Politechnika Świętokrzyska

Sprawozdanie z projektu

Przedmiot: Hurtownie i eksploracja danych

Autor: Bartosz Dygas

Grupa: 1ID21A

1. Opis słowny

Hurtownia jest zbudowana w oparciu o sieć zakładów samochodowych. Dane w hurtowni są mapowane w strukturę gwiazdy. Zakłady obsługują w poszczególnych warsztatach, na poszczególnych halach i podnośnikach samochody należące do klientów. Do każdej naprawy wyznaczony jest jeden pracownik, który tą naprawą nadzoruje i po każdej naprawie generowana jest faktura. Naprawa jest też związana z awarią lub zestawem awarii, której uległ samochód i do której trzeba użyć konkretnych zestawów narzędzi i części. Migracja danych z bazy danych do hurtowni danych następuje raz dziennie.

Do tabel warsztatów, hal, podnosników – rekordy są dodawane bardzo rzadko. Następuje to zwykle kilka razy do roku gdy wybudowany zostanie nowy budynek firmy. Dodanie rekordów częściej następuje dla tabeli pracowników, narzędzi i części, natomiast dla tabel pojazdów, faktur, klientów rekordy są dodawane praktycznie codziennie.

Dla hurtowni przygotowanych zostało kilkanaście zapytań (w języku SQL), które pomogą w analizie danych zawartych w hurtowni. Zostały również zbudowane 3 modele eksploracji danych – klasyfikacja, asocjacja i grupowanie. Dzięki tym modelom można będzie pogrupować naprawy, zbadać jakie awarie najczęsciej następują wspólnie w danej marce samochodów oraz przewidzieć płeć klienta za pomocą narzędzia klasyfikacji.

2. Projekt hurtowni danych

Na tabele wymiarów składają się:

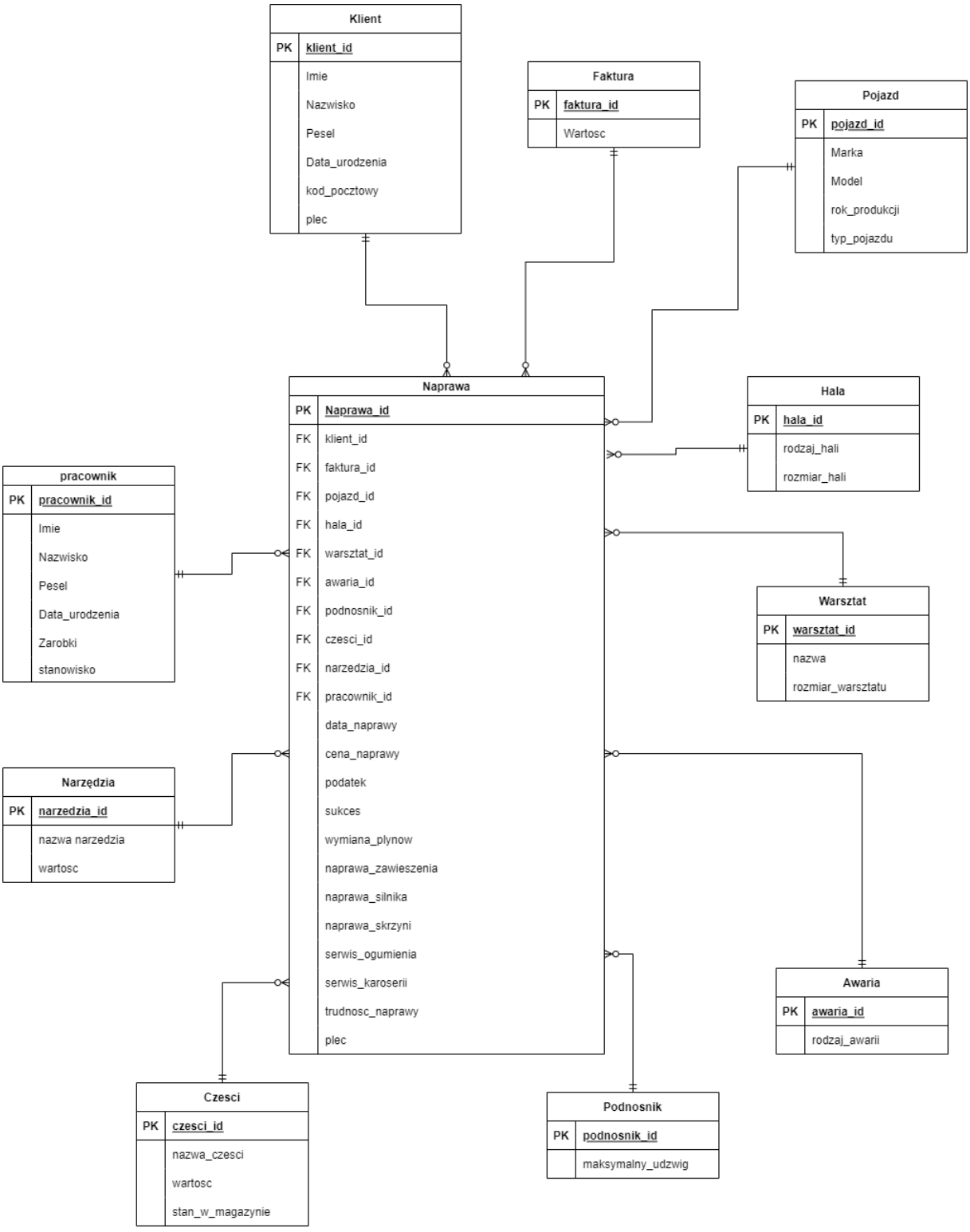
- Awaria - tabela zawierająca wszystkie spotkane rodzaje awarii samochodów
- Podnosnik – tabela zawiera wszystkie podnosniki uzywane w firmie
- Czesci – tabela zawiera zestawy czesci
- Narzedzia – tabela zawiera zestawy narzedzi
- Pracownik – wszyscy pracownicy
- Klient – wszyscy klienci
- Faktura – wszystkie faktury
- Pojazdów – wszystkie pojazdy z rozróżnieniem na markę, model, typ, rok produkcji
- Hala – rodzaje hal w warsztatach
- Warsztat – wszystkie warsztaty należące do firmy

Tabelą faktów jest tabela napraw. Zawiera ona klucze obce, za pomocą których łączy się z tabelami wymiarów oraz miary, które pozwalają na dokładniejszą eksplorację danych.

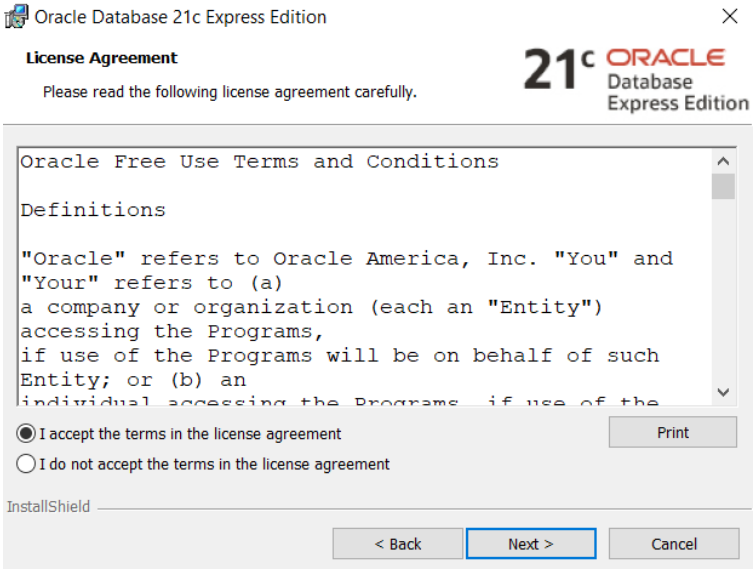
Miary tabeli faktów:

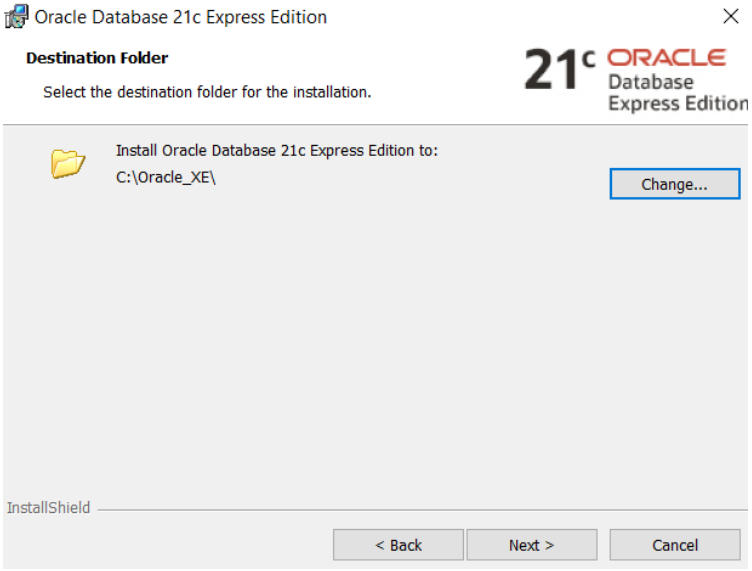
- Data_naprawy – data w formacie rok/miesiąc/dzień
- Cena_naprawy – całkowita cena jaką klient musi zapłacić za naprawę
- Podatek – podatek jaką zakład musi zapłacić za naprawę
- Sukces – przybiera wartości TAK/NIE i mówi o tym, czy naprawa się udała
- Wymiana_płynów – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy zostały wymienione płyny
- Naprawa_zawieszenia – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy było naprawiane zawieszenie
- Naprawa_silnika – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy był naprawiany silnik
- Naprawa_skrzyni – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy była naprawiana skrzynia
- Serwis_ogumienia – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy był dokonywany serwis opon
- Serwis_karoserii – przybiera wartości TAK/NIE i mówi o tym, czy podczas naprawy były dokonywane naprawy blacharskie
- Trudnosc_naprawy – przybiera wartości BARDZO_LATWY/LATWY/PRZECIETNY/TRUDNY/BARDZO_TRUDNY i mówi o tym, jak trudna i czasochłonna była naprawa według pracowników
- Plec – plec klienta

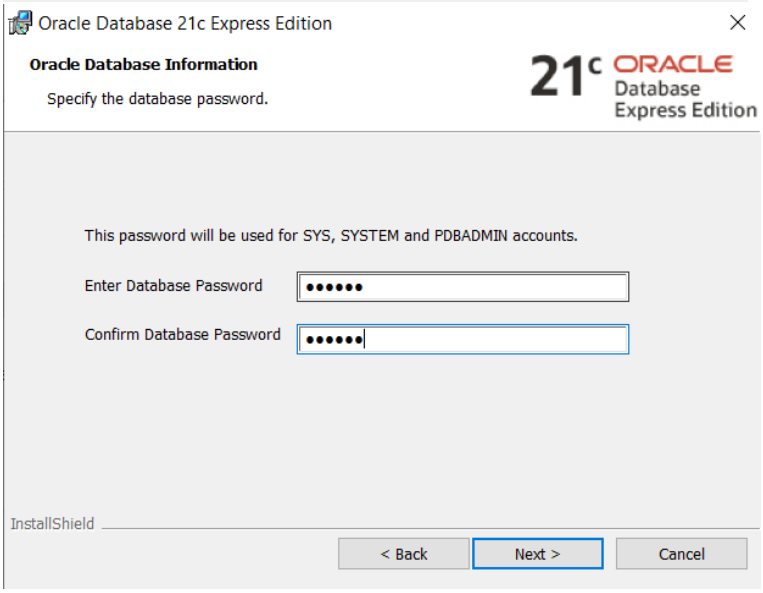
Schemat hurtowni danych

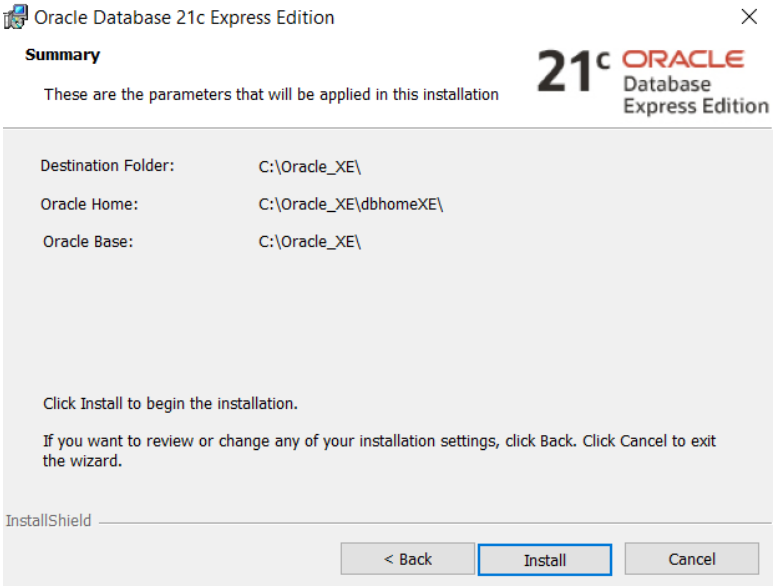


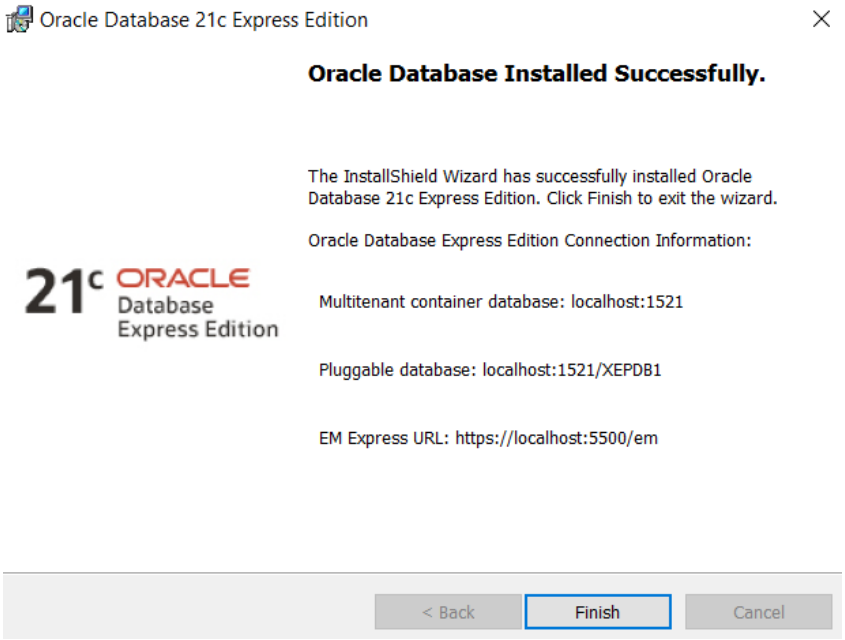
3. Instalacja środowiska pracy





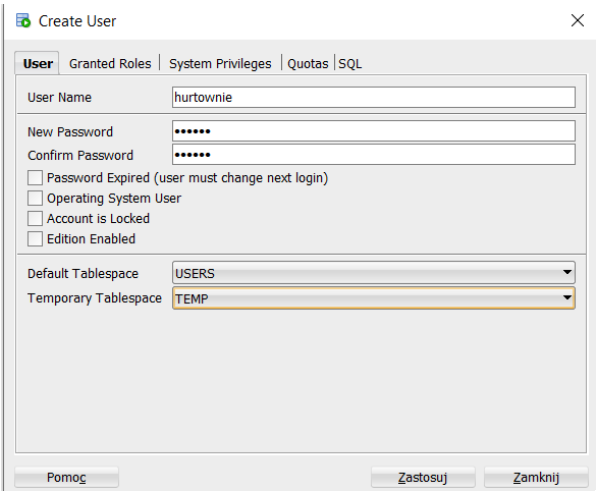






Po instalacji programu Oracle Database 21c Express Edition można skonfigurować Oracle developer. Pierwszym krokiem jest połączenie się z bazą danych, z kontem administratora:

Następnie możliwe jest stworzenie nowego użytkownika, dla którego stworzymy hurtownię.



