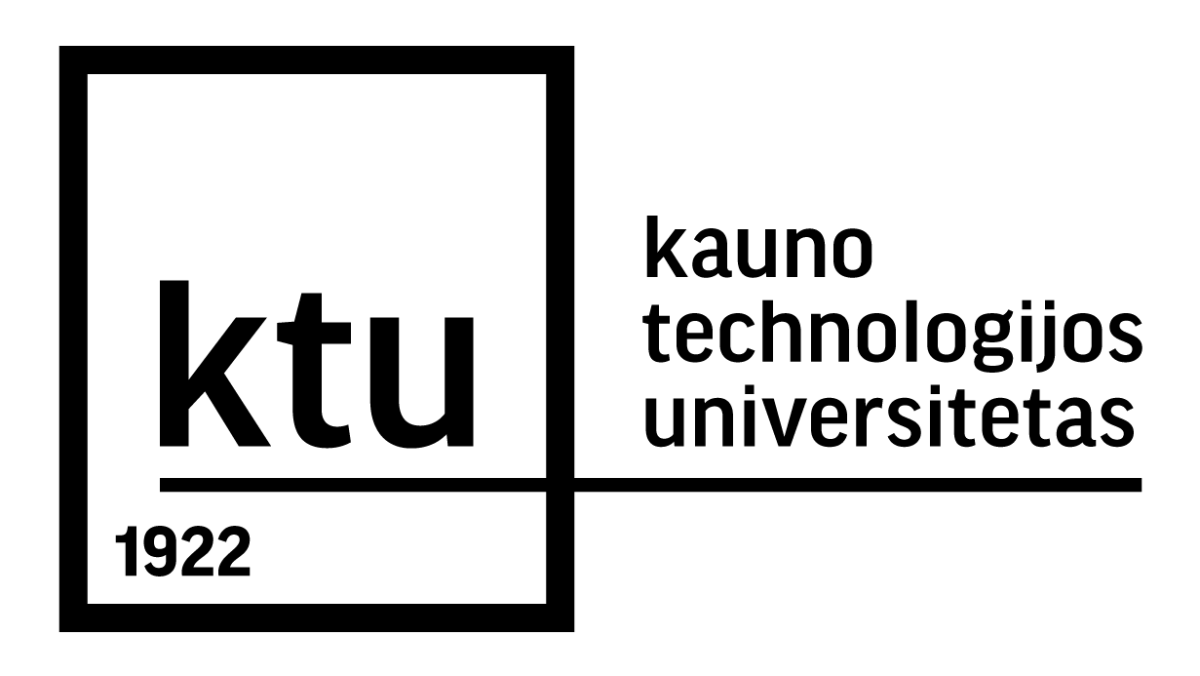
KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS



Intelektikos pagrindai (P176B101)

Laboratorinis darbas Nr.1

Atliko:

IFF-9/8 gr. studentas

Lukas Navašinskas

2022 m. kovo 7 d.

Priėmė:

lekt. Nečiūnas Audrius

doc. Paulauskaitė-Tarasevičienė Agnė

KAUNAS 2022

Contents

[1. Duomenų rinkinys 3](#_Toc65774127)

# Duomenų rinkinys

Laboratoriniui darbui pasirinktas automobilių specifikacijų rinkinys. Rinkinį sudaro šie atributai:

Tolydiniai atributai: Year, Engine Cylinders, Engine Displacement, City MPG, Highway MPG, Annual Fuel Cost, Tailpipe CO2 in Grams/Mile

Kategoriniai atributai: Drive, Transmission, Turbocharger, Supercharger, Fuel Type

