

Tema Invatare Automata - Partea 1

Ariana-Maria Lazăr-Andrei 342CA

Contents

1 Explorarea și Vizualizarea datelor	3
1.1 Inchiriere biciclete	3
1.1.1 Vizualizare ca serie de timp a datelor din închirierea bicicletelor - existență trend-uri sau ciclicitate	5
1.1.2 Corelații cu target-ul, corelații între atrbute	7
1.1.3 Existența datelor lipsă	10
1.2 Autovit	10
1.2.1 Corelații cu target-ul, corelații între atrbute	13
1.2.2 Existența datelor lipsă	15
2 Extragerea, standardizarea, selecția de atrbute și suplimentarea valorilor lipsă	15
2.1 Standardizarea atrbutelor	15
2.2 Encodarea atrbutelor categorice sau ordinale	16
2.3 Imputarea valorilor lipsă	16
2.4 Discretizarea atrbutelor numerice	16
2.5 Selectia atrbutelor relevante	16
3 Utilizarea algoritmilor de Învățare Automată	16
3.1 Linear Regression	16
3.1.1 Inchiriere biciclete	16
3.1.2 Autovit	16
3.2 SVR	17
3.2.1 Inchiriere biciclete	17
3.2.2 Autovit	17
3.3 Random Forest Regressor	17
3.3.1 Inchiriere biciclete	17
3.3.2 Autovit	18
3.4 Gradient Boosted Regressor	18
3.4.1 Inchiriere biciclete	18
3.4.2 Autovit	21
3.5 Quantile Regressor	24
3.5.1 Inchiriere biciclete	24
3.5.2 Autovit	27
3.6 Procedura de cautare a hiperparametrilor	31
3.7 Evaluarea algoritmilor pe datasetul Bikes	31
3.7.1 Linear Regression	31
3.7.2 Support Vector Regressor	31
3.7.3 Random Forest Regressor	31
3.7.4 Gradient Boosting Regressor	31
3.7.5 Quantile Regressor	32
3.8 Evaluarea algoritmilor pe datasetul Autovit	32
3.8.1 Linear Regression	32
3.8.2 Support Vector Regressor	32

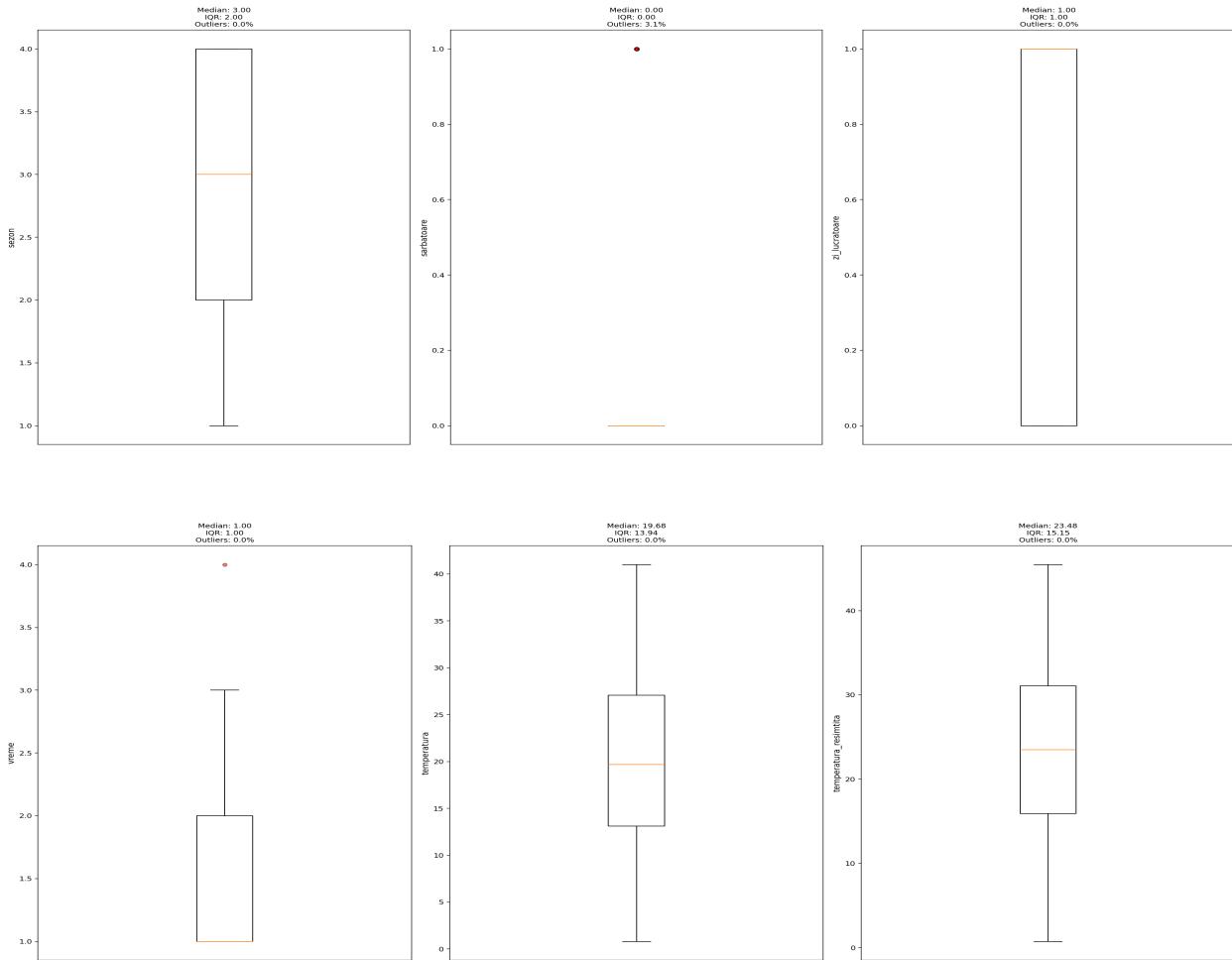
3.8.3	Random Forest Regressor	32
3.8.4	Gradient Boosting Regressor	32
3.8.5	Quantile Regressor	32

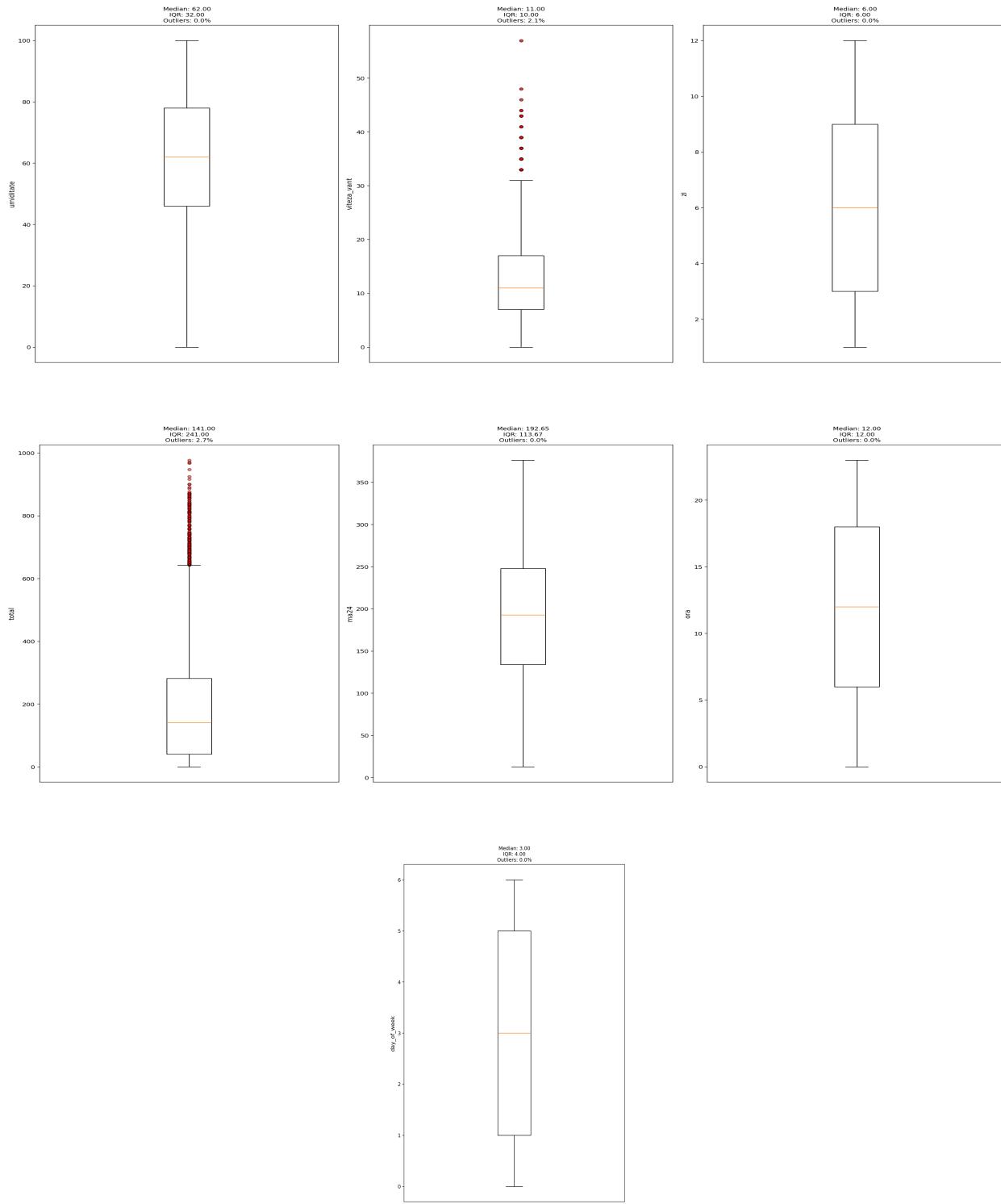
1 Explorarea și Vizualizarea datelor

1.1 Inchiriere biciclete

Table 1: Analiza atribute

Atribut	Non-null	Mean	Std	Min	25%	50%	75%	Max
sezon	6878	2.51	1.12	1.00	2.00	3.00	4.00	4.00
sarbatoare	6878	0.03	0.17	0.00	0.00	0.00	0.00	1.00
zi_lucratoare	6878	0.68	0.47	0.00	0.00	1.00	1.00	1.00
vreme	6878	1.42	0.64	1.00	1.00	1.00	2.00	4.00
temperatura	6878	20.08	8.17	0.82	13.12	19.68	27.06	41.00
temperatura_resimtita	6878	23.51	8.87	0.76	15.91	23.49	31.06	45.46
umiditate	6878	61.99	19.83	0.00	46.00	62.00	78.00	100.00
viteza_vant	6878	12.58	8.25	0.00	7.00	11.00	17.00	57.00
zi	6878	6.50	3.45	1.00	3.00	6.00	9.00	12.00
total	6878	188.87	179.67	1.00	41.00	141.00	282.00	977.00





Putem observa faptul există atribută care variază mult, sunt disperse într-un interval mare, au outlieri sau un interval concentrat de valori. Medianele indică simetrie pozitivă/negativă (și extrema dacă avem și whiskers de dimensiune mare).

