

Projeto Final de Curso – Eng. ML

Eduardo Montovanelli Dalmaso
<https://github.com/eduardomdalmaso/kobe-dataset>

Agenda do Trabalho

O aluno deve preencher essa apresentação com os resultados da sua implementação do modelo. Os códigos devem ser disponibilizados em repositório próprio, público, para inspeção.

Essa apresentação é padronizada para que os alunos possam incluir os seus resultados, com figuras, tabelas e descrições sobre o projeto de curso. Os resultados aqui descritos serão confrontados com os códigos disponibilizados.

Roteiro

- Objetivo da modelagem
- Arquitetura da solução
 - Diagrama
 - Bibliotecas
 - Artefatos e Métricas
- Pipeline de processamento dos dados
 - Descrição dos dados
 - Análise Exploratória
 - Seleção base de teste
- Pipeline de Treinamento do Modelo
 - Validação Cruzada
 - Regressão Logística
 - Árvore de Decisão
 - Seleção, finalização e registro
- Aplicação do Modelo
 - Model as a Service localmente
 - Interface para aplicação na base de produção
 - Monitoramento do modelo

Objetivo da modelagem

Em homenagem ao jogador da NBA Kobe Bryant (falecido em 2020), foram disponibilizados os dados de 20 anos de arremessos, bem sucedidos ou não, e informações correlacionadas.

O objetivo desse estudo é aplicar técnicas de inteligência artificial para prever se um arremesso será convertido em pontos ou não.

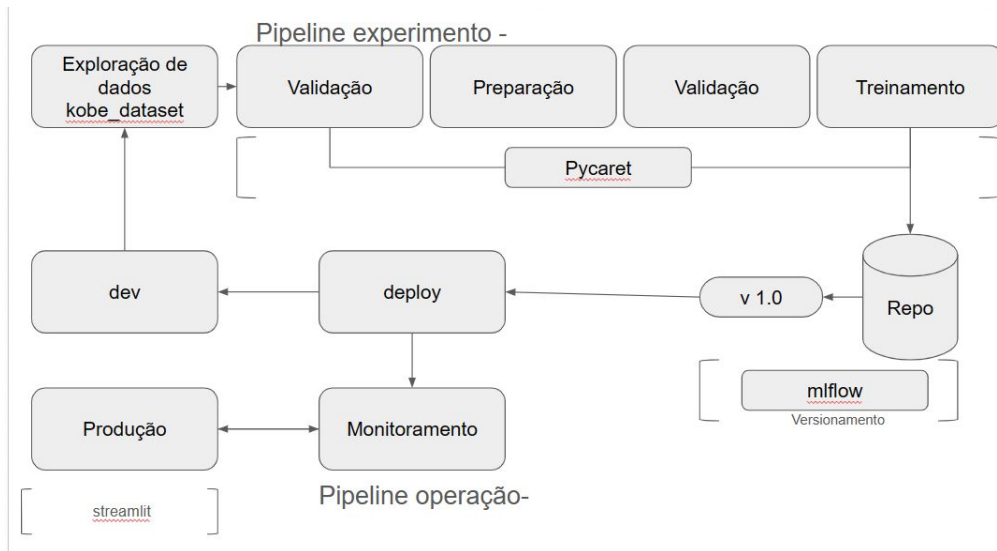


Arquitetura da Solução

Arquitetura da Solução

Diagrama

Pipeline ajuda na automatização do projeto, reprodutibilidade, escalabilidade e manutenção, além da governança sobre o processo.



Arquitetura da Solução

Bibliotecas

PyCaret:

- Log de experimentos automático: Registra automaticamente os parâmetros do modelo, métricas de desempenho e outras informações relevantes para cada experimento.
- Comparação de modelos: Permite comparar facilmente o desempenho de diferentes modelos em um único painel.
- O Pycaret facilita a comparação de vários modelos de machine learning em termos de desempenho e métricas. Ele fornece uma tabela resumida com os resultados de diferentes algoritmos, permitindo que você tome decisões informadas sobre qual modelo usar.

