Feature Instability Search for Multi-Way Explainability

Kelly Sean
Jiang Meng

1 University of Notre Dame
skelly26@alumni.nd.edu, mjiang2@nd.edu

Abstract
Deep neural networks (DNNs) and other black box functions (BBFs) are notoriously difficult to interpret given their numerous layers of nonlinear functions and weights, which calls for methods to retrospectively explain the model’s output. Trust, and therefore explainability of the model’s output is essential if used in safety-critical applications, where it is crucial that decisions derived from the model reflect a dependence on contextually meaningful features. Current explainability frameworks (1) Lack a reliable quantitative definition of explainability, (2) Aren’t evaluated on a true ground truth measure, & (3) Fail to account for multi-way feature interactions. This paper proposes the concept of feature instability as a proxy for the importance of a given input parameter’s features in determining the output of a BBF, where instability refers to the distance a BBF’s input feature must change to alter the output of the BBF. We propose DFEST, an approach to quantify the influence of multi-way feature interactions in the output of a low dimensional BBF model. These proposed methods define a completely synthetic ground truth explainability, which have not been previously conceptualized, and estimate ground truth feature interaction explainability at scale through informed outside-in search. Repository: https://github.com/spkell/DFEST-Explainability

Introduction
Deep neural networks (DNNs) today outperform humans in many tasks by learning to extract relevant information from data. However, due to their inherent complexity of high-dimensional mathematics, neural connections, layers, the process in which their architectures are defined, and how the training process converges, deep learning models are often referred to as black boxes lacking human interpretability. As a result, they are often avoided in safety-critical systems like health diagnosis(Fink et al. 2018),(Chander et al. 2018). Aside from model trustworthiness, non-interpretable models risk adversarial attack vectors through noise identified in model parameters(Samuel et al. 2021a). We will refer to arbitrary binary prediction black box functions (BBFs) throughout this paper in place of a specific type of black box model such as MLPs, DNNs, or other non-linear neural networks, although a logistic regression is demonstrated in this paper’s evaluation section for simplicity.

In addition to generating the predictions produced by BBFs with high fidelity, it is equally important to understand the reasons behind the predictions, as 2 models can come to the same conclusion with drastically different reasoning. To address this, the study of Trusted AI(Polak and Krzanowski 2021)(Cohen et al. 2019) examines the importance of fairness, robustness, explainability, transparency, accountability, and value alignment in AI development.

Explainable AI (XAI) XAI(Molnar 2022) provides the means for technical trust, solvable through rationality. Frameworks of XAI are categorized in several ways. (1) An intrinsic method typically restricts the complexity of the model to enable interpretability, such as the intrinsic interpretability of decision trees(Freitas 2014), whereas post-hoc methods examine mappings of input to output post training. (2) Model specific methods interpret the weights of the model to formulate explainability, while model agnostic methods use post-hoc analysis, with no access to model parameters. Lastly, (3) Global interpretability attempts to com-
prehend the whole model, which are nearly un-interpretable by humans for models with more than 4-5 parameters due to constraints of our visual system (Lipton 2016). Local explainability examines specific BBF input-output pairs, which closely reflects small groups of feature contributions that may be overlooked in a global interpretation.

Related Work

Researchers have worked on various model-interpretable tools, which includes but are not limited to LIME (Ribeiro, Singh, and Guestrin 2016a), DLIME (Zafar and Khan 2019), Eli5 (Fan et al. 2019), SHAP (Lundberg and Lee 2017), LEN (Ciravegna et al. 2021), counterfactual (Verma, Dicker- son, and Hines 2020), and influence functions (Hines et al. 2022). LIME explains the prediction of a classifier by learning an interpretable model locally around the prediction (Ribeiro, Singh, and Guestrin 2016a). Furthermore, Shapley Additive exPlanations (SHAP), are a unified approach for interpreting predictions, in which they use game theoretic approach to explain the output of a given model by assigning each feature with an importance score using shapley values (Lundberg and Lee 2017). Random Forest Importances (Tsang, Cheng, and Liu 2017). Random Forest Importances (Li et al. 2019) are considered to approximate the ground truth of one-way feature explainability through a permutation importance mechanism, similarly to LIME (Ribeiro, Singh, and Guestrin 2016a) and SHAP (Lundberg and Lee 2017). Counterfactual approaches perturb the BBF input by iteratively removing input features ad-hoc in search of a changed model output. Influence functions (Adler et al. 2016) approach interpretability by adjusting training data of model to measure a change in a model’s output, to calculate influence $I_{loss}(x_{test})$ of a perturbed training input on the model’s loss function.

