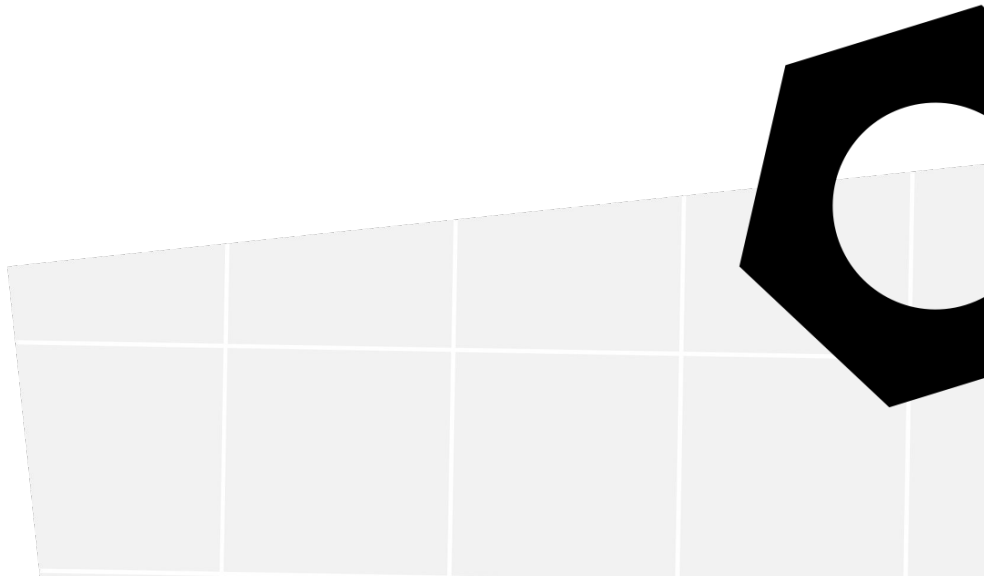
Projekt praktyczny (regresja)

