

\_\_\_\_\_

PÓS-GRADUAÇÃO

**XP Educação**

**Relatório do Projeto Aplicado**

Avaliação da Usabilidade e Eficácia da Yggdrasil Decision Forests (YDF) em Previsão de Dados para Contextos de empresas Data-Driven

Matheus Poletto

Orientador(a): Davidson Oliveira

11/11/2024

****

**MATHEUS POLETTO**

**XP EDUCAÇÃO**RELATÓRIO DO PROJETO APLICADO

Avaliação da Usabilidade e Eficácia da Yggdrasil Decision Forests (YDF) em Previsão de Dados para Contextos Data-Driven

Relatório de Projeto Aplicado desenvolvido para fins de conclusão do curso de Pós-Graduação em Cientista de Dados.  
  
Orientador (a): Davidson Oliveira

**Florianópolis/SC  
11/11/2024**

**Sumário**

[1. CANVAS do Projeto Aplicado 4](#_Toc184230746)

[1.1.1 Análise de Contexto 4](#_Toc184230747)

[1.1.2 Personas 5](#_Toc184230748)

[1.1.3 Benefícios e Justificativas 6](#_Toc184230749)

[1.1.4 Hipóteses 7](#_Toc184230750)

[1.2 Solução 9](#_Toc184230751)

[1.2.1 Objetivo SMART 9](#_Toc184230752)

[1.2.2 Premissas e Restrições 9](#_Toc184230753)

[1.2.3 Backlog de Produto 10](#_Toc184230754)

[2. Área de Experimentação 11](#_Toc184230755)

[2.1 Sprint 1 11](#_Toc184230756)

[2.1.1 Solução 11](#_Toc184230757)

[● Evidência do planejamento: 11](#_Toc184230758)

[● Evidência da execução de cada requisito: 12](#_Toc184230759)

[● Evidência dos resultados: 13](#_Toc184230760)

[2.1.2 Lições Aprendidas 14](#_Toc184230761)

[2.2 Sprint 2 14](#_Toc184230762)

[2.2.1 Solução 14](#_Toc184230763)

[● Evidência do planejamento: 14](#_Toc184230764)

[● Evidência da execução de cada requisito: 15](#_Toc184230765)

[● Evidência dos resultados: 15](#_Toc184230766)

[2.2.2 Lições Aprendidas 15](#_Toc184230767)

[2.3 Sprint 3 16](#_Toc184230768)

[2.3.1 Solução 16](#_Toc184230769)

[● Evidência do planejamento: 16](#_Toc184230770)

[● Evidência da execução de cada requisito: 16](#_Toc184230771)

[● Evidência dos resultados: 16](#_Toc184230772)

[2.3.2 Lições Aprendidas 16](#_Toc184230773)

[3. Considerações Finais 17](#_Toc184230774)

[3.1 Resultados 17](#_Toc184230775)

[3.2 Contribuições 17](#_Toc184230776)

[3.3 Próximos passos 17](#_Toc184230777)

## 1. CANVAS do Projeto Aplicado

Figura 1, que representa todas as etapas propostas do Projeto Aplicado.



Figura 1 – Canvas do Projeto Aplicado.

### 1.1.1 Análise de Contexto

Devido ao contexto atual do big-data, cada vez se torna mais relevante a utilização dos dados dentro de uma organização. Quem tem informação está um passo à frente e quem sabe o que fazer com essa informação, está ainda mais a frente. No dia-a-dia do cientista de dados, uma das tarefas mais relevantes é a previsão de resultados com base em uma análise de um banco de dados existente para aplicação em uma situação futura.

O objetivo principal desse trabalho, vista estudar as aplicações específicas da Yggdrasil Decision Forests (YDF), uma poderosa biblioteca para python desenvolvida pela Google, que tem como função a ingestão de dados e consequente previsão.

A Figura 2 apresenta o modelo de matriz CSD (certezas, suposições e dúvidas).

A Figura 3 apresenta a Análise do Contexto do Problema – POEMS.



Figura 2 – Matriz CSD.



Figura 3 – POEMS.

### 1.1.2 Personas

A persona deste trabalho será um cientista de dados em ascensão de carreira. Ele se chama José das Dores, 26 anos, é um cientista de dados em início de carreira buscando um campo de especialização. Possui experiência corporativa na área de engenharia mecânica, onde sabe lidar com e resolver problemas. Atualmente procura uma ferramenta para sua principal atuação no campo de ciência de dados. Hoje utiliza principalmente as ferramentas python e suas principais bibliotecas voltadas a ciência de dados e SQL.



Figura 4 – Mapa de Empatia.

### 1.1.3 Benefícios e Justificativas

A principal tarefa de um cientista de dados no dia a dia é realizar previsões. Há muitas ferramentas disponíveis para isso, e cada uma delas é mais adequada a um contexto específico. Neste projeto, que utiliza exclusivamente o Yggdrasil Decision Forests (YDF) e seu modelo de Gradient Boosted Decision Trees (GBDT), a escolha se justifica pelos diversos benefícios que ele proporciona.

Um dos principais diferenciais do YDF é a capacidade de lidar com dados sem necessidade de pré-tratamento rigoroso. Em um cenário de big data, onde a geração de dados é exponencial, o tratamento de dados costuma ser uma etapa fundamental e muitas vezes demorada. Ao eliminar essa necessidade, o YDF permite acelerar o desenvolvimento do projeto e reduzir possíveis erros, uma vez que essa variável é minimizada no processo. Isso torna o YDF uma solução poderosa para aplicações em larga escala, otimizando o tempo e os recursos envolvidos.



Tabela 1 – *Blueprint*.

