SHAP

合肥工業大學

李诗惠

2022/7/12

简介

可解释的机器学习是指使人类可以理解机器学习系统的行为和预测的方法和模型。部分学者认为**可解释性**是人类能够理解决策原因的程度；或者说可解释性是人类能够一致地预测模型结果的程度。

A Unified Approach to Interpreting Model Predictions

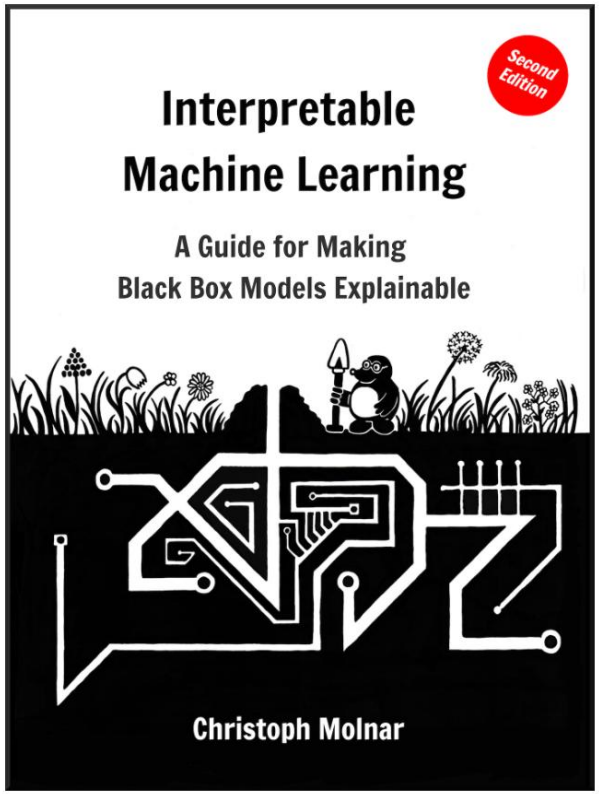
Scott M. Lundberg
Paul G. Allen School of Computer Science
University of Washington
Seattle, WA 98105
slund1@cs.washington.edu

Su-In Lee
Paul G. Allen School of Computer Science
Department of Genome Sciences
University of Washington
Seattle, WA 98105
suinlee@cs.washington.edu

ARTICLES
https://doi.org/10.1038/s42256-019-0138-9
nature machine intelligence

From local explanations to global understanding with explainable AI for trees

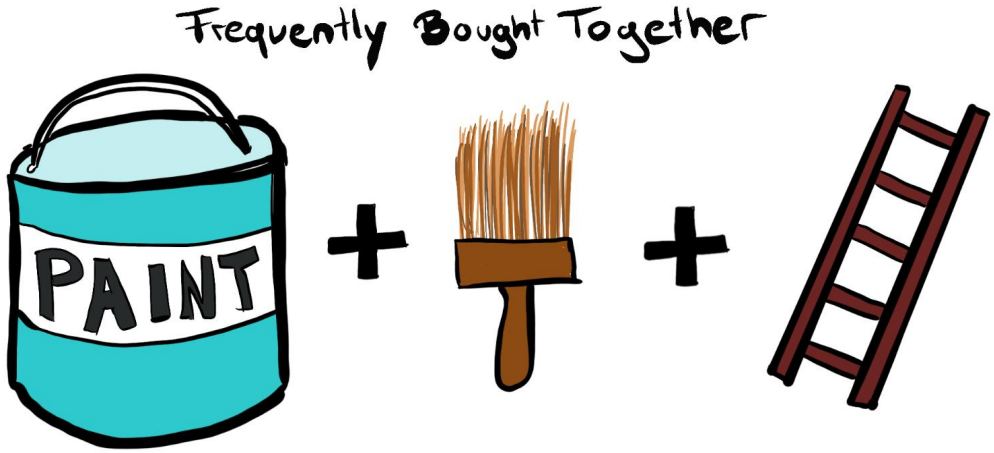
Scott M. Lundberg[1,2], Gabriel Erion[2,3], Hugh Chen[2], Alex DeGrave[2,3], Jordan M. Prutkin[4], Bala Nair[5,6], Ronit Katz[7], Jonathan Himmelfarb[7], Nisha Bansal[7] and Su-In Lee[2*]



简介

某些情况下，人们只关心机器学习产生某些预测或行为的结果，并不关心为什么会有这样的结果。然而，了解“为什么”可以帮助人们更好地了解问题、数据以及模型可能失败的原因。机器学习模型的可解释性越高，人们就越容易理解为什么做出某些决定或预测。

