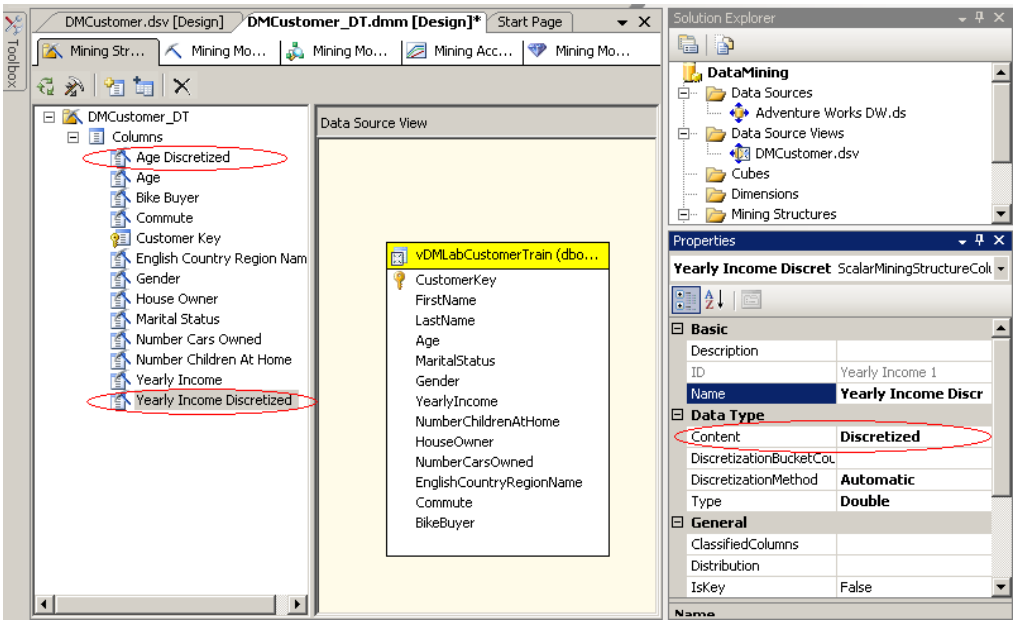
Data Mining – podstawy analizy danych
Część druga

W części pierwszej dokonaliśmy procesu analizy danych treningowych w oparciu o algorytm drzewa decyzyjnego. Proces analizy danych treningowych może być realizowany przez różne algorytmy. W takim razie możemy sobie zadawać pytanie: czy analiza danych przy pomocy różnych algorytmów da takie same wyniki czy też będą rozbieżności? A tym samym pojawiają się dalsze pytania:
- Czy wybrany przez nas algorytmy jest prawidłowy?
- Czy prezentacja danych jest zgodna z naszymi oczekiwaniami?

Aby odpowiedzieć sobie na te pytania możemy wykonać szereg analiz przy pomocy różnych algorytmów a następnie porównać efekty ich działania z danymi sprawdzającymi (weryfikacja algorytmów).
W SQL2005 istnieje możliwość realizacji wielu analiz przy pomocy wielu algorytmów w jednym projekcie. Oprócz popularnego algorytmu drzewa decyzyjnego – będący w istocie jednym z algorytmów klasyfikujących – bardzo popularny algorytmem jest naiwny klasyfikator Bayes (naive Bayes). Właśnie w naszym projekcie zastosujemy ten drugi algorytm do porównania wyników.
Proces analizy danych przy pomocy innych algorytmów, może wymagać przekształcenia niektórych typów danych. Wynika to z faktu, iż niektóre z algorytmów np. Klasyfikator Bayes'a – wymagają danych zdefiniowanych przedziałowych. Przykładem, w naszym przypadku są dochody roczne, które mogą przyjmować dowolne wartości. Aby można wykorzystać te dane należy zdefiniować skończoną liczbę przedziałów. W procesie przekształcenia typów danych, możemy wykorzystać właściwości modelu analitycznego, a w szczególności widoku źródeł danych, gdzie typ danych, możemy w prosty sposób dopasować do wymagań algorytmu.
W naszym przypadku rozbudujemy mechanizmy związane z widokiem źródła danych o dodatkową kolumnę, zawierająca dyskretne dane dotyczące przychodów rocznych oraz wieku naszych klientów. Pozostałe, niepasujące dane do tego algorytmu zignorujemy.



