Wydział Matematyki i Nauk Informacyjnych Politechnika Warszawska

Wstęp do uczenia maszynowego (projekt grupowy)

Sprawozdanie z projektu

Autorzy:
Bartek Eljasiak, Mateusz Grzyb, Mariusz Słapek
Prowadzący:
Michał Turski

Poniższy raport zawiera informację, co zostało przez zrobione, podczas trzech etapów projektu, oraz jakie są tego wnioski.

Ponieważ, wszystkie wyniki są w plikach .ipynb w tym dokumencie zawrzemy najbardziej istotne, naszym zdaniem, wnioski.

1. Zbiór danych

Nazwa: Online Shoppers Purchasing Intention Dataset Autorzy: C. Okan Sakar, Yomi Kastro Opis: The dataset consists of feature vectors belonging to 12,330 sessions. The dataset was formed so that each session would belong to a different user in a 1-year period to avoid any tendency to a specific campaign, special day, user profile, or period. Źródło: https://archive.ics.uci.edu/ml/

1.1. Słownik pojęć

Poniżej znajduje się opis zmiennych zbioru danych (będziemy potem używać w sprawozdaniu):

— Zmienne numeryczne

- 1. Administrative number of different pages visited related to the administrative concerns of the website
- 2. Informational number of different pages visited related to the information of the website and other useful contents of the website
- 3. ProductRelated number of different pages visited related to different products of the website.
- 4. BounceRate Percentage of users who left the website from the landing page
- 5. ExitRate Percentage of users who left from the page the visit
- 6. Page Values Page Value is the average value for a page that a user visited before making a transaction.
- 7. SpecialDay 0 day of the session is not within 10 days of a special day. Between 0.1 and 0.5 day of the session is between 10 days and 5 days away from a special day. Between 0.6 and 0.9 day of the session is between 4 days and 1 day away from a special day. 1 day of the session is a special day.

— Zmienne kategoryczne

- 1. Administrative_Duration time spent on Administrative pages in seconds
- 2. Informational_duration time spent on Informational pages in seconds
- 3. ProductRelated_Duration time spent on pages related to products in seconds
- 4. Browse ID of bowsers from which the session took place
- 5. Region ID of Regions from which the session took place
- 6. Traffic Type ID of different types of sources from which the users landed on the website
- 7. User Type whether the user is a returning user or a new user or of any other type
- 8. Revenue whether the user contributed to the revenue by purchasing or not
- 9. Weekend whether the session was on a weekend or not

2. Etap 1

Na pierwszy etap składa się głównie eksploracyjna analiza danych (EDA).

2.1. Wykonane prace

- zostały zbadane typy zmiennych i braki danych
- została przeprowadzona automatyczna analiza danych (przy pomocy wykorzystania pakietu pandas_profiling.

- przeprowadzona została analiza zmiennych:
 - 1. rozkłady zmiennych kategorycznych
 - 2. rozkłady zmiennych numerycznych
 - 3. statystyki pozycyjne i rozproszenia zmiennych numerycznych
- zbadane została korelacja zmiennych znajdujących się w zbiorze danych
- przeprowadzona została analiza wielowymiarowa
 - 1. Czy użytkownicy średnio odwiedzają więcej stron i spędzają na nich więcej czasu w trakcie weekendu? Jak weekend wpływa na odsetek sesji zakończonych przychodem?
 - 2. Jak bliskość dnia specjalnego wpływa na średnią ilość stron danego rodzaju odwiedzanych przez użytkownika? Jak wpływa na odsetek sesji zakończonych przychodem?
 - 3. W jakich miesiącach odsetek sesji zakończonych przychodem jest największy?

4.

