Baza danych firmy spedycyjnej - raport

Maciej Dzimira, Michał Fluder, Łukasz Łaszczuk, Krzysztof Siniecki

24.06.2020

Opis bazy danych

Baza danych składa się z dziewięciu encji -- tablic zawierających informacje:

- o pracownikach firmy i ich pensjach,
- o flocie firmy,
- o klientach i wykonywanych dla nich zleceniach,
- o cenach paliwa w danym dniu,
- o wartościach akcji firmy,
- o długach i opłatach.

Schemat bazy

Schemat bazy danych został przedstawiony na wykresie 1.

O firmie

Nasza firma spedycyjna jest spora, choć nie należałaby do czołówki największych firm w Polsce. Flota składa się z 75 pojazdów: 59 samochodów ciężarowych oraz 16 samochodów dostawczych. Liczba pracowników zmienia się w każdym roku. Wzrasta wraz z upływem czasu, choć losowo występują również zwolnienia. Ilość pracowników jest zawsze większa niż ilość pojazdów i waha się pomiędzy 81 a 122. Ilość nowych zleceń jako kwantyzacja rozkładu lognormalnego zazwyczaj mieściła się w przedziale 10-30. Zlecenia na terenie Polski (dystans mniejszy niż 800km) realizowano samochodami dostawczymi, natomiast zagraniczne samochodami ciężarowymi. Samochód jak i pracownik realizujący wybierani byli losowo z puli wolnych w tym czasie samochodów/pracowników.

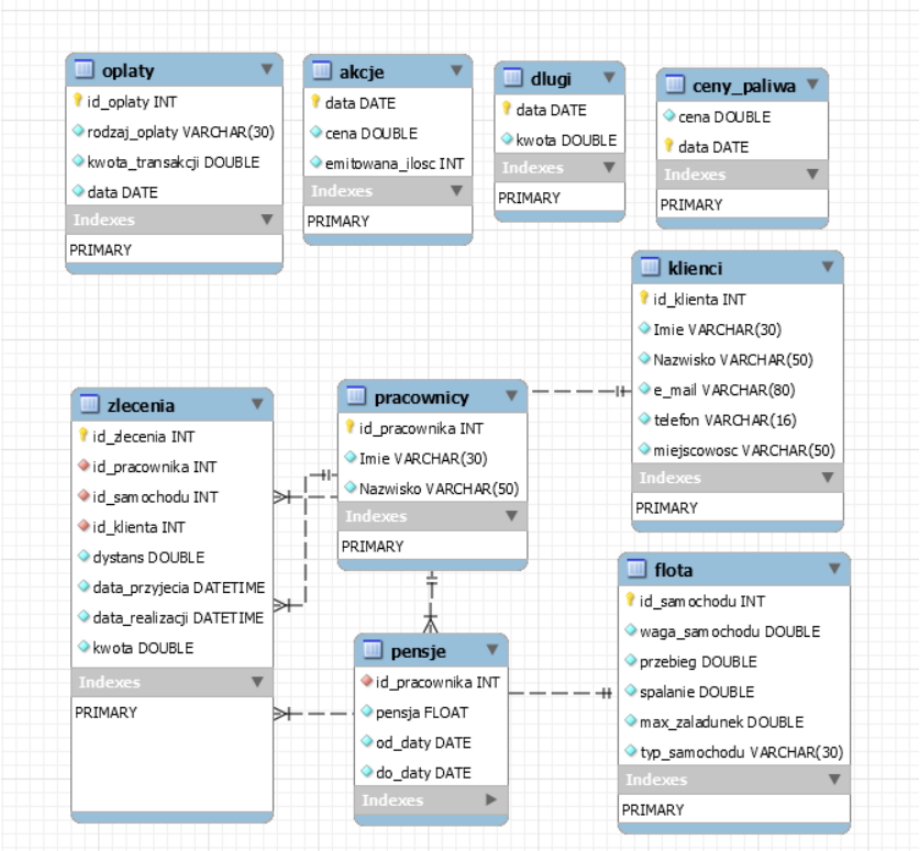


Figure 1: Schemat bazy

Rozpatrywany okres czasu

Dane zostały wygenerowane dla okresu od 1 stycznia 2010 do 1 kwietnia 2020. Daty te występowały we wszystkich tabelach, można więc przyjąć że początkiem działalności rynkowej jak i dniem pierwszej emisji akcji spółki (IPO) był dzień 01.01.2010. Ceny paliwa i akcji zmieniały się codziennie.