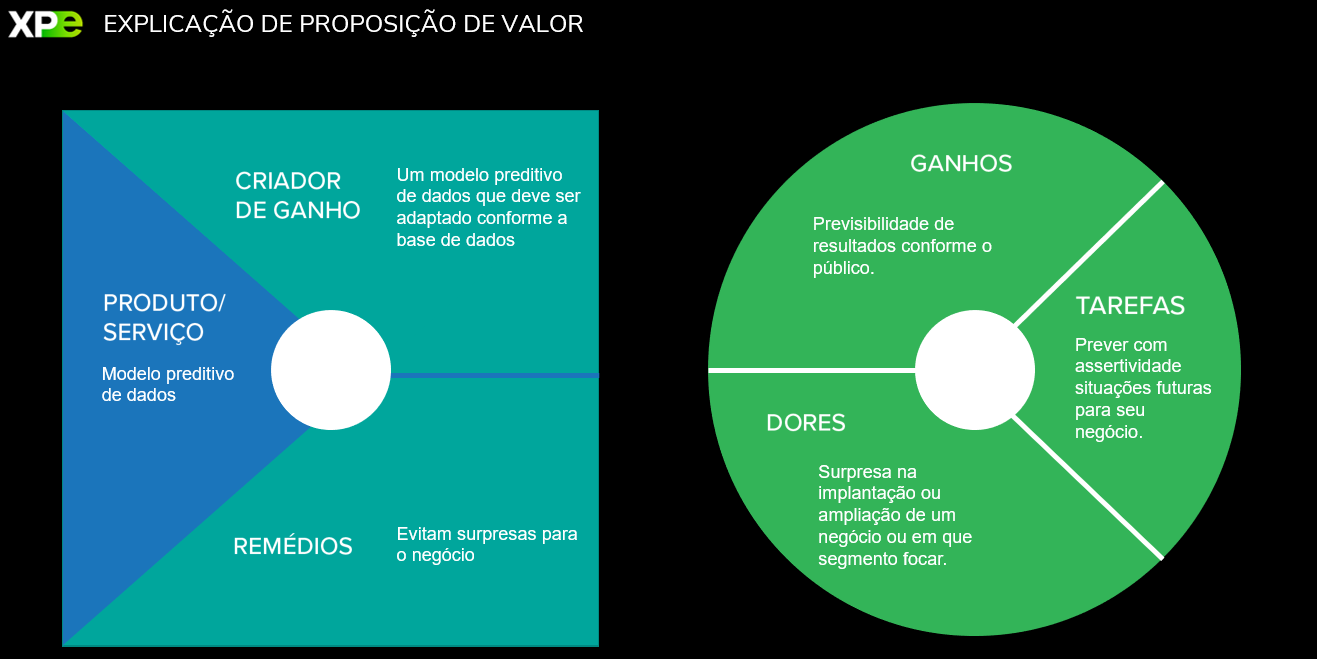


Figura 5 – Proposição de Valor.

### 1.1.4 Hipóteses

Dentre todas as soluções que um cientista de dados consegue aplicar, a predição certamente é um de seus principais trunfos. Dado o grande número de ferramentas e metodologias que é possível aplicar, a pessoa fica sem saber em qual se aprimorar e por que. Afunilando um pouco esse leque de ferramentas, para as ferramentas de modelos de aprendizado supervisionado, temos as florestas de decisão.

Existe uma dificuldade inicial do cientista de dados em como aplicar um modelo de floresta de decisão. O YDF, antigo TFDF (tensor flow decision forests) é uma solução nova e aprimorada, que compete de frente com outras bibliotecas como Scikit-Learn ou XGBoost. A tabela 2 traz um pouco do problema e da insegurança ao praticar e escolher uma ferramenta para se aperfeiçoar e dominar.



Tabela 2 – Matriz de Observações e Hipóteses.

Abaixo, na matriz de priorização de ideias, temos uma noção de quais as principais ideias e as mais impactantes.



Tabela 3 – Matriz BASICO.

## 1.2 Solução

### 1.2.1 Objetivo SMART

Este estudo tem por objetivo trazer informações de elaboração e como fazer o aprimoramento de um modelo de floresta de decisão.

O que buscamos aqui é estudar o modelo, validar sua assertividade dentro do que é possível prever segundo a própria biblioteca YDF e depois verificar qual foi seu acerto, enviando o arquivo resultante para a plataforma Kaggle verificar o nível de acerto. Lembrando que o que a Kaggle faz é praticamente impossível no mundo real, pois eles têm os dados corretos e no dia-a-dia temos que confiar unicamente em nossas previsões e na assertividade do modelo.

### 1.2.2 Premissas e Restrições

Apesar de o mercado já dispor de várias ferramentas para o machine learning, foi decidido usar neste projeto apenas o YDF e GBDT para estudo específico das ferramentas, de modo a conseguir chegar numa conclusão de sua utilização.

Devido ao curto tempo de desenvolvimento, não será possível explorar outros estudos de caso com outras ferramentas de modo a fazer um comparativo entre soluções.

A ideia principal é ter um resultado de assertividade inicial e conseguir melhorá-lo através do treinamento do modelo e inserção de novas variáveis. O não atendimento dessa condição resulta em possível busca por uma ferramenta melhor.

Ainda, para a utilização dessa ferramenta, é necessário um banco de dados relacional e com dados confiáveis, pois serão os inputs do modelo.



Tabela 4 – Matriz de Riscos.

### 1.2.3 Backlog de Produto

Nessa etapa, será descrito como será o passo a passo da solução do problema que o projeto propõe para alcançar o objetivo final do trabalho.