Counterfactual Explainability

Counterfactual XAI can be defined by methods that “explain a model’s prediction by assigning credit to each input feature based on how much it influenced the prediction” (Janizek, Sturmfels, and Lee 2020), which implicitly includes feature interactions. Counterfactual approaches to interpretability such as SHAP may be closer aligned to reliability and robustness, as they scrutinize model performance in the face of parameter & input variation. These methods explore the representation space of a model after training to shed light on the model’s decision making.

Evaluation of Interpretability

Some claim that if a system is useful in a practical application, it implies that the system is useful to be interpretable, while it does need to be functionally evaluable to be interpretable. Thus, we should determine a functional basis for interpretability to vet system performance in human application judged uses (Guidotti 2021).

Current XAI methods lack a reliable quantitative definition of explainability, and are instead typically evaluated on the basis of qualitative methods such as predictability, reliability, faithfulness, and consistency. Current XAI methods largely ignore the deep non-linear interactions of BBFs that result in supra-additive (Berthoud 2013) feature interactions, s.t. the importance of features in producing a BBF's output $I(f_1) + I(f_2) < I(f_1, f_2)$. Friedman’s H-Statistic (Inglis, Parnell, and Hurley 2021) hints at the study of multi-way feature interactions, however cannot practically scale past 2-way interactions due to complexity constraints. This phenomena is clearly described in Bayesian statistics with causal knowledge (Lad 1999).

Scope

In this work, we consider the following research questions:

- Can the performance of explainability frameworks be...
evaluated against a quantifiable ground truth explainability based on a local search of a model’s input and output in representational decision space?

• Can supra-additive feature interactions be represented and searched for in local representation space?

• Can an informed search on the output decision space of a BBF model generate more expressive and accurate explanations than state-of-the-art explainability frameworks, with high order feature interactions?

To address these questions, we propose a novel post-hoc diagnostic approach, namely Feature Stability Descent and Tensor Search for Explainability (DFEST) to quantify the importance of multi-way feature interactions in the output of BBFs with continuous features. DFEST introduces:

• The concept of feature (in)stability as a measure of explainability of the output of a model;

• A synthetic model with predetermined ground truth multi-way feature explainability to evaluate explainability frameworks;

• An informed stability descent based search algorithm as an attempt to quantifying multi-way feature stability, or importance, for a given recommendation;

• And a feature importance ranking evaluation loss function capable of comparing one-way feature explainability frameworks (LIME, SHAP) to more expressive frameworks (DFEST).

Problem Definition

In this section, we introduce the problem definition, and describe our novel approach, DFEST, in explaining the results generated by BBFs, while leveraging multi-way feature interactions. Moreover, we provide a detailed explanation of feature instability, as we posit the concept as one of the most promising quantifiable measures of explainability.

Problem Definition: Current methods to explain the most important features for a BBF’s output are unable to account for multi-way feature interactions, and lack a reliable ground truth metric to evaluate XAI frameworks.

Feature Instability

Let \( \hat{y} = BBF(x_{\text{train}}) \) denote a generic BBF. Let \( x_s = f_1, \ldots, f_n \) be an n-dimensional vector, denote the input data to a BBF, composed of \( n \) continuous numerical features \( f_i \), from which we want to explain a specific BBF output \( y_s \), s.t. we are blind to the internal BBF parameters. Feature instability is defined as the minimum feature-scaled distance \( \Delta'(x_s, x_q) \) in representational decision space \( \mathbb{R}^n \) between \( x_s \) and another location \( x_q \in \mathbb{R}^n \) s.t. \( y_s \neq y_q \) (Fig. 1). Further, the dominant feature instability of \( f_1 \) over \( f_2 \) in Fig. 1 presents a logical, quantifiable measure of \( f_1 \)'s importance in explaining how the model produces a prediction.

\[
\text{k-way Feature Instability} = \frac{1}{\text{Distance}} = \frac{1}{\Delta'(x_s, x_q)} \quad (1)
\]

Multi-way feature interaction can thus be defined through feature instability, where small perturbations in multiple features \( f \in x_s \) can result in a different \( \hat{y} \), and \( k \) denotes the number of important features involved in the feature interaction. Feature instability can function as a direct quantifiable metric for explainability, as the path of least perturbations to a differential \( \hat{y} \) indicates the features upon which the BBF most readily changes its output. In other words, feature instability is the minimum distance to a decision boundary for a set of input features.

Following from this definition, a continuous feature \( f_i \) in which a small change in \( f_i \) and no others result in a large \( \delta \hat{y} \) implies that \( f_i \) is highly unstable for the query \( \hat{y} = BBF(x_s) \). On the contrary, if \( f_i \) requires a much larger magnitude of change to encounter a solution than other feature dimensions, then \( f_i \) can be said to be stable w.r.t. \( x_s \) and the respective BBF.

