RoomSage

Forecasting task

Hubert Jóskowiak

Poznań, 2019

Spis treści

1 Opis projektu **3**

2 Zaproponowane rozwiązanie **3**

3 Przygotowanie danych **4**

4 Wykorzystane algorytmy uczenia maszynowego **5**

5 Przedstawienie wyników **7**

6 Wnioski **9**

1. Opis projektu

- Dane wejściowe: próbka danych z systemu Google AdWords oraz systemu hotelowego dla jednego hotelu w okresie dwóch lat.
- Cel: stworzenie modeli prognostycznych dla kolumn *clicks* oraz *conversions* przewidującej wartość danej zmiennej w kolejnym dniu.
- Definicje poszczególnych kolumn w dostępnym zestawie danych:
  - *impressions* - liczba pojawień się reklamy na stronie wyników wyszukiwania lub w witrynie Google Network,
  - *clicks* - liczba kliknięć w reklamę,
  - *conversions* - liczba konwersji wybranych do optymalizacji,
  - *cost* - suma kosztów CPC oraz CPM,
  - *total_conversion_value* - libcza konwersji dla wszystkich rodzajów konwersji,
  - *average_position* - pozycja reklamy w stosunku do pozycji innych reklamodawców,
  - *reservations* – liczba rezerwacji dokonanych danego dnia,
  - *price* – łączna cena zapisana w rezerwacjach z danego dnia.

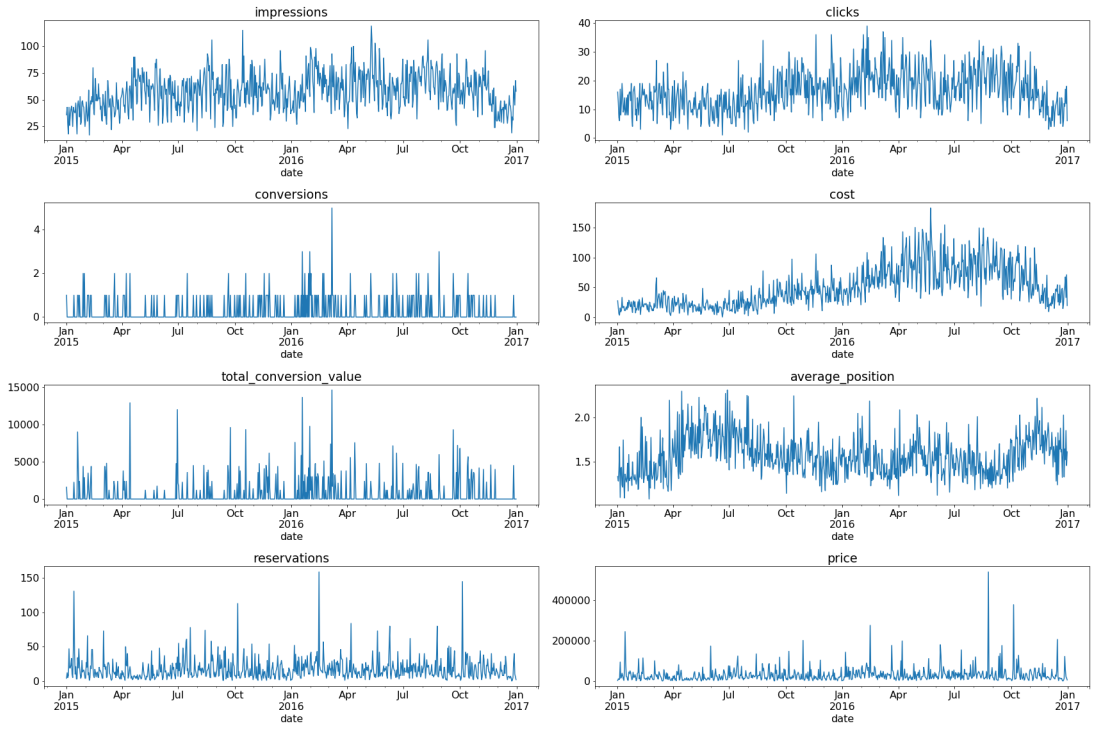
2. Zaproponowane rozwiązanie

- Baseline - uśrednienie wartości dla danej zmiennej z ostatnich 7 dni.
  - Wartość RMSE dla kolumny *clicks* : 5,72
  - Wartość RMSE dla kolumny *conversions* : 0,59
- Wykorzystanie algorytmów uczenia maszynowego:
  - Klasyfikator *Random Forest*
  - Klasyfikator *XGBoost*
  - Głębokie sieci neuronowe

- Wykorzystane oprogramowanie
  - Python (wersja 3.5.5)
  - Biblioteki:
    - numpy
    - pandas
    - matplotlib.pyplot
    - sklearn
    - pprint
    - tensorflow
    - xgboost

3. Przygotowanie danych

- Pobranie danych
- Dodanie informacji o dniu tygodnia - kodowanie one-hot
- Wykrycie błędów w danych (rysunek 3.1) - brak brakujących danych, brak danych odbiegających znacznie od średniej
- Sprawdzenie czy występują niezerowe wartości zmiennej *total_conversion_value* przy zerowych wartościach zmiennej *conversions*
  - Liczba przypadków : 0
- Sprawdzenie czy występują zerowe wartości zmiennej *price* dla niezerowej liczby rezerwacji
  - Liczba przypadków : 2 - przypisanie najmniejszej występującej wartości dla zmiennej *price* równej 200
- Dodanie kolumny ze średnią wartością danej zmiennej w poprzednich siedmiu dniach - prognoza na dany dzień na potrzeby baseline'u
- Przygotowanie danych do uczenia maszynowego - podział na zbiór testowy oraz treningowy.
  - Zbiór treningowy: Dane z losowo wybranych 621 dni (85 %)
  - Zbiór testowy: Dane z 110 dni nie znajdujących się w zbiorze treningowym (15 %)



Rys. 3.1: Wykrycie błędów

4. Wykorzystane algorytmy uczenia maszynowego

- Dobór parametrów dla klasyfikatora Random Forest z wykorzystaniem RandomizedSearchCV

Parametr	**Testowane wartości**	**Optymalna wartość dla *clicks***	**Optymalna wartość dla *conversions***
n_estimators	100, 200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000	800	1800
max_features	'auto', 'sqrt'	'auto'	'auto'
max_depth	5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, None	30	40
min_samples_split	2, 5, 10	5	10
min_samples_leaf	1, 2, 4	2	2
bootstrap	True, False	True	True

- Dobór parametrów dla klasyfikatora XGBoost z wykorzystaniem RandomizedSearchCV

Parametr	**Testowane wartości**	**Optymalna wartość dla *clicks***	**Optymalna wartość dla *conversions***
n_estimators	100, 200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000	200	100
eta	0.001, 0.1, 0.3, 0.5	0.1	0.3
max_depth	5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100	5	30
min_child_weight	1, 2, 5, 10	5	10

- Parametry klasy RandomizedSearchCV
  - proces losowania wartości parametrów z rozkładu, jest wykonywany 200 razy,
  - zbiór treningowy jest dzielony na 3 zbiory, na których przeprowadzana jest walidacja krzyżowa.
- Architektura sieci neuronowej

Typ warstwy	**Liczba neuronów**	**Funkcja aktywacji**
fully connected	256	ReLU
fully connected	256	ReLU
fully connected	256	sigmoid
fully connected	1	-

- Parametry sieci neuronowej

Tab. 4.1: Parametry sieci neuronowej

Parametr	**Wartość**
Rozmiar pojedynczego zbioru uczącego (*batch*)	100
Liczba epok	2000
Współczynnik uczenia	0.001
Funkcja kosztu	RMSE
Algorytm optymalizacji	Adam

