Kernel function:

Original lime has used Ridge as kernel function and in this project, we will implement linear regression, Lasso and logistic regression in lime to observe the effect of different kernel functions on the performance of lime.

Linear regression: linear regression refers to utilizing linear regression function to model the relationship between dependent variable (target) with one or more independent variables (features). The goal of linear regression is to fit a line, a plane or a higher-dimensional hyperplane so that the prediction given by linear regression model could be as close as possible to actual (real) data. For one instance, the relationship between target and features can be represented by linear regression model as follow, it could be easily seen as weighted sum of different features,

where

represents prediction of target,

represents feature,

represents intercept, it represents the feature weight for ‘constant feature’, which is always 1 in all instances. The meaning of is it represents prediction of target when features has been standardised and are at their mean value,

represents feature weight for different feature, also called coefficient. It reflects the effect of feature on prediction,

represents error term which reflects the difference between prediction and actual data.Both negative errors and positive errors are taken into consideration.

The ordinary least squares method is used in linear regression to minimize the squared difference between prediction of target and actual target, the formula is as below,

To solve overfitting and collinearity problem in standard linear regression, Lasso and Ridge are implemented in lime as well .

Lasso: Lasso is short for ‘least absolute shrinkage and selection operator’. Based on linear regression, Lasso operates feature selection and regulates selected feature weight. It constructs a penalty function in linear regression model where L1-norm is used. Here L1-norm refers to the sum of absolute value of feature weight should be less than or equal to a threshold. With penalty, values of some feature weight (coefficients) are estimated as 0 and some are shrunk, thereby achieving feature selection and feature weight regulation. The objective function(目标函数) of Lasso is as below,

where

represents L1-norm,

represents shrinkage factor, the larger the , the more feature weights are estimated as 0. As a result, regression model has fewer features and more easier to understand.

Ridge: Similar to Lasso, Ridge also constructs a penalty function based on standard linear regression. However, Ridge uses L2-norm as penalty function, here L2-norm refers to the sum of squares of feature weights should be less than or equal to a threshold. Therefore, no feature weights are shrunk to 0 in Ridge. The objective function of Ridge is as below,

where

represents L2-norm,

represents shrinkage factor, the larger the , the more feature weights are shrunk in Ridge.

Logistic regression: logistic regression is a classification method, usually for binary classification (class=A, class=B). Instead of fitting a line, a plane or a higher-dimensional hyperplane, logistic regression uses logistic function to transform the value of output in linear equation to [0, 1]. The result in logistic regression represents the probability for a particular instance which is predicted as class A. The logistic function is defined as below,

,

Replace in formula with linear equation in linear regression: and the equation of logistic regression as below,

where

represents the probability of this particular instance that is predicted as class 1.

A threshold is set in logistic regression, when the output is greater than threshold, then the instance is predicted as class = 1, otherwise, the instance is predicted as class = 0.

Evaluation Metrics

In order to reasonably evaluate the performance of lime, we have formulated two evaluation metrics, fidelity and interpretability.

Fidelity: fidelity is used to measure the similarity between the classification prediction given by lime and the black box model. It could be also interpreted as the proportion of instances that lime and black box model make the same classification prediction in the whole dataset. The equation of fidelity is shown as below,

,

where

represents the number of instances that lime and black box model make the same classification prediction

represents the total number of instances in the whole dataset

The calculation process of fidelity is as follow,

1. For each instance, calculate the final classification prediction made by lime. Lime classifies features into two types (for simplicity use positive features and negative features), corresponding to different class of target and also outputs corresponding feature weight. According to sum of feature weight for different type of feature, the large type will be chosen as the final classification prediction of target. The output of lime is shown as below,

|  |  |  |  |
| --- | --- | --- | --- |
| negative feature | | positive feature | |
| feature 5 | weight5 | feature 1 | weight1 |
| feature 6 | weight6 | feature 2 | weight2 |
| feature 7 | weight7 | feature 3 | weight3 |
| feature 8 | weight18 | feature 4 | weight4 |

Two calculation steps are shown as below.

1. Calculate the sum of positive feature weight and negative feature weight.
2. Compare positive feature weight and negative feature weight,

If , then the instance will be classified as positive; else if , then the instance will be classified as negative.