4. Zasilenie hurtowni danych

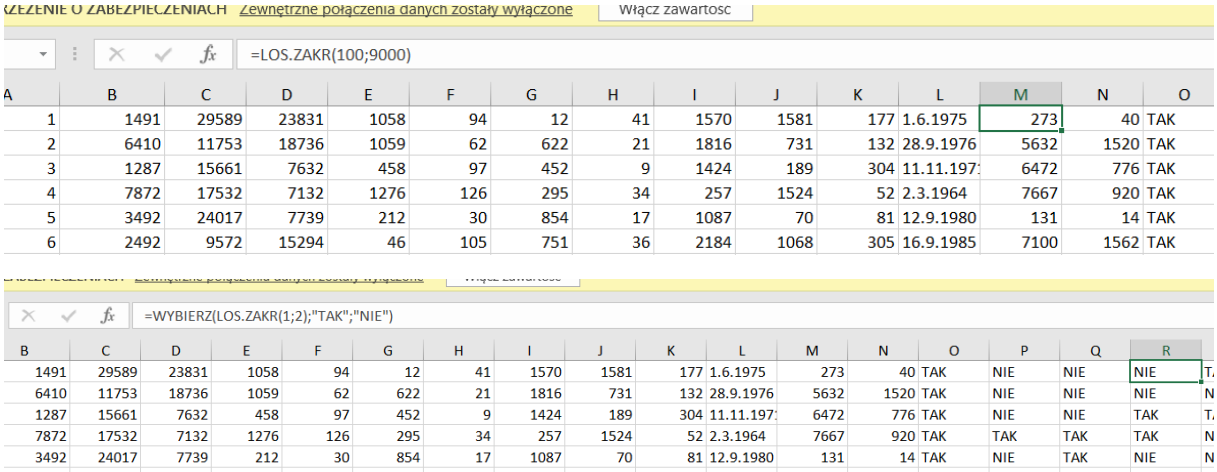
Aby zasilić hurtownię danych trzeba najpierw przygotować odpowiednie tabele.

```
CREATE TABLE klient
(
   klient_id numeric(10) not null,
   imie varchar2(50),
   nazwisko varchar2(100) not null,
   pesel varchar2(11),
   data_urodzenia DATE,
   kod_pocztowy varchar2(10),
   plec varchar2(1),
   CONSTRAINT klient_pk PRIMARY KEY (klient_id)
);

CREATE TABLE faktura
(
   faktura_id number(10) not null,
   Wartosc number(10,2),
   CONSTRAINT faktura_pk PRIMARY KEY (faktura_id)
);

CREATE TABLE pojazd
(
   pojazd_id number(10) not null,
   marka varchar2(40),
   model varchar2(80),
   rok_produkcji number(4),
   typ_pojazdu varchar2(25),
   CONSTRAINT pojazd_pk PRIMARY KEY (pojazd_id)
);

CREATE TABLE hala
(
   hala_id number(10) not null,
   rodzaj_hali varchar2(40),
   rozmiar_hali number(8),
   CONSTRAINT hala_pk PRIMARY KEY (hala_id)
);

CREATE TABLE warsztat
(
   warsztat_id number(10) not null,
   nazwa varchar2(50),
   rozmiar_warsztatu number(8),
   CONSTRAINT warsztat_pk PRIMARY KEY (warsztat_id)
);
```

```sql
CREATE TABLE awaria
(
  awaria_id number(10) not null,
  rodzaj_awarii varchar2(250) not null,
  CONSTRAINT awaria_pk PRIMARY KEY (awaria_id)
);

CREATE TABLE podnosnik
(
  podnosnik_id number(10) not null,
  maksymalny_udzwig number(6),
  CONSTRAINT podnosnik_pk PRIMARY KEY (podnosnik_id)
);

CREATE TABLE czesci
(
  czesci_id number(10) not null,
  nazwa_czesci varchar(100),
  wartosc number(7),
  stan_w_magazynie varchar(20),
  CONSTRAINT czesci_pk PRIMARY KEY (czesci_id)
);

CREATE TABLE narzedzia
(
  narzedzia_id number(10) not null,
  nazwa_narzedza varchar(100),
  wartosc number(7),
  CONSTRAINT narzedzia_pk PRIMARY KEY (narzedzia_id)
);

CREATE TABLE pracownik
(
  pracownik_id number(10) not null,
  imie varchar2(50),
  nazwisko varchar2(100) not null,
  pesel varchar2(11),
  data_urodzenia DATE,
  zarobki number(10),
  stanowisko varchar2(60),
  CONSTRAINT pracownik_pk PRIMARY KEY (pracownik_id)
);
```

```sql
CREATE TABLE naprawa
(
  naprawa_id number(12) not null,
  klient_id number(10),
  faktura_id number(10),
  pojazd_id number(10),
  hala_id number(10),
  warsztat_id number(10),
  awaria_id number(10),
  podnosnik_id number(10),
  czesci_id number(10),
  narzedzia_id number(10),
  pracownik_id number(10),
  data_naprawy date,
  cena_naprawy number(10,2),
  podatek number(10,2),
  sukces varchar(3),
  wymiana_plynow varchar(3),
  naprawa_zawieszenia varchar(3),
  naprawa_silnika varchar(3),
  naprawa_skrzyni varchar(3),
  serwis_ogumienia varchar(3),
  serwis_karoserii varchar(3),
  plec varchar(1),
  trudnosc_naprawy varchar(25),
  CONSTRAINT naprawa_pk PRIMARY KEY (naprawa_id),
  CONSTRAINT fk_klient FOREIGN KEY (klient_id) REFERENCES klient(klient_id),
  CONSTRAINT fk_faktura FOREIGN KEY (faktura_id) REFERENCES faktura(faktura_id),
  CONSTRAINT fk_pojazd FOREIGN KEY (pojazd_id) REFERENCES pojazd(pojazd_id),
  CONSTRAINT fk_hala FOREIGN KEY (hala_id) REFERENCES hala(hala_id),
  CONSTRAINT fk_warsztat FOREIGN KEY (warsztat_id) REFERENCES warsztat(warsztat_id),
  CONSTRAINT fk_awaria FOREIGN KEY (awaria_id) REFERENCES awaria(awaria_id),
  CONSTRAINT fk_podnosnik FOREIGN KEY (podnosnik_id) REFERENCES podnosnik(podnosnik_id),
  CONSTRAINT fk_czesci FOREIGN KEY (czesci_id) REFERENCES czesci(czesci_id),
  CONSTRAINT fk_narzedzia FOREIGN KEY (narzedzia_id) REFERENCES narzedzia(narzedzia_id),
  CONSTRAINT fk_pracownik FOREIGN KEY (pracownik_id) REFERENCES pracownik(pracownik_id)

);
```

Kolejnym krokiem jest wygenerowanie odpowiednich danych. Dane zostały wygenerowane w programie Microsoft Excel za pomocą funkcji LOS.ZAKR oraz WYBIERZ.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	1491	29589	23831	1058	94	12	41	1570	1581	177	1.6.1975	273	40	TAK
2	6410	11753	18736	1059	62	622	21	1816	731	132	28.9.1976	5632	1520	TAK
3	1287	15661	7632	458	97	452	9	1424	189	304	11.11.197	6472	776	TAK
4	7872	17532	7132	1276	126	295	34	257	1524	52	2.3.1964	7667	920	TAK
5	3492	24017	7739	212	30	854	17	1087	70	81	12.9.1980	131	14	TAK
6	2492	9572	15294	46	105	751	36	2184	1068	305	16.9.1985	7100	1562	TAK

B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1491	29589	23831	1058	94	12	41	1570	1581	177	1.6.1975	273	40	TAK	NIE	NIE	NIE
6410	11753	18736	1059	62	622	21	1816	731	132	28.9.1976	5632	1520	TAK	NIE	NIE	NIE
1287	15661	7632	458	97	452	9	1424	189	304	11.11.197	6472	776	TAK	NIE	NIE	TAK
7872	17532	7132	1276	126	295	34	257	1524	52	2.3.1964	7667	920	TAK	TAK	TAK	TAK
3492	24017	7739	212	30	854	17	1087	70	81	12.9.1980	131	14	TAK	NIE	TAK	NIE

Do załadowania danych do tabel użyte zostały skrypty składające się z plików .ctl potrzebny do zlokalizowania danych oraz bazy, do której dane mają być wgrane oraz pliku .bat wgrywającym wszystkie dane do tabel.

Przykładowy plik .ctl:

czesci.ctl — Notatnik

Plik Edycja Format Widok Pomoc

```
LOAD DATA
INFILE 'czesci.csv'

insert
into table czesci
fields terminated by ',' optionally enclosed by '"'
trailing nullcols
(czesci_ID,nazwa_czesci,wartosc,stan_w_magazynie)
```

W pliku tym zawarta jest nazwa tabeli, z której wgrywać będziemy dane, tabela i jej kolumny w hurtowni danych, do których wgrywane będą dane oraz separator kolumn w pliku .csv.

Plik .bat:

```
@echo off
set /p user=Podaj login:
set /p password=Podaj haslo:
sqlldr %user%/%password% control=awaria.ctl data=awaria.csv log=awaria.log
sqlldr %user%/%password% control=klient.ctl data=klient.csv log=klient.log
sqlldr %user%/%password% control=czesci.ctl data=czesci.csv log=czesci.log
sqlldr %user%/%password% control=faktura.ctl data=faktura.csv log=faktura.log
sqlldr %user%/%password% control=hala.ctl data=hala.csv log=hala.log
sqlldr %user%/%password% control=narzedzia.ctl data=narzedzia.csv log=narzedzia.log
sqlldr %user%/%password% control=podnosnik.ctl data=podnosnik.csv log=podnosnik.log
sqlldr %user%/%password% control=warsztat.ctl data=warsztat.csv log=warsztat.log
sqlldr %user%/%password% control=pracownik.ctl data=pracownik.csv log=pracownik.log
sqlldr %user%/%password% control=pojazd.ctl data=pojazd.csv log=pojazd.log
sqlldr %user%/%password% control=naprawa.ctl data=naprawa4.csv log=naprawa.log
echo Successfull
pause
```

Plik .bat wymaga podania loginu i hasła użytkownika. Za pomocą narzędzia sql loader pobiera dane z pliku .ctl oraz dane z pliku .csv. Dokonuje próby ładowania danych i wyniki umieszcza w pliku .log.

