# IUM projekt - 2021Z

## Zadanie 8, wariant 3

## Etap 1

Michał Łątkowski, Vladyslav Kyryk

Wersja 2

Ostatnia aktualizacja dokumentu: 30.12.2021

**Kontekst**

W ramach projektu wcielamy się w rolę analityka pracującego w firmie „eSzoppping” – sklepu internetowego z elektroniką i grami komputerowymi. Praca na tym stanowisku nie jest łatwa – zadanie dostajemy w formie enigmatycznego opisu i to do nas należy doprecyzowanie szczegółów tak, aby dało się je zrealizować. To oczywiście wymaga zrozumienia problemu, przeanalizowania danych, czasami negocjacji z szefostwem. Poza tym, oprócz przeanalizowania zagadnienia i wytrenowania modeli, musimy przygotować je do wdrożenia produkcyjnego – zakładając, że w przyszłości będą pojawiać się kolejne ich wersje, z którymi będziemy eksperymentować.

Jak każda szanująca się firma internetowa, eSzoppping zbiera dane dotyczące swojej działalności – są to:

- baza użytkowników,
- katalog produktów,
- historia sesji użytkowników,
- dane dotyczące wysyłki zakupionych produktów.

**Treść zadania**

„Mamy problemy z odpowiednim zapełnianiem półek magazynowych. Nigdy nie wiadomo, co tak naprawdę będzie potrzebne w najbliższym tygodniu, co powinniśmy zamówić. Może da się coś z tym zrobić?”

**Definicja problemu biznesowego**

Potrzeba optymalizacji zapełniania półek magazynowych w celu uniknięcia niepotrzebnych zakupów, a także nadmiarów/braków na magazynie.

**Zadanie modelowania**

Predykcja zachowań klientów, a konkretnie zapotrzebowania na każdy produkt w najbliższym tygodniu. Model ma przewidywać ile sztuk każdego z produktów musi być dostępnych na magazynie (ewentualnie ile trzeba zamówić, ale wtedy musimy wiedzieć ile obecnie jest na magazynie, a nie mamy takiej informacji).

Uwaga: definicja zadania modelowania jest mocno niejasna, nie wynika z niej co dokładnie model będzie robić

Do analizy szeregów czasowych oraz tworzenia predykcji użyjemy modelu ARIMA. Na wejściu model dostanie rok oraz numer tygodnia, dla którego trzeba zwrócić prognozę. Model musi zwracać n liczb całkowitych (gdzie n = liczba wszystkich produktów, sprzedawanych w sklepie). Każda z liczb będzie oznaczała ile sztuk danego produktu będzie sprzedanych (zgodnie z prognozą modelu) w określonym tygodniu.

**Założenia**

- Większym problemem (oprócz sytuacji skrajnych) jest brak danego produktu, niż jego nadwyżka. Sytuacje skrajne to np. wysoka cena czy duży rozmiar produktu, przez co jeszcze bardziej nie chcemy przechowywać w nadmiarze tego produktu na magazynie, w porównaniu do tańszych/mniejszych rzeczy. Na przykład, objętości telewizora i gry komputerowej różnią się o kilka rzędów wielkości, i wtedy o ile nadmiar 10 gier może nie być problemem dla nas, to 10 telewizorów zajmuje sporo miejsca. Dlatego bardzo przydatne byłyby dane o rozmiarze produktów, ale nawet jeśli takie dane nie istnieją, możemy oszacować wielkość produktu za pomocą kilku zmiennych porządkowych (np. mały, duży, bardzo duży), biorąc pod uwagę, że mamy tylko 319 produktów, z których 243 to gry komputerowe. Prawdopodobnie da się oszacować rozmiar produktów patrząc na kategorię, do której należą, ewentualnie potem sprawdzając czy nie ma żadnych “wyjątków” w tej kategorii.
- Model ma przewidywać ile dokładnie produktów będzie sprzedanych, żeby zbadać jego dokładność. Ewentualnie osoba korzystająca z przewidywań

modelu, czyli ta, która jest odpowiedzialna za zamówienie produktów, może zamówić o 1-3 sztuki więcej - na zapas.

**Kryteria sukcesu**

- **Biznesowe**
  - Zmniejszenie liczby nadmiarowych produktów na magazynie.
  - Zwiększenie zysku poprzez redukcję problemu braków magazynowych.
  - Im dokładniej model przewiduje ile sztuk każdego z produktów będzie sprzedanych w przyszłym tygodniu, tym lepiej. Brak i nadmiar będą miały własne wagi (hiperparametry modelu). W modelu bazowym te wagi będą takie same, ale w modelu zaawansowanym, będą one różne, ponieważ brak produktu powinien mieć bardziej negatywny wpływ na jakość modelu, niż nadmiar (oprócz sytuacji skrajnych).