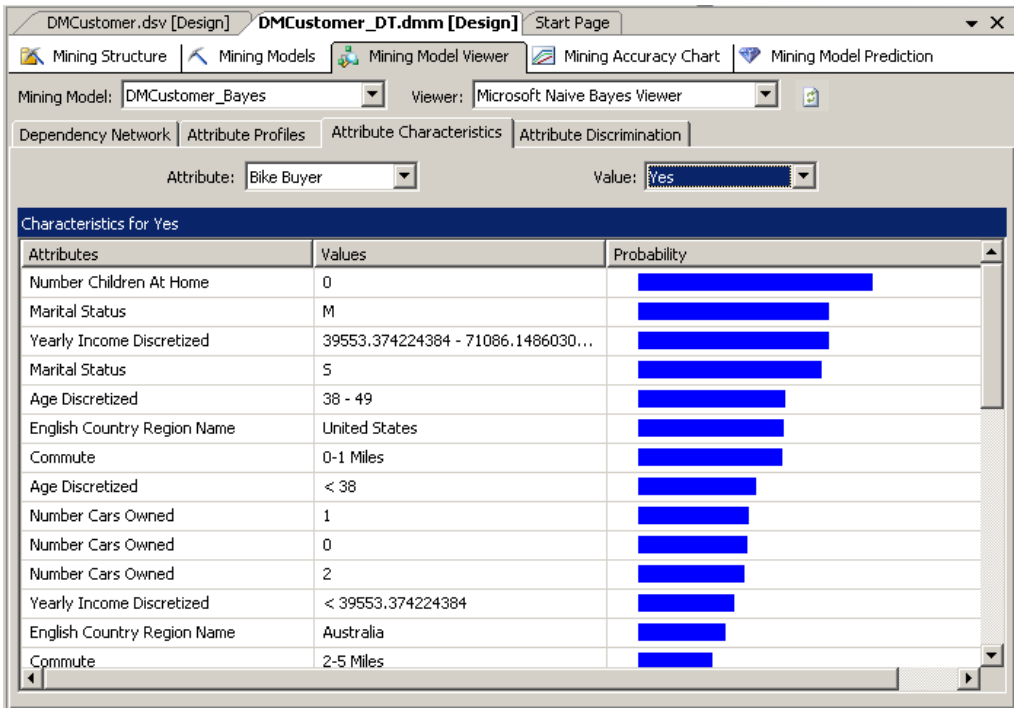
Rys 1. Definicja danych przedziałowych.

Efektem przygotowania powyższych danych jest możliwość zdefiniowania modelu analizy opartego na procesach analizy klasyfikatorem Bayes'a, włączając typy danych określone jako przedziałowe.

Structure	DMCustomer_DT	DMCustomer_Bayes
	Microsoft_Decision_Trees	Microsoft_Naive_Bayes
Age	Input	Ignore
Age Discretized	Ignore	Input
Bike Buyer	PredictOnly	PredictOnly
Commute	Input	Input
Customer Key	Key	Key
English Country Region Name	Input	Input
Gender	Input	Input
House Owner	Input	Input
Marital Status	Input	Input
Number Cars Owned	Input	Input
Number Children At Home	Input	Input
Yearly Income	Input	Ignore
Yearly Income Discretized	Ignore	Input

Rys 2. definicja modelu analizy danych opartego o klasyfikator Bayes'a

Każdy z algorytmów ma specyficzny sposób prezentacji danych. W przypadku drzewa decyzyjnego była to architektura drzewiasta. W przypadku klasyfikatora Bayes'a są to wykresy pokazujące zależności pomiędzy wartościami wejściowymi i wartością predykcji – charakterystyka cech oraz charakterystyka dyskryminacji poszczególnych parametrów wejściowych. Podobnie jak przypadku drzewa decyzyjnego, mamy również zaprezentowane mechanizmy zależności w postaci sieci zależności – „Dependenci Network".



Rys 3. Wykres pokazujący prawdopodobieństwo wpływy danej cechy na podjęcie decyzji o zakupie roweru w algorytmie Bayes'a.

