SMU Data Science Review

Volume 3 | Number 1 Article 4

2020

QLIME-A Quadratic Local Interpretable Model-Agnostic Explanation Approach

Steven Bramhall
Southern Methodist University, sbramhall@smu.edu

Hayley Horn
hhorn@smu.edu

Michael Tieu
mtieu@smu.edu

Nibhrat Lohia
Southern Methodist University, nlohia@smu.edu

QLIME: A Quadratic Local Interpretable Model-Agnostic Explanation Approach

Steven Bramhall, Hayley Horn, Michael Tieu, and Nibhrat Lohia

Masters of Science in Data Science, Southern Methodist University, Dallas TX 75205, USA {sbramhall, hhorn, mtieu, nlohia}@smu.edu

Abstract. In this paper, we introduce a proof of concept that addresses the assumption and limitation of linear local boundaries by Local Interpretable Model-Agnostic Explanations (LIME), a popular technique used to add interpretability and explainability to black box models. LIME is a versatile explainer capable of handling different types of data and models. At the local level, LIME creates a linear relationship for a given prediction through generated sample points to present feature importance. We redefine the linear relationships presented by LIME as quadratic relationships and expand its flexibility in non-linear cases and improve the accuracy of feature interpretations. We coin this use of quadratic relationships as QLIME and demonstrate its viability by comparing its utility to LIME on a black box model. Using data from a global staffing company, the goal of the model is to predict successful candidate placements. QLIME adds explainability to the model and shows an improvement for both successful and unsuccessful predictions, and our quadratic approach is validated with mean squared error (MSE) improvements of 3.8% and 2.9%.

1 Introduction

Artificial intelligence (AI) powers the logic behind many everyday activities and the growth of AI integration continues. AI is especially prevalent in product and movie recommendations, chatbots, disease diagnoses, or forecast predictions [1]. By 2021, global investment in these fields are expected to reach upwards of $52 billion USD with an estimated revenue of nearly $2.6 trillion USD [2]. However, AI models often face challenges in adoption.

Data science is an umbrella term that encompasses both AI and machine learning (ML). The models used in AI are developed and refined for better performance and a focus on improving metrics such as accuracy at the expense of interpretation [3] [2]. These opaque solutions are known as black box models and can obscure features and include a mechanism of bias and misunderstanding [4].

The lack of interpretability arising from black box models challenges workplace adoption [5]. While these types of algorithms are helpful with analyzing data and making predictions and decisions, their immediate use can bring about some skepticism due to a lack of transparency [6]. According to a 2019 study by

the International Institute for Analytics, only 10% of companies have successfully deployed enterprise AI models [7]. Understanding the rationale behind a model's predictions would help with the acceptance of a model. Especially true of financial [8], medical [9], and legal [10] industries, regulatory bodies have strict requirements on model explainability.

Explainable AI (XAI) is a research field with a goal of bringing transparency to black box models and making them interpretable and understandable to human users. Explainability comes from the result of a number of topics in AI. These topics include 1) transparency which is the right to interpretable explanations regarding predictions, 2) causality being the right to a mechanistic understanding behind a learned model, 3) bias to ensure that learned models have an unbiased view of the world, 4) fairness to verify that decisions from learned models were made fairly, and 5) safety to bring confidence in the reliability of a learned model such as confidence intervals [11].

Attention towards the explainability of black box models has surged recently in both academia and practice, partly due to the widespread interest towards expanded use of black box models [2]. Currently, there are a handful of explainability frameworks for data scientists to apply to their models. Choosing an explainability framework is largely dependent on the data, model application, and requirements.

Our contributions in this paper are motivated by a case from a global staffing company. Employee matching is either overly laborious or leverages a black box system that provides matches with no details on the match criteria. As a result, placement managers individually assign job candidates to jobs in a subjective manner, versus a data driven approach. This results in a higher than desired turnover and uncompleted job assignments.

Prediction models can be a huge benefit for staffing companies but the lack of transparency hinders its adoption [12]. We address interpretability by applying a Local Interpretable Model-Agnostic Explanations (LIME) based technique and propose the use of a Quadratic Local Interpretable Model-Agnostic Explanations (QLIME) technique. QLIME expands on the linear limitations of LIME by redefining these linear relationships to be quadratic. In doing so, we maintain the integrity of LIME as a computationally quick and powerful explainer while adjusting for non-linear relationships using a quadratic approximation. We utilize this QLIME framework on a black box model predicting employee placement success to test its viability. The novelty of QLIME improves the mean squared error (MSE) that LIME's linear relationship provides at the local level. This proof of concept addresses LIME's linear limitation when boundaries are non-linear.