1. For each instance, compare the classification prediction made by lime and black box model, if they are the same, then the value of is added by 1, .
2. After comparing every instance in the dataset and getting the value of , calculate .

The higher the fidelity, the more similar between lime and black box model and the better the performance of lime.

*Interpretability:* Interpretability is used to measure the ability of lime to explain black box model. (Whether the rank of feature weight of different features given by lime is reasonable.) Two indicators are combined to represent the interpretability, the number of important features and the sensitivity-weight value.

*The number of important features:* important features refer to features whose absolute value of feature weight is greater than or equal to threshold we set. In experiment, the threshold is set to 0.015. In lime, the value of feature weight is in the interval [0, 1] and we only focus on the features that we think they are important. When humans receive explanation, usually they prefer to simple explanation rather than a complex one, therefore, the less the number of important features, the simple the explanation and the better the performance of lime.

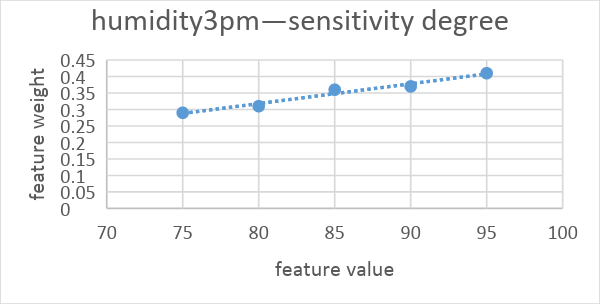
*Sensitivity-weight value*: sensitivity-weight value is used to measure rationality of feature weight given by lime. In this paper, we define the concept of sensitive feature. Sensitive feature is contrastive. When two features both change the same value, such as increasing or decreasing 20%, we observe the change of feature weight in lime, the feature with more change of feature weight is more sensitive than the feature with fewer change of feature weight and thereby we could rank the sensitivity degree of different features.

The calculation process of Sensitivity-weight value is as below, here take one instance as an example

1. For features whose absolute value of feature weight is more than or equal to 0.015 (important features), rank features according to the feature weight, the result is as below,

|  |  |  |
| --- | --- | --- |
| feature | weight | weight rank |
| feature 1 | 0.2 | 4 |
| feature 2 | 0.4 | 2 |
| feature 3 | 0.3 | 3 |
| feature 4 | 0.5 | 1 |

1. For important features, calculate sensitivity degree of these features and rank the feature according to sensitivity degree. Here take feature = feature1 as an example.
   1. Increase value of feature1 5% every time until it achieves the upper bound, record feature value and corresponding feature weight output by lime each time, then apply linear regression on data (feature value, feature weight) to fit a line. The slope of the line can be used as sensitivity degree since it shows the effect of feature value on feature weight. The bigger the slope, the more sensitive the feature and the higher the sensitivity degree. The line is fitted as below,



* 1. Rank all features based on sensitivity degree. The result is as below,

|  |  |  |
| --- | --- | --- |
| feature | sensitivity degree | sensitivity degree rank |
| feature 1 | 0.2 | 4 |
| feature 2 | 0.4 | 2 |
| feature 3 | 0.3 | 3 |
| feature 4 | 0.5 | 1 |

1. Compare weight rank with sensitivity degree rank and calculate sensitivity-weight value. The calculation steps are as below,
   1. Disparity is used to represent the difference value between weight rank and sensitivity degree rank for each feature, disparity = |weight rank – sensitivity degree rank|, for each instance, calculate the disparity, the result is as below,

|  |  |  |  |
| --- | --- | --- | --- |
| feature | weight rank | sensitivity degree rank | disparity |
| feature 1 | 1 | 3 | 2 |
| feature 2 | 2 | 1 | 1 |
| feature 3 | 3 | 2 | 1 |
| feature 4 | 4 | 4 | 0 |

* 1. Calculate the Sensitivity-weight value for one particular instance, the equation is as below,

(要不要除以feature个数？)

Where

is the sum of disparity of all features in one instance

is the number of features in this instance

4．Calculate the sensitivity-weight value for the whole dataset. Sum the sensitivity-weight value for every instance and divided by the number of instances in the dataset.

The less the sensitivity-weight value, the more reasonable the feature weight given by lime.