1.1.1 Vizualizare ca serie de timp a datelor din închirierea bicicletelor - existență trend-uri sau ciclicitate

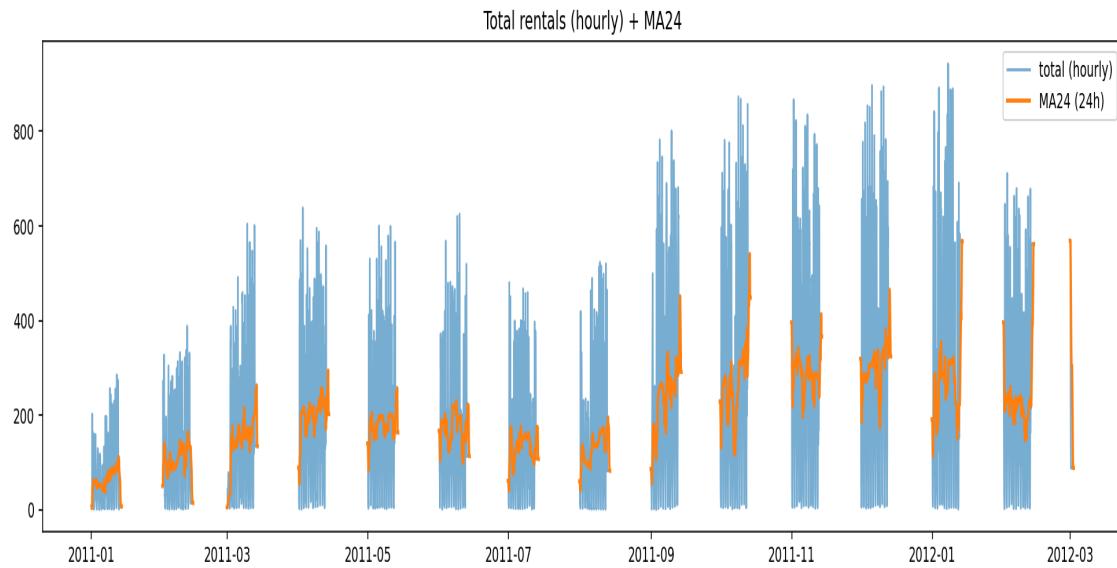


Figure 1: Datele orare arată o variabilitate zilnică extremă cu vârfuri clare în orele 7-9 AM, respectiv 5-7 PM și valori minime noaptea. Media mobilă pe 24 de ore evidențiază atât ciclicitatea săptămânală cât și tendința crescătoare.