Feature Instability Discussion

Feature instability is closely related to faithfulness & permutation importance(Zhou et al. 2021), as well as sensitivity & infidelity in relation to saliency expectations, which seeks to minimize explanation infidelity by measuring the difference in function values after significant perturbations on the input(Yeh et al. 2019). Within these methods are LIME, C-LIME, KernelSHAP, Occlusion, Vanilla Gradients, Gradients x Input, SmoothGrad, and Integrated Gradients, which all perform local linear function approximation of a black box model, differing in loss function and neighborhood(Han, Srivinas, and Lakkaraju 2022). There is no consensus definition of feature attribution whereas the feature instability measure clearly & intuitively defines explainability. Work by (Bastings et al. 2021) proposes feature permutation importance.
evaluation functions separately for precision and mean rank, however these functions ignore multi-way feature interactions, which are both represented in Eq. 3 of DFEST. Overall, feature instability provides a principled definition of multi-way feature explainability that can be utilized as a foundational objective to create new explainability methods, as an alternative to the concept of location function approximation.

**Feature Space**

Explaining the most important feature interactions responsible for the output of $\hat{y}$ through the discovery of multi-way feature instability is formalized as a local optimization search through n-dimensional feature space surrounding $x_s$. The BBF input ($f_i \in f$), can be conceptualized as a single coordinate in a tensor consisting of every possible input the model can take. In this, each dimension of the tensor corresponds to the range of 3 standard deviations ($\sigma_s(f_i) - \sigma_{-3}(f_i)$) of possible values of an input feature of the model, defined as domain $D$, while the position has a value corresponding to $y'. A high dimensional feature space tensor has complexity $O(n^d)\forall x \in f_i$, where $x$ is the number of unique values each feature $f_i$ could exhibit, and $n$ is the number of distinct features input to the model.

In this paper, space is structured as a graph, where the source node represents $x_s$, while query nodes ($x_q$) are generated as a means to provide the BBF with feature combinations that approach a decision boundary. Each node contains input features corresponding to its position in feature space ($x_q$), and a feature-dependent scaled euclidean distance $\Delta'(x_s, x_q)$ from the given query node to the source node. Solutions are nodes with positions near the source node in feature space which led to a change in the BBF output.

**Method**

DFEST is a novel model agnostic XAI method that leverages local informed search to measure post-hoc multi-way feature instability of local BBF predictions. DFEST introduces (1) The concept of feature stability as a measure of explainability of the output of a model, and (2) an informed stability descent based search algorithm capable of quantifying multi-way feature stability of a given binary prediction. (Fig. 1)

**Algorithm 1: N-Dimensional Uniform Search Algorithm**

1. $D = \sigma_3(f_i) - \sigma_{-3}(f_i) \forall i \in d$ and $\forall f_i$ in training set.
2. $\mu = \frac{d}{nSteps} \forall d \in D$
3. $F = \frac{1}{3} \forall d \in D$
4. featSteps = $(\mu f_0, \ldots, \mu f_d)$
5. $\hat{y} = BBF(x_s)$
6. for $0 \rightarrow k$ do
7. $x_u = \text{Normalize} (\text{Rand}(0,1) \forall i \in n)$
8. for $step \in nSteps$ do
9. $x_q = (x_u \times \text{featSteps[step]}) + x_s$
10. $y' = BBF(x_q)$
11. if $y' \neq y'$ then
12. $x_q,distance = \Delta'(x_s, x_q)$
13. $R_U,\text{insert}(x_q)$
14. end if
15. end for
16. end for
17. return $R_U$

This method is closely related counterfactuals, attribution, & influence functions, as they attempt to measure post-hoc explainability through perturbations in feature space. However, these methods lack a reliable quantified definition such as feature instability, or a method to identify multi-way feature interactions.