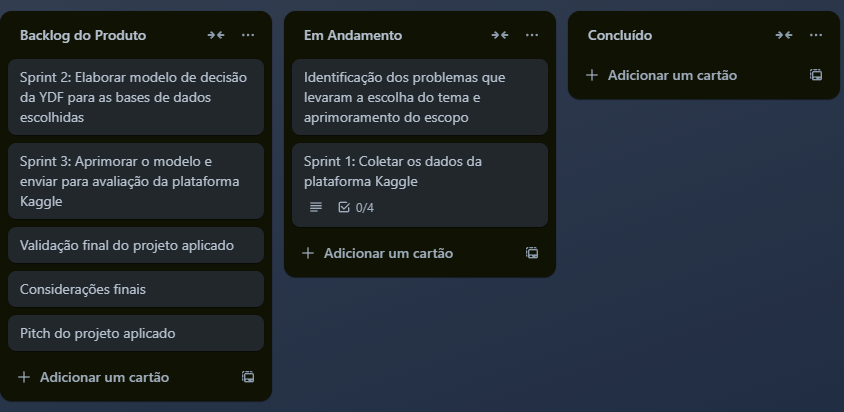


Figura 6 – Backlog de Produto no Trello.

# 2. Área de Experimentação

Esta seção tem a finalidade de mostrar a execução das atividades propostas no Backlog do Produto. Serão mostrados em detalhes todas as etapas, dividas em 03 (três) Sprints. Cada um deles representa um conjunto de tarefas a serem executadas, de forma que, juntos, representam a solução do problema proposto. A ferramenta escolhida para isso foi o Trello, onde através dele, todas as tarefas foram divididas. Cada cartão representa uma etapa ou Sprint. Dentro de cada um, estão as anotações e checklist das atividades.

## 2.1 Sprint 1

### 2.1.1 Solução

#### Evidência do planejamento:

Todo planejamento do Sprint 1 foi catalogado dentro do Trello. Nele foram criados vários Checklists para organizar cada etapa. Isso permite uma maior organização das etapas, de forma a visualizar de forma detalhada tudo o que tem que ser feito, além de registrar todas as tarefas. A Figura 7 mostra em detalhes o resultado deste planejamento.

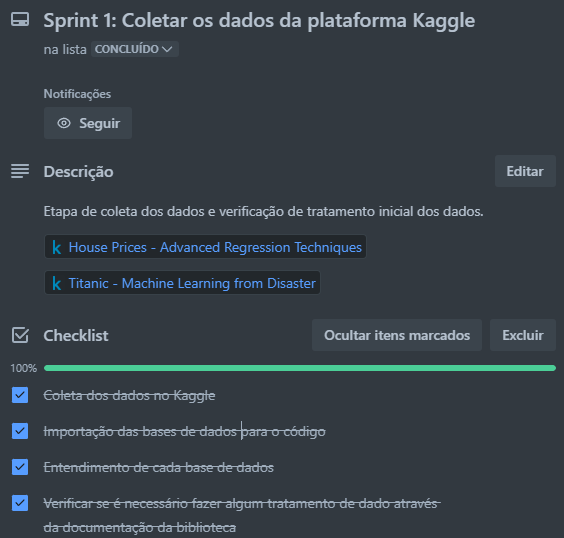


Figura 7 – Sprint 1 no Trello.

As bases de dados utilizadas foram obtidas do site Kaggle, através dos links <https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques> e <https://www.kaggle.com/competitions/titanic>.

#### Evidência da execução de cada requisito:

A Sprint 1 se propunha a resolver 4 principais problemas, que serão abordados abaixo.

Foram importadas e estudadas as composições de cada base de dados para o entendimento das variáveis para melhorar o treinamento dos modelos.

Coleta de dados no Kaggle: Os dados foram baixados e armazenados em disco local.

Importação das bases de dados para o código: Conforme Figuras 8 e 9, a importação das bases de dados foram realizadas.

Entendimento de cada base de dados: foram analisadas as variáveis e alguns dados das bases de dados, porém sem profundidade, uma vez que analisaremos a fundo na etapa de aprimoramento do modelo.

Verificar se é necessário fazer tratamento de dados: conforme verificado na página oficial da biblioteca, não é necessário (<https://ydf.readthedocs.io/en/latest/>).