Etap drugi – predykcja danych i weryfikacja danych

Proces analizy danych treningowych pozwala uzyskać informację o zależnościach między danymi wejściowymi a parametrem analizowanym. W tym przypadku realizujemy operacje „post factum" – mamy odpowiedź w naszych zbiorach danych a jedynie analizujemy ich zależność.
Kolejnym krokiem w procesie analizy danych jest predykacja danych. W tym przypadku mamy tylko dane wejściowe – a nie posiadamy odpowiedzi, w jaki sposób to wpłynie na wartość oczekiwaną. W naszym projekcie możemy to sobie wyobrazić, że zamierzamy sprzedawać rowery w nowej lokalizacji. Zebraliśmy podobne informacje o naszych potencjalnych klientach. Na podstawie tychże danych zamierzamy ocenić możliwości sprzedaży rowerów poszczególnym klientom.

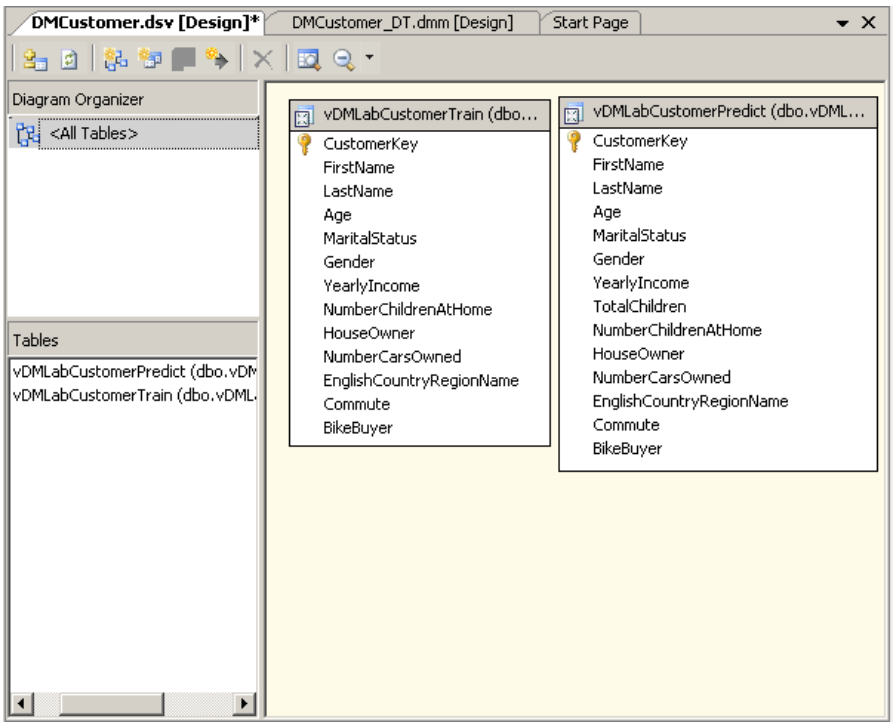
W celu realizacji naszego projektu wykorzystamy skrypt tworzący widok z określonymi danymi:

```
/******  View [dbo].[vDMLabCustomerPredict]  ******/

CREATE VIEW [dbo].[vDMLabCustomerPredict]  AS
SELECT
        c.[CustomerKey],
        c.[FirstName],
        c.[LastName],
        CASE
                WHEN Month(GetDate()) < Month(c.[BirthDate])
                        THEN DateDiff(yy,c.[BirthDate],GetDate()) - 1
                WHEN Month(GetDate()) = Month(c.[BirthDate])
                AND Day(GetDate()) < Day(c.[BirthDate])
                        THEN DateDiff(yy,c.[BirthDate],GetDate()) - 1
                ELSE DateDiff(yy,c.[BirthDate],GetDate())
        END AS [Age],
        c.[MaritalStatus],
        c.[Gender],
        c.[YearlyIncome],
        c.TotalChildren,
        c.[NumberChildrenAtHome],
        CASE c.[HouseOwnerFlag] WHEN 0 THEN 'No' ELSE 'Yes' END as HouseOwner,
        c.[NumberCarsOwned],
        g.EnglishCountryRegionName,
        c.[CommuteDistance] As Commute,
        CASE x.[Bikes]
                WHEN 0 THEN 'No'
                ELSE 'Yes'
```

```
        END AS [BikeBuyer]
    FROM
        [dbo].[DimCustomer] c INNER JOIN (
            SELECT
                [CustomerKey]
                ,[Region]
                ,[Age]
                ,Sum(
                    CASE [EnglishProductCategoryName]
                        WHEN 'Bikes' THEN 1
                        ELSE 0
                    END) AS [Bikes]
            FROM
                [dbo].[vDMPrep]
            GROUP BY
                [CustomerKey]
                ,[Region]
                ,[Age]
            ) AS [x]
        ON c.[CustomerKey] = x.[CustomerKey]
    join dimgeography g
    on c.geographykey = g.geographykey
go
```