Kurs Data Science

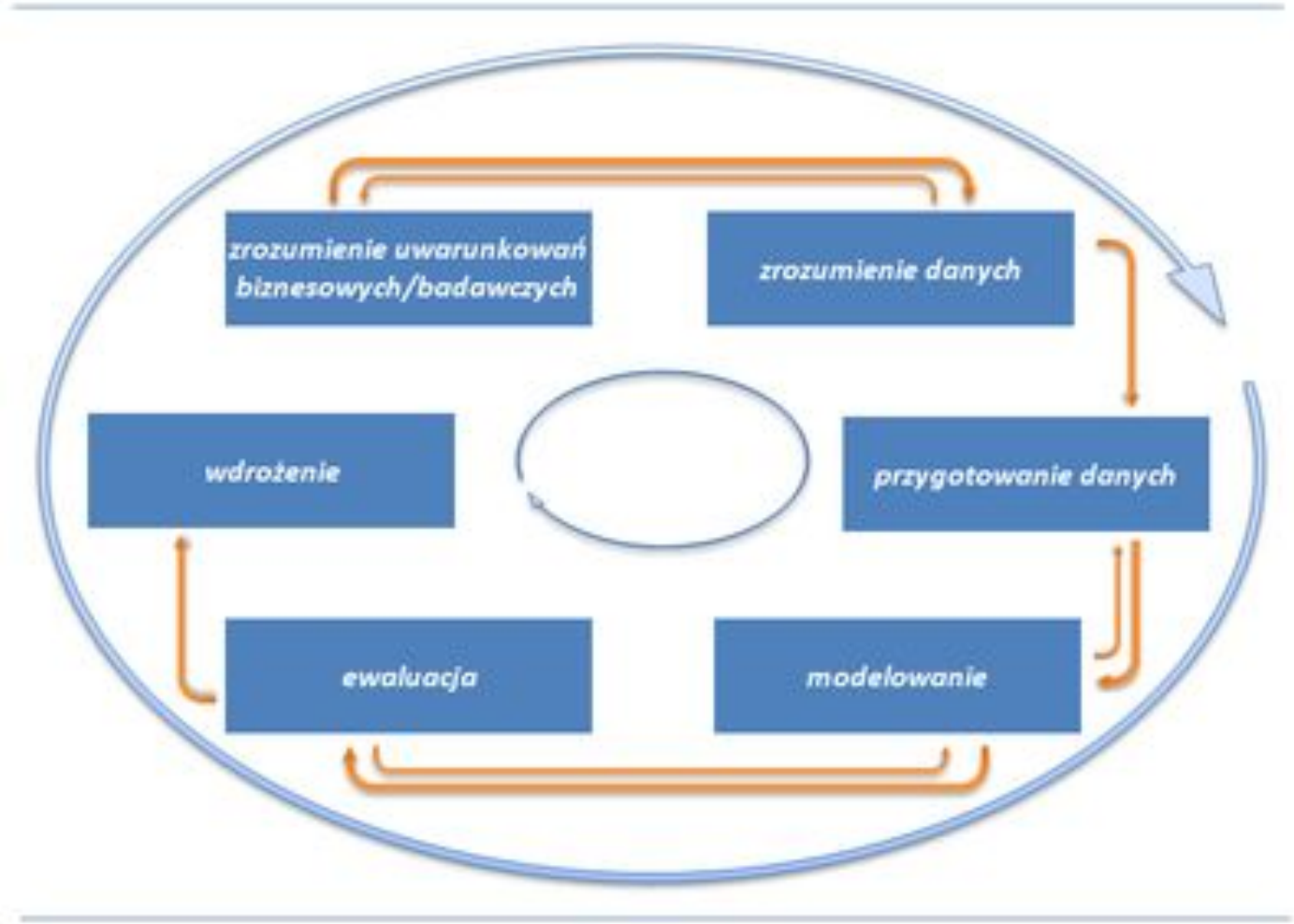


software
development
academy

UWAGI

Projekt praktyczny jest podsumowaniem dotychczasowych zajęć poprzez zebranie wymagań, analizę danych i implementację kodu źródłowego.

Cykl życia i budowy algorytmu machine learning – metodologia CRISP (ang. Cross Industry Standard Process)



PLAN DNIA

- Podział na grupy projektowe (max. 2-3 osoby)
- Zrozumienie i analiza problemu, który prezentują dane
- Wstępne zapoznanie się z przedstawioną bazą danych
- Podsumowanie kroku 2 i 3 oraz dokładne opisanie i przedstawienie hipotez, które będą rozwiązywane
- Analiza, wizualizacja danych oraz przeprowadzenie analizy statystycznej w języku Python
- Użycie i przetestowanie różnych algorytmów regresji do analizy problemów liniowych i nieliniowych
- Ocena działania algorytmu
- Wyjaśnienie dlaczego użyty został akurat ten algorytm
- Propozycja jego użycia

Wymagania do projektu praktycznego z regresji

1. Projekt powinien być dostarczony jako prezentacja Powerpoint z wynikami (można to samo zrobić w Jupyter Notebook dla chętnych) oraz kody języka Python jako Jupyter Notebook.
2. Prace powinny być rozwiązywane na maszynie lokalnej (komputer/laptop), na początku trzeba zainstalować niezbędne pakiety wykorzystując polecenie pip
3. Kod powinien być dzielony pomiędzy członkami zespołu przy użyciu zdalnego repozytorium git, np. na platformie Github

NARZĘDZIA

Podczas zajęć istnieje możliwość korzystania z poniższych narzędzi pomocniczych:

- Google Colaboratory (https://colab.research.google.com/)
- Github (https://github.com)
- Sklearn (https://scikit-learn.org/stable/)
- Scipy (https://www.scipy.org/)

OPIS SZCZEGÓŁOWY

software
development
academy

1. Podział na grupy projektowe

- Grupy 2 lub 3 osobowe
- Należy dobrać się samodzielnie (w przypadku problemów z dobraniem się w zespołu trener dokona podziału losowego)

2. Zrozumienie i analiza problemu, który prezentują dane

- Zapoznanie się z wybranym zestawem danych i opisem zadania do wykonania
- Pobranie bazy danych na swój komputer
- Zdefiniowanie co wiemy na dany temat
- Efektem tego punktu powinien być krótki 1-2 stronicowy wstęp teoretyczny analizujący problem od strony biznesowej/merytorycznej

3. Wstępne zapoznanie się z przedstawioną bazą danych

- Zapoznanie się z pobraną bazą danych (podłączenie się do niej przez Pythona z użyciem znanych pakietów pandas, numpy)
- Przygotowanie danych, a w tym:
  - Podział na cechy numeryczne, kategoryczne
  - Przygotowanie danych (pozbycie się lub transformacja wartości niekompletnych)
  - Agregacja niezbędnych informacji
  - Czyszczenie danych
  - Transformacja, dyskretyzacja, skalowanie, grupowanie itp. – użyj to, czego dotyczy Twoje zagadnienie
- Wyświetlenie podstawowych statystyk (średnia, odchylenie standardowe, mediana) przydatnych do analizy problemu

