sampling to calculate scores [Lundberg and Lee, 2017]. Global scores like LOCO are calculated on some finite set of input data, yet are used to infer across the entire population of data [Lei et al., 2017]. This inherent lack of reproducibility seriously undermines the credibility of these analyses [Yu and Kumbier, 2020].

To address this, we cast a large class of local and global attributions into a unifying framework, based on properties of unbiasedness and asymptotic normality. For any method in this framework, we propose techniques to verify that the observed importance rankings are correct with high-probability guarantees. Amongst the highest-ranking features, these techniques assess the stability of both the top-K set as well as their ordering relative to one another.

We provide retrospective tools that analyze the rankings of given feature attributions (Section 4). Then, we propose two sampling methods which ensure the K highest-ranked features are correct with probability exceeding $1 - \alpha$ (Section 5). The first of these methods leverages the Sequential Probability Ratio Test (SPRT) from Wald [1945] to sample continually. The second approach, more efficient for certain

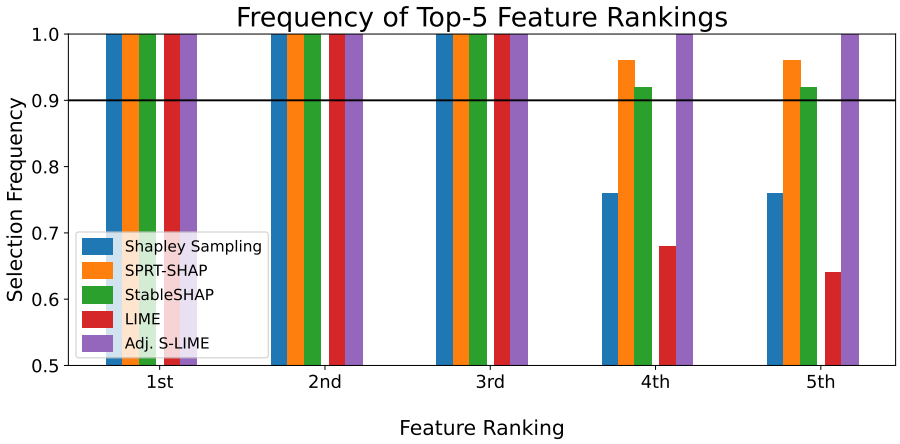

Figure 1: Instability of top-5 feature rankings, Adult Census Income dataset. StableSHAP, SPRT-SHAP, and Adj. S-LIME are our contributions, run at $\alpha = 0.1$; meaning 90% of replicates should return the same feature set. The computational budget of StableSHAP and Shapley Sampling are the same.

scores, iterates sample size calculations on ambiguously-ranked pairs. Applied to Shapley values, we refer to these algorithms as SPRT-SHAP and StableSHAP. Figure 1 highlights the improvement in stability of our sampling algorithms relative to baselines on SHAP and LIME. [1]

2 FEATURE IMPORTANCE SCORES

2.1 LOCAL FEATURE IMPORTANCE

A broad range of research has addressed the question of how best to attribute variable importance for individual predictions. SHAP [Lundberg and Lee, 2017] and LIME [Ribeiro et al., 2016] are amongst the most popular local attribution methods. Both are model-agnostic and entail random Monte Carlo computation. We provide a brief review of these two techniques, then introduce global feature importance.

2.1.1 SHAP

SHAP (SHapley Additive exPlanations) is a special case of the Shapley value from game theory [Shapley, 1952]. Shapley values, and SHAP by extension, uniquely satisfy several reasonable desiderata for credit allocation. For any subset S of the d variables, define the value function $v : S \to \mathbb{R}$ and weighting kernel $w_S = \binom{d-1}{d-|S|-1}^{-1}$. The Shapley value for variable j is

$$\phi_j(v) = \frac{1}{d} \sum_{S \subseteq [d] \setminus \{j\}} w_S(v(S \cup \{j\}) - v(S)). \quad (1)$$

SHAP popularized the use of Shapley values for local feature attribution. In this context, $v(S)$ is the prediction the model $\hat{f}$ would have made on input x if it only had access to the features x_S with indices in $S \subseteq [d]$. In theory, this prediction may be obtained by refitting the model on those features [Strumbelj et al., 2009, Lipovetsky and Conklin, 2001]. For computational convenience, however, it is common to sample the features in its complement S^c, concatenate them onto x_S, and take the prediction $\hat{f}(x_S, x_{S^c})$. The unknown features are usually sampled from their marginal distributions, as in SHAP [Lundberg and Lee, 2017, Strumbelj and Kononenko, 2014]. Other works propose using conditional distributions $X_{S^c}|X_S$ instead [Aas et al., 2021, Frye et al., 2020].

Computing the exact Shapley value (1) requires evaluating $v(S)$ for $O(2^d)$ terms. When d is large, this is computationally prohibitive, so approximation algorithms must be used instead. The basic approach is Shapley Sampling (Algorithm 7) [Strumbelj and Kononenko, 2014]. Rather than using all subsets of $[d]\setminus\{j\}$, this algorithm samples n subsets at random.

$$\hat{\phi}_j(v) = \frac{1}{n} \sum_{i=1}^{n} v(S_j^i \cup \{j\}) - v(S_j^i). \quad (2)$$

This sample average is unbiased for the true Shapley value, meaning $\mathbb{E}[\hat{\phi}_j] = \phi_j$ [Strumbelj et al., 2009]. Lundberg and Lee [2017] later proposed KernelSHAP, a more efficient estimator than Shapley Sampling. KernelSHAP obtains all d Shapley values at once in a linear regression framework.

2.1.2 LIME

Local Interpretable Model-agnostic Explanations, or LIME, explains an individual prediction via a local surrogate

[1] Our Python code and experimental results are at `https://github.com/jeremy-goldwasser/feature-rankings`.

[Ribeiro et al., 2016]. Around the input in question, LIME first samples a large number of random data points. These are passed through the model, producing a collection of labeled pseudodata. The pseudodata is then used to fit an inherently interpretable model, which summarizes the salient factors driving the original model's prediction.

The explanation model is typically a linear model, possibly regularized for smoothness or sparsity. For example, the default choice in the `lime` Python package is a sparse linear model with K nonzero coefficients. Because linear models associate each feature with a unique regression coefficient, they provide a natural framework for feature attributions.

In this setting, LIME importance rankings may be obtained in a number of ways. One could score each feature as the product of its regression coefficient and observed value, with the K highest scores corresponding to the most important features. When the explanation model is sparse, another option is to take the first K features that enter the Lasso path with decreasing λ [Tibshirani, 1996, Tibshirani and Taylor, 2011]. Finally, the authors propose K-Lasso, ranking the top K features with the order they enter the Least Angle Regression path [Efron et al., 2004].

2.2 GLOBAL FEATURE IMPORTANCE

A number of methods summarize a model's general behavior with global feature importance scores. The global scores $\hat{\phi}_j \ \forall j$ are calculated from a finite number of inputs x_i. They may be used as a proxy for ϕ_j, the population score describing the model across the entire sample space $\mathcal{X}$.

Most global attribution methods are a sample average of scores on individual data points. For local score s_j^i on feature j of input i, these attributions are

$$\hat{\phi}_j = \frac{1}{n} \sum_{i=1}^{n} s_j^i. \tag{3}$$

One approach merely takes the average of local feature attribution scores. For example, SAGE is the mean LossSHAP score; this is a Shapley value based on the per-sample loss, rather than the prediction. A global Shapley method that does not require labels is the average absolute SHAP value [Lundberg and Lee, 2017, Lundberg et al., 2020]. Similarly, [Ribeiro et al., 2016] computes the mean absolute LIME score to present global explanations. Analogous strategies have been suggested for counterfactual explanations [Ley et al., 2022]. More broadly, van der Linden et al. [2019] proposes global importance scores that average the absolute value of any local metric.

Another subset of global scores taking the form (3) compares model performance before and after controlling for feature j. The performance is evaluated with some loss function $\mathcal{L}$ on labeled data points $\{(x_i, y_i)\}_{i=1}^n$. These methods modify either the data or model to produce some altered prediction $\tilde{y}_{ij}$. Their local scores take the form

$$s_j^i = \mathcal{L}(y_i, \hat{y}_i) - \mathcal{L}(y_i, \tilde{y}_{ij}).$$

Here, we survey choices of $\tilde{y}_{ij}$ for a number of popular loss-based frameworks.

- **Permutation Importance** [Breiman, 2001, Fisher et al., 2019]. Averaging across unseen data points, this approach takes predictions after permuting the values of feature j. Its original formulation, Mean Decrease in Accuracy (MDA), used out-of-bag training inputs in random forests.
- **Conditional Variable Importance** [Strobl et al., 2008]. Permutation Importance samples features from their marginal distribution, producing unlikely data with potentially inaccurate predictions. Conditional Variable Importance instead samples each feature from its distribution conditional on the other features.
- **Leave-One-Covariate-Out** [Lei et al., 2017, Verdinelli and Wasserman, 2023]. LOCO retrains the model without each feature j, then predicts $\tilde{y}_{ij}$ using all other features.
- **Permute-and-relearn**. This technique retrains after imputing random values of a given feature from its marginal [Hooker et al., 2021] or conditional [Mentch and Hooker, 2016] distribution.

A separate strategy trains an interpretable surrogate model on the predictions of the black-box model [Molnar, 2022, Hinton et al., 2015]. When a linear model is used, its regression coefficients may be taken as importance scores in the same fashion as LIME.

3 RANK VERIFICATION REVIEW

3.1 STABLE IMPORTANCE RANKINGS

A small body of work seeks to identify the most important features from estimated feature importances. Neuhof and Benjamini [2024] established population rankings in retrospect via simultaneous confidence intervals, adjusted for multiple testing with procedures such as Holm's method [Holm, 1979]. While valid at level α, the use of such multiple testing corrections drastically reduces the power of this procedure. The p-values are inflated by a factor of $O(d^2)$, the total number of pairwise comparisons.

A number of works propose sampling algorithms designed to stabilize the set of K highest Shapley values [Narayanam and Narahari, 2008, Pliatsika et al., 2024, Chabrier et al., 2024, Kolpaczki et al., 2021, Kariyappa et al., 2024]. The latter two present probabilistic guarantees for their top-K bandit algorithms, albeit in terms of unknown parameters

based on the gaps between Shapley values. Kariyappa et al. [2024] only guarantees ϵ-approximate solutions — allowing incorrect top-K features to be included, so long as they are within ϵ of the true top K. The algorithms are further hindered by the use of loose bounds like the Bonferroni correction.

