

UNIWERSYTET GDAŃSKI – WYDZIAŁ EKONOMICZNY

Sebastian Masłowski

Numer albumu: 270049

Kierunek studiów: EKONOMIA

**RYNEK NIERUCHOMOŚCI W LONDYNIE –
DASHBOARD ANALITYCZNY
Z ZASTOSOWANIEM NARZĘDZI BI**

Praca magisterska wykonana
w Katedrze Ekonomii
Międzynarodowej i Rozwoju
Gospodarczego
pod kierunkiem
prof. UG, dr hab. Stanisława
Umińskiego

Sopot 2024

SPIS TREŚCI

WSTĘP	3
ROZDZIAŁ 1. CHARAKTERYSTYKA GOSPODARKI LONDYNU ORAZ RYNKU NIERUCHOMOŚCI	6
1.1 Historia rozwoju oraz charakterystyka stolicy Wielkiej Brytanii.....	6
1.2 Charakterystyka rynku nieruchomości w metropoliach.....	9
1.3 Determinanty kształtowania się cen nieruchomości w Londynie.....	13
ROZDZIAŁ 2. OKREŚLENIE PROBLEMU BADAWCZEGO	17
2.1 Problem badawczy	17
2.2 Metoda eksploracji problemu badawczego i uzasadnienie jej doboru.....	17
ROZDZIAŁ 3. OCZYSZCZANIE, PRZETWARZANIE ORAZ PRZYGOTOWYWANIE DANYCH Z UŻYCIEM SQL.....	21
3.1 Zbiór danych z różnych źródeł oraz przygotowanie mapy przy pomocy QGIS	21
3.2 Oczyszczanie danych oraz modelowanie	24
ROZDZIAŁ 4. BUDOWA DASHBOARDU ANALITYCZNEGO W POWER BI Z UŻYCIEM POWER QUERY I JĘZYKA DAX.....	48
4.1 Karta najem krótkoterminowy	48
4.2 Karta model najmu krótkoterminowego.....	55
4.3 Karta rynek pracy	59
4.4 Karta siła nabywcza	65
4.5 Karta przestępcość	79
4.6 Karta korelacje	93
4.7 Karta miernik syntetyczny	98
ZAKOŃCZENIE	102
TITLE AND ABSTRACT	104
BIBLIOGRAFIA	105
SPIS RYSUNKÓW	109
ZAŁĄCZNIKI.....	114

WSTĘP

Zapewnienie odpowiednich warunków mieszkaniowych dla siebie oraz najbliższej rodziny zawsze było, jest i będzie jednym z kluczowych priorytetów jednostek na całym świecie. Jednak w ciągu ostatnich dziesięcioleci rozwinięte gospodarki, a szczególnie metropole oraz inne istotne pod względem ekonomicznym lub turystycznym obszary, borykają się z problemem nierównowagi między cenami nieruchomości, a poziomem dochodów. Przekłada się to na narastający kryzys na rynku nieruchomości.

Okoliczności te zdeterminowane są przez określone wydarzenia oraz czynniki, które zostaną szczegółowo przeanalizowane w niniejszej pracy magisterskiej. Należy zwrócić szczególną uwagę na fakt, iż część z owych determinant jest uniwersalna i dotyczy wszystkich metropolii, niezależnie od lokalizacji. Ze względu na historyczne wydarzenia, charakterystykę miasta oraz specyfikę Wielkiej Brytanii, część czynników kształtujących rynek nieruchomości odnosi się wyłącznie do Londynu. W obliczu kryzysu niezbędne staje się podejmowanie świadomych decyzji, które pozwalają zaspokajać możliwie największą liczbę potrzeb, przy ograniczonym budżecie. Umiejętność podejmowania świadomych decyzji, uwarunkowana jest posiadaniem wiedzy na temat rynku nieruchomości. W zdobyciu owej wiedzy pomocny okazać się może stworzony na potrzeby niniejszej pracy magisterskiej dashboard analityczny, który zawiera między innymi informacje na temat czynników kształtujących rynek nieruchomości w Londynie.

Celem pracy jest uwypuklenie czynników kształtujących kryzys mieszkaniowy w Londynie oraz prezentacja determinant i trendów za pomocą stworzonego na ten cel dashboardu analitycznego. Zostały zbadane poniższe zależności:

- **korelacja między ceną nieruchomości, a ceną najmu,**
- **korelacja między ceną nieruchomości, a liczbą lat pracy potrzebnych do jej zakupu,**
- **korelacja między liczbą miejsc pracy, a wysokością średniego wynagrodzenia,**
- **korelacja między odległością od dworca głównego, a wysokością średniego wynagrodzenia,**

- **korelacja między odlegością od pałacu Buckingham, a liczbą przestępstw,**
- **korelacja między odlegością od dworca głównego, a gęstością zaludnienia.**

Metoda badawcza zastosowana w trzecim i czwartym rozdziale pracy magisterskiej opiera się na wykorzystaniu odpowiednich języków programowania oraz programów do zbierania, przekształcania, modelowania, oczyszczania i wizualizowania danych. Użyte technologie to m.in. SQL Server, Power BI, Power Query, DAX, Python (wraz z biblioteką Pandas), QGIS, Gretl oraz Excel.

Pierwszy podrozdział rozdziału pierwszego przedstawia teoretyczne podstawy pracy magisterskiej. Omawia on cechy gospodarki oraz historię Londynu, która określa jego charakter i umożliwia głębsze zrozumienie sytuacji tego miasta. Kolejne dwa podrozdziały rozdziału pierwszego skupiają się na czynnikach kształtujących rynek nieruchomości bezpośrednio w Londynie oraz ogólnie w metropoliach.

Rozdział metodologiczny koncentruje się na określeniu problemu badawczego oraz opisie metod i technologii zastosowanych w celu jego rozwiązania. Jest on stosunkowo krótki, jednak istotny ze względu na znaczenie zrozumienia ról, jakie pełnią języki programowania i programy (*software*) w różnych aspektach pracy.

Zbiór danych oraz mapy przygotowane w oparciu o różne źródła są częścią rozdziału trzeciego. Rozdział ten opisuje również proces transformacji danych, w którym wystąpiły wyzwania, takie jak sprowadzanie liczby dzielnic do tej samej liczby pomiędzy źródłami danych, czy ujednolicanie znaków diakrytycznych. Proces przetwarzania danych zawierał również agregację i generowanie brakujących obserwacji na podstawie trendów w już istniejących danych. Generowanie danych pozwoliło podtrzymać maksymalną rozpiętość czasową dostępnych obserwacji dotyczących kryzysu na rynku nieruchomości. Natomiast grupowanie danych umożliwiło poprawną komunikację między źródłami danych za pośrednictwem kalendarza.

Czwarty rozdział dotyczy bezpośrednio środowisk Power BI, Power Query oraz DAX. Rozdział ten podzielony jest na podrozdziały, z których każdy reprezentuje oddzielną kartę dashboardu. Podrozdziały zawierają procesy budowy miar i kolumn kalkulacyjnych oraz objaśnienia wizualizacji zawartych w kartach. Rozdział ten opisuje również rozwiązania wykraczające poza użycie narzędzi business intelligence. W zależności od potrzeb, w poszczególnych kartach opisywany jest kod SQL, Python lub interpretacje pochodzące z oprogramowania GRETL. Znaczącym wsparciem, szczególnie w podrozdziale dotyczącym miernika syntetycznego, okazał się program Excel.

Wiele procesów oraz kodów, które umożliwiły eksplorację danych nie zostało zawartych w poniższej pracy magisterskiej z racji na pośredni wpływ. Praca zawiera głównie takie rozwiązania, które bezpośrednio przełożyły się na efekt końcowy. Należy również podkreślić, iż nie wszystkie rozwiązania o bezpośrednim przełożeniu zostały zawarte w pracy. Powtarzalne elementy rozwiązań pomijano z racji na zbliżony do istniejących rozwiązań schemat. Poniższa praca magisterska zawiera kluczowe rozwiązania oraz narrację umożliwiającą zrozumienie całego procesu budowy dashboardu analitycznego oraz problemu jakim jest kryzys na rynku nieruchomości.

ROZDZIAŁ 1. CHARAKTERYSTYKA GOSPODARKI LONDYNU ORAZ RYNKU NIERUCHOMOŚCI

Tematyka teoretycznej części pracy magisterskiej koncentruje się na charakterystyce rynku nieruchomości w Londynie. W pierwszej kolejności przedstawiona zostaje historia rozwoju miasta, co umożliwia głębsze zrozumienie pozycji City of London oraz metropolii londyńskiej. Kolejne części rozdziału teoretycznego poświęcone są analizie determinantów kształtujących rynek nieruchomości ogółem oraz tych, które wpływają bezpośrednio na rynek Londynu.

1.1 Historia rozwoju oraz charakterystyka stolicy Wielkiej Brytanii

Początki historii Londynu sięgają czasów Cesarstwa Rzymskiego. W roku 43 przed naszą erą powstało tak zwane „Londinium”, powszechnie znane jako City of London. W okresie panowania Rzymian miasto pełniło funkcję głównego portu oraz stanowiło najważniejsze centrum handlowe Brytanii. Spuścizna Cesarstwa Rzymskiego umożliwiła City of London ustabilizowanie swojej potęgi gospodarczej jak i strategicznej w regionie oraz dała podwaliny pod utworzenie oraz utrzymanie najstarszej, nieprzerwanie funkcjonującej demokracji świata¹. ‘The Corporation of London’ obwieściła się jako niezależna gmina w 1191 roku. Połączenie bogactwa, funkcjonujących instytucji demokratycznych oraz prawnych czy skutecznych systemów policyjnych wewnętrz ‘City’ stało się równoznaczne z brakiem możliwości podporządkowania sobie owego regionu przez monarchów panujących na wyspach. City samo o sobie stanowiło, opodatkowywało, sądziło oraz rządziło.

Na przestrzeni wieków populacja Londynu znacząco się rozwinęła. Jego prosperująca gospodarka i pozycja w regionie przyciągnęły wielu migrantów czy mieszkańców wsi i okolicznych miast. Szacuje się, że już w roku 1625 miasto zamieszkiwało 400 000 osób, czyli 20 krotnie więcej niż w jakiejkolwiek innej miejscowości na wyspach w tamtym okresie. City of London pozostawał odrębny w swoich strukturach wobec przedmieść, które stanowiły miejsce zamieszkania dla większości z okolicznej ludności. Spowodowało to w roku 1632 zwrócenie się przez monarchów do korporacji z prośbą o rozszerzenie swojego funkcjonowania łącznie

¹ *The City of London's Strange History*, 2014, [w:] ft.com, <https://www.ft.com/content/7c8f24fa-3aa5-11e4-bd08-00144feabdc0> [dostęp 22.05.2024].

z przywilejami i zachowaniem instytucji na przyległe tereny. Zamiast jednak rozszerzenia demokratycznych wpływów i zapewniania legalnej ochrony dla nowych obywateli przedmieść Westminster, Southwark, Clerkenwell i Whitechapel, City of London odmówiło uznania Londynu jako jego integralnej części. Wydarzenie te mające miejsce w roku 1637 nazwane ‘wielką odmową’ trwale zdefiniowało kształt i charakter miasta podzielonego na Londyn metropolitarny oraz City of London będący odrębnym podmiotem demokratycznym zachowującym swoją charakterystykę po dziś dzień.

City of London stał się głównym centrum finansowym świata krótko po roku 1795, kiedy to Republika Holandii upadła pod naciskiem armii Napoleona. Wydarzenie to spowodowało, że wielu bankierów pochodzących z Amsterdamu będącego jednym z najważniejszych centrów finansowych tamtych czasów przeniosło się do Londynu. Pomocne w rozwoju potęgi finansowej miasta stało się zorientowanie systemu na rynek, a nie na bank, jak miało to miejsce w Amsterdamie. Londyńska elita finansowa umocniła się dzięki rozwiniętej żydowskiej mniejszości kulturowej, która przyczyniła się w znaczącym stopniu do opanowania i doskonalenia najbardziej z wyrafinowanych na tamte czasy instrumentów finansowych².

Obecnie City of London pozostał odrębną demokracją i uznawany jest potocznie za 33 dzielnicę Londynu, którą w rzeczywistości nie jest. Korporacja posiada Guildhall, który pełni rolę administracyjną i ceremonialną oraz Mansion House będący siedzibą wybieranego corocznie burmistrza City. Nawet policja w regionie City of London różni się od tej metropolitarnej i jest zarządzana bezpośrednio przez korporację. Specyficzna, a przede wszystkim korzystna z punktu widzenia biznesu charakterystyka tego regionu, wraz z silnym potencjałem gospodarczym przyciągnęła wielu inwestorów. Oczywistym następstwem rozwoju City, był trwały rozwój Londynu metropolitarnego, który również stał się siedzibą dla licznych biznesów oraz centrów finansowych. Wyróżnić należy przede wszystkim takie centra biznesowe jak: Westminster, Camden & Islington, Canary Wharf (będącym jednym z dwóch głównych centrów finansowych), Lambeth & Southwark oraz oczywiście City of London.

Londyn jest siedzibą licznych instytucji oraz przedsiębiorstw. The London Stock Exchange, będący największym rynkiem wymiany papierów wartościowych w Europie, ma swoją siedzibę w Londynie, podobnie jak połowa z 100 największych notowanych przedsiębiorstw na tej giełdzie. Londyn pełni funkcję głównej siedziby dla ponad 100

² O. Coispeau, *Finance Masters: A Brief History of International Financial Centers in the Last Millennium*, 2017, s. 44.

spośród 500 największych przedsiębiorstw w Europie. Ponad 70% przedsiębiorstw z FTSE 100 oraz 75% z Fortune 500 posiada biura w metropolitarnej części Londynu. Według badań przeprowadzonych na zlecenie Deloitte, Londyn jest najbardziej zróżnicowanym międzynarodowo środowiskiem na świecie, przyciągającym liderów biznesu z 95 krajów oraz absolwentów z 134 krajów³.

Londyn wypracował swoją pozycję na rynku jako potentat finansowy oraz gospodarka o znacznym udziale usług, co pozwoliło osiągnąć poziom 22% ogólnego udziału PKB w UK⁴. Główne czynniki, które pozwoliły osiągnąć Londynowi status opartej o usługach gospodarki, z rozwiniętymi centrami biznesowymi przed pozostałymi europejskimi miastami to:

- Język angielski funkcjonujący w Londynie jest językiem ojczystym Wielkiej Brytanii, Stanów Zjednoczonych oraz kolonii Brytyjskich,
- Pełnienie roli stolicy Brytanii w czasie panowania Rzymian,
- Pozycja miasta w Europie, która jako kontynent posiada większą od amerykańskiej wielkość populacji⁵, a co za tym idzie rynek,
- Centralna strefa czasowa pozwalająca Londynowi być pomostem pomiędzy rynkami Azji i Ameryki⁶,
- Środowisko sprzyjające rozwojowi biznesu w szczególności City of London, gdzie wyboru rządu nie dokonuje większość mieszkańców, a członkowie biznesu rezydujący w tym obszarze. City of London jest biznesową demokracją,
- Niskie podatki dla korporacji,
- English contract law będącym najważniejszym i najczęściej używanym z praw w świecie międzynarodowego biznesu⁷,

³ *London is the Soft Power and High Skills Capital of the World*, 2016, [w:] deloitte.com, <https://www2.deloitte.com/uk/en/pages/press-releases/articles/london-soft-power-and-high-skills-capital.html> [dostęp 22.05.2024].

⁴ *Regional Gross Value Added (Income Approach) NUTS3 Tables*, 2014, [w:] web.archive.org, <https://webarchive.nationalarchives.gov.uk/ukgwa/20160105160709/http://www.ons.gov.uk/ons/rel/regional-accounts/regional-gross-value-added-income-approach-december-2013/rft-nuts3.xls> [dostęp 22.05.2024].

⁵ *European Movement Uk: The Economic Benefits to the Uk of Eu Membership*, [w:] web.archive.org, <https://web.archive.org/web/20150629121205/http://www.euromove.org.uk/index.php> [dostęp 22.05.2024].

⁶ J. Werdiger, *London Wants to Tap Chinese Currency Market*, 2012, [w:] nytimes.com, <https://archive.nytimes.com/dealbook.nytimes.com/2012/01/16/london-wants-to-tap-chinese-currency-market/> [dostęp 22.05.2024].

⁷ *English Common Law is the most widespread legal system in the world*, 2008, [w:] sweetandmaxwell.co.uk, <https://www.sweetandmaxwell.co.uk/about-us/press-releases/061108.pdf> [dostęp 22.05.2024].

- Rozwinięta infrastruktura lotnicza (przykładowo lotnisko Heathrow),
- Wysoka jakość życia.

Obecnie ponad 85% osób zatrudnionych w Londynie pracuje w usługach. Ciekawostką jest, że City of London posiadające powierzchnie 2.9 km², zawiera 7.74km² powierzchni biurowej, czyli ponad 2,5 krotność powierzchni samej dzielnicy! To ukazuje jak wysoka oraz gęsta zabudowa znajduje się w City oraz jak rozwinięty i preżny gospodarczo jest ten obszar.

1.2 Charakterystyka rynku nieruchomości w metropoliach

Rynek nieruchomości kształtuje wiele uniwersalnych czynników, które w znaczącym stopniu determinują ostateczną cenę na rynkach pierwotnych oraz wtórnego. Jednym z czynników będących determinantą cen nieruchomości jest gęstość zaludnienia. Im wyższa gęstość, tym droższe nieruchomości⁸. Zasada ta dotyczy wielu aglomeracji na świecie. Oczywiście istnieją odstępstwa od reguły, gdy centrum traci atrakcyjność z racji na zbyt gęstą zabudowę i ogólne zaniedbanie, a przedmieścia zaczynają stanowić atrakcyjną alternatywę. Zjawisko to występuje w indywidualnych przypadkach, jak przykładowo w mieście Łódź i to dla odosobnionych osiedli. Rodzi się jednak pytanie, dlaczego wysoka gęstość zaludnienia oraz ogólnej populacji w metropoliach, determinuje kształtowanie się cen na wysokim poziomie w sektorze nieruchomości?

Rozsądne planowanie przestrzenne, wykorzystujące gęstą oraz wysoką zabudowę, umożliwia maksymalizację zagospodarowania ograniczonych gruntów w metropoliach. Tak zaplanowane miasto umożliwia zapewnienie odpowiedniego dostępu do usług, sklepów czy komunikacji publicznej. Miasta niewykorzystujące planowania przestrzennego lub korzystające z niego w sposób niedostateczny, często mierzą się z przesadną gęstością zabudowy w centrum miasta lub niedostatecznym zagęszczeniem zabudowy na przedmieściach, czy w skrajnych dzielnicach. Mieszkańcy zamieszkujący dzielnice o niskim zagęszczeniu ludności, często borykają się z problemami komunikacyjnymi czy ograniczonym dostępem do usług oraz z przymusem wykorzystywania własnego środka transportu⁹. Zakładając, iż zasięg

⁸ D. Cvijanovic, *Real Estate Finance: How Demographics Drive Housing Prices*, 2012, [w:] hec.edu, <https://www.hec.edu/en/real-estate-finance-how-demographics-drive-housing-prices> [dostęp 22.05.2024].

⁹ D.Z.W. Wang, *Financial sustainability of rail transit service: The effect of urban development*, 2016, s. 3.

oddziaływania przystanku kolejowej wynosi do 500 metrów w linii prostej od przystanku, można zaznaczyć na mapie koło o takim promieniu. W przypadku odpowiednio zaplanowanej przestrzeni, takie koło oddziaływania zamieszuje około 10 tysięcy ludzi w zabudowie miejskiej. Jednakże, gdy zabudowa nie jest wystarczająco gęsta, liczba ta może ograniczyć się już tylko do 5 tysięcy osób. Można więc założyć, że w przypadku podwójnego zagęszczenia zabudowy w okolicy przystanku kolejowego kolejowej, korzystne może być uruchomienie czterech składów kolejowych na godzinę zamiast dwóch. Zmiana ta bezpośrednio przełożyłaby się na poprawę komfortu mieszkańców regularnie dojeżdżających do centrów miast lub metropolii. Należy pamiętać, że zagęszczanie zabudowy stanowi powszechnie zjawisko w metropoliach.

Podstawową problematyką związaną z zagęszczaniem zabudowy, jest ograniczenie dostępności terenów zielonych, co negatywnie wpływa na jakość życia. Znalezienie równowagi pomiędzy zrównoważonym zagęszczaniem zabudowy umożliwiającym rozwój miasta oraz jego przedmieście, a zachowaniem dostatecznej powierzchni terenów zielonych staje się kluczowe. Gęstość zaludnienia oraz obecność terenów zielonych w pobliżu nieruchomości są istotnie dodatnio skorelowane z ceną¹⁰. Obserwacja ta prowadzi do konkluzji, iż mieszkańcy chcą korzystać z dobrodziejstw miasta, jednocześnie zachowując kontakt z naturą. Połączenie owych wyzwań stanowi wyzwanie w planowaniu przestrzennym.

Posiadanie ponadprzeciętnej powierzchniowo nieruchomości w mieście, takiej jak przykładowo wolno stojący dom jednorodzinny, powinno zostać przywilejem oraz swego rodzaju luksusem pozostającym w zasięgu nielicznych, aniżeli standardem. Niekorzyści płynące z rozrzedzania zabudowy oddziałują na wszystkich mieszkańców. Wolnostojąca zabudowa powinna być elementem zagospodarowania przestrzennego głównie na terenach wiejskich, czy w miastach satelickich oraz w odpowiedniej proporcji na przedmieściach. Doskonałym przykładem dobrego rozwoju przestrzennego miasta w Polsce są Tychy. Czasy PRL-u ukazały, iż możliwym staje się zaplanowanie przestrzennie miasta w taki sposób, aby mieszkańcy posiadaли zrównoważony dostęp zarówno do przychodni, szkół, komunikacji miejskiej czy

¹⁰ L. Jong-Won, L. Sang-Woo, K. Hai Gyong, J. Hyun-Kil, P. Se-Rin, *Green Space and Apartment Prices: Exploring the Effects of the Green Space Ratio and Visual Greenery*, 2023, s.4.

różnorodnych usług (w tym gastronomicznych) z racji na kompaktowy oraz gęsty charakter miasta jak i terenów zielonych¹¹.

Czynniki determinujące ceny nieruchomości, są swego rodzaju naczyniami połączonymi, które wzajemnie na siebie oddziałują. Występowanie terenów zielonych w najbliższym sąsiedztwie bezpośrednio wpływa na wyższe ceny nieruchomości. Gęsta zabudowa nie oddziałuje w sposób bezpośredni na cenę, tak jak robi to przykładowo zielone drzewo przy bloku. Sama w sobie gęsta zabudowa jest zjawiskiem negatywnym z perspektywy mieszkańca, a jednak skorelowana jest dodatnio z cenami. Wynika to z faktu, iż z gęstością zabudowy skorelowane są między innymi takie zmienne jak liczba miejsc pracy, czy średnie wynagrodzenie. To właśnie wzrost dostępności pobliskich usług i komunikacji wynikający z zagęszczania zabudowy determinuje popyt, a co za tym idzie – wzrost cen nieruchomości.

Określenie kluczowych aspektów powiązanych z decyzjami inwestycyjnymi w przestrzeń biurową, magazynową lub produkcyjną stanowi wyzwanie, ze względu na mnogość czynników zależnych od indywidualnych preferencji inwestorów i firm. Istnieją jednak uniwersalne determinanty takie jak:

- infrastruktura drogowa, kolejowa,
- dostępność kapitału ludzkiego,
- obecność innych przedsiębiorstw oraz korporacji,
- wymogi prawne, regulacyjne, specjalne strefy ekonomiczne,
- potencjał dzielnicy,
- prestiż dzielnicy wpływający na wizerunek.

Atrakcyjność biznesowa, na którą wpływają w/w czynniki, zachęca inwestorów do lokowania kapitału w metropoliach. Inwestycje przyciągają przedsiębiorstwa, a te z kolei zatrudniają specjalistów. Miasto rozwija się, a dzięki wzrostowi liczby miejsc pracy w sektorze prywatnym, rośnie również zapotrzebowanie na specjalistów w sektorze państwowym czy usługowym. Zawody takie jak policjant, nauczyciel, fryzjer, lekarz i tym podobne istnieją w Londynie dzięki innym profesjom i są zależne od działalności innych branż. Proces ten skutkuje koncentracją perspektyw zawodowych w obrębie aglomeracji, jednocześnie determinując, poprzez konkurencyjność rynkową, wzrost wynagrodzeń, co z kolei przyczynia się do nasilenia migracji do metropolii.

¹¹ J. Dudek-Klimiuk, *History of Green Areas of Tychy Their Origins and Role in the Structure of the City*, 2016, s.7.

Niestety wyższe wynagrodzenia determinują również wyższe ceny nieruchomości¹². Taki stan rzeczy skłania pracowników do mieszkania w bardziej przestępnych cenowo dzielnicach oraz dojeżdżania do biur. Zwiększyły się popyt na mieszkania w okolicach centrów biznesowych i centrach metropolitarnych przekłada się na wzrost cen gruntów. Konsekwencją tego zjawiska jest maksymalizacja inwestycji na tych gruntach, napędzana dążeniem do wykorzystania przestrzeni w zmaksymalizowany sposób. W rezultacie dochodzi do zagęszczania zabudowy w centrach miast, gdzie ceny gruntów osiągają znaczące wartości. Przesadna gęstość zabudowy oraz często ekstremalnie wysokie ceny nieruchomości wymuszają na znacznej części pracowników mieszkanie w zewnętrznych dzielnicach miast lub przedmieściach.

Dobrze zaplanowane i skomunikowane przedmieścia oraz miasta satelickie, stają się substytutem rozwiązań na ogólnie rozwinięty kryzys mieszkaniowy na świecie¹³. Przedmieścia jednak powinny być budowane w sposób zrównoważony pod względem gęstości zaludnienia tak, aby nie „przytłoczyć” mieszkańców, a jednocześnie sprawić, aby wszelkie usługi czy komunikacja stały się rentowne. Jednym z rozwiązań jest urozmaicanie rodzajów zabudowy w przemyślany sposób. Stosowanie zabudowy szeregowej, wielorodzinnej, usługowo-mieszkaniowej, a na samym końcu wolnostojących domów jednorodzinnych umożliwia stworzenie odpowiednich warunków do inwestycji infrastrukturalnych, czy kreacji lokalnych biznesów.

Sukces miast satelickich zależy od dostępnej infrastruktury umożliwiającej komunikację z większym miastem. Jednym z najlepszych rozwiązań tego typu jest kolej, która jednocześnie emitując stosunkowo mało CO₂ do atmosfery jest w stanie przetransportować znaczącą liczbę osób, zazwyczaj do samego centrum metropolii. Zakup nieruchomości na cele mieszkaniowe w mieście satelickim, uwzględniając wszystkie czynniki, staje się interesującą alternatywą, w szczególności w obliczu coraz popularniejszej pracy hybrydowej i rosnących cen nieruchomości w metropoliach na całym świecie.

W obszarach metropolitalnych, charakteryzujących się wysoką gęstością zaludnienia, często obserwuje się wzrost wskaźników przestępcości. Jednakże, paradoksalnie, przestępstwa te rzadko mają znaczący wpływ na ceny nieruchomości w tych regionach. Jedyne kategorie przestępstw, które wydają się wywierać istotny wpływ na wartość nieruchomości w metropoliach, to napaści i rabunki. Pomimo

¹² D. Kwon, O. Sorenson, *The Silicon Valley Syndrome*, 2021, s.3.

¹³ V. Kumar Nirmal, *Satellite Cities: the Only Hope of Megacities A Case of Indian Scenario*, 2015, s.4.

zwiększych zagrożeń związanych z przestępcością w środowiskach miejskich, ceny nieruchomości pozostają stosunkowo odporne na te czynniki¹⁴.

1.3 Determinanty kształtowania się cen nieruchomości w Londynie

Poza uniwersalnymi czynnikami, które kształtują ceny nieruchomości na całym świecie, istnieją również determinanty wpływające na ceny w określonych aglomeracjach, krajach czy regionach. Uniwersalne czynniki kształtuje ceny na rynku nieruchomości często dotyczą lokalnych aspektów, jak przykładowo bliskości przystanku tramwajowego, drzewa rosnącego za oknem, czy drogi szybkiego ruchu przed mieszkaniem. Narracja prowadzona wokół determinant sektora mieszkaniowego w Londynie orbitować będzie w obrębie czynników lokalnych, lecz w kontekście obejmującym mechanizmy większe niż aspekty lokalne. W przypadku Londynu uwaga zostanie poświęcona czynnikom zewnętrznym wynikającym z lokalizacji owej aglomeracji w Wielkiej Brytanii, jak również wewnętrznym determinantom, takim jak obecność centrów finansowych.

Zaprezentowaną ustawę mieszkaniową pod nazwą ‘Right to Buy’ w roku 1980 przez Margaret Thatcher, ówczesnego premiera Wielkiej Brytanii, uznaje się za jedną z głównych przyczyn aktualnego kryzysu mieszkaniowego w Londynie, jak i na wyspach. Ustawa wprowadziła możliwość wykupu mieszkań socjalnych przez wynajmujących za połowę rynkowej kwoty. Badania pokazują, iż przynajmniej 36% mieszkań sprzedanych w ten sposób jest aktualnie wynajmowanych przez prywatnych właścicieli¹⁵. Program faktycznie początkowo umożliwił wielu rodzinom posiadanie własnego mieszkania, co przyczyniło się do poprawy jakości ich życia, jednak długoterminowo okazał się być nieodpowiednim wykorzystaniem środków finansowych obywateli jak i rządu. Pozbawiając się mieszkań socjalnych po zaniżonych cenach rząd nie dość, że utracił znaczącą ilość pieniędzy, to dodatkowo uniemożliwił czerpanie zysków z najmu z tych nieruchomości. Zyski te umożliwiły budowanie większej liczby mieszkań socjalnych, przy braku zaangażowania środków z innych sfer budżetowych. Jednocześnie Wielka Brytania utraciła w dużym stopniu kontrolę nad rynkiem najmu, pozostawiając ją w rękach prywatnych inwestorów oraz wolnego rynku, który bezlitośnie

¹⁴ M. Maximino, *The Impact of Crime on Property Values: Research Roundup*, 2014, [w:] journalistsresource.org, <https://journalistsresource.org/economics/the-impact-of-crime-on-property-values-research-roundup/> [dostęp 22.05.2024].

¹⁵ T. Copley, *From Right to Buy to Buy to Let*, 2014, s.2.

drenuje po dziś dzień kieszenie Londyńczyków oraz pozostałych mieszkańców wysp wysokimi cenami najmu oraz zakupu nieruchomości.

Okres wprowadzenia ustawy mieszkaniowej jest zbieżny z początkiem rozwoju finansjalizacji w Wielkiej Brytanii. Charakteryzuje się ona nieproporcjonalnym wzrostem udziału usług finansowych w PKB oraz transformacją gospodarki przemysłowej w gospodarkę usługową. Zwiększenie liczby nieruchomości w rękach prywatnych będące rezultatem wprowadzenia ustawy ‘Right to Buy’, spotęgowało użycie mieszkań jako formy instrumentu finansowego. Prawo do spekulacji cenami nieruchomości stało się istotniejsze, aniżeli prawo do posiadania przyzwoitego domu czy mieszkania przez przeciętnego mieszkańca wysp. Finansjalizacja doprowadziła do znaczającej niestabilności finansowej gospodarstw domowych, nierówności społecznych oraz nierównych możliwości rozwoju¹⁶. System skonstruowany w ten sposób umożliwia czerpanie jeszcze większych zysków przez już wzbogaconych, jednocześnie uniemożliwiając posiadanie własnego mieszkania czy domu dla pozostałych, poprzez przymus mierzenia się z drogimi cenami najmu oraz zakupu nieruchomości. Problemem nawet nie staje się brak możliwości zbudowania oszczędności przy wysokich cenach najmu, lecz możliwości opłacenia najmu w ogóle. Sama wysokość najmów dla części społeczeństwa staje się nieosiągalna finansowo, co skutkuje wzrostem zakresu biedy oraz bezdomności. Problem pogłębił się również z racji zwiększającej się dostępności kredytów w tamtym okresie. Ceny nieruchomości dostosowały się do zdolności kredytowej mieszkańców, która była i jest znaczaco wyższa niż ich możliwości oszczędzania.

Postępujący proces finansjalizacji, nie tylko w Wielkiej Brytanii, ale i w całej rozwiniętej gospodarczo części świata, wpłynął w znaczącym stopniu na powstanie wielu centrów finansowych w Londynie. Przyczyniła się do tego, pozycja Londynu na arenie międzynarodowej oraz indywidualny charakter City of London. Efektem owego procesu jest stały wzrost zapotrzebowania na specjalistów z branży finansowej oraz powstanie znaczącej liczby nowych miejsc pracy. Położone głównie w City of London oraz Canary Wharf, ale i również w Islington, Camden, Southwark oraz Lambeth centra finansowe, przyciągnęły do pracy znaczną liczbę, dobrze zarabiających pracowników. Skutkiem wysokich zarobków w branży finansowej, a także jej obecności w poszczególnych dzielnicach, są wzrosty cen nieruchomości w dzielnicach będących siedzibami

¹⁶ G. Blakeley, *Financialization, Real Estate and COVID-19 in the UK*, 2020, s.5.

korporacji. Rynek zaczął dostosowywać się do zdolności kredytowej osób pracujących w prosperujących branżach, pozostawiając daleko w tyle pozostałe grupy zawodowe. Biznes przyciągnął za sobą w sposób pośredni wiele nowych miejsc pracy, jednak nie tak dobrze płatnych jak te korporacyjne. Rezultatem stała się więc obecność znaczającej liczby pracowników w Londynie, którzy nie są w stanie pozwolić sobie, ani na wyśrubowane ceny najmu, ani na zakup nieruchomości za cenę, która jest zależna od zarobków najlepiej zarabiającej grupy.

