**Hurtownie Danych - laboratorium**

**Lista 8**

***MDX, KPI, Data Mining***

**Wstęp teoretyczny**

Sortowanie zbioru - ORDER:

|  |
| --- |
| **ORDER** ( {<Set>}, <wyrażenie> [, <Flaga>])  Funkcja Order sortuje krotki zbioru na podstawie wartości wyrażenia.  <Flaga> ::= ASC | DESC | BASC | BDESC   * ASC, DESC – uporządkowanie uwzględniające sortowanie zgodnie z hierarchią, np. dla hierarchii Rok-Miesiąc-Dzień, funkcja wybierze najpierw najlepszy rok, potem w tym roku najlepszy miesiąc, a z tego miesiąc najlepszy dzień. * BASC, BDESC – sortowanie z pominięciem ograniczenia wynikającego z hierarchii, np. dla powyższego przykładu funkcja zwróci dzień, w którym wartość miary była odpowiednio najmniejsza lub największa. |

Filtrowanie wyniku – FILTER

|  |
| --- |
| **FILTER** ( {<Set>}, <wyrażenie\_logiczne> )  Funkcja sprawdza spełnienie warunku <wyrażenie\_logiczne> dla każdej krotki w określonym zbiorze <Set>. Zwraca zbiór krotek, dla których warunek jest spełniony.  Przykład:  Filtruje wiersze, wyświetlając tylko daty, w których sprzedaż była większa niż 10000.  SELECT [Measures].[Internet Sales Amount] ON COLUMNS,  FILTER(  [Date].[Date].[Date].MEMBERS,  [Measures].[Internet Sales Amount]>10000  )  ON ROWS  FROM [Cube] |

Źródła:

<https://docs.microsoft.com/en-us/sql/mdx/order-mdx?view=sql-server-ver15>

<https://docs.microsoft.com/en-us/sql/mdx/filter-mdx?view=sql-server-ver15>

**Wybrane algorytmy eksploracji danych**

|  |
| --- |
| **Regresja liniowa** oblicza liniową zależność pomiędzy zmienną zależną i niezależnymi, w celu predykcji kolejnych wartości zmiennej zależnej.  y=ax+b  gdzie:   * x – zmienna niezależna (wymaganie VS: ciągłe wartości liczbowe) * y – zmienna zależna (również wartości liczbowe) * a – współczynnik kierunkowy prostej * b – wyraz wolny prostej   Celem jest wyznaczenie współczynników a i b tak, aby sumaryczna odległość punktów od wyznaczonej prostej była minimalna.  Model regresji liniowej stosuje się w zadaniach predykcji trendu |

|  |
| --- |
| **Drzewo decyzyjne** jest algorytmem klasyfikacji wykorzystywanym w zadaniach modelowania predykcji zarówno dla atrybutów ciągłych jak i dyskretnych.  W przypadku atrybutów dyskretnych algorytm predykuje wartość atrybutu na podstawie wartości określonego zestawu danych wejściowych. W szczególności, algorytm identyfikuje te atrybuty, które są skorelowane z atrybutem lub wartością predykowaną.  *Przykład:*  Chcemy określić cechy klientów, którzy prawdopodobnie kupią rower. Jeśli dziewięciu na dziesięciu młodszych klientów kupiło rower, ale tylko dwóch na dziesięciu starszych klientów to zrobiło, to algorytm ustala, że ​​wiek jest dobrym predykatorem zakupu roweru.  W przypadku atrybutów ciągłych, algorytm wyznacza model regresji liniowej, który determinuje miejsce podziału dla drzewa decyzyjnego. Możliwe jest prognozowanie więcej niż jednej wartości. Ponadto, jeśli dane wejściowe zawierają zagnieżdżoną tabelę, to algorytm buduje osobne drzewo decyzyjne dla każdej predykowanej wartości.  Algorytm drzewa decyzyjnego wybiera atrybuty (węzły), na podstawie których dokonywany jest podział (tworzona nowa gałąź w drzewie). Jeśli dany atrybut jest skorelowany z wartością predykowaną, to dodawany jest nowy wierzchołek do drzewa.  W celu wyznaczenia najlepszych atrybutów do kolejnych poziomów drzewa wykorzystywany jest algorytm selekcji cech (ang. *feature selection*), dzięki czemu tworzony model zawiera jedynie niezbędne atrybuty, co poprawia wydajność czasową i pamięciową algorytmu.  Ważnym aspektem jakości drzew decyzyjnych jest ich głębokość (czyli liczba atrybutów, które decydują o predykcji). Zbyt czuły lub przetrenowany model może nie być przydatny dla nowych danych.  Drzewa decyzyjne stosuje się w zadaniach ciągłej i dyskretnej predykcji wartości oraz w wyznaczaniu grup atrybutów o podobnych wartościach, np. analiza koszykowa (wyznaczenie cech klientów, którzy kupują określone produkty) i rekomendacja oparta o filtrowanie kolaboratywne (*„inni klienci kupili również…”*). |