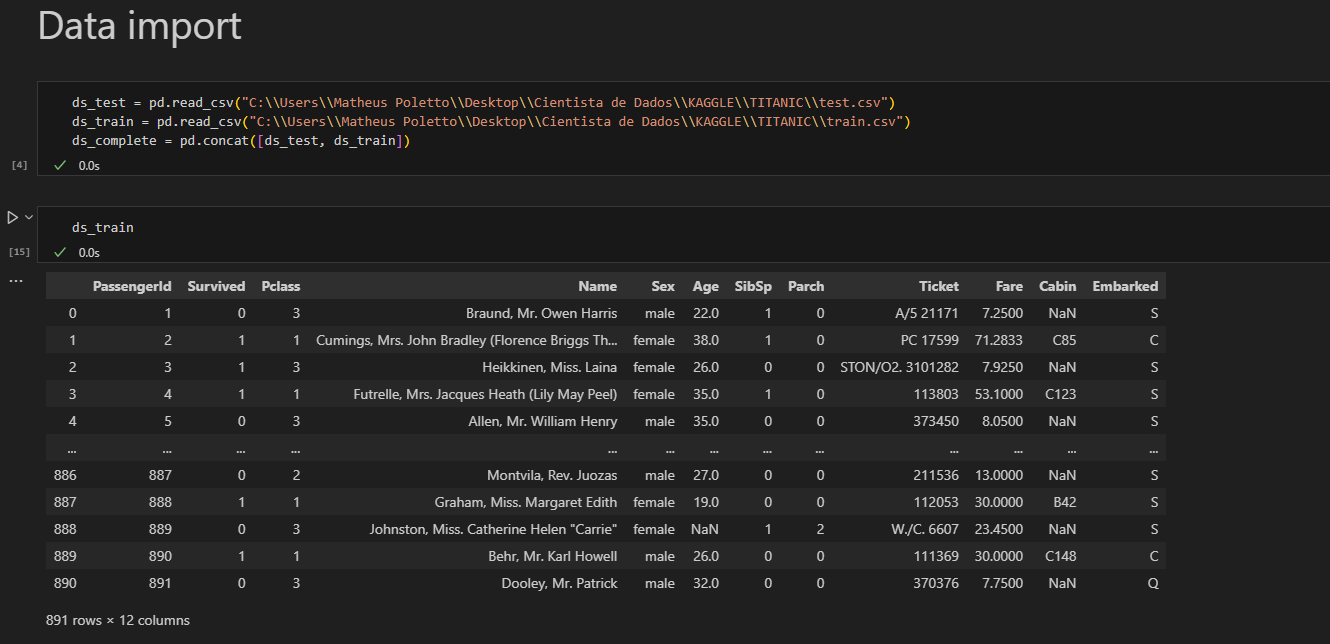


Figura 8 – Importação das tabelas TITANIC para o notebook (VSCode).

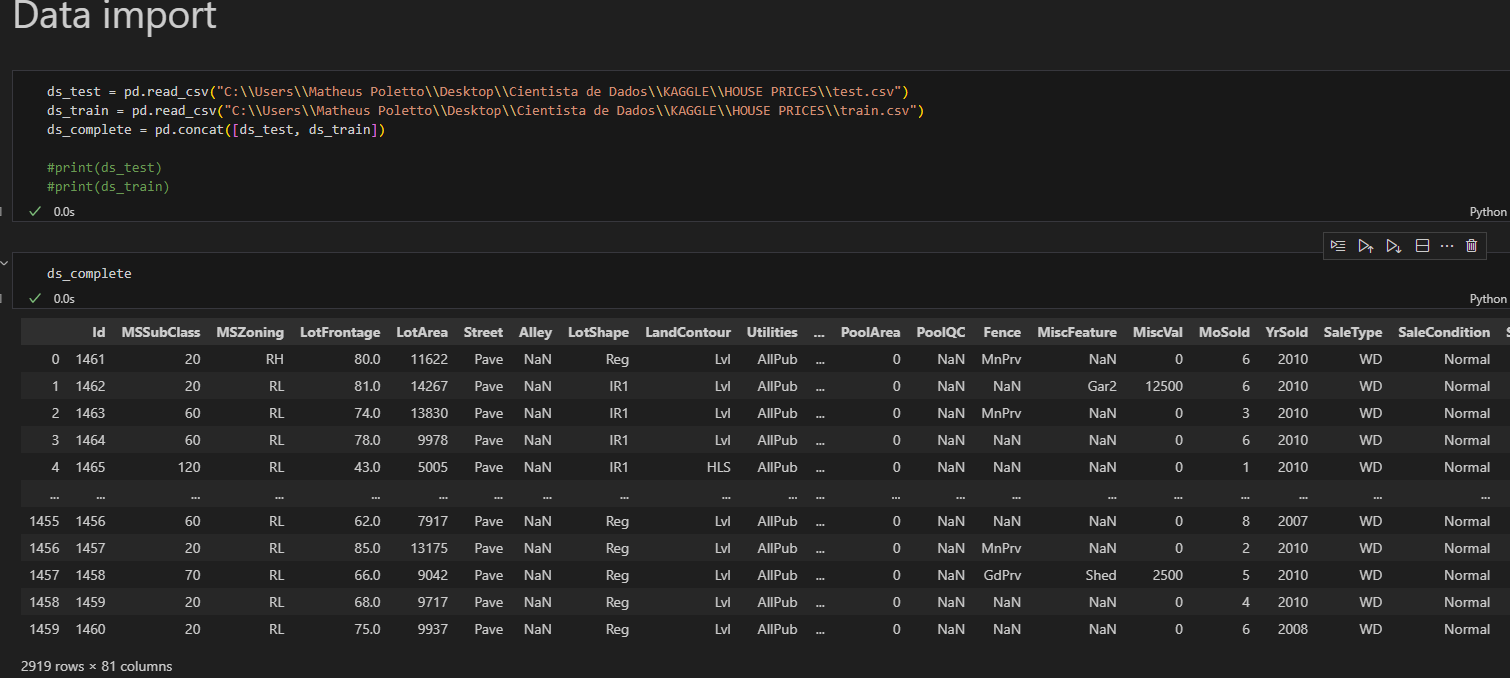


Figura 9 – Importação das tabelas HOUSE PRICES para o notebook (VSCode).

#### Evidência dos resultados:

Conforme citado anteriormente neste trabalho, a não necessidade de qualquer tratamento de dados adianta muito o trabalho do cientista de dados.

A título de exemplo, a Figura 10 representa todas as etapas de como seria um tratamento de dados:

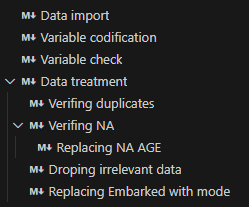


Figura 10 – Estrutura básica de um código de tratamento de dados.

Enquanto a YDF, como faz o tratamento por conta, fica parecido com isso:

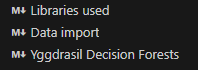


Figura 11 – Estrutura básica de um código de YDF GBDT.

### 2.1.2 Lições Aprendidas

Como o próprio material da ferramenta YDF GBDT cita, é desnecessário e até não recomendado fazer qualquer tipo de tratamento de dados, pois a escolha que o ser humano faz, pode não ser a melhor. Por exemplo, enquanto o cientista de dados faria a substituição de um valor faltante pela média da variável, a YDF GBDT encontra por si só qual é a melhor decisão.

Em etapas futuras, faremos a análise da relevância de tratamento de dados quando utilizamos a YDF com GBDT.

## 2.2 Sprint 2

### 2.2.1 Solução

#### Evidência do planejamento:

Todo planejamento do Sprint 2 foi catalogado dentro do Trello, assim como na Sprint 1. A Figura 12 mostra em detalhes o resultado deste planejamento.