Londyn jest unikalnym miastem w skali Wielkiej Brytanii, Europy oraz świata. Unikalność tego miasta wraz z niepowtarzalnym charakterem, możliwościami oraz znaczącą spuścizną historyczną sprawia, iż miasto staje się częstym wyborem urlopowym wielu turystów. Londyn również chce utrzymać pozycję jednego z najlepszych miast do odwiedzenia na świecie¹⁷. Nie powinno to dziwić z racji na fakt, iż turystyka w takiej skali odgrywa znaczącą rolę w gospodarce miasta. Pomimo jednak czerpania przez Londyn niezliczonych korzyści z turystyki, miasto mierzy się z wieloma problemami z niej wynikającymi. Wewnętrzne dzielnice Londynu, będące częścią tak zwanego ‘Inner London’, borykają się z przeludnieniem oraz wysokim poziomem przestępcości, wynikającym w znaczącym stopniu z wysokiej liczby turystów. Idealnym zobrazowaniem owej problematyki jest dzielnica Westminster, która na skutek wzmożonej turystyki, obecności najmu krótkoterminowego oraz wysokich cen nieruchomości, rok rocznie przoduje pod względem liczby przestępstw - powyższych obserwacji można dokonać przy pomocy dashboardu stworzonego na potrzeby poniższej pracy magisterskiej. Jednak kluczowym i jednocześnie globalnym problemem wynikającym z znaczącej liczby turystów, jest popularność najmu krótkoterminowego dostępnego za pomocą platform takich jak Airbnb, czy Booking. Niejednokrotnie zostało udowodnione, iż portale takie jak Airbnb zarabiają na nielegalnym najmie, który powoduje wzrosty cen najmów długoterminowych w okolicy, redukcje dostępności mieszkań na wynajem dla mieszkańców oraz pogłębia segregacje¹⁸. Krótkoterminowy najem w dzielnicach atrakcyjnych turystycznie umożliwia czerpanie obfitych korzyści finansowych przez inwestorów, aniżeli najem długoterminowy tych samych nieruchomości. Powoduje to, że ceny najmu długoterminowego w atrakcyjnych

¹⁷ C. Maxim, *Challenges Faced by World Tourism Cities – London’s Perspective*, 2017, s.2.

¹⁸ D. Lee, *How Airbnb Short-Term Rentals Exacerbate Los Angeles’s Affordable Housing Crisis: Analysis and Policy Recommendations*, 2016, s.3.

dzielnicach potrafią osiągnąć poziom niedostępny nawet dla najlepiej zarabiających grup społecznych Londynu.

Nieruchomości pełnią rolę instrumentów finansowych, których cenę rynkową determinuje między innymi możliwość osiągnięcia oczekiwanej stopy zwrotu przez inwestorów w określonym czasie. Najem krótkoterminowy umożliwia osiągnięcie wyższej stopy zwrotu niż najem długoterminowy, co bezpośrednio przekłada się na wzrost rynkowych cen nieruchomości w lokalizacjach atrakcyjnych turystycznie. Konsekwencją zależności cen nieruchomości w Londynie od spekulacji, wynagrodzeń najbardziej zamożnych grup społecznych oraz możliwości finansowych turystów jest trudność zamieszkania w Londynie przez osoby nie będące specjalistami lub nie posiadające własnych nieruchomości.

Londyn składa się z części metropolitalnej oraz „dzielnicy” City of London, która charakteryzuje się występowaniem odrębnego od londyńskiego systemu demokratycznego. Taki układ administracyjny stanowi podstawę istnienia wielu czynników kształtujących rynek nieruchomości w Londynie. Istotny wpływ na rynek wywarło wprowadzenie prawa „Right to Buy” oraz rozwój finansjalizacji na terenie Wielkiej Brytanii. Specyficzne dla Londynu i całego kraju determinanty, w połączeniu z powszechnymi mechanizmami działającymi na rynkach nieruchomości, kształtują sektor mieszkaniowy w stolicy.

ROZDZIAŁ 2. OKREŚLENIE PROBLEMU BADAWCZEGO

Część metodologiczna koncentruje się na przedstawieniu problemu badawczego oraz metod jego eksploracji w kontekście rynku nieruchomości Londynu. Omówiona jest również zasadność prowadzenia badań, a także rola i funkcja narzędzi wykorzystanych w tym celu.

2.1 Problem badawczy

Światowe metropole od lat dotknięte są wszelkiego rodzaju kryzysami na rynkach nieruchomości. Problem potęguje fakt, iż w aglomeracjach występuje ograniczona podaż mieszkań, która spotyka się z wzmożonym popytem ze strony konsumentów. Popyt z podażą spotykają się na znaczącym pułapie cenowym, często odbiegającym kilkukrotnie od standardów przyjętych w innych miejscowościach krajów będących siedzibą owych metropolii.

Niestety niedobory podażowe oraz zawyżony popyt umożliwiają dokonywanie spekulacji cenami nieruchomości, co skutkuje nieadekwatnym ich wzrostem. Proces ten podsycany jest przez kapitał pochodzący z bogaczej części rozwartwiającego się społeczeństwa. Zjawisko to skłania do rezygnacji z mieszkania w metropoliach i poszukiwania alternatyw przez biedniejszą, z rozwartwionych części społeczeństwa. Problem rynkowy pozostaje niezmienny; jednak zdobycie informacji oraz zrozumienie faktycznej sytuacji rynkowej może dostarczyć wiedzy umożliwiającej zidentyfikowanie optymalnych rozwiązań dla indywidualnych jednostek mierzących się z kryzysem mieszkaniowym. Problemem badawczym jest więc **uwypuklenie czynników będących determinantami cen na rynku nieruchomości wraz z wskazaniem stopnia ich oddziaływania rynkowego na przykładzie Londynu**. Zrozumienie pozwoli interesariuszom podejmować adekwatne oraz świadome decyzje na podstawie zgromadzonej w ten sposób wiedzy rynkowej.

2.2 Metoda eksploracji problemu badawczego i uzasadnienie jej doboru

Wielowymiarowość oraz wielowątkowość narracji powiązanej z tematyką problemu badawczego stały się czynnikami determinującymi budowę dashboardu analitycznego. Wykorzystanie dashboardu jako formy prezentacji czynników

ksztaltujących rynek umożliwiło przekazanie informacji w sposób klarowny, zwięzły oraz dostosowany do różnorodnych odbiorców. Przedstawienie złożonej i spójnej narracji stało się prostsze dzięki podzieleniu dashboardu na poszczególne karty zawierające odrębną tematykę. Kolorowe wykresy oraz mapa Londynu umożliwiają eksplorację osobistych preferencji użytkowników. Wizualizacje powstały w środowisku Power BI z zastosowaniem kolumn kalkulacyjnych oraz miar stworzonych w języku DAX.

Niezbędne do konstrukcji dashboardu analitycznego są dane, które na potrzeby pracy magisterskiej zostały pozyskane z różnorodnych źródeł. Dane mają na celu wsparcie rozwiązania postawionego problemu badawczego. Źródła danych wykazywały pewne deficyty, takie jak brakujące informacje dla wybranych lat oraz zróżnicowaną liczbę dzielnic w Londynie pomiędzy źródłami. W celu skutecznego radzenia sobie z tymi wyzwaniami, zastosowano język SQL w środowisku RDBMS Microsoftu, znanego jako SQL Server. Poprzez wykorzystanie SQL-a opracowano szereg zapytań, które umożliwiły przeprowadzenie wstępnej analizy dotyczącej jakości baz danych, identyfikację brakujących danych oraz ocenę istotności zawartych w nich informacji z perspektywy pracy magisterskiej. W ramach tych działań, wykorzystano logikę CTE, funkcje okien oraz inne zaawansowane techniki SQL, które umożliwiły wygenerowanie brakujących danych na podstawie istniejących już obserwacji w bazie. Język SQL okazał się koniecznym narzędziem w procesie oczyszczania danych, manipulacji znakami diakrytycznymi, grupowania oraz agregacji danych zgodnie z określonymi oczekiwaniemi. Język SQL stanowi kluczowy element poniższej pracy poprzez umożliwienie przygotowania danych, modelowania oraz eksportu do środowiska Power BI.

Kluczową rolę w obszarze narzędzi Business Intelligence pełni oprogramowanie Power BI. Jest to narzędzie, które cieszy się uznaniem w branży, a jego wszechstronność jest wzmacniona przez możliwość korzystania z języka Data Analysis Expression (DAX) oraz Power Query M (PQM). Power Query jest narzędziem ETL, służącym do ekstrakcji, transformacji oraz ładowania danych. Wykorzystanie PQM okazało się niezwykle pomocne w importie danych z różnych źródeł, takich jak baza danych SQL, czy plików płaskich w formatach CSV oraz XLSX. Dodatkowo, PQM posłużyło do oczyszczania danych, zmiany typów danych, usuwania brakujących danych, edycji informacji związanych z datą oraz w wielu innych czynnościach.

Z kolei, język DAX odgrywał istotną rolę w konstrukcji dashboardu. Za jego pomocą powstały kolumny kalkulatoryne oraz miary, które umożliwiły kreację zaawansowanych wizualizacji oraz prezentację wyników analiz. Bez zastosowania języka DAX, stworzenie większości wizualizacji w dashboardzie byłoby niemożliwe. Środowisko Power BI pozwoliło dokonywać edycji modelu danych, określać nowe relacje bądź usuwać niepotrzebne, czy pracować nad estetyką i ogólnie pojętym designem. Właściwości wbudowane w Power BI okazały się użyteczne przy projektowaniu, budowie menu oraz w innych aspektach, które przełożyły się na ostateczny efekt.

Jednym z podstawowych elementów każdej z kart dashboardu analitycznego jest mapa Londynu. Jest to mapa kształtów zapisana w rozszerzeniu SHP pochodząca z roku 2018 z oficjalnego źródła. Niezbędny w celu poprawnego wyświetlenia mapy w Power BI okazał się program QGIS. Program ten umożliwił eksplorację mapy, a informacje zawarte w mapie pozyskane za pomocą programu QGIS stanowiły wyznacznik podziału dzielnicowego dla różnych źródeł danych, które zawierały odmienną liczbę dzielnic. Dzięki temu oprogramowaniu udało się również rozwiązać problem z renderingiem, a następnie przy pomocy programu MapShaper udało się wygładzić granice mapy i wyeksportować mapę w rozszerzeniu JSON obsługiwany przez Power BI.

Część danych wykorzystanych w dashboardzie powstała na podstawie pozostałych już istniejących obserwacji, przy pomocy kolumn kalkulatorynych języka DAX lub uzupełnienia braków odpowiednim zapytaniem SQL. Jednak w jednej z tabel zaistniała potrzeba uzyskania wartości odległości, pomiędzy współrzędnymi geograficznymi. Najlepszym z rozwiązań okazało się użycie języka Python z bibliotekami Harvesine oraz Pandas.

Odległości bazujące na współrzędnych geograficznych powstały między innymi w celu stworzenia korelacji pomiędzy zmiennymi. Weryfikacja poprawności owych korelacji wymagała wykonania testów normalności rozkładu, homoskedastyczności lub heteroskedastyczności, determinacji lub indeterminacji, istotności statystycznej i tym podobnych. Do tego celu posłużyło oprogramowanie GRETl. Wnioski z tego programu zostały opisane wewnątrz dashboardu pod liniami korelacji.

Nieoceniony wkład w pracę magisterską i budowę dashboardu miał program Microsoft Excel. To właśnie w tym programie powstał miernik syntetyczny, który zebrał wiele z determinant rynku nieruchomości w całość. Program ten również przydał się do wprowadzania ręcznych zmian, które nie były warte automatyzowania lub stworzenia

rozwiązań w języku SQL. Excel wspierał wiele z procesów analitycznych umożliwiając przejrzysty wgląd w informacje na każdym etapie budowy dashboardu analitycznego.

Niezbędnymi narzędziami do eksploracji problemu badawczego okazały się być: SQL Server, Excel, Power BI, Data Analysis Expressions (DAX), Power Query M, Python wraz z bibliotekami, oprogramowanie GRETL oraz QGIS. Zastosowanie owych narzędzi umożliwiło uwypuklenie czynników będących determinantami cen na rynku nieruchomości w Londynie.

ROZDZIAŁ 3. OCZYSZCZANIE, PRZETWARZANIE ORAZ PRZYGOTOWYWANIE DANYCH Z UŻYCIEM SQL

Niezbędnym elementem budowy dashboardu analitycznego jest przetwarzanie danych. W przypadku pracy magisterskiej skupia się ono na połączeniu danych z różnych źródeł, ich oczyszczaniu, budowie modelu oraz konfiguracji mapy. Poniższa część pracy zawiera gotowe rozwiązania w języku SQL oraz oprogramowaniu QGIS.

3.1 Zbiór danych z różnych źródeł oraz przygotowanie mapy przy pomocy QGIS

Do stworzenia dashboardu analitycznego potrzebne są dane. Istotna jest nie tylko ilość danych, która umożliwia tworzenie wielowymiarowych analiz, ale również ich jakość. Na potrzeby poniższej pracy magisterskiej zostały wykorzystane dane z różnych źródeł tak, aby maksymalnie wyczerpać możliwość eksploracji problemu badawczego i uwypuklić czynniki na niego wpływające.

Pierwszym zbiorem danych, który został wykorzystany w analizach jest zbiór z strony kaggle dotyczący rynku mieszkaniowego w Londynie¹⁹. Zbiór zawiera wiele istotnych informacji takich jak średnie ceny mieszkań na przestrzeni 20 lat, czy średnie wynagrodzenie w poszczególnych dzielnicach. Dane z powyższego zbioru charakteryzują się wysoką przejrzystością oraz zwierają nieocenioną wartość dodaną w postaci wielu już zagregowanych informacji przydatnych w analizach.

Drugim zbiorem danych jest zbiór pochodzący z strony data.london.gov.uk dotyczący liczby przestępstw w Londynie na przestrzeni 8 lat²⁰. Przestępstwa w bazie nie są zagregowane, dlatego też należy ową agregację wykonać. Następstwem braku grupowania tego źródła danych jest znacząca liczba obserwacji (ponad 13 milionów). Dane podzielone są na kategorie i podkategorie, które zliczone są wewnątrz poszczególnych miesięcy oraz podkategorii jednocześnie. Baza wymaga dopasowania do pozostałych danych w umiarkowanym stopniu, jednak pochodzi z wiarygodnego źródła i stanowi wysoką wartość dodaną w analizach.

¹⁹ J. Cirtautas, *Housing in London*, 2020, [w:] kaggle.com, <https://www.kaggle.com/datasets/justinas/housing-in-london> [dostęp 22.05.2024].

²⁰ London Crime Data, [w:] data.london.gov.uk, <https://data.london.gov.uk/> [dostęp 22.05.2024].

Trzecim i ostatnim głównym źródłem danych jest baza zawierająca obserwacje z najmu krótkoterminowego z platformy Airbnb. Źródło danych pochodzi z webscrapingu portalu i zostało udostępnione również poprzez platformę kaggle²¹. Baza najmu jest bazą najbardziej zanieczyszczoną. Zawiera wiele kolumn zbędnych z punktu widzenia budowy dashboardu oraz obserwacje, które Londynu nie dotyczą. Przykładem takich obserwacji są mieszkania na wynajem w Berlinie czy Madrycie.

Ostatnim równie istotnym źródłem danych do poniższej pracy magisterskiej jest Londyńska mapa kształtów. Pochodzi ona z strony data.london.gov.uk²² i zawiera ostatnie aktualizację uwzględniającą zmiany granic dzielnicowych w Londynie. Ostatnia edycja dotyczyła zmian w dzielnicach Bexley, Croydon, Redbridge oraz Southwark i miała miejsce 3 maja 2018 roku. Plik z tego dnia został zastosowany w pracy magisterskiej.

Dane w pierwszej kolejności zostały przetworzone za pomocą RDBMS stworzonego przez przedsiębiorstwo Microsoft o nazwie SQL Server z użyciem odpowiedniego dialekta języka SQL. Nastąpiło pobranie danych z powyżej wymienionych źródeł w postaci plików płaskich CSV, a następnie import owych plików do środowiska SQL Server. Powyżej wymieniony proces wymagał zmiany języka oprogramowania systemu Windows 11 na angielski (US) w celu odczytywania plików płaskich. Poza tym proces przebiegł bez przeszkód z wyjątkiem bazy najmu.

Podczas wielokrotnych prób wgrania bazy Airbnb do środowiska SQL Server następowały liczne problemy, które ustąpiły po zastosowaniu niżej wymienionych kroków:

- zmiany typu danych na 1252 (latino),
- usunięcie obserwacji niedotyczących Londynu,
- zapisanie pliku płaskiego bez rozszerzenia UTF-8, a następnie import do SQL Server.

Ostatnim krokiem przygotowania źródeł było sprawdzenie działania mapy kształtów. Niestety wskutek importu shapemap do środowiska Power BI napotkany został problem z jej renderingiem.

²¹ J. Arvidsson, *Airbnb Global Listings*, 2023, [w:] kaggle.com, <https://www.kaggle.com/datasets/joebeachcapital/airbnb/data> [dostęp 22.05.2024].

²² *Statistical GIS Boundary Files for London*, 2018, [w:] data.london.gov.uk, <https://data.london.gov.uk/dataset/statistical-gis-boundary-files-london> [dostęp 22.05.2024].



Rysunek 1. Problem z renderingiem mapy

Źródło: Microsoft Power BI

W rozwiązaniu problemu pomogło GISowskie oprogramowanie do edycji map. Na potrzeby pracy magisterskiej użyty został program QGIS. Problem renderingu generuje układ odniesienia, który w Power BI jest inny niż w plikach od mapy kształtów. Układ odniesienia, czyli tak zwany CRS (Coordinate Reference System) definiuje sposób, w jaki dane przestrzenne są odwzorowywane na płaszczyźnie lub w przestrzeni trójwymiarowej. Spójnym z odczytem Power BI układem odniesienia jest EPSG:4326 – WGS 84, który należy ustawić w środowisku QGIS dla mapy Londynu²³. Następnie w opcjach warstwy konieczne jest skonfigurowanie możliwości zmiany rozmiaru, umożliwiając mapie przyjęcie dowolnych wymiarów. w takiej konfiguracji plik zostaje nadpisany.

Jednak dalej nie rozwiązuje to w pełni problemu renderingu z racji na brak obsługiwanej przez Power BI rozszerzenia mapy. Następnie komplet nadpisanych plików o rozszerzeniach cpg, dbf, prj, qmd, shp oraz shx zostaje wgrany do programu służącego konwersji zwanego mapshaper. Zostanie zastosowana również funkcjonalność wygładzania granic mapy, która umożliwi poprawę jej czytelności. Po dokonaniu tego procesu następuje eksport pliku do rozszerzenia TopoJSON, a następnie import do środowiska Power BI. Powyższym sposobem udaje się osiągnąć zamierzony efekt.

²³ D. Eldersveld, *Overcoming Potential Power Bi Shape Map Rendering Issues - Dataveld*, 2016, [w:] dataveld.com, <https://dataveld.com/2016/09/01/overcoming-potential-power-bi-shape-map-rendering-issues/> [dostęp 22.05.2024].



Rysunek 2. Naprawiona mapa za pomocą QGIS

Źródło: Oprogramowanie QGIS

3.2 Oczyszczanie danych oraz modelowanie

Oczyszczanie danych oraz tworzenie odpowiednich relacji między tabelami stanowi fundamentalny etap w procesie prowadzenia analiz. Poza koniecznością dokładnej edycji danych, takiej jak przykładowo eliminacja znaków diakrytycznych i redukcja szumu, istotną rolę odgrywa transformacja danych w sposób przystosowany do celu analizy. W kontekście omawianej pracy magisterskiej, istnieje nieoczywisty aspekt procesu dopasowania danych, a mianowicie weryfikacja liczby dzielnic oraz ich nazw między różnymi tabelami. Taki scenariusz ukazuje istotność wyznaczenia z góry założonych celów w trakcie procesu oczyszczania oraz modelowania danych. Dzięki temu możliwe jest podjęcie odpowiednich kroków wcześniej oraz zaoszczędzenia znacznej ilości czasu.

Następstwem importu plików płaskich do środowiska SQL Server jest powstanie czterech tabel znajdujących się na serwerze lokalnym. Tabele te na początkowym etapie przetwarzania danych zawierały następujące robocze nazwy:

- dbo.housing_in_london_yearly_variables,
- dbo.housing_in_london_monthly_variables,
- dbo.london_crime_by_lsoa,
- dbo.AirbnbONLYneedLONDON.

Ostatnia z tabel zawiera specyfczną nazwę z racji na jej charakter. Jest to tabela, która wywodzi się z źródła danych najmu krótkoterminowego jednak usunięte zostały z niej obserwacje spoza obszaru Londynu adekwatnym kodem SQL. W toku przekształceń wkluczono również szereg kolumn, które nie były istotne z punktu widzenia analizy oraz mogły opóźnić proces obliczeń²⁴. Poddane usunięciu zostały kolumny takie jak: Listing Url, ScrapeID, Experiences Offered, Neighborhood Overview, Notes, Transit, Access, Interaction, House Rules, ThumbnailUrl, Medium Url, Picture Url, XL Picture Url, Host URL, Host About, Host Acceptance Rate, Host Thumbnail Url, Host Picture Url i inne. Podczas próby konwersji typów danych w owej tabeli pojawił się problem, który spowodowany był przestawionymi danymi w części komórek.

Market	Country	Latitude	Longitude	Property Type
London	United Kingdom	51.51780907	-0.126325271	Apartment
London	United Kingdom	51.50322967	-0.082720052	Apartment
London	United Kingdom	51.50094077	-0.081015828	Apartment
London	United Kingdom	51.49954891	-0.080798045	Apartment
London	United Kingdom	51.4243695	-0.113378277	Apartment
London	United Kingdom	51.43433796	-0.05519697	House
England	GB"	London	United Kingdom	51.33600425
St. Johns Wood	NW8 London"	London	United Kingdom	51.54223146
England	GB"	London	United Kingdom	51.45035368
England	GB"	London	United Kingdom	51.4484496
England	GB"	London	United Kingdom	51.6150061

Rysunek 3. Przesunięcie danych pomiędzy kolumnami

Źródło: SQL Server

Z racji na nieznaczną liczbę obserwacji objętych tą problematyką (102 w stosunku do 52257 obserwacji istniejących w tamtym czasie), zapadła decyzja o usunięciu przesuniętych wierszy następującym kodem.

```
DELETE FROM dbo.AirbnbONLYneedLONDON
WHERE Market <> 'London' ;
```

Rysunek 4. Usunięcie przesuniętych danych

Źródło: SQL Server

²⁴ R. Sherman, *Business Intelligence Guidebook: From Data Integration to Analytics*, 2014, s. 292.

Po dokonaniu wstępnej eksploracji danych można zauważyc znaczące różnice w liczbie dzielnic pomiędzy tabelami. Budowa wizualizacji z użyciem mapy kształtów wymaga ujednolicenia liczby dzielnic i ich nazewnictwa na takie, odpowiadające tym zawartym w mapie. Spis dzielnicowy zawierający 33 dzielnice zgodny z mapą znajdująca się będzie w tabeli dbo.GIS_London_Polygons_Names. Jest to jedyna poprawna liczba zgodna z zmianami granic z dnia 3 maja 2018 roku z uwzględnieniem ‘City of London’ jako dzielnicę Londynu. Spis ten w następnych etapach pracy stanowić będzie centralną tabelę modelu danych.

	NAME	GSS_CODE
1	Barking and Dagenham	E09000002
2	Barnet	E09000003
3	Bexley	E09000004
4	Brent	E09000005
5	Bromley	E09000006
6	Camden	E09000007
7	City of London	E09000001
8	Croydon	E09000008
9	Ealing	E09000009
10	Enfield	E09000010

Rysunek 5. Przykładowy spis

dzielnic z kodami

Źródło: SQL Server

Na tym etapie analizy za klucz posłużą nazwy dzielnic, które są jedynym punktem styczności pomiędzy tabelami. W następnych krokach rolę kluczy głównych i obcych pełnił kod dzielnic, przypisany do każdego obszaru w jednolity sposób.

Posiadając punkt odniesienia w postaci spisu dzielnicowego można zastosować logikę zagnieżdżenia podzapytania w zapytaniu SQL w taki sposób, aby w rezultacie otrzymać tylko takie dzielnice, które w owym spisie się nie znajdują²⁵. Logika ta została zastosowana na tabeli dbo.housing_in_london_yearly_variables poprzez poniższe zapytanie.

²⁵ K. Kellenberger, S. Shaw, *Beginning T-SQL - Subsection: Using a Subquery Containing NULL with NOT IN*, 2014, s. 129.

```

SELECT DISTINCT area FROM dbo.housing_in_london_yearly_variables
WHERE area NOT IN (
    SELECT G.NAME
    FROM dbo.GIS_London_Polygons_Names AS G
    WHERE G.NAME IS NOT NULL
);

```

Rysunek 6. Zapytanie wyszukujące obszary w bazie dbo. housing yearly nie znajdujące się w tabeli centralnej

Źródło: SQL Server

Kod ten wykazał istnienie 18 obszarów, które są odmienne od tych zawartych w mapie kształtów.

	area
1	east
2	east midlands
3	england
4	england and wales
5	great britain
6	inner london
7	london
8	north east
9	north west
10	northern ireland
11	outer london
12	scotland
13	south east
14	south west
15	united kingdom
16	wales
17	west midlands
18	yorkshire and the humber

Rysunek 7. Obszary w tabeli dbo. housing yearly odmienne od istniejących w punkcie odniesienia

Źródło: SQL Server

Obszary te dotyczą wysp brytyjskich jako całych lub odnoszą się do podziału dzielnicowego na poziomie hierarchii, wyżej niż ta, która jest obiektem zainteresowania z punktu budowy dashboardu. Obszary takie jak Inner czy Outer London zawierają w sobie inne dzielnice, więc agregują informacje z nich zawarte. Zgodnie z przyjętymi odgórnymi założeniami, obszary te podlegają usunięciu, co zostaje wyegzekwowane poniższym kodem.

```

DELETE FROM dbo.housing_in_london_yearly_variables
WHERE area NOT IN (
    SELECT G.NAME
    FROM dbo.GIS_London_Polygons_Names AS G
    WHERE G.NAME IS NOT NULL
);


```

Rysunek 8. Usunięcie zbędnych obszarów z dbo. housing yearly

Źródło: SQL Server

Analogiczny proces z wykorzystaniem zagnieźdżenia podzapytania SQL należy zastosować do pozostałych tabel. Tabela dbo.housing_in_london_monthly_variables zawiera 12 obszarów niezgodnych z spisem.

	area
1	north east
2	yorks and the humber
3	north west
4	south east
5	london
6	england
7	outer london
8	south west
9	west midlands
10	east of england
11	east midlands
12	inner london

Rysunek 9. Obszary odmienne od punktu odniesienia w tabeli dbo. housing monthly

Źródło: SQL Server

Obszary nie zawierające się wewnętrz mapy kształtów, podobnie jak w poprzedniej tabeli, dotyczą obszarów wykraczających poza podział dzielnicowy Londynu i dotyczą głównie Anglii. Obserwacje te należy poddać usunięciu co zostało przeprowadzone za pomocą poniższego kodu.

```
DELETE FROM dbo.housing_in_london_monthly_variables
WHERE area NOT IN (
    SELECT G.NAME
    FROM dbo.GIS_London_Polygons_Names AS G
    WHERE G.NAME IS NOT NULL
);
```

**Rysunek 10. Usunięcie zbędnych obszarów z dbo.
housing monthly**

Źródło: SQL Server

Tabela dbo.AirbnbONLYneedLondon zawiera znaczną liczbę obserwacji, które w przeciwieństwie do pozostałych źródeł danych, nie są pogrupowane w żaden sposób. w przypadku, gdy każda z obserwacji stanowi niezależną od siebie informację, niezgodne z sztuką analityczną byłoby nie wzięcie pod uwagę obserwacji występujących regularnie pod nazwą dzielnic, których nazwy mogą różnić się od tych zawartych w spisie. Szczególne znaczenie ma to w przypadku, gdy nazwa dzielnic stanowi na tym etapie analiz klucz, więc owe obszary zostałyby pominięte w dalszych działaniach. Potrzeba więc stworzyć zapytanie, które pogrupuje obserwacje względem indywidualnych nazw obszarów, a następnie uwypukli liczebność tych grup²⁶. Powyższy proces można wykonać następującym zapytaniem.

```
SELECT COUNT (Price) AS Policzone,
"Neighbourhood Cleansed"
FROM dbo.AirbnbONLYneedLONDON
GROUP BY "Neighbourhood Cleansed"
ORDER BY Policzone;
```

Rysunek 11. Zapytanie grupujące obserwacje według dzielnic w tabeli dbo. Airbnb

Źródło: SQL Server

Rezultatem takiego zapytania są dwie kolumny. Jedna z nich zawiera nazwę obszaru, a druga liczbę obserwacji, która w danym obszarze występuje. Łączna liczba grup, a co za tym idzie indywidualnych obszarów istniejących w bazie wynosi 41.

²⁶ K. Kellenberger, S. Shaw, *op.cit.*, s. 150.

	Policzone	Neighbourhood Cleansed
1	1	Llucmajor
2	1	East Village
3	1	Whitehorse
4	1	Reuilly
5	1	Ménilmontant
6	1	Berruguete
7	3	
8	3	United Kingdom
9	93	Havering
10	110	Bexley
11	136	Barking and Dagenham
12	139	Sutton
13	252	Harrow
14	265	Hillingdon
15	294	Enfield
16	301	Kingston upon Thames

Rysunek 12. Obszary pogrupowane od najmniej licznych w tabeli dbo. Airbnb

Źródło: SQL Server

Zapytanie uwzględniało sortowanie grup od najmniej do najbardziej licznej. Grupy zawierające pomiędzy jedną, a trzema obserwacjami można wykluczyć i uznać takie obszary za nieistotne z punktu widzenia analizy.

```
DELETE FROM dbo.AirbnbONLYneedLONDON
WHERE [Neighbourhood Cleansed] NOT IN (
    SELECT G.NAME
    FROM dbo.GIS_London_Polygons_Names AS G
    WHERE G.NAME IS NOT NULL
);
```

Rysunek 13. Usunięcie nieistotnych obszarów z tabeli dbo. Airbnb

Źródło: SQL Server

Uwzględniając różnicę pomiędzy liczebnością grup przed i po usunięciu tych z małą liczbą obserwacji, otrzymujemy 33 dzielnice. Można więc przyjąć wstępne założenie, że wszystkie z nich są zgodne z 33 dzielnicami zawartymi w spisie.

Tabela zawierająca liczbę przestępstw z podziałem na kategorie i podkategorie zawiera tylko 33 dzielnice, co zostało zweryfikowane odpowiednim zapytaniem w języku SQL. Liczba dzielnic po zastosowaniu kodów SQL jest zgodna pomiędzy wszystkimi czterema tabelami. W celu weryfikacji poprawności nazw obszarów należy przyrównać

je pomiędzy sobą oraz ze spisem dzielnic z mapy kształtów. Sposób ten potwierdzi zgodność dzielnicową pomiędzy wszystkimi źródłami danych. Do przeprowadzenia tego procesu potrzebne jest użycie kwerendy z wszystkich źródeł w jednym zapytaniu SQL. Do osiągnięcia celu przyda się zastosowanie komendy JOIN²⁷.

```
SELECT DISTINCT M.area, Y.area, G.NAME, L.borough, A.[Neighbourhood Cleansed]
FROM dbo.housing_in_london_monthly_variables M
JOIN dbo.GIS_London_Polygons_Names G ON M.area = G.NAME
JOIN dbo.housing_in_london_yearly_variables Y ON G.NAME = Y.area
JOIN dbo.london_crime_by_lsoa L ON G.NAME = L.borough
JOIN dbo.AirbnbONLYneedLONDON A ON G.NAME = A.[Neighbourhood Cleansed]
ORDER BY M.area, Y.area, G.Name, L.borough ASC;
```

Rysunek 14. Zapytanie umożliwiające wyświetlenie wszystkich wspólnych dzielnic z tabel

Źródło: SQL Server

Efektem powyższego zapytania jest pięć kolumn: po jednej z każdego źródła danych oraz jedna z spisu dzielnic, który stanowi punkt odniesienia.