4. Podsumowanie kroku 2. i 3. oraz dokładne opisanie i przedstawienie hipotez, które będą rozwiązywane

- Zebranie wyników z punktu 3. i wiedzy z punktu 2.
- Opis tego, co zamierzamy zrobić wykorzystując zdobyte informacje
- Opis czego nie możemy zrobić na danej bazie danych z uwagi np. na brak informacji, zbyt dużo wartości odstających
- Podanie kilku algorytmów/modeli uczenia maszynowego rozwiązujących problemy regresji, który będziemy chcieli użyć w tym projekcie

5. Analiza, wizualizacja danych oraz przeprowadzenie analizy statystycznej w języku Python

Skoro już wiemy co będziemy analizować musimy dokładniej opisać dane za pomocą:

- Wykresów:

✔ Wykresy liniowe, histogramy opisujące zależności między pojedynczymi zmiennymi

✔ Wykresy opisujące zależności między zmiennymi (Q-Q plot, wykres punktów, mapy ciepła, wykresy słupkowe)

✔ Wykresy zmienności w czasie (wykres liniowy)

✔ Zaawansowana wizualizacja z wykorzystaniem plotly lub seaborn

5. Analiza, wizualizacja danych oraz przeprowadzenie analizy statystycznej w języku Python

- Statystyki matematycznej:
  - Współczynniki korelacji
  - Wariancja, kowariancja, odchylenie standardowe
  - Współczynnik zmienności
  - Pozostałe współczynniki statystyczne określające z jakim rozkładem statystycznym mamy do czynienia
  - Jeżeli będzie bardziej pomocne to macierz korelacji

6. Użycie i przetestowanie różnych algorytmów regresji do analizy problemów liniowych i nieliniowych

- Wykorzystując wszystkie informacje dotyczące danego problemu, które zgromadziliśmy, a szczególnie opis danych z poprzedniego punktu, zaproponować 3 algorytmy które będą implementowane
- Implementacja algorytmów regresji liniowej (1 i wielu zmiennych), wielomianowej oraz drzewa decyzyjnego
- Zadbanie o minimalizację funkcji kosztu, jeżeli trzeba to z użyciem gradient descent lub innej wbudowanej funkcji w pakiecie sklearn
- Wyświetlenie wartości funkcji kosztu na wykresie dla 10 i 20 iteracji oraz wag algorytmu
- Predykcja algorytmu dla 3 losowych wartości spoza zbioru danych

7. Ocena działania algorytmu

- Podział danych na zbiór treningowy i walidacyjny
- Zastosowanie walidacji krzyżowej i leave-one-out (przetasować zbiór tak, aby zawsze zaczynać od innego miejsca w danych)
- Ocenić skuteczność modeli (współczynnik determinacji, 2 metody błędu prognozy)
- Rozwiązanie problemu z przetrenowaniem lub niedotrenowaniem algorytmu
- Znalezienie złotego środka między obciążeniem a wariancją, najlepiej pokazując to na wykresie

7. Wyjaśnienie dlaczego użyty został akurat ten algorytm

- Wyciągając wnioski z poprzednich punktów opowiedzieć maksymalnie na 1-2 slajdach dlaczego użyliśmy akurat tego modelu, mając na uwadze:

- problemy liniowe i nieliniowe
- wpływ statystyk takich jak korelacja na zmienne
- wartości odstające
- wagi algorytmu
- ocena skuteczności jego działania
- zdolności predykcyjne

8. Propozycja jego użycia

- Maksymalnie na 1-2 slajdach odpowiedzieć na pytania:

Jaki problem jesteśmy w stanie rozwiązać używając naszego algorytmu?

Jak możemy go użyć? Z użyciem jakich narzędzi przekazywać komuś do użycia?

Jak interpretować wyniki algorytmu?

Jak interpretować wyniki skuteczności algorytmu?

Jak możemy go ulepszyć w przyszłości?

PODSUMOWANIE

1. Określenie trudności podczas pracy w zespole wirtualnym.
2. Określenie, która część sprawiła najwięcej problemów.
3. Wskazanie tematów, które wymagają uzupełnienia lub powtórzenia zagadnień.
4. Czy sposób przeprowadzenia projektu rozwija uczestników i sprawia, że lepiej rozumieją problem, który poznawali na zajęciach?
5. Czy sposób przeprowadzenia projektu może być pomocny przy przyszłych zagadnieniach?