5. Zapytania do hurtowni danych

5.1 Zapytania Rollup

5.1.1 Pierwsze zapytanie typu Rollup sprawdza ile pieniędzy dla firmy wygenerowali poszczególni pracownicy z podziałem na lata. Dzięki temu można wynagrodzić najlepszych pracowników premiami i awansami.

```
select
    prac.nazwisko, prac.imie, nap.rok, nap.cena
from pracownik prac inner join(select
    EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok,
    pracownik_id,
    sum(cena_naprawy) as cena
from naprawa
group by rollup(EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), pracownik_id)
)nap on prac.pracownik_id = nap.pracownik_id
order by nap.cena DESC
```

Wynik:

	NAZWISKO	IMIE	ROK	CENA
1	Boguszewski	Wojciech	1976	98013
2	Górski	Marcin	1966	97932
3	Abacki	Michal	1964	97764
4	Kamiński	Tomek	1997	96803
5	Kowalski	Rafa³	1960	93864
6	Abacki	Sylwester	1992	92853
7	Kamiński	Damian	1993	90800
8	Wolny	Damian	1986	90187
9	Kowalczyk	Adam	1972	87734
10	Zieliński	Aleksander	1996	87456
11	Kamiński	Aleksander	1998	86628
12	Boguszewski	Bogdan	1963	86467
13	Wolny	Tomek	1960	86112
14	Boguszewski	Marcin	1963	85835
15	Czerwiński	Aleksander	1998	85750
16	Wiœniewski	Sebastian	1980	85417
17	Jab³onowski	Szymon	1989	85118
18	Czerwiński	Rafa³	1984	85017
19	Czerwiński	Damian	1994	84946
20	Kowalski	Aleksander	1972	84440
21	Danielak	Tomek	1986	84427
22	Wójcik	Rafa³	1987	84027

5.1.2. Zapytanie sprawdza ile napraw dokonały poszczególne warsztaty z podziałem na lata. Dzięki temu można na przykład zainwestować w zakłady, w których dokonuje się większej liczby napraw.

```
select war.nazwa as nazwa_warsztatu, nap.liczba_napraw, nap.rok
from warsztat war inner join(select warsztat_id, COUNT(naprawa_id) as liczba_napraw,
EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok
from naprawa
group by rollup(EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), warsztat_id)
)nap on war.warsztat_id = nap.warsztat_id
order by nap.liczba_napraw DESC
```

Wynik:

	NAZWA_WARSZTATU	LICZBA_NAPRAW	ROK
1	mekmech	38	1997
2	mechex	38	1991
3	warmek	36	1969
4	warmech	36	1976
5	mekmek	35	1979
6	fixmech	35	1999
7	fixauto	35	1972
8	autonet	34	1984
9	autoauto	34	1993
10	autoauto	34	1994
11	autoauto	34	1980
12	iwemech	34	1987
13	naprawawar	34	1980
14	fixex	33	1962
15	fixauto	33	1977
16	iweauto	33	1984
17	iweex	33	1962
18	maksnet	33	1961
19	werauto	33	1998
20	naprawaauto	33	1985
21	iwemat	33	1977
22	makswar	33	1967

5.1.3. Ostatnie zapytanie Rollup sprawdza ile razy wykonana została naprawa silnika w poszczególnych latach. Pozwala to oszacować awaryjność silników w poszczególnych latach i na tej podstawie można przewidzieć prawdopodobne przyszłe rodzaje usterek i napraw.

```
28  select
29  EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok,
30  naprawa_silnika as naprawa_silnika,
31  count(naprawa_id) as liczba
32  from naprawa
33  group by rollup(EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), naprawa_silnika)
34  order by rok DESC
```

Wynik:

	ROK	NAPRAWA_SILNIKA	LICZBA
1	(null)	(null)	100000
2	1999	(null)	2483
3	1999	TAK	1257
4	1999	NIE	1226
5	1998	TAK	1311
6	1998	NIE	1220
7	1998	(null)	2531
8	1997	NIE	1247
9	1997	TAK	1243
10	1997	(null)	2490
11	1996	(null)	2563
12	1996	TAK	1270
13	1996	NIE	1293
14	1995	(null)	2471
15	1995	TAK	1193
16	1995	NIE	1278
17	1994	NIE	1291
18	1994	TAK	1238
19	1994	(null)	2529
20	1993	NIE	1258
21	1993	(null)	2516

5.2 Zapytania Cube

5.2.1 Zapytanie oblicza średnią cenę naprawy w zależności od zestawu użytych narzędzi do tej naprawy oraz w zależności od roku naprawy. Pozwala to na oszacowanie jakich narzędzi najlepiej używać w celu zminimalizowania kosztów naprawy.

```
41  SELECT rok, narz.NAZWA_NARZEDZA AS uzyte_narzedzia, srednia_cena_naprawy
42  FROM narzedzia narz
43  RIGHT JOIN (
44      SELECT EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok,
45      narzedzia_id,
46      avg(cena_naprawy) as srednia_cena_naprawy
47      FROM naprawa
48      GROUP BY CUBE (EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), narzedzia_id)
49  ) nap ON nap.narzedzia_id = narz.narzedzia_id
50  order by srednia_cena_naprawy
51
```

Wynik:

	...	UZYTE_NARZEDZIA	SREDNIA_CENA_NAPRAWY
1	1967	komputer suwmiarka	100
2	1976	wywa¿nik suwmiarka	100
3	1977	miernik komputer	100
4	1996	latarka suwmiarka mlotek	100
5	1963	napelniacz pistolet miernik	101
6	1988	latarka suwmiarka	101
7	1964	miernik pistolet komputer	101
8	1992	mlotek srubokret	102
9	1979	klucz	102
10	1982	srubokret	103
11	1997	wywa¿nik	103
12	1976	komputer napelniacz komputer	103
13	1992	klucz	103
14	1976	napelniacz pistolet miernik	104
15	1992	pistolet	104
16	1990	mlotek	105
17	1981	napelniacz komputer pistolet	105
18	1960	napelniacz miernik miernik	105
19	1977	mlotek wywa¿nik klucz	106
20	1977	napelniacz mlotek	106
21	1965	latarka komputer pistolet	107
22	1966	napelniacz napelniacz	107
23	1995	komputer srubokret komputer	109

5.2.2 Następne zapytanie pozwala na sprawdzenie zależności średniego zapłaconego podatku od wymiany płynów w konkretnym dniu.

```
SELECT TO_DATE(data_naprawy, 'RR.MM.DD') as data, round(avg(podatek),2) as sredni_podatek, wymiana_plynow
FROM naprawa
    GROUP BY CUBE (TO_DATE(data_naprawy, 'RR.MM.DD'), wymiana_plynow)
    order by sredni_podatek
```

Wynik:

	DATA	SREDNI_PODATEK	WYMIANA_PLYNOW
1	67/12/11	11	NIE
2	80/08/20	12	TAK
3	79/05/15	16	TAK
4	87/12/01	17	NIE
5	83/04/07	18	NIE
6	90/12/09	18	NIE
7	88/01/19	21	TAK
8	85/11/25	22	NIE
9	83/05/24	23	NIE
10	63/02/20	24	TAK
11	99/05/17	24	TAK
12	74/10/16	25	NIE
13	77/02/02	26	NIE
14	80/06/07	26	NIE
15	82/02/04	27	NIE
16	78/10/13	28	TAK
17	94/03/28	28	NIE
18	96/01/13	30	TAK
19	70/12/09	30	NIE
20	66/08/10	30	TAK
21	82/02/26	31	NIE
22	97/09/23	31	TAK
23	80/06/06	32	TAK
24	68/11/16	32	TAK
25	63/04/13	32	TAK
26	60/11/07	32	TAK
27	90/01/03	33	TAK
28	75/08/23	33	TAK
29	75/08/23	33	(null)
30	78/03/18	34	TAK
31	79/01/24	34	(null)

5.2.3 Zapytanie pozwala sprawdzić zależność między naprawą zawieszenia, serwisem opon i średnim podatkiem. Dzięki temu firma może w przyszłości skupić się na serwisach, przez które może zapłacić mniejszy podatek

```
64  SELECT serwis_ogumienia, round(avg(podatek),2) as sredni_podatek, naprawa_zawieszenia
65  FROM naprawa
66      GROUP BY CUBE (serwis_ogumienia, naprawa_zawieszenia)
67      order by sredni_podatek
```

Wynik:

	SERWIS_OGUMI...	SREDNI_PODATEK	NAPRAWA_ZAWIESZENIA
1	NIE	904,31	NIE
2	NIE	905,3	(null)
3	NIE	906,28	TAK
4	(null)	906,67	NIE
5	(null)	907,78	(null)
6	(null)	908,89	TAK
7	TAK	909,01	NIE
8	TAK	910,26	(null)
9	TAK	911,52	TAK

5.3 Grouping sets

5.3.1 Zapytanie pozwala sprawdzić wydatki na naprawy poszczególnych klientów oraz w poszczególnych latach. Pomoże to zakładowi określi klientów którzy przynoszą największe zyski i zorientować się, w których latach zyski były najlepsze, a w których gorsze.

```
74  SELECT rok, klient.imie, klient.nazwisko, koszt_napraw
75  FROM klient
76  RIGHT JOIN (
77      SELECT EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok,
78      sum(cena_naprawy) as koszt_napraw,
79      klient_id
80      FROM naprawa
81      GROUP BY GROUPING SETS (EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), klient_id)
82  ) nap ON nap.klient_id = klient.klient_id
83  order by koszt_napraw DESC
```

Wynik:

	ROK	IMIE	NAZWISKO	KOSZT_NAPRAW
1	1967	(null)	(null)	12028204
2	1972	(null)	(null)	11947186
3	1992	(null)	(null)	11777506
4	1976	(null)	(null)	11723370
5	1970	(null)	(null)	11696490
6	1993	(null)	(null)	11593271
7	1991	(null)	(null)	11543131
8	1982	(null)	(null)	11539655
9	1987	(null)	(null)	11524928
10	1984	(null)	(null)	11470909
11	1996	(null)	(null)	11465470
12	1980	(null)	(null)	11463486
13	1963	(null)	(null)	11456717
14	1965	(null)	(null)	11448670
15	1989	(null)	(null)	11447576
16	1998	(null)	(null)	11419368
17	1975	(null)	(null)	11403804
18	1977	(null)	(null)	11398488
19	1981	(null)	(null)	11392811
20	1988	(null)	(null)	11362490
21	1997	(null)	(null)	11348420
22	1990	(null)	(null)	11303345

.
.
.

41	(null)	Adrian	Czerwiński	126617
42	(null)	Aleksander	Kamiński	126171
43	(null)	Wojciech	Kowalczyk	123424
44	(null)	Marcin	Kamiński	121894
45	(null)	Cezary	Górski	121257
46	(null)	Marcin	Wiœniewski	119661
47	(null)	Piotr	Abacki	118620
48	(null)	Rafa³	Szymański	117860
49	(null)	Adam	Jab³onowski	117268
50	(null)	Michal	Danielak	116914
51	(null)	Adam	Abacki	114872
52	(null)	Wojciech	Górski	114833
53	(null)	Adam	Szymański	114610
54	(null)	Krzysztof	Kamiński	114562
55	(null)	Cezary	Wiœniewski	114395
56	(null)	Sylwester	Górski	114197

5.3.2 Liczba napraw w poszczególnych latach oraz liczba napraw z wykorzystaniem poszczególnych części. Pomoże to m.in. oszacować jakie części wykorzystywane są najczęściej i w jakie warto się zaopatrywać.

```
SELECT rok, nazwa_czesci, liczba_napraw
FROM czesci
RIGHT JOIN (
    SELECT EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok,
    count(czesci_id) as liczba_napraw,
    czesci_id
    FROM naprawa
    GROUP BY GROUPING SETS (EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), czesci_id)
) nap ON nap.czesci_id = czesci.czesci_id
order by liczba_napraw DESC
```

Wynik:

	ROK	NAZWA_CZESCI	LICZBA_NAPRAW
1	1972	(null)	2605
2	1967	(null)	2595
3	1970	(null)	2569
4	1992	(null)	2569
5	1996	(null)	2563
6	1991	(null)	2560
7	1976	(null)	2556
8	1984	(null)	2541
9	1981	(null)	2533
10	1998	(null)	2531
11	1994	(null)	2529
12	1980	(null)	2528
13	1987	(null)	2523
14	1993	(null)	2516
15	1982	(null)	2508

.

.

.

41	(null)	rozrz¹d	54
42	(null)	klocki	54
43	(null)	olej plyn hamulcowy drzwi	54
44	(null)	drazek plyn hamulcowy klapa	51
45	(null)	tarcze	51
46	(null)	alternator drzwi	50
47	(null)	wahacz alternator wydech	50
48	(null)	drzwi silnik drazek	49
49	(null)	klocki alternator	48
50	(null)	plyn chlodniczy sprzeglo	48
51	(null)	silnik tarcze sprzeglo	48
52	(null)	olej skrzynia	48
53	(null)	plyn chlodniczy tarcze	48
54	(null)	chlodnica silnik	47
55	(null)	olej skrzynia	47
56	(null)	elektronika plyn hamulcowy	47

5.3.3 Zapytanie oblicza średnią cenę naprawy oraz średni podatek dla poszczególnych marek samochodów w poszczególnych latach. Pomaga to w oszacowaniu najbardziej dochodowych marek.

```
104  SELECT rok, p.marka, srednia_cena_naprawy, sredni_podatek
105  from pojazd p
106  RIGHT JOIN(
107      SELECT EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')) as rok, pojazd_id,
108      round(avg(cena_naprawy),2) as srednia_cena_naprawy,
109      round(avg(podatek),2) as sredni_podatek
110      FROM naprawa nap
111      GROUP BY GROUPING SETS((EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD'))),
112      (EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')), pojazd_id))
113  ) nap ON P.pojazd_ID = nap.pojazd_id
114  order by rok;
```

Wynik:

	ROK	MARKA	SREDNIA_CENA_NAPRAWY	SREDNI_PODATEK
1	1960	saab	1624	438
2	1960	mazda	2986	537
3	1960	jeep	2992	777
4	1960	jeep	500	95
5	1960	jeep	5054	808
6	1960	cadillac	8192	1474
7	1960	polonez	2720	761
8	1960	(null)	4562,58	912,21
9	1960	renault	3176	730
10	1960	honda	7775	1399
11	1960	hyundai	3004	600
12	1960	alfa romeo	8696	1304
13	1960	audi	1904	533
14	1960	mazda	5192	1194
15	1960	ford	1830	347
16	1960	citroen	2771	692
17	1960	opel	5654	1300
18	1960	mazda	5607	1457
19	1960	fiat	4710	989
20	1960	saab	4885	1465

5.4 Partycje obliczeniowe

5.4.1 Zapytanie sprawdza zależność trudności naprawy od jej ceny, wypisuje sumę cen za daną trudność oraz oblicza udział w tej sumie dla poszczególnego rekordu. Pomaga to w oszacowaniu czy trudniejsze naprawy są bardziej opłacalne, czy może warto z nich zrezygnować.

```
SELECT trudnosc_naprawy, cena_naprawy,
       sum(cena_naprawy) over (partition by trudnosc_naprawy) as suma_cen_za_trudnosc,
       round(100*cena_naprawy/(sum(cena_naprawy) over (partition by trudnosc_naprawy)), 5) "Udzial%"
from naprawa
order by cena_naprawy Desc;
```

Wynik:

	TRUDNOSC_NAPRAWY	CENA_NAPRAWY	SUMA_CEN_ZA_TRUDNOSC	Udzial%
1	bardzo latwa	9000	90925711	0,0099
2	bardzo trudna	9000	90975460	0,00989
3	trudna	9000	90145391	0,00998
4	przecietna	9000	90992134	0,00989
5	przecietna	9000	90992134	0,00989
6	latwa	9000	91563338	0,00983
7	trudna	8999	90145391	0,00998
8	przecietna	8999	90992134	0,00989
9	latwa	8999	91563338	0,00983
10	latwa	8999	91563338	0,00983
11	latwa	8999	91563338	0,00983
12	bardzo trudna	8999	90975460	0,00989
13	bardzo latwa	8999	90925711	0,0099
14	bardzo latwa	8999	90925711	0,0099
15	bardzo latwa	8999	90925711	0,0099
16	bardzo latwa	8999	90925711	0,0099
17	bardzo latwa	8999	90925711	0,0099