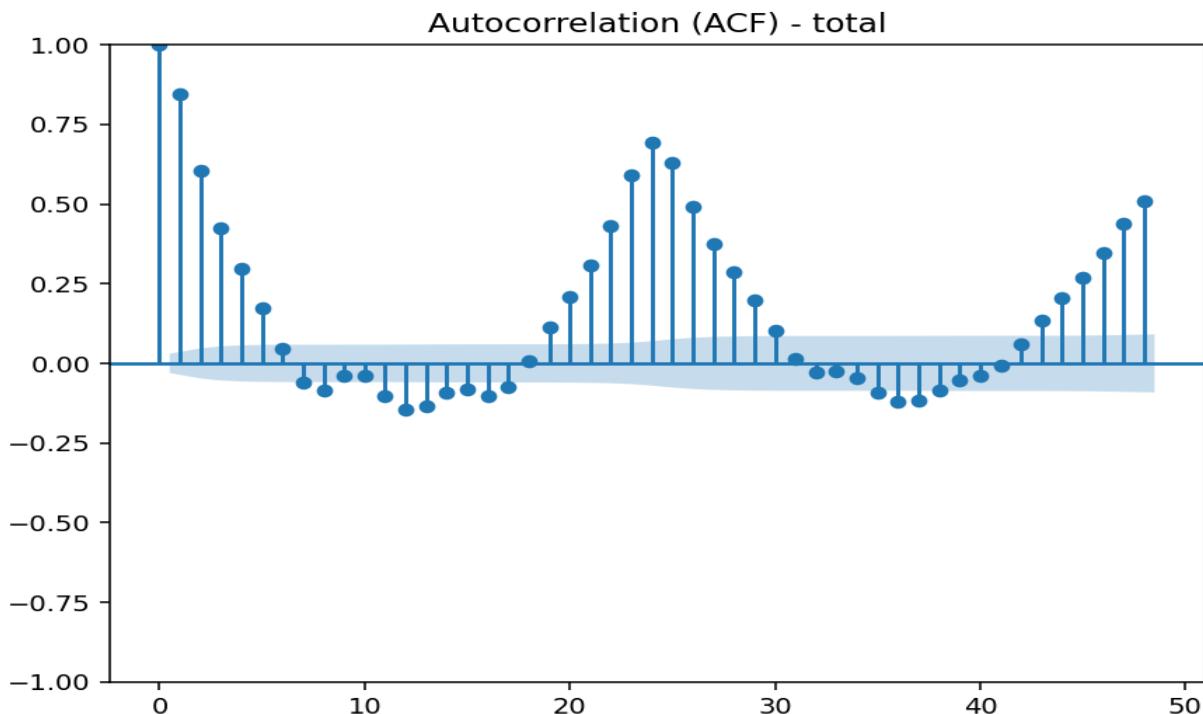


Figure 2: ciclicitate zilnică și săptămânală foarte pronunțată

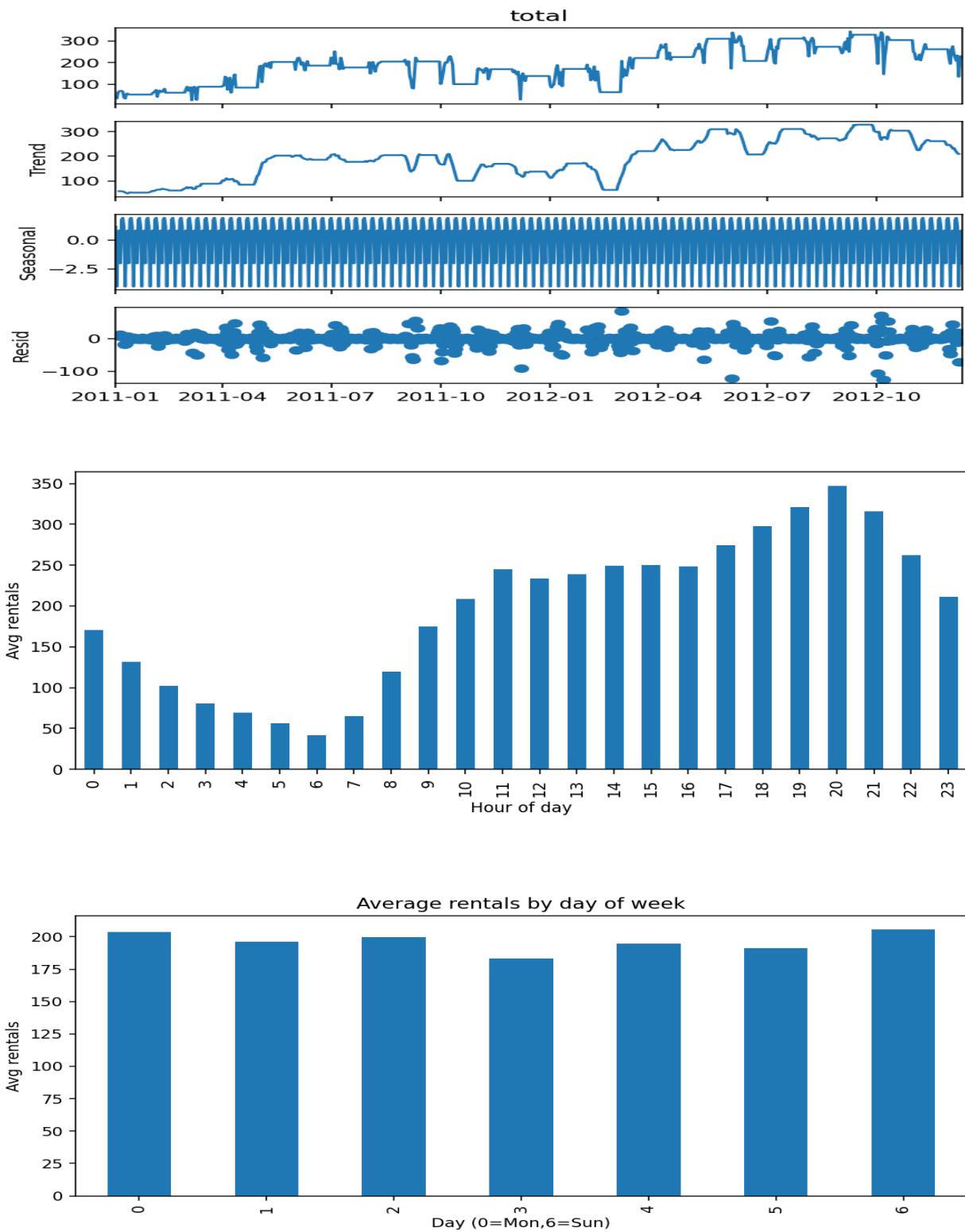


Figure 3: variație relativ uniformă

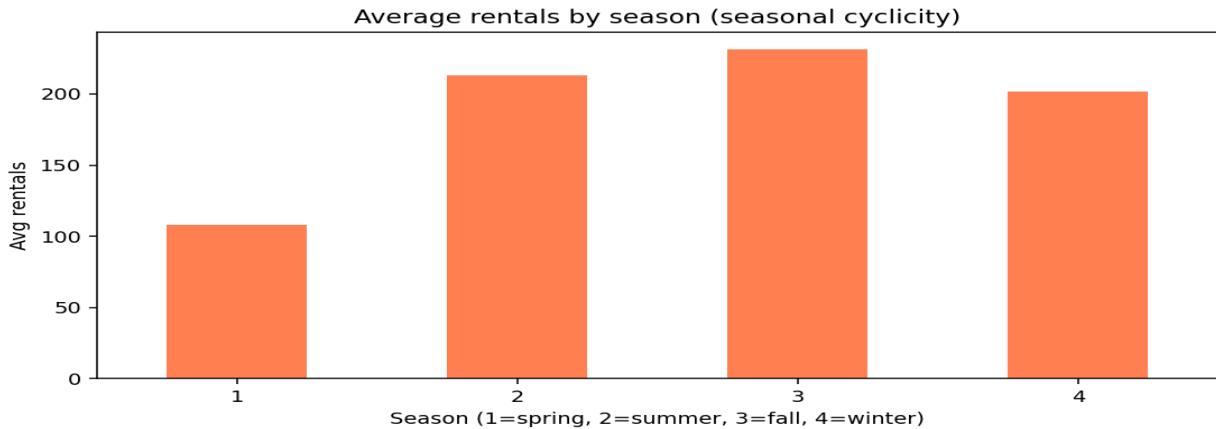


Figure 4: Analiza pe sezoane relevă diferențe semnificative între anotimpuri (diferență de aproximativ 2.2x a minimului față de sezonul de vârf).

1.1.2 Corelații cu target-ul, corelații între atrbute

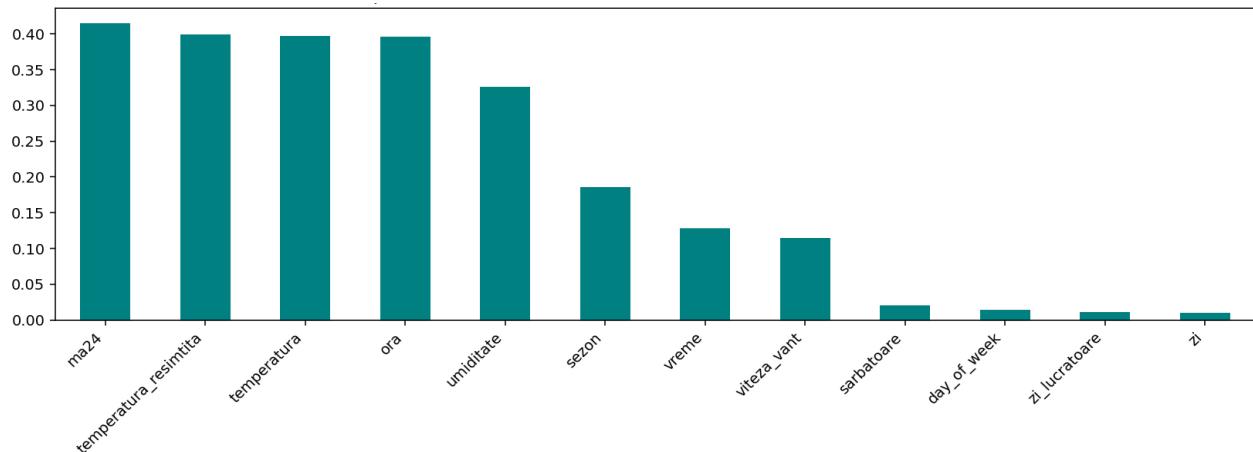
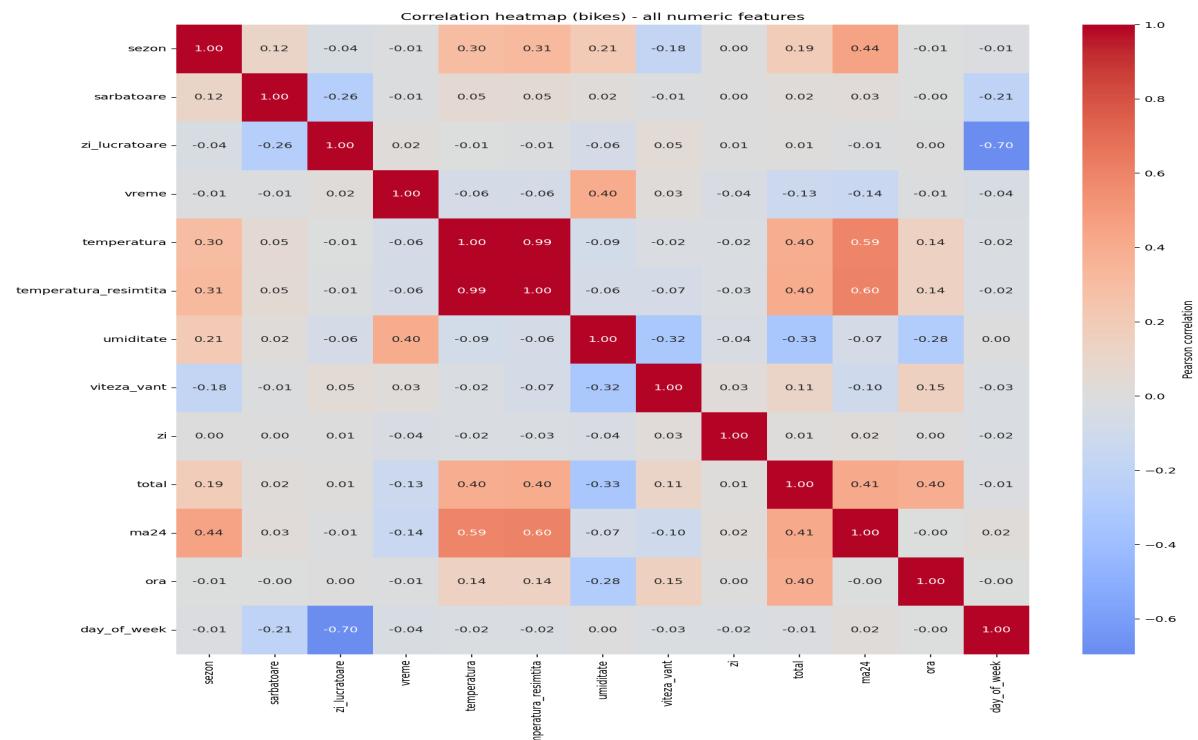


Table 2: Corelație cu target-ul pentru toate atrbutele principale

Atribut	Corelație cu target
sezon	0.21
sârbatoare	0.04
zi_lucratoare	0.03
vreme	0.00
temperatura	0.09
temperatura_resimtita	0.09
umiditate	0.11
viteza_vant	0.05
ocazionali	0.22
inregistrati	0.28
total	1.00
zi	0.05



Targetul are corelatie moderata pozitiva cu temperatura, relatia este non-liniara. In schimb, pentru umiditate corelatia este negativa. Viteza vantului poate fi asociata cu temperatura pentru o predictie corecta. Temperatura si temperatura resimtita au corelatie foarte mare, ceea ce este tratat prin feature engineering.