MLflow:

- Experimentos rastreáveis: Rastreia experimentos de ML com código, dados, resultados e artefatos em um único local.
- Compartilhamento de experimentos: Permite compartilhar experimentos com outros colaboradores e reproduzir resultados com facilidade.
- Gerenciamento de versões: Permite gerenciar diferentes versões de modelos e experimentos.

Arquitetura da Solução

Bibliotecas

MLflow

- Rastrear métricas e parâmetros durante o desenvolvimento de modelos, é importante registrar métricas como acurácia e erro quadrático médio, bem como os hiperparâmetros utilizados em cada execução, para poder comparar diferentes versões do modelo
- Empacotar modelos para implantar um modelo em produção, é necessário empacotá-lo junto com suas dependências para que ele possa ser executado em outro ambiente. O MLflow ajuda nesse processo de empacotamento.

Arquitetura da Solução

Bibliotecas

Streamlit:

- Criação de interfaces de usuário: Permite criar interfaces de usuário interativas para seus modelos de ML com apenas algumas linhas de código.
- Compartilhamento de modelos: Permite compartilhar seus modelos de ML com outras pessoas através de interfaces web.
- Coleta de feedback: Permite coletar feedback dos usuários sobre o desempenho do modelo e iterar no processo de desenvolvimento.

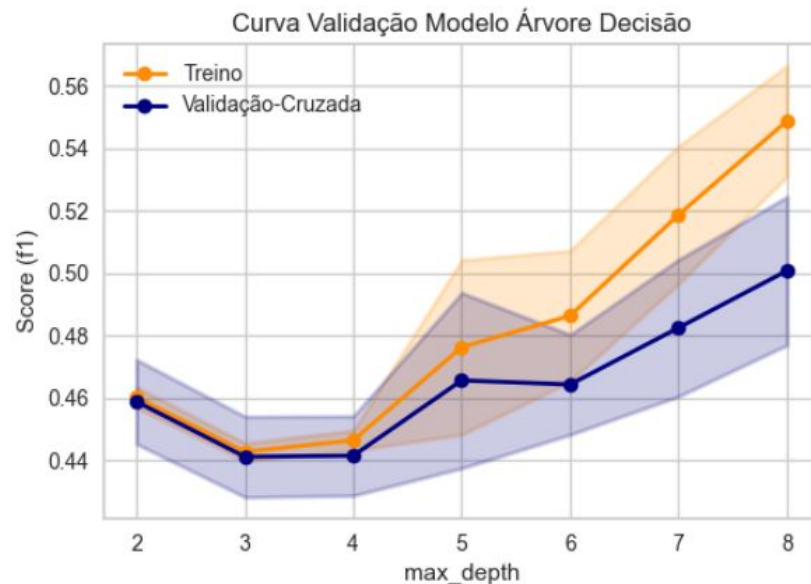
Essas ferramentas proporcionam:

- Rastreamento de Experimentos (PyCaret, MLflow)
- Treinamento e Avaliação do Modelo (PyCaret, MLflow)
- Monitoramento da Saúde do Modelo (MLflow)
- Atualização do Modelo (MLflow)
- Provisionamento (Deployment) (Streamlit)

Arquitetura da Solução

Artefatos e Métricas

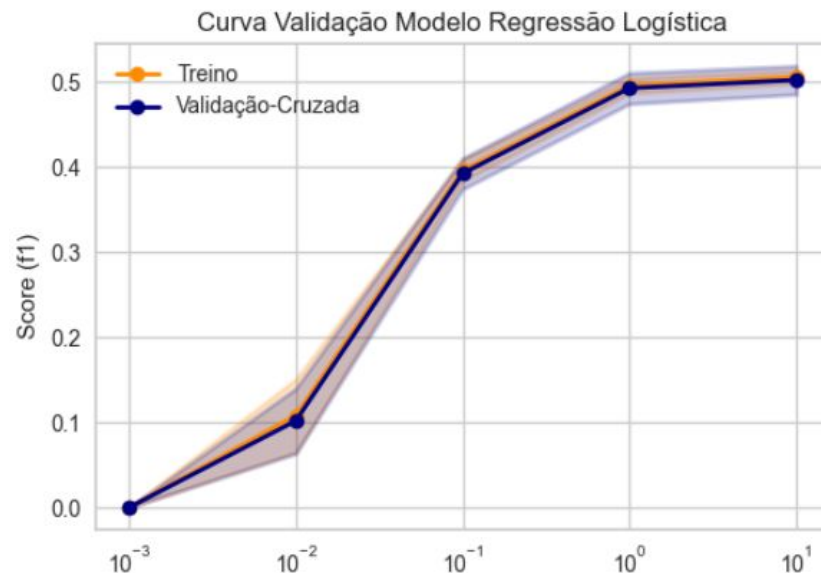
- **Curva de validação AD:** Válida a performance por meio de validação cruzada do treino e teste para árvore de decisão usando o score F1 por max_depth (profundidade da árvore).