Generowanie danych

Skrypt do wypełnienia bazy danych został napisany w języku Python. Do wygenerowania losowych zmiennych kategorycznych (takich jak imiona, nazwiska i adresy) wykorzystana została biblioteka Faker (wersja polskojęzyczna), zaś do wygenerowania danych liczbowych wykorzystana została biblioteka random. Baza po każdym generowaniu wygląda tak samo, ponieważ przy wypełnianiu danych korzystamy z tego samego zestawu liczb losowych.

Instrukcja do wygenerowania bazy

Aby móc wygenerować bazę danych oraz być w stanie odtworzyć rezultaty naszej pracy, należy pobrać repozytorium z kodami dostępnymi na stronie github. Przykładowo można to zrobić używając polecenia:

```
git clone https://github.com/lukaszlaszczuk/projekt-bazy-danych.git
```

W celu wygenerowania bazy, najpierw należy upewnić się, czy zainstalowane są wszystkie potrzebne biblioteki języka Python. Lista niezbędnych bibliotek (wraz z ich wersjami) znajduje się w pliku requirements.txt. Możemy je zainstalować w łatwy sposób przy użyciu systemu pip, za pomocą komendy:

```
pip install -r requirements.txt
```

Jeżeli wszystkie potrzebne biblioteki są już zainstalowane, to możemy wygenerować raport wynikowy. W raporcie znajduje się kod tworzący schemat bazy oraz wykonujący skryptowe wypełnianie tabel bazy danych. Program potrzebuje kilku informacji, które musimy podać, edytując plik `projekt-bazy-danych/project/database-info/database_credentials.xlsx`. Informacje te to:

- nazwa hostu;
- nazwa użytkownika;
- hasło do bazy danych (w przypadku braku zostawiamy puste).

Po uzupełnieniu odpowiednich informacji możemy wygenerować raport wynikowy używając następującego polecenia w folderze `projekt-bazy-danych/project/raport`:

```
stitch raport.md -o raport_wynikowy.pdf
```

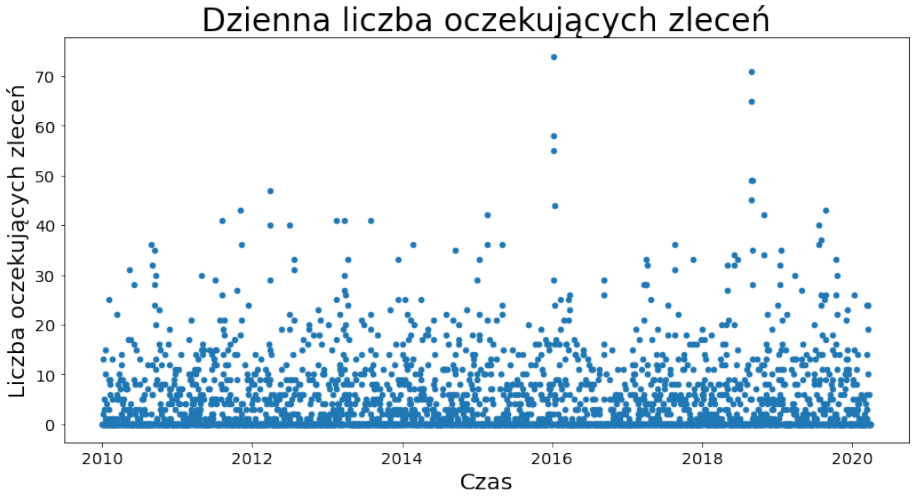
Polecenie to wygeneruje końcowy raport z analizą.

