# Relatório de Modelo de Série Temporal para Previsão de Vendas Semanais

Gerado por: [ueliton\_viana@outlook.com](mailto:ueliton_viana@outlook.com) e H2o Cloud

Gerado em: 27 de Nov de 2022

Modelo de Série Temporal para Previsão de Vendas Semanais1

1. Experimento Geral2
2. [Visão geral do Dataset](#_heading=h.bj7ecn214d3h)3
3. [Methodologia](#_heading=h.jz4leguz5ipf)5
4. [Amostragem de dados 1](#_heading=h.tyjcwt)0
5. [Estratégia de Validação e Ajuste](#_heading=h.ar7084j8kbjw)10
6. [Evolução do RMSE](#_heading=h.3p814xomrmow)13
7. [Transformações de Recursos Finais](#_heading=h.oehfh73py4vx)15
8. [Modelo Final](#_heading=h.lnxbz9) 18
9. [Implantação 2](#_heading=h.1ksv4uv)0
10. Shapley Values [2](#_heading=h.44sinio)1
11. [Apêndice 23](#_heading=h.z337ya)

## 

**1) Experimento Geral**

Modelo de LightGBM para prever Vendas\_Semanais com 12 recursos originais do conjunto de dados de entrada walmart\_train.csv. Este experimento de regressão foi concluído em 30 minutos e 23 segundos (0:30:23), usando 3 dos 12 recursos originais e 48 dos 48 recursos projetados.

### Performance

| **Dataset** | **RMSE** |
| --- | --- |
| Validação Interna | 140.831,95 |
| Dados de Teste | 81.820,71 |

### Configurações de A.T.I.

| **Nome** | **Descrição** | **Valor Selecionado** | **Valores Possíveis** |
| --- | --- | --- | --- |
| **Accuracy** | Controla os limites de precisão dos modelos | 8 | 1-10 |
| **Time** | Controla a duração do experimento | 5 | 1-10 |
| **Interpretability** | Controla a complexidade dos modelos | 8 | 1-10 |

### Especificação do sistema

|  | **Systema** | **Memória do sistema** | **CPUs** | **GPUs** |
| --- | --- | --- | --- | --- |
|  | Docker/Linux | 54 GB | 32 | 1 |

## 

## 2) [Visão geral do Dataset](#_heading=h.bj7ecn214d3h)

Essa tabela provê informações sobre o dataset usado no experimento.

|  | **Conjunto de Dados** | **Tamanho** | **Linhas** | **Colunas** |
| --- | --- | --- | --- | --- |
|  | Treinamento e Validação | 138.1 KiB | 3.330 | 13 |
|  | Teste | 58.7 KiB | 1.395 | 13 |
|  |  |  |  |  |

### Dados de treinamento

Os dados de treinamento consistem apenas em colunas numéricas.

O resumo das colunas é mostrado abaixo:

#### Colunas numéricas

| **Nome** | **Tipo Lógico** | **Tipo Armazenamento** | **min** | **mean** | **max** | **std** | **unique** | **freq of mode** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Date | date, raw, time column | time [%Y-%m-%d] | 2010-04-02 |  | 2011-08-26 |  | 74 | 45 |
| Store | raw | int | 1.000 | 23.000 | 45.000 | 12.989 | 45 | 74 |
| Type | num, categ, catlabel, ohe hot cat | int | 1.000 | 1.644 | 3.000 | 0.704 | 3 | 1,628 |
| Size | N/A | int | 34.875,00 | 13.0287,60 | 219,622,00 | 63.121,59 | 40 | 222 |
| Vendas Semanais | N/A | real | 209.975,28 | 1.039.793,70 | 3.818.685,09 | 562.177,82 | 3,330 | 1 |

**Colunas booleanas**

| **nome** | **tipo lógico** | **tipo armaz.** | **min** | **mean** | **max** | **std** | **freq of max value** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| EhFeriado | numeric, categorical, catlabel, ohe\_categorical | bool | False | 0.05405405405405406 | True | 0.22615829174691943 | 180 |
| Super\_Bowl | numeric, categorical, catlabel, ohe\_categorical | bool | False | 0.013513513513513514 | True | 0.11547685024017315 | 45 |
| Dia\_do\_Trabalho | numeric, categorical, catlabel, ohe\_categorical | bool | False | 0.013513513513513514 | True | 0.11547685024017315 | 45 |
| Thanksgiving | numeric, categorical, catlabel, ohe\_categorical | bool | False | 0.013513513513513514 | True | 0.11547685024017315 | 45 |
| Natal | numeric, categorical, catlabel, ohe\_categorical | bool | False | 0.013513513513513514 | True | 0.11547685024017315 | 45 |
| Type\_A | N/A | bool | False | 0.4888888888888889 | True | 0.49995160140912004 | 1,628 |
| Type\_B | N/A | bool | False | 0.37777777777777777 | True | 0.4849044634842886 | 1,258 |
| Type\_C | N/A | bool | False | 0.13333333333333333 | True | 0.33998568697454284 | 444 |

### Detecção de Vazamento nos Dados

A IA do sistema H2o pode realizar a detecção de deslocamento entre os conjuntos de dados de treinamento, validação e teste. Ela faz isso treinando um modelo binomial para prever a qual conjunto de dados um registro pertence. Por exemplo, ser capaz de separar os dados de treinamento e teste com uma AUC de 0,8 usando apenas a coluna: C1 como preditor. Isso indica que há algum tipo de desvio na distribuição de C1 entre os dados de treinamento e teste.

Para este experimento, a AI verificou os dados do treino e do teste em busca de qualquer mudança nas distribuições, mas não encontrou nenhuma. Isso indica que todos os preditores/colunas nos dados de treinamento e teste são da mesma distribuição.

## 

## 

## 3) [Metodologia](#_heading=h.jz4leguz5ipf)

Essa seção tem o objetivo de descrever o experimento.

### Suposições e Limitações

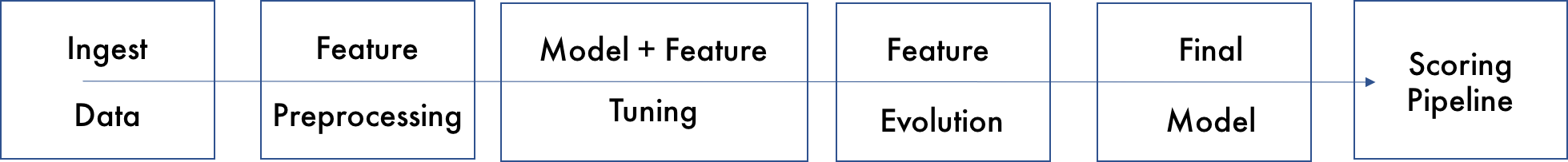
O sistema utilizado treina todos os modelos com base nos dados de treinamento fornecidos (neste caso: walmart\_train.csv). É pressuposto da IA que esse conjunto de dados seja representativo dos dados que serão vistos durante o teste.