3.2 GAUSSIAN METHODOLOGY

Given a set of random variables, it is often of interest to verify whether the highest observed value - or values - indeed matches the population ranking with high probability. The methods discussed in 3.1 do so coarsely, relying on multiple testing adjustments, loose bounds, and unknown parameters. In contrast, more powerful verification methods have been studied under various probability distributions via selective inference [Hung and Fithian, 2019, Taylor and Tibshirani, 2015].

Goldwasser et al. [2025] introduced the first selective inference-based methods to verify the ranks of Gaussian data with unequal variances. For $j \in [d] = \{1, \ldots, d\}$, consider independent random variables $X_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$, where σ_j is known and μ_j is unknown. Let x_j denote the realized values of these random variables X_j. For notational convenience, sort the variables according to their order statistics, such that X_1 has the highest observed value x_1, X_2 is second-largest, etc.

The primary objective is to verify the "winner" X_1 as the "best," meaning $\mu_1 > \mu_j$ for all $j > 1$. Its rank is verified upon rejecting the null hypothesis H_{01} that μ_1 is *not* the highest, conditioned on the selection event that X_1 wins.

$$H_{01} : \bigcup_{j>1} \underbrace{\left\{\mu_1 \leq \mu_j \mid X_1 > \max_{k>1} X_k\right\}}_{H_{01j}}. \tag{4}$$

To test this null, define $\bar{\mu}_{1j} = \frac{\sigma_j^2 x_1 + \sigma_1^2 x_j}{\sigma_1^2 + \sigma_j^2}$, $\bar{\sigma}_{1j}^2 = \frac{\sigma_1^4}{\sigma_1^2 + \sigma_j^2}$, and $\bar{\eta}_{1j} = \max(\bar{\mu}_{1j}, \max_{k \neq 1,j} x_k)$. This parameterizes a normal distribution with mean $\bar{\mu}_{1j}$ and variance $\bar{\sigma}_{1j}^2$, whose left tail has been truncated at $\bar{\eta}_{1j}$. For standard normal CDF $\Phi(\cdot)$, further define p_{1j}:

$$p_{1j} = \frac{1 - \Phi(\frac{x_1 - \bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}{1 - \Phi(\frac{\bar{\eta}_{1j} - \bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}. \tag{5}$$

p_{1j} is the tail mass of this truncated normal distribution above x_1. Goldwasser et al. [2025] shows it is a valid p-value for H_{01j} (4). Furthermore, H_{01} may be tested at level α with

$$p_1^* = \max_{j>1} p_{1j}. \tag{6}$$

The winner is verified when $p_1^* \leq \alpha$, rejecting H_{01}. This is equivalent to rejecting H_{01j} with significant p-values $p_{1j} \leq \alpha$ for all j.

This result has useful extensions for top-K rank verification, still without explicit multiple testing adjustments. These verify all ranks $i \leq K$ by rejecting nulls of the form

$$H_{0i} : \bigcup_{j} \underbrace{\left\{\mu_i \leq \mu_j \mid X_i > \max_{k>i} X_k\right\}}_{H_{0ij}}. \tag{7}$$

Procedure 1 (Top-K Ranks). This outputs an integer $K \geq 0$ such that the probability that the top-K rankings are correct is at least $1 - \alpha$. It iterates the test in Equation (6) on successive ranks until a failure to reject (Alg. 3).

First, test H_{01} with $p_1^* = \max_{j>1} p_{1j}$. If $p_1^* \leq \alpha$, then test $H_{02} = \bigcup_{j>2} H_{02j}$ with $p_2^* = \max_{j>2} p_{2j}$. Continuing for $K \geq 0$ rejections, stop at the first failure to reject, wherein $p_{(K+1)j} > \alpha$ for some $j > K + 1$.

Procedure 2 (Top-K Set). This test evaluates whether the set of K largest observed elements are guaranteed to have the highest means with probability exceeding $1 - \alpha$ (Alg. 4). Unlike in Procedure 1, here K is fixed a priori, and the orderining within the top K does not matter.

Procedure 2 tests $\widetilde{H}_{0i} = \bigcup_{j>K} H_{0ij}$, i.e. that μ_i is not actually in the top K set, with $\tilde{p}_i^* = \max_{j>K} p_{ij}$. When all K nulls $\widetilde{H}_{0i}$ reject, the top-K set is verified. This is equivalent to having $p_{ij} \leq \alpha$ for all $i \leq K$ and $j > K$.

Corollaries 1 and 2 in Goldwasser et al. [2025] establish the validity of Procedures 1 and 2. We refer the reader to the original manuscript for the proofs.

4 RETROSPECTIVE SCORE VERIFICATION

Under mild assumptions, most of the scores discussed in Section 2 are normally distributed and unbiased. Their variance can be well-approximated with simple techniques.

To see this, consider first attributions of the form $\hat{\phi}_j = \frac{1}{n} \sum_{i=1}^n s_j^i$, such as Shapley Sampling and LOCO. When the samples used to compute $\hat{\phi}_j$ are selected at random, $\hat{\phi}_j$ is unbiased for the population score ϕ_j. Moreover, with sufficiently large n, the distribution of $\hat{\phi}_j$ converges to a normal distribution centered around ϕ_j by the central limit theorem. The sample variance $\hat{\sigma}_j^2 = \frac{1}{n-1} \sum_{i=1}^n (s_j^i - \bar{s}_j)^2$ is unbiased for σ_j^2, converging at an $O(n^{-1/2})$ rate.

In addition, the scores of local (LIME) and global surrogate methods are linear regression coefficients. Assuming normal errors, Ordinary Least Squares coefficients are unbiased, and their variance can be estimated with standard formulas. Similarly, KernelSHAP expresses Shapley values as linear regression coefficients, solving with a weighted least squares. Covert and Lee [2020] proved it is asymptotically normal with negligible bias. They also introduced a variance estimator, studied and improved upon by Goldwasser and

Hooker [2023]. Finally, LIME with K-Lasso selects features with scaled correlations that are asymptotically normal [Zhou et al., 2021].

Further assume the importance scores $\hat{\phi}_j$ are independent. This certainly holds in some cases. For example, Shapley Sampling (2) uses different subsets S_j^i for each feature, so the resulting attributions are independent. However, mild correlation may exist in methods like KernelSHAP, where the same data is used to estimate multiple scores. Nevertheless, our empirical results indicate the ensuing procedures are always valid, and in fact somewhat conservative. Appendix D evaluates the merits of alternative approaches which use correlated testing.

Main result. Let $\hat{\phi}_1, \ldots, \hat{\phi}_d$ be a set of such feature importance scores. If the user desires, these may be absolute values. Assume they are independent, normal, unbiased, and with known oracle variance ($\hat{\sigma}_j^2 = \sigma_j^2$). Then the procedures from Goldwasser et al. [2025] may be applied to verify the observed importance rankings.

Let $(\hat{k})$ denote the k^{th} largest score.Procedure 1 (Alg. 3) yields some non-negative integer K such that with probability at least $1 - \alpha$,

$$\phi_{(\hat{1})} > \ldots > \phi_{(\hat{K})} > \max_{\ell > K} \phi_{(\hat{\ell})}.$$

Procedure 2 (Alg. 4) tests the stability of the top-K set, for any user-defined $K \in \{1, \ldots, d-1\}$. When its test rejects at level α, the set of K most important features is correct, again with the same high-probability guarantee.

The test (6) may also be applied simultaneously, indicating whether the k^{th}-ranked feature is indeed higher than all lower ranks for all k of interest. Doing so is akin to the `lm` function in `R`, which tests the statistical significance of all coefficients in a linear model. Not all significant results necessarily hold at level α due to multiple testing; nevertheless, this provides a concise summary of which ranks are likely stable.

5 STABILIZED TOP-K ALGORITHMS

This section introduces algorithms that guarantee the K highest-ranking features, and perhaps their relative ordering, are correct with probability at least $1 - \alpha$, where K and α are predetermined by the user. This is motivated by numerous applications in which the K most important features are analyzed, e.g. Narayanam and Narahari [2008], Goli and Mohammadi [2022], Ghorbani and Zou [2020]. Again, features may be sorted via their absolute values.

The algorithms sample until Procedure 1 or 2 verifies the top-K features. To guarantee validity, Section 5.1 uses modified hypothesis tests based on the SPRT [Wald, 1945], whereas Section 5.2 performs sample size calculations to efficiently obtain each attribution.

5.1 SPRT APPROACH

A naive top-K ranking strategy would draw samples until Procedure 1 from Section 3.2 rejects for the K highest feature importances. An analogous approach for the top-K set samples until Procedure 2 rejects. However, doing so would not necessarily control the error rate at level α. This is because standard hypothesis tests like the p-values p_{ij} from Goldwasser et al. [2025] are not valid under *optional stopping*, when data is accumulated until the moment it indicates a significant result. This process inflates the Type I error rate because it permits the data to be tested multiple times without adjustment.

To address this, we modify Procedures 1 and 2 so the tests they conduct are valid under optional stopping. The canonical choice in this setting is the Sequential Probability Ratio Test [Wald, 1945]. After any number of samples have been drawn, SPRT computes the likelihood ratio

$$T = \frac{\max_{\theta \in H_1} \mathbb{P}(X \mid \theta)}{\max_{\theta \in H_0} \mathbb{P}(X \mid \theta)}. \tag{8}$$

Set Type I and II error rates α and β. The test accepts H_1 when $T \geq \frac{1-\beta}{\alpha}$, accepts H_0 when $T \leq \frac{\beta}{1-\alpha}$, and continues sampling otherwise.

In the context of feature importance rankings, we construct the SPRT likelihood ratio T with the following theorem. The proof, in Appendix C, involves maximum likelihood estimation in a selective inference framework.

Theorem 1. *Assume $X_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$ independently, where σ_j is known. Let ϕ and Φ be the standard normal PDF and CDF, and recall the definitions of $\bar{\mu}_{1j}$, $\bar{\sigma}_{1j}$, and $\bar{\eta}_{1j}$ from Section 3.2. For any Type I and II error thresholds α and β, a valid SPRT test statistic for $H_{01j} : \mu_1 \leq \mu_j \mid X_1 > \max_{k>1} X_k$ is*

$$T_{1j} = \left[\frac{\phi(0)}{1 - \Phi(\frac{\bar{\eta}_{1j} - x_i}{\bar{\sigma}_{1j}})}\right] \left[\frac{\phi(\frac{x_1 - \bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}{1 - \Phi(\frac{\bar{\eta}_{1j} - \bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}\right]^{-1}.$$

If $T_{1j} \geq \frac{1-\beta}{\alpha}$ for all j such that $X_i > X_j$, then verify μ_i as larger than all μ_j.