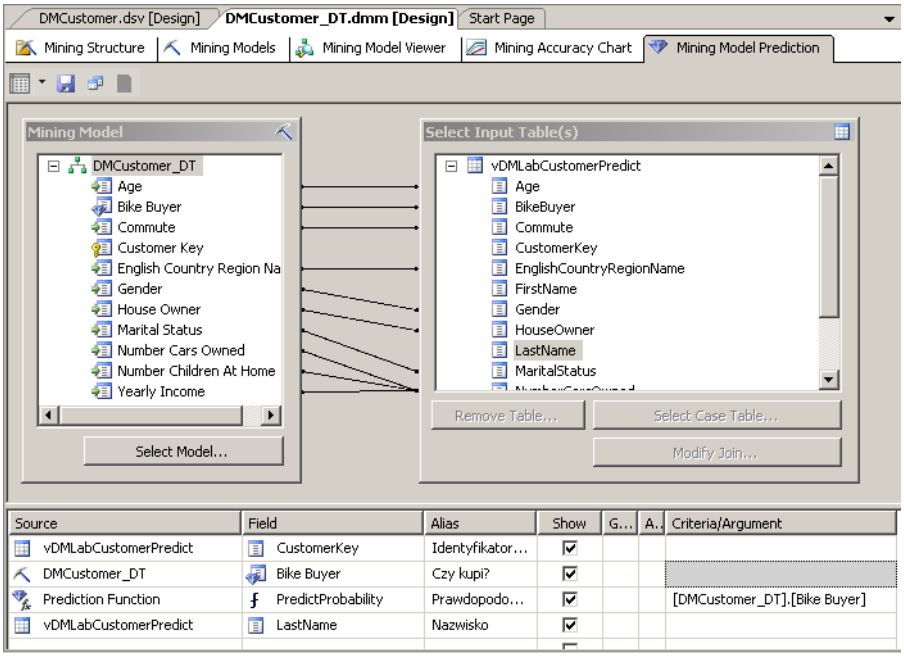
Proces predykcji rozpoczynamy od odpowiedniego skonfigurowania widoku źródeł danych, dołączając zdefiniowany widok.



Rys 4. Dołączenie do projektu widoku do analizy predykcji

Możemy w teraz przystąpić do predykcji dołączonego zbioru danych. Oczywiście predykcję danych możemy zrealizować przy pomocy jednego z wcześniej zdefiniowanych algorytmów: drzewa decyzyjnego bądź klasyfikatora Bayes'a.

Na zakładce „data mining prediction" definiujemy mechanizmy predykcji danych:
W pierwszej kolejności musimy określić dane wejściowe do predykcji – opcja „select case table" pozwala wskazać wcześniej zdefiniowany widok. Następnie wskazujemy przy pomocy opcji „select model" wybieramy odpowiedni model analityczny – w naszym wypadku skorzystamy z algorytmu wykorzystującego drzewo decyzyjne.
Kolejnym krokiem jest wskazanie odpowiadających sobie pól w modelu i tabeli wyboru. W przypadku, gdy nazwy pól w tabeli są identyczne, kreator budowy modelu predykcji automatyczne powiąże odpowiednie pola. W naszym przypadku wystarczy jedynie zweryfikować wskazane powiązania.



Rys 5. Definicja modelu predykcji danych

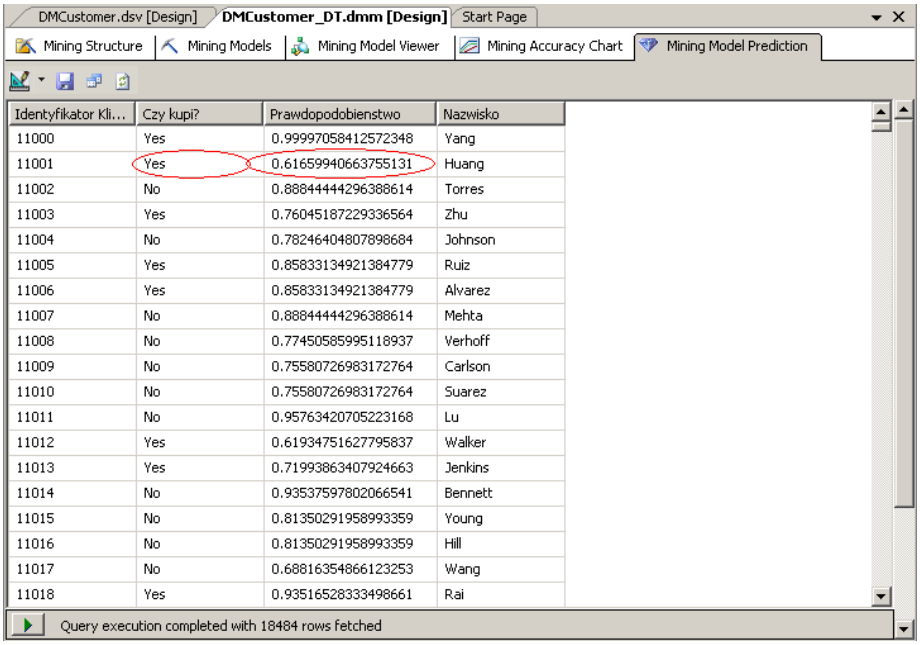
Ponadto poniżej zdefiniujemy pola jakie chcemy przeglądać w efekcie predykcji danych. Zwracam tutaj uwagę, że cały proces analizy danych oparty jest o metody statystyczne. W związku z tym, istnieje możliwość wykorzystania wbudowanych funkcji statystycznych do prezentacji np. prawdopodobieństwa trafności analizy danych. W naszym przypadku wykorzystamy funkcję predict() aby zobrazować tę informację. Ponadto w procesie analizy danych uzyskamy informację o identyfikatorze klienta oraz jego nazwisku. Dodatkowe pola, znajdujące się w tabeli wejściowej mogą być również wykorzystywane w procesie predykcji.

Zdefiniowany model predykcji, przy pomocy interfejsu graficznego, może być również zrealizowany przy pomocy języka DMX (Data Mining Expression):

SELECT
(t.[CustomerKey]) as [Identyfikator Klienta],

```
  ([DMCustomer_DT].[Bike Buyer]) as [Czy kupi?],
  (PredictProbability([DMCustomer_DT].[Bike Buyer])) as [Prawdopodobienstwo],
  (t.[LastName]) as [Nazwisko]
From
  [DMCustomer_DT]
PREDICTION JOIN
  OPENQUERY([Adventure Works DW],
    'SELECT
      [CustomerKey],
      [LastName],
      [Age],
      [MaritalStatus],
      [Gender],
      [YearlyIncome],
      [NumberChildrenAtHome],
      [HouseOwner],
      [NumberCarsOwned],
      [EnglishCountryRegionName],
      [Commute],
      [BikeBuyer]
    FROM
      [dbo].[vDMLabCustomerPredict]
    ') AS t
ON
  [DMCustomer_DT].[Age] = t.[Age] AND
  [DMCustomer_DT].[Marital Status] = t.[MaritalStatus] AND
  [DMCustomer_DT].[Gender] = t.[Gender] AND
  [DMCustomer_DT].[Yearly Income] = t.[YearlyIncome] AND
  [DMCustomer_DT].[Number Children At Home] = t.[NumberChildrenAtHome] AND
  [DMCustomer_DT].[House Owner] = t.[HouseOwner] AND
  [DMCustomer_DT].[Number Cars Owned] = t.[NumberCarsOwned] AND
  [DMCustomer_DT].[English Country Region Name] = t.[EnglishCountryRegionName] AND
  [DMCustomer_DT].[Commute] = t.[Commute] AND
  [DMCustomer_DT].[Bike Buyer] = t.[BikeBuyer]
```

Efektem predykcji danych jest tabela pokazująca szukane zależności.