Portanto, pode realizar a detecção de mudança entre o treino e os dados de teste. Se for detectada uma mudança na distribuição, isso pode indicar que os dados que serão usados para o teste podem ter distribuições não representadas nos dados de treinamento.

Para este experimento, o sistema realizou a detecção de mudança, mas não encontrou mudanças significativas na distribuição dos dados de trem e teste.

### Experimento do Pipeline

Para este experimento, a DriverlessAI realizou as seguintes etapas para encontrar o modelo final ideal:



Os passos são descritos abaixo:

* **Ingest Data**
* detecção dos tipos de dados nas colunas
* **Feature Preprocessing**
* transformação de recursos e valores numéricos
* **Model and Feature Tuning**

Este estágio combina o ajuste aleatório de hiperparâmetros com seleção e geração de recursos. Os recursos em cada iteração são atualizados usando a importância variável da iteração anterior como probabilística antes de decidir quais novos recursos criar. O modelo e os recursos com melhor desempenho são então passados para o estágio de evolução do recurso.

* encontrou os parâmetros ideais para modelos constant, xgboost e lightgbm treinando modelos com diferentes parâmetros
* os melhores parâmetros são aqueles que geram o menor RMSE nos dados de validação interna
* **41** modelos treinados e pontuados para avaliar recursos e parâmetros do modelo
* **Feature Evolution**

Esta etapa usa um algoritmo genético para encontrar o melhor conjunto de parâmetros do modelo e transformações de recursos a serem usados no modelo final.

* encontrou a melhor representação dos dados para o treinamento do modelo final, criando e avaliando **48** recursos em **33** iterações
* 364 modelos treinados e pontuados para avaliar melhor os recursos de engenharia
* **Final Model**
* criou o melhor modelo das iterações de engenharia de recursos
* nenhum conjunto **ensemble** é feito porque uma coluna de **tempo** foi fornecida
* **Create Scoring Pipeline**
* criou e exportou o pipeline de pontuação MOJO e Python
* MOJO Scoring Pipeline: ueliton\_viana@outlook.com/h2oai\_experiment\_e03b7a70-6e0a-11ed-8c76-0e8c0d6727b2/mojo\_pipeline/mojo.zip
* Python Scoring Pipeline: ueliton\_viana@outlook.com/h2oai\_experiment\_e03b7a70-6e0a-11ed-8c76-0e8c0d6727b2/scoring\_pipeline/scorer.zip

**Otimização dos modelos**

Modelos treinados durante todo o experimento para determinar os melhores parâmetros, conjunto de dados do modelo e modelo final ideal. As etapas são descritas a seguir:

| **Etapa DriverlessAI** | **Tempo(segundos)** | **Número de Modelos** |
| --- | --- | --- |
| **Data Preparation** | 13.78 | 0 |
| **Model and Feature Tuning** | 226.48 | 41 |
| **Feature Evolution** | 1,430.16 | 364 |
| **Final Pipeline Training** | 82.04 | 1 |

### Configurações do experimento

Abaixo estão as configurações selecionadas para o experimento por ueliton\_viana@outlook.com. Os parâmetros definidos representam os parâmetros de alto nível.

**Parâmetros Definidos**

| **Parâmetros** | **Valor** |
| --- | --- |
| is\_classification | False |
| ativacao\_de\_gpus | True |
| seed | False |
| accuracy | 8 |
| time | 5 |
| interpretability | 8 |
| num\_prediction\_periods (intervalo de tempo nos dados: 1 semana) | 1 |
| num\_gap\_periods (tempo perdido entre treino e teste) | 0 |
| is\_timeseries | True |
| is\_image | False |

Essas configurações de precisão, tempo e interpretabilidade são mapeadas para a seguinte configuração interna do experimento DriverlessAI:

| **Parâmetros Internos** | **Valor** |
| --- | --- |
| data filtered | False |
| tuna o alvo transformado | False |
| número de iterações de engenharia de recursos | 50 |
| número de modelos treinados por iteração | 8 |
| rodada em que ocorreu a parada precoce | 10 |
| restrição de monotonicidade | True |
| número de combinações de modelo | 10 |
| número de base-learners no ensemble | 0 |
| coluna de tempo | Date |
| grupo de colunas ligadas ao tempo | ['Date', 'Loja'] |
| intervalo de tempo nos dados (semana) | week |
| número de período de previsão | 1 |
| intervalo de tempo entre treino e teste | 0 |

#### Detalhes

* **data filtered:** DriverlessAI pode filtrar os dados de treinamento, dependendo do número de linhas e da configuração de precisão.
* para esse experimento, os dados não foram filtrados.
* **tuna o alvo transformado:** se DriverlessAI avaliou o desempenho do modelo se o alvo foi transformado.
* ex: o desempenho do modelo pode ser melhor ao prever o log da coluna de destino em vez da coluna de destino bruta.
* **número de iterações de engenharia de recursos**: o número de iterações realizadas da engenharia de recursos.
* **número de modelos treinados por iteração:** para cada iteração de engenharia de recursos, o DriverlessAI treina vários modelos. Cada modelo é treinado com um conjunto diferente de preditores ou recursos. O objetivo desta etapa é determinar quais tipos de recursos levam ao **menor RMSE**.
* **rodada em que ocorreu a parada precoce:** se DriverlessAI não observar nenhuma melhoria após **10** iterações de engenharia de recursos, a etapa de engenharia de recursos será interrompida automaticamente.
* **restrição de monotonicidade:** se ativado, os modelos terão apenas relacionamentos monótonos entre os preditores e a variável de destino.
* **número de combinações de modelo:** o número de combinações de ajuste de modelos avaliados para determinar as configurações de modelo ideais para os modelos constant, xgboost e lightgbm.
* **número de base-learners no ensemble:** o número de modelos-base usados para criar o ensemble final.
* **coluna de tempo:** a coluna que fornece a passagem do tempo. Se uma coluna de tempo for fornecida, a engenharia de recursos e a validação do modelo respeitarão a causalidade do tempo. Se a coluna de tempo estiver desativada, nenhuma ordem de tempo é usada para modelagem e os dados podem ser embaralhados aleatoriamente (qualquer causalidade temporal potencial será ignorada).
* **grupo de colunas ligadas ao tempo:** as colunas que compõem os grupos de séries temporais.
* intervalo de tempo nos dados (semana): o intervalo de tempo nos dados.
* **número de período de previsão:** o número de períodos (semanas) que deseja-se realizar a previsão.
* **intervalo de tempo entre treino e teste:** o intervalo entre os dados disponíveis e o período de previsão desejado.

## 4) [Amostragem de dados](#_heading=h.tyjcwt)

No DriverlessAI, a amostragem de dados (Data Sampling) é uma etapa de pré-processamento realizada antes do início do treinamento do modelo; não está relacionado à amostragem feita durante o treinamento do modelo. DriverlessAI não realiza amostragem de dados, a menos que o conjunto de dados seja grande ou altamente desequilibrado. Se um conjunto de dados é considerado grande depende da configuração de precisão do experimento e do parâmetro config.toml static\_threshold\_data\_size\_large.

O DriverlessAI não realizou nenhuma amostragem dos dados.

## 

## 

## 5) [Estratégia de Validação e Ajuste](#_heading=h.ar7084j8kbjw)

Os dados de treino são divididos automaticamente entre treinamento e validação, ordenando os dados por Data. O modelo foi ajustado para prever 1 semana sem intervalo entre o treinamento e a previsão.

### Divisões de validação com base no tempo para ajuste de modelo e evolução de recursos

A cor azul representa a divisão dos dados de treinamento, enquanto a cor cinza representa a divisão dos dados de validação. As seguintes divisões de dados foram usadas:

**etapa\_divisao\_1\_train**

* inicio 2010-04-02
* final 2011-06-24

**etapa\_divisao\_1\_valid**

* inicio 2011-07-01
* final 2011-08-26

**etapa\_divisao\_2\_train**

* inicio 2010-04-02
* final 2011-04-22

**etapa\_divisao\_2\_valid**

* inicio 2011-04-29
* final 2011-06-24

**etapa\_divisao\_3\_train**

* inicio 2010-04-02
* final 2011-02-18

**etapa\_divisao\_3\_valid**

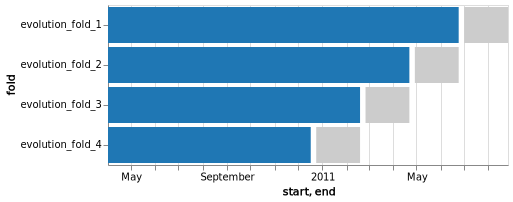
* inicio 2011-02-25
* final 2011-04-22

**etapa\_divisao\_4\_train**

* inicio 2010-04-02
* final 2010-12-17

**etapa\_divisao\_4\_valid**

* inicio 2010-12-24
* final 2011-02-18



### Intervalo de Tempo no Treinamento do Pipeline Final

**final\_pipeline\_train**

* inicio 2010-04-02
* final 2011-08-26



## Ajuste de modelo

A tabela abaixo mostra a pontuação e o tempo de treinamento dos modelos constant, xgboost e lightgbm avaliados pelo DriverlessAI. A tabela mostra os modelos de ajuste de parâmetros avaliados, ordenados com base em uma combinação de menor pontuação e menor tempo de treinamento.

| **ordem dos testes** | **booster** | **n\_features** | **tempo de treinamento** | **scores (RMSE)** |
| --- | --- | --- | --- | --- |
| 5 | lightgbm | 39 | 7.72 | 123.766.67 |
| 0 | lightgbm | 92 | 12.02 | 126.228.81 |
| 4 | lightgbm | 37 | 16.33 | 127.007.17 |
| 6 | lightgbm | 36 | 11.35 | 128.559.08 |
| 8 | lightgbm | 37 | 7.35 | 128.600.00 |
| 7 | lightgbm | 32 | 10.56 | 128.897.82 |
| 2 | lightgbm | 42 | 5.15 | 132.657.92 |
| 3 | lightgbm | 114 | 12.70 | 139.728.60 |
| 1 | gbtree | 114 | 23.51 | 148.081.59 |
| 9 | constant | 1 | 2.98 | 559.824.25 |

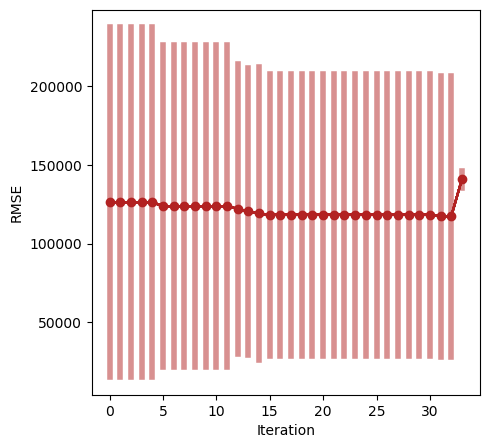
## 

## 6) [Evolução do RMSE](#_heading=h.3p814xomrmow)

Durante o estágio de ajuste de modelo e recurso, o DriverlessAI avalia os efeitos de diferentes tipos de algoritmos, parâmetros de algoritmo e recursos. O objetivo do estágio de ajuste de modelo e recurso é determinar o melhor algoritmo e parâmetros a serem usados durante o estágio de evolução do recurso.

No estágio de evolução de recursos, DriverlessAI treinou modelos constantes, xgboost e lightgbm (364), onde cada modelo avaliou um conjunto diferente de recursos. O Estágio de Evolução de Recursos usa um algoritmo genético para pesquisar o grande espaço de engenharia de recursos.

O gráfico abaixo mostra o efeito que o estágio de ajuste de modelo e recurso e o estágio de evolução de recurso tiveram no desempenho.



* Com base nas configurações do experimento e nos tipos de coluna no conjunto de dados, o DriverlessAI conseguiu explorar os seguintes transformadores:
* **InteractionsTransformer**: o Interactions Transformer adiciona, divide, multiplica e subtrai duas colunas numéricas nos dados para criar um novo recurso.
* **CatOriginalTransformer**: o Categorical Original Transformer aplica uma transformação de identidade que deixa os recursos categóricos como estão. Este transformador funciona com modelos que podem lidar com valores de recursos não numéricos.
* **FrequentTransformer**: o Frequent Transformer calcula a frequência para cada valor na(s) coluna(s) categórica(s) e usa isso como um novo recurso. Essa contagem pode ser a contagem bruta ou a contagem normalizada.
* **CatTransformer**: o Categorical Transformer classifica uma coluna categórica em ordem lexicográfica e usa o índice de ordem criado como um novo recurso. Este transformador funciona com modelos que podem lidar com recursos categóricos.
* **OneHotEncodingTransformer**: o transformador One-hot Encoding converte uma coluna categórica em uma série de recursos booleanos executando a codificação one-hot. Os recursos booleanos são usados ​​como novos recursos.
* **IsHolidayTransformer**: o transformador Is Holiday determina se uma coluna de data é feriado. Uma coluna booleana indicando se a data é um feriado é adicionada como um novo recurso. Cria um recurso separado para feriados nos Estados Unidos, Reino Unido, Alemanha, México e Banco Central Europeu. Outros países disponíveis no pacote de férias do python podem ser adicionados por meio do arquivo de configuração.
* **DatesTransformer**: o Date Transformer recupera quaisquer valores de data ou hora, incluindo: Ano, Trimestre, Mês, Dia, Dia do Ano, Semana, Dia da Semana, Hora, Minuto, Segundo.

**EwmaLagsTransformer**: o transformador de média móvel ponderada exponencial (EWMA) calcula a média móvel ponderada exponencial dos atrasos da coluna que se fará a previsão (Vendas semanais).

**Colunas Deletadas**

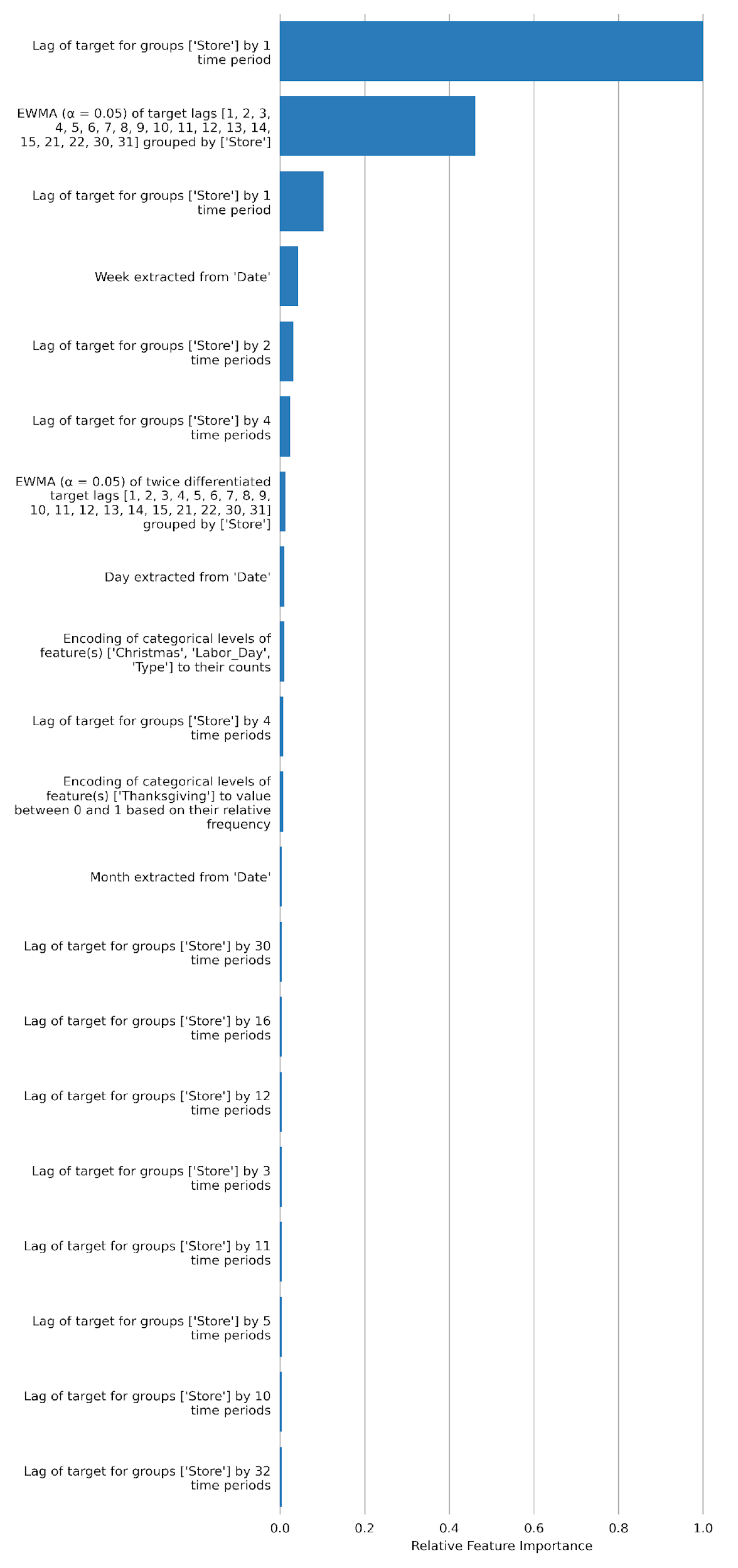
Abaixo está a lista completa de recursos descartados devido a vários motivos.

| **Nome** | **Razão** |
| --- | --- |
| EhFeriado | Modelo DAI Deletou Automaticamente |
| Tamanho da Loja (Vazamento de Dados) | Usuário Deletou |
| Super\_Bowl | Modelo DAI Deletou Automaticamente |
| Type\_A (O modelo aplica dummies otimizadas) | Usuário Deletou |
| Type\_B (O modelo aplica dummies otimizadas) | Usuário Deletou |
| Type\_C (O modelo aplica dummies otimizadas) | Usuário Deletou |

## 7) [Transformações de Recursos Finais](#_heading=h.oehfh73py4vx)

O resultado do Estágio de Evolução de Recursos é um conjunto de recursos a serem usados no modelo final. Alguns desses recursos foram criados automaticamente pelo DriverlessAI. Os principais recursos usados no modelo final são mostrados abaixo, ordenados por importância. As feições da tabela estão limitadas às 50 primeiras, restritas àquelas com importância relativa maior ou igual a 0,003. Se nenhum transformador foi aplicado, o recurso é uma coluna original.

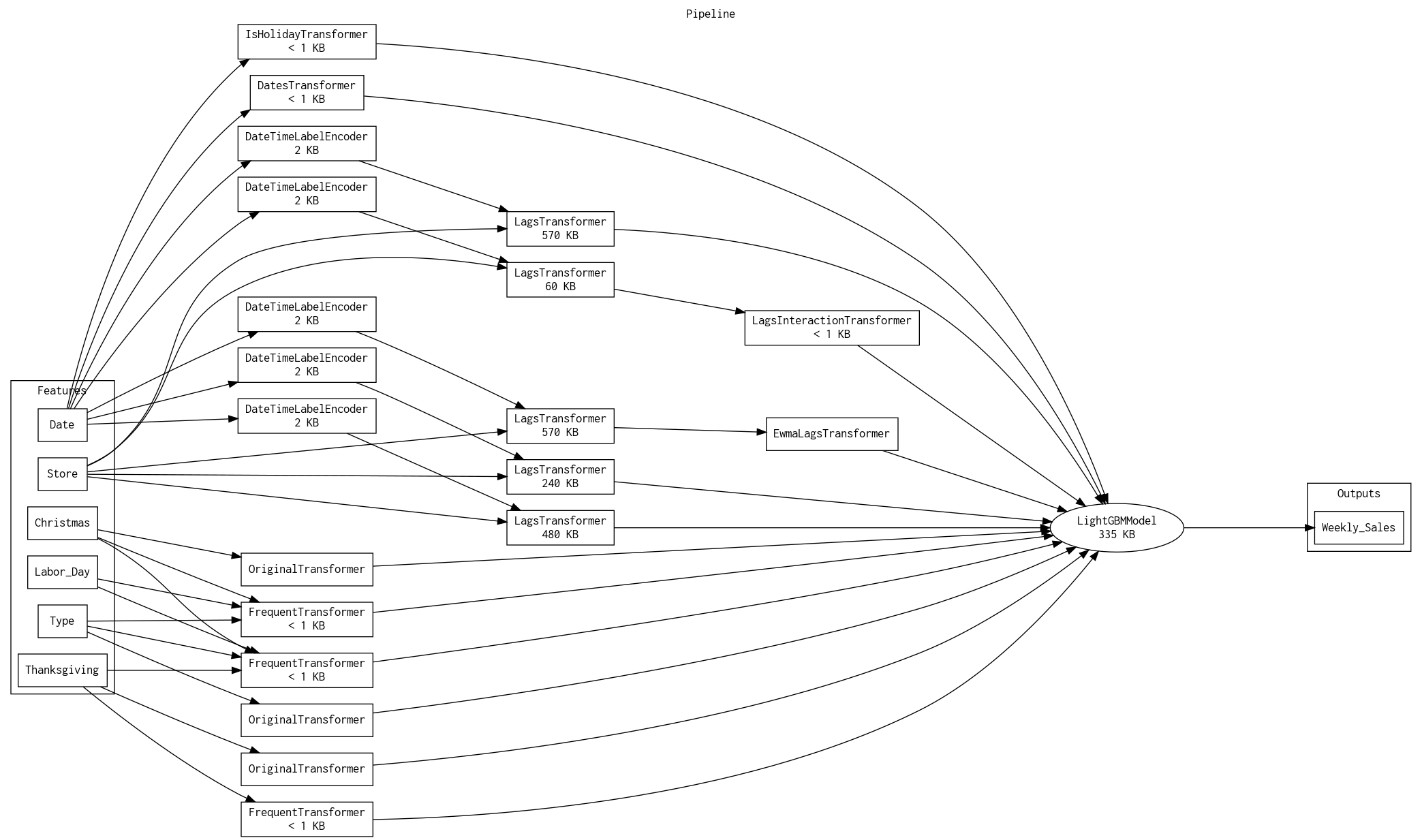
|  | **Feature** | **Descrição** | **Transformação** | **Import. Relativa** |
| --- | --- | --- | --- | --- |
| 1 | 13\_TargetLag: Date: Store.1 | Atraso da meta para grupos ['Loja'] por 1 período de tempo | Lags | 1.0 |
| 2 | 23\_EWMA(0.05)(0)TargetLags: Date: Store.1~2~3~4~5~6~7~8~9~10~11~12~13~14~15~21~22~30~31 | EWMA (α = 0,05) de atrasos vendas semanais [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 21, 22, 30, 31] agrupados por ['Loja'] | Exponential Weighted Moving Average | 0.462 |
| 3 | 40\_TargetLag: Date: Store.1 | Atraso da meta para grupos ['Loja'] por 1 período de tempo | Lags | 0.1034 |
| 4 | 12\_Date: Date~get\_week | Semana extraída de 'Data' | Date Expansion | 0.0429 |
| 5 | 13\_TargetLag: Date: Store.2 | Atraso das vendas semanais por grupos ['Loja'] por 2 períodos de tempo | Lags | 0.0315 |
| 6 | 13\_TargetLag: Date: Store.4 | Atraso das vendas semanais por grupos ['Loja'] por 4 períodos de tempo | Lags | 0.0246 |
| 7 | 23\_EWMA(0.05)(2)TargetLags: Date: Store.1~2~3~4~5~6~7~8~9~10~11~12~13~14~15~21~22~30~31 | EWMA (α = 0,05) de atrasos das vendas semanais duas vezes diferenciados [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 21, 22, 30, 31] agrupados por ['Loja'] | Exponential Weighted Moving Average | 0.0131 |
| 8 | 12\_Date: Date~get\_day | Dia extraído de 'Data' | Date Expansion | 0.0107 |
| 9 | 20\_Freq: Natal: Dia\_do\_Trabalho: Tipo | Codificação de níveis de frequência nos dados categóricos ['Natal','Dia\_do\_Trabalho', 'Tipo'] | Frequency Encoding | 0.0103 |
| 10 | 35\_TargetLag: Date: Store.4 | Atraso das vendas semanais por grupos ['Loja'] por 4 períodos de tempo | Lags | 0.0086 |
| 11 | 10\_Freq: Thanksgiving | Codificação de níveis categóricos de recursos ['Ação de Graças'] para valores entre 0 e 1 com base em sua frequência relativa | Frequency Encoding | 0.0076 |
| 12 | 12\_Date: Date~get\_month | Meses extraídos de 'Data' | Date Expansion | 0.0047 |
| 13 | 19\_Freq: Natal: Dia\_do\_Trabalho: Thanksgiving: Type | Codificação de níveis categóricos de característica(s) ['Natal', 'Dia\_do\_Trabalho', 'Ação de Graças', 'Tipo'] para suas contagens | Frequency Encoding | 0.0037 |
| 14 | 0\_Natal | Natal (Original) | None | 0.0034 |



## 8) [Modelo Final](#_heading=h.lnxbz9)

**Pipeline**

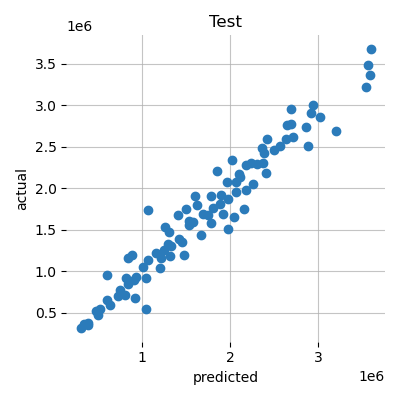
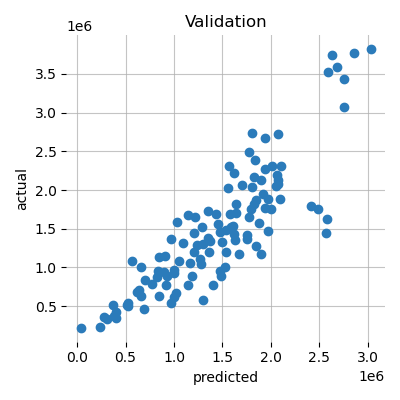
Modelo de Pipeline final do LightGBM tem ensemble\_level=0 transformando **6** recursos originais em **51** recursos, ajustando à retenção do modelo baseada em tempo. Esta imagem abaixo, está disponível em alta resolução nos arquivos em anexo com o nome pipeline.png:



**Performance do Modelo Final**

| **Métrica** | **Otimizado** | **Melhor pontuação é** | **Pontuações finais do conjunto na validação** | **Desvio padrão do conjunto final na validação** | **RMSE Final no Teste** | **Desvio Padrão no Teste** |
| --- | --- | --- | --- | --- | --- | --- |
| **RMSE** | \* | lower | 140.832 | 5.328.68 | 81.820.72 | 5.328.68 |
| GINI |  | higher | 0.9796695 | 0.001144529 | 0.9918682 | 0.001144529 |
| **MAE** |  | lower | 74984.23 | 1925.66 | 52683.84 | 1925.66 |
| MAPE |  | lower | 7.194266 | 0.1966101 | 5.123156 | 0.1966101 |
| MER |  | lower | 4.541406 | 0.1019572 | 3.683587 | 0.1171577 |
| MSE |  | lower | 1.983364e+10 | 1.235267e+09 | 6.69463e+09 | 1.235267e+09 |
| R2COD |  | higher | 0.937231 | 0.003929691 | 0.9811788 | 0.003929691 |
| R2 |  | higher | 0.9372913 | 0.004044363 | 0.9815589 | 0.004044363 |
| RMSLE |  | lower | 0.1183814 | 0.005789305 | 0.07261892 | 0.005789305 |
| RMSPE |  | lower | 11.73719 | 0.4055685 | 7.475851 | 0.453842 |
| SMAPE |  | lower | 7.14579 | 0.1308216 | 5.084216 | 0.1308216 |

**Atual vs Predito**



## 9) [Implantação](#_heading=h.1ksv4uv)

Para este experimento, os pipelines de pontuação Python e MOJO estão disponíveis para produzir o pipeline de modelo final para uma determinada linha de dados ou tabela de dados.

### Pipeline de Pontuação do Python

Este pacote contém um modelo exportado e exemplos de código-fonte Python 3.6 para produzir modelos criados usando o H2O DriverlessAI. O Python Scoring Pipeline está localizado aqui:

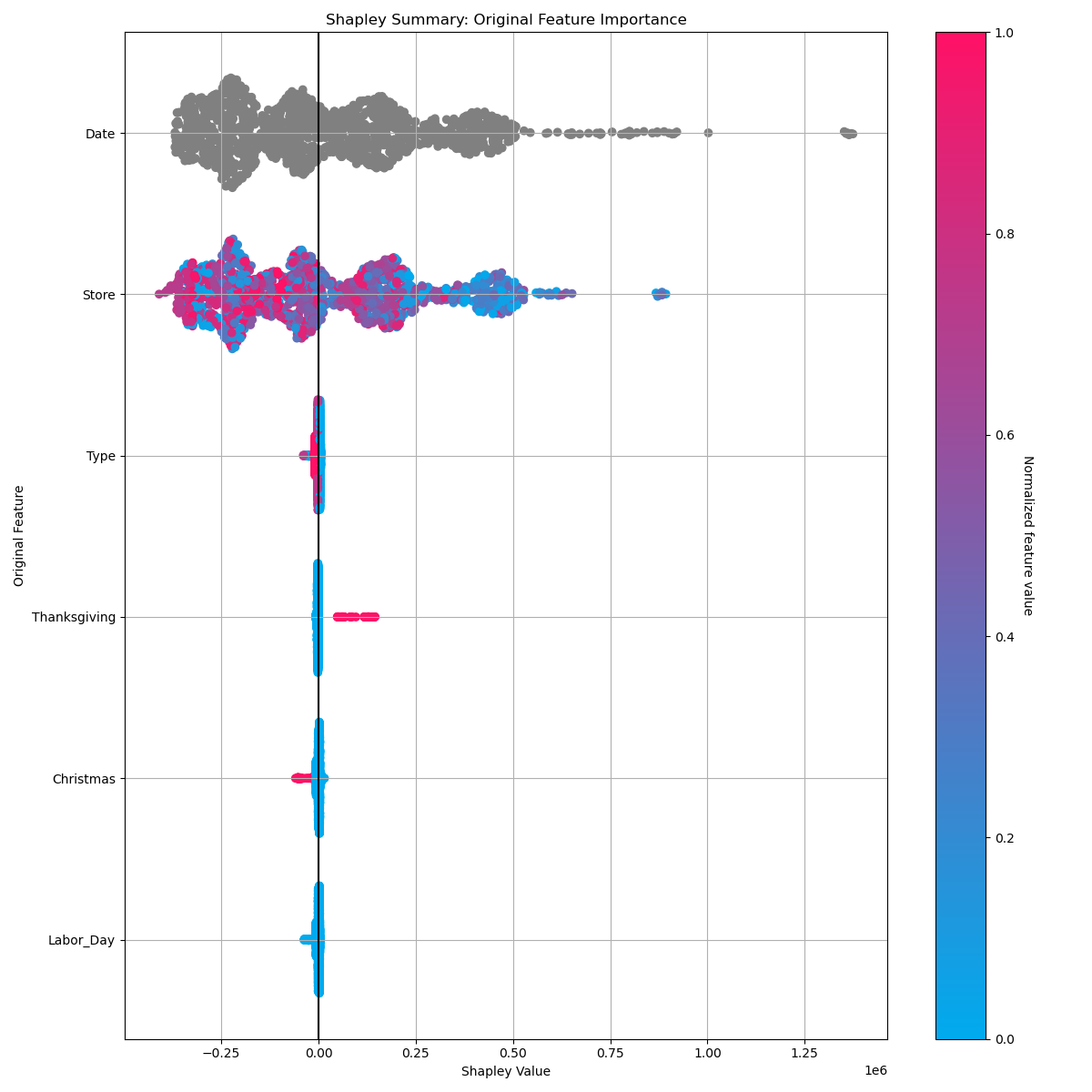
* **ueliton\_viana@outlook.com/h2oai\_experiment\_e03b7a70-6e0a-11ed-8c76-0e8c0d6727b2/scoring\_pipeline/scorer.zip**

## 10) Shapley Values

##### Contribuições de Shapley no conjunto de dados de **teste**

As explicações de Shapley são uma técnica com suporte teórico confiável que apresenta contribuições de características (globais e locais) consistentes. Para problemas de regressão, as contribuições de recursos locais de Shapley mais a soma do termo de viés para a previsão do modelo final.

Esta seção usa o método Naive Shapley da DriverlessAI para calcular as explicações locais de Shapley para recursos originais. Essas explicações são aproximações para os recursos originais e são baseadas na frequência com que os recursos são usados em *recursos transformados* e na *importância desses recursos transformados para o modelo final*. A importância de cada feição transformada é distribuída igualmente para todas as feições originais que ajudaram a criá-la. Isso é então somado para cada feição original.



## 11) [Apêndice](#_heading=h.z337ya)

### Detalhes finais do modelo

| **Índice do modelo** | **Tipo** | **Tamanho** | **Recursos** | **Transformação no Target** |
| --- | --- | --- | --- | --- |
| 0 | LightGBMModel | 1 | 51 | identity\_noclip |

**Índice do Modelo: Modelo Final - Modelo Único**

| **parâmetro** | **valor** |
| --- | --- |
| IS\_FINAL | True |
| \_build\_info | {'commit': 'a093413', 'version': '1.10.3.1'} |
| accuracy | 8 |
| bagging\_seed | 420805510 |
| booster | lightgbm |
| boosting\_type | gbdt |
| categorical\_feature |  |
| class\_weight |  |
| colsample\_bytree | 0.8 |
| deterministic | False |
| device\_type | cpu |
| disable\_gpus | False |
| encoder | {'bad\_fit': False, 'copy': True, 'description': 'DateTimeLabelEncoder', 'dummy\_tr': None, 'estimated\_output\_feature\_count': 1, 'experiment\_description\_origin': 'Unknown', 'feature\_desc': None, 'fit\_exception': None, 'fitted\_output\_feature\_names': None, 'full\_feature\_desc': None, 'input\_feature\_names': ['Date'], 'is\_custom': False, 'is\_orig': True, 'is\_raw': False, 'labels': None, 'monthly': False, 'out\_types': {}, 'output\_feature\_extra\_descriptions': None, 'output\_feature\_names\_backup': None, 'output\_features\_to\_drop': [], 'random\_state': 42, 'target': None, 'time\_bins': [1.269864e+18, 1.2704688e+18, 1.2710736e+18, 1.2716784e+18, 1.2722832e+18, 1.272888e+18, 1.2734928e+18, 1.2740976e+18, 1.2747024e+18, 1.2753072e+18, 1.275912e+18, 1.2765168e+18, 1.2771216e+18, 1.2777264e+18, 1.2783312e+18, 1.278936e+18, 1.2795408e+18, 1.2801456e+18, 1.2807504e+18, 1.2813552e+18, 1.28196e+18, 1.2825648e+18, 1.2831696e+18, 1.2837744e+18, 1.2843792e+18, 1.284984e+18, 1.2855888e+18, 1.2861936e+18, 1.2867984e+18, 1.2874032e+18, 1.288008e+18, 1.2886128e+18, 1.2892176e+18, 1.2898224e+18, 1.2904272e+18, 1.291032e+18, 1.2916368e+18, 1.2922416e+18, 1.2928464e+18, 1.2934512e+18, 1.294056e+18, 1.2946608e+18, 1.2952656e+18, 1.2958704e+18, 1.2964752e+18, 1.29708e+18, 1.2976848e+18, 1.2982896e+18, 1.2988944e+18, 1.2994992e+18, 1.300104e+18, 1.3007088e+18, 1.3013136e+18, 1.3019184e+18, 1.3025232e+18, 1.303128e+18, 1.3037328e+18, 1.3043376e+18, 1.3049424e+18, 1.3055472e+18, 1.306152e+18, 1.3067568e+18, 1.3073616e+18, 1.3079664e+18, 1.3085712e+18, 1.309176e+18, 1.3097808e+18, 1.3103856e+18, 1.3109904e+18, 1.3115952e+18, 1.3122e+18, 1.3128048e+18, 1.3134096e+18, 1.3140144e+18, 1.3146192e+18, 1.315224e+18, 1.3158288e+18, 1.3164336e+18, 1.3170384e+18, 1.3176432e+18, 1.318248e+18, 1.3188528e+18, 1.3194576e+18, 1.3200624e+18, 1.3206672e+18, 1.321272e+18, 1.3218768e+18, 1.3224816e+18, 1.3230864e+18, 1.3236912e+18, 1.324296e+18, 1.3249008e+18, 1.3255056e+18, 1.3261104e+18, 1.3267152e+18, 1.32732e+18, 1.3279248e+18, 1.3285296e+18, 1.3291344e+18, 1.3297392e+18, 1.330344e+18, 1.3309488e+18, 1.3315536e+18, 1.3321584e+18, 1.3327632e+18, 1.3330656e+18, 1.3339728e+18, 1.3345776e+18, 1.3351824e+18, 1.3357872e+18, 1.336392e+18, 1.3369968e+18, 1.3376016e+18, 1.3382064e+18, 1.3388112e+18, 1.339416e+18, 1.3400208e+18, 1.3406256e+18, 1.3412304e+18, 1.3418352e+18, 1.34244e+18, 1.3430448e+18, 1.3436496e+18, 1.3442544e+18, 1.3448592e+18, 1.345464e+18, 1.3460688e+18, 1.3466736e+18, 1.3472784e+18, 1.3478832e+18, 1.348488e+18, 1.3490928e+18, 1.3496976e+18, 1.3503024e+18, 1.3509072e+18, 1.351512e+18, 1.3521168e+18, 1.3527216e+18, 1.3533264e+18, 1.3539312e+18, 1.354536e+18, 1.3551408e+18, 1.3557456e+18, 1.3563504e+18, 1.3569552e+18, 1.35756e+18, 1.3581648e+18, 1.3587696e+18, 1.3593744e+18, 1.3599792e+18, 1.360584e+18, 1.3611888e+18, 1.3617936e+18, 1.3623984e+18, 1.3630032e+18, 1.363608e+18, 1.3642128e+18, 1.3648176e+18, 1.3654224e+18, 1.3660272e+18, 1.366632e+18, 1.3672368e+18, 1.3678416e+18, 1.3684464e+18, 1.3690512e+18, 1.369656e+18, 1.3702608e+18, 1.3708656e+18, 1.3714704e+18, 1.3720752e+18, 1.37268e+18, 1.3732848e+18, 1.3738896e+18, 1.3744944e+18, 1.3750992e+18, 1.375704e+18, 1.3763088e+18, 1.3769136e+18, 1.3775184e+18, 1.3781232e+18, 1.378728e+18, 1.3793328e+18, 1.3799376e+18, 1.3805424e+18, 1.3811472e+18, 1.381752e+18, 1.3823568e+18, 1.3829616e+18, 1.3835664e+18, 1.3841712e+18, 1.384776e+18, 1.3853808e+18, 1.3859856e+18, 1.3865904e+18, 1.3871952e+18, 1.3878e+18, 1.3884048e+18, 1.3890096e+18, 1.3896144e+18, 1.3902192e+18, 1.390824e+18, 1.3914288e+18, 1.3920336e+18, 1.3926384e+18, 1.3932432e+18, 1.393848e+18, 1.3944528e+18, 1.3950576e+18, 1.3956624e+18], 'ts': 1270166400000000000, 'weekdays\_only': False} |
| ensemble\_level | 0 |
| eval\_metric | rmse |
| experiment\_description | Minimal Lag Size-gupiwufu |
| feature\_fraction\_seed | 420805509 |
| gamma | 0.01 |
| gpu\_device\_id | 0 |
| gpu\_id | 0 |
| gpu\_platform\_id | 0 |
| gpu\_use\_dp | False |
| grow\_policy | depthwise |
| importance\_type | gain |
| interpretability | 8 |
| label\_counts |  |
| labels | None |
| layer | 0 |
| learning\_rate | 0.05 |
| lossguide | False |
| max\_bin | 127 |
| max\_delta\_step | 0.0 |
| max\_depth | 7 |
| max\_leaves | 128 |
| min\_child\_samples | 5 |
| min\_child\_weight | 0.001 |
| min\_data\_in\_bin | 3 |
| min\_split\_gain | 0.0 |
| model\_class\_name | LightGBMModel |
| model\_id | Final Model - Single Model |
| model\_origin | DefaultIndiv: do\_te:catlabel,accuracy:8,interp:8,num\_as\_cat:True |
| monotone\_constraints | [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, -1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, -1] |