Uwaga: ustalone biznesowe kryteria sukcesu, zabrakło analitycznych kryteriów

- **Analityczne**
  - Brak/nadmiar każdego z produktów nie przekracza 5 sztuk w tygodniu w porównaniu z prognozą modelu.
  - Braki dotyczą maksymalnie 40% liczby wszystkich produktów (np. jeśli mamy 300 unikalnych produktów to nie może być ponad 120 brakujących różnych produktów w jednym tygodniu). Nadmiar jest mniej krytyczny.

**Analiza danych z perspektywy realizacji zadań modelowania (trzeba ocenić, czy dostarczone dane są wystarczające – może czegoś brakuje, może coś trzeba poprawić, domagać się innych danych, ...)**

Statystyki i wykresy są podane w notebooku, załączonym razem z obecnym dokumentem.

Uwaga: dane wejściowe są jakoś opisane, ale nie jest explicite wybrany ich zakres na potrzeby modelu, nie ma odniesienia do zmiennej celu

Zmienna celu - liczba sprzedanych sztuk każdego z produktów w tygodniu. Do trenowania modelu użyjemy wszystkich danych, oprócz 20% najnowszych tygodni, które użyjemy do testowania dla testowania, czyli np. jeśli mamy dane dla zakresu 100 tygodni, to 80 tygodni użyjemy do wytrenowania modelu, a 20 najnowszych do testowania.

Uwaga: brak weryfikacji, czy dane wydają się nadawać do modelowania (czy zmienne wejściowe coś mówią o zmiennej wyjściowej).

Sumując liczbę rekordów (w zbiorze sessions z wartością “BUY_PRODUCT” w atrybucie “event_type”) dla każdego produktu oddzielnie, możemy przeanalizować popyt na dany produkt w kolejnych tygodniach i wytrenować model do prognozowania na szeregach czasowych (ARIMA).

**NOWE DANE - ANALIZA:**

**Deliveries**

- Zbiór danych zbędny w kontekście zadania,
- Purchase_id zawsze unikalne, nie ma pustych wartości,
- Purchase_timestamp zawsze jest w poprawnym formacie i nie ma pustych wartości,
- Delivery_timestamp jak wyżej,
- Delivery_company - 3 unikalne wartości, brak pustych wartości.

**Products**

- Product_id jest unikalny, nie ma pustych wartości,
- Product_name jest unikalny, nie ma pustych wartości,
- Category_path nie ma pustych wartości, jest 15 różnych wartości tej zmiennej, wszystkie wydają się być poprawne w kontekście zadania,
- Price - brak pustych wartości, mediana 41, średnia 248, maksymalna wartość 7639, minimalna 1. Analiza wartości minimalnych pokazuje, że są 3 gry kosztujące 1zł (mogłoby to być sensowne, jeśli np. są dołączane do innych gier). Wartości maksymalne są powiązane z komputerami, więc wydają się być rozsądne.

**Sessions**

- Session_id nie ma pustych wartości, nie jest unikatowe, ale nie musi być - w jednej sesji jest wiele działań (oglądanie/kupno różnych produktów),
- Timestamp - ma poprawny format, nie ma pustych wartości,
- User_id - nie ma pustych wartości, jest 300 różnych wartości tej zmiennej,
- Product_id - brak pustych wartości, zmienna przyjmuje 197 różnych wartości, a 319 produktów jest w zbiorze *products*, czyli nie mamy informacji o sprzedaży 122 produktów (38%), co jest podejrzane. **Prosimy o sprawdzenie, czy istnieją dane (w zbiorze sessions) dla tych**

