GM_Car_cost_predict

Bio:

Ismim Orifjonov Abdulaziz

Astrum IT Akademiyasining Data Science yo’nalishi bo’yicha o’qiyman va Yordamchi mentor bo’ib ishlayman!

Bugungi mavzu: Ma’lum bir datalardan foydalanib yangi ishlab chiqarilgan mashinani narxini belgilab beradigan dastur

Avalam bor bizga taqdim etilgan datani ko’rib olsak

ID	symboling	name	fueltypes	aspiration	doornumbers	carbody	drivewheels	enginelocation	wheelbase	carlength	carwidth	carheight	curbweight	enginetype	cylindernumber	enginesize	fuelsystem	boreratio	stroke	compressionratio	horsepower	peakrpm	citympg	highwaympg	price
1	3	alfa-romero giulia	gas	std	two	convertible	rwd	front	88.6	168.8	64.1	48.8	2548	dohc	four	130	mpfi	3.47	2.68	9	111	5000	21	27	13495
2	3	alfa-romero stelvio	gas	std	two	convertible	rwd	front	88.6	168.8	64.1	48.8	2548	dohc	four	130	mpfi	3.47	2.68	9	111	5000	21	27	16500
3	1	alfa-romero Quadrifoglio	gas	std	two	hatchback	rwd	front	94.5	171.2	65.5	52.4	2823	ohcv	six	152	mpfi	2.68	3.47	9	154	5000	19	26	16500
4	2	audi 100 ls	gas	std	four	sedan	fwd	front	99.8	176.6	66.2	54.3	2337	ohc	four	109	mpfi	3.19	3.4	10	102	5500	24	30	13950
5	2	audi 100ls	gas	std	four	sedan	4wd	front	99.4	176.6	66.4	54.3	2824	ohc	five	136	mpfi	3.19	3.4	8	115	5500	18	22	17450
6	2	audi fox	gas	std	two	sedan	fwd	front	99.8	177.3	66.3	53.1	2507	ohc	five	136	mpfi	3.19	3.4	8.5	110	5500	19	25	15250
7	1	audi 100ls	gas	std	four	sedan	fwd	front	105.8	192.7	71.4	55.7	2844	ohc	five	136	mpfi	3.19	3.4	8.5	110	5500	19	25	17710
8	1	audi 5000	gas	std	four	wagon	fwd	front	105.8	192.7	71.4	55.7	2954	ohc	five	136	mpfi	3.19	3.4	8.5	110	5500	19	25	18920
9	1	audi 4000	gas	turbo	four	sedan	fwd	front	105.8	192.7	71.4	55.9	3086	ohc	five	131	mpfi	3.13	3.4	8.3	140	5500	17	20	23875
10	0	audi 5000s (diesel)	gas	turbo	two	hatchback	4wd	front	99.5	178.2	67.9	52	3053	ohc	five	131	mpfi	3.13	3.4	7	160	5500	16	22	17859.167
11	2	bmw 320i	gas	std	two	sedan	rwd	front	101.2	176.8	64.8	54.3	2395	ohc	four	108	mpfi	3.5	2.8	8.8	101	5800	23	29	16430
12	0	bmw 320i	gas	std	four	sedan	rwd	front	101.2	176.8	64.8	54.3	2395	ohc	four	108	mpfi	3.5	2.8	8.8	101	5800	23	29	16925
13	0	bmw x1	gas	std	two	sedan	rwd	front	101.2	176.8	64.8	54.3	2710	ohc	six	164	mpfi	3.31	3.19	9	121	4250	21	28	20970
14	0	bmw x3	gas	std	four	sedan	rwd	front	101.2	176.8	64.8	54.3	2765	ohc	six	164	mpfi	3.31	3.19	9	121	4250	21	28	21105
15	1	bmw z4	gas	std	four	sedan	rwd	front	103.5	189	66.9	55.7	3055	ohc	six	164	mpfi	3.31	3.19	9	121	4250	20	25	24565
16	0	bmw x4	gas	std	four	sedan	rwd	front	103.5	189	66.9	55.7	3230	ohc	six	209	mpfi	3.62	3.39	8	182	5400	16	22	30760
17	0	bmw x5	gas	std	two	sedan	rwd	front	103.5	193.8	67.9	53.7	3380	ohc	six	209	mpfi	3.62	3.39	8	182	5400	16	22	41315
18	0	bmw x3	gas	std	four	sedan	rwd	front	110	197	70.9	56.3	3505	ohc	six	209	mpfi	3.62	3.39	8	182	5400	15	20	36880
19	2	chevrolet impala	gas	std	two	hatchback	fwd	front	88.4	141.1	60.3	53.2	1488	l	three	61	2bbl	2.91	3.03	9.5	48	5100	47	53	5151
20	1	chevrolet monte carlo	gas	std	two	hatchback	fwd	front	94.5	155.9	63.6	52	1874	ohc	four	90	2bbl	3.03	3.11	9.6	70	5400	38	43	6295
21	0	chevrolet vega 2300	gas	std	four	sedan	fwd	front	94.5	158.8	63.6	52	1909	ohc	four	90	2bbl	3.03	3.11	9.6	70	5400	38	43	6575
22	1	dodge rampage	gas	std	two	hatchback	fwd	front	93.7	157.3	63.8	50.8	1876	ohc	four	90	2bbl	2.97	3.23	9.41	68	5500	37	41	5572
23	1	dodge challenger se	gas	std	two	hatchback	fwd	front	93.7	157.3	63.8	50.8	1876	ohc	four	90	2bbl	2.97	3.23	9.4	68	5500	31	38	6377
24	1	dodge d200	gas	turbo	two	hatchback	fwd	front	93.7	157.3	63.8	50.8	2128	ohc	four	98	mpfi	3.03	3.39	7.6	102	5500	24	30	7957
25	1	dodge monaco (sw)	gas	std	four	hatchback	fwd	front	93.7	157.3	63.8	50.6	1967	ohc	four	90	2bbl	2.97	3.23	9.4	68	5500	31	38	6229
26	1	dodge colt hardtop	gas	std	four	sedan	fwd	front	93.7	157.3	63.8	50.6	1989	ohc	four	90	2bbl	2.97	3.23	9.4	68	5500	31	38	6692
27	1	dodge colt (sw)	gas	std	four	sedan	fwd	front	93.7	157.3	63.8	50.6	1989	ohc	four	90	2bbl	2.97	3.23	9.4	68	5500	31	38	7609
28	1	dodge coronet custom	gas	turbo	two	sedan	fwd	front	93.7	157.3	63.8	50.6	2191	ohc	four	98	mpfi	3.03	3.39	7.6	102	5500	24	30	8558
29	-1	dodge dart custom	gas	std	four	wagon	fwd	front	103.3	174.6	64.6	59.8	2535	ohc	four	122	2bbl	3.34	3.46	8.5	88	5000	24	30	8921