生病原因



产品推荐：基于经常购买的产品组合

简介

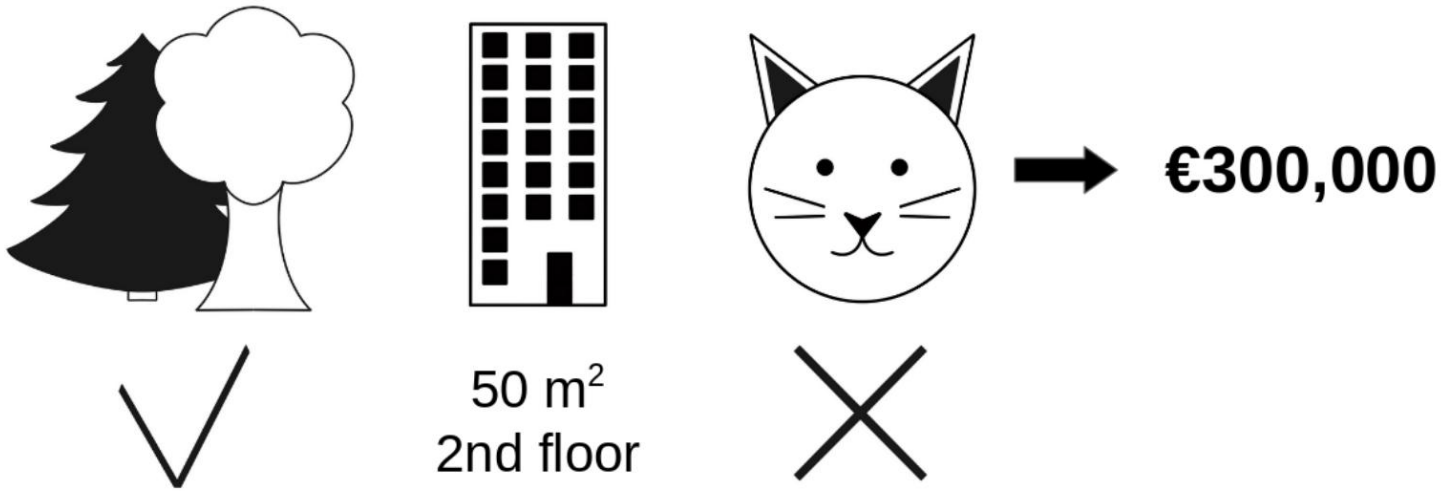
目前许多解释机器学习模型局部预测的方法都属于可加特征归因方法，比如LIME、DeepLIFT、分层相关传播、Shapley回归值、Shapley采样值和定量输入影响，而**SHAP(SHapley Additive exPlanations)**则是统一了上面六种方法的解释预测框架。

SHAP的应用方向有很多，比如TreeExplainer、DeepExplainer、GradientExplainer、KernelExplainer。TreeExplainer是TreeSHAP的实现，TreeSHAP用于基于树的机器学习模型，例如决策树、随机森林和梯度提升树。

Shapley值

SHAP基于博弈论最优的**Shapley值**解释个体预测，SHAP的目标是通过计算每个特征对预测的贡献来解释实例 x 的预测。

假设已经训练了一个机器学习模型来预测公寓价格。有一个公寓面积为50m^2，位于二楼，附近有公园，禁止养猫，这个公寓价格的预测是300,000欧元。当所有公寓的平均预测为310,000欧元时，与平均预测相比，每个特征值对预测的贡献是多少?

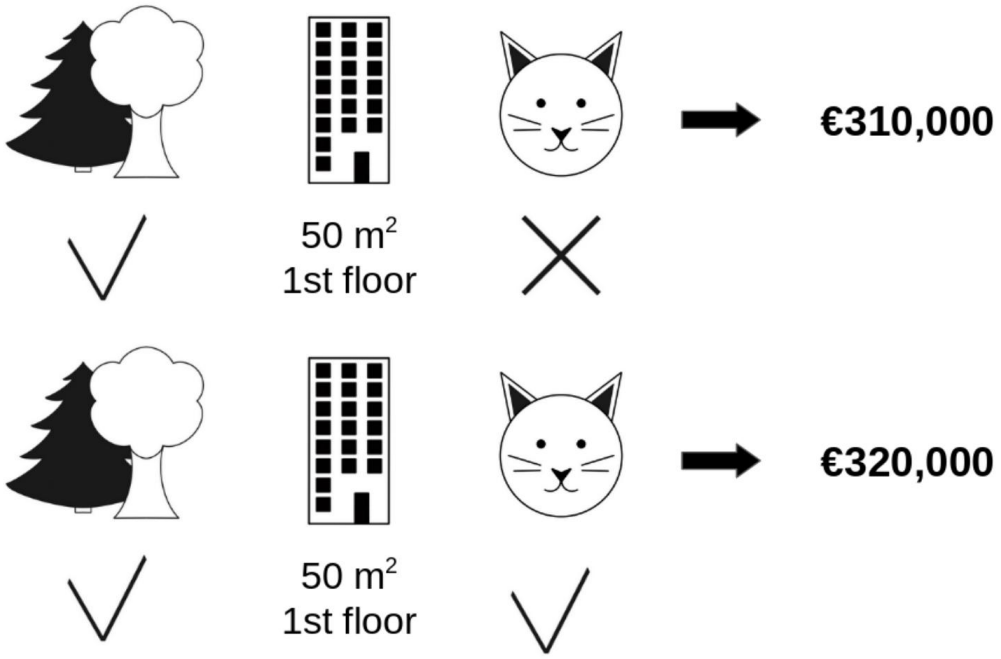


Shapley值

如何计算目标公寓实例（park=nearby，cat=banned，size=50，floor=2nd）其中一个特征的Shapley值？以该公寓实例的cat=banned为例：

对于这些联盟中的每一个，我们计算带有和不带有cat=banned特征值的预测公寓价格，并计算差值来获得边际贡献，**Shapley值是所有可能联盟中特征值的(加权)平均边际贡献。**

- 空联盟
- park=nearby
- size=50
- floor=2nd
- park=nearby 和 size=50
- park=nearby 和 floor=2nd
- size=50 和floor=2nd
- park=nearby 和size=50 和floor=2nd.