2.2. Najważniejsze wnioski

- Dane obejmują 12330 sesji. Zmienne są typów bool, int, float i str. Nie występują braki danych.
- w danych występuje 125 identycznych wierszy,
- w kolumnach Administrative i Administrative Duration zera statnowią 47% danych,
- w kolumnach Informational i Informational_Duration zera stanowia 79% danych.
- w danych brakuje sesji z miesięcy styczeń i kwiecień,
- w każdym z rozkładów zmiennych 'OperatingSystems', 'Browser', 'Region' i 'TrafficType' występuje zdecydowany lider pod względem liczności,
- większość odwiedzających to powracający odwiedzający,
- większość sesji odbyła się w dzień roboczy, ale nie brakuje informacji o sesjach odbytych podczas weekendu (dane dość wyrównane biorąc pod uwagę liczbę dni roboczych oraz liczbę dni podczas weekendu)
- wszystkie rozkłady wykazują silną dodatnią asymetrię,
- użytkownicy dużo częściej odwiedzają strony związane z produktem, niż strony administracyjne i informacyjne (i spędzają na nich średnio ponad 10 razy więcej czasu),
- ilość stron danego typu, jaką odwiedził użytkownik, jest dodatnio skorelowana z ilością czasu, jaki na nich spędził (wartości 0.6-0.86),
- większość sesji nie przyniosła dochodu
- użytkownicy średnio odwiedzają nieco więcej stron każdego rodzaju i spędzają na nich nieco więcej czasu w trakcie weekendu
- w trakcie weekendu odsetek sesji zakończonych dochodem jest nieco większy
- w pobliżu dni specjalnych użytkownicy odwiedzają średnio mniej stron informacyjnych i administracyjnych oraz średnio nieco więcej stron związanych z produktami, w czasie sesji, w porównaniu do dni zwyczajnych
- trzy najlepsze pod tym względem miesiące to listopad, październik i wrzesień. Luty mocno odstaje od reszty obserwacji, jako najgorszy miesiąc pod tym względem

3. Etap 2

Na drugi etap złożyła się inżynieria cech (selekcja zmiennych itd.) oraz wstępne uczenia maszynowe.

3.1. Wykonane prace

— została przeprowadzona normalizacja zmiennych ciągłych (tzw. skalowanie). Wszystkie zmienne numeryczne, oprócz zmiennej 'SpecialDay', a zatem zmienne 'Administrative', 'Administrative_Duration', 'Informational', 'Informational_Duration', 'ProductRelated', 'Product-

- tRelated_Duration', 'BounceRates', 'ExitRates' i 'PageValues' poddamy skalowaniu. Zakres wartości zmiennej 'SpecialDay' już wynosi [0, 1], więc nie jest to dla niej konieczne.
- zmienne typu True/False został zmienione na wartości numeryczne (odpowiednio 1\0). Zmienne 'Weekend' i 'Revenue' przyjmują wartości 'True' i 'False', co przełożymy na wartości 1 i 0.
- kodowanie zmiennej 'Month'. Zmienną 'Month' zakodujemy w postaci kolejnych wierzchołków dwunastokąta foremnego, wpisanego w okrąg jednostkowy. W tym celu stworzymy dwie zmienne Month_x i Month_y, obie o zakresie wartości [-1, 1]. W ten sposób porządek między kolejnymi miesiącami zostanie zachowany (kąt), ale odległość między np. grudniem i styczniem będzie taka sama, jak między styczniem a lutym.
- zostało przeprowadzone enkodowanie zmiennych kategorycznych (zastosowaliśmy onehot encoding)
- przeszliśmy do fazy modelowania został sprawdzone takie algorytmy jak:
 - 1. AgglomerativeClustering
 - 2. KMeans
- stworzenie modelu XGBoost do przewidywania zmiennej 'Revenue'

3.2. Najważniejsze wnioski

- dla grup o wyższym Revenue_ratio średnie PageValues jest zauważalnie wyższe,
- sesje nowych użytkowników zawierają się niemal wyłącznie w grupach o wyższym Revenue_ratio (grupa 1 to niemal same tego typu sesje),
- dla grup o wyższym Revenue_ratio średnie Month_x jest ujemne, a średnie Month_y dodatnie, a dla grup o niższym Revenue_ratio jest dokładnie odwrotnie,
- dla grup o wyższym Revenue_ratio średnie BounceRates jest niższe (szczególnie niskie jest dla grupy 1),
- zaskakująco, sesje z grup o wyższym Revenue_ratio, są znacznie oddalone od dni specjalnych (co jest zgodne ze wcześniejszą analizą),
- sesje z grup o wyższym Revenue_ratio zawierają średnio większą liczbę odwiedzonych stron administracyjnych,
- wśród sesji z grup o wyższym Revenue_ratio większy ich odsetek dotyczy weekendu (co jest zgodne ze wcześniejszą analiza).