**Solution Definition**

DFEST functions to identify the ordered set of the $k$ most unstable feature-interaction clusters, denoted as $\{c_0, \ldots, c_k\} \subset C \forall c_i < k$ s.t. $c_i \supseteq \text{Sign}(x_s - x_q)$, where $\text{Sign}()$ denotes a step function s.t. $\text{Sign}(x > 0) = 1$, and $\text{Sign}(x = 0) = 0$, returning a vector of length $|x_s|$ (MATLAB 2022). The ordering of elements in $C$ is given by the cluster instability $I(c_j) > I(c_{j+1}) \forall j < ||P(x_s)||$, the number of elements in the power set of feature-interactions in $x_s$. Furthermore: (a) Given $x_s$ and $y_s$, $x_s$ is iteratively perturbed by small amounts into $x_q$, until $y_q \neq y_q$, making $x_q$ an instability solution, though likely sub-optimal, and a continuous valued member of an instability cluster $c_i$, where $c_i$ is the hypothetical complete set of all counterfactual points $x_q$ diverging from $x_s$ into cluster $i$. (b) Instability $(c_i | x_s, x_q) = \frac{1}{\Delta'(x_s, x_q)}$, where $(x_s - x_q) = \min(x_s - x_q) \forall x_q \in c_i$; (c) Different features $f_i \in x_s$ exist on different value scales, and thus must be scaled by their expected input range, with a perturbation in $f_i$ is equivalent in terms of feature instability to a similar degree of perturbation of $\hat{f}_j$. $\Delta'(x_s, x_q)$ is defined as the euclidean distance between the features in $x_s$ and $x_q$, s.t. $x_s, -x_q = \frac{1}{D(f_j)} * (f_i \in x_s - f_i \in x_q)$, where $D(f_j) = \sigma_3(f_i) - \sigma_{-3}(f_i) \forall f_i \in x \in X_{train}$. Although $x_{train}$ is noted as a feature in the DFEST workflow, training data is only needed to obtain an approximate measure of how the domain of each individual feature differs, for the sake of normalization. A subject matter expert should be able to accurately predict this value in the absence of training data.

**Feature Stability Descent and Tensor Search (DFEST)**

DFEST is composed of a uniform vector search to approximate an even distribution of solutions from the origin in n-dimensional feature search space, followed by distance informed search. DFEST is performed on a 2D feature space in the illustrations included in this paper for the sake of clarity (Fig. 3), with higher dimensional search results of the synthetic ground truth model illustrated in Fig. 4 & 5. The n-dimensional feature space to be searched for the $k$ feature interactions having the lowest stability with respect to a query recommendation, is defined by the continuous space of interactions between every feature column input to the BBF.

We divide the DFEST approach into two subsections:

1. Uniform Search
2. Heuristic Cluster Search
Uniform Search
A n-dimensional decision boundary consists of the closest positions \( x_q \) surrounding \( x_s \in \mathbb{R}^n \), which give \( y_s \neq y_q \). With the goal of identifying \( x_q \) to reach the decision boundary with Distance \( \Delta' = \min(\Delta'(x_s, x_q)) \), heuristics are essential to avoid the impractical scale of a n-dimensional brute force search over the feature space with complexity \( O(n^k) \).

Evenly Distributed Search Over Surface of \( n \)-Sphere
To identify heuristics to begin an informed search, sparse solution nodes are uncovered with a high degree of coverage over the surface of the n-dimensional decision boundary. This can be abstracted to the simpler task of creating an even distribution of points on the surface of an n-sphere(Marsaglia 1972).

Uniformly Distributed Search of Feature Space
Given a large number of n-dimensional unit vectors, with each dimension representing feature values of all \( f_i \in x_s \), an iterative search along each vector can be performed with steps. Refer to Fig. A.1 A5-A9 to demonstrate how Uniformly Distributed Search precedes Informed Cluster Search. The algorithm to identify \( R_U \) sub-optimal solutions for use in Informed Cluster Search is outlined in Algorithm 1. Additionally, let \( R_H \) denote locally optimal feature stability solutions uncovered by Informed Cluster Search. A feature scaler \( F \) is defined for each feature dimension of the unit vector, to amplify the step size as a function of the domain of each respective feature dimension, s.t. \( F = [D(f_1), \ldots, D(f_n)]^{-1} \). \( F \) ensures a common step multiplier across all features is amplified for feature columns that have a higher range, such that a step in each dimension of \( x_q \) corresponds to a comparable change in conceptual distance from \( x_s \). \( F \) can alternatively be described as the element-wise inverse of the domain of each feature, given by the training data/expected input domain, s.t. \( x \ast F = x_{\text{scaled}} \). Let \( x_u \) denote a randomly generated unit vector of a n-sphere representing feature combination slope, and \( \mu \) represent the step size for each feature to be scaled by.

The uniformly distributed points \( x_q \in R_U \) aren’t strictly required for the informed cluster search, however it provides an even distribution of samples to search for a global optimum solution of feature instability. The inclusion of this component is comparable to stochastic gradient descent with restarts, where each sub-optimal solution is persisted as a SGD restart position even spread from each other(Loshchilov and Hutter 2016).

By leveraging a graph structure, each unique combination of nodes that are traversed to as a solution, or adjacent node, is able to store a measure of distance from the source node \( (x_s) \), to specify the node’s position \( (x_q) \) in the priority queue of Informed Cluster Search. The even distribution of permutations in n-dimensional space provides a low complexity method to identify neighboring locations in feature space that lie on a decision boundary of \( y \).

Informed Cluster Search
The Informed Cluster Search component of DFEST is comparable to a conventional A* search, however it includes (1) a direct descent from \( x_q \) toward \( x_s \) until a minimum distance is reached, and (2) an adjacent cluster generation strictly orthogonal to the source node’s cluster. Refer to Fig. 3 for a demonstration of how the Informed Cluster Search obtains \( R_U \) from the uniformly distributed search & performs a local search similar to stochastic gradient descent (SGD).

We present the Informed Cluster Search in the step-wise representation:

- Perform Uniformly Distributed Search to identify \( R_U \) to be used as initial nodes in priority queue.
- Pop the query node closest to the source node from the queue, & append to solutions list if \( y' \neq y \), which is determined by querying the BBF.
- Calculate the feature-interaction cluster being expressed by the query node, and mark as visited, or discard if already visited.
- Search directly from query node to source node with defined number of steps, similarly to the precursor search method, to approximate the minimum distance expressed within the feature-interaction cluster.
- Calculate the feature-interaction clusters of the query node and adjacent nodes.
- Generate query nodes at the center of adjacent feature-interaction clusters, and push those to the priority queue.

Adjacent Cluster Search
Whenever a valid solution \( x_q \) is discovered in Informed Cluster Search, every 1-way feature change is queried for the solution’s feature-interaction cluster \( c_i \) to identify directly adjacent clusters to search next. Adjacent clusters whose center \( x_{q_i} \) is a valid solution s.t. \( y_{q_i} \neq x_s \) undergo a linear search to the direction of \( x_s \), in order to find the minimum \( x_q \in c_i \). The center \( x_{q_j} \) of the adjacent cluster \( c_j \) derived as follows:

Algorithm 2: Center \( x_{q_j} \) of adjacent cluster
1: \( x_{q_j} = x_s + (c_i \ast F \ast m) \)
2: \( r = \text{Distance}(F \ast m) \)
3: \( r = \sqrt{\sum_i |F|^2 m^2} \)
4: \( r = m \ast \sqrt{\sum_i |F|^2} \)
5: \( m = \frac{\sum_i |F|^2}{\sum_i |F|^2} \)
6: \( x_{q_j} = x_s + \frac{F \ast x_s \ast c_i}{\sum_i |F|^2} \)

Where \( r = \text{Distance}(x_s, x_{q_j}) \), and \( m \) is calculated multiplier.

Ground Truth Evaluation of DFEST
There is no inherent ground truth to the most important multi-way feature-interactions of a BBF trained on real world data, and static high dimensional feature-interactions are difficult to efficiently produce(Barr et al. 2020). We construct a deterministic synthetic model that designates a set of the top \( k \) most unstable (i.e. important) feature-interactions. Demonstrating adequate performance of explainability frameworks on a synthetic ground truth model builds trust in the generalizability of the frameworks to
explanations of real world BBFs. Given feature instability as a measure of feature-interaction importance for a given \( \hat{y} \), a brute force full coverage explainability measure can theoretically guarantee optimal instability discovery. This ground truth method can be conceptualized as Friedman’s H-Statistic, which provides an n-feature partial dependence function. However, these types of methods scale exponentially for n features and their range of values, and are thus impractical as a ground truth for comparison of today’s standard explainability techniques. DFEST can approximate a brute force method using the same objective function, while utilizing an outside-in search strategy with heuristics of distance to map out a decision boundary in feature space closest to the source node. Although this “model” is functionally a single parameter decision tree, the nature of the output is comparable to that of an arbitrary BBF, as it is only the binary output that DFEST and other counterfactual methods take into account when explaining a prediction. The synthetic model dynamically calculates the quantifiable feature instability of any arbitrary feature cluster \( c_i \) in O(n) complexity, as a feature weighted ascent from the predetermined optimal feature instability cluster \( c_{\min} \).

**Feature-Interaction Cluster Distance (Equation 2)**

\[
\Delta(x_s, c_{\text{Query}}) = \Delta(x_s, c_{\text{Min}}) + \left[ \sum_{i} c_{\min_i} - c_{\text{Query}}_i \times m_i \right] \times \text{increment}
\]