30	3	dodge coronet custom (sw)	gas	turbo	two	hatchback	fwd	front	95.9	173.2	66.3	50.2	2811	ohc	four	156	mfi	3.6	3.9	7	145	5000	19	24	12964
31	2	honda civic	gas	std	two	hatchback	fwd	front	86.6	144.6	63.9	50.8	1713	ohc	four	92	1bbl	2.91	3.41	9.6	58	4800	49	54	6479
32	2	honda civic cvcc	gas	std	two	hatchback	fwd	front	86.6	144.6	63.9	50.8	1819	ohc	four	92	1bbl	2.91	3.41	9.2	76	6000	31	38	6855
33	1	honda civic	gas	std	two	hatchback	fwd	front	93.7	150	64	52.6	1837	ohc	four	79	1bbl	2.91	3.07	10.1	60	5500	38	42	5399
34	1	honda accord cvcc	gas	std	two	hatchback	fwd	front	93.7	150	64	52.6	1940	ohc	four	92	1bbl	2.91	3.41	9.2	76	6000	30	34	6529
35	1	honda civic cvcc	gas	std	two	hatchback	fwd	front	93.7	150	64	52.6	1956	ohc	four	92	1bbl	2.91	3.41	9.2	76	6000	30	34	7129
36	0	honda accord lx	gas	std	four	sedan	fwd	front	96.5	163.4	64	54.5	2010	ohc	four	92	1bbl	2.91	3.41	9.2	76	6000	30	34	7295
37	0	honda civic 1500 gl	gas	std	four	wagon	fwd	front	96.5	157.1	63.9	58.3	2024	ohc	four	92	1bbl	2.92	3.41	9.2	76	6000	30	34	7295
38	0	honda accord	gas	std	two	hatchback	fwd	front	96.5	167.5	65.2	53.3	2236	ohc	four	110	1bbl	3.15	3.58	9	86	5800	27	33	7895
39	0	honda civic 1300	gas	std	two	hatchback	fwd	front	96.5	167.5	65.2	53.3	2289	ohc	four	110	1bbl	3.15	3.58	9	86	5800	27	33	9095
40	0	honda prelude	gas	std	four	sedan	fwd	front	96.5	175.4	65.2	54.1	2304	ohc	four	110	1bbl	3.15	3.58	9	86	5800	27	33	8845
41	0	honda accord	gas	std	four	sedan	fwd	front	96.5	175.4	62.5	54.1	2372	ohc	four	110	1bbl	3.15	3.58	9	86	5800	27	33	10295
42	0	honda civic	gas	std	four	sedan	fwd	front	96.5	175.4	65.2	54.1	2465	ohc	four	110	mpfi	3.15	3.58	9	101	5800	24	28	12945
43	1	honda civic (auto)	gas	std	two	sedan	fwd	front	96.5	169.1	66	51	2293	ohc	four	110	2bbl	3.15	3.58	9.1	100	5500	25	31	10345
44	0	isuzu MU-X	gas	std	four	sedan	rwd	front	94.3	170.7	61.8	53.5	2337	ohc	four	111	2bbl	3.31	3.23	8.5	78	4800	24	29	6785
45	1	isuzu D-Max	gas	std	two	sedan	fwd	front	94.5	155.9	63.6	52	1874	ohc	four	90	2bbl	3.03	3.11	9.6	70	5400	38	43	8916.5
46	0	isuzu D-Max V-Cross	gas	std	four	sedan	fwd	front	94.5	155.9	63.6	52	1909	ohc	four	90	2bbl	3.03	3.11	9.6	70	5400	38	43	8916.5
47	2	isuzu D-Max	gas	std	two	hatchback	rwd	front	96	172.6	65.2	51.4	2734	ohc	four	119	spfi	3.43	3.23	9.2	90	5000	24	29	11048
48	0	jaguar xj	gas	std	four	sedan	rwd	front	113	199.6	69.6	52.8	4066	dohc	six	258	mpfi	3.63	4.17	8.1	176	4750	15	19	32250
49	0	jaguar xf	gas	std	four	sedan	rwd	front	113	199.6	69.6	52.8	4066	dohc	six	258	mpfi	3.63	4.17	8.1	176	4750	15	19	35550
50	0	jaguar xk	gas	std	two	sedan	rwd	front	102	191.7	70.6	47.8	3950	ohcv	twelve	326	mpfi	3.54	2.76	11.5	262	5000	13	17	36000
51	1	maxda rx3	gas	std	two	hatchback	fwd	front	93.1	159.1	64.2	54.1	1890	ohc	four	91	2bbl	3.03	3.15	9	68	5000	30	31	5195
52	1	maxda glc deluxe	gas	std	two	hatchback	fwd	front	93.1	159.1	64.2	54.1	1900	ohc	four	91	2bbl	3.03	3.15	9	68	5000	31	38	6095
53	1	mazda rx2 coupe	gas	std	two	hatchback	fwd	front	93.1	159.1	64.2	54.1	1905	ohc	four	91	2bbl	3.03	3.15	9	68	5000	31	38	6795
54	1	mazda rx-4	gas	std	four	sedan	fwd	front	93.1	166.8	64.2	54.1	1945	ohc	four	91	2bbl	3.03	3.15	9	68	5000	31	38	6695
55	1	mazda glc deluxe	gas	std	four	sedan	fwd	front	93.1	166.8	64.2	54.1	1950	ohc	four	91	2bbl	3.08	3.15	9	68	5000	31	38	7395
56	3	mazda 626	gas	std	two	hatchback	rwd	front	95.3	169	65.7	49.6	2380	rotor	two	70	4bbl	3.33	3.255	9.4	101	6000	17	23	10945
57	3	mazda glc	gas	std	two	hatchback	rwd	front	95.3	169	65.7	49.6	2380	rotor	two	70	4bbl	3.33	3.255	9.4	101	6000	17	23	11845
58	3	mazda rx-7 gs	gas	std	two	hatchback	rwd	front	95.3	169	65.7	49.6	2385	rotor	two	70	4bbl	3.33	3.255	9.4	101	6000	17	23	13645
59	3	mazda glc 4	gas	std	two	hatchback	rwd	front	95.3	169	65.7	49.6	2500	rotor	two	80	mpfi	3.33	3.255	9.4	135	6000	16	23	15645
60	1	mazda 626	gas	std	two	hatchback	fwd	front	98.8	177.8	66.5	53.7	2385	ohc	four	122	2bbl	3.39	3.39	8.6	84	4800	26	32	8845
61	0	mazda glc custom l	gas	std	four	sedan	fwd	front	98.8	177.8	66.5	55.5	2410	ohc	four	122	2bbl	3.39	3.39	8.6	84	4800	26	32	8495
62	1	mazda glc custom	gas	std	two	hatchback	fwd	front	98.8	177.8	66.5	53.7	2385	ohc	four	122	2bbl	3.39	3.39	8.6	84	4800	26	32	10595
63	0	mazda rx-4	gas	std	four	sedan	fwd	front	98.8	177.8	66.5	55.5	2410	ohc	four	122	2bbl	3.39	3.39	8.6	84	4800	26	32	10245