SHAP

SHAP将模型的预测值解释为每个输入特征的归因值之和：

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j$$

g是解释模型，z': 相应特征是否能被观察到(1或0)，M是输入特征的数目，ϕ_j是每个特征的归因值(Shapley值)，ϕ_0是解释模型的常数(所有训练样本的预测均值)。

由于树模型的输入必须是结构化数据，对于实例 x，z'应该是所有值为1的向量，即所有特征均能被观察到的，于是该公式简化为：

$$g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j$$

SHAP

SHAP在Shapley值满足效率性，对称性，虚拟性和可加性的基础上，还具有三个理想的属性：局部准确性(Local accuracy)，缺失性(Missingness)和一致性(Consistency)。

① **局部准确性：**表示特征归因的总和等于我们要解释的模型的输出，也就是说对于每一个样本，各个特征的归因值与常数归因值之和等于模型的输出值$f(x)$。

$$\hat{f}(x) = g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j x'_j$$

② **缺失性：**$x'_j = 0$ 的特征没有归因影响。(某个特征在实例中观察不到)

$$x'_j = 0 \Rightarrow \phi_j = 0$$

③ **一致性：**如果模型发生更改，使得特征值的边际贡献增加或保持不变(与其他特征无关)，则归因值不应降低。

SHAP

SHAP值的计算过程：

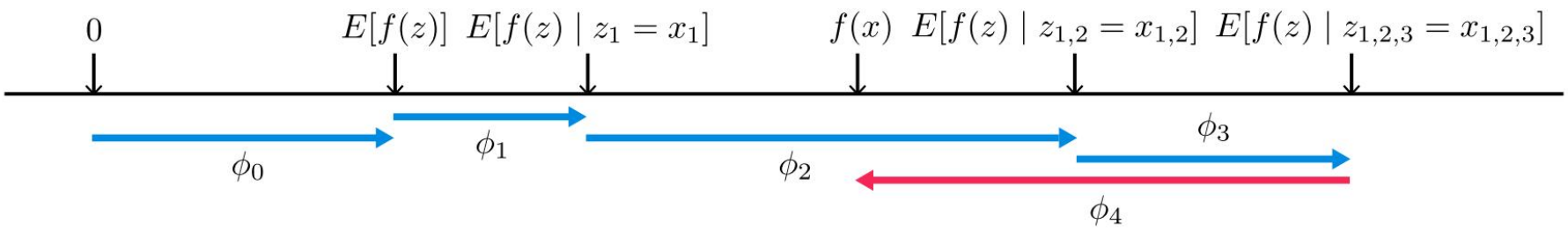
以模型 f 对于样本$\{x_1 = a_1, x_2 = a_2, x_3 = a_3, x_4 = a_4\}$的预测解释为例。

①S为空集时，$\phi_0 = f_x(\emptyset) = E[f(x)]$，其中$E[f(x)]$为模型预测值的期望，可用训练样本的模型预测值的平均值近似。

②S顺序加入特征x_1，此时$\phi_1 = f_x(\{x_1\}) - f_x(\emptyset) = E[f(x)|x_1] - E[f(x)]$，即$\{x_1 = a_1\}$时的模型预测值期望−模型预测值期望。

……

⑤直至加入最后一个特征x_4。



注：此图只显示单个排序情况。

SHAP

在实际情况下，当模型是非线性的或输入特征不是独立时，SHAP值应该对所有可能的特征排序计算加权平均值。

$$\phi_j = \sum_{S \subseteq \{x_1, \cdots, x_p\} \setminus \{x_j\}} \frac{|S|!(p - |S| - 1)!}{p!} (f_x(S \cup \{x_j\}) - f_x(S))$$

$\frac{|S|!(p - |S| - 1)!}{p!}$ 为所有可能的特征排列情况

$\{x_1, \ldots, x_p\} \setminus \{x_j\}$为不包括$\{x_j\}$的所有输入特征可能的集合

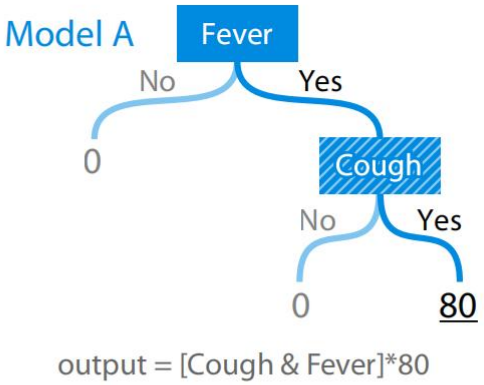
$f_x(S)$为特征子集S的预测情况

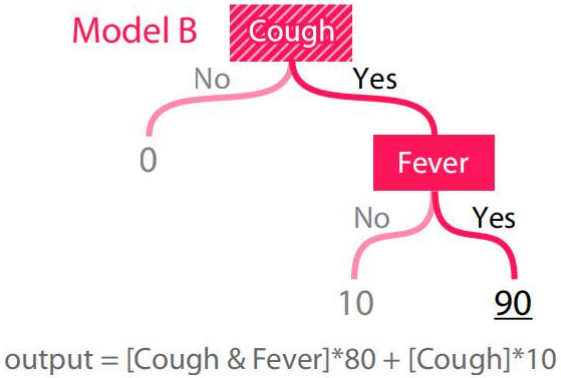
$f_x(S \cup \{x_j\})$为在特征子集S的基础上加特征x_j的预测情况