We employ Theorem (1) to learn the correct top-K rankings (Alg. 1) and set (Alg. 5) with probability $1 - \alpha$. In essence, our algorithms run computation until Procedures 1 and 2 verify the top K. However, the procedures are modified from Section 3.2: here, they test each null H_{0ij} with the likelihood ratio T_{ij}, rather than p-values p_{ij} (5). By SPRT, these tests can be conducted at any n without regard for optional stopping.

5.2 RESAMPLING APPROACH

While valid, SPRT is fairly conservative, as it must hold for any number of samples. An alternative approach tests with

Algorithm 1 Rank Stability via SPRT (SPRT-SHAP)

Require: Desired rankings $K > 0$, error rate $\alpha \in [0, 1]$, total sample budget n_{max}, samples between tests n_{btwn}
Ensure: $\hat{\phi}_1, \ldots, \hat{\phi}_d$ whose top-K rankings are correct with probability $\geq 1 - \alpha$
  $n \leftarrow 0$
  while $n < n_{max}$ **do**
    Generate n_{btwn} new samples
    $n \leftarrow n + n_{btwn}$
    $\hat{\phi}_n \leftarrow$ Feature importances fit on all n samples
    $\hat{\Sigma}_n \leftarrow$ Variances of all feature importances
    $K' \leftarrow$ Procedure 1 (Alg. 3) on $\hat{\phi}_n$ & $\hat{\Sigma}_n$, rejecting tests H_{0ij} if $T_{ij} > \frac{1-\beta}{\alpha}$ (Thm. 1)
    if $K' >= K$ **then**
      return $\hat{\phi}$, "Verified"
    end if
  end while
  return $\hat{\phi}$, "Failed to verify"

the original, more powerful p-values p_{ij} (5). To account for optional stopping, it throws out all data used to compute $\hat{\phi}_i$ and $\hat{\phi}_j$ when null H_{0ij} fails to reject. As a result, subsequent tests are independent of previous results. To guarantee validity, the earlier assumptions of independence, normality, unbiasedness, and oracle variance are still necessary.

This resampling strategy is only more efficient on certain importance scores. Shapley Sampling, for example, computes each attribution $\hat{\phi}_j$ in isolation with separate subsets S_j (2). Therefore when H_{0ij} fails to reject at level α, only $\hat{\phi}_i$ and $\hat{\phi}_j$ must be recomputed; data for all $k \neq i, j$ may be kept. This is not the case for KernelSHAP, in which the same set of samples is used to compute all d attributions $\hat{\phi}_j$. In that case, removing the entire dataset each time a pairwise test fails to reject is not viable.

Similarly, resampling may be more efficient than SPRT on global attributions computed in isolation (e.g. average Shapley Sampling, loss-based methods), but not jointly (e.g. average KernelSHAP, surrogate methods). Thus, all attributions that may benefit from this approach take the form

$$\hat{\phi}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} s_j^i.$$

When some pairwise null H_{0ij} fails to reject, the goal is to recompute $\hat{\phi}_i$ and $\hat{\phi}_j$ with as few samples as possible for H_{0ij} to subsequently reject. Here we present two methods to approximate this. Let $\tilde{\sigma}_j^2$ be the sample variance of s_j^i, so $\sigma_j^2 = \text{Var}(\hat{\phi}_j) = \frac{\tilde{\sigma}_j^2}{n_j}$. Goldwasser et al. [2025] show that when the runner-up has highest p-value, Equation (5) reduces to a Z-test at level $\alpha/2$. Assuming this occurs,

$$\frac{\hat{\phi}_i - \hat{\phi}_j}{\sqrt{\frac{\tilde{\sigma}_i^2}{n_i} + \frac{\tilde{\sigma}_j^2}{n_j}}} < Z_{1-\alpha/2}, \tag{9}$$

where $Z_{1-\alpha/2}$ is the upper $\frac{\alpha}{2}$ quantile of the standard normal distribution.

Suppose we want the new number of samples n_i, n_j to be the same n' for both features. Then solving Equation (9) for n' leads to the sample size

$$n' = \Big(\frac{Z_{1-\alpha/2}}{\hat{\phi}_i - \hat{\phi}_j}\Big)^2 (\tilde{\sigma}_i^2 + \tilde{\sigma}_j^2). \tag{10}$$

Alternatively, we may want the sample size to scale with the variance. Intuitively, more samples should be used to stabilize highly variable features. Defining $n_j' := \frac{\tilde{\sigma}_j^2}{\tilde{\sigma}_j^2} n_i'$ and solving for n_i' yields

$$n_i' = 2\Big(\frac{Z_{1-\alpha/2}}{\hat{\phi}_i - \hat{\phi}_j}\Big)^2 \tilde{\sigma}_i^2, \;\; n_j' = 2\Big(\frac{Z_{1-\alpha/2}}{\hat{\phi}_i - \hat{\phi}_j}\Big)^2 \tilde{\sigma}_j^2. \tag{11}$$

These lower bounds estimate the minimum number of samples needed to obtain an anticipated significant result. To avoid narrowly missing the mark, it is reasonable to choose values of n' that exceed them by a small buffer, e.g. 10%. It is entirely possible that more optimal choices of n_i' and n_j' exist; we leave this as an open problem.

Algorithm 2 Rank Stability via Resampling (StableSHAP)

Require: Desired rankings $K > 0$, error rate $\alpha \in [0, 1]$, per-feature sample budget n_{max}, intial per-feature samples n_{init}, buffer $c \geq 1$
Ensure: $\hat{\phi}_1, \ldots, \hat{\phi}_d$ whose top-K rankings are correct with probability $\geq 1 - \alpha$
  $\hat{\phi} \leftarrow$ Feature importances fit on n_{init} samples
  $\hat{\Sigma} \leftarrow$ Variances of all feature importances
  while $n_j < n_{max}\ \forall j$ **do**
    $K' \leftarrow$ Procedure 1 (Alg. 3) on $\hat{\phi}_n$ & $\hat{\Sigma}_n$, rejecting tests H_{0ij} if $T_{ij} > \frac{1-\beta}{\alpha}$ (Thm. 1)
    if $K' >= K$ **then**
      return $\hat{\phi}$, "Verified"
    else
      $n_i', n_j' \leftarrow$ Est. samples to reject, Eq. (10) or (11).
      $n_i', n_j' \leftarrow \min(\lceil cn_i' \rceil, n_{max}), \min(\lceil cn_j' \rceil, n_{max})$
      $\hat{\phi}_i, \hat{\phi}_j \leftarrow$ Attributions fit on n_i', n_j' samples
      $\hat{\Sigma}_i, \hat{\Sigma}_j \leftarrow$ Variances of feature importances
    end if
  end while
  return $\hat{\phi}$, "Failed to verify"

Running for the maximal number of samples is not guaranteed to yield K rejections at the desired error tolerance. Nevertheless, it may suffice even when the anticipated requisite budget is higher, since Equations (10) and (11) rely on plug-in variance estimates.

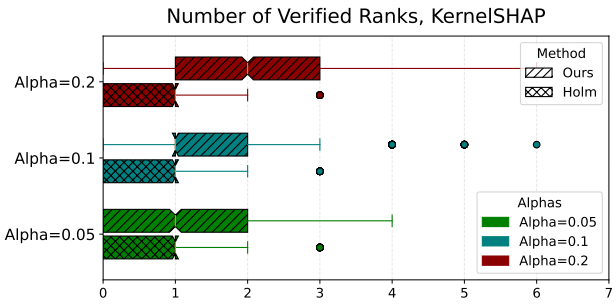

Figure 2: Comparing Number of Stable Features in Retrospect.

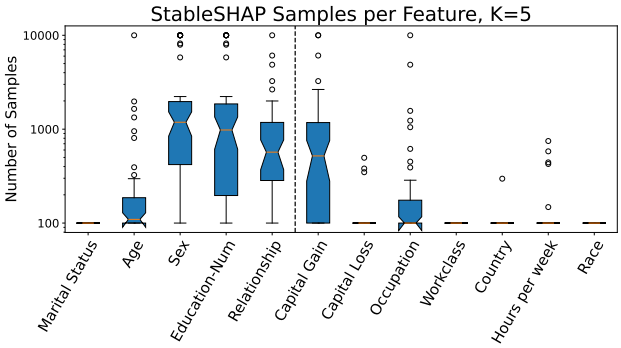

Figure 3: StableSHAP Sample Allocation by Feature Ranking.

6 EXPERIMENTS

We evaluated our retrospective and top-K methods' ability to stabilize rankings from SHAP and LIME. These attributions describe neural network classifiers fit on the Adult Census Income, Portuguese Bank, BRCA, Wisconsin Breast Cancer, and German Credit datasets. Further results in Appendix E demonstrate the efficiency of the resampling approach. Appendix F contains more detailed information on these experiments.

6.1 RETROSPECTIVE STABILITY

We tested the efficacy of our retrospective tools across a range of datasets, attribution methods, and significance levels. To do so, we randomly selected 30 inputs from the test set of each benchmark dataset. On each data point, we ran Shapley Sampling and KernelSHAP 50 times. For each iteration, we identified the number of stable ranks (Procedure 1) and assessed the stability of the top-5 set (Procedure 2).

We then computed the family-wise error rate (FWER) of the ranking and set procedures on each input. The FWER is the fraction of iterations with an error in the ranking or set of supposedly stable features. Iterations that do not verify any rankings or the top-K set are counted as error-free. We used the most common top-K ranking as the ground truth; this is almost certainly correct, since the Shapley estimators are unbiased.

Table 1 reports the maximal error rates across the 30 data points. In all 60 settings, the 30 FWERs are at most α, indicating these procedures successfully control the FWER. This supports the application of Goldwasser et al. [2025] in the context of feature importances.