Bizga shunga o’hshash 205 qatorlik xalqaro bozorda sotilyapkan mashinalarning datalari taqdim etilgan edi va biz shu ma’lumotlarga tayanib yangi mashinaning narxini chiqazib berishimiz kerak,

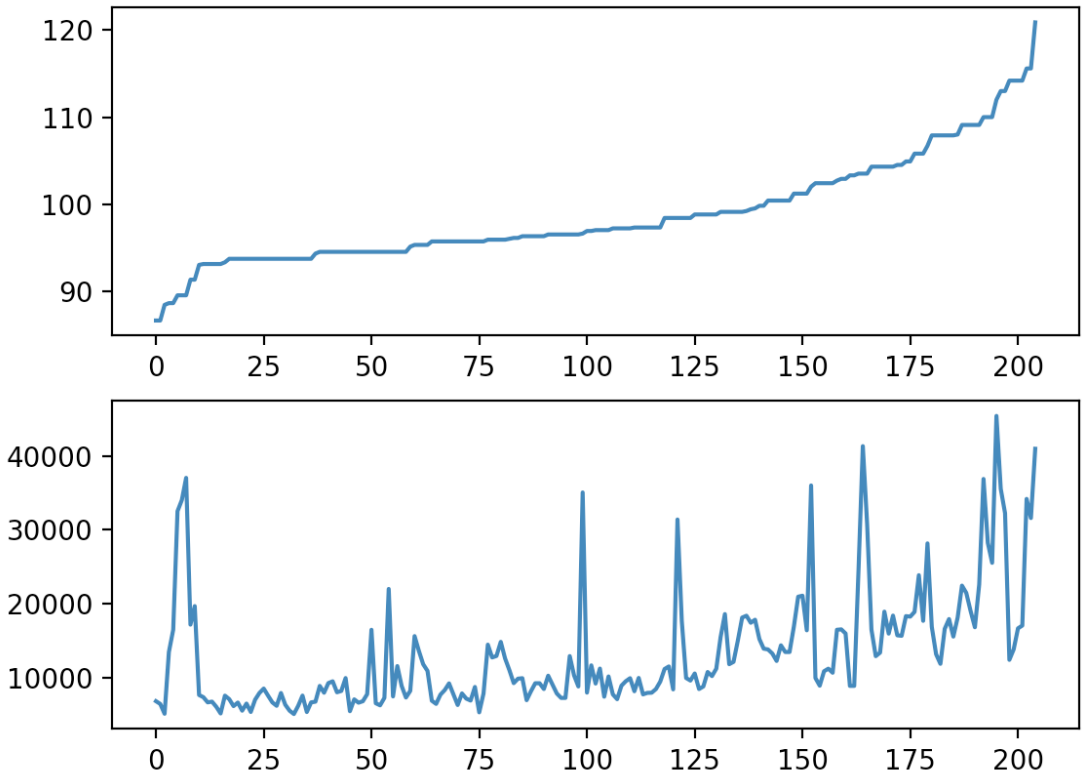
Mening birinchi o’rinda boshlagan ishim bu datani tozalash bo’ldi va men o’ylaymanki bu har bir data science sohadagi xodimni birinchi ishi bo’ladi chunki kamdan kam holatlarda sizga churoylik bir hil bosh joylari yo’qsiz ma’lumot taqdim etiladi, Bunga sabab nima? Bunga sabab shuki chunki kompaniyalarda ma’lumotlarni yozadigan xodim bitta bo’lmaydi shundan kelib chiqib har bir xodimning o’zini yozish stili va prinsiplari bo’ladi. A data science chiga bir dona ma’lumotham yo’ki boshqa shakilda, yo’ki bo’lmasa predikt qilish kerak bo’gan loyiha bo’sa shunchaki modellar ishlamaydi. Hop datani tozalab bo’ldik endi bo’ldi hamma datani modelga tiqib predikt bo’lib ketadi degani emas orada hali biz dependent va independent value larni topib aniqlab olishimiz

kerak. Meninig bu loyihamda dependent value bu price va independent value bu qaysi stolbesdagi ma’lumotlar price o’zgarishiga turtki beryapti shu ma’lumotlar, hali bu ma’lumotlarga yana qaytamiz albatta😊

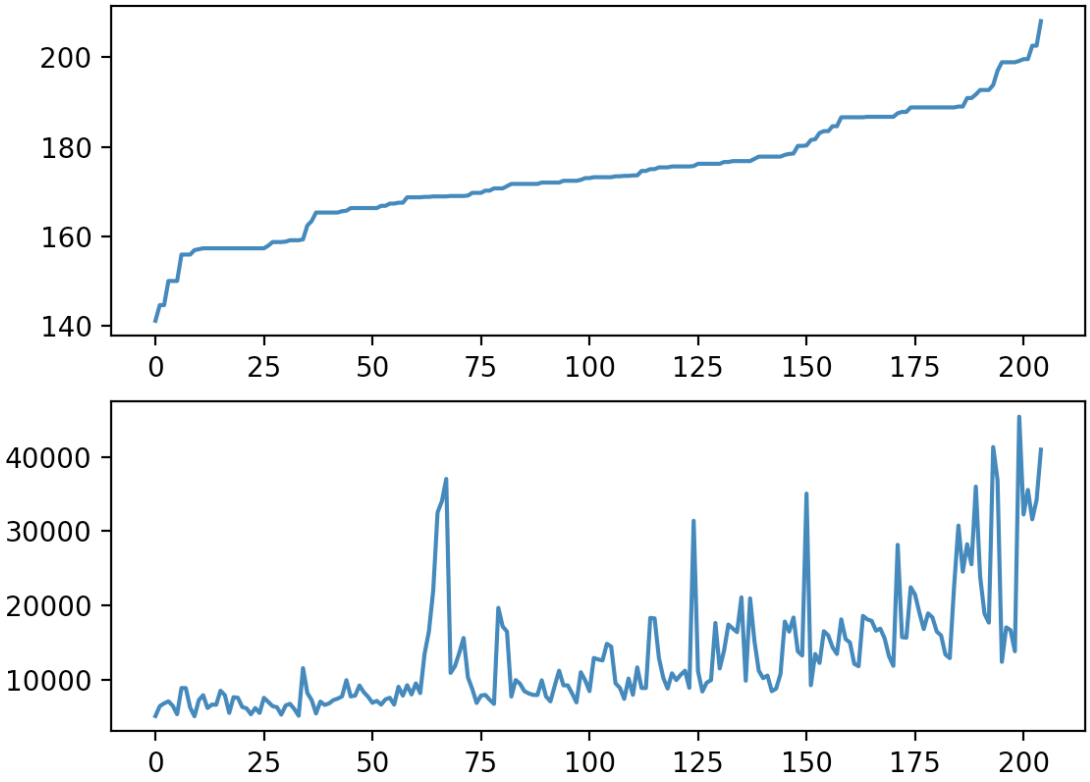
```
# ---------------------------------Clean and collect data-------------------------------------------------
df = pd.read_csv('auto_costs.csv')
df.name = df.name.map(lambda x: x.replace('-', ' '))
df.cylindernumber = df.cylindernumber.map(lambda x: x.strip())
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('four', '4'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('six', '6'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('five', '5'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('three', '3'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('twelve', '12'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('two', '2'))
df.cylindernumber = df.cylindernumber.map(lambda x: x.replace('eight', '8'))




sort_by_wheelbase = df.sort_values(by=['wheelbase'], ascending=True)
sort_by_carlength = df.sort_values(by=['carlength'], ascending=True)
sort_by_carwidth = df.sort_values(by=['carwidth'], ascending=True)
sort_by_carheight = df.sort_values(by=['carheight'], ascending=True)
sort_by_curbweight = df.sort_values(by=['curbweight'], ascending=True)
sort_by_enginesize = df.sort_values(by=['enginesize'], ascending=True)
sort_by_boreratio = df.sort_values(by=['boreratio'], ascending=True)
sort_by_stroke = df.sort_values(by=['stroke'], ascending=True)
sort_by_compressionratio = df.sort_values(by=['compressionratio'], ascending=True)
sort_by_horsepower = df.sort_values(by=['horsepower'], ascending=True)
sort_by_peakrpm = df.sort_values(by=['peakrpm'], ascending=True)
sort_by_citympg = df.sort_values(by=['citympg'], ascending=True)
sort_by_highwaympg = df.sort_values(by=['highwaympg'], ascending=True)
sort_by_cylindernumber = df.sort_values(by=['cylindernumber'], ascending=True)
```