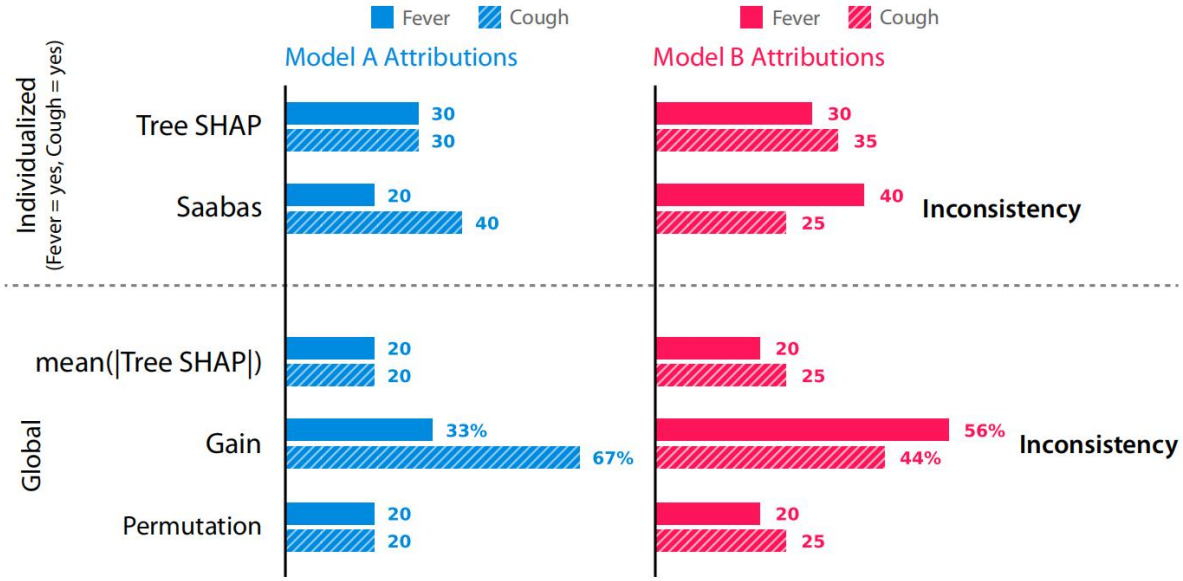
TreeSHAP

SHAP值是唯一**一致**的**个性化**特征归因方法。

- 对于二元特征发烧(Fever)和咳嗽(Cough)，模型A只是一个简单的"和"函数，模型B是相同的函数，但是当为咳嗽时预测值会增加(加10分)，使得模型更依赖于咳嗽，这时因咳嗽更重要，导致在模型B中咳嗽先分裂。