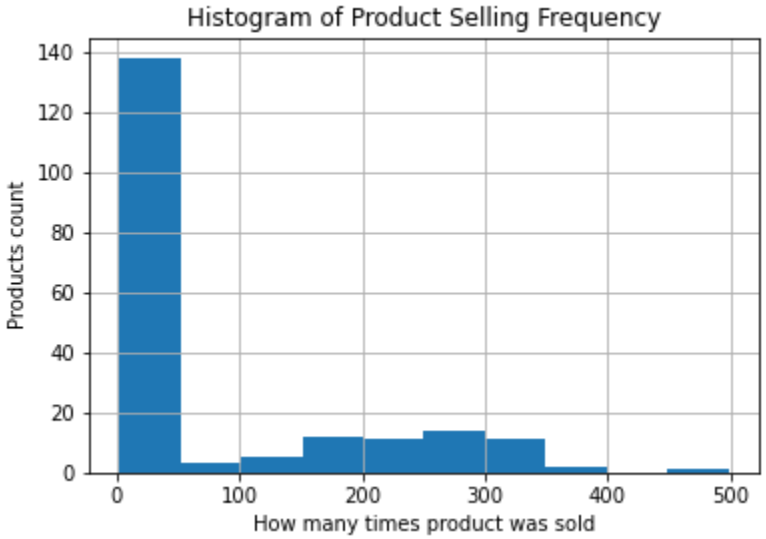
**produktów.** Jeśli takie dane nie istnieją, model nie będzie mógł robić przewidywań dla tej grupy produktów,

- Event_type - brak pustych wartości, przyjmuje 2 wartości - VIEW_PRODUCT albo BUY_PRODUCT. Kupiono tylko 16056 produktów, ale biorąc pod uwagę, że jest ich tylko 197 unikalnych, przypuszczamy, że taka liczba wystarczy, żeby wytrenować model (co prawda, oszacowanie jego jakości nie jest trywialne),
- Offered_discount - brak pustych wartości, 5 unikalnych wartości (0, 5, 10, 15, 20),
- purchase_id - tutaj puste wartości oczywiście są, 16056 nie jest puste (te, które mają event_type równy BUY_PRODUCT).

**Users**

- User_id - brak pustych wartości, 300 unikalnych wartości, co się zgadza z liczbą unikalnych wartości user_id w sessions,
- Name - brak pustych wartości,
- City - brak pustych wartości, 7 unikalnych wartości, wszystkie poprawne,
- Street - brak pustych wartości, wszystkie unikatowe.

Jakość przewidywań dla różnych produktów może się różnić, ponieważ częstotliwość sprzedaży produktów różni się (138/197 produktów było sprzedano mniej niż 50 razy, wtedy jak najbardziej popularny produkt był sprzedany 498 razy, a najmniej popularny tylko 2 razy), co widać na załączonym histogramie.



**STARE DANE - ANALIZA:**
**Ogólne wnioski dotyczące jakości danych w każdym zbiorze:**

**Sessions**

- Najważniejszy zbiór danych w kontekście naszego zadania, ponieważ tam mamy informację o tym jaki produkt został kupiony i kiedy, co właśnie pozwoli nam przewidzieć ile produktów będzie kupionych w przyszłości.
- W kolumnach user_id i product_id są puste wartości (około 5%).
- Sesje bez product_id możemy potraktować jako rozpoczęte, ale użytkownik nie obejrzał/kupił żadnego produktu, ale musimy je wyrzucić, bo dla naszego zadania zmienna product_id jest krytyczna.
- Sesje bez user_id możemy potraktować jako sesje niezarejstrowanych użytkowników, chociaż w teorii system powinien nadawać im jakieś tymczasowe id. **Chcielibyśmy dowiedzieć się, czy możemy używać takich sesji w naszym modelu, jeśli wartość product_id nie jest pusta, czy raczej musimy traktować takie sesje jako błędne?**
- Tylko 2975 z 173632 produktów zostały kupione - wartość “BUY_PRODUCT” w kolumnie “event_type” (reszta to “VIEW_PRODUCT”).
- Możliwe, że nam trochę pomogą dane o tym, jak często jakiś produkt był oglądany (możemy uwzględnić zależność pomiędzy tym jak często produkt był oglądany, a potem kupiony), jeśli np. będziemy już znać stany magazynowe, ale wciąż musimy wiedzieć, ile jakich produktów było kupionych. Dlatego potrzebujemy więcej danych odnośnie zakupu. W naszym zadaniu najważniejsze są sesje z atrybutem “BUY_PRODUCT”. Im więcej takich takich sesji będzie dostępnych dla modelu, tym lepsza będzie jego jakość.
- Dane są tylko od 1 stycznia 2021 roku. Warto sprawdzić czy nie ma żadnych danych, z okresu przed tą datą. Nie mamy nawet całego roku, co utrudni nam pracę nad modelem, ponieważ ciężej uwzględnić w takim przypadku sezonowość. Również należy wziąć pod uwagę fakt, że często jest duży wzrost w sprzedażach w okresie przedświątecznym w grudniu, co można zauważyć, analizując statystyki naszych konkurentów na rynku. Nie mając danych z kilku poprzednich lat, prawie nie potrafimy ocenić sezonowość.