# Duomenų rinkinio kokybės analizė

Tolydinių atributų analizė:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Atributo pavadinimas | Kiekis (Eilučių sk.) | Trūkstamos reikšmės, % | Kardinalumas | Minimali reikšmė | Maksimali reikšmė | 1-asis kvartilis | 3-iasis kvartilis | Vidurkis | Mediana | Standartinis nuokrypis |
| Year | 38113 | 0 | 34 | 1984 | 2017 | 1991 | 2009 | 2000.195 | 2001 | 10.465 |
| Engine Cylinders | 37977 | 0.357 | 9 | 2 | 16 | 4 | 6 | 5.737 | 6 | 1.752 |
| Engine Displacement | 37979 | 0.352 | 66 | 0 | 8.4 | 2.2 | 4.3 | 3.318 | 3 | 1.362 |
| City MPG | 38113 | 0 | 93 | 6 | 150 | 15 | 20 | 17.981 | 17 | 6.850 |
| Highway MPG | 38113 | 0 | 83 | 9 | 122 | 20 | 27 | 24.081 | 24 | 7.027 |
| Annual Fuel Cost | 38113 | 0 | 60 | 500 | 6050 | 1600 | 2350 | 1970.675 | 1950 | 532.555 |
| Tailpipe CO2 in Grams/Mile | 38113 | 0 | 592 | 0 | 1269.571 | 388 | 555.438 | 472.761 | 467.737 | 122.200 |

*pav. 1 Tolydinio tipo atributų kokybės analizės lentelė*

Kategorinių atributų analizė:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Atributo pavadinimas | Kiekis (Eilučių sk.) | Trūkstamos reikšmės, % | Kardinalumas | Moda | Modos dažnumas | Moda, % | 2-oji Moda | 2-osios Modos dažnumas | 2-oji Moda, % |
| Drive | 36924 | 3.120 | 7 | Front-Wheel | 13351 | 35.030 | Rear-Wheel Drive | 13018 | 34.156 |
| Transmission | 38102 | 0.0289 | 46 | Automatic 4-Speed | 11042 | 28.972 | Manual 5-Speed | 8323 | 21.838 |
| Turbocharger | 5239 | 86.254 | 2 | None | 32874 | 13.746 | Yes | 5239 | 13.756 |
| Supercharger | 693 | 98.182 | 2 | None | 37420 | 1.818 | Yes | 693 | 1.818 |
| Fuel Type | 38113 | 0 | 14 | Regular | 25258 | 66.271 | Premium | 10133 | 26.587 |

pav. 2 Kategorinio tipo atributų kokybės analizės lentelė

|  |
| --- |
|  |
|  |

Atributų histogramos

# Duomenų kokybės problemos ir sprendimai

Duomenų rinkinio atributai turėjo trūkstamų reikšmių bei išskirčių. Įrašai kurie turėjo tuščių reikšmių, bei išskirčių buvo ištrinti. Išskirčių radimui buvo pasinaudota „python“ biblioteka „pandas“, randami kvantiliai ir pagal juos atrenkami duomenys.

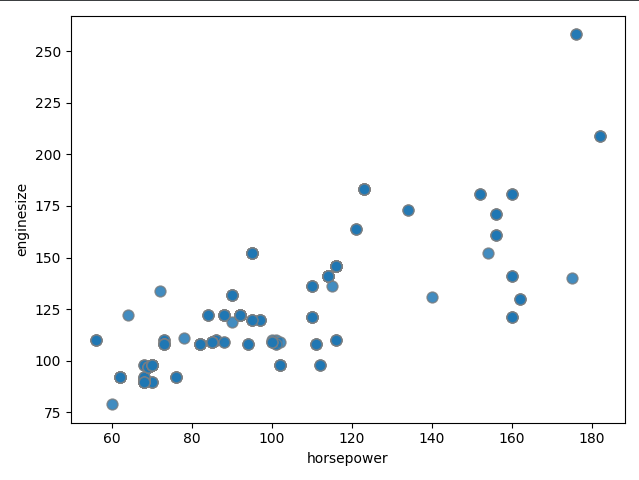
Kodo fragmentas išskirčių radimui ir šalinimui:

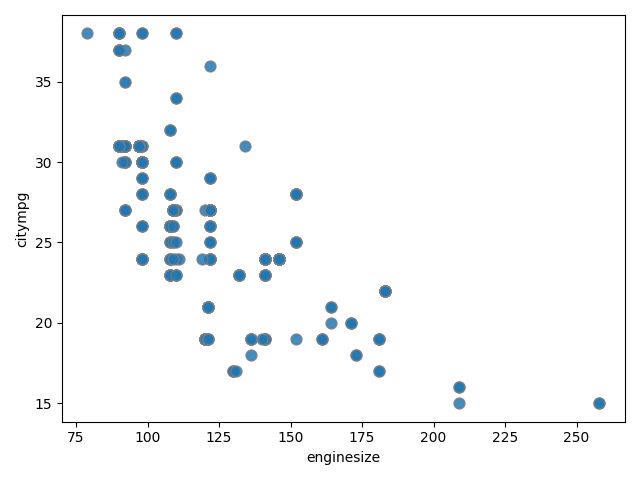
def deleteOutliers(data):  
 q\_low = data["enginesize"].quantile(0.0005)  
 q\_hi = data["enginesize"].quantile(0.9995)  
 data = data[(data["enginesize"] < q\_hi) & (data["enginesize"] > q\_low)]  
  
 q\_low = data["symboling"].quantile(0.0005)  
 q\_hi = data["symboling"].quantile(0.9995)  
 data = data[(data["symboling"] < q\_hi) & (data["symboling"] > q\_low)]  
  
 q\_low = data["compressionratio"].quantile(0.005)  
 q\_hi = data["compressionratio"].quantile(0.9995)  
 data = data[(data["compressionratio"] < q\_hi) & (data["compressionratio"] > q\_low)]  
  
 q\_low = data["horsepower"].quantile(0.0005)  
 q\_hi = data["horsepower"].quantile(0.9995)  
 data = data[(data["horsepower"] < q\_hi) & (data["horsepower"] > q\_low)]  
  
 q\_low = data["citympg"].quantile(0.0005)  
 q\_hi = data["citympg"].quantile(0.9995)  
 data = data[(data["citympg"] < q\_hi) & (data["citympg"] > q\_low)]  
  
 q\_low = data["highwaympg"].quantile(0.0005)  
 q\_hi = data["highwaympg"].quantile(0.9995)  
 data = data[(data["highwaympg"] < q\_hi) & (data["highwaympg"] > q\_low)]  
 return data

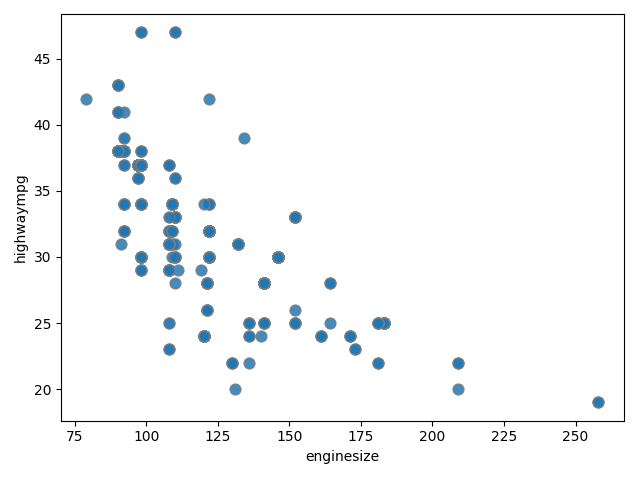
Kodo fragmentas ištrinti eilutes su trūkstamomis reikšmėmis:

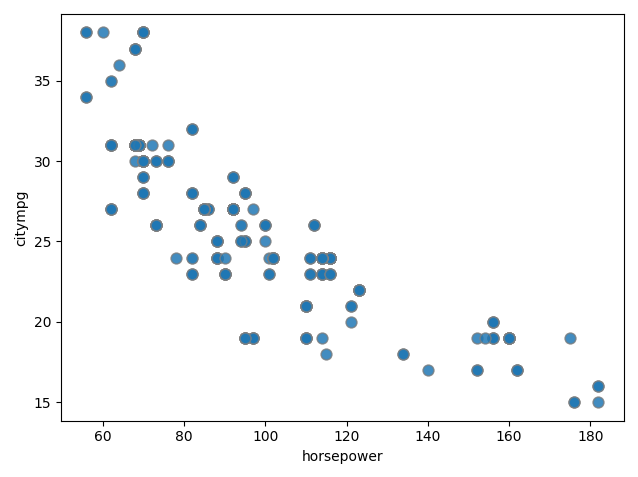
def deleteNaN(data):  
 data = data.dropna()

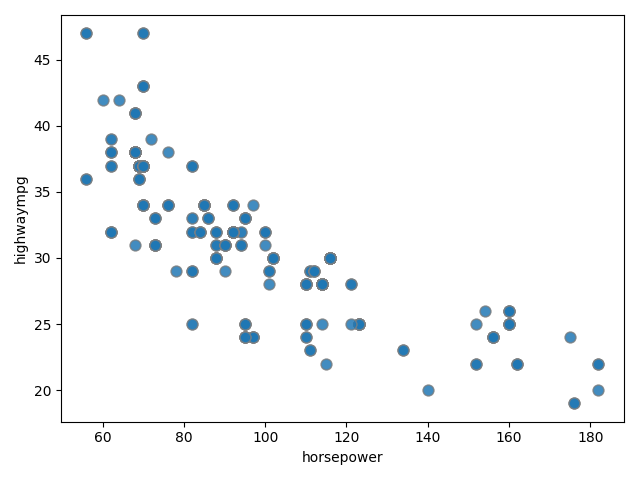
# Tolydinio tipo atributų vizualizacija



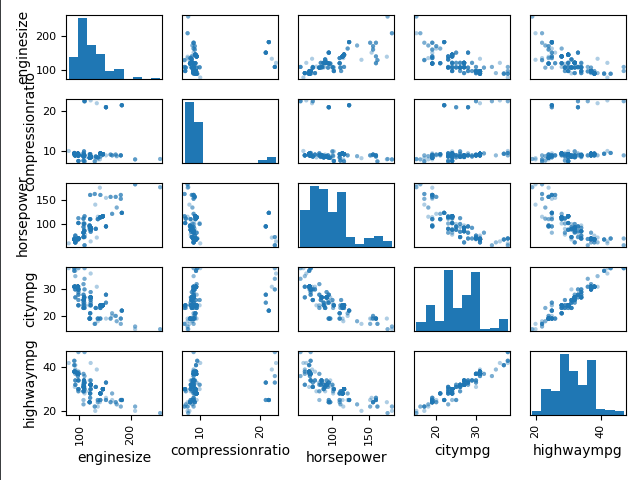




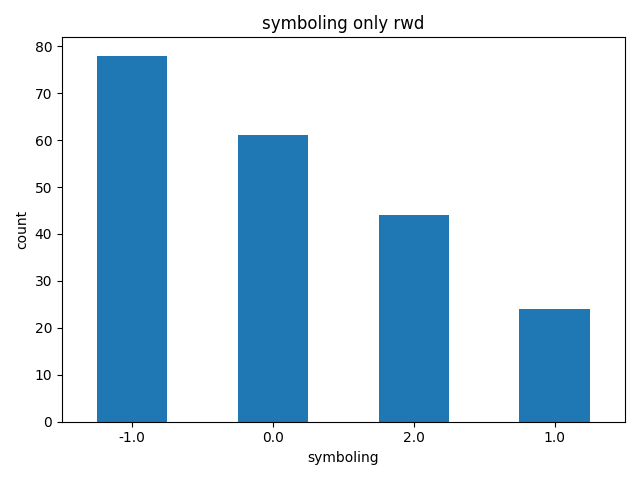