²⁷ K. Kellenberger, S. Shaw, *op.cit.*, s. 113.

	area	area	NAME	borough	Neighbourhood Cleansed
1	barking and dagenham				
2	barnet	barnet	Barnet	Barnet	Barnet
3	bexley	bexley	Bexley	Bexley	Bexley
4	brent	brent	Brent	Brent	Brent
5	bromley	bromley	Bromley	Bromley	Bromley
6	camden	camden	Camden	Camden	Camden
7	city of london				
8	croydon	croydon	Croydon	Croydon	Croydon
9	ealing	ealing	Ealing	Ealing	Ealing
10	enfield	enfield	Enfield	Enfield	Enfield
11	greenwich	greenwich	Greenwich	Greenwich	Greenwich
12	hackney	hackney	Hackney	Hackney	Hackney
13	hammersmith and fulham				
14	haringey	haringey	Haringey	Haringey	Haringey
15	harrow	harrow	Harrow	Harrow	Harrow
16	havering	havering	Havering	Havering	Havering
17	hillingdon	hillingdon	Hillingdon	Hillingdon	Hillingdon
18	hounslow	hounslow	Hounslow	Hounslow	Hounslow
19	islington	islington	Islington	Islington	Islington
20	kensington and chelsea				
21	kingston upon thames				
22	lambeth	lambeth	Lambeth	Lambeth	Lambeth
23	lewisham	lewisham	Lewisham	Lewisham	Lewisham
24	merton	merton	Merton	Merton	Merton
25	newham	newham	Newham	Newham	Newham
26	redbridge	redbridge	Redbridge	Redbridge	Redbridge
27	richmond upon thames				
28	southwark	southwark	Southwark	Southwark	Southwark
29	sutton	sutton	Sutton	Sutton	Sutton
30	tower hamlets				
31	waltham forest				
32	wandsworth	wandsworth	Wandsworth	Wandsworth	Wandsworth
33	westminster	westminster	Westminster	Westminster	Westminster

Rysunek 15. Wspólne dzielnice pomiędzy źródłami danych (33)

Źródło: SQL Server

Rezultat spełnił założone oczekiwania. Wszystkie źródła danych zawierają ujednoliconą liczbę dzielnic o tym samym nazewnictwie. Nazwy dzielnic jednak nie powinny być kluczami w modelu danych, dlatego należy zweryfikować, czy wszystkie źródła danych zawierają kod dzielnicowy, który będzie pełnił tę funkcję. Odpowiednim zapytaniem w języku SQL, również wykorzystującym komendę JOIN, zostają połączone kolumny z kodami przy użyciu nazw dzielnic.

	code	code	GSS_CODE	lsoa_code	Neighbourhood Cleansed
20...	E12000008	E09000012	E09000012	E01001770	Hackney
20...	E12000008	E09000012	E09000012	E01001782	Hackney
20...	E12000008	E09000012	E09000012	E01001756	Hackney
20...	E12000008	E09000012	E09000012	E01001826	Hackney
20...	E12000008	E09000012	E09000012	E01001803	Hackney
20...	E12000008	E09000012	E09000012	E01001789	Hackney
20...	E12000008	E09000012	E09000012	E01001825	Hackney
20...	E12000008	E09000012	E09000012	E01033701	Hackney
20...	E12000008	E09000012	E09000012	E01001757	Hackney
20...	E12000008	E09000012	E09000012	E01001812	Hackney
20...	E12000008	E09000012	E09000012	E01001763	Hackney
20...	E09000012	E09000012	E09000012	E01001769	Hackney
20...	E09000012	E09000012	E09000012	E01001790	Hackney
20...	E09000012	E09000012	E09000012	E01001732	Hackney
20...	E09000012	E09000012	E09000012	E01001839	Hackney

Rysunek 16. Niepasujące do siebie kody dzielnicowe

Źródło: SQL Server

Niestety, kody dzielnicowe nie wykazują spójności, a jedno z dostępnych źródeł danych w ogóle ich nie zawiera. W związku z tym, każdej tabeli zostanie przypisany kod dzielnicowy zgodny ze spisem dzielnic. Proces ten zostanie zrealizowany poprzez wykonanie poniższych zapytań dla wszystkich czterech źródeł danych.

```

SELECT A.ID, A.[Neighbourhood Cleansed], G.GSS_CODE, A.Latitude, A.Longitude, A.[Property Type],
       A.[Room Type], A.Accommodates, A.Bathrooms, A.Bedrooms, A.Beds, A.Price, A.[Review Scores Rating],
       A.[Reviews per Month]
INTO dbo.AirBnbGSS_Code
FROM dbo.AirbnbONLYneedLONDON A
JOIN dbo.GIS_London_Polygons_Names G ON G.NAME = A.[Neighbourhood Cleansed];

SELECT M.date, M.area, M.average_price, M.houses_sold, M.no_of_crimes,
       G.GSS_CODE
INTO dbo.housing_monthly_GSS_Code
FROM dbo.housing_in_london_monthly_variables M
JOIN dbo.GIS_London_Polygons_Names G ON G.NAME = M.area;

SELECT Y.area, G.GSS_CODE, Y.date, Y.median_salary, Y.life_satisfaction, Y.mean_salary, Y.recycling_pct,
       Y.population_size, Y.number_of_jobs, Y.area_size, Y.no_of_houses
INTO dbo.housing_yearly_GSS_Code
FROM dbo.housing_in_london_yearly_variables Y
JOIN dbo.GIS_London_Polygons_Names G ON G.NAME = Y.area;

SELECT C.borough, G.GSS_CODE, C.major_category,
       C.minor_category, C.value, C.year, C.month
INTO dbo.crime_GSS_Code
FROM dbo.london_crime_by_lsoa C
JOIN dbo.GIS_London_Polygons_Names G ON C.borough = G.NAME;

```

Rysunek 17. Kod SQL narzucający jednolity kod dzielnicowy wszystkim źródłom danych

Źródło: SQL Server

Proces przyniósł zamierzony efekt i umożliwił stworzenie wzajemnie spójnych kluczy, we wszystkich z istniejących źródeł danych. Od tego momentu spis dzielnicowy będzie centralną tabelą, a wszystkie z pozostałych źródeł stanowić będą odgałęzienia od tabeli centralnej. Źródła danych w momencie przejęcia kodu GSS za klucze przyjmują zmienioną nazwę z dopiskiem GSS_Code.

	NAME	GSS_CODE	GSS_CODE	GSS_CODE	GSS_CODE	GSS_CODE
1	Barking and Dagenham	E09000002	E09000002	E09000002	E09000002	E09000002
2	Barnet	E09000003	E09000003	E09000003	E09000003	E09000003
3	Bexley	E09000004	E09000004	E09000004	E09000004	E09000004
4	Brent	E09000005	E09000005	E09000005	E09000005	E09000005
5	Bromley	E09000006	E09000006	E09000006	E09000006	E09000006
6	Camden	E09000007	E09000007	E09000007	E09000007	E09000007
7	City of London	E09000001	E09000001	E09000001	E09000001	E09000001
8	Croydon	E09000008	E09000008	E09000008	E09000008	E09000008
9	Ealing	E09000009	E09000009	E09000009	E09000009	E09000009
10	Enfield	E09000010	E09000010	E09000010	E09000010	E09000010
11	Greenwich	E09000011	E09000011	E09000011	E09000011	E09000011
12	Hackney	E09000012	E09000012	E09000012	E09000012	E09000012
13	Hammersmith and Fulham	E09000013	E09000013	E09000013	E09000013	E09000013
14	Haringey	E09000014	E09000014	E09000014	E09000014	E09000014
15	Harrow	E09000015	E09000015	E09000015	E09000015	E09000015
16	Havering	E09000016	E09000016	E09000016	E09000016	E09000016
17	Hillingdon	E09000017	E09000017	E09000017	E09000017	E09000017
18	Hounslow	E09000018	E09000018	E09000018	E09000018	E09000018
19	Islington	E09000019	E09000019	E09000019	E09000019	E09000019
20	Kensington and Chelsea	E09000020	E09000020	E09000020	E09000020	E09000020
21	Kingston upon Thames	E09000021	E09000021	E09000021	E09000021	E09000021
22	Lambeth	E09000022	E09000022	E09000022	E09000022	E09000022
23	Lewisham	E09000023	E09000023	E09000023	E09000023	E09000023
24	Merton	E09000024	E09000024	E09000024	E09000024	E09000024
25	Newham	E09000025	E09000025	E09000025	E09000025	E09000025
26	Redbridge	E09000026	E09000026	E09000026	E09000026	E09000026
27	Richmond upon Thames	E09000027	E09000027	E09000027	E09000027	E09000027
28	Southwark	E09000028	E09000028	E09000028	E09000028	E09000028
29	Sutton	E09000029	E09000029	E09000029	E09000029	E09000029
30	Tower Hamlets	E09000030	E09000030	E09000030	E09000030	E09000030
31	Waltham Forest	E09000031	E09000031	E09000031	E09000031	E09000031
32	Wandsworth	E09000032	E09000032	E09000032	E09000032	E09000032
33	Westminster	E09000033	E09000033	E09000033	E09000033	E09000033

Rysunek 18. Wszystkie źródła danych zawierające spójny kod dzielnicowy - klucz główny tabel

Źródło: SQL Server

Obiektem zainteresowania inwestora czerpiącego zyski z najmu krótkoterminowego jest osiągnięcie możliwie jak najwyższej ceny oraz stopy zwrotu dla nieruchomości. Popularność nieruchomości na wynajem odgrywa znaczącą rolę, ponieważ na wolnym rynku to popyt określa jak wysoka cena jest tą, która będzie

akceptowalna przez turystę oraz umożliwi rezerwację mieszkania przez jak największą liczbę osób. Wiedza tego pokroju jest znaczącym źródłem informacji dla inwestora. Do stworzenia kolumny zawierającej popularność najmów zostanie wykorzystana liczba opinii zostawiona przez użytkowników portalu Airbnb. Tabela dbo.AirBnbGSS_Code, w kolumnie ‘Reviews per Month’, zawiera puste komórki, co uniemożliwia zmianę typu danych oraz przeprowadzenie dalszych działań. Język SQL obsługuje wartości NULL, czyli tak zwane puste wartości, które nie są 0. Wskutek wgrania danych z plików płaskich silnik nie wstawił wartości NULL tam, gdzie z założenia powinien, dlatego należy to wykonać ręcznie.

```
UPDATE dbo.AirBnbGSS_Code  
SET [Reviews per Month] = NULL  
WHERE [Reviews per Month] = '';
```

Rysunek 19. Zamiana pustych wartości na NULL

Źródło: SQL Server

Poza wartościami pustymi istnieją wartości błędne zawierające inne wartości niż liczbowe. Takie obserwacje również poddać należy zamianie na wartości NULL²⁸. Po dokonaniu tego procesu możliwa jest zmiana typu danych na liczbowy.

```
UPDATE dbo.AirBnbGSS_Code  
SET [Reviews per Month] = NULL  
WHERE TRY_CAST([Reviews per Month] AS FLOAT) IS NULL  
      AND [Reviews per Month] IS NOT NULL;
```

Rysunek 20. Przekształcenie danych nie liczbowych oraz niepustych wartości w NULL

Źródło: SQL Server

Zostało przyjęte założenie, że na 10 osób korzystających z usługi najmu krótkoterminowego, opinię na portalu Airbnb zostawia 1 osoba. Średnia wielkość przychodu dla poszczególnej nieruchomości będzie więc obliczona działaniem:

$$- \text{Ilość opinii} * \text{Cena} * 10$$

²⁸ A. Villazon, *Useful SQL Server functions: TRY_CAST & TRY_CONVERT*, 2020, [w:] andrewvillazon.com, <https://www.andrewvillazon.com/sql-server-try-cast-convert/> [dostęp 22.05.2024].

Została dodana kolumna do tabeli dbo.AirBnbGSS_Code o nazwie AverageRevenueMonth, która będzie zawierała ową wielkość zysku dla mieszkania.

```
--Dodaj nową kolumnę do tabeli
ALTER TABLE dbo.AirBnbGSS_Code
ADD AverageRevenueMonth FLOAT;

--Ustawienie wartości dla nowej kolumny na podstawie warunku:
UPDATE dbo.AirBnbGSS_Code
SET AverageRevenueMonth = [Reviews per Month] * 10 * Price
WHERE [Reviews per Month] > 0;
```

Rysunek 21. Stworzenie kolumny zawierającej dane dotyczące średniego zysku z najmu krótkoterminowego

Źródło: SQL Server

Dalsza eksploracja danych wykazała wiele brakujących informacji w tabeli dbo.housing_yearly_GSS_Code. Kolumna mean_salary, zawierająca pogrupowane latami informacje dotyczące średnich zarobków w poszczególnych dzielnicach, posiada 7 brakujących obserwacji.

area	GSS_CODE	date	median_salary	life_satisfaction	mean_salary
merton	E09000024	2018-12-01	31182	7.84	#
croydon	E09000008	2016-12-01	31479	7.68	#
bromley	E09000006	2006-12-01	25715		#
hackney	E09000012	2001-12-01	24095		#
haringey	E09000014	2000-12-01	19772		#
hackney	E09000012	2000-12-01	24083		#
merton	E09000024	2019-12-01	31699		#

Rysunek 22. Brakujące informacje dla 7 z obserwacji oznaczonych jako # w bazie dbo. housing yearly

Źródło: Microsoft Power BI

Niestety specyfika tego źródła danych nie pozwala na zignorowanie braków, ponieważ nie są to tylko pojedyncze obserwacje, a zagregowany efekt ze zbioru pojedynczych obserwacji. Potrzebne więc jest wygenerowanie tych informacji, ponieważ nie dysponujemy dostępem do danych przed agregacją. Wpierw należy zamienić znak '#' na wartość NULL, aby móc dowolnie manipulować typem danych w środowisku bazodanowym.

```
-- Zmiana znaków # na NULL
UPDATE dbo.housing_yearly_GSS_Code
SET mean_salary = NULL
WHERE mean_salary = '#';

-- Zmiana typu danych z VAR na INT
ALTER TABLE dbo.housing_yearly_GSS_Code
ALTER COLUMN mean_salary INT;
```

Rysunek 23. Zamiana typów danych oraz obserwacji zawierającej # na NULL

Źródło: SQL Server

Dalsze działania podzielone zostały na dwa etapy. Pierwszy etap dotyczyć będzie wstępnej eksploracji działania, a drugi jej przeprowadzenia w środowisku bazodanowym. Dane w tabeli dbo.housing_yearly_GSS_Code są zagregowane po dzielnicach oraz latach. Jeżeli przykładowo dla dzielnicy Bromley brakuje obserwacji dla roku 2006, to w celu uzupełnienia braku można wziąć pod uwagę wartości z lat 2005 i 2007 dla tej dzielnicy i wyliczyć średnią z tych dwóch wartości, a następnie wstawić w brakujące miejsce. Potrzebny jest kod SQL, który uwzględnia wszystkie brakujące obserwacje, a następnie uzupełni je średnią z wartości okalających je. Niezbędne do przeprowadzenia tego procesu będą funkcje LAG oraz LEAD, które mogą posłużyć do stworzenia dodatkowych kolumn zawierających wartości poprzedzające oraz następujące. W tym wypadku należy stworzyć jedną kolumnę z wartościami następującymi o jedną pozycję w przód oraz stępującymi o jedną pozycję w tył. Niezbędne do stworzenia takiego działania jest również użycie logiki okien w języku SQL, gdyż LAG i LEAD są funkcjami okiennymi i możliwość użycia ich istnieje tylko wewnętrz okna²⁹. Istotną funkcją okna jest również, posortowanie wartości wewnątrz po dzielnicach. Zobrazowanie pełni sytuacji i jednocześnie pierwszy etap budowy kodu SQL umożliwi poniższe działanie.

²⁹ K. Kellenberger, S. Shaw, *op.cit.*, s. 179.

```

SELECT area, year,
COALESCE(mean_salary, (LAG(mean_salary) OVER (ORDER BY area) +
LEAD(mean_salary) OVER (ORDER BY area)) / 2) AS mean_salary,
LAG(mean_salary) OVER (ORDER BY area) AS Preview,
LEAD(mean_salary) OVER (ORDER BY area) AS Future
FROM dbo.housing_yearly_GSS_Code;

```

Rysunek 24. Zapytanie umożliwiające znalezienie zesłorocznych oraz przyszłorocznych wartości dla pustych obserwacji z kolumny mean_salary

Źródło: SQL Server

	area	year	mean_salary	Preview	Future
91	bromley	2005	29955	27617	NULL
92	bromley	2006	32488	29955	35021
93	bromley	2007	35021	NULL	33378
94	bromley	2008	33378	35021	33329

Rysunek 25. Zobrazowanie funkcjonowania kodu SQL dla zesłorocznych i przyszłorocznych wartości

Źródło: Microsoft Power BI

Zapytanie ukazało, że kod działa poprawnie i uzupełnia brakujące informacje. Używając logiki CTE oraz funkcji UPDATE³⁰, zapytanie zostaje zmodyfikowane w taki sposób, aby nie tylko wyświetlało rozwiązańe, ale również wprowadzało owe wartości do tabeli dbo.housing_yearly_GSS_Code na stałe, tym samym kończąc etap drugi.

```

WITH Podzapytanie AS (
    SELECT mean_salary,
        LAG(mean_salary) OVER(ORDER BY area) AS Previous,
        LEAD(mean_salary) OVER(ORDER BY area) AS Future
    FROM dbo.housing_yearly_GSS_Code
)
UPDATE Podzapytanie
SET mean_salary = COALESCE(mean_salary, (Previous + Future) / 2);

```

Rysunek 26. Wprowadzenie średniej z okalających wartości w miejsce pustych obserwacji w kolumnie mean_salary

Źródło: SQL Server

³⁰ J. Zednick, *SQL CTE (Common Table Expressions) With Examples – More Organized Queries and Procedures*, 2020, [w:] janzednick.cz, <https://janzednick.cz/en/sql-cte-common-table-expressions-with-clause-more-organized-queries-and-procedures/> [dostęp 22.05.2024].

Niestety wskutek niedopatrzenia, kod nie uwzględnił dwóch pustych komórek, a mianowicie wartości dla dzielnic Hackney dla lat 2000 i 2001. Źródło problemu wynikało z bliskości obserwacji od siebie. Kod błędnie zakładał, że każda wartość NULL będzie posiadać obie z okalających wartości jednak jedną z tych obserwacji nie posiada górnej z nich, a druga tej dolnej. Na tym etapie najprostszym rozwiązaniem problemu będzie ręczne wstawienie do bazy odpowiednich wartości.

	area	GSS_CODE	date	median_salary	life_satisfaction	mean_salary	recycling_pct	population_size	number_of_jobs	area_size	no_of_houses	year
1	hackney	E09000012	1999-12-01	23249		39629	2	199087				1999
2	hackney	E09000012	2000-12-01	24083		NULL	1	203381	102000.0			2000
3	hackney	E09000012	2001-12-01	24095		NULL	1	207246	108000.0	1905	87208	2001
4	hackney	E09000012	2002-12-01	24582		32321	3	210961	108000.0	1905	88557	2002

Rysunek 27. Dwie w dalszym ciągu występujące obserwacje z powodu braku okalających dla nich wartości

Źródło: Microsoft Power BI

Między latami 2002 i 1999 istnieją 3 okresy, dlatego należy wykonać działanie, w którym różnica pomiędzy wartością z roku 2002 i 1999 zostaje podzielona przez 3 i wynik będzie reprezentował średnioroczną zmianę dla kolumny mean_salary. Różnica_mean_salary = (39629/32321) /3 = 2436 w okresie 1999 – 2002. Istnieje trend malejący, więc należy odjąć 39629 – 2436 = 37193 dla roku 2000 oraz 37193 – 2436 = 34757 dla roku 2001.

```

UPDATE dbo.housing_yearly_GSS_Code
SET mean_salary = 37193
WHERE area IN ('hackney') AND year = 2000;

UPDATE dbo.housing_yearly_GSS_Code
SET mean_salary = 34757
WHERE area IN ('hackney') AND year = 2001;

```

Rysunek 28. Ręcznie wprowadzone wartości w puste obserwacje w dzielnicy Hackney

Źródło: SQL Server

	area	GSS_CODE	date	median_salary	life_satisfaction	mean_salary	recycling_pct	population_size	number_of_jobs	area_size	no_of_houses	year
1	hackney	E09000012	1999-12-01	23249		39629	2	199087				1999
2	hackney	E09000012	2000-12-01	24083		37193	1	203381	102000.0			2000
3	hackney	E09000012	2001-12-01	24095		34757	1	207246	108000.0	1905	87208	2001
4	hackney	E09000012	2002-12-01	24582		32321	3	210961	108000.0	1905	88557	2002

Rysunek 29. Rezultat przekształceń w kolumnie mean_salary

Źródło: Microsoft Power BI

Kolejną kolumną dotkniętą brakiem danych z tabeli dbo. housing_ yearly_ GSS_ Code jest kolumna median_salary. Należy wykonać procedurę zmiany pustych wartości na wartości NULL, aby móc wykonać konwersję typu danych na integer. Konwersja typu danych jest konieczna z racji wykonywania obliczeń na kolumnie.

```
-- Zamieniamy puste wartości na NULL
UPDATE dbo.housing_yearly_GSS_Code
SET median_salary = NULL
WHERE median_salary = ' ';

-- Zmieniamy typ danych z VARCHAR na INTEGER
ALTER TABLE dbo.housing_yearly_GSS_Code
ALTER COLUMN median_salary INT;
```

Rysunek 30. Zamiana typów danych oraz pustych obserwacji na NULL

Źródło: SQL Server

Następnie w podobny sposób co w przypadku kolumny mean_salary należy napisać drugą część kodu, która wprowadzała zmiany w tabeli, a nie tylko wyświetlała. Edytując nazwę kolumn w poprzednim zapytaniu wprowadzamy kod bezpośrednio do RDBMS.

```
WITH Podzapytanie AS (
    SELECT median_salary,
           LAG(median_salary) OVER(ORDER BY area) AS Previous,
           LEAD(median_salary) OVER(ORDER BY area) AS Future
    FROM dbo.housing_yearly_GSS_Code
)
UPDATE Podzapytanie
SET median_salary = COALESCE(median_salary, (Previous + Future) / 2);
```

Rysunek 31. Wprowadzenie średniej z okalających wartości w miejsce pustych obserwacji w kolumnie median_salary

Źródło: SQL Server

Eksploracja danych umożliwiła zweryfikowanie, iż brakujących danych w tabeli dbo.housing_ yearly_ GSS_ Code jest wiele więcej. Na tym etapie dla wygody obcowania z danymi należy zamienić wszystkie puste wartości na NULL-e, aby móc swobodnie poruszać się pomiędzy odpowiednimi typami danych.

```
□ UPDATE dbo.housing_yearly_GSS_Code  
SET population_size = NULL  
WHERE population_size = ' ';  
  
□ UPDATE dbo.housing_yearly_GSS_Code  
SET number_of_jobs = NULL  
WHERE number_of_jobs = ' ';  
  
□ UPDATE dbo.housing_yearly_GSS_Code  
SET no_of_houses = NULL  
WHERE no_of_houses = ' ';  
  
□ UPDATE dbo.housing_yearly_GSS_Code  
SET recycling_pct = NULL  
WHERE recycling_pct = ' ';  
  
□ UPDATE dbo.housing_yearly_GSS_Code  
SET life_satisfaction = NULL  
WHERE life_satisfaction = ' ';
```

Rysunek 32. Seryjna zamiana pustych wartości na bazodanowy NULL

Źródło: SQL Server

Kolumna population_size w tabeli dbo.housing_yearly_GSS_Code posiada brakujące obserwacje dla roku 2019. W celu wygenerowania danych zostaną użyte obserwacje z lat 2018 i 2017, a różnica pomiędzy tymi latami zostanie dodana do roku 2018. Sposób ten umożliwia otrzymanie szacunkowej wielkości populacji dla poszczególnych dzielnic na rok 2019. Zmienna Population_2018 oraz Population_2017 zostaje wydobyta z bazy danych przy pomocy funkcji okna, logiki wielokrotnego CTE³¹ oraz funkcji LAG, użytej wewnątrz okna przesuniętego o jedną i dwie pozycje. Jedno z zapytań wydobywających dane z tabeli pozwala eksplorować problem oraz dobrą możliwie najlepsze rozwiązanie, z kolei drugie zapytanie wprowadza owo rozwiązanie do źródła danych.

³¹ T. Babic, *How to Write Multiple CTEs in SQL*, 2022, [w:] learnsq.com, <https://learnsq.com/blog/multiple-cte/> [dostęp 22.05.2024].

```

= ((Population_2018 - Population_2017) / Population_2018 + 1) * Population_2018

Population_2018
LAG(population_size) OVER (PARTITION BY area ORDER BY year ASC) AS Preview

Population_2017
LAG(population_size, 2) OVER (PARTITION BY area ORDER BY year ASC) AS Preview2

```

EKSPLORACJA PROBLEMU

```

;WITH virtual_table AS (
    SELECT area, year, population_size
    FROM dbo.housing_yearly_GSS_Code
    WHERE year IN (2017, 2018, 2019)
    GROUP BY area, year, population_size),
virtual_table2 AS (
    SELECT area, year, population_size,
    LAG(population_size) OVER
    (PARTITION BY area ORDER BY year ASC) AS Preview,
    LAG(population_size, 2) OVER
    (PARTITION BY area ORDER BY year ASC) AS Preview2
    FROM virtual_table)

SELECT area, year, population_size,
COALESCE(population_size,
((Preview - Preview2)/Preview + 1) *
Preview, 0) AS Kalkulacja
FROM virtual_table2
ORDER BY area, year ASC;

```

To jest tylko zapytanie które nie wprowadza zmian w bazie danych, lecz pozwala wyświetlić informacje oczekiwany sposób i umożliwić dokonanie analizy

ROZWIĄZANIE NA BAZIE DANYCH

```

;WITH virtual_table AS (
    SELECT area, year, population_size
    FROM dbo.housing_yearly_GSS_Code
    WHERE year IN (2017, 2018, 2019)
    GROUP BY area, year, population_size),
virtual_table2 AS (
    SELECT area, year, population_size,
    LAG(population_size) OVER
    (PARTITION BY area ORDER BY year ASC) AS Preview,
    LAG(population_size, 2) OVER
    (PARTITION BY area ORDER BY year ASC) AS Preview2
    FROM virtual_table),
virtual_table3 AS (
    SELECT area, year, population_size,
    COALESCE(population_size,
    ((Preview - Preview2)/Preview + 1) *
    Preview, 0) AS Kalkulacja
    FROM virtual_table2)

UPDATE dbo.housing_yearly_GSS_Code
SET population_size = vt3.Kalkulacja
FROM dbo.housing_yearly_GSS_Code h
JOIN virtual_table3 vt3 ON
h.area = vt3.area AND h.year = vt3.year
WHERE h.year = 2019;

```

	area	year	population_size	Kalkulacja
1	barking and dagenham	2017	210711	210711.000000
2	barking and dagenham	2018	211998	211998.000000
3	barking and dagenham	2019	NULL	213285.000000
4	barnet	2017	387803	387803.000000
5	barnet	2018	392140	392140.000000
6	barnet	2019	NULL	396477.000000

PROBLEM

	area	year	population_size
1	barking and dagenham	2017	210711
2	barking and dagenham	2018	211998
3	barking and dagenham	2019	213285
4	barnet	2017	387803
5	barnet	2018	392140
6	barnet	2019	396477

REZULTAT

ROZWIĄZANY

Rysunek 33. Kompletne rozwiązanie problemu pustych wartości w kolumnie population_size dla roku 2019

Źródło: SQL Server

Analogiczne rozwiązanie należy zastosować również w innych miejscach, w których brakuje danych dla konkretnych lat. Kolumna number_of_jobs posiada puste komórki dla lat 1999 oraz 2019. Na podstawie wcześniejszego przykładu stwierdzono, iż bardziej adekwatnym rozwiązaniem problemu generowania brakujących wartości będzie wyznaczenie liniowego trendu dla wszystkich dostępnych okresów, a nie tylko dla

okalających. Należy wziąć dwa najbardziej skrajne okresy (w tym wypadku rok 2018 i 2000), a następnie odjąć je od siebie i przedzielić przez wartości z roku 2000. Następnie uzyskany wynik przedzielić przez liczbę okresów (w tym wypadku 18), a później do rezultatu dodać 1 (odpowiednik 100%) i pomnożyć przez wartości z roku 2018. w ten sposób można uzyskać średnioroczną zmianę wartości w kolumnie z okresemieniem kierunku zmiany. Jest to zdecydowanie dokładniejsza metoda niż uwzględnianie jedynie wartości okalających do estymacji.

```
= (((number_of_jobs_2018 - number_of_jobs_2000) / number_of_jobs_2000) / 18 ) + 1) *
number_of_jobs_2018

number_of_jobs_2018

LAG(number_of_jobs) OVER (PARTITION BY area ORDER BY year ASC) AS Preview3

number_of_jobs_2000

LAG(number_of_jobs,2) OVER (PARTITION BY area ORDER BY year ASC) AS Preview4
```

ROZWIĄZANIE

	area	year	number_of_jobs	Kalkulacja
1	barking and dagenham	2000	57000	57000.000000
2	barking and dagenham	2018	66000	66000.000000
3	barking and dagenham	2019	NULL	66578.947368
4	barnet	2000	138000	138000.000000
5	barnet	2018	170000	170000.000000
6	barnet	2019	NULL	172190.016103

PROBLEM ↗

	area	year	number_of_jobs
1	barking and dagenham	2000	57000
2	barking and dagenham	2018	66000
3	barking and dagenham	2019	66579
4	barnet	2000	138000
5	barnet	2018	170000
6	barnet	2019	172190

;WITH virtual_table4 AS (
SELECT area,year, number_of_jobs
FROM dbo.housing_yearly_GSS_Code
WHERE year IN (2000, 2018, 2019)
GROUP BY area, year, number_of_jobs),
virtual_table5 AS (
SELECT area, year, number_of_jobs,
LAG(number_of_jobs) OVER
(PARTITION BY area ORDER BY year ASC) AS Preview3,
LAG(number_of_jobs,2) OVER
(PARTITION BY area ORDER BY year ASC) AS Preview4
FROM virtual_table4),
virtual_table6 AS(
SELECT area, year, number_of_jobs,
COALESCE(number_of_jobs, (((Preview3 - Preview4)/
Preview4)/18)+1)*Preview3,0) AS Kalkulacja
FROM virtual_table5)
UPDATE dbo.housing_yearly_GSS_Code
SET number_of_jobs = vt6.Kalkulacja
FROM dbo.housing_yearly_GSS_Code h
JOIN virtual_table6 vt6 ON
h.area = vt6.area AND h.year = vt6.year
WHERE h.year = 2019;

Rysunek 34. Kompletne rozwiązanie problemu pustych wartości w kolumnie `number_of_jobs` dla roku 2019

Źródło: SQL Server

Efektem jest uzupełnienie komórek dla roku 2019 wygenerowanymi danymi. Proces ten należy powtórzyć dla roku 1999, aby tego dokonać trzeba odwrócić filtrację wewnątrz okien, ponieważ nie istnieją dane dla lat sprzed roku 1999 w tabeli

dbo.housing_yearly_GSS_Code, więc opóźnione dane funkcją LAG bez odwrócenia filtracji byłyby wartością NULL. Dodatkowo należy w pierwszym podzapytaniu CTE uwzględnić lata 1999, 2000 oraz 2018 w klauzuli WHERE, co zdeterminuje zawartość danych wewnątrz okien. Zmiany również wymaga rok w klauzuli WHERE, w kodzie UPDATE, na końcu całego zapytania.

ROZWIAZANIE 1999	POPRZEDNIE ROZWIAZANIE 2019
<pre><code>;WITH virtual_table7 AS (SELECT area,year, number_of_jobs FROM dbo.housing_yearly_GSS_Code WHERE year IN (1999, 2000, 2018) ● GROUP BY area, year, number_of_jobs), virtual_table8 AS (SELECT area, year, number_of_jobs, LAG(number_of_jobs) OVER (PARTITION BY area ORDER BY year DESC) AS Preview5, LAG(number_of_jobs,2) OVER (PARTITION BY area ORDER BY year DESC) AS Preview6 FROM virtual_table7), virtual_table9 AS(SELECT area, year, number_of_jobs, COALESCE(number_of_jobs, (((Preview3 - Preview4)/ Preview4)/18)+1)*Preview3,0) AS Kalkulacja FROM virtual_table8) UPDATE dbo.housing_yearly_GSS_Code SET number_of_jobs = vt9.Kalkulacja FROM dbo.housing_yearly_GSS_Code h JOIN virtual_table9 vt9 ON h.area = vt9.area AND h.year = vt9.year WHERE h.year = 1999; ●</code></pre>	<pre><code>;WITH virtual_table4 AS (SELECT area,year, number_of_jobs FROM dbo.housing_yearly_GSS_Code WHERE year IN (2000, 2018, 2019) ● GROUP BY area, year, number_of_jobs), virtual_table5 AS (SELECT area, year, number_of_jobs, LAG(number_of_jobs) OVER (PARTITION BY area ORDER BY year ASC) AS Preview3, LAG(number_of_jobs,2) OVER (PARTITION BY area ORDER BY year ASC) AS Preview4 FROM virtual_table4), virtual_table6 AS(SELECT area, year, number_of_jobs, COALESCE(number_of_jobs, (((Preview3 - Preview4)/ Preview4)/18)+1)*Preview3,0) AS Kalkulacja FROM virtual_table5) UPDATE dbo.housing_yearly_GSS_Code SET number_of_jobs = vt6.Kalkulacja FROM dbo.housing_yearly_GSS_Code h JOIN virtual_table6 vt6 ON h.area = vt6.area AND h.year = vt6.year WHERE h.year = 2019;</code></pre>
REZULTAT	<ul style="list-style-type: none"> ● WHERE year IN (2000, 2018, 2019) Zmiana zawartości okien ● WHERE year IN (1999, 2000, 2018) ● ORDER BY year ASC Zmiana kierunku filtracji wewnątrz okien ● ORDER BY year DESC ● WHERE h.year = 1999; Zmiana lokalizacji wprowadzenia rezultatu kodu ● WHERE h.year = 2019;

Rysunek 35. Kompletne rozwiązanie problemu pustych wartości w kolumnie number_of_jobs dla roku 1999

Źródło: SQL Server

Uwagę również zwraca fakt, iż w kolumnie number_of_jobs wartości są zaokrąglone do tysięcy, czego nie uwzględniał kod SQL. Należy więc zaokrąglić całą kolumnę do tysięcy, aby ujednolicić dane.

```
UPDATE dbo.housing_yearly_GSS_Code
SET number_of_jobs = ROUND(number_of_jobs / 1000.0, 0) * 1000;
```

Rysunek 36. Zaokrąglenie wartości do tysięcy w kolumnie number_of_jobs

Źródło: SQL Server

Dla kolumny number_of_houses w tabeli dbo.housing_yearly_GSS_Code brakuje obserwacji dla lat 1999, 2000 oraz 2019. Dane należy wygenerować w analogiczny sposób, jak w przypadku kolumny number_of_jobs, z tą różnicą, że należy wziąć pod uwagę 17, a nie 18 okresów z racji brakujących danych również w roku 2000. Należy pamiętać, że wprowadzenie danych w puste komórki powinno być wykonane w odpowiedniej kolejności. Najlepiej rozpocząć od roku 2019, a następnie po edycji zawartości okien, zmianie kierunku ich filtracji oraz lokalizacji wprowadzania danych, wykonać kod dla roku 2000, a na końcu dla 1999. Najstarszy rok powinien być uwzględniony na końcu, aby uniknąć przypadkowego uwzględnienia pustych komórek wewnętrz okien, które mogłyby pochodzić z roku 2000, będącego rokiem sąsiadującym dla 1999. Można również zwiększyć liczbę okresów z 17 do 18 w trakcie obliczeń po uzupełnieniu danych dla roku 2000. Jednakże ten proces nie wpłynie na końcowy wynik, z racji na zachowanie tendencji liniowej.