Analiza danych

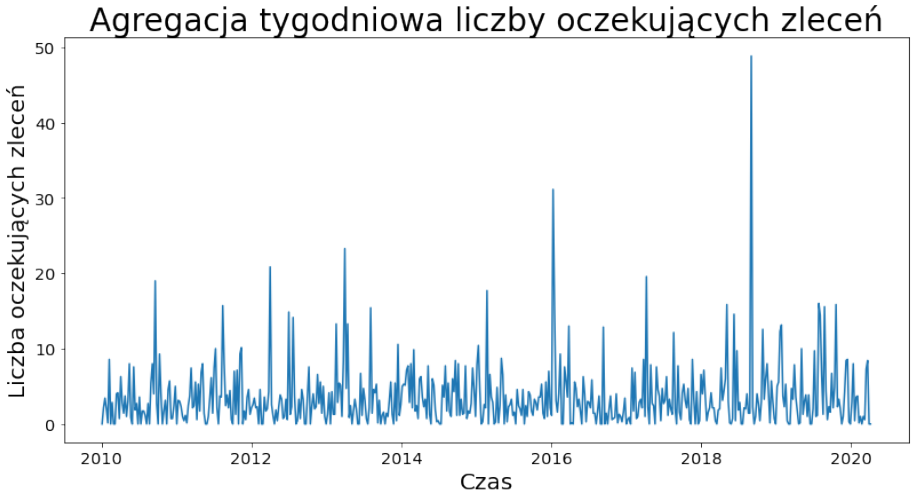
1. Liczba oczekujących zleceń w czasie

W naszej firmie liczba kierowców była zawsze większa od liczby dostępnych samochodów. Zdarzało się, że wszystkie samochody były aktualnie w trasie. W takich przypadkach zlecenie musiało poczekać na podjęcie realizacji. Zwizualizujemy jak ta liczba zmieniała się w czasie. Przygotowaliśmy dwa wykresy:

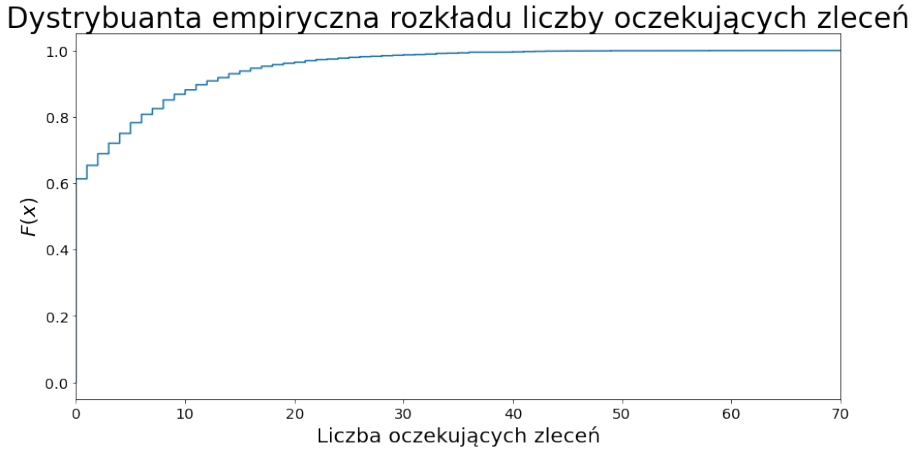
- Dzienna liczba oczekujących zleceń;
- Pierwszy wykres był dość niewyraźny, więc zagregowaliśmy tygodniowo liczbę oczekujących zleceń i policzyliśmy średnią w każdym z tygodni



Dane zaprezentowane w ten sposób nie są czytelne, dlatego zagregowaliśmy tygodniowo liczbę oczekujących zleceń i wyciągnęliśmy dla każdego z nich średnią.