Figure 2 shows the number of verified ranks on the German Credit dataset. We benchmark our method against the approach from Neuhof and Benjamini [2024], which adjusts one-sided p-values with Holm's method. This conservative multiple testing procedure has less statistical power than the tests we utilize from Goldwasser et al. [2025]. Experimental results reflect this improvement. For all error thresholds α, our retrospective procedure tends to verify more top ranks than the benchmark. The boxplots corresponding to our method consistently have longer tails, as well as a higher median for $\alpha = 0.2$.

Examining our method, the median number of verified ranks rises from one to two as α goes to 0.2. In the most extreme case, the ranking of the top six features is stable. However, a substantial fraction of inputs had only zero or one stable ranking. For $\alpha = 0.05$, the single top-ranking feature could not be verified on over 25% of inputs. These results emphasize the fragile reliability of Shapley rankings, and the need for algorithms that enforce their stability.

6.2 TOP-K RANK VERIFICATION

Empirical results demonstrate that our sampling algorithms identify the K most important features with probability $1 - \alpha$. Applied to Shapley estimation, we refer to these algorithms as SPRT-SHAP (1) and StableSHAP (2). SPRT-SHAP obtains Shapley values with KernelSHAP, whereas StableSHAP is based on Shapley Sampling.

Table 2 summarizes their performance across a range of datasets, desired ranks K, and significance levels α. The evaluation metric is the maximum error rate, introduced in Section 6.1 for Table 1. As before, FWERs for 30 inputs are computed empirically over 50 runs. In all settings for which the algorithms converged within the given sample budget, the FWER was below α.

StableSHAP is highly adaptive to the significance level. Its empirical FWERs were closest to α, getting up to a 10% FWER with $\alpha = 0.1$ and 20% with $\alpha = 0.2$. In contrast, SPRT-SHAP produced more stable top-K rankings, if it converged at all. This matches our intuition that SPRT is more conservative, requiring a higher evidence threshold to reject.

Moreover, StableSHAP allocates computation in a highly

Table 1: Maximum error rate of retrospective rank (R) and set (S) procedures, across 30 samples. $K = 5$ for set.

			Shapley Sampling						KernelSHAP					
			$\alpha = 0.05$		$\alpha = 0.1$		$\alpha = 0.2$		$\alpha = 0.05$		$\alpha = 0.1$		$\alpha = 0.2$	
Dataset	N	D	R	S	R	S	R	S	R	S	R	S	R	S
`Adult`	32,561	12	2%	4%	6%	4%	10%	10%	4%	4%	4%	6%	8%	12%
`Bank`	45,211	16	2%	0%	4%	2%	12%	8%	2%	0%	4%	0%	8%	2%
`BRCA`	572	20	2%	2%	8%	4%	10%	8%	2%	2%	2%	4%	10%	6%
`Credit`	1,000	20	2%	0%	8%	4%	10%	6%	2%	0%	2%	0%	6%	2%
`WBC`	569	30	2%	2%	8%	6%	10%	8%	6%	2%	6%	4%	10%	10%

Table 2: Maximum error rate of top-K rank (R) and set (S) procedures, across 30 input data points. NAs indicate the procedure does not reject all K tests on the provided inputs.

	StableSHAP								SPRT-SHAP							
	$K = 2$				$K = 5$				$K = 2$				$K = 5$			
	$\alpha = 0.1$		$\alpha = 0.2$		$\alpha = 0.1$		$\alpha = 0.2$		$\alpha = 0.1$		$\alpha = 0.2$		$\alpha = 0.1$		$\alpha = 0.2$	
Dataset	R	S	R	S	R	S	R	S	R	S	R	S	R	S	R	S
`Adult`	8%	2%	16%	16%	6%	6%	14%	14%	0%	0%	2%	0%	NA	0%	NA	8%
`Bank`	6%	2%	14%	0%	10%	10%	20%	16%	0%	0%	2%	2%	NA	0%	NA	2%
`BRCA`	6%	6%	14%	10%	10%	10%	20%	20%	NA	0%	NA	0%	NA	0%	NA	2%
`Credit`	4%	2%	8%	2%	4%	4%	12%	16%	0%	0%	0%	0%	NA	0%	NA	2%
`WBC`	0%	4%	10%	12%	6%	4%	20%	4%	0%	0%	4%	0%	NA	0%	0%	0%

efficient manner. It adaptively focuses on the features whose rankings are both ambiguous and relatively high. Figure 3 displays its sample allocations for a given input from the Adult dataset, running 50 times at $\alpha = 0.1$. StableSHAP uses only the initial 100 samples for the highest-ranked feature, Marital Status, as it wins by a wide margin. It also avoids precise estimation of ranks beyond the top K and its runner-up, Capital Gain. The main exception, Occupation, has considerably higher variance, thereby raising its p-values (Figure 6).

In contrast, most existing Shapley algorithms allot the same budget for all features. Figure 5 in Appendix E compares the performance of StableSHAP and Shapley Sampling, given the same inputs and computational budget. By and large, StableSHAP demonstrates improved stability, with fewer misranked features.

Figure 4 visualizes the average runtime of StableSHAP and SPRT-SHAP. The methods share the same minimal and maximal budget, and explain the same data. While both often achieve top-2 rank stability with the initial number of samples, StableSHAP is generally more efficient. SPRT-SHAP often takes a larger number of samples to converge, as evidenced by its higher third quartiles. Overall, these sample budgets are similar to the defaults in the `shap` package.

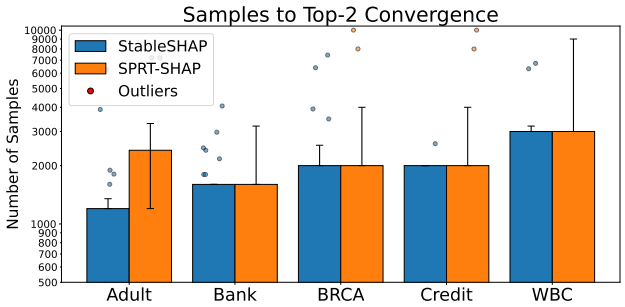

Figure 4: Sampling Efficiency of StableSHAP vs SPRT-SHAP.

6.3 LIME EXPERIMENTS

LIME, introduced in Section 2.1.2, fits an interpretable model on data randomly generated around a point of interest. The default model in the `lime` Python package is K-Lasso, which iteratively selects K features along the Least-Angle Regression (LARS) path. Our experiments modify Algorithm 2 to achieve top-K rank stability for LIME with K-Lasso.

LIME regenerates data at each step along the LARS path, so our methods are not applicable in their current form. Rather, to ensure the FWER is controlled at level-α, a multiple testing correction must be used. We apply the Bonferroni

Table 3: Maximum error rate (%) for top-5 ranking procedure on LIME with K-Lasso.

Dataset	$K = 2$ $\alpha = 0.1$	$K = 2$ $\alpha = 0.2$	$K = 5$ $\alpha = 0.1$	$K = 5$ $\alpha = 0.2$
Adult	0%	2%	NA	NA
Bank	2%	0%	0%	0%
BRCA	2%	0%	NA	NA
Credit	6%	8%	0%	0%
WBC	2%	2%	0%	0%

correction, verifying each feature selection at level α/K. Note this assumes LARS does not deselct any features, in which case more than K tests would be performed.

At each step, LARS iteratively chooses the predictor that is most correlated with the current residuals. These scaled correlations $\hat{c}_j$ are asymptotically normal by the central limit theorem. Therefore top-K rank stability can be achieved by running either algorithm in Section 5 at level α/K until the top rank is stable, for all K steps.

We did not implement our algorithms on LIME from scratch. Rather, we repurposed an existing method, S-LIME, to adhere generally within our framework [Zhou et al., 2021]. More details are in Appendix F.2

We ran the adjusted S-LIME procedure on 5 datasets at $\alpha = 0.1$ and 0.2. As before, we compare 30 inputs, computing the error rate with 50 runs across each. Table 3 displays the experimental results.

For all inputs, the error rate is always controlled at level α. In fact, this procedure is very conservative, with a maximal error rate of 2% for four out of five datasets. This may be attributed to its use of the Bonferroni correction, lowering the significance threshold by a factor of 5. Future experiments could run our top-K algorithms on LIME with OLS, requiring no such correction.

7 DISCUSSION

In this paper we present methods to obtain stable orderings of feature importance scores. For a user-defined error rate α, our retrospective procedures verify the highest observed rankings and top-K set. Our top-K algorithms efficiently run computation until the population rankings are attained with high probability. Our statistical guarantees, contingent on normal assumptions or asymptotic arguments, are unanimously justified by empirical results.

Our methods can be used to rank Shapley values in any context, not specifically for feature attributions. Other use cases include feature selection [Cohen et al., 2007], federated learning [Liu et al., 2022], data valuation [Ghorbani and Zou, 2019], multi-agent RL [Li et al., 2021], and ensembling [Rozemberczki and Sarkar, 2021]. Outside of ML, they have been applied in fields as diverse as ecology [Haake et al., 2007], online advertising [Zhao et al., 2018], supply chain management [Xu et al., 2018], and financial portfolio optimization [Shalit, 2020].

In addition to their analytical utility, our top-K methods run relatively efficiently. The resampling approach adaptively budgets computation towards the more important features. SPRT for Shapley importances enables use of KernelSHAP, a highly efficient algorithm. While SPRT is fairly conservative due to its anytime-valid requirement, future work could use more relaxed procedures that restrict the number of potential rejection times. Testing procedures that do so have been proposed in the context of clinical trials, e.g. Pocock [1977], O'Brien and Fleming [1979], Peto et al. [1976].

Our methods provide concrete statistical guarantees on their rankings and selections, with higher power and more reasonable assumptions than prior work [Kariyappa et al., 2024, Kolpaczki et al., 2021, Zhou et al., 2021]. Moreover, they may be applied in conjunction with methods that stabilize the attributions themselves [Goldwasser and Hooker, 2023, Mitchell et al., 2022].

References

Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artif. Intell.*, 298:103502, 2021. doi: 10.1016/J.ARTINT.2021.103502.

Vaishak Belle and Ioannis Papantonis. Principles and practice of explainable machine learning. *Frontiers Big Data*, 4:688969, 2021. doi: 10.3389/fdata.2021.688969.

Roger L. Berger. Multiparameter hypothesis testing and acceptance sampling. *Technometrics*, 24(4):295–300, 1982. doi: 10.2307/1267823. URL https://doi.org/10.2307/1267823.

Richard Berk. *Machine Learning Risk Assessments in Criminal Justice Settings*. Springer, 2019. ISBN 978-3-030-02271-6. doi: 10.1007/978-3-030-02272-3.

Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, 2001. doi: 10.1023/A:1010933404324.

Lisa Chabrier, Anton Crombach, Sergio Peignier, and Christophe Rigotti. Effective pruning for top-k feature search on the basis of shap values. *IEEE Access*, 12:163079–163092, 2024. doi: 10.1109/ACCESS.2024.3489958.

Shay B. Cohen, Gideon Dror, and Eytan Ruppin. Feature selection via coalitional game theory. *Neural Comput.*, 19(7):1939–1961, 2007. doi: 10.1162/NECO.2007.19.7.1939.

Ian Covert and Su-In Lee. Improving kernelshap: Practical shapley value estimation via linear regression. *CoRR*, abs/2012.01536, 2020.

Rahul Dubey and Arti Chandani. Application of machine learning in banking and finance: a bibliometric analysis. *Int. J. Data Anal. Tech. Strateg.*, 14(3), 2022. doi: 10.1504/ijdats.2022.128268.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–451, 2004. ISSN 00905364.

Munira Ferdous, Jui Debnath, and Narayan Ranjan Chakraborty. Machine learning algorithms in healthcare: A literature survey. In *11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020, Kharagpur, India, July 1-3, 2020*, pages 1–6. IEEE, 2020. doi: 10.1109/ICCCNT49239.2020.9225642.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.*, 20:177:1–177:81, 2019.

Christopher Frye, Colin Rowat, and Ilya Feige. Asymmetric shapley values: incorporating causal knowledge into model-agnostic explainability. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning, 2019.

Amirata Ghorbani and James Y. Zou. Neuron shapley: Discovering the responsible neurons. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Jeremy Goldwasser and Giles Hooker. Stabilizing estimates of shapley values with control variates. *CoRR*, abs/2310.07672, 2023. doi: 10.48550/ARXIV.2310.07672.

Jeremy Goldwasser, Will Fithian, and Giles Hooker. Gaussian rank verification, 2025. URL `https://arxiv.org/abs/2501.14142`.

Alireza Goli and Hatam Mohammadi. Developing a sustainable operational management system using hybrid Shapley value and Multimoora method: case study petrochemical supply chain. *Environment, Development and Sustainability: A Multidisciplinary Approach to the Theory and Practice of Sustainable Development*, 24(9):10540–10569, September 2022. doi: 10.1007/s10668-021-01844-.

Claus-Jochen Haake, Akemi Kashiwada, and Francis Edward Su. The shapley value of phylogenetic trees. *Journal of Mathematical Biology*, 56(4):479–497, September 2007. ISSN 1432-1416. doi: 10.1007/s00285-007-0126-2.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL `https://arxiv.org/abs/1503.02531`.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Giles Hooker, Lucas Mentch, and Siyu Zhou. Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, 31:1–16, 2021.

Kenneth Hung and William Fithian. Rank verification for exponential families. *The Annals of Statistics*, 47(2), April 2019. ISSN 0090-5364. doi: 10.1214/17-aos1634.

Shashank Kariyappa, Lefteris Tsepenekas, Fabien Lécué, and Daniele Magazzeni. Shap@k: Efficient and probably approximately correct (pac) identification of top-k features. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13068–13075, 2024. doi: 10.1609/aaai.v38i12.29205.

Patrick Kolpaczki, Viktor Bengs, and Eyke Hüllermeier. Identifying top-k players in cooperative games via shapley bandits. In Thomas Seidl, Michael Fromm, and Sandra Obermeier, editors, *Proceedings of the LWDA 2021 Workshops: FGWM, KDML, FGWI-BIA, and FGIR, Online, September 1-3, 2021*, volume 2993 of *CEUR Workshop Proceedings*, pages 133–144. CEUR-WS.org, 2021.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression, 2017.

Dan Ley, Saumitra Mishra, and Daniele Magazzeni. Global counterfactual explanations: Investigations, implementations and improvements, 2022. URL https://arxiv.org/abs/2204.06917.

Jiahui Li, Kun Kuang, Baoxiang Wang, Furui Liu, Long Chen, Fei Wu, and Jun Xiao. Shapley counterfactual credits for multi-agent reinforcement learning. In Feida Zhu, Beng Chin Ooi, and Chunyan Miao, editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 934–942. ACM, 2021. doi: 10.1145/3447548.3467420.

Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 17(4):319–330, October 2001. doi: 10.1002/asmb.446.

Zelei Liu, Yuanyuan Chen, Han Yu, Yang Liu, and Lizhen Cui. Gtg-shapley: Efficient and accurate participant contribution evaluation in federated learning. *ACM Trans. Intell. Syst. Technol.*, 13(4):60:1–60:21, 2022. doi: 10.1145/3501811.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4765–4774, 2017.

Scott M. Lundberg, Gabriel G. Erion, Hugh Chen, Alex J. DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.*, 2(1):56–67, 2020. doi: 10.1038/s42256-019-0138-9.

Varun Mandalapu, Lavanya Elluri, Piyush Vyas, and Nirmalya Roy. Crime prediction using machine learning and deep learning: A systematic review and future directions. *IEEE Access*, 11:60153–60170, 2023. doi: 10.1109/ACCESS.2023.3286344.

Lucas Mentch and Giles Hooker. Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research*, 17: 841–881, 2016. URL http://jmlr.org/papers/v17/14-168.html.

Rory Mitchell, Joshua Cooper, Eibe Frank, and Geoffrey Holmes. Sampling permutations for shapley value estimation. *J. Mach. Learn. Res.*, 23:43:1–43:46, 2022.

Christoph Molnar. *Interpretable Machine Learning*. 2 edition, 2022. URL https://christophm.github.io/interpretable-ml-book.

Ramasuri Narayanam and Y. Narahari. Determining the top-k nodes in social networks using the shapley value. In Lin Padgham, David C. Parkes, Jörg P. Müller, and Simon Parsons, editors, *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 3*, pages 1509–1512. IFAAMAS, 2008.

Bitya Neuhof and Yuval Benjamini. Confident feature ranking. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1468–1476. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/neuhof24a.html.

Peter C. O'Brien and Thomas R. Fleming. A multiple testing procedure for clinical trials. *Biometrics*, 35(3):549–556, 1979. ISSN 0006341X, 15410420.

R Peto, MC Pike, P Armitage, NE Breslow, DR Cox, SV Howard, N Mantel, K McPherson, J Peto, and PG Smith. Design and analysis of randomized clinical trials requiring prolonged observation of each patient. i. introduction and design. *Br J Cancer*, 34(6):585–612, Dec 1976. doi: 10.1038/bjc.1976.220.

Venetia Pliatsika, Joao Fonseca, Kateryna Akhynko, Ivan Shevchenko, and Julia Stoyanovich. Sharp: A novel feature importance framework for ranking, 2024. URL https://arxiv.org/abs/2401.16744.

Stuart J. Pocock. Group sequential methods in the design and analysis of clinical trials. *Biometrika*, 64(2):191–199, 1977. ISSN 00063444.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi, editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144. ACM, 2016. doi: 10.1145/2939672.2939778.

Benedek Rozemberczki and Rik Sarkar. The shapley value of classifiers in ensemble games. In Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong, editors, *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 1558–1567. ACM, 2021. doi: 10.1145/3459637.3482302.

Haim Shalit. The shapley value of regression portfolios. *Journal of Asset Management*, 21(6):506–512, 2020.

Lloyd Shapley. A value for n-person games, march 1952.

Carolin Strobl, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. *BMC Bioinformatics*, 9:307, 2008. doi: 10.1186/1471-2105-9-307. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-9-307.

Erik Strumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.*, 41(3):647–665, 2014. doi: 10.1007/S10115-013-0679-X.

Erik Strumbelj, Igor Kononenko, and Marko Robnik-Sikonja. Explaining instance classifications with interactions of subsets of feature values. *Data Knowl. Eng.*, 68(10):886–904, 2009. doi: 10.1016/J.DATAK.2009.01.004.

Jonathan Taylor and Robert J. Tibshirani. Statistical learning and selective inference. *Proceedings of the National Academy of Sciences*, 112(25):7629–7634, 2015. doi: 10.1073/pnas.1507583112. URL https://doi.org/10.1073/pnas.1507583112. This contribution is part of the special series of Inaugural Articles by members of the National Academy of Sciences elected in 2012. Contributed by Robert J. Tibshirani, May 7, 2015 (sent for review March 2, 2015; reviewed by Rollin Brant and John D. Storey).

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 00359246.

Ryan J. Tibshirani and Jonathan Taylor. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3), June 2011. ISSN 0090-5364. doi: 10.1214/11-aos878. URL http://dx.doi.org/10.1214/11-AOS878.

Ilse van der Linden, Hinda Haned, and Evangelos Kanoulas. Global aggregations of local explanations for black box models, 2019. URL https://arxiv.org/abs/1907.03039.

Isabella Verdinelli and Larry Wasserman. Feature importance: A closer look at shapley values and loco, 2023. URL https://arxiv.org/abs/2303.05981.

Abraham Wald. Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, 16:117–186, 1945.

Z. Xu, Z. Peng, L. Yang, and X. Chen. An improved shapley value method for a green supply chain income distribution mechanism. *International Journal of Environmental Research and Public Health*, 15(9):1976, Sep 2018. doi: 10.3390/ijerph15091976.

Bin Yu and Karl Kumbier. Veridical data science. *Proceedings of the National Academy of Sciences*, 117(8):3920–3929, February 2020. ISSN 1091-6490. doi: 10.1073/pnas.1901326117.

Kaifeng Zhao, Seyed Hanif Mahboobi, and Saeed R. Bagheri. Shapley value methods for attribution modeling in online advertising, 2018.

Zhengze Zhou, Giles Hooker, and Fei Wang. S-lime: Stabilized-lime for model explanation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '21. ACM, August 2021. doi: 10.1145/3447548.3467274.