The synthetic ground truth model takes a feature-interaction cluster as input \( c_{\text{query}} \), e.g. \((1,1,0,0,...,0)\) denoting an interaction of the first 2 features, along with the relative feature importance of every feature in the cluster \( (m_1, ..., m_n) \), where \( n \) is the importance multiplier of the respective feature. The resulting feature space gives a single minimum feature-interaction cluster, with progressively less significant feature-interaction importances for other clusters \( c_{\text{query}}'s \) further from \( x_s \) than the \( c_{\min} \), and added noise from \( m \). Let \( c_{\text{Query}}_i \) represent the \( i \)th feature’s inclusion where \(+1, -1\) denote directional significance of the feature in \( c_{\text{Query}} \), and \( 0 \) denote disclusion from the feature interaction. Every possible feature-interaction cluster decision boundary distance \( \Delta \) from \( x_s \) to the center of \( c_{\text{Query}} \) in the defined feature space is dynamically generated by calculating the difference of each feature position between \( c_{\min} \) and \( c_{\text{Query}} \). The decision boundary distance for \( c_{\text{Query}} \) is defined for the synthetic ground truth model as the distance from \( x_s \) to the center of \( c_{\text{Query}} \)’s decision boundary (Equation 2).

**Algorithm 3: Feature-Interaction Cluster Determination**

1: Input \( x_q = (f_1, ..., f_n)_q \)
2: \( c_{\text{cluster}} = [0_0, ..., 0_d] \)
3: if \( f_{\text{max}} > 0 \) then
4: \( c_{\text{cluster}}[f_{\text{max}}] = 1 \)
5: else
6: \( c_{\text{cluster}}[f_{\text{max}}] = -1 \)
7: end if
8: for \( f_i \in f \) do
9: if \( \left| \frac{f_{\text{max}}}{f_i} \right| \leq 2 \) then
10: if \( f_i > 0 \) then
11: \( c_{\text{cluster}}[f_i] = 1 \)
12: else
13: \( c_{\text{cluster}}[f_i] = -1 \)
14: end if
15: end if
16: end for
17: return \( c_{\text{cluster}} \) as \( c_q \)

With a synthetic ground truth feature model defining feature instability in arbitrary feature clusters outlined in Equation 2, DFEST must leverage Algorithm 3 to identify the feature cluster that \( x_q \) is a member of w.r.t. \( x_s \), in order to search adjacent clusters in the Informed Cluster Search. Given feature parameters in n-dimensional feature space: \((f_1, ..., f_n)\), the corresponding feature-interaction cluster is computed w.r.t. the distance of \( f_{\text{max}} \), the feature with the largest magnitude distance from the same scaled feature in the source node, and that of \( f_i \) is orthogonal to \( f_{\text{max}} \). Fig. 2 illustrates the search of a feature space, demonstrating the cluster calculation described in Algorithm 2. Algorithm 3 is based on pairwise ratios between the feature with the largest relative change in magnitude from the source to query nodes w.r.t. every other feature in the query node. This feature importance clustering process strictly defines a static clustering threshold calculation s.t. any node can be clustered in O(n) complexity. Feature importance clustering has parallels to integrated Hessians(Janizek, Sturmfels, and Lee 2020) and agglomerative contextual decomposition (ACD), which iteratively forms hierarchical clusters based on interaction scores(Singh, Murdoch, and Yu 2018). Similarly, contextual decomposition captures combinations of token inputs to LSTM models leading to differential outputs(Murdoch, Liu, and Yu 2018).

**Synthetic Ground Truth Discussion** Ground truth explainability can help overcome the disagreement problem (Krishna et al. 2022) between different explainability methods, when evaluated on a well defined explainability interpretation such as feature instability. Work by (Bastings et al. 2021) & (Zhou et al. 2021) construct single-way ground truth feature permutation importances using partially synthetic data grounded by a model’s tendency to assign high feature importance to “shortcuts” and artifacts. These ground truth models were >100% accurate, meaning they still partially depended on unimportant features, and did not thoroughly represent valid ground truth explainability. Conversely, the fully synthetic ground truth explanations proposed by DFEST provides inherent multi-feature importances that are not subject to error. Mechanistic interpretability involves reverse engineering BBFs to provide mechanistic explanations for a model with predefined outputs, which is accomplished in DFEST by creating a completely known and deterministic synthetic ground truth explainability model. Furthermore, recent work by TracrLindner et al. 2023) conceptualizes a ground truth transformer model which can be used to evaluate current and future explainability methods over transformers.
Loss Function for Feature Importance Ranking Evaluation

Rank-aware evaluation methods are typically leveraged to evaluate the top recommendations produced by RecSys models. Leading evaluation methods include Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), and Normalized Discounted Cumulative Gain (NDCG) (Järvelin and Kekäläinen 2002). These metrics are extremely valuable in the practical evaluation of RecSys models (Gupta et al. 2020), as conventional accuracy loss functions and decision support metrics that leverage confusion matrices such as precision, recall, and F1 scores operate at the individual and total dataset level, respectively. Rank-aware evaluation metrics target the top k recommendations, enabling a more real world application performance evaluation.