Aby w lepszy sposób zwizualizować rozkład zaobserwowanej realizacji wektora losowego liczby oczekujących zleceń, pokażemy wykres dystrybuanty empirycznej tej zmiennej.



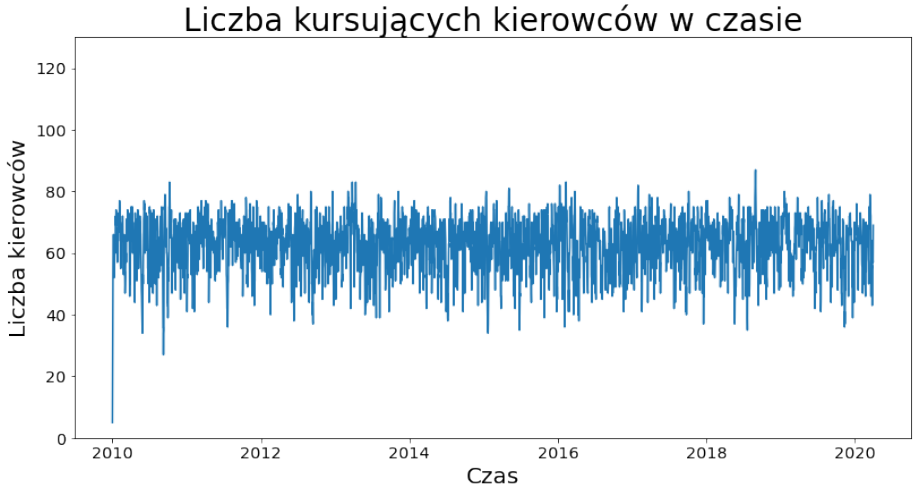
Jak widać z wykresu dystrybuanty empirycznej przez ponad 60% dni nie mieliśmy zleceń ozekujących do realizacji, a w około 80% przypadków zmienna losowa przyjmowała wartości mniejsze niż 10. Rozkład tej zmiennej jest prawostronnie skośny

2. Wykres liczby zmieniających się pracowników

W naszej bazie danych założyliśmy, że lista pracowników zmienia się rocznie (na początku każdego roku zwalniamy lub zatrudniamy nowych pracowników). Przygotowaliśmy więc:

- Wykres zatrudnionych pracowników na dany rok;
- Wykres liczby pracowników, którzy są w trasie w danym dniu.

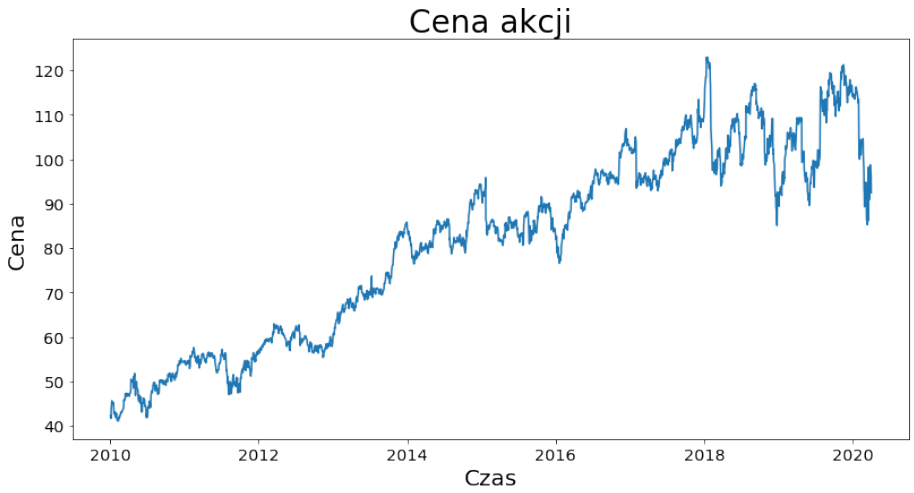
Z wykresu możemy zauważyć rosnący trend zatrudnionych pracowników. Nasza firma się rozwija!