**Products**

- W zbiorze nie ma pustych wartości, product_id jest poprawnym kluczem głównym. Product_name i category_path to faktycznie zawsze stringi.
- W kolumnie 'price' mamy pewne wartości odstające, które w większości są błędne. Wśród tych wartości znajdują się wartości skrajnie duże, jak i ujemne:
    - 20 produktów (6.3%) z ujemną ceną - taka sytucja jest niedopuszczalna i ceny muszą być poprawione.
    - 19 (6.8%) z ceną ponad 1,000,000. 15/19 produktów to są gry. Nie wiemy w jakiej walucie są ceny, ale patrząc na inne ceny, wartości 1,000,000 występować nie powinny.
    - Ze względu na wysoką liczbę błędów, również prosimy o sprawdzenie cen na inne produkty, ponieważ nie wiemy co jest przyczyną pomyłek i mogą one pojawić się w zakresie od 0 do kilku tysięcy. W zbiorze znajdują się np. telefony z ceną 15.

**Deliveries**

- Kolumny delivery_timestamp i delivery_company zawierają puste wartości. W przyszłości należałoby albo dostać lepsze dane, albo te kolumny odsiać.
- Pomyślna konwersja typów sugeruje, że typy danych w zbiorze są takie jak oczekiwane.
- Zbiór deliveries - może się przydać purchase_timestamp (razem z purchase_id). Chociaż jeśli jest on kopią timestamp ze zbioru sessions, to wtedy nie dodaje nam żadnej nowej informacji.
- Delivery_company oraz delivery_timestamp raczej zbędne w kontekście zadania.

**Users**

- W zbiorze nie ma pustych wartości, każda wartość, która powinna być stringiem, jest stringiem, user_id to faktycznie unikalny identyfikator.
- Zbiór users w tym momencie prawie nie ma żadnej przydatności.
- Zbiór users - można poszukać czy są jacyś klienci, którzy często coś kupują. Może są klienci, którzy np. co miesiąc kupują nową grę komputerową (ale do tego wystarczy nam wartość user_id w zbiorze sessions). Ewentualnie można spróbować jakoś ich posegmentować (pogrupować), chociaż biorąc pod uwagę dostępne atrybuty, segmentacja raczej nie będzie udana. Gdybyśmy dostali więcej atrybutów w zbiorze

users, to moglibyśmy wychwycić jakieś podobieństwa pomiędzy nimi, co by pozwoliło lepiej przewidywać ich zachowania w kontekście kupna produktów.

**Podsumowanie analizy danych**

- Prosimy o sprawdzenie, czy nie ma więcej sesji z atrybutem BUY_PRODUCT, takie sesje są dla nas najważniejsze i od ich liczby prawdopodobnie będzie w dużej mierze zależeć jakość modelu.

Uwaga: Możliwe, że coś udałoby się znaleźć - proszę napisać o jakiej liczbie rekordów mówimy?

W nowej paczce danych mamy informacje o sprzedażach produktów w okresie ponad 2 lat (od 2019-08-01 do 2021-12-20), co już pozwala na uwzględnienie zjawisk sezonowych (np. wzrost sprzedaży na święta), w porównaniu z poprzednimi danymi, gdzie nie mieliśmy nawet całego roku. Obecna liczba rekordów (w zbiorze sessions z wartością “BUY_PRODUCT” w atrybucie “event_type”) wydaje się być wystarczająca na stworzenie wstępnego modelu, którego jakość będzie rosnąć wraz z otrzymaniem nowych danych w przyszłości.

- Wszystkie ceny produktów muszą być sprawdzone i poprawione, zwłaszcza ceny ujemne i bardzo wysokie. Nie jest to krytyczne dla naszego modelu, ponieważ przewidujemy liczbę, ale ceny mogą się przydać w przypadku doboru wag braku/nadmiaru, jeśli np. nadmiar drogich produktów jest dla nas bardzo niepożądany.
- Przydałyby się nam dane, dotyczące stanu magazynu w czasie, wtedy moglibyśmy przewidywać ile produktów trzeba dokupić. Mając takie dane będziemy mogli odjąć od predykcji, ile produktów będzie sprzedanych, liczba na magazynie i wtedy zamawiający nie będzie musiał sprawdzać ile produktów ma na stanie. Nie jest to obowiązkowe, ale uczyni to model bardziej użytecznym i pomocnym.
- Jeśli są dostępne jakieś dodatkowe atrybuty dla zbioru users, mogą one nam się przydać dla polepszenia przewidywania zachowania klientów.