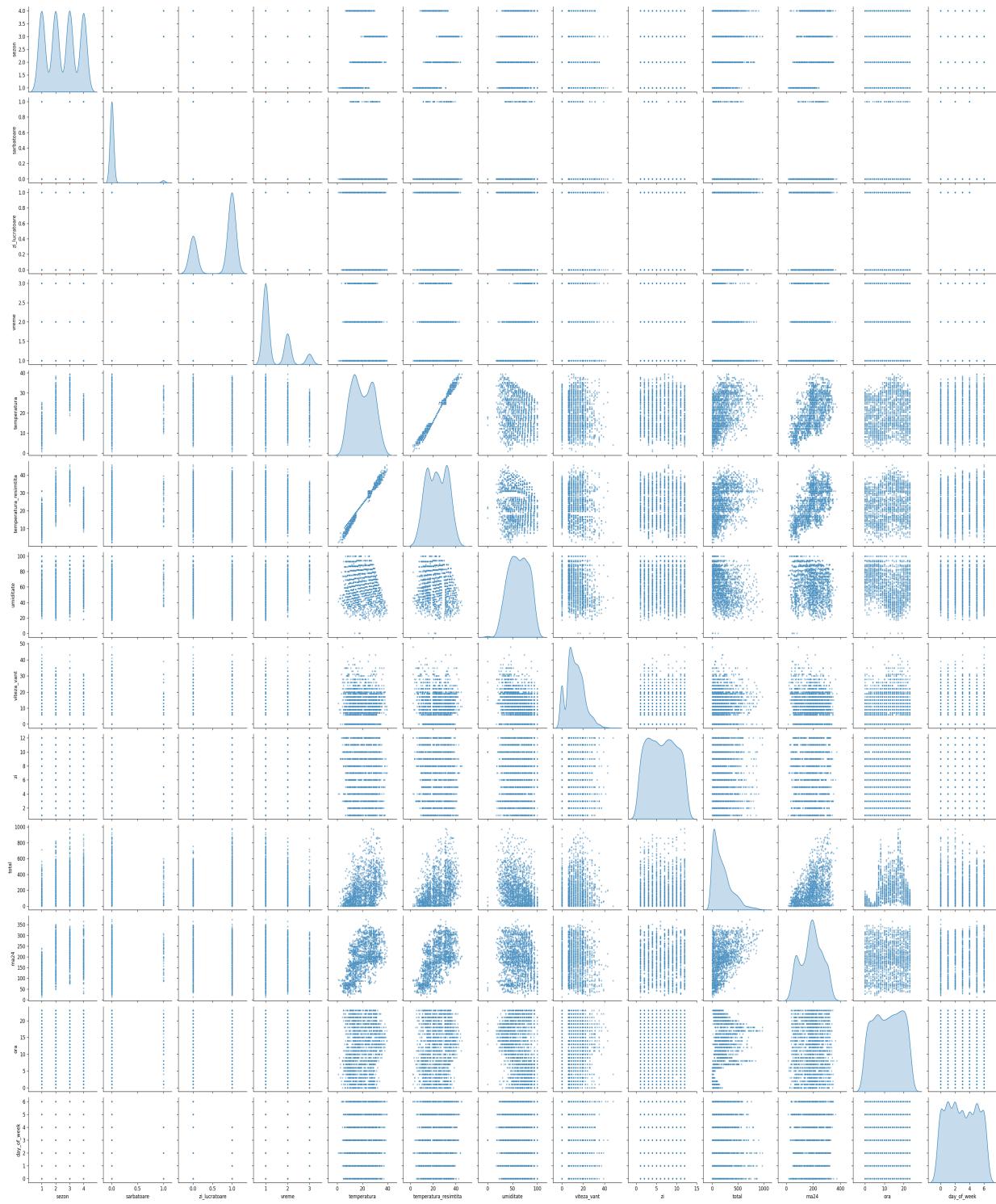


Figure 5: Pairplot inchiriere biciclete

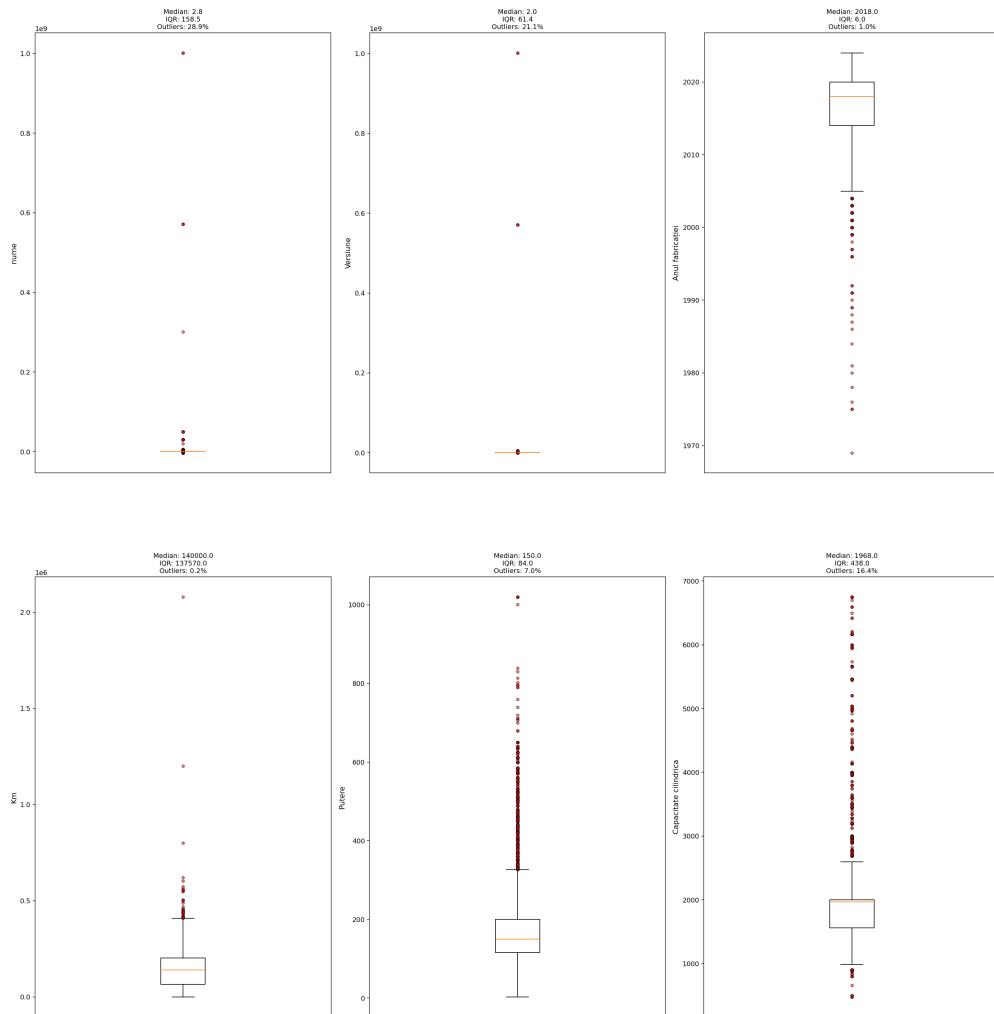
Graficul relevă relații directe (inregistrati-total sau temperatura-temperatura_resimtita), relații non-liniare (umiditate-total), distribuții neuniforme (histograme asimetrice pentru total sau valori discrete pentru sezon), precum și prezența outlierilor (total, ocazionali).

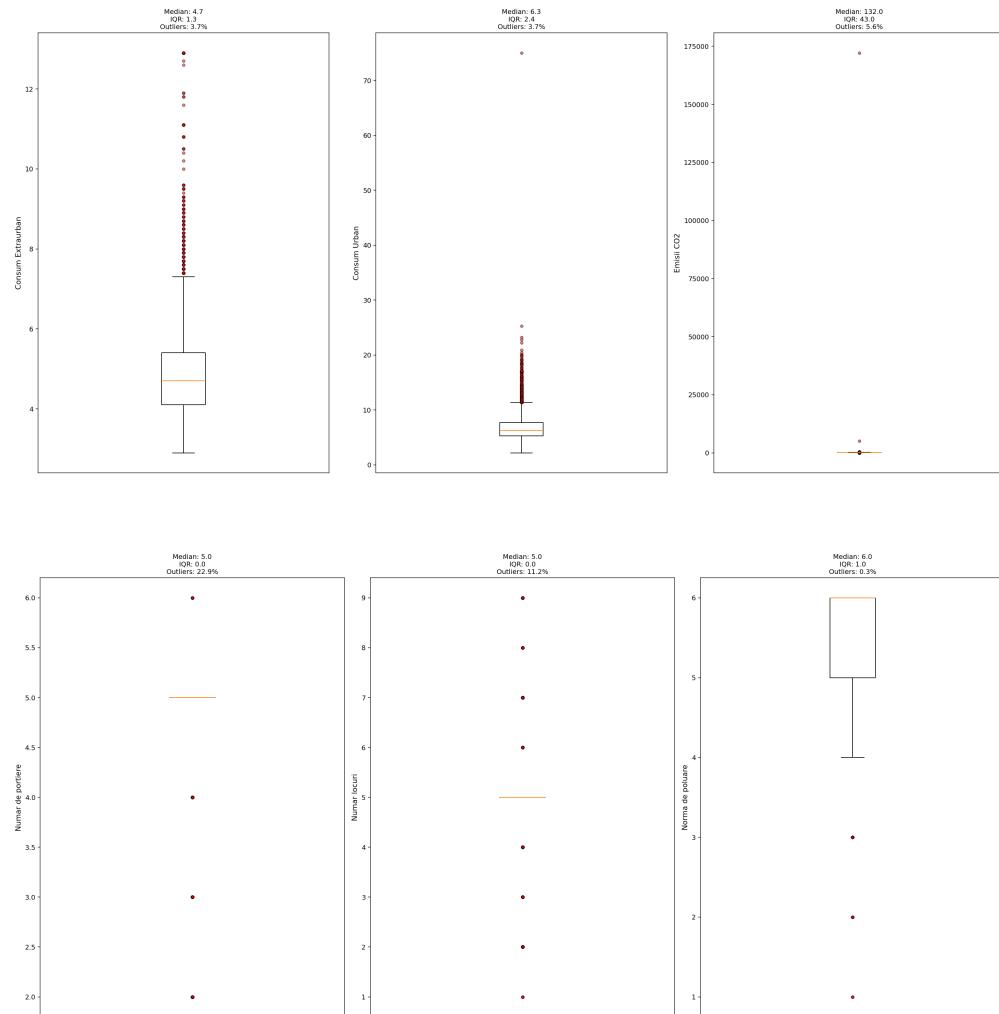
1.1.3 Existența datelor lipsă

Table 3: Procent date lipsă pentru toate atributele principale

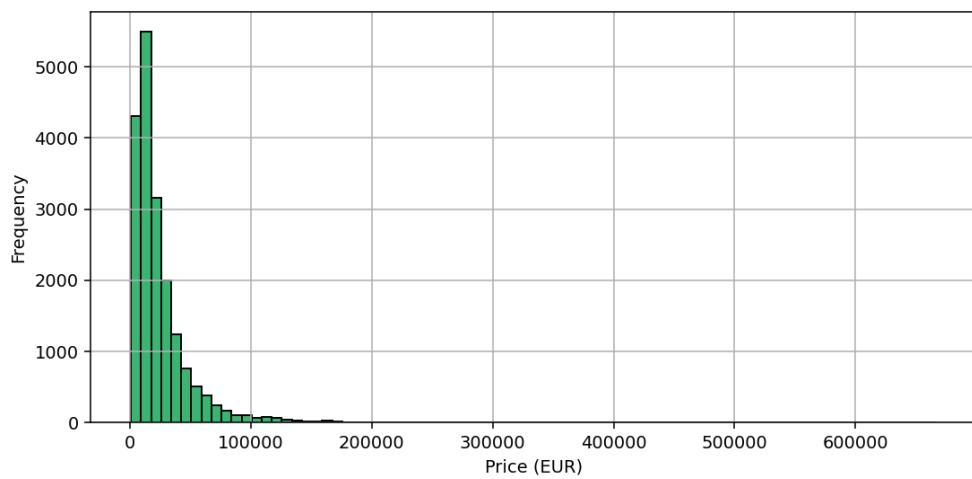
Atribut	Missing%
sezon	0.00
sarbatoare	0.00
zi_lucratoare	0.00
vreme	0.00
temperatura	0.00
temperatura_resimtita	0.00
umiditate	0.00
viteza_vant	0.00
ocazionali	0.00
inregistrati	0.00
total	41.70
zi	0.00