5. Przedstawienie wyników

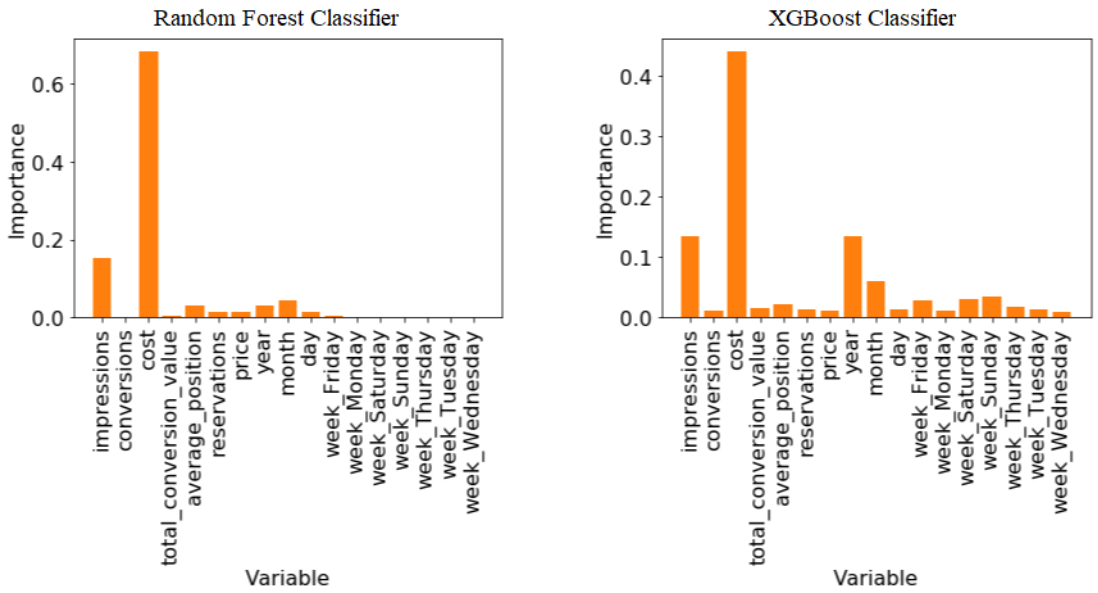
- Wartości RMSE uzyskane dla zmiennej *clicks*

	RMSE zb. treningowego	RMSE zb. testowego
Baseline	-	5.72
Random Forest	1.31	2.61
XGBoost	**0.52**	**2.56**
Neural Network	6.65	6.95

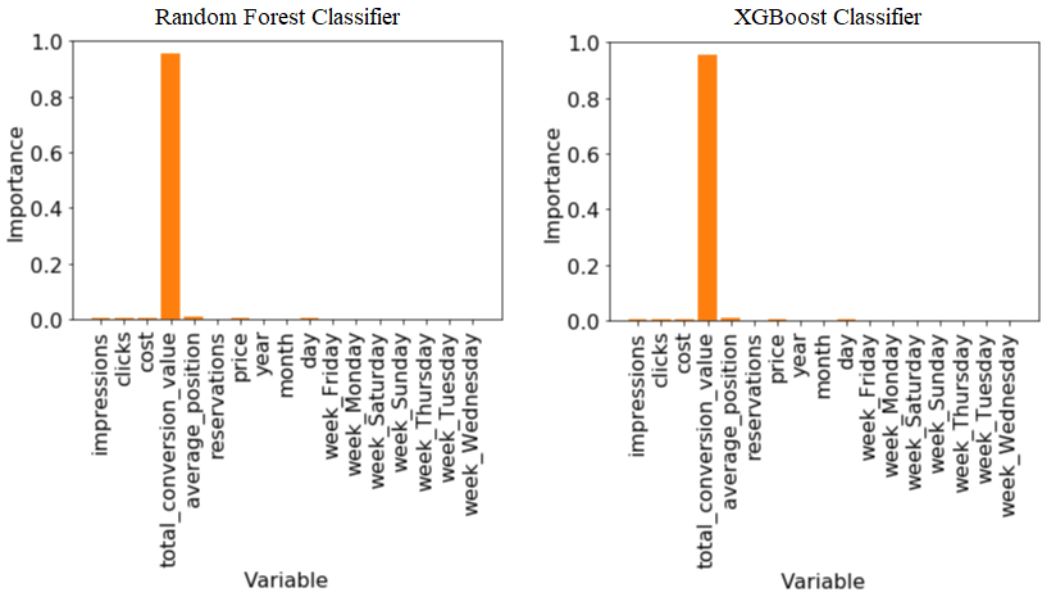
- Wartości RMSE uzyskane dla zmiennej *conversions*

	RMSE zb. treningowego	RMSE zb. testowego
Baseline	-	0.59
Random Forest	0.14	0.25
XGBoost	**0.11**	**0.24**
Neural Network	0.53	0.57