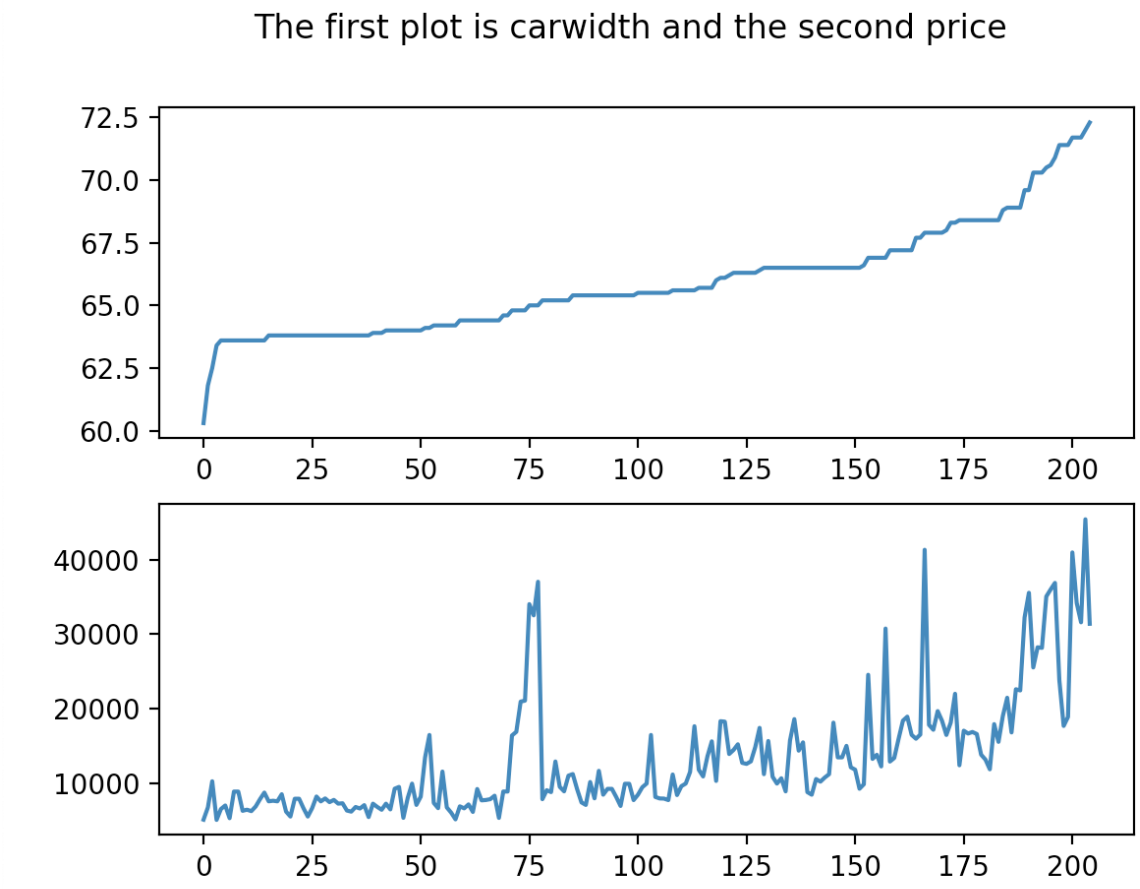
Mashinalarning narhini o’zgarishiga majbur qilyapkan ma’lumotlarni topish

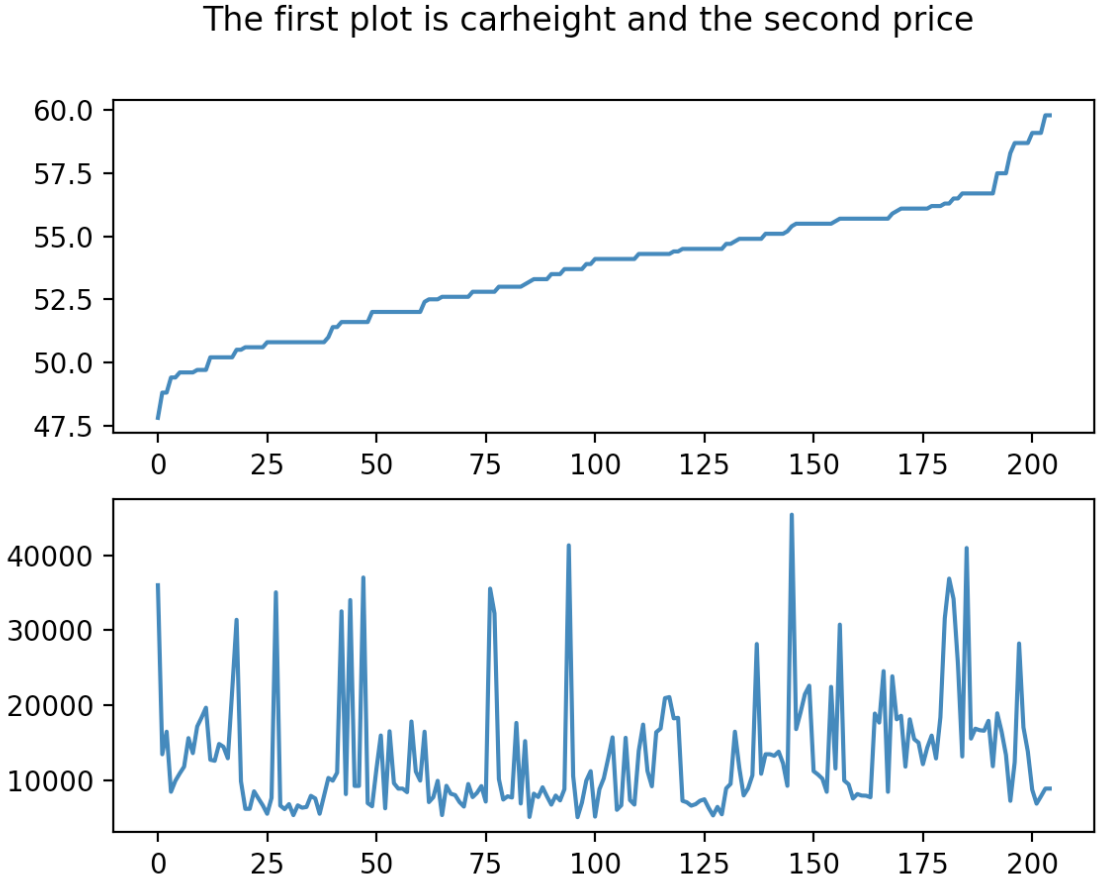
The first plot is wheelbase and the second price



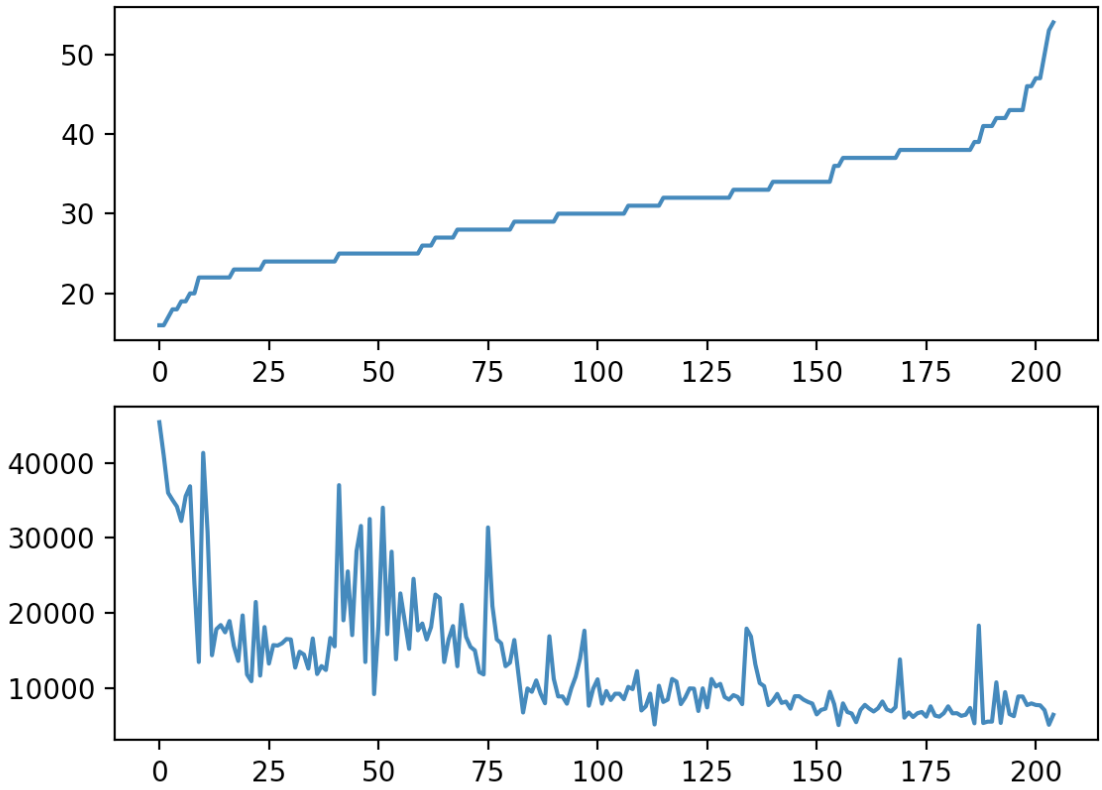
The first plot is carlength and the second price

