Enhancements to UnRAvEL Algorithm for Improved Stability, Efficiency, and Local Fidelity in Explanations

Aditi Roy
MT23010

1. Introduction

In machine learning, black-box models often provide excellent predictions but are hard to interpret. The Uncertainty-driven Robust Active Learning with Explanation Locality (UnRAvEL) algorithm was created to offer more understandable explanations for these models by selecting important samples and training a simpler model close to a target instance.

Despite its good performance, there are areas where UnRAvEL can be improved for better stability, efficiency, and interpretability. This project suggests an improved version of UnRAvEL, with features like adaptive exploration, better acquisition functions, and more consistent explanations across multiple levels.

2. Background and Motivation

The original UnRAvEL algorithm focuses on creating explanations by actively sampling data points with high uncertainty and training a surrogate model to approximate the behavior of a black-box model locally. While effective, UnRAvEL has notable limitations:

- **Stability:** The explanations generated across different runs can vary due to differences in sampled points, leading to inconsistent interpretations. This variability undermines trust in the explanations, especially when decisions require high reliability.
- **Efficiency:** The iterative process of selecting uncertain data points, retraining the surrogate model, and generating explanations can be computationally expensive. This limits scalability, especially when dealing with large datasets or complex models
- **Fidelity:** In complex or highly non-linear regions of the feature space, the surrogate model may struggle to accurately approximate the black-box model. This misalignment reduces the explanatory power of the approach.
- **Interpretability:** Relying on a single explanation method can result in incomplete insights. Complementary explanation methods could provide a broader and more informative perspective on model behavior.

3. Objectives

The primary objectives of this Enhanced version of Unravel project are as follows:

- **Improve Stability:** Enhance the consistency of generated explanations across multiple iterations, minimizing variability between runs.
- **Enhance Computational Efficiency:** Reduce training time and computational overhead by optimizing the sampling process and resource utilization.
- **Increase Fidelity:** Improve the surrogate model's approximation of the black-box model in the local neighborhood by using ensemble models and advanced feature selection techniques.
- **Improve Interpretability:** Provide complementary and consistent explanations by integrating multiple explanation methods, ensuring a clearer understanding for users.

4. Proposed Enhancements:

The following enhancements are proposed for the UnRAvEL algorithm:

4.1. Adaptive Sampling Based on Uncertainty and Coverage

Instead of a fixed exploration domain around x_0 with a constant width defined by σ_D introduce an adaptive exploration domain that dynamically adjusts based on the density of already sampled points.
The main objective of the adaptive sampling is use this dynamic exploration exploration to select new points for training the surrogate model such that we explore regions with high uncertainity in the model's prediction and we exploit regions that are already well represented in the training data , avoiding redundant sampling highlighting the concept of coverage. This coverage refers to the density of samples that are already in the region. If it is sparsely sampled, we explore it further. The steps involved in adaptive sampling process are:

1) **Calculation of uncertainties (exploration) and entropy (exploration):** Uncertainity of the model

represents the model's lack of confidence in its prediction. It is quantified using standard deviation(variance) of the model's prediction. Given a point x, the prediction of the Gaussian Process model $\hat{f}(x)$ at that point has a mean $\mu(x)$ and a standard deviation $\sigma(x)$, i.e.,

$$\hat{f}(x) = \mu(x) \pm \sigma(x)$$

Where:$\mu(x)$ is the predicted mean of the function at x. $\sigma(x)$ is the uncertainty (or standard deviation) of the prediction.The uncertainty at a point is defined as:

$$\text{Uncertainty}(x) = \sigma(x)$$

This represents how confident the model is about its prediction. Points with high uncertainty $\sigma(x)$ are candidates for exploration because the model has low confidence in those regions.
After this, we did the Entropy sampling which measures the amount of uncertainity or randomness in a distribution. Higher Entropy means more uncertainity. Here, in this enhancement project, we do the entropy based exploration by generating random values for each point in the pool.