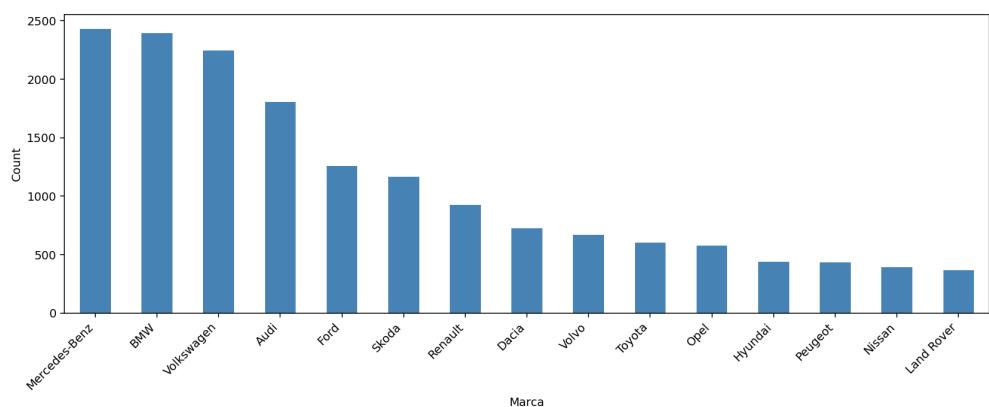
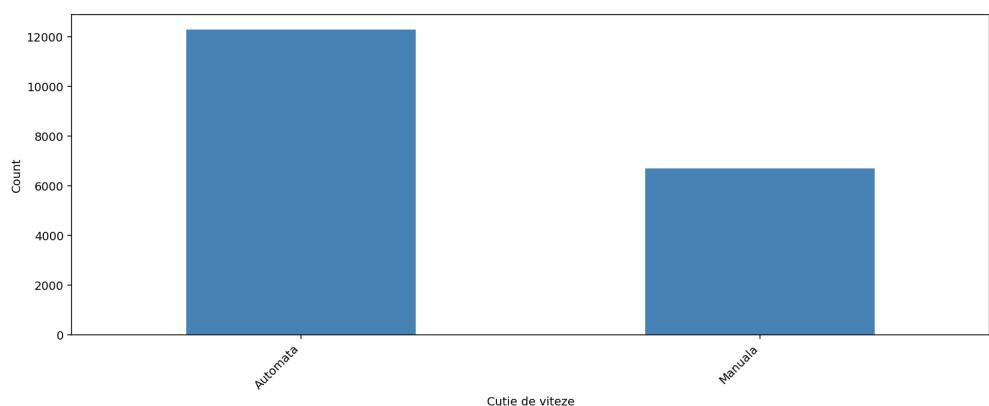
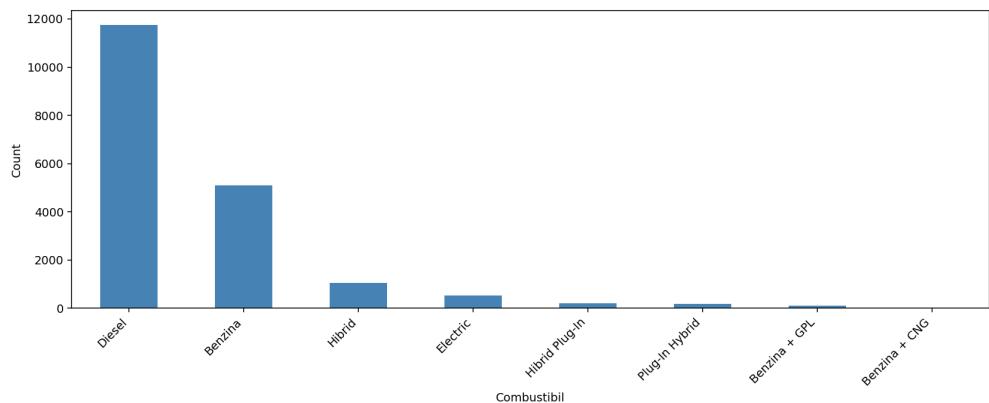
1.2 Autovit

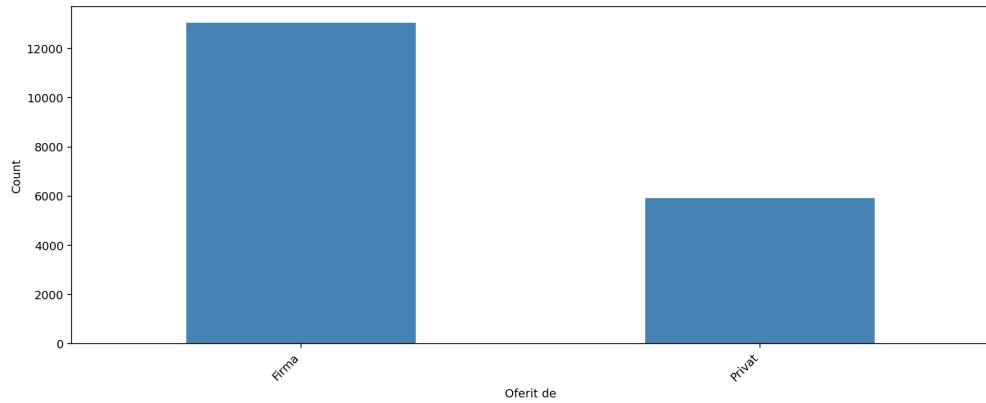




Boxploturile relevă variabilitate extremă cu majoritatea variabilelor concentrate în zona inferioară și peste 28% outlieri dispersați, necesitând transformări logaritmice și tratarea valorilor extreme.







1.2.1 Corelații cu target-ul, corelații între atrbute

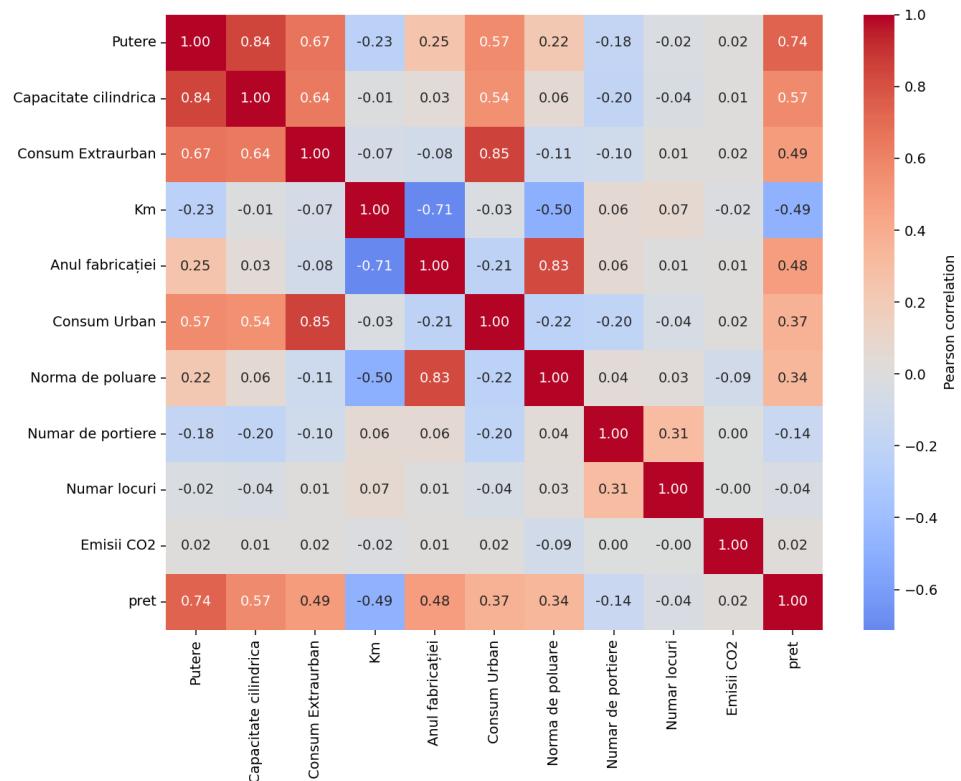


Figure 6: Corelatii intre atrbute

Putem afirma ca predictorii cei mai relevanți pentru target sunt Putere, Capacitate_cilindrica, Anul_fabricatiei și Km. Putere și Capacitate_cilindrica sunt puternic corelate, ce ar sugera eliminarea atributului redundant Capacitate_cilindrica.

