

# Wyniki badań opracowane w ramach zrealizowanego projektu

o numerze RPMA.01.02.00-14-9523/17

**„Prace badawczo-rozwojowe w spółce Yield Riser nad wykorzystaniem Sztucznych Sieci Neuronowych w innowacyjnym systemie do optymalizacji przychodów z reklam na witrynach w modelu RTB”**

Projekt współfinansowany ze środków Europejskiego Funduszu Rozwoju Regionalnego w ramach Regionalnego Programu Operacyjnego Województwa Mazowieckiego na lata 2014-2020, Działanie 1.2 Działalność badawczo – rozwojowa przedsiębiorstw.

Przygotowali:  
Andrzej Wodecki  
Rafał Moczadło  
Jakub Szczęsny  
Paweł Treściński

# Spis treści

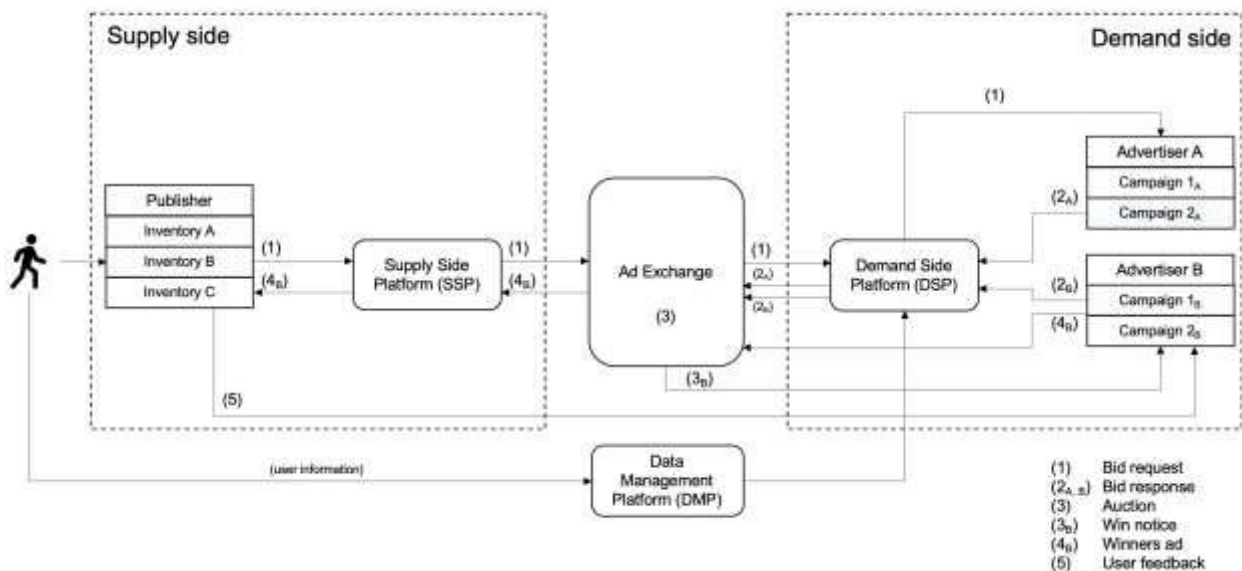
## Spis treści

<b>SPIS TREŚCI</b> .....	<b>2</b>
<b>1. WPROWADZENIE</b> .....	<b>3</b>
<b>2. PRZEGLĄD LITERATURY</b> .....	<b>4</b>
PERSPEKTYWA REKLAMODAWCY .....	4
PERSPEKTYWA WYDAWCÓW .....	5
<b>3. PRACE BADAWCZO-ROZWOJOWE PODJĘTE W PROJEKCIE</b> .....	<b>6</b>
ZADANIE 1: OPRACOWANIE KONCEPCJI PROJEKTU WYKORZYSTANIA ELEMENTÓW SZTUCZNEJ INTELIGENCJI PRZY PLATFORMACH EMITUJĄCYCH REKLAMY NA STRONACH INTERNETOWYCH W SYSTEMIE RTB .....	6
<i>Problem badawczy</i> .....	6
<i>Analiza eksploracyjna danych historycznych i pierwsze eksperymenty z cenami minimalnymi</i> .....	7
<i>Heurystyka wynikająca z eksploracyjnej analizy danych i zrealizowanych eksperymentów</i> .....	12
<i>Analiza dostępnych modeli Sztucznej Inteligencji oraz możliwości ich wykorzystania w projekcie</i> .....	13
ZADANIE 2: ZAPROJEKTOWANIE I ZBADANIE PLATFORMY DO OPTYMALIZACJI SYSTEMÓW SSP DLA YIELDRISER .....	19
ZADANIE 3: OPRACOWANIE PROTOTYPU PLATFORMY DO AUTOMATYZACJI POWIERZCHNI REKLAMOWEJ PRZY WYKORZYSTANIU PLATFORM SSP ORAZ ELEMENTÓW SZTUCZNEJ INTELIGENCJI .....	22
<i>Ogólny opis platformy</i> .....	22
<i>Moduł Sztucznej Inteligencji</i> .....	22
<b>BIBLIOGRAFIA</b> .....	<b>34</b>

## 1. Wprowadzenie

Reklama internetowa stała się w ostatnich latach przemysłem wartym wiele miliardów dolarów. Według Zenith Agency (Austin, Barnard and Hutcheon, 2019), globalne wydatki na reklamę cyfrową w 2019 r. przekroczyły 290 miliardów dolarów, a w 2020 r. można spodziewać się kwoty przekraczającej 320 miliardów dolarów. Reklamodawcy mogą kupować tzw. sloty reklamowe zarówno w drodze umów bezpośrednich (nieprogramowych), jak i aukcji w czasie rzeczywistym (marketing programowy). Według danych (Austin, Barnard and Hutcheon, 2019), marketing programowy stanowił 65% udziału w wydatkach na reklamę cyfrową w 2019 r. Co więcej, oczekuje się wzrostu do poziomu 69% w 2020 r.

Aukcja w czasie rzeczywistym (RTB – *Real Time Bidding*), jako sposób handlu reklamami cyfrowymi poprzez aukcje odbywające się w czasie rzeczywistym jest najbardziej popularnym mechanizmem marketingu programowego. Na rynku RTB można w uproszczeniu wyróżnić stronę popytową (organizacje zainteresowane zakupem powierzchni reklamowej), podażową (udostępnienie tej powierzchni), organizatora rynku oraz firmy wspierające transakcję na różne sposoby. W praktyce, w każdym z tych segmentów istnieje wiele podmiotów o różnych modelach biznesowych i funkcjach docelowych. Uproszczony wykres rynku został przedstawiony na Rys. 1.



Rys. 1. Ekosystem reklamy internetowej. Źródło: Opracowanie własne.

Typowy proces transakcyjny na rynku RTB wygląda następująco. Internauta odwiedza jedną z wyświetlanych przez wydawcę powierzchni reklamowych, najczęściej za pośrednictwem przeglądarki internetowej lub aplikacji mobilnej. Wydawnictwo przekazuje te informacje (tzw. bid request): (1) do platformy typu Supply Side Platform (SSP) wzbogaconej o opis dostępnych narzędzi reklamowych. Platforma SSP (może być ich wiele na rynku) przekazuje ww. informacje do mechanizmu licytacji. Ad Exchange, który inicjuje aukcję: dystrybuuje informacje o narzędziach, wzbogacone o dane użytkowników (demografia, profil behawioralny) uzyskanych z platform typu Data Management Platforms (DMP), z pośród wielu platform Demand Side Platforms (DSP) oczekujących na oferty Reklamodawców. Na DSP zapytanie ofertowe (ang. bid request) jest wstępnie dostosowywane do wcześniej zgłoszonej powierzchni i wymagań użytkowników (kampanii reklamowych). Reklamodawcy otrzymują dane ofertowe i najczęściej za pomocą algorytmów zaimplementowanych w swoich systemach określają cenę, jaką są skłonni zapłacić (tzw. bid response): (2)). Odpowiedzi (bid response) na oferty przekazywane są za pośrednictwem DSP do Ad Exchange, gdzie następuje finalizacja aukcji (3), a informacja o wygranej (ang. win notice) przesyłana jest do Reklamodawcy.

Reklamodawca przekazuje swoją reklamę (poprzez DSP, Ad Exchange i SSP) do Wydawcy, który emituje ją na wskazanym narzędziu. Po emisji, dane, w szczególności informacja o opinii użytkownika (5), są przekazywane Reklamodawcy. Transakcja zostaje sfinalizowana, aukcja zostaje zakończona. Cały proces (od momentu wejścia użytkownika na stronę internetową Wydawnictwa do wyświetlenia reklamy) trwa około 0,1 sekundy.

Rynek RTB można porównać do gry w trybie multi player, gdzie każdy gracz ma swoje cele. Wydawnictwo chce maksymalizować zyski z reklam, co wymaga, aby ceny były z jednej strony jak najwyższe, a z drugiej konkurencyjne wobec innych wydawców. Reklamodawca musi także inteligentnie licytować: oferować ceny, które minimalizują koszty pozyskania klienta, ale są wyższe od cen innych reklamodawców. Jest to utrudnione przez brak przejrzystości informacji na rynku: 1. Platformy DSP i SSP są jego aktywnymi uczestnikami: DSP często masowo kupują przestrzeń od wydawców i licytują tylko po znanych cenach; 2. reklamodawcy nie mają informacji o cenie sprzedaży aukcyjnej, której nie wygrali; oraz 3. wydawcy nie znają wszystkich cen oferowanych przez reklamodawców na aukcjach.

Nic dziwnego, że wszyscy uczestnicy rynku aktywnie poszukują rozwiązań (metod, algorytmów, technologii), które zoptymalizowałyby ich cele. W niniejszym podsumowaniu skupimy się na perspektywie Wydawcy, a samych Reklamodawców traktując w dużym skrócie. Głównym celem jest określenie tzw. strategii optymalizacji (minimalnych) cen wywoławczych, która jest nie tylko efektywna, ale również możliwa do wdrożenia w średniej wielkości agencji marketingowej RTB. Prezentowane strategie (przedstawione w różnych omówionych poniżej pracach badawczych) w większości przypadków starają się prognozować strategie reklamodawców w celu maksymalizacji zysku z danej emisji reklam, co 1. sprawia, że są to poważne wyzwania obliczeniowe dla małych agencji RTB oraz 2. nie uwzględniają innych celów wydawniczych, takich jak optymalne wykorzystanie powierzchni reklamowej (mierzone parametrem Coverage (wskaźnik pokrycia).

W naszym projekcie stosujemy znacząco inną heurystykę: zamiast zgadywać wartość dodaną dla reklamodawców, stosujemy zaawansowane algorytmy szeregowych do krótkoterminowego (3-dniowego) prognozowania parametru Coverage (wskaźnik pokrycia) i ustalenia ceny wywoławczej, aby utrzymać ten parametr na możliwie maksymalnym poziomie. Dzięki temu jest on nie tylko efektywny obliczeniowo (możliwe jest codzienne przekwalifikowanie modelu), ale także łatwy do zrozumienia. Przedstawiona poniżej strategia zwiększa przewagę konkurencyjną spółki Yield Riser i zwiększa zyski klientów (Wydawców).

## 2. Przegląd literatury

Każdy z opisanych powyżej uczestników rynku RTB ma swoją funkcję celową, którą stara się maksymalizować. Ze względu na asymetrię informacyjną i złożoność mechanizmów aukcyjnych rodzi to wiele wyzwań. Dostęp do danych i duże praktyczne znaczenie problemu motywuje badaczy do formułowania i sprawdzania hipotez, które pomagają osiągnąć przewagę konkurencyjną na tych rynkach. Poniżej przedstawiamy zarys aktualnych wyzwań badawczych oraz próby ich rozwiązania zarówno z perspektywy wydawców, jak i reklamodawców.

### Perspektywa reklamodawcy

Głównym celem reklamodawców jest zakup najlepszej możliwej powierzchni reklamowej po możliwie najniższej cenie. W większości przypadków zakup ten jest częścią szerszej kampanii reklamowej, która ma pewne konkretne cele (produkt, odbiorca, itp.), budżet i czas realizacji. Reklamodawca, po otrzymaniu informacji o aktualnie dostępnej powierzchni reklamowej i odwiedzającym go kliencie

(zapytanie ofertowe [bid request] wzbogacone o dane informacyjne użytkownika) musi bardzo szybko podjąć decyzję o udziale w aukcji i określić cenę, jaką będzie gotów zapłacić. Czyni to w warunkach niepełnej informacji: nie zna zarówno ceny minimalnej (wywoławczej) ustalonej przez Wydawcę, jak i cen oferowanych przez innych Reklamodawców. W aukcjach drugiej ceny (second-price), wyłącznie zwycięzca aukcji zna cenę sprzedaży - pozostali licytujący wiedzą tylko, że ktoś zaoferował wyższą cenę, co nazywamy danymi cenzurowanymi.

Jedną z najbardziej popularnych strategii radzenia sobie z takimi przypadkami jest tzw. prognozowanie krajobrazowe (ang. landscape forecast), polegające na prognozowaniu rozkładu prawdopodobieństwa cen oferowanych przez rynek dla każdej aukcji. Istnieje wiele podejść do rozwiązania tego problemu. Jedna grupa metod opiera się na analizie przetrwania w naukach medycznych, biorąc pod uwagę tylko przegrywające logi (por. (Wang et al., 2016), (Wu, Yeh and Chen, 2018), (Wu, Yeh and Chen, 2015). Ren et al. (Ren et al., 2019) z kolei stosuje metody analizy przetrwania dla obsługi cenzury, i głębokich sieci neuronowych do modelowania warunkowego prawdopodobieństwa zwycięstwa w odniesieniu do każdej ceny ofertowej.

Inne podejście, oparte na wzmocnionym uczeniu (reinforcement learning, RL), zostało opisane w (Cai et al., 2017). Algorytm reprezentujący Reklamodawcę (nazywany *Agentem* w ramach RL) na każdym etapie obserwuje parametry kampanii (pozostały czas trwania, budżet) oraz zapytanie ofertowe o konkretną emisję reklamy (wzbogacone o dane użytkownika). Za pomocą takiego zestawu informacji (zwanego *Stanem lub Staturem* (ang. *State*)), *Agent* decyduje o cenie oferty i wykonuje *Działanie* (ang. *Action*). Po zakończeniu aukcji otrzymuje *Nagrodę* (ang. *Reward*), która jest pochodną wyników aukcji oraz informacji zwrotnej od użytkownika (w przypadku wygrania aukcji). Na koniec, w oparciu o schemat *State > Action > Reward*, *Agent* aktualizuje swoją *Politykę*. Po wielu iteracjach *Agent* poznaje optymalną politykę dla danej kampanii. Główną wadą tego podejścia polega na tym, że w wielu przypadkach algorytmy RL wymagają tysięcy iteracji, aby nauczyć się optymalnej polityki, co wymaga czasu i zużywa dużo mocy obliczeniowej. Najczęstszym podejściem do rozwiązania tego problemu jest symulacja rynku RTB za pomocą danych historycznych, ale podejście to nie może być wdrażane w rzeczywistych zastosowaniach (*Agent* przeszkolony w zakresie danych historycznych prawdopodobnie nie będzie zachowywał się optymalnie w nowych, realnych sytuacjach).

Kolejnym wyzwaniem podejmowanym we współczesnych badaniach nad optymalizacją procesów ustalania cen zakupu powierzchni na rynkach RTB jest wysoka dynamika tych środowisk. W związku z tym proces podejmowania decyzji zakupowych nie może być traktowany jako kwestia optymalizacji stacjonarnej. Problemem zajmuje się wiele grup badaczy (Cai et al., 2017; Lu et al., 2019; Yang et al., 2020), jako rozwiązanie proponując różne opcje usprawnienia (procesu) uczenia.

Przedstawiona powyżej perspektywa reklamodawców jest często podstawą do tworzenia strategii optymalizacji cen.

## Perspektywa wydawców

Głównym celem wydawców jest maksymalizacja zysku ze sprzedaży ich powierzchni reklamowej, zarówno w kanałach off-line (np. poprzez umowy bezpośrednie), jak i on-line (np. na rynkach RTB). W przypadku każdego nowego wydania muszą niezwłocznie podjąć decyzję dotyczącą kanału dystrybucji i poziomu ceny minimalnej. Zbyt niska cena generuje ryzyko utraty zysku, a zbyt wysoka, ryzyko braku emisji (Li, Ni and Yuan, 2018). W systemach aukcji drugiej ceny (second-price auction), sytuację dodatkowo komplikuje fakt, że na rynku jest wielu reklamodawców oferujących bardzo niskie ceny zakupu. Motywuje to do stworzenia strategii ustalania cen wywoławczych, chroniącą wydawców przed praktycznie swobodnym dostępem do ich powierzchni.

Strategia ustalania cen wywoławczych jest bardzo ważna nie tylko dla wydawców (którym gwarantuje się pewien poziom cen sprzedaży powierzchni reklamowej), ale także dla całego ekosystemu RTB (ustalającego równowagę między popytem a podażą) (Li et al., 2017). Wydawcy, podobnie jak reklamodawcy, nie znają zachowań swoich konkurentów (innych wydawców) i potencjalnych klientów. Sytuację komplikuje fakt, że powierzchnia reklamowa jest sprzedawana zarówno on-line (obecnie dominująca forma sprzedaży), jak i off-line (bezpośrednia sprzedaż B2B). W związku z tym wydawca musi ustalić odrębne ceny wywoławcze dla obu kanałów, ale także bardzo szybko podjąć decyzję, który kanał będzie oferował daną powierzchnię reklamową.

Naukowcy od lat zajmują się kwestią optymalizacji cen wywoławczych w systemach aukcyjnych (Myerson, 1981; Riley and Samuelson, 1981). Najczęstsze aktualnie strategie cen wywoławczych koncentrują się na prognozowaniu strategii cenowych reklamodawców (Li et al., 2017). W innych metodach stosuje się iteracyjną zmianę ceny w kierunku gradientu odpowiednio zbudowanej relaksacji Lagrange'a (Radovanovic and Heavlin, 2012). Podobne, iteracyjne podejście do optymalizacji cen w czasie rzeczywistym zostało zaproponowane przez Austin i in. (Austin et al., 2016). Z kolei Li i in. badają teoretyczne i empiryczne metody wpływu poziomu cen wywoławczych na zyski wydawców oferujących swoją przestrzeń jednocześnie w wielu kanałach (Li, Ni and Yuan, 2018).

### 3. Prace badawczo-rozwojowe podjęte w projekcie

Celem projektu „Prace badawczo-rozwojowe w spółce Yield Riser nad wykorzystaniem Sztucznych Sieci Neuronowych w innowacyjnym systemie do optymalizacji przychodów z reklam na witrynach w modelu RTB” było opracowanie zaawansowanej platformy wykorzystującej elementy Sztucznej Inteligencji w celu wsparcia lub nawet zastąpienia pracy optymalizatora poprzez automatyzację optymalizacji powierzchni reklamowej przy wykorzystaniu platform SSP oraz elementów Sztucznej Inteligencji.

W rezultacie tych prac badawczo-rozwojowych miała powstać platforma do automatyzacji optymalizacji powierzchni reklamowej przy wykorzystaniu platform SSP oraz elementów Sztucznej Inteligencji, stosowana w działalności spółki Yield Riser, co udało się osiągnąć. W ofercie Yield Riser Spółka ta wykorzystuje zaimplementowane na skutek wdrożenia niniejszego projektu rozwiązanie bazujące na mechanizmach Sztucznej Inteligencji, optymalizujące proces monetyzacji przestrzeni reklamowej na witrynach internetowych.

Poniżej opis prac badawczych i rozwojowych, które doprowadziły do powstania tej platformy.

#### Zadanie 1: opracowanie koncepcji projektu wykorzystania elementów Sztucznej Inteligencji przy platformach emitujących reklamy na stronach internetowych w systemie RTB

##### Problem badawczy

Istnieje wiele strategii określania cen zakupu i sprzedaży powierzchni reklamowej na rynkach RTB. Większość z nich koncentruje się na wskaźnikach „przychodów”, które mogą zostać obliczone na podstawie takich parametrów jak Ad Request eCPM lub Estimated Revenue (szacowany przychód)., Dodatkowo w naszym podejściu pod uwagę powinniśmy brać dynamikę rynku. Na koniec należy wspomnieć, że każdy system wspomagający podejmowanie decyzji musi zostać dobrze zrozumiany przez jego użytkowników - co nie ma miejsca w przypadku zaawansowanych systemów symulacji rynku opartych np. na algorytmach głębokiego uczenia się.

Konsekwentnie, problem badawczy sprowadza się do zaproponowania takiej strategii ustalania cen wywoławczych dla wydawców, która spełniałaby następujące warunki:

1. osiąga wysokie marże na poszczególnych emisjach reklam, dbając jednocześnie o (wysoki) wskaźnik porycia dla całej witryny (mierzony parametrem Coverage)
2. traktuje rynek RTB jako środowisko dynamiczne (dynamika w czasie rzeczywistym)
3. jest łatwa do zrozumienia i interpretacji przez konsultantów RTB odpowiedzialnych za ustalanie poziomów cen wywoławczych
4. jest wydajna obliczeniowo: model dla setek narzędzi może być codziennie szkolony na serwerach po uzasadnionej cenie.

Spółka Yield Riser działa po stronie podaży na rynku RTB i zarządza portalami swoich klientów (wydawców) poprzez Google AdManager (GAM). Każdego dnia nasi konsultanci monitorują efektywność narzędzi reklamowych (zwanym AdUnits) analizując wiele dostępnych danych, ale koncentrując się na trzech najważniejszych miarach efektywności: Estimated Revenue (szacunkowy przychód z danego AdUnit), Ad request eCPM (który mierzy przychód z danego umieszczenia reklamy na 1000 wyświetleń) oraz parametrze pokrycia Coverage (miara efektywności wykorzystania danego narzędzia).

W swojej codziennej praktyce nasi konsultanci tworzą tzw. reguły cenowe, aby ustalić optymalne minimalne ceny ofertowe dla swoich klientów (wydawców), w oparciu o raporty Ad Exchange i osobiste doświadczenia. Ww. „ręczny” proces jest czasochłonny i podatny na błędy. Głównym celem projektu było zautomatyzowanie tego procesu przy użyciu najnowocześniejszych, samo- optymalizujących się algorytmów sztucznej inteligencji.

Głównym problemem badawczym było sformułowanie metody określania cen wywoławczych dla powierzchni reklamowej oferowanej przez wydawców na rynku RTB. Wymagało to znalezienia kluczowego wskaźnika/wskaźników najlepiej sprawdzającego się w automatycznej optymalizacji. Dopiero kolejnym celem badawczym była identyfikacja Sztucznej Inteligencji, która najlepiej przewiduje wskaźnik/wskaźniki uznane za decydujące.

### Analiza eksploracyjna danych historycznych i pierwsze eksperymenty z cenami minimalnymi

Projekt badawczy, realizowany zgodnie z metodologią CRISP-DM (Chapman et al., 2004) został podzielony na następujące etapy: *zrozumienie danych, przygotowanie danych, modelowanie i ewaluacja oraz wdrożenie*.

Pierwszym wyzwaniem dla zespołu DataScience było zrozumienie metod działania konsultantów firmy, ich własnych spostrzeżeń w zakresie wpływu różnych czynników na efektywność sprzedaży powierzchni reklamowej oraz sposobów pomiaru efektywności biznesowej ich działań.

Aby to osiągnąć, przeprowadzono pogłębione wywiady z kluczowymi konsultantami firmy, których efektem były wstępne heurystyki.

Schemat działania wszystkich trzech uczestniczących w projekcie konsultantów YR jest podobny:

1. po 1-2 dniach sprzedaży danej reklamy na witrynie internetowej przy cenie (Floor Price) 0 zł konsultant tworzy raport, w którym bada najważniejsze z punktu widzenia optymalizacji cenowej wskaźniki (metryki), którymi są:

**Ad request** (skrót AR) - ilość wywołań reklamowych na danych tagach (tagi to słowa kluczowe np. czytadelko.pl\_970x250, jak wskazał w opisie optymalizacji jeden z konsultantów w heurystyce wstępnej)

**Coverage** - procentowa ilość sprzedanych odsłon względem wszystkich Ad Requestów

**Ad Request eCPM** - aktualna cena za 1000 Ad requestów, co jest jednym z głównych czynników decydującym o wysokości Floor Price (minimalna cena zakupu powierzchni reklamowej na danej stronie)

- znając Ad Request eCPM uruchamia raport Tag|Days|Branding types, dzięki któremu jest w stanie ustalić cenę Anonymous i Branded<sup>1</sup> dla danej jednostki.
- po wprowadzeniu przez konsultanta Floor Price dla Branded i Anonymous, konsultant obserwuje, jak zachowują się wskaźniki Ad Request eCPM oraz Coverage. Bardzo ważnym wskaźnikiem jest Coverage, czyli wypełnienie. Gdy utrzymuje się na wysokim poziomie, wtedy Ad Request eCPM jest najwyższy. Jeśli Coverage spada – to należy obniżyć Floor Price, natomiast jeśli rośnie, to można zwiększać cenę.

Kolejnym krokiem była weryfikacja heurystyk stosowanych w Firmie oraz rozwinięcie intuicji związków pomiędzy różnymi parametrami kontrolowanymi przez konsultantów YieldRiser a najważniejszymi parametrami wynikowymi, takimi jak Ad request eCPM (AREcpm) czy Coverage.

W tym celu:

- Przeprowadzono pogłębioną, eksploracyjną analizę danych historycznych** (por. Załącznik 1. Raport. Eksploracyjna analiza danych). Przeanalizowano w niej dane historyczne (1 rok) blisko 1300 tzw. tag'ów (powierzchni reklamowych), z czego głębszej analizie poddano 274 najbardziej istotne.
- Zrealizowano wiele (ponad 20) eksperymentów testujących** wrażliwość parametrów efektywności tag'ów reklamowych na zmiany cen minimalnych (por. Załącznik 2. Eksperymenty 'zero' oraz Załącznik 3. Eksperymenty 'regiony').

Po przeprowadzeniu analizy eksploracyjnej danych z raportów Google Ad Manager (GAM), **przeprowadzono setki testów A/B w celu zweryfikowania wpływu różnych strategii cen wywoławczych na kluczowe wskaźniki** (głównie Ad request eCPM i Coverage). Potwierdziły one wnioski wynikające z teorii gry: w aukcjach drugiej ceny optymalną strategią jest ustawienie ceny wywoławczej na najniższym możliwym poziomie, tak aby cena zakupu była ustalana przez samych kupujących, poprzez aukcję (Wang, Zhang and Yuan, 2017).

W ramach zadania 1 wykonano w projekcie następujące czynności:

- Przeanalizowano 1297 tagów pod względem
  - danych dziennych o przychodach, AREcpm, Coverage, liczby requestów dla danego tagu
  - ich miar zmienności (odchylenie standardowe i współczynnik zmienności)
  - danych zagregowanych per tag o poziomie przychodów i AREcpm dla Branded i Anonymous, relacji ich AREcpm, udziału Branded do Anonymous (według AR i przychodów)
  - danych zagregowanych per tag o przychodach, AREcpm, AR z zapytań z krajów UE, USA, Nowej Zelandii, Kanady i Australii („kraje premium”)

---

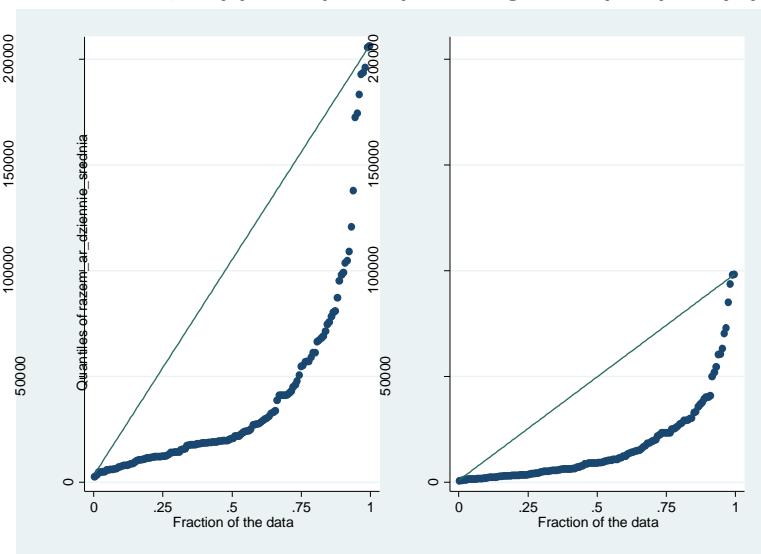
<sup>1</sup> Obecnie nie stosuje się już podziału na Anonymus i Branded, jednak taki podział istniał w trakcie prowadzenia badań



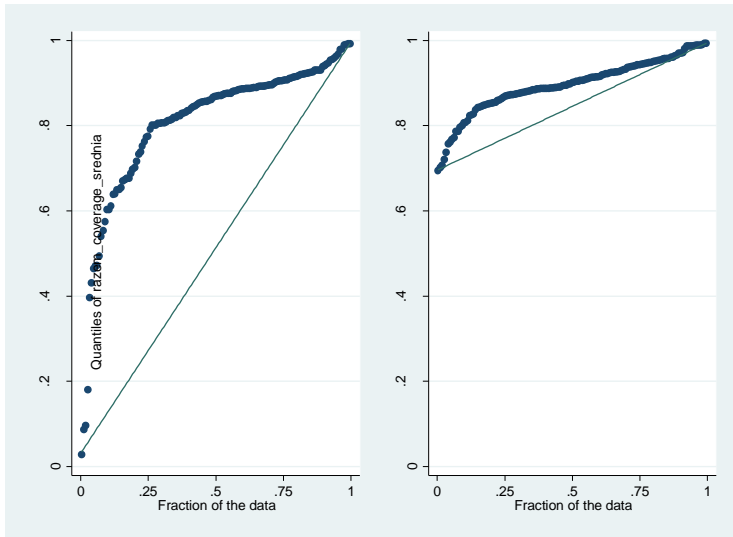
- danych zagregowanych o koncentracji przychodów per tag generowanych przez 10 największych kupujących (Advertisers)
  - danych zagregowanych o bidach, win i close CPM per tag
2. Z przeprowadzonej analizy wynikało, że istotność biznesowa tagów charakteryzuje się dużą koncentracją:
    - tylko 850 wykazało jakiegokolwiek przychody
    - 209 wykazało przychody powyżej 1000 zł
    - 110 tagów wygenerowało 80% przychodu
    - Pierwszych 10 tagów: 29% przychodów
  3. Do dalszej analizy wzięto 274 tagi o przychodach
    - powyżej 1000 zł w całym okresie
    - powyżej 80 zł w marcu 2019
  4. Analiza objęła relacje między 34 miarami opisującymi każdy z tagów. Pierwsza klasyfikacja dotyczyła podziału tagów na dwie grupy według poziomu AREcpm - powyżej i poniżej średniej (analiza z podziałem na kwartyle rozkładu dała podobne wyniki)
    - Dla tagów o AREcpm wyższym od średniej (0,40/1,24) istotnie niższa jest przeciętna dzienna liczba AR (41000/17100), ale jednocześnie wyższy coverage (0,80/0,90)

Przykładowe wyniki analiz zaprezentowano poniżej, zaś kompletne raporty można znaleźć w Załączniku 2. 3. i 4.

**DZIENNE AR (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**

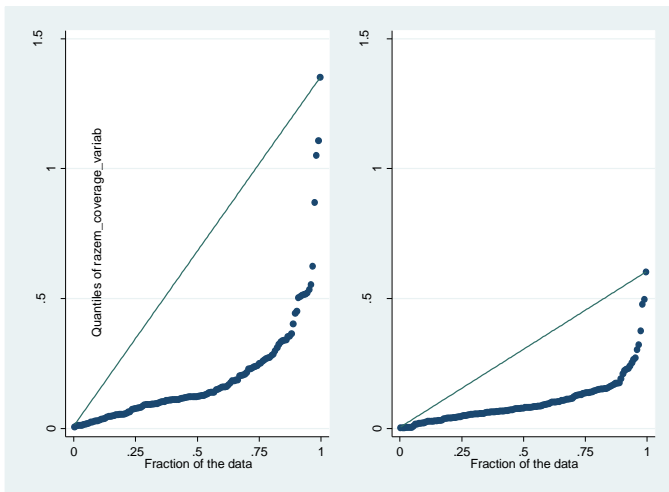


**COVERAGE (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



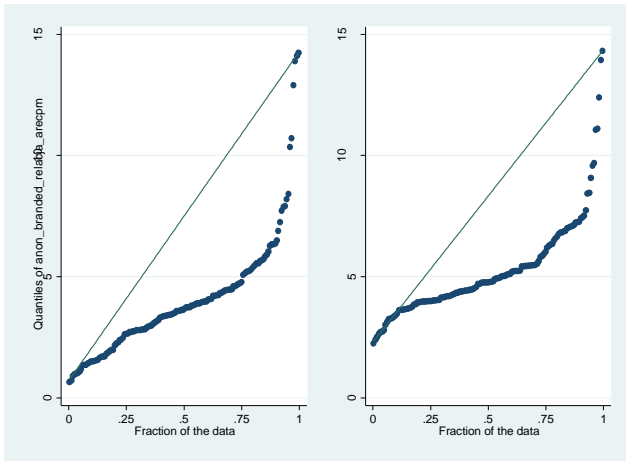
- Droższe tagi mają dużo stabilniejszy coverage (istotnie niższy współczynnik zmienności dziennej per tag)

### ZMIENNOŚĆ COVERAGE (lewy panel: poniżej średniego AREcpm; prawy: powyżej)

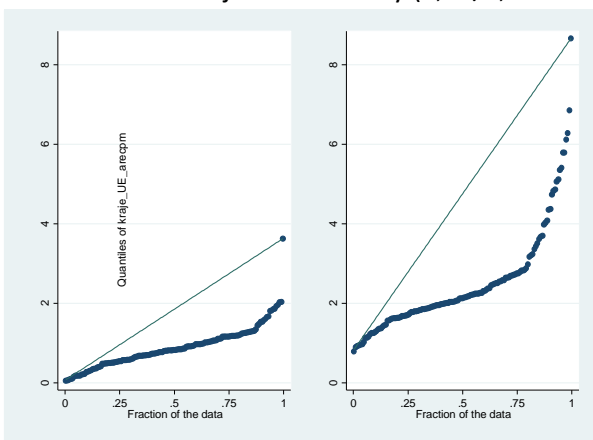


- Brak istotnych różnic w udziale Branded/Anonymous między tagami drogimi i tanimi
- W tagach drogich relacja Branded/Anonymous AREcpm dużo wyższa niż w tanich (4,07/5,54)

### RELACJA AREcpm Branded/Anonymous (lewy panel: poniżej średniego AREcpm; prawy: powyżej)

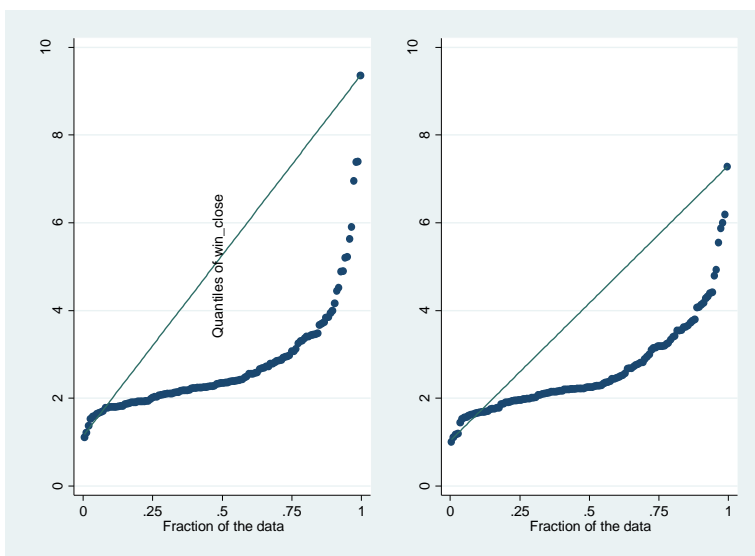


- Dla odsetek kupionych z krajów premium poziom relacji AREcpm dla tagów droższych i tanich jest zachowany (0,89/2,51 wobec 0,40/1,24 na całej próbie)



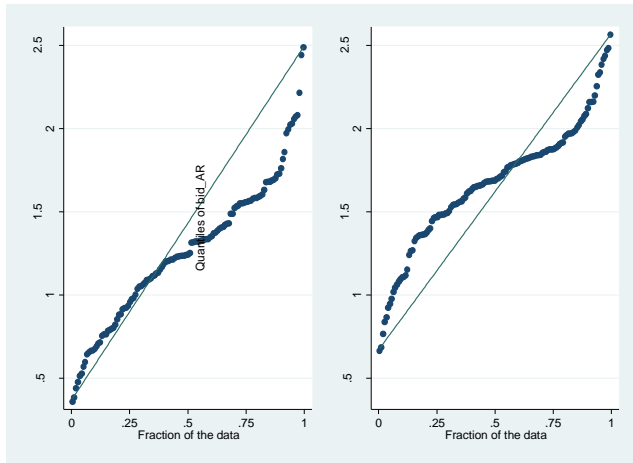
- Dla tagów droższych relacja winCPM do closeCPM jest niższa (3,18/2,25)

**RELACJA winCPM do closeCPM (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



- Relacja BIDów do AdRequestów jest natomiast znacznie wyższa dla tagów droższych (1,27/1,67)

## RELACJA BID/AR (lewy panel: poniżej średniego AREcpm; prawy: powyżej)



Badania wykazały, że kluczowym wskaźnikiem dla optymalizacji przychodów z powierzchni reklamowych na witrynach internetowych jest wskaźnik pokrycia (Coverage), który pokazuje procentową ilość sprzedanych odseton względem wszystkich Ad Requestów. Badania wykazały, że wskaźnik ten jest decydujący, ponieważ ustawiając zbyt wysoką cenę minimalną istnieje ryzyko ograniczenia wyświetlonych reklam, a tym samym zmniejszenia wskaźnika Ad Request eCPM.

## Heurystyka wynikająca z eksploracyjnej analizy danych i zrealizowanych eksperymentów

We wrześniu 2019 r. firma Google ogłosiła stopniowe przejście od aukcji drugiej ceny do aukcji pierwszej ceny, co miało silny wpływ na strategię optymalizacji cen wywoławczych. Google ograniczył liczbę możliwych do ustawiania reguł cenowych do 100, gdzie w przypadku aukcji drugiej ceny liczba ta była nieograniczona i przekraczała 1000 ustawionych reguł na koncie Yield Riser. Spowodowało to konieczność wprowadzenia pewnych poprawek co do sposobu optymalizacji np. rezygnacja z podziału na ceny Anonymous i Branded oraz konieczność przypisywania tagów do 100 reguł cenowych, zamiast możliwości ustalania odrębnych reguł dla każdego tagu.

Nawet przy zmianie strategii Google, nadal za kluczowy wskaźnik maksymalizacji zysków klientów został uznany wskaźnik pokrycia (miara efektywności wykorzystania danego narzędzia). Następnie, w oparciu o analizę raportów GAM i doświadczenia naszych yield managerów, opracowano strategię cen wywoławczych opartą na prognozie tego wskaźnika na kolejne trzy dni:

1. Jeżeli przewidywany wskaźnik pokrycia (coverage) jest wyższy niż 95%: zwiększyć cenę
2. Jeżeli przewidywany wskaźnik pokrycia wynosi od 90% do 94%: utrzymać istniejącą cenę
3. Jeżeli przewidywany wskaźnik pokrycia jest mniejszy niż 90%: należy obniżyć cenę.

Podnoszenie lub obniżenie cen realizowane jest w dyskretnych krokach, z wykorzystaniem tzw. koszyków cenowych: odpowiedniej struktury ujednociającej reguły cenowe, konfigurowalnej z poziomu interfejsu użytkownika. Jest to najczęściej od 10 do 20 poziomów o rozpiętości rzędu kilku groszy. Podniesienie/obniżenie ceny sprowadza się w efekcie do zmiany reguły cenowej z koszyka aktualnego do koszyka wyższego/niższego.

Heurystyka ta stała się w efekcie podstawą zarówno dla opracowania prototypu, jak i stworzenia docelowego rozwiązania sztucznej inteligencji w przedmiotowym projekcie.

Praktyczna realizacja tej strategii wymagała stworzenia modelu przewidywania wskaźnika pokrycia. Pierwszym etapem było pozyskanie, analiza i przygotowanie do modelowania odpowiednich szeregów czasowych (*przygotowywanie danych*). W kolejnym etapie (*modelowanie i ocena*) konieczne było przeprowadzenie analizy dostępnych obecnie modeli Sztucznej Inteligencji oraz możliwości ich wykorzystania przy projekcie. W tym zakresie wybrano, zaimplementowano i oceniono różne algorytmy prognozowania szeregów czasowych z zastosowaniem Sztucznej Inteligencji. Najlepszy z nich został zaimplementowany w systemie produkcyjnym z dedykowanym dashboardem Business Intelligence (BI), dla danego narzędzia (reklamy), wizualizującym wybrane parametry GAM, wiele dodatkowych zbiorów danych, szeregi czasowe z prognozami i rekomendacjami obniżenia, utrzymania lub zwiększenia ceny wywoławczej. Szczegóły dotyczące metod stosowanych na tych etapach zostały przedstawione poniżej, a wyniki badań w dalszej części niniejszego opracowania.

### Analiza dostępnych modeli Sztucznej Inteligencji oraz możliwości ich wykorzystania w projekcie

W ramach prac badawczych z udziałem naukowców, specjalnie zatrudnionych przez YieldRiser do tego projektu (Andrzej Wodecki – Politechnika Warszawska) zostały przeanalizowane dostępne modele Sztucznej Inteligencji pod kątem możliwości ich zastosowania w projekcie. Wybrano najlepiej oceniane algorytmy trenowania modeli prognozujące szeregi czasowe, takie jak:

1. Metody bazowe (naiwna, średnia)
2. Modele autoregresyjne: ARIMA i SARIMA
3. Modele z wygładzaniem eksponentialnym (ESM)
4. Modele wykorzystujące sieci neuronowe:
  - a. Perceptrony wielowarstwowe (MultiLayer Perceptrons – MPLs)
  - b. Sieci konwolucyjne (Convolutional Neural Nets – CNN)
  - c. Sieci rekurencyjne z pamięcią długo i krótkoterminową (Long-Short term Memory Recurrent Neural Nets - LSTM)

### Przygotowanie danych i inżynieria

Jak wspomniano powyżej, głównym źródłem danych był Google AdManager (GAM), który dostarcza wiele konfigurowalnych raportów (wszystkie definicje, wymiary i miary zob. np. (Ad Exchange historical report - Google Ad Manager Help, 2020) Przykładowymi wymiarami są kupujący (np. marki i ich wertykalia), segmenty zapasów (w tym reguły cenowe, strony, rozmiary i kanały), geografia (kraje), szczegóły dotyczące urządzeń mobilnych (urządzenia, systemy operacyjne) oraz czas (dni, tygodnie i miesiące, 2 lata wstecz). Dla każdego z tych wymiarów możemy otrzymać wiele różnych miar, w tym najważniejsze - zapytania ofertowe (wszystkie, dopasowane i CTR), szacowany koszt dotarcia (areCPM), matched eCPM i Coverage.

Dane mogą być pozyskiwane bezpośrednio z interfejsu GAM lub poprzez interfejs programowania aplikacji (API). W niniejszej analizie (jednowymiarowa analiza i planowanie szeregów czasowych) wykorzystujemy podzbiór dostępnych danych reprezentatywny dla niskich, średnich i wysokich wartości pokrycia oraz jeden szereg czasowy *Coverage*.

Mimo że dane uzyskane z GAM są bardzo klarowne, najważniejszą i najbardziej czasochłonną częścią tego projektu było przygotowanie danych. Proces podzielono na dwie fazy: przygotowanie zestawów danych do prognozowania szeregów czasowych oraz inżynieria funkcji dodatkowej analizy, która następnie została wyświetlona w dedykowanym dashboardzie BI.

## Modelowanie

Wynikiem opracowanej na tym etapie projektu heurystyki było w szczególności wskazanie modeli predykcyjnych (prognozowanie szeregów czasowych) jako podstawy dla rekomendacji poziomów cenowych. Z tego względu w realizowanych w zadaniu 1 pracach w pierwszej kolejności skoncentrowano się na identyfikacji algorytmów w najlepszy możliwy sposób sprawdzających się na zbiorach danych generowanych przez systemy YR. Poniżej prezentujemy najważniejsze wnioski wynikające z przeprowadzonych na tym etapie analiz.

Głównym celem fazy modelowania było określenie najlepszego możliwego algorytmu prognozowania szeregów czasowych dla parametru (wskaźnika pokrycia) Coverage. W naszej analizie zbadaliśmy szerokie spektrum algorytmów prognostycznych. W naszym projekcie stosujemy tylko tzw. *metody proste, auto-regresywne (ARIMA i SARIMA), Exponential Smoothing (wygładzanie wykładnicze) oraz algorytmy Facebook Prophet*, ograniczając ich zastosowanie do jednoetapowego prognozowania. Poniżej przedstawiamy ww. metody, zaś wyniki oceny modelowania opisano w kolejnej sekcji.

Istnieją dwie różne proste (naïve) strategie prognozowania (więcej informacji (Brownlee, 2018a): naïve gdy wcześniejsze obserwacje są wykorzystywane bezpośrednio do prognozowania przyszłych wartości i zwykła, gdy do prognozowania przyszłości wykorzystuje się różne statystyki z wcześniejszych obserwacji. „Naiwna” prognoza polega na wykorzystaniu poprzednich obserwacji bezpośrednio jako prognozy, bez wprowadzania żadnych zmian (tzw. prognoza uporczywa). W przypadku, gdy chcemy skorygować prognozę o wpływy sezonowe, możemy wyważyć dane historyczne o tzw. sezonowe time lags. I tak np. przy założeniu sezonowości = 7 dni (jak w naszym przypadku), jako prognozę uporczywą można przyjąć obserwację opartą o okres 7 dni wstecz. Strategia uśrednionej (zwykłej) prognozy jako prognozę wykorzystuje uśrednioną wcześniejszą wartość. Wszystkie wcześniejsze obserwacje są zbierane i uśredniane, za pomocą średniej lub mediany, bez innych operacji na danych (Brownlee, 2018a). Liczba poprzednich uśrednionych obserwacji jest kontrolowana przez parametr modelu, który później może być przeszukiwany (tzw. grid search) w ramach procedury optymalizacji modelu. Ponieważ *proste metody* są łatwe do zrozumienia i wdrożenia, są one doskonałymi kandydatami do modelu wzorcowego (ang. benchmark model.). W naszym przypadku, wykonaliśmy więcej pracy dzięki przestrzeni hiper-parametrycznej dla metod zarówno „naiwnych” (naïve), jak i uśrednionych, aby znaleźć najlepszy prosty model.

Modele typu Auto Regressive Integrated Moving Average (ARIMA) okazały się bardzo skuteczne w modelowaniu szeregów czasowych (parametru) Coverage. W skrócie, *autoregresja* oznacza, że model wykorzystuje zależność między obserwacją a pewną liczbą opóźnionych obserwacji (ang. lagged observations), *integracja* oznacza różnicowanie surowych obserwacji w celu „unieruchomienia” szeregów czasowych, a *średnia ruchoma* wykorzystuje zależność pomiędzy obserwacją a szacunkowym błędem z modelu średniej ruchomej mającego zastosowanie względem opóźnionych obserwacji.

Każdy z tych aspektów został odzwierciedlony w parametrach modelu ARIMA. Przy standardowej notacji w postaci ARIMA ( $p, d, q$ ),  $p$  oznacza liczbę opóźnionych obserwacji w modelu (nazywaną również *kolejnością opóźnień*) (ang. lag order),  $d$  jest liczbą razy odzwierciedlającą zróżnicowanie obserwacji w rzędzie (*stopień zróżnicowania*), a  $q$  jest wielkością okna średniej ruchomej.

W modelu ARIMA skonstruowano model regresji (Brownlee, 2018b), z wykorzystaniem przygotowanych danych, różniących się w szczególności tym, że szereg czasowy jest nieruchomy. Różnicowanie (ang. differencing) jest stosunkowo prostą transformacją stosowaną względem szeregów czasowych, która eliminuje trendy i sezonowe struktury poprzez obliczanie różnicy pomiędzy kolejnymi obserwacjami:

$$y'_t = y_t - y_{t-1}$$

Czasami proces ten może być powtarzany - liczba tych iteracji nazywana jest *kolejnością różnicującą*.

Rozszerzenie SARIMA (Hipel and McLeod, 1994) obsługuje bezpośrednio modelowanie sezonowej składowej (danego) szeregu. Wykorzystuje różnicowanie w celu usunięcia dodatkowych sezonowych wpływów (czynników). W tym przypadku wprowadza się cztery dodatkowe parametry sezonowe: *P*: sezonowa kolejność autoregresywna, *D*: sezonowa kolejność różnicowa, *Q*: sezonowa średnia krocząca i *m*: liczba etapów czasowych dla jednego okresu sezonowego. Parametry sezonowe mogą być zidentyfikowane poprzez staranne badanie wykresów autokorelacji (korelacje ostatnich etapów czasowych).

*Exponential smoothing (wygładzanie wykładnicze)* jest metodą prognozowania szeregów czasowych dla danych jednorodnych, która może zostać poszerzona o dane z regularnym trendem lub składową sezonową (Brownlee, 2018a). W tych metodach prognoza jest liniową sumą ważoną ostatnich obserwacji z przeszłości, ale w przeciwieństwie do rodziny metod Box-Jenkins ARIMA, model wyraźnie wykorzystuje wykładniczo malejące znaczenie obserwacji z przeszłości. W tym badaniu wdrażamy *potrójne wygładzanie wykładnicze (ETS) dla ES*, znane również jako *Holt-Winters ES*. Szczegółowy opis i lista hiper-parametrów modelu np. (Exponential smoothing, 2020).

*Facebook Prophet* to projekt typu open-source skupiający się na prognozowaniu szeregów czasowych, oparty na modelu addytywnym, w którym nieliniowe trendy są dopasowane do sezonowości rocznej, tygodniowej i dziennej oraz wpływu wakacji.

Ogólna procedura dla wszystkich zaimplementowanych algorytmów Sztucznej Inteligencji była następująca. Po załadowaniu zbioru danych przygotowaliśmy zbiór testowy: podzieliliśmy zbiór danych szkoleniowych i zbiór danych testowych w proporcji 9:1 oraz zastosowaliśmy procedurę walidacji kroczącej (bardzo skuteczną w przypadku oceny modelu prognozowania szeregów czasowych). Następnie zdefiniowano model i skonfigurowano odpowiednią (zależną od modelu) przestrzeń hiper-parametru oraz wykonano wyszukiwanie metodą przeszukiwania siatkowego (ang. grid-search). Na koniec oceniono modele (wykorzystując RMSE jako miernik jakości) i zidentyfikowano najlepiej funkcjonującą konfigurację modelu.

### *Ocena modelu i wybór najlepszego modelu*

Z perspektywy uczenia maszynowego kluczowym modelem determinującym jakość rozwiązania jest model predykcyjny (a dokładniej: predykcji szeregu czasowego dla wskaźnika Coverage w oparciu o jego wartości historyczne dla wskazanego tag'u).

W efekcie, głównym celem etapu była identyfikacja algorytmu trenowania modelu predykcyjnego w *najlepszy możliwy sposób* prognozującego przyszłe wartości współczynnika Coverage. Jakość algorytmu oznaczała przy tym nie tylko jakość predykcji modelu, ale też czas jego trenowania (docelowy system powinien mieć możliwość efektywnego, regularnego „dotrenowywania” modelu w miarę napływających danych historycznych).

Po przeprowadzeniu podstawowej analizy szeregów czasowych, w tym autokorelacji i kontroli stacjonarności, w celu określenia najlepszego modelu prognozowania wdrażamy, punktujemy i oceniamy opisane powyżej modele Sztucznej Inteligencji. Mając za miernik jakości prognozy Root Mean Square Error (RMSE) (średnia kwadratowa błędów), naszym celem jest zidentyfikowanie najlepszego z nich. W przypadku, gdy uzyskamy zadowalającą prognozę (błąd prognozowania na

poziomie 10% Coverage lub niższym), proponujemy ją jako pierwszą, prostą metodę ustalenia reguły wyceny.

Jako metrykę ewaluacji dla modeli prognozowania szeregów czasowych proponujemy RMSE (Root Mean Squared Error) jako łatwy do zrozumienia miernik jakości prognozowania:

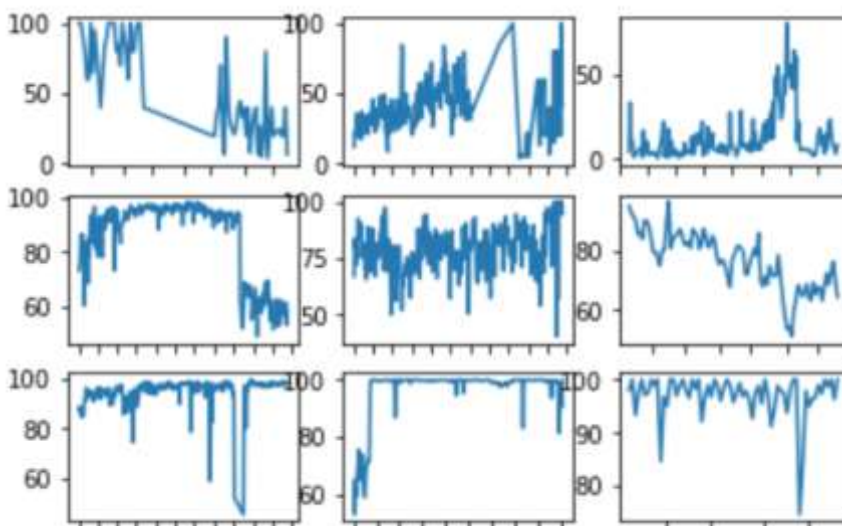
$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2} \quad (1)$$

gdzie  $y$  i  $\hat{y}$  stanowią obserwowane i przewidywane wartości szeregów czasowych. Stąd, RMSE, będący pierwiastkiem kwadratowym średniej wartości z różnic podniesionych do kwadratu między prognozą a rzeczywistą obserwacją, mierzy średnią wielkość błędu.

W porównaniu z alternatywną miarą - średnim bezwzględnym błędem (MAE):  $\frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|$ , RMSE nadaje stosunkowo dużą wagę poważnym błędom (błędy są podnoszone do potęgi przed uśrednieniem). Ponieważ w naszej analizie poważne błędy są niepożądane, wybieramy RMSE jako model metryczny do prognozowania wyników.

W celu oceny modelu stosuje się *procedurę walidacji kroczącej* (Brownlee, 2018b). Po pierwsze, zestaw danych jest dzielony na zestaw danych szkoleniowych i zestaw danych testowych. Następnie, poczynając od początku szeregu czasowego, do przeszkolenia modelu wykorzystuje się minimalną liczbę próbek, dokonuje się prognozy na kolejny etap czasowy z wykorzystaniem przeszkolonego modelu i kolejno ocenia się prognozę na podstawie znanej wartości. Na koniec, okno czasowe jest rozszerzane o znaną wartość i proces jest powtarzany.

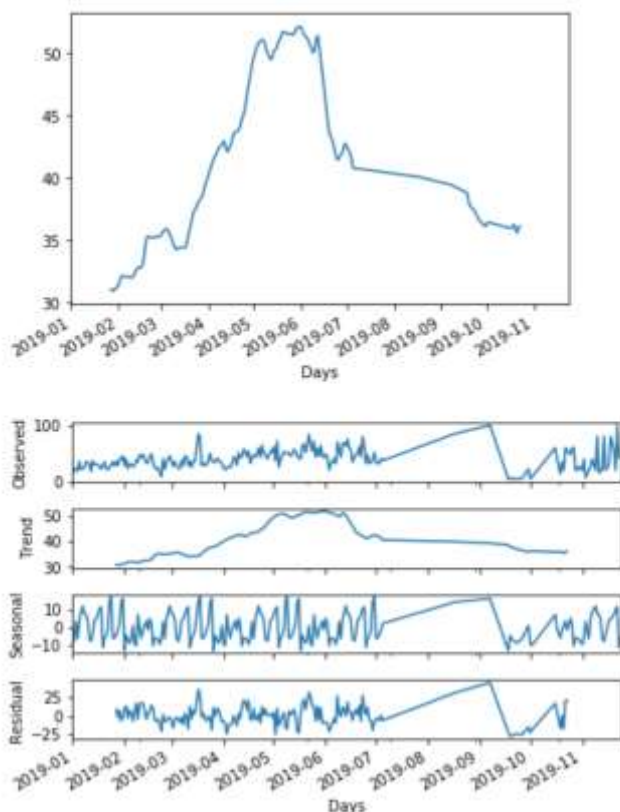
W pierwszej kolejności przeprowadzono analizę danych rozpoznawczych (EAO). Zbiór danych, z szeregiem czasowym dla parametru Coverage o wielkości około 600 AdUnits, został podzielony na trzy tercyle obejmujące niski, średni i wysoki poziom wskaźnika pokrycia Coverage. Następnie wybrano losowo trzech przedstawicieli dla każdej kategorii. Wizualizacja poglądowa ww. reprezentatywnych AdUnits przedstawiona została na Rys. 1.



Rys. 1. Wykres liniowy szeregów czasowych wskaźnika pokrycia dla niskich, średnich i wysokich AdUnits wskaźnika. Źródło: obliczenia własne.