2) **Defining dynamic exploration domain width based on density of samples:** This step involves the calculation of density of points in the sample pool which helps to decide whether to explore(samples in the sparse region) or exploit(samples near existing points). For the calculation of density we first compute the distance between a candidate point x and the set of previously sampled points D as the Euclidean distance.The distance between x and D is given by:

$$\text{distance}(x, D) = \|x - D\|$$

where $\|x - D\|$ is the Euclidean distance between x and each point in D. Next, we calculate a density score for x. This can be interpreted as how densely packed the points are around x. A higher density means the region is already well-sampled, and we should avoid exploring it further.The density score is inversely proportional to the distance, i.e.,

$$\text{Density}(x) = \frac{1}{1 + \|x - D\|^2}$$

where $\|x - D\|^2$ is the squared Euclidean distance. This formula assigns a higher score to regions with dense samples (lower distances) and a lower score to regions with sparse samples (higher distances).

3) **Dynamic exploration domain:** Now, based on the density of sampled points, we dynamically adjust the exploration width. This adjustment allows us to balance exploration and exploitation.We calculate the dynamic exploration width $w(x)$ based on the density of samples in the neighborhood of x. This can be defined as:

$$w(x) = \exp\left(-\text{Density}(x)\right)$$

Where: $\text{Density}(x)$ is the density score for point x, and $\max(\text{Density}(D))$ is the maximum density across all existing points D. This function $w(x)$ ensures that:

- In regions of high density, $w(x)$ becomes small, focusing the search on areas that are already well-sampled (exploitation).
- In regions of low density, $w(x)$ becomes large, encouraging the model to explore these areas (exploration).

4) **Final sampling score:** Finally, we combine the uncertainty (exploration) and entropy (exploration) with the dynamic exploration width and density to calculate a final sampling score for each candidate point x.The final sampling score is a weighted sum of the uncertainty term and the entropy term, adjusted by the dynamic exploration width:

$$\begin{aligned}\text{Sampling Score}(x) = \text{Uncertainty}(x) \\ \times (1 - \text{Acquisition Weight})\end{aligned} \tag{1}$$

And then, we apply the dynamic exploration domain $w(x)$:

$$\text{Final Score}(x) = \text{Sampling Score}(x) \times w(x) \tag{2}$$

4.2. Enhanced acquisition function with hybrid approach

The Hybrid Acquisition Function combines several components to select the next most informative sample for active learning. The goal of the hybrid function is to balance exploration (searching for uncertain regions of the input space) with exploitation (sampling in areas where the model's uncertainty is high, or it believes the model is likely to provide valuable information). The function also incorporates diversity penalization to avoid selecting samples that are too similar to the already sampled points. The hybrid acquisition function combines the following:

- **Faithful Uncertainty Reduction (FUR)** FUR score is calculated for a point x using the equation: where

$$\mathbf{x}_n = \arg\max_{\mathbf{x}} \underbrace{-\left\| \left(\mathbf{x} - \mathbf{x}_0 - \frac{\overline{\sigma}\epsilon}{\log(n)} \right) \right\|_2}_{\text{T1}} + \underbrace{\sigma_n(\mathbf{x})}_{\text{T2}},$$

σ is the empirical mean of the standard deviation of individual features in the training data, $\epsilon \sim \mathcal{N}(0, 1)$, x_0 is the index sample, and $\sigma_n(x_n)$ is the standard deviation of f_e obtained until the nth sample x_n.

- **Diversity Penalization and the dynamic exploration width from adaptive sampling** The diversity penalty ensures samples are not too similar to already sampled points, which is mainly distance between x and D is given by:

$$\text{distance}(x, D) = \|x - D\|$$

where $\|x - D\|$ is the Euclidean distance between x and each point in D.
The dynamic width encourages exploration in sparse areas which is given as :

$$w(x) = \exp\left(-\text{Density}(x)\right)$$

Where: Density(x) is the density score for point x

- **Acquisition Function Score** The final acquisition score combines FUR, Diversity Penalization, and Dynamic Exploration Width:

$$\alpha(x) = (\text{FUR}(x) + \lambda \cdot \text{distance}(x, D))$$
$$\cdot \text{w}(x)$$

Where λ is Diversity weighting factor and $\alpha(x)$ is the final acquisition score for sample x

4.3. Surrogate Model Refinement

Iterative Model Fine-tuning with Weighted Points:
Use a weighting scheme where samples closer to x_0 have higher importance when training the Gaussian Process Regressor (GPR), leading to improved local fidelity. This can be achieved by assigning a weight

$$w(x) = \exp\left(-\frac{\|x - x_0\|^2}{2\sigma^2}\right)$$

to each sample x, emphasizing samples in the core neighborhood of x_0.

