

Publicação de Aplicação para Predição de Diagnóstico de Câncer

Bel Cogo e Bruno Hoffmann

2024/2 | Ciência de Dados & Big Data

Relembrando do Grau A...

- Modelo capaz de prever o diagnóstico de câncer de pacientes;
- Dados: 1500 pacientes;
- Algoritmo usado: Random Forest;
- Métricas:
 - Acurácia: 89.3%;
 - Precisão: 92.6%;
 - Recall: 78.9%;
 - F1-Score: 85.2%.

DEMO APP

App Predição de Câncer

☐ Exibir análise dos dados

Predição de Diagnóstico

Idade em Anos

24 - +

IMC

23,00 - +

Tempo de Atividade Física Semanal (em Horas)

2,00 - +

Quantidade de Alcool Consumido na Semana

0,00 - +

Possuí Risco Genético

- ☒ Baixo
☐ Médio
☐ Alto

Gênero

- ☐ Feminino
☒ Masculino

☐ É fumante?

☐ Possui Histórico de Câncer?

Predizer Diagnóstico

Resultado da Predição: Não Diagnosticado

Construção dos gráficos

Importando dados

```
# Carrega o dataset e manipula dados quantitativos em qualitativos para melhor visualização dos gráficos
dataset = pd.read_csv('./src/resources/The_Cancer_data_1500_V2.csv')
dataset_for_plots = dataset.copy()
dataset_for_plots['GeneticRisk'] = dataset_for_plots['GeneticRisk'].map({0: 'Baixo', 1: 'Médio', 2: 'Alto'})
dataset_for_plots['Gender'] = dataset_for_plots['Gender'].map({0: 'Masculino', 1: 'Feminino'})
dataset_for_plots['Smoking'] = dataset_for_plots['Smoking'].map({0: 'Não', 1: 'Sim'})
dataset_for_plots['CancerHistory'] = dataset_for_plots['CancerHistory'].map({0: 'Sem histórico', 1: 'Com histórico'})
```

Renderizando gráfico

```
st.pyplot(plot.get_figure())
```

```
st.pyplot(fig)
```

```
st.plotly_chart(graph)
```

Criando plot

```
plt.clf()
plot = sns.countplot(x=dataset_for_plots['Gender'])

plt.title('Distribuição por Gênero')
plt.ylabel('Contagem')
plt.xlabel('Gênero')
```


Como consumir o modelo?

Exportando modelo e *encoders*

```
import pickle

# Salvando modelo treinado em arquivo pkl
model_file_name = './src/resources/trained_random_forest.pkl'
pickle.dump(classifier, open(model_file_name, 'wb'))

# Salva encoders para usar na UI
scaler_encoder_file_name = './src/resources/scaler_encoder.pkl'
one_hot_encoder_file_name = './src/resources/one_hot_encoder.pkl'

pickle.dump(one_hot_encoder, open(one_hot_encoder_file_name, 'wb'))
pickle.dump(scaler, open(scaler_encoder_file_name, 'wb'))
```

Importando modelo e *encoders*

```
# Importação do modelo e dos encoders usados no treinamento
model = pickle.load(open('./src/resources/trained_random_forest.pkl', 'rb'))
one_hot_encoder = pickle.load(open('./src/resources/one_hot_encoder.pkl', 'rb'))
scaler_encoder = pickle.load(open('./src/resources/scaler_encoder.pkl', 'rb'))
```