Kolumna recycling_pct zawiera puste wartości o oznaczeniu ‘na’. Zmiana typu danych na liczbowy nie jest możliwa, gdy niektóre komórki zawierają litery. W związku z tym należy podążyć za logiką narzuconą przez silnik bazodanowy, zgodnie z którą puste wartości w bazie danych są reprezentowane przez NULL. Należy więc dokonać zamiany komórek z oznaczeniem ‘na’ na NULL. Recycling_pct również zawiera brakujące wartości dla lat 2000 i 2019. Następuje uzupełnienie danych przy pomocy rozwiązania stosowanego w wcześniejszych etapach pracy magisterskiej.

	area	GSS_CODE	date	median_salary	life_satisfaction	mean_salary	recycling_pct	population_size	area_size	no_of_houses	year	number_of_jobs
1	havering	E09000016	2019-12-01	32321	NULL	37161	39	259581	11446	102518	2019	109000
2	havering	E09000016	2018-12-01	30181	7.68	34484	37	257810	11446	101993	2018	108000
3	havering	E09000016	2017-12-01	29134	7.58	33432	37	256039	11446	101716	2017	103000
4	havering	E09000016	2016-12-01	28906	7.53	33025	37	252783	11446	101273	2016	100000
5	havering	E09000016	2015-12-01	30591	7.69	33850	32	249085	11446	100261	2015	93000
6	havering	E09000016	2014-12-01	30003	7.62	33536	32	245974	11446	99619	2014	90000
7	havering	E09000016	2013-12-01	29858	7.51	32758	32	242080	11446	99463	2013	91000
8	havering	E09000016	2012-12-01	28326	7.36	31759	35	239733	11446	99226	2012	84000
9	havering	E09000016	2011-12-01	28415	7.35	31947	36	237927	11446	99184	2011	82000
10	havering	E09000016	2010-12-01	28235	NULL	32068	31	236234	11446	98805	2010	82000
11	havering	E09000016	2009-12-01	28317	NULL	30993	34	234127	11446	98289	2009	80000
12	havering	E09000016	2008-12-01	28800	NULL	32386	27	231793	11446	97551	2008	87000
13	havering	E09000016	2007-12-01	26079	NULL	29418	24	229789	11446	97132	2007	82000
14	havering	E09000016	2006-12-01	26394	NULL	28459	20	228198	11446	96191	2006	86000
15	havering	E09000016	2005-12-01	24296	NULL	27736	18	226990	11446	95727	2005	89000
16	havering	E09000016	2004-12-01	18240	NULL	21505	16	225769	11446	95182	2004	92000
17	havering	E09000016	2003-12-01	18250	NULL	21167	8	225248	11446	94553	2003	92000
18	havering	E09000016	2002-12-01	16123	NULL	19520	5	225054	11446	94194	2002	91000
19	havering	E09000016	2001-12-01	17940	NULL	20002	9	224717	11446	93792	2001	88000
20	havering	E09000016	2000-12-01	17418	NULL	19309	6	225141		93348	2000	89000
21	havering	E09000016	1999-12-01	17165	NULL	18786	8	225712		92896	1999	88000

Rysunek 37. Podglądowy przekrój danych dla pojedynczej dzielnicy i wszystkich dostępnych lat z tabeli dbo. housing yearly

Źródło: SQL Server

Kolumna life_satisfaction w tabeli dbo.housing_yearly_GSS_Code zawiera dane jedynie dla 8 z 21 lat. z tego powodu dane dla ostatniego roku zostaną dodane do tabeli centralnej, która nie posiada wymiaru czasowego. Taka procedura zapewni zachowanie przekrojowej informacji związanej z tą kolumną oraz poprawi czytelność dashboardu. Wraz z kolumną life_satisfaction do tabeli centralnej zostanie przeniesiona również kolumna area_size z tabeli dbo.housing_yearly_GSS_Code. Wielkość powierzchni dzielnic pozostawała stała w czasie, dlatego też, w celu optymalizacji modelu danych, kolumna ta zostanie zapisana jako informacja przekrojowa. Obie kolumny zostają usunięte z tabeli dbo.housing_yearly_GSS_Code.

```
SELECT L.NAME, H.GSS_CODE, H.life_satisfaction
INTO dbo.GIS_London_Polygons_Names_Updated
FROM dbo.housing_yearly_GSS_Code H
JOIN dbo.GIS_London_Polygons_Names L ON H.GSS_CODE = L.GSS_CODE
WHERE H.year = 2018;

ALTER TABLE dbo.housing_yearly_GSS_Code
DROP COLUMN area_size,
DROP COLUMN life_satisfaction;
```

Rysunek 38. Kod zapisujący dane z roku 2018 dla kolumny life_satisfaction w tabeli centralnej oraz usunięcie kolumn area_size i life_satisfaction z dbo. housing yearly

Źródło: SQL Server

Tabela dbo.housing_monthly_GSS_Code zawiera kolumnę o nazwie houses_sold, w której występują braki danych dla lat 2019 i 2020. Należy najpierw zamienić puste obserwacje na NULL, w celu umożliwienia konwersji typu danych w kolumnie, a następnie zastosować dopracowany w wcześniejszych etapach pracy magisterskiej sposób uzupełniania brakujących danych. Pamiętać należy o zachowaniu ostrożności, gdyż agregacja w tabeli dbo.housing_mothly_GSS_Code przebiega miesiącami, a nie latami, jak to miało miejsce w tabeli dbo.housing_yearly_GSS_Code. w wyniku realizacji kodów i weryfikacji wyników stwierdzono, że pozostały 3 obserwacje z wartością NULL dla kolumny houses_sold. Po dokładnej analizie okazuje się, że owe obserwacje są zduplikowane z innymi wewnątrz bazy. Podlegają więc usunięciu.

Z racji na znaczenie informacji zawartych w kolumnie AverageRevenuePerMonth w tabeli dbo.AirBnbGSS_Code, konieczne jest uwzględnienie jedynie tych obserwacji, dla których wartości średniego miesięcznego zysku z najmu krótkoterminowego są większe niż zero. Głównymi elementami

składającym się na wysokość zysku z najmu jest popularność apartamentu oraz cena najmu. Dlatego, jeśli obserwacje zawierają zerowy zysk w kolumnie, sugeruje to brak opinii, gdyż apartament posiada zazwyczaj cenę niezerową. To z kolei może oznaczać, że cena lub standard mieszkania do wynajęcia nie odpowiada rynkowemu, co może wpływać na wyniki analizy. W związku z tym, odrzucane są wszystkie obserwacje, które nie posiadają opinii, a co za tym idzie, zawierają zerowy przychód z najmu³².

```
SELECT *
INTO AirbnbGSS_Code_PBI
FROM dbo.AirbnbGSS_Code
WHERE AverageRevenueMonth > 0;
```

Rysunek 39. Usunięcie obserwacji posiadających zysk równy 0 z dbo. Airbnb poprzez tworzenie nowej tabeli z pominięciem obserwacji

Źródło: SQL Server

Osiągnięty został etap wstępnej obróbki danych, który umożliwia rozpoczęcie pracy w środowisku Power BI. Niektóre ze zmian oraz procesów oczyszczania i obróbki danych nie zostało opisanych, z racji na dość prosty charakter oraz małe znaczenie z perspektywy budowy dashboardu. Poza udokumentowaną częścią pracy magisterskiej dokonano wielu zmian typów danych, zmian lokalizacji kolumn lub ich usunięcia, edycji błędnych danych, usunięcia zduplikowanych obserwacji i tym podobnych działań. Wszystkie źródła danych zostały połączone z Power BI za pomocą DirectQuery, z środowiskiem SQL Server lub plikami płaskimi. Na tym etapie istnieją 4 źródła danych z jedną tabelą centralną. Model danych w kolejnych etapach będzie rozbudowywany za pomocą użycia SQL Server oraz narzędzi Power BI.

Kluczowym elementem przetwarzania danych w pracy magisterskiej jest ich generowanie przy pomocy napisanych zapytań w środowisku SQL Server. Złożone rozwiązania w języku SQL umożliwiły uzupełnienie brakujących danych w komórkach. Niezbędne okazały się procesy zmiany typów danych, edycji języków, usuwania kolumn, filtrowania przy pomocy klauzuli WHERE, łączenia danych za pomocą JOIN, używania logiki CTE oraz zagnieżdżeń, a także wiele innych. Nieocenioną rolę w konfiguracji mapy, która odbyła się również w tej części pracy, odegrało oprogramowanie Mapshaper wraz z QGIS.

³² S. Gigoyan, *Creating a table using the SQL SELECT INTO clause*, 2021, [w:] mssqltips.com, <https://www.mssqltips.com/sqlservertip/6977/sql-select-into-create-table/> [dostęp 22.05.2024].

ROZDZIAŁ 4. BUDOWA DASHBOARDU ANALITYCZNEGO W POWER BI Z UŻYCIEM POWER QUERY I JĘZYKA DAX

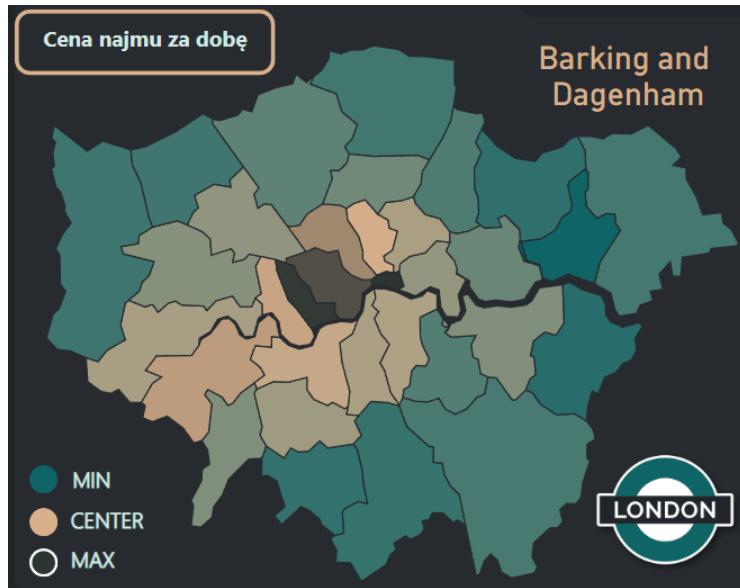
Poniższy rozdział pracy magisterskiej opisuje proces budowy dashboardu analitycznego składającego się z kart: najmu krótkoterminowego, modelu najmu krótkoterminowego, rynku pracy, siły nabywczej, przestępcości, korelacji oraz miernika syntetycznego. Dashboard powstał przy użyciu narzędzi Power BI w tym języka DAX oraz Power Query.

4.1 Karta najem krótkoterminowy

Pierwsza z stron dashboardu dotyczy najmu krótkoterminowego. Założeniem owej karty jest umożliwienie użytkownikowi eksploracji sektora najmu dobowego w Londynie. w owej karcie istnieją dwie odmienne perspektywy. Pierwszą z nich jest perspektywa potencjalnego turysty chcącego skorzystać z najmu mieszkania, a druga dotyczy spojrzenia potencjalnego inwestora, który chciałby zakupić nieruchomości na wynajem. Karta zawiera dwie dodatkowe informacje z perspektywy inwestora, które turysty raczej nie dotyczą, a mianowicie, zwrot z inwestycji w latach oraz średni miesięczny zysk z mieszkania.

Charakterystyczną cechą kart w dashboardzie magisterskim jest mapa osadzona zazwyczaj w lewym górnym rogu. Jest to mapa kształtów z wydzielonymi 33 dzielnicami, których kolorystyka zależy od poziomu zmiennej osadzonej wewnętrz mapy. w karcie najem krótkoterminowy jest to cena najmu za dobę, która została wydobыта z tabeli przy pomocy miary opartej o funkcję AVERAGE³³.

³³ M.Allington, *Super Charge Power BI: Power BI is Better when You Learn to Write Dax*, 2018, s. 62.



Rysunek 40. Mapa Londynu - cena najmu za dobę

Źródło: Microsoft Power BI

Mapa jest interaktywna, więc umożliwia wybór dzielnicy. Skutkuje to filtracją pozostałych wizualizacji na karcie, jak przykładowo średniego zysku miesięcznego z najmu. Informacje zawarte w wizualizacjach dostosowywane są do kontekstu wynikającego z mapy. z tego względu pozostałe wykresy, macierze oraz inne elementy prezentują informacje dotyczące wybranej dzielnicy na mapie, bądź ogółu Londynu w przypadku braku wyboru dzielnicy. Niektóre z wizualizacji mogą również wpływać na mapę, która reprezentuje zależne od kontekstu wartości.

Prawy górny róg karty zajmują 3 wartości. Pierwsza z nich to zwrot z inwestycji w latach, druga dotyczy średniej ceny najmu za dobę dla wybranej dzielnicy oraz trzecia zawierająca średni zysk miesięczny z najmu krótkoterminowego. Druga i trzecia karta została stworzona za pomocą miar korzystających z funkcji AVERAGEX. Pierwsza z nich funkcjonuje również na podstawie miary, lecz z użyciem funkcji DIVIDE. Funkcja ta dzieli średnią cenę nieruchomości na rok 2019 przez średni miesięczny zysk z najmu, pomnożony przez liczbę miesięcy w roku. Wszystkie z trzech wizualizacji filtryują się przez zaznaczoną dzielnicę na mapie, prezentując informację dotyczące wybranej z nich lub całości Londynu w przypadku braku filtracji. Kolor beżowy odpowiada za wartości pozytywne z punktu widzenia inwestora. Im wyższa średnia cena najmu lub mniejsza liczba lat potrzebnych na zwrot kapitału z inwestycji, tym lepiej z perspektywy właściciela apartamentu. Kolor morski odnosi się do wartości negatywnych z punktu widzenia inwestora, a kolor patynowy jest neutralny.



Rysunek 41. Karty informacyjne - zwrot z inwestycji, średnia cena najmu, średni zysk

Źródło: Microsoft Power BI

Poniżej znajdują się dwa wykresy: jeden słupkowy, a drugi kołowy. Wykres kołowy prezentuje podział procentowy typów najmu krótkoterminowego. Dostępny najem dzieli się na trzy kategorie: cały dom lub mieszkanie, pokój lub łóżko w pokoju wieloosobowym. Dla ogółu Londynu udział tych kategorii wynosi odpowiednio: 52.72% dla całego domu/mieszkania, 46.12% dla pokoju, a jedynie 1.16% dla łóżka w pokoju wieloosobowym.

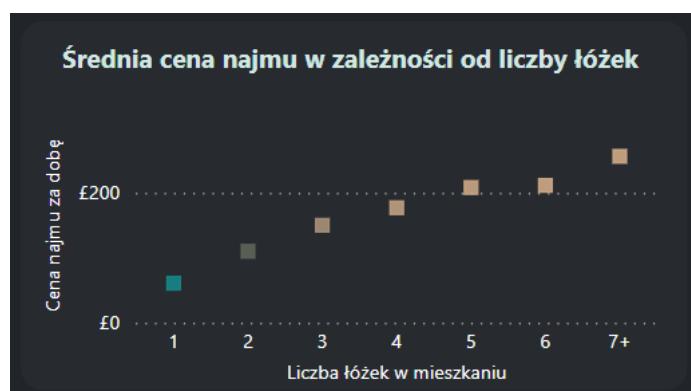
Wykres słupkowy ukazuje liczbę dostępnych ofert najmu w zależności od liczby łóżek w ofertach. Z wykresu wynika, iż najpopularniejszy jest najem zawierający 1 lub 2 łóżka. Apartamenty zawierające więcej niż 7 łóżek zostały przypisane do kategorii 7+ ze względu na ich rzadkość. Dlatego też wykres słupkowy ma rozpiętość od 1 do 7+ łóżek. Do modelu danych została dodana dodatkowa kolumna kalkulacyjna, która, używając funkcji SWITCH, przypisuje obserwacje do odpowiednich kategorii, tak aby zebrać wszystkie apartamenty zawierające więcej niż 7 łóżek w grupę 7+. Wykres kołowy i słupkowy filtrują się wzajemnie.



Rysunek 42. Wykres kołowy udział typu najmu oraz wykres słupkowy liczba łóżek

Źródło: Microsoft Power BI

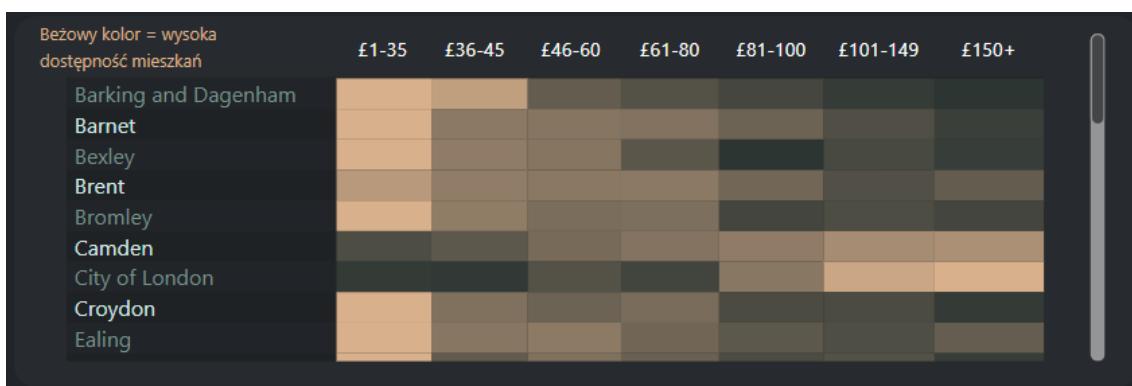
Prawy dolny róg zajmuje wizualizacja reprezentująca średnią cenę najmu w zależności od liczby łóżek. Wizualizacja grupuje obserwacje ofertowe portalu Airbnb, a następnie ukazuje średnią cenę dla każdej z grup w zależności od liczby łóżek. Wizualizacja ta filtrowana jest również przez mapę, co pozwala na dowolną eksplorację w zależności od potrzeb. Założyć można scenariusz, że grupa znajomych planuje podróż do Londynu i poszukuje mieszkani na wynajem, które będzie posiadać cztery łóżka. Dzięki tej wizualizacji mogą oszacować, w których z interesujących ich dzielnic będą w stanie znaleźć tego rodzaju najem w najlepszej cenie. Następnie swoje poszukiwania mogą zawieźć właśnie do tego regionu.



Rysunek 43. Wykres - średnia cena najmu zależna od liczby łóżek

Źródło: Microsoft Power BI

Lewy dolny róg zawiera macierz, która reprezentuje procentowy stosunek liczebności określonych przedziałów cenowych najmu krótkoterminowego w ogólnym udziale najmu. Macierz jest formatowana na podstawie udziału procentowego³⁴. Suma udziałów procentowych wszystkich przedziałów dla każdej z dzielnic wynosi 100%. Jeżeli liczebność określonego przedziału jest wyższa niż 1/7 z 100%, to kolor staje się bardziej beżowy, co oznacza wysoką dostępność mieszkań na wynajem. Jeżeli liczebność jest mniejsza, kolor staje się ciemnobrązowy, co oznacza niską dostępność. Przykładowo, dla dzielnicy Barking and Dagenham, przedziały £1-35 oraz £36-45 są zaznaczone jasno beżowym kolorem, co jest równoznaczne z wysoką dostępnością mieszkań w tym przedziale cenowym. Z kolei przedział £150+ jest oznaczony ciemnobrązowym kolorem, co oznacza, że istnieje mało nieruchomości dostępnych na wynajem krótkoterminowy w tej dzielnicy w tak wysokiej cenie.



Rysunek 44. Macierz przedziałów cenowych najmu w poszczególnych dzielnicach

Źródło: Microsoft Power BI

Otrzymanie poprawnych wyników wymaga przypisania obserwacji do przedziałów oraz wyznaczenia ich granic. Macierz w dashboardzie posiada 7 kategorii cenowych, a każda z nich zawiera podobną liczbę obserwacji. Istotne jest zachowanie podobnej liczebności obserwacji wewnętrz przedziałów, ze względu na punkt odniesienia wynoszący 1/7 z 100%. Dopiero po przefiltrowaniu przez dzielnice za pomocą macierzy, będzie możliwe zauważenie zmiany liczebności przedziałów, która determinuje formatowanie warunkowe koloru komórek. Obserwacje zostały podzielone na 7 równych

³⁴ S. Murray, *Power BI Conditional Formatting for Matrix and Table Visuals*, 2019, [w:] mssqltips.com, <https://www.mssqltips.com/sqlservertip/6265/power-bi-conditional-formatting-for-matrix-and-table-visuals/> [dostęp 23.05.2024].

grup za pomocą języka SQL³⁵. Zadaniem kodu SQL było pozyskanie najwyższej ceny najmu w każdej z wyodrębnionych grup cenowych. Warto zauważyć, że najwyższa cena w danej grupie, na przykład w grupie numer 3, jest równocześnie najniższą ceną grupy numer 4. z tego powodu, w celu uproszczenia konstrukcji zapytania SQL, wystarczy wyznaczyć jedynie maksymalne wartości w poszczególnych grupach.

```

;WITH CTE AS (
    SELECT Price, NTILE(7) OVER (ORDER BY Price) AS Grupy
    FROM dbo.AirBnbGSS_Code_PBI),
    CTE2 AS (
        SELECT MAX(Price) AS Obliczone, Grupy
        FROM CTE
        GROUP BY Grupy)
SELECT *
FROM CTE2;

```

The screenshot shows a SQL query in the left pane and its results in the right pane. The query uses a Common Table Expression (CTE) to calculate price ranges and then groups them by their NTILE value. The results are a table with two columns: 'Obliczone' (calculated values) and 'Grupy' (groups).

	Obliczone	Grupy
1	35	1
2	45	2
3	61	3
4	81	4
5	103	5
6	150	6
7	999	7

Rysunek 45. Wyznaczenie granic przedziałów o takiej samej liczbie obserwacji na podstawie ceny w celu stworzenia macierzy

Źródło: SQL Server

Następnie, przy użyciu funkcji SWITCH w języku DAX³⁶, tworzona jest kolumna kalkulacyjna Price Range, która przypisuje określone obserwacje do odpowiednich przedziałów cenowych. Wartości 61, 81 oraz 103 zostały zaokrąglone w dół.

```

1 Price Range =
2     SWITCH(
3         TRUE(),
4             'AirBnbGSS_Code_PBI'[Price]<= 35, "£1-35",
5             'AirBnbGSS_Code_PBI'[Price]<= 45, "£36-45",
6             'AirBnbGSS_Code_PBI'[Price]<= 60, "£46-60",
7             'AirBnbGSS_Code_PBI'[Price]<= 80, "£61-80",
8             'AirBnbGSS_Code_PBI'[Price]<= 100, "£81-100",
9             'AirBnbGSS_Code_PBI'[Price]<= 149, "£101-149",
10            "£150+"
11     )

```

Rysunek 46. Podział obserwacji na grupy za pomocą kolumny kalkulacyjnej stworzonej w języku DAX

Źródło: Microsoft Power BI

³⁵ J. Celko, *The NTILE Function*, 2023, [w:] red-gate.com, <https://www.red-gate.com/simple-talk/databases/theory-and-design/the-ntile-function/> [dostęp 23.05.2024].

³⁶ M.Allington, *op.cit.*, s. 62.

Macierz dostępna w środowisku Power BI nie posiada funkcjonalności sortowania kolumn według określonych potrzeb. Proces ten wykonywany jest automatycznie w zależności od wartości danych liczbowych, jednak nazwy przedziałów w pracy magisterskiej są danymi tekstowymi, więc będą posortowane przez silnik Power BI w sposób losowy. z racji na istotność sortowania poniższej macierzy i jego wpływ na późniejszą czytelność oraz odbiór, przedziały należy przypisać do liczb od 1 do 7 przy pomocy funkcji SWITCH, a następnie użyć tak stworzonej kolumny do budowy macierzy. Taki sposób umożliwi sortowanie od najmniejszego do największego przedziału odpowiednio od lewej do prawej strony. Niedogodnością związaną z tym rozwiązaniem jest konieczność dodania pól tekstowych do nagłówka każdej z kolumn, zawierających nazwy przedziałów zamiast cyfr.

Macierz automatycznie filtryuje dane przez przedziały, które są kolumnami oraz dzielnice, które stanowią wiersze. Najpierw należy stworzyć miarę, która zliczy liczbę nieruchomości na wynajem w poszczególnych dzielnicach. Pomocną funkcją języka DAX będzie COUNTX. Stworzona miara nosi nazwę ‘CountOfPriceRange’ i nie wymaga dodatkowej filtracji wewnątrz miary, ponieważ sama struktura wizualizacji, którą jest macierz, odpowiada za tę filtrację. Rezultatem jest macierz wyświetlająca liczbę nieruchomości w danym przedziale. Nie jest to oczekiwany efekt z racji na brak punktu odniesienia. Przykładowo dzielnica Barking and Dagenham posiada 43 mieszkania dostępne w przedziale £1-35, a dzielnica Westminster zawiera ich 106. Jednak owe 106 mieszkań w dzielnicy Westminster jest stosunkowo niską dostepnością najmu zważywszy na to, że dzielnica ta posiada, aż 1996 mieszkań dostępnych na wynajem w przedziałach £100+. Dzielnica Barking and Dagenham posiada tylko kilka nieruchomości w przedziałach cenowych powyżej £100, głównie z powodu mniejszej atrakcyjności turystycznej tej dzielnicy. Nie biorąc pod uwagę rozkładu procentowego każdej z dzielnic, można byłoby błędnie oszacować, że dzielnica Westminster oferuje stosunkowo dużo mieszkań w niższych przedziałach cenowych, a więc i procentowy udział tych mieszkań stanowić będzie znaczącą część w rozkładzie. Zależność nominalna (nie procentowa) wobec pozostałych dzielnic wynika z popularności samej w sobie dzielnicy Westminster. Efekt skali determinuje istnienie stosunkowo sporej liczby ofert za niską cenę, ale również znaczco większej liczby ofert za wyższą stawkę. Należy tu przypomnieć, iż przedziały posiadają zbliżony udział procentowy liczby nieruchomości. Filtracja na poszczególne dzielnice dopiero sprawia, że następują odmienne rozkłady.

Z użyciem miary ‘CountOfPriceRange’ powstaje inna miara, która umożliwia osiągnięcie zamierzonego efektu. Używając zmiennej VAR ‘Neighbourhood Total’³⁷ należy najpierw zachować filtracje w owej mierze dla dzielnic Londynu, a następnie użyć owej zmiennej jako mianownika w dzieleniu przez tą miarę. Sposób ten umożliwił utworzenie procentowego punktu odniesienia wewnątrz dzielnic, eliminując jednocześnie czynnik zakłócający odbiór poprawnej narracji budowanej przez macierz, czyli popularności turystycznej dzielnic.

```
1 % of Price Range =
2 VAR NeighbourhoodTotal =
3     CALCULATE(
4         [CountOfPriceRange],
5         ALLEXCEPT(AirBnbGSS_Code_PBI, AirBnbGSS_Code_PBI[Dzielnica])
6     )
7
8 RETURN
9 DIVIDE(
10    [CountOfPriceRange],
11    NeighbourhoodTotal,
12    0
13 )
```

Rysunek 47. Procentowy udział danej grupy w macierzy dla dzielnic Londynu

Źródło: Microsoft Power BI

4.2 Karta model najmu krótkoterminowego

Aglomeracje charakteryzują się występowaniem złożonych zależności, które często stanowią istotny obiekt badań ekonomicznych. Londyn, z uwagi na swoją skalę i złożoność, z pewnością należy do tego typu aglomeracji. Kluczowym zagadnieniem poddanym analizie w tej części pracy magisterskiej jest wyznaczenie faktycznego centrum najmu krótkoterminowego w Londynie. W takim centrum, ceny najmu są zwykle wyższe i odzwierciedlają one popularność danego obszaru. Badanie to nie należy do najprostszych, ponieważ Londyn nie jest typowym miastem turystycznym, jak przykładowo Alicante w Hiszpanii, gdzie bliskość morza determinuje wyższe ceny najmu. Turystyczne miejscowości charakteryzują się zazwyczaj prostą zależnością: im bliżej plaży, tym wyższe ceny najmu krótkoterminowego. Jednak w przypadku Londynu sytuacja nie jest tak oczywista. Z tego powodu w analizie uwzględniono dwa potencjalne centra najmu. Pierwszym z nich jest Pałac Buckingham, znajdujący się w dzielnicy

³⁷ M.Allington, *op.cit.*, s. 150.

Westminster, a drugim jest dworzec główny w City of London, czyli centrum biznesowe miasta. Celem badania jest zbadanie czy centrum Londynu ma większy wpływ na kształtowanie się cen najmu krótkoterminowego w aglomeracji, czy też dominuje wpływ dzielnicy posiadającej największą liczbę zabytków, popularnych wśród turystów.

Niezbędne do przeprowadzenia owego procesu jest wydobycie dwóch kolumn danych, które na tym etapie nie są jeszcze częścią modelu w pracy magisterskiej. Pierwsza z kolumn zawierać będzie odległości pomiędzy każdą obserwacją, będącą ofertą najmu krótkoterminowego, a dworcem centralnym w dzielnicy City of London. Druga kolumna z kolei zawierać będzie odległości pomiędzy tymi obserwacjami, a lokalizacją pałacu Buckingham w dzielnicy Westminster. Oba z tych punktów nie zostały dobrane w sposób losowy. City of London jest naturalnym centrum Londynu, będącym jednocześnie, wraz z Canary Wharf, centrum finansowym miasta, Wysp Brytyjskich, a nawet całej Europy. Pałac Buckingham, będący siedzibą rodziny królewskiej, z kolei jest jedną z najczęściej odwiedzanych atrakcji turystycznych miasta, wraz z ogromną liczbą innych zabytków znajdujących się w dzielnicy Westminster.

Obliczenie odległości pomiędzy określonymi punktami zawierającymi szerokość i długość geograficzną możliwe są na różne sposoby. Istnieje możliwość użycia wzorów z sinusami oraz cosinusami, jednak próbny rezultat z użyciem tej metody zawierał wiele odchyleń od stanu faktycznego. W celu osiągnięcia poprawnych obliczeń dotyczących odległości pomiędzy ofertami najmu, działania zostały przeprowadzone w języku Python wraz z odpowiednimi bibliotekami w środowisku Google Colaboratory. Dane zostały wyeksportowane z modelu znajdującego się w Power BI w postaci pliku płaskiego i przygotowane do opracowania w środowisku Python³⁸.

³⁸ Y. Jun, *Efficient Euclidean distance computation in pandas*, [w:] towardsdatascience.com, <https://towardsdatascience.com/efficient-euclidean-distance-computation-in-pandas-66b472f6b0ba> [dostęp 23.05.2024].

```

import haversine as hs
from haversine import Unit
import pandas as pd

# Wczytanie danych z pliku CSV
df = pd.read_csv('/OdleglosciMiędzySzerokościąADługością.csv')

# Współrzędne punktu loc1
loc1 = (51.5111142321689, -0.0903417291741786)

# Funkcja do obliczania odległości między loc1 a innymi punktami
def oblicz_odleglosci(row):
    loc2 = (row['Latitude'], row['Longitude'])
    return hs.haversine(loc1, loc2, unit=Unit.KILOMETERS)

# Obliczanie odległości dla każdego wiersza w DataFrame
df['odleglosc'] = df.apply(oblicz_odleglosci, axis=1)

# Wyświetlenie pierwszych kilku wierszy z obliczonymi odległościami
df.to_csv('odleglosci_do_lokalizacji.csv', index=False)

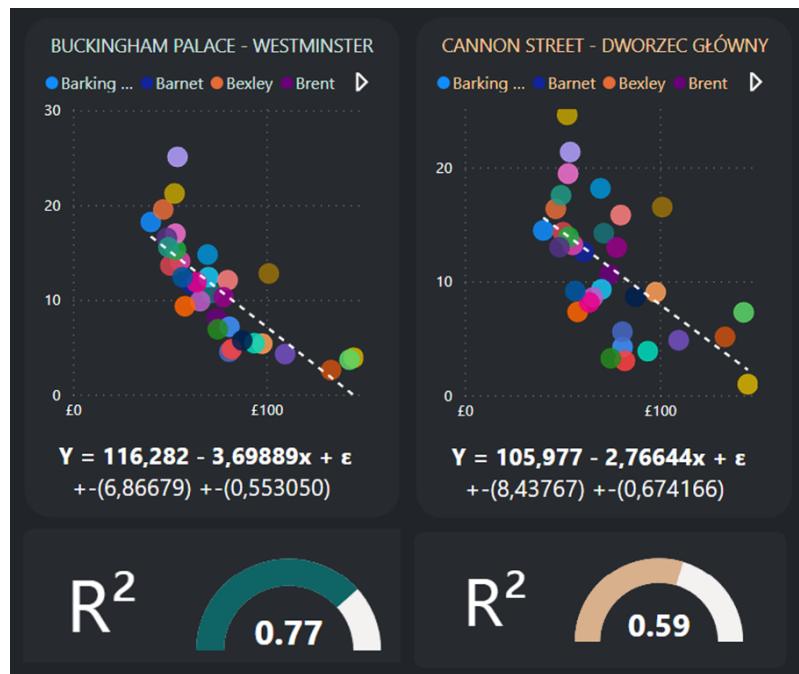
```

Rysunek 48. Obliczanie odległości pomiędzy lokalizacją apartamentów na wynajem, a Dworcem Głównym Londynu

Źródło: Google Colaboratory - Python

Punktem loc1 jest dworzec główny znajdujący się w City of London. Efektem powyższego kodu jest dopisanie do każdej z obserwacji odległości do tego punktu. Analogiczny proces należy przeprowadzić zmieniając współrzędne punktu loc1 na loc2 i uwzględniając w punkcie loc2 dokładną lokalizację pałacu Buckingham. Dane należy z powrotem przetworzyć i załadować do środowiska Power BI. Na tym etapie dane zostały również pogrupowane na dzielnice i zimportowane do tabeli centralnej. Proces ten nie umożliwił jedynie dostarczenia informacji o odległościach pomiędzy poszczególnymi ofertami najmu, lecz także dał możliwość wydobycia danych dotyczących uśrednionych wartości odległości dla wszystkich z dzielnic.

Centralne miejsce w karcie zajmuje mapa kształtów zawierająca informacje na temat średniej ceny najmu za dobę. Lewy górny róg zawiera model korelacji pomiędzy średnią ceną najmu, a średnią odlegością od ofert najmu do pałacu Buckingham w poszczególnych dzielnicach. Prawy róg dashboardu obejmuje analogiczny model regresji, jednak odnoszący się do odległości od Dworca Głównego – Cannon Street w City of London.



Rysunek 49. Korelacje pomiędzy ceną najmu, a odległością od Pałacu Westminster lub Dworca Głównego

Źródło: Microsoft Power BI

Interpretacja modelu korelacji – odległość od Pałacu Buckingham: Jeżeli odległość od Pałacu Buckingham wzrasta o 1 km to cena za najem/doba maleje średnio o 3,699£/doba ze średnim błędem szacunku $+-(0,553\text{£})$ przy założeniu ceteris paribus. Test White'a - H_0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = $0,00974085 <$ poziomu istotności alfa 0.05, więc należy odrzucić hipotezę zerową na rzecz alternatywnej, więc heteroskedastyczność występuje, co jest niekorzystne dla modelu. Test normalności rozkładu reszt - H_0 - składnik losowy ma rozkład normalny = statystyka P dla testu = $0,000511527 <$ od poziomu istotności alfa 0,05, więc należy odrzucić hipotezę zerową na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - $H_0 = p >$ poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = $1,76e-07$ i jest mniejsza od alfa. Należy odrzucić H_0 na rzeczy hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna Odległość od Buckingham Palace objaśnia model w 77%.

