

Specyfikacja: trening i ewaluacja modelu

Cel

Sekwencja skryptów Python, których zadaniem będzie wczytanie danych treningowych, stworzenie modelu i jego ewaluacja na danych testowych.

Lista kontrolna

Skrypt, który stworzymy, będzie realizował następujące zadania:

1. Wczytanie danych

- ☐ Wczytuje plik treningowy data_init.csv
- ☐ Zapisuje go do pliku data_train.csv

2. Trenowanie modelu

- ☐ Wczytuje dane treningowe data_train.csv
- ☐ Przygotowuje dane do modelowania
- ☐ Trenuje model korzystając algorytmu LinearRegression
- ☐ Drukuje na ekranie parametry modelu
- ☐ Zapisuje model do pliku model_1.0.pkl w folderze model

3. Ewaluacja modelu

- ☐ Wczytuje model z pliku model/model_1.0.pkl
- ☐ Wczytuje dane treningowe z pliku data/data_test.csv
- ☐ Generuje predykcje i ocenia model
- ☐ Drukuje wyniki ewaluacji na ekranie komputera

Architektura

Artefakty

1. Wejście

1. Inicjalny plik treningowy: data_init.csv
2. Plik testowy: data_test.csv

2. Wyjście

1. model: model_1.0

Komponenty

1. Wczytanie danych (1. read.py)

1. Wejście: data_init.csv
2. Działania:
 1. wczytuje dane
 2. zapisuje na dysku
3. Wyjście: data_train.csv

2. Trening (2. train.py):

1. Wejście: data_train.csv
2. Działania:
 1. Wczytuje dane treningowe data_train.csv
 2. Przygotowuje dane do modelowania
 3. Trenuje model korzystając algorytmu LinearRegression
 4. Drukuje na ekranie parametry modelu
 5. Zapisuje model do pliku model_1.0.pkl w folderze model
3. Wyjście: model_1.0.pkl

3. Ewaluacja (3. test.py):

1. Wejście:
 1. data_test.csv
 2. model_1.0.pkl
2. Działania:
 1. Wczytuje model z pliku model/model_1.0.pkl
 2. Wczytuje dane treningowe z pliku data/data_test.csv
 3. Generuje predykcje i ocenia model
 4. Drukuje wyniki ewaluacji na ekranie komputera
3. Wyjście: ekran

Decyzje

Projektując to rozwiązanie, musimy podjąć następujące decyzje:

1. Wybór miary jakości modelu
 1. RMSE
 2. R2
2. Algorytm(y) ML: LinearRegression

Podsumowanie

Artefakty				
	wejście	opis	wyjście	opis
	data_init.csv	inicjalny plik treningowy	model_1.0	wytrenowany model
	data_test.csv	plik testowy		
Komponenty				
	wejście	specyfikacja	wyjście	
1. read.py	data_init.csv	Wczytuje plik treningowy data_init.csv	data_train.csv	
		Zapisuje go do pliku data_train.csv		
2. train.py	data_train.csv	Wczytuje dane treningowe data_train.csv	model_1.0.pkl	
		Przygotowuje dane do modelowania		
		Trenuje model korzystając algorytmu LinearRegression		
		Drukuje na ekranie parametry modelu		
		Zapisuje model do pliku model_1.0.pkl w folderze model		
3. test.py	data_test.csv	Wczytuje model z pliku model/model_1.0.pkl	ekran	
	model_1.0.pkl	Wczytuje dane treningowe z pliku data/data_test.csv		
		Generuje predykcje i ocenia model		
		Drukuje wyniki ewaluacji na ekranie komputera		