Dodatkowo wykonaliśmy dekompozycję szeregów czasowych wskaźnika pokrycia - jej przykład znajduje się na Rys. 2.





Rys. 2. Dekompozycja szeregów czasowych wskaźnika pokrycia dla próbnego AdUnit. Źródło: obliczenia własne

Jak widać, w ramach średniej trudno jest zidentyfikować jakiejkolwiek trendy czy sezonowość (co jest oczywiste biorąc pod uwagę, że AdUnits należą do bardzo różnych kategorii stron internetowych). W większości próbnych AdUnits występują pewne oczywiste wyjątki. W celu sprawdzenia stacjonarności szeregu czasowego wykonaliśmy serię testów AdFuller (ADF). W większości okazują się one być stacjonarne: dla czterech (z dziewięciu) AdUnits statystyki ADF są większe niż wartość krytyczna wynosząca 1%, a więc te zbiory danych nie wydają się być stacjonarne. Ale na poziomie 5% większość szeregów czasowych jest stacjonarna. Wykresy autokorelacji wykazały, że z małymi wyjątkami nie obserwujemy znaczącej autokorelacji.

Po przeanalizowaniu danych i przygotowaniu ich do modelowania, zgodnie z naszą metodologią badawczą, rozpoczęliśmy od identyfikacji modelu wzorcowego. W tym celu wybraliśmy model prognozowania uporczywego: prostą heurystykę, w której wartość z etapu "t" jest wykorzystywana jako prognoza dla etapu t+1. RMSE dla ww. modelu bazowego, uśredniony dla 9 AdUnits = 11,20.

Następnie zastosowaliśmy różne techniki modelowania szeregów czasowych dla dziewięciu reprezentatywnych AdUnits. Zaczęliśmy od modeli auto-regresyjnych. Skupiliśmy się na modelu ARIMA, dostosowanym pod względem sezonowym. Przeszukano następujące wartości parametrów:  $p = (0, 1, 2)$ ;  $d = (0, 1)$ ;  $q = (0, 1, 2)$ ;  $t = 'n'$  (brak trendu),  $'c'$  (stała),  $'t'$  (liniowa) i  $'ct'$  (stała z trendem liniowym);  $P = (0, 1, 2)$ ;  $D = (0, 1)$ ;  $Q = (0, 1, 2)$ ,  $m = (0, 7)$ . Przeszukiwanie metodą grid search tej przestrzeni parametrów wykazało, że najlepszy model ma następującą konfigurację: Model (2, 1, 2), (0, 0, 2, 7), 'n', gdzie:

1. Dla elementów trendów:
  - a.  $p = 2$  < auto-regresja trendu
  - b.  $d = 1$  < różnica trendu

- c.  $q = 2 <$  średnia krocząca trendu
2. Dla elementów sezonowych:
- a.  $P = 0 <$  sezonowa auto-regresja
  - b.  $D = 0 <$  różnica sezonowa
  - c.  $Q = 0 <$  sezonowa średnia krocząca
  - d.  $m = 7 <$  liczba etapów czasowych dla pojedynczego okresu sezonowego.
  - e.  $t = 'n' <$  brak parametru trendu.

To RMSE (uśrednione dla 9 jednostek AdUnits, z wyłączeniem jednej wartości „odstającej”) zaokrąglone do 8,48. Podsumowując: Model SARIMA znacząco poprawia nasz benchmark, choć obliczenia były niezwykle czasochłonne.

Następnie oceniliśmy modele Exponential Smoothing (ETS) (wyrównanie wykładnicze). Ponieważ obliczenia ETS są również niezwykle czasochłonne, nie przeprowadziliśmy systematycznego wyszukiwania typu grid search, lecz wypróbowaliśmy różne ręczne konfiguracje. Około 10 eksperymentów nad tymi modelami wykazało najlepszy model z następującą konfiguracją hiperparametru:  $t =$  Brak (trend jest spadkowy);  $d =$  Fałszywy (trend nie powinien być tłumiony);  $s =$  dodatek (składnik sezonowy jest addytywny);  $p = 7$  (okres sezonowy = 7 etapów czasowych) i  $b =$  fałszywy (nie wykonujemy przekształcenia mocy (ang. power transform) dla szeregu). RMSE (uśrednione dla 9 AdUnits) = 8,37, co jest korzystniejsze, niż nasz benchmark i modele SARIMA. Inne ustawienia modelu, z nieco gorszymi (ale porównywalnymi) RMSE, wykonano z  $t =$  "dodaj" (trend dodatni),  $s =$  "mul" (sezonowość mnoga (multiplikatywna)), oraz różnymi wartościami okresu sezonowego  $p$  (1, 2, 20). Wnioski: Model ETS znacząco poprawia nasz wzorzec (benchmark), a jego jakość jest porównywalna z najlepszym modelem SARIMA.

Na koniec, oceniliśmy sztuczną inteligencję Facebook Prophet. Co ciekawe, jej podstawowa konfiguracja spowodowała, że RMSE (uśrednione dla 9 jednostek AdUnits) = 10,6, co jest gorszym wynikiem niż ETS i SARIMA, przy czym „reprezentanci” AdUnits przy niskim współczynniku pokrycia Coverage znacząco „odstają” (w ujęciu RMSE). Obliczenia dla reprezentantów średniego i wysokiego współczynnika pokrycia, w połączeniu z diagnostyką natywną biblioteki *fbprophet* (z procedurą walidacji krzyżowej, opisaną w (Prophet Diagnostics, 2020) przyniosły bardziej obiecujące wyniki: wynik wyniósł RMSE = 6,71.

Ocenę wszystkich ocen modeli progностycznych szeregów czasowych przedstawiono w tabeli 1.

Model	RMSE
Linia odniesienia (uporczywość)	11,20
SARIMA	8,48
ETS	8,37
Facebook Prophet (współczynnik pokrycia średni i wysoki)	6,71

Tabela 1. Ocena modeli prognozowania szeregów czasowych. Źródło: Opracowanie własne.

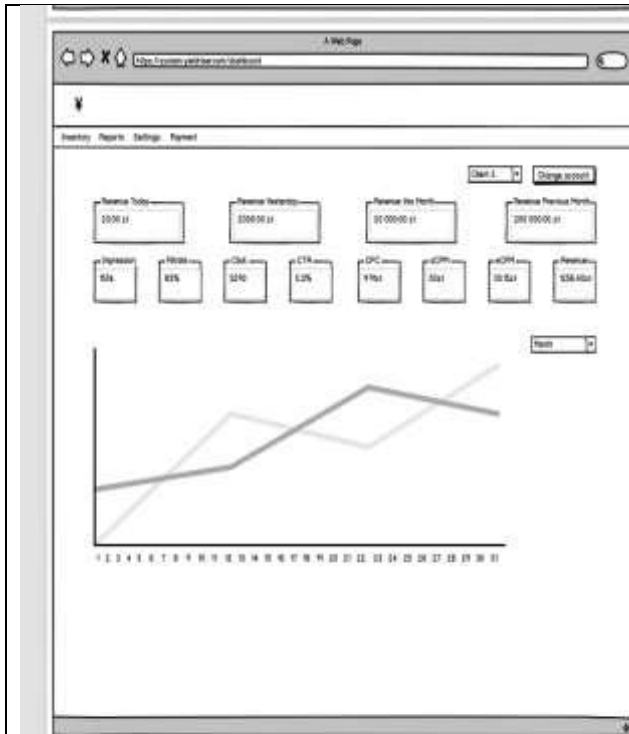
Ponieważ heurystyka przy ustalaniu minimalnych cen ofertowych uwzględnia wysokie (>90%) wartości pokrycia Coverage i szkolenie modelu Facebook Prophet odbyło się znacznie szybciej niż w przypadku SARIMA i ETS, uznaliśmy Facebook Prophet za najbardziej optymalną Sztuczną Inteligencję do celów niniejszego projektu.

## Zadanie 2: Zaprojektowanie i zbadanie platformy wykorzystującej rozwiązania z etapu 1 do celów usprawnienia pracy osób odpowiedzialnych za optymalizację systemów SSP wewnątrz przedsiębiorstwa

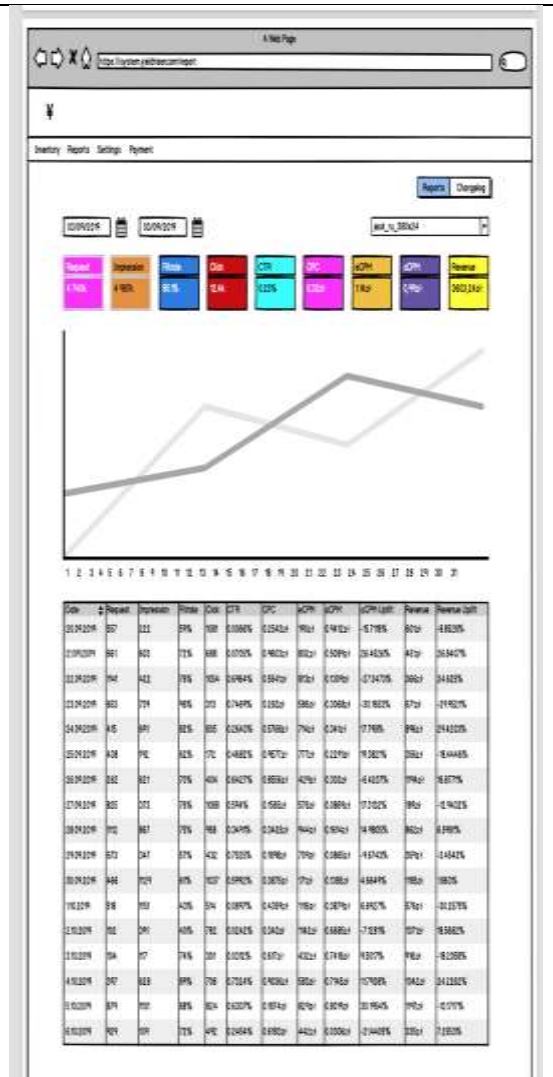
W ramach Zadania nr 2 zaprojektowano platformę wykorzystującą efekty badań z Zadania 1 umożliwiającą użytkownikom – konsultantom YR oraz klientom YR – wydawcom posiadającym swoje własne konto Google ADX, intuicyjną obsługę panelu tzn. niewymagającą znajomości języków programowania.

W pierwszym kroku zaprojektowano interfejs do obsługi platformy przez pracowników YR, w tym celu wykonano makiety graficzne, przy wykorzystaniu najlepszych praktyk z dziedziny User Experience. Przeprowadzono konsultacje z pracownikami Yield Riser w celu optymalnego zaprojektowania modelu obsługi panelu przez użytkowników i zbadano zapotrzebowanie na rodzaj i sposób wyświetlania danych.

Wynikiem przeprowadzonych z pracownikami YR konsultacji były makiety systemu, dostępne w Załączniku 3. Wybrane ekrany prezentują schematy poniżej:



Unit Name	Unit ID	Unit ID	Revenue	Expenses	Profit	Expenses Profit
Unit_1	100000	100000	1000	100	900	0.1
Unit_2	200000	200000	2000	200	1800	0.1
Unit_3	300000	300000	3000	300	2700	0.1
Unit_4	400000	400000	4000	400	3600	0.1
Unit_5	500000	500000	5000	500	4500	0.1
Unit_6	600000	600000	6000	600	5400	0.1
Unit_7	700000	700000	7000	700	6300	0.1
Unit_8	800000	800000	8000	800	7200	0.1
Unit_9	900000	900000	9000	900	8100	0.1
Unit_10	1000000	1000000	10000	1000	9000	0.1
Unit_11	1100000	1100000	11000	1100	9900	0.1
Unit_12	1200000	1200000	12000	1200	10800	0.1
Unit_13	1300000	1300000	13000	1300	11700	0.1
Unit_14	1400000	1400000	14000	1400	12600	0.1
Unit_15	1500000	1500000	15000	1500	13500	0.1
Unit_16	1600000	1600000	16000	1600	14400	0.1
Unit_17	1700000	1700000	17000	1700	15300	0.1
Unit_18	1800000	1800000	18000	1800	16200	0.1
Unit_19	1900000	1900000	19000	1900	17100	0.1
Unit_20	2000000	2000000	20000	2000	18000	0.1
Unit_21	2100000	2100000	21000	2100	18900	0.1
Unit_22	2200000	2200000	22000	2200	19800	0.1
Unit_23	2300000	2300000	23000	2300	20700	0.1
Unit_24	2400000	2400000	24000	2400	21600	0.1
Unit_25	2500000	2500000	25000	2500	22500	0.1
Unit_26	2600000	2600000	26000	2600	23400	0.1
Unit_27	2700000	2700000	27000	2700	24300	0.1
Unit_28	2800000	2800000	28000	2800	25200	0.1
Unit_29	2900000	2900000	29000	2900	26100	0.1
Unit_30	3000000	3000000	30000	3000	27000	0.1
Unit_31	3100000	3100000	31000	3100	27900	0.1



Szczególną uwagę poświęcono pulpitowi prezentacji danych. Po kilku iteracjach przybrał on formę zaprezentowaną poniżej:



Rys. 4. Interaktywny dashboard z zaleceniami opartymi na prognozie współczynnika pokrycia Coverage. Źródło: Opracowanie własne.

Na tym etapie dokonano analizy i doboru najefektywniejszego środowiska programistycznego do wykorzystania w odwzorowaniu projektu teoretycznego z zadania 1. W rezultacie jako język programowania wybrano Python, gdyż najlepiej odpowiadał założonym celom.

W toku prowadzonych prac utworzono dwa modele podawania danych do analizy:

1. wgrzywanie raportów
2. importowanie danych przez API po podaniu danych logowania, gdyż system musi umożliwiać importowanie danych w sposób manualny oraz automatyczny poprzez połączenie z API Google AdExchange.

W toku prac nad platformą uzgodniono, że domyślnym sposobem będzie połączenie poprzez API, natomiast w związku z pracą na wielu kontach ADX, w tym na kontach wydawców, niektórzy z nich nie udostępnią danych logowania zewnętrznej firmie. W takim wypadku niezbędna jest możliwość wgrania wygenerowanego przez wydawcę raportu.

W przypadku API, platforma ma możliwość zapamiętywania danych logowania wielu kont, w związku z działaniami prowadzonymi przez Yield Riser na wielu kontach Google ADX.

Ponieważ ustalenie zakresu danych do analizy przez Sztuczną Inteligencję determinowało wybór najlepszego rozwiązania z zakresu SI, część prac przewidzianych we wniosku o dofinansowanie dla Zadania 2, konieczne było do wykonania już w Zadaniu 1 (przeprowadzenie prac badawczych nad najlepszym wskaźnikiem/wskaźnikami do procesu automatyzacji optymalizacji powierzchni reklamowej witryn internetowych).

### Zadanie 3: Opracowanie prototypu platformy do automatyzacji powierzchni reklamowej przy wykorzystaniu platform SSP oraz elementów Sztucznej Inteligencji

#### Ogólny opis prototypu platformy

W ramach zadania nr 3 opracowano platformę Yield Riser AI Optimization System (YAOS – System Optymalizacyjny Yield Riser oparty o mechanizmy sztucznej inteligencji). **Oprogramowanie Yield Riser AI Optimization System (YAOS) w fazie wstępnej/testowej było prototypem platformy docelowej.**

W wyniku realizacji projektu stworzony został system, który w oparciu o mechanizmy sztucznej inteligencji, od pierwszego uruchomienia analizuje i uczy się zachowania reklamodawców oraz sugeruje wprowadzenie zmian.

Yield Riser AI Optimization System jest usługą sieciową, która pozwala wydawcom internetowym zwiększać ich dochody pochodzące ze sprzedaży przestrzeni reklamowych na ich stronach internetowych. System przetwarza dane z Google AdManager (platformy RTB). Na podstawie danych System Yield Riser przewiduje zmiany parametrów opisujących jednostki reklamowe (ad units), następnie rekomenduje ustalenie optymalnego poziomu ceny minimalnej miejsc reklamowych na stronie wydawcy. Oprogramowanie ma charakter oprogramowania samodzielnego, wykorzystującego i bazującego na mechanizmach sztucznej inteligencji.

Niezbędne do tego celu dane przechowywane są w bazach Google AdManager (GAD). System pobiera dane, czyści je, przelicza dodatkowe parametry nieuwzględnione w raporcie GAD, dokonuje przewidywania i rekomenduje podjęcie decyzji dla użytkownika końcowego, aby zwiększyć, zmniejszyć lub utrzymać cenę minimalną.

#### Moduł Sztucznej Inteligencji

Moduł sztucznej inteligencji dokonuje przewidywań za pomocą metody szeregów czasowych, uczy się na podstawie danych z przeszłości i obecnie występujących trendów. Oprócz rekomendacji system wyświetla interaktywny raport (Dashboard) z danymi, które pomagają podjąć ostateczną decyzję dotyczącą ustalenia ceny minimalnej za miejsce reklamowe.

Jako algorytm predykcyjny będący podstawą systemu rekomendującego poziomy cenowe wybrano Facebook Prophet. Jest on jednym z wiodących modułów sztucznej inteligencji, udostępniany na licencji Creative Commons. Celem projektu było bowiem wykorzystanie Sztucznej Inteligencji do optymalizacji przychodów z reklam. Cel ten udało się osiągnąć.

Wykorzystany algorytm Sztucznej Inteligencji dekomponuje szereg czasowy do trzech składowych: trendu, sezonowości i świąt:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t.$$

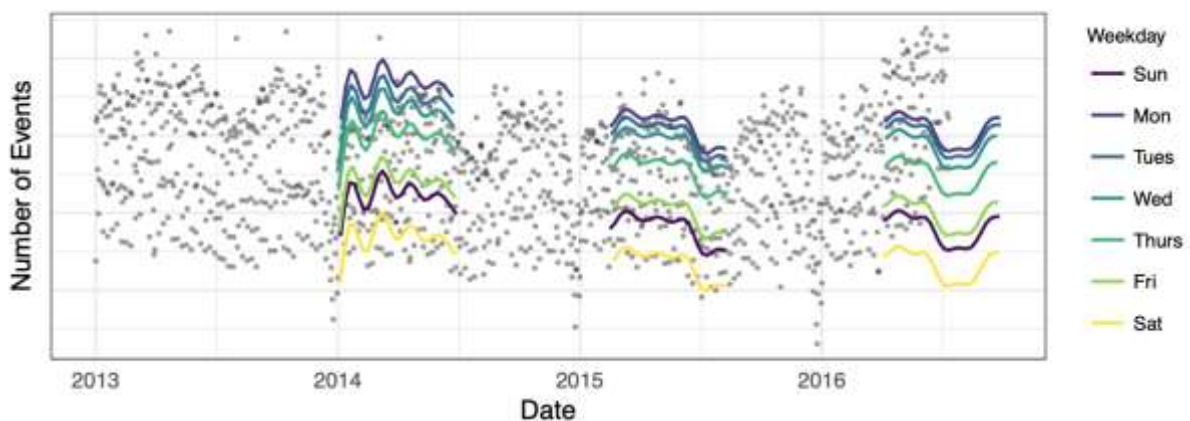
W równaniu tym  $g(t)$  reprezentuje trend (zmiany nie-okresowe),  $s(t)$  modeluje sezonowość (np. tygodniową czy roczną, zaś  $h(t)$  reprezentuje efekty świąt lub okresów wolnych od pracy (nieregularne, trwające do kilku dni). Na koniec,  $\epsilon_t$  błąd nie odwzorowany przez model, posiadający rozkład normalny.

Do modelowania trendu zastosowano dwa podejścia: model wysyczonego wzrostu oraz modeli liniowy z punktami zmiany. Efekty periodyczne (sezonowość)  $s(t)$  są aproksymowane z wykorzystaniem szeregow Fouriera, zaś komponent efektów świąt jest konfigurowalny: analityk może samodzielnie wprowadzić odpowiednie daty jako parametr algorytmu.

Zastosowane w tej metodzie podejście addytywne ma, w porównaniu do innych uznanych metod takich jak np. generatywne modele ARIMA, wiele zalet. W szczególności jest elastyczne, nie wymaga interpolacji pustych wartości, trenowanie jest bardzo szybkie (co jest bardzo ważne w implementacji praktycznej), zaś sam model jest stosunkowo prosty w interpretacji, zaś jego parametry zrozumiałe dla analityków.

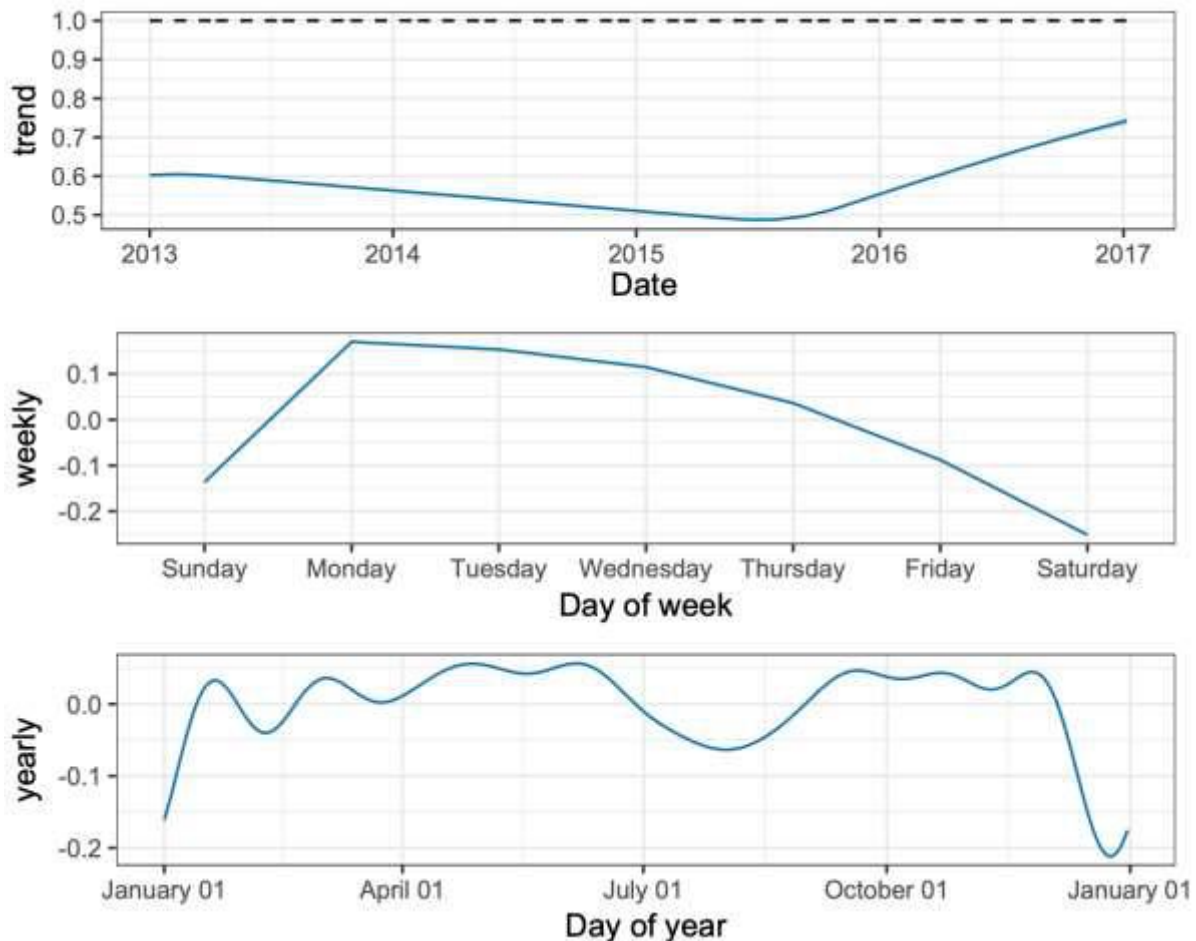
Analitycy wykorzystujący model w praktyce mogą go dostosować do własnych potrzeb korzystając z wielu prostych w interpretacji parametrów. W szczególności, algorytm umożliwia kontrolę zmiennych takich jak potencjał rynku, daty istotnych zmian (np. uruchomienie nowej kampanii czy wprowadzenie produktu na rynek), święta i dni wolne od pracy czy parametry wygładzania modelu.

Addytywne podejście do modelowania szeregu czasowego umożliwia stosunkowo prostą interpretację jego wyników. Poniższy rysunek przedstawia przykładowe wyniki.



Rysunek 5. Wyniki modelowania szeregów czasowych

Jak widać, wykorzystany algorytm Sztucznej Inteligencji stosunkowo dobrze modeluje sezonowość roczną oraz tygodniową, uwzględniając przy tym efekty świąt. Co więcej, umożliwia bardziej zaawansowaną analitykę: dekompozycję na składowe addytywne trendu oraz sezonowości tygodniowych i rocznych (por. Rysunek 6).



Rysunek 6. Składowe prognozy SI.

Sposób działania zaimplementowanego ostatecznie w środowisku produkcyjnym modułu sztucznej inteligencji jest następujący:

1. W pierwszym kroku, wykorzystując dedykowane skrypty, importujemy dane o aktualnych wartościach parametru Coverage wszystkich tag'ów.
2. Zaktualizowane szeregi czasowe są przekazywane do modułu predykcyjnego, który:
  - a. Jest w oparciu o nie „dotrenowywany” (w rytmie dziennym, z wykorzystaniem metody *fit* pakietu FB Prophet)
  - b. Po wytrenowaniu zwraca prognozy parametru Coverage na najbliższe 3 dni (metoda *predict* pakietu FB Prophet).
  - c. Czynność ta jest realizowana „off-line”, najczęściej w nocy.
3. Na podstawie tak obliczonych predykcji i zgodnie z heurystyką opisaną w poprzednim punkcie system dla każdego z tag'ów rekomenduje podniesienie ceny, pozostawienie jej na obecnym poziomie lub obniżenie. Bezpośrednim rezultatem jest rekomendacja zmiany (bądź nie) aktualnego koszyka cenowego wybranego TAG'u.



4. Rekomendacje te są przekazywane do pulpitu informacyjnego użytkownika.

## Bibliografia

- Anon 2020. Ad Exchange historical report - Google Ad Manager Help. [online] Available at: <<https://support.google.com/admanager/table/7612037?hl=en>> [Accessed 10 Jun. 2020].
- Anon 2020. Autoregressive integrated moving average. In: Wikipedia. [online] Available at: <[https://en.wikipedia.org/w/index.php?title=Autoregressive\\_integrated\\_moving\\_average&oldid=950625193](https://en.wikipedia.org/w/index.php?title=Autoregressive_integrated_moving_average&oldid=950625193)> [Accessed 1 Jun. 2020].
- Anon 2020. Exponential smoothing. In: Wikipedia. [online] Available at: <[https://en.wikipedia.org/w/index.php?title=Exponential\\_smoothing&oldid=957231778](https://en.wikipedia.org/w/index.php?title=Exponential_smoothing&oldid=957231778)> [Accessed 1 Jun. 2020].
- Anon 2020. Prophet. [online] Prophet. Available at: <<http://facebook.github.io/prophet/>> [Accessed 1 Jun. 2020].
- Anon 2020. Prophet Diagnostics. [online] Prophet. Available at: <<http://facebook.github.io/prophet/docs/diagnostics.html>> [Accessed 1 Jun. 2020].
- Austin, A., Barnard, J. and Hutcheon, N., 2019. Programmatic Marketing Forecasts 2019. [online] Zenith. Available at: <<https://s3.amazonaws.com/media.mediapost.com/uploads/ProgrammaticMarketingForecasts2019.pdf>>.
- Austin, D., Seljan, S., Monello, J. and Tzeng, S., 2016. Reserve Price Optimization at Scale. In: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). pp.528–536.
- Brownlee, J., 2018a. Deep Learning for Time Series Forecasting. [online] Available at: <[www.machinelearningmastery.com](http://www.machinelearningmastery.com)>.
- Brownlee, J., 2018b. Introduction to Time Series Forecasting in Python. [online] Available at: <[www.machinelearningmastery.com](http://www.machinelearningmastery.com)>.
- Cai, H., Ren, K., Zhang, W., Malialis, K., Wang, J., Yu, Y. and Guo, D., 2017. Real-Time Bidding by Reinforcement Learning in Display Advertising. Proceedings of the Tenth ACM International Conference on Web Search and Data Mining - WSDM '17, pp.661–670.
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R., 2004. CRISP-DM 1.0. SPSS.
- Choi, H., Mela, C.F., Balseiro, S. and Leary, A., 2019. Online Display Advertising Markets: A Literature Review and Future Directions. SSRN Electronic Journal. [online] Available at: <<https://www.ssrn.com/abstract=3070706>> [Accessed 4 May 2020].
- Hipel, K.W. and McLeod, A.I. eds., 1994. Chapter 12 Seasonal Autoregressive Integrated Moving Average Models. In: Developments in Water Science, Time Series Modelling of Water Resources and Environmental Systems. [online] Elsevier.pp.419–462. Available at: <<http://www.sciencedirect.com/science/article/pii/S0167564808706737>> [Accessed 10 Jun. 2020].

- Li, J., Ni, X. and Yuan, Y., 2018. The Reserve Price of Ad Impressions in Multi-Channel Real-Time Bidding Markets. *IEEE Transactions on Computational Social Systems*, 5(2), pp.583–592.
- Li, J., Ni, X., Yuan, Y., Qin, R., Wang, X. and Wang, F.-Y., 2017. The impact of reserve price on publisher revenue in real-time bidding advertising markets. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). [online] 2017 IEEE International Conference on Systems, Man and Cybernetics (SMC). Banff, AB: IEEE.pp.1256–1261. Available at: <<http://ieeexplore.ieee.org/document/8122785/>> [Accessed 7 May 2020].
- Lu, J., Yang, C., Gao, X., Wang, L., Li, C. and Chen, G., 2019. Reinforcement Learning with Sequential Information Clustering in Real-Time Bidding. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. [online] CIKM '19: The 28th ACM International Conference on Information and Knowledge Management. Beijing China: ACM.pp.1633–1641. Available at: <<https://dl.acm.org/doi/10.1145/3357384.3358027>> [Accessed 4 May 2020].
- Makridakis, S., Spiliotis, E. and Assimakopoulos, V., 2018. Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PLOS ONE*, 13(3), p.e0194889.
- Myerson, R.B., 1981. Optimal Auction Design. *Mathematics of Operations Research*, 6(1), pp.58–73.
- Radovanovic, A. and Heavlin, W.D., 2012. Risk-aware revenue maximization in display advertising. In: Proceedings of the 21st international conference on World Wide Web - WWW '12. [online] the 21st international conference. Lyon, France: ACM Press.pp.91–100. Available at: <<http://dl.acm.org/citation.cfm?doid=2187836.2187850>> [Accessed 5 Jun. 2020].
- Ren, K., Qin, J., Zheng, L., Yang, Z., Zhang, W. and Yu, Y., 2019. Deep Landscape Forecasting for Real-time Bidding Advertising. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '19. [online] the 25th ACM SIGKDD International Conference. Anchorage, AK, USA: ACM Press.pp.363–372. Available at: <<http://dl.acm.org/citation.cfm?doid=3292500.3330870>> [Accessed 4 May 2020].
- Riley, J.G. and Samuelson, W.F., 1981. Optimal Auctions. *The American Economic Review*, 71(3), pp.381–392.
- Wang, J., Zhang, W. and Yuan, S., 2017. Display Advertising with Real-Time Bidding (RTB) and Behavioural Targeting. arXiv:1610.03013 [cs]. [online] Available at: <<http://arxiv.org/abs/1610.03013>> [Accessed 11 May 2020].
- Wang, Y., Ren, K., Zhang, W., Wang, J. and Yu, Y., 2016. Functional Bid Landscape Forecasting for Display Advertising. In: European Conference on Machine Learning and Knowledge Discovery in Databases - Volume 9851, ECML PKDD 2016. [online] Riva del Garda, Italy: Springer-Verlag.pp.115–131. Available at: <[https://doi.org/10.1007/978-3-319-46128-1\\_8](https://doi.org/10.1007/978-3-319-46128-1_8)> [Accessed 3 Jun. 2020].
- Wu, W., Yeh, M.-Y. and Chen, M.-S., 2018. Deep Censored Learning of the Winning Price in the Real Time Bidding. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '18. [online] London, United Kingdom: Association for Computing Machinery.pp.2526–2535. Available at: <<https://doi.org/10.1145/3219819.3220066>> [Accessed 3 Jun. 2020].

Wu, W.C.-H., Yeh, M.-Y. and Chen, M.-S., 2015. Predicting Winning Price in Real Time Bidding with Censored Data. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15. [online] Sydney, NSW, Australia: Association for Computing Machinery.pp.1305–1314. Available at: <<https://doi.org/10.1145/2783258.2783276>> [Accessed 3 Jun. 2020].

Yang, C., Lu, J., Gao, X., Liu, H., Chen, Q., Liu, G. and Chen, G., 2020. MoTiAC: Multi-Objective Actor-Critics for Real-Time Bidding. arXiv:2002.07408 [cs]. [online] Available at: <<http://arxiv.org/abs/2002.07408>> [Accessed 5 May 2020].

S. Makridakis, E. Spiliotis, V. Assimakopoulos, *Statistical and Machine Learning forecasting methods: Concerns and ways forward*, <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0194889>, 2018

J. Brownlee, *Introduction to Time Series Forecasting in Python*, [www.machinelearningmastery.com](http://www.machinelearningmastery.com), 2018

J. Brownlee, *Deep Learning for Time Series Forecasting*, [www.machinelearningmastery.com](http://www.machinelearningmastery.com), 2018

*Augmented Dickey-Fuller test*, Wikipedia, [https://en.wikipedia.org/wiki/Augmented\\_Dickey%E2%80%93Fuller\\_test](https://en.wikipedia.org/wiki/Augmented_Dickey%E2%80%93Fuller_test), accessed: 20.11.2018.

*Autoregressive Integrated Moving Average*, [https://en.wikipedia.org/wiki/Autoregressive\\_integrated\\_moving\\_average](https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average), accessed: 20.11.2018.

*Exponential Smoothing*, [https://en.wikipedia.org/wiki/Exponential\\_smoothing](https://en.wikipedia.org/wiki/Exponential_smoothing), accessed: 20.11.2018.

*MEA and RMSE - Which Metric is Better?* <https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d>, accessed: 22.11.2018

Maravall, Banco de Espania, [https://www.bde.es/bde/en/secciones/servicios/Profesionales/Programas\\_estadi/Notas\\_introduccion\\_3638497004e2e21.html](https://www.bde.es/bde/en/secciones/servicios/Profesionales/Programas_estadi/Notas_introduccion_3638497004e2e21.html), accessed: 22.11.2018

A Kaggle's Guide to Model Stacking in Practice, <http://blog.kaggle.com/2016/12/27/a-kagglers-guide-to-model-stacking-in-practice/>,

Introduction to Ensembling/Stacking in Python, <https://www.kaggle.com/arthurthok/introduction-to-ensembling-stacking-in-python>

Prophet – Diagnostics. <https://facebook.github.io/prophet/docs/diagnostics.html>

Python, <https://www.python.org/>

PHP: Hypertext Preprocessor, <https://www.php.net/>

Angular, <https://angular.io/>

Symfony, <https://symfony.com/>

SQLite, <https://www.sqlite.org/index.html>

PostgreSQL, <https://www.postgresql.org/>

Conda, <https://docs.conda.io/en/latest/>

Prophet, [https://facebook.github.io/prophet/docs/quick\\_start.html](https://facebook.github.io/prophet/docs/quick_start.html)

Docker, <https://www.docker.com/>

Apache, <https://httpd.apache.org/>

GitLab, <https://about.gitlab.com/>

Rest API, [https://en.wikipedia.org/wiki/Application\\_programming\\_interface](https://en.wikipedia.org/wiki/Application_programming_interface),  
[https://en.wikipedia.org/wiki/Representational\\_state\\_transfer](https://en.wikipedia.org/wiki/Representational_state_transfer)

Google AdManager, <https://admanager.google.com/>

Google AdManager API, <https://developers.google.com/ad-manager/api/start>

SciKit – Learn, <https://scikit-learn.org/stable/>

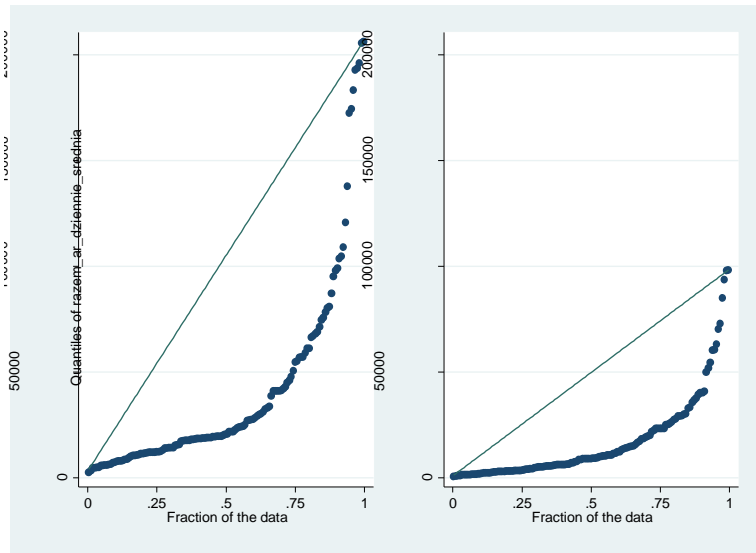
SciPy, <https://www.scipy.org/>

## Załącznik 1. Raport. Eksploracyjna analiza danych 04.2019

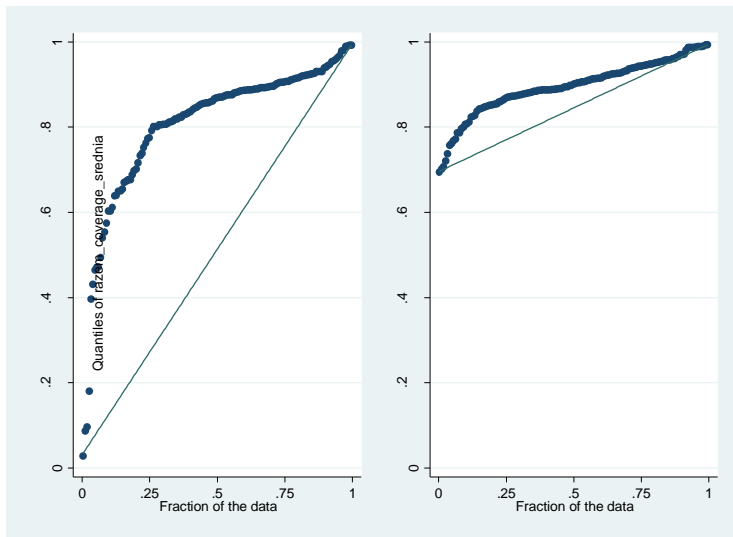
Klasyfikacja – wstępne wnioski

1. Przeanalizowano 1297 tagów z okresu 10.04.2018-10.04.2019 pod względem
  - danych dziennych o przychodach, AREcpm, Coverage, liczby requestów dla danego tagu
  - ich miar zmienności (odchylenie standardowe i współczynnik zmienności)
  - danych zagregowanych per tag o poziomie przychodów i AREcpm dla Branded i Anonymous, relacji ich AREcpm, udziału Branded do Anonymous (według AR i przychodów)
  - danych zagregowanych per tag o przychodach, AREcpm, AR z zapytań z krajów UE, USA, Nowej Zelandii, Kanady i Australii („kraje premium”)
  - danych zagregowanych o koncentracji przychodów per tag generowanych przez 10 największych kupujących (Advertisers)
  - danych zagregowanych o bidach, win i close CPM per tag (dla okresu 180 dni)
2. Z przeprowadzonej analizy wynika, że istotność biznesowa tagów charakteryzuje się dużą koncentracją:
  - tylko 850 wykazało jakiegokolwiek przychody
  - 209 wykazało przychody powyżej 1000 zł
  - 110 tagów wygenerowało 80% przychodu
  - Pierwszych 10 tagów: 29% przychodów
3. Do dalszej analizy wzięto 274 tagi o przychodach
  - powyżej 1000 zł w całym okresie
  - powyżej 80 zł w marcu 2019
4. Analiza objęła relacje między 34 miarami opisującymi każdy z tagów. Pierwsza klasyfikacja dotyczyła podziału tagów na dwie grupy według poziomu AREcpm - powyżej i poniżej średniej (analiza z podziałem na kwartyle rozkładu dała podobne wyniki)
  - Dla tagów o AREcpm wyższym od średniej (0,40/1,24) istotnie niższa jest przeciętna dzienna liczba AR (41000/17100), ale jednocześnie wyższy coverage (0,80/0,90)

**DZIENNE AR (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**

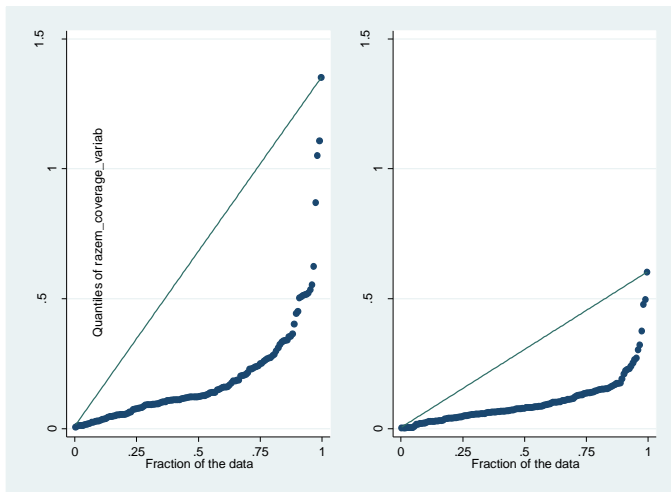


**COVERAGE (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



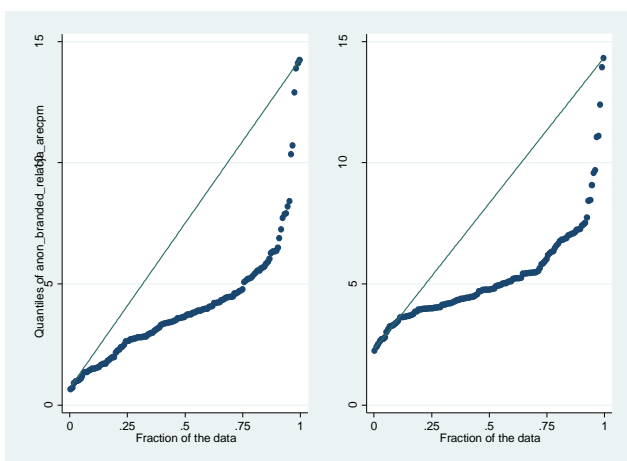
- Droższe tagi mają dużo stabilniejszy coverage (istotnie niższy współczynnik zmienności dziennej per tag)

**ZMIENNOŚĆ COVERAGE (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



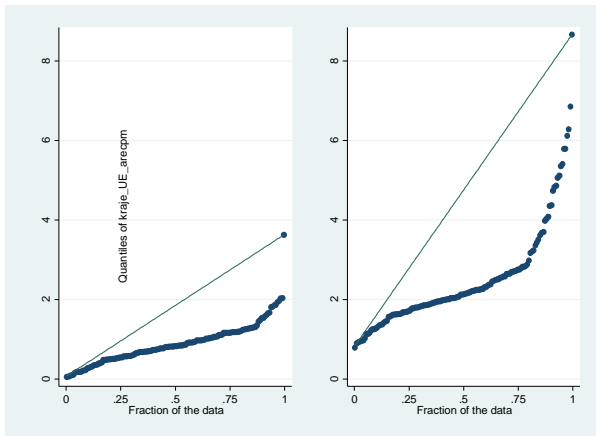
- Brak istotnych różnic w udziale Branded/Anonymous między tagami drogimi i tanimi
- W tagach drogich relacja Branded/Anonymous AREcpm dużo wyższa niż w tanich (4,07/5,54)

**RELACJA AREcpm Branded/Anonymous (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**

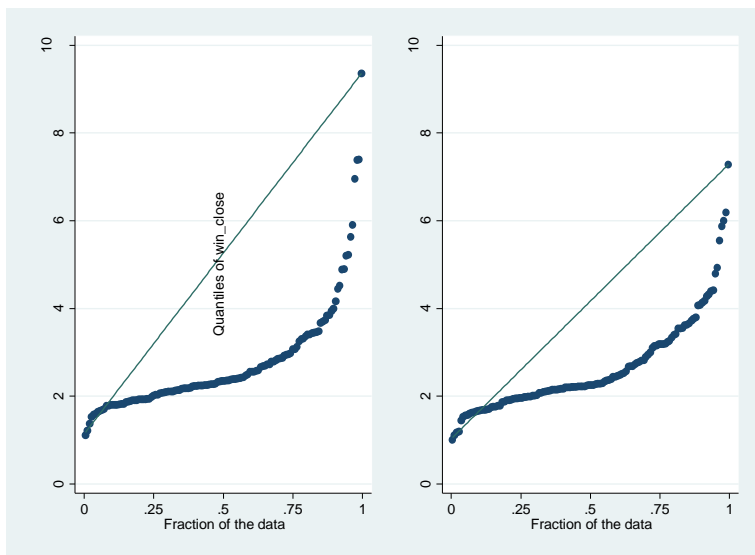


- Dla odsetków kupionych z krajów premium poziom relacji AREcpm dla tagów drogich i tanich jest zachowany (0,89/2,51 wobec 0,40/1,24 na całej próbie)

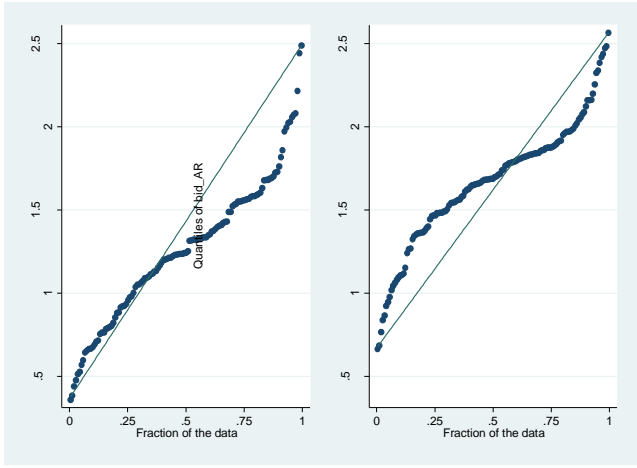




- Dla tagów droższych relacja winCPM do closeCPM jest niższa (3,18/2,25)  
**RELACJA winCPM do closeCPM (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



- Relacja BIDów do AdRequestów jest natomiast znacznie wyższa dla tagów drogich (1,27/1,67)  
**RELACJA BID/AR (lewy panel: poniżej średniego AREcpm; prawy: powyżej)**



# Załącznik 2 - Eksperymenty 'zero'

---

1	Eksperyment gd_test_336x280_1_T_ (od 2018-11-13 do 2018-12-10) .....	2
2	Eksperyment gd_test_336x280_1_T_ (od 2018-12-12 do 2019-1-1) .....	8
3	Eksperyment cg_test_300x250_mobile_2_T_ (od 2018-12-20 do 2018-12-27) .....	15
4	Eksperyment cg_test_300x250_mobile_2_T_ (od 2018-12-28 do 2018-12-30) .....	21
5	Eksperyment cg_test_300x250_mobile_2_T_ (od 2019-1-1 do 2019-1-7) .....	27
6	Eksperyment di_test_300x250_desktop_T_ (od 2018-12-20 do 2018-12-30) .....	33
7	Eksperyment cg_test_300x250_mobile_2_T_ (od 2019-1-8 do 2019-1-16) .....	40
8	Eksperyment cg_test_300x250_mobile_2_T_ (od 2019-1-20 do 2019-1-31) .....	46
9	Eksperyment di_test_300x250_desktop_T_ (od 2019-1-1 do 2019-1-16) .....	52
10	Eksperyment cg_test_336x280_1_T_ (od 2019-1-12 do 2019-1-16) .....	62
11	Eksperyment cg_test_336x280_1_T_ (od 2019-1-20 do 2019-1-31) .....	69
12	Eksperyment 93_test_300x600_T_ (od 2019-1-12 do 2019-1-16) .....	76
13	Eksperyment 93_test_300x600_T_ (od 2019-1-17 do 2019-1-31) .....	83
14	Eksperyment ch_test_160x600_left_T_ (od 2019-1-12 do 2019-1-16) .....	90
15	Eksperyment ch_test_160x600_left_T_ (od 2019-1-20 do 2019-1-31) .....	98
16	Eksperyment j5_test_970x250_T_ (od 2019-1-12 do 2019-1-15) .....	105
17	Eksperyment ku_test_300x250_mobile_1_T_ (od 2019-1-20 do 2019-1-31) .....	112
18	Eksperyment di_test_300x250_desktop_T_ (od 2019-1-22 do 2019-1-31) .....	119
19	Eksperyment ip_test_750x200_LIST_D_E_F_T_ (od 2019-1-23 do 2019-1-31) .....	129

20	Eksperyment i0_test_970x250_atf_T_ (od 2019-1-23 do 2019-1-31).....	136
----	---	-----

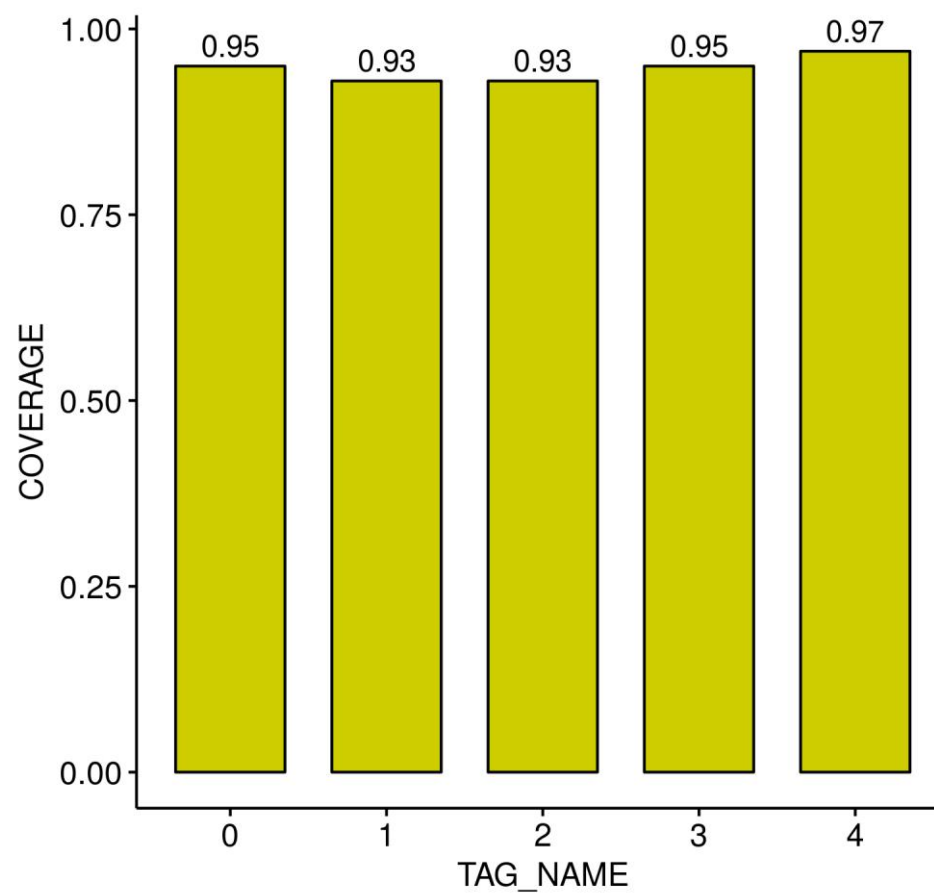
## 1 Eksperyment gd\_test\_336x280\_1\_T\_ (od 2018-11-13 do 2018-12-10)

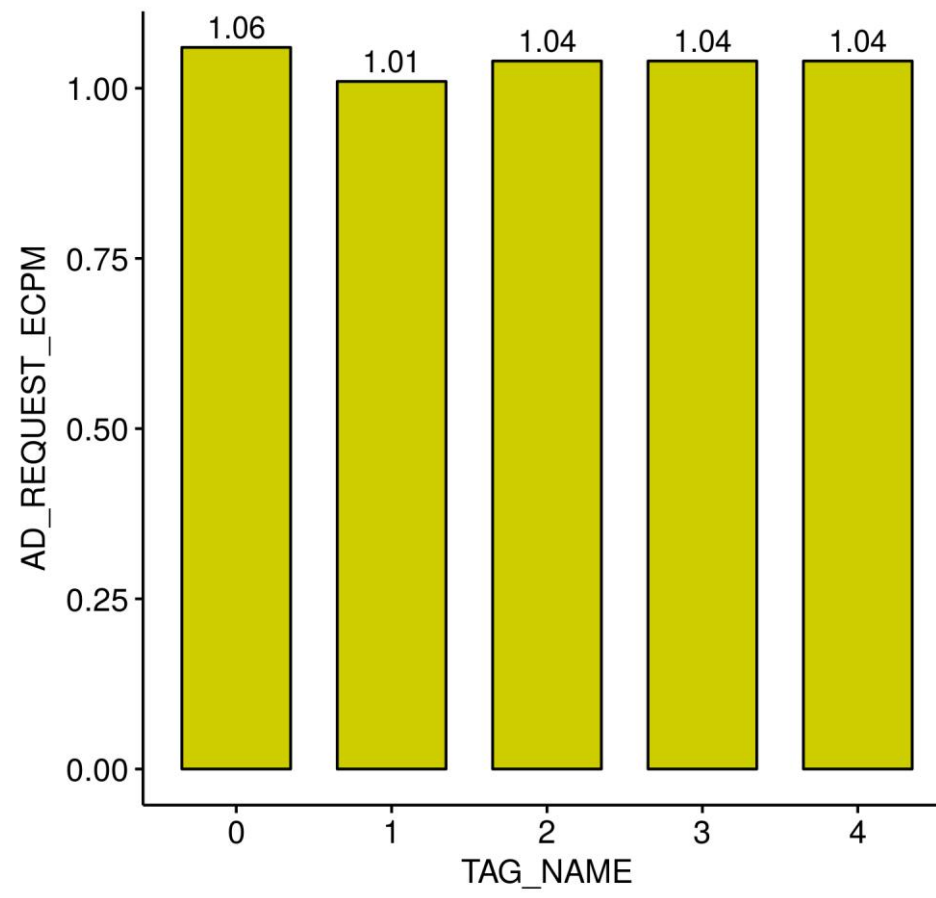
Data wygenerowania raportu: 2019-01-02

**Cel eksperymentu:** Badanie podziału rotacyjnego tagu gd\_test\_336x280\_1\_T\_ Na tagu \_0 kierowane 20% ruchu, na \_1 50%, na \_3, \_4 oraz \_5 po 10% ruchu.