State-of-the-art explainability frameworks provide a ranked list of feature-interactions with associated magnitude as output, and thus require a rank-aware evaluation when considering the top k feature-interaction importance’s. Eq. 3 describes the rank-aware feature-importance evaluation proposed in our work to evaluate DFEST and LIME against a ground truth measure.

Feature-Interaction Importance Ranking Loss Function (Equation 3)

\[
\text{loss} = \sum_{i=1}^{k} \left[ \min_{j=1}^{k} \left( \left| c_{\text{index}} \right| \right) \right] \\
- g_{i,\text{index}} \left[ \sum_{x} \left( c_{x} - g_{j} + 1 \right) \right] - 1 \times \frac{1}{k}
\]

The loss function evaluates the ranking of explainability frameworks (DFEST, LIME) on (1) the distance in ranking position of a feature-interaction cluster from its ground truth ranking position (He et al. 2017), as explicitly defined by the synthetic ground truth model (Wu 2022), and (2) the difference from the ground truth feature-interaction cluster. The 2nd measure is required to account for LIME’s inability to represent multi-way feature-interactions. In short, loss is defined by the sum of the similarity in ranking of each of the top k feature-interaction clusters generated by an evaluation framework and a synthetic ground truth model. The feature-interaction importance ranking loss function is defined by Equation 3, where k is the number of unique feature-interaction clusters (i.e., 8), \( c_1, \ldots, c_k \) are the top k DFEST feature-interaction clusters, \( g_1, \ldots, g_k \) are the top k ground truth feature-interaction clusters, and \( \text{index} \) is the rank of feature-interaction cluster out of k.

Dynamic Synthetic Ground Truth Function for LIME

LIME’s architecture only accepts well defined models with trained weights, thus the source code for LIME was altered to allow for direct calls to the synthetic ground truth model within the LimeTabularExplainer class (Ribeiro, Singh, and Guestrin 2016b). Specifically, only the function `explain_instance()` leveraged the BBF for which the goal was to explain the output of. Thus, when LIME produced a neighborhood of perturbed data points to train a regression model based on the output of the BBF, it is trivial to redirect calls intended for the BBF model to the well defined synthetic ground truth model.

Evaluation & Experiments

The results of DFEST and LIME were expected to have very similar outcomes given the same input, for one-way feature interactions. As DFEST is capable of searching through multi-dimensional feature space, it is able to reflect the non-linearity responsible for the output of deep models, in the form of multi-way feature-interaction instability. For feature space dimensions up to 64 features, DFEST demonstrated significantly lower loss values than both LIME and a random cluster ranking, as illustrated in Fig. 4. The time complexity is also noted. DFEST has various tunable hyperparameters, k precursor solutions, unit vector step size, k A* solutions, and A* adjacent node learning rate. Evaluation of DFEST with various hyperparameter combinations on a ground truth model indicates the optimal settings to be used for real BBFs with n-dimensional feature space, as illustrated in Fig. 5.

Datasets

In the first experiment, a synthetic model was constructed to function as a ground truth in the evaluation of DFEST & LIME, using the feature-interaction importance loss function. In the second experiment, the Wisconsin Breast Cancer (WBC) dataset (Dua and Graff 2017) was used to train a sklearn logistic regression model (Pedregosa et al. 2011), consisting of 30 continuous numerical features, with binary training labels.

DFEST & LIME Evaluation on Synthetic Model

DFEST performance is preserved as the number of dimensions increases, as long as the feature search space is appro-
R5 demonstrates that a sufficiently high number of pre-defined ground truth minimum node. The curve in Fig. (10,000) passed as input to Informed Cluster Search remained constant through each iteration. The decreasing curve illustrated in Fig. 5 demonstrates the ability of DFEST to achieve a meaningful increase in performance while searching the feature space for optimal multi-way feature instability solutions. However, as \( R_H \) increases, the feature space is more thoroughly searched, and the top k optimal solutions more likely to be found. This is demonstrated at the increase from the 16-32 dimensional feature space. DFEST’s feature interaction cluster rankings typically include the correct top rankings with the correct magnitude for each cluster, however may be in a slightly different order.

The decreasing curve illustrated in Fig. 5 demonstrates the ability of DFEST to achieve a meaningful increase in performance while searching the feature space for optimal multi-way feature instability, as a function of the number of solutions identified by Informed Cluster Search. This is due to additional time allowed for DFEST to perform stability descent along feature clusters that are approaching the pre-defined ground truth minimum node. The curve in Fig. 5 demonstrates that a sufficiently high number of \( R_H \) approaches high performance.