A PROCEDURES 1 AND 2

Algorithm 3 Procedure 1 [Goldwasser et al., 2025]

Require: Significance level α, ordered data $X_1 > X_2 > \ldots > X_d$
Ensure: Integer $K \geq 0$ verifying the top-K population rankings are correct with probability at least $1 - \alpha$
  $K \leftarrow 0$ ▷ Number of verified ranks
  while $K < d - 1$ **do**
    $i \leftarrow K + 1$ ▷ Index of tested feature
    for $j \in [i + 1 : d]$ **do**
      Test $H_{0ij} : \{\mu_i < \mu_j \mid X_i > \max_{k>i} X_k\}$. ▷ Use original p-value p_{ij} or SPRT statistic T_{ij}
    end for
    if all nulls H_{0ij} reject **then** ▷ Equivalently, test H_{0i} with $p_i^* \leftarrow \max_j p_{ij}$ or $T_{i*}^n \leftarrow \min_j T_{ij}$ (SPRT)
      $K \leftarrow K + 1$
    else
      return K
    end if
  end while
  if $K = d - 1$ **then**
    $K \leftarrow d$
  end if
  return d

Algorithm 4 Procedure 2 [Goldwasser et al., 2025]

Require: Significance level α, ordered data $X_1 > X_2 > \ldots > X_d$
Ensure: Boolean whether top-K population set is correct with probability at least $1 - \alpha$
  for $i \in [1 : K]$ **do**
    for $j \in [K + 1 : d]$ **do**
      Test $H_{0ij} : \{\mu_i < \mu_j \mid X_i > \max_{k>K} X_k\}$. ▷ Use original p-value p_{ij} or SPRT statistic T_{ij}
    end for
    if some null H_{0ij} fails to reject **then** ▷ i.e. Test H_{0i} with $p_i^* \leftarrow \max_j p_{ij}$ or $T_{i*}^n \leftarrow \min_j T_{ij}$ (SPRT)
      return False ▷ Cannot verify i[th] rank belongs in top-K set
    end if
  end for
  return True

B TOP-K SET ALGORITHMS

Algorithms 5 and 6 stabilize the top K set. Like Algorithms 1 and 2, they employ the SPRT and resampling approaches, respectively.

C SPRT PROOF

Here we prove Theorem 1, presenting the likelihood ratio necessary for SPRT. Returning to the terminology of Section 3.2, let $X_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$ with independence and known σ_j; sort $X_1 > X_2 > \ldots X_d$.

Our objective is to verify feature i as having the largest mean in a set of random variables, given it has the largest observed value X_i. Given a valid test to do so, Procedure 1 iterates from $i = 1$ onwards until a failure to reject at $i = K + 1$. At each step, i is tested against all lower ranks. Procedure 2 conducts this test on $i = 1$ through K against features $j = K + 1, \ldots, d$. Because rankings within the top-K set do not matter, it only conditions on $\{X_1 > \max_{j>K} X_j\}$.

Algorithm 5 Set Stability via SPRT

Require: Desired set size $K > 0$, error rate $\alpha \in [0, 1]$, total sample budget n_{max}, number of samples between tests n_{btwn}, samples between tests n_{btwn}.
Ensure: Estimated feature importances $\hat{\phi}_1, \ldots, \hat{\phi}_d$ whose top-K set is correct with probability $\geq 1 - \alpha$.
  $n \leftarrow 0$
  while $n < n_{max}$ **do**
    Generate n_{btwn} new samples
    $n \leftarrow n + n_{btwn}$
    $\hat{\phi}_n \leftarrow$ Feature importances fit on all n samples
    $\hat{\Sigma}_n \leftarrow$ Variances of all feature importances
    Status $\leftarrow$ Procedure 2 (Alg. 4) on $\hat{\phi}_n$ & $\hat{\Sigma}_n$, rejecting H_{0ij} if $T_{ij} > \frac{1-\beta}{\alpha}$
    if Status==“Verified” **then**
      return $\hat{\phi}$, “Verified”
    end if
  end while
  return $\hat{\phi}$, “Failed to verify”

Algorithm 6 Set Stability via Resampling

Require: Desired rankings $K > 0$, error rate $\alpha \in [0, 1]$, per-feature sample budget n_{max}, intial per-feature samples n_{init}, buffer $c \geq 1$.
Ensure: Estimated feature importances $\hat{\phi}_1, \ldots, \hat{\phi}_d$ whose top-K rankings are correct with probability $\geq 1 - \alpha$.
  $\hat{\phi} \leftarrow$ Feature importances fit on n_{init} samples
  $\hat{\Sigma} \leftarrow$ Variances of all feature importances
  while $n_j < n_{max}\ \forall j$ **do**
    Status $\leftarrow$ Procedure 2 (Alg. 4) on $\hat{\phi}_n$ & $\hat{\Sigma}_n$, rejecting H_{0ij} if $T_{ij} > \frac{1-\beta}{\alpha}$
    if Status==“Verified” **then**
      return $\hat{\phi}$, “Verified”
    else
      $n'_i, n'_j \leftarrow$ Estimated samples to reject, Eq. (10) or (11).
      $n'_i, n'_j \leftarrow \min(\lceil cn'_i \rceil, n_{max}), \min(\lceil cn'_j \rceil, n_{max})$
      $\hat{\phi}_i, \hat{\phi}_j \leftarrow$ Attributions fit on n'_i, n'_j samples
      $\hat{\Sigma}_i, \hat{\Sigma}_j \leftarrow$ Variances of feature importances
    end if
  end while
  return $\hat{\phi}$, “Failed to verify”

Without loss of generality, let $i = 1$. Define the null hypothesis

$$\begin{aligned} & H_0 : \mu_1 \textit{ not} \text{ best} \mid X_1 \text{ wins} \\ \iff \quad & \mu_1 \leq \max_{j>1} \mu_j \mid X_1 > \max_{k>1} X_k \\ \iff \quad & \bigcup_{j>1} \underbrace{\{\mu_1 \leq \mu_j \mid X_1 > \max_{k>1} X_k\}}_{H_{0j}}. \end{aligned}$$

Analogously define the alternate hypothesis

$$\begin{aligned} & H_1 : \mu_1 \textit{ is} \text{ best} \mid X_1 \text{ wins} \\ \iff \quad & \mu_1 > \max_{j>1} \mu_j \mid X_1 > \max_{k>1} X_k \\ \iff \quad & \bigcap_{j>1} \underbrace{\{\mu_1 > \mu_j \mid X_1 > \max_{k>1} X_k\}}_{H_{1j}}. \end{aligned}$$

A classical result states that a valid p-value for a union null hypothesis is the maximum p-value of its constituent nulls [Berger, 1982]. Therefore it suffices to construct valid tests for all H_{0j}; rejecting H_0 when all tests reject is a valid level-α procedure.

Ostensibly, SPRT tests H_{0j} with a likelihood ratio T_{1j}, not a p-value. However, T_{1j} may be interpreted probabilistically: Because SPRT is a valid test, the probability under the null that T exceeds threshold $\gamma_1 = \frac{1-\beta}{\alpha}$ is at most α. Therefore the SPRT p-value is conceivably the probability of randomly obtaining a higher T than the observed quantity, given that the null is true. Amongst all nulls H_{0j}, the p-value is highest for the lowest T_{1j}. So it suffices to compute T_{1j} $\forall j > 1$, and reject H_0 if the smallest $T_{1j} \geq \gamma_1$. (Equivalently, T_{1j} must exceed γ_1 for all j.)

Accepting the null is not a concern for reasonable α and β. For example, when $\alpha = 0.05$ and $\beta = 0.2$, the null is accepted when T_{1j} is below roughly 0.21. In practice, however, T_{1j} should never even be below 1. Conditioning on the selection event that $X_1 > X_j$, the data will always be likelier when the means mirror this discrepancy with $\Delta > 0$.

Let $\Delta = \mu_1 - \mu_j$. Under the null $\Delta \leq 0$, and under the alternate $\Delta > 0$. Also let $A_1 = \{X_1 > \max_{k>1} X_k\}$, the event that X_1 wins. The j^{th} likelihood ratio is then

$$T_{1j} = \frac{\max_{\Delta>0} \mathbb{P}_{\mu_1-\mu_j=\Delta}(X_1, \ldots, X_d \mid A_1)}{\max_{\Delta\leq 0} \mathbb{P}_{\mu_1-\mu_j=\Delta}(X_1, \ldots, X_d \mid A_1)}.$$

Following the same argument of Goldwasser et al. [2025], these probabilities may be simplified by conditioning on additional variables. The ensuing test will be valid at level-α for all possible realizations; as a result, the unconditional test would still be valid after marginalizing them out.

In particular, we condition on the values of non-tested variables X_{-1j}. We also condition on $U(X) = \frac{X_1}{\sigma_1^2} + \frac{X_j}{\sigma_j^2}$ taking its realized value, $u = \frac{x_1}{\sigma_1^2} + \frac{x_j}{\sigma_j^2}$. The purpose of this is to remove the influence of a nuisance parameter, which would otherwise prohibit inference on Δ. We refer the reader to the manuscript for greater detail. The ratio that is tested instead is

$$T_{1j} = \frac{\max_{\Delta>0} \mathbb{P}_{\mu_1-\mu_j=\Delta}(X_1 \mid A_1, X_{-1j}, U)}{\max_{\Delta\leq 0} \mathbb{P}_{\mu_1-\mu_j=\Delta}(X_1 \mid A_1, X_{-1j}, U)}. \tag{12}$$

Ignoring constant factors, the conditional likelihood is proportional to the following:

$$X_1 \mid \{A_1, X_{k\neq 1,j}, U\} \propto \exp\left[-\left(\frac{1}{2\sigma_1^2} + \frac{\sigma_j^2}{2\sigma_1^4}\right) X_1^2 + \left(\frac{\sigma_j^2 u + \Delta}{\sigma_1^2}\right) X_1\right] \mathbf{1}_{A_1}. \tag{13}$$

Goldwasser et al. [2025] showed that this likelihood under the null is maximized at $\Delta = 0$. The conditional distribution for X_1 that results is a truncated normal. Its parameters are defined in Section 3.2: Mean $\bar{\mu}_{1j} = \frac{\sigma_j^2 x_1 + \sigma_1^2 x_j}{\sigma_1^2 + \sigma_j^2}$, variance

$\bar{\sigma}_{1j}^2 = \frac{\sigma_1^4}{\sigma_1^2+\sigma_j^2}$, and truncation at $\bar{\eta}_{1j} = \max(\bar{\mu}_{1j}, \max_{k\neq 1,j} x_k)$. The denominator of (12) is thus

$$\frac{\phi(\frac{x_1-\bar{\mu}_{1j}}{\bar{\sigma}_{1j}})\frac{1}{\bar{\sigma}_{1j}}}{1-\Phi(\frac{\bar{\eta}_{1j}-\bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}. \tag{14}$$