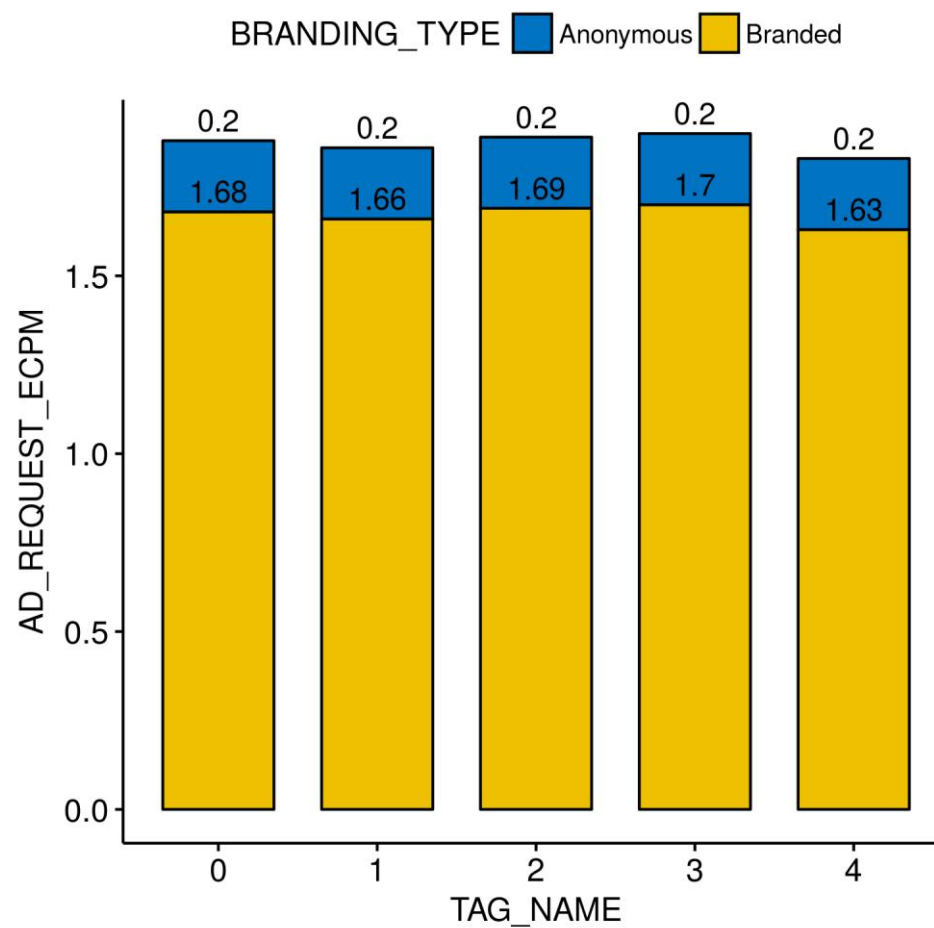
**Wnioski z eksperymentu:** Testy rotacyjne wskazują na wahania dzienne wskaźnika AdRequesteCPM 0,130 (13,3%). Maksymalne wahanie wystąpiło w dniu 2018-12-06 (0,224; 34,1%). W całym okresie wahanie wskaźnika to 0,05. Testy wskazują na zrównoważony poziom wskaźnika ARe w całym zakresie czasowym testu i pozwalają przeprowadzać kolejne eksperymenty ze zróżnicowanymi regułami na tagach o różnej rotacji (50%, 20%, 10% x3)

## 1.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

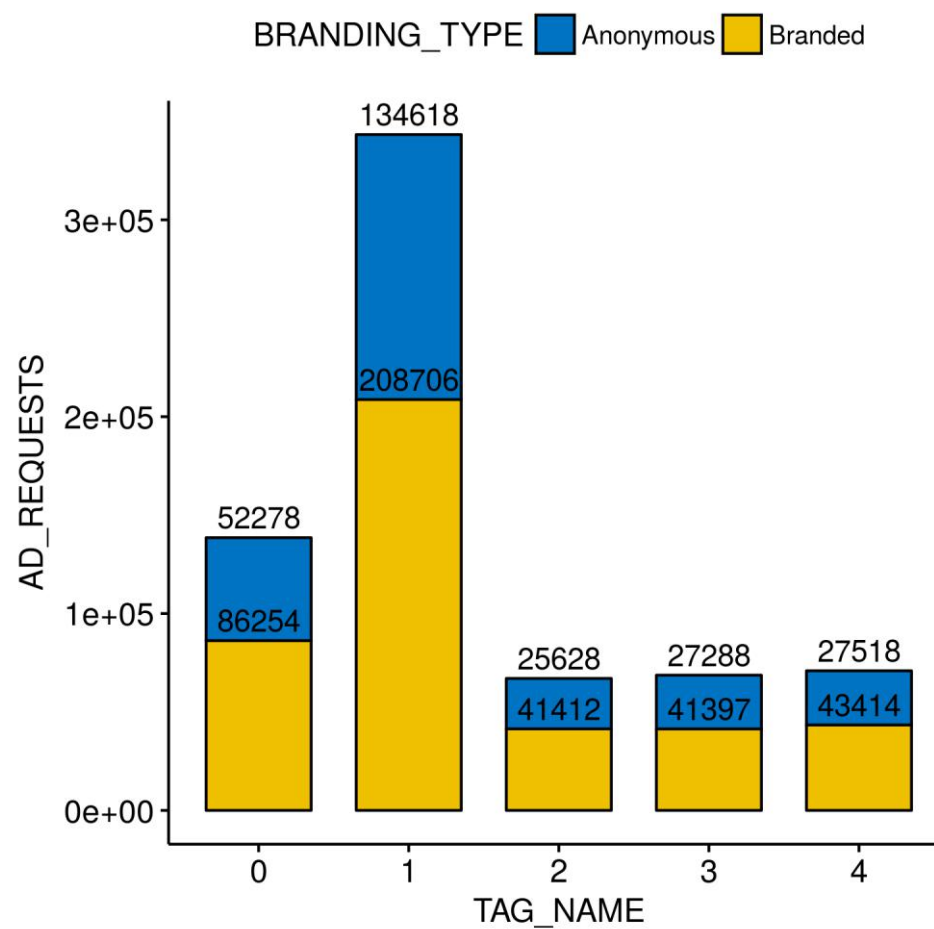




## 1.2 Wykres AdR ECPM z podziałem na Branding Type

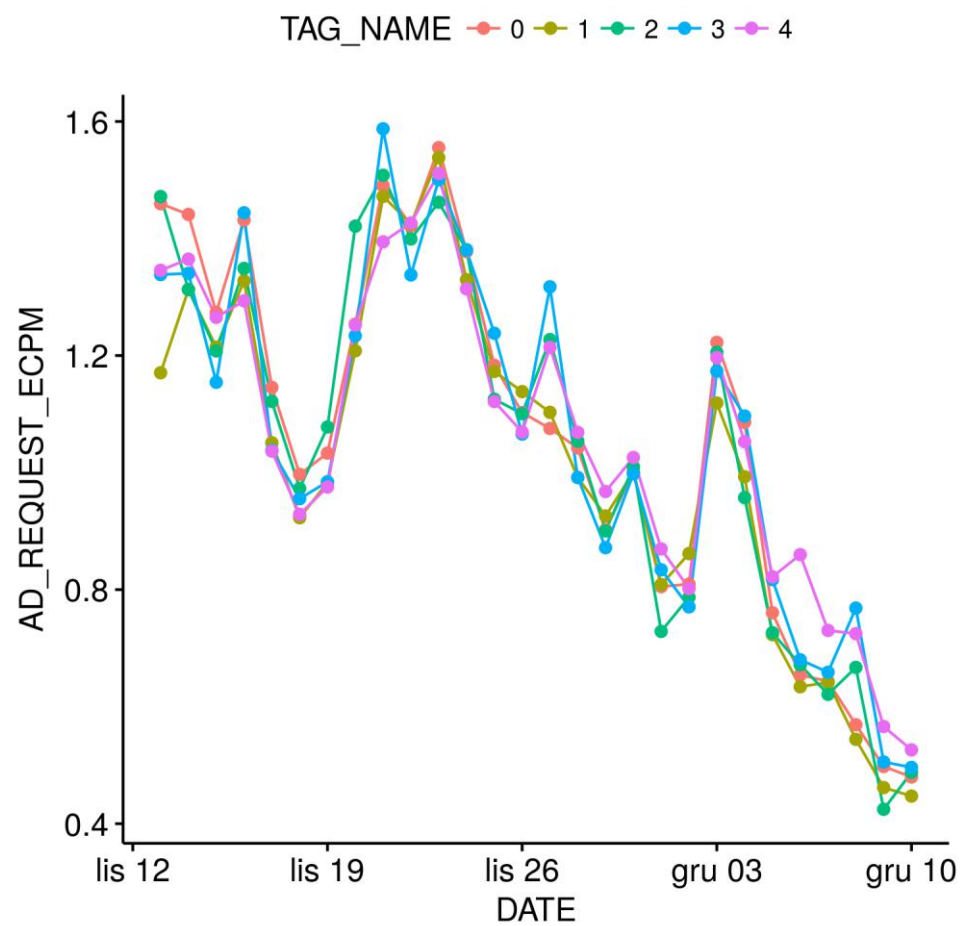


### 1.3 Wykres Ad requests z podziałem na Branding Type





## 1.4 Wykres trendu dziennego AdRequest eCPM



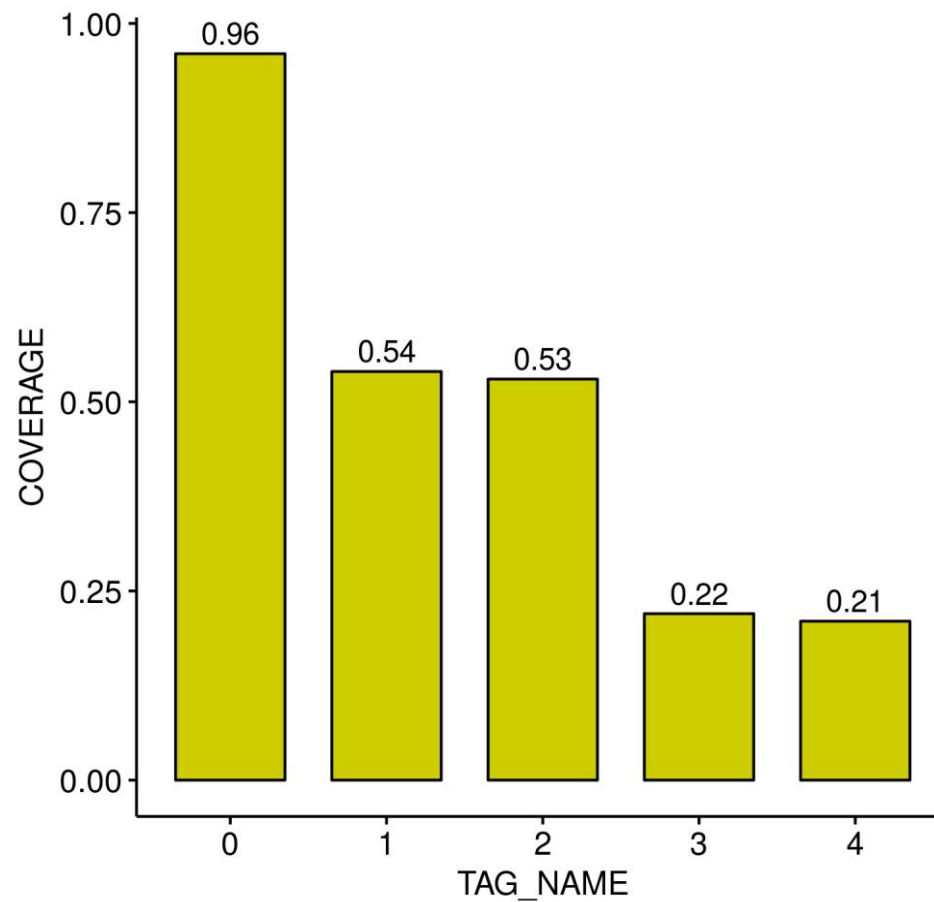
## 2 Eksperyment gd\_test\_336x280\_1\_T\_ (od 2018-12-12 do 2019-1-1)

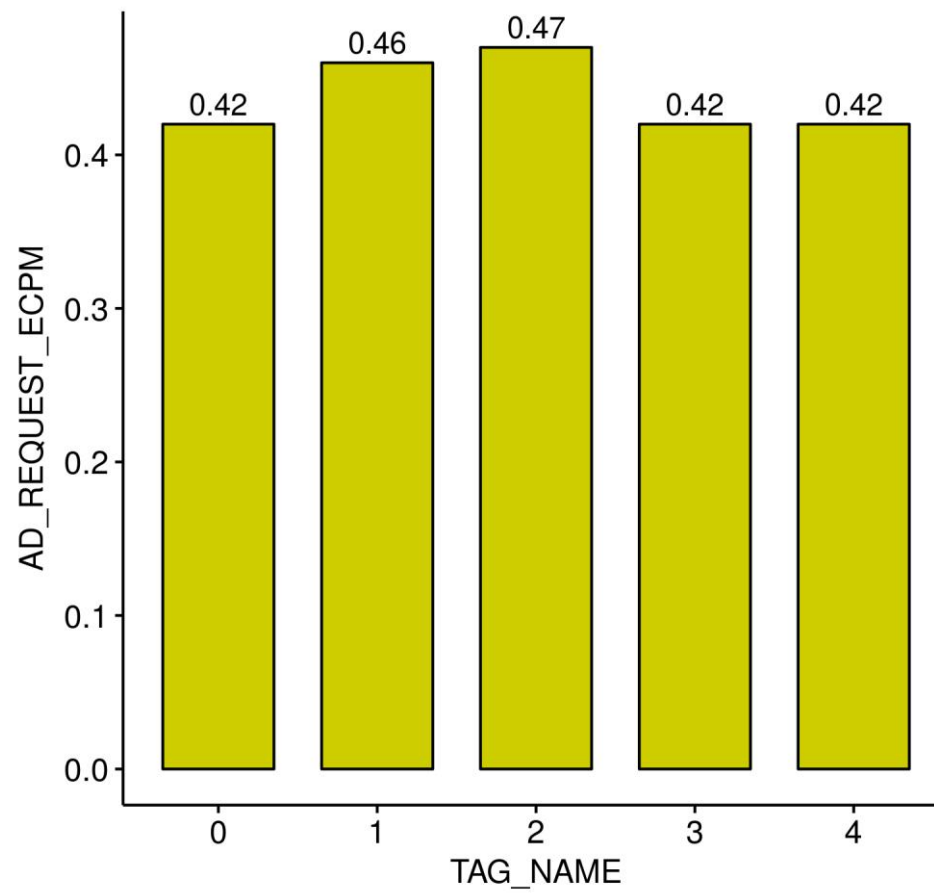
Data wygenerowania raportu: 2019-01-02

Data	tag_ad_unit	rule_name	wejsciove_bran ded	wejsciove_anony mous	wyjsciove_bran ded	wyjsciove_anony mous	Uwagi
2018-12-11	test_336x280_1_T_1	test_336x280_1_T_1(YR)	1.14	0.12	1.14	0.12	
2018-12-11	test_336x280_1_T_2	test_336x280_1_T_2	1.14	0.12	0	0	
2018-12-11	test_336x280_1_T_3	test_336x280_1_T_3(1)	1.14	0.12	2.28	0.12	kraje UE+USA, Australia, Nowa Zelandia
2018-12-11	test_336x280_1_T_3	test_336x280_1_T_3(2)	1.14	0.12	1.14	0.12	
2018-12-11	test_336x280_1_T_4	test_336x280_1_T_4	1.14	0.12	1.14	1.14	
2018-12-11	test_336x280_1_T_5	test_336x280_1_T_5(1)	1.14	0.12	2.24	0	kraje UE+USA, Australia, Nowa Zelandia
2018-12-11	test_336x280_1_T_5	test_336x280_1_T_5(2)	1.14	0.12	0	0	

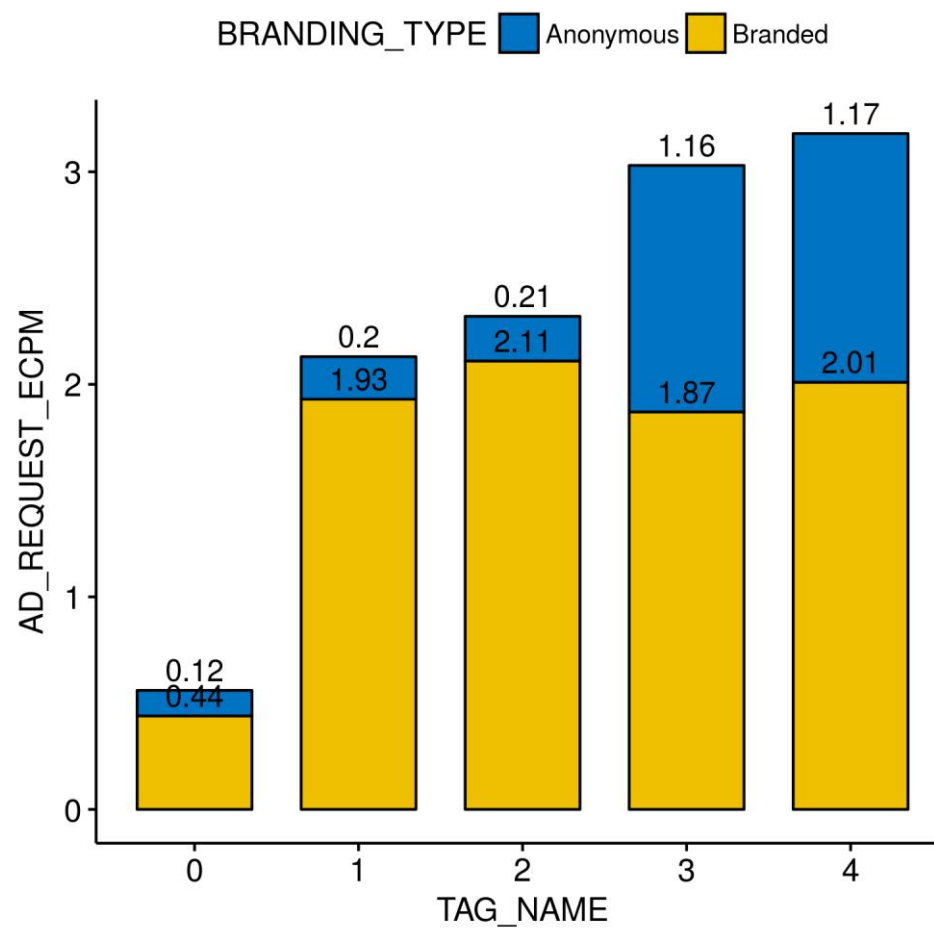
<b>Data</b>	<b>tag_ad_unit</b>	<b>rule_name</b>	<b>wejsciove_bran ded</b>	<b>wejsciove_anony mous</b>	<b>wyjsciove_bran ded</b>	<b>wyjsciove_anony mous</b>	<b>Uwagi</b>
2018-12-12	test_336x280_1_T_5	test_336x280_1_T_5(2)	0	0	1.14	1.14	
2018-12-12	test_336x280_1_T_5	test_336x280_1_T_5(1)	2.28	0	2.8	2.28	kraje UE+USA, Australia, Nowa Zelandia

## 2.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

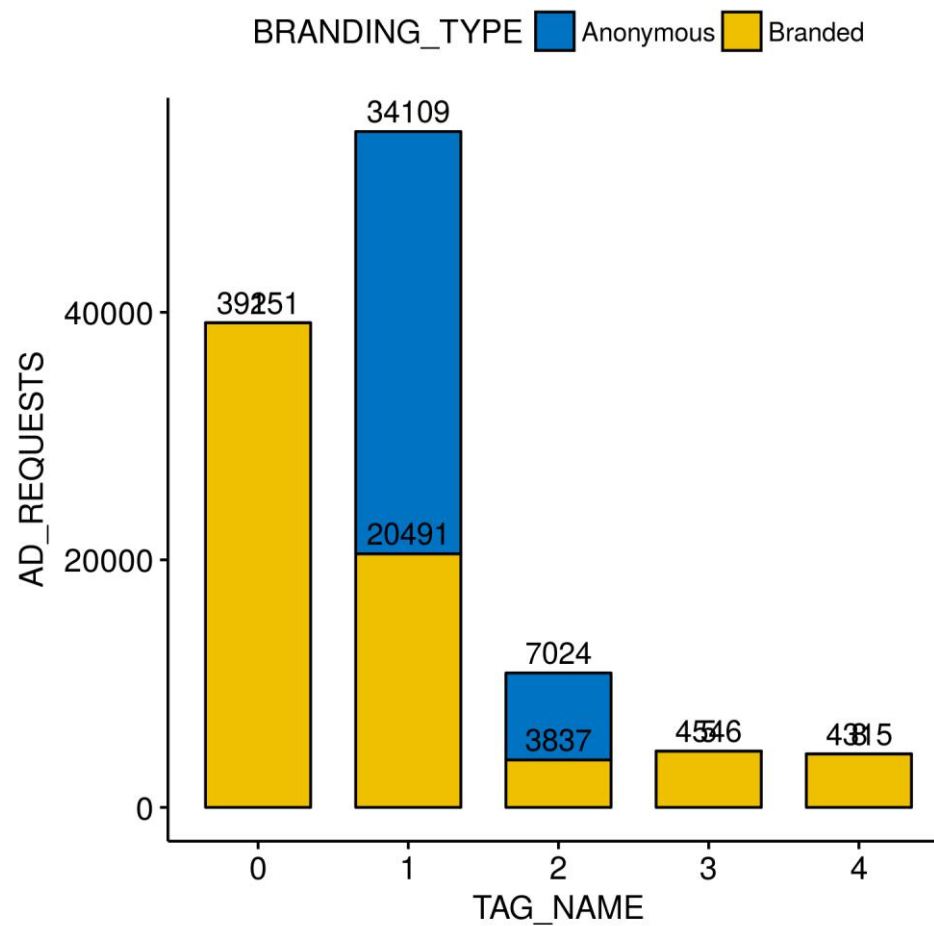




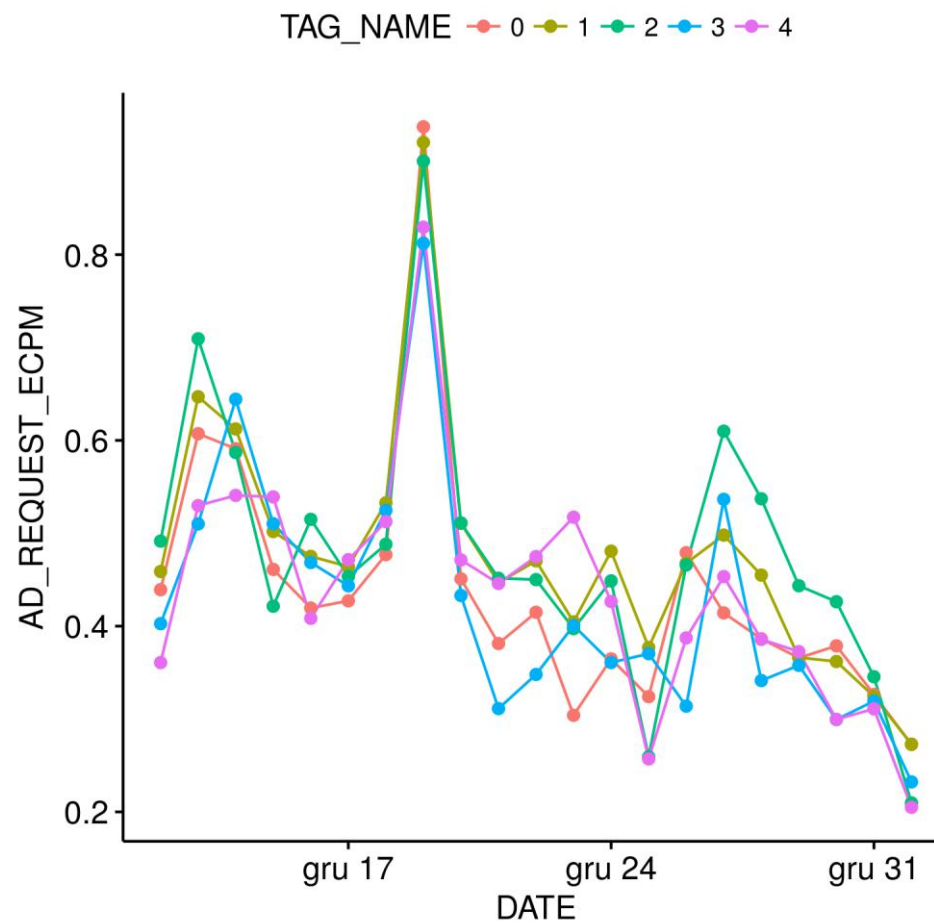
## 2.2 Wykres AdR ECPM z podziałem na Branding Type



### 2.3 Wykres Ad requests z podziałem na Branding Type



## 2.4 Wykres trendu dziennego AdRequest eCPM



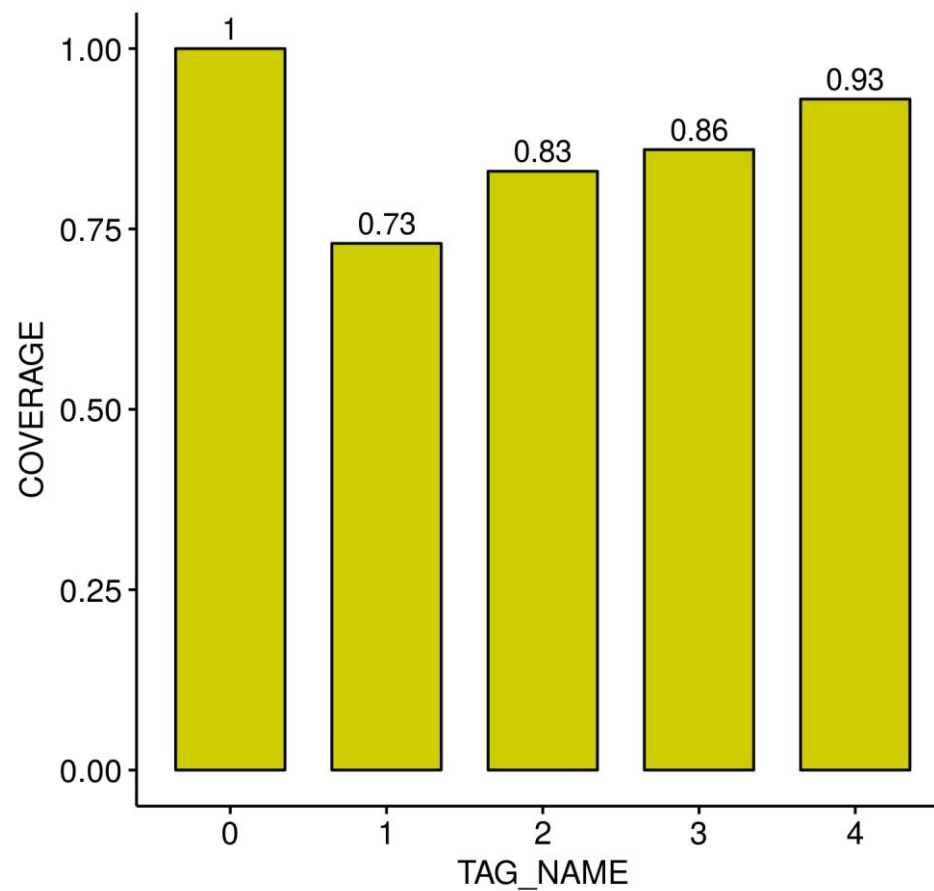


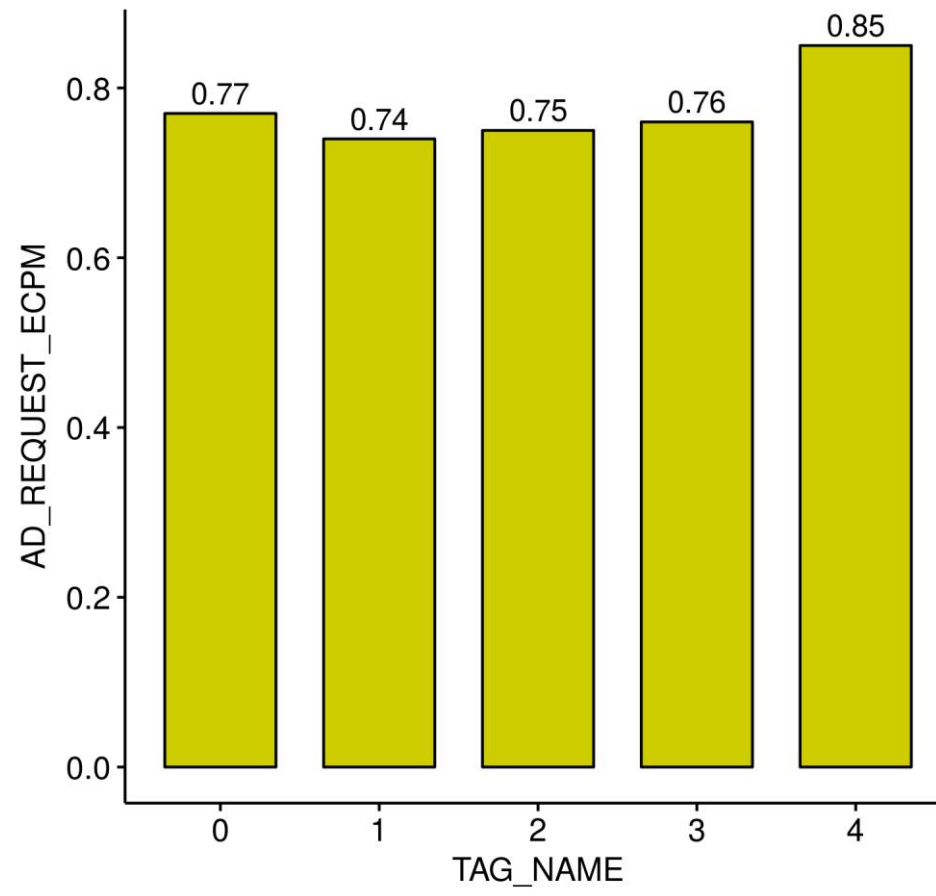
### 3 Eksperyment cg\_test\_300x250\_mobile\_2\_T\_ (od 2018-12-20 do 2018-12-27)

Data wygenerowania raportu: 2018-12-29

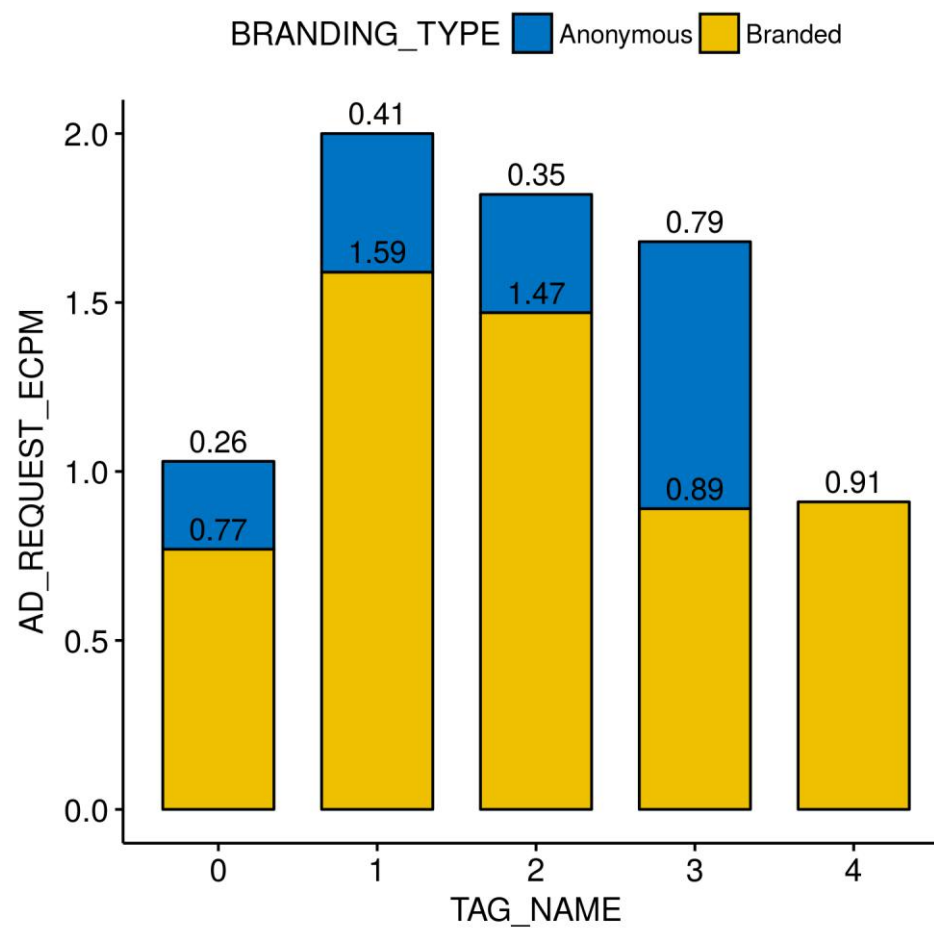
<b>data</b>	<b>tag_ad_unit</b>	<b>rule_name</b>	<b>wejsciove_branded</b>	<b>wejsciove_anonymous</b>	<b>wyjsciove_branded</b>	<b>wyjsciove_anonymous</b>	<b>na</b>
2018-12-18	cg_test_300x250_mobile_2_T_2	cg_test_300x250_mobile_2_T_2	0.9	0.22	0	0	
2018-12-19	cg_test_300x250_mobile_2_T_5	cg_test_300x250_mobile_2_T_5_B	0.9	-	0.22	-	
2018-12-20	cg_test_300x250_mobile_2_T_4	cg_test_300x250_mobile_2_T_4_A	-	0.22	0.3	-	
2018-12-20	cg_test_300x250_mobile_2_T_1	cg_test_300x250_mobile_2_T_1	0.9	0.22	0.9	0.3	
2018-12-20	cg_test_300x250_mobile_2_T_3	cg_test_300x250_mobile_2_T_3	0.9	0.22	0.82	0.22	

### 3.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

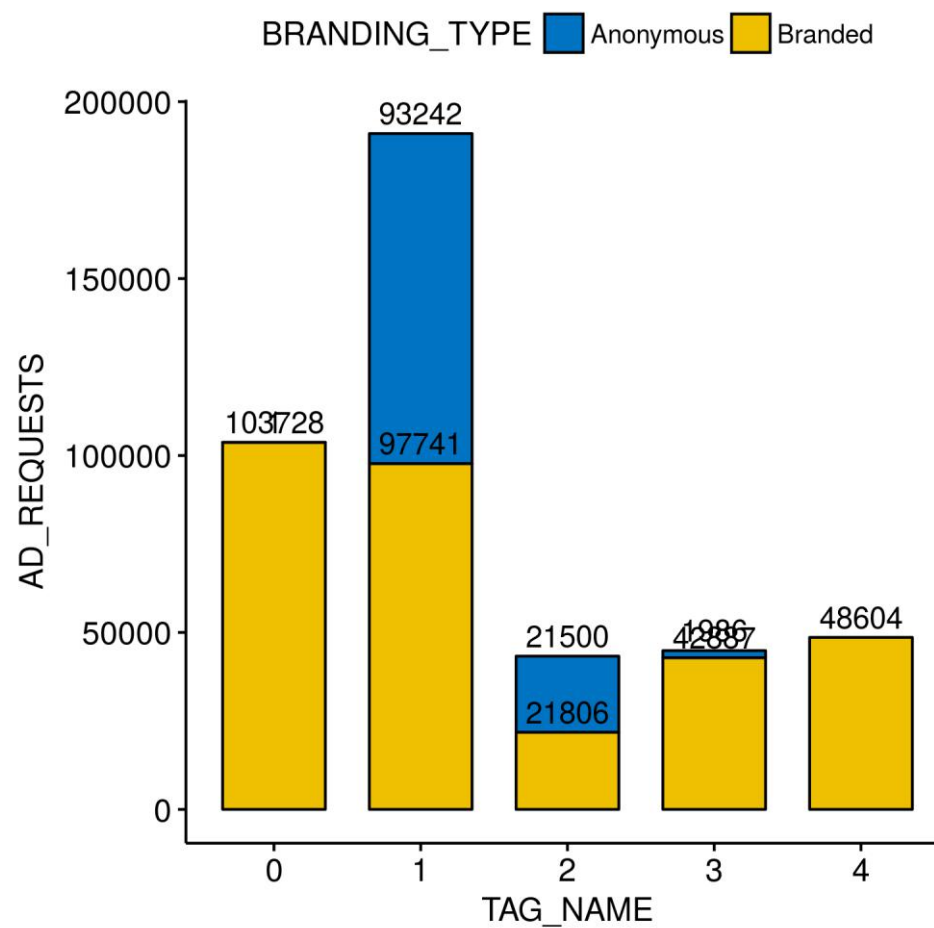




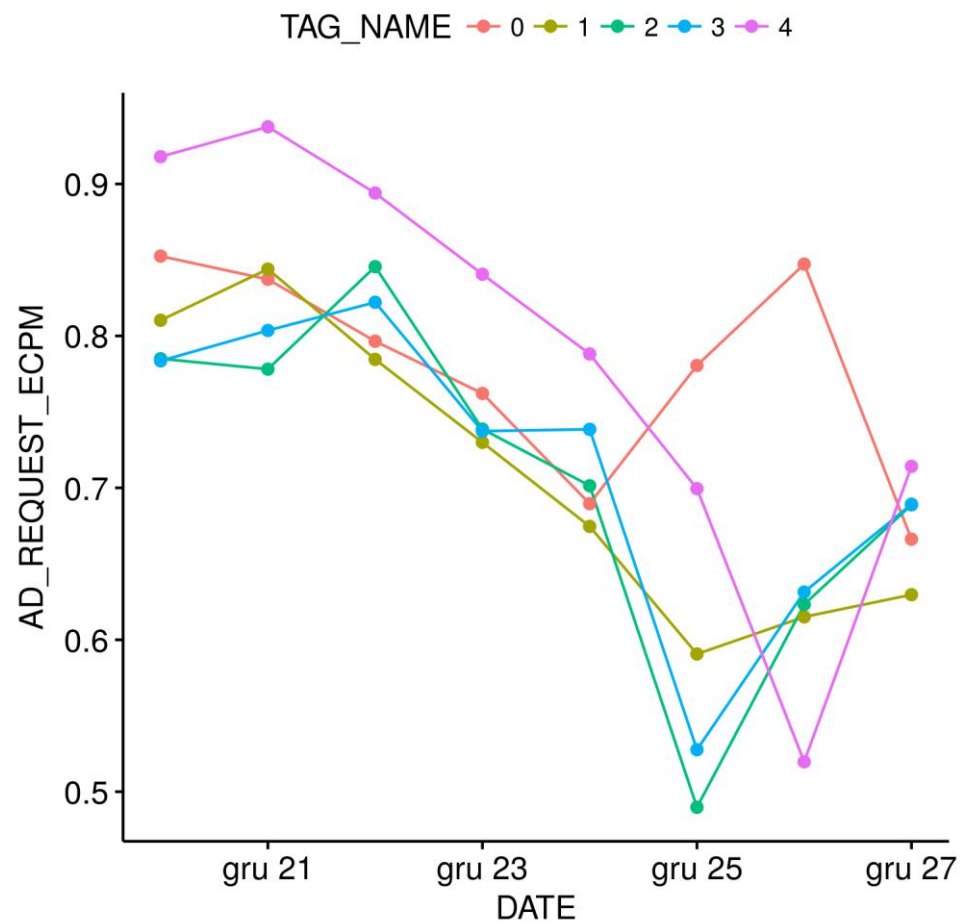
### 3.2 Wykres AdR ECPM z podziałem na Branding Type



### 3.3 Wykres Ad requests z podziałem na Branding Type



### 3.4 Wykres trendu dziennego AdRequest eCPM



#### 4 Eksperyment cg\_test\_300x250\_mobile\_2\_T\_ (od 2018-12-28 do 2018-12-30)

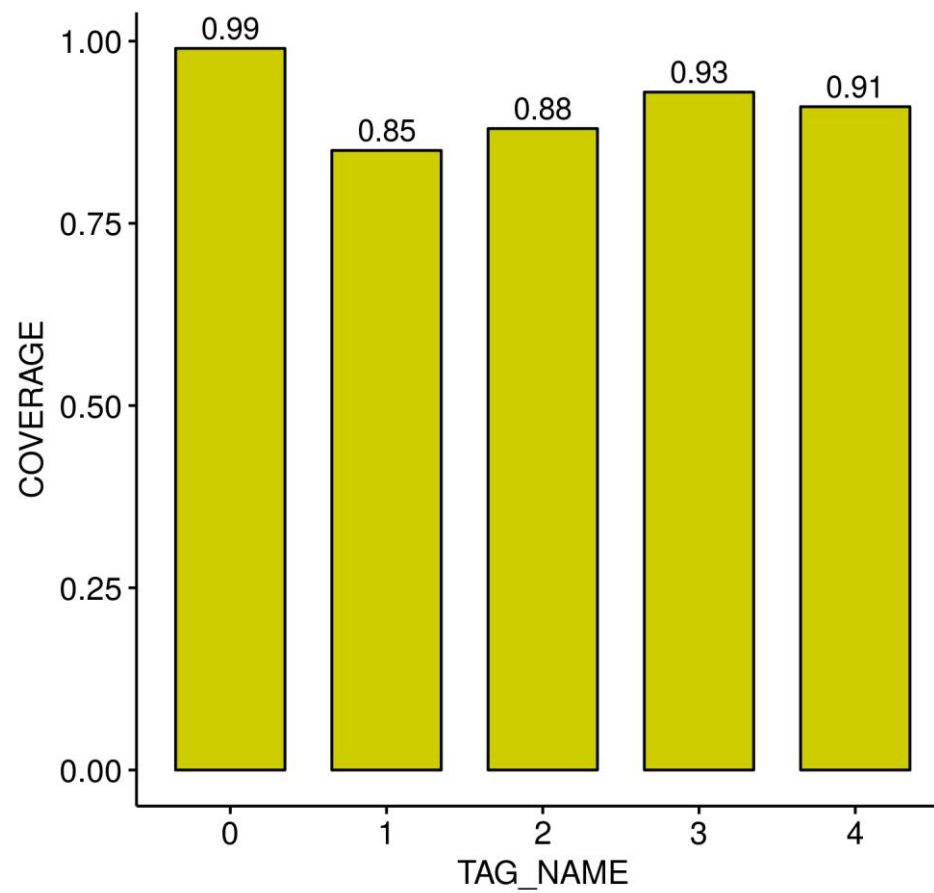
Data wygenerowania raportu: 2018-12-31

Cel: Obniżenie floor (w stosunku do poprzedniej wartości 0.3) na \_3 w celu optymalnego dopasowania flooru na branded. Dodatkowo optymalizacja \_1 i \_2 przez YR.

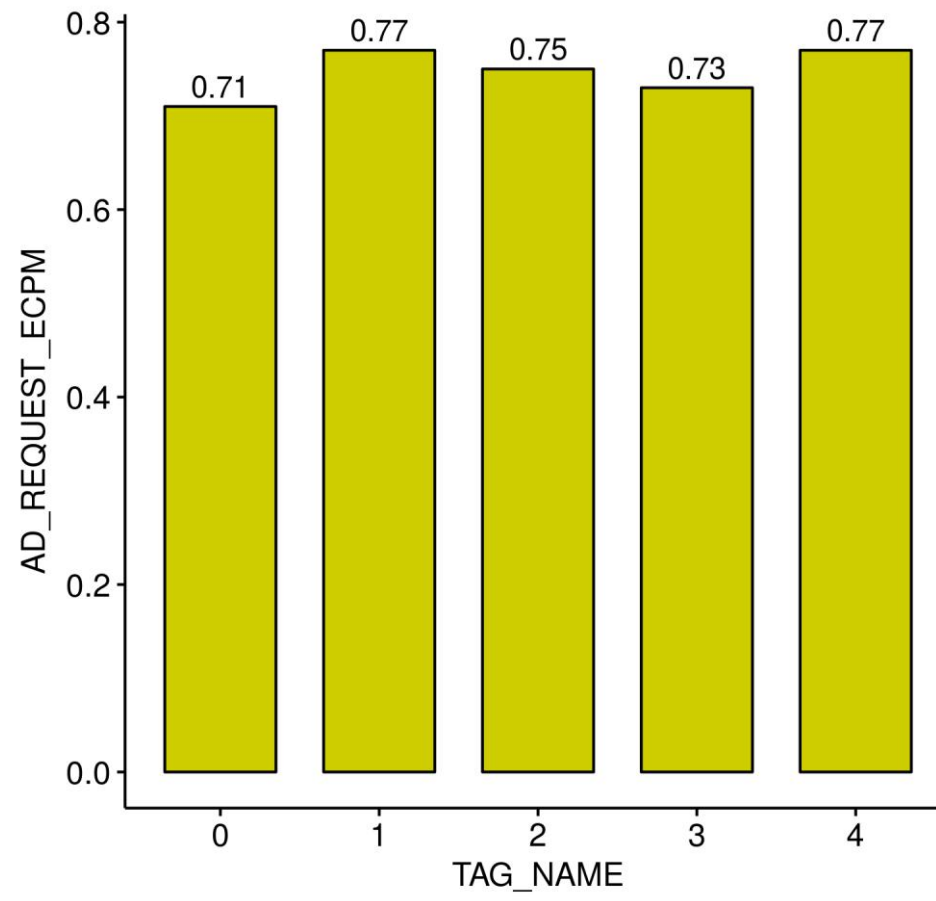
Wniosek: Obniżenie spowodowało spadek ARe (pomimo wzrostu Coverage)

data	tag_ad_unit	rule_name	wejsciove_branded	wejsciove_anonymous	wyjsciove_branded	wyjsciove_anonymous
2018-12-18	cg_test_300x250_mobile_2_T_0	cg_test_300x250_mobile_2_T_0	0.9	0.22	0	0
2018-12-28	cg_test_300x250_mobile_2_T_1	cg_test_300x250_mobile_2_T_1	0.9	0.3	0.7	0.2
2018-12-28	cg_test_300x250_mobile_2_T_2	cg_test_300x250_mobile_2_T_2	0.82	0.22	0.72	0.17
2018-12-28	cg_test_300x250_mobile_2_T_3	cg_test_300x250_mobile_2_T_3_A	0.3	-	0.18	-
2018-12-19	cg_test_300x250_mobile_2_T_4	cg_test_300x250_mobile_2_T_4_B	0.9	-	0.22	-

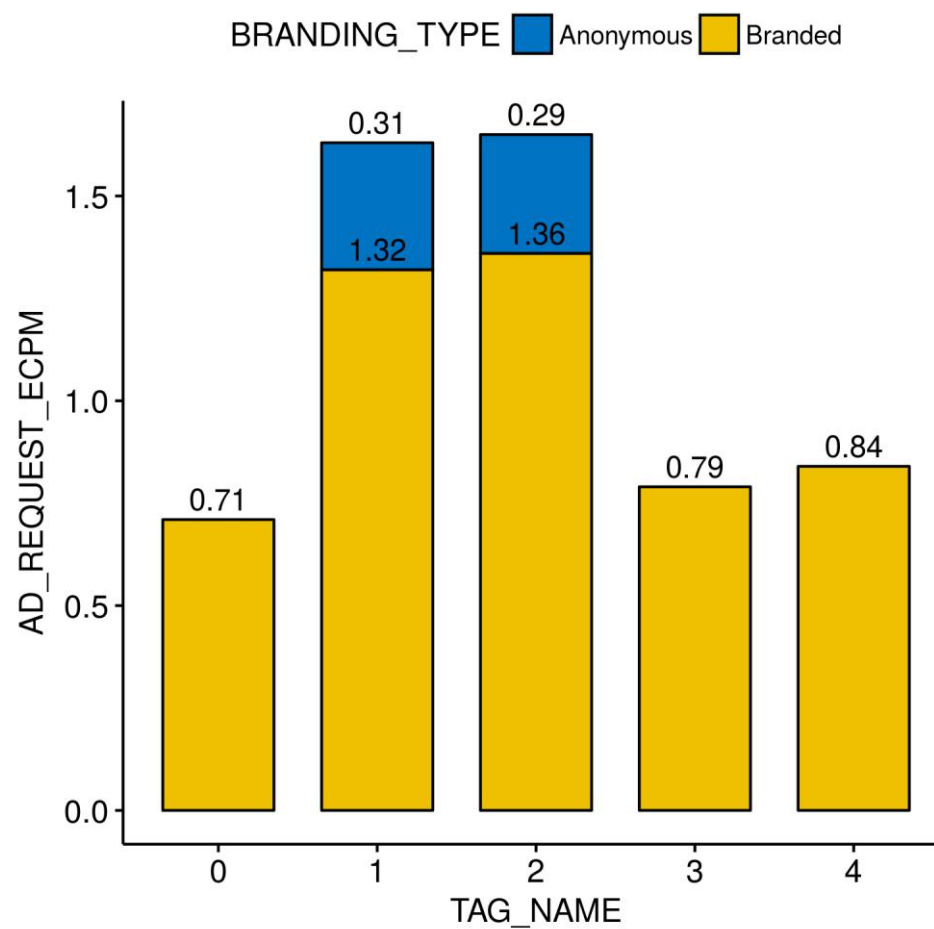
#### 4.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



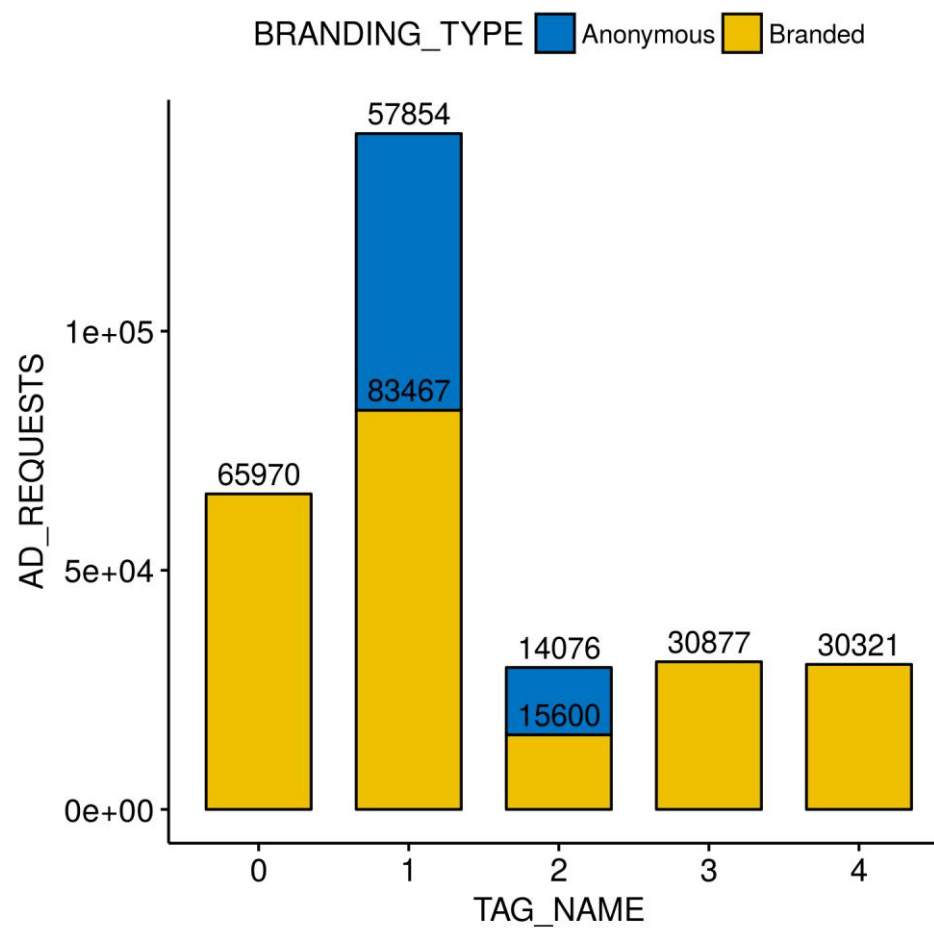




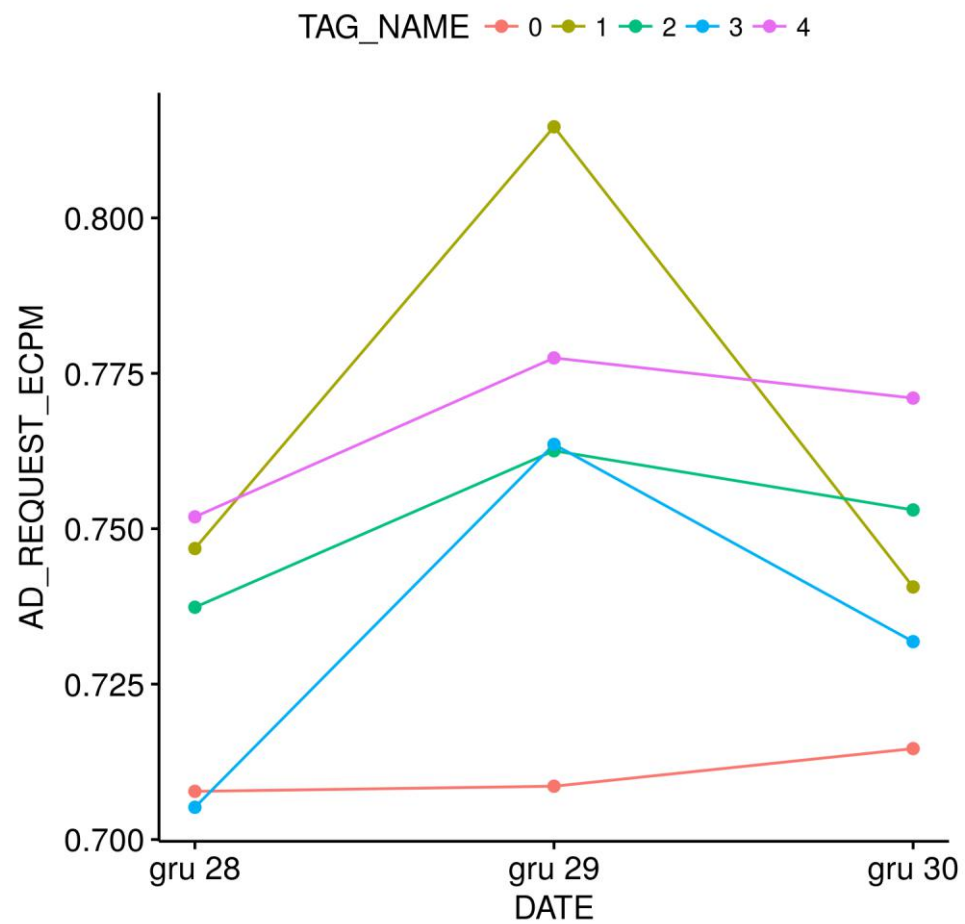
## 4.2 Wykres AdR ECPM z podziałem na Branding Type



### 4.3 Wykres Ad requests z podziałem na Branding Type



#### 4.4 Wykres trendu dziennego AdRequest eCPM



## 5 Eksperyment cg\_test\_300x250\_mobile\_2\_T\_ (od 2019-1-1 do 2019-1-7)

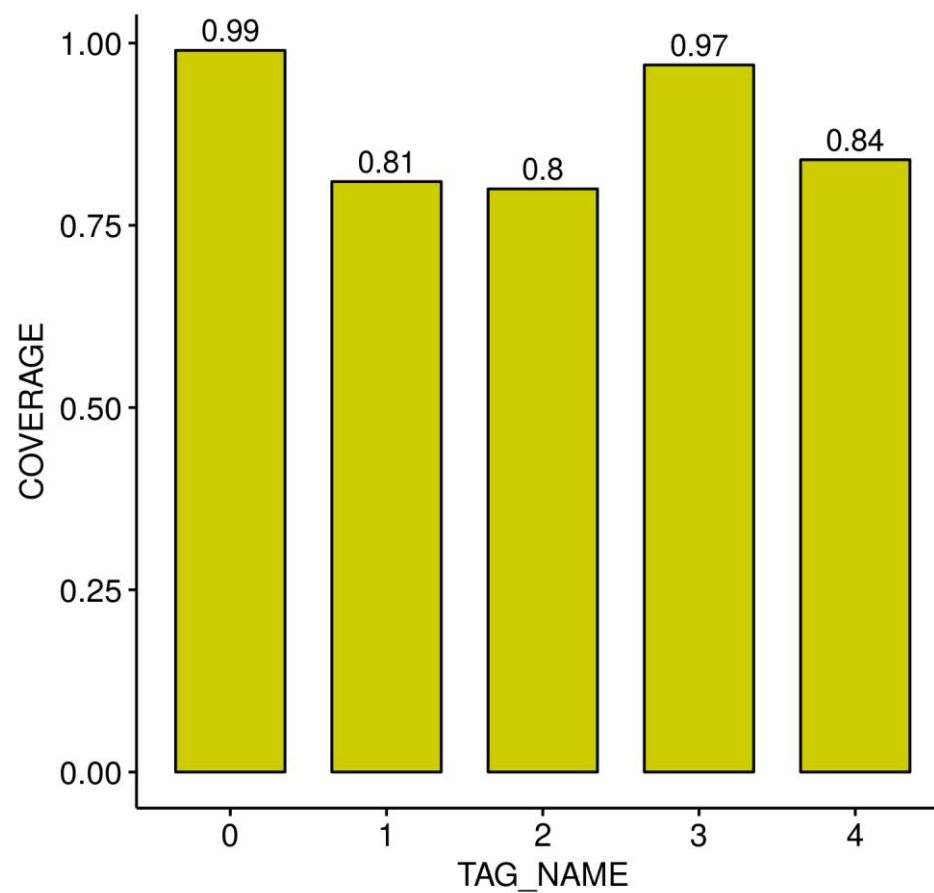
Data wygenerowania raportu: 2019-01-08

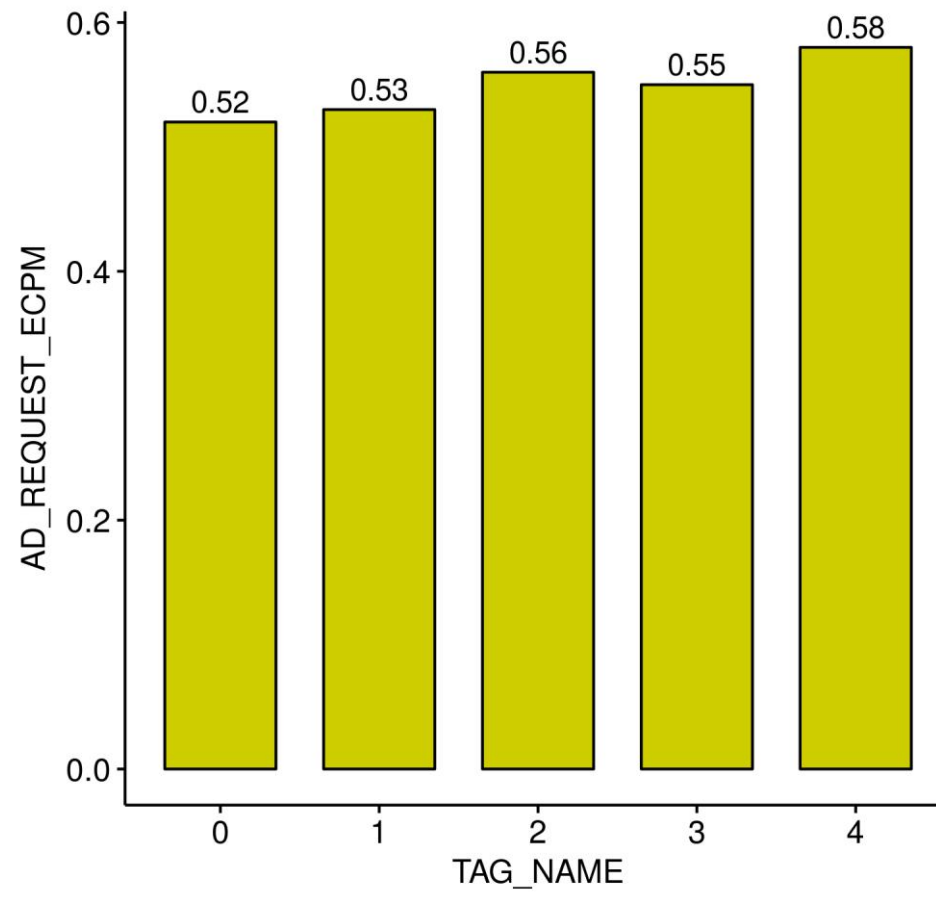
**Cel eksperymentu:** Porównanie na \_3 i \_4 licytacji branded i anonymous. Na \_4 od dłuższego czasu ustawiony optymalny fl B = 0.22 i zablokowane A; na \_3 fl B = 0.22 i fl A = 0. W międzyczasie dostosowanie \_1 i \_2 przez YR.

**Wnioski:** W okresie eksperymentu najwyższe ARe na \_4. Po dopuszczeniu Anonymous na poziomie 0 – obniżenie całkowitego ARe (zgodnie z przewidywaniami, które wynikają z modelu). Prawdopodobnie zablokowanie ruchu Anonymous powoduje takie samo działanie, jak ustawienie tej samej reguły co z Branded (0.22 w tym przypadku) i ostatecznie obniżenie całkowitego ARe. Kolejny eksperyment to ustawienie fl \_B = 0.11 na \_3.

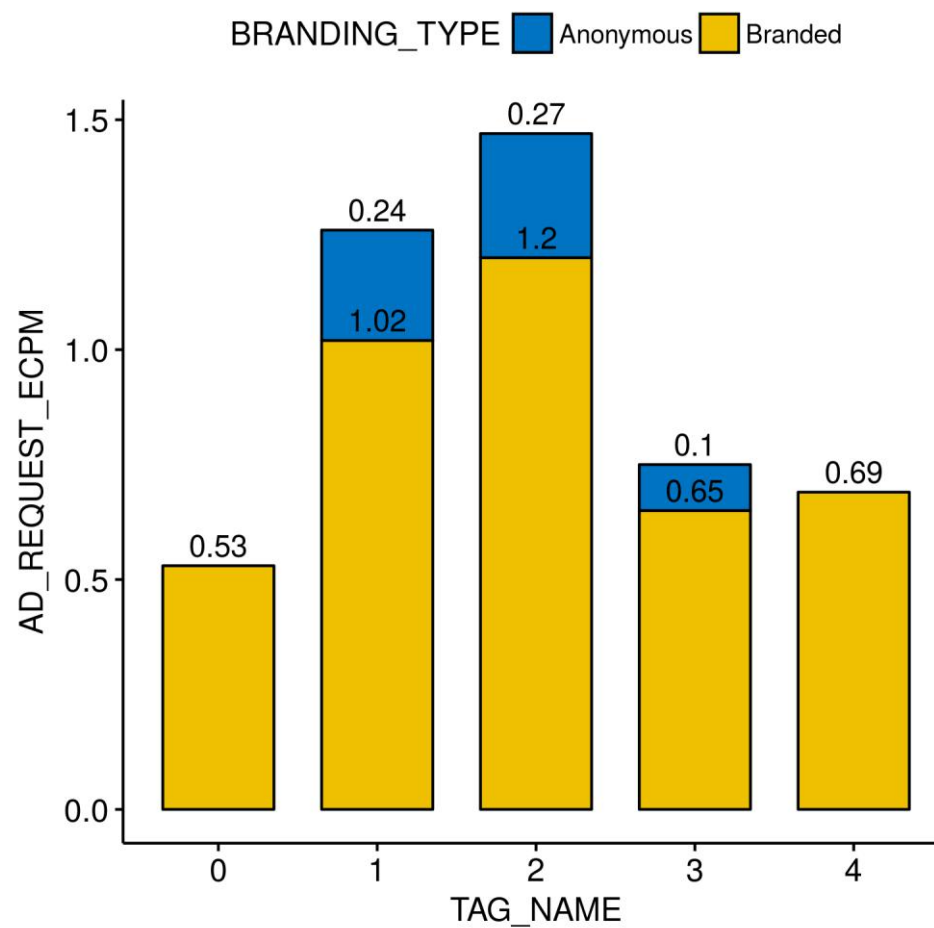
data	tag_ad_unit	rule_name	wejsciowe_ branded	wejsciowe_a nonymous	wyjsciove _branded	wyjsciove_ anonymous
2018-12-18	cg_test_300x250_mobile_2_T_0	cg_test_300x250_mobile_2_T_0	0.9	0.22	<b>0</b>	<b>0</b>
2019-01-02	cg_test_300x250_mobile_2_T_1	cg_test_300x250_mobile_2_T_1	0.7	0.2	<b>0.6</b>	<b>0.15</b>
2018-12-28	cg_test_300x250_mobile_2_T_2	cg_test_300x250_mobile_2_T_2	0.82	0.22	<b>0.72</b>	<b>0.17</b>
2018-12-31	cg_test_300x250_mobile_2_T_3	cg_test_300x250_mobile_2_T_3 _A	0.18	-	<b>0.22</b>	<b>0</b>
2018-12-19	cg_test_300x250_mobile_2_T_4	cg_test_300x250_mobile_2_T_4 _B	0.9	-	<b>0.22</b>	-

## 5.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



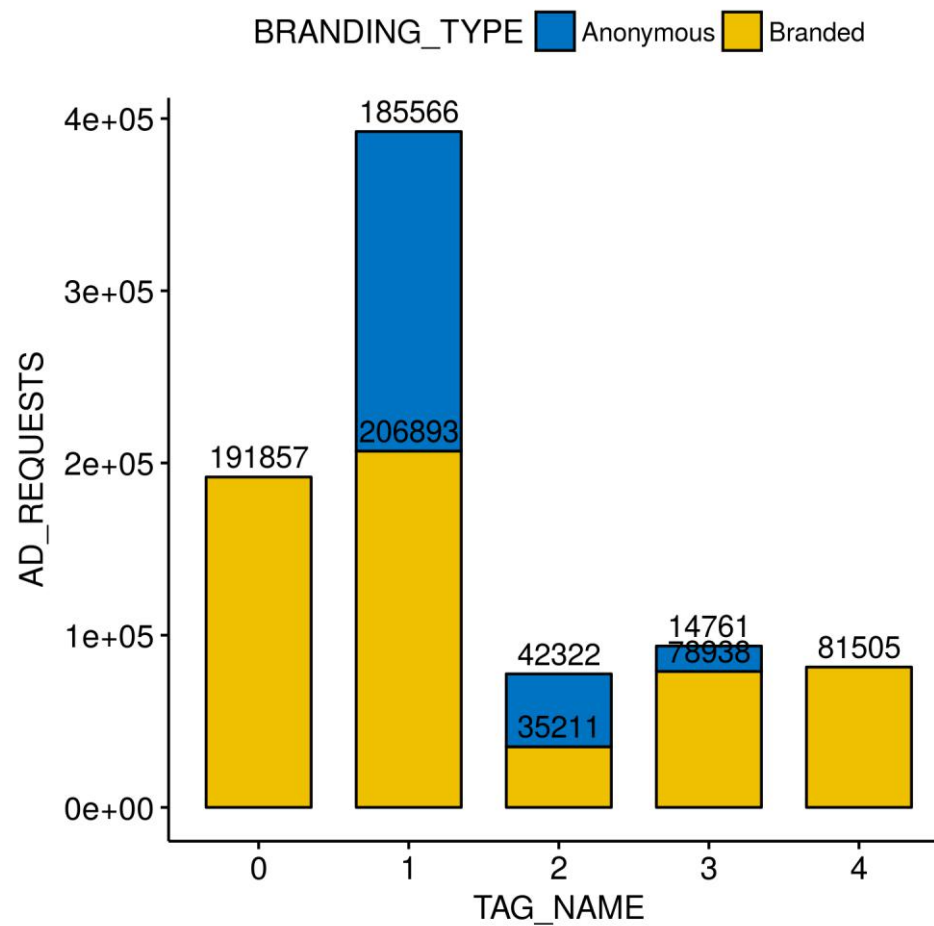


## 5.2 Wykres AdR ECPM z podziałem na Branding Type

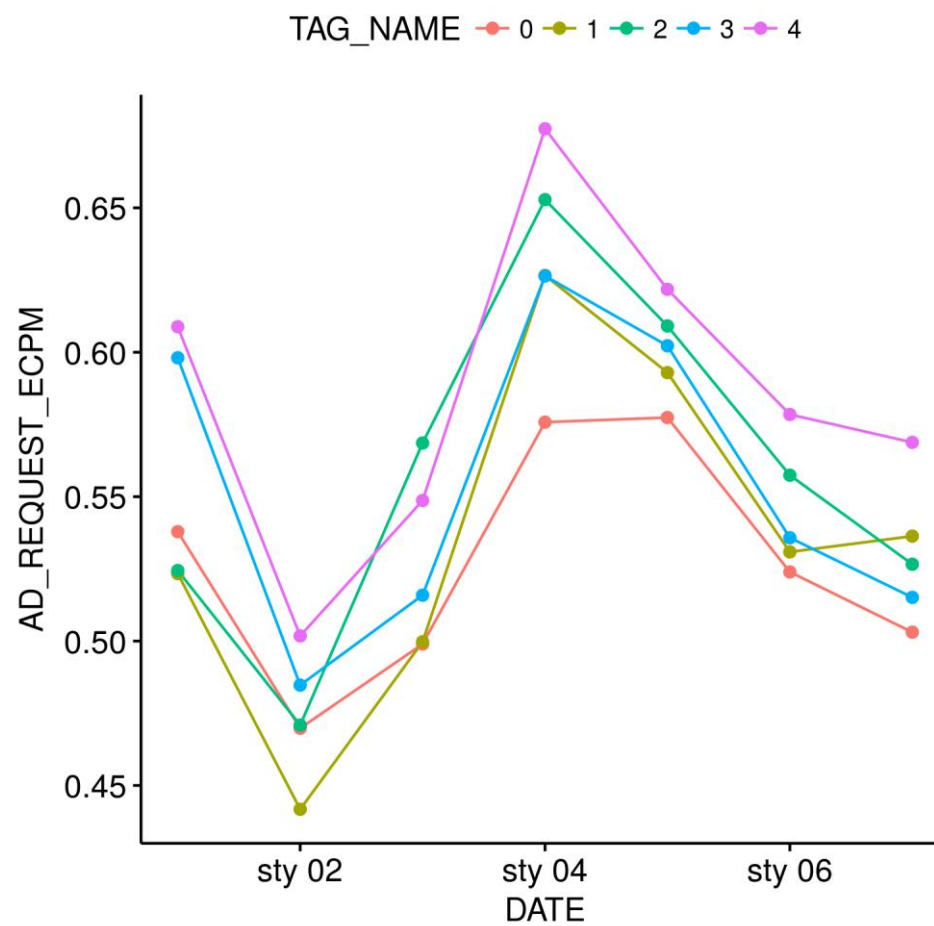




### 5.3 Wykres Ad requests z podziałem na Branding Type



## 5.4 Wykres trendu dziennego AdRequest eCPM



## 6 Eksperyment di\_test\_300x250\_desktop\_T\_ (od 2018-12-20 do 2018-12-30)

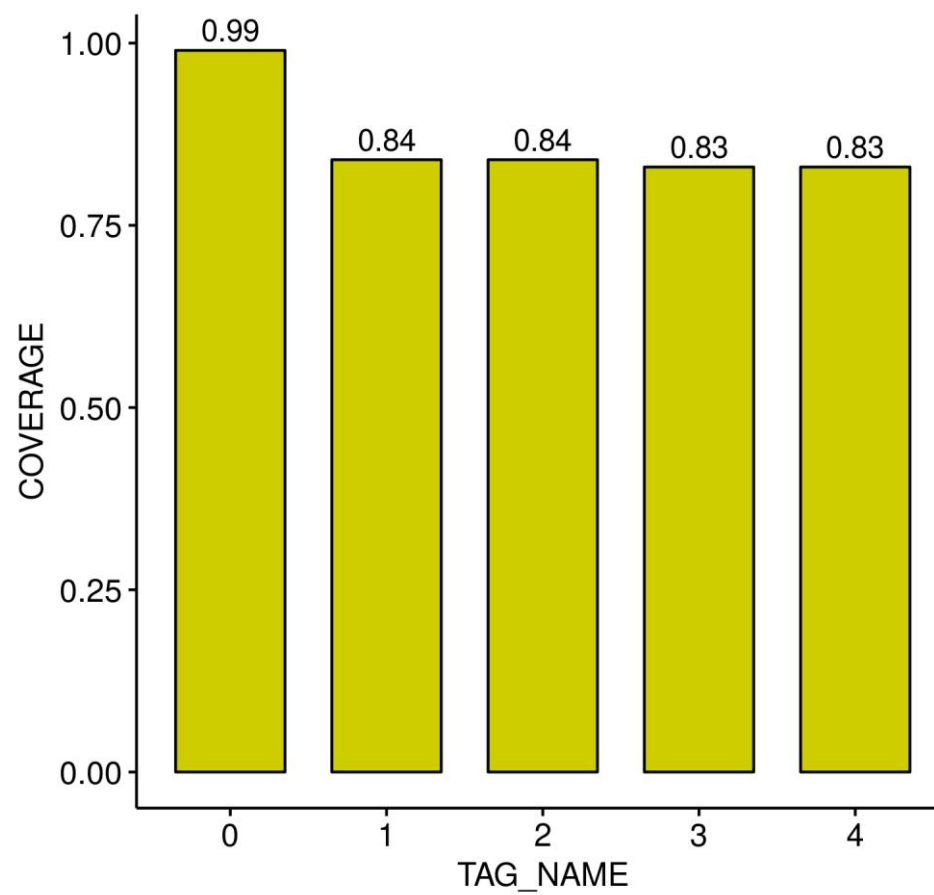
Data wygenerowania raportu: 2019-01-02

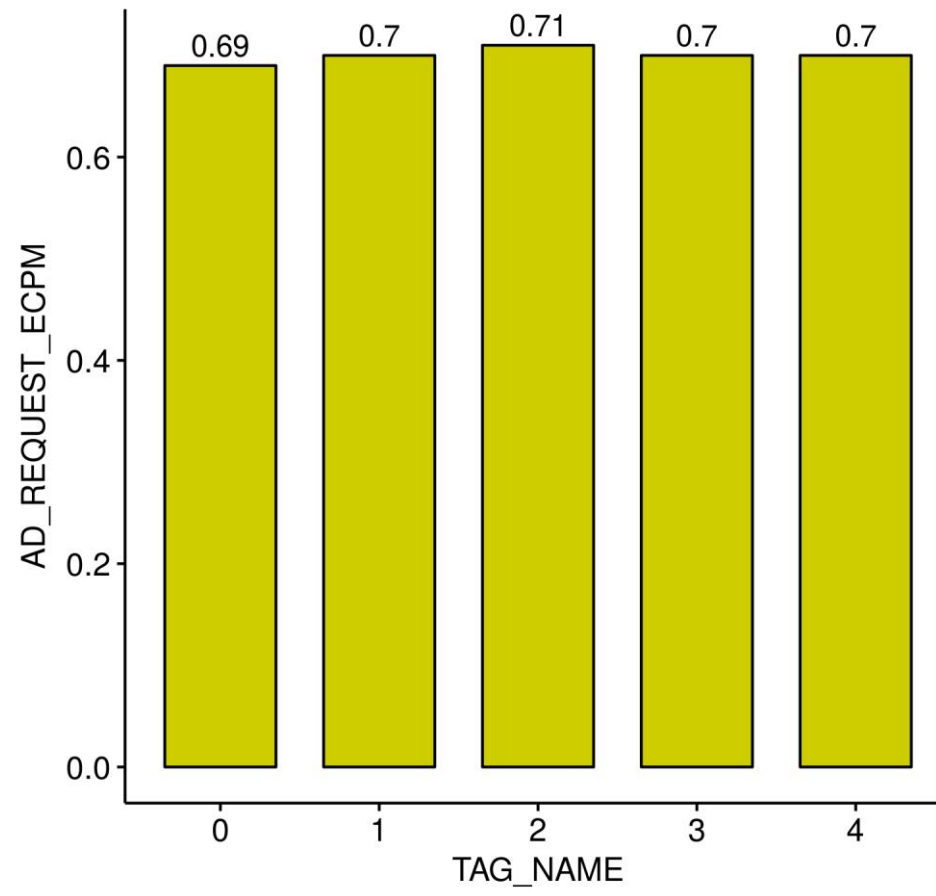
**Cel eksperymentu:** Testy rotacji przed właściwym eksperymentem. Na wszystkich tagach oprócz zerówki ustawione te same reguły.

**Wnioski:** Testy rotacji **przebiegły pomyślnie**. Na wszystkich tagach poziom zrównoważony (nawet na zerówce). Największe wahanie dzienne w dniu 2018-12-30 (0,07; 11%).

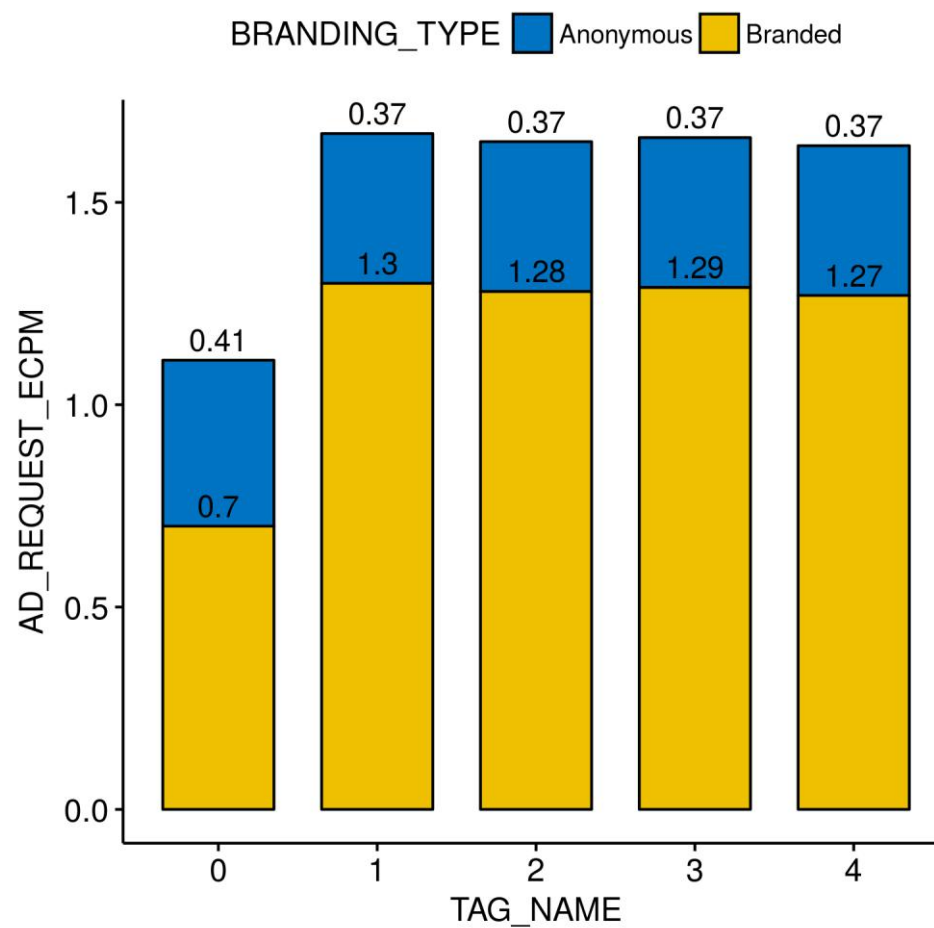
data	tag_ad_unit	rule_name	wejsciove_ branded	wejsciove_ anonymous	wyjsciove_ _branded	wyjsciove_anony mous	na
2018-12-21	di_test _300x250_desktop_T_2	di_test_300x2 50_desktop_T_2(Mob i Desk)	0,24/0,35	0,9/0,7	0	0	(mob/desk) reguła zmienona około 21 dnia 20.12

## 6.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

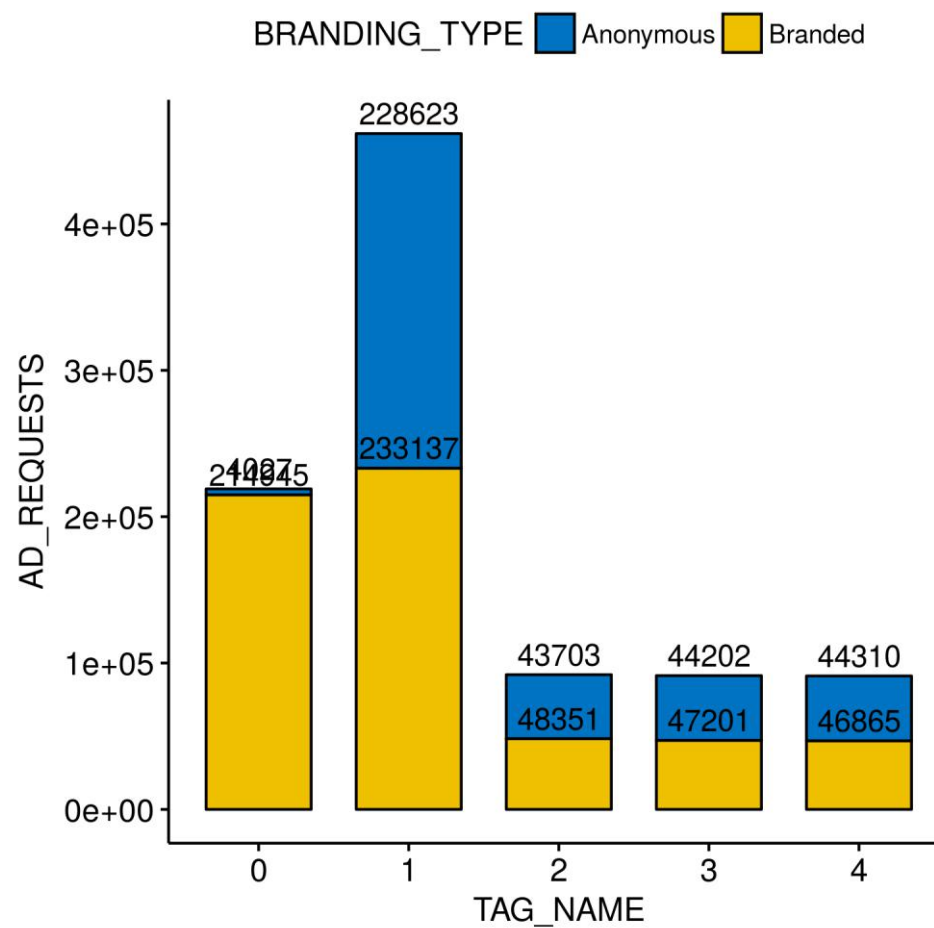




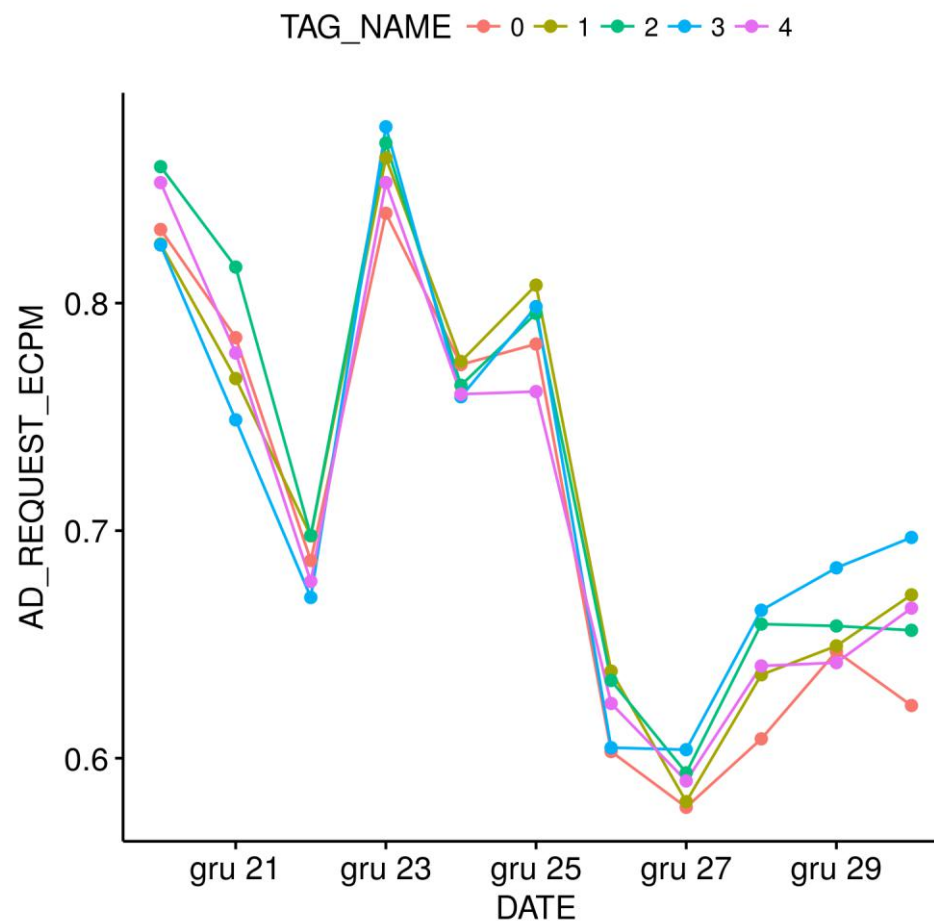
## 6.2 Wykres AdR ECPM z podziałem na Branding Type



### 6.3 Wykres Ad requests z podziałem na Branding Type



## 6.4 Wykres trendu dziennego AdRequest eCPM



## 6.5 Wahania dzienne AdRequest eCPM

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2018-12-20	0.825573	0.859970	0.034397	0.04081415



<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2018-12-21	0.748705	0.815842	0.067137	0.08582293
2018-12-22	0.670641	0.697857	0.027216	0.03977499
2018-12-23	0.839493	0.877491	0.037998	0.04426133
2018-12-24	0.758847	0.774365	0.015518	0.02024247
2018-12-25	0.761132	0.807907	0.046775	0.05962248
2018-12-26	0.602910	0.638265	0.035355	0.05697021
2018-12-27	0.578426	0.603747	0.025321	0.04283806
2018-12-28	0.608509	0.665029	0.056520	0.08876060
2018-12-29	0.641982	0.683667	0.041685	0.06288995
2018-12-30	0.623154	0.697001	0.073847	0.11187626

## 7 Eksperyment cg\_test\_300x250\_mobile\_2\_T\_ (od 2019-1-8 do 2019-1-16)

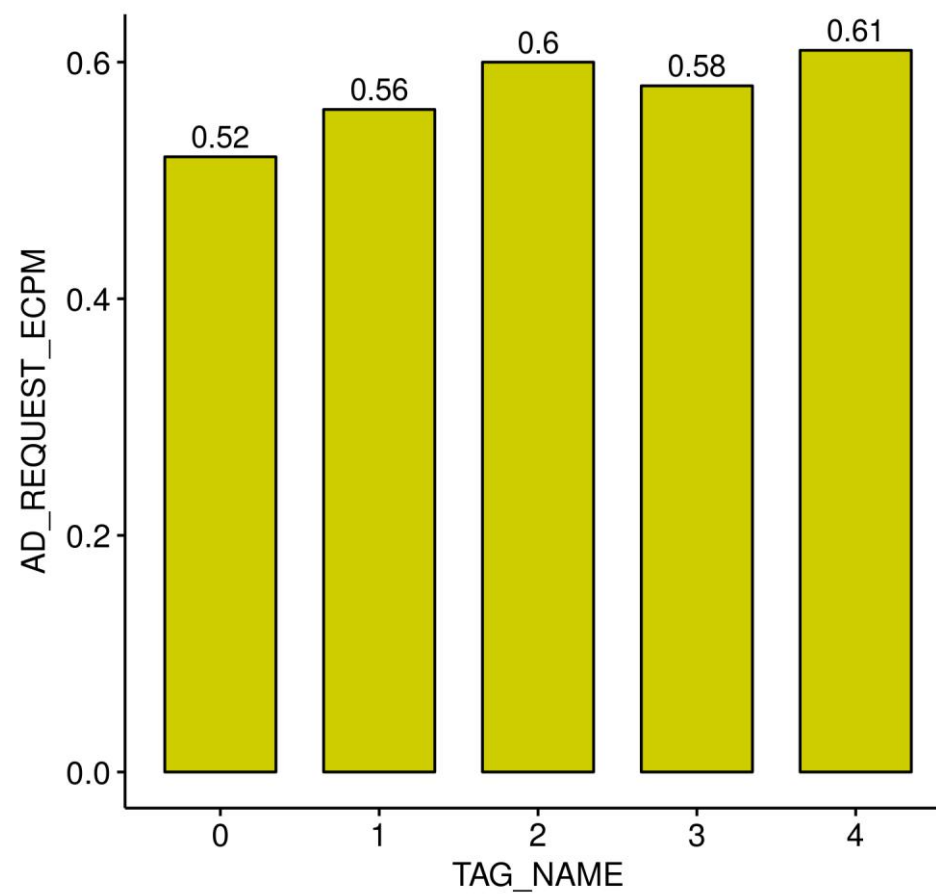
Data wygenerowania raportu: 2019-01-17

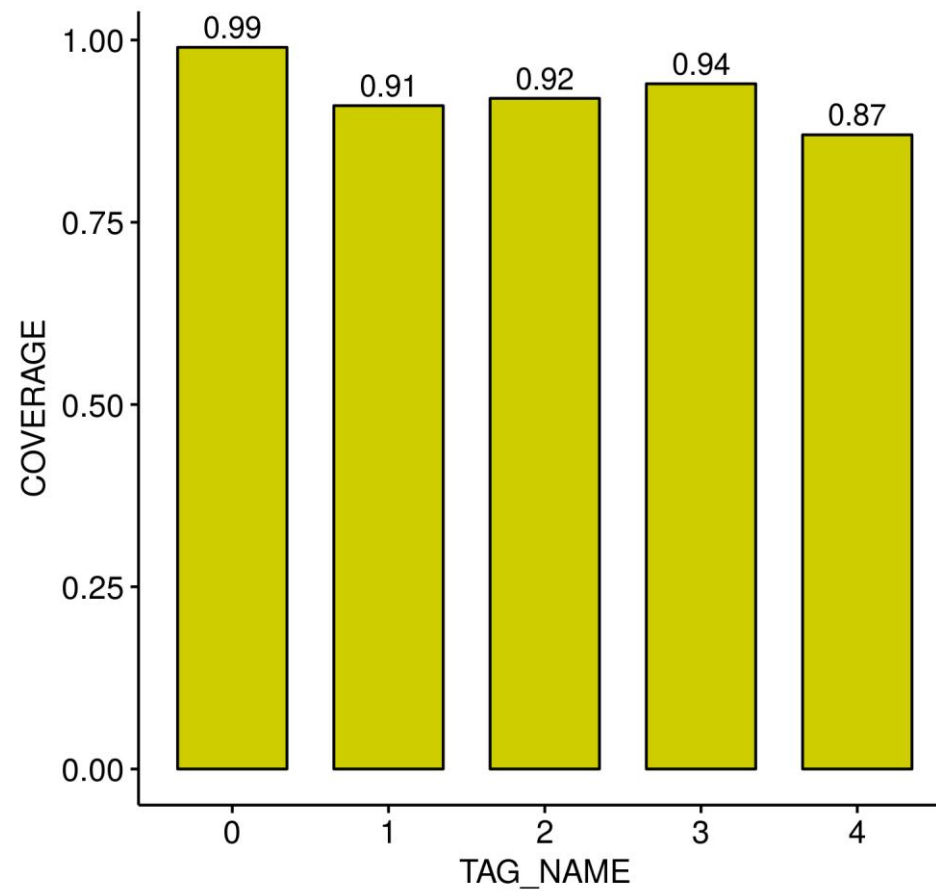
Cel: Zbadanie zależności licytacji A/B. Na \_3 ustawione A = 0.11 B = 0.22; na \_4: A zablokowane, B = 0.22 (takie samo jak na \_3)

Obserwacje: Obniżone ARe na \_3 w porównaniu do \_4 – nieznacznie. Najniższe Coverage na \_4 (ze względu na najwyższą regułę)

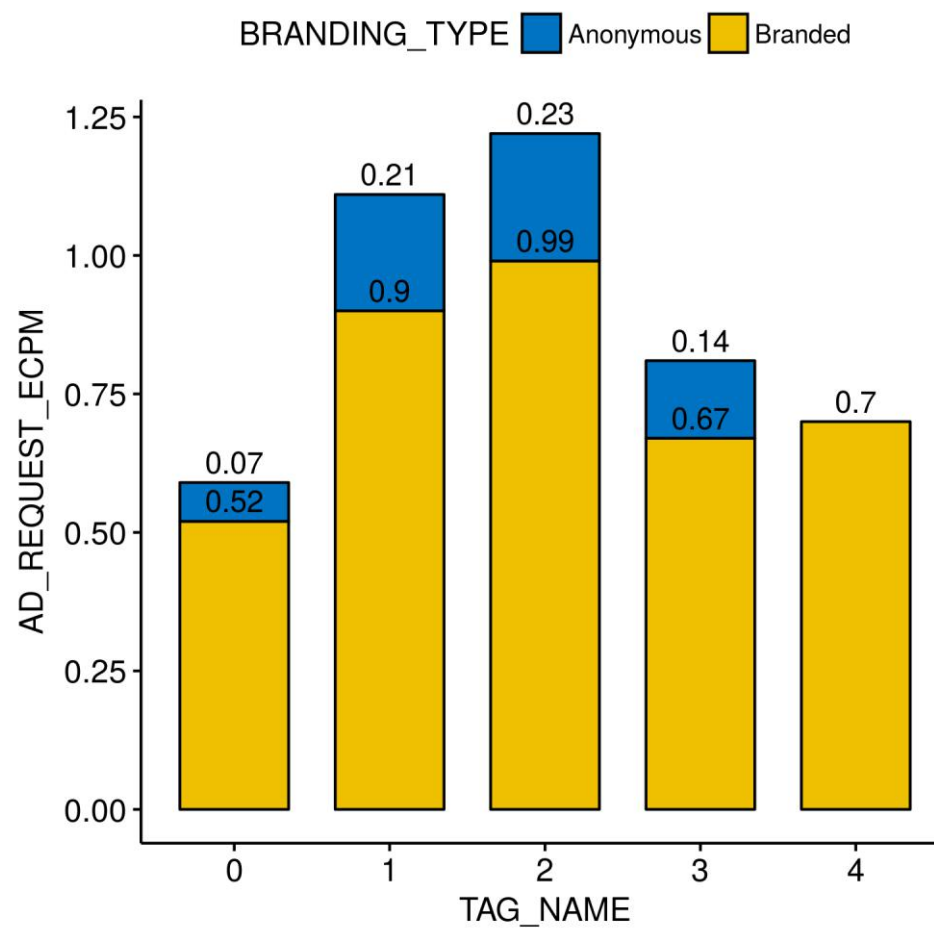
data	tag_ad_unit	rule_name	wejsciove_branded	wejsciove_anonymous	wyjsciove_branded	wyjsciove_anonymous	uwagi
2018-12-18	cg_test_300x250_mobile_2_T_0	cg_test_300x250_mobile_2_T_0	0.9	0.22	0	0	
2019-01-07	cg_test_300x250_mobile_2_T_1	cg_test_300x250_mobile_2_T_1	0.6	0.15	0.45	0.1	
2019-01-07	cg_test_300x250_mobile_2_T_2	cg_test_300x250_mobile_2_T_2	0.72	0.17	0.57	0.12	
2019-01-08	cg_test_300x250_mobile_2_T_3	cg_test_300x250_mobile_2_T_3_A	0.22	0	0.22	0.11	
2018-12-19	cg_test_300x250_mobile_2_T_4	cg_test_300x250_mobile_2_T_4_B	0.9	-	0.22	-	

## 7.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

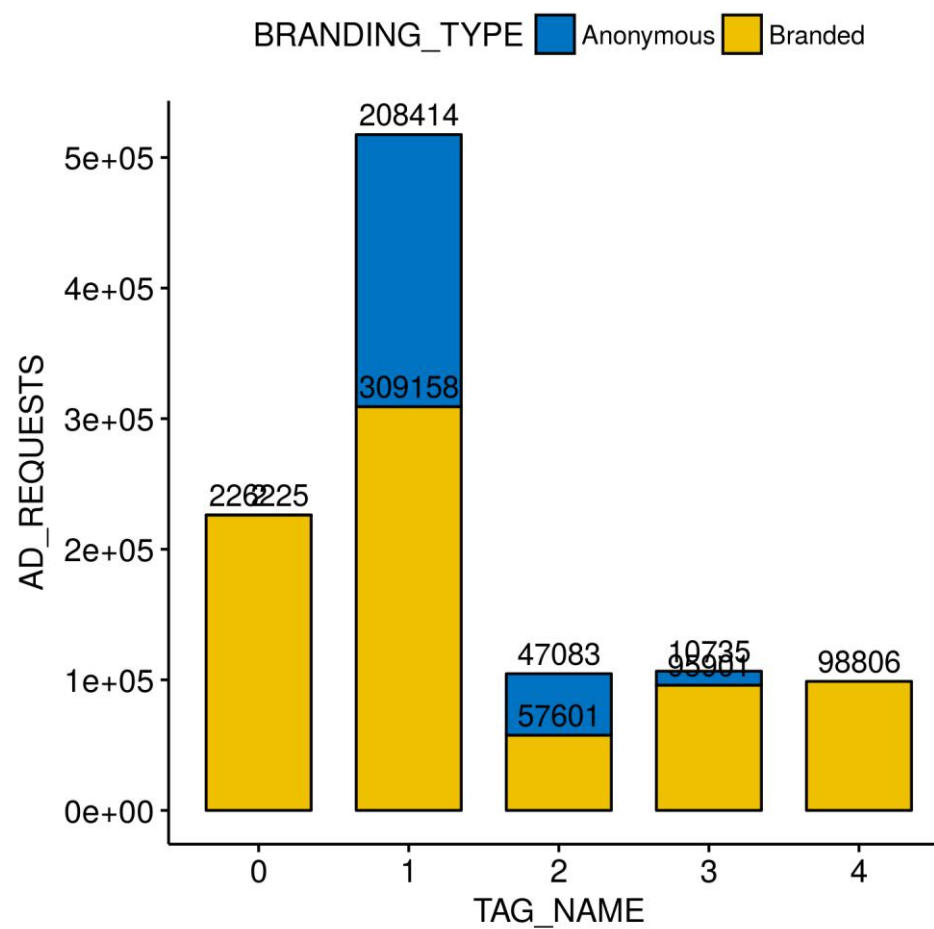




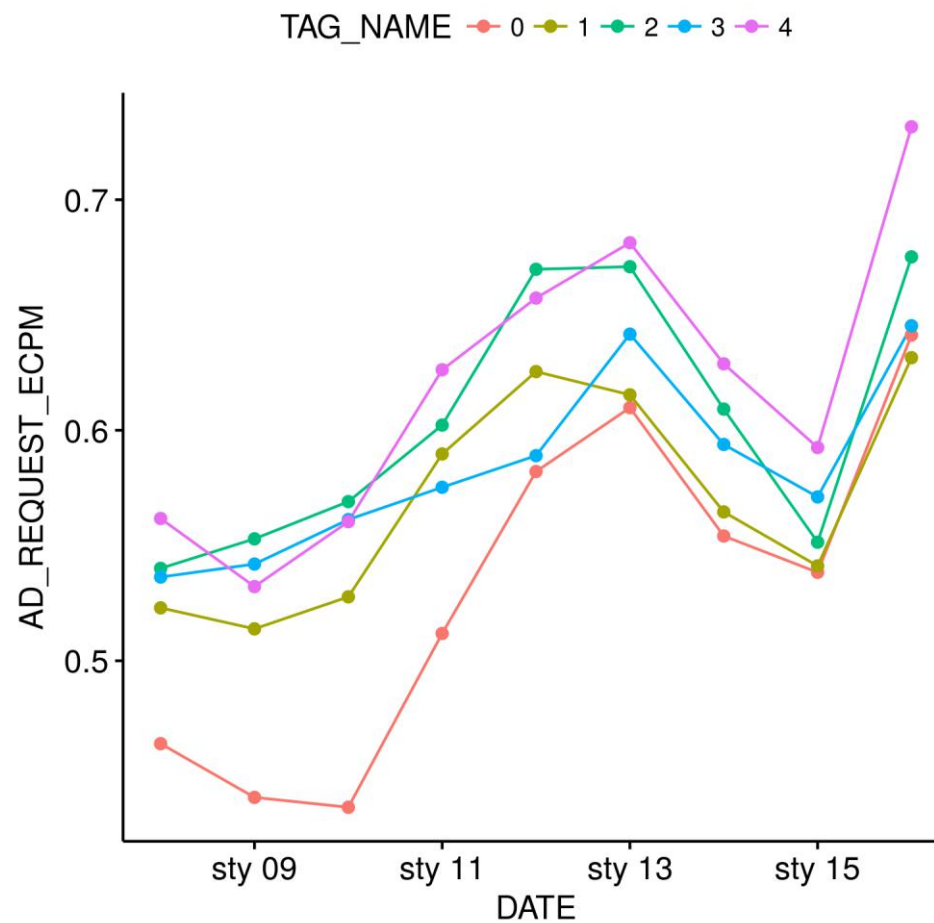
## 7.2 Wykres AdR ECPM z podziałem na Branding Type



### 7.3 Wykres Ad requests z podziałem na Branding Type



#### 7.4 Wykres trendu dziennego AdRequest eCPM



## 8 Eksperyment cg\_test\_300x250\_mobile\_2\_T\_ (od 2019-1-20 do 2019-1-31)

Data wygenerowania raportu: 2019-02-01

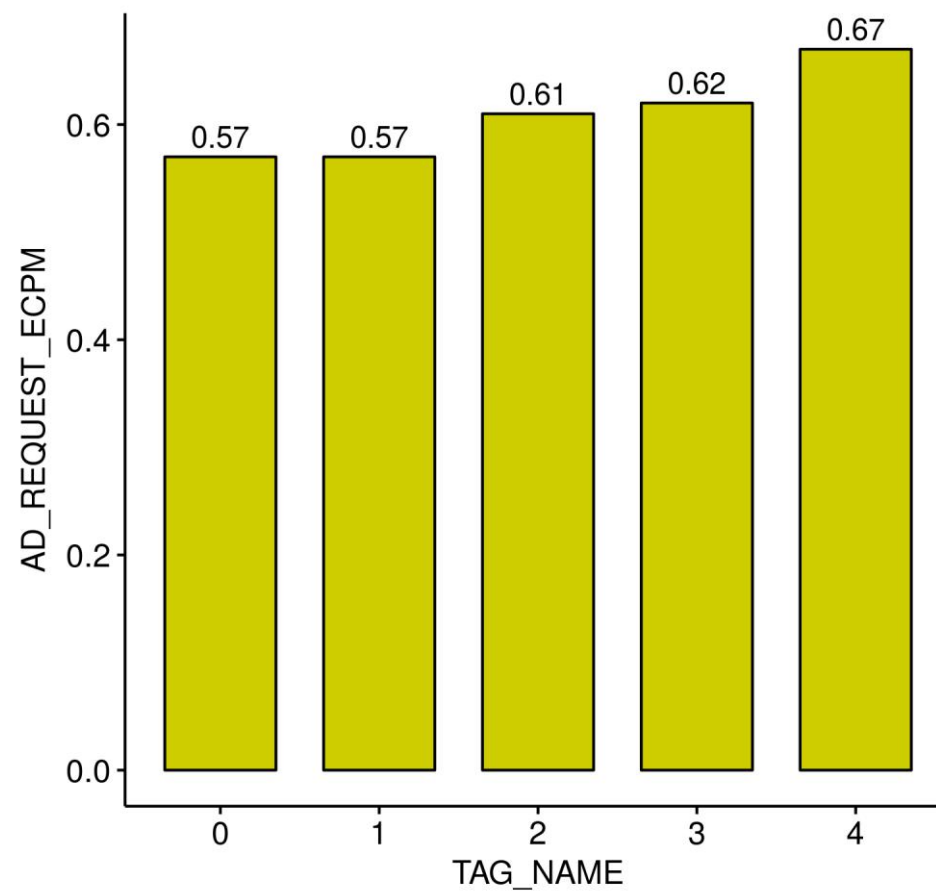
Cel eksperymentu: Zbadanie zależności licytacji B/A. Na \_3 ustawione reguły A = 0.22; B = 0.26; na \_4 A zablokowane, B = 0.22

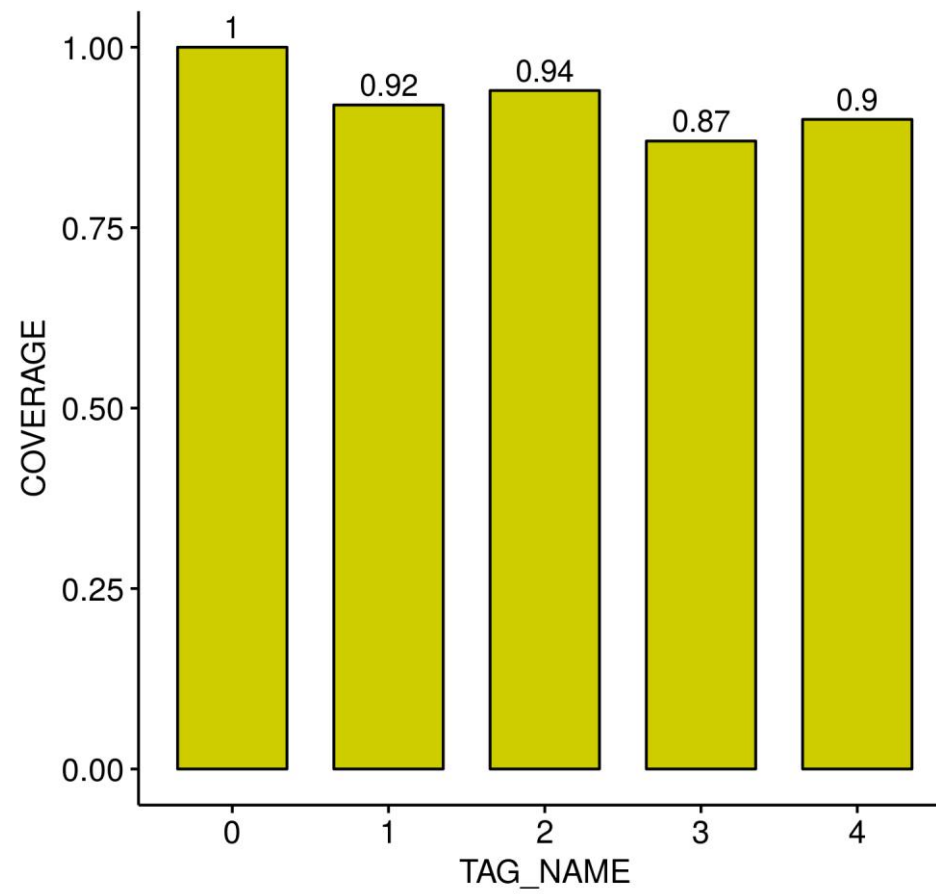
Wnioski: Na tej grupie tagów głównymi licytującymi są Branded – po podwyższeniu reguły na B, obniżenie znaczne coverage i ARe (na \_3 w porównaniu do \_4).

data	tag_ad_unit	wejsciove_branded	wejsciove_anonymous	wyjsciove_branded	wyjsciove_anonymous
2018-12-18	cg_test_300x250_mobile_2_T_0	0.9	0.22	<b>0</b>	<b>0</b>
2019-01-10	cg_test_300x250_mobile_2_T_1	0.45	0.1	<b>0.5</b>	<b>0.12</b>
2019-01-10	cg_test_300x250_mobile_2_T_2	0.57	0.12	<b>0.54</b>	<b>0.11</b>
2019-01-18	cg_test_300x250_mobile_2_T_3	0.22	0	<b>0.26</b>	<b>0.22</b>
2018-12-19	cg_test_300x250_mobile_2_T_4	0.9	-	<b>0.22</b>	-

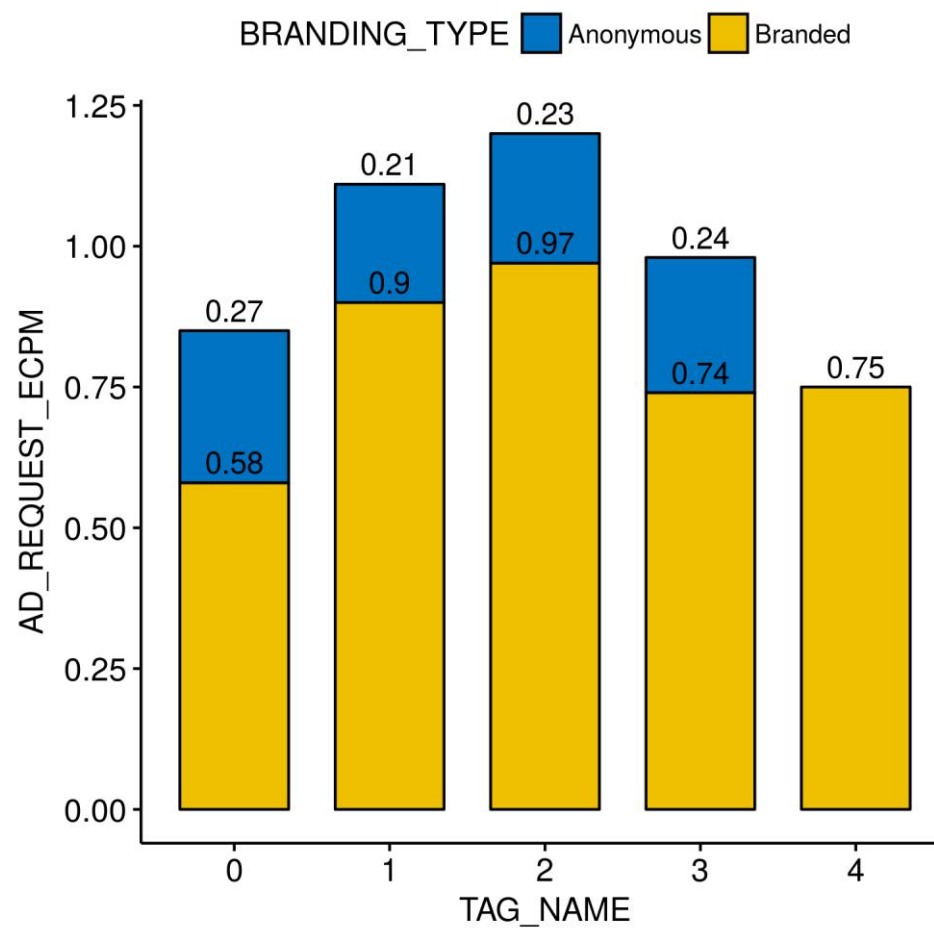


## 8.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

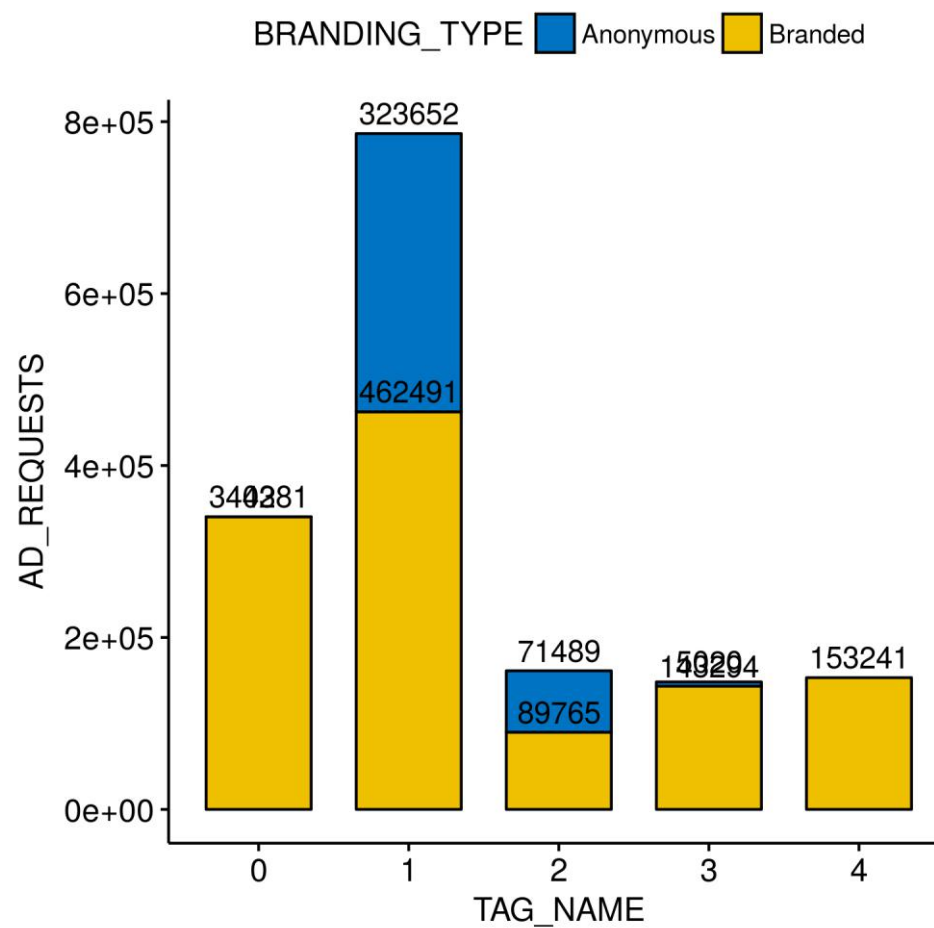




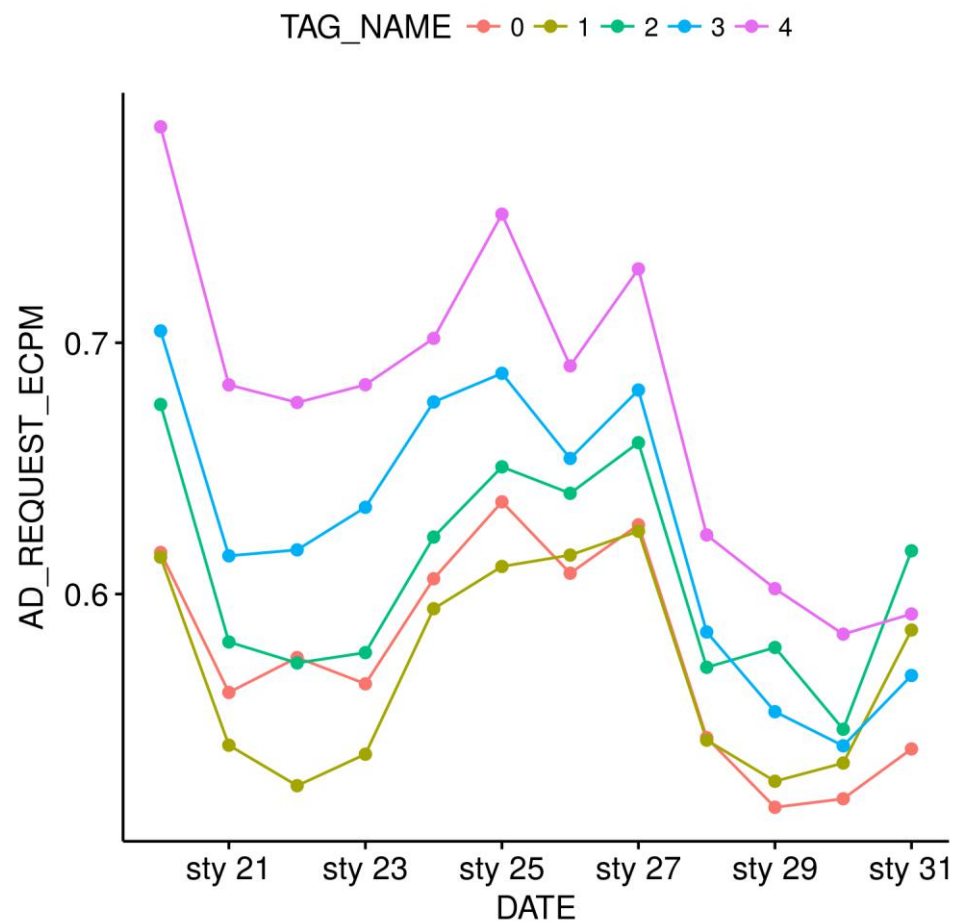
## 8.2 Wykres AdR ECPM z podziałem na Branding Type



### 8.3 Wykres Ad requests z podziałem na Branding Type



## 8.4 Wykres trendu dziennego AdRequest eCPM



## 9 Eksperyment di\_test\_300x250\_desktop\_T\_ (od 2019-1-1 do 2019-1-16)

Data wygenerowania raportu: 2019-01-22

**Cel eksperymentu:** Ustalenie czy bity są zależne od floorów. Poboczny cel: optymalizacja mobile + desktop. W międzyczasie optymalizacja przez YR na \_1. Reguły ustawione rosnąco: na \_0 zerówka, na \_1 YR (poniżej 0.38), na \_2 0.38, na \_3 0.75 i na \_4 1.13.

