Projeto Final de Programação

Relatório Técnico

PUC-Rio
Departamento de Informática
Pós-Graduação em Informática

Professora

Clarisse Sieckenius de Souza

Aluno
Bianca Moreira Cunha - 2220956

Turma
2023.2

Data de Entrega
08 de dezembro de 2023

Sumário

1. Introdução

Este relatório apresenta o trabalho desenvolvido para a disciplina de Projeto Final de Programação. Neste trabalho foi desenvolvida a biblioteca ExpViz, uma biblioteca python com o objetivo de reunir em um pacote implementações métodos de explicação de modelos de machine learning que vêm sendo muito utilizados na comunidade de XAI (Explainable AI), como SHAP e LIME. A ideia é facilitar o acesso a diferentes métodos para que esses possam ser comparados e analisados de forma mais ágil e também de iniciar uma tentativa de melhoria da interpretabilidade, uma vez que junto com as visualizações, a biblioteca também gera explicações verbais parametrizadas para a predição ou o modelo.

2. Descrição e objetivos gerais do Software

A área de Explainable AI vem crescendo significativamente nos últimos anos, por conta da necessidade de tornar possível interpretar e entender decisões tomadas por modelos chamados de caixa-preta, como redes neurais, principalmente quando eles são utilizados para questões sensíveis. Alguns métodos para melhorar a interpretabilidade de modelos foram propostos, mas há uma falta de avaliação para entender se as suas saídas realmente ajudam a melhorar a interpretabilidade ou não.

Tendo isso em mente, a motivação para a criação da biblioteca ExpViz foi inicialmente para facilitar o trabalho de pesquisadores que, como eu, estão estudando como os métodos de explicação existentes podem ou não ajudar nessa questão, oferecendo um acesso mais fácil para esses métodos juntando eles em um lugar e utilizando uma mesma sintaxe para gerar explicações. Além disso, há a tentativa de dar um passo na direção de melhorar a interpretabilidade dessas explicações, ao gerar explicações verbais para elas de forma parametrizada para que seja personalizada para os dados do usuário. Assim, além de facilitar o trabalho de pesquisadores, a biblioteca também pode ajudar usuários que queiram entender os resultados de modelos treinados para domínios específicos e têm dificuldade de interpretar as visualizações existentes.

3. Especificação de requisitos do software

Considerando o contexto de Explainable AI em que a solução proposta está inserida e os seus objetivos, foram definidos os requisitos necessários para o seu desenvolvimento. Nesta seção então serão apresentados os requisitos funcionais e não funcionais do software.

3.1. Requisitos funcionais

[RF1] O sistema deve permitir que o usuário escolha um dos métodos conhecidos de explicabilidade de modelos de machine learning, fornecendo um modelo treinado por ele e suas features

[RF2] O sistema deve ser capaz de exibir visualizações das explicações geradas pelo método selecionado pelo usuário.

[RF3] O sistema deve ser capaz de fornecer uma explicação textual do que está sendo exibido na visualização para auxiliar o usuário a entendê-la.

[RF4] O sistema deve ter um atributo que explique ao usuário como cada tipo de visualização disponível funciona e quais são os cenários em que cada uma é mais adequada.

[RF5] O sistema deve possibilitar que o usuário gere visualizações e explicações de diferentes métodos de explicação utilizando uma mesma a simples sintaxe.

3.2. Requisitos não-funcionais

[RNF1] O sistema deve ser disponibilizado como uma biblioteca python para poder ser utilizada preferencialmente em notebooks.

[RNF2] O sistema deve ser escalável, de forma que novas classes de métodos de explicação ou novas classes de visualização possam ser adicionadas facilmente.

[RNF3] O sistema deve ser implementado de forma que possibilite fácil manutenção e atualização, seguindo as melhores práticas da Engenharia de Software.