Interpretacja modelu korelacji – odległość od Dworca Głównego – Cannon Street: Jeżeli odległość od Dworca Głównego na Cannon Street wzrasta o 1 km to cena za najem/doba maleje średnio o 2,766£ / doba ze średnim błędem szacunku $+-(0,674\text{£})$ przy

założeniu ceteris paribus. Test White'a - H0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = $0,255843 >$ poziomu istotności alfa 0.05, więc istnieje brak podstaw do odrzucenia hipotezy zerowej, więc heteroskedastyczność nie występuje, co jest korzystne dla modelu (homoskedastyczność). Test normalności rozkładu reszt - H0 - składnik losowy ma rozkład normalny = statystyka P dla testu = $0,00962501 <$ od poziomu istotności alfa 0,05, więc należy odrzucić hipotezę zerową na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - H0 = p $>$ poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = 0,0003 i jest mniejsza od alfa. Należy odrzucić H0 na rzeczy hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna Odległość od Cannon Street objaśnia model w 59%.

Znacząca różnica w wartości współczynnika determinacji wskazuje na to, że faktycznym centrum Londynu najmu krótkoterminowego jest Pałac Buckingham. Im bliżej dzielnicy Westminster i znajdującego się w niej Pałacu Buckingham tym większa średnia cena najmu za dobę. Jest to wyjątkowo interesująca informacja, gdyż zazwyczaj w miastach nie będących typowym kurortem turystycznym bądź nie pełniącym funkcji kurortu, centrum miasta jest lokalizacją, w której najem krótkoterminowy osiąga najwyższe ceny.

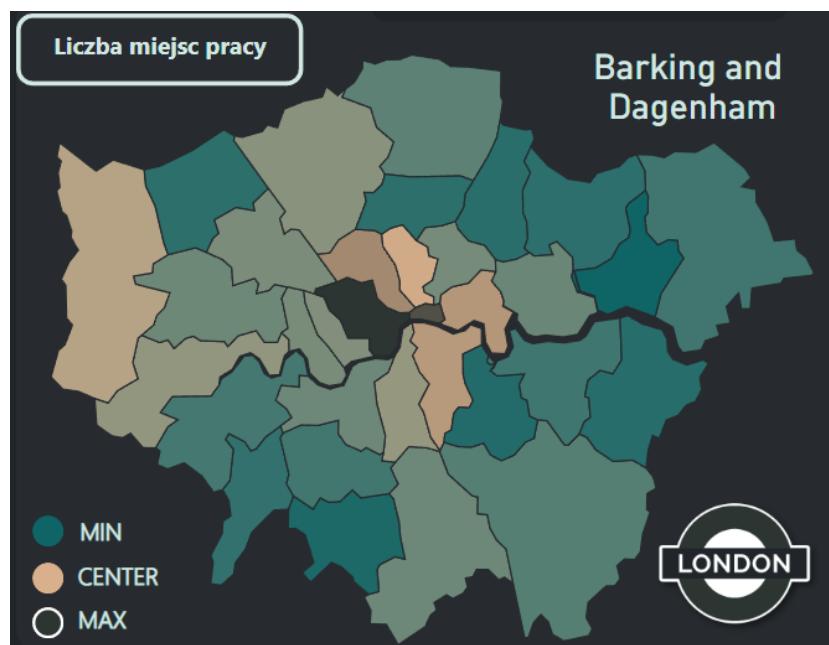
Karta model najmu krótkoterminowego zawiera również kafelki, na których znajdują się wnioski, interpretacje czy informacje takie jak średnia odległość od Pałacu Buckingham, średnia odległość od Dworca Głównego, czy średnia cena najmu za dobę. Wyżej wymienione miary filtrują się w zależności od doboru dzielnicy z centralnie osadzonej mapy w karcie.

4.3 Karta rynek pracy

Podstawą funkcjonowania dużych ośrodków miejskich jest rozwinięty rynek pracy, szczególnie w porównaniu z mniejszymi miejscowościami oraz wsiami. Jest to również czynnik determinujący krajowe migracje wewnętrzne do miast. Jednak nawet wewnątrz miasta, szczególnie tak rozwiniętego jak Londyn, można zauważyć różnice w charakterze dzielnic. Niektóre z nich pełnią rolę sypialni, podczas gdy inne są miejscem funkcjonowania prężnie rozwijającego się biznesu. Obszary aktywne gospodarczo charakteryzują się również stosunkowo wyższymi cenami nieruchomości.

Karta ‘rynek pracy’ pozwala dokonać eksploracji z perspektywy inwestora oraz osób zainteresowanych zamieszkiem w Londynie. Karta ta została opracowana na podstawie danych typu szeregów czasowych, co pozwala zauważać rozwijające się trendy na rynku pracy w Londynie, ze szczególnym uwzględnieniem poszczególnych dzielnic.

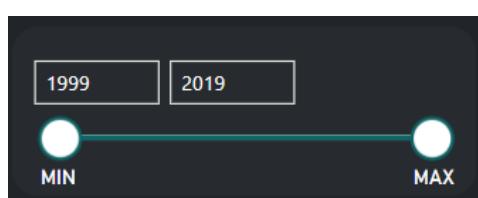
Mapa kształtów w pracy magisterskiej zazwyczaj znajduje się w lewym górnym rogu karty. Mapa na karcie 'rynek pracy' przedstawia dane dotyczące liczby miejsc pracy w poszczególnych dzielnicach na przestrzeni lat 1999–2019.



Rysunek 50. Mapa Londynu - liczba miejsc pracy

Źródło: Microsoft Power BI

Ze względu na zmienność czasową danych, na karcie konieczne stało się dodanie suwaka, dzięki któremu użytkownik dashboardu może wybrać okres analizy.



Rysunek 51. Suwak lat 1999 – 2019

Źródło: Microsoft Power BI

Użycie standardowej miary z zastosowaniem funkcji AVERAGE w celu ukazania liczby miejsc pracy w Londynie spowoduje wyświetlenie się średniej dla wszystkich wybranych na suwaku okresów. Oznacza to, że liczba miejsc pracy zostanie zsumowana dla wybranych lat i podzielona przez ich liczbę. Może to prowadzić do zniekształconego odbioru danych, ponieważ przykładowo gwałtowne zmiany w zatrudnieniu mogą pozostać niezauważone ze względu na uwzględnienie tych lat w średniej. Stworzenie miary, która pozwoli na wyświetlanie na mapie liczby miejsc pracy dla najnowszego roku wybranego na suwaku, umożliwi otrzymanie klarownego obrazu sytuacji. Czyli, gdy suwak będzie obejmował na przykład lata od 2005 do 2015, mapa zaprezentuje liczbę miejsc pracy w poszczególnych dzielnicach dla roku 2015. Suwak zawiera kolumnę 'year' z tabeli 'housing_yearly_GSS_Code_PBI', dlatego pierwsza część miary w języku DAX odnosi się do maksymalnej wartości z tej kolumny. Następnie, przy pomocy funkcji CALCULATE, nadawana jest filtracja w taki sposób, aby rezultatem miary była średnia liczba miejsc pracy dla maksymalnego roku wybranego z kolumny 'year'. Dane przefiltrowane przez dzielnice oraz lata schodzą do maksymalnej głębi logicznej bazy, a co za tym idzie użycie funkcji AVERAGE jako funkcji arytmetycznej nie ma znaczenia. Tabela jest źródłem zagregowanym po latach i dzielnicach. Ze względu na stałą obecność podwójnej filtracji w dashboardzie przez lata oraz dzielnice, możliwe jest zastosowanie dowolnej funkcji arytmetycznej. Zachowana jednak zostanie funkcja AVERAGE ze względu na późniejsze użycie tej samej miary w innym miejscu dashboardu.

```

1 Ilość_miejsc_pracy_MAX =
2 VAR MaxYear = MAX(housing_yearly_GSS_Code_PBI[year])
3 RETURN
4     CALCULATE(
5         AVERAGE(housing_yearly_GSS_Code_PBI[number_of_jobs]),
6         FILTER(
7             housing_yearly_GSS_Code_PBI,
8             housing_yearly_GSS_Code_PBI[year] = MaxYear
9         )
10    )

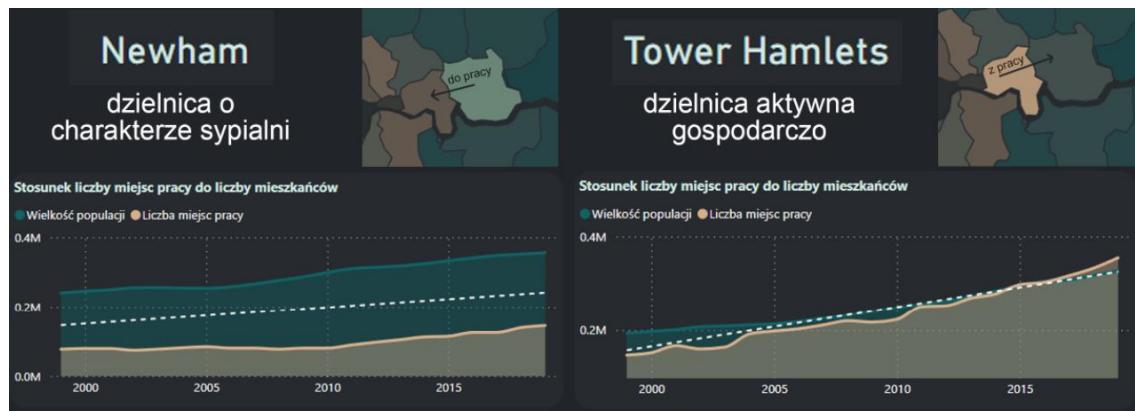
```

Rysunek 52. Kod DAX - średnia liczba miejsc pracy w wybranych latach na suwaku

Źródło: Microsoft Power BI

Poniżej mapy, w lewym dolnym rogu, umieszczony jest wykres o nazwie 'Stosunek liczby miejsc pracy do liczby mieszkańców'. W celu stworzenia tego wykresu,

wykonano dwie miary w języku DAX, wykorzystując funkcję arytmetyczną AVERAGE. Pierwsza z nich zawiera średnią wielkość populacji przy braku filtracji po mapie oraz określoną wielkość populacji, gdy dzielnica jest wybrana. Druga z miar działa w analogiczny sposób, lecz prezentuje dane dotyczące liczby miejsc pracy. Gdy liczba miejsc pracy jest wyraźnie niższa od wielkości populacji, to dana dzielnica ma charakter sypialni. z takich dzielnic znacząca liczba osób dojeżdża do pracy do innych obszarów Londynu, zazwyczaj położonych bliżej centrum, gdzie znajdują się liczne przedsiębiorstwa.



Rysunek 53. Dzielnice o charakterze sypialni lub charakterze biznesowym

Źródło: Microsoft Power BI

Środkową część karty zajmują dwa wykresy. Pierwszy z nich, znajdujący się na górze, dotyczy wzrostu liczby miejsc pracy na km^2 . Drugi natomiast odnosi się do rosnącego trendu gęstości zaludnienia na km^2 . Wykresy zmieniają się w zależności od doboru dzielnicy na mapie. Jednak bez punktu odniesienia czytelność owych wykresów byłaby utrudniona. Na potrzeby wykresów należy więc stworzyć dwie miary. Jedną, która będzie filtrowała się przez dzielnice oraz drugą, która pozostanie na wykresie jako forma reprezentacji dla ogółu Londynu i stanowić będzie punkt odniesienia. Miara ‘Liczba miejsc pracy na km^2 w wybranej dzielnicy’ powstaje przy pomocy funkcji DIVIDE, która dzieli liczbę miejsc pracy z tabeli housing_yearly_GSS_Code_PBI przez powierzchnię poszczególnych dzielnic znajdującej się w centralnej tabeli modelu o nazwie GIS_London_Polygons_PBI.

```

1 Liczba miejsc pracy na KM2 w WYBRANEJ DZIELNICY =
2 DIVIDE(housing_yearly_GSS_Code_PBI[Liczba miejsc pracy],
3 |     AVERAGE(GIS_London_Polygons_PBI[KM2]),0)

```

Rysunek 54. Dzielenie w języku DAX

Źródło: Microsoft Power BI

Druga miara blokuje wszystkie filtry za pomocą funkcji ALL, użytej wewnętrz CALCULATE dla dzielnic zawartych w GIS_London_Polygons_PBI[Name]³⁹. Dzięki zastosowaniu funkcji ALL, wybór dzielnicy na mapie nie wpływa na wykres i tworzy się punkt odniesienia. Na podstawie obu miar powstają wykresy liniowe.

```

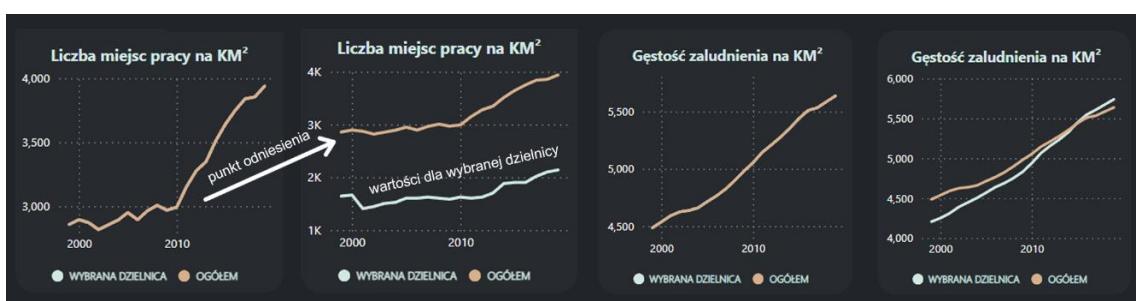
1 Liczba miejsc pracy na KM2 OGÓŁEM =
2 CALCULATE(
3 DIVIDE(housing_yearly_GSS_Code_PBI[Liczba miejsc pracy],
4 |     AVERAGE(GIS_London_Polygons_PBI[KM2]),0
5 |),
6 ALL(GIS_London_Polygons_PBI[NAME])
7 )

```

Rysunek 55. Dzielenie w języku DAX z zablokowaną filtracją

Źródło: Microsoft Power BI

Analogiczne rozwiązania zostają zastosowane również dla wykresu ‘Gęstość zaludnienia na km²’ z tą różnicą, że wykorzystane są inne dane do obliczeń. Oczywiście miary również różnią się nazewnictwem, lecz ich składnia pozostaje zbliżona.



Rysunek 56. Karty zawierające wykresy - w zależności od istnienia filtracji pojawia się punkt odniesienia

³⁹ M. Russo, A. Ferrari, *The Definitive Guide to Dax: Business Intelligence with Microsoft Power BI, SQL Server Analysis Services, and Excel - Published with The Authorization of Microsoft Corporation by Pearson Education*, 2020, s. 189.

Źródło: Microsoft Power BI

Całą prawą stronę karty zajmują informacje dotyczące:

- wielkości populacji,
- liczby miejsc pracy,
- liczby wakatów na 1 osobę,
- mediany wynagrodzeń,
- średniej wynagrodzeń.

Wartości te występują podwójnie z racji na zwiększenie czytelności. Wartość MIN (minimum) dotyczy pierwszego z zaznaczonych okresów na suwaku, z kolei MAX (maksimum) odpowiada za najwyższy z okresów. Domyślnie MIN = 1999, a MAX = 2019.

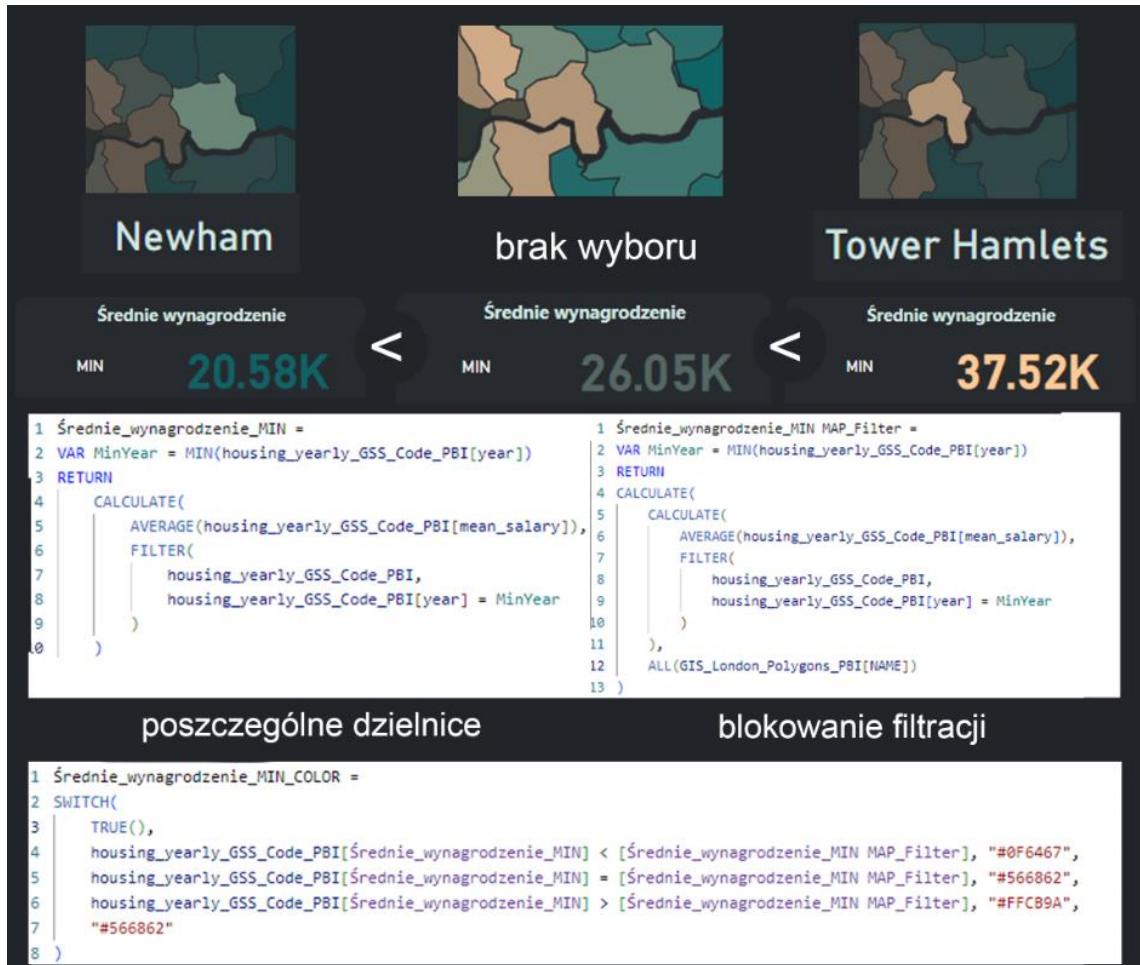


Rysunek 57. Karty informacyjne – strona rynek pracy

Źródło: Microsoft Power BI

Kolor czcionki uzależniony jest od wartości miar w stosunku do średnich wartości z całego miasta. Kolor beżowy dotyczy wartości powyżej średniej, a morski poniżej średniej. Patynowy jest równoznaczny z wartościami średnimi dla Londynu i występuje głównie w przypadku braku filtracji przez dzielnicę. Funkcjonalność ta została stworzona przy pomocy funkcji SWITCH, a następnie użycie powstałej miary w formatowaniu warunkowym⁴⁰.

⁴⁰ S. Shekhar, *3 easy steps to use SWITCH function to make conditional coloured bar charts in Power BI*, 2024, [w:] medium.com, <https://medium.com/microsoft-power-bi/3-easy-steps-to-use-switch-function-to-make-a-conditional-coloured-bar-charts-in-power-bi-d54826fce99f> [dostęp 23.05.2024].



Rysunek 58. Miary DAX pisane pod formatowanie warunkowe kolorów kart

Źródło: Microsoft Power BI

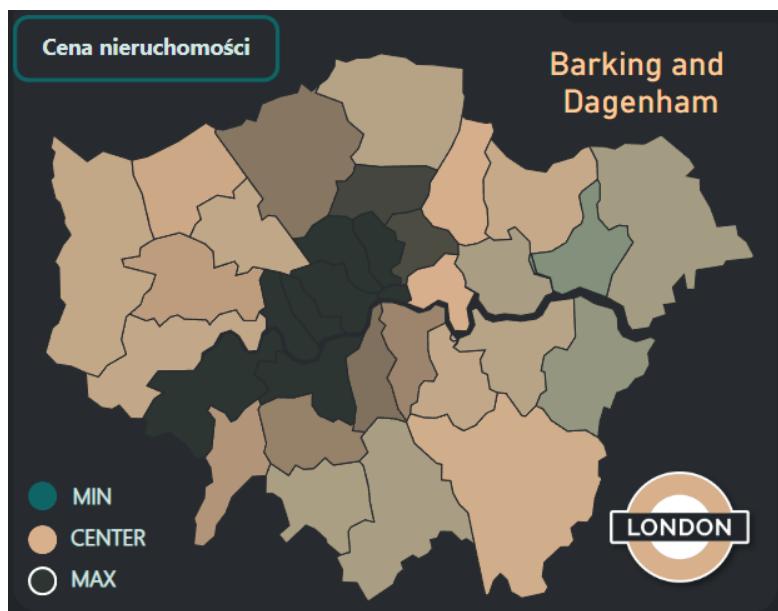
Analogiczny proces zastosowany został do wszystkich kart tego typu. Odpowiedni komplet 3 miar na każdą z kart znajduje się w folderze Rynek_Pracy zawierającym sumarycznie 30 miar obsługujących głównie formatowanie warunkowe.

4.4 Karta siła nabywcza

Mieszkanie jest jednym z podstawowych dóbr niezbędnych do zaspokojenia fundamentalnych potrzeb. Jeszcze niedawno nieruchomości były rozpatrywane głównie w takich kategoriach. Niestety, na przestrzeni dziesięcioleci perspektywa uległa deformacji. Następstwem zmian i postrzegania nieruchomości również jako dobro inwestycyjne jest stały wzrost cen nieruchomości, nieadekwatny do wzrostu przeciętnych zarobków. Problem nie dotyczy wyłącznie Londynu oraz Wysp Brytyjskich, ale wszystkich rozwiniętych gospodarek. Karta ‘siły nabywczej’ umożliwi eksplorację tego

problemu i pozwoli na zrozumienie wielkości dysproporcji zarobków oraz cen mieszkań, oraz trendów występujących w Londynie.

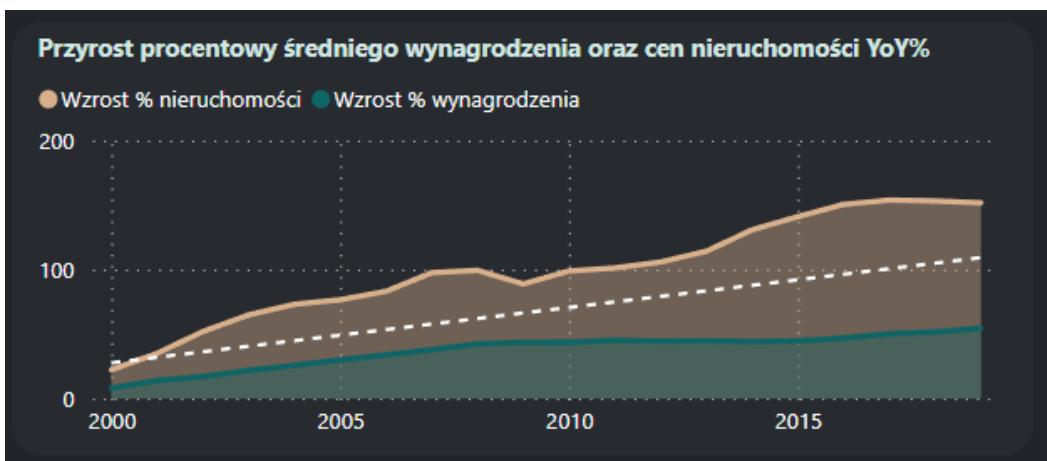
Lewy górny róg zajmuje mapa kształtów Londynu zawierająca dane dotyczące cen nieruchomości. Miara odpowiedzialna za wyświetlanie cen wyciąga informacje z najwyższego wybranego roku na suwaku. Domyślnie więc kolory dzielnic ukazują średnie ceny mieszkań na rok 2019.



Rysunek 59. Mapa Londynu - cena nieruchomości

Źródło: Microsoft Power BI

Poniżej w lewym dolnym rogu zawarty jest wykres o nazwie ‘Przyrost procentowy średniego wynagrodzenia oraz cen nieruchomości YoY%’. Reprezentuje on rok roczny procentowy przyrost wynagrodzeń, który zestawiony jest z rokiem rocznym procentowym przyrostem cen nieruchomości. Wykres ten jest filtrowany przez dzielnice.



Rysunek 60. Procentowy przyrost średnich wynagrodzeń oraz cen nieruchomości rok do roku – suma zmian procentowych

Źródło: Microsoft Power BI

Stworzenie powyższego wykresu stanowiło jedno z największych wyzwań całej pracy magisterskiej. W celu zestawienia procentowego przyrostu wartości nieruchomości oraz przyrostów wartości wynagrodzeń, należy znaleźć wspólny czasowy punkt odniesienia pomiędzy dwoma różnymi tabelami będącymi źródłem danych. Silnik DAX nie zrozumie wymiaru czasowego bez relacji pomiędzy obiema tabelami, z stworzonym w tym celu kalendarzem⁴¹. Dla opracowania wspomnianej wizualizacji, a także z myślą o dostępności przyszłych rozwiązań, został przygotowany kalendarz, który koncentruje się wyłącznie na rocznej zmienności. Następnie została nadana relacja pomiędzy ‘Calendar_Year_PBI’, a ‘housing_monthly_GSS_Code_PBI’ oraz ‘housing_yearly_GSS_Code_PBI’. Niestety, zawarta w tabeli kolumna ‘average_price’, zawierająca potrzebne do wykresu dane dotyczące cen nieruchomości, jest zagregowana na miesiące, co skutkuje brakiem możliwości połączenia danych z kalendarzem Calendar_Year_PBI. Do rozwiązania problemu użyto języka SQL, który zgrupował dane miesięczne latami.

⁴¹ S. Bakhshi, *Expert Data Modeling with Power BI: Get the Best Out of Power BI by Building Optimized Data Models for Reporting and Business Needs*, 2021, s. 76.

```

CREATE TABLE housing_yearly_monthly_PBI (
    area VARCHAR(100),
    GSS_CODE VARCHAR(10),
    year INT,
    average_price_flat DECIMAL(18, 2)
);

INSERT INTO dbo.housing_yearly_monthly_PBI (area, GSS_CODE, year, average_price_flat)
SELECT area, GSS_CODE, year, SUM(sumowanie)/SUM(houses_sold) AS average_price_flat
FROM (
    SELECT area, average_price, houses_sold, GSS_CODE, (average_price * houses_sold)
    AS sumowanie, year
    FROM dbo.housing_monthly_GSS_Code_PBI
) AS cte
GROUP BY area, GSS_CODE, year
ORDER BY area, GSS_CODE, year;

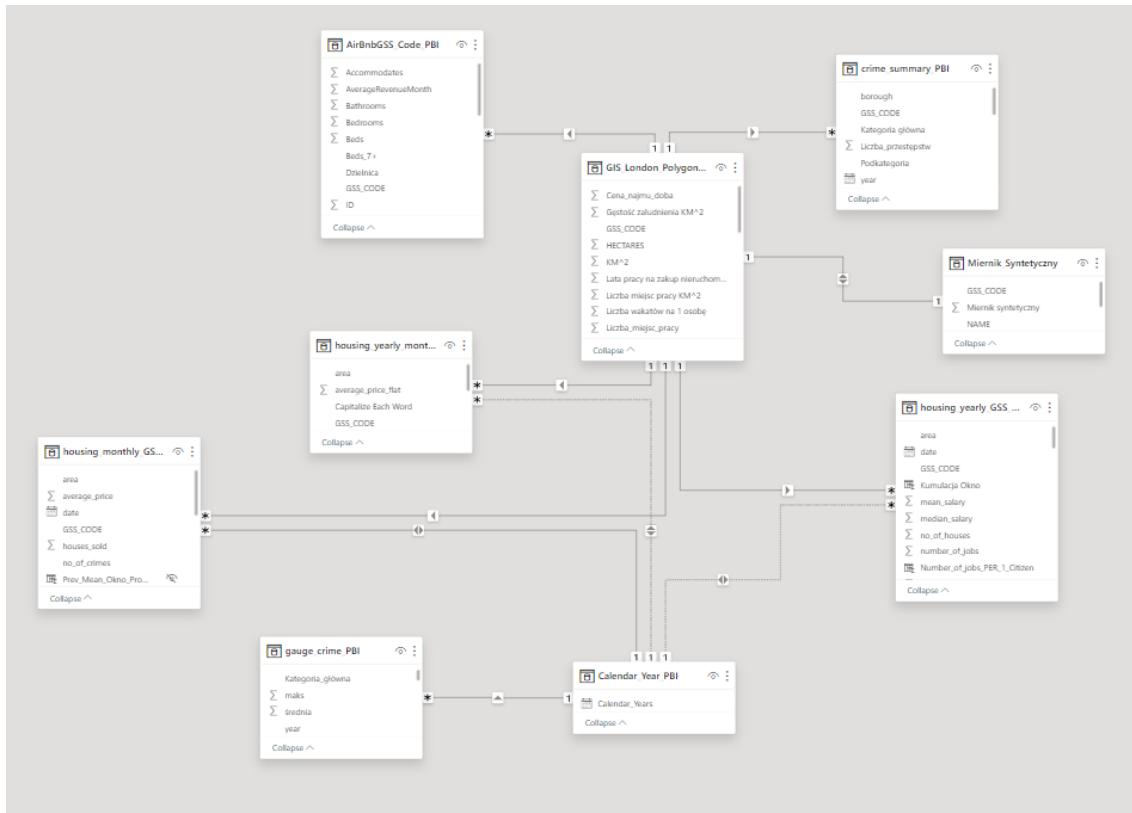
```

Rysunek 61. Grupowanie czasowe danych miesięcznych na roczne w celu stworzenia wizualizacji

Źródło: Microsoft Power BI

Stworzona w ten sposób tabela może zawierać relacje pomiędzy Calendar_Year_PBI, co za tym idzie, powstaje pośrednia relacja pomiędzy dwoma niezbędnymi źródłami danych. w tym miejscu również należy zaprezentować model danych, na którym oparty jest dashboard. Zawiera on informacje i relacje, które dotychczas nie zostały stworzone, jednak będą następstwami dalszych działań w poniższej pracy magisterskiej⁴².

⁴² C. Adamson, *The Complete Reference Star Schema – Section: Factless Fact Tables*, 2010, s. 291 – 306.



Rysunek 62. Model danych przed dodaniem tabeli miernika syntetycznego

Źródło: Microsoft Power BI

Działania na tabelach `housing_yearly_monthly_PBI` oraz `housing_yearly_GSS_Code_PBI` będą zbliżone. Różnice mogą dotyczyć uwzględnienia innych kolumn z danymi. Jednak efektem będzie powstanie analogicznych miar. Cały proces zostanie omówiony na podstawie średniej ceny mieszkania z tabeli `housing_yearly_monthly_PBI` powstały za pomocą języka SQL w procesie pogrupowania innego z źródeł danych.

Celem owych działań jest znalezienie procentowej różnicy pomiędzy cenami mieszkań na przestrzeni lat z podziałem na dzielnice oraz stworzenie kolumny zawierającej zsumowane wartości różnic z każdego kolejnego okresu. Przykładowo, jeżeli w dzielnicy Barnet w roku 1999 mieszkanie średnio kosztowało 136905£, a w latach 2000 i 2001 odpowiednio 167916£ i 185730£, to różnice pomiędzy tymi okresami będą wynosić odpowiednio 31011£ i 17814£. Procentowy przyrost wartości średniej ceny na rynku nieruchomości odpowiednio będzie wynosić w takim wypadku 22.65% oraz 10.61%. Jednak wyświetlenie na wykresie takich wartości nie oddałoby dysproporcji pomiędzy sumaryczną procentową zmianą średniej ceny nieruchomości, a średnich zarobków w Londynie. Dlatego istnieje potrzeba stworzenia kolumny, która będzie reprezentować tzw. Cumulative Total, charakteryzujące się bieżącym

sumowaniem wartości z każdej kolejnej komórki. Na przytoczonym przykładzie pierwsza komórka byłaby pusta z racji na brak punktu odniesienia w stosunku do roku 1999, dla roku 2000 komórka wykazywałaby 22.65%, a w roku 2001 odpowiednio 33.26% (22.65+10.61). Rozwiążanie te umożliwi prezentację dysproporcji w przejrzysty sposób.

Język DAX zawiera zbliżoną logikę tworzenia okien co język SQL, która jest niezbędna do kreacji powyższej wizualizacji. Zacząć należy od odpowiedniego ułożenia danych wewnętrz okien. Dane będą posortowane od najstarszych do najmłodszych latami, a pogrupowane za pomocą funkcji PARTITIONBY na regiony zawarte w tabeli ‘housing_yearly_monthly_PBI’ jako [area]. W funkcji SUMMARIZE również należy umieścić kolumny poddawane analizie, czyli area – warunek grupujący, year – warunek filtrujący oraz average_price_flat – potrzebna informacja. Używając opisanych warunków, należy okalać je funkcjami MINX oraz OFFSET, a następnie przesunąć o jedną pozycję, ustawiając wartość -1 wewnętrz funkcji OFFSET. W powyższy sposób zostaje stworzona kolumna kalkulacyjna w tabeli ‘housing_yearly_monthly_PBI’⁴³.

```
1 Prev_Mean_Okno_Property =
2 MINX(
3   OFFSET(
4     -1,
5     SUMMARIZE('housing_yearly_monthly_PBI', housing_yearly_monthly_PBI[area],
6     housing_yearly_monthly_PBI[year], housing_yearly_monthly_PBI[average_price_flat]),
7     ORDERBY(housing_yearly_monthly_PBI[year]),
8     PARTITIONBY(housing_yearly_monthly_PBI[area])),
9   'housing_yearly_monthly_PBI'[average_price_flat]
10 )
```

Rysunek 63. Kolumna kalkulacyjna przedstawiająca opóźnione o jedną pozycję obserwacje

Źródło: Microsoft Power BI

Efektem kolumny kalkulacyjnej ‘Prev_Mean_Okno_Property’ są dane ‘average_price_flat’ przesunięte o jedną pozycję wewnętrz okien dzielnicowych. Kolejnym krokiem jest wyznaczenie różnic pomiędzy okresami, odejmując od ‘average_price_flat’ przesuniętą o jedną pozycję kolumnę ‘Prev_Mean_Okno_Property’.