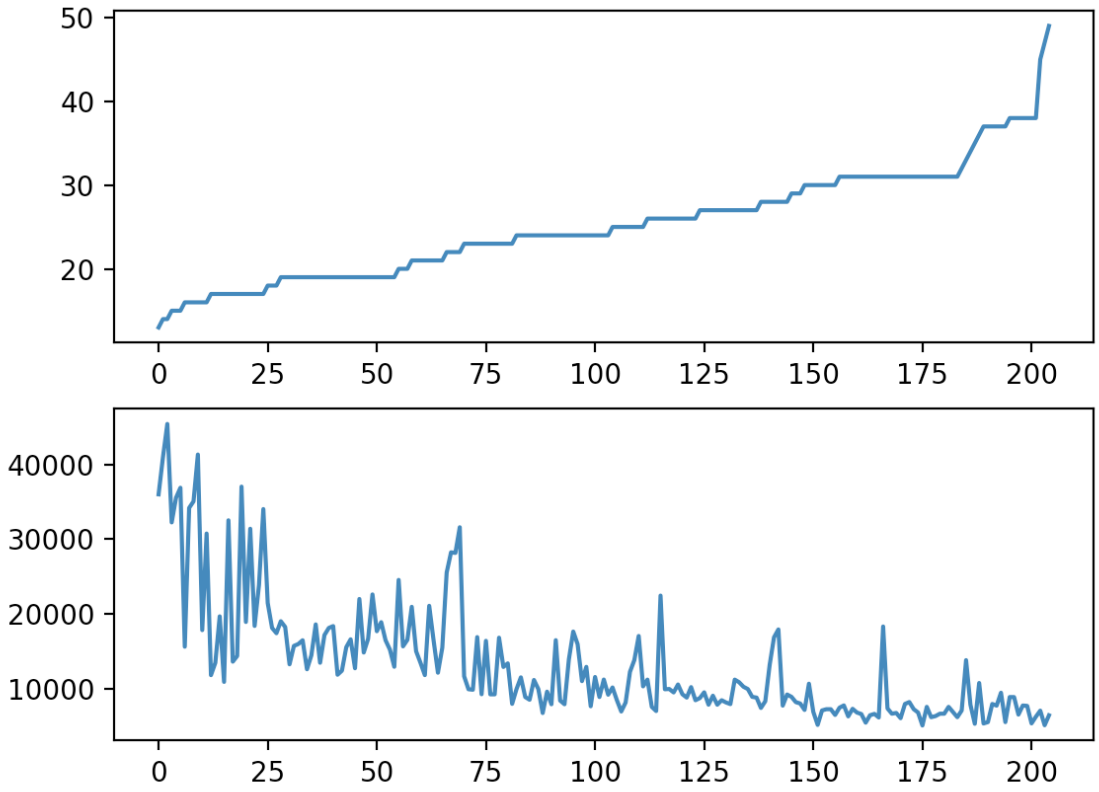



The first plot is carwidth and the second price
72.5
70.0
67.5
65.0
62.5
60.0
0
25
50
75
100
125
150
175
200
40000
30000
20000
10000
0
25
50
75
100
125
150
175
200

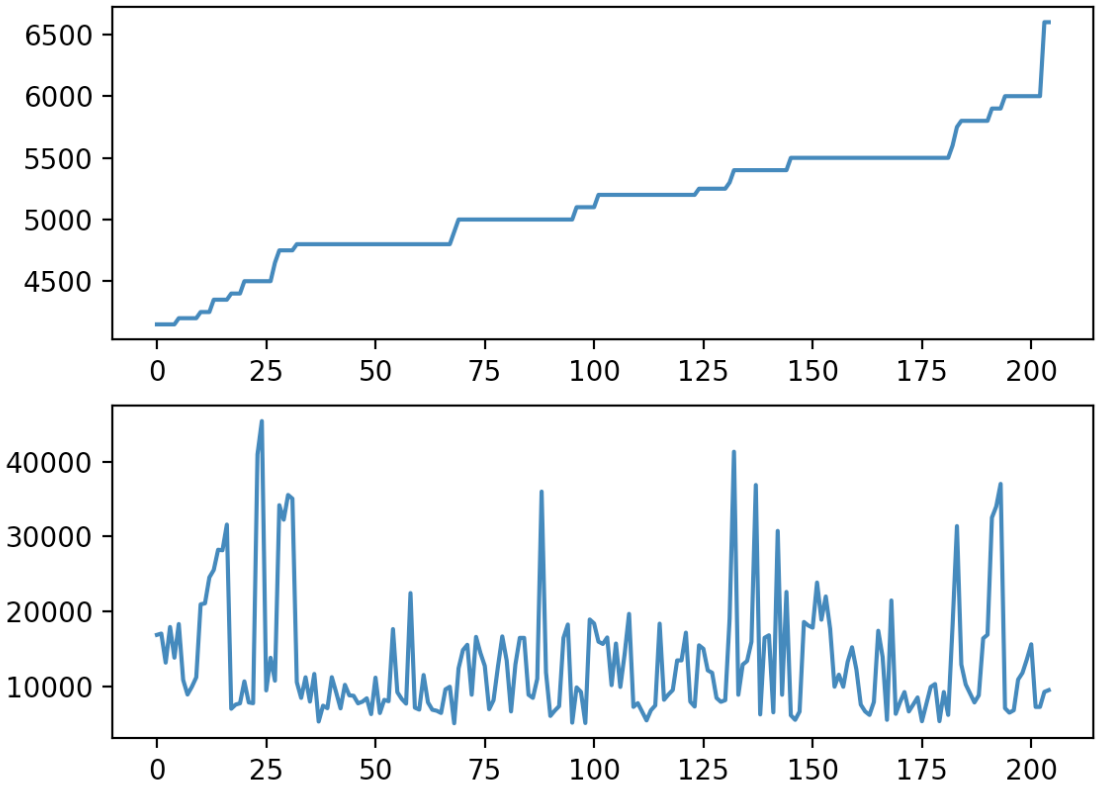

The first plot is carheight and the second price
60.0
57.5
55.0
52.5
50.0
47.5
0
25
50
75
100
125
150
175
200
40000
30000
20000
10000
0
25
50
75
100
125
150
175
200