4. Descrição da arquitetura

ExpViz em composta pelos pacotes *exp_explainers* e *visualizations*. Cada um deles contém uma classe abstrata base e classes que implementam os seus métodos. No pacote *exp_explainers* a classe abstrata base se chama *Explainer*, e é dela que as classes de cada método de explicação vão herdar para ter suas implementações específicas. Cada classe de método de explicação terá o seu conjunto de visualizações. As visualizações vêm do pacote *visualizations*, que contém a classe abstrata base *Plot*. Cada classe de visualização tem uma função que irá exibir tanto o gráfico da explicação quanto a explicação textual parametrizada para ajudar na interpretação da visualização, considerando os dados fornecidos pelo usuário.

Essa estrutura faz com que a solução seja escalável, uma vez que é possível adicionar novos métodos de explicação e novos tipos de visualização de forma fácil, sem que os que já existem tenham que ser alterados.

4.1. Diagrama de classes

O diagrama de classes da arquitetura descrita anteriormente está apresentado na Figura 1.

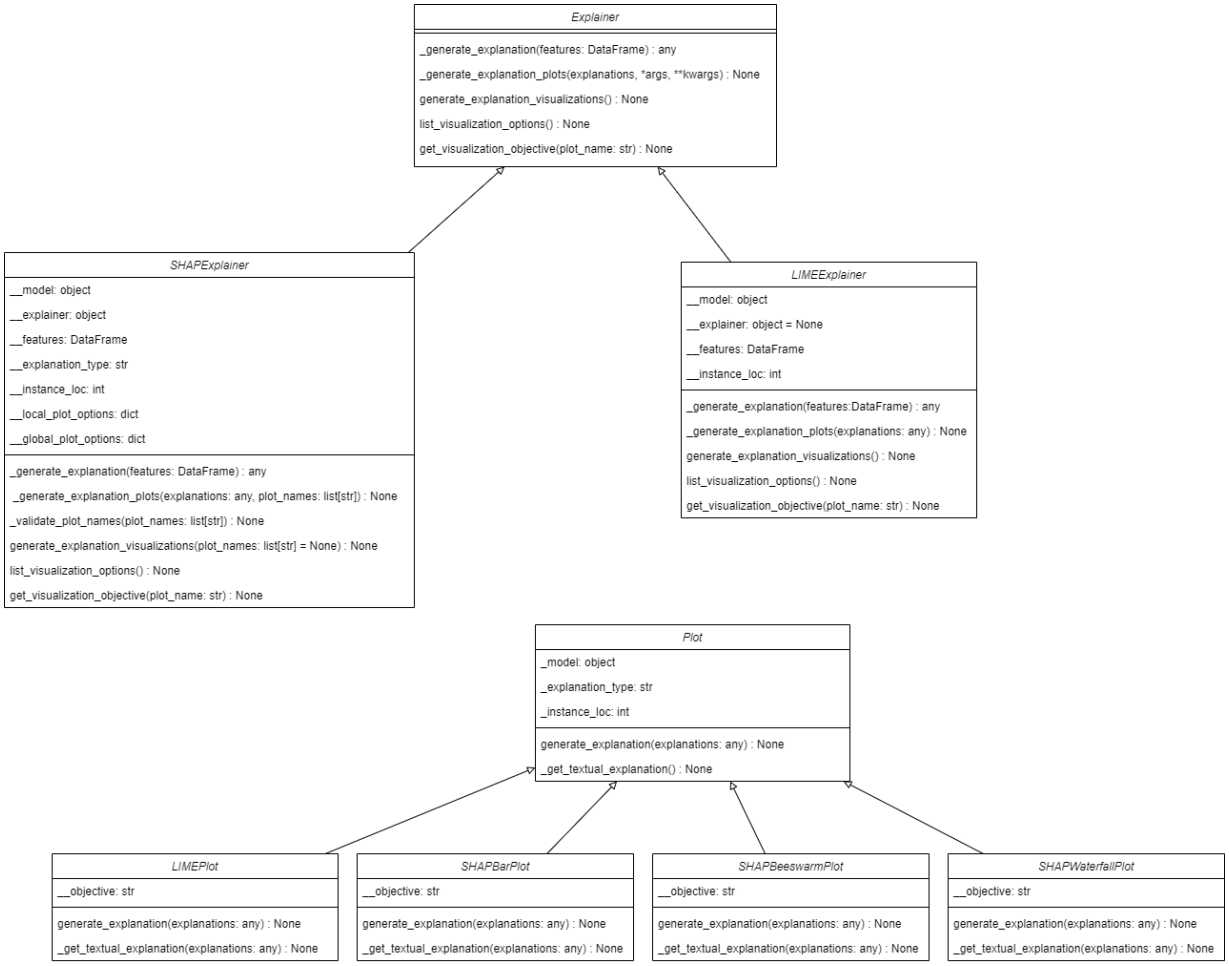

Figura 1. Diagrama de classes

5. Cenários de uso do software

Nesta seção serão apresentados alguns cenários de uso do software, tanto casos em que o usuário tem sucesso, descritos na seção 5.1, quanto cenários em que ele encontra problemas, descritos na seção 5.2.

5.1. Cenários de uso com sucesso

5.1.1. Cenário 1

Um cientista de dados pode utilizar a biblioteca ExpViz para gerar explicações para um modelo de machine learning que ele esteja utilizando ou desenvolvendo com o intuito de entender como o seu modelo se comporta ou como cada atributo do modelo impacta predições específicas. Ele pode querer testar explicações geradas por diferentes tipos de métodos de explicação para entender qual o atende melhor. Para isso ele deve instalar a biblioteca em seu ambiente de trabalho seguindo as orientações na documentação, utilizar um notebook importando a biblioteca nele, para que possa ver as visualizações e explicações com maior facilidade, e deve ler a documentação para entender como a biblioteca deve ser