⁴³ M. Russo, A. Ferrari, *Understanding apply semantics for window functions in DAX*, 2024, [w:] sqlbi.com, <https://www.sqlbi.com/articles/understanding-apply-semantics-for-window-functions-in-dax/> [dostęp 23.05.2024].

```
1 Różnica_Okno_Property = [average_price_flat]-[Prev_Mean_Okno_Property]
```

Rysunek 64. Kolumna kalkulacyjna obliczająca różnicę pomiędzy średnią ceną mieszkania, a średnią ceną mieszkania opóźnioną o jedną pozycję

Źródło: Microsoft Power BI

Dzięki posiadaniu kolumny zawierającej różnice pomiędzy poszczególnymi okresami możliwe jest wyznaczenie procentowej zmiany tych wartości. Proces ten polega na pomnożeniu wartości różnic przez 100, a następnie podzieleniu uzyskanej wartości przez średnią cenę nieruchomości z poprzedniego roku. Niezbędna do obliczeń jest funkcja IFERROR z racji na istnienie pustych komórek dla roku 1999 w każdym z okien⁴⁴. Niemożliwe jest przeprowadzenie dzielenia przez zero, czy pustą wartość. Rezultatem takiego działania, a w tym wypadku również nie użycia funkcji IFERROR, skutkowałoby błędami.

```
1 Różnica_%_Okno_Property =
2 IFERROR(
3 ([Różnica_Okno_Property] * 100)/[Prev_Mean_Okno_Property], BLANK())
```

Rysunek 65. Kolumna kalkulacyjna obliczająca procentową zmianę

Źródło: Microsoft Power BI

Na podstawie kolumny zawierającej procentowe różnice pomiędzy okresami można stworzyć kolumnę kalkulacyjną, która przeprowadza sumowanie tych różnic. Dane należy wyświetlić w taki sposób, aby każdy następny rok zawierał sumę różnic z lat poprzednich. Kod DAX zawiera trzy zdefiniowane zmienne o nazwach: aktualna data, dzielnica oraz tabela filtrowana. Pierwsza zmienna definiuje rok, druga z zmiennych dzielnice, a trzecia tworzy tabelę opartą na filtracji przy pomocy pierwszej i drugiej z nich. Następnie z użyciem funkcji CALCULATE oraz SUM, przeprowadzone jest sumowanie przez wirtualną tabelę ‘TabelaFiltrowana’.

⁴⁴ M. Russo, A. Ferrari, *op.cit.*, s. 54.

```

1 Kumulacja_Okno_Property =
2 VAR AktualnaData = housing_yearly_monthly_PBI[year]
3 VAR Dzielnica = housing_yearly_monthly_PBI[area]
4 VAR TabelaFiltrowana =
5     FILTER(housing_yearly_monthly_PBI,
6             housing_yearly_monthly_PBI[year] <= AktualnaData &&
7             housing_yearly_monthly_PBI[area] = Dzielnica)
8 RETURN
9 CALCULATE(
10    SUM(housing_yearly_monthly_PBI[Różnica_%_Okno_Property]),
11    TabelaFiltrowana)

```

Rysunek 66. Kolumna kalkulacyjna sumująca procentowe zmiany rok do roku na przestrzeni 20 lat

Źródło: Microsoft Power BI

Dla pełnej ilustracji operacji przeprowadzonych za pomocą ostatnich czterech kolumn kalkulacyjnych zamieszczono rysunek, który umożliwia wizualizację ich efektów.

average_price_flat	year	Prev_Mean_Okno_Property	Różnica_Okno_Property	Różnica_%_Okno_Property	Kumulacja_Okno_Property	area
65456.78	1999		RÓŻNICA	65456.78		Barking and Dagenham
77797.03	2000	65456.78	12340.25	18.8525161182692	18.8525161182692	Barking and Dagenham
88813.38	2001	77797.03	11016.35	14.1603734744116	33.0128895926808	Barking and Dagenham
112844.78	2002	88813.38	24031.4	27.0583103581915	60.0711999508723	Barking and Dagenham
142882.57	2003	112844.78	30037.79	26.618679215822	86.6898791666942	Barking and Dagenham
157695.98	2004	142882.57	14813.41	10.3675416812562	97.0574206479504	Barking and Dagenham
163394	2005	157695.98	5698.019999999999	3.61329439089062	100.670715238841	Barking and Dagenham
168358.26	2006	163394	4964.260000000001	3.03821437751693	103.708929616358	Barking and Dagenham

Rysunek 67. Zobrazowane rozwiązywanie budowy wykresu YoY%

Źródło: Microsoft Power BI

Wykres zestawia dane z dwóch źródeł, dla których czasowym punktem odniesienia jest tabela Calendar_Year_PBI. Należy stworzyć miarę, która przy pomocy funkcji USERELATIONSHIP używa relacji pomiędzy tabelą housing_yearly_monthly_PBI, a Calendar_Year_PBI⁴⁵. Miara ta wewnętrz funkcji CALCULATE zawierać będzie funkcję AVERAGE, dzięki której przy braku występowania filtracji przez mapę wyświetlna będzie średnia wartość z tabeli ‘Kumulacja Okno Property’ dla wszystkich z dzielnic. Dopiero w momencie wyboru dzielnicy na mapie, wykres zobrazuje istniejące w powstałych kolumnach kalkulacyjnych wartości.

⁴⁵ M. Russo, A. Ferrari, *op.cit.*, s. 184.

```

1 Wzrost % nieruchomości =
2 CALCULATE(
3     AVERAGE(housing_yearly_monthly_PBI[Kumulacja Okno Property]),
4     USERELATIONSHIP(housing_yearly_monthly_PBI[year], Calendar_Year_PBI[Calendar_Years])
5 )

```

Rysunek 68. Użycie Calendar_Year_PBI w celu powiązania danych z różnych źródeł dla tych samych okresów

Źródło: Microsoft Power BI

Cały proces, zawierający stworzenie czterech kolumn kalkulacyjnych oraz jednej miary DAX, należy powielić również w tabeli housing_yearly_GSS_Code_PBI, bazując na zawartych w niej danych o średnim wynagrodzeniu. Efektem powielenia wszystkich z pięciu kroków jest uzyskanie miary ‘Wzrost % wynagrodzenia’. Następstwem zestawienia powstałej miary z miarą ‘Wzrost % nieruchomości’ jest powstanie wykresu, reprezentującego znaczącą dysproporcję pomiędzy przyrostem przeciętnych wynagrodzeń, a średnich kosztów zakupu nieruchomości na przestrzeni ostatnich 20 lat.

Środkową część karty zajmują dane dotyczące średnich cen nieruchomości oraz średniego wynagrodzenia. Karty wykorzystują rozwiązania analogiczne do tych zastosowanych na karcie 'Rynek pracy', w tym funkcje MIN i MAX, które odnoszą się odpowiednio do pierwszego i ostatniego zaznaczonego okresu na suwaku. Środkowy panel, poza wyżej wymienionymi danymi oraz suwakiem, zawiera średni współczynnik recyklingu. Dane zostały poddane formatowaniu warunkowemu. Kolor beżowy odpowiada za wartości powyżej średniej, a morski poniżej średniej. Patynowy odnosi się do średniej dla wszystkich z dzielnic. Formatowanie zostało wykonane za pomocą funkcji SWITCH i zastosowane w ustawieniach wizualizacji w sposób analogiczny do tego na karcie 'Rynek Pracy'.



Rysunek 69. Karty informacyjne - Power BI

Źródło: Microsoft Power BI

Miary pod wizualizacją ‘Średnia cena nieruchomości’ zostały stworzone w analogiczny sposób co miary zawarte w karcie ‘Rynek pracy’. Jednak wizualizacja ‘Średnie wynagrodzenie’ wymagała bardziej złożonego rozwiązania. Problem wynika z różnicy w liczbie lat w tabeli ‘Calendar_Year_PBI’, a liczbie lat w tabeli ‘housing_yearly_GSS_Code_PBI’. Tabela, która pełni rolę kalendarza w modelu danych, powinna obejmować minimalną rozpiętość czasową, która jest co najmniej równa najdłuższemu okresowi obecnemu w innych tabelach w modelu danych. Zastosowanie tego samego kodu, co w wizualizacji obok, spowoduje, że przy wyborze roku 1999 lub 2019 na suwaku, wartość zostanie zastąpiona wartością BLANK. Powodem takiego stanu rzeczy jest użycie funkcji MIN oraz MAX. Funkcje te odnoszą się do wartości minimalnych oraz maksymalnych tabeli Calendar_Year_PBI. Wartością minimalną dla ‘Calendar_Year_PBI’ jest rok 1995, a maksymalną 2020. Dla takich lat nie istnieją dane w tabeli ‘housing_yearly_GSS_Code_PBI’, co skutkuje wyświetlaniem wartości BLANK. z pomocą w rozwiązyaniu problemu przychodzą funkcje IF oraz ISBLANK⁴⁶. Tworząc miarę należy w pierwszej kolejności określić trzy zmienne. Pierwszą ‘MinYear’ zawierającą minimalny rok z kolumny [Calendar_Years], drugą ‘MinRok’ zawierającą

⁴⁶ M. Russo, A. Ferrari, *op.cit.*, s. 270.

minimalny rok z kolumny [year] z tabeli ‘housing_yearly_GSS_Code_PBI’. Trzecią zmienną jest kod, który był już stosowany w innych miejscach pracy magisterskiej i użyty zostałby w przypadku braku problemów z tabelą ‘Calendar_Year_PBI’.

```
1 Średnie wynagrodzenie MIN =
2 VAR MinYear = MIN(Calendar_Year_PBI[Calendar_Years])
3 VAR MinRok = MIN(housing_yearly_GSS_Code_PBI[year])
4 VAR Kalkulacje =
5 CALCULATE(
6     AVERAGE(housing_yearly_GSS_Code_PBI[mean_salary]),
7     FILTER(
8         housing_yearly_GSS_Code_PBI,
9         housing_yearly_GSS_Code_PBI[year] = MinYear
10    ),
11    USERELATIONSHIP(housing_yearly_GSS_Code_PBI[year], Calendar_Year_PBI[Calendar_Years])
12 )
```

Rysunek 70. Średnie wynagrodzenie MIN – miara DAX

Źródło: Microsoft Power BI

Funkcja IF pozwala na ustawienie warunku, dzięki któremu, gdy w wizualizacji pojawiłaby się pusta wartość, silnik DAX wykonuje alternatywne działanie, co pozwala uniknąć tak poważnego błędu. Wewnątrz tej funkcji należy zagnieździć funkcję ISBLANK, aby silnik DAX mógł określić, czy wartość jest pusta, czy nie. w ten oto sposób powstaje pełna wersja kodu, która wyświetla informacje również dla roku 1999. Analogicznym sposobem, tylko z użyciem funkcji MAX, należy napisać odpowiednią miarę, aby rozwiązać problem dla roku 2019. Następnie należy użyć obie miary wewnątrz wizualizacji.

```

1 Średnie wynagrodzenie MIN =
2 VAR MinYear = MIN(Calendar_Year_PBI[Calendar_Years])
3 VAR MinRok = MIN(housing_yearly_GSS_Code_PBI[year])
4 VAR Kalkulacje =
5 CALCULATE(
6     AVERAGE(housing_yearly_GSS_Code_PBI[mean_salary]),
7     FILTER(
8         housing_yearly_GSS_Code_PBI,
9         housing_yearly_GSS_Code_PBI[year] = MinYear
10    ),
11    USERELATIONSHIP(housing_yearly_GSS_Code_PBI[year], Calendar_Year_PBI[Calendar_Years])
12  )
13 VAR Kalkulacje2 =
14 IF(
15     ISBLANK(Kalkulacje),
16     CALCULATE(
17         AVERAGE(housing_yearly_GSS_Code_PBI[mean_salary]),
18         FILTER(
19             housing_yearly_GSS_Code_PBI,
20             housing_yearly_GSS_Code_PBI[year] = MinRok
21           ),
22         USERELATIONSHIP(housing_yearly_GSS_Code_PBI[year], Calendar_Year_PBI[Calendar_Years])
23       ),
24     CALCULATE(
25         AVERAGE(housing_yearly_GSS_Code_PBI[mean_salary]),
26         FILTER(
27             housing_yearly_GSS_Code_PBI,
28             housing_yearly_GSS_Code_PBI[year] = MinYear
29           ),
30         USERELATIONSHIP(housing_yearly_GSS_Code_PBI[year], Calendar_Year_PBI[Calendar_Years])
31     )
32   )
33 RETURN
34 Kalkulacje2

```

Rysunek 71. Rozwiązywanie problemu braku danych dla lat z Calendar_Year wybranych po lewej stronie suwaka (analogiczne rozwiązanie dla prawej strony – zamiast MIN → MAX) – średnie wynagrodzenie

Źródło: Microsoft Power BI

Prawy górny róg karty ‘Siła nabywcza’ zajmuje wizualizacja pod nazwą ‘Średnia liczba sprzedanych nieruchomości rocznie’. Wizualizacja ma swoje źródło w tabeli ‘housing_monthly_GSS_Code_PBI’, która zagregowana jest po miesiącach. Kod w języku DAX będzie obliczał średnią liczbę sprzedanych nieruchomości miesięcznie, więc satysfakcjonującym rozwiązaniem jest mnożenie wyników przez liczbę miesięcy w roku.



Rysunek 72. Wykres - średnia liczba sprzedanych nieruchomości rocznie

Źródło: Microsoft Power BI

Miara wyświetlająca średnią liczbę sprzedanych miesięcznie nieruchomości dla wybranej dzielnicy została stworzona w oparciu o funkcję AVERAGEX. z kolei miara, pełniąca funkcję punktu odniesienia, została stworzona przy pomocy użycia funkcji ALL, która blokuje filtrację przez mapę Londynu. Wartości z obu miar mnożone są również przez liczbę miesięcy w roku.

```

1 Średnia ilość sprzedanych nieruchomości (miesięcznie) DLA WSZYSTKICH DZIELNIC =
2 CALCULATE(
3     AVERAGEX(
4         housing_monthly_GSS_Code_PBI,
5         housing_monthly_GSS_Code_PBI[houses_sold]
6     ) * 12,
7     ALL(GIS_London_Polygons_PBI[NAME])
8 )

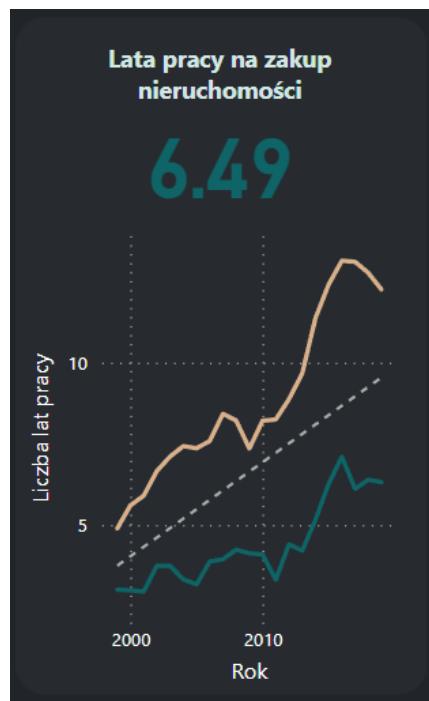
```

Rysunek 73. Miara DAX do punktu odniesienia na wykresie - średnia liczba sprzedanych nieruchomości rocznie

Źródło: Microsoft Power BI

Prawy dolny róg zajmuje ostatnia z wizualizacji dostępnej na karcie ‘Siła nabycwca’. Wizualizacja ta dotyczy lat pracy potrzebnych w celu zakupu nieruchomości. Miara jest wyrażona w latach, a z wynagrodzeń nie zostały potrącone ani wydatki na życie, ani podatki. Założeniem wizualizacji jest przeznaczenie 100% wynagrodzenia jednej osoby na cele mieszkaniowe. Uwzględnienie czynników poza płacowych, które mają wpływ na ostateczną wysokość dochodu rozporządzalnego, mogłoby spowodować

zaburzenia w interpretacji wykresu. Głównym celem jest zobrazowanie w sposób przejrzysty i nie zakłócony czynnikami zewnętrznymi trendu w czasie.



Rysunek 74. Wykres - lata pracy na zakup nieruchomości

Źródło: Microsoft Power BI

Do budowy wizualizacji potrzebne są dwie miary. Jedna bez zablokowanej filtracji oraz druga z zablokowaną filtracją przez dzielnicę z użyciem funkcji CALCULATE i ALL. Obie miary powstają poprzez podzielenie miary 'Średnia cena nieruchomości MAX' przez miarę 'Średnie wynagrodzenie MAX' przy użyciu funkcji DIVIDE.

zablokowana filtracja przez dzielnice <pre> 1 Lata pracy na zakup nieruchomości OGÓŁEM = 2 CALCULATE(3 DIVIDE(housing_monthly_GSS_Code_PBI[Średnia cena nieruchomości MAX], 4 housing_yearly_GSS_Code_PBI[Średnie wynagrodzenie MAX],0), 5 ALL(GIS_London_Polygons_PBI[NAME]) 6) </pre>	brak blokowania filtracji <pre> 1 Lata pracy na zakup nieruchomości DLA WYBRANEJ DZIELNICY = 2 DIVIDE(housing_monthly_GSS_Code_PBI[Średnia cena nieruchomości MAX], 3 housing_yearly_GSS_Code_PBI[Średnie wynagrodzenie MAX],0) </pre>
---	--

Rysunek 75. Zablokowana filtracja jako kreacja punktu odniesienia w obliczu filtracji przez mapę

Źródło: Microsoft Power BI

Dzięki wcześniejszemu rozwiązaniu problemów związanych z filtracją przez suwak informacji dotyczących średniego wynagrodzenia, ułatwiony został proces budowy wizualizacji "Lata pracy na zakup nieruchomości". Proces ten ukazuje istotność podziału pracy nad dashboardem na etapy oraz rozsądne planowanie jego budowy. Rozpoczęcie tworzenia karty "Siła nabywcza" od tej wizualizacji mogłoby utrudnić wykrycie problemu z pustymi wartościami, a rozwiązanie mogłoby być bardziej złożone niż w przypadku skupienia się wyłącznie na wydobyciu wartości średniego wynagrodzenia. Podział pracy na etapy, a następnie utrzymanie odpowiedniej kolejności wykonywanych działań, pozwala zaoszczędzić znaczną ilość czasu poprzez skrócenie drogi w poszukiwaniu możliwych problemów.

4.5 Karta przestępcość

Poziom przestępcości w różnych dzielnicach miasta może istotnie wpływać na decyzję dotyczącą wyboru najlepszej lokalizacji do zamieszkania, inwestycji czy też turystyki miejskiej. Wyższa liczba przestępstw w danej dzielnicy może prowadzić do wzrostu poczucia zagrożenia wśród mieszkańców. Karta przestępcości pozwalaając na ocenę stopnia bezpieczeństwa w poszczególnych obszarach miasta, staje się użytecznym narzędziem dla turystów, inwestorów oraz mieszkańców. Analiza zmienności liczby przestępstw w czasie, umożliwia obserwację ewoluujących trendów, a klasyfikacja przestępstw według kategorii i podkategorii, dostarcza pełniejszego obrazu w kontekście miejskim.

Na potrzeby dalszej analizy zaleca się zgrupowanie istniejącego źródła danych dotyczącego przestępstw. Niniejsza procedura pozwoli na efektywniejsze działania z wykorzystaniem wspomnianego źródła. Ograniczenie liczby obserwacji z użyciem grupowania z podziałem na kategorie, podkategorie, dzielnice oraz lata zwiększy czytelność źródła danych. Początkowa tabela, zawierająca ponad 13 milionów wierszy, została skondensowana do 8262 obserwacji, zachowując jednocześnie możliwość prowadzenia planowanych analiz w opracowywanym dashboardzie. Niektóre z podkategorii nie zawierają żadnych informacji bądź informacje błędne. w związku z tym przy okazji przeprowadzania operacji w języku SQL, podkategorie te nie zostaną uwzględnione.

```

CREATE TABLE crime_summary_PBI (
    borough VARCHAR(50),
    GSS_CODE VARCHAR(10),
    major_category VARCHAR(50),
    minor_category VARCHAR(50),
    count_crime INT,
    year INT
);

INSERT INTO dbo.crime_summary_PBI (borough, GSS_CODE, major_category, minor_category,
count_crime, year)
SELECT borough, GSS_CODE, major_category, minor_category, SUM(value) AS count_crime,
year
FROM dbo.crime_GSS_Code_PBI
WHERE minor_category NOT IN ('Rape', 'Other Sexual', 'Counted per Victim', 'Other
Fraud & Forgery')
GROUP BY borough, major_category, minor_category, year, GSS_CODE
ORDER BY year, borough, count_crime;

```

Rysunek 76. Agregacja danych dotycząca liczby przestępstw oraz usunięcie wybranych kategorii

Źródło: SQL Server

Karta ‘przestępstwa’ zawiera łącznie 9 mierników promieniowych. Pierwszy z nich dotyczy ogółu przestępstw, drugi liczby przestępstw na 1000 mieszkańców, a pozostałe 7 z nich, podziału na kategorie. Do stworzenia mierników promieniowych z podziałem na kategorie posłużyła specjalna tabela przygotowana na ten cel w środowisku SQL. Zawiera ona maksymalne i średnie wartości dla wszystkich z kategorii w poszczególnych latach. z racji na małą liczbę obserwacji (63), dodanie owej tabeli nie powinno obciążyć modelu danych. Po imporcie tabeli będącej efektem zapytania SQL, należy utworzyć relację pomiędzy tabelą ‘gauge_crime_PBI’, a ‘Calendar_Year_PBI’.

```

;WITH cte AS (
    SELECT borough, major_category, year, SUM(count_crime) AS suma
    FROM dbo.crime_summary_PBI
    GROUP BY major_category, borough, year)
SELECT major_category, year,
MAX(suma) AS maksymalne, AVG(suma) AS srednia
FROM cte
GROUP BY major_category, year
ORDER BY year ASC;

```

Rysunek 77. Kod SQL tworzący tabelę wspierającą kreację mierników promieniowych

Źródło: SQL Server

Lewy górny róg karty zajmuje mapa kształtów Londynu, zawierająca dane dotyczące liczby przestępstw w poszczególnych dzielnicach, dla najwyższego roku zaznaczonego na suwaku znajdującym się w innym miejscu karty.



Rysunek 78. Mapa Londynu - liczba przestępstw

Źródło: Microsoft Power BI

Miara użyta do zobrazowania danych na mapie bazuje na podobnej strukturze co już istniejące miary w modelu danych dla pozostałych kart dashboardu. Pierwsza z zmiennych VAR przy pomocy funkcji VALUES zwraca unikalne wartości dla kolumny 'crime_summary_PBI'[year]. Druga z zmiennych zwraca przy pomocy funkcji MAXX najwyższą wartość z unikalnych wartości pochodzących z pierwszej zmiennej. Za pomocą funkcji FILTER, zagnieźdzonej wewnętrz funkcji CALCULATE, zwracane są wartości sumowania dla najwyższego z zaznaczonych na suwaku okresów.

```

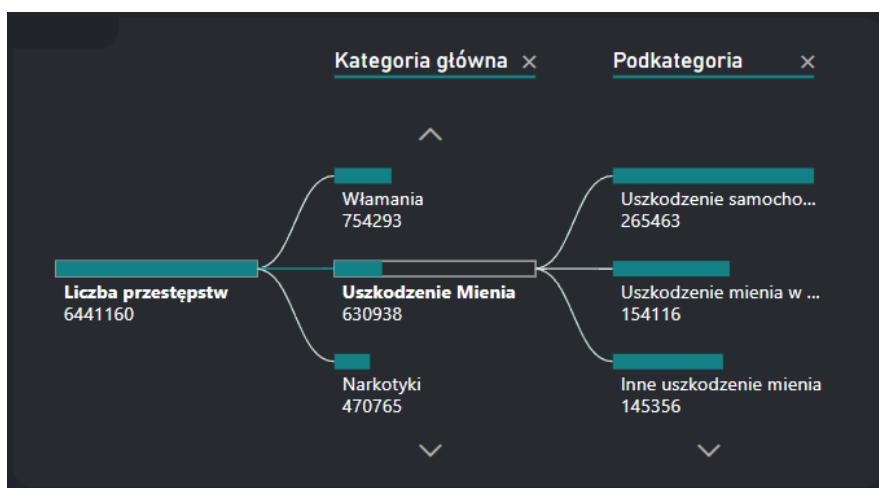
1 Liczba przestępstw dla maksymalnego wybranego roku =
2 VAR SelectedYears = VALUES(crime_summary_PBI[year])
3 VAR MaxYear = MAXX(SelectedYears, crime_summary_PBI[year])
4 RETURN
5   CALCULATE(
6     SUM(crime_summary_PBI[Liczba_przestepstw]),
7     FILTER(
8       crime_summary_PBI,
9       crime_summary_PBI[year] = MaxYear
10      )
11    )

```

Rysunek 79. Miara wyświetlająca sumę przestępstw dla najwyższego roku z wybranych na suwaku lat

Źródło: Microsoft Power BI

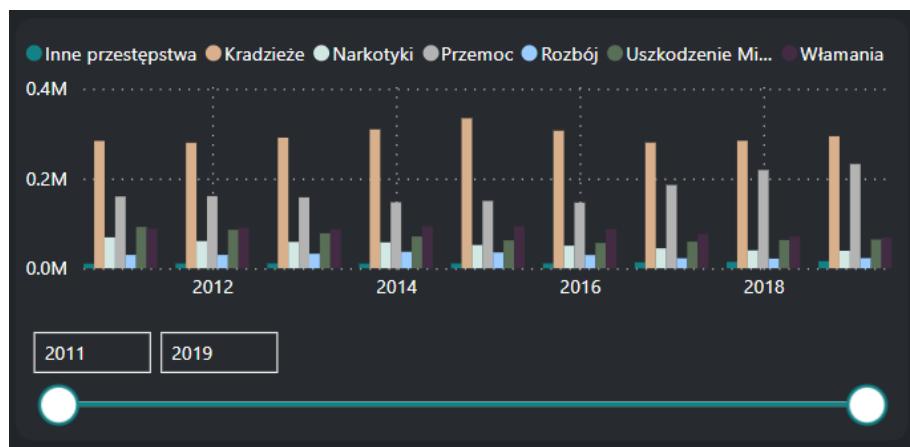
Prawy górny róg karty przedstawia drzewo dekompozycji, obrazujące strukturę podziału liczby przestępstw na kategorie i podkategorie. Dane te zostały przetłumaczone z języka angielskiego na polski za pomocą Power Query. Drzewo dekompozycji filtrowane jest przez mapę, dzięki czemu można zaobserwować liczbę przestępstw w określonych kategoriach w wybranej przez użytkownika dzielnicy dashboardu. Drzewo sumuje wartości z całego zaznaczonego na suwaku okresu. Gdy na suwaku ustalony jest przedział czasowy od 2011 do 2019 roku, a na mapie wybrana jest konkretna dzielnica, na przykład Tower Hamlets, liczba przypadków morderstw wynosząca 45 odnosi się do całego okresu ośmiu lat dla dzielnicy Tower Hamlets.



Rysunek 80. Drzewo - kategorie i podkategorie przestępstw

Źródło: Microsoft Power BI

Pod drzewem dekompozycji znajduje się wykres słupkowy, który obrazuje zmienność liczby przestępstw w czasie z podziałem na kategorie.



Rysunek 81. Wykres - liczba przestępstw w wybranych kategoriach na przestrzeni lat

Źródło: Microsoft Power BI

Na spodzie dashboardu, wzdłuż całej szerokości karty, umieszczonych jest 7 mierników promieniowych z podziałem na kategorię przestępstw. Mierniki te dotyczą:

- kradzieży,
- przemocy,
- włamań,
- uszkodzenia mienia,
- narkotyków,
- rozboju,
- innych przestępstw.

Każdy z wymienionych mierników powinien zawierać 5 miar dostarczających odpowiednich informacji do każdego z nich (razem 35 miar). Wszystkie mierniki zawierają:

- wartość minimalną dla danej kategorii,
- wartość maksymalną dla danej kategorii,
- średnią wartość dla całego Londynu dla danej kategorii,
- liczbę przestępstw dla danej kategorii ogółem,
- miarę odpowiadającą za formatowanie warunkowe.

Wartość minimalna jest równa 0, gdyż nie może wystąpić mniej przestępstw niż zero, niezależnie od okoliczności. Miara określająca zero będzie więc uniwersalnie stosowana do każdego z mierników.

```

MIN =
VAR Minimum = 0
RETURN
Minimum

```

Rysunek 82. Miara DAX - wartość 0

Źródło: Microsoft Power BI

Następne w kolejności jest określenie wartości maksymalnej dla kategorii. Cały proces budowy mierników zostanie zaprezentowany na kategorii 'Kradzież'. Najpierw należy stworzyć pierwszą zmienną z użyciem funkcji VALUES, która zwraca będzie unikalne wartości. Druga zmienna z użyciem funkcji MAXX zwraca najwyższą wartość z tych zawartych w pierwszej zmiennej. Trzecią zmienną jest zmienna o nazwie 'Kradzieże', w której określona jest funkcja CALCULATE, wewnętrz której funkcja MAX zwraca dane z przefiltrowanej tabeli. Warunek filtracji określony jest przy pomocy funkcji FILTER, na podstawie tabeli 'gauge_crime_PBI'. Jego warunkami jest wzięcie pod uwagę tylko obserwacji należących do kategorii 'Kradzieże' dla maksymalnego roku wybranego na suwaku.

```

1 KradzieżeMAX =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaksYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kradzieze =
5   CALCULATE(
6     MAX (gauge_crime_PBI[maks]),
7     FILTER(gauge_crime_PBI,
8       gauge_crime_PBI[Kategoria_główna] = "Kradzieże" &&
9       gauge_crime_PBI[year] = MaksYear))
10 RETURN
11 Kradzieze

```

Rysunek 83. Miara - maksymalna liczba przestępstw w wybranej kategorii jako górny punkt odniesienia miernika promieniowego

Źródło: Microsoft Power BI

Zbliżonym sposobem powstaje miara odpowiadająca za ukazywanie średniej liczby przestępstw w kategorii 'Kradzieże'. Powstała miara będzie punktem odniesienia wewnętrz mierników. Istniejący tego typu punkt odniesienia pozwoli zaobserwować, w jakim stopniu liczba przestępstw odbiega od średniej w danej dzielnicy, czy to w aspekcie pozytywnym, czy negatywnym. Dotyczy to wszystkich z kategorii.

```

1 KradzieżeTARGET =
2 CALCULATE(
3 AVERAGE(gauge_crime_PBI[średnia]),
4 FILTER(gauge_crime_PBI,
5     gauge_crime_PBI[Kategoria_główna] = "Kradzieże" &&
6     gauge_crime_PBI[year] IN VALUES (crime_summary_PBI[year])))

```

Rysunek 84. Miara - średnia liczba przestępstw jako cel miernika promieniowego

Źródło: Microsoft Power BI



Rysunek 85. Mierniki promieniowe dla wybranej na mapie dzielnicy, w której występuje mniej przestępstw niż średnia w Londynie

Źródło: Microsoft Power BI



Rysunek 86. Mierniki promieniowe dla wybranej na mapie dzielnicy, w której występuje więcej przestępstw niż średnia w Londynie

Źródło: Microsoft Power BI

Najważniejszą z miar, umieszczonych wewnętrz mierników promieniowych, jest miara odpowiadająca za wyświetlanie liczby przestępstw dla wybranych dzielnic lub całości Londynu. Jednym z problemów wynikających z budowy miary jest fakt, że w obliczu braku filtracji przez mapę, suma przestępstw ukazuje się dla całego Londynu. Ta wartość nie jest reprezentatywna. Mierniki, w obliczu braku filtracji, powinny ukazywać wartości średnie, dla najwyższego wybranego na suwaku roku, z podziałem na kategorie. Zadaniem owej miary jest w przypadku braku filtracji, zapełnienie miernika promieniowego do linii celu, a w momencie filtracji przez wybraną dzielnicę, wyświetlenie faktycznej liczby przestępstw dla danej kategorii wewnątrz miernika. Miara z użyciem funkcji IF oraz ISFILTERED warunkuje, czy wyświetlona w mierniku jest

suma przestępstw dla danej kategorii oraz dzielnicy, czy średnia liczba przestępstw dla kategorii⁴⁷.

```
1 Kradzieże =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 RETURN
5 IF(
6     ISFILTERED(GIS_London_Polygons_PBI[NAME]) || ISFILTERED('crime_summary_PBI'[year]),
7     CALCULATE(
8         SUM('crime_summary_PBI'[Liczba_przestępstw]),
9         FILTER(
10            'crime_summary_PBI',
11            'crime_summary_PBI'[Kategoria_główna] = "Kradzieże" &&
12            'crime_summary_PBI'[year] = MaxYear
13        )
14    ),
15    CALCULATE(
16        AVERAGE(gauge_crime_PBI[średnia]),
17        FILTER(
18            gauge_crime_PBI,
19            gauge_crime_PBI[Kategoria_główna] = "Kradzieże" &&
20            gauge_crime_PBI[year] IN VALUES (crime_summary_PBI[year])
21        )
22    )
23 )
```

Rysunek 87. Wartość miernika dla określonej kategorii - brak filtracji = średnia, filtracja = suma dla filtrowanej dzielnicy

Źródło: Microsoft Power BI

Ostatnią z miar jest miara odpowiadająca za formatowanie warunkowe. Kolor miernika promieniowego zależy od tego, czy wartość przekracza linię celu oraz czy wartość osiąga maksymalny pułap. Pomocna przy tworzeniu owej miary jest funkcja SWITCH.

```
1 KradziezeCOLOR =
2 SWITCH(
3     TRUE(),
4     crime_summary_PBI[Kradzieże] >= [KradzieżeMAX], "#2C3532",
5     crime_summary_PBI[Kradzieże] >= [KradzieżeTARGET], "#D8B08C",
6     "#566862"
7 )
```

Rysunek 88. Miara - zmiana koloru w zależności od liczby przestępstw w wybranej kategorii

Źródło: Microsoft Power BI

⁴⁷ S. Cagliari, *Exploring the Filter Context with DAX functions*, 2022, [w:] towardsdatascience.com, <https://towardsdatascience.com/exploring-the-filter-context-with-dax-functions-422211c1118e>, [dostęp 23.05.2024].