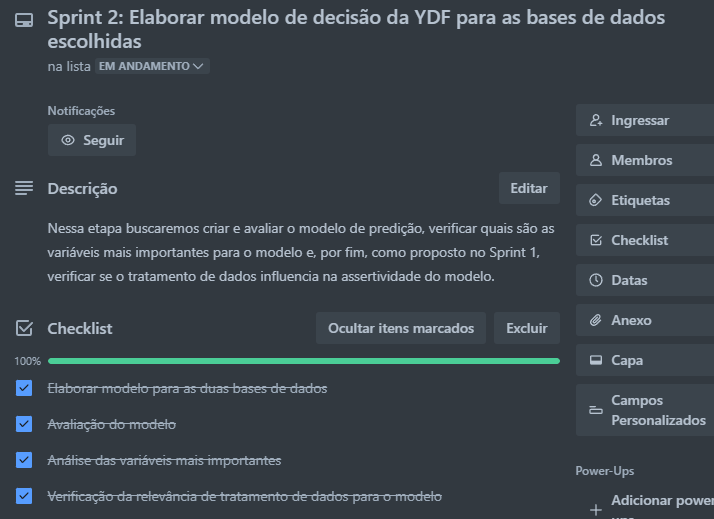


Figura 12 – Sprint 2 no Trello.

#### Evidência da execução de cada requisito:

A Sprint 2 se propunha a resolver 4 problemas, que serão abordados abaixo.

Elaborou-se o modelo para cada base de dados, avaliou-se o modelo aplicando-o em uma planilha de teste, evidenciou-se as variáveis mais importantes e foi verificado a relevância do tratamento de dados para a assertividade do modelo.

Elaborar modelo para as duas bases de dados: foram elaborados os modelos individualmente por base de dados.

Avaliação do modelo: após a elaboração, foi testado via “evaluate” assertividade do modelo, com base nos dados de treino.

Análise das variáveis mais importantes: foi verificado e evidenciado as variáveis mais importantes para o modelo de cada base de dados.

Verificação da relevância de tratamento de dados para o modelo: foi testado em cada base de dados se realmente o tratamento de dados não é benéfico ao modelo.

#### Evidência dos resultados:

A elaboração dos modelos é relativamente simples. Precisamos colocar qual a coluna que vamos treinar:

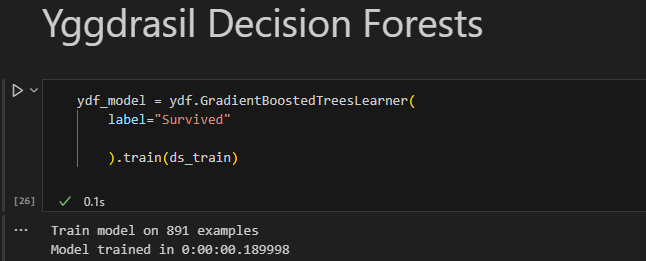


Figura 13 – Criando o modelo para o TITANIC no VSCode.

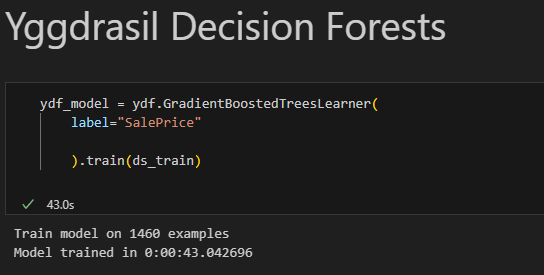


Figura 14 – Criando o modelo para o HOUSE PRICES no VSCode.

A avaliação do modelo é o acerto que o modelo consegue atingir com os dados fornecidos. É como se ele fosse predizer a própria base de dados utilizada para treinar:

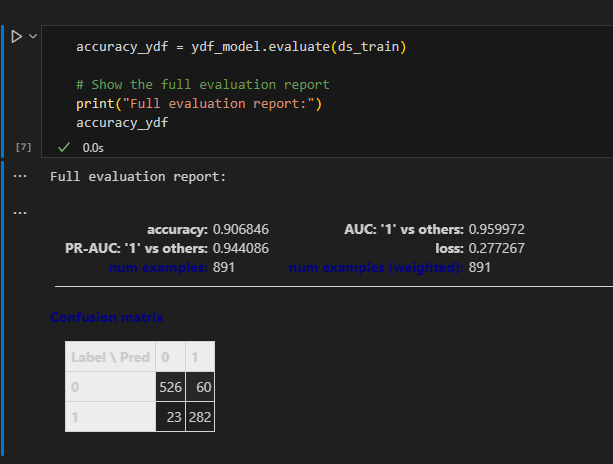


Figura 15 – Verificando a assertividade do modelo treinado, TITANIC.

As variáveis mais importantes dependem do modelo treinado e, obviamente, das variáveis disponíveis. Abaixo segue a relevância das variáveis de cada modelo:

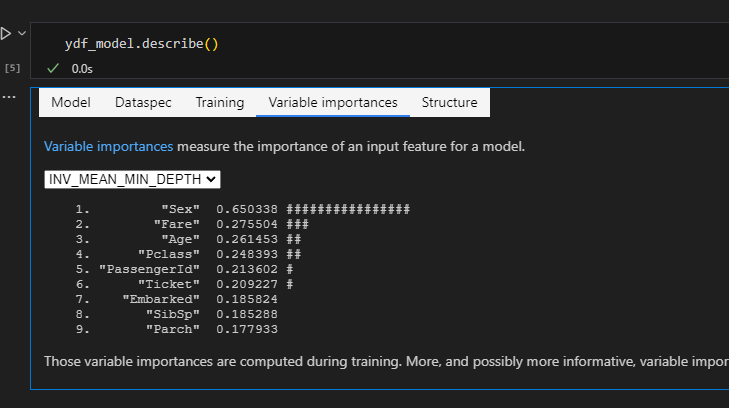


Figura 16 – Verificando a relevância de cada variável para o modelo do TITANIC.