5.4.2 Następne zapytanie porównuje cenę naprawy z wykorzystaniem różnych zestawów narzędzi. Dzięki temu możemy zobaczyć, które narzędzia używane są w naprawach przynoszących największy zysk.

```
SELECT n.nazwa_narzedza as nazwa_narzedzia, nap.cena_naprawy, nap.suma_cen_wykorzystujac_narzedzie,nap.udzial
from narzedzia n
right join (select narzedzia_id, cena_naprawy, sum(cena_naprawy) over (partition by narzedzia_id) as suma_cen_wykorzystujac_narzedzie,
round(100*cena_naprawy/(sum(cena_naprawy) over (partition by narzedzia_id)), 5) as udzial
from naprawa) nap on n.narzedzia_id = nap.narzedzia_id
order by nap.cena_naprawy Desc;
```

Wynik:

	NAZWA_NARZEDZIA	CENA_NAPRAWY	SUMA_CEN_WYKORZYSTUJAC_NARZEDZIE	UDZIAL
1	klucz	9000	216791	4,15146
2	suwmiarka klucz napelniacz	9000	263516	3,41535
3	mlotek klucz pistolet	9000	271681	3,31271
4	klucz komputer komputer	9000	190696	4,71955
5	wywa¿nik komputer wywa¿nik	9000	238855	3,76798
6	latarka wywa¿nik srubokret	9000	194322	4,63149
7	suwmiarka srubokret mlotek	8999	167144	5,38398
8	komputer wywa¿nik latarka	8999	293900	3,06193
9	srubokret komputer wywa¿nik	8999	268327	3,35374
10	wywa¿nik klucz pistolet	8999	176437	5,1004
11	pistolet napelniacz klucz	8999	211423	4,2564
12	mlotek klucz miernik	8999	242085	3,71729
13	miernik	8999	254318	3,53848
14	mlotek klucz napelniacz	8999	277169	3,24676
15	mlotek klucz	8999	225589	3,98911
16	mlotek komputer	8999	246901	3,64478
17	latarka suwmiarka	8999	260696	3,45191
18	mlotek komputer	8999	306672	2,93441

5.4.3 Zapytanie pokazuje zależność między rodzajem awarii, a zapłaconym podatkiem. Dzięki temu można zrezygnować z napraw, które generują największe podatki dla zakładu.

```
SELECT a.rodzaj_awarii, nap.podatek, nap.suma_podatkow_dla_awarii,nap.udzial
from awaria a
right join (select awaria_id, podatek, sum(podatek) over (partition by awaria_id) as suma_podatkow_dla_awarii,
round(100*podatek/(sum(podatek) over (partition by awaria_id)), 5) as udzial
from naprawa) nap on a.awaria_id = nap.awaria_id
order by nap.podatek Desc;
```

Wynik:

	RODZAJ_AWARII	PODATEK	SUMA_PODATKOW_DLA_AWARII	UDZIAL
1	uszkodzony klocki hamulcowe	2699	101598	2,65655
2	obluzowany skrzynia	2698	100215	2,69221
3	obluzowany podwozie	2697	96506	2,79464
4	brak drzwi	2697	97500	2,76615
5	zardzewialy opona	2697	95995	2,80952
6	uszkodzony podwozie	2696	104882	2,57051
7	zardzewialy alternator	2696	91549	2,94487
8	cieknacy klocki hamulcowe	2696	119418	2,25762
9	porysowany opona	2696	103516	2,60443
10	zardzewialy skrzynia	2695	101938	2,64376
11	zardzewialy skrzynia	2695	87905	3,06581
12	zardzewialy podwozie	2695	84312	3,19646
13	brak karoseria	2695	84937	3,17294
14	cieknacy zaciski	2695	93801	2,8731
15	halasujacy si³owniki	2695	67423	3,99715
16	uszkodzony podwozie	2694	75039	3,59013
17	porysowany silnik	2693	96331	2,79557
18	zardzewialy drzwi	2693	112468	2,39446
19	cieknacy klapa	2692	83300	3,23169
20	uszkodzony alternator	2692	97302	2,76664

5.5 Okna ruchome

5.5.1 Zapytanie pozwala na oszacowanie, w który dzień i miesiąc zakłady przynoszą największe zyski. Dzięki temu można lepiej wyznaczyć urlopy pracownikom, przestoje warsztatów, czy inne przerwy w pracy.

```
SELECT distinct
    (EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))) AS MIESIAC,
    (EXTRACT(Day from TO_DATE(data_naprawy, 'RR.MM.DD'))) AS DZIEN,
     SUM(Cena_naprawy) OVER
        (PARTITION BY EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD')),
        EXTRACT(Day from TO_DATE(data_naprawy, 'RR.MM.DD'))
        ORDER BY EXTRACT(Day from TO_DATE(data_naprawy, 'RR.MM.DD'))
        RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS KWOTA_DZIENNA,
    SUM(cena_naprawy)
        OVER (PARTITION BY EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))
        ORDER BY EXTRACT(Day from TO_DATE(data_naprawy, 'RR.MM.DD'))
        RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS KWOTA_MIESIECZNA
FROM naprawa
ORDER BY MIESIAC, DZIEN;
```

Wynik:

	MIESIAC	DZIEN	KWOTA_DZIENNA	KWOTA_MIESIECZNA
1	1	1	1378370	1378370
2	1	2	1283690	2662060
3	1	3	1447340	4109400
4	1	4	1179157	5288557
5	1	5	1421700	6710257
6	1	6	1243265	7953522
7	1	7	1328005	9281527
8	1	8	1442929	10724456
9	1	9	1349217	12073673
10	1	10	1235527	13309200
11	1	11	1289347	14598547
12	1	12	1330428	15928975
13	1	13	1416027	17345002
14	1	14	1406738	18751740
15	1	15	1344717	20096457
16	1	16	1415284	21511741
27	1	27	1333530	36450336
28	1	28	1336879	37787215
29	2	1	1276346	1276346
30	2	2	1328551	2604897
31	2	3	1292636	3897533
32	2	4	1364311	5261844
33	2	5	1314438	6576282
34	2	6	1387378	7963660
35	2	7	1181178	9144838
36	2	8	1380633	10525471
37	2	9	1336095	11861566

5.5.2 Następne pytanie pozwala na określenie w którym miesiącu i roku płacone są najwyższe, a w którym najniższe podatki. To zapytanie może pomóc w decyzji, kiedy rozliczać się z podatków.

```
SELECT distinct
    (EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD'))) AS ROK,
    (EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))) AS MIESIAC,
     SUM(podatek) OVER
        (PARTITION BY EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD')),
        EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))
        ORDER BY EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))
        RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS KWOTA_MIESIECZNA,
    SUM(podatek)
        OVER (PARTITION BY EXTRACT(Year from TO_DATE(data_naprawy, 'RR.MM.DD'))
        ORDER BY EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'))
        RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS KWOTA_ROCZNA
FROM naprawa
ORDER BY ROK, MIESIAC;
```

Wynik:

	ROK	MIESIAC	KWOTA_MIESIECZNA	KWOTA_ROCZNA
1	1960	1	150559	150559
2	1960	2	179990	330549
3	1960	3	170661	501210
4	1960	4	219659	720869
5	1960	5	202691	923560
6	1960	6	196364	1119924
7	1960	7	168875	1288799
8	1960	8	168130	1456929
9	1960	9	186048	1642977
10	1960	10	205435	1848412
11	1960	11	179807	2028219
12	1960	12	187547	2215766
13	1961	1	175171	175171
14	1961	2	183578	358749
15	1961	3	202031	560780
16	1961	4	208488	769268
17	1961	5	186342	955610

5.5.3 Następne zapytanie pozwala na zliczenie trudnosci napraw dla poszczegolnych marek samochodow oraz ogólnym zliczeniu występujących trudności napraw. Pomoże to zorientować się w tym, jak trudne naprawy są obsługiwane najczęściej, oraz które samochody są najtrudniejsze, a które najłatwiejsze w naprawach.

```
SELECT P.marka, nap.trudnosc_naprawy,suma_dla_samochodu, suma_wszystkich
FROM pojazd P
JOIN (
    SELECT DISTINCT
        pojazd_id,
        trudnosc_naprawy,
        COUNT(*) OVER (PARTITION BY trudnosc_naprawy,
        pojazd_id ORDER BY pojazd_ID RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS suma_dla_samochodu,
         COUNT(*) OVER (PARTITION BY trudnosc_naprawy ORDER BY pojazd_ID
         RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS suma_wszystkich
    FROM naprawa
) nap ON nap.pojazd_ID = P.pojazd_ID ;
```

Wynik:

	MARKA	TRUDNOSC_NAPRAWY	SUMA_DLA_SAMOCHODU	SUMA_WSZYSTKICH
1	alfa romeo	bardzo latwa	1	17
2	volkswagen	bardzo latwa	2	24
3	mercedes	bardzo latwa	1	41
4	renault	bardzo latwa	1	45
5	honda	bardzo latwa	1	74
6	polonez	bardzo latwa	1	92
7	mercedes	bardzo latwa	3	95
8	fiat	bardzo latwa	1	100
9	volvo	bardzo latwa	1	102
10	cadillac	bardzo latwa	3	137
11	fiat	bardzo latwa	2	146
12	audi	bardzo latwa	1	176
13	lancia	bardzo latwa	1	179
14	lancia	bardzo latwa	3	182
15	citroen	bardzo latwa	1	183
16	ford	bardzo latwa	1	185
17	jeep	bardzo latwa	1	192
18	bmw	bardzo latwa	1	203
19	saab	bardzo latwa	1	215

5.6 Zapytania rankingowe

5.6.1 Pierwsze zapytanie rankingowe pomoże ustalić jakie rodzaje napraw ze względu na ich trudność przynoszą największe zyski

```
SELECT
    trudnosc_naprawy,
    SUM(cena_naprawy),
    DENSE_RANK() OVER (ORDER BY SUM(cena_naprawy) DESC) AS RANK
FROM naprawa
GROUP BY trudnosc_naprawy;
```

Wynik:

	TRUDNOSC_NAPRAWY	SUM(CENA_NAPRAWY)	RANK
1	latwa	91563338	1
2	przecietna	90992134	2
3	bardzo trudna	90975460	3
4	bardzo latwa	90925711	4
5	trudna	90145391	5

5.6.2 Drugie zapytanie pozwala ustalić, w którym miesiącu występuje największa liczba napraw.

```
SELECT
    EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD')) AS MIESIAC,
    COUNT(*) AS ILOSC,
    DENSE_RANK() OVER (ORDER BY COUNT(*) DESC) AS RANK
FROM NAPRAWA
GROUP BY EXTRACT(Month from TO_DATE(data_naprawy, 'RR.MM.DD'));
```

Wynik:

	MIESIAC	ILOSC	RANK
1	6	8490	1
2	12	8478	2
3	3	8467	3
4	1	8358	4
5	5	8346	5
6	11	8321	6
7	7	8320	7
8	10	8263	8
9	9	8252	9
10	2	8243	10
11	4	8236	11
12	8	8226	12

5.6.3 Ostatnie zapytanie rankingowe pozwala ustalić, który warsztat generuje największe zyski z napraw

```
SELECT W.nazwa, NAP.suma_PLN_z_napraw, nap.ranking FROM warsztat W
JOIN
(
    SELECT
        warsztat_id,
        SUM(cena_naprawy) AS suma_PLN_z_napraw,
        DENSE_RANK() OVER (ORDER BY SUM(cena_naprawy) DESC) AS ranking
    FROM naprawa
    GROUP BY warsztat_id
) nap ON W.warsztat_ID = nap.warsztat_id
order by nap.ranking;
```

Wynik:

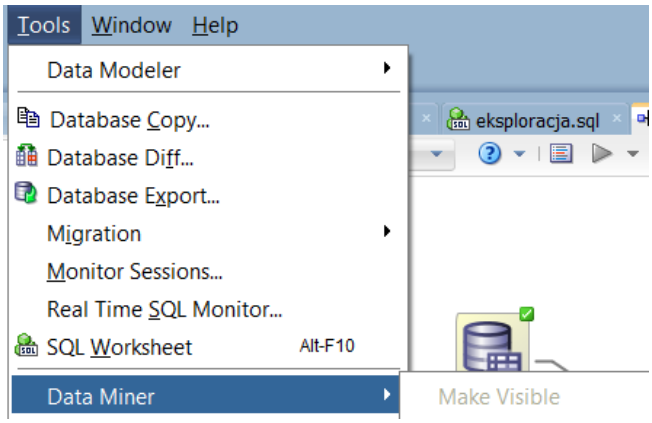
	NAZWA	SUMA_PLN_Z_NAPRAW	RANKING
1	mechauto	3928656	1
2	mekex	3855873	2
3	fixmech	3847145	3
4	fixmech	3835339	4
5	warmek	3819842	5
6	fixauto	3814315	6
7	naprawaauto	3761754	7
8	fixfix	3761461	8
9	werwar	3755383	9
10	warmech	3735283	10
11	autowar	3733897	11
12	maksmech	3733321	12
13	automat	3718488	13
14	fixauto	3718309	14
15	autoex	3714095	15
16	automat	3712418	16
17	mekmech	3707812	17
18	warmek	3702122	18
19	fixnet	3681243	19
20	iwewar	3679586	20
21	naprawamech	3674575	21

6. Model eksploracji danych

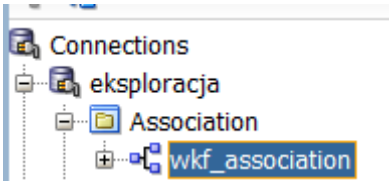
6.1 Model asocjacji

Odkrywanie asocjacji jest metodą eksploracji danych pozwalającą na znajdowanie związków między występowaniem grup elementów w zbiorach danych.