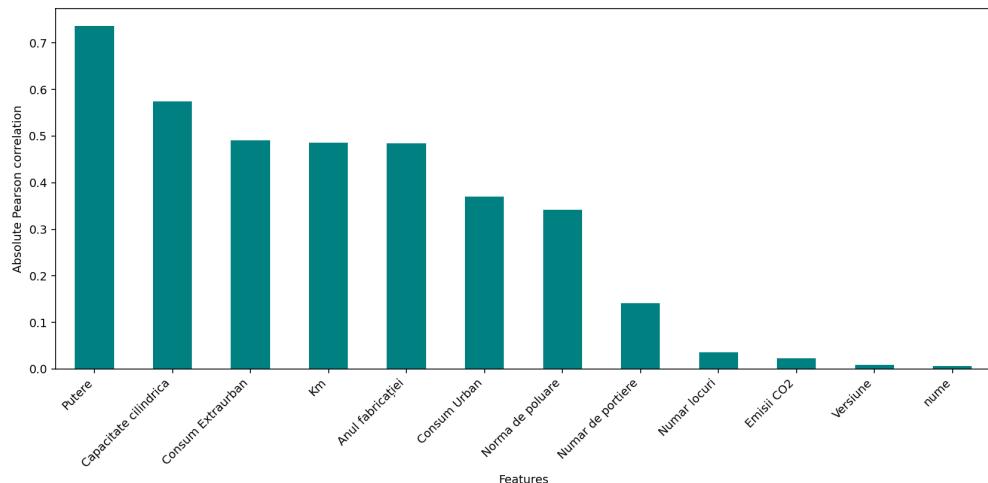
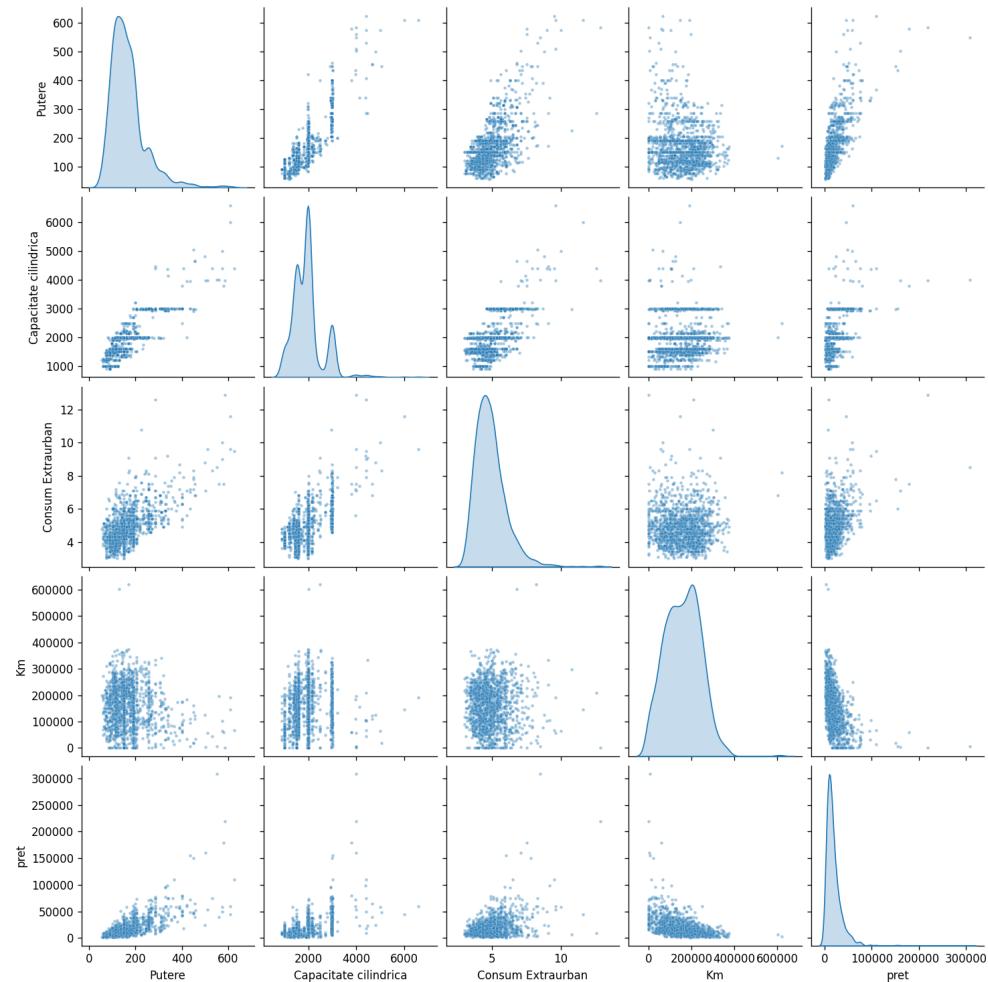


Figure 7: Corelatia cu targetul



Am testat eliminarea unor atribute puternic corelate, dar nu am obtinut o imbunatatiri notabile. Am decis sa nu elimin niciun atribut in solutia finala. In cazul Linear Regression,

Table 4: Corelație cu target-ul pentru atrbutele principale

Atribut	Corelație cu target
nume	0.01
pret	1.00
Versiune	0.01
Anul fabricației	0.48
Km	0.49
Putere	0.74
Capacitate cilindrica	0.58
Consum Extraurban	0.49
Consum Urban	0.37
Emisii CO2	0.02
Numar de portiere	0.14
Numar locuri	0.04
Norma de poluare	0.34

SVR si Quantile Regression am utilizat VarianceThreshold(1e-12) pentru a elimina categoriile cu varianță aproape nula rezultate după OneHotEncoding: 'cat_Are_VIN_(Serie_sasiu)_Da', 'cat_Primul_proprietar_(de_nou)_Da', 'cat_Fara_accident_in_istoric_Da', 'cat_Tuning_Da'.

Modelele arbore nu includ în splituri coloanele constante, respectiv cele liniare pot fi afectate de acest aspect.

1.2.2 Existența datelor lipsă

Table 5: Procent date lipsă pentru atrbutele principale

Atribut	Missing%
nume	31.60
pret	0.00
Versiune	38.00
Anul fabricației	0.00
Km	0.20
Putere	0.00
Capacitate cilindrica	2.80
Consum Extraurban	56.00
Consum Urban	31.20
Emisii CO2	49.30
Numar de portiere	1.20
Numar locuri	22.30
Norma de poluare	34.50

2 Extragerea, standardizarea, selecția de atrbute și suplimentarea valorilor lipsă

2.1 Standardizarea atrbutelor

Am standardizat atrbutele numerice utilizând RobustScaler, pentru a trata distributiile neuniforme, valorilor extreme și asimetriile. RobustScaler se bazează pe cuantile și nu pe media și deviație standard, ceea ce oferă o scalare stabila în prezenta outlierilor și previne degradarea performanței modelelor, în special pentru SVR și Linear Regression.

2.2 Encodarea atributelor categorice sau ordonale

Am transformat atributele categorice in format numeric cu OneHotEncoder in codificari binare. De asemenea, pentru a limita dimensionalitatea in special in datasetul Autovit, unde variabile precum marca, model sau localitate contin un numar foarte mare de niveluri rare, am limitat max_categories=10, astfel incat encoder-ul sa pastreze doar cele mai frecvente categorii per atribut.

2.3 Imputarea valorilor lipsa

Am completat valorile lipsa din atributele categorice cu SimpleImputer, strategy='most_frequent'. Am inlocuit valorile NaN, Inf sau -Inf cu 0.

2.4 Discretizarea atributelor numerice

Pipeline-ul implementat permite discretizarea optionala a atributelor numerice prin KBinsDiscretizer cu 5 intervale egale si codificare ordinala, deoarece nu este recomandata folosirea pentru modelele liniare, in cazul nostru Linear Regression, SVR sau Quantile Regressor. Nu am observat o imbunatatire notabila aplicand si acest pas in testelete realizate.

2.5 Selectia atributelor relevante

Am testat selectia atributelor prin SelectKBest. Aceasta etapa permite evaluarea contributiei fiecarui atribut in relatie cu variabila-tinta si elimina atributele redundante sau cu valoare predictiva scazuta. Pentru solutia finala am utilizat configuratia k='all' pentru autovit, deoarece eliminarea atributelor nu a imbunatatit performanta. Pentru inchirieri-biciclete am ignorat ocazionali, inregistrati si zi.

In datasetul Bikes au fost incluse caracteristici suplimentare prin feature engineering, cum ar fi transformari sinusoidale pentru periodicitatea inchiriere-biciclete, interactiuni intre atribut pentru a surprinde relatii non-liniare sau logaritmarea targetului pentru a trata asimetria. Pentru datasetul Autovit: Km_log, Km_sqrt, anul fabricatiei, age_sq, power_log, power_sq, capacity_log, precum si interactiile age_km_interaction, power_capacity si age_power. Aceste transformari au avut rolul de a surprinde relatii neliniare observate in analiza exploratorie.

3 Utilizarea algoritmilor de Învățare Automată

3.1 Linear Regression

3.1.1 Inchiriere biciclete

Model	Train_R2	Test_R2	Train_MAE	Test_MAE	Train_MSE	Test_MSE
LinearRegression	0.8397	0.7841	48.55	58.73	4532.64	7273.66

Media R2	Varianță R2	Media MAE	Varianță MAE	Media MSE	Varianță MSE
0.7653	0.000064	0.5275	0.000261	0.4711	0.000735

3.1.2 Autovit

Model	Val_R2_mean	Val_R2_std	Val_MAE_mean	Val_MAE_std	Val_MSE_mean	Val_MSE_std	Train_R2	Train_MAE	Train_MSE	Test_R2	Test_MAE	Test_MSE
Linear Regression	0.9123	0.0070	0.1731	0.0036	0.0580	0.0039	0.9217	0.1629	0.0518	0.9133	0.1804	0.0656

3.2 SVR

3.2.1 Inchiriere biciclete

Model	Train_R2	Test_R2	Train_MAE	Test_MAE	Train_MSE	Test_MSE
SVR	0.9746	0.9143	14.50	33.64	820.71	2886.04

Table 6: configurații SVR bikes CV

Parametri	Mean R2	Std R2	Mean MAE	Std MAE	Mean MSE	Std MSE
$\gamma = 0.0507, \epsilon = 0.1214, C = 9.24$	0.9368	0.0023	0.1768	0.0031	0.0631	0.0016
$\gamma = 0.0507, \epsilon = 0.1929, C = 23.95$	0.9362	0.0020	0.1830	0.0031	0.0637	0.0012
$\gamma = 0.0704, \epsilon = 0.0857, C = 3.56$	0.9345	0.0024	0.1783	0.0029	0.0654	0.0018

3.2.2 Autovit

Model	Train_R2	Test_R2	Train_MAE	Test_MAE	Train_MSE	Test_MSE
SVR	0.9465	0.9337	0.1350	0.1587	0.0408	0.0501