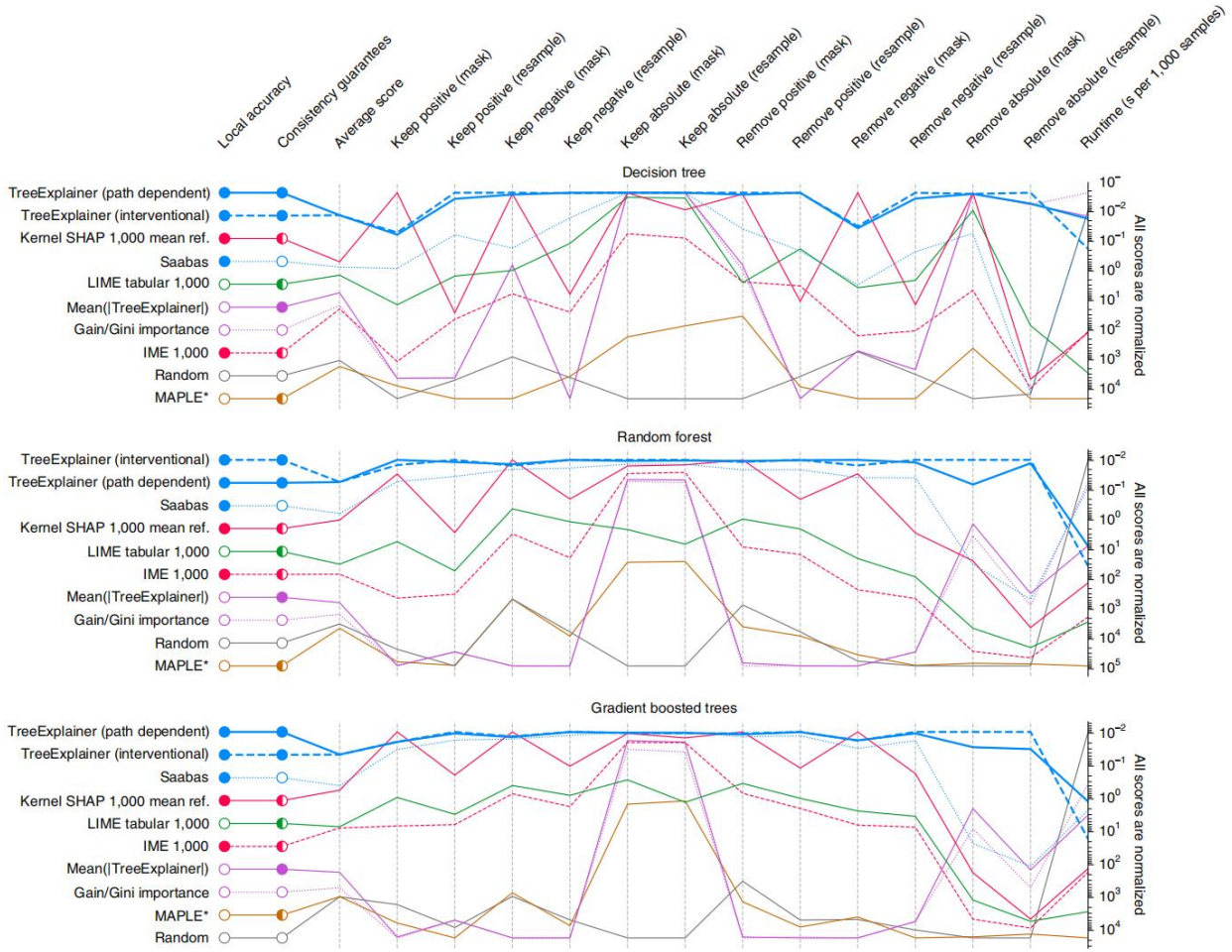






TreeSHAP

TreeExplainer的表现优于以前的方法，不仅因为它有关于一致性的理论保证，而且还通过一组测量解释质量的量化指标展示了改进的性能。



- 解释方法在慢性肾脏疾病数据集的15种不同评价指标和3种分类模型上的表现。
- 每一列表示一个评估度量，每一行表示一种解释方法。

TreeSHAP可视化应用

(1) 安装SHAP

```
pip install shap
```

或

```
conda install -c conda-forge shap
```

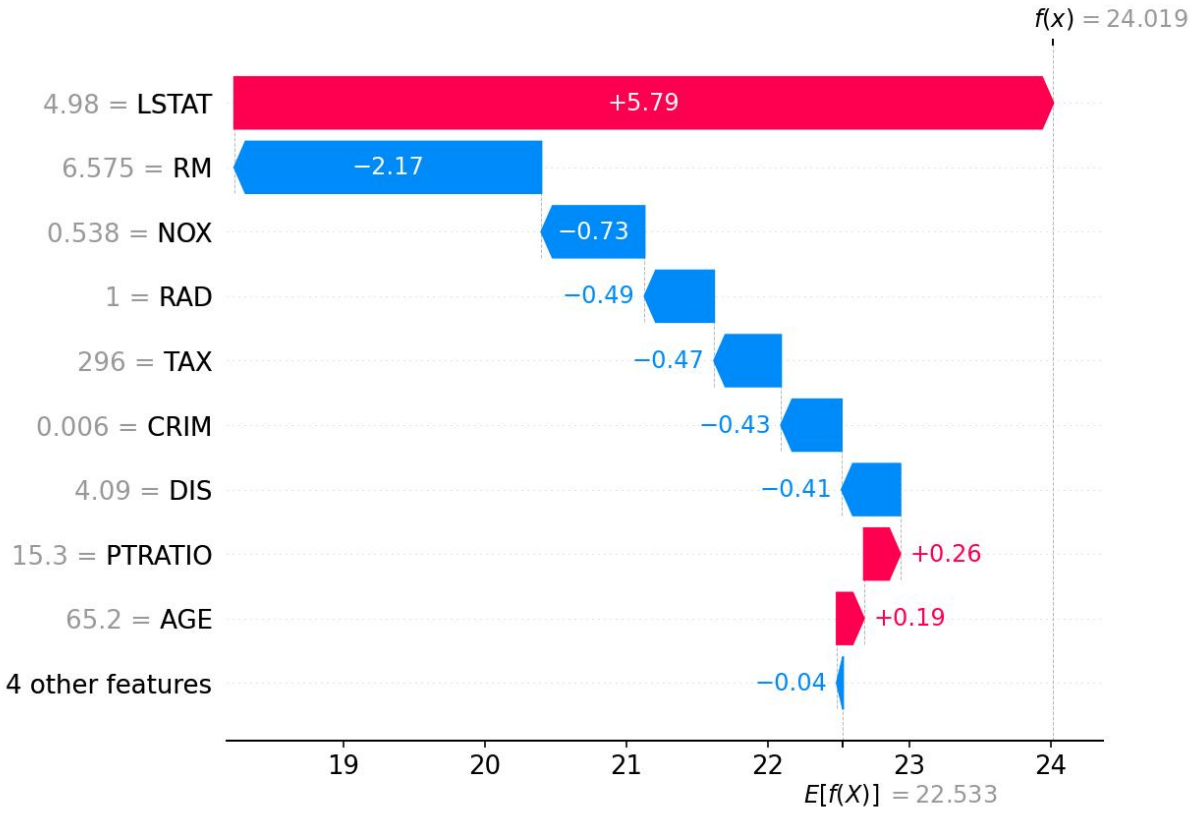
(2) 训练并解释模型

```
import xgboost
import shap

# train an XGBoost model
X, y = shap.datasets.boston()
model = xgboost.XGBRegressor().fit(X, y)

# explain the model's predictions using SHAP
# (same syntax works for LightGBM, CatBoost, scikit-learn, transformers, Spark, etc.)
explainer = shap.Explainer(model)
shap_values = explainer(X)
```

TreeSHAP可视化应用

(3) 可视化

```
# visualize the first prediction's explanation
shap.plots.waterfall(shap_values[0])
```

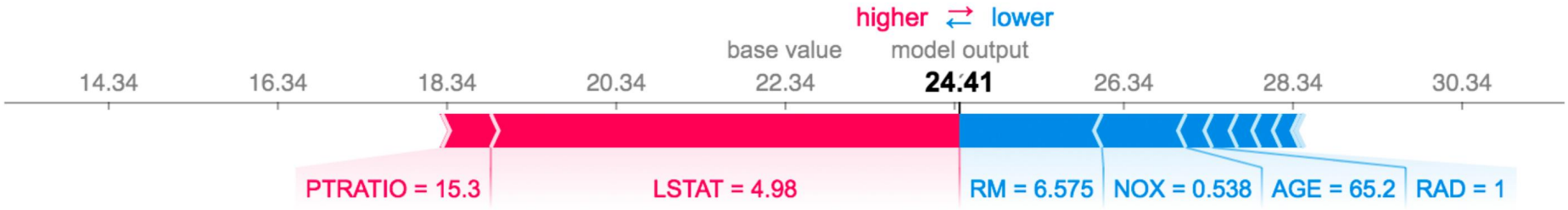


waterfall图：显示了每个有助于将模型输出从基值推到最终输出的特征。将预测推高的特征以红色显示，将预测推低的特征以蓝色显示。

TreeSHAP可视化应用

force图：将诸如Shapley值之类的特征属性可视化为“力”，每个特征值都是增加或减少预测的力量，这些力量在数据实例的实际预测中相互平衡。

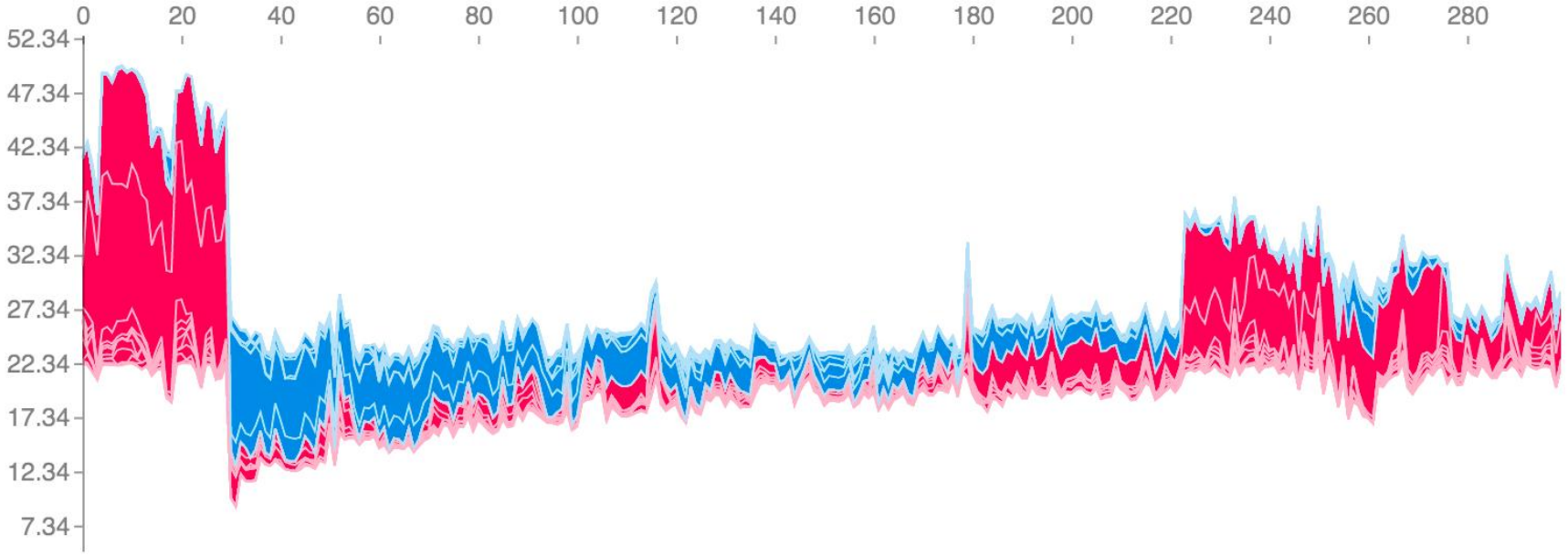
```
# visualize the first prediction's explanation with a force plot
shap.plots.force(shap_values[0])
```