The remainder of this paper is organized as follows. Section II presents an overview of some examples of available explainers. Section III outlines the methodology of our black box model and QLIME implementation. In Section IV, we compare our results of using QLIME and LIME and discuss some implications in Section V. Finally, we present our conclusions and opportunities for future research in Section VI.

2 Overview of Explainer Frameworks

As modeling methods are developed with uninterpretable outputs, different approaches to create frameworks for explaining feature importance are undertaken. Most applications for AI fall into data categories for image, text, tabular, or any combination of these data types. However, not all explainers are versatile enough to accommodate all three data types. SHAP (SHapley Additive exPlanations) and LIME are the two primary explainers that can handle these three data types.

Interpretation of explanations must also be human understandable [13]. For example, the Principal Component Analysis (PCA) method may be used to reduce feature dimensionality to produce inputs to the model [14]. These inputs may not be human understandable, and therefore, the variables that feed the PCA should be considered in the explanation not the PCA's feature reduced set.

2.1 SHAP

In 2017, a unified framework for interpreting models called SHAP [15] was developed, named after the "Shapley Values" technique in game theory developed by L. S. Shapley. The Shapley values describe how much any feature contributes to the prediction of a model, and SHAP considers all possible permutations of a model. This makes SHAP comprehensive and as a result it is computationally intensive. SHAP is a post hoc explainer, meaning it is applied after the model, and uses a class called additive feature attribution. SHAP is model agnostic, and can be used with images and tabular data. However, due to the computational requirements of SHAP, we selected a LIME based post hoc explainer.

2.2 LIME

The LIME explainer is a local and thus less computationally intensive alternative to SHAP. LIME trains local interpretable models to approximate individual predictions in a series of steps. This process involves perturbing the inputs, followed by observing the output of the general (black box) model to understand how the predictions change with different observations. If perturbations at the local level produce a change in the general model, the feature is deemed important. The level of change determines the ranking of feature importance.

LIME explanations are determined by equation 1 below [13]. G represents potentially interpretable models such as general linear models and decision trees. Each individual model in G is represented by g. The measure of complexity of g is $\Omega(g)$. For example, the number of trees could be a measure of complexity in a decision tree. In this case, the number of trees would be represented by $\Omega(g)$. The general or black box model being explained is f. The fidelity or measure of how faithful g is in approximating f is L. Fidelity looks at how closely the local model represents the general model at the local level. The explanation of L occurs within the defined locality Π_x and the goal of LIME is to minimize L and $\Omega(g)$ [13] [16].

$$\xi(x) = argminL(f, g, \Pi_x) + \Omega(g), g \in G \quad (1)$$

The LIME technique can be a great option for creating interpretable model explanations but there are limitations. A linear function may not be enough to explain local perturbations for non-linear behaviors. In some cases, reducing the locality may be helpful but each problem may require creative and custom solutions and a local boundary that is too narrow risks efficacy. A deterministic approach to capture a sample relevant to the data may be more appropriate than random perturbations [17]. Figure 1 shows an example of weighted perturbations around a point of interest within local boundaries.

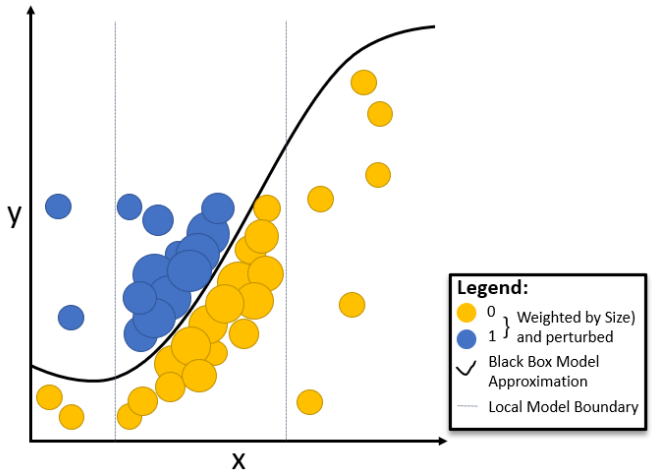

Fig. 1. Perturbed and Weighted Points Around the Point of Interest.