For arbitrary Δ, the conditional likelihood is also a truncated normal. Its proportional density (13) may be rearranged to complete the square.

$$\begin{aligned}
X_1 \mid \{A_1, X_{k\neq 1,j}, U\} &\propto \exp\left[\left(-\frac{\sigma_1^2+\sigma_j^2}{2\sigma_1^4}\right)X_1^2 + \left(\frac{\sigma_j^2 u + \Delta}{\sigma_1^2}\right)X_1\right]\mathbf{1}_{A_1} \\
&\propto \exp\left[-\frac{\sigma_1^2+\sigma_j^2}{2\sigma_1^4}\left(X_1^2 - \frac{2\sigma_1^2(\sigma_j^2 u+\Delta)}{\sigma_1^2+\sigma_j^2}X_1\right)\right]\mathbf{1}_{A_1} \\
&\propto \exp\left[-\frac{\sigma_1^2+\sigma_j^2}{2\sigma_1^4}\left(X_1 - \frac{\sigma_1^2\sigma_j^2 u + \sigma_1^2\Delta}{\sigma_1^2+\sigma_j^2}\right)^2\right]\mathbf{1}_{A_1},
\end{aligned}$$

This is a truncated normal distribution again with variance $\bar{\sigma}_{1j}^2$ and truncated at $\bar{\eta}_{1j}$. Its mean decomposes to the following:

$$\tilde{\mu}_{1j}^{\Delta} = \frac{\sigma_1^2\sigma_j^2 u + \sigma_1^2\Delta}{\sigma_1^2+\sigma_j^2} = \frac{\sigma_j^2 x_1 + \sigma_1^2 x_j + \sigma_1^2\Delta}{\sigma_1^2+\sigma_j^2}.$$

The numerator of Equation (12) is

$$\max_{\Delta>0} \frac{\phi(\frac{x_i-\tilde{\mu}_{1j}^{\Delta}}{\bar{\sigma}_{1j}})\frac{1}{\bar{\sigma}_{1j}}}{1-\Phi(\frac{\bar{\eta}_{1j}-\tilde{\mu}_{1j}^{\Delta}}{\bar{\sigma}_{1j}})}. \tag{15}$$

This expression cannot be manipulated into a closed-form solution for the optimal Δ, due to the integral in the denominator. It could be optimized numerically, but doing so would be relatively slow. Furthermore, the SPRT algorithm must compute T_{ij} for many i, j, and n, so this is impractical. Instead, a practical solution optimizes only the numerator of (15). The resulting choice of Δ may be slightly sub-optimal, producing a test that is barely more conservative.

$$\begin{aligned}
\Delta^* &= \arg\max_{\Delta>0} \phi(\frac{x_i - \tilde{\mu}_{1j}^{\Delta}}{\bar{\sigma}_{1j}}) \\
&= \arg\max_{\Delta>0} \exp\left[-\frac{(x_1-\tilde{\mu}_{1j}^{\Delta})^2}{2\bar{\sigma}_{1j}^2}\right] \\
&= \arg\min_{\Delta>0} (x_1 - \tilde{\mu}_{1j}^{\Delta})^2 \\
&= \arg\min_{\Delta>0} \left(\frac{\sigma_1^2 x_1 + \sigma_j^2 x_1}{\sigma_1^2+\sigma_j^2} - \frac{\sigma_j^2 x_1 + \sigma_1^2 x_j + \sigma_1^2\Delta}{\sigma_1^2+\sigma_j^2}\right)^2 \\
&= \arg\min_{\Delta>0} \left(\frac{\sigma_1^2}{\sigma_1^2+\sigma_j^2}(x_1 + x_j - \Delta)\right)^2 \\
&= x_1 - x_j.
\end{aligned}$$

This is an intuitive choice: The optimal difference in means Δ^* is the observed difference, $x_1 - x_j$. Furthermore, the conditional mean of X_1 under the alternate is the observed value x_1:

$$\tilde{\mu}_{1j}^{\Delta^*} = \frac{\sigma_j^2 x_1 + \sigma_1^2 x_j + \sigma_1^2 x_1 - \sigma_1^2 x_j}{\sigma_1^2+\sigma_j^2} = \frac{\sigma_j^2 x_1 + \sigma_1^2 x_1}{\sigma_1^2+\sigma_j^2} = x_1.$$

Therefore the alternate likelihood (15) is

$$\frac{\phi(0)\frac{1}{\bar{\sigma}_{1j}}}{1-\Phi(\frac{\bar{\eta}_{1j}-x_1}{\bar{\sigma}_{1j}})}.$$

Composing this with (14) yields

$$T_{1j} = \left[\frac{\phi(0)}{1-\Phi(\frac{\bar{\eta}_{1j}-x_1}{\bar{\sigma}_{1j}})}\right]\left[\frac{\phi(\frac{x_i-\bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}{1-\Phi(\frac{\bar{\eta}_{1j}-\bar{\mu}_{1j}}{\bar{\sigma}_{1j}})}\right]^{-1}.$$

Finally, the test statistic for H_0 is $\min_{j>1} T_{1j}$.

D CORRELATED TESTING

The methodology we have presented assumes estimated ranks are independent. This assumption is necessary in order to apply the hypothesis tests in Equation (5) and Theorem 1. As described in Section 4, however, feature importances may be weakly correlated with one another. Violating this assumption is a limitation of our methodology. Fortunately, this limitation does not seem to affect practice, as the empirical FWERs of KernelSHAP are all controlled at level α.

Nevertheless, it turns out that alternative tests can be constructed which take correlation into account. Unfortunately, they require making assumptions that are at least as concerning as indepedence itself.

To test each hypothesis H_{0ij} (7), it is necessary to construct a density that isolates the parameter of interest, $\mu_i - \mu_j$. The joint density of all observations $X_{1:d}$ has a host of "nuisance parameters" — other unknown quantities that prohibit inference on $\mu_i - \mu_j$. To remove their inference, a requisite step conditions on the non-tested variables $X_k = x_k$, where $k \neq i, j$. When all variables are independent, the nuisance parameters μ_k disappear in the density of $X_{i,j}|X_{k\neq i,j}$. The remaining density can be rewritten in terms of $\mu_i - \mu_j$ and a final nuisance parameter, μ_j.

When variables are correlated, the conditional distribution $X_{i,j}|X_{k\neq i,j}$ does *not* eliminate the nuisance parameters $\mu_{k\neq i,j}$. This is because the conditional means $\mathbb{E}[X_{i,j} \mid X_{k\neq i,j}]$ depend on μ_k. To see this, let $X \sim \mathcal{N}(\mu, \Sigma)$, with indices $a = \{i, j\}$ and $b = [d]\backslash\{i, j\}$. Express the unconditional mean and variance as

$$\mu = \begin{bmatrix}\mu_a \\ \mu_b\end{bmatrix} \quad \text{and} \quad \Sigma = \begin{bmatrix}\Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb}\end{bmatrix}.$$

Classical results state the conditional distribution is normal with means and variance

$$\mu_{a|b} = \mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(x_b - \mu_b),$$
$$\Sigma_{a|b} = \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}$$

Thus, the density of $X_{i,j}|X_{k\neq i,j}$ still contains all means $\mu_{k\neq i,j}$, vectorized in μ_b. These nuisance parameters prohibit inference on $\mu_i - \mu_j$.

A strong assumption enables us to circumvent this restriction. The conditional means μ_i have their own ranking: Conditioned on X_k, we may compare $\mu_{i|b}$ and $\mu_{j|b}$. For all i and j, we may assume the unconditional ordering of μ_i and μ_j is equal to the conditional ordering of $\mu_{i|b}$ and $\mu_{j|b}$. When this holds, a valid test of H_{0ij} performs inference on the null that $\mu_{i|b} - \mu_{j|b}$.

It is possible to construct a test on these conditional means that takes their correlation ρ_{ij} into account, using the conditional variance $\Sigma_{a|b}$. The argument to do so is identical to that of Goldwasser et al. [2025]:

1. Express the density in terms of the difference in means $\mu_{i|b} - \mu_{j|b}$ and a single nuisance parameter.
2. Conditioning on the ancillary statistic removes the nuisance parameter, and expresses the distribution in terms of X_i.
3. Apply the null hypothesis, under which $\mu_{i|b} - \mu_{j|b} = 0$.
4. Completing the square reveals the mean and variance of a truncated normal distribution.
5. Identify the truncation event for X_i winning in terms of $X_{k\neq i,j}$ and the ancillary statistic.

This approach may be followed for top-K SPRT algorithms. The optimal densities under the null and alternative can be calculated, producing a likelihood ratio akin to Theorem (1).

We do not include these results and their proofs because the assumption of mean order preservation is very strong, and may be easily violated. For example, consider the simple three-variable case case with $\mu_1 > \mu_2 > \mu_3$. X_1 is independent of the other variables, but positive correlation exists between X_2 and X_3. Then whenever X_3 is observed to be relatively high, the conditional mean of X_2 is raised, perhaps higher than that of X_1.

Table 4: Model Accuracies on Benchmark Datasets

Dataset	Accuracy
Adult	82.0%
Bank	82.8%
BRCA	77.0%
Credit	72.0%
WBC	91.8%

In practice, it is impossible to know whether observed values are unusually high or low relative to their mean. Therefore this assumption is impossible to verify. Future work could explore conditions under which it holds. It is possible that this correlated test of conditional means is preferable to assuming independence, but further work is necessary to demonstrate its validity.

E ADDITIONAL SHAP EXPERIMENTS

E.1 STABLESHAP EFFICIENCY

We investigated how the adaptive allocation strategy of StableSHAP improved stability relative to Shapley Sampling. To do so, we compared their performance given equal computational budgets. The algorithms explained 30 test set predictions of a neural network trained on the Adult dataset. On each input, we ran StableSHAP (Alg. 2) to convergence 50 times, tracking the average number of total samples $n = \sum_j n_j$ (Equation (2)). Then, we ran Shapley Sampling 50 times, evenly distributing the same total number of samples across the d features for each input. We repeated this procedure for $K = 2$ and $K = 5$, as well as $\alpha = 0.1$ and $\alpha = 0.2$.