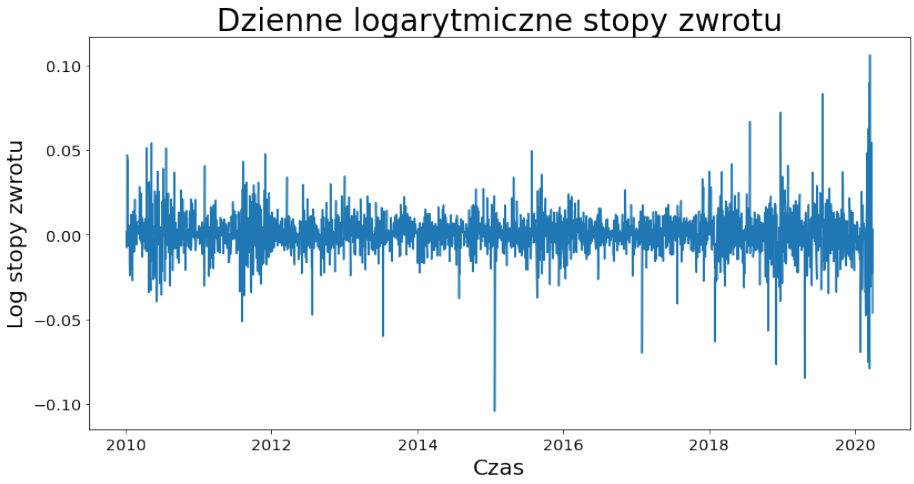


Na wykresie możemy zaobserwować pewną okresowość liczby kursujących. Przez zdecydowaną większość czasu więcej niż 40 kierowców było w trasie.

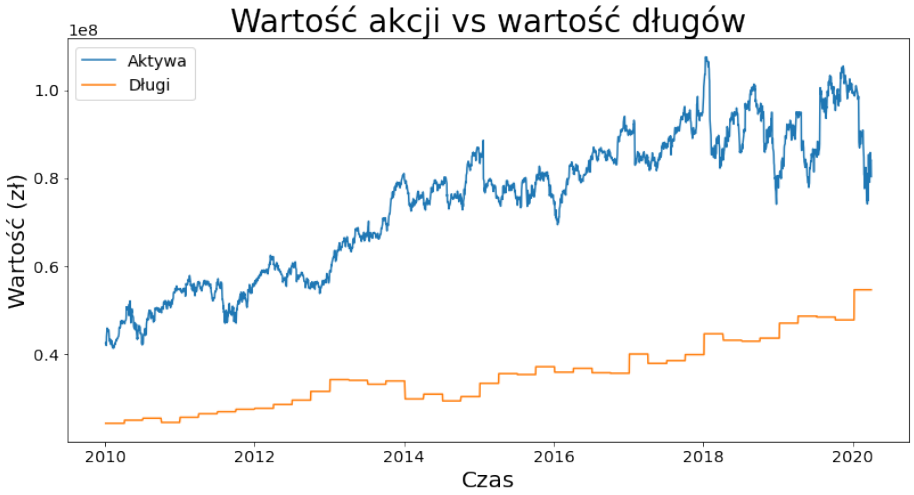
3. Bilans finansowy firmy - czy jest w stanie się utrzymać?

W tej części postaramy się sprawdzić, jakie jest prawdopodobieństwo zbankrutowania firmy spedycyjnej. Na podstawie akcji oraz długów firmy sprawdzimy przy pomocy modelu KMV prawdopodobieństwo bankructwa (Default Probability) dla rocznego horyzontu czasowego, przy stopie wolnej od ryzyka równej $0,5\%$. Aby móc skorzystać z modelu KMV będziemy potrzebować: całkowitej wartości akcji, całkowitej wartości długu oraz dziennych logarytmicznych stóp zwrotu z akcji. Zaprezentujemy te zmienne na wykresach.