3 Methodology

The need for explainability arises from more complex, black box models that offer limited, if any, human understandable interpretations. While there are many models that are interpretable (such as generalized linear models or random forests), neural networks and ensembles are becoming more popular.

We use data from a staffing agency to create an ensemble of three models. This black box model serves as an example to which we can obtain explanations. Using the LIME framework as a foundation, we also introduce the intuition behind obtaining and using a quadratic approximation. LIME is available as an open source Python package. To implement our quadratic transformation, we created customized Python code that provides access to each step in the LIME framework shown in Figure 6.

3.1 Data

Our data comes from a global industrial staffing company utilizing a franchise model with hundreds of locations within the United States. The company has grown through acquisitions and as a result the data falls short on consistency in formatting and completion. To remedy data issues, we used a combination of data munging techniques using both Python and Alteryx to condition the data into a format appropriate for analysis.

Our final data set contains 64 features and one predictor label. Table 1 shows a categorized summary of the data with indications of created, existing, and consolidated features. To capture resume and job posting information, a Natural Language Processing (NLP) technique was used to engineer a similarity feature. The similarity calculation uses cosine similarity and a spaCy English dictionary, a technique leveraged for similar semantic analysis [18]. We created our target variable based on the number of hours an employee worked at a job with a success cutoff of 80+ hours. Data with expected assignments less than this cutoff were removed to prevent undue bias for the unsuccessful class. Our binary classifier models use 1 to indicate a job placement success and a 0 to represent unsuccessful job placements. The 0 and 1 indicators are used in remainder of this paper. Figure 2 shows the balance of the label which is relatively balanced with a 43/57 split.

Table 1. The Model Features and Label.

Category	Created	Existing	Consolidated	Grand Total
Geography		3		3
Job Details		2	24	26
Pay		1		1
Schedule	1	6		7
Skills and Experience	1	1	24	26
NLP	1			1
Label	1			1
Grand Total	4	13	48	65

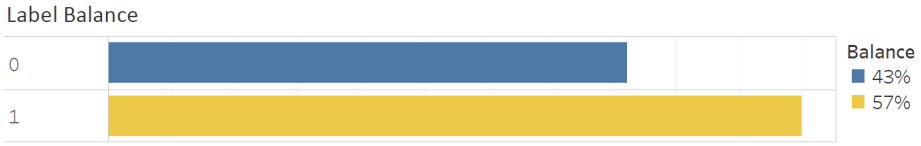

Fig. 2. Balance of Data Labels.

The data is comprised of 50MM+ records regarding employee or job geography, skills and skill requirements, experience, and availability needs and preferences. After cleaning and filtering to recent years (2017+) and relevant geographies, we were left with 2.5MM records. Figure 3 shows the employee placements made by the staffing company in the US.

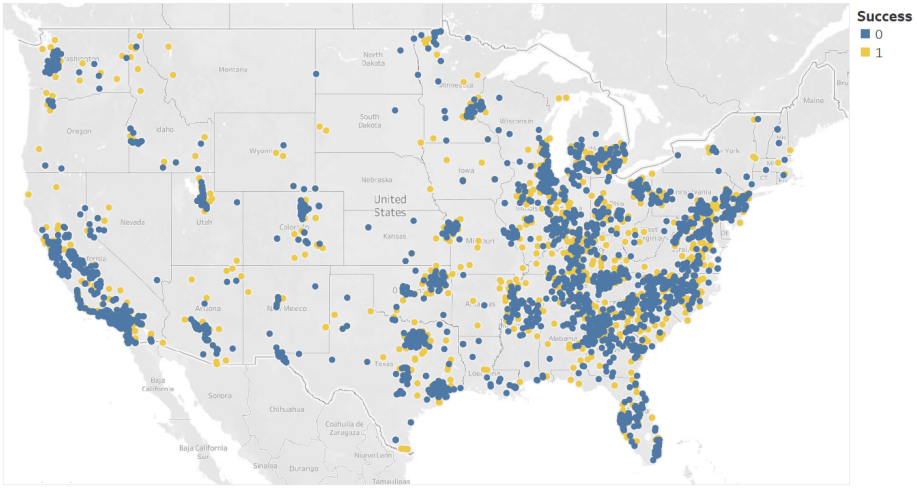

Fig. 3. Employee Placements in the US, color coded by Success.