utilizada, quais métodos de explicação estão disponíveis nela e quais visualizações estão disponíveis e como cada uma funciona.

5.1.2. Cenário 2

Um pesquisador que esteja estudando a interpretabilidade de explicações geradas por métodos de explicação de modelos de machine learning, como SHAP e LIME, pode utilizar a biblioteca ExpViz para gerar explicações e visualizações de diferentes métodos existentes na literatura com facilidade, já que estão concentrados em uma biblioteca. Para isso ele deve instalar a biblioteca em seu ambiente de trabalho seguindo as orientações na documentação, utilizar um notebook importando a biblioteca nele, para que possa ver as visualizações e explicações com maior facilidade, e deve ler a documentação para entender como a biblioteca deve ser utilizada e quais métodos de explicação estão disponíveis nela.

5.2. Cenários de uso com problemas

5.2.1. Cenário 1

Um cientista de dados que queira entender o comportamento de um modelo de machine learning que está utilizando, mas que não esteja familiarizado com o seu dataset e os atributos utilizados para treinar o modelo, pode ter dificuldade para entender as saídas das funções da biblioteca ou pelo menos para fazer uso adequado dessa informação que ele recebe. O ideal é que o usuário estude o seu dataset e entenda os atributos e as relações entre eles, para que a informação entregue pela biblioteca seja utilizada de forma otimizada.

5.2.2. Cenário 2

Um usuário que tente usar a biblioteca ExpViz sem utilizar um IPython notebook ou uma interface que possibilite a exibição de gráficos e imagens, como por exemplo um terminal, pode ter dificuldades, já que uma parte importante das saídas da biblioteca são as visualizações. Além disso, em termos de comparação de métodos de explicação e de suas visualizações, o notebook torna a exibição de múltiplas visualizações para observação em paralelo muito mais fácil, e não utilizá-lo pode tornar a análise mais difícil e demorada. O ideal é que o usuário use um IPython notebook para as análises utilizando a biblioteca, para que tenha uma melhor visualização das saídas.