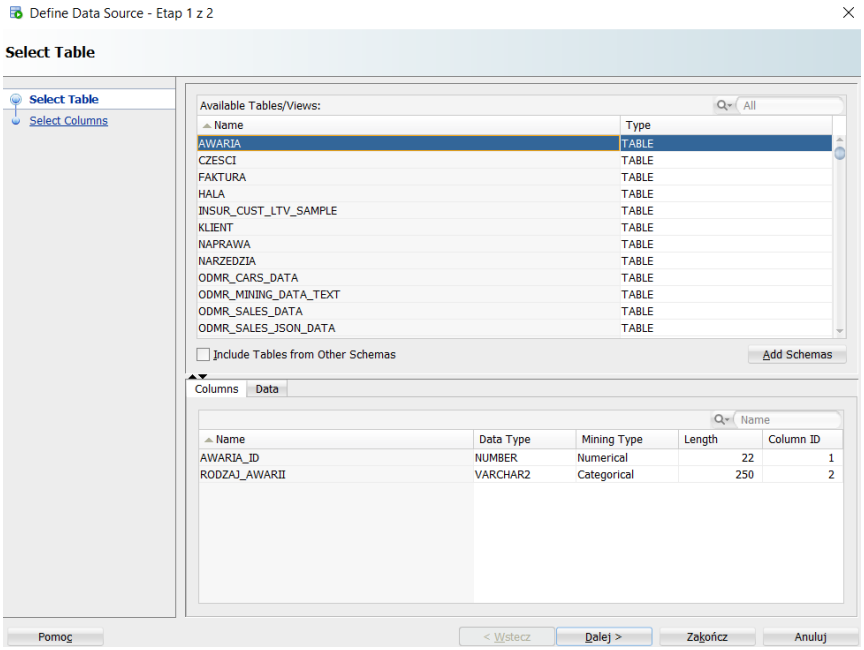
Pierwszym krokiem do zbudowania modelu odkrywania asocjacji jest uruchomienie narzędzia Data Miner w środowisku Oracle SQL developer.

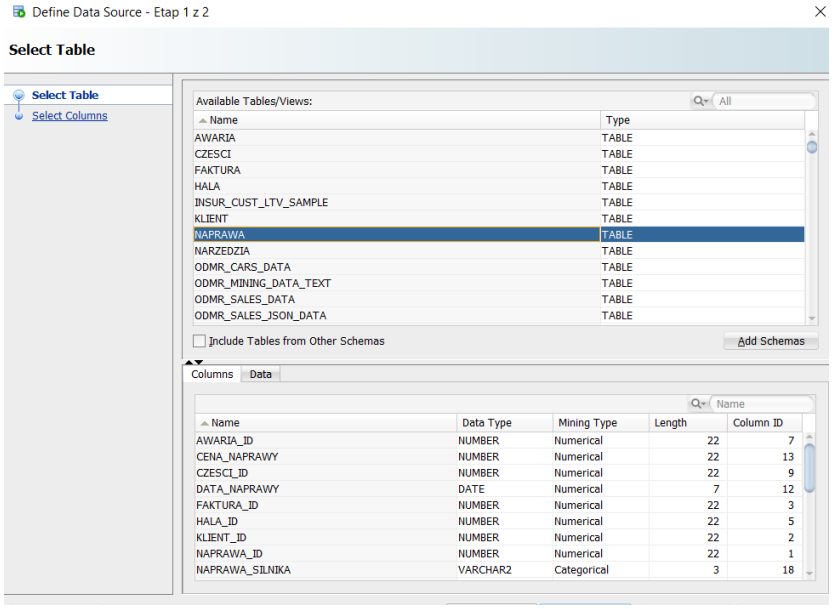


Następne kroki polegają na połączeniu się użytkownika, u którego znajdują się odpowiednie dane, do data minera, stworzenie nowego projektu i miejsca pracy (workspace).

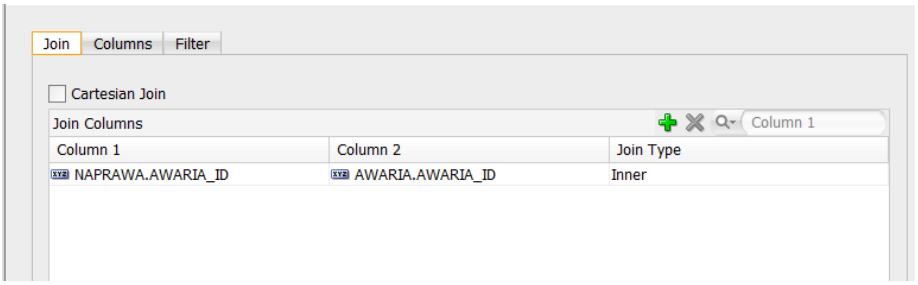


Moim celem w tym modelu jest sprawdzenie, które awarie towarzyszą wspólnie konkretnym pojazdom. W tym celu wybieram dwa obiekty Data_source i w ich miejsce wstawiam tabele – naprawa i awaria.

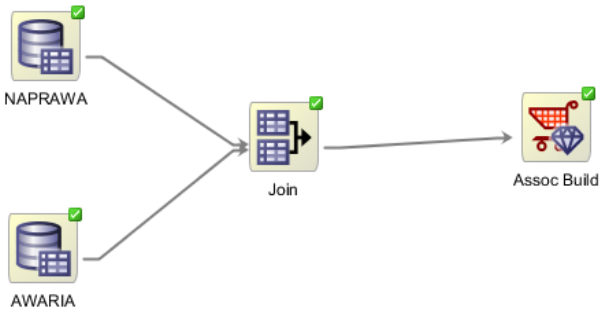




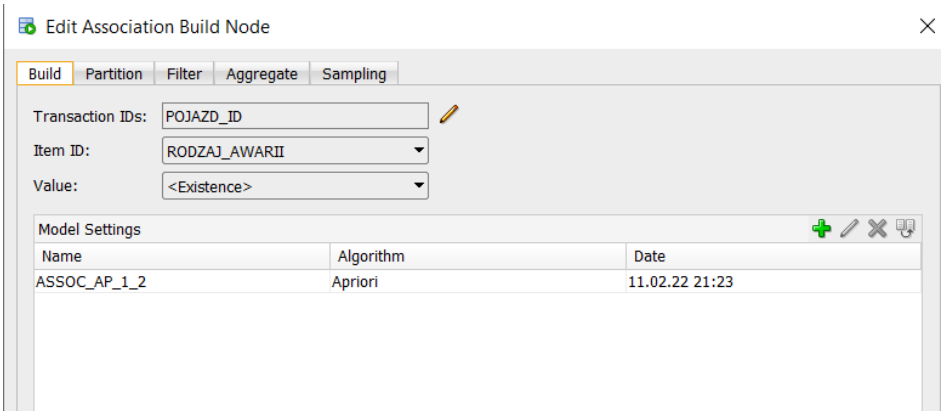
Następnie łączę obie tabele za pomocą narzędzia Join:



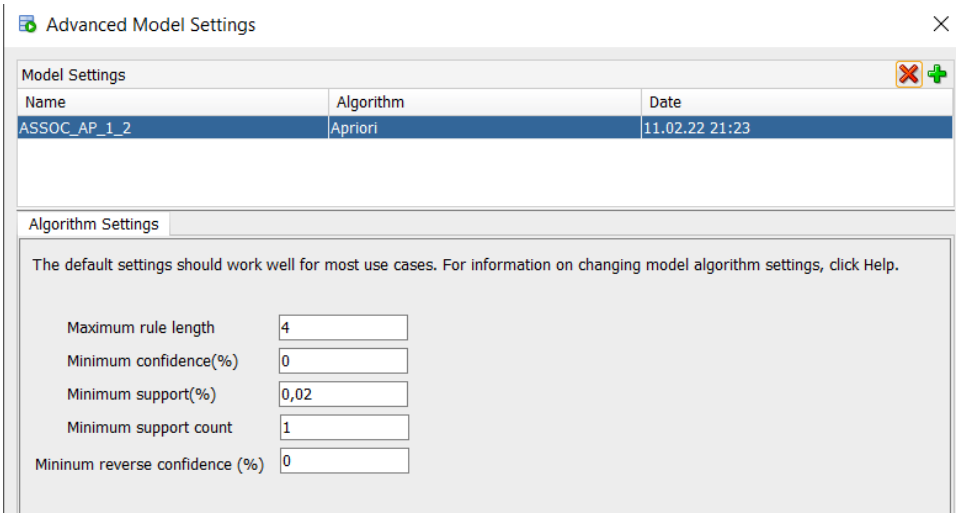
Na końcu należy jeszcze dodać narzędzie asocjacji i połączyć wszystkie obiekty strzałkami. Tak wygląda prezentowany model:



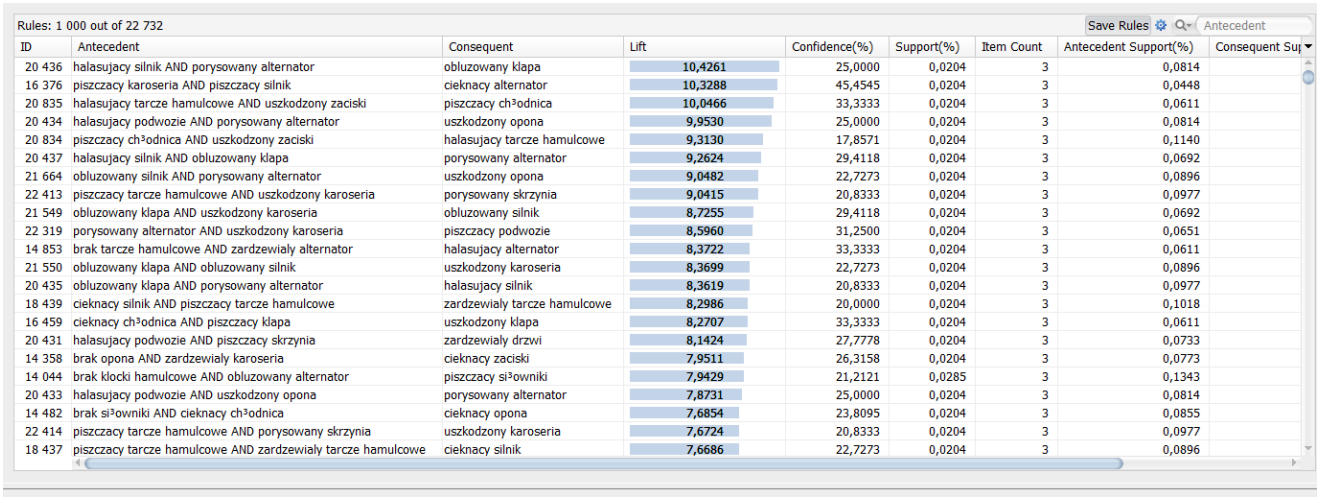
Po wybraniu asocjacji należy wybrać id transakcji – w moim przypadku będą to pojazdy, dla których będę badał awarie oraz itemID czyli wspomniana już awaria.



W opcjach zaawansowanych można wyznaczyć maksymalną długość reguł asocjacyjnych, minimalny poziom ufności i wsparcia:



Po uruchomieniu powyższego modelu można przyjrzeć się wynikom.



Rules: 1 000 out of 22 732 — Save Rules — Antecedent

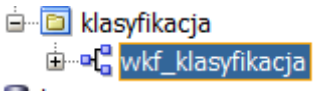
ID	Antecedent	Consequent	Lift	Confidence(%)	Support(%)	Item Count	Antecedent Support(%)	Consequent Su
20 436	halasujacy silnik AND porysowany alternator	obluzowany klapa	10,4261	25,0000	0,0204	3	0,0814	
16 376	piszczacy karoseria AND piszczacy silnik	cieknacy alternator	10,3288	45,4545	0,0204	3	0,0448	
20 835	halasujacy tarcze hamulcowe AND uszkodzony zaciski	piszczacy ch³odnica	10,0466	33,3333	0,0204	3	0,0611	
20 434	halasujacy podwozie AND porysowany alternator	uszkodzony opona	9,9530	25,0000	0,0204	3	0,0814	
20 834	piszczacy ch³odnica AND uszkodzony zaciski	halasujacy tarcze hamulcowe	9,3130	17,8571	0,0204	3	0,1140	
20 437	halasujacy silnik AND obluzowany klapa	porysowany alternator	9,2624	29,4118	0,0204	3	0,0692	
21 664	obluzowany silnik AND porysowany alternator	uszkodzony opona	9,0482	22,7273	0,0204	3	0,0896	
22 413	piszczacy tarcze hamulcowe AND uszkodzony karoseria	porysowany skrzynia	9,0415	20,8333	0,0204	3	0,0977	
21 549	obluzowany klapa AND uszkodzony karoseria	obluzowany silnik	8,7255	29,4118	0,0204	3	0,0692	
22 319	porysowany alternator AND uszkodzony karoseria	piszczacy podwozie	8,5960	31,2500	0,0204	3	0,0651	
14 853	brak tarcze hamulcowe AND zardzewialy alternator	halasujacy alternator	8,3722	33,3333	0,0204	3	0,0611	
21 550	obluzowany klapa AND obluzowany silnik	uszkodzony karoseria	8,3699	22,7273	0,0204	3	0,0896	
20 435	obluzowany klapa AND porysowany alternator	halasujacy silnik	8,3619	20,8333	0,0204	3	0,0977	
18 439	cieknacy silnik AND piszczacy tarcze hamulcowe	zardzewialy tarcze hamulcowe	8,2986	20,0000	0,0204	3	0,1018	
16 459	cieknacy ch³odnica AND piszczacy klapa	uszkodzony klapa	8,2707	33,3333	0,0204	3	0,0611	
20 431	halasujacy podwozie AND piszczacy skrzynia	zardzewialy drzwi	8,1424	27,7778	0,0204	3	0,0733	
14 358	brak opona AND zardzewialy karoseria	cieknacy zaciski	7,9511	26,3158	0,0204	3	0,0773	
14 044	brak klocki hamulcowe AND obluzowany alternator	piszczacy si³owniki	7,9429	21,2121	0,0285	3	0,1343	
20 433	halasujacy podwozie AND uszkodzony opona	porysowany alternator	7,8731	25,0000	0,0204	3	0,0814	
14 482	brak si³owniki AND cieknacy ch³odnica	cieknacy opona	7,6854	23,8095	0,0204	3	0,0855	
22 414	piszczacy tarcze hamulcowe AND porysowany skrzynia	uszkodzony karoseria	7,6724	20,8333	0,0204	3	0,0977	
18 437	piszczacy tarcze hamulcowe AND zardzewialy tarcze hamulcowe	cieknacy silnik	7,6686	22,7273	0,0204	3	0,0896	

Po wynikach widać m.in. że hałasujący silnik i porysowany alternator często spotykane są wspólnie z obluzowaną klapą. Czy też piszcząca chłodnica i uszkodzone zaciski towarzyszą hałasującym tarczom hamulcowym.