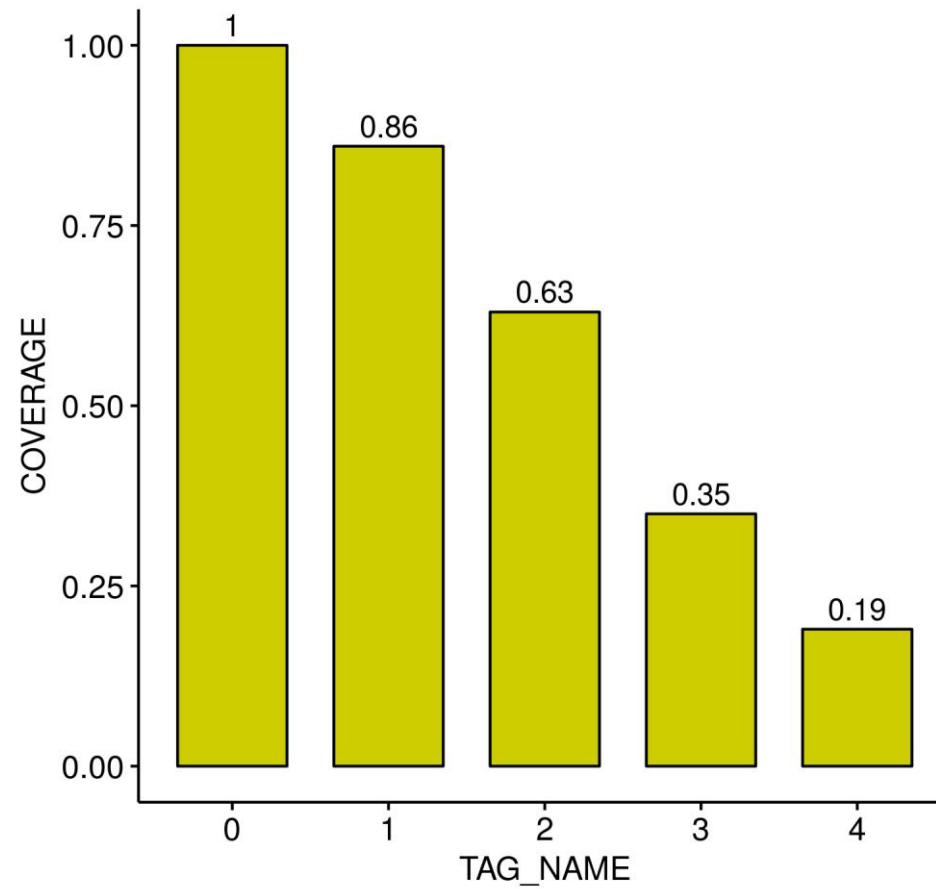
### Wnioski:

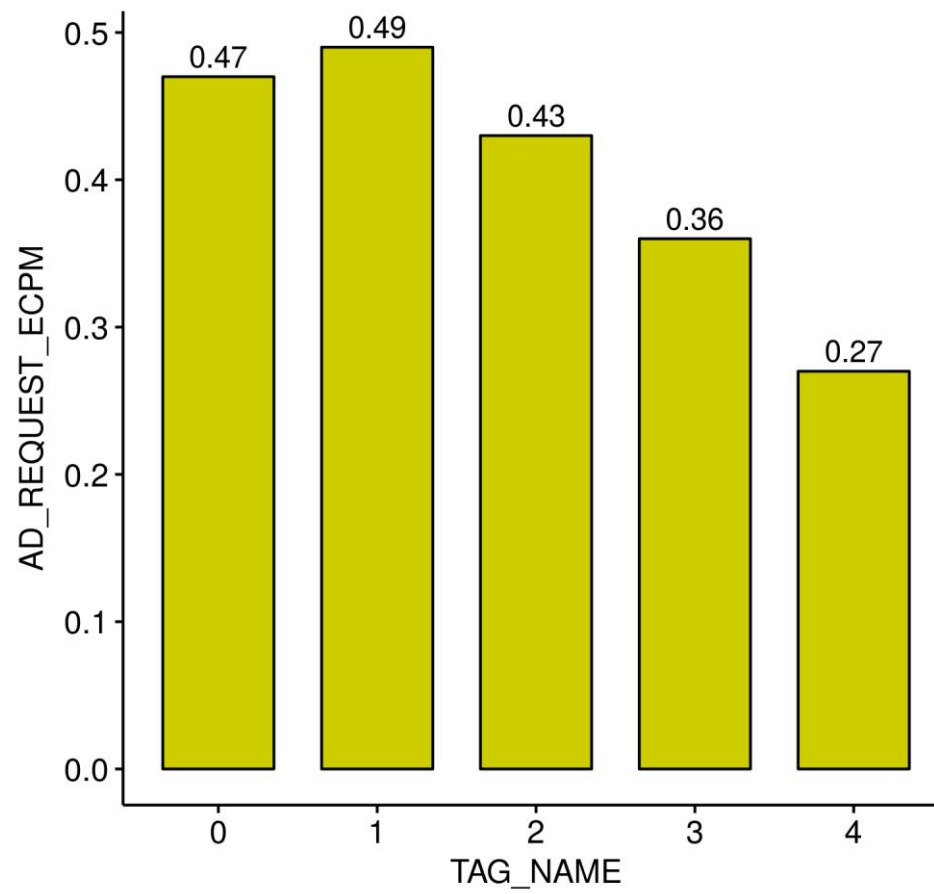
1. Ustawione reguły w niewielkim stopniu wpływają na licytację (wykres 6). Największy udział procentowy bidów 0.00 – 0.38 na tagu \_0, ale przybliżone wartości występują na pozostałych tagach. \_1 funkcjonuje najdłużej na rynku – najniższy odsetek bidów 0.00 – 0.38, na tagach \_2, \_3 i \_4 odsetek podobny.
2. Wraz ze wzrostem floorów obniżka na Coverage (wykres 1). Najbliżej optymalnych reguł jest tag \_1 (wykres 2).
3. Główni odbiorcy tych tagów korzystają z Desktop i Mobile devices (wykres 7 drugi Ad Requests). Najlepsze ARe dla mobile i desktop na \_1 – wniosek: optymalne floory są w przedziale 0.00 – 0.38. Na wyższych poziomach odpowiednia obniżka ARe (już dla 0.38 jest obniżka).

data	tag_ad_unit	rule_name	wyjsciowe_branded	wyjsciowe_anonymous
2018-12-21	_0	Mob + desktop	<b>0</b>	<b>0</b>
2018-12-31	_2	Mobile	<b>0.38</b>	<b>0.38</b>
2018-12-31	_3	Mobile	<b>0.75</b>	<b>0.75</b>
2018-12-31	_4	Mobile	<b>1.13</b>	<b>1.13</b>
2018-12-31	_2	Desktop	<b>0.38</b>	<b>0.38</b>
2018-12-31	_3	Desktop	<b>0.75</b>	<b>0.75</b>
2018-12-31	_4	Desktop	<b>1.13</b>	<b>1.13</b>
2019-01-02	_1	Mobile	0.75	0.18

data	tag_ad_unit	rule_name	wyjsciove_branded	wyjsciove_anonymous
2019-01-02	_1	Desktop	0.6	0.3
2019-01-07	_1	Mobile	0.7	0.16

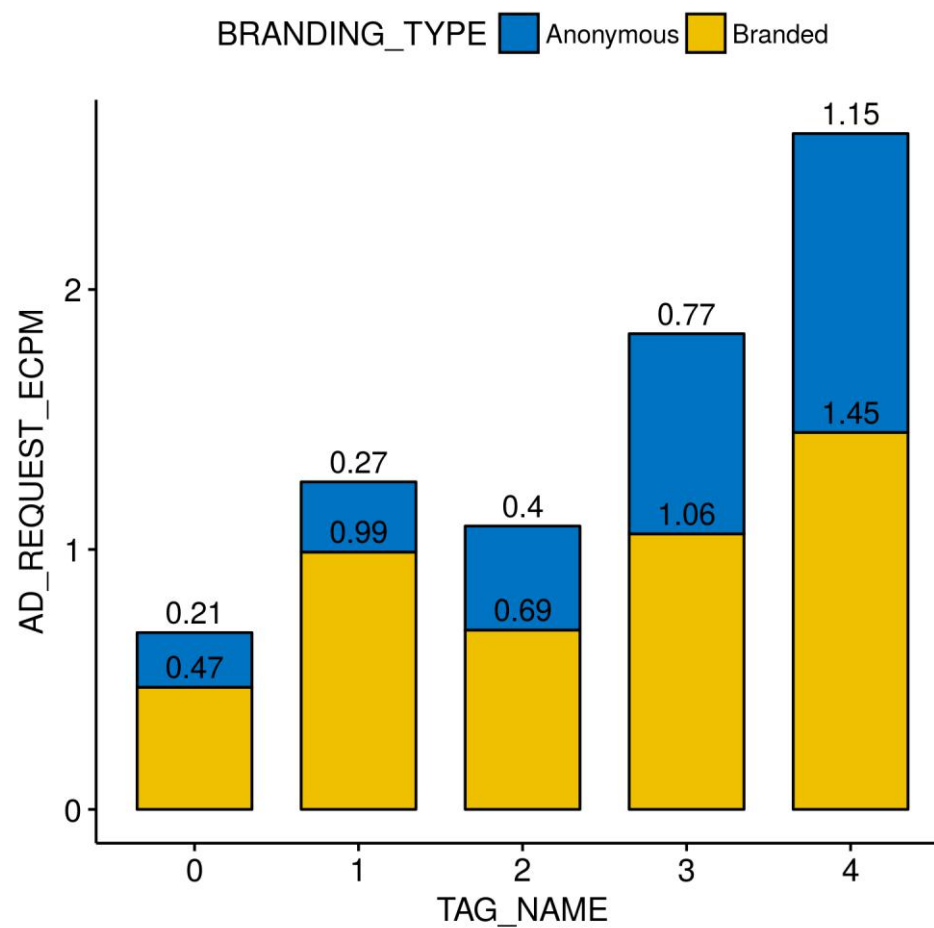
### 9.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



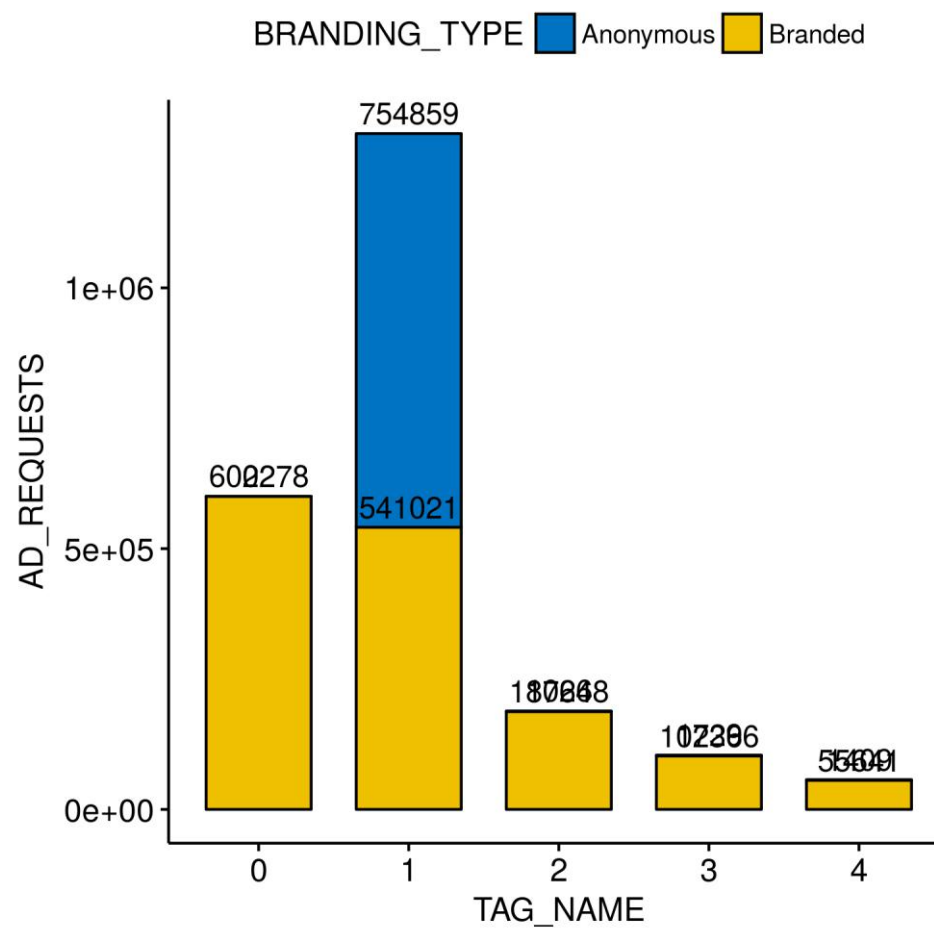




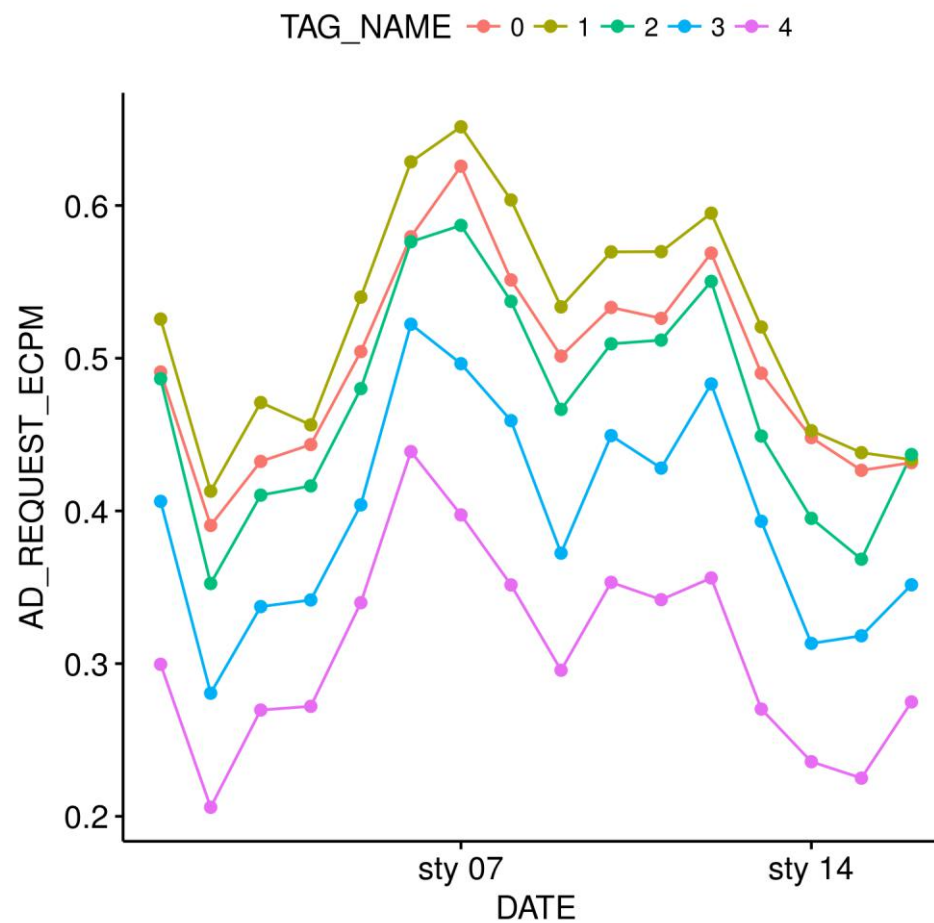
## 9.2 Wykres AdR eCPM z podziałem na Branding Type



### 9.3 Wykres Ad requests z podziałem na Branding Type



## 9.4 Wykres trendu dziennego AdRequest eCPM



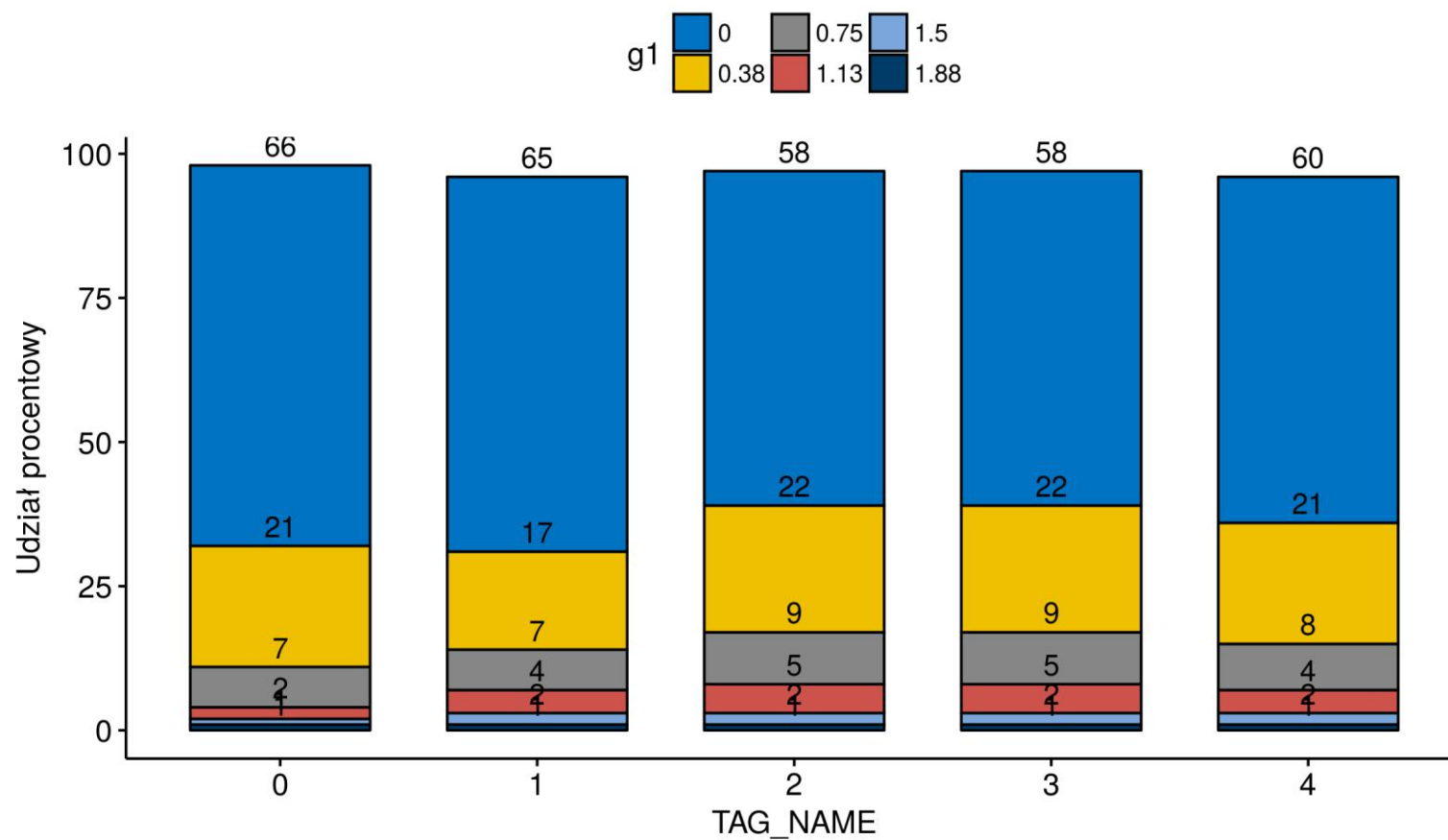
## 9.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-01	0.299553	0.525667	0.226114	0.5480090

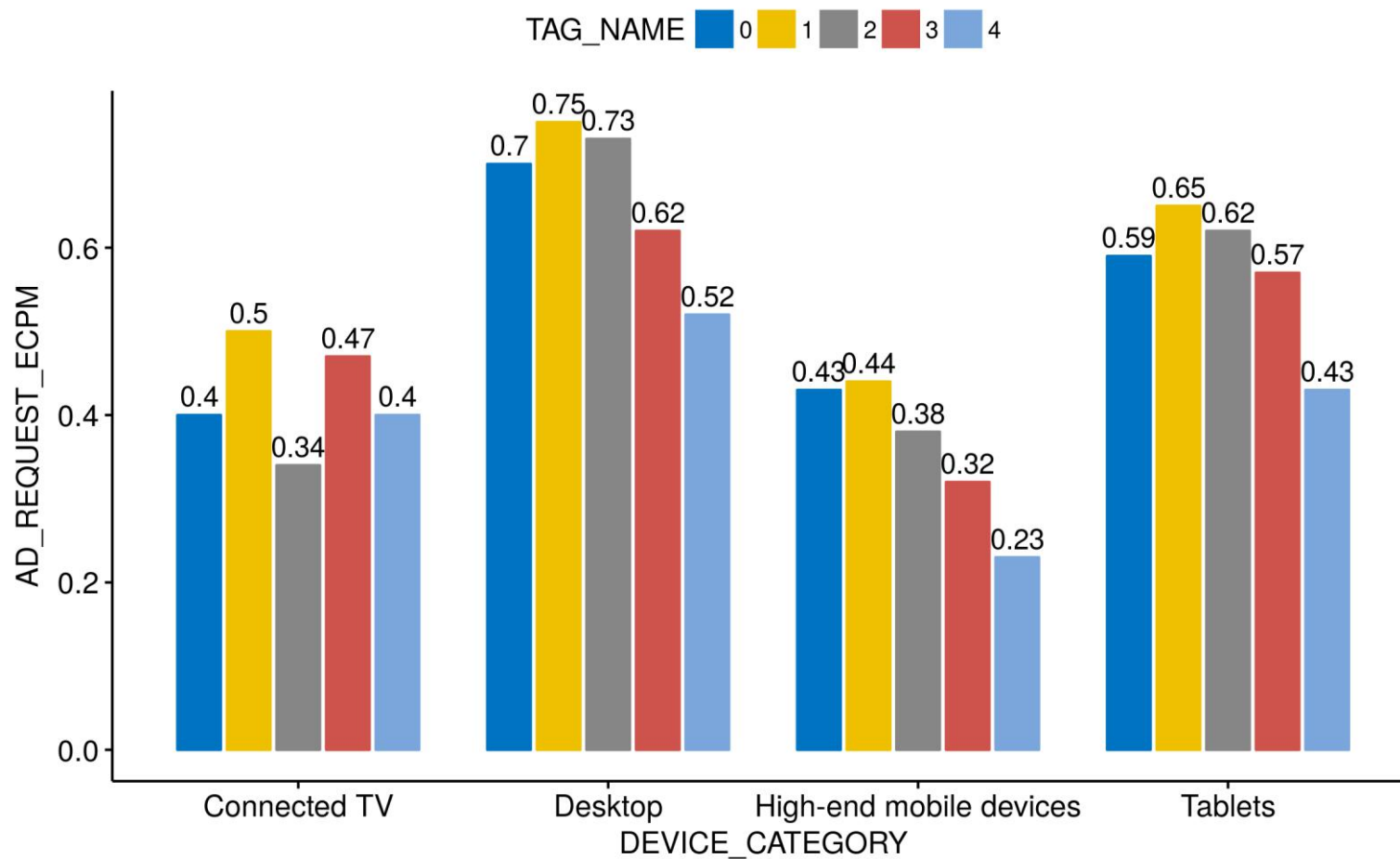
<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-02	0.205916	0.412929	0.207013	0.6690302
2019-01-03	0.269624	0.470985	0.201361	0.5437714
2019-01-04	0.272041	0.456369	0.184328	0.5061106
2019-01-05	0.339916	0.540016	0.200100	0.4548079
2019-01-06	0.438852	0.628589	0.189737	0.3554988
2019-01-07	0.397396	0.651556	0.254160	0.4845980
2019-01-08	0.351541	0.603634	0.252093	0.5278467
2019-01-09	0.295671	0.533622	0.237951	0.5738647
2019-01-10	0.353221	0.569620	0.216399	0.4689844
2019-01-11	0.341876	0.569800	0.227924	0.5000110
2019-01-12	0.356085	0.595004	0.238919	0.5024114
2019-01-13	0.270218	0.520457	0.250239	0.6329756
2019-01-14	0.235762	0.452513	0.216751	0.6298384
2019-01-15	0.225009	0.438168	0.213159	0.6428420
2019-01-16	0.274912	0.436915	0.162003	0.4551752

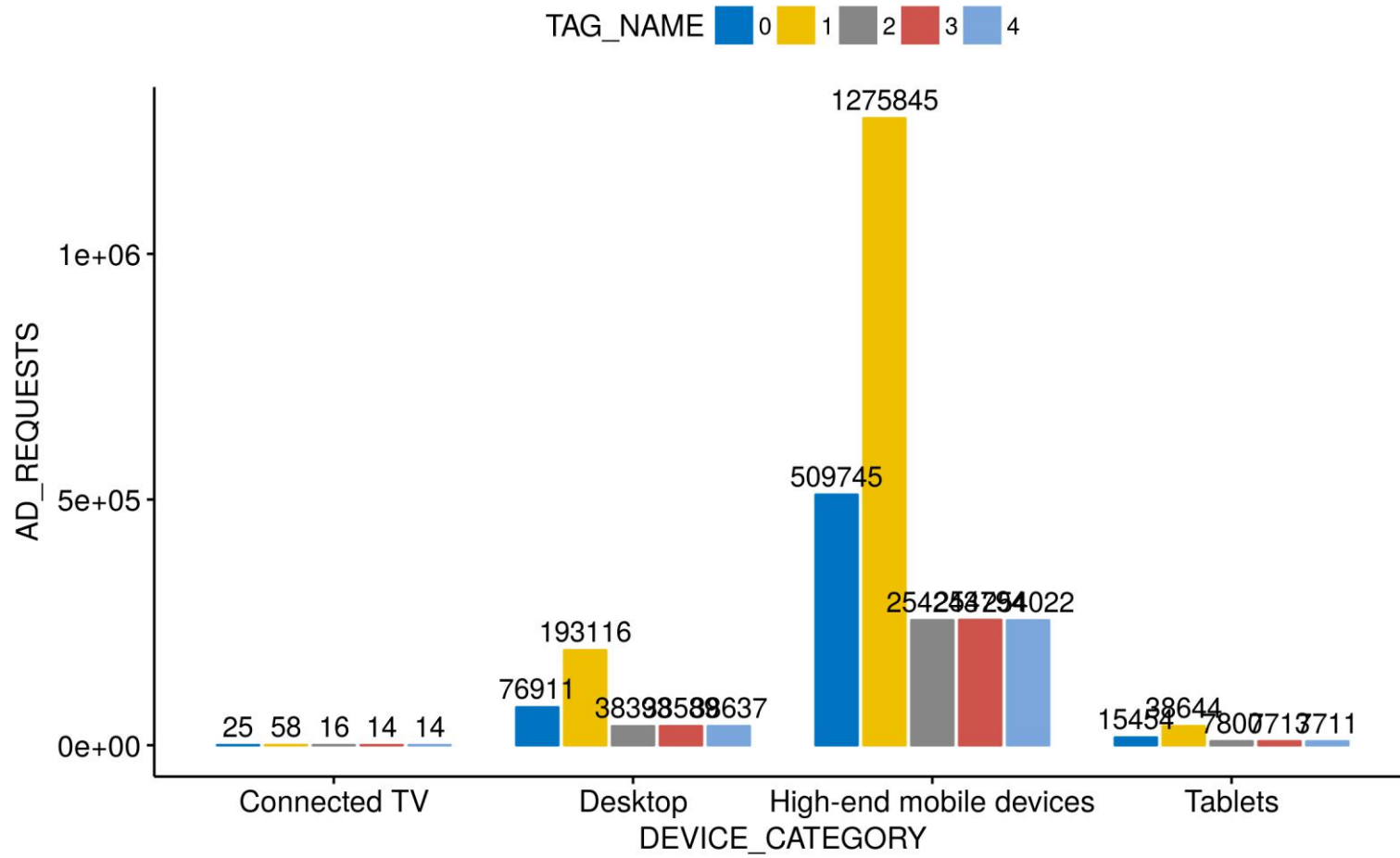
## 9.6 Udział procentowy bidów w podziale na TAG\_NAME i na BID\_RANGES

Kolory oznaczają dolne granice przedziałów bidowania



### 9.7 ARE i Requests w podziale na TAG\_NAME i DEVICE\_CATEGORY





## 10 Eksperyment cg\_test\_336x280\_1\_T\_ (od 2019-1-12 do 2019-1-16)

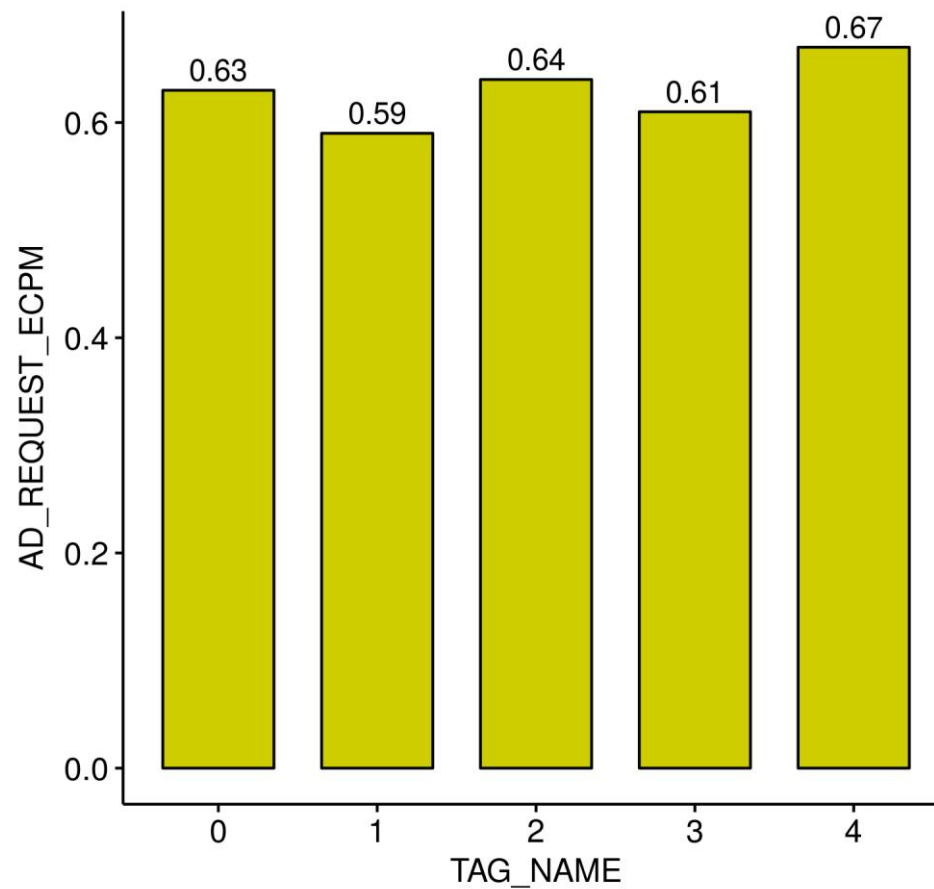
Data wygenerowania raportu: 2019-01-17

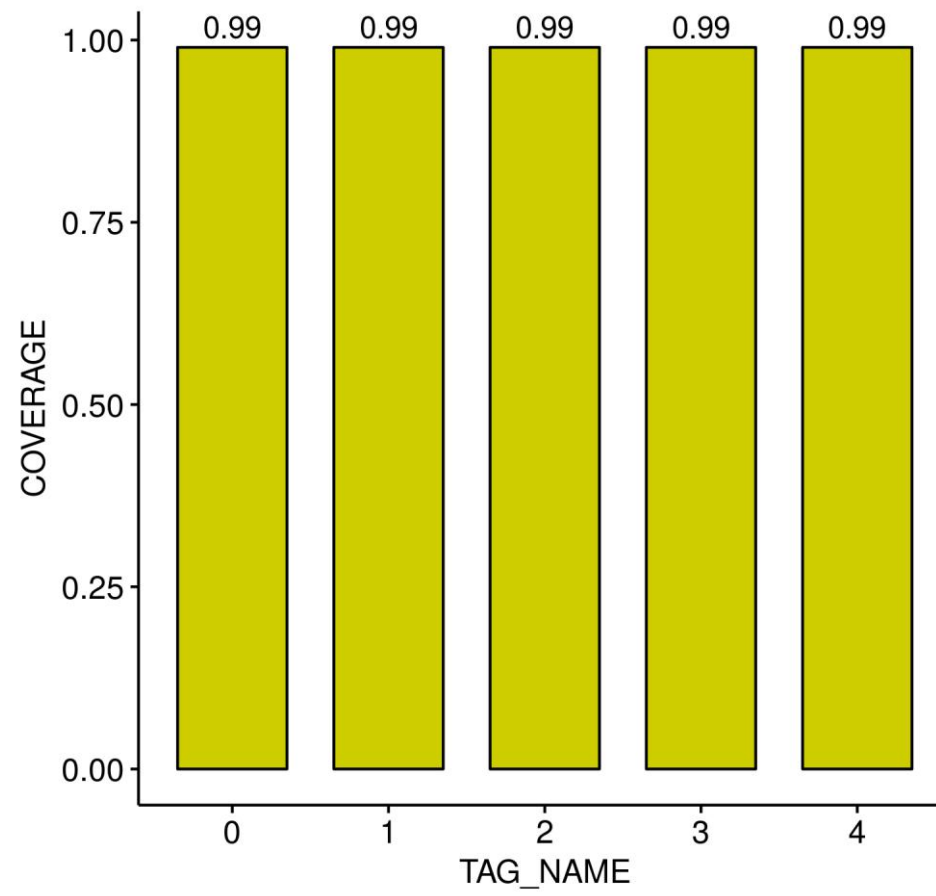
Cel: Kolejny eksperyment ma na celu zbadanie stabilności reguł z  $A = B = 0$ . Chcemy zbadać jak bardzo wahają się zerówki, być może są bardziej stabilne niż inne reguły (ze względu na większą liczbę licytujących) i nie ma potrzeby na tagu zerówka zajmowała aż 20%.

Wnioski: Wahania dzienne ARe 27% tuż po ustawieniu, w kolejnych dniach maksymalnie 16%. Coverage za cały okres na wszystkich tagach 99% - taki sam.

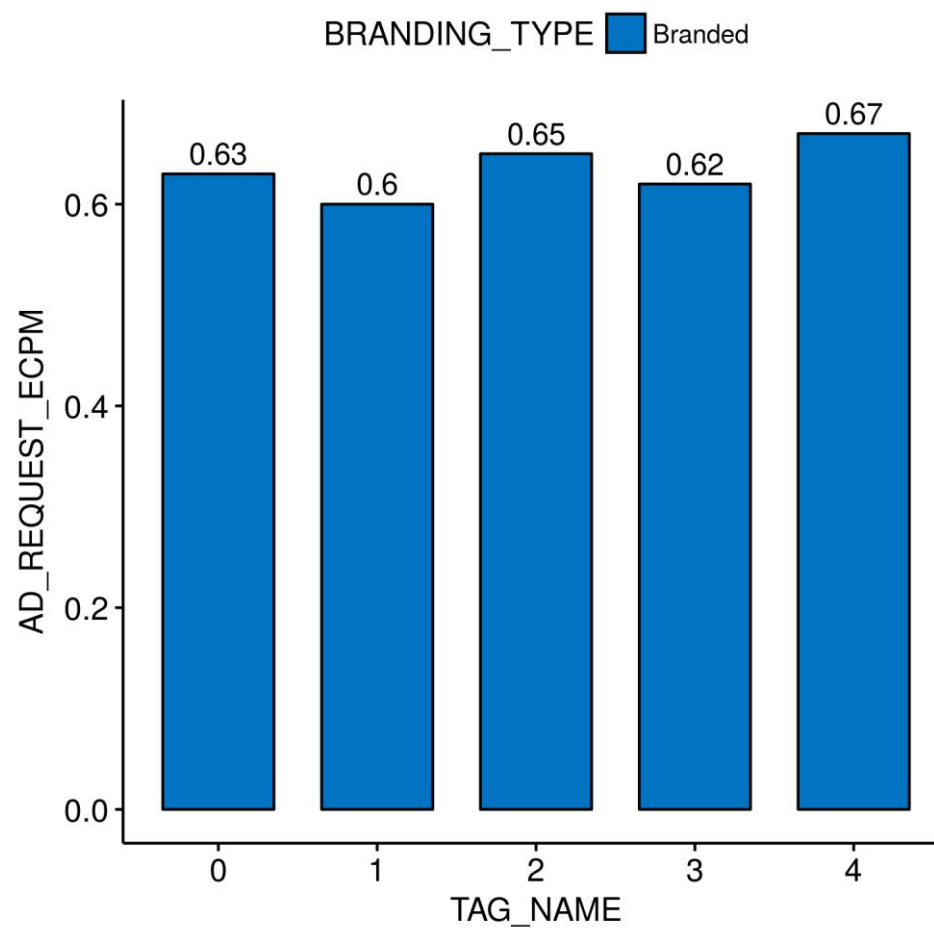


### 10.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

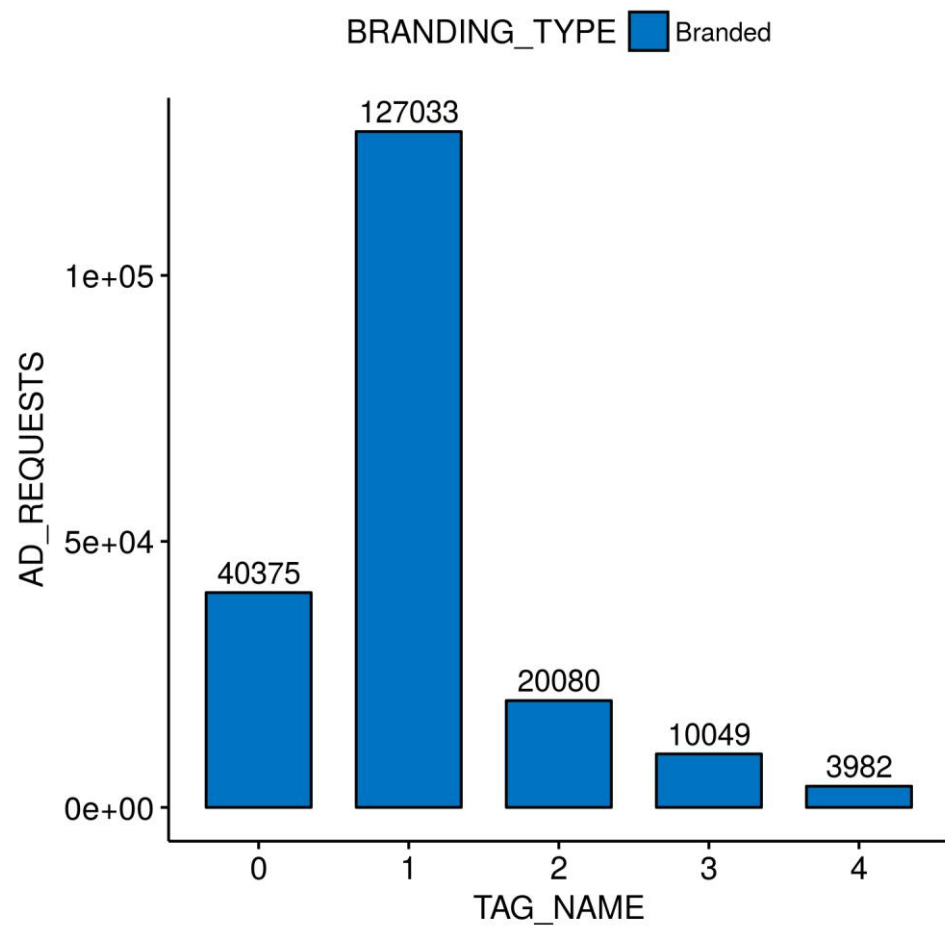




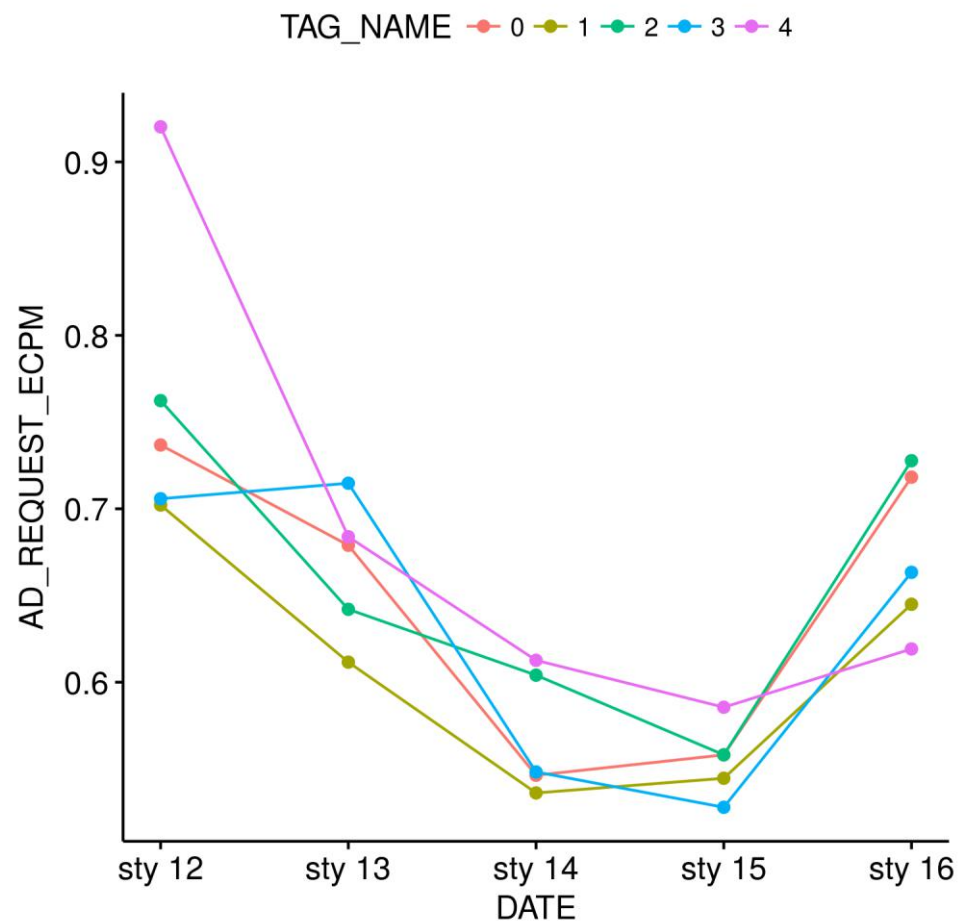
## 10.2 Wykres AdR ECPM z podziałem na Branding Type



### 10.3 Wykres Ad requests z podziałem na Branding Type



### 10.4 Wykres trendu dziennego AdRequest eCPM



### 10.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-12	0.702150	0.920257	0.218107	0.2688684

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-13	0.611557	0.714728	0.103171	0.1555789
2019-01-14	0.536139	0.612623	0.076484	0.1331590
2019-01-15	0.527872	0.585564	0.057692	0.1036288
2019-01-16	0.619176	0.727724	0.108548	0.1611820

## 11 Eksperyment cg\_test\_336x280\_1\_T\_ (od 2019-1-20 do 2019-1-31)

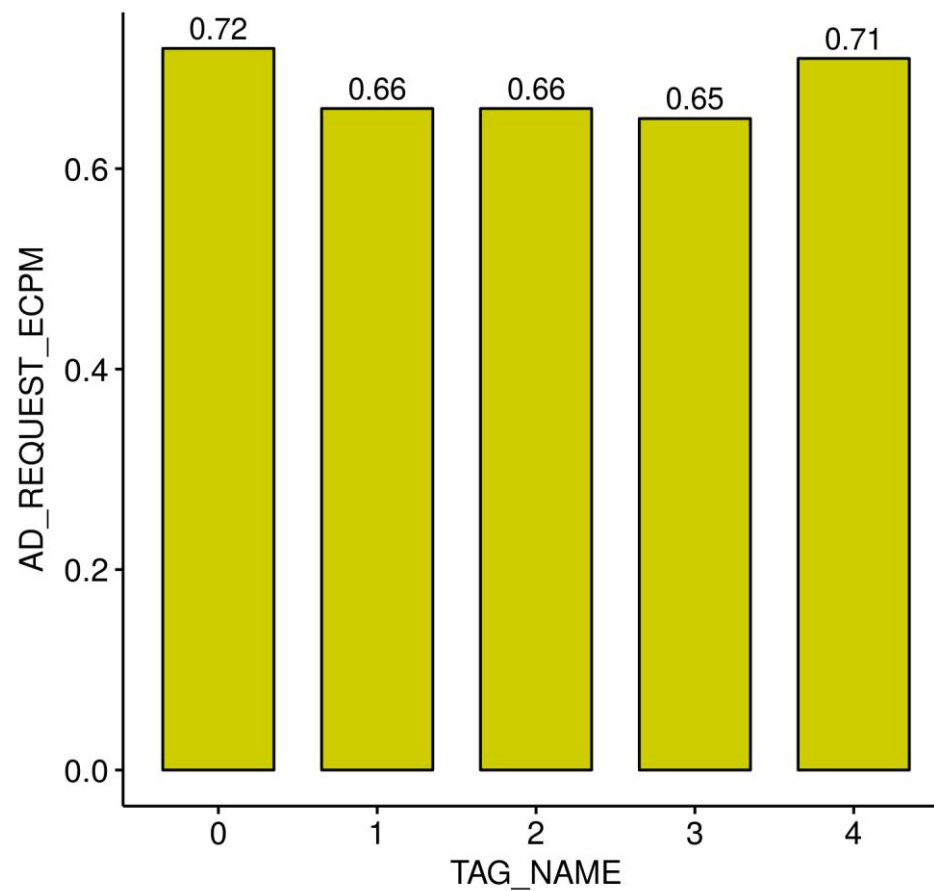
Data wygenerowania raportu: 2019-02-01

data	tag_ad_unit	rule_name	wejsciowe_brand ed	wejsciowe_anonymo us	wyjsciowe_brand ed	wyjsciowe_anonym ous
2019-01-10	cg_test_336x280_1_T_0	cg_test_336x280_1_T_0	0.5	0.1	0	0
2019-01-11	cg_test_336x280_1_T_1	cg_test_336x280_1	0.5	0.1	0	0
2019-01-11	cg_test_336x280_1_T_2	cg_test_336x280_1	0.5	0.1	0	0
2019-01-11	cg_test_336x280_1_T_3	cg_test_336x280_1	0.5	0.1	0	0
2019-01-11	cg_test_336x280_1_T_4	cg_test_336x280_1	0.5	0.1	0	0

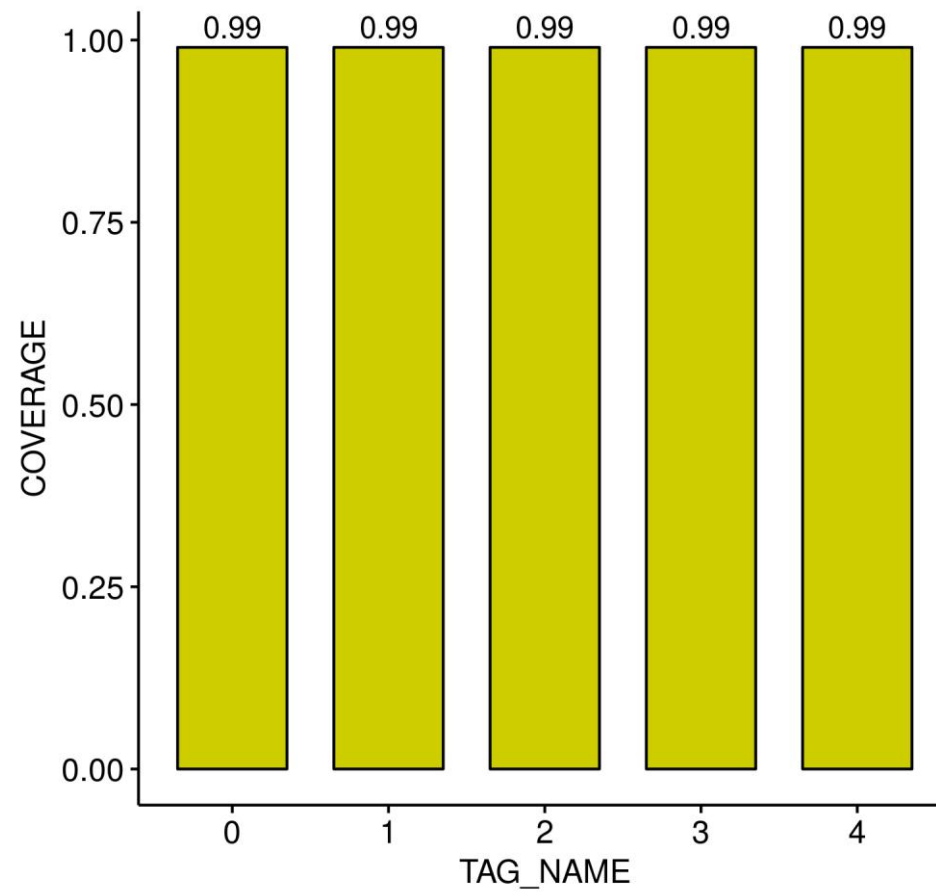
**Cel:** Kolejny eksperyment ma na celu zbadanie stabilności reguł z  $A = B = 0$ . Chcemy zbadać jak bardzo wahają się zerówki, być może są bardziej stabilne niż inne reguły (ze względu na większą liczbę licytujących) i nie ma potrzeby na tagu zerówka zajmowała aż 20%.

**Wnioski:** Ustalenie rotacji po równo (1/1/1/1) daje bardziej stabilne rezultaty (do 16% dziennie). W między czasie była kampania na tej paczce tagów.

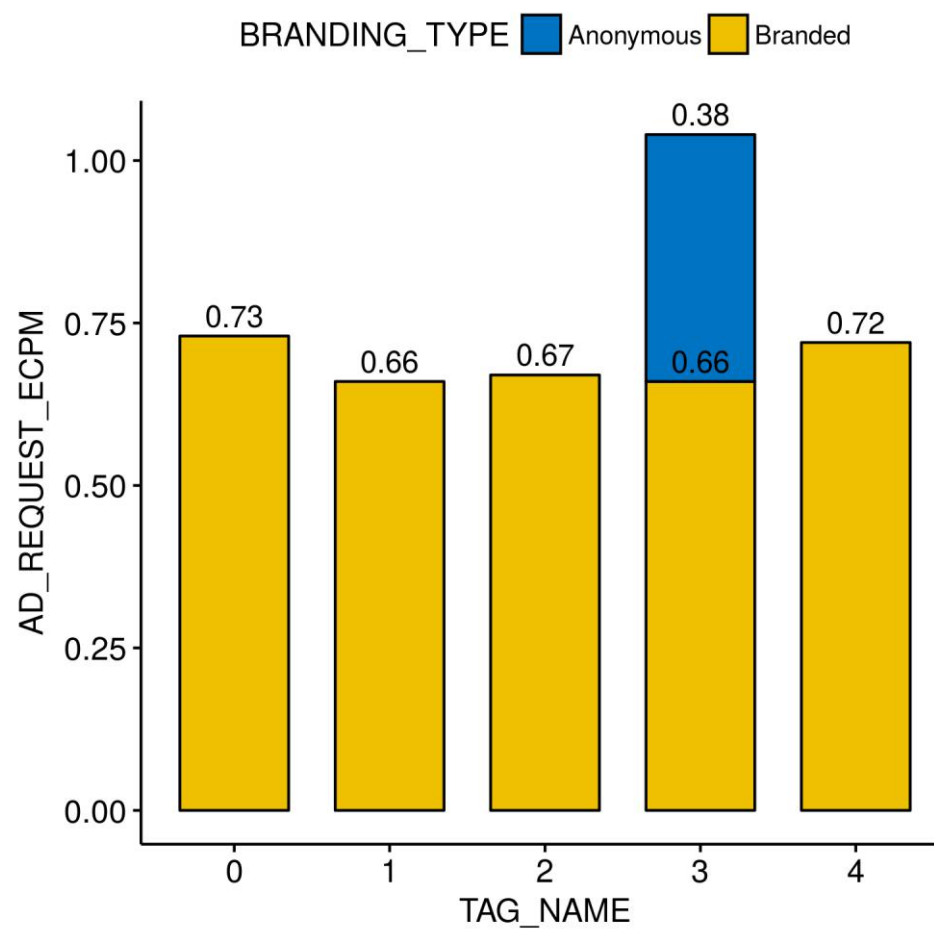
### 11.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



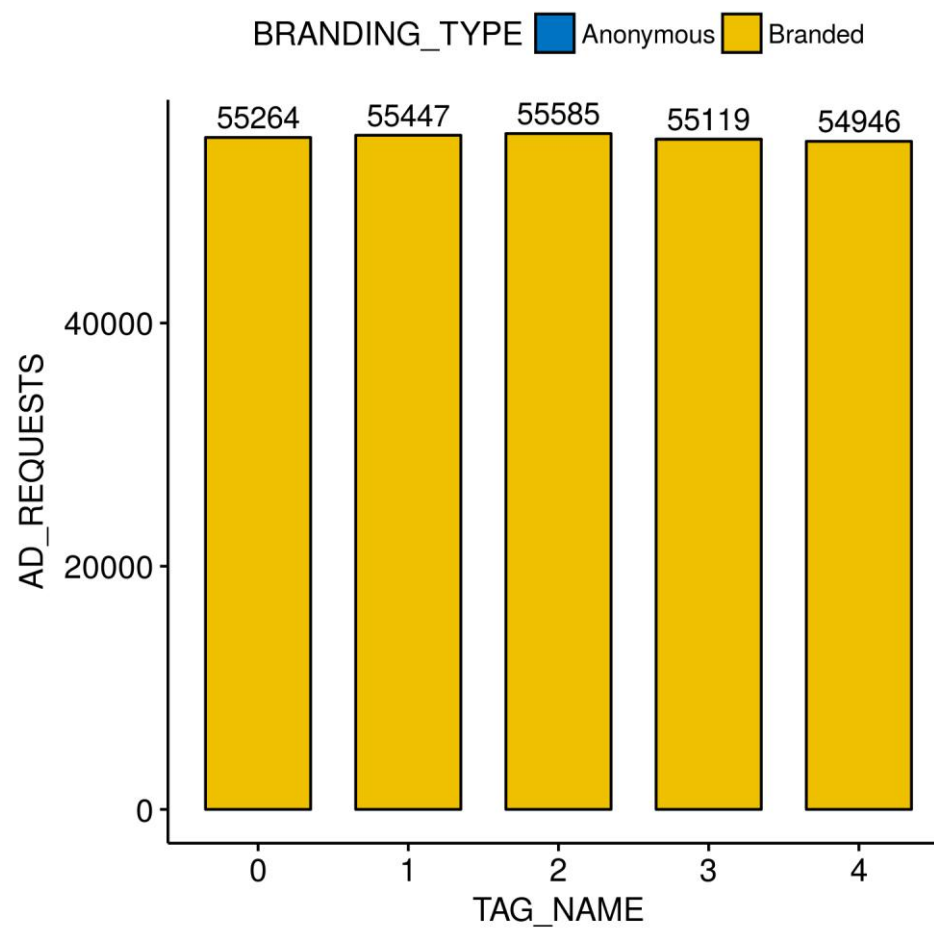




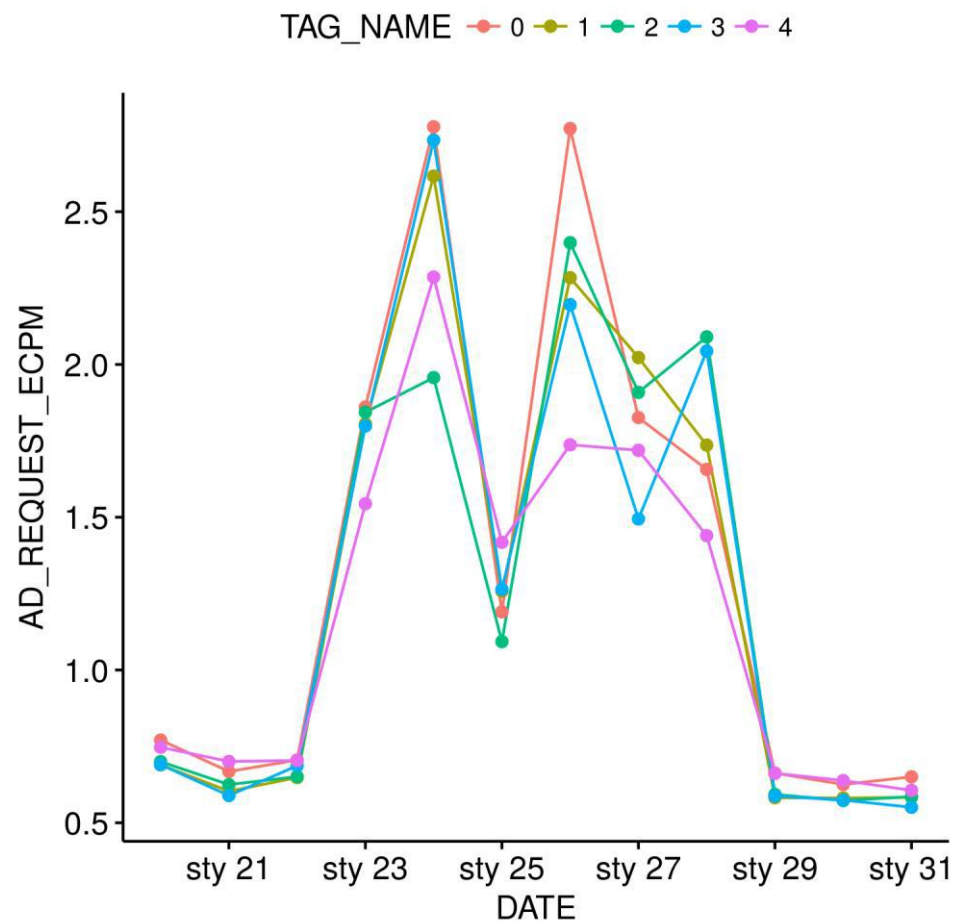
## 11.2 Wykres AdR ECPM z podziałem na Branding Type



### 11.3 Wykres Ad requests z podziałem na Branding Type



### 11.4 Wykres trendu dziennego AdRequest eCPM



### 11.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-20	0.689589	0.770665	0.081076	0.111104370

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-21	0.589009	0.700668	0.111659	0.17315809
2019-01-22	0.647907	0.705318	0.057411	0.08485063
2019-01-23	1.544144	1.860431	0.316287	0.18580116
2019-01-24	1.956785	2.777854	0.821069	0.34683489
2019-01-25	1.092777	1.418092	0.325315	0.25912543
2019-01-26	1.737170	2.772233	1.035063	0.45906875
2019-01-27	1.494420	2.022688	0.528268	0.30039908
2019-01-28	1.439722	2.090173	0.650451	0.36853844
2019-01-29	0.582044	0.662521	0.080477	0.12932551
2019-01-30	0.572604	0.638162	0.065558	0.10829178
2019-01-31	0.550658	0.650406	0.099748	0.16609939

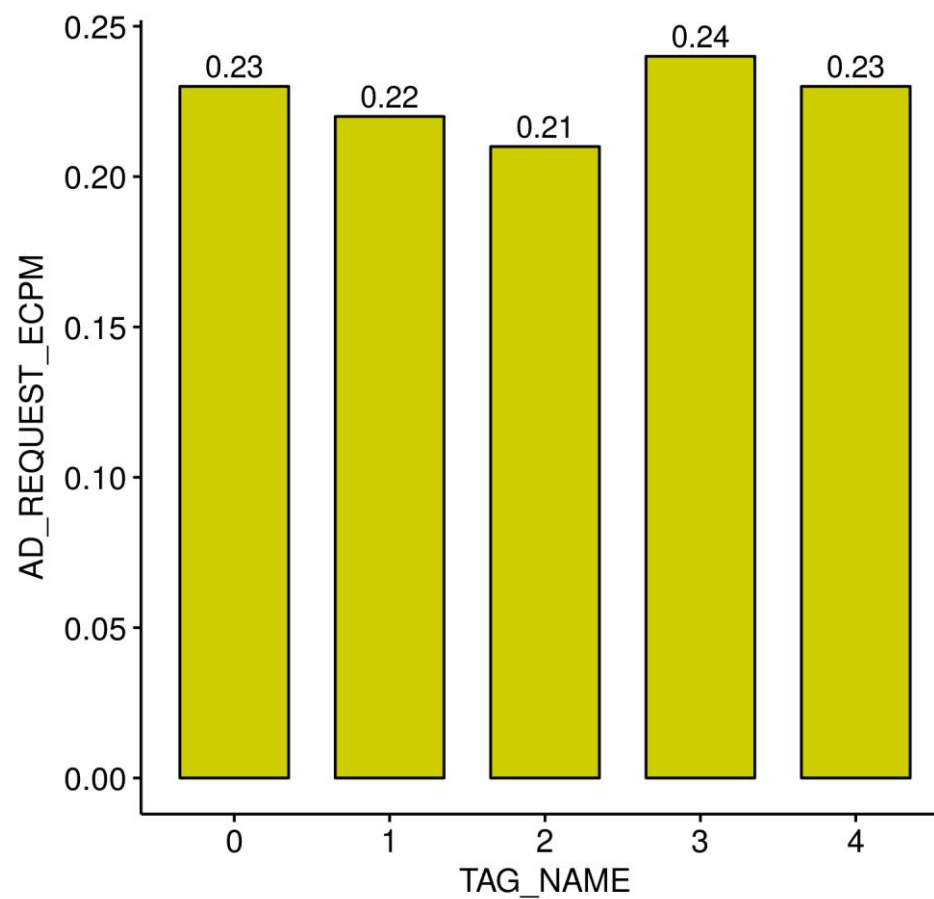
## 12 Eksperyment 93\_test\_300x600\_T\_ (od 2019-1-12 do 2019-1-16)

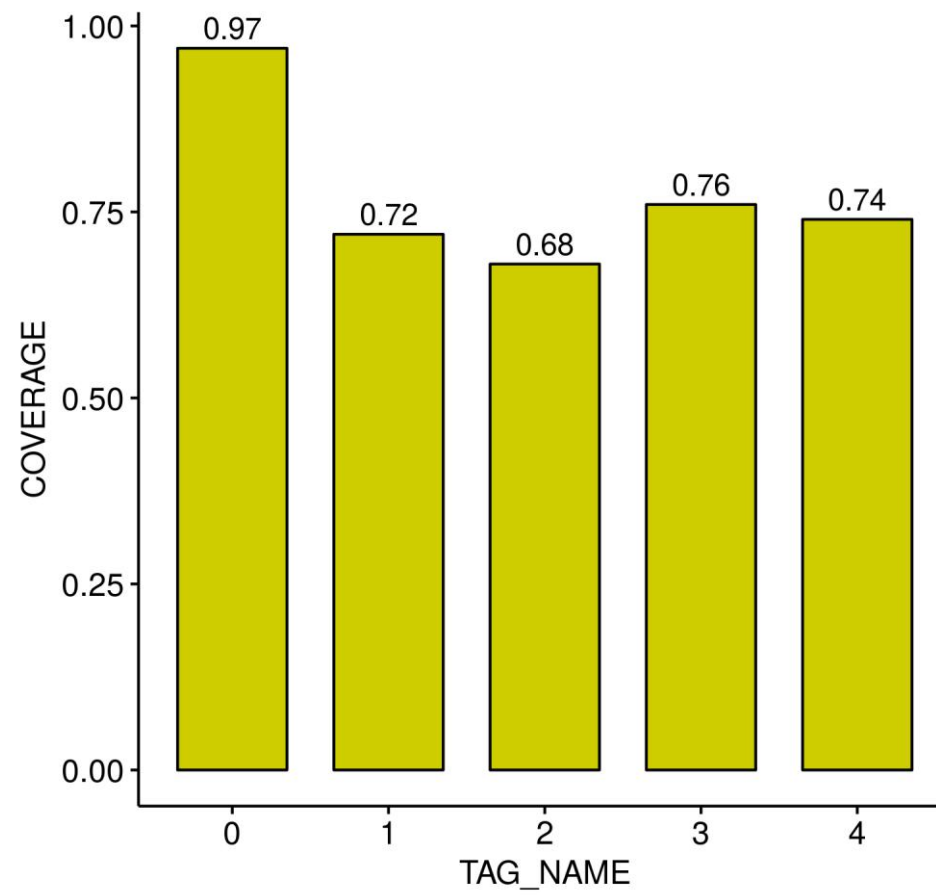
Data wygenerowania raportu: 2019-01-17

Cel: na paczce 93\_test\_300x600\_T zerówka była sprzedawana wczoraj po 0.24, dlatego (pomimo i tak niskich reguł) tutaj obniżamy reguły plus wykonamy eksperyment porównania licytacji branded i anonymous. Celem eksperymentu jest zbadanie, czy ustawienie reguły wykluczającej anonymous jest takie samo jak gdyby ustawić ten sam floor co dla branded (prawdopodobnie nie). Dlatego prosba o ustawienie reguł ze standardową rotacją (20/50/10/10/10): \_0: A = B = 0 (to już jest); \_1 i \_2: YR - Wasza optymalizacja; \_3: A = B = 0.10; \_4: B = 0.10; A zablokowane.