|  |
| --- |
| **Naiwny klasyfikator Bayesowski** jest algorytmem klasyfikacji opartym na twierdzeniu Bayesa (zakłada niezależność zdarzeń), dzięki czemu pozwala na szybkie generowanie modeli. Jest często stosowany do szybkiego znalezienia relacji pomiędzy atrybutami wejściowymi a zmienną predykowaną.  Algorytm oblicza prawdopodobieństwa warunkowe każdego stanu wartości (dla każdej wejściowej kolumny) w zależności od wartości zmiennej predykowanej. Dla predykcji dobre są takie cechy, których rozkłady dla różnych wartości zmiennej predykowanej różnią się miedzy sobą.  Naiwy klasyfikator Bayesowski stosuje się w zadaniach predykcji wartości dyskretnych (określanie prawdopodobieństwa zajścia określonych zdarzeń). |

|  |
| --- |
| Celem **algorytmów grupowania** jest wyznaczenie klastrów elementów, które mają podobne cechy. Model jest trenowany na podstawie relacji pomiędzy elementami w zbiorze danych oraz klastrami, które algorytm wyznacza.  Po wstępnym zdefiniowaniu klastrów początkowych, algorytm oblicza, w jakim stopniu klastry reprezentują zawartość grupy, a następnie ponownie zdefiniować grupy tak, aby dopasowanie elementów do grup było lepsze. Algorytm wykonywany jest iteracyjnie do momentu, gdy kolejne iteracje nie dają poprawy.  Algorytmy te nie służą określaniu predykcji wartości konkretnej zmiennej.  Algorytmy grupowania znajdują zastosowanie w zadaniach znajdowania grup o podobnych cechach, identyfikacji anomalii i tworzenia prognoz. |

|  |
| --- |
| **Modele asocjacyjne** budowane są na podstawie często powtarzających się zależności w zbiorze danych. Opierają się na częstości występowania wspólnych wartości atrybutów. Nie uwzględniają kolejności tych atrybutów. Wynikiem tej metody są reguły asocjacyjne w postaci implikacji. W przypadku zagadnień transakcyjnych jest to postać:  *Jeśli klient kupił produkt A, B i C, to prawdopodobnie zakupi również produkt D.*  Wskaźnikami jakości dla reguł decyzyjnych są:   * wsparcie (ang. *support*) – jak często dana reguła występuje w całym zbiorze * zaufanie (ang. *confidence*) – względna częstość poprawności reguły   Reguły asocjacyjne są przydatne w zadaniach analizy koszykowej oraz rekomendacji opartej na zawartości (ang. *content-based recommendation*, czyli rekomendacja jest determinowana na podstawie już zakupionych produktów) |

Źródła:

<https://docs.microsoft.com/en-us/analysis-services/data-mining/data-mining-algorithms-analysis-services-data-mining?view=asallproducts-allversions>