Neighborhood Clustering and Local Models:
Instead of a single GPR model, apply clustering in the feature space to divide the neighborhood into sub-regions and train separate GPR models for each sub-region. This multi-model approach could better approximate complex, nonlinear decision boundaries within the local neighborhood of x_0.

4.4. Enhanced Stability metric

The function evaluates the consistency of feature importance across different methods by computing two metrics: Jaccard distance and Spearman correlation. For each randomly selected sample, the top k features are chosen using two different methods (simulated by random selection here).

The Jaccard distance measures the similarity between the top k feature sets from both methods, where a lower value indicates higher similarity. The Jaccard distance is defined as:

$$\text{Jaccard Distance} = \frac{|A \cap B|}{|A \cup B|}$$

where A and B are the sets of top k features selected by the two methods.

The Spearman correlation evaluates the rank-order consistency of feature importance between the two methods. The Spearman correlation coefficient ρ is defined as:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

where d_i is the difference between the ranks of corresponding values, and n is the number of features.

The function then computes the mean Jaccard distance and the spearman's coefficient across all selected samples as a stability metric. This quantifies how consistently the feature rankings align across different explanations or models.

5. Dataset

Five datasets from the UCI Machine Learning repository were chosen for the tabular data-based experiments, based on their relevance in the literature, feature novelty, and prediction tasks. The datasets were pre-processed using the same pipeline, including removal of missing values, frequency encoding of categorical features, and feature standardization. The datasets are:

- **Parkinson's:** A classification task with 195 biomedical recordings and 22 features, aiming to predict whether a patient has Parkinson's disease.
- **Cancer:** A classification task with 569 entries and 30 features, predicting whether breast cancer is malignant or not.
- **Adult:** A classification task with 14 features, predicting whether an individual's income exceeds $50,000.
- **Boston:** A regression task with 506 entries and 13 features, predicting house prices based on neighborhood characteristics.
- **Bodyfat:** A regression task with 14 features, predicting body fat percentage based on physical properties.

6. Data Preprocessing

The processing part of the project prepares the input data for machine learning tasks by performing several key steps. First, it removes rows with missing values in the target column to ensure clean data. Then, for categorical features, it applies frequency encoding using `LabelEncoder` to convert categorical values into numerical representations. The dataset is then split into features (X) and the target variable (y). Next, the function performs a train-test split, either using a specified training size (`n_train`) or a default of 80% for training and 20% for testing. To standardize the features, it uses `StandardScaler` to normalize the feature values, ensuring they have a mean of 0 and a standard deviation

of 1. Finally, it returns the preprocessed training and testing sets, along with the total number of samples, the number of features, and the list of feature names.

7. Proposed Algorithm

The **Enhanced UnRAvEL** algorithm is an active learning method designed to iteratively improve the training of a surrogate model while balancing exploration and exploitation. Below is a simplified breakdown of its key steps:

7.1. Initialization

The process starts with an initial sample, x_0, and its prediction from the black-box model. A Gaussian Process Regressor (GPR) model is set up with an Automatic Relevance Determination (ARD) kernel. The GPR model is used as a surrogate to approximate the black-box model's predictions.

7.2. Adaptive Sampling

In each iteration, the algorithm selects new data points to sample based on two factors:

- **Uncertainty**: Areas of high uncertainty are prioritized for sampling, meaning the model is unsure about the predictions in those areas.
- **Entropy**: A measure of disorder or randomness, which helps the model explore new, unknown regions.

The exploration domain adjusts dynamically based on the density of existing data points, ensuring the model explores areas with low sample density (more unexplored regions) and exploits areas where more data is available.

