

Predição de Diabetes usando Flask e Random Forest Classifier– Projeto da Disciplina de Tópicos em Software

- Dataset utilizado: load_diabetes da biblioteca do Scikit-learn.

Link da documentação desse dataset:

https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_diabetes.html#examples-using-sklearn-datasets-load-diabetes

- Atributos da Tabela

Atributo	Significado
Pregnancies	Número de vezes grávida
Glucose	Concentração de glicose no plasma após 2 horas de teste oral de tolerância à glicose
BloodPressure	Pressão arterial diastólica em mm Hg
SkinThickness	Espessura da dobra da pele do tríceps em mm
Insulin	Insulina sérica de 2 horas em U/ml
BMI	Índice de massa corporal
DiabetesPedigreeFunction	Função de pedigree de diabetes
Age	Idade
Outcome	Resultado binário indicando presença ou ausência de diabetes

Esse código é um exemplo de uma aplicação web simples que utiliza um modelo de Machine Learning para fazer previsões sobre a ocorrência de diabetes com base em dados fornecidos pelo usuário. A aplicação é construída usando o framework Flask para o backend e HTML com Bootstrap para o frontend.

Importação de bibliotecas para o funcionamento da API

- Flask: Framework web em Python.
- Request: permite acessar dados enviados com a solicitação HTTP
- jsonify: converte objetos Python em JSON.
- CORS: permite requisições entre domínios diferentes
- RandomForestClassifier do Scikit-Learn: implementa um classificador de floresta aleatória.
- Joblib(Recomendado pelo professor): usado para carregar o modelo treinado
- Numpy: Biblioteca para manipulação de arrays.

Observações Gerais

- O modelo treinado ('diabetes_model.joblib') deve estar presente no mesmo diretório que o script Flask
- A página HTML usa o jQuery e o Popper.js para o Bootstrap.
- Certifique-se de ter todas as dependências instaladas usando `pip install Flask, flask-cors, scikit-learn, joblib e numpy`
- Foi alocado também um arquivo .csv, uma tabela de um dataset parecido do Kaggle, que utilizamos os dados dela apenas para fazer testes.
- **Importante: é necessário rodar o arquivo `train_model.py` primeiro pra que ele crie o arquivo `'diabetes_model.joblib'`, que é o modelo treinado.**

Execução do Código

- Execute o script Python (`app.py`) para iniciar o servidor Flask.
- Abra o arquivo HTML em um navegador
- Insira os dados no formato especificado e clique no botão "Predict"
- A previsão ou mensagens de erro serão exibidas na página.

Sobre o Dataset

O dataset utilizado tem como resposta (Prediction) números inteiros que variam entre 25 e 346, assim como diz a documentação. Não está muito claro, mas segundo alguns testes feitos, os números mais comuns de aparecer são 84 e 346, acreditamos pelo número de vezes que aparecem, que 84 seja não diabetes e 346 seja diabetes, já que são mais incomuns de aparecer. Optamos por deixar apenas um campo para preenchimento para o usuário, porque fica mais fácil e mais rápido para fazer os testes. O sistema utilizado (código) foi baseado nos exemplos disponibilizados pelo professor.