3.2 Generating a black box model

While there are many cases where a single model is best suited for a problem, situations arise where multiple models compete against each other for their utility. Ensembling multiple models together is increasing in popularity but is limited in use due to the added complexity of doing so. For the purposes of creating a black box model with high complexity to gauge the performance of both QLIME and LIME, we look at a stacked ensemble of three models:

1. Generalized Linear Model (GLM): A common and well understood approach to predicting outputs based on given inputs are linear models. These models can be very elementary and use only one variable or can be complex and use a variety of variables to predict an output [19].
2. Random Forest (RF): Classifiers represent another approach to machine learning. A random forest classifier is a decision tree algorithm that utilizes several decision tree models to aggregate a single prediction [20].
3. Gradient Boosting Machine (GBM): An alternative decision tree approach is known as gradient boosting machine, where a gradient descent is used to gauge a forward stage-wise additive model and thus minimize the loss [21].

Each of the four models (GLM, RF, GBM, and ensemble) were created using H2O.ai, an open-source machine learning product. All models were trained and tested using an 80/20 train/test split. The ensemble model uses a stacked technique. This technique is an improvement over an averaging ensemble where each model contributes the same amount to the prediction. A weighted average ensemble weights the more successful models more than the others. Our stacked approach aims to improve upon the weighted average approach by taking the outputs of sub-models and uses them as inputs to learn the best way to combine the input predictions to optimize final predictions [22].

3.3 QLIME

The intuition behind LIME is to bridge the gap between interpretability and model creation (particularly those in the more complex categories), essentially turning "black box" algorithms into "white-box" ones. LIME's framework produces linear approximations of key features in order to demonstrate what features or attributes hold the greatest influential impact on our models. Incorrect patterns within the classifier outputs can then be corrected. While this brings interpretability into the ever-growing pool of classifiers, this process holds an inherent limitation by assuming all boundaries are linear [23]. Fundamentally, LIME relies on sparse linear explanations to approximate feature importance by looking at the magnitude and direction of coefficients. Least squared error is used as a loss function for the linear regressions. This brings quantitative and qualitative understanding of our models, but its reliance becomes heavily unfavorable in nonlinear relationships between features and predictions.

We relate this process to taking the derivative of a higher order (quadratic) function in order to arrive at a simpler, linear generalization of the relationship between a variable and our output within a certain localized boundary. This generalization is essentially the slope or correlation between these two entities. This slope is expected to change as our features change in value; for the purposes of this paper, the dynamic slope will be treated as a static one, as listed in our assumptions below.

For QLIME, we treat our linear approximations as tangential steps (derivatives) within a larger, more complex function. We use this approach to integrate our linear relationships to generate a quadratic function to better explain the relationship between our features and our outputs. Figure 4 shows an example of the improvement QLIME has over LIME for a non-linear boundary and the following assumptions are made:

1. The range (domain) of our feature(s) in LIME's linear approximations is equal to the range of our QLIME approximation and is dependent on our data
2. The linear regression captures the relationship between our features and predictions in a tangential fashion, and can therefore be represented as a derivative function of F, where F' = G.

3. The relationship between features and predictions then can be categorized as a higher order function by integrating G and reintroducing a bias, C:

$$\int G = F + C \tag{2}$$

4. The change in this relationship is either infinitely small or zero. That is, we make the assumption that there is only one curvature of best fit between our features and predictions.
5. For relationships where there is minimal gain in the loss function using a higher order function, a linear relationship will be approximated by the restricted range.

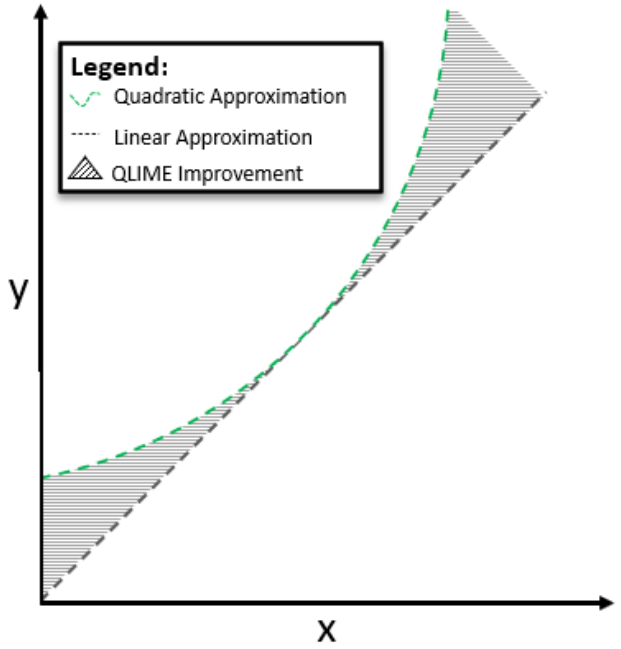

Fig. 4. A General Comparison of a Quadratic and Linear Approximation.

Where a linear relationship may result in some of our predictions being very close to or on the line itself, a higher order function may help differentiate these points into their proper categories. LIME outputs provide the weights of our features to indicate their significance and importance in their respective linear equations. Using these weights as a foundation, we can integrate them to backtrack and retrieve a higher order function in the following manner:

If the coefficient between feature x and prediction G is 0.8, then we can establish the equation:

$$G(x) = 0.8x \tag{3}$$

Taking the integral of this relationship results in:

$$\int G dx = 0.8 \int x dx \tag{4}$$

Since F' = G (as established in our assumptions above):

$$F = \int G dx \tag{5}$$

$$F = 0.4x^2 + C \tag{6}$$

A LIME coefficient that is greater than one will be more suggestive of a higher order relationship between our features and outputs due to a greater vertical stretch. A smaller LIME coefficient (between zero and one) will produce a vertical compression of our higher order relationship, yielding evidence of a more linear relationship between our features and our outputs. In both scenarios, we achieve a quadratic fit that is a parent function to our LIME linear equation and fit more uniformly to our data. Figure 5 shows where a quadratic fit to the data would improve a misclassification that a linear fit may produce.

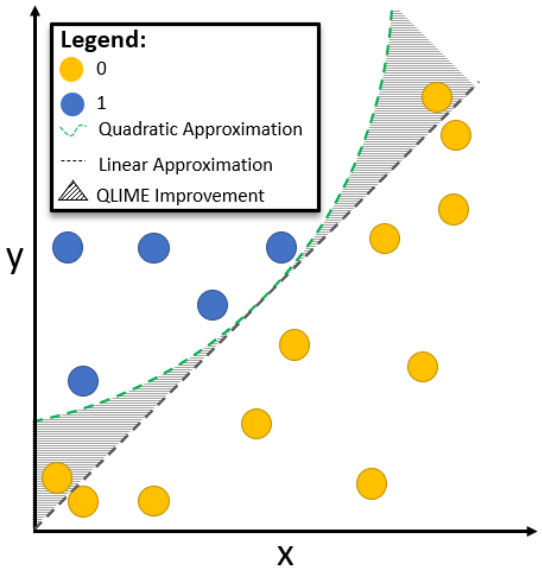

Fig. 5. QLIME Misclassification Improvement.

Our workflow in Figure 6 involves multiple steps starting with the execution of the black box model to generate predictions. The target data or job candidate to interpret is selected, and a dense field of perturbations around the target data are created and weighted. Weights are determined by calculating the Euclidean distance between the target data and neighborhood observations. These weights are included in the data to create predictions using the black box model. The predictions are added as a feature to the data set used in the local model. The local model in LIME uses a linear model and this is where the quadratic transformation occurs. The coefficients are used for to explain the predictions of the target data or candidate.