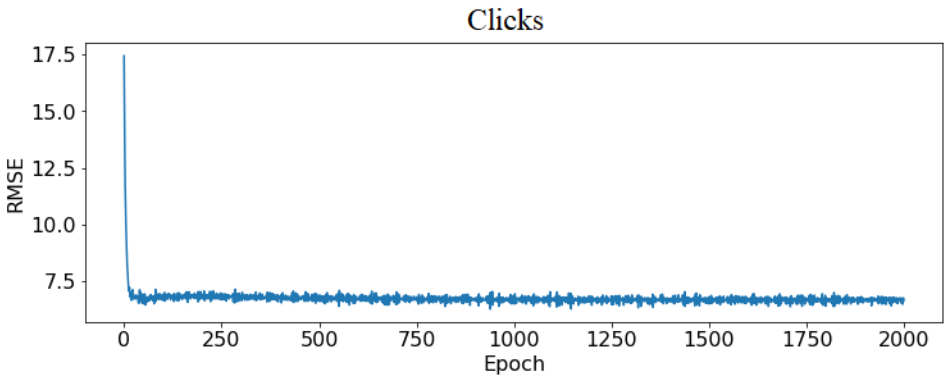
- Wpływ cech na proces uczenia (dla zmiennej *clicks*)

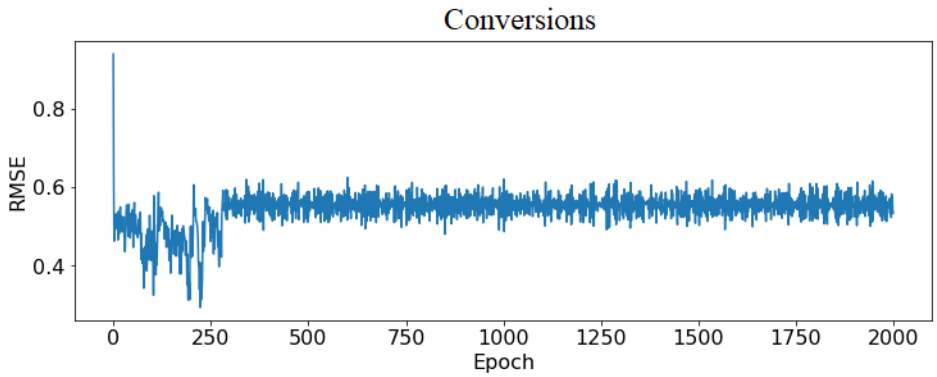


- Wpływ cech na proces uczenia (dla zmiennej *conversion*)



- Trening sieci neuronowej dla obu modeli





6. Wnioski

- Zbiór danych podzielono na zbiór treningowy (621 dni) oraz zbiór testowy (110 dni).
- Zbudowano baseline, zgodnie z którym prognoza optymalizowanej zmiennej została przeprowadzona w oparciu o wartość tej zmiennej w poprzednich siedmiu dniach (jako średnia tych wartości). Dla zmiennej *clicks* wartość rmse dla zbioru testowego wynosiła 5,72, a dla zmiennej *conversions* 0,59
- Zastosowanie klasyfikatora Random Forest, które parametry zostały dopasowane z wykorzystaniem klasy RandomizedSearchCV pozwoliło na zmniejszenie wartości rmse do wartości 2,61 dla zmiennej *clicks* oraz 0,25 dla zminnej *conversions*. Oznacza to poprawę w stosunku do baseline'u o około 55%.
- Zastosowanie klasyfikatora XGBoost pozwoliło na jeszcze lepsze dopasowanie modelu do danych testowych. Wartość rmse dla zmiennej *clicks* wynosi 2,56, a dla zmiennej *conversions* 0,24 co oznacza poprawę w stosunku do baseline'u nawet o blisko 60 %.
- Zaprojektowana sieć neuronowa dla zmiennej *conversions* pozwoliła na minimalne obniżenie wartości rmse w stosunku do baseline'u, natomiast w przypadku trudniejszego do optymalizacji parametru *clicks* (z powodu większego zakresu możliwych wartości) sieć neuronowa nie poradziła sobie z zadaniem - średni błąd kwadratowy wynosi aż 6,95 - jest to spowodowane najprawdopodobniej zbyt małą liczbą danych w procesie uczenia.