[IUM] Przewidywanie dokonania zakupu

Bartosz Świtalski (300279)
Piotr Frątczak (300207)

17 maja 2021

1 Kontekst

W ramach projektu wcielamy się w rolę analityka pracującego w firmie *eSzoppping* – sklepu internetowego z elektroniką i grami komputerowymi.

1.1 Obecna sytuacja

Firmowi konsultanci nie wiedzą, czy dana sesja zakończy się zakupem i nie mogą przez to podjąć stosownych decyzji - którym sesjom przyglądać się uważniej.

1.2 Oczekiwania

Usługa pomoże konsultantom skupić się na użytkownikach, którzy najprawdopodobniej dokonają zakupu, dzięki czemu ograniczona zostanie liczba przypadków, w których klient rezygnuje z zakupu w ostatniej chwili lub nie może dokonać zakupu z powodu problemów technicznych, co przełoży się na większą liczbę sprzedanych produktów.

2 Słownik dziedziny problemu

- **klient/użytkownik** - osoba korzystająca ze sklepu internetowego *eSzopping*.
- **zleceniodawca** - osoba zlecająca zadanie wyprodukowania usługi rozwiązującej problem biznesowy, dostarczająca dane i ostatecznie oceniająca wynik.
- **konsultant** - pracownik firmy *eSzoppping*, który kontaktuje się z klientem w razie potrzeby, jego zadaniem jest skłonić klienta do zakupu oraz pomóc z problemami technicznymi.
- **usługa** - program wykonujący zadanie modelowania.
- **zakup** - zamówienie produktu z oferty sklepu i opłacenie zamówienia.
- **sesja** - okres korzystania ze sklepu internetowego o tym samym identyfikatorze sesji.

3 Problem biznesowy

Przewidywanie, czy użytkownik przeglądający sklep internetowy dokona zakupu.

3.1 Biznesowe kryterium sukcesu

Usługa powinna przewidywać czy dojdzie do zakupu z dokładnością na poziomie **co najmniej 70%** (zgodnie z danymi, prawie połowa sesji kończy się zakupem).

4 Zadanie modelowania

Będziemy przewidywać czy dana sesja użytkownika zakończy się zakupem na podstawie dotychczasowej aktywności w ramach sesji.

4.1 Typ zadania modelowania

Klasyfikacja

4.2 Możliwe wyniki

- `tak` - sesja zakończy się zakupem,
- `nie` - sesja nie zakończy się zakupem.

4.3 Analityczne kryterium sukcesu

Osiągnięcie:

$$75\% \leq \frac{\text{liczba sesji sklasyfikowanych jako } \texttt{tak}}{\text{rzeczywista liczba sesji zakończona zakupem}}$$
$$= \frac{\text{liczba sesji sklasyfikowanych jako } \texttt{nie}}{\text{rzeczywista liczba sesji niezakończona zakupem}}.$$

5 Założenia

- Jako sesję zakończoną zakupem rozumie się każdą sesję, w której przynajmniej raz doszło do zakupu,
- Usługa działa dla użytkowników zalogowanych jak i niezalogowanych,
- Usługa będzie działać nieprzerwanie,
- Model zostanie zbudowany jedynie na podstawie danych dostarczonych przez zleceniodawcę,
- Podejście do problemu i model mogą zostać zmodyfikowane w kolejnych iteracjach pracy nad modelem,
- Im więcej danych o aktywności użytkownika w ramach sesji (historii sesji), tym lepiej model będzie przewidywać wynik,
- Model będzie przewidywać wynik dla każdej sesji błyskawicznie, aby dla dużego natężenia na stronie, konsultanci nie musieli czekać kilku minut z decyzją którego klienta obsłużyć pierwszego.

6 Struktura danych wykorzystywanych do modelowania

- katalog produktów
  - id (`product_id`)
  - kategoria (`category_path`)
  - cena (`price`)
- użytkownicy
  - id (`user_id`)
- historia sesji
  - id (`session_id`)
  - stempel czasowy (`timestamp`)
  - id użytkownika (`user_id`)
  - id produktu (`product_id`)
  - typ sesji (`event_type`)
  - zaoferowana zniżka (`offered_discount`)

7 Analiza dostarczonych danych

Wstępną analizę dostarczonych danych przeprowadzimy za pomocą pakietów `Pandas` oraz `json_lines` dla języka `Python`. Będziemy sprawdzać:

- czy dostarczone dane mają braki (np. wartości *null*),
- czy dostarczone dane są prawidłowe pod kątem formatu (np. czy cena produktu to *float*),
- czy dostarczone dane zachowują więzy integralności (np. czy użytkownik z historii sesji istnieje w spisie użytkowników),
- czy dostarczone dane są sensowne (np. czy cena produktu mieści się w realnym zakresie).

7.1 Użytkownicy

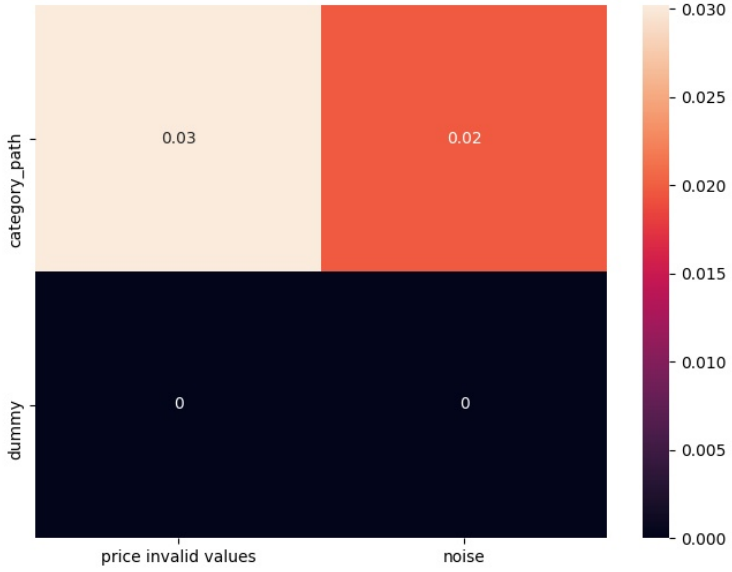
7.1.1 Id

Przeprowadzona analiza nie wykazała obecności jakichkolwiek błędów w potrzebnych danych.

7.2 Katalog produktów

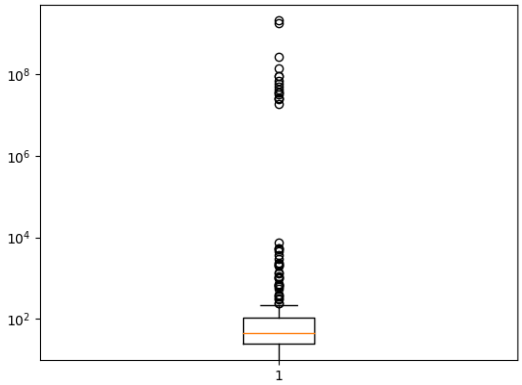
7.2.1 Cena

Występują dwojakie błędy w cenach produktów: ceny ujemne oraz zauważalnie i niedopuszczalnie odbiegające od średniej (które mogą być spowodowane błędami przy wprowadzaniu np. cena urządzenia wielofunkcyjnego *Kyocera* równa 2048500000 - cena kilka rzędów większa w porównaniu do wyników wyszukiwaniu w internecie).



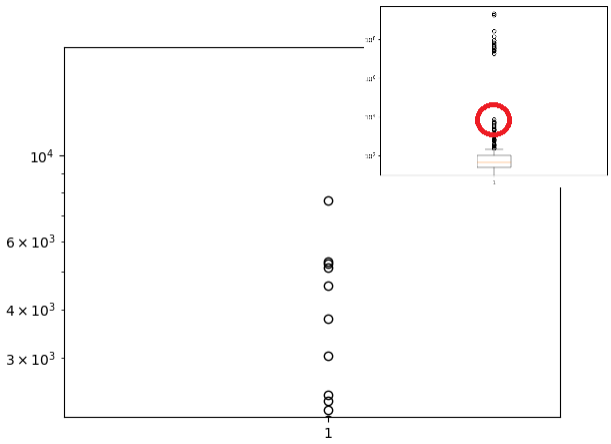
Rysunek 1: współczynnik informacji wzajemnej ceny (`price`) i kategorii produktu (`category_path`)

Analiza wykazała, że błędy są `MCAR`.



Rysunek 2: rozkład cen produktów

Ceny zauważalnie i niedopuszczalnie odbiegające od średniej to takie, których rząd wielkości jest większy niż 10^4.



Rysunek 3: rozkład cen produktów (rozwiązania dopuszczalne)

Dane z katalogu produktów		
# błędnych	# prawidłowych	% prawidłowych
29	290	90.9%

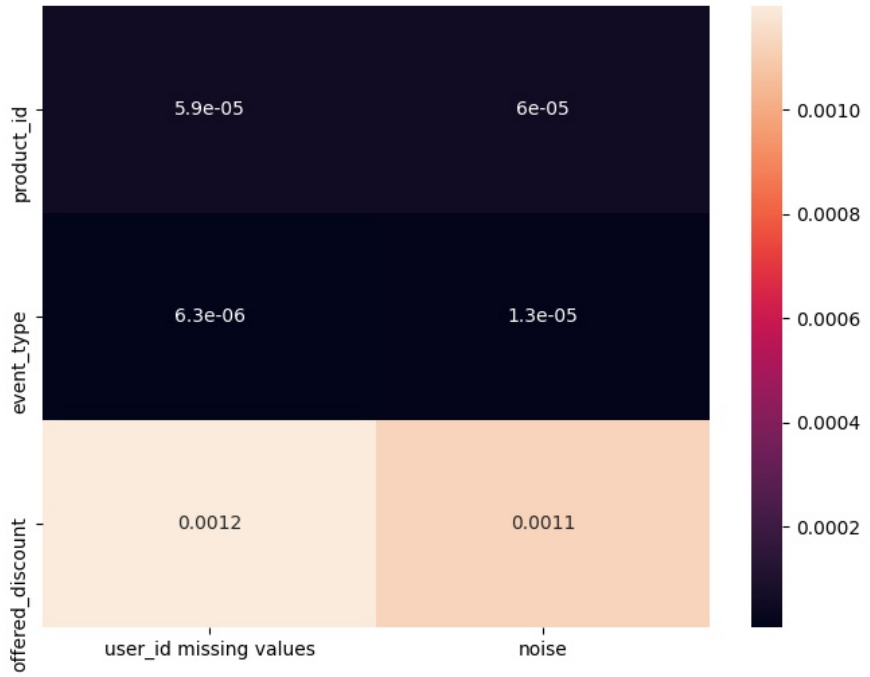
7.2.2 Id i kategoria

Nie zaobserwowano błędów, ani brakujących wartości dla atrybutów `product_id` i `category_path`.

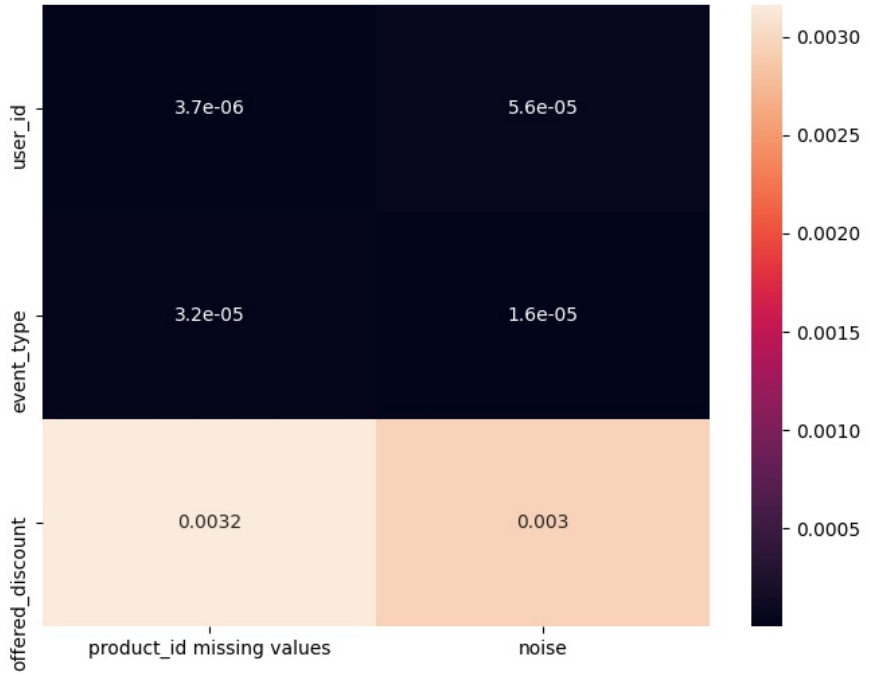
7.3 Historia sesji

7.3.1 Id użytkownika i id produktu

Przeprowadzona analiza wykazała, że są braki w atrybutach `user_id` oraz `product_id`. Aby sprawdzić rodzaj brakujących danych, dodaliśmy kolumnę pomocniczą, zawierającą 0 lub 1 w zależności, czy w sprawdzanej kolumnie jest brakująca wartość, czy nie.



Rysunek 4: tabela współczynnika informacji wzajemnej dla atrybutu `user_id`



Rysunek 5: tabela współczynnika informacji wzajemnej dla atrybutu `product_id`

Uzyskane wyniki pozwalają na stwierdzenie, że są to braki `MCAR`. Podczas dalszej analizy danych będziemy zatem próbować odtworzyć część danych, a niemożliwą do reprodukcji resztę usuniemy.