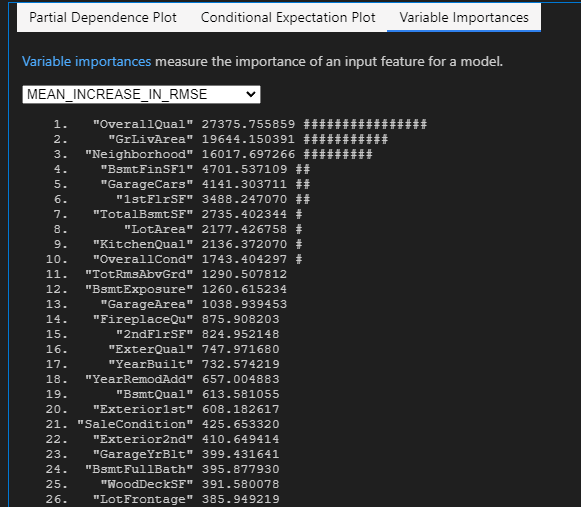


Figura 16 – Verificando a relevância de cada variável para o modelo do HOUSE PRICES.

A última etapa proposta para a Sprint 2 é verificar a relevância do tratamento de dados para o modelo. Portanto, foi incluída a etapa de tratamento de dados na base de dados TITANIT, conforme mostrado abaixo:

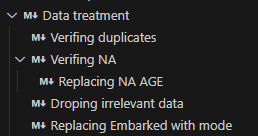


Figura 17 – Tratamento de dados.

Foram feitos os seguintes tratamentos:

Verificação de duplicatas: não haviam;

Verificação de dados faltantes: haviam dados faltantes na coluna Age, Cabin e Embarked.

Os dados da coluna Age foram substituídos conforme código abaixo:

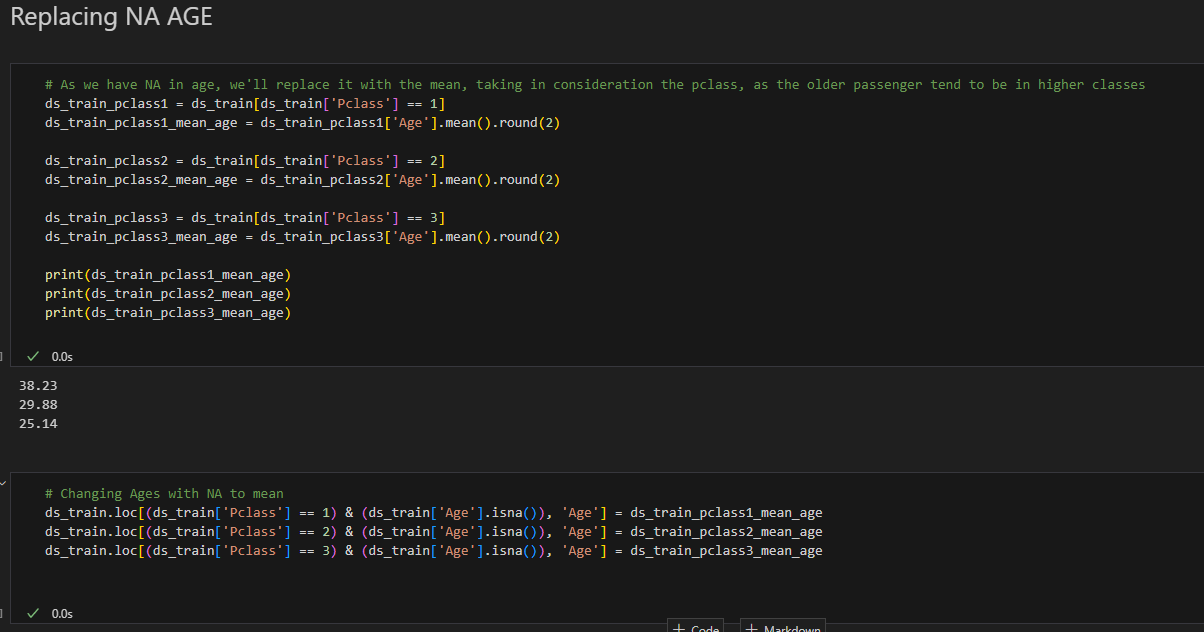


Figura 18 – Replacing NA AGE.

Calculou-se a idade média para cada Pclass, considerando que esse era o dado mais relevante na matriz de correlação, por representar a riqueza de cada passageiro. Isso ocorre porque, quanto melhor a classe, maior tende a ser o poder aquisitivo do passageiro. Após o cálculo, a idade média correspondente foi atribuída a cada dado faltante com base na Pclass.

A coluna Cabin foi removida pois não exista nenhum padrão visível.

Os dados faltantes da coluna Embarked foram substituídos pela moda:

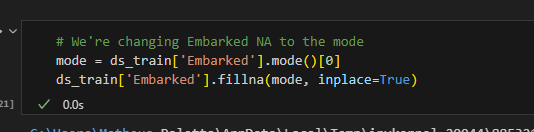


Figura 19 – Removendo nulos da coluna Embarked.

Uma vez feito o tratamento, treinamos novamente o modelo e analisamos a efetividade. Pode-se notar que a acurácia foi reduzida de 0,9068 para 0,8900, conforme a figura 20 abaixo. Pode parecer pouco, mas ao longo do treinamento do modelo, faz grande diferença.