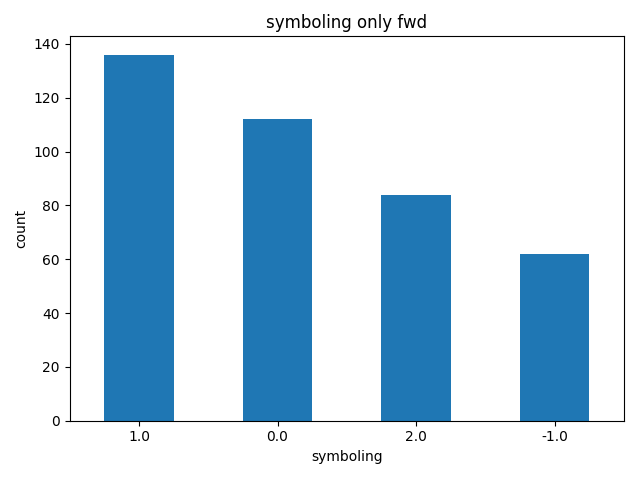


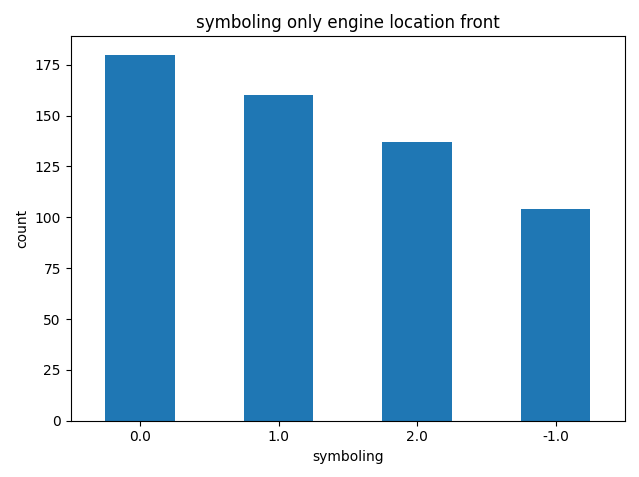
# Scatter plot Matrix diagrama

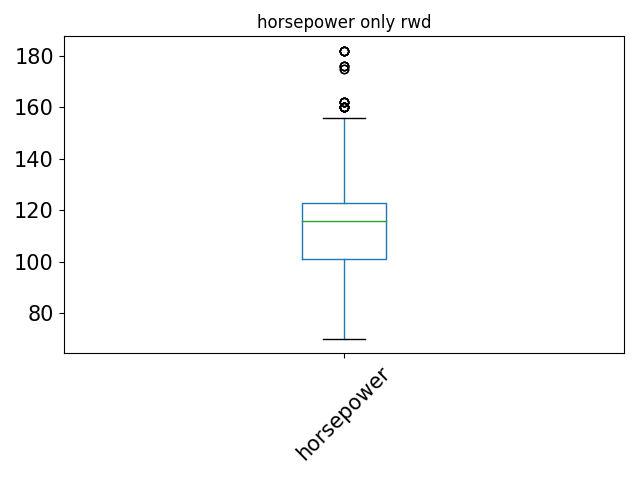


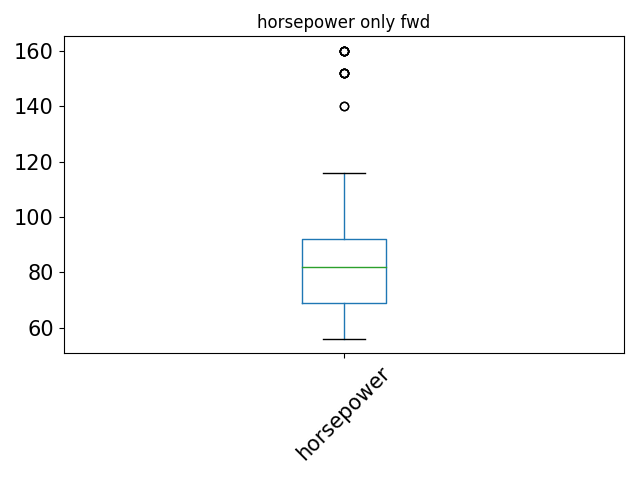
# Kategorinio tipo atributų vizualizacija