| monotone\_constraints\_method | intermediate |
| monotone\_penalty | 0.0 |
| monotonicity\_constraints | True |
| n\_estimators | 152 |
| n\_gpus | 1 |
| n\_jobs | 4 |
| ngenes | 14 |
| ngenes\_max | 100 |
| num\_class | 1 |
| num\_classes | 1 |
| num\_layers | 1 |
| num\_leaves | 128 |
| num\_threads | 4 |
| objective | mse |
| optuna\_trial |  |
| optuna\_trial\_state | unused |
| outer\_trial | [0, 2] |
| pred\_gap | 0 |
| pred\_periods | 1 |
| random\_state | 420805508 |
| reg\_alpha | 5.0 |
| reg\_lambda | 0.5 |
| resumed\_experiment\_id |  |
| scale\_pos\_weight | 1.0 |
| score\_f\_name | RMSE |
| seed | 420805508 |
| silent | True |
| str\_uuid | ret\_75982994-02ce-40bb-a57f-1fa0a0704109 |
| subsample | 0.9 |
| subsample\_for\_bin | 200000 |
| subsample\_freq | 5 |
| target | Weekly\_Sales |
| test\_data\_name | [Test] |
| test\_shape | [1, 1] |
| tgc | ['Date', 'Loja'] |
| time\_column | Date |
| time\_tolerance | 5 |
| train\_dataset\_name | walmart\_train.csv |
| train\_shape | [3330, 9] |
| tsp | {'datetime\_format': '%Y-%m-%d', 'dtlabel\_encoder': {'bad\_fit': False, 'copy': True, 'description': 'DateTimeLabelEncoder', 'dummy\_tr': None, 'estimated\_output\_feature\_count': 1, 'experiment\_description\_origin': 'Unknown', 'feature\_desc': None, 'fit\_exception': None, 'fitted\_output\_feature\_names': None, 'full\_feature\_desc': None, 'input\_feature\_names': ['Date'], 'is\_custom': False, 'is\_orig': True, 'is\_raw': False, 'labels': None, 'monthly': False, 'out\_types': {}, 'output\_feature\_extra\_descriptions': None, 'output\_feature\_names\_backup': None, 'output\_features\_to\_drop': [], 'random\_state': 42, 'target': None, 'time\_bins': [1.269864e+18, 1.2704688e+18, 1.2710736e+18, 1.2716784e+18, 1.2722832e+18, 1.272888e+18, 1.2734928e+18, 1.2740976e+18, 1.2747024e+18, 1.2753072e+18, 1.275912e+18, 1.2765168e+18, 1.2771216e+18, 1.2777264e+18, 1.2783312e+18, 1.278936e+18, 1.2795408e+18, 1.2801456e+18, 1.2807504e+18, 1.2813552e+18, 1.28196e+18, 1.2825648e+18, 1.2831696e+18, 1.2837744e+18, 1.2843792e+18, 1.284984e+18, 1.2855888e+18, 1.2861936e+18, 1.2867984e+18, 1.2874032e+18, 1.288008e+18, 1.2886128e+18, 1.2892176e+18, 1.2898224e+18, 1.2904272e+18, 1.291032e+18, 1.2916368e+18, 1.2922416e+18, 1.2928464e+18, 1.2934512e+18, 1.294056e+18, 1.2946608e+18, 1.2952656e+18, 1.2958704e+18, 1.2964752e+18, 1.29708e+18, 1.2976848e+18, 1.2982896e+18, 1.2988944e+18, 1.2994992e+18, 1.300104e+18, 1.3007088e+18, 1.3013136e+18, 1.3019184e+18, 1.3025232e+18, 1.303128e+18, 1.3037328e+18, 1.3043376e+18, 1.3049424e+18, 1.3055472e+18, 1.306152e+18, 1.3067568e+18, 1.3073616e+18, 1.3079664e+18, 1.3085712e+18, 1.309176e+18, 1.3097808e+18, 1.3103856e+18, 1.3109904e+18, 1.3115952e+18, 1.3122e+18, 1.3128048e+18, 1.3134096e+18, 1.3140144e+18, 1.3146192e+18, 1.315224e+18, 1.3158288e+18, 1.3164336e+18, 1.3170384e+18, 1.3176432e+18, 1.318248e+18, 1.3188528e+18, 1.3194576e+18, 1.3200624e+18, 1.3206672e+18, 1.321272e+18, 1.3218768e+18, 1.3224816e+18, 1.3230864e+18, 1.3236912e+18, 1.324296e+18, 1.3249008e+18, 1.3255056e+18, 1.3261104e+18, 1.3267152e+18, 1.32732e+18, 1.3279248e+18, 1.3285296e+18, 1.3291344e+18, 1.3297392e+18, 1.330344e+18, 1.3309488e+18, 1.3315536e+18, 1.3321584e+18, 1.3327632e+18, 1.3330656e+18, 1.3339728e+18, 1.3345776e+18, 1.3351824e+18, 1.3357872e+18, 1.336392e+18, 1.3369968e+18, 1.3376016e+18, 1.3382064e+18, 1.3388112e+18, 1.339416e+18, 1.3400208e+18, 1.3406256e+18, 1.3412304e+18, 1.3418352e+18, 1.34244e+18, 1.3430448e+18, 1.3436496e+18, 1.3442544e+18, 1.3448592e+18, 1.345464e+18, 1.3460688e+18, 1.3466736e+18, 1.3472784e+18, 1.3478832e+18, 1.348488e+18, 1.3490928e+18, 1.3496976e+18, 1.3503024e+18, 1.3509072e+18, 1.351512e+18, 1.3521168e+18, 1.3527216e+18, 1.3533264e+18, 1.3539312e+18, 1.354536e+18, 1.3551408e+18, 1.3557456e+18, 1.3563504e+18, 1.3569552e+18, 1.35756e+18, 1.3581648e+18, 1.3587696e+18, 1.3593744e+18, 1.3599792e+18, 1.360584e+18, 1.3611888e+18, 1.3617936e+18, 1.3623984e+18, 1.3630032e+18, 1.363608e+18, 1.3642128e+18, 1.3648176e+18, 1.3654224e+18, 1.3660272e+18, 1.366632e+18, 1.3672368e+18, 1.3678416e+18, 1.3684464e+18, 1.3690512e+18, 1.369656e+18, 1.3702608e+18, 1.3708656e+18, 1.3714704e+18, 1.3720752e+18, 1.37268e+18, 1.3732848e+18, 1.3738896e+18, 1.3744944e+18, 1.3750992e+18, 1.375704e+18, 1.3763088e+18, 1.3769136e+18, 1.3775184e+18, 1.3781232e+18, 1.378728e+18, 1.3793328e+18, 1.3799376e+18, 1.3805424e+18, 1.3811472e+18, 1.381752e+18, 1.3823568e+18, 1.3829616e+18, 1.3835664e+18, 1.3841712e+18, 1.384776e+18, 1.3853808e+18, 1.3859856e+18, 1.3865904e+18, 1.3871952e+18, 1.3878e+18, 1.3884048e+18, 1.3890096e+18, 1.3896144e+18, 1.3902192e+18, 1.390824e+18, 1.3914288e+18, 1.3920336e+18, 1.3926384e+18, 1.3932432e+18, 1.393848e+18, 1.3944528e+18, 1.3950576e+18, 1.3956624e+18], 'ts': 1270166400000000000, 'weekdays\_only': False}, 'frequency\_in\_si': 1.65, 'period': 604800000000000.0, 'period\_in\_si': 1.0, 'test\_periods': 31, 'time\_column': 'Date', 'time\_resolution\_si': 'wk', 'train\_periods': 74} |
| ufapt | [] |
| uses\_gpu | True |
| valid\_data\_name | [Valid] |
| valid\_shape | [1, 1] |
| verbose | -1 |
| nfolds | 1 |