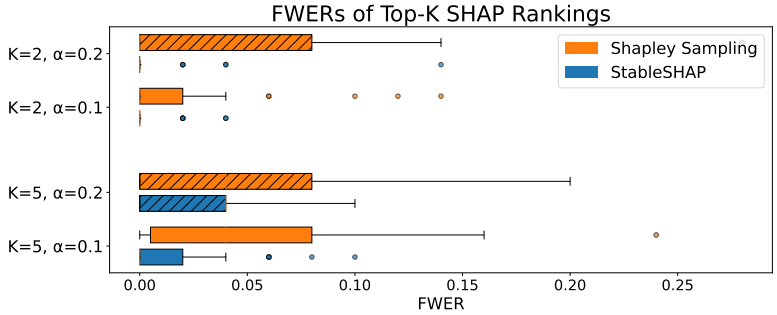

Figure 5: Stability of StableSHAP and Shapley Sampling, Equal Computation.

Figure 5 displays empirical error rates on top-K Shapley rankings. For all input data points, ranks K, and tolerances α, StableSHAP successfully achieved FWER control, as anticipated. In contrast, rankings from Shapley Sampling were considerably less stable. In all settings, their error rates were higher than StableSHAP's, often exceeding α. This indicates that StableSHAP is more efficient than typical Shapley Sampling methods, which allocate samples equally across features.

E.2 VARIANCES

Figure 6 shows the variances of the SHAP estimates in Figure 3. They follow the same order by magnitude, with dashed line at $K = 5$. While lower SHAP values generally have smaller variance, the 8^{th} and 11^{th} ranks have high variance. This accounts for the iterations in Figure 3 for which Occupation and Hours of Week require more than 100 samples.

F EXPERIMENTAL DETAILS

For the models, we trained two-layer feedforward neural networks in Pytorch for 20 epochs. The hidden layer had 50 neurons. To discourage overfitting to the more common class, we batched the two classes in sampling each batch from the

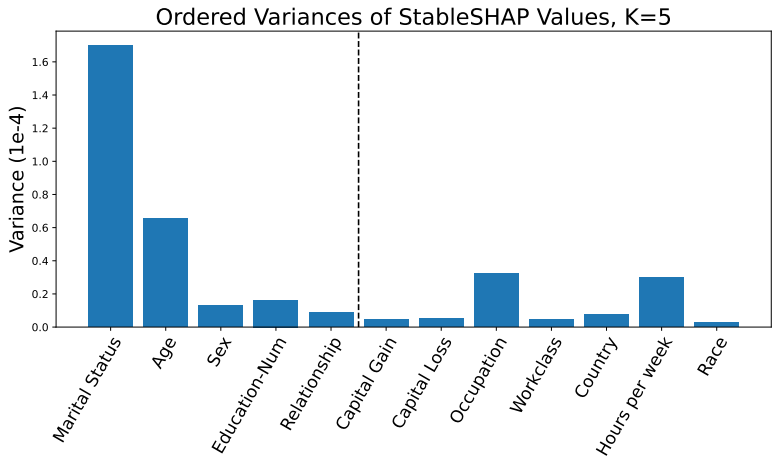

Figure 6: Variances of StableSHAP values displayed in Figure 3

training data. We trained networks for 20 epochs in batches of 32 samples, using the Adam optimizer with a learning rate of 0.001.

Datasets for experimental results were pulled from the UCI Machine Learning repository. We random split the data 75%/25% into training and test sets using `sklearn`. Accuracies of the five model are contained in Table 4

Computation was conducted using a slurm partition. Each entry in the tables was run with a separate job, though not with internal parallelization. In constructing Tables 2 and 3, we only considered input data points for which our algorithms successfully rejected all K tests on at least 90% of the iterations.

Our experiments treat the most common observed top-K as ground truth. In theory, this could be wrong. Fortunately, however, it is straightforward to show that the probability this surrogate ground truth is incorrect is essentially 0.

To do so, recall that Table 1 shows the most common top-K always occurs with frequency exceeding $1-\alpha$, according to the theory. To be conservative, suppose $\alpha = 0.2$, and the most common top-K appears $50 \times (1-\alpha) = 40$ times. Now suppose the most common top-K is actually incorrect. Then the probability of observing the *wrong* top-K at least 40 times is upper-bounded by the binomial probability

$$\mathbb{P}(\text{observe top-}K\text{ at least 40x} \mid \text{it is wrong}) \leq \sum_{s=40}^{50} \binom{n}{s} 0.2^s (1-0.2)^{50-s} \approx 1.29 \times 10^{-19}.$$

With this infinitesimal probability in mind, we can safely conclude that our "proxy" ground truth is indeed correct. Thus, synthetic experiments with exhaustively computed ground truth SHAP values is unnecessary. Doing so would require $\mathcal{O}(2^d)$ samples for each input, which is computationally prohibitive in high dimensions.

F.1 SHAP

Our experiments ranked the absolute values of SHAP estimates. The sum of all SHAP values for input x is $f(x) - E[f(X)]$, which may be positive or negative. In the latter case, the most important features are the ones with the largest negative values. Further, an individual sample may have features with both large positive and negative SHAP values. An alternative approach would be flip the sign of the SHAP values if their sum is negative, then rank these preprocessed values.

Because absolute values have non-negative support, their distribution is no longer Gaussian, but rather a folded normal. To avoid this, we used the same normal distribution, but flipped the sign if the SHAP estimate was negative. In theory, the true mean could then be the opposite sign as its observation. In essence, this does not consider the distribution of the absolute SHAP values. Instead, it merely establishes a ranking based on their absolute value. This subtle distinction may have virtually no practical difference, as the mass of the highest-ranking features may be almost entirely positive or negative.

Shapley Sampling (2) takes the average of n values of $v(S_j^i \cup \{j\}) - v(S_j^i)$. To select subsets $S_j^i \subseteq [d] \backslash \{j\}$, each iteration randomly permutes the d features, then takes the features that precede j. This is formalized in Algorithm 7.

Algorithm 7 Shapley Sampling

Require: Input x, dataset X, Shapley value function $v(S)$, player $j \in [d]$, number of samples $n > 0$
Ensure: $\hat{\phi}_j(v)$, an unbiased estimate of $\phi_j(v)$
  $\hat{\phi}_j \leftarrow 0$
  for $i = 1$ **to** n **do**
    $\pi_i \leftarrow$ random permutation of $[d]$
    $S_j^i \leftarrow$ elements of π_i before j
    $\hat{\phi}_j \leftarrow \hat{\phi}_j + v(S_j^i \cup \{j\}) - v(S_j^i)$
  end for
  $\hat{\phi}_j(v) \leftarrow \frac{1}{n}\hat{\phi}_j(v)$

To generate Figure 1, we used results from the experiment generating Figure 5. This compared StableSHAP and Shapley Sampling on the Adult Census Income dataset. We chose an input for which Shapley Sampling was less stable. On the same input, we ran SPRT-SHAP, LIME, and the adjusted S-LIME procedure described in Appendix F.2.

Our SHAP estimates computed $v(S) = \hat{\mathbb{E}}[f(X)|X_S = x_s]$ with 10 samples per subset S. Features were sampled from their marginal distributions.

We implemented KernelSHAP as described in Covert and Lee [2020]. The same paper introduces a method for estimating its variance. However, Goldwasser and Hooker [2023] found that a considerably more stable approach was to compute a bootstrap estimate. This method takes the covariance matrix of Shapley estimates fit on bootstrapped versions of the data. In our experiments, we used this approach with 250 bootstrapped samples.

Our retrospective experiments (Table 1) used $2d + 2048$ samples of subsets S. This is the default for KernelSHAP in the `shap` package. Note this is more than enough to guarantee convergence by CLT. Concerns of coarse approximation would only be legitimate under perhaps $n = 30$ random samples. For each of 30 input data points, the empirical FWER was computed over 50 runs. The table displays the maximum FWER over these 30 inputs.

Similarly, our top-K experiments computed empirical FWERs over 50 runs, taking the maximum across 30 inputs. NA values in Table 2 correspond to settings of method, dataset, K, and α that proved incapable of converging with reasonable frequency. Specifically, we ceased computation after our methods failed to converge on fewer than 10% of inputs after at least 10 attempts.

StableSHAP used 100 initial samples per feature, and a maximum of 10,000. n_i' and n_j' were scaled according to their relative variances, following Equation (11). SPRT-SHAP iterations were capped at 50,000 samples. The SPRT test was conducted every 1,000 iterations. For the Type II error rate, we set $\beta = 0.2$. This indicates the null will be accepted at most 20% of the time. (In practice, the null is never accepted.)

StableSHAP and SPRT-SHAP do not necessarily use the same inputs in Table 2. To construct Figure 4, we explicitly selected 30 inputs for which both top-2 ranking algorithms converged. For BRCA, there were not enough test set samples for which SPRT-SHAP stabilized the top-2 ranks under reasonable α; therefore we used training samples on this dataset.

F.2 LIME

Our LIME procedure can be implemented using Zhengze Zhou's `slime` repository almost entirely off-the-shelf. In our experiments, we added only 3 lines of code, which flag when the maximum number of samples have been used but not all hypothesis tests reject.

S-LIME generates enough samples such that each highest-ranking feature beats its runner-up with probability exceeding $1 - \alpha$, in a manner similar to Algorithm 2.

To verify the winner, Test (6) takes the highest p-value p_{1j}. Comparing to the runner-up is equivalent to a level-$\alpha/2$ Z-test. Unless all variances are equal, ranks below the runner-up may have higher p-values [Hung and Fithian, 2019]. To run S-LIME in this context, it is necessary to assume the runner-up always has the highest p-value.

Making the requisite assumptions, running S-LIME at level $\alpha/2K$ for each test is therefore a valid level-α ranking procedure for LIME with K-Lasso. The factors of 2 and K account for selection and multiple testing, respectively.

To control FWER at level α, the "alpha" parameter passed to `slime()` should be $\frac{\alpha}{2K}$. We used 1,000 initial samples, and a maximum of 100,000. Another parameter, "tol," denotes the tolerance level of the hypothesis tests. Setting $tol = 0$ corresponds to the algorithm in the paper, and is of course a viable option.

The creators of the package found that having a small positive tolerance could yield comparable results with considerably greater efficiency. In our experiments, we set $tol = 10^{-4}$, which ran more quickly while controlling the FWER at level α. While higher values of *tol* made S-LIME run more quickly, it also resulted in rare instances in which the algorithm allegedly converged without actually controlling the FWER. The arbitrary nature of this choice is a legitimate limitation of the S-LIME method. That said, our experimental errors are always below α, and more a conservative approach could set $tol = 0$.