Wszystkie miary załadowane do miernika promieniowego w Power BI tworzą komplementarną całość. Proces ten należy w sposób analogiczny odtworzyć dla wszystkich siedmiu odrębnych kategorii, a następnie w celu uproszczenia nawigacji po dashboardzie umieścić je w odpowiednich folderach. Łącznie w celu stworzenia karty zawierającej mierniki promieniowe na dole strony ‘Przestępcość’ zostało użytych 29 miar (nie 35, ponieważ jedna z miar = 0 została użyta siedmiokrotnie).

Karta ‘Przestępcość’, poniżej mapy, zawiera dwa kolejne mierniki promieniowe o nazwach ‘Liczba przestępstw’ oraz ‘Liczba przestępstw na 1000 osób’. Jak nazwy wskazują, pierwszy z nich pokazuje liczbę przestępstw z podziałem na dzielnice wybierane z mapy, a drugi ową liczbę podzieloną przez liczbę tysięcy. Istotną kwestią jest ilościowy punkt odniesienia w przypadku przestępstw, z racji na różnice pomiędzy dzielnicami w liczbie mieszkańców, czy nawet powierzchni, która ową liczbę często determinuje. Przykładem może być dzielnica Croydon dla której liczba przestępstw wykracza poza średnią Londynu dla roku 2019, jednak miernik kalkulowany na 1000 osób wskazuje poniżej średniej.



Rysunek 89. Mierniki promieniowe - liczba przestępstw oraz liczba przestępstw na 1000 osób dla dzielnicy Croydon

Źródło: Microsoft Power BI

Obiektem zainteresowania jest opisanie procesu powstania drugiej z tych miar, a mianowicie miernika promieniowego o nazwie ‘Liczba przestępstw na 1000 osób’, z racji na potrzebę zaangażowania danych z różnych źródeł zawartych w modelu oraz faktu, iż miernik ‘Liczba przestępstw’ wynika de facto z tego procesu, tylko bez uwzględnienia podziału na wielkość populacji. Proces ten jest jednym z najbardziej zaawansowanych procesów zastosowanych w pracy magisterskiej wykorzystujących język DAX.

Stworzenie miar odpowiedzialnych za zwracanie liczby przestępstw na 1000 osób wymaga wykonania dzielenia przez wielkość populacji. Proces ten zależny jest od

kontekstu. Dla jednych z miar, mianownikiem w dzieleniu powinna być wielkość populacji dla wybranej na mapie dzielnicy, a dla drugich - wielkość populacji dla Londynu ogółem. Należy więc stworzyć miary, które spełnią owe założenia. z pomocą przychodzi funkcja SUMMARIZE, tworząca sumaryczną tabelę na podstawie sumy wielkości populacji. Funkcja FILTER natomiast wybiera maksymalny rok z wcześniej określonych lat na suwaku w zmiennej VAR ‘MaxYear’, co jest powszechnie stosowanym rozwiązaniem w pracy magisterskiej. Następnie należy okalać kod funkcją CALCULATE, odpowiadającą za umożliwienie zmiany kontekstu oraz dokonania filtracji, oraz funkcji AVERAGEX. Funkcja ta wyświetla średnią wartość z sumy wielkości populacji dla MaxYear z wybranego na suwaku i działa tylko w mierze oznaczonej 2B, z racji na zablokowaną filtrację. Miara 1B z kolei wykorzystywana jest tylko w przypadku wyboru dzielnicy na mapie, więc wyświetlana jest wybrana z sum wewnętrz SUMMARIZE.

```

1 Wielkość populacji pod CRIME =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kalkulacje =
5
6     CALCULATE(
7         AVERAGEX(
8             SUMMARIZE(
9                 housing_yearly_GSS_Code_PBI,
10                housing_yearly_GSS_Code_PBI[year],
11                housing_yearly_GSS_Code_PBI[areal],
12                "Liczba populacji", SUM(housing_yearly_GSS_Code_PBI[population_size])),
13                [Liczba populacji]),
14                FILTER(
15                    housing_yearly_GSS_Code_PBI,
16                    housing_yearly_GSS_Code_PBI[year] = MaxYear)
17            )
18
19 RETURN
20 Kalkulacje
21

```



```

1 Wielkość populacji pod CRIME niefiltrowane =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kalkulacje =
5 CALCULATE(
6     CALCULATE(
7         AVERAGEX(
8             SUMMARIZE(
9                 housing_yearly_GSS_Code_PBI,
10                housing_yearly_GSS_Code_PBI[year],
11                housing_yearly_GSS_Code_PBI[areal],
12                "Liczba populacji", SUM(housing_yearly_GSS_Code_PBI[population_size])),
13                [Liczba populacji]),
14                FILTER(
15                    housing_yearly_GSS_Code_PBI,
16                    housing_yearly_GSS_Code_PBI[year] = MaxYear)
17            ),
18            ALL(GIS_London_Polygons_PBI[NAME])
19        )
20 RETURN
21 Kalkulacje
22

```

Rysunek 90. Miara DAX obliczająca wielkość populacji przy filtracji lub jej braku

Źródło: Microsoft Power BI

Otrzymane tym sposobem miary posłużą jako mianownik w procesie dzielenia przez liczbę przestępstw. w pierwszej kolejności powstaną punkty odniesienia dla danych. Punktami odniesienia będzie wartość minimalna, maksymalna oraz średnia liczba przestępstw Londynie z podziałem na dzielnice. Za wartość minimalną zostało uznane 0, gdyż nie jest możliwe wystąpienie ujemnej liczby przestępstw. Wartością maksymalną z kolei jest liczba przestępstw dla najniebezpieczniejszej z Londyńskich dzielnic dla wybranego na suwaku roku. Wartość maksymalna zmienia się w zależności od doboru lat na suwaku. Miara DAX znajduje więc maksymalną liczbę przestępstw dla dzielnicy w maksymalnym wybranym roku. Tym samym sposobem powstaje miara

zwracającą średnią liczbę przestępstw z tą różnicą, iż użyta zostanie funkcja arytmetyczna AVERAGEX zamiast MAXX. z użyciem funkcji DIVIDE należy przedzielić wyniki przez wielkość populacji.

```

1 Maks Przestępstw Ogółem per Capita =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kalkulacje =
5 CALCULATE(
6     CALCULATE(
7         MAXX(
8             SUMMARIZE(
9                 crime_summary_PBI,
10                crime_summary_PBI[year],
11                crime_summary_PBI[borough],
12                "Srednia ilosc przestepstw", SUM(crime_summary_PBI[Liczba_przestepstw])),
13                [Srednia ilosc przestepstw]),
14                FILTER(
15                    crime_summary_PBI,
16                    crime_summary_PBI[year] = MaxYear)
17            ),
18            ALL(GIS_London_Polygons_PBI[NAME])
19        )
20 RETURN
21 DIVIDE(
22     (Kalkulacje * 1000),
23     housing_yearly_GSS_Code_PBI[Wielkość populacji pod CRIME niefiltrowane], 0)
24 
```



```

1 Srednia Przestępstw Ogółem TARGET per Capita =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kalkulacje =
5 CALCULATE(
6     CALCULATE(
7         AVERAGEX(
8             SUMMARIZE(
9                 crime_summary_PBI,
10                crime_summary_PBI[year],
11                crime_summary_PBI[borough],
12                "Srednia ilosc przestepstw", SUM(crime_summary_PBI[Liczba_przestepstw])),
13                [Srednia ilosc przestepstw]),
14                FILTER(
15                    crime_summary_PBI,
16                    crime_summary_PBI[year] = MaxYear)
17            ),
18            ALL(GIS_London_Polygons_PBI[NAME])
19        )
20 RETURN
21 DIVIDE(
22     (Kalkulacje * 1000),
23     housing_yearly_GSS_Code_PBI[Wielkość populacji pod CRIME niefiltrowane], 0)
24 
```

Rysunek 91. Miary obliczające średnią oraz maksymalną liczbę przestępstw

Źródło: Microsoft Power BI

Niestety okazuje się, iż rezultaty miary odpowiadającej za zwracanie maks są błędne. Miara 3A, bez użycia funkcji DIVIDE po RETURN, pozostaje jednak w modelu i służy zwracaniu wartości maksymalnej dla miernika promieniowego, bez podziału na wielkość populacji. Błędne wartości wynikają z przedzielenia liczby przestępstw w danych dzielnicach przez średnią liczbę ludności w Londynie. w przypadku dzielnicy Westminster, będącej jednocześnie najniebezpieczniejszą dzielnicą Londynu rokrocznie, podzielenie liczby przestępstw przez średnią liczbę ludności ogółem zwraca wartość mniejszą niż stan faktyczny, co sprawia, że miernik wyświetla dane większe niż wartość maksymalna. Nie istnieje jednak możliwość zastosowania wielkości populacji bez zablokowanej filtracji, ze względu na zmienność wyniku w zależności od wyboru dzielnicy na mapie. Jest to niedozwolone ze względu na charakter maksymalnej wartości, która służy jako punkt odniesienia. Rozwiązaniem jest wyszukanie wartości maksymalnych dopiero po wykonaniu podziału. Aby to osiągnąć, należy bazować na fizycznych danych, a nie na wirtualnych miarach, ponieważ silnik DAX nie zinterpretuje rezultatu dzielenia. Podjęto więc decyzję o stworzeniu tabeli zawierającej wartości maksymalne dzielenia dla wszystkich lat z przedziału 2011-2019. Przygotowana w ten sposób tabela zostaje zaimportowana do modelu, z nawiązaniem relacji pomiędzy tabelą Calendar_Year_PBI pełniącą rolę kalendarza.

```

;WITH cte1 AS(
    SELECT borough, GSS_CODE, year, SUM(count_crime) AS suma_przestepstw
    FROM dbo.crime_summary_PBI
    GROUP BY borough, year, GSS_CODE),
cte2 AS ( SELECT area, GSS_CODE, population_size, year
    FROM dbo.housing_yearly_GSS_Code_PBI),
cte3 AS (
    SELECT (suma_przestepstw * 1000)/population_size AS dzialanie, borough, c1.year
    FROM cte1 c1
    JOIN cte2 c2 ON c1.GSS_CODE = c2.GSS_CODE and c1.year = c2.year)
SELECT MAX(dzialanie) AS maksymalna_wartosc, year
FROM cte3
GROUP BY year
ORDER BY year ASC;

```

Rysunek 92. Rozwiązywanie problemu maksymalnej wartości mniejszej niż realna w mierniku - liczba przestępstw na 1000 osób

Źródło: Microsoft Power BI

	maksymalna_wartosc	year
1	216.3838566002677815121	2011
2	216.4586478928606527352	2012
3	212.1133304825847179301	2013
4	232.5111876961061673782	2014
5	244.4335304726804485367	2015
6	217.4836640900756823119	2016
7	194.3005604666742920635	2017
8	185.6268897557613072018	2018
9	181.7928772399681025533	2019

Rysunek 93. Maksymalne liczby przestępstw na 1000 osób w dzielnicach Londynu na przestrzeni 8 lat w wybranych latach

Źródło: Microsoft Power BI

Następnie przy pomocy miary ‘Liczba przestępstw 1000 os MAX’ zostaje przypisana wartość maksymalna na mierniku promieniowym w zależności od wyboru przedziału lat na suwaku dostępnym w karcie.

3A+

```
1 Liczba przestępstw 1000 os MAX =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaksYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR obliczenie =
5     CALCULATE(
6         MAX (Maksima_Crime_Capita[maksymalna_liczba_przestepstw_per_Capita]),
7         FILTER(Maksima_Crime_Capita,
8             |   Maksima_Crime_Capita[year] = MaksYear))
9 RETURN
10 obliczenie
```

Rysunek 94. Odniesienie się w mierze do wartości maksymalnych policzonych w środowisku SQL Server

Źródło: Microsoft Power BI

Istotnym elementem miernika promieniowego jest centralny punkt wyświetlający wartość. W tym wypadku jest to liczba przestępstw na 1000 osób w wybranej dzielnicy. Co jednak, gdy nie ma wybranej dzielnicy? Należy przewidzieć taką sytuację, ponieważ domyślnie karta nie jest filtrowana. Dopiero gdy użytkownik zaczyna z niej korzystać i dokonuje wyboru dzielnic na mapie, funkcjonuje filtrowanie. Pomocna w rozwiązyaniu tego problemu jest wcześniej używana kombinacja funkcji IF oraz ISFILTERED, która w zależności od filtracji zwraca sumę liczby przestępstw dla wybranej dzielnicy, gdy filtracja wystąpi, lub średnią liczbę przestępstw w przypadku braku filtracji dla wszystkich dzielnic. Mianownikiem w owym działaniu jest wielkość populacji z swobodną filtracją, czyli zależną od wyboru na mapie. Działanie w drugiej części funkcji IF wymaga podzielenia przez 33 (liczbę dzielnic) w celu uzyskania średniej z całkowitej sumy populacji w Londynie.

1A

```

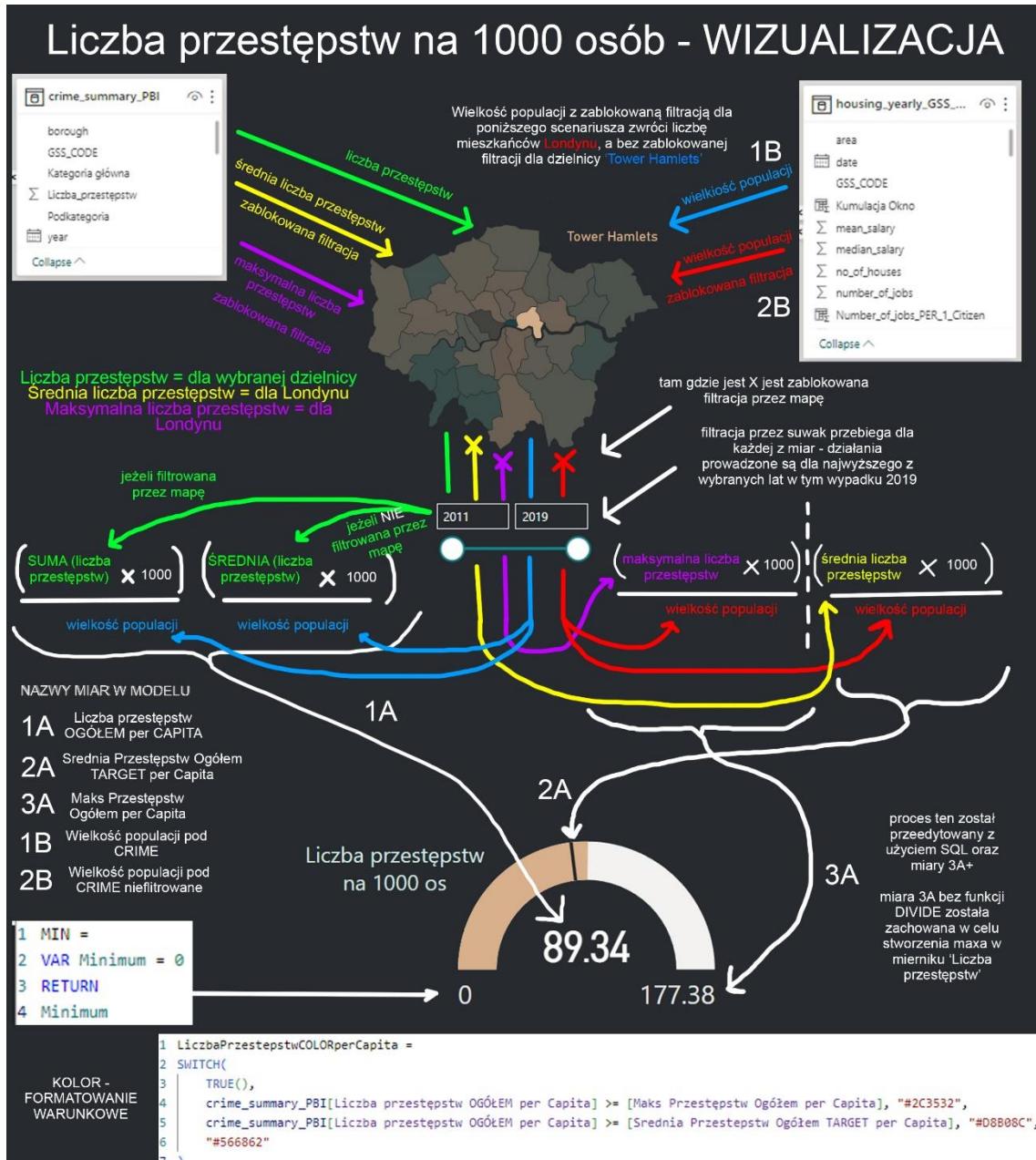
1 Liczba przestępstw OGÓŁEM per Capita =
2 VAR SelectedYears = VALUES('crime_summary_PBI'[year])
3 VAR MaxYear = MAXX(SelectedYears, 'crime_summary_PBI'[year])
4 VAR Kalkulacje =
5 IF(
6     ISFILTERED(GIS_London_Polygons_PBI[NAME]) || ISFILTERED('crime_summary_PBI'[year]),
7     CALCULATE(
8         SUM('crime_summary_PBI'[Liczba_przestepstw]),
9         FILTER(
10            'crime_summary_PBI',
11            'crime_summary_PBI'[year] = MaxYear
12        )
13    ),
14    CALCULATE(
15        DIVIDE(
16            AVERAGEX(
17                SUMMARIZE(crime_summary_PBI,
18                    crime_summary_PBI[year],
19                    "Srednia ilosc przestepstw", SUM(crime_summary_PBI[Liczba_przestepstw])),
20                    [Srednia ilosc przestepstw]),
21                    33, 0),
22        FILTER(
23            crime_summary_PBI,
24            crime_summary_PBI[year] = MaxYear
25        )
26    )
27 )
28 RETURN
29 DIVIDE(
30     (Kalkulacje * 1000),
31     housing_yearly_GSS_Code_PBI[Wielkość populacji pod CRIME], 0)

```

Rysunek 95. Miara liczba przestępstw ogółem na 1000 osób - wynik zależny od istnienia filtracji

Źródło: Microsoft Power BI

Jako dopełnienie procesu budowy miernika należy użyć prostej miary zwracającej zero jako wartość minimalną, oraz funkcji odpowiedzialnej za formatowanie warunkowe, z zastosowaniem funkcji SWITCH. Wartości powyżej średniej, poniżej średniej lub osiągające maksymalną wartość miernika mają przypisane kolory za pomocą formatowania warunkowego. Do stworzenia miernika promieniowego niezbędne stało się użycie 7 miar. Likwidując procesy odpowiedzialne za dzielenie przez wielkość populacji po deklaracji RETURN na końcu miar, otrzymywane są miary, które spełniają rolę budowy miernika promieniowego bez podziału na liczbę mieszkańców Londynu. Poniżej znajduje się schemat obrazujący cały proces.



Rysunek 96. Objasnenie komplementarnego rozwiæzania budowy miernika promieniowego - liczba przestepstw na 1000 osób

Źródło: Microsoft Power BI

4.6 Karta korelacji

Dashboard magisterski zawiera dwie karty z korelacjami. Jedna z nich zawiera korelacje dodatnie, a druga ujemne. Korelacje te umożliwiają użytkownikowi dashboardu zrozumienie istnienia pewnych często fundamentalnych zależności. Celem owych kart jest zobrazowanie najważniejszych korelacji wynikających z dostępnych danych, a następnie ich zinterpretowanie. Współzależności powstały przy użyciu tabeli centralnej,

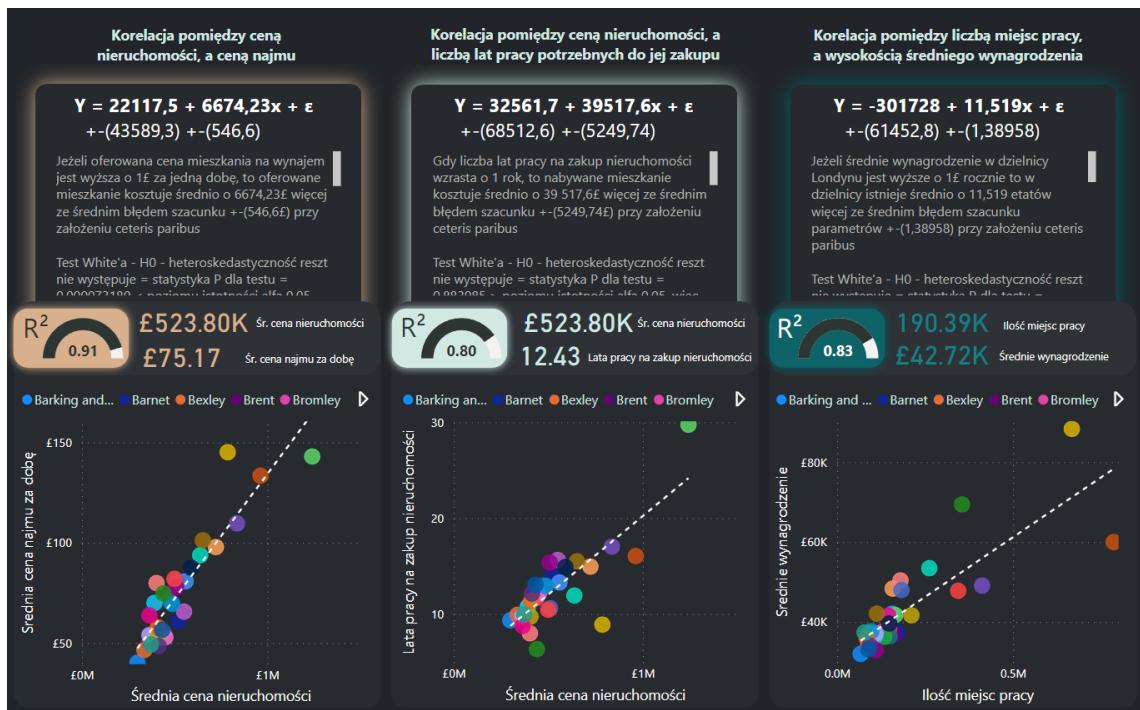
która została zmodyfikowana w tym celu. Dodane zostały zagregowane wartości mierników wykorzystywanych w kartach dashboardu pracy magisterskiej. Statystyka współczynnika determinacji pochodzi z obliczeń języka DAX w środowisku Power BI. z środowiska Power BI pochodzi również sama wizualizacja. Interpretacja została opracowana na podstawie testów wykonanych w oprogramowaniu GRETL oraz Statistica. Korelacje powstały na podstawie danych przekrojowych z jedną zmienną objaśniającą dla roku 2019.

Istnieje dodatnia korelacja pomiędzy średnią ceną nieruchomości w danej dzielnicy, a średnią ceną najmu krótkoterminowego w niej. Wynika to z kalkulacji ROI z najmu, który nieruchomość powinna spełnić, aby inwestor otrzymał oczekiwany stopę zwrotu. Jeżeli oferowana cena mieszkania na wynajem jest wyższa o 1£ za jedną dobę, to oferowane mieszkanie kosztuje średnio o 6674,23£ więcej ze średnim błędem szacunku $+(546,6\text{£})$ przy założeniu ceteris paribus. Test White'a - H0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = 0,000073180 < poziomu istotności alfa 0.05. Należy odrzucić hipotezę zerową na rzecz alternatywnej, więc heteroskedastyczność występuje, co jest niekorzystne dla modelu. Test normalności rozkładu reszt - H0 - składnik losowy ma rozkład normalny = statystyka P dla testu = 0,000889171 < od poziomu istotności alfa 0,05. Należy odrzucić hipotezę zerową na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - H0 = p > poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = 0,00000000000222 i jest mniejsza od alfa. Odrzucić należy hipotezę zerową na rzecz alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna średnia cena nieruchomości objaśnia model w 91%.

Występuje dodatnia korelacja pomiędzy średnią ceną nieruchomości w dzielnicach Londynu, a latach pracy potrzebnych do ich zakupu. Ta korelacja nie jest oczywista, gdyż w dzielnicach charakteryzujących się wyższymi cenami nieruchomości występują lepiej opłacane prace. Jednak widełki płacowe posiadają mniejszą rozpiętość niż widełki cen nieruchomości, co oznacza, że nawet wysokie wynagrodzenia nie zawsze rekompensują koszty zakupu drogich nieruchomości. Rezultatem jest wzrost popularności dzielnic sypialnianych. Gdy liczba lat pracy na zakup nieruchomości wzrasta o 1 rok, to nabywane mieszkanie kosztuje średnio o 39517,6£ więcej ze średnim błędem szacunku $+(5249,74\text{£})$ przy założeniu ceteris paribus. Test White'a - H0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = 0,883985

> poziomu istotności alfa 0,05. Istnieje brak podstaw do odrzucenia hipotezy zerowej. Heteroskedastyczność reszt nie występuje - homoskedastyczność jest korzystna dla modelu. Test normalności rozkładu reszt - H_0 - składnik losowy ma rozkład normalny = statystyka P dla testu = $0,0000129493 <$ od poziomu istotności alfa 0,05. Należy odrzucić hipotezę zerową na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - $H_0 = p >$ poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = $1,75e-08$ i jest mniejsza od alfa. Odrzucana jest hipoteza zerowa na rzecz alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna średnia cena nieruchomości objaśnia model w 80%.

Istnieje dodatnia korelacja pomiędzy liczbą miejsc pracy, a wysokością średniego wynagrodzenia w dzielnicach Londynu. Dzielnice, w których istnieje znacząca liczba miejsc pracy, często są siedzibami międzynarodowych firm i korporacji, które konkurują o kapitał ludzki. Ten konkurencyjny kontekst przyczynia się między innymi do wzrostu wynagrodzeń, który jest uzależniony od liczby dostępnych miejsc pracy. Jeżeli średnie wynagrodzenie w dzielnicy Londynu jest wyższe o 1£ rocznie to w dzielnicy istnieje średnio o 11,519 etatów więcej ze średnim błędem szacunku parametrów $+-(1,38958)$ przy założeniu ceteris paribus. Test White'a - H_0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = $0,0182198 <$ poziomu istotności alfa 0,05. Należy odrzucić hipotezę zerową na rzecz alternatywnej. Heteroskedastyczność występuje, co jest niekorzystne dla modelu. Jednak przy założeniu bardziej restrykcyjnego poziomu istotności alfa jak np. 0,01 istnieje możliwość przyjęcia H_0 , a co za tym idzie wykluczenia heteroskedastyczności i potwierdzenia homoskedastyczności. Test normalności rozkładu reszt - H_0 - składnik losowy ma rozkład normalny = statystyka P dla testu = $6,17664e-007 <$ od poziomu istotności alfa 0,05. Odrzucana jest hipoteza zerowa na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - $H_0 = p >$ poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = $2,31e-09$ i jest mniejsza od alfa. Odrzucana jest H_0 na rzeczy hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna liczba miejsc pracy objaśnia model w 83%.



Rysunek 97. Korelacje dodatnie

Źródło: Microsoft Power BI

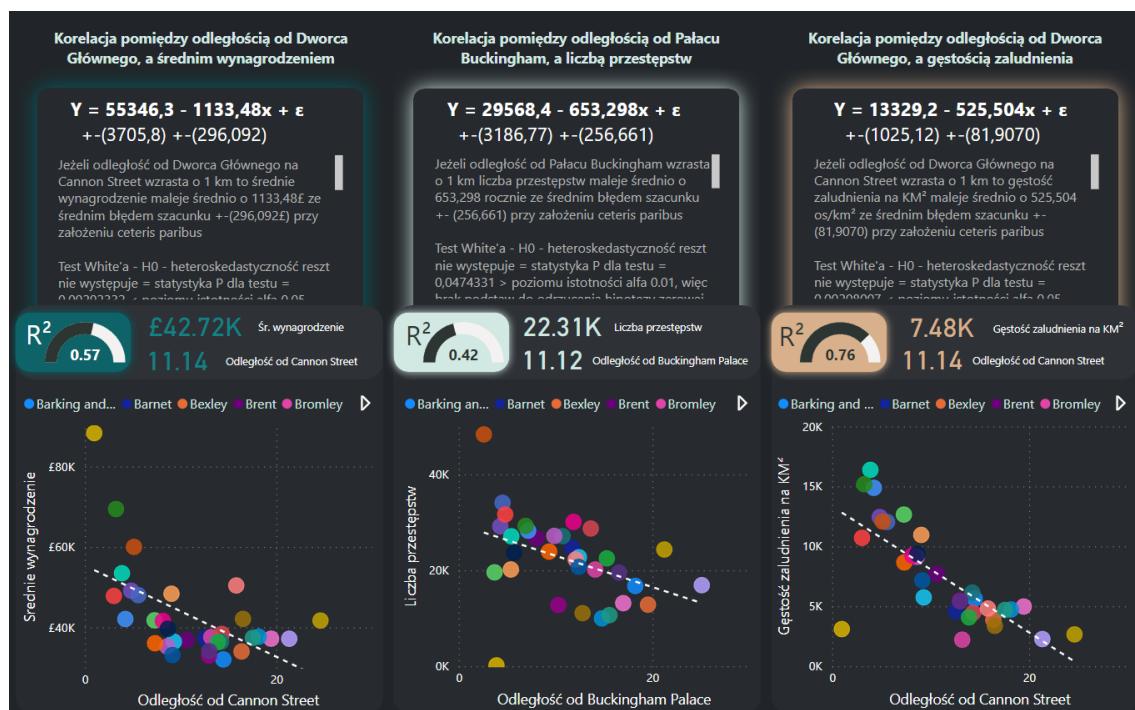
Im więcej miejsc pracy w dzielnicy tym wyższe średnie wynagrodzenie, więc które z dzielnic zawierają najwięcej miejsc pracy i najwyższe średnie wynagrodzenia? Okazuje się, że istnieje ujemna korelacja między odległością od dworca głównego, znajdującego się w sercu Londynu, a wysokością średnich wynagrodzeń. Oczywiście, nie jest to reguła bez wyjątków, ponieważ istnieją sytuacje, takie jak lotnisko w Heathrow, które jest jednym z największych pracodawców w mieście, mimo że znajduje się na obrzeżach miasta. Tego typu obiekty działają proaktywnie na kształtowanie się wynagrodzeń w pobliskiej okolicy. Jeżeli odległość od Dworca Głównego na Cannon Street wzrasta o 1 km to średnie wynagrodzenie maleje średnio o 1133,48£ ze średnim błędem szacunku +-(296,092£) przy założeniu ceteris paribus. Test White'a - H0 - heteroskedastyczność reszt nie występuje = statystyka P dla testu = 0,00292332 < poziomu istotności alfa 0.05. Należy odrzucić hipotezę zerową na rzecz alternatywnej, więc heteroskedastyczność występuje, co jest niekorzystne dla modelu. Test normalności rozkładu reszt - H0 - składnik losowy ma rozkład normalny = statystyka P dla testu = 0,000696101 < od poziomu istotności alfa 0,05. Odrzucana jest hipoteza zerowa na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - H0 = p > poziomu istotności alfa 0,05.

Statystyka P dla testu wynosi = 0,0006 i jest mniejsza od alfa. Odrzucamy H₀ na rzecz hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R² - zmienna Odległość od Cannon Street objaśnia model w 57%.

Pałac Buckingham oraz inne zabytki w dzielnicy Westminster przyciągają znaczną liczbę turystów, a sama dzielnica przoduje w liczbie przestępstw w niemal każdej kategorii. Nie jest zaskoczeniem istnienie ujemnej korelacji pomiędzy liczbą przestępstw, a wzrostem odległości od Pałacu Buckingham. Jeżeli odległość od Pałacu Buckingham wzrasta o 1 km liczba przestępstw maleje średnio o 653,298 rocznie ze średnim błędem szacunku +- (256,661) przy założeniu ceteris paribus. Test White'a - H₀ - heteroskedastyczność reszt nie występuje = statystyka P dla testu = 0,0474331 > poziomu istotności alfa 0,01. Brak podstaw do odrzucenia hipotezy zerowej. Heteroskedastyczność nie występuje, co jest korzystne dla modelu. Test normalności rozkładu reszt - H₀ - składnik losowy ma rozkład normalny = statystyka P dla testu = 0,00144604 < od poziomu istotności alfa 0,01. Należy odrzucić hipotezę zerową na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - H₀ = p > poziomu istotności alfa 0,05. Statystyka P dla testu wynosi = 0,0161 i jest większa od alfa 0,01, ale mniejsza od poziomu istotności alfa 0,05. Odrzucana jest H₀ na rzecz hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej, **przy przyjęciu poziomu istotności na poziomie 0,05. Współczynnik determinacji R² - zmienna Odległość od Buckingham Palace objaśnia model w 42%.

Bliskość centrum miasta często ściśle powiązana jest z lepszą komunikacją miejską, większymi możliwościami zatrudnienia czy nauki oraz pozostałymi czynnikami, które wpływają na atrakcyjność samego centrum i okolic. Czynniki te przekładają się na wysoką gęstość zaludnienia w centrum. Zbadano zatem korelację, która ukazuje ujemną zależność pomiędzy odlegością od dworca głównego w City of London, a gęstością zaludnienia. Jeżeli odległość od Dworca Głównego na Cannon Street wzrasta o 1 km to gęstość zaludnienia na km² maleje średnio o 525,504 os/km² ze średnim błędem szacunku -(81,9070) przy założeniu ceteris paribus. Test White'a - H₀ - heteroskedastyczność reszt nie występuje = statystyka P dla testu = 0,00208007 < poziomu istotności alfa 0,05. Należy odrzucić hipotezę zerową na rzecz alternatywnej. Heteroskedastyczność występuje, co jest niekorzystne dla modelu. Test normalności rozkładu reszt - H₀ - składnik losowy ma rozkład normalny = statystyka P dla testu = 0,00171414 < od

poziomu istotności alfa 0,05. Odrzucana jest hipoteza zerowa na rzecz alternatywnej. Składnik losowy nie ma rozkładu normalnego. Test na indywidualną istotność parametru strukturalnego Beta - $H_0 = p >$ poziomu istotności alfa 0,05. Statystyka P dla testu wynosi $= 3,78e-07$ i jest mniejsza od alfa. Odrzucana jest H_0 na rzeczy hipotezy alternatywnej. Parametr istotnie różni się od 0, czyli wpływa istotnie statystycznie na poziom zmiennej objaśnianej. Współczynnik determinacji R^2 - zmienna odległość od Cannon Street wyjaśnia model w 76%.