TreeSHAP可视化应用

该图由许多force图组成，每个force图都解释了一个实例的预测，垂直旋转force图，并根据它们的聚类相似性将它们并排放置。

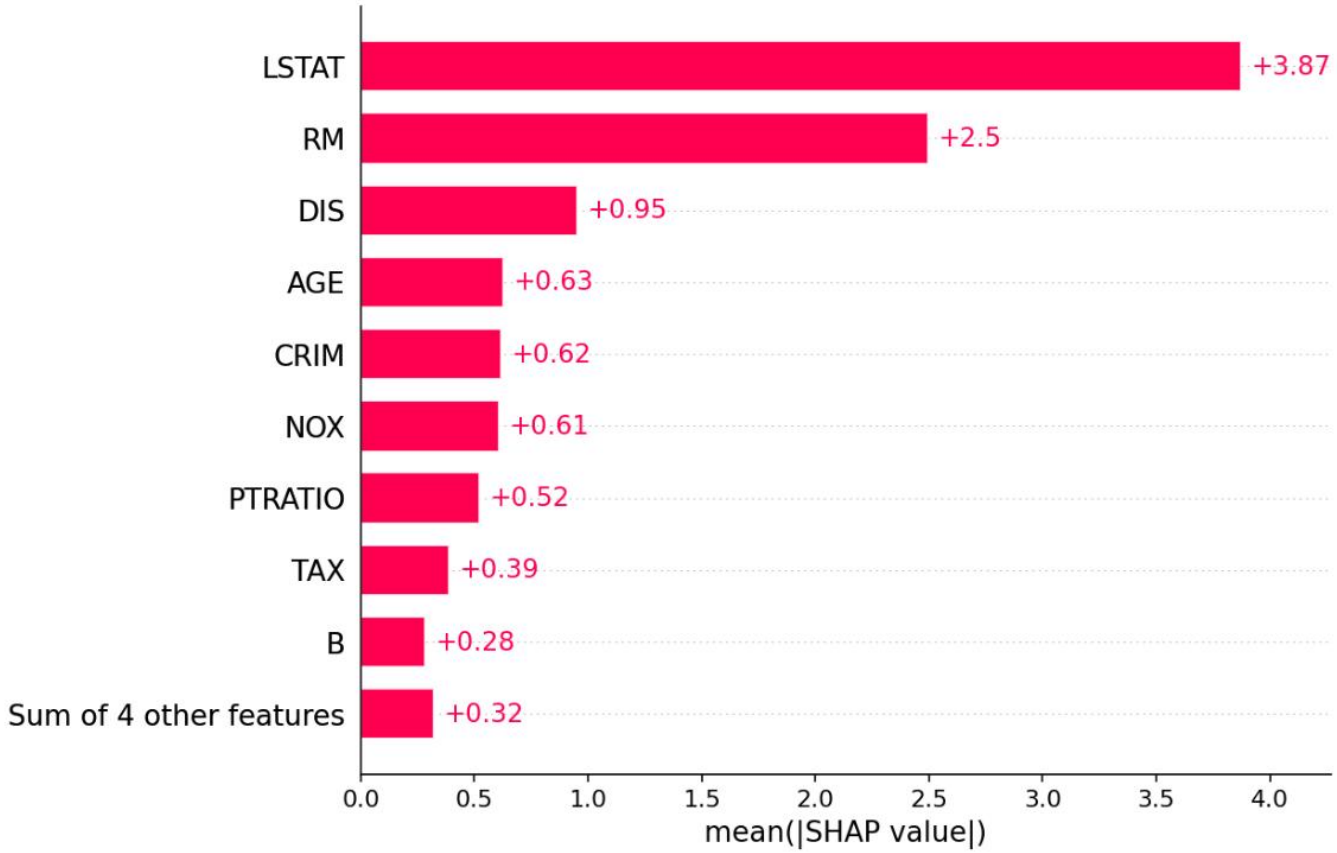
```
# visualize all the training set predictions
shap.plots.force(shap_values)
```