DFEST & LIME Evaluation on the WBC Dataset

DFEST offers an explainability output that has both dimension to explain the contributed instability of each feature in a feature-interaction cluster, and is dimensionless in terms of a ranked order of top k important clusters, which can be compared relatively across frameworks. The feature-interaction cluster matrix presented in Fig. A.2 demonstrates the top k most unstable feature-interaction clusters, descending in importance from left to right. The relative contribution of every feature to its cluster is preserved, and patterns of significant feature stability and instability is observed for features which are either all green or all white for all 50 clusters returned by DFEST. The comparison of DFEST & LIME in Fig. A.3 demonstrates considerable, though significantly different, overlap in reported feature importances, with both representations having similar expressiveness. However, LIME lacks the additional k-way cluster analysis offered by DFEST in Fig. A.2. It is likely that comparison of the methods interpreting BBF output would show greater differences due to LIME’s ignorance of non-linear feature interactions.

Given the strict threshold for feature importance clustering, the top 50 feature interaction clusters of Fig. A.2 clearly demonstrates that the clusters with the highest instability tend to have the same core unstable features, which are the MOST unstable features in each of those clusters.

**Conclusion**

This paper demonstrates a novel conceptualization of a principled feature importance metric for explainability & implementation of a completely deterministic and interpretable ground truth explainability measure, capable of both single feature and n-way feature interaction measures. To address the original objectives of this paper, (1) We implemented a fully synthetic ground truth explainability measure and used it to compare the feature importance accuracy of DFEST against LIME. Future experiments hope to simplify the synthetic ground truth model by defining a local decision boundary as the surface of an off-origin n-sphere, where the origin is the model input to be explained; (2) Supra-additive effects are represented by the synthetic ground truth model & DFEST, however more should be done to visualize & interpret these interactions. (3) Fig. 4 & A.3 compare the relative importance of different features between DFEST and LIME, with DFEST demonstrating higher expressivity in feature interactions.

Feature instability offers a well defined measure of multi-way feature explainability over continuous space for BBF predictions, however is not well defined for textual or image data. Feature instability specifically conceptualizes a logical, quantitative, and deterministic definition of post-hoc explainability over a model’s decision space. DFEST is a proof-of-concept method designed to utilize the newly defined feature instability explainability measure and associated ground truth model, however is restricted to low dimensional DNNs due to scaling limitations as the number of model inputs increases. Future methods can address categorical features by projecting the features to continuous space with learned embedding vectors, however this perpetuates the scaling issue of high dimensionality. DFEST is indeterministic as shown in Fig. 4 & 5. Enabling measurement of multi-feature interaction stability has numerous implications, including identification of opportunities for generating targeted synthetic data for retraining a model, essentially pushing the decision barrier with precision to positions in feature space.
Acknowledgements

This work was partly supported by the ONR Research Expeditionary Cyber N00014-22-1-2507.

References


Inglis, A.; Parnell, A.; and Hurley, C. 2021. Visualizing variable importance and variable interaction effects in machine learning models.


Polak, P.; and Krzanowski, R. 2021. Towards Trusted AI (TAI) FIN. *Zaufanie do systemów sztucznej inteligencji*.


Tsai, C.-P.; Yeh, C.-K.; and Ravikumar, P. 2022. Faith-Shap: The Faithful Shapley Interaction Index.


Figure A.1: Workflow of Feature Stability Descent and Tensor Search Explainability (DFEST). At a high level, DFEST functions to explain the output of a model “BBF”: $y_s$ given input $x_s$ through a heuristic guided iterative counterfactual search over the feature space of the model. A) Evenly distributed gradient descent restart points in representation space are identified by the Uniform Distributed Search, B) Local minimums are found for these restart points using Informed Cluster Search & Adjacent Cluster Search; C) The top k informed solutions are evaluated by the rank-aware evaluation, comparing DFEST to the ground truth top k feature interactions.
Figure A.2: Top 50 feature interaction clusters for a given prediction from a model trained on the Wisconsin Breast Cancer Dataset. Y axis labels display feature labels, with cluster feature instability decreasing from left to right, i.e. leftmost clusters denote higher importance for explainability. Green boxes denote an increase if the feature is unstable (approaches decision boundary) when in combination with the other features in its column, while red boxes denote the reverse. White boxes denote features whose significant perturbation did not result in $\hat{y}_q \neq \hat{y}_s$, i.e. are stable.

Figure A.3: Feature importance comparison between DFEST & LIME for a given model prediction trained on the Wisconsin Breast Cancer Dataset. (A) DFEST: The Feature Importance of DFEST was averaged from the top 50 clusters, and represents which clusters are stable (0) and unstable (+/-). (B) LIME similarly displays features that contribute to and threaten the stability of the prediction.