Wartości NaN w historii sesji	
# w user_id	# w product_id
1837	1786

Nie jesteśmy w stanie odtworzyć atrybutu `product_id`, zatem sesje z pustym atrybutem `product_id` zostaną usunięte. Atrybut `user_id` jesteśmy w stanie odtworzyć, jeśli istnieje wpis z danej sesji z poprawnym atrybutem `user_id` (wtedy go kopiujemy).

Dane z historii sesji		
# wszystkich	36618	
# wstępie poprawnych	33089	90.36%
# poprawnych po reprodukcji	34702	94.77%

Po usunięciu nieprawidłowych produktów z pliku `products.jsonl` analiza wykazała 3972 naruszenia w więzach integralności na wartościach `product_id` (wartości `NaN`). Analiza nie wykazała naruszenia więzów integralności na wartościach `user_id`.

Dane z historii sesji		
# wszystkich	34702	94.77%
# naruszeń więzów integralności	3972	10.85%
# poprawnych	30730	83.92%

7.3.2 Typ sesji

Dane po reprodukcji były wolne od błędów dla atrybutu `event_type`.
Wyświetlenia, które zakończyły się zakupem, stanowią około 13.78% wszystkich wyświetleń.

Dane z historii sesji		
# poprawnych	30730	
# BUY_PRODUCT	4110	13.37%
# VIEW_PRODUCT	26620	86.63%

7.3.3 Id, stempel czasowy i zaoferowana zniżka

Dane dla atrybutów `session_id`, `timestamp` oraz `offered_discount` nie wykazały żadnych nieprawidłowości.

7.4 Wstępne wnioski

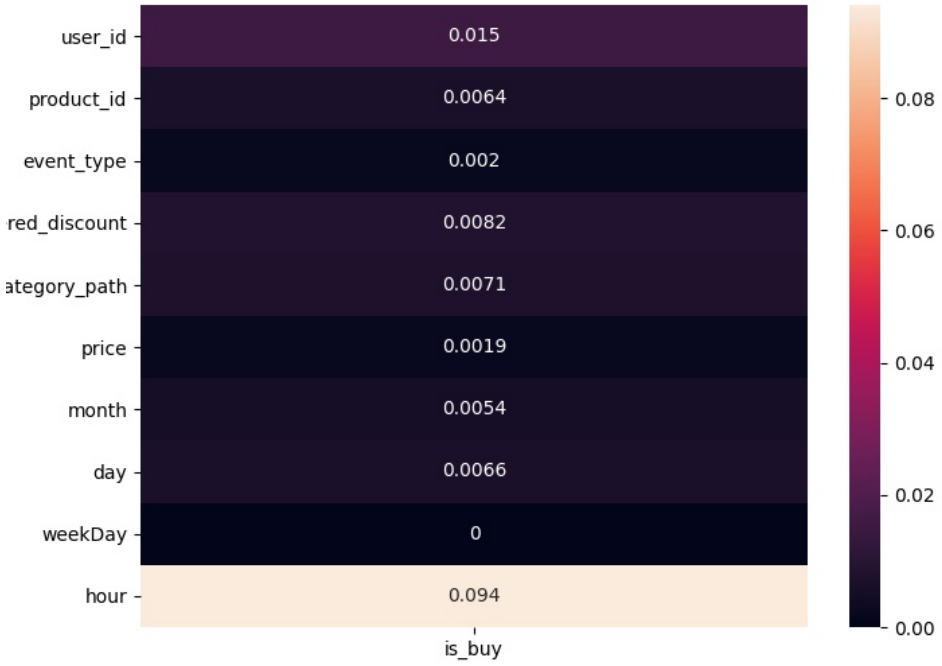
7.4.1 Sesje zakończone zakupem

Z otrzymanych danych (po połączeniu rekordów o tym samym atrybucie `session_id`) wynika, że 48,20% sesji kończy się zakupem.

Dane z historii sesji		
# wszystkich	9068	
# sesji zakończonych zakupem	4109	45.31%
# sesji bez zakupu	4959	54.69%

7.5 Zmienne wejściowe a zmienna celu

Aby zweryfikować czy zmienne wejściowe niosą informację o zmiennej celu, zbadano ich współczynnik wzajemnej informacji. Wyniki są bardzo niskie, bliskie 0. Jedynym wyjątkiem jest `session_id`, lecz powodem jest fakt, że `is_buy` została wyliczona na podstawie identyfikatora sesji. Wnioskujemy, że zmienne wejściowe bezpośrednio nie niosą informacji o zmiennej celu, lecz zgodnie z przyjętymi założeniami będzie można tę informację wyciągnąć po zbudowaniu odpowiedniego modelu.



Rysunek 6: tabela współczynnika informacji wzajemnej dla zmiennych wejściowych i zmiennej celu