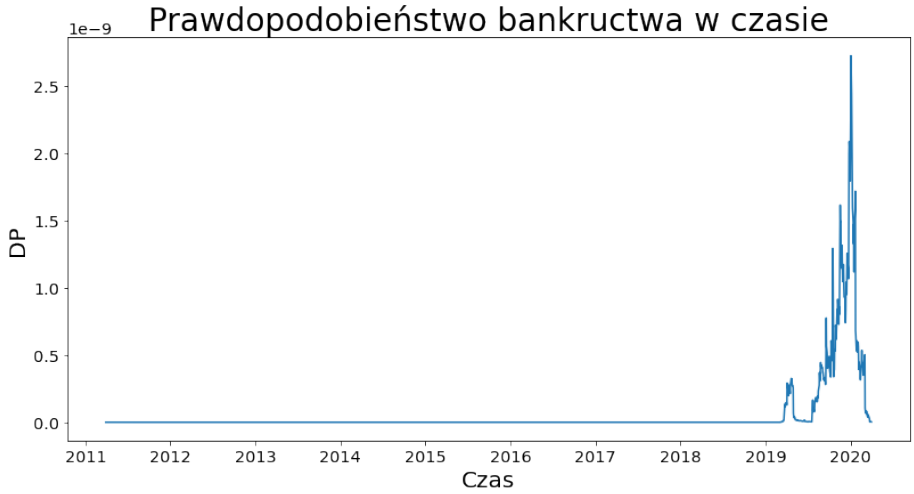




Obserwacja: Akcje były dość stabilne, w 2020 widzimy duży wzrost zmienności logarytmicznych stóp zwrotu. Spowodowane jest to kryzysem wywołanym przez koronawirus.



Wartość akcji jest przez cały okres większa od wartości długów.



Obserwacja: Prawdopodobieństwo bankructwa w modelu KMV jest tożsama z zerem dla zdecydowanej większości analizowanego okresu. Widzimy wzrost tego prawdopodobieństwa na przełomie lat 2019-2020 (lecz dalej jest to prawdopodobieństwo rzędu 10^{-9}.

4. Lista osób najdłużej czekających na zlecenie

- Pokazujemy imię i nazwisko oraz sumaryczny czas (dla wszystkich złożonych zleceń) oczekiwania na realizację zlecenia dla 10 klientów, którzy sumarycznie czekali na zlecenia najdłużej.

```
In [13]: najdluzej_czekajacy.iloc[:, ::-1]
```

	Imie i nazwisko	Czas_oczekiwania
0	Dagmara Szóstak	30 days
1	Mateusz Toma	24 days
2	Anna Maria Koss	24 days
3	Nataniel Żbik	23 days
4	Daniel Nalewajko	23 days
5	Anastazja Skiepko	23 days
6	Sylwia Towarek	22 days
7	Ada Mokwa	22 days
8	Elżbieta Kwapis	22 days
9	Maks Windak	22 days

Wniosek: Przepraszamy Panią Dagmarę.

5. Dodatkowe analizy:

- Którzy pracownicy realizowali zlecenia, które przynosiły najwięcej zysku?;
- Czy pensje tych pracwników były najwyższe?;
- Które samochody miały największy przebieg;
- Roczne zestawienie kosztów paliwa;
- Klienci, na których zarobiliśmy najwięcej.