<https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-linear-regression-algorithm?view=asallproducts-allversions>

<https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-decision-trees-algorithm?view=asallproducts-allversions>

<https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-naive-bayes-algorithm?view=asallproducts-allversions>

<https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-clustering-algorithm?view=asallproducts-allversions>

<https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-association-algorithm?view=asallproducts-allversions>

**Zadania do wykonania**

**Zad. 1. Zapytania MDX**

Napisać zapytania MDX, które:

1. Wyświetli liczbę sprzedanych produktów dla wybranych podkategorii produktu i płci sprzedawcy, w poszczególnych latach.
2. Wyświetli kwoty rabatów udzielonych na zakupione produkty dla kategorii i koloru.
3. Wyświetli liczbę różnych klientów, którzy dokonali zakupu w poszczególnych latach i miesiącach z uwzględnieniem ich regionu (CountryRegionCode).
4. Wyświetli liczbę transakcji obsługiwanych przez poszczególnych sprzedawców z uwzględnieniem kategorii produktu. Zestawienie posortuj malejąco. Uzasadnij wybór atrybutu sortowania.

Do powyższych zapytań zastosować filtrowanie. Uzasadnić wybór. Zinterpretować wynik.

**Zad. 2. Zapytania SQL**

Korzystając z bazy AdventureWorks napisać zapytania SQL, które dadzą analogiczny wynik jak w zadaniu 1.

**Zad. 3. KPI**

1. Przygotować wskaźnik KPI (zakładka KPI), która umożliwi podział klientów na dobrych i lepszych w zależności od liczby sztuk zamówionych produktów.

Tworząc nowy wskaźnik należy podać jego nazwę, wybrać (przeciągnąć) miarę, na podstawie której będzie dokonany podział zbioru, wybrać odpowiedni status (np. Shapes) i podać warunek:

*iif ( [Measures].[OrderQty] < , -1 /\*czerwony\*/, 1 /\*zielony\*/ )*

Należy uzasadnić wybór wartości progowej **.

Po przeprocesowaniu kostki, należy zobrazować działanie wskaźnika dla wybranych atrybutów w raporcie w Excelu.

1. Zaproponować własną miarę w zakładce *Calculation -> New Calculated Member*, (np. zysk z uwzględnieniem rabatu i frachtu), na podstawie której zostanie zdefiniowany odpowiedni wskaźnik KPI. Należy przeanalizować status opracowanego wskaźnika oraz jego trend. Wynik należy zaprezentować w wybranym kontekście.

**Zad. 4. Data Mining**

Wykorzystując opcję *Solution Explorer -> Mining Structures* należy przygotować analizę danych, korzystając z dostępnego kreatora.

Na podstawie wybranych atrybutów wymiarów Customer i Product należy przygotować predykcję wartości OrderQty i LineTotal, wykorzystując min. 3 różne algorytmy eksploracji dancyh:

- w zakładce *Mining Structure* należy wybrać interesujące atrybuty

- w zakładce *Mining Models* – właściwy algorytm

Na podstawie elementów dostępnych w pozostałych zakładkach (*Mining Model Viewer, Mining Accuracy Chart, Mining Model Prediction*) należy zinterpretować uzyskany wynik.

**Zad. 5. Data Mining**

Wykorzystując dane dostępne w bazie AdventureWorksDW, przygotować analizę danych w wybranym kontekście (wybrać najlepszy algorytm), np.:

* prognoza sprzedaży w kolejnym roku,
* analiza koszykowa, aby rekomendować klientowi dodatkowe produkty,
* grupowanie klientów o podobnych preferencjach zakupowych

Zinterpretować uzyskany wynik.

**Rozwiązania:**

**Wnioski:**

***Uwaga!***

* Sprawozdanie, bez wniosków podsumowujących aspekt zagadnień analizowanych na zajęciach laboratoryjnych i zawartych w sprawozdaniu, jest automatycznie oceniane negatywnie!