TreeSHAP可视化应用

通过对数据中每个特征的绝对Shapley 值进行平均得到全局重要性。

```
shap.plots.bar(shap_values)
```

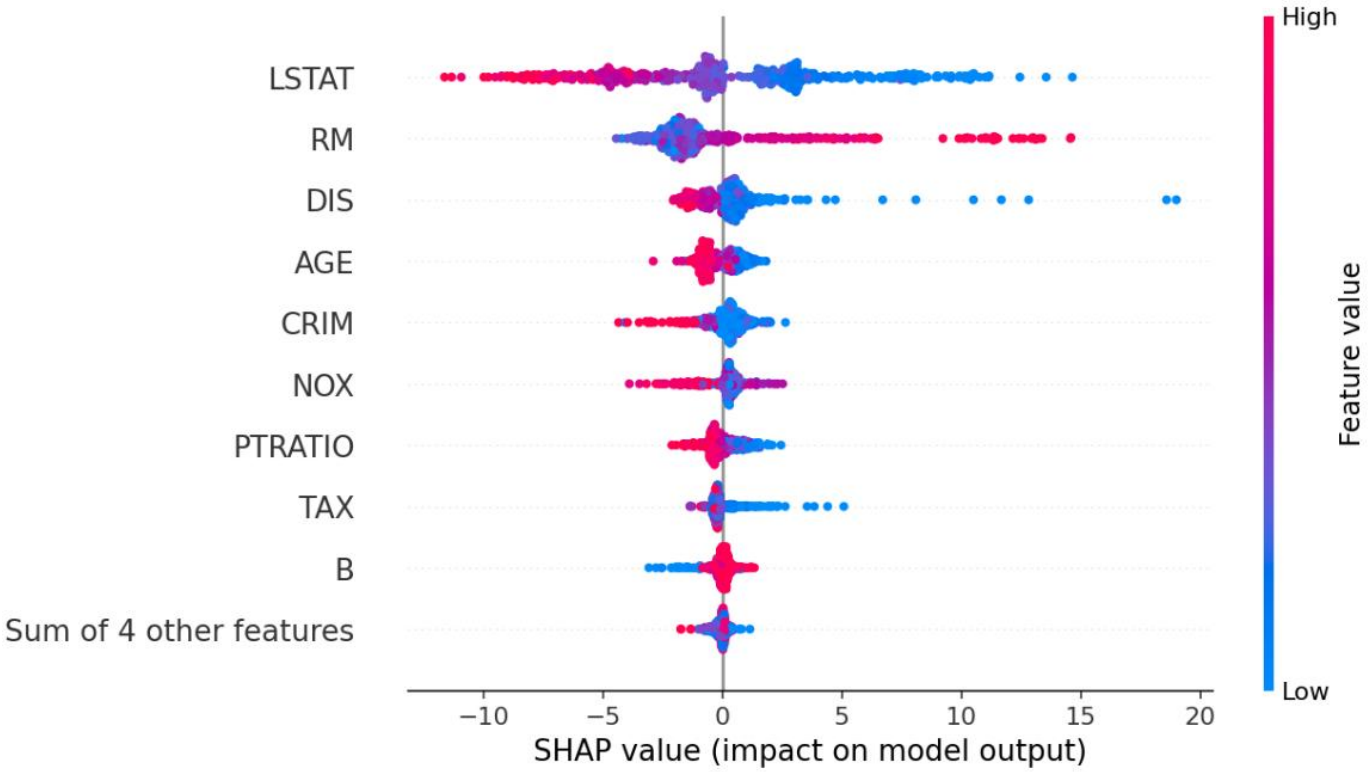


TreeSHAP可视化应用

beeswarm图：每个点都是特征和实例的Shapley值，重叠点在y轴方向抖动。这些特征是根据它们的重要性排序的。

```
shap.plots.beeswarm(shap_values)
```

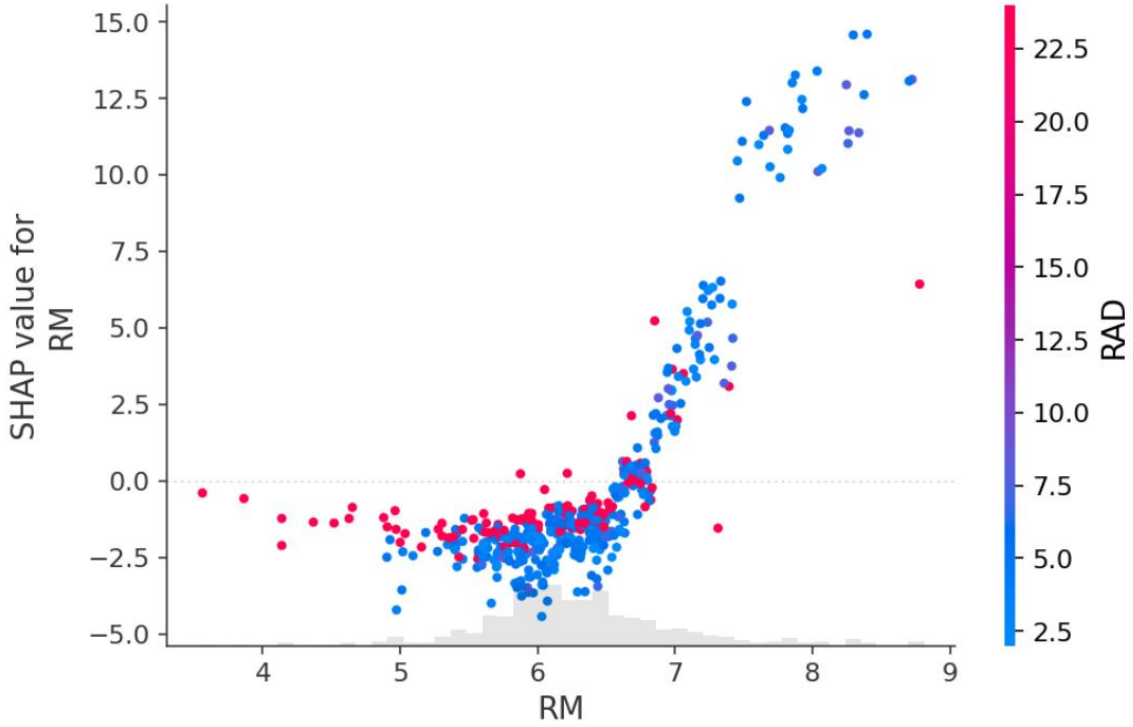
```
shap.summary_plot(shap_values, X_test)
```



TreeSHAP可视化应用

依赖散点图：单个RM值的垂直离散度表示与其他特征的相互作用效应。为了帮助揭示这些相互作用，我们可以用另一个特征着色。我们将整个Explanation对象传递给color参数时，散点图会尝试挑选出与RM交互作用最强的特征列。

```
# create a dependence scatter plot to show the effect of a single feature across the whole dataset
shap.plots.scatter(shap_values[:,"RM"], color=shap_values)
```



Q&A

合肥工業大學