Wnioski: Cena Branded ARe na \_3 i \_4 podobne, ze względu na blokadę A na \_4 obniżone Coverage i w konsekwencji obniżone ARe całkowite na \_4 (w porównaniu do \_3)

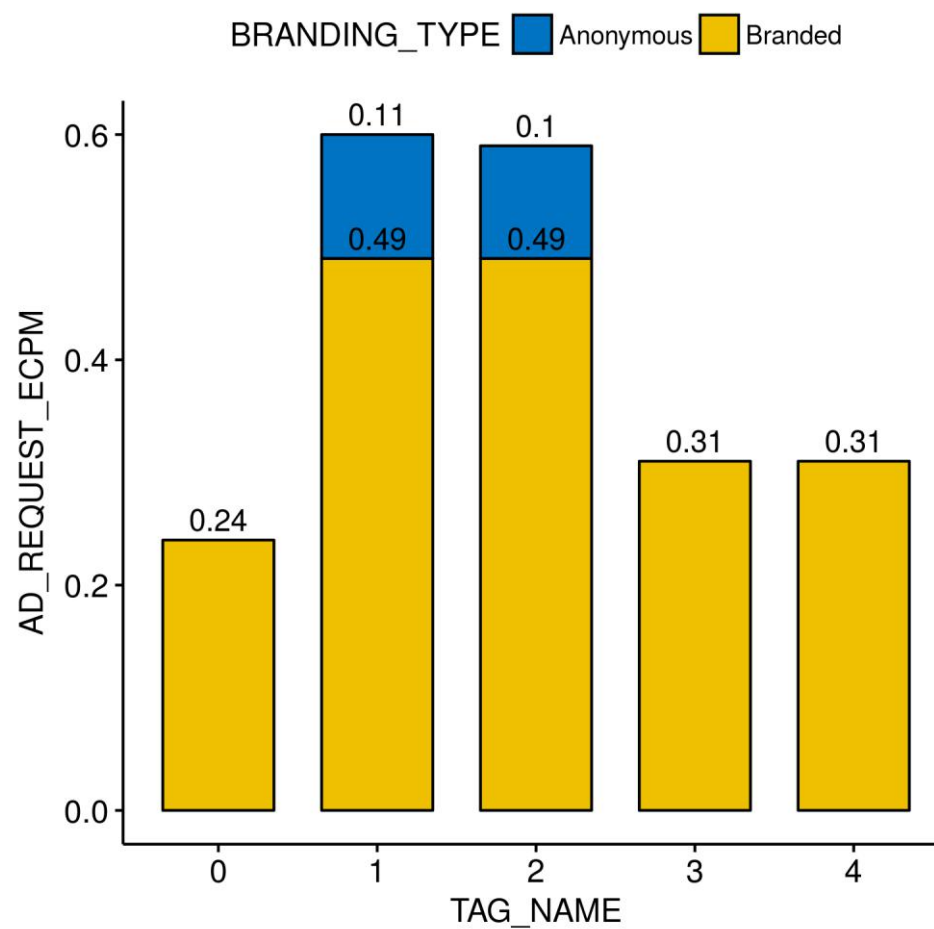
### 12.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



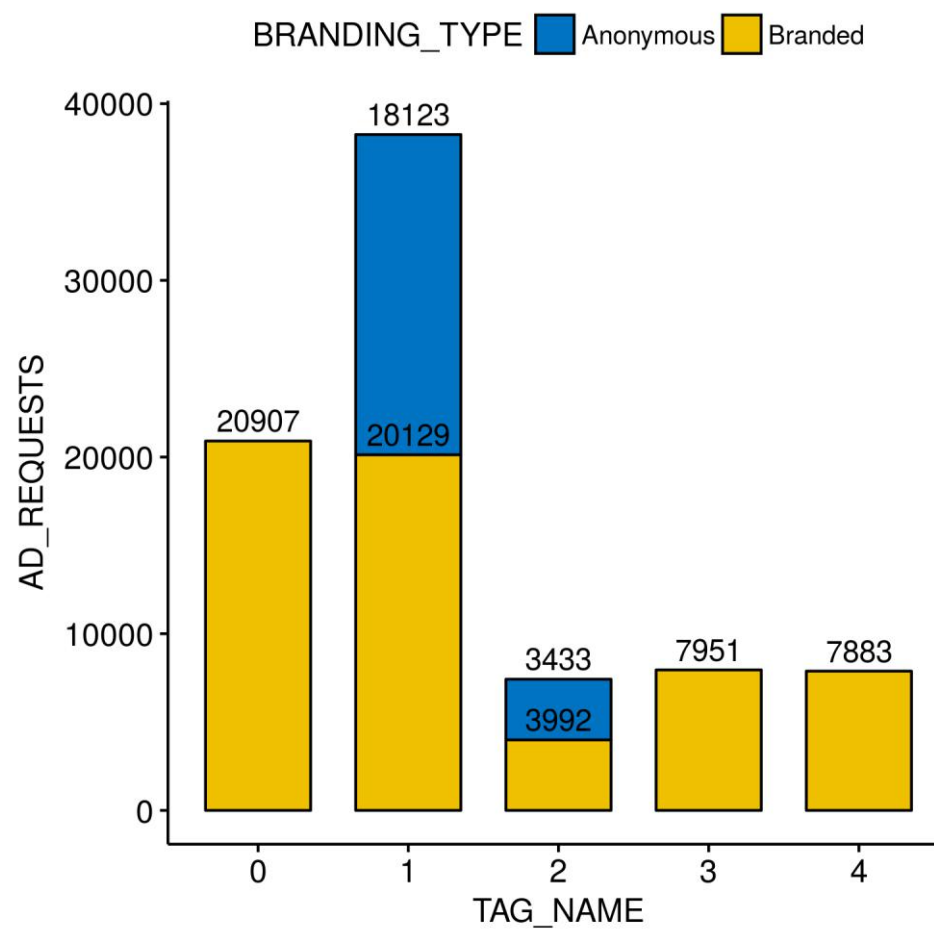




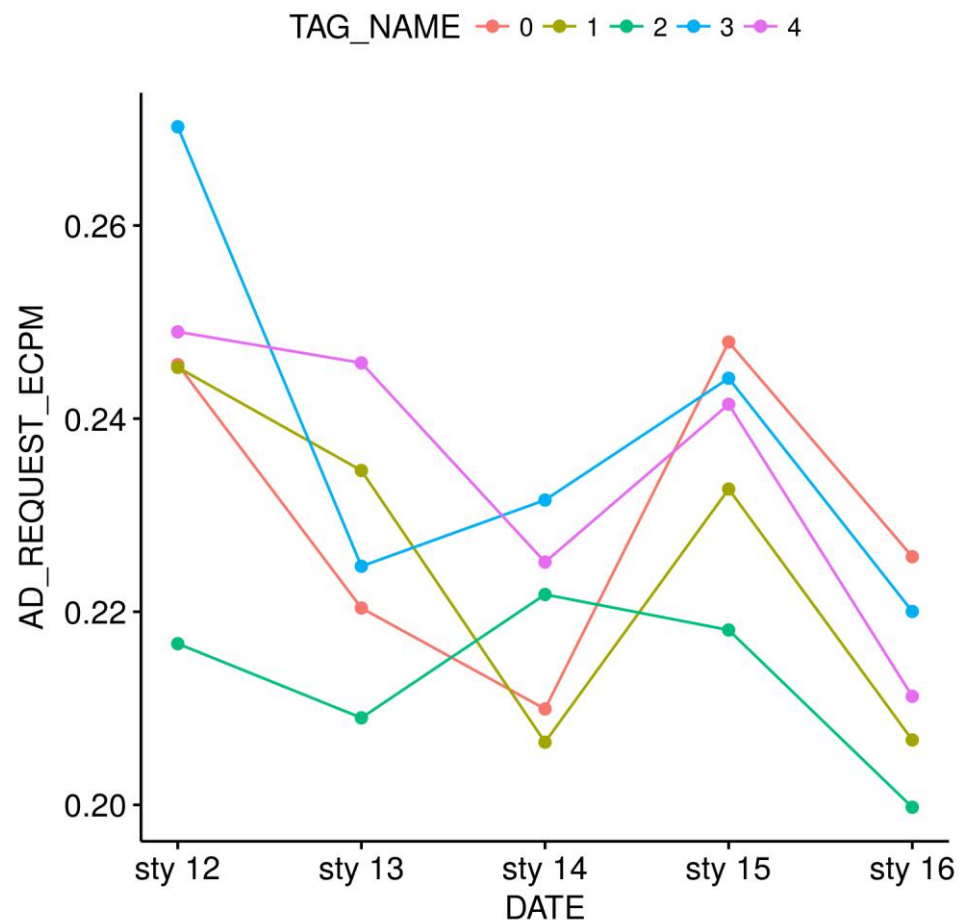
## 12.2 Wykres AdR ECPM z podziałem na Branding Type



### 12.3 Wykres Ad requests z podziałem na Branding Type



## 12.4 Wykres trendu dziennego AdRequest eCPM



## 12.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-12	0.216690	0.270222	0.053532	0.2198837

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-13	0.209010	0.245769	0.036759	0.1616565
2019-01-14	0.206492	0.231565	0.025073	0.1144737
2019-01-15	0.218108	0.247931	0.029823	0.1279850
2019-01-16	0.199748	0.225698	0.025950	0.1219896

### 13 Eksperyment 93\_test\_300x600\_T\_ (od 2019-1-17 do 2019-1-31)

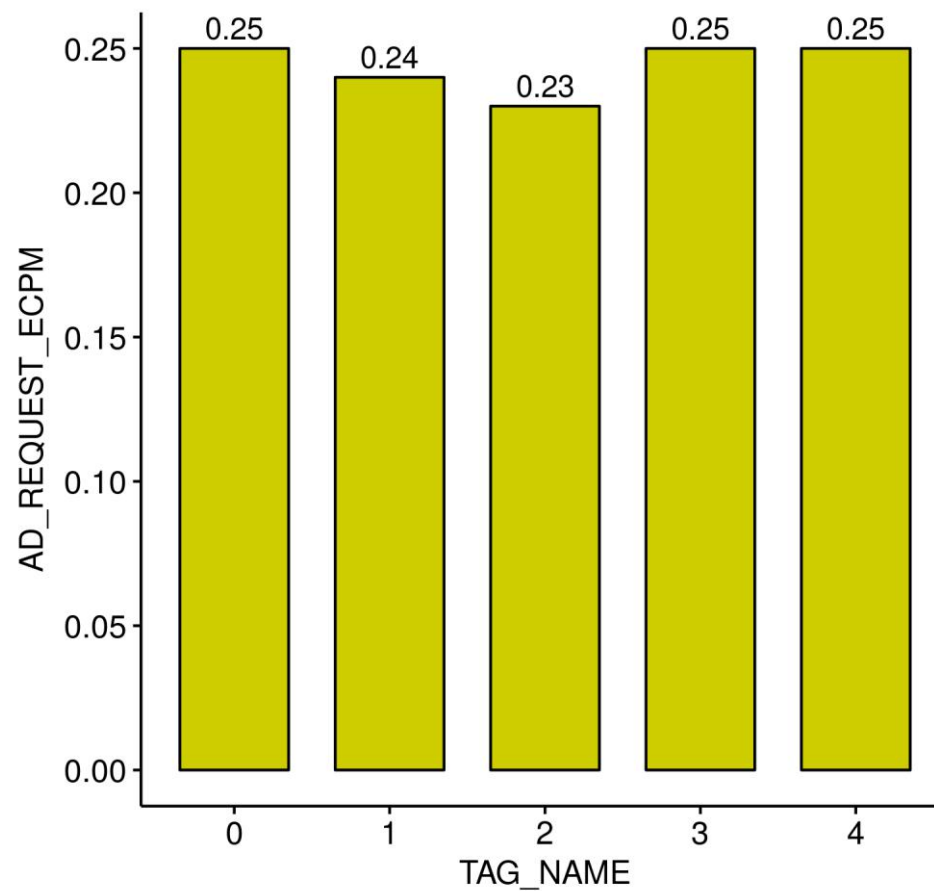
Data wygenerowania raportu: 2019-02-01

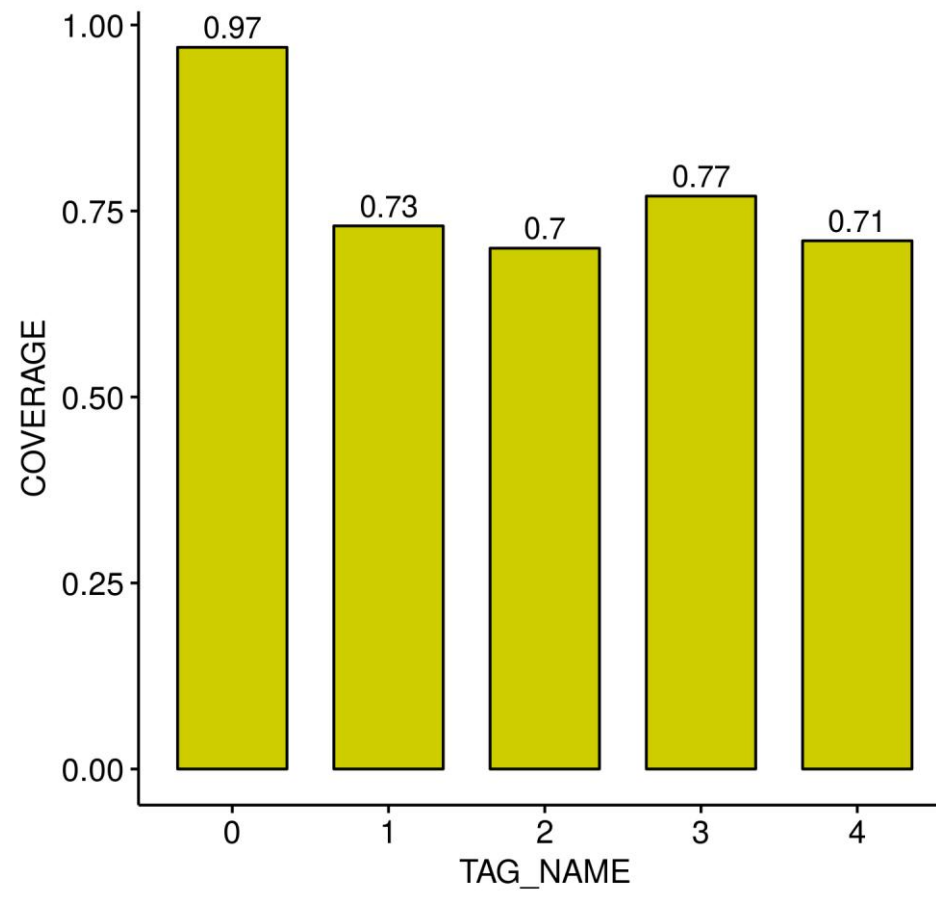
data	tag_ad_unit	rule_name	wyjsciove_branded	wyjsciove_anonymous
2019-01-10	93_test_300x600_T_0	93_test_300x600_T_0	0	0
2019-01-11	93_test_300x600_T_3	93_test_300x600_T_3	0.1	0.1
2019-01-18	93_test_300x600_T_4	93_test_300x600_T_4	0.15	0.1

Cel eksperymentu: Panowie na paczce 93\_test\_300x600\_T zerówka była sprzedawana wczoraj po 0.24, dlatego (pomimo i tak niskich reguł) tutaj obniżamy reguły plus wykonamy eksperyment porównania licytacji branded i anonymous. Celem eksperymentu jest zbadanie, czy ustawienie reguły wykluczającej anonymous jest takie samo jak gdyby ustawić ten sam floor co dla branded (prawdopodobnie nie). Dlatego prosba o ustawienie reguł ze standardową rotacją (20/50/10/10/10): \_0: A = B = 0 (to już jest); \_1 i \_2: YR - Wasza optymalizacja; \_3: A = B = 0.10; \_4: B = 0.15; A=0.1.

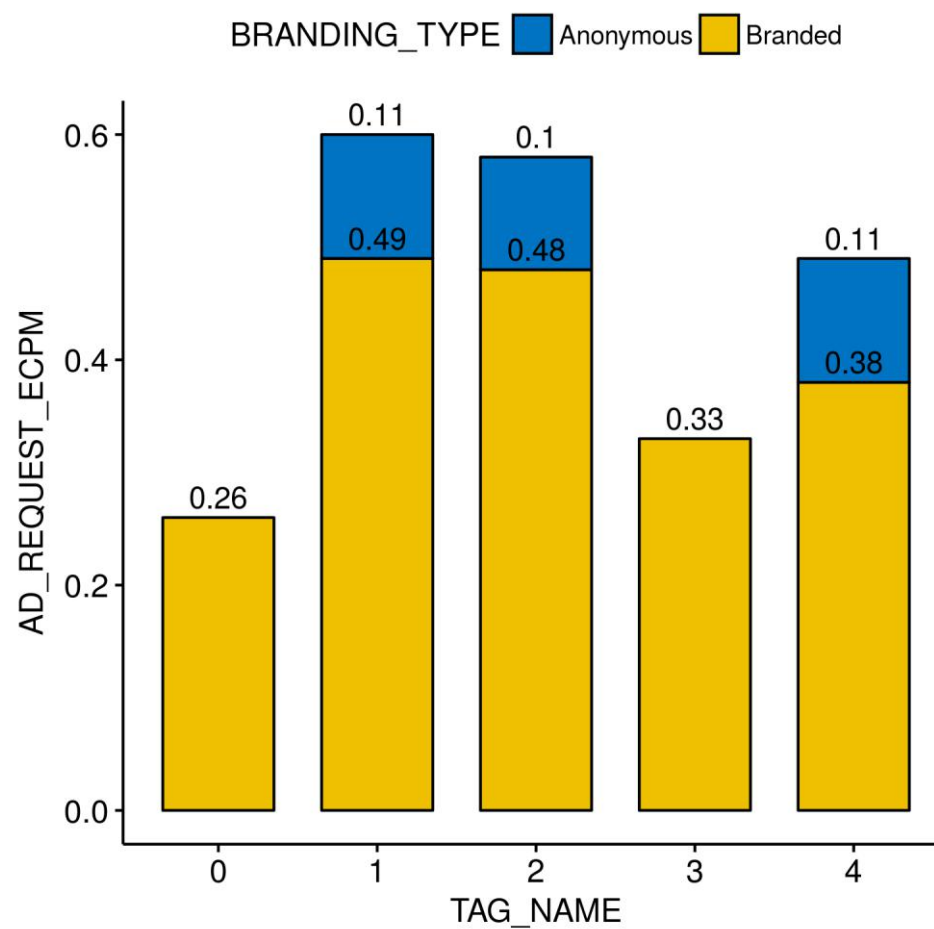
Wnioski: Porównywalne ARe na \_3 i \_4, zwiększone Coverage na \_3. W tym przypadku większą rolę odgrywają licytujący A.

### 13.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



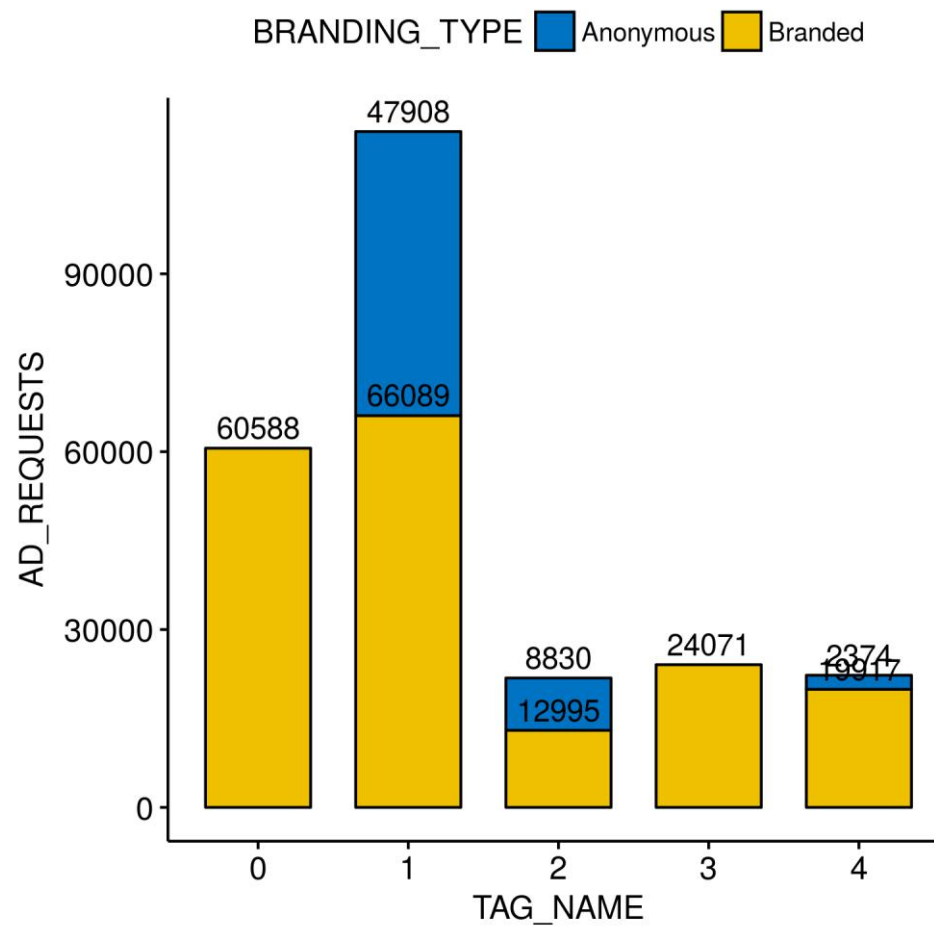


### 13.2 Wykres AdR ECPM z podziałem na Branding Type

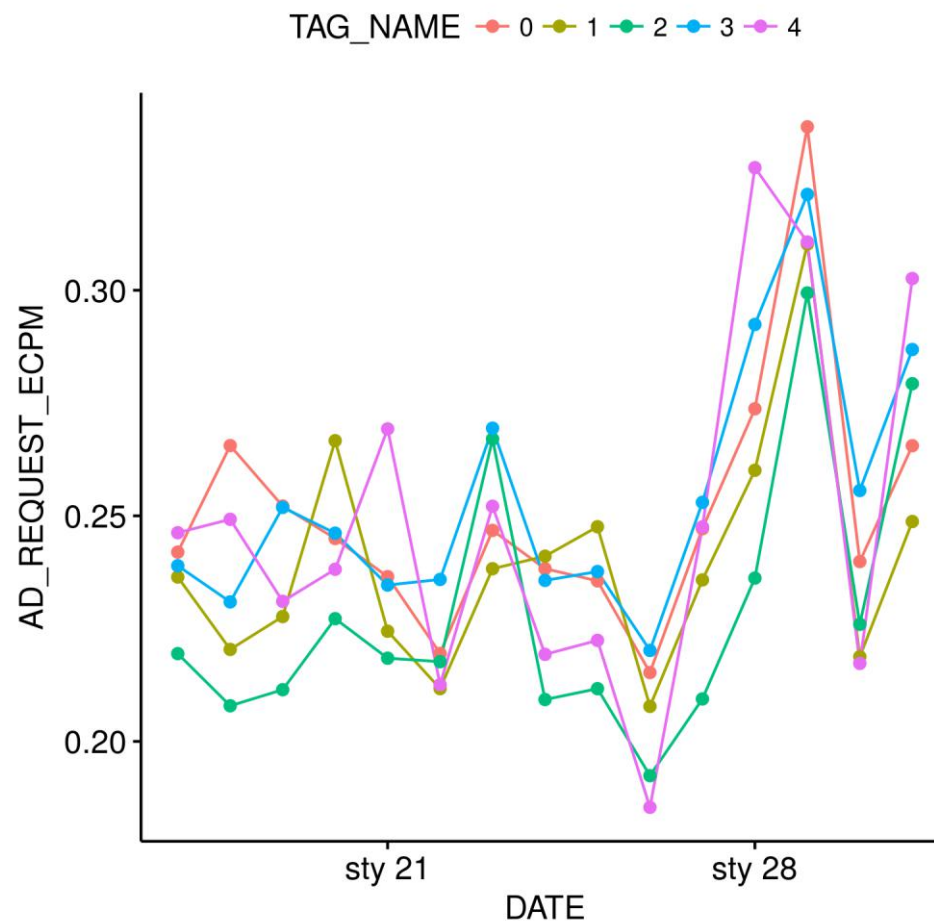




### 13.3 Wykres Ad requests z podziałem na Branding Type



### 13.4 Wykres trendu dziennego AdRequest eCPM



### 13.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-17	0.219457	0.246254	0.026797	0.1150800

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-18	0.207900	0.265565	0.057665	0.2435872
2019-01-19	0.211462	0.252153	0.040691	0.1755379
2019-01-20	0.227205	0.266651	0.039446	0.1597470
2019-01-21	0.218443	0.269268	0.050825	0.2084226
2019-01-22	0.211708	0.235900	0.024192	0.1080946
2019-01-23	0.238278	0.269425	0.031147	0.1226977
2019-01-24	0.209279	0.241047	0.031768	0.1410889
2019-01-25	0.211692	0.247580	0.035888	0.1562821
2019-01-26	0.185401	0.220170	0.034769	0.1714570
2019-01-27	0.209419	0.252988	0.043569	0.1884444
2019-01-28	0.236197	0.327170	0.090973	0.3229618
2019-01-29	0.299397	0.336226	0.036829	0.1158832
2019-01-30	0.217291	0.255613	0.038322	0.1620709
2019-01-31	0.248737	0.302604	0.053867	0.1954036

## 14 Eksperyment ch\_test\_160x600\_left\_T\_ (od 2019-1-12 do 2019-1-16)

Data wygenerowania raportu: 2019-01-17

Cel: Panowie ten tag po ustawieniu reguł (A:0.12 B:0.40) jest tylko trochę drożej sprzedający się niż \_0, dlatego poszukamy optymalnego poziomu ustawiając kolejne (co ustaloną kwotę) floory. Dlatego prosba o ustawienie na paczce ch\_test\_160x600\_left\_T\_ następujących reguł ze standardową rotacją (2/5/1/1/1): \_0: A = B = 0; \_1: YR - Wasza optymalizacja; \_2: A = 0.05; B = 0.1; \_3: A = 0.1; B = 0.2; \_4: A = 0.15; B = 0.3

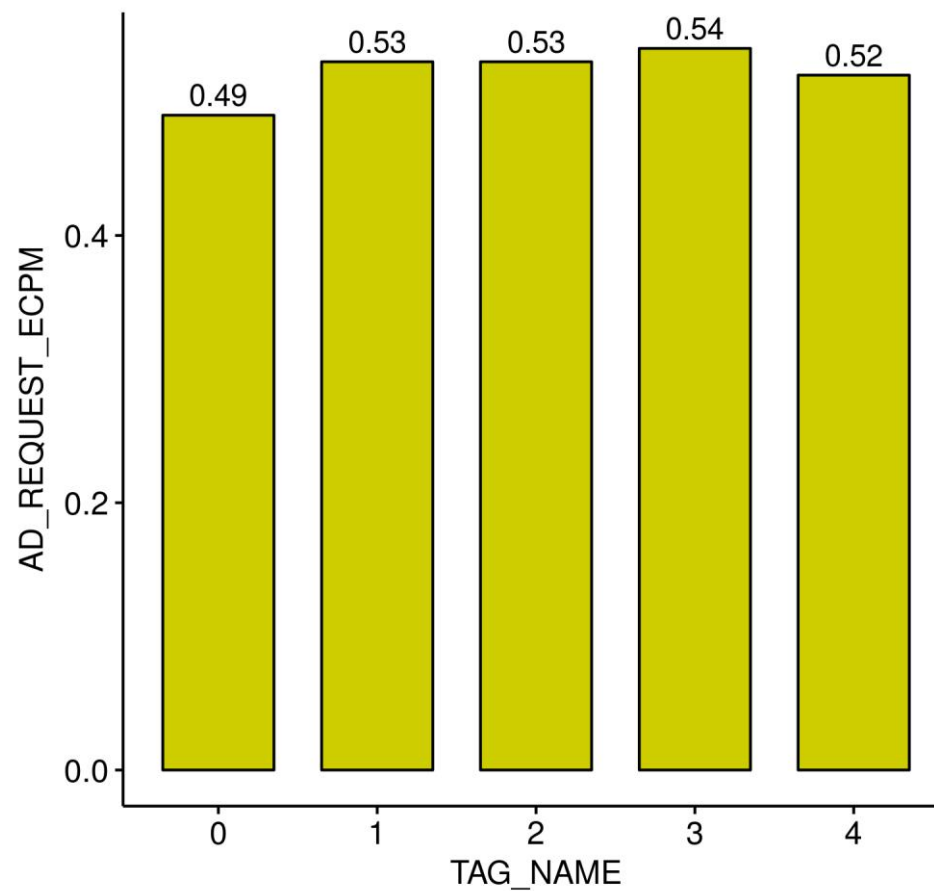
Celem eksperymentu jest poszukiwanie optymalnego fl A i fl\_B.

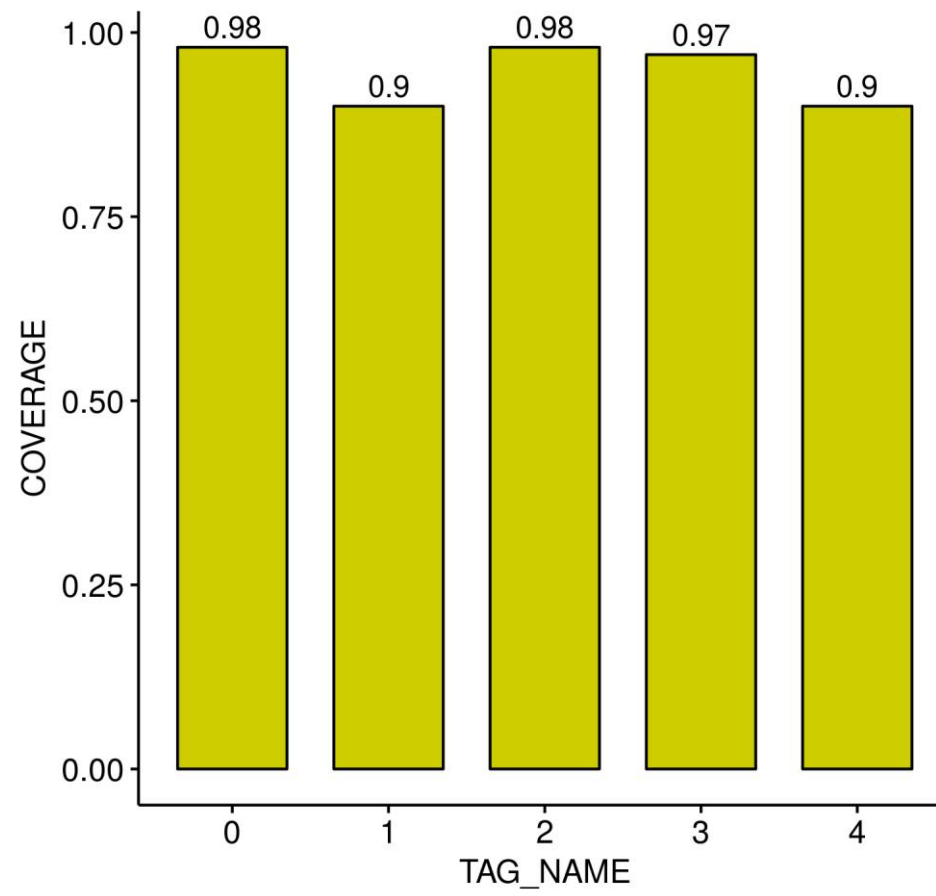
Wnioski: Im wyższe floor tym mniejsze Coverage. Optymalne fl na \_3 – przez wszystkie dni poza jednym eksperymentalnym ustawione A = 0.1 i B = 0.2 dawało w rezultacie najwyższe ARe.

data	tag_ad_unit	rule_name	wejsciowe_brannded	wejsciowe_anonymous	wyjsciowe_brannded	wyjsciowe_anonymous
2019-01-10	ch_test_160x600_left_T_0	ch_test_160x600_left_T_0	0.4	0.12	0	0
2019-01-11	ch_test_160x600_left_T_2	ch_test_160x600_left_T_2	0.4	0.12	0.1	0.05
2019-01-11	ch_test_160x600_left_T_3	ch_test_160x600_left_T_3	0.4	0.12	0.2	0.1
2019-01-11	ch_test_160x600_left_T_4	ch_test_160x600_left_T_4	0.4	0.12	0.3	0.15

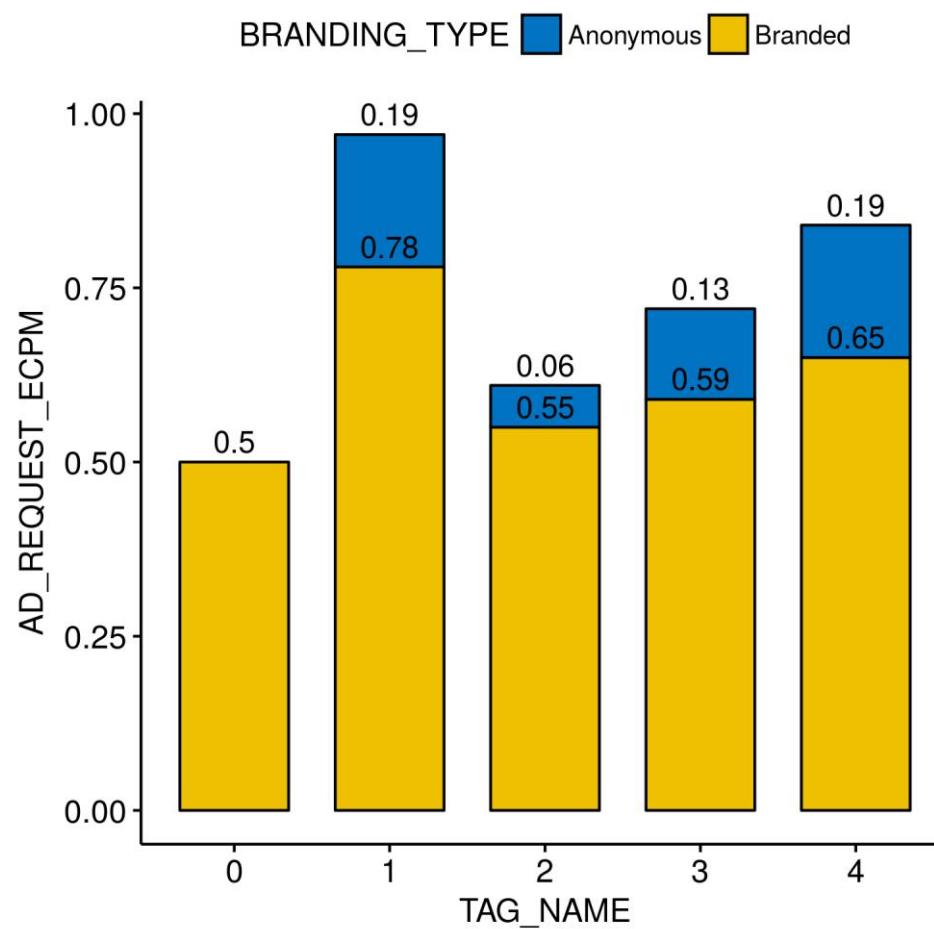
<b>data</b>	<b>tag_ad_unit</b>	<b>rule_name</b>	<b>wejsciowe_branded</b>	<b>wejsciowe_anonymous</b>	<b>wyjsciove_branded</b>	<b>wyjsciove_anonymous</b>
01-11						

### 14.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type



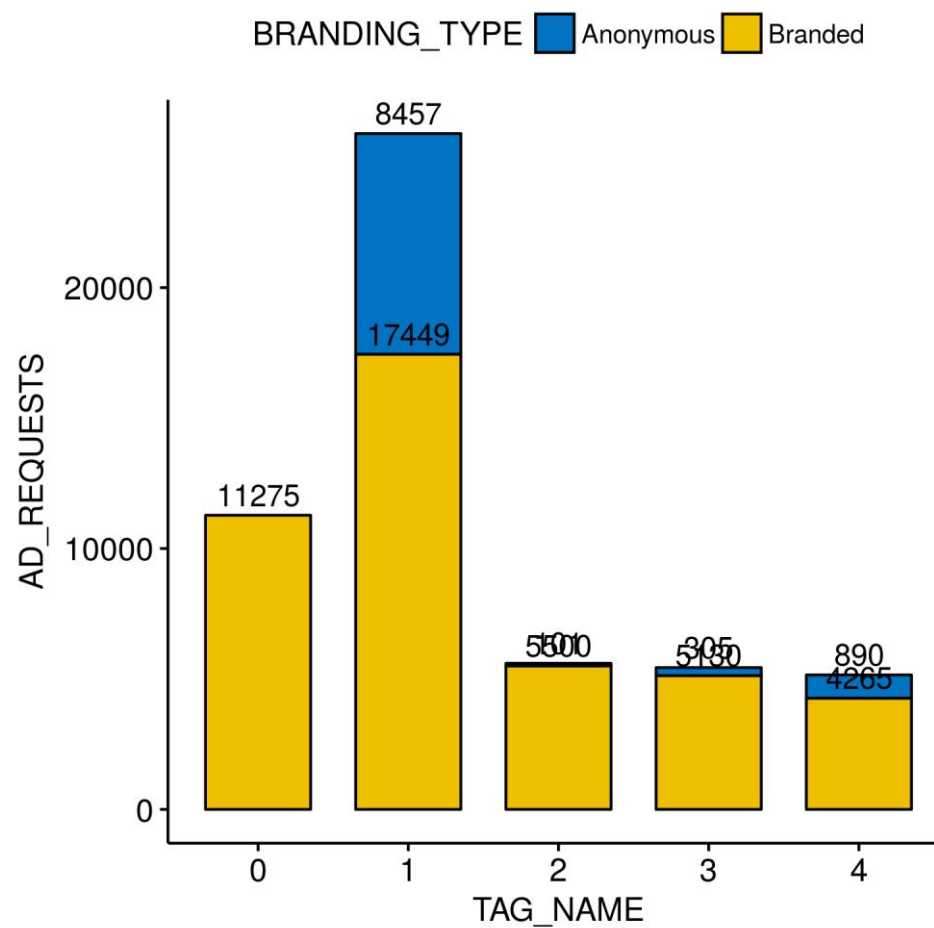


## 14.2 Wykres AdR ECPM z podziałem na Branding Type

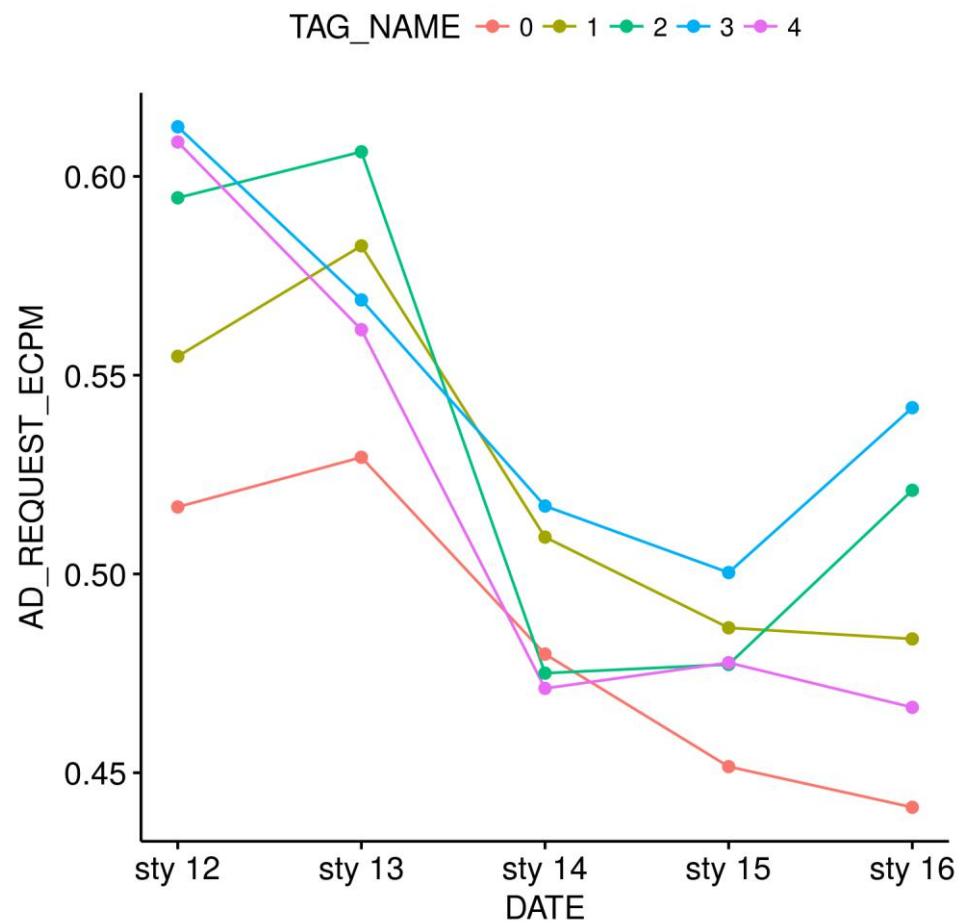




### 14.3 Wykres Ad requests z podziałem na Branding Type



#### 14.4 Wykres trendu dziennego AdRequest eCPM



#### 14.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-12	0.516851	0.612476	0.095625	0.16934865

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-13	0.529352	0.606197	0.076845	0.13534423
2019-01-14	0.471222	0.517086	0.045864	0.09281317
2019-01-15	0.451537	0.500353	0.048816	0.10256647
2019-01-16	0.441313	0.541812	0.100499	0.20444806

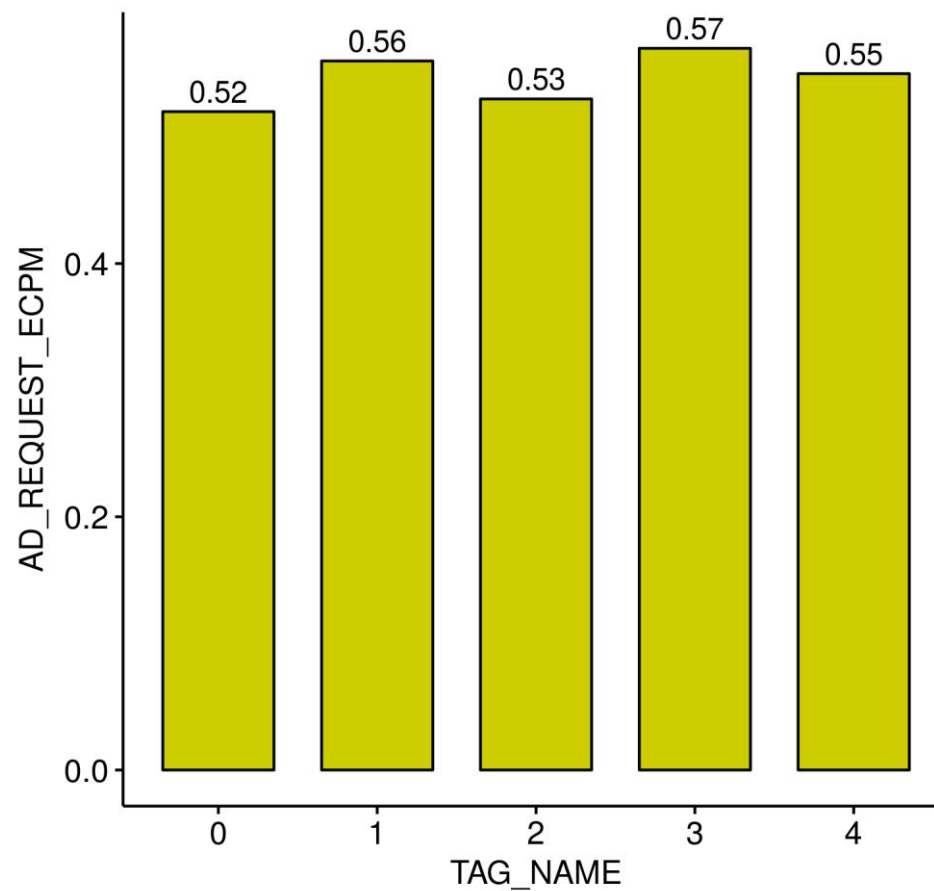
## 15 Eksperyment ch\_test\_160x600\_left\_T\_ (od 2019-1-20 do 2019-1-31)

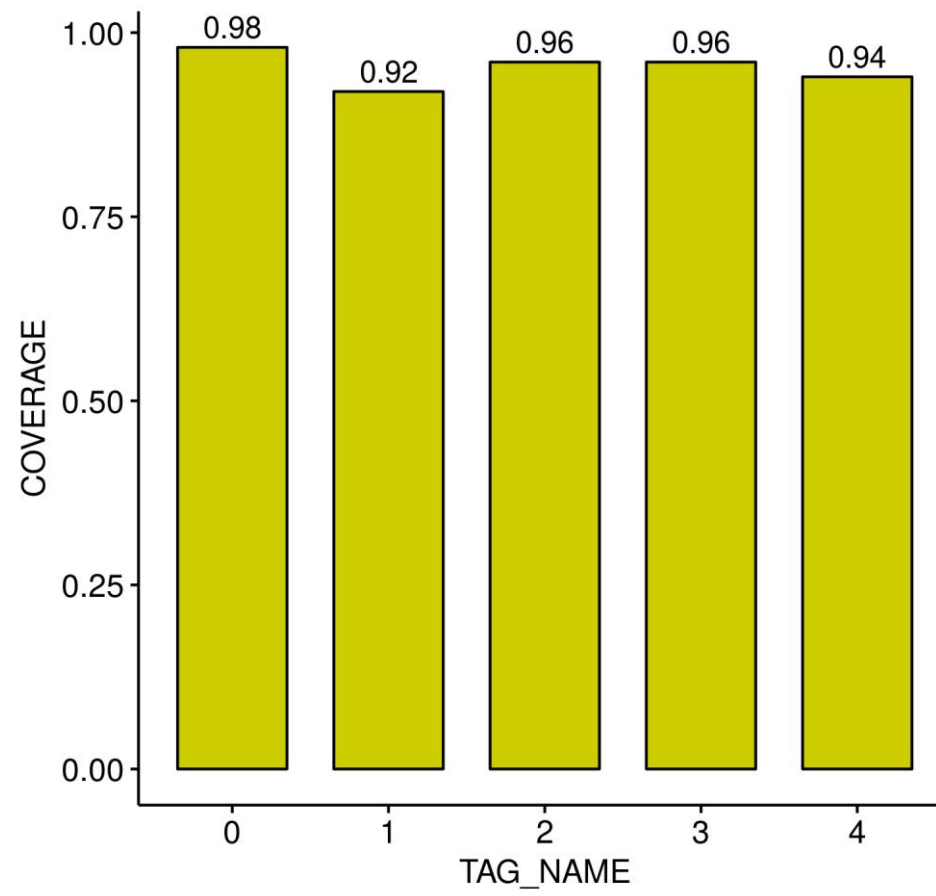
Data wygenerowania raportu: 2019-02-01

data	tag_ad_unit	wyjsciowe_branded	wyjsciowe_anonymous
2019-01-10	ch_test_160x600_left_T_0	0	0
2019-01-18	ch_test_160x600_left_T_2	0.12	0.08
2019-01-18	ch_test_160x600_left_T_3	0.15	0.1
2019-01-18	ch_test_160x600_left_T_4	0.18	0.12

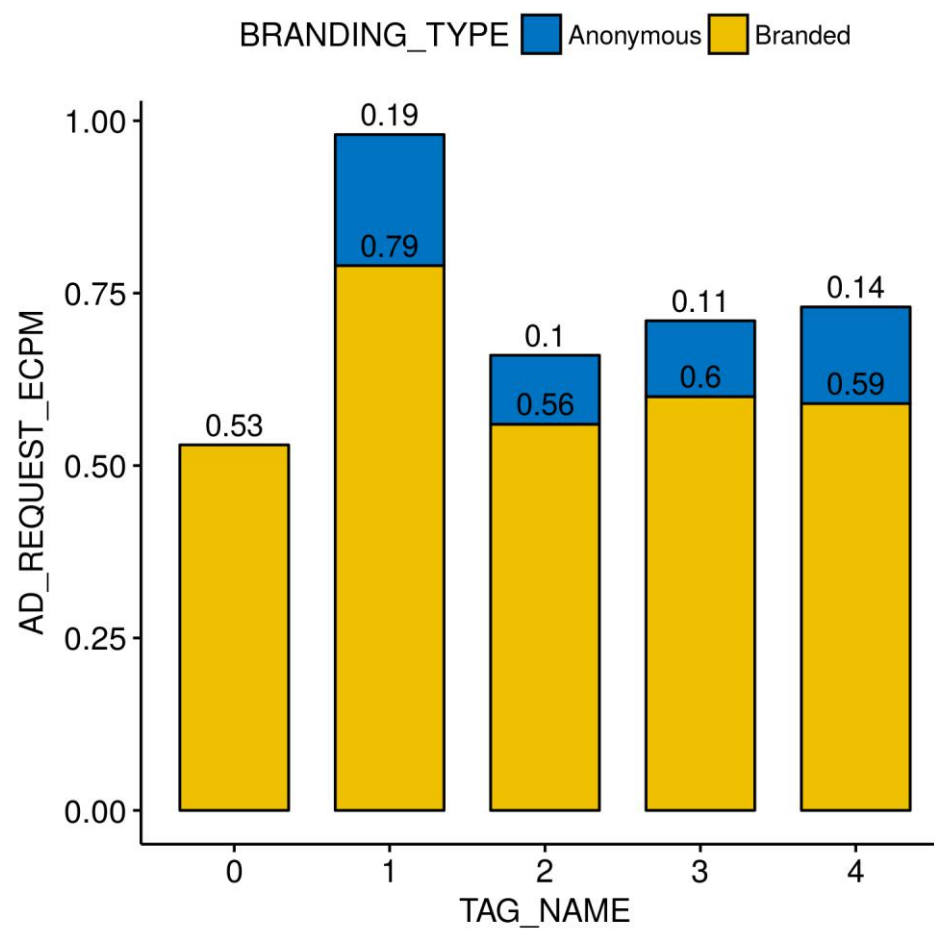
**Cele:** Panowie ten tag po ustawieniu reguł (A:0.12 B:0.40) jest tylko trochę drożej sprzedający się niż \_0, dlatego poszukamy optymalnego poziomu ustawiając kolejne (co ustaloną kwotę) floory. Dlatego prosba o ustawienie na paczce ch\_test\_160x600\_left\_T\_ następujących reguł ze standardową rotacją (2/5/1/1/1): \_0: A = B = 0; \_1: YR - Wasza optymalizacja; \_2: A = 0.08; B = 0.12 \_3: A = 0.1; B = 0.15 \_4: A = 0.12; B = 0.18 **Wnioski:** Optymalny poziom na \_3, najlepsze ARE w totalu i przez większość dni eksperymentu.