Identyfikator Kli...	Czy kupi?	Prawdopodobienstwo	Nazwisko
11000	Yes	0.99997058412572348	Yang
11001	Yes	0.61659940663755131	Huang
11002	No	0.88844444296388614	Torres
11003	Yes	0.76045187229336564	Zhu
11004	No	0.78246404807898684	Johnson
11005	Yes	0.85833134921384779	Ruiz
11006	Yes	0.85833134921384779	Alvarez
11007	No	0.88844444296388614	Mehta
11008	No	0.77450585995118937	Verhoff
11009	No	0.75580726983172764	Carlson
11010	No	0.75580726983172764	Suarez
11011	No	0.95763420705223168	Lu
11012	Yes	0.61934751627795837	Walker
11013	Yes	0.71993863407924663	Jenkins
11014	No	0.93537597802066541	Bennett
11015	No	0.81350291958993359	Young
11016	No	0.81350291958993359	Hill
11017	No	0.68816354866123253	Wang
11018	Yes	0.93516528333498661	Rai

Rys 6. Efekt predykcji danych

Z uzyskanych wyników możemy stwierdzić, że klient Huang z prawdopodobieństwem 0,6165 zakupi rower.

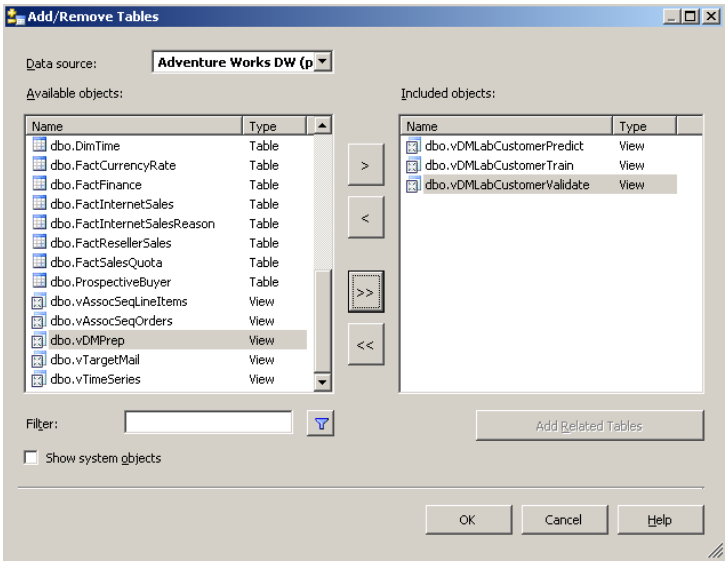
Jak wcześniej wspomnieliśmy, wybrany przez nas model analizy (drzewo decyzyjne), niekoniecznie musi być najlepszym algorytmem do rozwiązania naszego problemu. Wobec tego pojawia się pytanie, czy wybór innego algorytmu nie pozwoliłby osiągnąć lepszych efektów?
Ostatnim elementem analizy danych jest ocena „jakości algorytmu". Możemy to przedstawić w następujący sposób:
Po kilku miesiącach od rozpoczęcia sprzedaży rowerów w nowej lokalizacji, uzyskaliśmy informację o faktycznej sprzedaży rowerów. Należałoby teraz porównać przewidywaną sprzedaż przez poszczególne algorytmy z faktycznymi danymi.
W tym celu zdefiniujemy trzeci widok, będący danymi o faktycznej sprzedaży rowerów.

```
/******  View [dbo].[vDMLabCustomerValidate]  ******/

CREATE VIEW [dbo].[vDMLabCustomerValidate]  AS
SELECT
                c.[CustomerKey],
                c.[FirstName],
                c.[LastName],
                CASE
                        WHEN Month(GetDate()) < Month(c.[BirthDate])
                                THEN DateDiff(yy,c.[BirthDate],GetDate()) - 1
                        WHEN Month(GetDate()) = Month(c.[BirthDate])
                        AND Day(GetDate()) < Day(c.[BirthDate])
                                THEN DateDiff(yy,c.[BirthDate],GetDate()) - 1
                        ELSE DateDiff(yy,c.[BirthDate],GetDate())
                END AS [Age],
```