The first plot is highwaympg and the second price

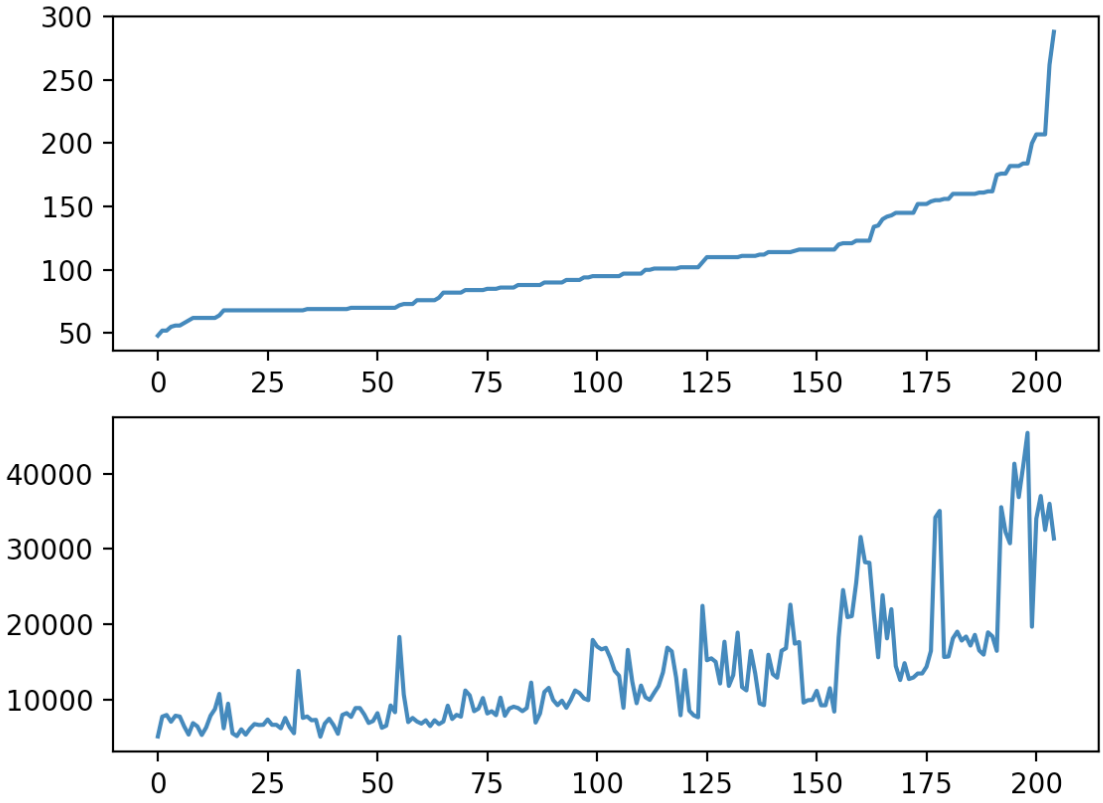



The first plot is citympg and the second price
50
40
30
20
0
25
50
75
100
125
150
175
200
40000
30000
20000
10000
0
25
50
75
100
125
150
175
200

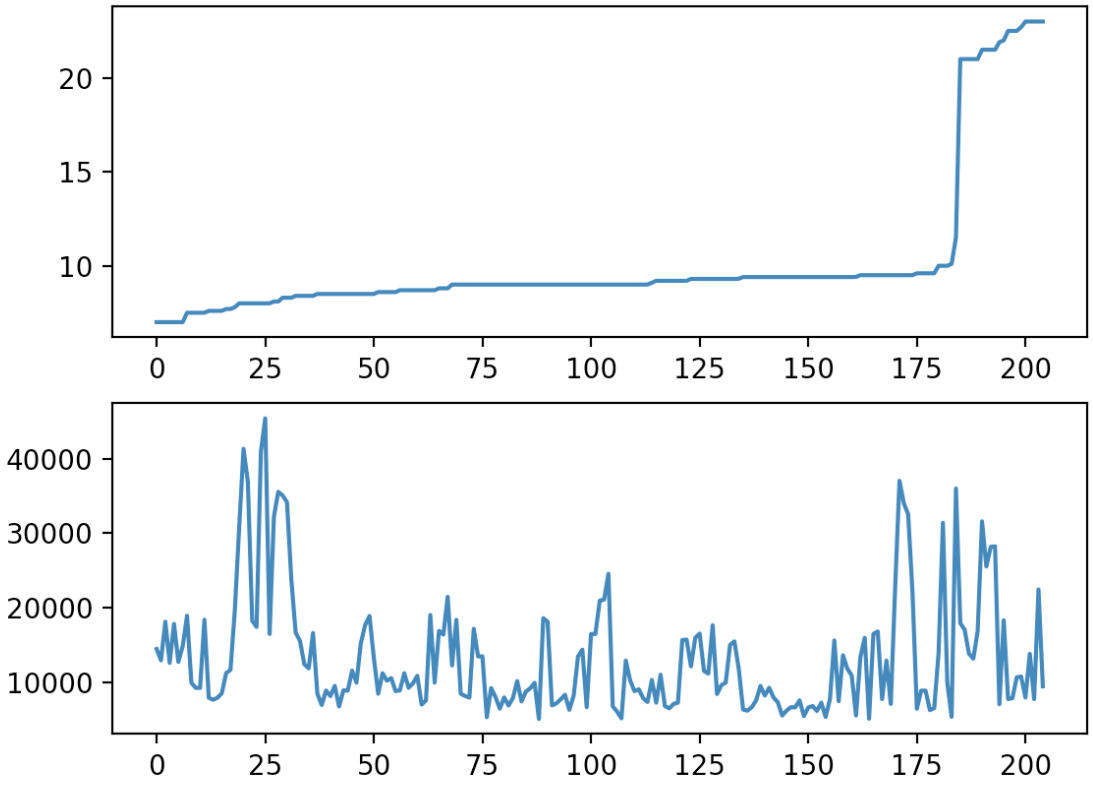
The first plot is peakrpm and the second price