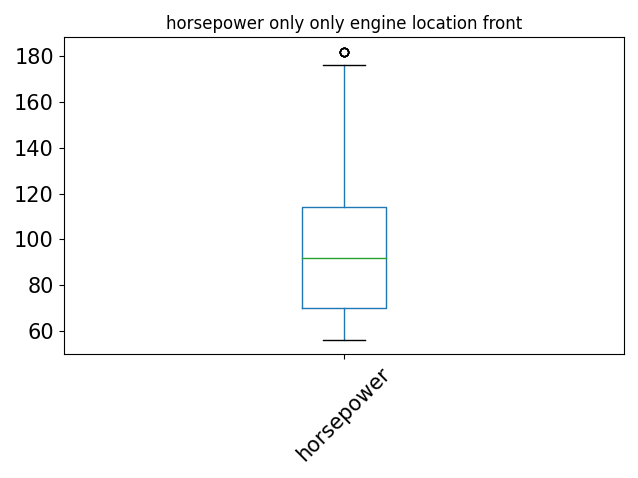


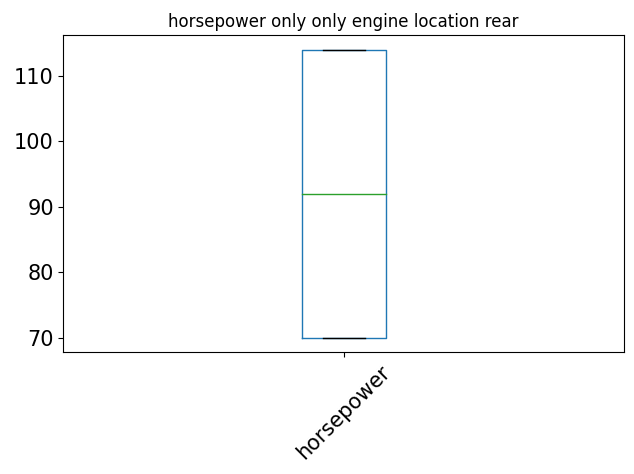




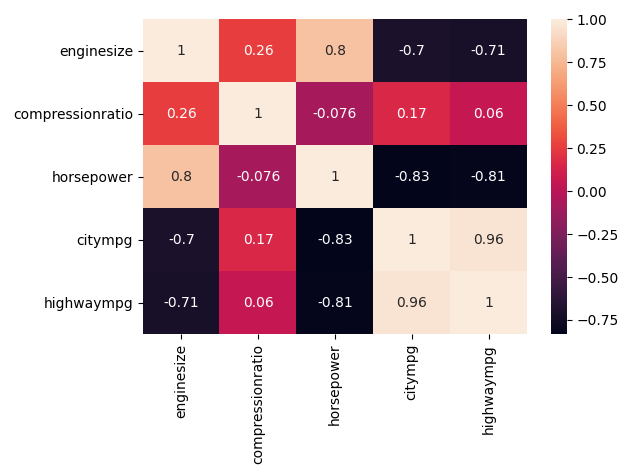








# Koreliacijos matrices diagrama



# Duomenų normalizacija

Programos kodas atlikti duomenų normalizacijai:

def normalize(data):  
 result = data.copy()  
 for feature\_name in data.columns:  
 max\_value = data[feature\_name].max()  
 min\_value = data[feature\_name].min()  
 result[feature\_name] = (data[feature\_name] - min\_value) / (max\_value - min\_value)  
 return result

Programos rezultatai:

