# Fast and accurate regional effect plots for automated tabular data analysis.

Vasilis Gkolemis[1,2], Christos Diou[1], Eirini Ntoutsi[3] and Theodore Dalamagas[2]

[1]Harokopio University of Athens
[2]ATHENA Research Center
[3]University of the Bundeswehr Munich

**Abstract**

The regional effect is a novel explainability method that can be used for automated tabular data understanding through a three-step procedure; a black-box machine learning model is trained on a tabular dataset, a regional effect method explains the ML model and the explanations are used to understand the data and and support decision making. Regional effect methods explain the effect of each feature of the dataset on the output within different subgroups, for example, how the age (feature) affects the annual income (output) for men and women separately (subgroups). Identifying meaningful subgroups is computationally intensive, and current regional effect methods face efficiency challenges. In this paper, we present regional RHALE (r-RHALE), a novel regional effect method designed for enhanced efficiency, making it particularly suitable for decision-making scenarios involving large datasets, i.e., with numerous instances or high dimensionality, and complex models such as deep neural networks. Beyond its efficiency, r-RHALE handles accurately tabular datasets with highly correlated features. We showcase the benefits of r-RHALE through a series of synthetic examples, benchmarking it against other regional effect methods. The accompanying code for the paper is publicly available.

**Keywords**

Explainability, Interpretability, Regional Effect, Decision Making, Tabular Data Understanding

## 1. Introduction

Latest advancements in Machine Learning (ML) for tabular data have provided models that can accurately learn complex data patterns. At the same time, eXplainable AI (XAI) [1, 2] has emerged to explain how these models operate. Combining ML with XAI is a promising strategy for data analysis. As shown in Figure 1, we can analyze a tabular dataset by explaining a black-box model that is trained on it.

Consider the task of deciding a promotional offer for bike rentals using a relevant dataset [3] with historical data. A detailed description of this task is presented in Section 5. The dataset includes features such as temperature, humidity, hour, working vs. non-working day, etc.. The target variable is the number of bikes rented per hour. We focus on the `hour` feature, but the methodology is applicable to any other feature.

Standard data analysis methods, such as aggregation-based queries, pairwise plots (Figure 2a) or global effect plots (Figure 2b), indicate that bike rentals peak around 8:30 AM and 5:00 PM, due to people moving from and to work.

We propose an improved pipeline (Figure 1) which can provide more detailed insights. We, first, fit a ML model, like a neural network, to the dataset and then use a regional effect XAI method [4, 5, 6, 7, 8]. The pipeline identifies two distinct patterns: on weekdays (Figure 2c), rentals peak at 8:30 AM and 5:00 PM, but on weekends (Figure 2d), rentals rise from 9:00 AM, peak at 12:00 PM, and decline by 4:00 PM, indicating recreational use.

According to that we should opt for a different promotional offer on working and non-working days. The key advantage of our approach is the automatic extraction of these patterns from the data, without any domain expertise, which would be challenging with traditional aggregation-based methods.

The automated extraction of significant subregions, like "working" vs. "non-working" days is computationally intensive. Current regional effect methods, such as r-PDP, r-ALE, and r-SHAPDP[1] face computational limitations when the dataset is large or the black-box model is expensive to evaluate. Additionally, r-PDP struggles with tabular datasets with correlated features.

To address these challenges, we introduce r-RHALE, a regional effect method built on RHALE [9, 10], which:

- is efficient, making it suitable for datasets with numerous instances and expensive black-box models, such as deep neural networks
- handles appropriately tabular datasets with correlated features

We demonstrate these advantages with two synthetic examples. For the experiments, we use the python package Effector [11].

*Workshop on Tabular Data Analysis (TaDA) @ VLDB 2024*


[1]The prefix *r-*<name> is a shortcut for *regional-*<name>

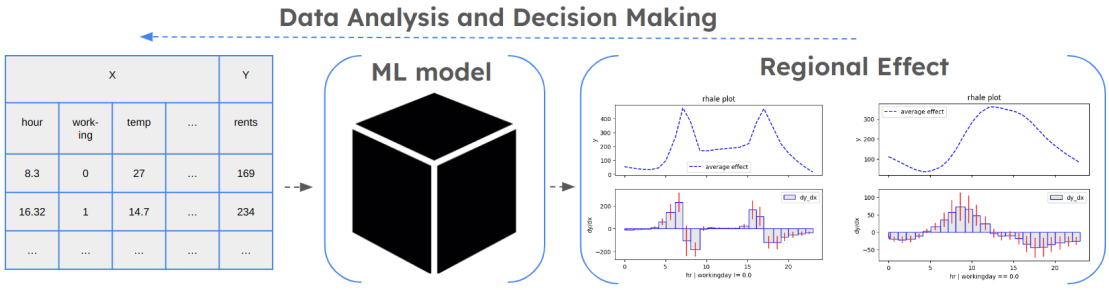



**Figure 1:** Data analysis and decision making pipeline: Utilizing regional effect plots to extract insights from tabular data.

## 2. Regional RHALE

r-RHALE builds on two papers. Gkolemis et al. (2023) [9] introduced RHALE, a global effect method for differentiable black-box models that improves on ALE by being faster and computing heterogeneity. As we will show below, the heterogeneity is crucial quantity for subregion detection. Herbinger et al. (2023) [4] proposed a generic framework for transforming global effect methods to regional, and applied it to PDP[12], ALE[13], and SHAP-DP[14]. This paper integrates these approaches.

**Notation.** Let $\mathcal{X} \in \mathbb{R}^d$ be the $d$-dimensional feature space, $\mathcal{Y}$ the target space and $f(\cdot) : \mathcal{X} \to \mathcal{Y}$ the black-box function. We use index $\mathtt{s} \in \{1, \dots, d\}$ for the feature of interest and $\mathtt{C} = \{1, \dots, d\} - s$ for the indices of all the other features. For convenience, we use $(x_s, \mathbf{x_c})$ to denote the input vector $(x_1, \cdots, x_s, \cdots, x_D)$, $(X_s, \mathbf{X_c})$ instead of $(X_1, \cdots, X_s, \cdots, X_D)$ for random variables and $\mathcal{X}_s, \mathcal{X}_c$ for the feature space and its complement, respectively. The training set $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N$ is sampled i.i.d. from the distribution $\mathbb{P}_{X,Y}$.

**globalRHALE.** RHALE estimates the effect of feature $x_s$ on the output $y$ (Figure 2b), as:

$$f(x_s) = \underbrace{\underbrace{\sum_{k=1}^{k_{x_s}} \frac{z_k - z_{k-1}}{|\mathcal{S}_k|} \sum_{i:\mathbf{x}^{(i)} \in \mathcal{S}_k} \overbrace{\frac{\partial f}{\partial x_s}(x_s^{(i)}, \mathbf{x}_c^{(i)})}^{\text{instance effect}}}_{\mu_k \text{ (interval effect)}}}_{\text{global effect}} \quad (1)$$

The feature axis $x_s$ is divided into $K_s$ variable-size intervals $\{\mathcal{Z}_k\}_{k=1}^{K_s}$, where each interval spans $[z_{k-1}, z_k)$. Let $\mathcal{S}_k$ be the set of instances with the $s$-th feature in the $k$-th interval, i.e., $\mathcal{S}_k = \{x^{(i)} : z_{k-1} \leq x_s^{(i)} < z_k\}$. The interval boundaries are determined by solving an optimization problem as described in Gkolemis et al. (2023).

To understand Eq. (1), we proceed step by step. The instance effect, $\frac{\partial f}{\partial x_s}(x_s^{(i)}, \mathbf{x}_c^{(i)})$, measures the change in the output when the $s$-th feature changes slightly from $(x_s^{(i)}, \mathbf{x}_c^{(i)})$ to $(x_s^{(i)} + \delta, \mathbf{x}_c^{(i)})$. We then average the instance effects for all instances in the $k$-th bin to obtain the bin effect, $\mu_k$. The global effect is the sum of the bin effects.

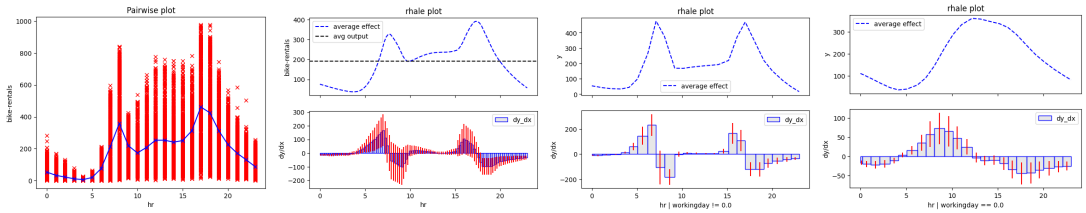
**Heterogeneity.** Heterogeneity measures the deviation of instance effects from the bin effect:

$$H_s = \sum_{k=1}^{K_s} \frac{z_k - z_{k-1}}{|\mathcal{S}_k|} \sum_{i:\mathbf{x}^{(i)} \in \mathcal{S}_k} \left[ \frac{\partial f}{\partial x_s}(x_s^{(i)}, \mathbf{x}_c^{(i)}) - \mu_k \right]^2 \quad (2)$$

Zero heterogeneity indicates that the effect of $x_s$ on the output is independent of other features, i.e., $f(\mathbf{x}) = f_s(x_s) + f_c(\mathbf{x_c})$. In this ideal case, the feature effect explanation is reliable for all instances. As heterogeneity increases, the feature effect explanation becomes less accurate for individual instances, reflecting a stronger dependence on other features $\mathbf{x_c}$.

For example, the effect of hour on bike rentals significantly depends on the day type, resulting in high heterogeneity and inaccurate average explanations for non-working days. By splitting data into working and non-working days, regional effects reduce heterogeneity, providing reliable explanations within each subregion.

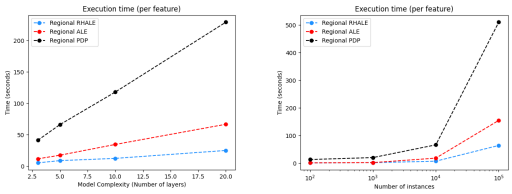
**r-RHALE.** Regional effects aim to identify subregions with reduced heterogeneity by conditioning on one ore more of the features in $\mathtt{C}$. For continuous features, this condition is based on whether the feature value is above or below a threshold $\tau$, and for categorical features, whether it equals or differs from $\tau$. A CART-based algorithm iterates over all features in $\mathbf{x}_c$ and tests various thresholds $\tau$ to find the one that maximally reduces heterogeneity. r-RHALE combines the heterogeneity measure from Eq. (2) with this CART-based algorithm, as detailed in [4, 11].




(a) Pairwise plot (b) Global effect (c) Effect on working days (d) Effect on non-working days

**Figure 2:** r-RHALE applied to the bike-sharing dataset; (a) global effect of feature "hour" on the bike-rentals (b) regional effect on feature "working days" (c) regional effect on feature "non-working days".

**Computational Advantage.** r-RHALE offers a computational advantage over other methods due to its approach to computing heterogeneity. According to Eq. (2), the term $\frac{\partial f}{\partial x_s}(x_s^{(i)}, \mathbf{x}_c^{(i)})$ needs to be computed only once for all instances. When executed in a batched manner with support for automatic differentiation, the computational time is comparable to a single evaluation of $f$. In contrast, other methods require multiple evaluations of $f$ to compute regional effects, resulting in slower execution times, especially for complex and computationally intensive functions $f$.




(a) (b)

**Figure 3:** Execution time applied (a) neural networks of varying number of layers, i.e., varying inference times and (b) on datasets with a varying number of instances (log scale).

## 3. Execution time comparison

In this example, we show that r-RHALE executes significantly faster than r-PDP and r-ALE. We do not include r-SHAPDP in the comparison, because its execution time is prohibitive high, e.g., more than 30 minutes, even for relatively light models and datasets. In the example, we observe that r-RHALE executes fast even under a (a) slow-inference black-box model and (b) a large tabular dataset.

**Slow-inference black-box model:** We generate a dataset with $N = 10^4$ instances, $D = 10$ features and train deep neural networks (DNN) with layers ranging from $L = 3$ to $L = 20$. More layers means bigger inference time, so our findings generalize to any slow-inference black-box model.

In Figure 3a, we observe that r-RHALE's execution time increases at a slower rate compared to r-ALE and r-PDP. Even for complex models like DNNs with 20 layers, r-RHALE requires less than 15 seconds to generate regional effect plots for a single feature. This translates to approximately 4-5 minutes for a typical tabular dataset with 20 features. In contrast, r-PDP requires about 4 minutes per feature, totaling roughly an hour for all features, while r-ALE needs about 1 minute per feature, resulting in approximately 20 minutes for all features.

**Large tabular dataset:** We define a deep neural network (DNN) with $L = 5$ layers and a synthetic dataset with $D = 20$ features and a varying number of instances $N \in \{10^3, 10^4, 10^5\}$ (log scale).

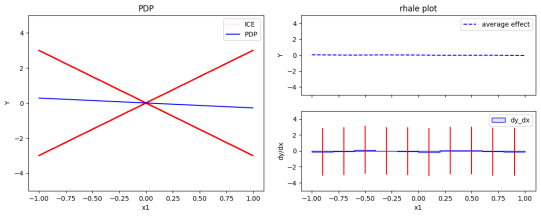
In Figure 3b, we observe that r-RHALE's execution time increases at a slower rate compared to r-ALE and r-PDP. r-RHALE is more than twice as fast as r-ALE and ten times faster than r-PDP. For large datasets, this means that r-RHALE executes 20 minutes and 2 hours faster compared to r-ALE and r-PDP. The efficiency gain would be even more pronounced with a heavier black-box model, as demonstrated in the previous example.

## 4. Correlated Features

In this example, we demonstrate that, unlike r-PDP, r-RHALE handles well tabular datasets with correlated features.

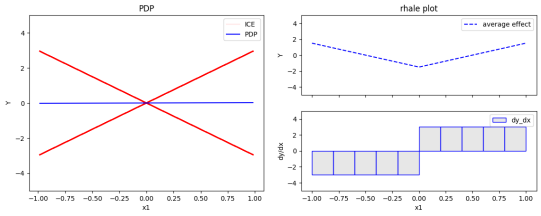
We use the model $y = 3x_1 I_{x_3>0} - 3x_1 I_{x_3\leq 0} + x_3$ with two different data-generating distributions. In the non-correlated setting, all variables are uniformly distributed, $x_i \sim \mathcal{U}(-1, 1)$. In the correlated setting, $x_1$ and $x_2$ maintain the same distributions, but $x_3 = x_1$.

These two versions illustrate that r-PDP produces the same regional effect regardless of correlations, while r-RHALE accurately distinguishes between the two cases. We focus on the effect of $x_1$ on $y$.




(a) Global PDP ($x_1$) (b) Global RHALE ($x_1$)

**Figure 4:** Global plots for the non-correlated setting.




(a) Global PDP ($x_1$) (b) Global RHALE ($x_2$)

**Figure 5:** Global plots for the correlated setting.

**Non-correlated setting.** The effect of $x_1$ arises from the interaction terms $3x_1 I_{x_3>0}$ and $3x_1 I_{x_3\leq 0}$. The global effect will be $3x_1$ when $x_3 > 0$ (half the time, given $x_3 \sim \mathcal{U}(-1, 1)$) and $-3x_1$ when $x_3 \leq 0$ (the other half). This results in an overall zero global effect with high heterogeneity. By splitting into two subregions, $x_3 > 0$ and $x_3 \leq 0$, we obtain two regional effects, $3x_1$ and $-3x_1$, each with zero heterogeneity.

In Figure 4, both r-PDP and r-RHALE correctly identify the global effect. The global effect is zero but with high heterogeneity, indicated by the two red lines in the r-PDP plot (Figure 4a) and the red bars in the r-RHALE plot (Figure 4b). Due to space limitations, we do not illustrate the regional effects, which, in both cases, match the ground truth.

**Correlated setting.** In the correlated case, with $x_3 = x_1$, the effect becomes $y = 3x_1 I_{x_1>0} - 3x_1 I_{x_1\leq 0}$. This is because the interaction terms simplify to $3x_1 I_{x_1>0}$ and $-3x_1 I_{x_1\leq 0}$. When $x_1 > 0$, $x_3 > 0$, so only the $3x_1$ term is active. Similarly, when $x_1 \leq 0$, $x_3 \leq 0$, making only the $-3x_1$ term active.

In Figure 5, we observe that only r-RHALE correctly estimates the global and regional effects. r-RHALE (Figure 5b) accurately computes the effect as $3x_1 I_{x_1>0} - 3x_1 I_{x_1\leq 0}$ with no heterogeneity and does not identify subregions. In contrast, r-PDP (Figure 5a) treats the features as independent, resulting in the same global effect as in the uncorrelated case and incorrectly identifying subregions for $x_3 > 0$ and $x_3 \leq 0$.