### 15.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

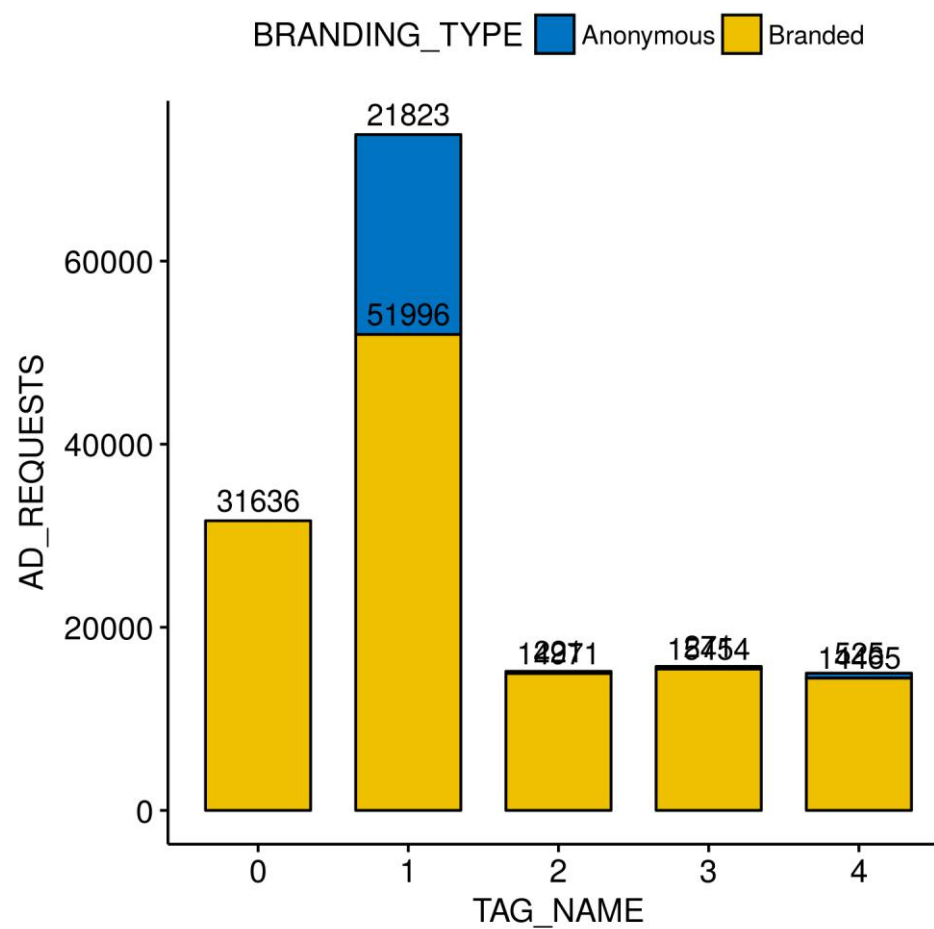




## 15.2 Wykres AdR ECPM z podziałem na Branding Type

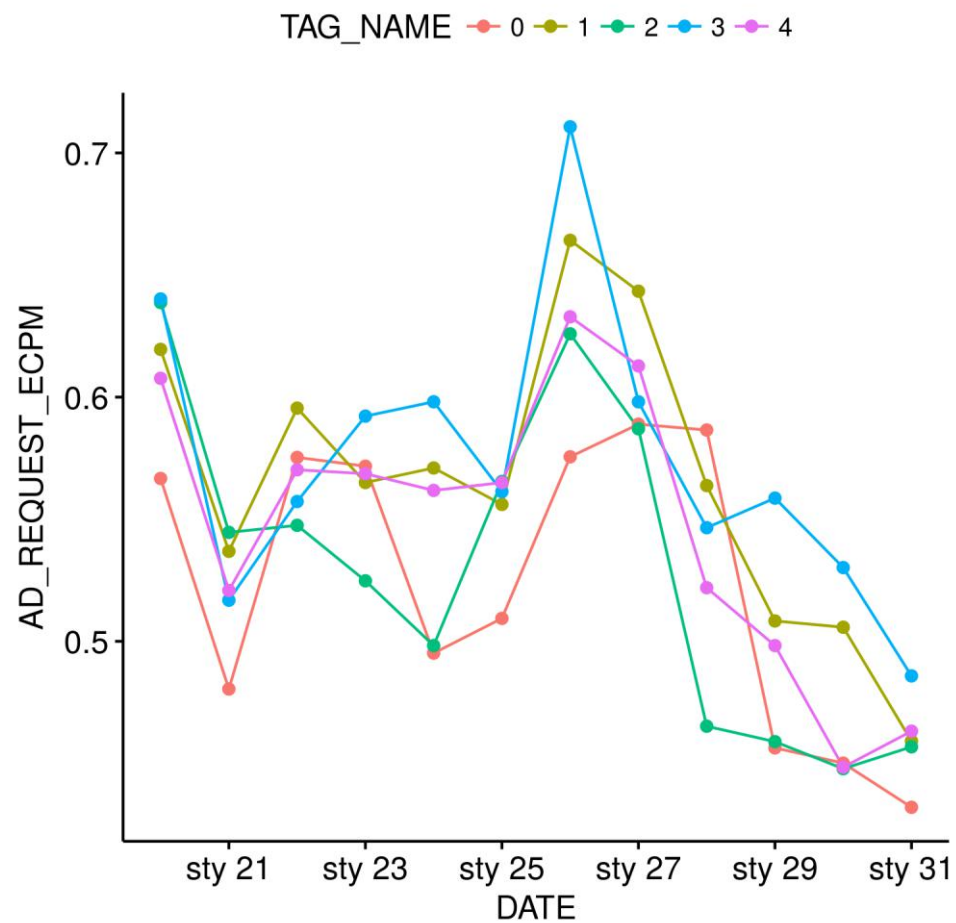


### 15.3 Wykres Ad requests z podziałem na Branding Type





### 15.4 Wykres trendu dziennego AdRequest eCPM



### 15.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-20	0.566686	0.640190	0.073504	0.12180870

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-21	0.480465	0.544621	0.064156	0.12517194
2019-01-22	0.547504	0.595487	0.047983	0.08396042
2019-01-23	0.524770	0.592218	0.067448	0.12076764
2019-01-24	0.495163	0.598094	0.102931	0.18830156
2019-01-25	0.509385	0.565452	0.056067	0.10432652
2019-01-26	0.575556	0.710703	0.135147	0.21013964
2019-01-27	0.587051	0.643375	0.056324	0.09155203
2019-01-28	0.465226	0.586528	0.121302	0.23066611
2019-01-29	0.456352	0.558640	0.102288	0.20155430
2019-01-30	0.447826	0.530194	0.082368	0.16843827
2019-01-31	0.432030	0.485839	0.053809	0.11724767

## 16 Eksperyment j5\_test\_970x250\_T\_ (od 2019-1-12 do 2019-1-15)

Data wygenerowania raportu: 2019-01-16

Panowie na paczce tagów j5\_test\_970x250\_T\_ to wydaje się, że nie poszalejemy - zerówka jest sprzedawana za 0.14 z pokryciem 93%. Niezależnie od tego spróbujemy ustalić optymalny poziom a później zoptymalizować to po krajach.

\_0: A = B = 0

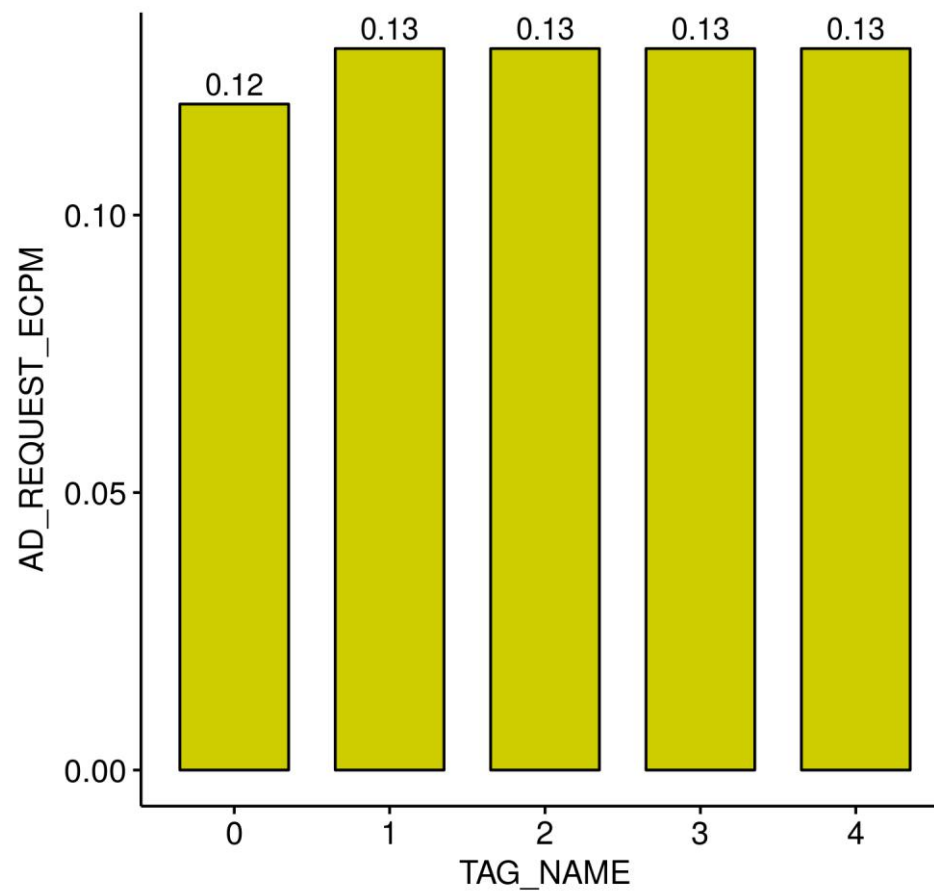
\_1: YR - Wasza optymalizacja

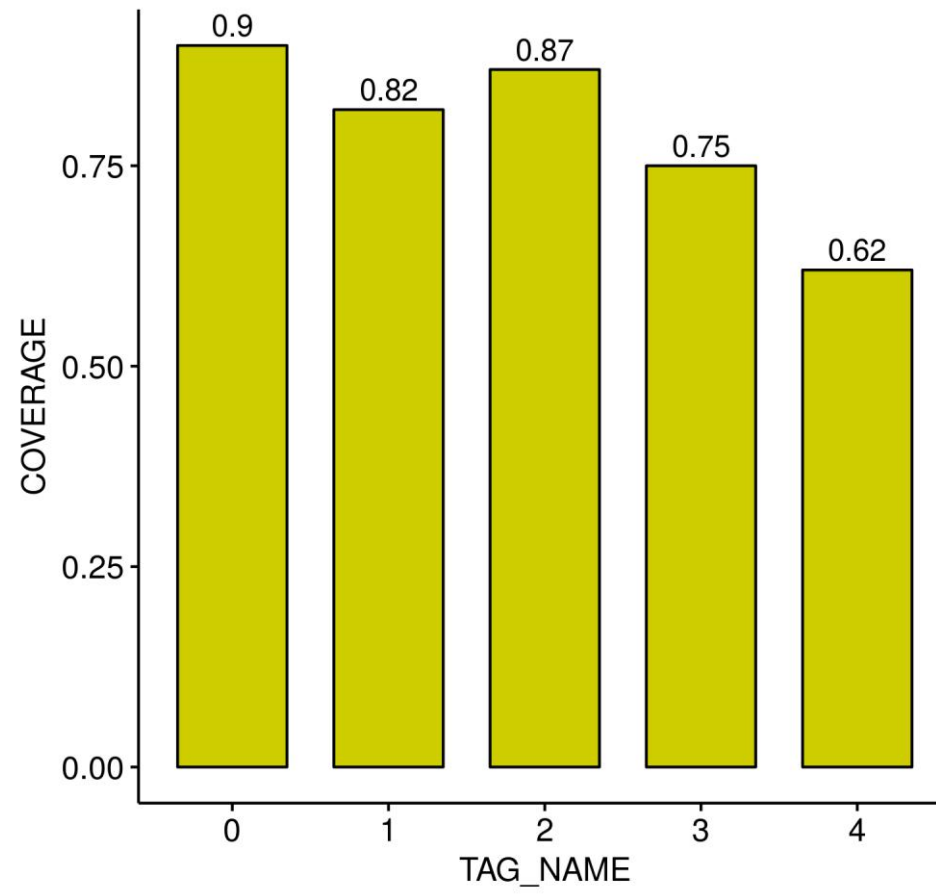
\_2: A = 0.02; B = 0.04

\_3: A = 0.04; B = 0.08

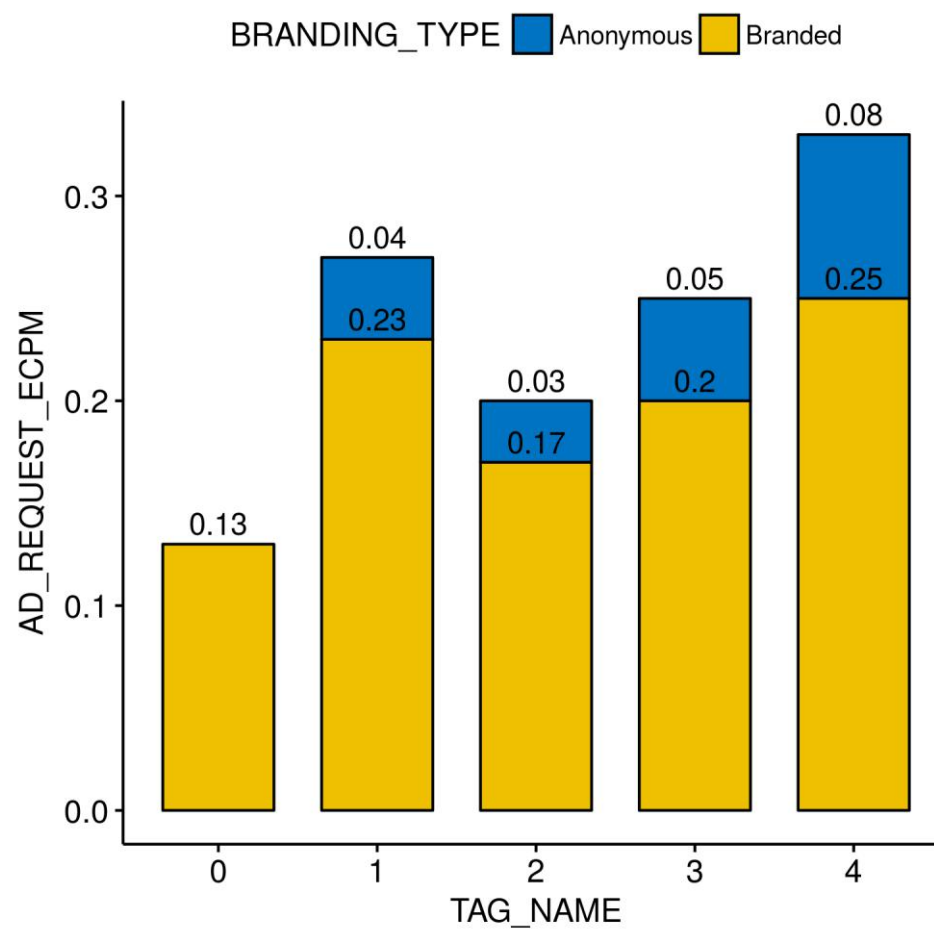
\_4: A = 0.06; B = 0.12

### 16.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

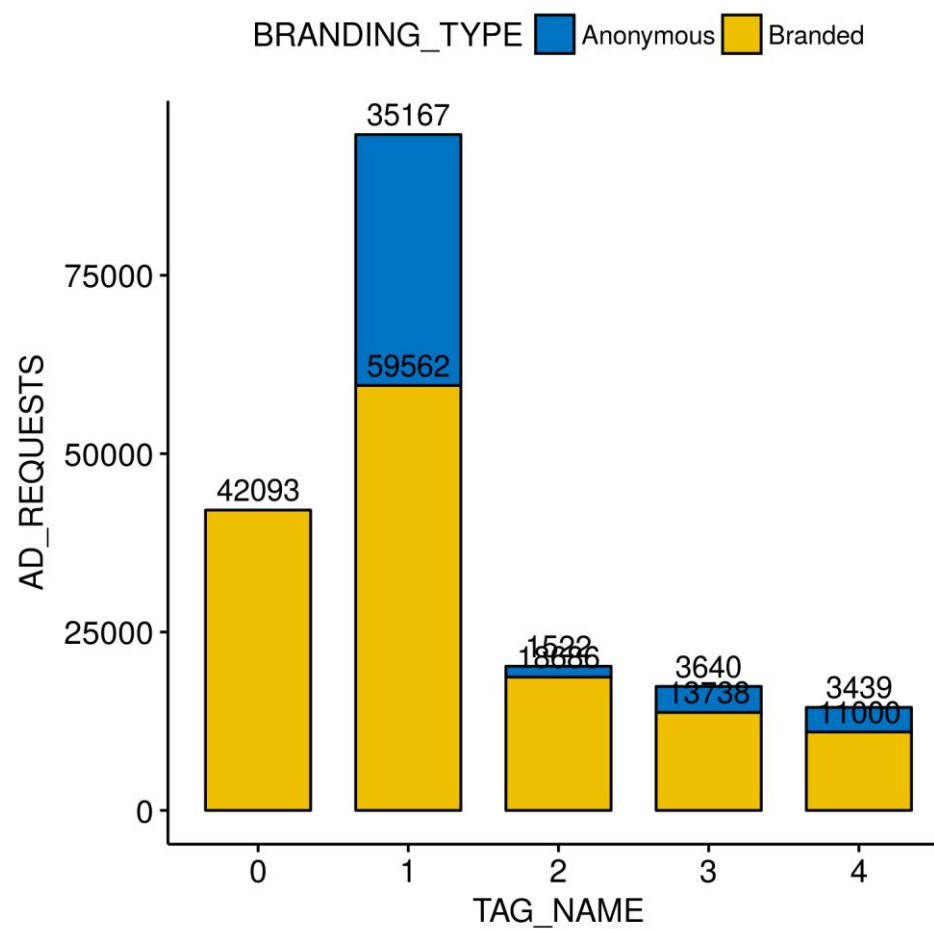




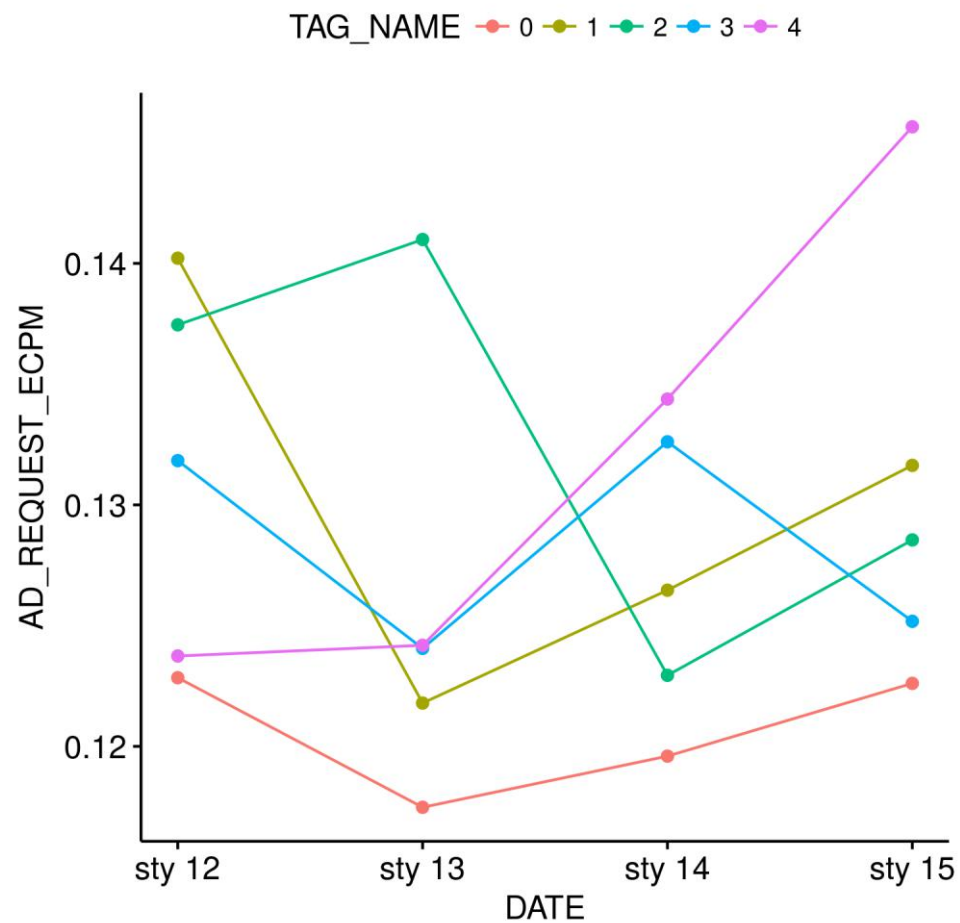
## 16.2 Wykres AdR ECPM z podziałem na Branding Type



### 16.3 Wykres Ad requests z podziałem na Branding Type



### 16.4 Wykres trendu dziennego AdRequest eCPM



### 16.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-12	0.122843	0.140214	0.017371	0.1320702



<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-13	0.117477	0.140989	0.023512	0.1819350
2019-01-14	0.119595	0.134382	0.014787	0.1164436
2019-01-15	0.122609	0.145656	0.023047	0.1718226

## 17 Eksperyment ku\_test\_300x250\_mobile\_1\_T\_ (od 2019-1-20 do 2019-1-31)

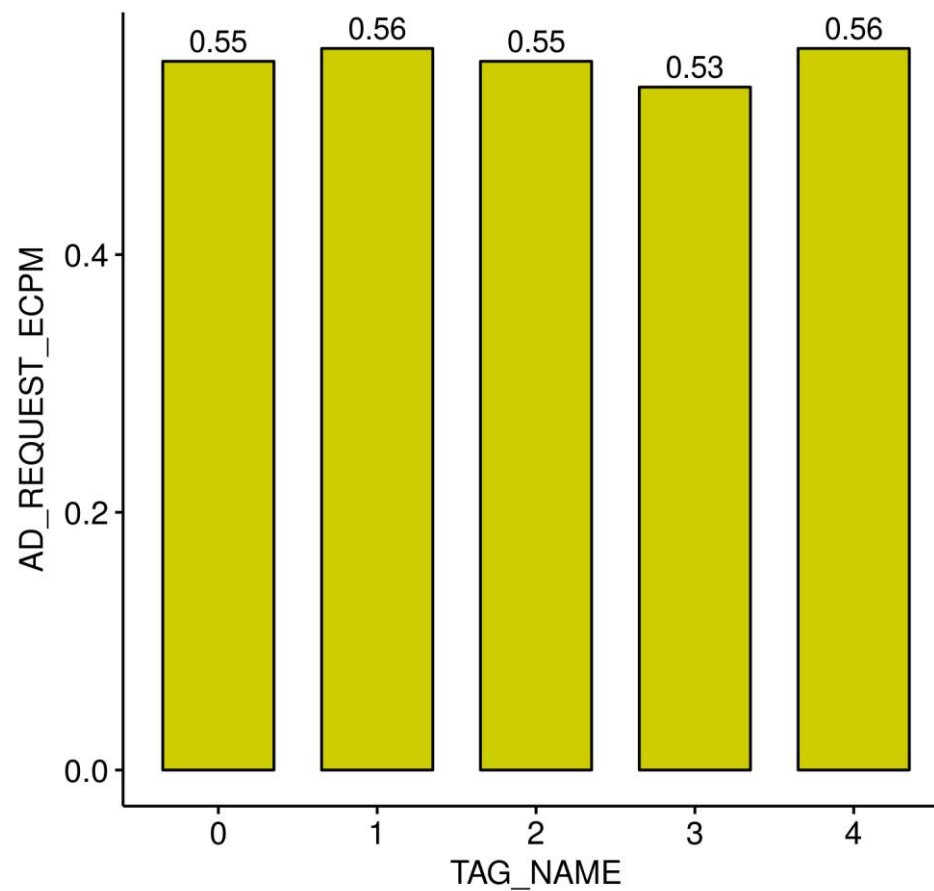
Data wygenerowania raportu: 2019-02-01

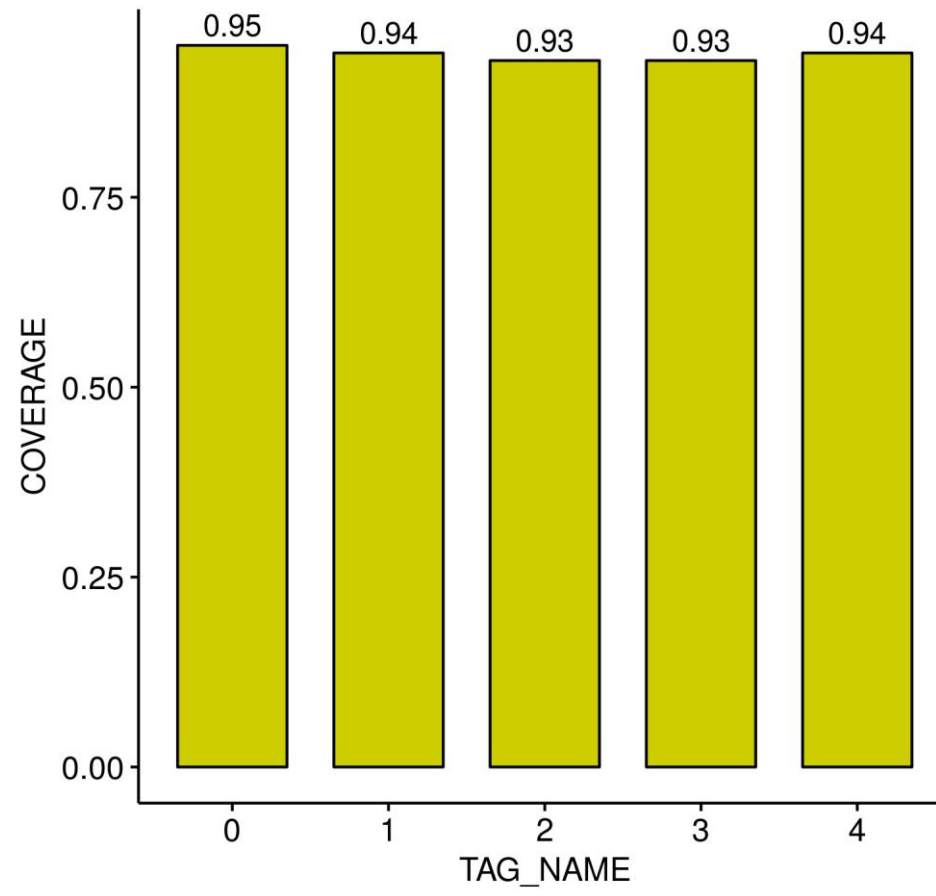
Cel: Eksperyment rotacyjny przed właściwą optymalizacją. Na wszystkich tagach (poza zerówką) ustawione reguły: A: 0,02 zł oraz B: 0,13 zł.

Wnioski: Wahania dzienne do 13%.

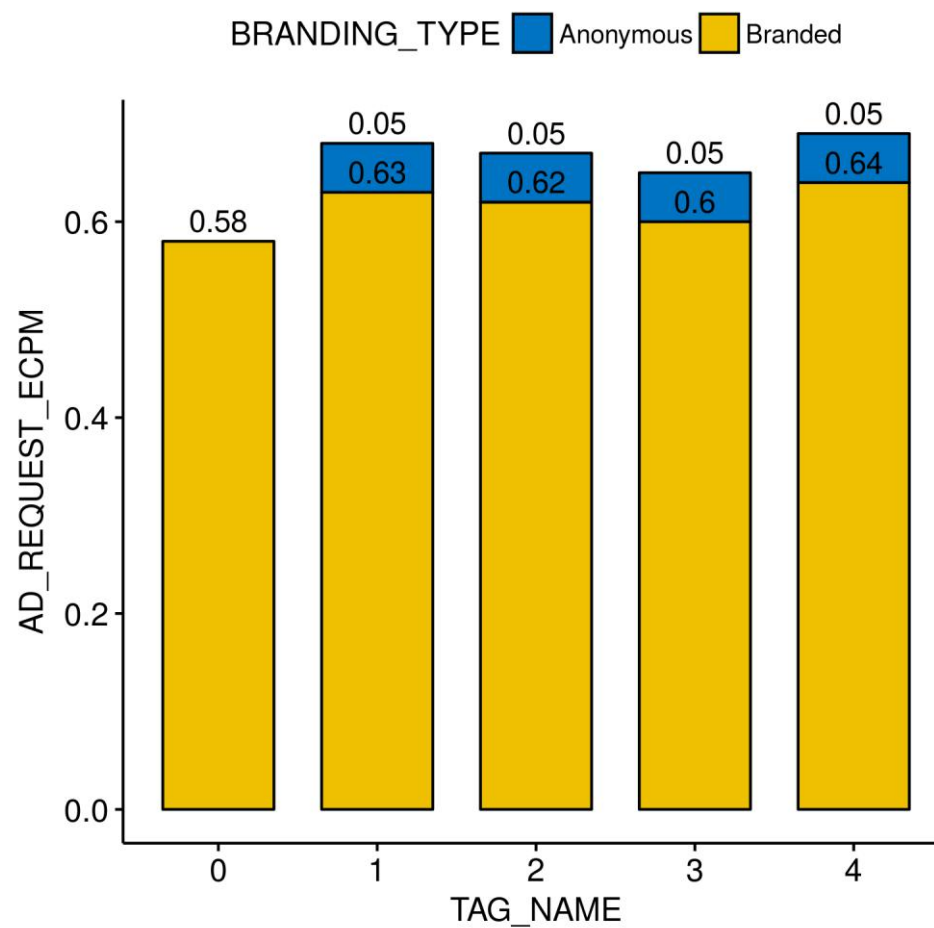
<b>data</b>	<b>tag_ad_unit</b>	<b>rule_name</b>	<b>wyjsciowe_branded</b>	<b>wyjsciowe_anonymous</b>
2019-01-11	ku_test_300x250_mobile_1_T_0	ku_test_300x250_mobile_1_T_0	0	0

### 17.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

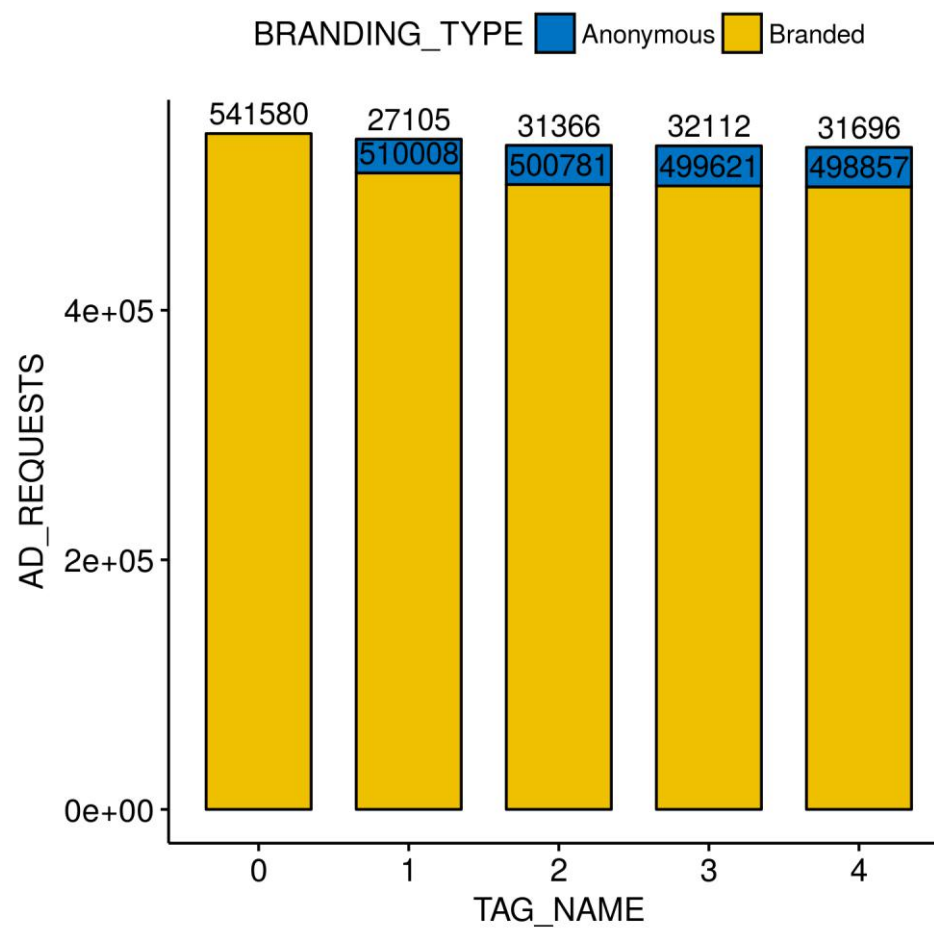




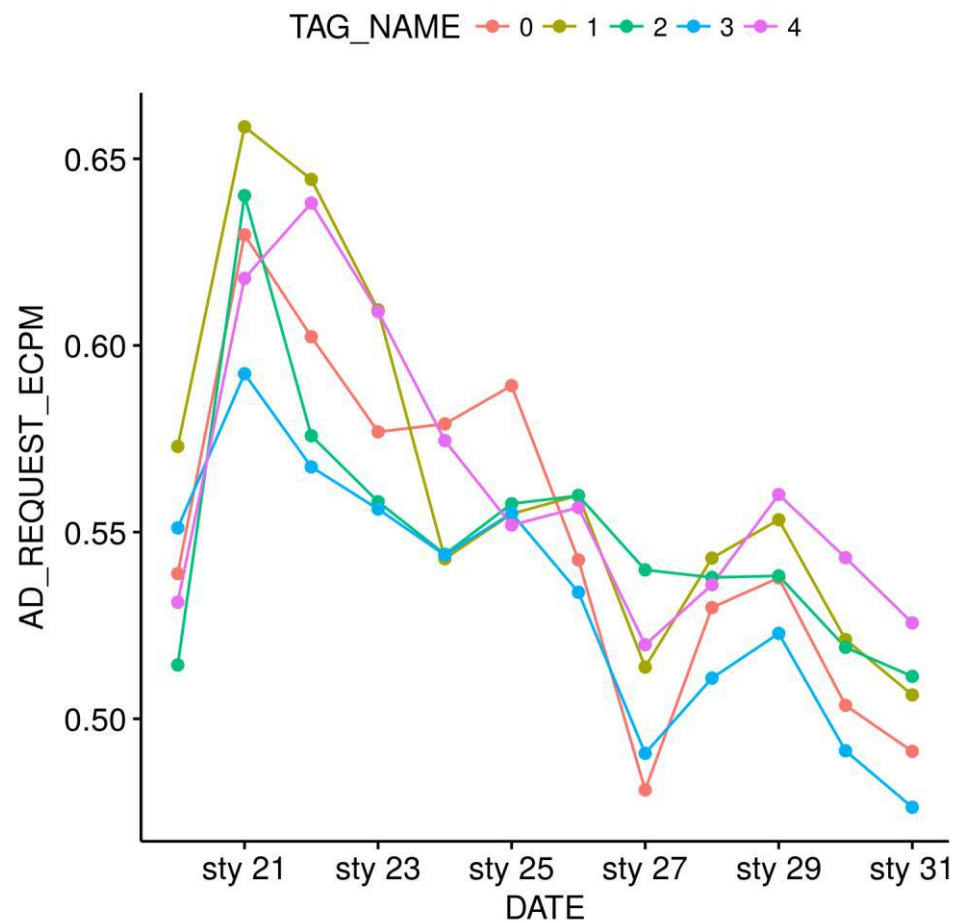
## 17.2 Wykres AdR ECPM z podziałem na Branding Type



### 17.3 Wykres Ad requests z podziałem na Branding Type



### 17.4 Wykres trendu dziennego AdRequest eCPM



### 17.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-20	0.514410	0.572942	0.058532	0.10765971

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-21	0.592417	0.658561	0.066144	0.10574766
2019-01-22	0.567455	0.644476	0.077021	0.12710460
2019-01-23	0.556170	0.609516	0.053346	0.09152722
2019-01-24	0.542860	0.578977	0.036117	0.06438903
2019-01-25	0.551874	0.589214	0.037340	0.06544631
2019-01-26	0.533884	0.559866	0.025982	0.04750994
2019-01-27	0.480944	0.539875	0.058931	0.11545827
2019-01-28	0.510891	0.543036	0.032145	0.06100043
2019-01-29	0.522864	0.560051	0.037187	0.06867944
2019-01-30	0.491422	0.543151	0.051729	0.10000068
2019-01-31	0.476294	0.525676	0.049382	0.09856982



## 18 Eksperyment di\_test\_300x250\_desktop\_T\_ (od 2019-1-22 do 2019-1-31)

Data wygenerowania raportu: 2019-02-01

Cel eksperymentu: Optymalizacja floorów z dokładnością do DEVICE\_CATEGORY. Ustawione reguły schodkowo:

\_1 (50%) mobile: (Wasza optymalizacja)

\_2 (10%) mobile: A = 0.05; B = 0.05

\_3 (10%) mobile: A = 0.1; B = 0.1

\_4 (10%) mobile: A = 0.15; B = 0.15

\_1 (50%) desktop: (Wasza optymalizacja)

\_2 (10%) desktop: A = 0.1; B = 0.1

\_3 (10%) desktop: A = 0.2; B = 0.2

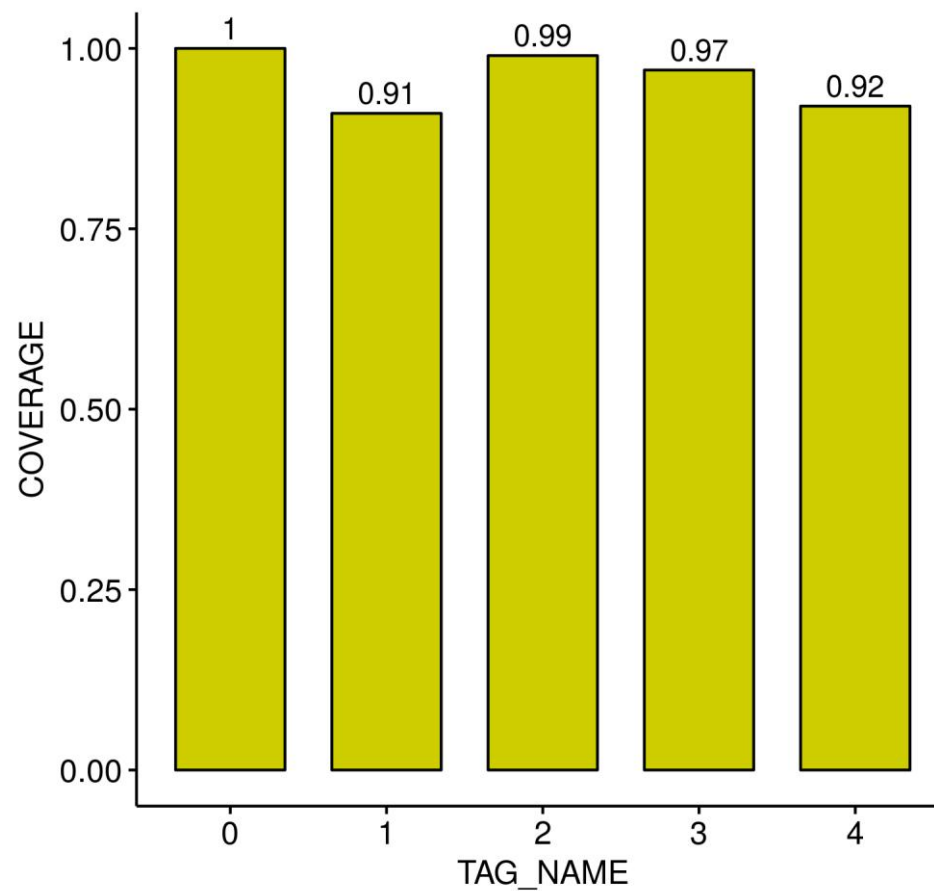
\_4 (10%) desktop: A = 0.25; B = 0.25

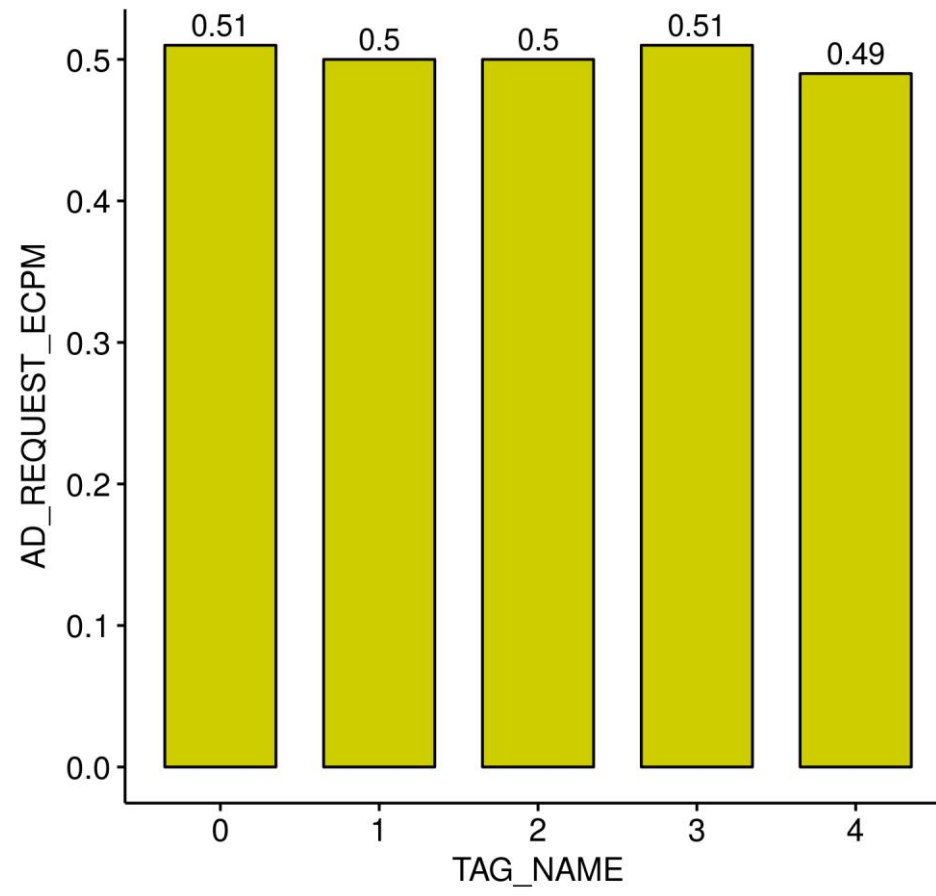
**Wnioski:** Na desktopie optymalne ARE na \_2 (wykres 1.6 ARE i Requests w podziale na TAG\_NAME i DEVICE\_CATEGORY), ale tam jest mniej requestów. Główny dochód ARE jest tworzony na Mobile – tam podwyższenie reguł nic nie zmienia/minimalnie pogarsza sytuację. W totalu ARE porównywalne na wszystkich tagach. Uwaga: W międzyczasie kampania do 31.01.

data	tag_ad_unit	rule_name	wyjsciowe_brande d	wyjsciowe_anonimo us
2019-01-22	di_test_300x250_desktop_T_1	di_test_300x250_desktop	0.5	0.25
2019-01-22	di_test_300x250_desktop_T_2	di_test_300x250_desktop_T_2(mob)	0.05	0.05
2019-01-22	di_test_300x250_desktop_T_3	di_test_300x250_desktop_T_3(mob)	0.1	0.1

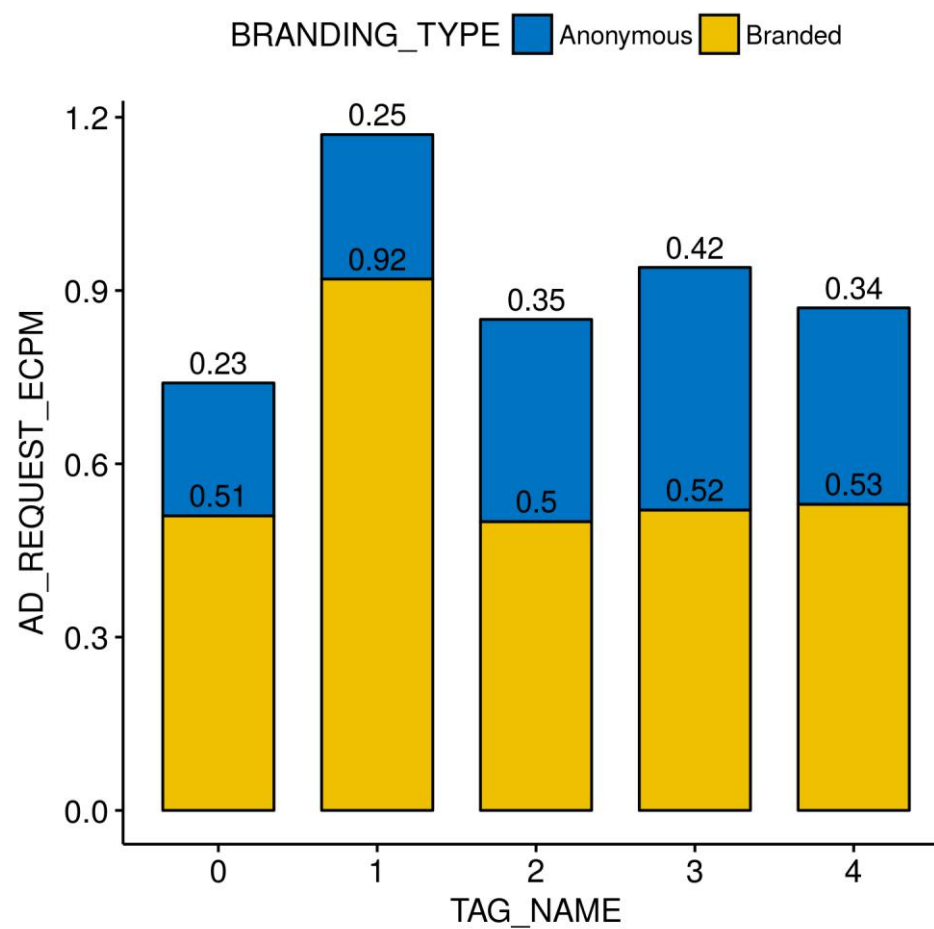
<b>data</b>	<b>tag_ad_unit</b>	<b>rule_name</b>	<b>wyjsciove_brande d</b>	<b>wyjsciove_anonymo us</b>
2019 -01- 22	di_test_300x250_desktop_T _4	di_test_300x250_desktop_T_4(de sk)	0.25	0.25

### 18.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

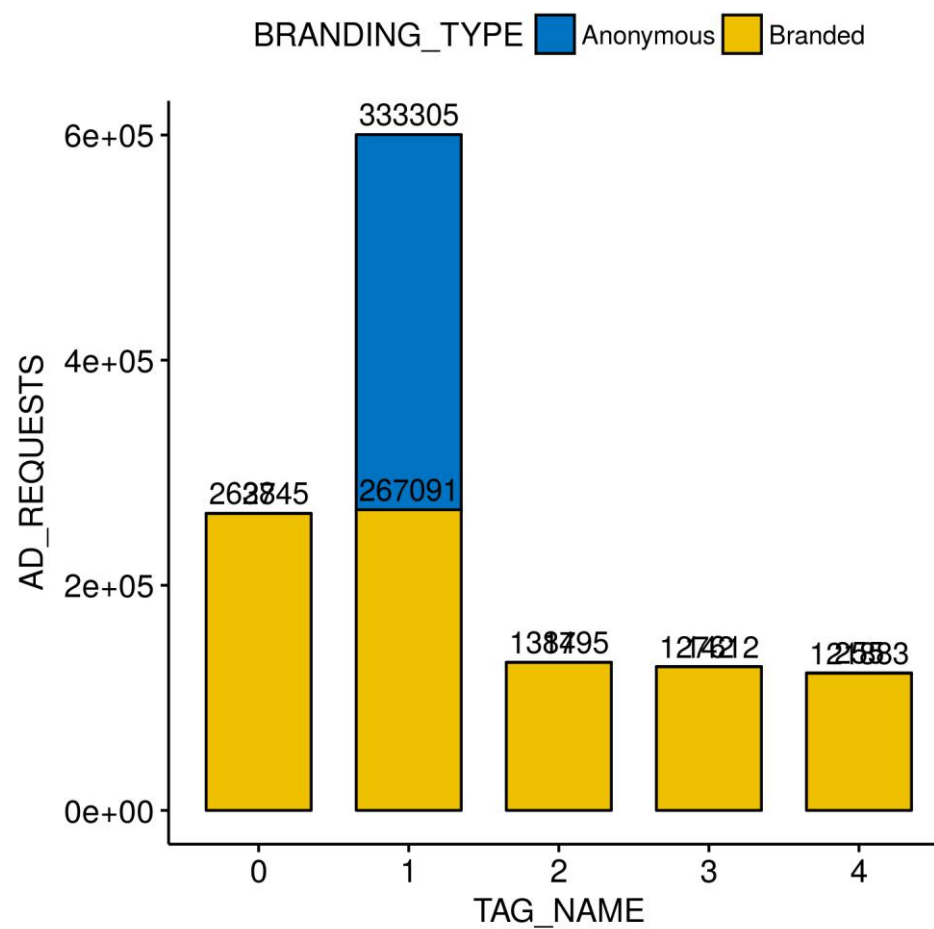




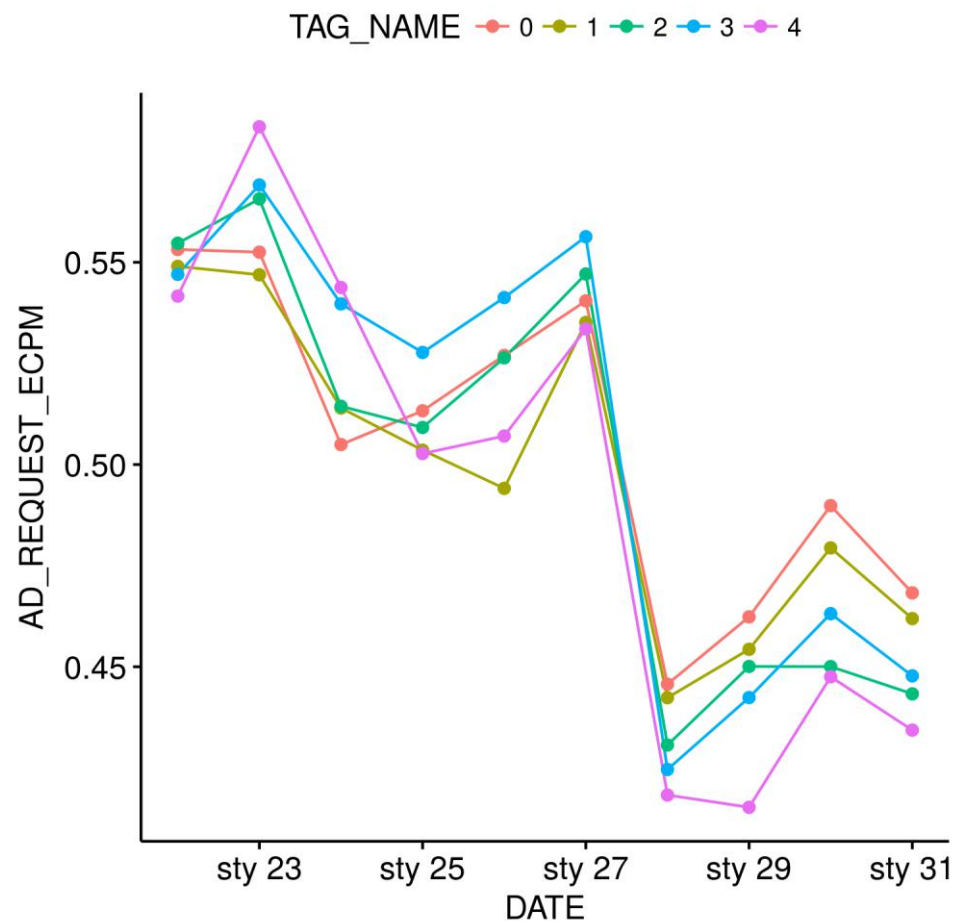
## 18.2 Wykres AdR eCPM z podziałem na Branding Type



### 18.3 Wykres Ad requests z podziałem na Branding Type



### 18.4 Wykres trendu dziennego AdRequest eCPM



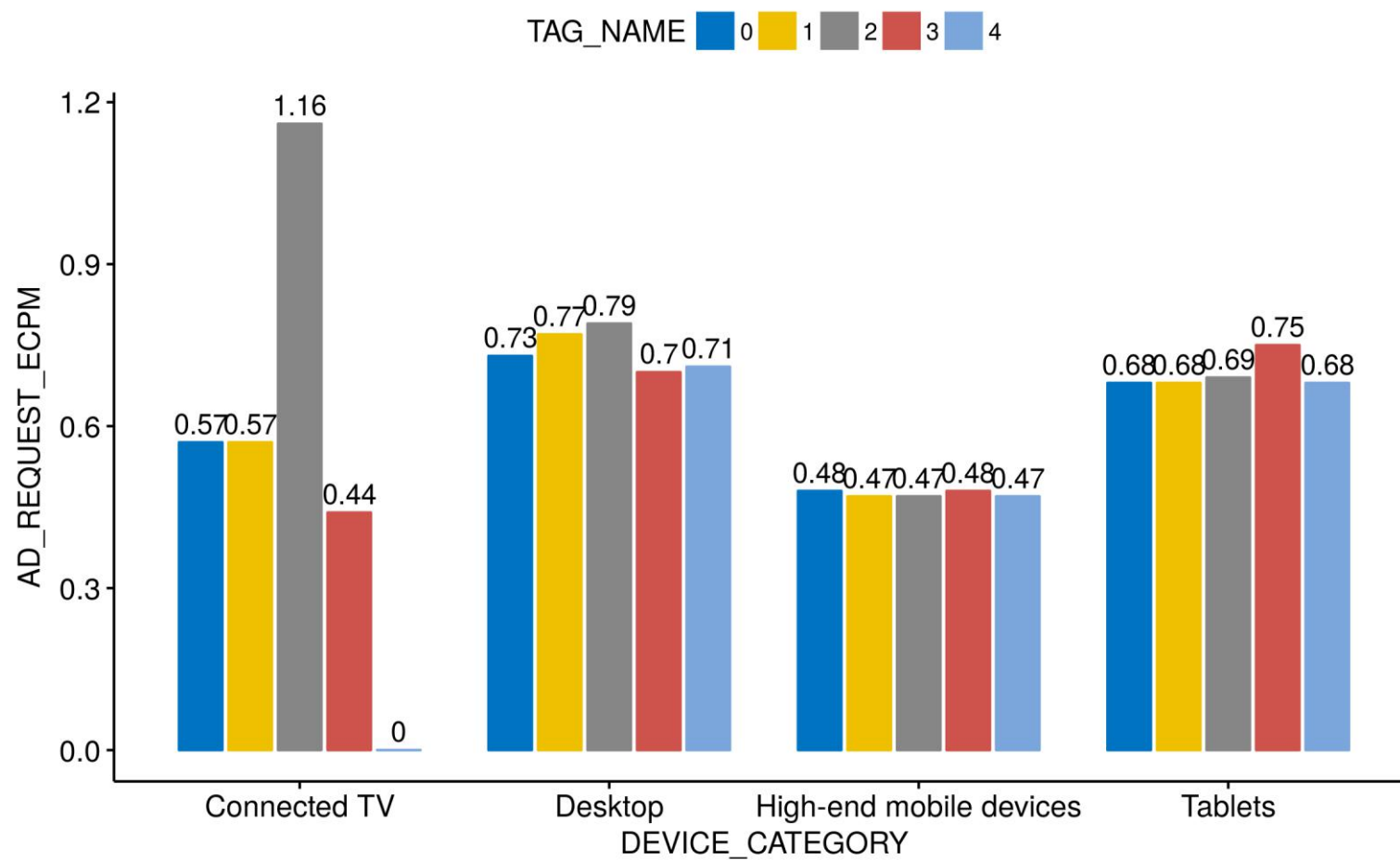
### 18.5 Wahania dzienne ARe

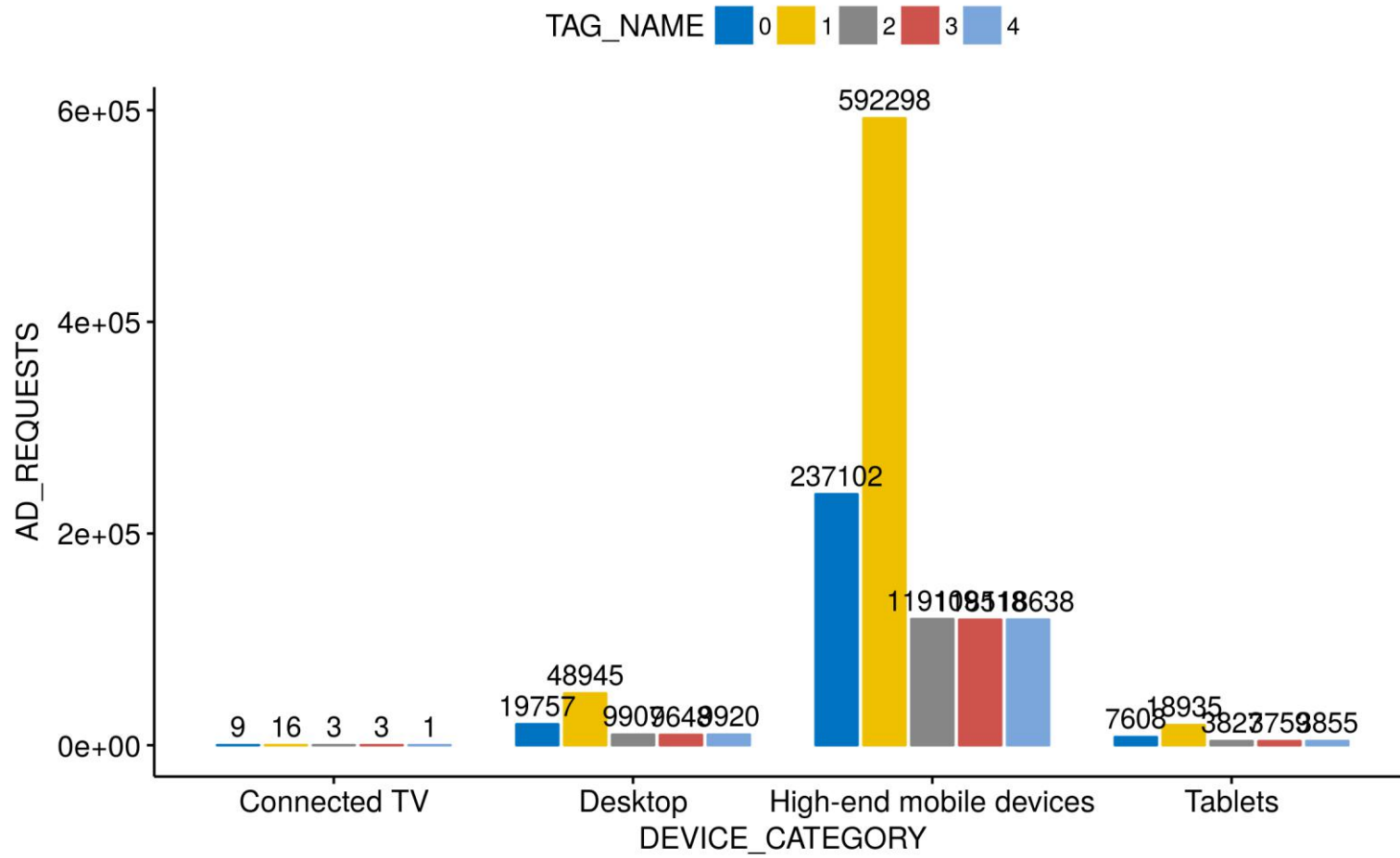
DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-22	0.541637	0.554710	0.013073	0.02384829