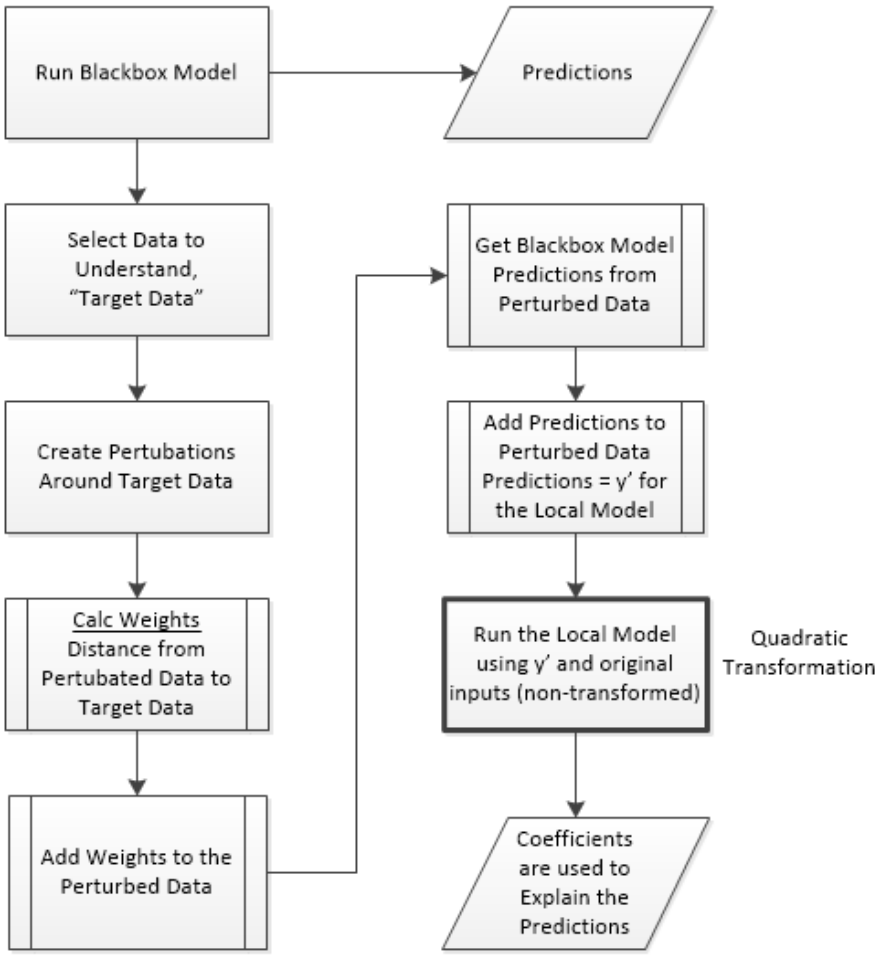

Fig. 6. QLIME Workflow.

4 Results

A generalized linear model, gradient boost model, random forest model, and stacked model using all three models were developed with parameters set to minimize the loss. The minimized loss is denoted by the area under the curve (AUC) value, a common performance metric for classification problems. These AUC values are shown in Table 2. We select the stacked ensemble model since it is a better representation of a black box model and refer to this model as our general model as opposed to the local model in LIME.

Table 2. The Model Results.

Model	AUC
GLM	0.6297
GBM	0.7093
RF	0.7648
Stacked Ensemble	0.7545

The top six features (of either ordinal, categorical, or numerical values) deemed as important for the general model using LIME are shown in Table 3 below. The most influential features are categorized as Job Shift Times, Location, Pay Rate, Education, Similarity of Job Description to Position Held, and Employee Job History. None of the features listed are surprising and make logical sense to impact the number of hours an employee works. For this data set in particular, neither LIME nor QLIME differ in their important features shown at the local level.

Table 3. Important Features.

Rank	Features
1	Job Shift Times
2	Location
3	Pay Rate
4	Education
5	Similarity of Job Description to Position Held
6	Employee Job History

To demonstrate the benefit of using QLIME over LIME, we select the most important continuous numeric feature, Pay Rate. Continuous data types best illustrate this type of improvement since we integrate over a continuous range of values local to the data. We iterate through several observations selected at random to determine the impact of introducing QLIME. Comparing the quadratic and linear fit to our data, we use the mean squared error (MSE) as our metric of comparison. In Table 4, we show the average MSE improvement that QLIME has over LIME for our label classifications.

Table 4. QLIME MSE Improvement Over LIME Using Pay Rate Feature.

Label Class	Improvement in MSE
0	3.8%
1	2.9%

5 Analysis

Using data from a staffing company to build a predictive black box model, our work shows that applying a quadratic transformation at the local level for a continuous variable, Pay Rate, improves the fidelity of the model's approximation over LIME's linear approximation. Here, we discuss the benefits of using a quadratic approximation, taking into account some implications on the locality aspect of this framework and combat against bias.