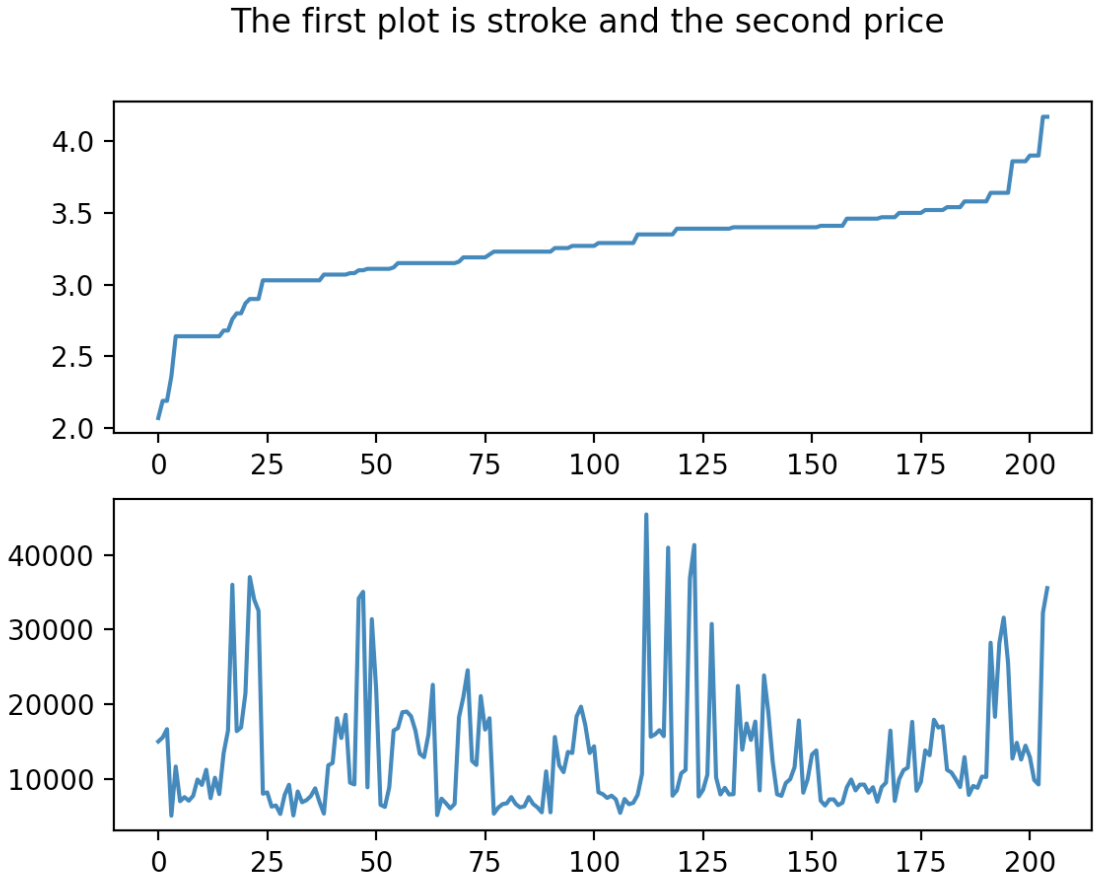


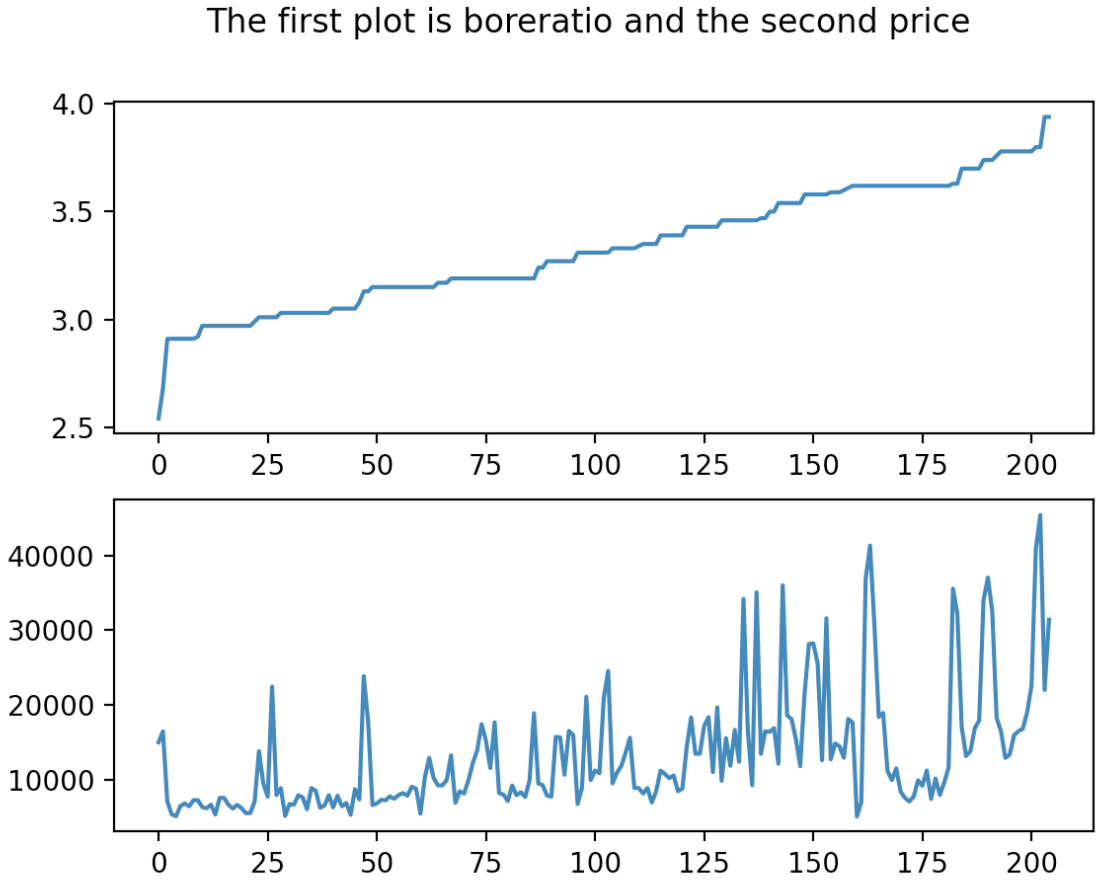
The first plot is horsepower and the second price



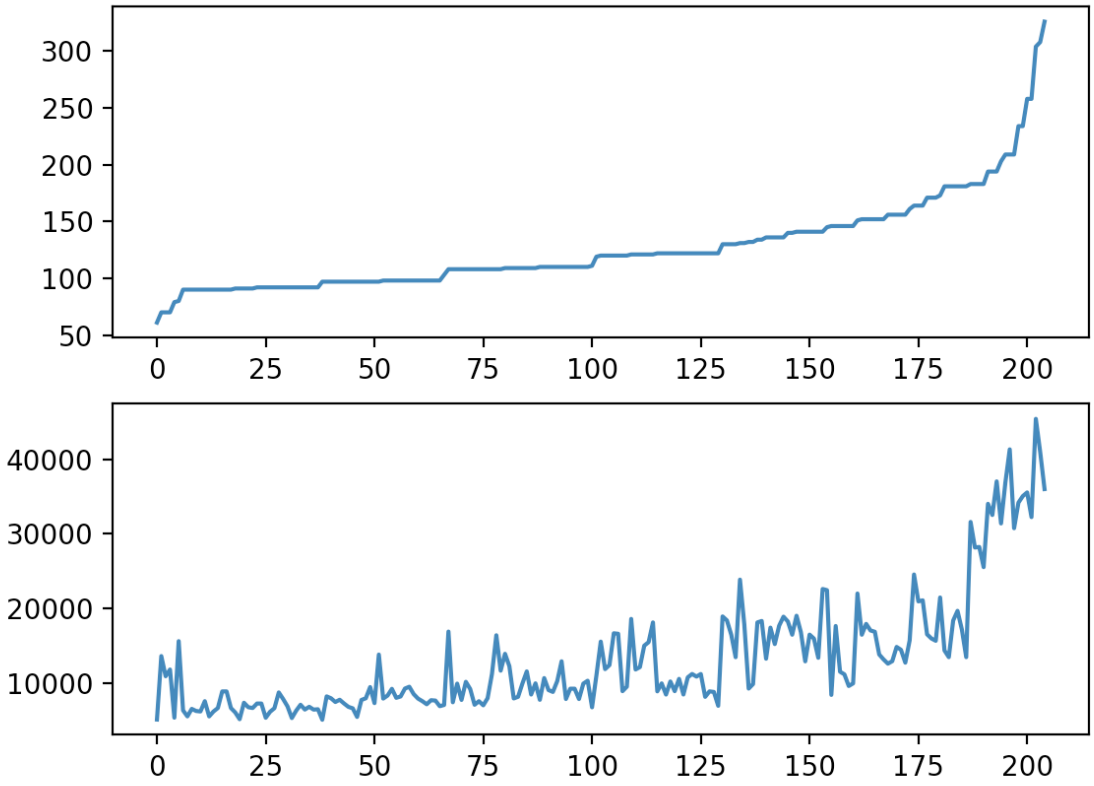
The first plot is compressionratio and the second price

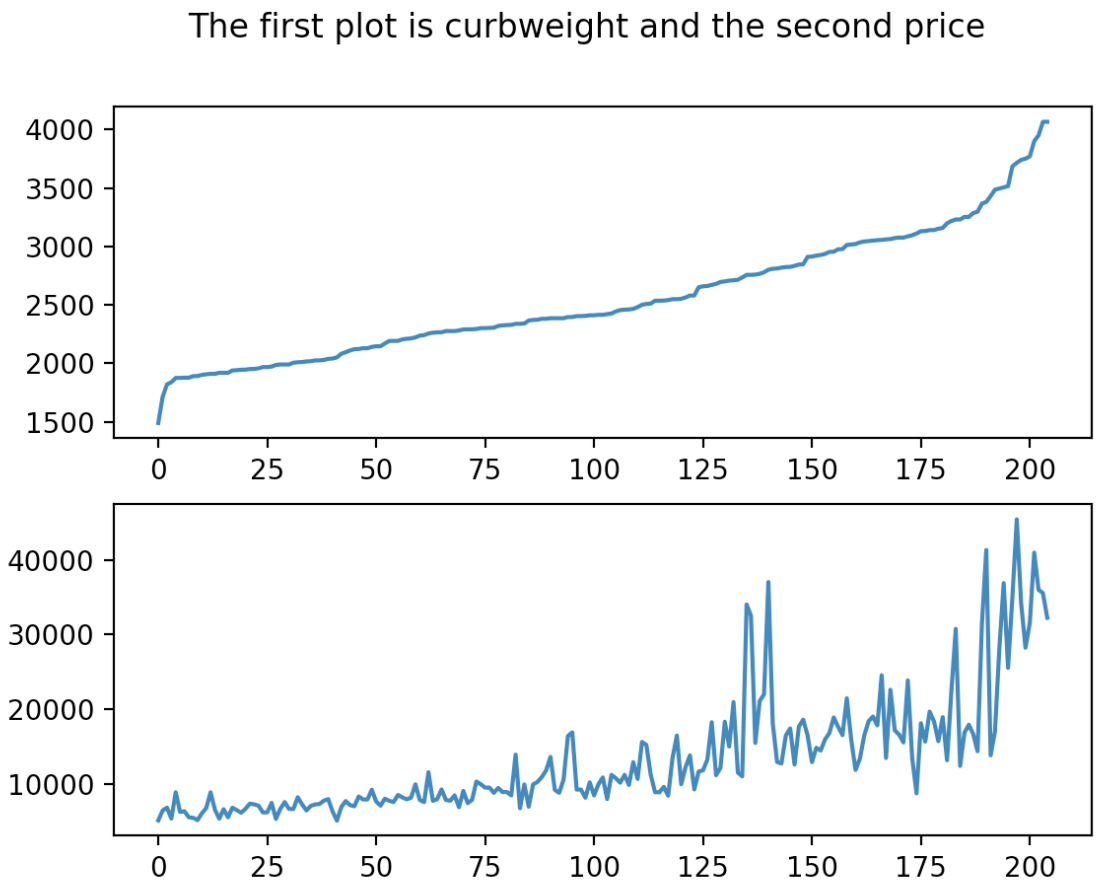



The first plot is stroke and the second price
4.0
3.5
3.0
2.5
2.0
0
25
50
75
100
125
150
175
200
40000
30000
20000
10000
0
25
50
75
100
125
150
175
200