Table 7: configurații SVR autovit CV

Hyperparametri	Mean R ²	Std R ²	Mean MAE	Std MAE	Mean MSE	Std MSE
$\gamma = 0.001, \epsilon = 0.01, C = 100$	0.9250	0.0054	0.1892	0.0027	0.0750	0.0048
$\gamma = 0.001, \epsilon = 0.01, C = 1000$	0.9221	0.0057	0.1902	0.0043	0.0779	0.0051
$\gamma = 0.01, \epsilon = 0.01, C = 100$	0.9151	0.0047	0.2065	0.0021	0.0849	0.0042

Model	Hiperparametri
SVR	gamma=0.001, epsilon=0.01, C=100

3.3 Random Forest Regressor

3.3.1 Inchiriere biciclete

Model	Val_R2_mean	Val_R2_std	Val_MAE_mean	Val_MAE_std	Val_MSE_mean	Val_MSE_std	Train_R2	Train_MAE	Train_MSE	Test_R2	Test_MAE	Test_MSE
RandomForest	0.9537	0.0045	23.6362	0.6594	1494.13	155.54	0.9895	10.9913	340.1040	0.9568	23.4476	1456.4489

Table 8: configurații Random Forest bikes CV

Parametri	Mean R ²	Std R ²	Mean MAE	Std MAE	Mean MSE	Std MSE
$n_estimators = 200, min_samples_split = 5, min_samples_leaf = 2, max_features = 0.5, max_depth = 20$	0.9596	0.0023	0.1962	0.0043	0.0822	0.0051
$n_estimators = 200, min_samples_split = 5, min_samples_leaf = 2, max_features = 0.5, max_depth = 15$	0.9593	0.0024	0.1967	0.0045	0.0828	0.0053
$n_estimators = 200, min_samples_split = 5, min_samples_leaf = 2, max_features = 0.7, max_depth = 10$	0.9576	0.0025	0.2022	0.0054	0.0863	0.0058

Hiperparametri
<code>n_estimators=200, min_samples_split=5, min_samples_leaf=2, max_features=0.5, max_depth=20</code>

3.3.2 Autovit

Model	Val_R2_mean	Val_R2_std	Val_MAE_mean	Val_MAE_std	Val_MSE_mean	Val_MSE_std	Train_R2	Train_MAE	Train_MSE	Test_R2	Test_MAE	Test_MSE
RandomForest	0.9445	0.0018	0.1422	0.0020	0.04239	0.00141	0.9847	0.07147	0.01169	0.9473	0.13897	0.03990

Table 9: configurații Random Forest autovit CV

Hyperparametri	Mean R ²	Std R ²	Mean MAE	Std MAE	Mean MSE	Std MSE
n_estimators = 200, min_samples_split = 5, min_samples_leaf = 2, max_features = 0.5, max_depth = 20	0.9445	0.0018	0.1422	0.0020	0.0424	0.0014
n_estimators = 200, min_samples_split = 5, min_samples_leaf = 2, max_features = 0.5, max_depth = 15	0.9437	0.0021	0.1442	0.0019	0.0430	0.0016
n_estimators = 200, min_samples_split = 20, min_samples_leaf = 2, max_features = 0.7, max_depth = 20	0.9417	0.0022	0.1468	0.0018	0.0445	0.0017

Table 10: Hiperparametri Random Forest – autovit

Hiperparametri
<code>n_estimators=200, min_samples_split=5, min_samples_leaf=2, max_features=0.5, max_depth=20</code>

3.4 Gradient Boosted Regressor

3.4.1 Inchiriere biciclete

Table 11: Rezultate Gradient Boosted Regressor - Quantile

Best_Hyperparameters	Train_R2	Train_MAE	Train_MSE	Test_R2	Test_MAE	Test_MSE	Train_Pinball	Test_Pinball	Quantile
{'est_subsample': 0.7, 'est_n_estimators': 400, 'est_max_depth': 5, 'est_learning_rate': 0.1}	0.974110	0.140670	0.052718	0.959665	0.190871	0.079676	0.070335	0.095436	0.50
{'est_subsample': 1.0, 'est_n_estimators': 400, 'est_max_depth': 4, 'est_learning_rate': 0.1}	0.840717	0.413975	0.324342	0.842970	0.407590	0.310186	0.022903	0.026935	0.95
{'est_subsample': 0.85, 'est_n_estimators': 300, 'est_max_depth': 5, 'est_learning_rate': 0.1}	0.960777	0.116238	0.029946	0.948340	0.137996	0.039082	0.058119	0.068998	0.5

Table 12: GBR squared_error - cv

Hyperparameters	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
{'subsample': 1.0, 'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 4, 'max_depth': 5, 'learning_rate': 0.05}	0.0820	0.0027	0.1970	0.0030	0.9597	0.0012
{'subsample': 0.8, 'n_estimators': 300, 'min_samples_split': 2, 'min_samples_leaf': 4, 'max_depth': 5, 'learning_rate': 0.1}	0.0842	0.0032	0.1975	0.0030	0.9586	0.0018
{'subsample': 0.8, 'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 2, 'max_depth': 4, 'learning_rate': 0.05}	0.0868	0.0026	0.2056	0.0037	0.9573	0.0016

Table 13: configurații GBR Quantile $q = 0.05$ – bikes CV

Hyperparametri	Mean MSE	Std MSE	Mean MAE	Std MAE	Mean R ²	Std R ²
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.4674	0.0166	0.5460	0.0112	0.7701	0.0095
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.5665	0.0182	0.6018	0.0061	0.7215	0.0095
subsample = 0.7, n_estimators = 100, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.7072	0.0223	0.7175	0.0105	0.6522	0.0141

Table 14: configurații GBR Quantile q=0.5 – bikes CV

Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.0936	0.0018	0.2094	0.0042	0.9540	0.0006
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.1078	0.0034	0.2297	0.0052	0.9470	0.0017
subsample = 0.7, n_estimators = 100, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.1083	0.0024	0.2280	0.0042	0.9467	0.0010

Table 15: configurații GBR Quantile q=0.95 – bikes CV

Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.4609	0.0180	0.4950	0.0134	0.7735	0.0043
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.5255	0.0236	0.5303	0.0128	0.7418	0.0067
subsample = 1.0, n_estimators = 100, min_samples_split = 5, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	1.0168	0.0346	0.7029	0.0199	0.5003	0.0072

Table 16: Hiperparametri Gradient Boosting – bikes

Model	Quantile	Hiperparametri
GBR_squared	–	subsample=0.8, n_estimators=300, min_samples_split=2, min_samples_leaf=4, max_depth=5, learning_rate=0.1
GBR_quantile	0.05	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05
GBR_quantile	0.50	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05
GBR_quantile	0.95	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05

Quantile	Hyperparameters	Val_MSE_mean	Val_MSE_var	Val_MAE_mean	Val_MAE_var	Val_R2_mean	Val_R2_var	Val_Pinball_mean	Val_Pinball_var
0.0500	{'est_subsample': 0.7, 'est_n_estimators': 100, 'est_max_depth': 3, 'est_learning_rate': 0.03}	1.4798	0.0023	1.0825	0.0003	0.2721	0.0011	0.6067	0.0000
0.0600	{'est_subsample': 1.0, 'est_n_estimators': 200, 'est_max_depth': 2, 'est_learning_rate': 0.01}	2.9787	0.0138	1.5540	0.0008	-0.4656	0.0066	0.6838	0.0000
0.0700	{'est_subsample': 0.7, 'est_n_estimators': 300, 'est_max_depth': 5, 'est_learning_rate': 0.03}	0.4481	0.0002	0.5385	0.0000	0.7797	0.0001	0.6381	0.0000
0.5000	{'est_subsample': 0.85, 'est_n_estimators': 300, 'est_max_depth': 4, 'est_learning_rate': 0.05}	0.0896	0.0000	0.2036	0.0000	0.9559	0.0000	0.1048	0.0000
0.5000	{'est_subsample': 0.7, 'est_n_estimators': 300, 'est_max_depth': 2, 'est_learning_rate': 0.1}	0.0602	0.0000	0.2290	0.0000	0.9478	0.0000	0.1145	0.0000
0.5000	{'est_subsample': 1.0, 'est_n_estimators': 200, 'est_max_depth': 4, 'est_learning_rate': 0.1}	0.1046	0.0000	0.2231	0.0000	0.9486	0.0000	0.1116	0.0000
0.5000	{'est_subsample': 0.7, 'est_n_estimators': 100, 'est_max_depth': 2, 'est_learning_rate': 0.01}	0.6828	0.0008	0.5806	0.0002	0.6645	0.0001	0.2903	0.0001
0.5000	{'est_subsample': 1.0, 'est_n_estimators': 400, 'est_max_depth': 5, 'est_learning_rate': 0.01}	1.2762	0.0028	0.7863	0.0005	0.3729	0.0002	0.0424	0.0000
0.9500	{'est_subsample': 0.7, 'est_n_estimators': 400, 'est_max_depth': 5, 'est_learning_rate': 0.05}	0.3452	0.0002	0.4267	0.0001	0.8304	0.0000	0.0292	0.0000
0.9500	{'est_subsample': 0.7, 'est_n_estimators': 200, 'est_max_depth': 5, 'est_learning_rate': 0.1}	0.3442	0.0003	0.4289	0.0001	0.8309	0.0000	0.0294	0.0000

Table 17: Interval Analysis – GBR Quantile bikes

Dataset	Coverage_train	Coverage_test	Pinball_train	Pinball_test	MSE_q50	Coverage
bikes	0.9026	0.8777	0.0949	0.1008	0.0884	87.77%