Rysunek 98. Korelacje ujemne

Źródło: Microsoft Power BI

4.7 Karta miernik syntetyczny

Ostatnią kartę dashboardu magisterskiego stanowi karta ukazująca miernik syntetyczny dla dzielnic Londynu z wnioskami z niego wynikającymi. Miernik zbudowany został w oparciu o efekty miar będących elementami samego dashboardu. Proces tworzenia miernika syntetycznego polega na zgrupowaniu mierników dotyczących w tym przypadku dzielnic Londynu, a następnie, przy pomocy przekształceń, stworzenie zmiennych syntetycznych zawierających się w przedziale pomiędzy 0 a 1. Podczas procesu przekształceń należy określić, czy mierniki stanowią stymulantę, destymulantę, czy też nominantę. Następnie, po dokonaniu przekształceń

należy wykonać średnią arytmetyczną z wszystkich otrzymanych wyników dla poszczególnych dzielnic⁴⁸. Im bliżej jedności znajdująć się będzie rezultat, tym wyższa jest atrakcyjność dzielnicy.

Przy określaniu, które mierniki będą stanowić stymulantę, destymulantę, czy też nominantę, należy uwzględnić założony punkt widzenia. Celem miernika jest ukazanie możliwie najbardziej atrakcyjnych dzielnic do zakupu nieruchomości na cele mieszkaniowe przez parę DINK, świadczącą stosunek pracy w przedsiębiorstwie z branżą finansową, która planuje po kilku latach wyprowadzkę na obrzeża Londynu i wynajęcie nabytej nieruchomości w mieście. W skład miernika syntetycznego wchodzą miary:

- odległość od Pałacu Buckingham (destymulanta),
- liczba przestępstw (destymulanta),
- średnie wynagrodzenie (stymulanta),
- liczba nieruchomości (stymulanta),
- mediana wynagrodzeń (stymulanta),
- średnia cena nieruchomości (destymulanta),
- średnia cena najmu za dobę (stymulanta),
- średni miesięczny zysk z najmu (stymulanta),
- gęstość zaludnienia na kilometr kwadratowy (stymulanta),
- liczba miejsc pracy na kilometr kwadratowy (stymulanta),
- lata pracy potrzebne na zakup nieruchomości (destymulanta),
- liczba wakatów pracy na 1 osobę (stymulanta).

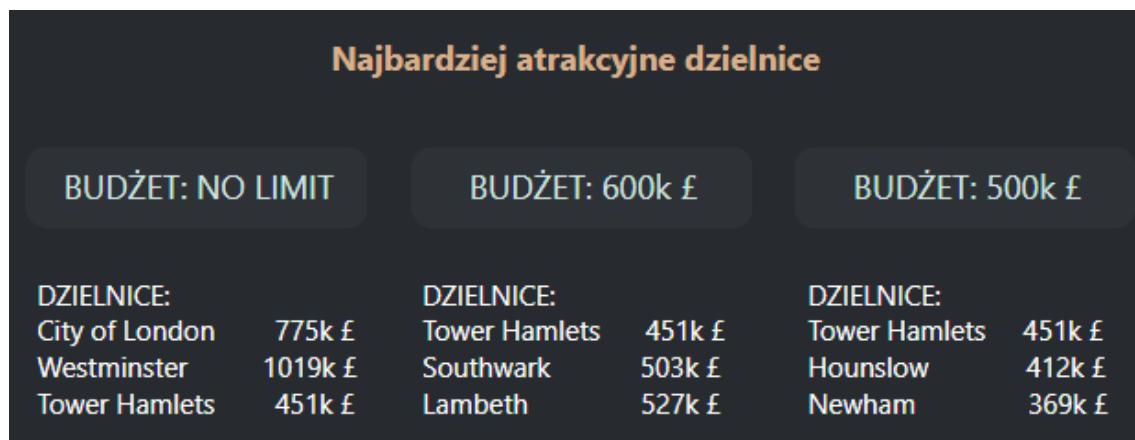
Im wyższa odległość od pałacu Buckingham, tym gorzej z punktu widzenia pary chcącej dokonać zakupu nieruchomości, ponieważ chcieliby ją w przyszłości wynajmować turystom. Dlatego uwzględniono odległość od Pałacu Buckingham, a nie od centrum Londynu, ponieważ to właśnie odległość od dzielnicy Westminster wpływa na ceny najmu w większym stopniu niż odległość od jakiejkolwiek innej dzielnicy Londynu. Wysoka liczba przestępstw jest niekorzystna zarówno z punktu widzenia zamieszkiwania przez właścicieli, jak i wynajmu, dlatego ta miara jest destymulantą. Średnie wynagrodzenie oraz mediana wynagrodzeń, im wyższe, tym lepsze, więc jako zmienne syntetyczne, te mierniki są stymulantami. Liczba nieruchomości stanowi

⁴⁸ K. Kompa, *Budowa mierników agregatowych do oceny poziomu rozwoju społeczno-gospodarczego*, s. 16.

stymulantę z racji na prostszy proces zakupu nieruchomości oraz większą konkurencję na rynku. Im więcej nieruchomości w dzielnicy tym lepiej z perspektywy kupujących. Średnia cena najmu oraz miesięczny zysk z najmu są stymulantą, gdyż para DINK zamierza czerpać korzyści finansowe z nieruchomości w przyszłości. Gęstość zaludnienia oraz liczba miejsc pracy na kilometr kwadratowy również jest stymulantą, gdyż im większe są owe mierniki tym większa dostępność usług, przedsiębiorstw czy komunikacji w pobliżu. Lata pracy potrzebne na zakup nieruchomości jako zmienna syntetyczna będzie destymulantą, gdyż im wyższa liczba lat pracy potrzebna do zakupu nieruchomości tym ceny nieruchomości są nieadekwatne wysokie w stosunku do możliwości zarobkowych. Liczba wakatów na 1 osobę będzie stymulantą, gdyż miara pokazuje, że im więcej wakatów per capita tym wyższe zarobki, ale i łatwiej jest dokonać zmiany pracy.

Wszystkie mierniki zostały zebrane w postaci tabeli z wykorzystaniem miar DAX w środowisku Power BI, a następnie wyeksportowane do skoroszytu w programie Microsoft Excel. Pierwszym krokiem była normalizacja z użyciem odchylenia standardowego, średniej dla każdego z mierników oraz funkcji ‘normalizuj’ w Microsoft Excel. Kolejnym krokiem było określenie wartości minimalnych i maksymalnych w każdym z mierników. Następnie, na podstawie tych wartości, zostały zastosowane wzory na przekształcenia destymulant oraz stymulant. Ze wszystkich powstałych w ten sposób zmiennych cząstkowych brana jest średnia arytmetyczna z podziałem na 33 dzielnice Londynu, która tworzy zmienne syntetyczne, w całości stanowiące miernik syntetyczny. Rezultat działań w oprogramowaniu Microsoft Excel zostaje wyeksportowany w postaci pliku płaskiego, a następnie zimportowany do modelu danych w środowisku Power BI.

Karta ‘Miernik syntetyczny’, będąca częścią dashboardu magisterskiego, zawiera mapę Londynu ukazującą wielkość zmiennych syntetycznych dla poszczególnych dzielnic, mieszczących się w przedziale pomiędzy 0 a 1. Znajduje się na niej również miernik promieniowy, pomagający określić stosunek danego miernika wobec pozostałych dzielnic oraz tabela. Po prawej stronie karty opisany jest punkt widzenia, będący determinantą przy określaniu stymulant i destymulant w mierniku. Prawy dolny róg zajmuje z kolei rekomendacja dzielnic do zakupu nieruchomości i zamieszkania w określonej dzielnicy przez parę DINK. Rekomendacja ta opiera się na możliwościach finansowych pary.



Rysunek 99. Najbardziej atrakcyjne dzielnice do zamieszkania przy określonym budżecie na zakup nieruchomości

Źródło: Microsoft Power BI

Za pomocą miar oraz kolumn kalkulacyjnych napisanych w języku DAX opracowano gotowe rozwiązania, umożliwiające stworzenie odpowiednich wizualizacji w Power BI. Wizualizacje te, wraz z licznymi kartami, składają się na dashboard magisterski. W tej części pracy niezbędne jest użycie języka SQL, co pozwoliło na edycję modelu oraz przetwarzanie danych w sposób umożliwiający tworzenie wizualizacji. Wszystkie miary zostały posegregowane w odpowiednich folderach w celu zachowania przejrzystości pliku pbix. Do interpretacji korelacji na kartach wykorzystano oprogramowanie GRETL, a do wyznaczania odległości pomiędzy współrzędnymi – język Python wraz z odpowiednimi bibliotekami.

ZAKOŃCZENIE

Funkcjonowanie City of London jako odrębnej jednostki demokratycznej wewnątrz Londynu, oraz korzystny dla miasta historyczny splot wydarzeń, który spowodował napływ finansistów z Amsterdamu, składają się na to, że Londyn jest wyjątkowym miastem nie tylko w skali Europy, ale i świata. Jego międzynarodowa pozycja finansowa, kulturowa oraz turystyczna przyciąga rzesze turystów z całego globu i tworzy miejsca pracy. Niestety, międzynarodowy charakter Londynu i jego wyjątkowość jeszcze bardziej pogłębiają istniejący już kryzys mieszkaniowy.

Pierwsza znacząca zmiana na rynku nieruchomości, nie tylko w Londynie, ale także w całej Wielkiej Brytanii, miała miejsce w latach 80. ubiegłego wieku. Patrząc z perspektywy czasu, wprowadzona ustawa mieszkaniowa „Right to Buy”, umożliwiająca wykupywanie socjalnych mieszkań za połowę ich wartości, miała istotny wpływ na wzrost cen nieruchomości oraz kosztów najmu w Londynie. Dodatkowo, w podobnym czasie rozwijała się bankowość, umożliwiająca dokonywanie zakupów nieruchomości na kredyt. Spowodowało to, że ceny nieruchomości zaczęły korelować z zdolnością kredytową mieszkańców, a nie z ich zdolnością oszczędzania, która zawsze była i nadal jest wyraźnie mniejsza. Ten trend utrzymuje się do dzisiaj, a rosnąca finansjalizacja Wielkiej Brytanii oraz traktowanie nieruchomości jako instrumentu finansowego, a nie podstawowego dobra, spotęgowała kryzys do granic możliwości.

Poza charakterystycznymi dla Londynu i Wielkiej Brytanii czynnikami determinującymi wzrost cen mieszkań, istnieje wiele uniwersalnych czynników, które wpływają na ceny w metropoliach, również w Londynie. Wyróżnić można takie czynniki jak gęstość zaludnienia, bliskość centrum czy obecność terenów zielonych. Wiele z tych czynników stało się przedmiotem zainteresowania w narracji prowadzonej w dashboardzie analitycznym.

Dashboard, powstały za pomocą narzędzi programistycznych i specjalistycznego oprogramowania, umożliwia dokonywanie dowolnej eksploracji rynku nieruchomości w Londynie. Dzięki przetworzeniu danych możliwe stało się uzyskanie istotnych informacji. Na przykład, dzielnica Westminster jest jedną z najbardziej atrakcyjnych turystycznie dzielnic Londynu, ze względu na wiele zabytków, atrakcji oraz historii. Dashboard ukazuje, że dzielnica ta posiada jedne z najwyższych cen najmu krótkoterminowego oraz nieruchomości. To właśnie w niej można osiągnąć największy zysk z najmu. Co ciekawe, dzielnica ta również rok rocznie jest niechlubnym liderem pod

względem liczby przestępstw.

Tower Hamlets z kolei charakteryzuje się wyjątkowo korzystnym stosunkiem ceny do jakości życia w porównaniu do innych dzielnic Londynu. Dzielnica ta jest siedzibą centrum korporacyjnego, zwanego Canary Wharf. Dzięki dashboardowi można również dowiedzieć się, że dzielnica ta nie ma charakteru sypialnego, ponieważ liczba miejsc pracy przewyższa liczbę mieszkańców. Dodając do tego fakt, że dzielnica ta znajduje się blisko samego centrum Londynu oraz możliwe jest znalezienie tutaj mieszkania do zakupu za połowę ceny, którą należałoby wydać w dzielnicy Westminster, inwestycja w dzielnicy Tower Hamlets wydaje się być opłacalna.

Kryzys nieruchomości w Londynie będzie narastał, a zmiana tej trajektorii bez załamania gospodarczego, konfliktu zbrojnego, kryzysu lub odejścia od nieruchomości jako dobra inwestycyjnego wydaje się mało prawdopodobna. Jeśli sytuacja będzie trwać, stosunek wynagrodzeń do cen nieruchomości będzie coraz bardziej rozbieżny. Bez naprawy rynku od strony podażowej i regulacji nieruchomości jako formy instrumentu finansowego, trudno spodziewać się poprawy. Nawet wyjście Wielkiej Brytanii z Unii Europejskiej nie wpłynęło pozytywnie na sytuację mieszkaniową przeciętnych Londyńczyków.

TITLE AND ABSTRACT

THE REAL ESTATE MARKET IN LONDON – AN ANALYTICAL DASHBOARD USING BUSINESS INTELLIGENCE TOOLS

Sebastian Masłowski 270049

The master's thesis describes the London real estate market using an analytical dashboard created in a Business Intelligence environment. The technologies used to create the dashboard include: Power BI, Power Query, DAX, SQL, Python with libraries, QGIS, Excel, and Gretl. The thesis consists of four chapters.

The first chapter is theoretical and describes the characteristics of the factors shaping the real estate market in London and in general. This chapter also includes the history of the development of Metropolitan London and the City of London, highlighting events that have determined the financial capital status of the United Kingdom.

The second chapter presents the methodological part. It lists the technologies and software used, as well as their role in building analytical solutions.

The third chapter begins the technical part of the thesis. The first part of this chapter describes the process of collecting data from various sources and maps. The second subsection concerns data cleaning, processing, and modeling. At this stage, missing data was generated using SQL scripts written for this purpose, employing multiple uses of CTE logic, window functions, nesting, and table joins. Python code was also used to measure the distance between central London or Buckingham Palace and individual rental properties.

The fourth chapter focuses on presenting the process of programming and building the dashboard in Power BI. This chapter is divided into seven subsections, each describing the construction of one of the dashboard's cards. At this stage of the work, all the technologies used throughout the thesis were employed, but the majority of the work consisted of codes created in the DAX language and operations in Power Query. Correlations were interpreted using the Gretl software. The result of the thesis is a dashboard enabling the exploration of the London real estate market.

BIBLIOGRAFIA

Pozycje zwarte

1. Adamson C., *The Complete Reference Star Schema*, The McGraw-Hill Companies, New York 2010.
2. Allington M., *Super Charge Power BI: Power BI is Better when You Learn to Write Dax*, Holy Macro Books, Merritt Island 2018.
3. Bakhshi S., *Expert Data Modeling with Power BI: Get the Best Out of Power BI by Building Optimized Data Models for Reporting and Business Needs*, Packt Publishing, Birmingham 2021.
4. Blakeley G., *Financialization, Real Estate and COVID-19 in the UK*, Oxford University Press and Community Development Journal, b.m. 2020.
5. Coispeau O., *Finance Masters: A Brief History of International Financial Centers in the Last Millennium*, World Scientific Publishing Co. Pte. Ltd, Singapore 2017.
6. Copley T., *From Right to Buy to Buy to Let*, Greater London Authority, London 2014.
7. Dudek-Klimiuk J., *History of Green Areas of Tychy Their Origins and Role in the Structure of the City*, Wydawnictwo Politechniki Krakowskiej, Kraków 2016.
8. Jong-Won L., Sang-Woo L., Hai Gyong K., Hyun-Kil J., Se-Rin P., *Green Space and Apartment Prices: Exploring the Effects of the Green Space Ratio and Visual Greenery*, Economic Valuation of Urban Green Spaces, b.m. 2023.
9. Kellenberger K., Shaw S., *Beginning T-SQL*, Apress, Berkeley 2014.
10. Kompa K., *Budowa mierników agregatowych do oceny poziomu rozwoju społeczno-gospodarczego*, SGGW w Warszawie, Warszawa b.r.
11. Kumar Nirmal V., *Satellite Cities: The Only Hope of Megacities A Case of Indian Scenario*, Amity School of Architecture & Planning, Amity University Haryana, Gurgaon 2015.
12. Kwon D., Sorenson O., *The Silicon Valley Syndrome*, Yale University and University of California, Los Angeles 2021.
13. Lee D., *How Airbnb Short-Term Rentals Exacerbate Los Angeles's Affordable Housing Crisis: Analysis and Policy Recommendations*, Harvard Law School Journals, Los Angeles 2016.

14. Maxim C., *Challenges Faced by World Tourism Cities – London’s Perspective*, London Geller College of Hospitality & Tourism, University of West London, London 2017.
15. Russo M., Ferrari A., *The Definitive Guide to Dax: Business Intelligence with Microsoft Power BI, SQL Server Analysis Services, and Excel*, Published with The Authorization of Microsoft Corporation by Pearson Education, b.m. 2020.
16. Sherman R., *Business Intelligence Guidebook: From Data Integration to Analytics*, Elsevier, Amsterdam 2014.
17. Wang D.Z.W., *Financial sustainability of rail transit service: The effect of urban development*, School of Civil and Environmental Engineering, Singapore 2016.

Artykuły

1. Babic T., *How to Write Multiple CTEs in SQL*, "LearnSQL", 2022, <https://learnsql.com/blog/multiple-cte/> [dostęp 22.05.2024].
2. Cagliari S., *Exploring the Filter Context with DAX functions*, "Towards Data Science", 2022, <https://towardsdatascience.com/exploring-the-filter-context-with-dax-functions-422211c1118e> [dostęp 23.05.2024].
3. Celko J., *The NTILE Function*, "Red Gate", 2023, <https://www.red-gate.com/simple-talk/databases/theory-and-design/the-ntile-function/> [dostęp 23.05.2024].
4. Cvijanovic D., *Real Estate Finance: How Demographics Drive Housing Prices*, "HEC", 2012, <https://www.hec.edu/en/real-estate-finance-how-demographics-drive-housing-prices> [dostęp 22.05.2024].
5. Eldersveld D., *Overcoming Potential Power Bi Shape Map Rendering Issues - Dataveld*, "Dataveld", 2016, <https://dataveld.com/2016/09/01/overcoming-potential-power-bi-shape-map-rendering-issues/> [dostęp 22.05.2024].
6. *English Common Law is the most widespread legal system in the world*, "Sweet and Maxwell", 2008, <https://www.sweetandmaxwell.co.uk/about-us/press-releases/061108.pdf> [dostęp 22.05.2024].
7. *European Movement UK: The Economic Benefits to the UK of Eu Membership*, "Web Archive", <https://web.archive.org/web/20150629121205/http://www.euromove.org.uk/index.php> [dostęp 22.05.2024].

8. Gigoyan S., *Creating a table using the SQL SELECT INTO clause*, "MSSQLTips", 2021, <https://www.mssqltips.com/sqlservertip/6977/sql-select-into-create-table/> [dostęp 22.05.2024].
9. Jun Y., *Efficient Euclidean distance computation in Pandas*, "Towards Data Science", <https://towardsdatascience.com/efficient-euclidean-distance-computation-in-pandas-66b472f6b0ba> [dostęp 23.05.2024].
10. *London is the Soft Power and High Skills Capital of the World*, "Deloitte", 2016, <https://www2.deloitte.com/uk/en/pages/press-releases/articles/london-soft-power-and-high-skills-capital.html> [dostęp 22.05.2024].
11. Maximino M., *The Impact of Crime on Property Values: Research Roundup*, "Journalist's Resource", 2014, <https://journalistsresource.org/economics/the-impact-of-crime-on-property-values-research-roundup/> [dostęp 22.05.2024].
12. Murray S., *Power BI Conditional Formatting for Matrix and Table Visuals*, "MSSQLTips", 2019, <https://www.mssqltips.com/sqlservertip/6265/power-bi-conditional-formatting-for-matrix-and-table-visuals/> [dostęp 23.05.2024].
13. *Regional Gross Value Added (Income Approach) NUTS3 Tables*, "Web Archive", 2014, <https://webarchive.nationalarchives.gov.uk/ukgwa/20160105160709/http://www.ons.gov.uk/ons/rel/regional-accounts/regional-gross-value-added--income-approach-december-2013/rft-nuts3.xls> [dostęp 22.05.2024].
14. Russo M., Ferrari A., *Understanding apply semantics for window functions in DAX*, "SQLBI", 2024, <https://www.sqlbi.com/articles/understanding-apply-semantics-for-window-functions-in-dax/> [dostęp 23.05.2024].
15. Shekhar S., *3 easy steps to use SWITCH function to make conditional coloured bar charts in Power BI*, "Medium", 2024, <https://medium.com/microsoft-power-bi/3-easy-steps-to-use-switch-function-to-make-a-conditional-coloured-bar-charts-in-power-bi-d54826fce99f> [dostęp 23.05.2024].
16. *The City of London's Strange History*, "FT", 2014, <https://www.ft.com/content/7c8f24fa-3aa5-11e4-bd08-00144feabdc0> [dostęp 22.05.2024].
17. Villazon A., *Useful SQL Server functions: TRY_CAST & TRY_CONVERT*, "Andrew Villazon", 2020, <https://www.andrewvillazon.com/sql-server-try-cast-convert/> [dostęp 22.05.2024].
18. Werdiger J., *London Wants to Tap Chinese Currency Market*, "NY Times", 2012, <https://archive.nytimes.com/dealbook.nytimes.com/2012/01/16/london-wants-to-tap-chinese-currency-market/> [dostęp 22.05.2024].

19. Zednicek J., *SQL CTE (Common Table Expressions) With Examples – More Organized Queries and Procedures*, "Jan Zednicek", 2020, <https://janzednicek.cz/en/sql-cte-common-table-expressions-with-clause-more-organized-queries-and-procedures/> [dostęp 22.05.2024].

Źródła danych i mapy

1. J. Cirtautas, *Housing in London*, 2020, [w:] kaggle.com, <https://www.kaggle.com/datasets/justinas/housing-in-london> [dostęp 22.05.2024].
2. *London Crime Data*, [w:] data.london.gov.uk, <https://data.london.gov.uk/> [dostęp 22.05.2024].
3. J. Arvidsson, *Airbnb Global Listings*, 2023, [w:] kaggle.com, <https://www.kaggle.com/datasets/joebeachcapital/airbnb/data> [dostęp 22.05.2024].
4. *Statistical GIS Boundary Files for London*, 2018, [w:] data.london.gov.uk, <https://data.london.gov.uk/dataset/statistical-gis-boundary-files-london> [dostęp 22.05.2024].

SPIS RYSUNKÓW

Rysunek 1. Problem z renderingiem mapy	23
Rysunek 2. Naprawiona mapa za pomocą QGIS.....	24
Rysunek 3. Przesunięcie danych pomiędzy kolumnami	25
Rysunek 4. Usunięcie przesuniętych danych	25
Rysunek 5. Przykładowy spis.....	26
Rysunek 6. Zapytanie wyszukujące obszary w bazie dbo. housing yearly nie znajdujące się w tabeli centralnej.....	27
Rysunek 7. Obszary w tabeli dbo. housing yearly odmienne od istniejących w punkcie odniesienia.....	27
Rysunek 8. Usunięcie zbędnych obszarów z dbo. housing yearly	28
Rysunek 9. Obszary odmienne od punktu odniesienia w tabeli dbo. housing monthly .	28
Rysunek 10. Usunięcie zbędnych obszarów z dbo. housing monthly	29
Rysunek 11. Zapytanie grupujące obserwacje według dzielnic w tabeli dbo. Airbnb ...	29
Rysunek 12. Obszary pogrupowane od najmniej licznych w tabeli dbo. Airbnb	30
Rysunek 13. Usunięcie nieistotnych obszarów z tabeli dbo. Airbnb.....	30
Rysunek 14. Zapytanie umożliwiające wyświetlenie wszystkich wspólnych dzielnic z tabel	31
Rysunek 15. Wspólne dzielnice pomiędzy źródłami danych (33)	32
Rysunek 16. Niepasujące do siebie kody dzielnicowe	33
Rysunek 17. Kod SQL narzucający jednolity kod dzielnicowy wszystkim źródłom danych	33
Rysunek 18. Wszystkie źródła danych zawierające spójny kod dzielnicowy - klucz główny tabel	34
Rysunek 19. Zamiana pustych wartości na NULL	35
Rysunek 20. Przekształcenie danych nie liczbowych oraz niepustych wartości w NULL	35
Rysunek 21. Stworzenie kolumny zawierającej dane dotyczące średniego zysku z najmu krótkoterminowego	36
Rysunek 22. Brakujące informacje dla 7 z obserwacji oznaczonych jako # w bazie dbo. housing yearly.....	36
Rysunek 23. Zamiana typów danych oraz obserwacji zawierającej # na NULL	37

Rysunek 24. Zapytanie umożliwiające znalezienie zeszłorocznych oraz przyszłorocznych wartości dla pustych obserwacji z kolumny mean_salary	38
Rysunek 25. Zobrazowanie funkcjonowania kodu SQL dla zeszłorocznych i przyszłorocznych wartości	38
Rysunek 26. Wprowadzenie średniej z okalających wartości w miejsce pustych obserwacji w kolumnie mean_salary	38
Rysunek 27. Dwie w dalszym ciągu występujące obserwacje z powodu braku okalających dla nich wartości.....	39
Rysunek 28. Ręcznie wprowadzone wartości w puste obserwacje w dzielnicy Hackney	39
Rysunek 29. Rezultat przekształceń w kolumnie mean_salary	39
Rysunek 30. Zamiana typów danych oraz pustych obserwacji na NULL	40
Rysunek 31. Wprowadzenie średniej z okalających wartości w miejsce pustych obserwacji w kolumnie median_salary	40
Rysunek 32. Seryjna zamiana pustych wartości na bazodanowy NULL.....	41
Rysunek 33. Kompletne rozwiązanie problemu pustych wartości w kolumnie population_size dla roku 2019.....	42
Rysunek 34. Kompletne rozwiązanie problemu pustych wartości w kolumnie number_of_jobs dla roku 2019.....	43
Rysunek 35. Kompletne rozwiązanie problemu pustych wartości w kolumnie number_of_jobs dla roku 1999.....	44
Rysunek 36. Zaokrąglenie wartości do tysięcy w kolumnie number_of_jobs.....	44
Rysunek 37. Podglądowy przekrój danych dla pojedynczej dzielnicy i wszystkich dostępnych lat z tabeli dbo. housing yearly	45
Rysunek 38. Kod zapisujący dane z roku 2018 dla kolumny life_satisfaction w tabeli centralnej oraz usunięcie kolumn area_size i life_satisfaction z dbo. housing yearly ...	46
Rysunek 39. Usunięcie obserwacji posiadających zysk równy 0 z dbo. Airbnb poprzez tworzenie nowej tabeli z pominięciem obserwacji.....	47
Rysunek 40. Mapa Londynu - cena najmu za dobę	49
Rysunek 41. Karty informacyjne - zwrot z inwestycji, średnia cena najmu, średni zysk	50
Rysunek 42. Wykres kołowy udział typu najmu oraz wykres słupkowy liczba łóżek ...	51
Rysunek 43. Wykres - średnia cena najmu zależna od liczby łóżek	51
Rysunek 44. Macierz przedziałów cenowych najmu w poszczególnych dzielnicach....	52

Rysunek 45. Wyznaczenie granic przedziałów o takiej samej liczbie obserwacji na podstawie ceny w celu stworzenia macierzy	53
Rysunek 46. Podział obserwacji na grupy za pomocą kolumny kalkulacyjnej stworzonej w języku DAX.....	53
Rysunek 47. Procentowy udział danej grupy w macierzy dla dzielnic Londynu.....	55
Rysunek 48. Obliczanie odległości pomiędzy lokalizacją apartamentów na wynajem, a Dworcem Głównym Londynu	57
Rysunek 49. Korelacje pomiędzy ceną najmu, a odlegością od Pałacu Westminster lub Dworca Głównego	58
Rysunek 50. Mapa Londynu - liczba miejsc pracy.....	60
Rysunek 51. Suwak lat 1999 – 2019.....	60
Rysunek 52. Kod DAX - średnia liczba miejsc pracy w wybranych latach na suwaku .	61
Rysunek 53. Dzielnice o charakterze sypialni lub charakterze biznesowym.....	62
Rysunek 54. Dzielenie w języku DAX	63
Rysunek 55. Dzielenie w języku DAX z zablokowaną filtracją	63
Rysunek 56. Karty zawierające wykresy - w zależności od istnienia filtracji pojawia się punkt odniesienia	63
Rysunek 57. Karty informacyjne – strona rynek pracy	64
Rysunek 58. Miary DAX pisane pod formatowanie warunkowe kolorów kart	65
Rysunek 59. Mapa Londynu - cena nieruchomości.....	66
Rysunek 60. Procentowy przyrost średnich wynagrodzeń oraz cen nieruchomości rok do roku – suma zmian procentowych	67
Rysunek 61. Grupowanie czasowe danych miesięcznych na roczne w celu stworzenia wizualizacji.....	68
Rysunek 62. Model danych przed dodaniem tabeli miernika syntetycznego	69
Rysunek 63. Kolumna kalkulacyjna przestawiająca opóźnione o jedną pozycję obserwacje	70
Rysunek 64. Kolumna kalkulacyjna obliczająca różnicę pomiędzy średnią ceną mieszkania, a średnią ceną mieszkania opóźnioną o jedną pozycję	71
Rysunek 65. Kolumna kalkulacyjna obliczająca procentową zmianę	71
Rysunek 66. Kolumna kalkulacyjna sumującą procentowe zmiany rok do roku na przestrzeni 20 lat.....	72
Rysunek 67. Zobrazowane rozwiążanie budowy wykresu YoY%.....	72

Rysunek 68. Użycie Calendar_Year_PBI w celu powiązania danych z różnych źródeł dla tych samych okresów	73
Rysunek 69. Karty informacyjne - Power BI	74
Rysunek 70. Średnie wynagrodzenie MIN – miara DAX.....	75
Rysunek 71. Rozwiązywanie problemu braku danych dla lat z Calendar_ Year wybranych po lewej stronie suwaka (analogiczne rozwiązanie dla prawej strony – zamiast MIN → MAX) – średnie wynagrodzenie.....	76
Rysunek 72. Wykres - średnia liczba sprzedanych nieruchomości rocznie.....	77
Rysunek 73. Miara DAX do punktu odniesienia na wykresie - średnia liczba sprzedanych nieruchomości rocznie	77
Rysunek 74. Wykres - lata pracy na zakup nieruchomości.....	78
Rysunek 75. Zablokowana filtracja jako kreacja punktu odniesienia w obliczu filtracji przez mapę.....	78
Rysunek 76. Agregacja danych dotycząca liczby przestępstw oraz usunięcie wybranych kategorii.....	80
Rysunek 77. Kod SQL tworzący tabelę wspierającą kreację mierników promieniowych	80
Rysunek 78. Mapa Londynu - liczba przestępstw	81
Rysunek 79. Miara wyświetlająca sumę przestępstw dla najwyższego roku z wybranych na suwaku lat	81
Rysunek 80. Drzewo - kategorie i podkategorie przestępstw	82
Rysunek 81. Wykres - liczba przestępstw w wybranych kategoriach na przestrzeni lat	83
Rysunek 82. Miara DAX - wartość 0.....	84
Rysunek 83. Miara - maksymalna liczba przestępstw w wybranej kategorii jako górný punkt odniesienia miernika promieniowego	84
Rysunek 84. Miara - średnia liczba przestępstw jako cel miernika promieniowego.....	85
Rysunek 85. Mierniki promieniowe dla wybranej na mapie dzielnicy, w której występuje mniej przestępstw niż średnia w Londynie	85
Rysunek 86. Mierniki promieniowe dla wybranej na mapie dzielnicy, w której występuje więcej przestępstw niż średnia w Londynie	85
Rysunek 87. Wartość miernika dla określonej kategorii - brak filtracji = średnia, filtracja = suma dla filtrowanej dzielnicy.....	86
Rysunek 88. Miara - zmiana koloru w zależności od liczby przestępstw w wybranej kategorii.....	86

Rysunek 89. Mierniki promieniowe - liczba przestępstw oraz liczba przestępstw na 1000 osób dla dzielnicy Croydon.....	87
Rysunek 90. Miara DAX obliczająca wielkość populacji przy filtracji lub jej braku	88
Rysunek 91. Miary obliczające średnią oraz maksymalną liczbę przestępstw	89
Rysunek 92. Rozwiązywanie problemu maksymalnej wartości mniejszej niż realna w mierniku - liczba przestępstw na 1000 osób.....	90
Rysunek 93. Maksymalne liczby przestępstw na 1000 osób w dzielnicach Londynu na przestrzeni 8 lat w wybranych latach.....	90
Rysunek 94. Odniesienie się w mierze do wartości maksymalnych policzonych w środowisku SQL Server.....	91
Rysunek 95. Miara liczba przestępstw ogółem na 1000 osób - wynik zależny od istnienia filtracji.....	92
Rysunek 96. Objaśnienie komplementarnego rozwiązania budowy miernika promieniowego - liczba przestępstw na 1000 osób	93
Rysunek 97. Korelacje dodatnie	96
Rysunek 98. Korelacje ujemne	98
Rysunek 99. Najbardziej atrakcyjne dzielnice do zamieszkania przy określonym budżecie na zakup nieruchomości	101

ZAŁĄCZNIKI

OŚWIADCZENIA

Załącznik nr 1 do zarządzenia Rektora UG nr 70/R/15 ze zm.

Oświadczam, że przedłożona praca dyplomowa została przygotowana przeze mnie samodzielnie, nie narusza praw autorskich, interesów prawnych i materialnych innych osób oraz wykorzystanie materiałów wytworzonych przez generatywne narzędzia sztucznej inteligencji odbyło się w zakresie uzgodnionym z promotorem.

10.06.2024



.....
data

.....
podpis

Załącznik nr 3 do zarządzenia Rektora UG nr 70/R/15

Wyrażam zgodę / nie wyrażam zgody* na udostępnienie osobom zainteresowanym mojej pracy dyplomowej dla celów naukowo-badawczych.

Zgoda na udostępnienie pracy dyplomowej nie oznacza wyrażenia zgody na kopowanie pracy dyplomowej w całości lub w części.

* niepotrzebne skreślić

10.06.2024



.....
data

.....
podpis