4. Etap 3

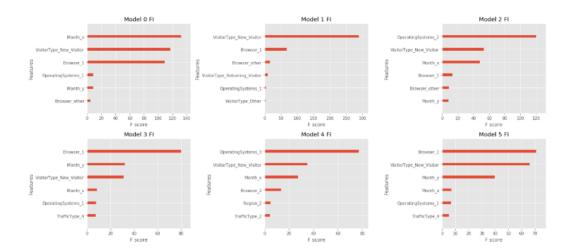
Podczas trzeciego etapu projektu zostały testowane inne modele, przeprowadzone zostało feature_importance oraz została stworzona aplikacja przy wykorzystaniu języka Python.

4.1. Wykonane prace

- wykonaliśmy klasteryzację danych (bez zmiennej 'Revenue') przy wykorzystaniu funkcji grouped_revenue, która zwraca wykresy liczności grup oraz odsetków sesji zakończonych przychodem dla każdej grupy
- została stworzona funkcja, która wytrenowała model do przewidywania przynależności do zadanej grupy (uwzględniliśmy strojenie hiperparametrów)
- wytrenowaliśmy model dla każdej z 6 grup. Z racji na niezbalansowanie danych, jako miarę użyliśmy BACC
- po wytrenowaniu modelu zwróciliśmy uwagę na Feature Importance dla każdej z tych sześciu grup
- zrobiliśmy aplikację w języku Python, gdzie mogliśmy badać nasze wnioski

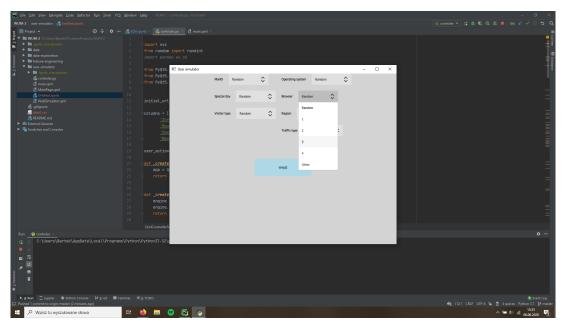
4.2. Najważniejsze wnioski

- podejście z XGBoostem pozwala na klasyfikowanie użytkowników na żywo w przeciwieństwie do Agglomerative Clustering
- poprzez stworzeniu paru modeli dla każdej grupy mogliśmy zobaczyć, że zmienne istotne dla każdej z grupy były różne. Możemy jednak zauważyć pewne podobieństwa niektórych grup (np. Model 2 oraz Model 4 albo Model 5 oraz Model 3). Poniżej są przedstawione wykresy Feature Imoortance dla każdej z grup.

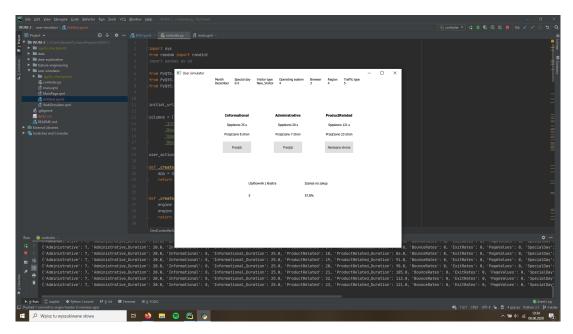


Rys. 1. Feature importance

4.3. Screeny naszej aplikacji



Rys. 2. Pierwsze okno naszej aplikacji



Rys. 3. Drugie okno naszej aplikacji