6.2 Klasyfikacja

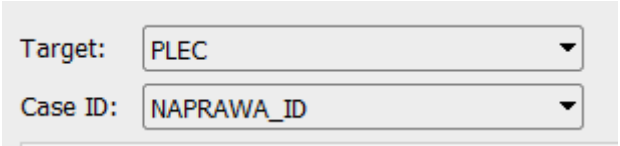
Klasyfikacja jest metodą analizy danych, której celem jest predykcja wartości określonego atrybutu w oparciu o pewien zbiór danych treningowych.

Do zbudowania modelu klasyfikacji stworzony został nowy projekt i nowy workspace



Model zbudowany jest tylko z 2 obiektów – data source, w którym znajduje się tabela i rekordy naprawy oraz Class Build pozwalającym na dokonanie klasyfikacji.

Celem klasyfikacji będzie przewidzenie płci klienta w zależności od dokonanych napraw. Co za tym idzie, zmienną celu jest płeć.

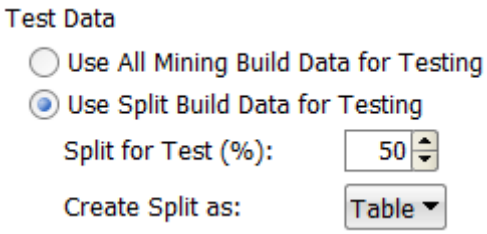


Klasyfikacja została dokonana w oparciu o 4 modele. Model naiwny Bayes'a, drzewo decyzyjne, maszynę wektorów nośnych oraz ogólny model liniowy

Model Settings

Name	Algorithm	Date	Data Usage
CLAS_GLM_1_3	Generalized Linear Model	11.02.22 21:59	
CLAS_SVM_1_3	Support Vector Machine	11.02.22 21:59	
CLAS_DT_1_3	Decision Tree	11.02.22 21:59	
CLAS_NB_1_3	Naive Bayes	11.02.22 21:59	

Podział na dane testowe – 50% rekordów został przeznaczony na dane testowe



Ogólny model liniowy:

Details

Name	Value
Number of Rows	49 943
Number of Parameters	24
Model Converged	Yes
Valid Covariance Matrix	Yes
Dependent Mean	0,5
-2 Log Likelihood of the Intercept Only Model	34 617,739705
-2 Log Likelihood of the Model	34 610,39954162
Likelihood Ratio Degrees of Freedom	23
Correct Prediction Percentage	50,6618%
Incorrect Prediction Percentage	49,3382%
Tied Cases Prediction	0,0000%
Rank Deficiency	0
Akaike's Criterion Intercept Only Model	34 619,739705
Akaike's Criterion Model	34 658,39954162
Likelihood Ratio Chi-square	7,34016338
Likelihood Ratio Chi-square Probability	0,99918773
Pseudo R-square Cox and Snell	0,00014696
Pseudo R-square Nagelkerke	0,00029392
Schwarz's Criterion Intercept Only Model	34 628,55834264
Schwarz's Criterion Model	34 870,04684484
Iterations	3

Target Value: K Sort by absolute value

Coefficients: 24 out of 24

Attribute	Value	Standardized Coefficient	Coefficient	Exp(Coefficient)
CENA_NAPRAWY		0,00764654	0,00000541	1,00000541
WARSZTAT_ID		0,00726051	0,00035638	1,00035644
PODATEK		-0,00673341	-0,00002032	0,99997968
POJAZD_ID		-0,00637897	-0,00000161	0,99999839
NARZEDZIA_ID		-0,00587273	-0,00001843	0,99998157
KLIENT_ID		0,00555701	0,00000433	1,00000433
CZESCI_ID		-0,00472905	-0,00000944	0,99999056
HALA_ID		0,00432200	0,00002092	1,00002093
NAPRAWA_ZAWIESZENIA	NIE	-0,00395607	-0,02870143	0,97170654
PODNOSNIK_ID		0,00283230	0,00035738	1,00035745
PRACOWNIK_ID		0,00260532	0,00004697	1,00004697
AWARIA_ID		-0,00245136	-0,00001542	0,99998458
TRUDNOSC_NAPRAWY	bardzo trudna	0,00199949	0,01811553	1,01828061
NAPRAWA_SILNIKA	TAK	-0,00181516	-0,01316948	0,98691686
FAKTURA_ID		0,00149234	0,00000028	1,00000028
WYMIANA_PLYNOW	NIE	0,00146285	0,01061346	1,01066998
SUKCES	TAK	-0,00134204	-0,00973666	0,99031058
SERWIS_OGUMIENIA	TAK	-0,00110471	-0,00801485	0,99201718
SERWIS_KAROSERII	TAK	0,00056505	0,00409955	1,00410797
TRUDNOSC_NAPRAWY	bardzo latwa	-0,00038601	-0,00349396	0,99651214
TRUDNOSC_NAPRAWY	trudna	0,00022470	0,00205634	1,00205846
TRUDNOSC_NAPRAWY	przecietna	-0,00003603	-0,00032653	0,99967352
NAPRAWA_SKRZYNI	TAK	0,00002070	0,00015019	1,00015020
<Intercept>		0,00000000	-0,00191534	0,99808650

Target Value: M ☑ Sort by absolute value

Coefficients: 24 out of 24

Attribute	Value	Standardized Coefficient	Coefficient	Exp(Coefficient)
CENA_NAPRAWY		-0,00764654	-0,00000541	0,99999459
WARSZTAT_ID		-0,00726051	-0,00035638	0,99964369
PODATEK		0,00673341	0,00002032	1,00002032
POJAZD_ID		0,00637897	0,00000161	1,00000161
NARZEDZIA_ID		0,00587273	0,00001843	1,00001843
KLIENT_ID		-0,00555701	-0,00000433	0,99999567
CZESCI_ID		0,00472905	0,00000944	1,00000944
HALA_ID		-0,00432200	-0,00002092	0,99997908
NAPRAWA_ZAWIESZENIA	NIE	0,00395607	0,02870143	1,02911729
PODNOSNIK_ID		-0,00283230	-0,00035738	0,99964268
PRACOWNIK_ID		-0,00260532	-0,00004697	0,99995303
AWARIA_ID		0,00245136	0,00001542	1,00001542
TRUDNOSC_NAPRAWY	bardzo trudna	-0,00199949	-0,01811553	0,98204757
NAPRAWA_SILNIKA	TAK	0,00181516	0,01316948	1,01325658
FAKTURA_ID		-0,00149234	-0,00000028	0,99999972
WYMIANA_PLYNOW	NIE	-0,00146285	-0,01061346	0,98944266
SUKCES	TAK	0,00134204	0,00973666	1,00978422
SERWIS_OGUMIENIA	TAK	0,00110471	0,00801485	1,00804706
SERWIS_KAROSERII	TAK	-0,00056505	-0,00409955	0,99590884
TRUDNOSC_NAPRAWY	bardzo latwa	0,00038601	0,00349396	1,00350007
TRUDNOSC_NAPRAWY	trudna	-0,00022470	-0,00205634	0,99794577
TRUDNOSC_NAPRAWY	przecietna	0,00003603	0,00032653	1,00032658
NAPRAWA_SKRZYNI	TAK	-0,00002070	-0,00015019	0,99984982
<Intercept>		0,00000000	0,00191534	1,00191717

Atrybuty Standardized Coefficient i Coefficient oznaczają współczynnik korelacji i mogą przyjmować wartości w granicach <-1, 1>. Im wartość bliższa jest 0, tym mniejsza jest korelacja między klasą, a zmienną opisową. Dane zostały posortowane według standaryzowanego współczynnika korelacji malejąco

Model osiągnął precyzję na poziomie 50,6618%. Jest to słaby wynik i jest spowodowany prawdopodobnie tym, że dane w hurtowni zostały wygenerowane losowo. Oznacza to, że szansa na poprawną klasyfikację wynosi około 50%.

Maszyna wektorów nośnych

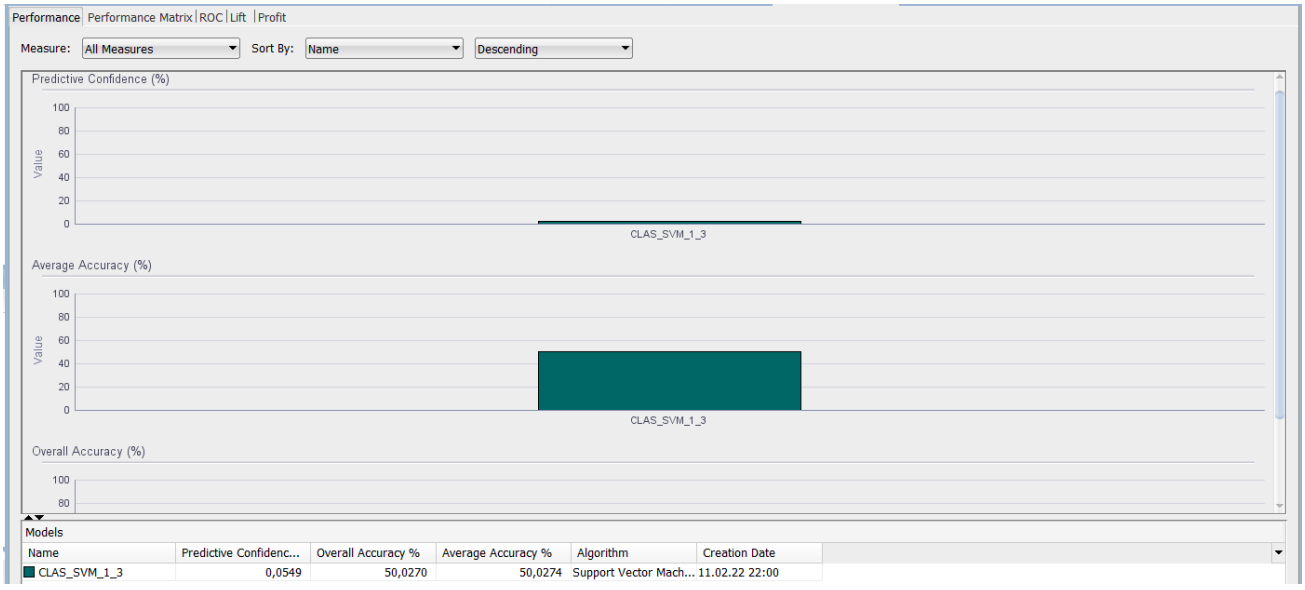
Target Value 1: M Target Value 2: K Sort by absolute value Fetch Size: 1 000 Query

Propensities: 32 out of 32 Attribute

Attribute	Value	Propensity for M	Propensity for K
NAPRAWA_ZAWIESZENIA	NIE	0,99998356	-0,99998356
NAPRAWA_ZAWIESZENIA	TAK	-0,99998356	0,99998356
WARSZTAT_ID		-0,00000343	0,00000343
TRUDNOSC_NAPRAWY	bardzo trudna	-0,00000338	0,00000338
POJAZD_ID		0,00000274	-0,00000274
CENA_NAPRAWY		-0,00000267	0,00000267
NARZEDZIA_ID		0,00000252	-0,00000252
KLIENT_ID		-0,00000247	0,00000247
PODATEK		0,00000231	-0,00000231
HALA_ID		-0,00000205	0,00000205
CZESCI_ID		0,00000189	-0,00000189
TRUDNOSC_NAPRAWY	bardzo latwa	0,00000175	-0,00000175
NAPRAWA_SILNIKA	TAK	0,00000139	-0,00000139
NAPRAWA_SILNIKA	NIE	-0,00000139	0,00000139
PRACOWNIK_ID		-0,00000118	0,00000118
AWARIA_ID		0,00000117	-0,00000117
PODNOSNIK_ID		-0,00000116	0,00000116
WYMIANA_PLYNOW	NIE	-0,00000105	0,00000105
WYMIANA_PLYNOW	TAK	0,00000105	-0,00000105
SUKCES	TAK	0,00000098	-0,00000098
SUKCES	NIE	-0,00000098	0,00000098
TRUDNOSC_NAPRAWY	przecietna	0,00000083	-0,00000083
TRUDNOSC_NAPRAWY	latwa	0,00000077	-0,00000077
SERWIS_OGUMIENIA	NIE	-0,00000073	0,00000073
SERWIS_OGUMIENIA	TAK	0,00000073	-0,00000073
SERWIS_KAROSERII	TAK	-0,00000063	0,00000063
SERWIS_KAROSERII	NIE	0,00000063	-0,00000063
FAKTURA_ID		-0,00000059	0,00000059
NAPRAWA_SKRZYNI	NIE	0,00000010	-0,00000010
NAPRAWA_SKRZYNI	TAK	-0,00000010	0,00000010
TRUDNOSC_NAPRAWY	trudna	0,00000003	-0,00000003
<Intercept>		0,00000001	-0,00000001

Model ten pozwala oszacować jakie zmienne miały największy wpływ na podjęcie dezycji o tym, czy klient jest kobietą czy mężczyzną. W tym przypadku Model stwierdził, że jeśli w naprawie występuje naprawa zawieszenia to klientem jest kobieta, w przeciwnym przypadku klientem jest mężczyzna. Pozostałe zmienne zostały praktycznie pominięte.

W przypadku użycia algorytmu maszyny wektorów nośnych szansa na poprawną klasyfikację również nieznacznie tylko przekroczyła 50%:

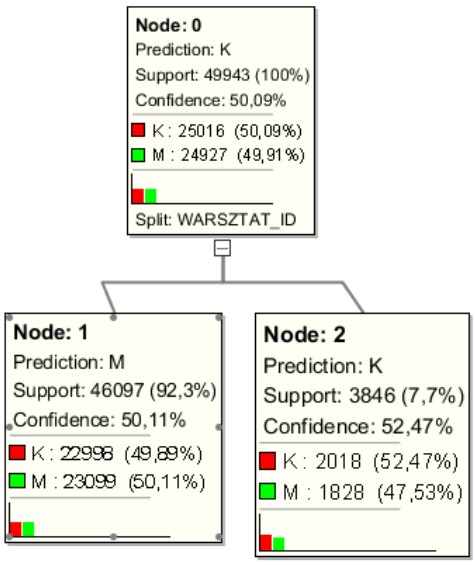


Drzewa decyzyjne

Algorytm ten buduje drzewa decyzyjne klasyfikujące dane na podstawie zbudowanych w trakcie uczenia zestawu reguł decyzyjnych. Jest to metoda przejrzysta i łatwa w odczytaniu.