1. Lista pracowników którzy średnio generowali największy zysk

```
In [15]: df
```

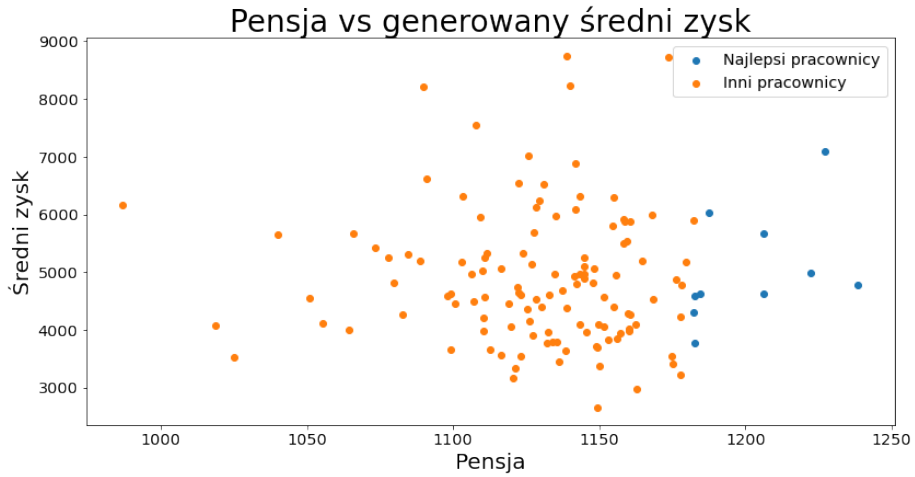
	id	sredni zysk	Imie	Nazwisko
0	103	1238.533333	Marika	Fiutak
1	122	1227.089464	Jakub	Misior
2	102	1222.445111	Marcelina	Pituła
3	3	1206.147358	Aurelia	Giża
4	112	1206.077426	Patryk	Bazylewicz
5	123	1187.574821	Dorota	Gomoła
6	90	1184.504275	Kalina	Kasznia
7	23	1182.793029	Tomasz	Hermann
8	71	1182.507576	Mateusz	Miros
9	106	1182.443658	Tomasz	Lesisz

2. Czy Ci pracownicy zarabiali najwięcej?

```
In [17]: podsumowanie
```

	miejsce_w_pensjach	sredni zysk	Pensja	Imie_x	Nazwisko_x
0	61	1238.533333	4781.32	Marika	Fiutak
1	5	1227.089464	7084.55	Jakub	Misior
2	49	1222.445111	4988.24	Marcelina	Pituła
3	67	1206.147358	4619.35	Aurelia	Giża
4	29	1206.077426	5671.14	Patryk	Bazylewicz
5	18	1187.574821	6027.48	Dorota	Gomoła
6	66	1184.504275	4632.31	Kalina	Kasznia
7	71	1182.793029	4590.83	Tomasz	Hermann
8	111	1182.507576	3780.39	Mateusz	Miros
9	85	1182.443658	4298.04	Tomasz	Lesisz

Sprawdzimy jeszcze jak najlepsi pracownicy wyglądają na tle innych pracowników.



Wniosek: "Najlepsi pracownicy" wcale nie mają największych pensji. Nie widzimy zauważalnie pozytywnej korelacji pomiędzy zyskiem a pensją.

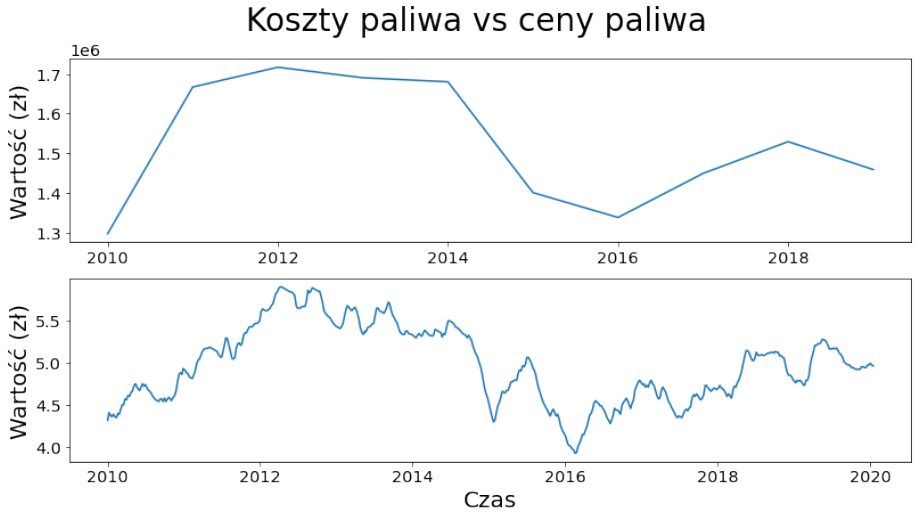
3. Które samochody miały największy przebieg