5.1 QLIME's Benefit

The important features remained the same in both LIME and QLIME. Both our quadratic proof of concept and LIME were implemented to quantify a continuous feature in both our class labels. QLIME's performance is validated as an improvement against LIME as measured by the MSE for Pay Rate on both label classes (Table 4). Although this improvement is small in magnitude, it is large in its implications. The quadratic estimator captured some subtleties within our data, and as a result, was a better fit for our data.

We attribute the improved performance to the non-linear approach within the local boundaries. In this paper, we demonstrated the benefit of using QLIME on tabular data with a non-linear boundary. LIME-based explainers can handle various other data types, and this improvement may be especially appreciated in image and audio data [24]. These data tend to encompass relationships that are non-linear in nature and their explainability may potentially benefit from a quadratic local model.

5.2 Locality Considerations

The locality of an observation is determined when fitting the local model by applying weights to the perturbed data. These weights are determined by calculating the distance between the perturbed data and the observation to be explained. There is no set rule for the locality (Π_x) range since the range is dependent on the data and model which requires some consideration.

A dense field of data around the target observation is recommended for best results to help the local model find boundaries [25] and a deterministic approach for creating perturbations can minimize unlikely data points in the perturbed data set [17]. An alternative method to generalize model decision logic involves input gradient penalties. This method is an extension of LIME's local perturbations, but can be more reliable in some instances [26].

5.3 Bias

There is a higher order need for explainability beyond understanding feature attribution, which is the need for model transparency to minimize algorithmic bias. Personally identifiable information (PII) may be necessary in some models, however this leaves room for models with discriminatory or unethical decisions. Although interpretability itself is not sufficient to prevent bias, explainability is necessary for evaluating at a high level why a model makes particular decisions [27].

Biased models can have real world impact. For instance, in crime data analysis, a biased algorithm that over- stated the risk probability of people of color lead to longer sentencing, higher bail and skewed recidivism predictions [28]. There has also been an increase in legislation that requires interpretability [29], and human understandable explanations of model outcomes [30].

In any case, bias can be introduced into models intentionally or unintentionally. Detection often relies on subject matter expertise to review the features captured by an explainer for unexpected feature importance values. Additional steps to check for evidence of bias disparity is recommended [31]. This is especially true of models that may have been designed to intentionally exploit data [30].

6 Conclusion and Future Work

In this paper, we propose a quadratic framework (QLIME) that addresses the linear limitations of LIME and helps generalize the interpretations of black box models. The expansion of LIME that QLIME provides maintains the integrity and efficiency of LIME as an explainer while providing quadratic relationships between variables and predictions.

The utility of QLIME is confirmed by comparing it to LIME in explaining a stacked black box ensemble model made of a generalized linear model, a random forest model, and a gradient boosting machine model. An improvement is seen using MSE as our comparison metric. This supports the proof of concept that the non-linear nature of QLIME offers an improvement over LIME, indirectly aiding the endeavor towards increased model adoption in industrial applications.

There are a number of methodologies to expand on the power of explainers. A limitation inherent to LIME (and subsequently QLIME) is the locality aspect. Expanding the local region would capture a larger population of possible values and further generalize the scope of work. Although varying by industrial and applicable domain, an investigation to expanding the local region warrants further investigation. However, larger ranges openly invite possible higher complexities. To offset this complication, increasing the order of our explainer beyond a quadratic approach to consider higher ordered polynomials (while keeping in mind the added complexity) is a viable option.

7 Acknowledgments

In memory of Jason Raguso (1971-2020), who provided guidance as our advisor on this project but passed before he could see our finished work. A special thanks to Nibhrat Lohia who adopted our team as our advisor.