The first plot is boreratio and the second price
4.0
3.5
3.0
2.5
0
25
50
75
100
125
150
175
200
40000
30000
20000
10000
0
25
50
75
100
125
150
175
200

The first plot is enginesize and the second price





Ma’lumotlarga izoh |^|

Va men tanlab olgan barcha ma’lumotlarimni Lineyniy Regressiyaga qoydim

```
# ----------------------Prediction----------------------
reg = linear_model.LinearRegression()
X = df[['carlength', 'carwidth', 'curbweight', 'enginesize', 'citympg', 'highwaympg', 'cylindernumber']]
y = df.price
SEED = 42
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = SEED)

reg.fit(X_train, y_train)
pred_val = reg.predict([[192.7, 71.4, 2844, 136, 19, 25, 5]])
print(pred_val)
```

```
[19429.04473352]
```

Ushbu ma’lumotlarimni berganimda mening predictim shunday javob berdi

Hop men javobni oldim lekin bu javob qanchali to’g’ri shuni tekshirishga qiziqdim

Va men solishtirdim mening predic value yimmi va aslida bo’gan javobni

```
# ----------------------------Actual and predicted values--------------------
y_pred = reg.predict(X_test)
df_preds = pd.DataFrame({'Actual': y_test.squeeze(), 'Predicted': y_pred.squeeze()})
# print(df_preds)
```

	Actual	Predicted
15	30760.000	25058.204070
9	17859.167	18592.794893
100	9549.000	11110.443008
132	11850.000	13608.618133
68	28248.000	24544.162129
95	7799.000	6641.935674
159	7788.000	8121.043659
162	9258.000	8105.646939
147	10198.000	10859.046537
182	7775.000	7569.335501
191	13295.000	16123.070756
164	8238.000	7428.233708
65	18280.000	16477.691951
175	9988.000	12077.100838
73	40960.000	38772.051504
152	6488.000	6674.067236
18	5151.000	-2922.880207
82	12629.000	18113.999154
86	8189.000	12244.098331
143	9960.000	10402.409978
60	8495.000	12315.429545
101	13499.000	21911.419979
98	8249.000	6802.975421
30	6479.000	2828.773830
25	6692.000	6662.458901
16	41315.000	25944.787330
168	9639.000	14715.906053
195	13415.000	16144.509748
97	7999.000	6349.716386
194	12940.000	15769.400519
67	25552.000	23821.615664
120	6229.000	6594.816253
154	7898.000	7191.579411
202	21485.000	22053.706931
79	7689.000	9120.834179
69	28176.000	23996.563728
145	11259.000	11143.571408
55	10945.000	8636.035090
45	8916.500	4517.957727
84	14489.000	18399.943075
146	7463.000	9337.126610

Va menga shunday javob chiqdi buyerda bizga ma’lumot hajmi kam bo’lgani uchun bu natija men judaham zo’r deb o’ylayman

Hop solishtirishga solishtirdik endi qanchali man notori javob berdim buni MSE-Mean Squared Error yordamida topiladi.

```
88  # ----------------------------Accurace of prediction--------------------
89  mae = mean_absolute_error(y_test, y_pred)
90  mse = mean_squared_error(y_test, y_pred)
91  rmse = np.sqrt(mse)
92
93  print(f'Mean absolute error: {mae}')
94  print(f'Mean squared error: {mse}')
95  print(f'Root mean squared error: {rmse}')
96  # print(reg.coef_)
97
98  accurancy = r2_score(y_test, y_pred)
99  print(accurancy)
100
101
```

```
Mean absolute error: 2720.309810967483
Mean squared error: 15524342.371948734
Root mean squared error: 3940.094208511864
Bizning predict 80.0% to'g'ri
```

Va men ushbu natijani oldim!!!

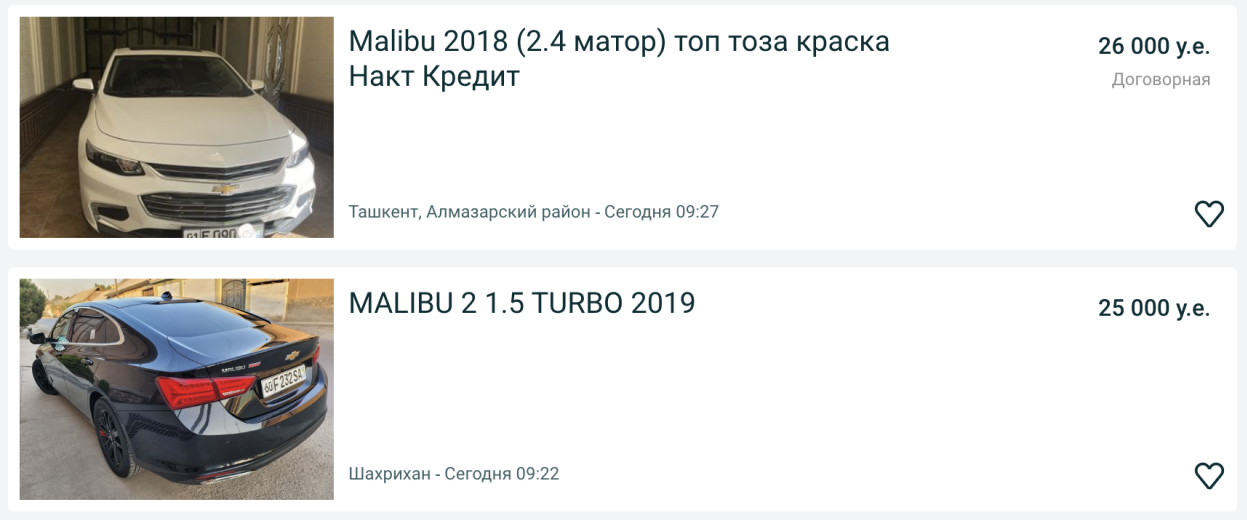
Asosiy maqsad

Ushbu dasturni yaratishdan mening asosiy maqsadim, O’zbekiston bozoridagi Malibu 2 Turbo avtomobilini, Haqiqiy haqqoniy narhini bilish edi va men ushbu natijaga erishdim:

```
59
60  # ----------------------Prediction----------------------
61  reg = linear_model.LinearRegression()
62  X = df[['carlength', 'carwidth', 'curbweight', 'enginesize', 'citympg', 'highwaympg', 'cylindernumber']]
63  y = df.price
64  SEED = 42
65  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = SEED)
66
67  reg.fit(X_train, y_train)
68  pred_val = reg.predict([[193.7, 73, 3273.8, 122, 23.5, 33.6, 4]])
69  print(pred_val)
70
```

```
[20123.36161707]
```

Va shuni bildimki Malibu 2 Turbo avtomobilining haqiqiy narxi 20ming do’llardan sag’al balanroq ekan.



va ko’rib turganingizdak mening tahminlarim mutloq to’g’ri bo’lib chiqdi