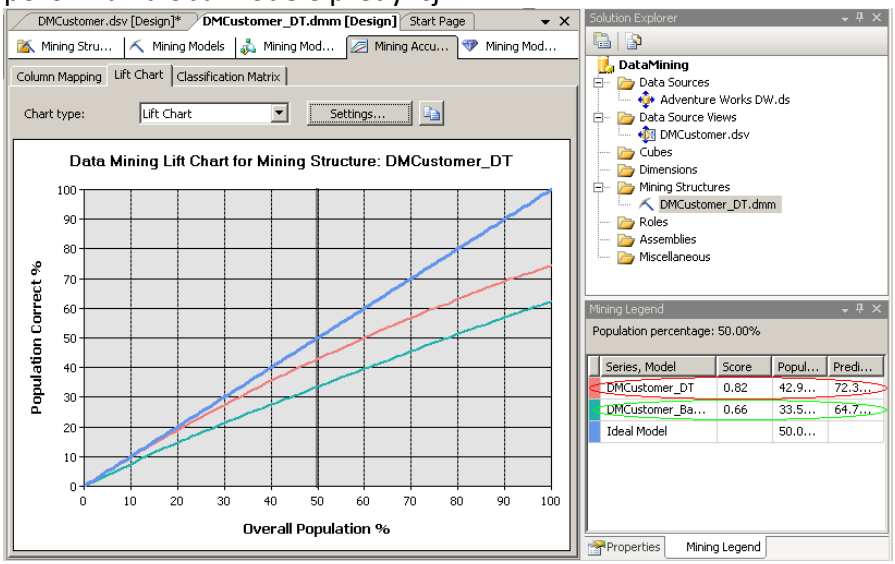
```
        c.[MaritalStatus],
        c.[Gender],
        c.[YearlyIncome],
        c.[NumberChildrenAtHome],
        CASE c.[HouseOwnerFlag] WHEN 0 THEN 'No' ELSE 'Yes' END as HouseOwner,
        c.[NumberCarsOwned],
        g.EnglishCountryRegionName,
        c.[CommuteDistance] As Commute,
        CASE x.[Bikes]
            WHEN 0 THEN 'No'
            ELSE 'Yes'
        END AS [BikeBuyer]
FROM
        [dbo].[DimCustomer] c INNER JOIN (
            SELECT
                [CustomerKey]
                ,[Region]
                ,[Age]
                ,Sum(
                    CASE [EnglishProductCategoryName]
                        WHEN 'Bikes' THEN 1
                        ELSE 0
                    END) AS [Bikes]
            FROM
                [dbo].[vDMPrep]
            GROUP BY
                [CustomerKey]
                ,[Region]
                ,[Age]
            ) AS [x]
        ON c.[CustomerKey] = x.[CustomerKey]
    join dimgeography g
    on c.geographykey = g.geographykey
go
```

Podobnie, jak w przypadku predykcji danych, również w tym przypadku podłączamy zdefiniowany dane do widoku źródła danych.



Rys 7. Rozbudowa widoków źródła danych

Następnie w modelu analizy danych w zakładce „Mining accuracy chart” wybieramy tabelę [dbo].[vDMLabCustomerValidate] jako tabelę wejściową. W naszym przypadku nie będziemy stosowali dodatkowych mechanizmów filtrujących. Wybieramy do procesu porównania oba modele predykcji.



Rys 8. Porównanie trafności algorytmów

Prezentowany wykres powyżej, pokazuje, że algorytm drzewa decyzyjnego znacznie lepiej przewiduje wyniki niż algorytm Bayes’a. Dla 100 procent populacji przewidywana trafność wyników wyniosła ponad 72 procent. Dla algorytmy Bayes’a wartość przewidywania wyniosła jedynie 64 procent. Na wykresie prezentowany jest również wykres idealnego algorytmu, gdy przewidywane jest 100 procent trafności.

Microsoft SQL Server 2005 umożliwia realizację pełnego procesu analizę danych: począwszy od procesu analizy danych treningowych poprzez predykcję danych aż do procesu weryfikacji danych związanych z predykcją.

NetSystem Tomasz Skurniak
CNI, CNE, MCT, MCSE, MCDBA, MCTS, MCITP
Ul. J. Burszty 25, 61-422 Poznań
E: Tomasz@Skurniak.pl
W: www.protis.pl
T: +48 601761013
F: +48 618308249