References

1. Sindhu Ghanta, Sriram Subramanian, Swaminathan Sundararaman, Lior Khermosh, Vinay Sridhar, Dulcardo Arteaga, Qianmei Luo, Dhananjoy Das, and Nisha Talagala. Interpretability and reproducability in production machine learning applications. *arXiv*, 2018.
2. Berrada M. Adadi, A. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE*, 6, 2018.
3. Radwa el Shawi, Youssef Sherif, Mouaz Al-Mallah, and Sherif Sakr. Interpretability in healthcare: A comparative study of local machine learning interpretability techniques. *IEEE*, 10.1109/CBMS.2019.00065, 2019.
4. Nina Olofsson. *A machine learning ensemble approach to churn prediction: Developing and comparing local explanation models on top of a black-box classifier*. PhD thesis, KTH Royal Institute of Technology School of Computer Science and Communication, Stockholm, Sweden, 6 2017.
5. Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José M. F. Moura, and Peter Eckersley. Explainable machine learning in deployment. *ArXiv*, abs/1909.06342, 2019.
6. D. Martens E. J. de Fortuny. Active learning-based pedagogical rule extraction. *IEEE transactions on neural networks and learning systems*, 26(11):2664–2677, 2015.
7. J. Decosmo. What nobody tells you about machine learning. *Forbes*, 4 2019.
8. Ramon-Jeronimo J. M. Florez-Lopez, R. Enhancing accuracy and interpretability of ensemble strategies in credit risk assessment. a correlated-adjusted decision forest proposal. *Elsevier*, 42, 2015.

9. Gajendra Katuwal and Robert Chen. Machine learning model interpretability for precision medicine. *arXiv*, 2016.
10. Haoxi Zhong, Yuzhon Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. Iteratively questioning and answering for interpretable legal judgment prediction. *Association for the Advancement of Artificial Intelligence*, 2020.
11. Hani Hagrass. Towards human-understandable, explainable ai. *IEEE*, 2018.
12. Gill N. Hall, P. *An Introduction to Machine Learning Interpretability*, pages 5–39. O'Reilly, Sebastopol, CA, 2019.
13. Mark Ibrahim, Melissa Louie, Ceena Modarres, and John W. Paisley. Global explanations of neural networks: Mapping the landscape of predictions. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, page 279–287, New York, NY, USA, 2019. Association for Computing Machinery.
14. Arnaz Malhi. Pca-based feature selection scheme for machine defect classification. *IEEE transactions on instrumentation and measurement*, 53(6):1517–1525, 12 2004.
15. Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *NIPS*, 2017.
16. Alexandra Hotti, Jacob Malmberg, and Marcus Ohman. *Implementing Machine Learning in the Credit Process of a Learning Organization While Maintaining Transparency Using LIME*. PhD thesis, KTH Royal Institute of Technology School of Computer Science and Communication, Stockholm, Sweden, 2018.
17. Muhammad Rehman Zafar and Naimul Mefraz Khan. Dlime: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems. *arXiv*, 2019.
18. Yun Zhu, Faizan Javed, and Ozgur Ozturk. Semantic similarity strategies for job title classification. *CoRR*, abs/1609.06268, 2016.
19. Nelder J.A. McCullagh, P. *Generalized Linear Models*, pages 21–41. CRC Press LLC, Sebastopol, CA, 1989.
20. A. Cutler, D. Cutler, and Stevens J.R. *Ensemble Machine Learning: Random Forests*. Springer, Boston, MA.
21. Lubke G.H. Miller, P.J. and Bergeman C.J. McArtor, D.B. Finding structure in data using multivariate tree boosting. *Psychological Methods*, 21(4):583–602, 2016.
22. Polley E.J. Hubbard A.E. Van der Laan, M.J. Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1):583–602, 2007.
23. Marco Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. *KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135 – 1144, 8 2016.
24. Sturm B.L. Dixon S. Mishra, S. Local interpretable model-agnostic explanations for music content analysis. In *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*, pages 537 – 543, Suzhou, China, October 2017.
25. et. al. Altman, T. Limitations of interpretable machine learning. https://compstat-lmu.github.io/iml_methods_limitations/lime-neighbor.html, 8 2019. Student Seminar, Computational Statistics, LMU 2019.
26. Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. *International Joint Conference on Artificial Intelligence*, 2017.
27. Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Michael Specter, and Lalana Kagal. Explaining explanations: An overview of in-

terpretability of machine learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018.
28. Julie Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias, Mar 2019.
29. Wojciech Samek and Klaus-Robert Müller. *Towards Explainable Artificial Intelligence*, pages 5–22. Springer International Publishing, Cham, 2019.
30. Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods, 2019.
31. Virginia Tsintzou, Evaggelia Pitoura, and Panayiotis Tsaparas. Bias disparity in recommendation systems. *ArXiv*, abs/1811.01461, 2018.