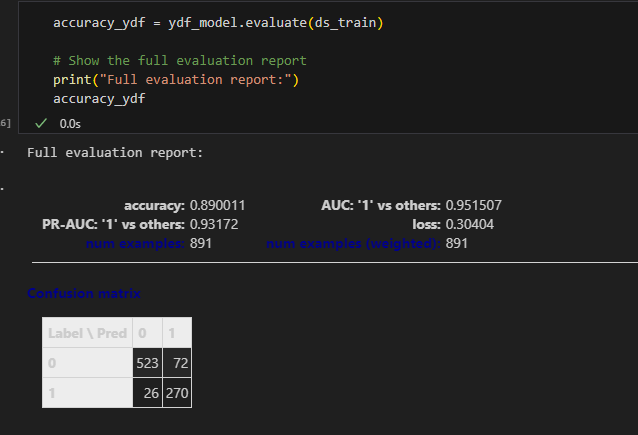


Figura 20 – Acurácia do modelo com tratamento de dados.

### 2.2.2 Lições Aprendidas

Nessa Sprint, pudemos evidenciar cada etapa que foi proposta.

Pudemos criar o modelo, avaliar o modelo, verificar as variáveis mais importantes e evidenciar que de fato o tratamento de dados é irrelevante e até prejudicial para o treinamento do modelo de GBDT.

Na próxima Sprint, buscaremos melhorar o modelo, incluindo novos parâmetros da ferramenta.

## 2.3 Sprint 3

### 2.3.1 Solução

#### Evidência do planejamento:

Todo planejamento do Sprint 3 foi catalogado dentro do Trello, assim como na Sprint 1 e 2. A Figura 20 mostra em detalhes o resultado deste planejamento.

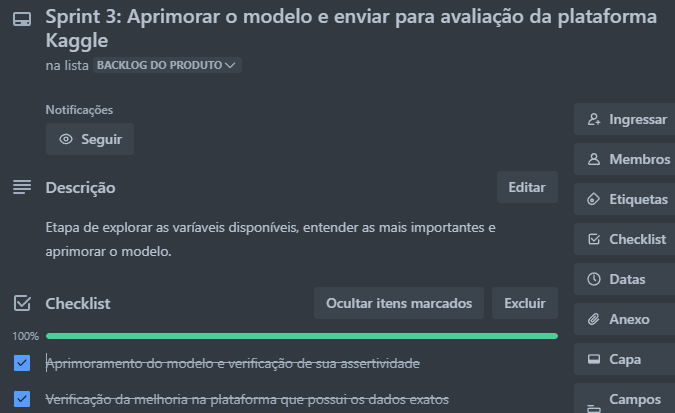


Figura 21 – Sprint 3 no Trello.

#### Evidência da execução de cada requisito:

A Sprint 3 se propunha a resolver 3 problemas, que serão abordados abaixo.

O modelo foi aprimorado utilizando as técnicas e ferramentas disponíveis na GBDT da YDF. É importante destacar que o aprimoramento e os testes são realizados na própria base de treinamento. Assim, mesmo que hipoteticamente alcancemos 100% de acurácia, é pouco provável que esse resultado se mantenha ao submetermos o modelo ao Kaggle, já que a base de dados utilizada para avaliação é diferente. Essa é, de longe, a etapa mais trabalhosa do projeto, pois envolve o teste de várias combinações para encontrar a melhor acurácia possível dentro de cada base de dados.

O primeiro e o último resultado de cada base de dados foram submetidos ao Kaggle para verificação se houve ou não melhoria, comparativamente ao que conseguimos no próprio modelo.

#### Evidência dos resultados:

Foi realizado o aprimoramento do modelo através do *model tuning* e seus *hyper parameters*:

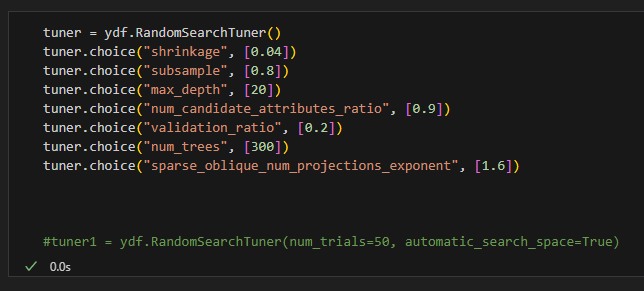


Figura 22 – Criação do tuner.

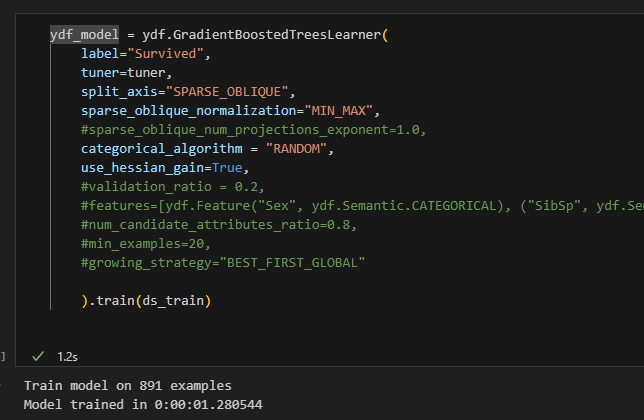


Figura 23 – Inclusão do tuner no modelo e outras variáveis.

Através do aprimoramento do modelo, conseguimos resultados relevantes para a acurácia do modelo:

Base de dados Titanic:

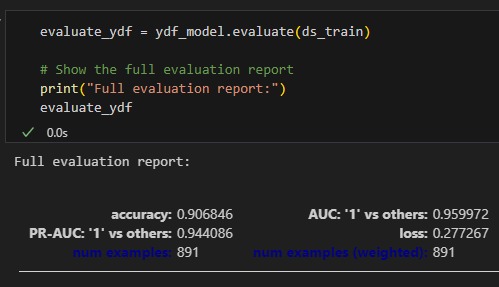


Figura 24 – Assertividade inicial do modelo.

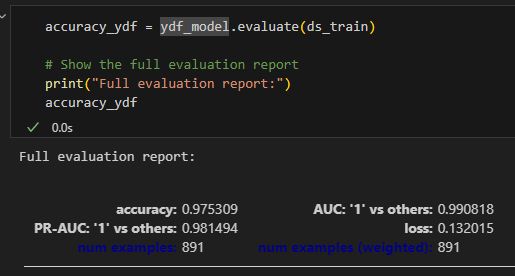


Figura 25 – Assertividade final do modelo.

A base de dados Titanic teve um aumento significativo de acurácia de 0.906846 para 0.975309.

Ao submetermos para o Kaggle, conseguimos os seguintes resultados:

Score do modelo original: 0.78708

Score do modelo final: 0.77990

Quanto mais próximo de 1, melhor.

Base de dados House Prices:

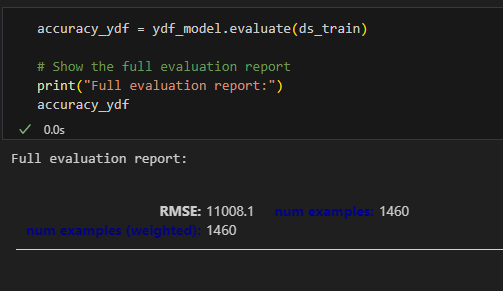


Figura 26 – Assertividade inicial do modelo.

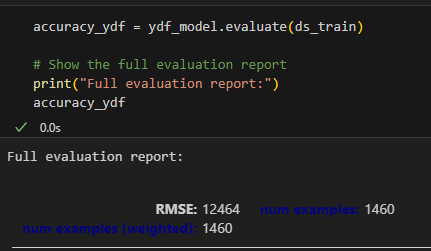


Figura 27 – Assertividade final do modelo.

Ao submetermos para o Kaggle, conseguimos os seguintes resultados:

Score do modelo original: 0.13381

Score do modelo final: 0.13244

Quanto maior, melhor.

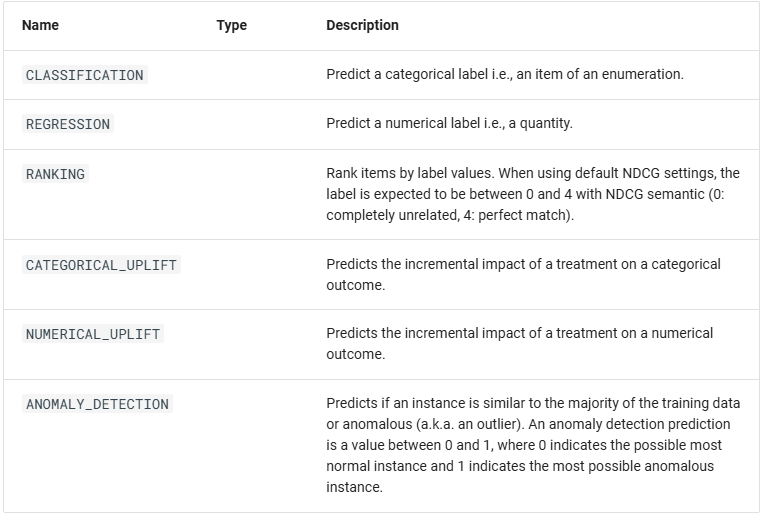
### 2.3.2 Lições Aprendidas

Pudemos trabalhar a melhoria de um modelo e descobrir suas mais variadas condições de *model tuning* para entender um pouco mais sobre a predição.

Não era esperado que o modelo melhorado desempenhasse pior ao fazer a submissão no Kaggle. Porém, conforme dito anteriormente, a acurácia alcançada pelo modelo é avaliada na própria base de dados de treino. Portanto, quando submetemos para avaliação, onde é outra base de dados, a assertividade pode não mudar.

Apesar de o GBDT não precisar de tratamento de dados, ainda precisamos indicar o tipo de tarefa que ele deve executar.

Naturalmente ele executa todas as tarefas como sendo “CLASSIFICATION”, que era o caso da base de dados TITANIC. Ao lidarmos com a base de dados HOUSE PRICES, precisamos solicitar que o GBDT trabalhe com a tarefa “REGRESSION”.



# 3. Considerações Finais

## 3.1 Resultados

A escolha da GBDT da YDF partiu de um anseio em verificar como desempenharia uma biblioteca relativamente nova e também aprender uma nova tecnologia.

Como grande ponto positivo, podemos destacar a não necessidade de tratamento de dados da GBDT. O tratamento de dados é uma etapa extremamente massiva e certamente é a que mais consome tempo de um cientista de dados ou de quem está buscado treinar algum modelo com machine learning.

Como grande ponto negativo, por se tratar de uma biblioteca nova, ainda é pouco falada e existem poucos conteúdos na internet. Isso dificulta a solução de algum problema específico.

Sobre o treinamento do modelo, conseguimos melhorar significativamente através do model tuning, embora não reflita na submissão ao Kaggle, é o principal parâmetro para análise da assertividade do modelo.

## 3.2 Contribuições

O Projeto Aplicado teve sua contribuição ao que foi proposto, sendo a elaboração e aprimoramento de um modelo de machine learning através da GBDT da YDF. Conseguimos observar desde a não necessidade de tratamento de dados como a análise da importância de cada variável para o modelo treinado.

## 3.3 Próximos passos

Para os próximos passos, existem caminhos a serem seguidos, entre eles:

* Escalabilidade: para casos de dados massivos, provavelmente é necessário trabalhar com computação distribuída;
* Ambiente de produção: para casos de negócios reais, é necessário ter uma aplicação pronta para uso, ao qual esse projeto não tinha como objetivo;
* Ferramentas similares: avaliação de outras bibliotecas/ferramentas com o mesmo objetivo e verificar qual desempenha melhor;
* Integração com tensor flow para criação de deep learning.