Chamando modelo para prever

```
# Monta registro do paciente
patient = {
    'Age': age,
    'Gender': int(gender == "Feminino"),
    'BMI': bmi,
    'Smoking': int(smoking),
    'GeneticRisk': get_formated_genetic_risk(genetic_risk),
    'PhysicalActivity': physical_activity,
    'AlcoholIntake': alcohol_intake,
    'CancerHistory': int(cancer_history)
}

# Coloca o registro em dataframe
df = pd.DataFrame([patient])
values = df.values

# Aplica encoders em cima do registro
hot_encoded_values = one_hot_encoder.transform(values)
scaler_encoded_values = scaler_encoder.transform(hot_encoded_values)

# Realiza a predição em cima dos dados.
results = model.predict(scaler_encoded_values)

# Retorna para usuário o resultado.
result_to_show = ":red[Diagnosticado]" if results[0] else ":green[Não Diagnosticado]"
st.subheader("Resultado da Predição: " + result_to_show)
```

```
# Método para obter o valor numérico de GeneticRisk
def get_formated_genetic_risk(risk):
    map = {
        'Baixo': 0,
        'Médio': 1,
        'Alto': 2
    }

    return map[risk]
```


Configuração AWS

The image shows two screenshots of the AWS Academy environment. The top screenshot is a lab interface with a sidebar on the left containing links like 'Página inicial', 'Módulos', 'Fóruns', and 'Lucid (Whiteboard)'. The main area shows a terminal with the command 'eee_W_3777661@runweb146627:~\$' and a terminal window titled 'Access the AWS Management Console' with the instruction '1. At the top of these instructions, choose'. The top right of the lab interface shows 'Used \$0 of \$50', a timer '03:51', and buttons for 'Start Lab', 'End Lab', 'AWS Details', 'Readme', 'Reset', and a close button. A red arrow points to the 'Start Lab' button. The bottom screenshot is the AWS Management Console 'Página inicial do console'. It features a 'Visitado recentemente' section with links to EC2, S3 Glacier, Lambda, S3, Elastic Kubernetes Service, Elastic Container Registry, Elastic Container Service, and AWS Artifact. There is also a 'Gerenciamento de cobrança e custos' link. The 'Aplicações (0)' section shows the current region as 'us-east-1 (Região atual)' and a search bar for 'Localizar aplicações'. It displays a message 'Nenhuma aplicação. Comece criando um aplicativo.' with a 'Criar aplicação' button. The top of the console shows the AWS logo, a search bar, and the current region 'Norte da Virgínia'.

ALLv2PT-... > Módulos > Laboratóri...
> Iniciar os laboratórios de aprendizagem da AWS Academy

Página inicial
Módulos
Fóruns
Lucid (Whiteboard)

AWS

Used \$0 of \$50 03:51 ▶ Start Lab ■ End Lab ⓘ AWS Details ⓘ Readme ↺ Reset ✕

eee_W_3777661@runweb146627:~\$

Access the AWS Management Console

1. At the top of these instructions, choose

aws Search [Option+S]

Norte da Virgínia ▼ voclabs/user1129270=Bruno_da_Siqueira_Hoffmann @ 6546-5447-9741 ▼

Página inicial do console Informações

Redefinir para layout padrão + Adicionar widgets

Visitado recentemente Informações

- EC2
- S3 Glacier
- Lambda
- S3
- Elastic Kubernetes Service
- Elastic Container Registry
- Elastic Container Service
- AWS Artifact

DynamoDB

Gerenciamento de cobrança e custos

Ver todos os serviços

Aplicações (0) Informações

Região: US East (N. Virginia)

us-east-1 (Região atual) Localizar aplicações

< 1 >

Nome | Descrição | Região | Conta de orig...

Nenhuma aplicação
Comece criando um aplicativo.

Criar aplicação

Vá para myApplications

Criação da Instância EC2

Painel X

Visualização Global do EC2

Eventos

▼ **Instâncias**

- Instâncias**
- Tipos de instância
- Modelos de execução
- Solicitações spot
- Savings Plans
- Instâncias reservadas
- Hosts dedicados
- Reservas de capacidade

▼ **Imagens**

- AMIs
- Catálogo de AMIs

Instâncias Informações

Última atualização less than a minute atrás

Conectar

Estado da instância ▼

Ações ▼

Executar instâncias ▼

Localizar instância por atributo ou tag (case-sensitive)

Todos os es... ▼

| | Name | ID da instância | Estado da inst... | Tipo de inst... | Verificação de stal | Status do alarm | Zona de dispon... | DNS IPv4 público | Endereço IP... |
|---|------|-----------------|-------------------|-----------------|---------------------|-----------------|-------------------|------------------|----------------|
| Nenhuma instância Você não tem nenhuma instância nessa região | | | | | | | | | |

Executar instâncias

Selecionar uma instância

Configuração de Grupos de Segurança

EC2 > Grupos de segurança > sg-007cff332b53bd0ff

sg-007cff332b53bd0ff - launch-wizard-1

Ações ▾

Detalhes

Nome do grupo de segurança

launch-wizard-1

ID do grupo de segurança

sg-007cff332b53bd0ff

Descrição

launch-wizard-1 created 2024-11-19T00:10:27.159Z

ID da VPC

vpc-09e816c9c3a638ba2

Proprietário

654654479741

Número de regras de entrada

2 Entradas de permissão

Número de regras de saída

1 Entrada de permissão

Regras de entrada

Regras de saída

Compartilhamento - novo

Associações da VPC - novo

Tags

Regras de entrada (2)

↻

Gerenciar tags

Editar regras de entrada

🔍 Pesquisar

< 1 > ⚙️

| <input type="checkbox"/> | Name ▾ | ID da regra do grup... ▾ | Versão do IP ▾ | Tipo ▾ | Protocolo ▾ | Intervalo de portas ▾ | Origem ▾ |
|--------------------------|--------|--------------------------|----------------|-------------------|-------------|-----------------------|-----------|
| <input type="checkbox"/> | - | sgr-0c5edd04342e7f2f1 | IPv4 | SSH | TCP | 22 | 0.0.0.0/0 |
| <input type="checkbox"/> | - | sgr-0c216fbb98031643a | IPv4 | TCP personalizado | TCP | 8501 | 0.0.0.0/0 |

[EC2](#) > [Instâncias](#) > [i-0bdb2bcd859ae6331](#) > Conectar-se à instância

Conectar-se à instância [Informações](#)

Conecte-se à sua instância i-0bdb2bcd859ae6331 (trabalho_gb_bel_bruno) usando qualquer uma destas opções

Conexão de instância do EC2

Gerenciador de sessões

Cliente SSH

Console de série do EC2

ID da instância

 i-0bdb2bcd859ae6331 (trabalho_gb_bel_bruno)

Tipo de conexão



Conectar-se usando o EC2 Instance Connect


Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.



Conectar-se usando o endpoint do EC2 Instance Connect

Conecte-se usando o cliente baseado em navegador do EC2 Instance Connect, com um endereço IPv4 privado e um endpoint da VPC.

☒ Endereço IPv4 público

 54.81.66.6

☐ Endereço IPv6

—

Nome de usuário

Insira o nome de usuário definido na AMI usada para iniciar a instância. Se você não definiu um nome de usuário personalizado, use o nome de usuário padrão, ubuntu.

 ubuntu



Observação: na maioria dos casos, o nome de usuário padrão, ubuntu, está correto. No entanto, leia as instruções de uso da AMI para verificar se o proprietário da AMI alterou o nome de usuário da AMI padrão.

Cancelar

Conectar

Conexão ambiente EC2

Configuração ambiente EC2

Configuração de pacotes do python

- `sudo apt update`
- `sudo apt-get update`
- `sudo apt upgrade -y`
- `sudo apt install python3-pip`
- `sudo apt install python3.12-venv`

Download do repositório git

- `git clone https://github.com/BrunoHoffmann15/trab-1-big-data-gb.git`

Acessar pastas

- `cd trab-1-big-data-gb`

Configurar venv

- `python3 -m venv venv`
- `source venv/bin/activate`

Download das dependências da aplicação

- `pip install -r src/requirements.txt`

Executar a aplicação

- `streamlit run src/app.py`

Obrigada!

Dúvidas?