W wyniku ich użycia otrzymujemy warunki, jakie muszą spełniać zmienne opisowe aby rekord został sklasyfikowany do poprawnej klasy. W tym przypadku również losowe generowanie danych sprawiło, że wyniki nie są zbyt użyteczne. Model podzielił rekordy w zależności od zmiennej warsztat_id. Według modelu wartość zmiennej warsztat_id mniejsza niż 118,5 wskazuje, że naprawa została wykonana dla klienta, który jest mężczyzną.

Ufność dla obu predykcji wynosi ok. 50%.

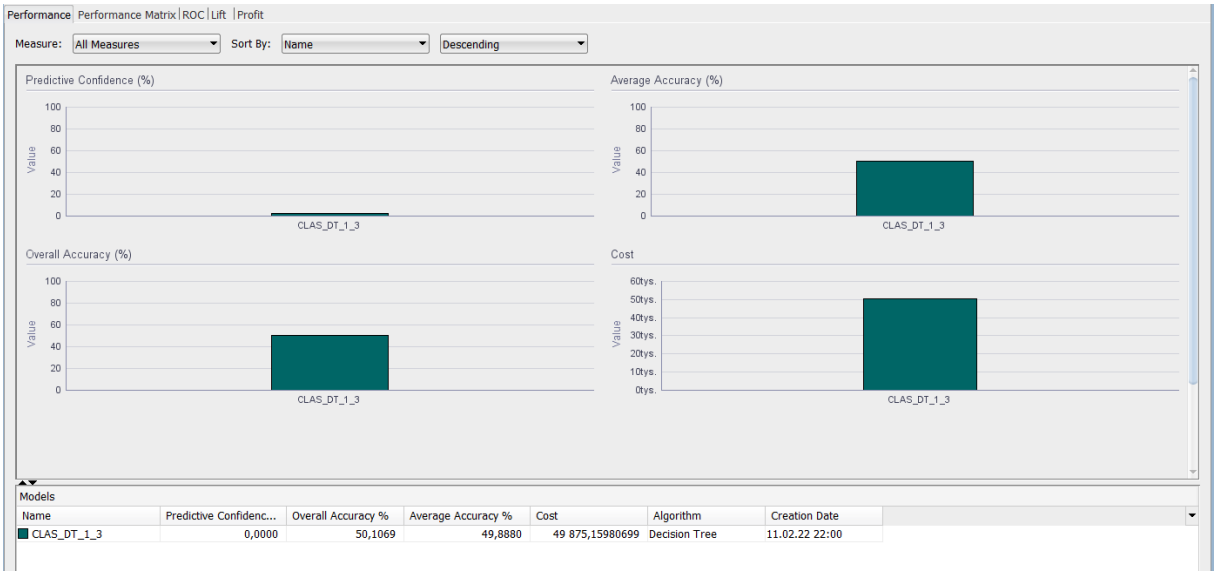


Node Rule:

```
If   WARSZTAT_ID <= 118,5

Then M
```

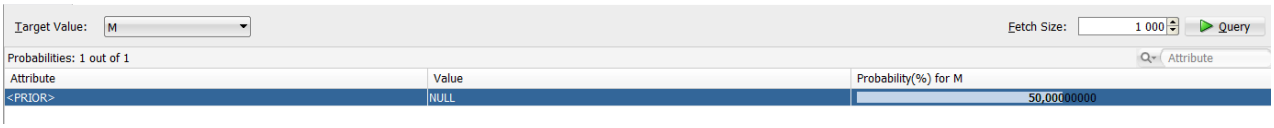
Confidence	0.5010955159771786
Support	0.9229922111206775

Szansa na poprawną klasyfikację ponownie nieznacznie przegracza 50%.

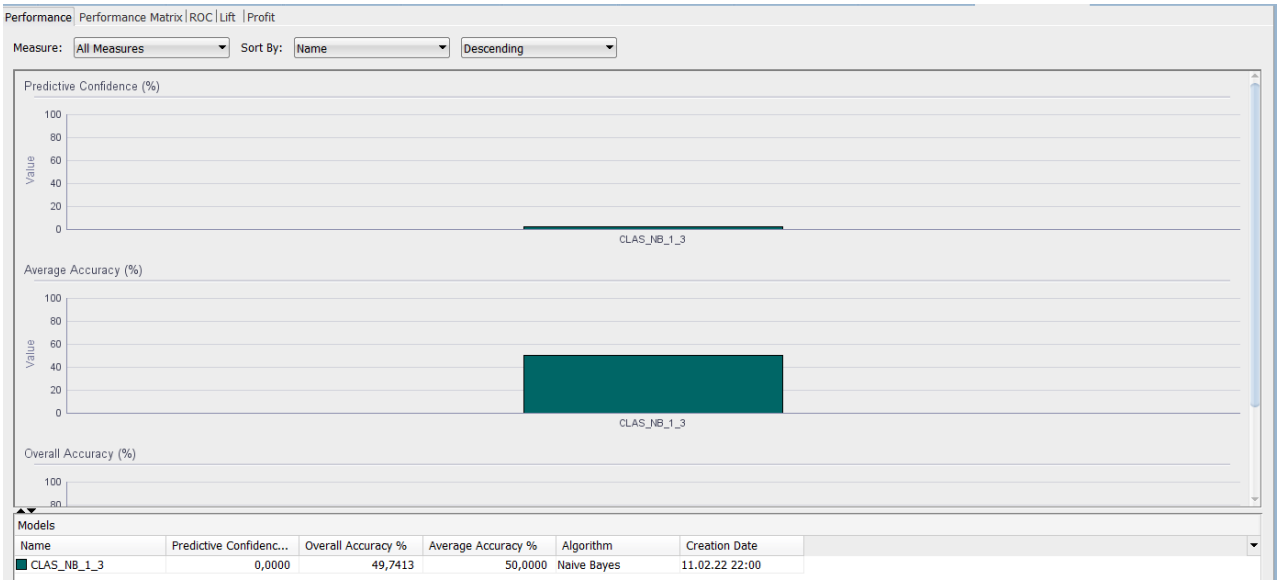


Naiwny algorytm Bayes'a

Niestety – jedyna wiadomość jaką można odczytać z tego algorytmu to prawdopodobieństwo wylosowania mężczyzny równa 50%. Ponownie może to być spowodowane losowym generowaniem danych do hurtowni.



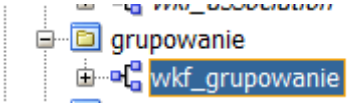
Tym razem szansa na poprawną kwalifikację spadła poniżej 50%:



6.3 Grupowanie

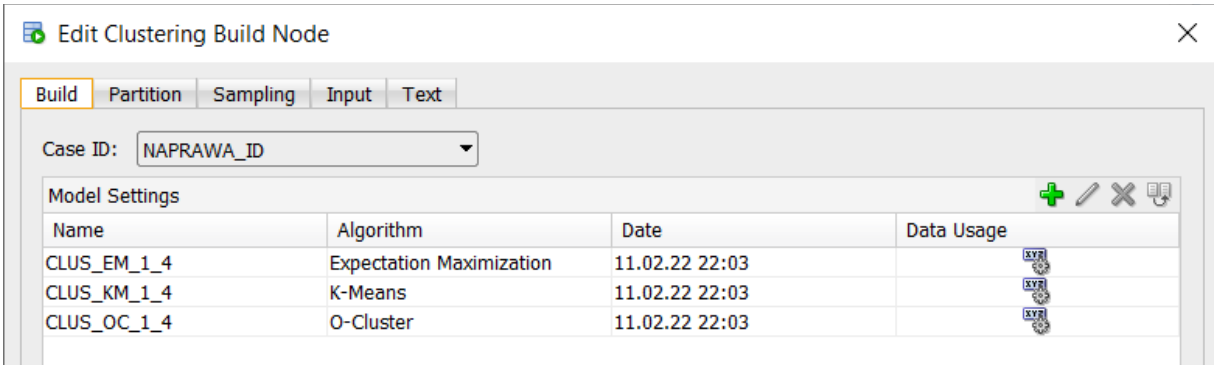
Grupowanie jest nienadzorowaną metodą eksploracji. Oznacza to, że nie wymaga ona ustalania zmiennej celu i cała operacja wykonuje się automatycznie.

Ponownie zbudowany został nowy projekt i nowy workspace.



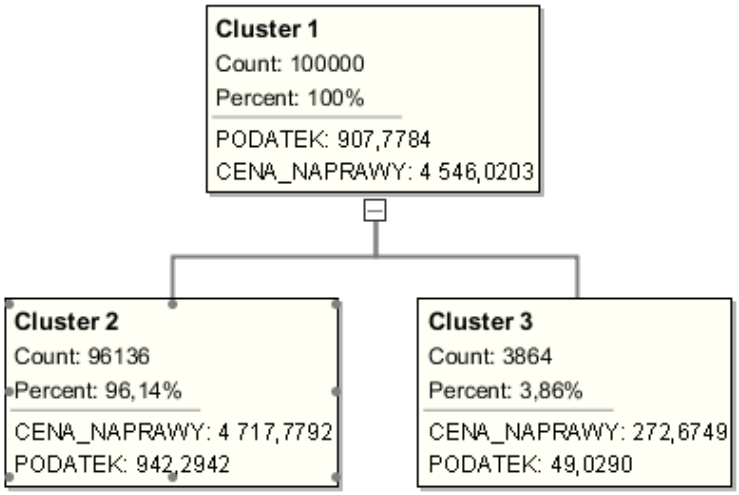
Model został zbudowany w oparciu o 2 obiekty – Data source posiadający rekordy z tabeli faktów – naprawa oraz obiekt grupowania.

Jako Case ID ustawiony został rekord NAPRAWA_ID będący kluczem podstawowym dla tabeli naprawa. Użyte zostały 3 algorytmy grupowania: Algorytm maksymalizacji oczekiwań, algorytm K-średnich oraz algorytm O-cluster. Rekordy zostały pogrupowane automatycznie.

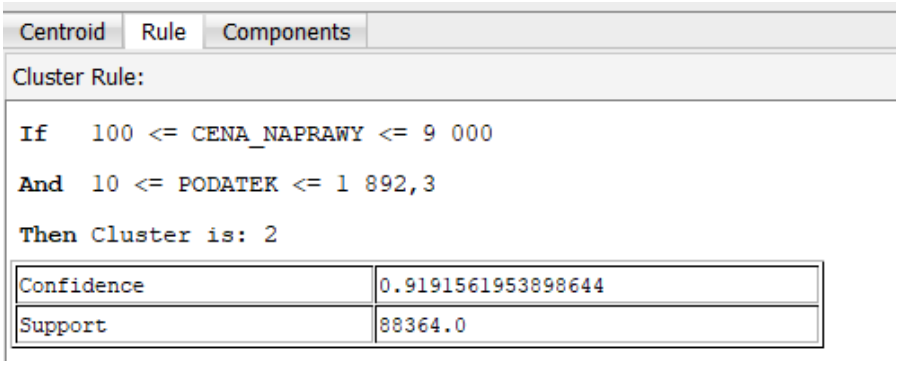


Algorytm maksymalizacji oczekiwań

Model ten buduje drzewo. W korzeniu nr 1 znajdują się wszystkie rekordy, które zostają rozbite na 2 grupy. W innych przypadkach możliwe jest dalsze rozbijanie rekordów. Jednak w przypadku mojej hurtowni rekordy zostały rozbite tylko na 2 grupy.



W zakładce „Rule” można sprawdzić jakie warunki musiał spełnić rekord żeby znaleźć się w konkretnej gałęzi drzewa.