7.3. Acquisition Function

The selection of the next sample is determined by an acquisition function that combines:

- **Faithful Uncertainty Reduction (FUR)**: Focuses on reducing uncertainty in the predictions.
- **Diversity Penalization**: Prevents choosing data points too similar to already selected samples, ensuring diversity in the sampled points.

The acquisition function balances these factors and adjusts the exploration strategy by considering both uncertainty and the density of samples.

7.4. Surrogate Model Update

After selecting the next data point, the algorithm adds it to the training set and retrains the surrogate model (GPR). This allows the model to improve its predictions over time.

7.5. Stability and Fidelity Evaluation

After each iteration, the algorithm evaluates the stability of feature importance across different methods (like ARD and LIME) using metrics like Jaccard distance and Spearman correlation. Additionally, it assesses the fidelity of the surrogate model by comparing its predictions to the true values using metrics like Mean Absolute Error (MAE) and R^2.

7.6. Repeat

These steps are repeated for a predefined number of iterations, progressively improving the surrogate model and refining the explanations provided by the model.

In essence, **Enhanced UnRAvEL** is a strategy that iteratively refines a surrogate model, explores the feature space adaptively, and ensures that the model becomes more accurate while maintaining the diversity and relevance of the sampled data. It also checks if the explanations remain stable and consistent across different methods.

8. Results

Based on the stability metric results across the different models and datasets, it is observed that a lower stability metric indicates higher consistency and similarity in the feature rankings. For the Parkinson's dataset, the UNRAVEL (0.60) and ENHANCED ENRAVEL (0.56) models have the lowest stability scores, signifying that these models exhibit the most consistent feature rankings. In contrast, LIME (0.85) and BAYLIME (0.82) have higher stability scores, which suggests that their feature rankings are less consistent, showing more variability across different samples. This pattern holds for other datasets as well. For example, in the Cancer dataset, BAYLIME (0.86) and LIME (0.87) demonstrate relatively higher stability, indicating less consistency in feature rankings compared to UNRAVEL (0.70) and ENHANCED ENRAVEL (0.51), which show lower stability scores and, therefore, higher consistency. Similarly, in the Adult dataset, BAYLIME (0.56) and LIME (0.46) have higher stability scores, while ENHANCED ENRAVEL (0.38) and UNRAVEL (0.50) have lower stability, reflecting more consistent feature rankings. The results are consistent across all datasets, where models with lower stability metrics, such as UNRAVEL and ENHANCED ENRAVEL, generally exhibit better performance in terms of feature ranking consistency. Therefore, for all datasets considered, UNRAVEL and ENHANCED ENRAVEL are more stable and reliable in selecting features consistently, while LIME and BAYLIME show less consistency in their rankings, as indicated by their higher stability metrics

9. Conclusion

The results strongly indicate that Enhanced UnRAvEL offers a substantial improvement over the other models,

Model's performance with Datasets	LIME	BAYLIME	UNRAVEL	ENHANCED ENRAVEL
Parkinsons	0.85	0.82	0.60	0.56
Cancer	0.87	0.86	0.70	0.51
Adult	0.46	0.56	0.50	0.38
Boston	0.79	0.82	0.85	0.45
Bodyfat	0.90	0.90	0.48	0.39

particularly in terms of both accuracy and stability. This suggests that the enhancements made to the original UnRAvEL algorithm—such as the hybrid acquisition function, diversity penalization, and adaptive sampling—are key factors in improving its performance across a variety of datasets. Enhanced UnRAvEL is highly suitable for applications where model stability and accurate, consistent feature importance are critical, making it a more reliable choice for complex, real-world problems.

References

[1] Aditya Saini and Ranjitha Prasad, *Select Wisely and Explain: Active Learning and Probabilistic Local Post-hoc Explainability*, Available: https://dl.acm.org/doi/abs/10.1145/3514094.3534191

[2] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, *"Why Should I Trust You?" Explaining the Predictions of Any Classifier*, Available: https://doi.org/10.48550/arXiv.1602.04938

[3] Xuechen Zhang, Samet Oymak, and Jiasi Chen, *Post-hoc Models for Performance Estimation of Machine Learning Inference*, Available: https://doi.org/10.48550/arXiv.2110.02459