<b>DATE</b>	<b>ARE_min</b>	<b>ARE_max</b>	<b>ARE_diff</b>	<b>ARE_diff_pr</b>
2019-01-23	0.546916	0.583499	0.036583	0.06472490
2019-01-24	0.504935	0.543772	0.038837	0.07406645
2019-01-25	0.502715	0.527703	0.024988	0.04850071
2019-01-26	0.494105	0.541274	0.047169	0.091111446
2019-01-27	0.533514	0.556318	0.022804	0.04184865
2019-01-28	0.418280	0.445699	0.027419	0.06347145
2019-01-29	0.415260	0.462329	0.047069	0.10726889
2019-01-30	0.447516	0.489841	0.042325	0.09030711
2019-01-31	0.434340	0.468268	0.033928	0.07517771



## 18.6 ARe i Requests w podziale na TAG\_NAME i DEVICE\_CATEGORY





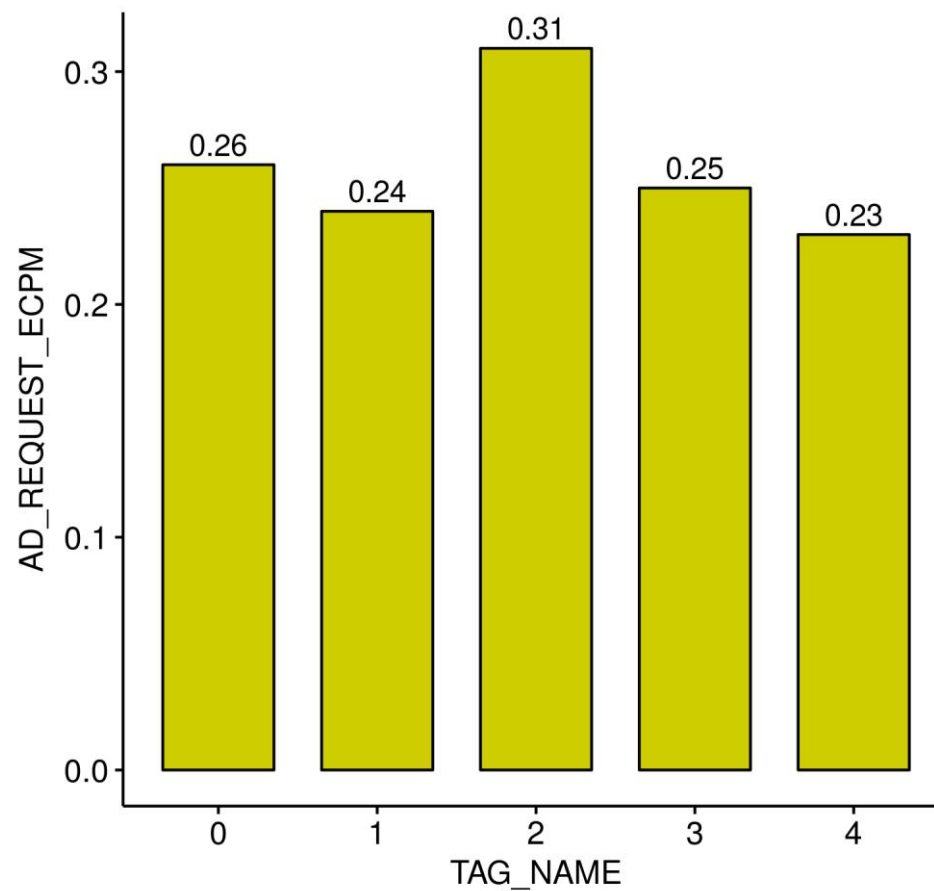
## **19 Eksperyment ip\_test\_750x200\_LIST\_D\_E\_F\_T\_ (od 2019-1-23 do 2019-1-31)**

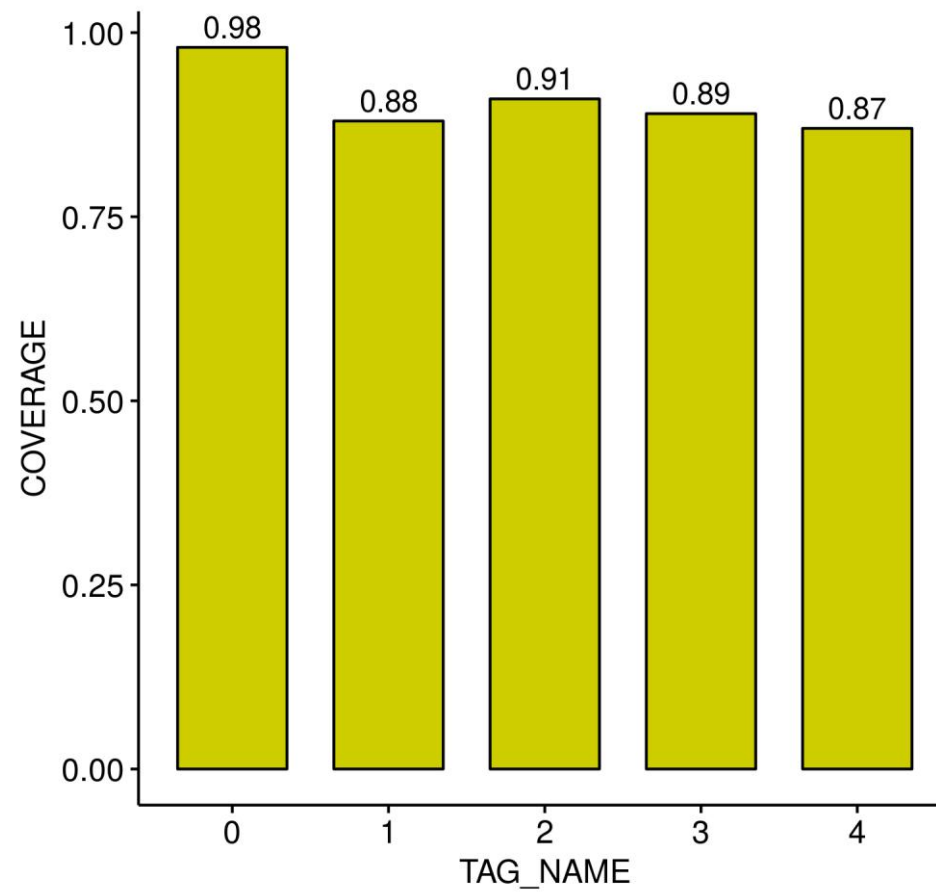
Data wygenerowania raportu: 2019-02-01

Cel eksperymentu: Eksperyment rotacji, 1/1/1/1 na \_1 do \_4 ustawione takie same reguły: A = 0.05 B = 0.10

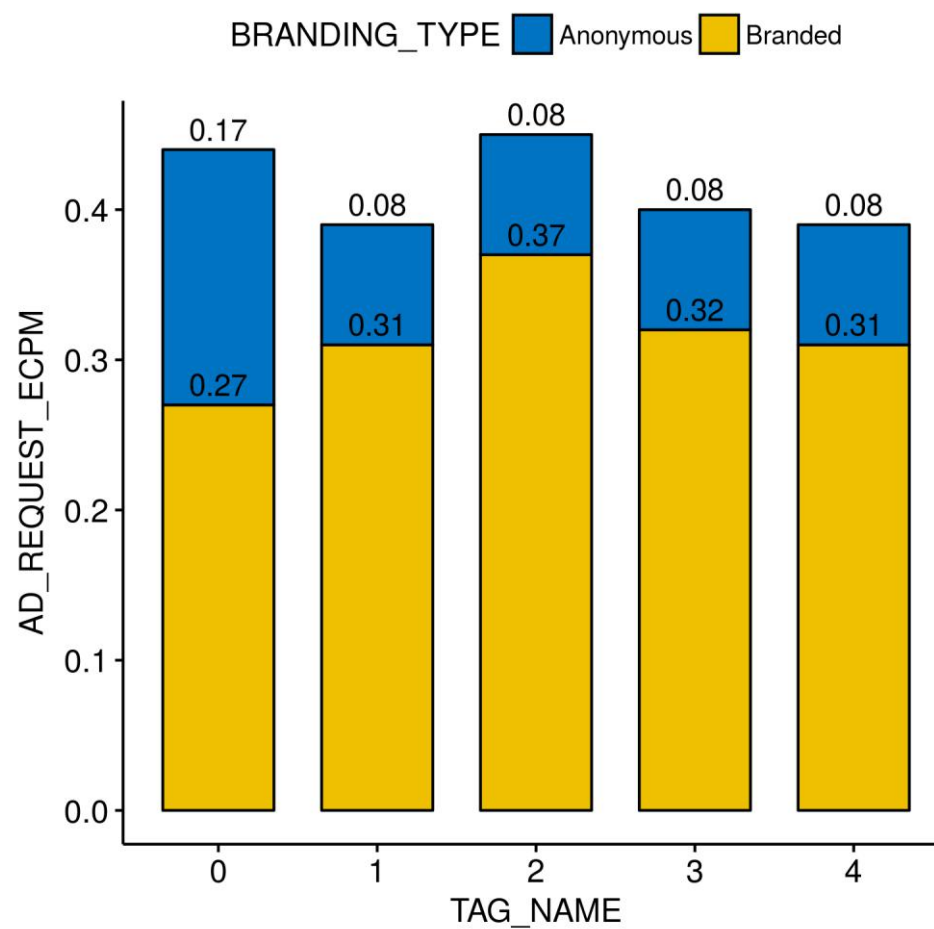
Wnioski: Z niewiadomych powodów znaczne podwyższenie ARe i Coverage (praktycznie codziennie) na \_2. Być może to efekt kampanii po stronie kupujących o której nie wiemy.

### 19.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

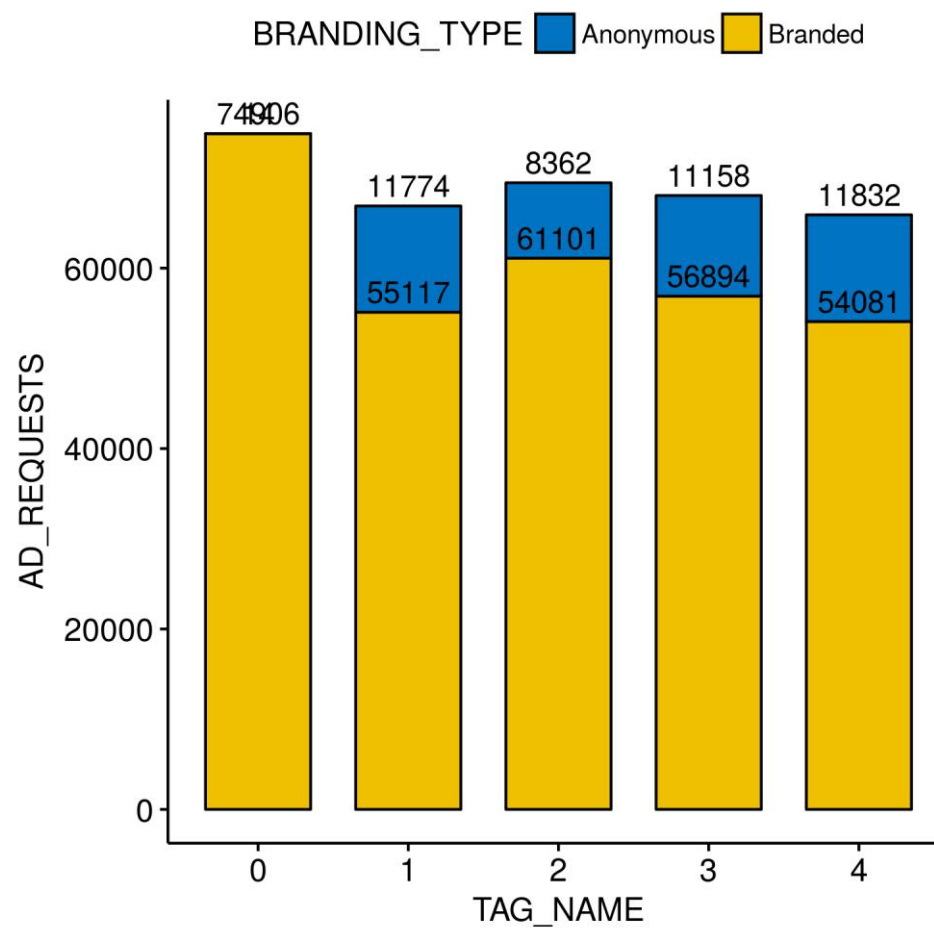




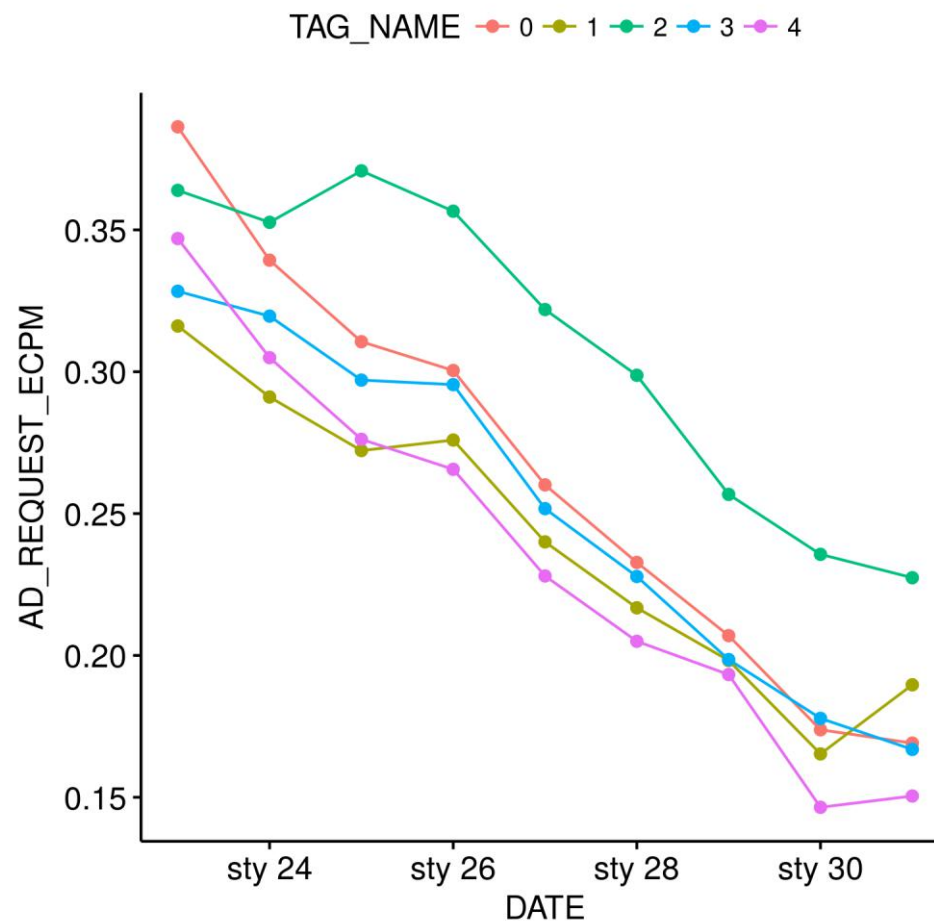
## 19.2 Wykres AdR ECPM z podziałem na Branding Type



### 19.3 Wykres Ad requests z podziałem na Branding Type



### 19.4 Wykres trendu dziennego AdRequest eCPM



### 19.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-23	0.316113	0.386335	0.070222	0.1999351



<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-24	0.291090	0.352659	0.061569	0.1912826
2019-01-25	0.272213	0.370812	0.098599	0.3066724
2019-01-26	0.265600	0.356585	0.090985	0.2924693
2019-01-27	0.228013	0.321929	0.093916	0.3415487
2019-01-28	0.204983	0.298741	0.093758	0.3722594
2019-01-29	0.193255	0.256765	0.063510	0.2822541
2019-01-30	0.146473	0.235629	0.089156	0.4666607
2019-01-31	0.150456	0.227391	0.076935	0.4072283

## 20 Eksperyment i0\_test\_970x250\_atf\_T\_ (od 2019-1-23 do 2019-1-31)

Data wygenerowania raportu: 2019-02-01

Reguły: \_0  $A = B = 0$

\_1 YR Wasza optymalizacja

\_2  $A = 0.20$   $B = 0.25$

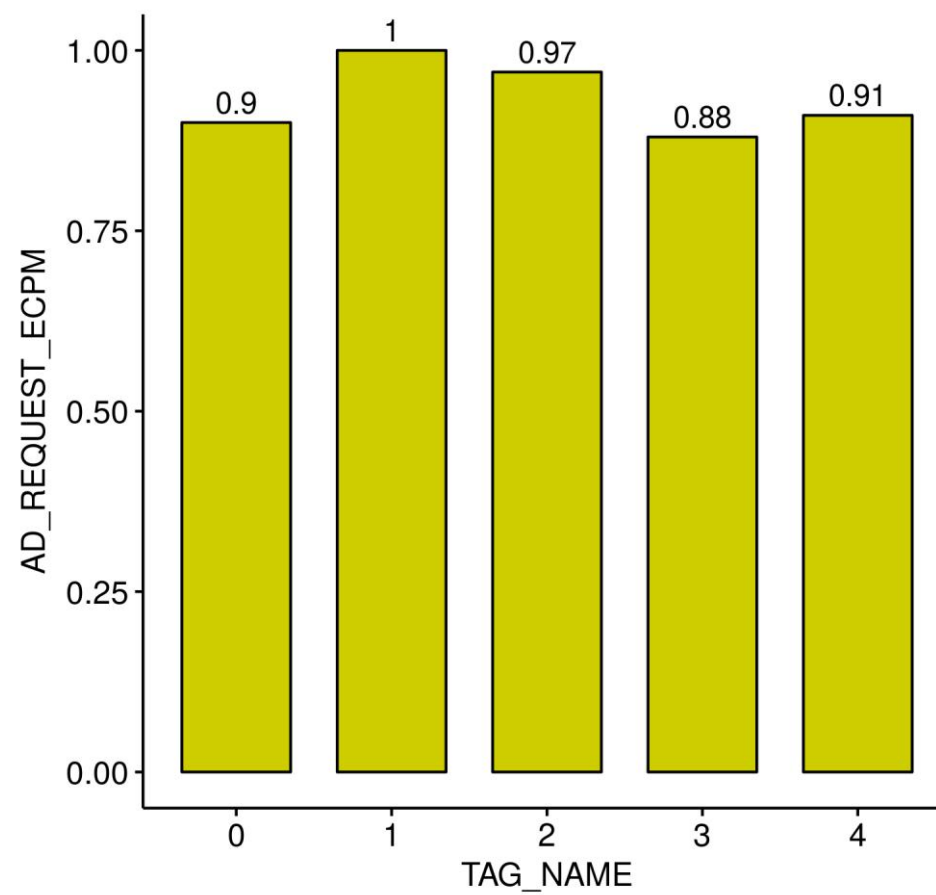
\_3  $A = 0.40$   $B = 0.45$

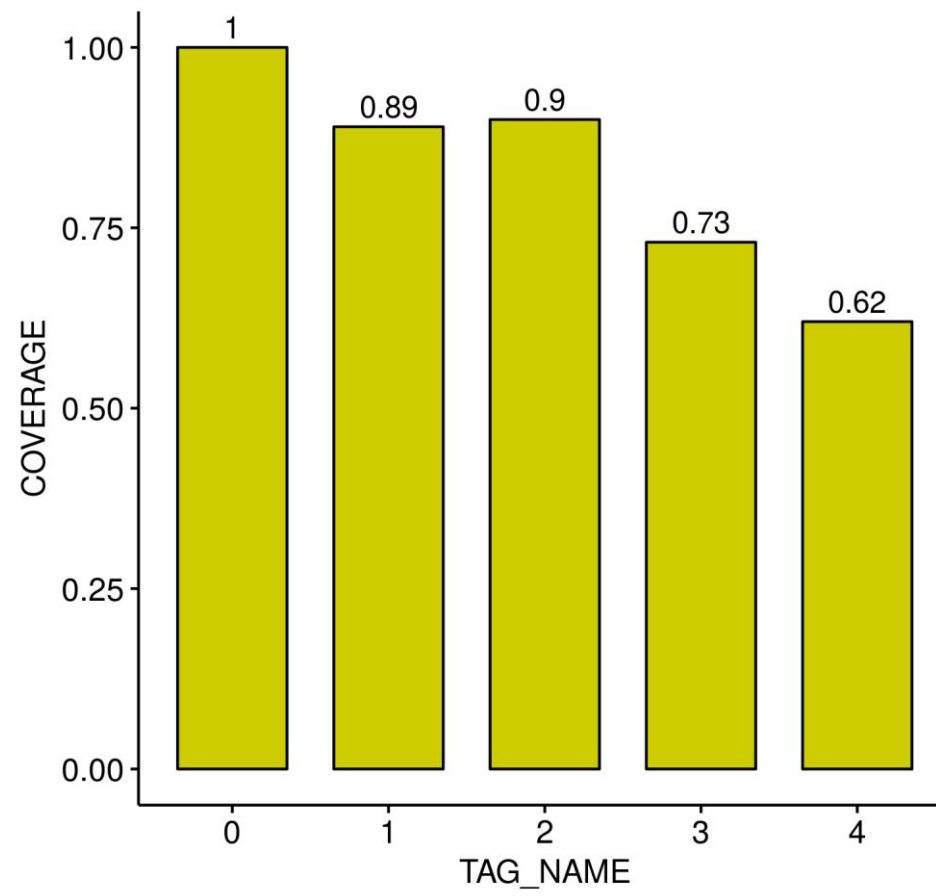
\_4  $A = 0.60$   $B = 0.65$

Cel: Optymalizacja ARe ze względu na ustawione schodkowo reguły.

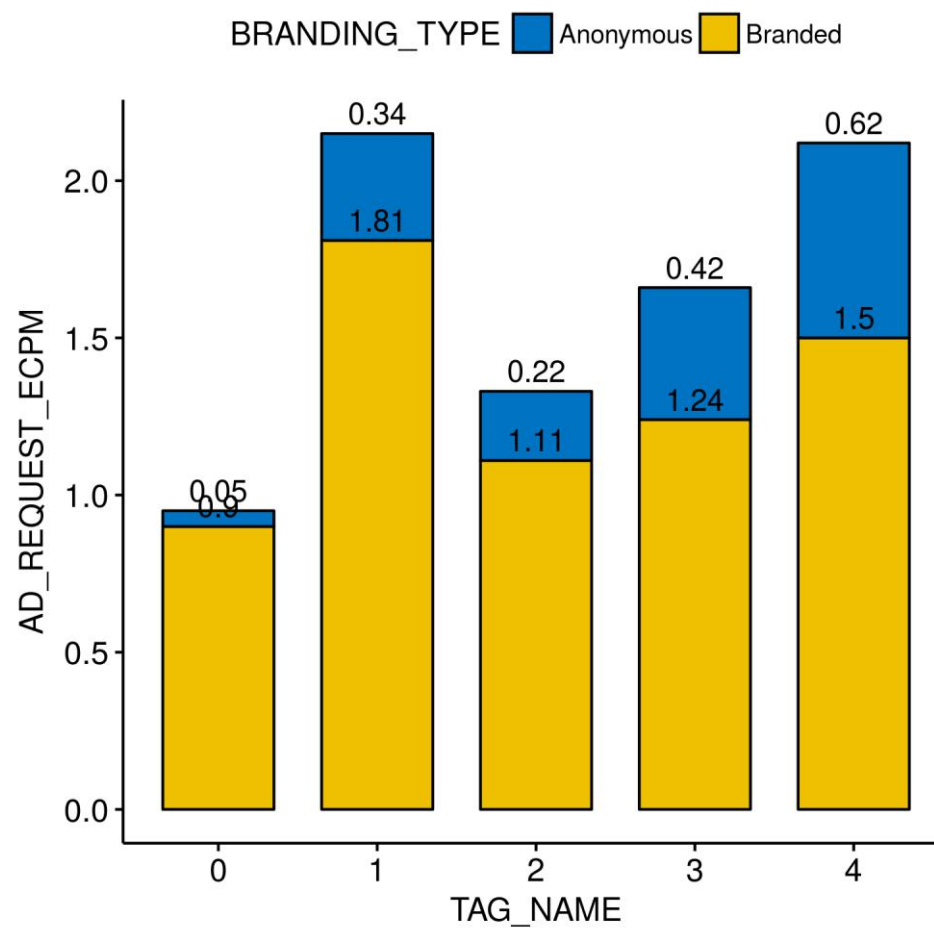
Wnioski: Najwyższy ARe na \_1 (brak informacji o ustawionych regułach) i \_2,. W kolejnym kroku ustawienie jeszcze niższych reguł.

## 20.1 Wykres Coverage i AdR ECPM bez podziału na Branding Type

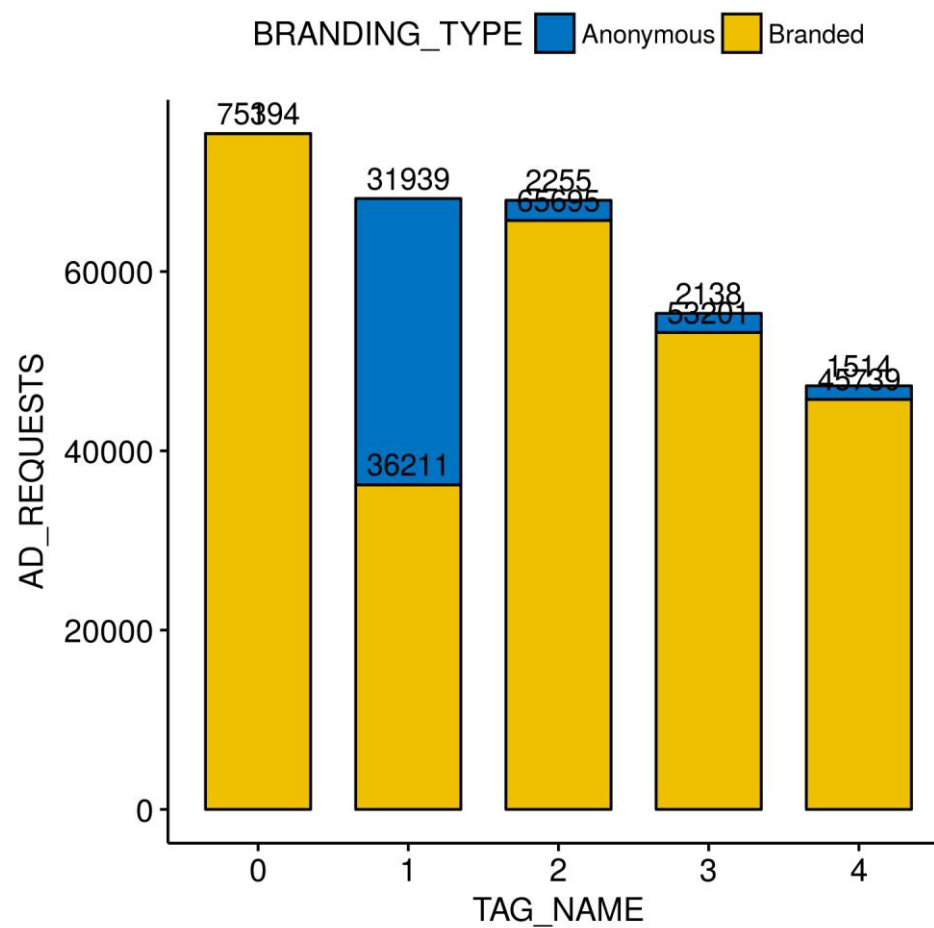




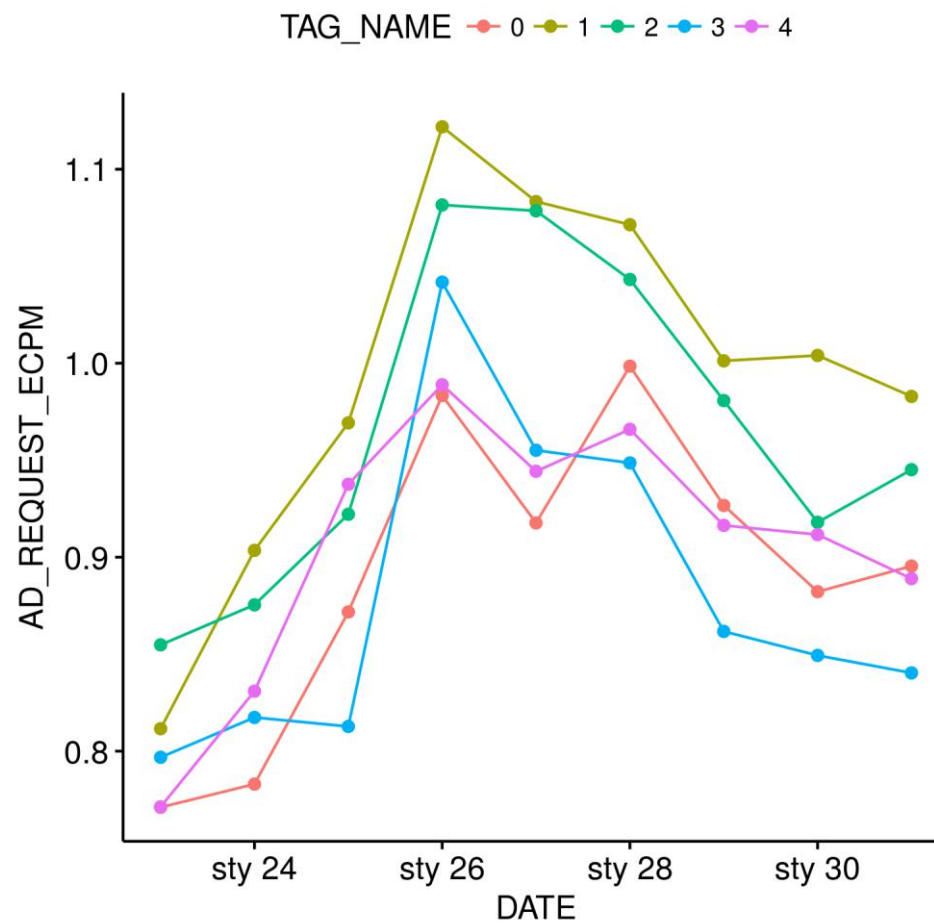
## 20.2 Wykres AdR ECPM z podziałem na Branding Type



### 20.3 Wykres Ad requests z podziałem na Branding Type



## 20.4 Wykres trendu dziennego AdRequest eCPM



## 20.5 Wahania dzienne ARe

DATE	ARe_min	ARe_max	ARe_diff	ARe_diff_pr
2019-01-23	0.771069	0.854822	0.083753	0.1030241

<b>DATE</b>	<b>ARe_min</b>	<b>ARe_max</b>	<b>ARe_diff</b>	<b>ARe_diff_pr</b>
2019-01-24	0.783019	0.903506	0.120487	0.1428820
2019-01-25	0.812735	0.969256	0.156521	0.1756698
2019-01-26	0.983350	1.121888	0.138538	0.1316127
2019-01-27	0.917677	1.083339	0.165662	0.1655779
2019-01-28	0.948587	1.071417	0.122830	0.1216136
2019-01-29	0.861741	1.001238	0.139497	0.1497569
2019-01-30	0.849355	1.004011	0.154656	0.1668920
2019-01-31	0.840332	0.982885	0.142553	0.1563752



### Załącznik 3. Eksperymenty 'regiony'

#### I. Dotychczasowe eksperymenty

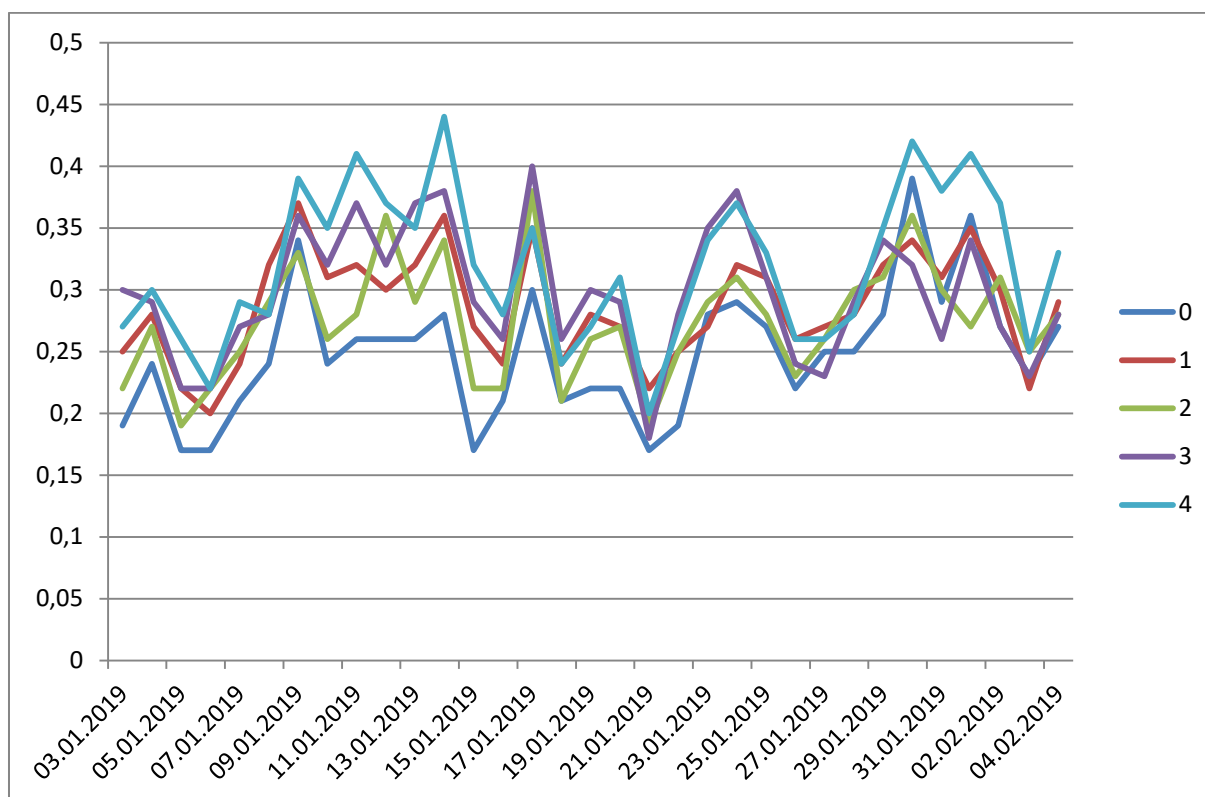
##### 1. Eksperyment ustawienia innych floor price dla krajów USA, Australia, Nowa Zelandia i UE.

Eksperyment główny: tag\_testowy(1-4); analiza dla 03-01-2019 do 04-02-2019

Celem eksperymentu było sprawdzenie, czy ustawienie wyższych floor price dla krajów, dla których historycznie ARecpm był wyższy, podniesie wartość ARecpm na danej regule i tagu. Ustawiono następujące reguły na tag\_testowy(1-4):

		fp_A	fp_B	kraje
_0	10%	0	0	wszystkie
_1	50%	0,04	0,60	wszystkie
_2_(1)	10%	0,04	1,20	USA, NZ, AST, UE
_2_(2)	10%	0,04	0,60	wszystkie
_3_(1)	5%	0,06	0,90	USA, NZ, AST, UE
_3_(2)	5%	0,04	0,60	wszystkie
_4_(1)	5%	0,08	1,20	USA, NZ, AST, UE
_4_(2)	5%	0,04	0,60	wszystkie

Dla okresu od 03-01-2019 do 04-02-2019 wyniki testu na pełnych tagach przedstawia poniższy wykres:



Wyniki wskazują, że tag 4 ustawiający regułę 2xYR dla badanych krajów zarówno w przypadku Branded, jak i Anonymous daje najlepsze efekty, znacznie odbiegające od poziomu optymalizacji YR oraz reguły zerowej:

Na 33 badane dni, reguła 4 dała najwyższy ARecpm dla 21 przypadków (reguła 3: 9 przypadków), zaś przeciętne odchylenie procentowe od reguły YR wyniosło 11% (reguła 3: 4%).

	0	1	2	3	4
średni ARecpm	0,25	0,29	0,27	0,30	0,32
ilość dni z maksymalnym wynikiem danego tagu	0	4	3	9	21
przeciętne odchylenie procentowe ARecpm od reguły YR	-14%	0%	-4%	4%	11%

Z analizy coverage wynika natomiast, że reguła 4 pomimo najwyższych floor price ma często wyższy coverage niż reguły o niższym poziomie floor price (z pominięciem reguły 0):

	0	1	2	3	4
średni coverage	0,86	0,73	0,71	0,72	0,75
ilość dni z maksymalnym wynikiem danego tagu (bez reguły 0)	ND	9	4	3	17
przeciętne odchylenie procentowe ARecpm od reguły YR	18%	0%	-2%	-1%	3%

W celu zrozumienia mechanizmu stojącego za powyższymi wynikami przeanalizowano poziomy ARecpm i Coverage dla poszczególnych krajów

POLSKA	0	1	2	3	4
średni ARecpm	0,20	0,23	0,22	0,23	0,25
ilość dni z maksymalnym wynikiem danego tagu	1	4	3	7	22
przeciętne odchylenie procentowe ARecpm od reguły YR	-14%	0%	-5%	2%	9%

NIEMCY	0	1	2	3	4
średni ARecpm	0,73	0,72	0,73	0,77	0,86
ilość dni z maksymalnym wynikiem danego tagu	4	6	6	7	11
przeciętne odchylenie procentowe ARecpm od reguły YR	4%	0%	4%	13%	21%

UK	0	1	2	3	4
średni ARecpm	0,51	0,55	0,56	0,55	0,61
ilość dni z maksymalnym wynikiem danego tagu	1	6	10	7	11
przeciętne odchylenie procentowe ARecpm od reguły YR	-5%	0%	3%	2%	10%

USA	0	1	2	3	4
średni ARecpm	1,71	2,01	2,12	1,75	2,01
ilość dni z maksymalnym wynikiem danego tagu	4	8	12	5	4
przeciętne odchylenie procentowe Arecpm od reguły YR	-9%	0%	7%	-8%	7%

POLSKA	0	1	2	3	4
średni Coverage	0,85	0,73	0,71	0,72	0,76
ilość dni z maksymalnym wynikiem danego tagu	NA	9	3	3	19
przeciętne odchylenie procentowe Coverage od reguły YR	18%	0%	-2%	-1%	4%

NIEMCY	0	1	2	3	4
średni Coverage	0,95	0,90	0,89	0,90	0,88
ilość dni z maksymalnym wynikiem danego tagu	NA	6	13	8	6
przeciętne odchylenie procentowe Coverage od reguły YR	5%	0%	-1%	0%	-3%

UK	0	1	2	3	4
średni Coverage	0,94	0,85	0,85	0,84	0,82
ilość dni z maksymalnym wynikiem danego tagu	NA	9	14	6	4
przeciętne odchylenie procentowe Coverage od reguły YR	10%	0%	1%	-1%	-4%

USA	0	1	2	3	4
średni Coverage	0,90	0,87	0,89	0,84	0,80
ilość dni z maksymalnym wynikiem danego tagu	NA	7	19	9	8
przeciętne odchylenie procentowe Coverage od reguły YR	4%	0%	2%	-3%	-8%

Z zestawienia widać, że tag 4 dawał znacznie lepsze wyniki (dla każdej z 3 zastosowanych miar) w przypadku Niemiec i UK, nie wpływała natomiast istotnie na wyniki dla USA. Co ciekawe, rozdzielanie ruchu wynikające z zastosowania odmiennych reguł cenowych dla grupy, w której znajdowała się Polska spowodowało, że również w przypadku Polski ARecpm był znacząco wyższy w 4 tagu, pomimo, że floor price w przy tej regule był taki sam jak dla optymalizacji YR.

Części wyjaśnienia dostarcza analiza Coverage. Widać na jej podstawie, że wprowadzenie floor price dla USA spowodowało znaczący spadek coverage, istotnie większy niż w przypadku UK i Niemiec. Dotyczy on jednak głównie reguły 4, ale już nie reguły 2, z czego wynika, że o spadku coverage decyduje głównie zwiększenie floor price dla Anonymous (reguła 2 ma ten sam floor price dla Branded co reguła 4, ale floor price dla Anonymous jest utrzymany na poziomie YR).

W przypadku Polski rozdzielanie ruchu spowodowało wzrost Coverage dla reguły 4, można więc sądzić, że efekt ten został wywołany przez podniesienie floor price na Anonymous dla krajów USA, NZ, AST i UE (ponownie, do takiego wniosku skłania brak wpływu na Coverage w regule 2).

W celu wyjaśnienia rozbieżności między efektami dla UK i Niemiec z jednej strony i brakiem efektów dla USA z drugiej strony oraz pozytywnym wpływem rozdzielania ruchu na wyniki Polski przeanalizowano dalsze rozbieżności, dodając do analizy wymiar targetowania (Branded/Anonymous).

POLSKA - ANONYMOUS	0	1	2	3	4
średni ARecpm	0,09	0,09	0,09	0,09	0,09
ilość dni z maksymalnym wynikiem danego tagu	8	18	15	13	19
przeciętne odchylenie procentowe Arecpm od reguły YR	-47%	0%	0%	-1%	1%

POLSKA - BRANDED	0	1	2	3	4
średni ARecpm	0,23	1,03	0,99	1,00	1,02
ilość dni z maksymalnym wynikiem danego tagu	0	13	8	6	9
przeciętne odchylenie procentowe Arecpm od reguły YR	-78%	0%	-4%	-3%	-1%

W przypadku Polski widać (zgodnie z oczekiwaniami) brak różnic w poziomie ARecpm dla Branded i Anonymous (który w tym przypadku jest równy Matchedecpm, gdyż Coverage jest tu równy 100%), różnica w ARecpm na poziomie tagów wynika więc w całości z poziomu Coverage.

W przypadku analizy pozostałych krajów widać, że poziom ARecpm dla Branded rośnie zgodnie z poziomem ustawionych reguł dla Niemiec i UK, a dla US wzrost ten jest wyraźnie mniejszy. Wynika to prawdopodobnie z faktu, iż w przypadku US poziom wyjściowy (dla reguły 0) jest wielokrotnie wyższy, niż w przypadku Niemiec i UK. Prawdopodobnie więc podwyższenie floor price dla branded w USA mogłoby dać lepsze efekty.

W przypadku ARecpm dla Anonymous nie widać tego typu różnic

NIEMCY - ANONYMOUS	0	1	2	3	4
średni ARecpm	NA	0,12	0,16	0,15	0,19
ilość dni z maksymalnym wynikiem danego tagu	NA	0	8	4	21
przeciętne odchylenie procentowe Arecpm od reguły YR	NA	0%	30%	23%	54%

NIEMCY - BRANDED	0	1	2	3	4
średni ARecpm	0,77	1,34	2,08	1,74	2,10
ilość dni z maksymalnym wynikiem danego tagu	0	1	13	4	15
przeciętne odchylenie procentowe Arecpm od reguły YR	-41%	0%	59%	36%	61%

UK - ANONYMOUS	0	1	2	3	4
średni ARecpm	NA	0,09	0,12	0,11	0,15
ilość dni z maksymalnym wynikiem danego tagu	NA	0	5	5	27
przeciętne odchylenie procentowe Arecpm od reguły YR	NA	0%	24%	21%	63%

UK - BRANDED	0	1	2	3	4
średni ARecpm	0,55	1,23	2,06	1,55	2,19
ilość dni z maksymalnym wynikiem danego tagu	0	0	14	5	14
przeciętne odchylenie procentowe Arecpm od reguły YR	-54%	0%	71%	30%	76%

USA - ANONYMOUS	0	1	2	3	4
średni ARecpm	NA	0,13	0,17	0,19	0,20

ilość dni z maksymalnym wynikiem danego tagu	NA	6	6	9	15
przeciętne odchylenie procentowe Arecpm od reguły YR	NA	0%	29%	43%	64%

USA - BRANDED	0	1	2	3	4
średni ARecpm	1,88	2,96	3,41	2,91	3,73
ilość dni z maksymalnym wynikiem danego tagu	0	6	10	8	9
przeciętne odchylenie procentowe Arecpm od reguły YR	-30%	0%	34%	8%	34%

#### DALSZE KROKI:

- a) Uruchomienie eksperymentu na 2-3 tagach równocześnie w podziale na 5 grup po 20%: zerówka (\_0), YR(\_1), kraje, odpowiednio, 1.5, 2 i 2.5 x YR, reszta YR (\_2, \_3, \_4). UWAGA: zgodnie z rekomendacjami z eksperymentów rotacyjnych konieczne jest wyczyszczenie dotychczasowych tagów
- b) Kontynuowanie eksperymentu na gd\_pyszności po powyższych zmianach

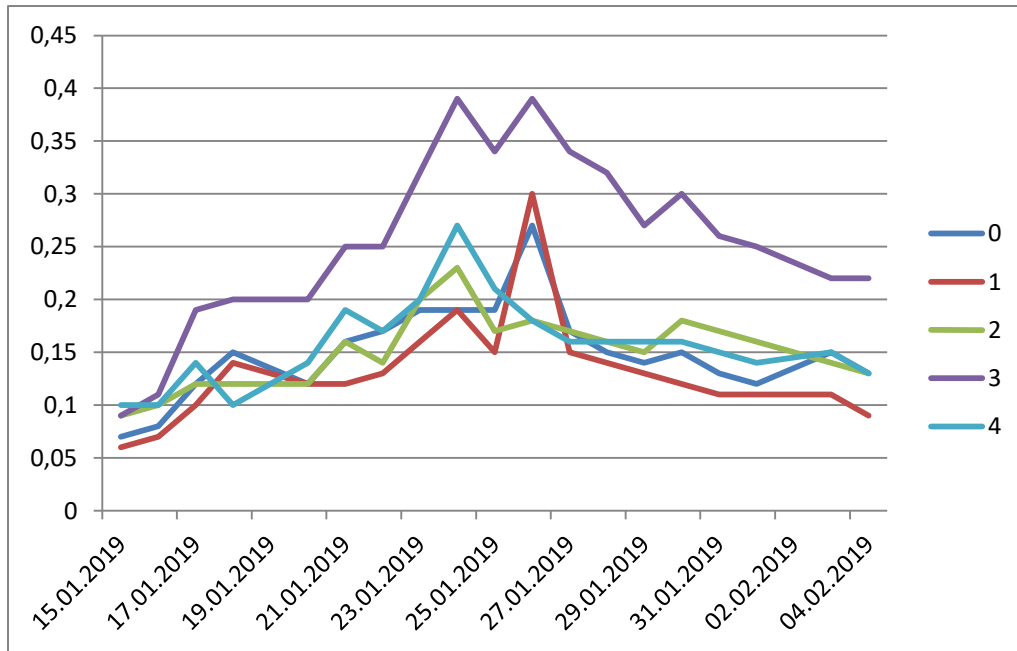
#### 2. Eksperyment grupowania reklamodawców.

Od 22 stycznia testowano możliwość przydzielenia danej reguły cenowej tylko wybranym reklamodawcom. Wybór dotyczył tag\_testowy i na tym etapie był losowy – potwierdził jednak możliwość takiego działania.

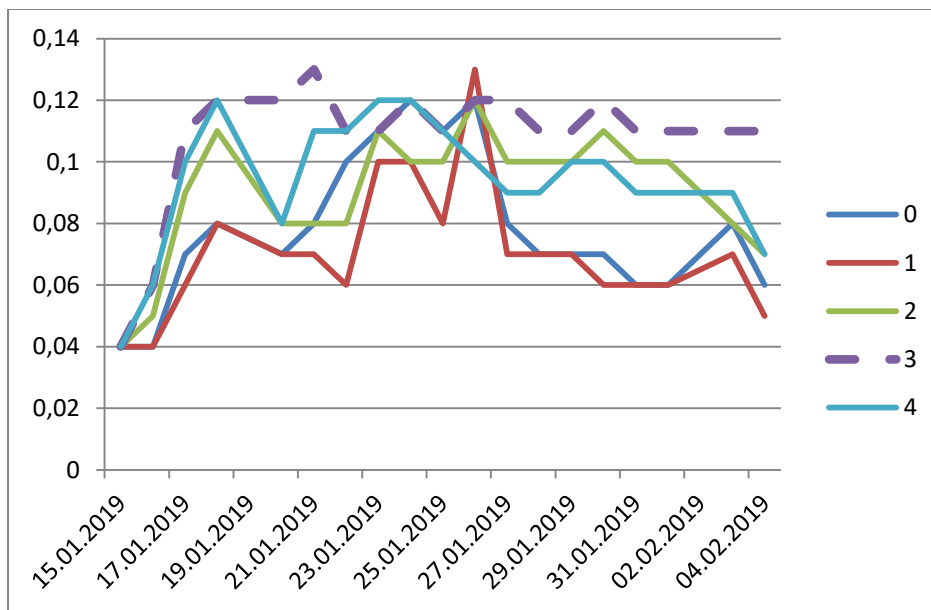
Ponieważ do eksperymentu wybrano zaledwie 3 reklamodawców, z których 1 był aktywny w znaczącym stopniu, tag/reguła cenowe, dla grupy wydzielonych reklamodawców charakteryzował się w badanym okresie bardzo niskim ARecpm ze względu na Coverage na poziomie ok 20-25% (pytanie techniczne: czy sposób „odcięcia” pozostałych reklamodawców nie powoduje, że Coverage jest sztucznie zaniżony tzn. reklamodawca spoza grupy wygrywa aukcję, ale nie ma matche, stąd Coverage jest niski).

**Ciekawym wnioskiem z eksperymentu i uzasadnieniem jego dalszej kontynuacji dla wybranych zgodnie z ustalonymi wcześniej zasadami reklamodawców jest jednak następująca obserwacja: w badanym okresie największy spośród trzech reklamodawców zmienił bardzo istotnie swoje zachowanie w ramach reguły, która obejmowała wydzieloną grupę, podwyższając na niej znacząco Average Bid i osiągając dużo wyższy procent wygranych licytacji, niż przy pozostałych tagach – prezentują to poniższe wykresy:**

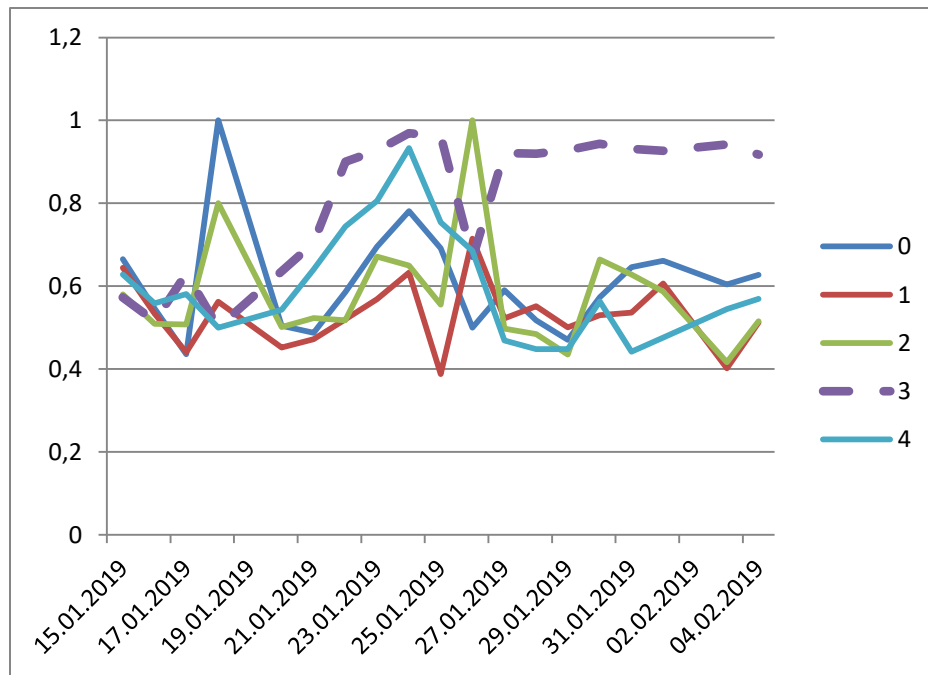
AVERAGE BID



CLOSE CPM



WINNING %



#### DALSZE KROKI:

Kontynuowanie eksperymentu na 2 tagach po uzgodnieniu na spotkaniu możliwości technicznych i ograniczeń związanych z Coverage

#### II. Rekomendacje kolejnych działań

1. Zautomatyzowanie procesu ustawiania reguł i zapisu zmian
2. Grupowanie tagów pod kątem struktury (struktura A i B, udział krajów innych niż Polska, koncentracja reklamodawców, koncentracja urzędzeń)
3. Wprowadzenie pierwszego eksperymentu automatycznego o następujących parametrach (na 3 tagach o różnych charakterystykach)
  - a. Podział tagu na 7 reguł:
    - i. Zerówka z podziałem na Aflorr0 i B florr0(\_0) – 10%
    - ii. Reguła główna, czyli reguła wygrywająca (o najwyższym ARecpm w poprzednim cyklu) z podziałem na Afloor i Bfloor (\_1) – 40%
    - iii. Challengers – 5 reguł (\_2 do \_6) z cenami ustawionymi na poziomie, odpowiednio, 0.5, 0.75, 1.25, 1.5, 2.0 x Afloor i Bfloor – każdy tag po 10%
  - b. Automatyczna analiza wyników zastosowania reguł po np. 1 tygodniu od zmiany według uzgodnionych z góry kryteriów (do ustalenia oprócz miary głównej jest sposób definiowania „zwycięstwo” danej reguły – odchylenie średnich, średnia odchylenia, liczba najwyższych wyników – oraz sposób traktowania outlierów)
  - c. Ustawienie nowej reguły głównej i przeliczenie reguł challengers. Alternatywnie można rozważyć ustawienie odchylenia challengers według wartości nominalnych (np. co 5, albo 10 groszy – zależnie od cen osiągniętych wyjściowo na tagu), a nie jako procent odchylenia od wygrywającej reguły.

4. Dalsze testowanie podziału na kraje (zgodnie z propozycją wyżej). Do tej części można ewentualnie dołączyć eksperyment automatyczny, w którym tagi \_2 do \_6 będą dodatkowo rozbite na kraje USA, NZ, AST i UE oraz resztę
5. Rozszerzenie eksperymentu podziału reklamodawców na grupy (po wyjaśnieniu wątpliwości związanych z niskim poziomem Coverage)