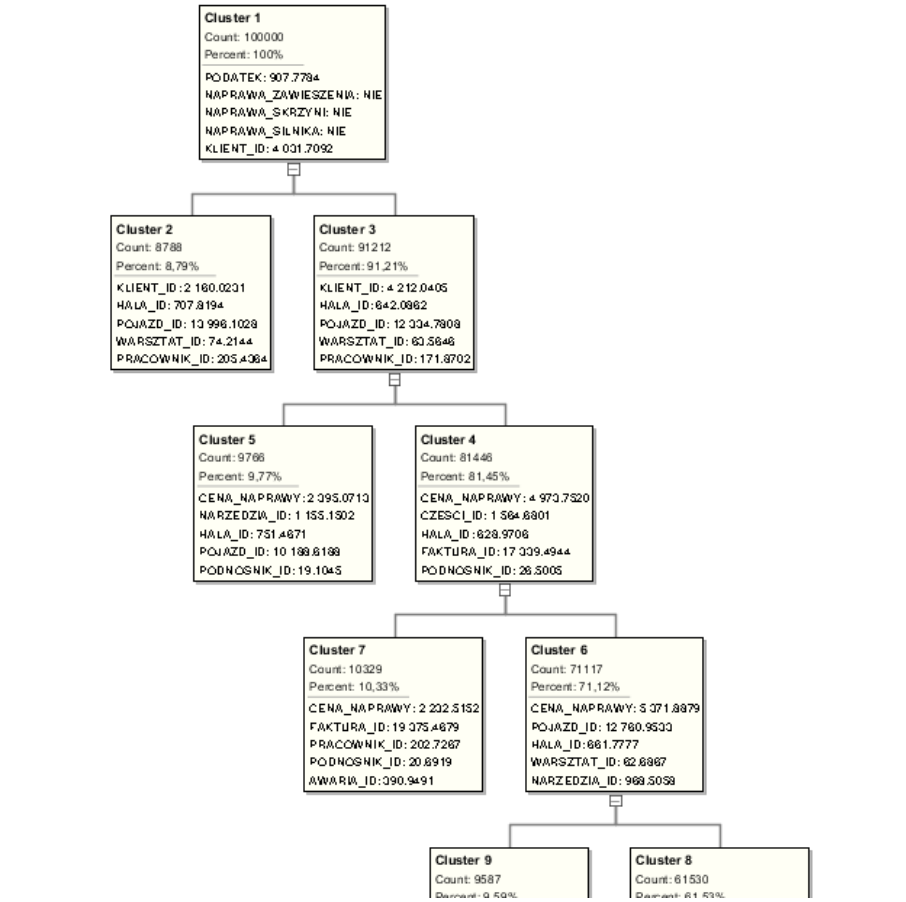


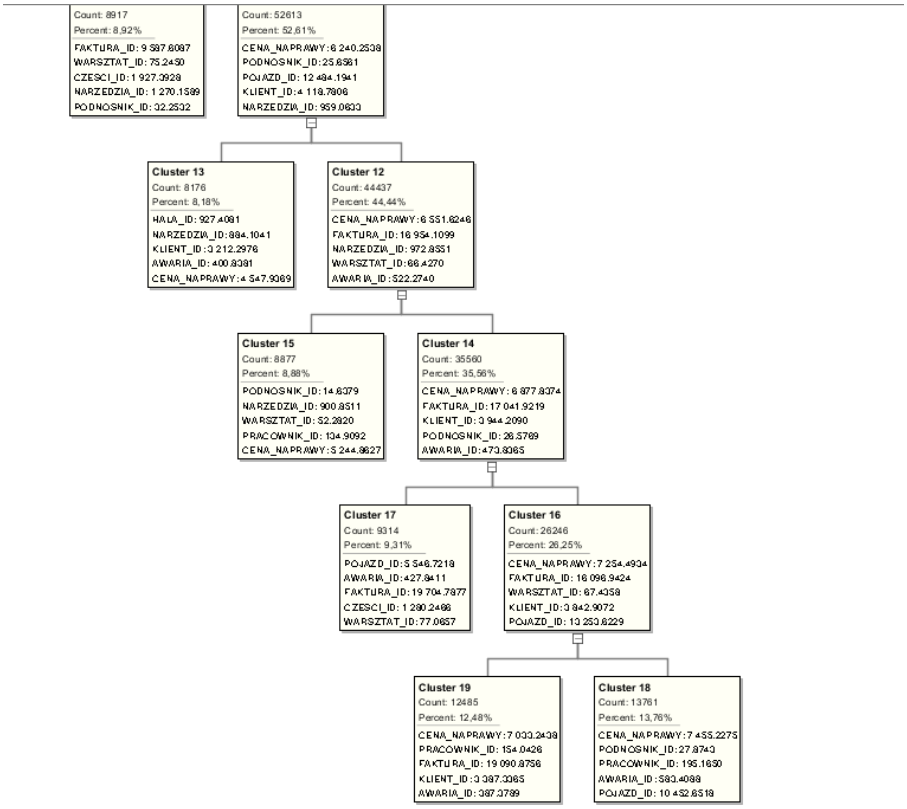
Algorytm k-średnich

Model ten również buduje drzewo, gdzie w korzeniu znajdują się wszystkie rekordy, które następnie zostają rozbite. Algorytm ten działa w oparciu o następujące kroki:

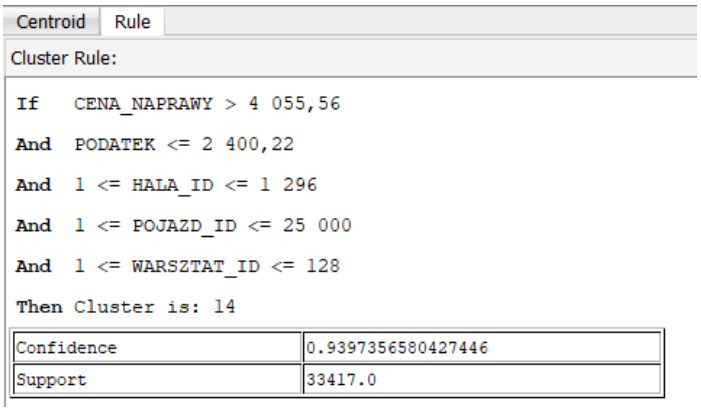
1. **Ustalamy liczbę skupień.**
   Jedną z metod ustalenia ilości skupień jest umowny jej wybór i ewentualna późniejsza zmiana tej liczby w celu uzyskania lepszych wyników. Wybór liczby skupień może być oparty również na wynikach innych analiz.
2. **Ustalamy wstępne środki skupień.**
   Środki skupień tak zwane centroidy możemy dobrać na kilka sposobów: losowy wybór k obserwacji, wybór k pierwszych obserwacji, dobór w taki sposób, aby zmaksymalizować odległości skupień. Jedną z najczęściej stosowanych metod jest kilkakrotne uruchomienie algorytmu i wybór najlepszego modelu, gdy wstępnie środki skupień były wybierane losowo.
3. **Obliczamy odległości obiektów od środków skupień.**
   Wybór metryki jest bardzo istotnym etapem w algorytmie. Wpływa ona na to, które z obserwacji będą uważane za podobne, a które za zbyt różniące się od siebie. Najczęściej stosowaną odległością jest odległość euklidesowa. Stosuje się również kwadrat tej odległości czy też odległość Czebyszewa.
4. **Przypisujemy obiekty do skupień**
   Dla danej obserwacji porównujemy odległości od wszystkich skupień i przypisujemy ją do skupienia, do którego środka ma najbliżej.
5. **Ustalamy nowe środki skupień**
   Najczęściej nowym środkiem skupienia jest punkt, którego współrzędne są średnią arytmetyczną współrzędnych punktów należących do danego skupienia.
6. **Wykonujemy kroki 3,4,5 do czasu, aż warunek zatrzymania zostanie spełniony.**
   Najczęściej stosowanym warunkiem stopu jest ilość iteracji zadana na początku lub brak przesunięć obiektów pomiędzy skupieniami.

W typ przypadku algorytm podzielił rekordy na 19 gałęzi drzewa.

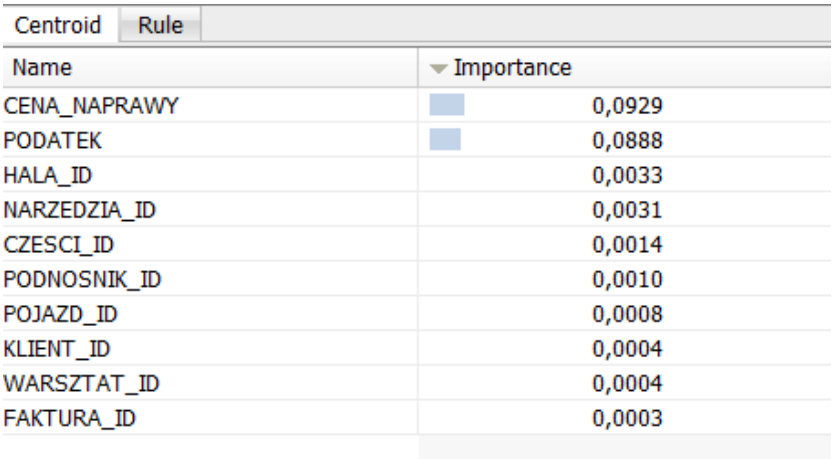




W tym przypadku również można sprawdzić jakie warunki musiał spełnić rekord by znaleźć się w odpowiedniej grupie:

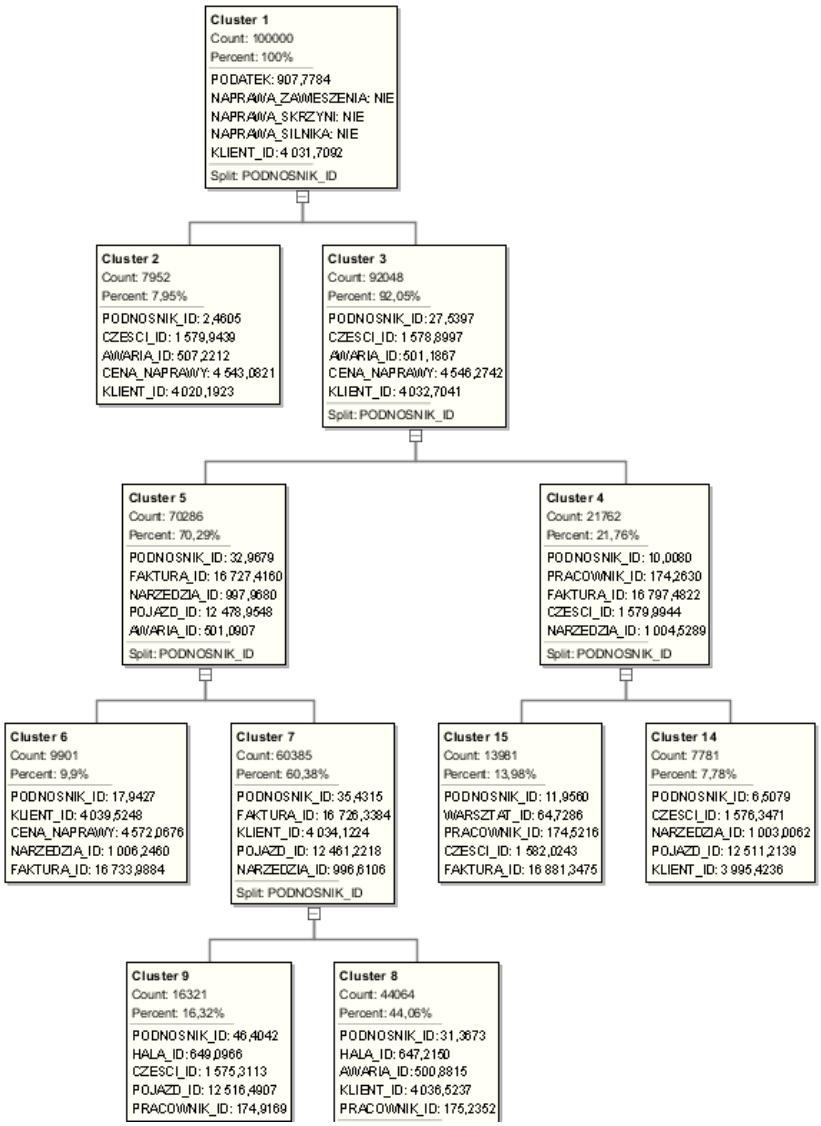


W zakładce Centroid można sprawdzić jakie atrybuty miały największy wpływ na znalezienie się rekordu w tej akurat gałęzi drzewa



Algorytm O-cluster

Algorytm ten, podobnie jak 2 poprzednie również buduje drzewo, gdzie w korzeniu znajdują się wszystkie rekordy i zostają w każdej następnej gałęzi dzielone na coraz mniejsze grupy

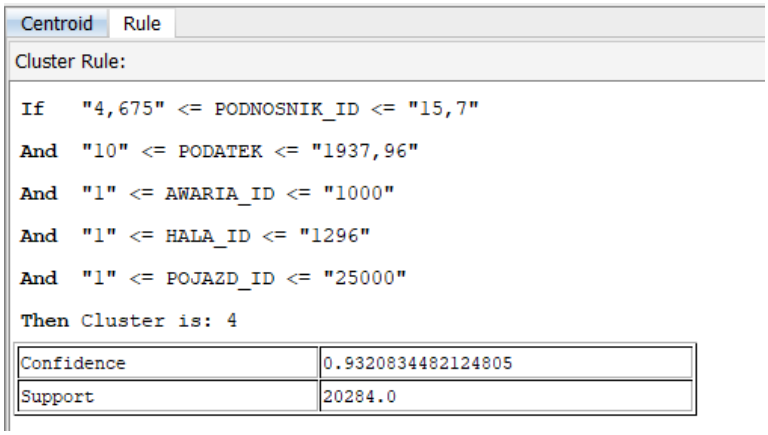


W tym przypadku również można sprawdzić w zakładce Centroid jaki atrybut miał największy wpływ na znalezienie się rekordu w konkretnej gałęzi drzewa

Centroid | Rule

Name	Importance
PODNOSNIK_ID	0,1104
PODATEK	0,0001
AWARIA_ID	0,0001
HALA_ID	0,0001
POJAZD_ID	0,0001
KLIENT_ID	0,0001
NARZEDZIA_ID	0,0000
CZESCI_ID	0,0000

Można też sprawdzić w zakładce Rule jakie dokładnie warunki musi spełnić rekord by znaleźć się w konkretnej gałęzi drzewa:



7. Wnioski

W ramach projektu zbudowana została hurtownia danych, wygenerowane zostały dane i zaprojektowane zostały skrypty pozwalające na szybkie załadowanie danych do hurtowni. Napisanych zostało 18 zapytań, po 3 z kategorii Cube, Rollup, Grouping Sets, Partycje obliczeniowe, Rankingi, Okna ruchome. Zapytania pozwalają na wyciągnięcie ciekawych wniosków z hurtowni takich jak: wyznaczenie najlepiej zarabiających warsztatów, sprawdzenie w jaki dzień miesiąca i w jaki miesiąc zyski są największe czy też wyznaczenie pracowników którzy w konkretnym roku zarobili najwięcej.

Zostały zbudowane również 3 modele eksploracji danych. Model asocjacji pozwolił przewidzieć jaki rodzaj awarii jest powiązany z innym dla konkretnego samochodu. Reguły asocjacyjne wyróżnione w tym modelu miały jednak niskie wsparcie spowodowane najprawdopodobniej losowym generowaniem danych.

Model klasyfikacji miał za zadanie przewidzieć płeć klienta na podstawie zmiennych. Wynik nie był jednak zbyt wiarygodny. Szansa na poprawną klasyfikację w 3 z 4 użytych algorytmów nieznacznie przekroczyła 50% co znaczy, że przewidywać można by równie dobrze rzucając monetą. W tym przypadku również najprawdopodobniejszym powodem takiej niewielkiej szansy na poprawną klasyfikację są losowo generowane dane.

Ostatnią metodą eksploracji danych było grupowanie. Za pomocą 3 algorytmów rekordy z tabeli faktów hurtowni zostały pogrupowane na różne grupy. Można było sprawdzić jaka miara tabeli miała największy wpływ na grupowanie w danej gałęzi drzewa oraz jakie warunki rekord musiał spełnić aby znaleźć się w konkretnej gałęzi drzewa.