Arquitetura da Solução

Artefatos e Métricas

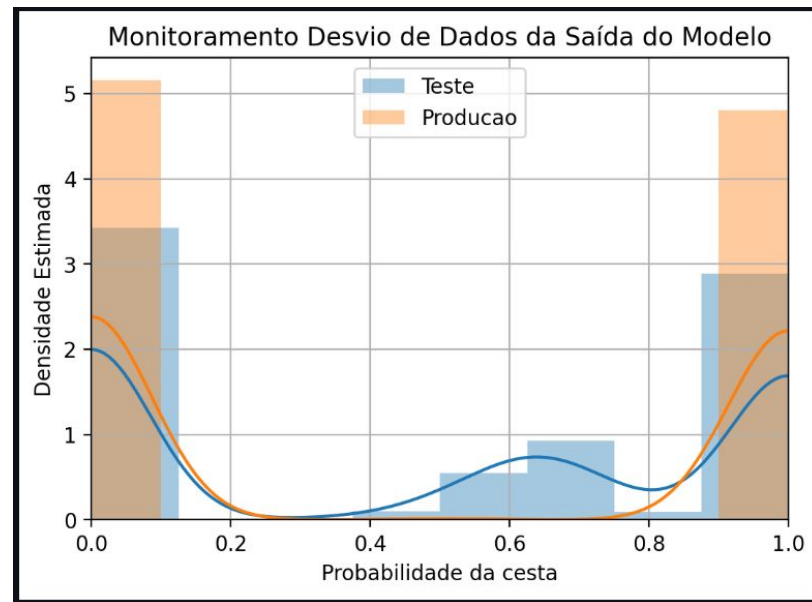
- **Curva de validação RL:** Válida a performance por meio de validação cruzada do treino e teste para regressão logística usando o score F1 por C.



Arquitetura da Solução

Artefatos e Métricas






- **Monitoramento produto final:** Comparação entre o treino em teste com a produção, usando streamlit.



Arquitetura da Solução

Artefatos e Métricas



- **base_test.parquet:** Base de teste da aplicação.
- **base_train.parquet:** Base de treino da aplicação.
- **data_filtered.parquet:** Base filtrada da base original.
- **predicton_prod:** Predição da aplicação na produção.
- **prediction_test:** Predição da aplicação pela base test.

 base_test.parquet	4/13/2024 9:10 PM	PARQUET File	45 KB
 base_train.parquet	4/13/2024 9:10 PM	PARQUET File	157 KB
 data_filtered.parquet	4/13/2024 9:10 PM	PARQUET File	203 KB
 prediction_prod.parquet	4/13/2024 9:14 PM	PARQUET File	194 KB
 prediction_test.parquet	4/13/2024 9:10 PM	PARQUET File	53 KB

Arquitetura da Solução

Artefatos e Métricas

- **dataset_kobe_dev.parquet:** Base de treino e teste para “treinar” nosso modelo para produção.
- **dataset_kobe_prod.parquet:** Base de produção o qual será usada para comparação na aplicação final.

 dataset_kobe_dev.parquet	4/13/2024 6:01 PM	PARQUET File	583 KB
 dataset_kobe_prod.parquet	4/13/2024 6:01 PM	PARQUET File	191 KB

Processamento de Dados

Pipeline de processamento dos dados

Descrição dos dados

- **dataset_kobe_dev.parquet:** Possui 24271 linhas e 25 colunas, possui 3986 valores NaN.

Tamanho do dataset original:

24271 rows × 25 columns

```
df_dev = pd.read_parquet("D:\\repositorios\\kobe-dataset\\data\\raw\\dataset_kobe_dev.parquet")  
df_dev.isnull().any(axis=1).sum()
```

3986

Pipeline de processamento dos dados

Descrição dos dados

- Colunas:

```
df_dev = pd.read_parquet("D:\\repositorios\\kobe-dataset\\data\\raw\\dataset_kobe_dev.parquet")
df_dev.columns
```

```
Index(['action_type', 'combined_shot_type', 'game_event_id', 'game_id', 'lat',
      'loc_x', 'loc_y', 'lon', 'minutes_remaining', 'period', 'playoffs',
      'season', 'seconds_remaining', 'shot_distance', 'shot_made_flag',
      'shot_type', 'shot_zone_area', 'shot_zone_basic', 'shot_zone_range',
      'team_id', 'team_name', 'game_date', 'matchup', 'opponent', 'shot_id'],
      dtype='object')
```

Pipeline de processamento dos dados

Descrição dos dados

- Colunas utilizadas:

```
df_dev.columns
```

```
Index(['lat', 'lon', 'minutes_remaining', 'period', 'playoffs',  
      'shot_distance', 'shot_made_flag'],  
      dtype='object')
```

Pipeline de processamento dos dados

Descrição dos dados

- Colunas info:

```
df_dev.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Index: 20285 entries, 1 to 30696
```

```
Data columns (total 7 columns):
```

#	Column	Non-Null Count	Dtype
0	lat	20285 non-null	float64
1	lon	20285 non-null	float64
2	minutes_remaining	20285 non-null	int64
3	period	20285 non-null	int64
4	playoffs	20285 non-null	int64
5	shot_distance	20285 non-null	int64
6	shot_made_flag	20285 non-null	float64

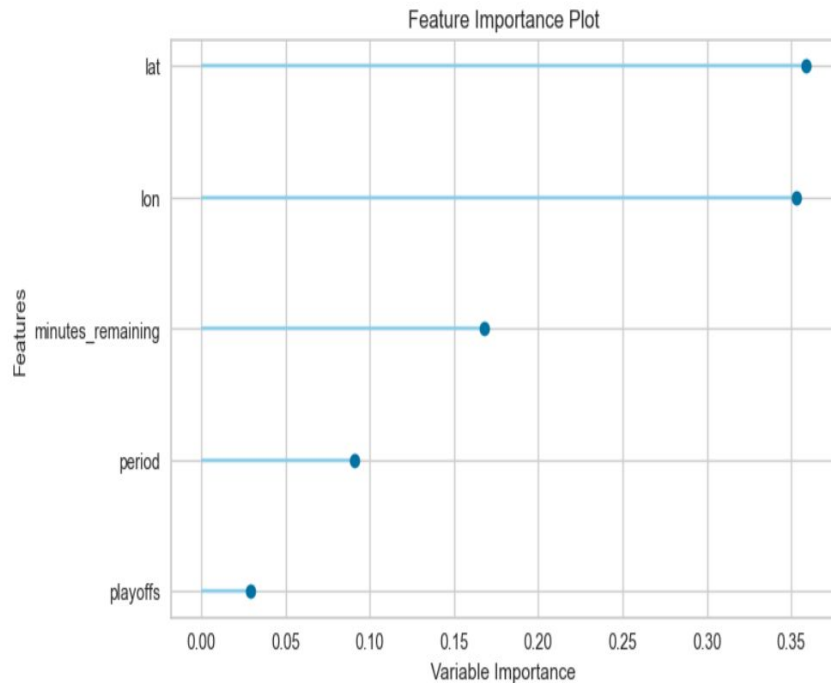
```
dtypes: float64(3), int64(4)
```

```
memory usage: 1.2 MB
```

Pipeline de processamento dos dados

Análise Exploratória

A latitude e a longitude tem maior influência que as demais, quase tem a mesma influência sobre o algoritmo.



Pipeline de processamento dos dados

Seleção base de teste

Distribuição foi 80% treino e 20% teste.

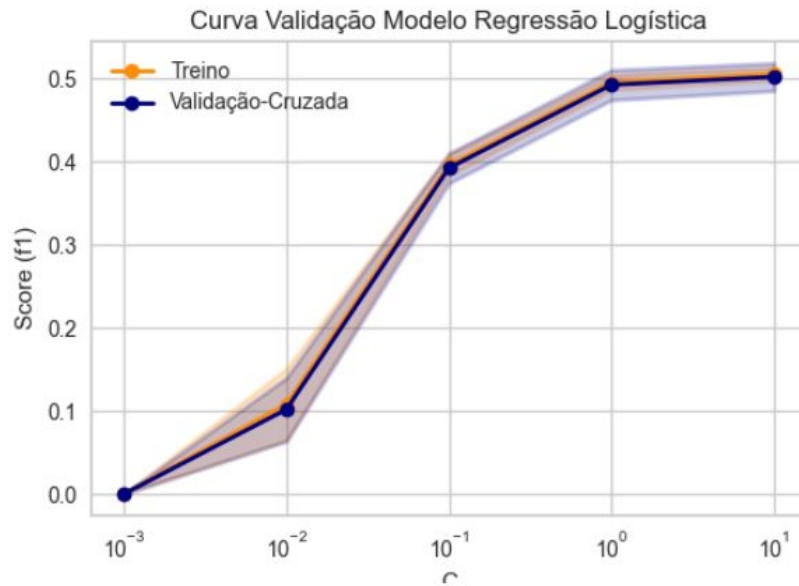
Treinamento do Modelo

Pipeline de Treinamento do Modelo

Regressão Logística - Validação Cruzada

Na regressão logística, o parâmetro C controla a regularização L2. Valores altos de C aumentam a regularização, penalizando coeficientes grandes e evitando overfitting. Valores baixos de C diminuem a regularização, permitindo que o modelo aprenda com mais flexibilidade.

Uma curva de validação ideal para o parâmetro C na regressão logística apresenta um mínimo em um valor específico de C . Esse valor indica o grau ideal de regularização para o modelo, onde ele encontra um equilíbrio entre aprendizado e generalização.



Pipeline de processamento dos dados

Regressão Logística - Classificação

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Logistic Regression	0.5588	0.5654	0.5023	0.5409	0.5209	0.1130	0.1133

Parameter: C

GridSearch: [0.001, 0.01, 0.1, 1, 10]

Scoring: f1

```
print('Log loss regressão logística:', log_loss(yhat_test.shot_distance, yhat_test.prediction_label))
```

✓ 0.0s

```
Log loss regressão logística: 16.6047789460833
```


Pipeline de Treinamento do Modelo

Árvore de Decisão - Validação Cruzada

A curva de aprendizado do parâmetro `max_depth` (profundidade máxima) na árvore de decisão fornece insights valiosos sobre como a complexidade do modelo afeta seu desempenho em termos de erro (geralmente taxa de erro de validação cruzada).

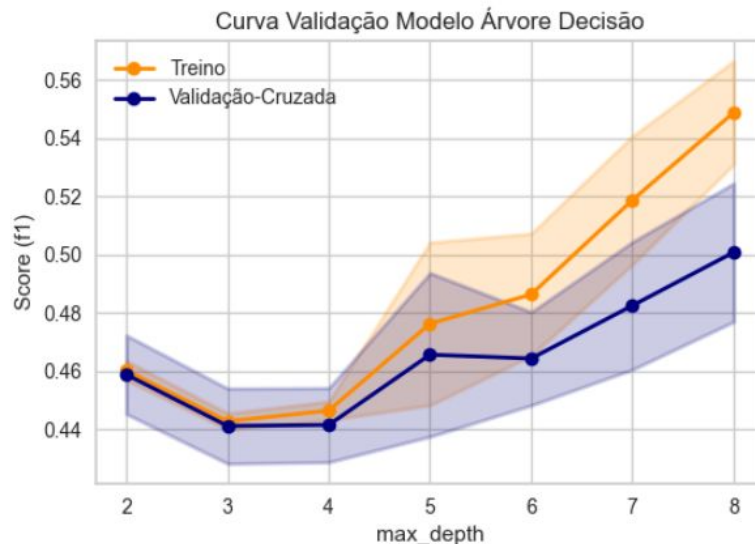
	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Decision Tree Classifier	0.5344	0.5142	0.5994	0.5106	0.5514	0.0738	0.0749

Parameter: `max_depth`

GridSearch: [2, 3, 4, 5, 6, 7]

Scoring: f1

Log loss árvore de decisão: 16.6047789460833

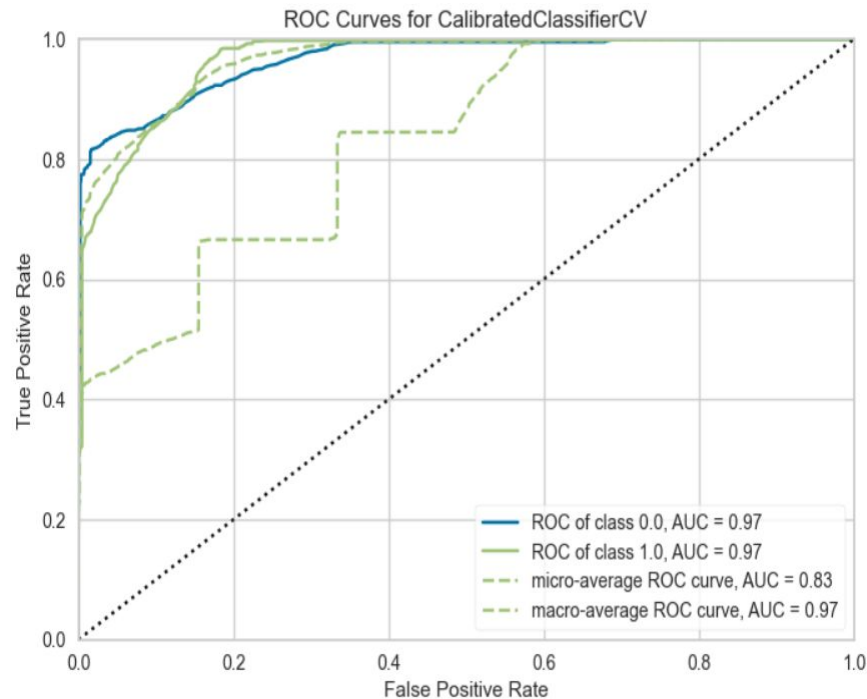


Pipeline de Treinamento do Modelo

Seleção, finalização e registro

Modelo foi o árvore de decisão, classificado como `list_model[0]`.

```
from pycaret.regression import *  
  
exp.evaluate_model(list_models[0])
```



Aplicação do Modelo

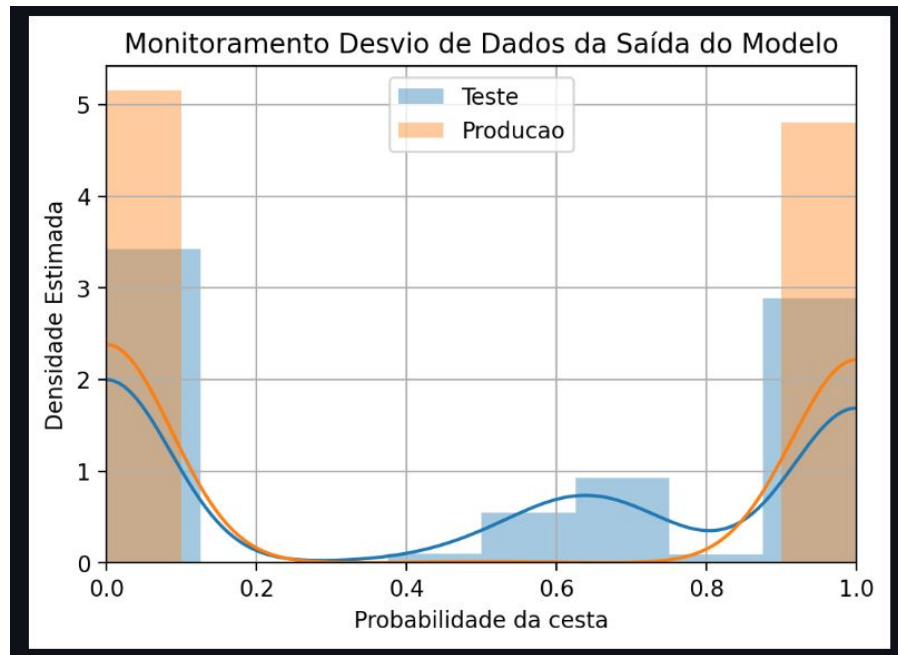
Pipeline de Aplicação do Modelo

Deployment

Modelo escolhido, árvore de decisão.

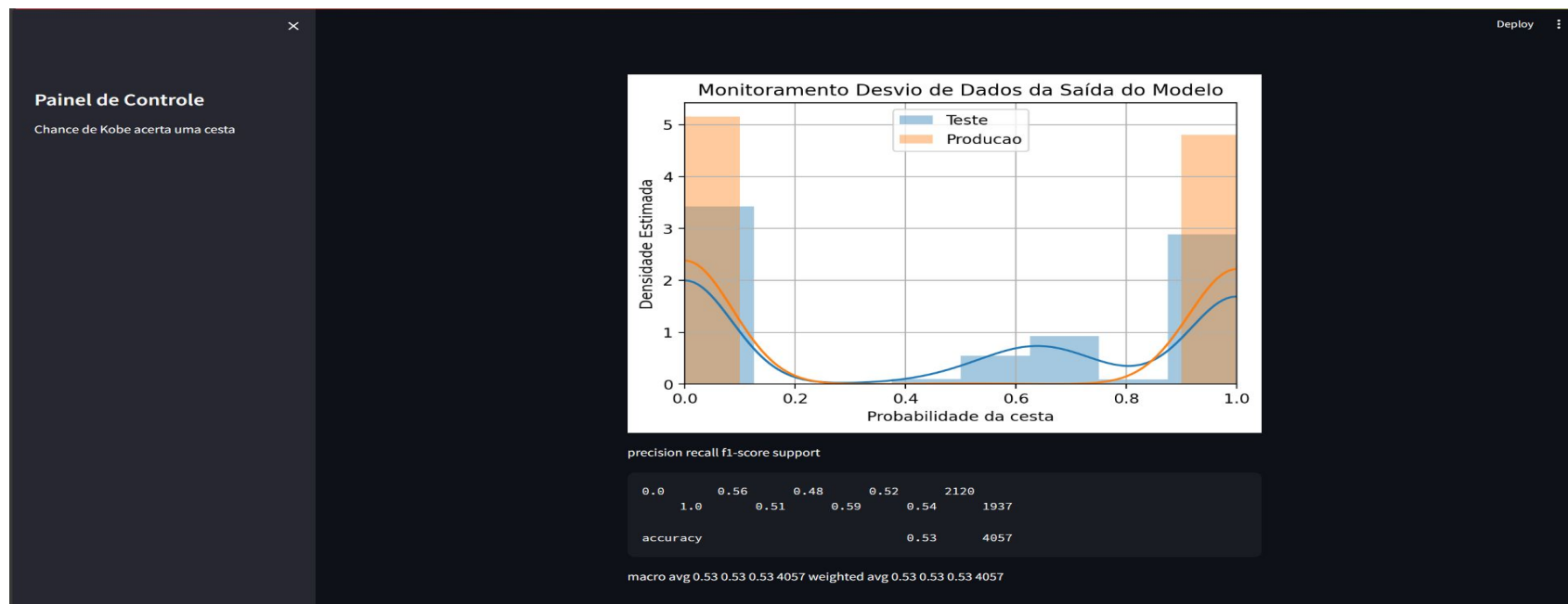
Métricas:

```
0.0    0.56    0.48    0.52    2120
      1.0    0.51    0.59    0.54    1937
accuracy          0.53    4057
```



Pipeline de Aplicação do Modelo

Interface Monitoramento



Pipeline de Aplicação do Modelo

Retreinamento

Estratégia reativa:

O modelo é re-treinado após a detecção de uma queda no desempenho. Essa detecção pode ser feita através de monitoramento de métricas (como acurácia, precisão, recall, etc.) ou através de alertas gerados por sistemas de monitoramento.

Estratégia preditiva:

O modelo é re-treinado proativamente, antes que uma queda no desempenho ocorra. Isso é feito através de modelos de previsão que estimam quando o desempenho do modelo atual vai diminuir.