Założyliśmy, że w tabeli flota mamy początkowy przebieg auta w czasie zakupu (możemy kupować używane samochody). Całkowity przebieg jest sumą początkowego przebiegu i sumy dystansów ze tabeli zlecenia.

```
In [21]: najbardziej_eksploatowane
```

	id	całkowity przebieg
0	35	1460248.0
1	34	1422726.0
2	53	1421068.0
3	16	1416927.0
4	30	1407044.0
5	12	1392597.0
6	48	1384850.0
7	52	1376478.0
8	19	1361955.0
9	49	1356401.0

4.Roczne zestawienie kosztów za paliwo

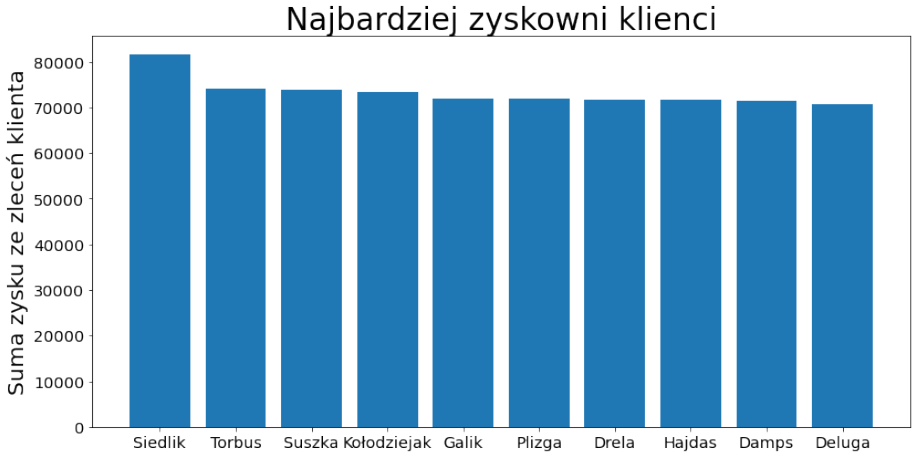


Wniosek: Widzimy zależność pomiędzy cenami paliwa a kosztami.

5. Lista klientów, na których zarobiliśmy najwięcej

```
In [25]: najlepsi_klienci.iloc[:, :3]
```

	Id	Imię	Nazwisko
0	443	Dariusz	Siedlik
1	274	Borys	Torbus
2	717	Róża	Suszka
3	816	Cyprian	Kołodziejak
4	1154	Błażej	Galik
5	223	Stefan	Plizga
6	197	Klara	Drela
7	1013	Jan	Hajdas
8	698	Jacek	Damps
9	375	Ryszard	Deluga



Podsumowanie

- W projekcie zasymulowaliśmy bazę danych firmy spedycyjnej.
- Wszystkie skrypty integrujące serwer bazy danych były napisane w języku Python. Jego uniwersalność pozwoliła nam na połączenie z językiem SQL w celu wygenerowania schematu bazy, stworzenie skryptu wypełniającego bazę odpowiednimi danymi, przeprowadzenie analizy danych oraz automatyczne wygenerowanie raportu, który wykonuje wszystkie poprzednie zadania przy użyciu jednego polecenia z linii komend;
- Według nas najtrudniejszym zadaniem było skryptowe wypełnienie bazy i na tę część poświęciliśmy najwięcej czasu. Niebanalne było również stworzenie struktury projektu w taki sposób, aby możliwe było wykonanie wszystkich poleceń przy użyciu jednego polecenia z linii komend.