## 5. Demonstration

In this section, we delve deeper into the example introduced in Section 1. The bike-sharing dataset [3] encompasses historical data on hourly bike rentals from 2011 to 2012 within the Capital bike share system, alongside relevant weather and seasonal information. The input features include year, month, day, hour, workday status, temperature, humidity, wind speed, and more, with the target variable being the number of bikes rented each hour.

Our aim is to analyze this data to propose an optimal time for a promotional offer. The focus of our analysis is on the feature "hour" to determine the best time of day for the promotion. This method can be applied to other features as well.

The proposed pipeline, depicted in Figure 1, comprises the following steps: first, we apply a neural network to the dataset. Subsequently, we use a regional effect method [4, 5] to assess the impact of the "hour" feature on bike rentals. The analysis reveals that the influence of "hour" differs between weekdays and weekends. On weekdays (Figure 2c), bike rentals peak around 8:30 AM and 5:00 PM, corresponding to commuting times. On weekends (Figure 2d), rentals increase from 9:00 AM, peak at 12:00 PM, and decrease after 4:00 PM, reflecting recreational use.

This finding aligns with common sense. The strength of the pipeline lies in its ability to automatically uncover such patterns without external input. Unlike traditional data analysis methods that require domain expert intervention to identify such subspaces, our approach leverages the machine learning model to discern complex relationships within the data, and the regional effect method to highlight significant subspaces.

## 6. Conclusion and Future Work

In this paper, we introduce a novel method for extracting insights from tabular data. Our approach involves first fitting a black-box model and then explaining its predictions using a regional effect method. The insights gained from the regional effect can then be applied to support decision-making processes.

To this end, we propose r-RHALE, an innovative regional effect method that builds upon the strengths of the global effect RHALE. r-RHALE offers significantly improved efficiency compared to existing methods and effectively handles datasets with correlated features.

## References


[1] T. Freiesleben, G. König, C. Molnar, A. Tejero-Cantero, Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena, arXiv preprint arXiv:2206.05487 (2022).

[2] M. T. Ribeiro, S. Singh, C. Guestrin, " why should i trust you?" explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[3] H. Fanaee-T, J. Gama, Event labeling combining ensemble detectors and background knowledge, Progress in Artificial Intelligence 2 (2014) 113–127.

[4] J. Herbinger, B. Bischl, G. Casalicchio, Decomposing global feature effects based on feature interactions, arXiv preprint arXiv:2306.00541 (2023).

[5] J. Herbinger, B. Bischl, G. Casalicchio, REPID: Regional Effect Plots with implicit Interaction Detection, 2022. URL: http://arxiv.org/abs/2202.07254. doi:10.48550/arXiv.2202.07254, arXiv:2202.07254 [cs, stat].

[6] M. Britton, Vine: Visualizing statistical interactions in black box models, arXiv preprint arXiv:1904.00561 (2019).

[7] C. A. Scholbeck, G. Casalicchio, C. Molnar, B. Bischl, C. Heumann, Marginal effects for non-linear prediction functions, Data Mining and Knowledge Discovery (2024) 1–46.

[8] L. Hu, J. Chen, V. N. Nair, A. Sudjianto, Surrogate locally-interpretable models with supervised machine learning algorithms, arXiv preprint arXiv:2007.14528 (2020).

[9] V. Gkolemis, T. Dalamagas, E. Ntoutsi, C. Diou, RHALE: Robust and heterogeneity-aware accumulated local effects, in: ECAI 2023, IOS Press, 2023, pp. 859–866.

[10] V. Gkolemis, T. Dalamagas, C. Diou, DALE: Differential accumulated local effects for efficient and accurate global explanations, in: Asian Conference on Machine Learning (ACML), 2022.

[11] V. Gkolemis, C. Diou, E. Ntoutsi, T. Dalamagas, B. Bischl, J. Herbinger, G. Casalicchio, Effector: A python package for regional explanations, 2024. arXiv:2404.02629.

[12] J. H. Friedman, B. E. Popescu, Predictive learning via rule ensembles, The annals of applied statistics (2008) 916–954. Publisher: JSTOR.

[13] D. W. Apley, J. Zhu, Visualizing the effects of predictor variables in black box supervised learning models, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 82 (2020) 1059–1086. Publisher: Wiley Online Library.

[14] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, Advances in neural information processing systems 30 (2017).