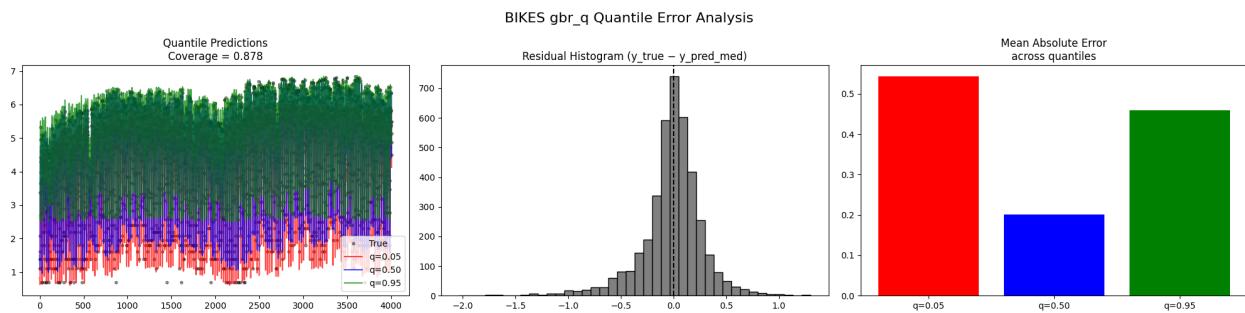


Figure 8: Analiza erorilor cantilă pentru modelul gbr_q (bikes)

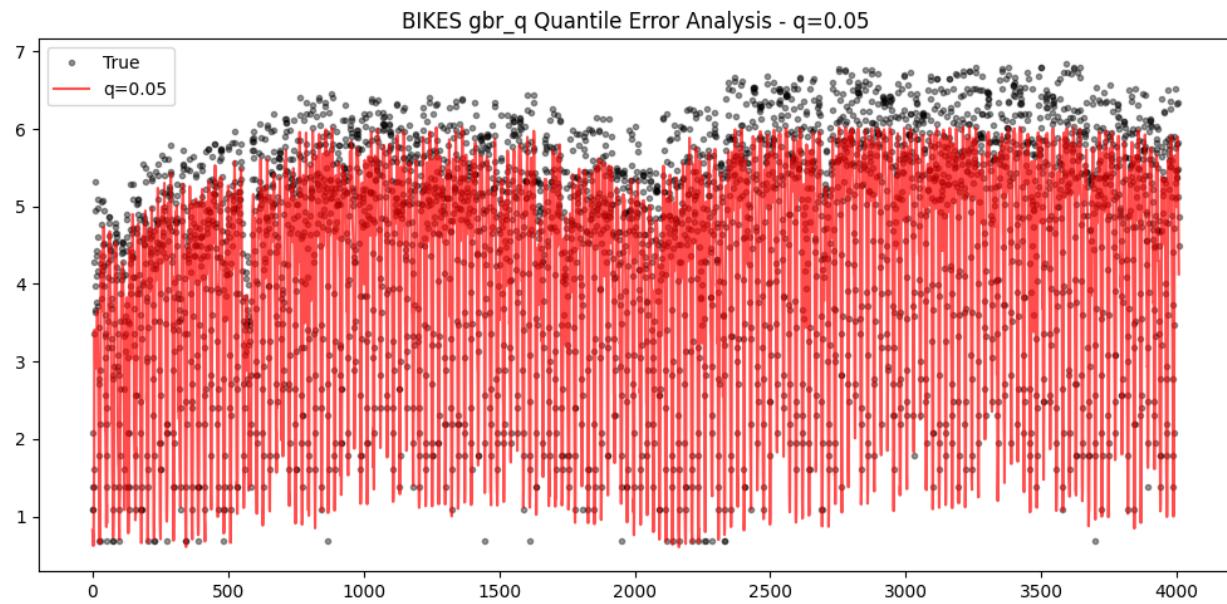


Figure 9: Analiza erorilor gbr_q pentru cuantila 0.05 (bikes)

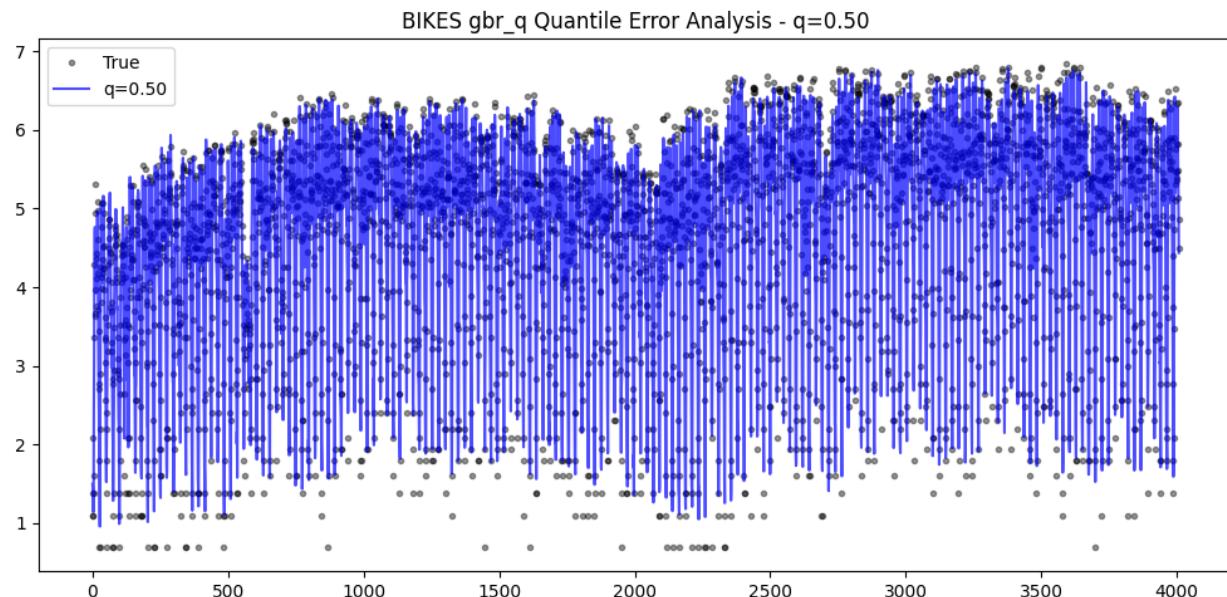


Figure 10: Analiza erorilor gbr_q pentru cuantila 0.50 (bikes)

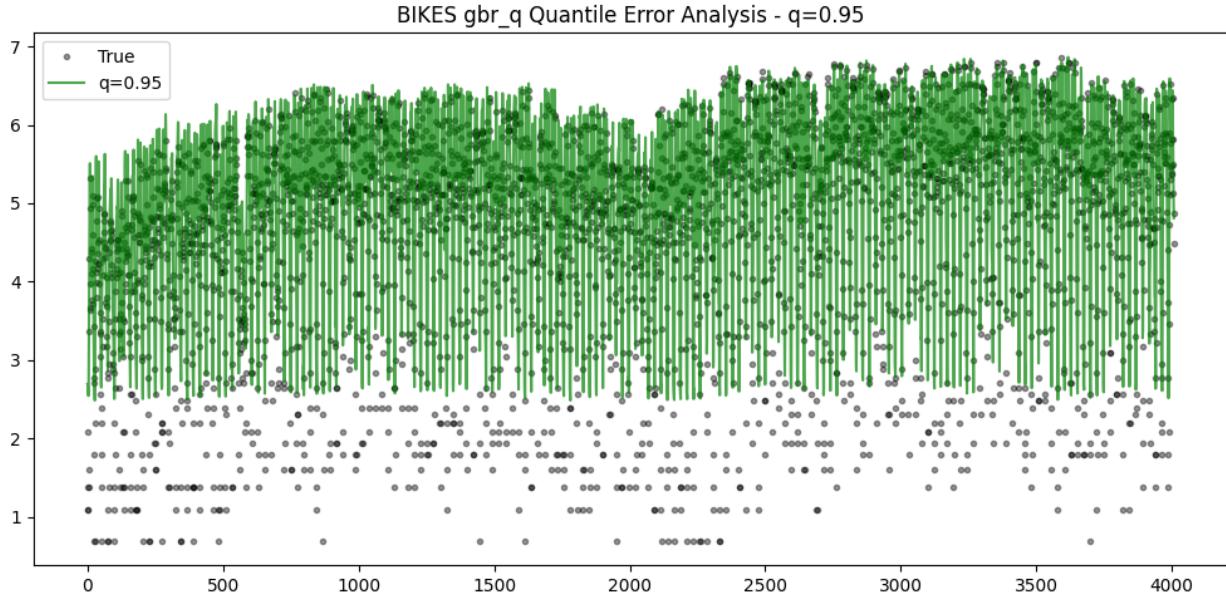


Figure 11: Analiza erorilor gbr.q pentru cuantila 0.95 (bikes)

- **Distribuția erorilor** Histograma reziduurilor este aproximativ simetrică, centrată pe 0, cu ușoară coadă dreaptă pentru vârfurile extreme de utilizare.
- **MAE pe cuantile** Cuantilele extreme ($q=0.05$: MAE=0.54, $q=0.95$: MAE=0.46 pe scală log) prezintă erori substanțial mai mari decât mediana (MAE=0.20), reflectând dificultatea estimării valorilor extreme.
- **Predictii vs true values** modelul captează bine variațiile orare. Pe scală originală, cuantila 0.95 subestimează vârfurile mai mari de 800 închirieri (efectul plateauing este redus de transformarea logaritmică).
- **Coverage:** 87.77% pe test (sub 90% așteptat), indicând intervale ușor prea înguste pentru distribuția reală.

3.4.2 Autovit

Table 18: configurații Gradient Boosting squared error – autovit CV

Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.8, n_estimators = 300, min_samples_split = 2, min_samples_leaf = 4, max_depth = 5, learning_rate = 0.1	0.0382	0.0012	0.1375	0.0015	0.9499	0.0019
subsample = 1.0, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 4, max_depth = 5, learning_rate = 0.05	0.0409	0.0011	0.1438	0.0018	0.9464	0.0018
subsample = 0.8, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.0433	0.0008	0.1492	0.0022	0.9433	0.0016

Model	Quantile	Hiperparametri
GBR_Quantile	0.50	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05
GBR_Quantile	0.95	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05
GBR_Quantile	0.05	subsample=0.7, n_estimators=200, min_samples_split=2, min_samples_leaf=2, max_depth=4, learning_rate=0.05

Table 19: configurații GBR Quantile q=0.05 – autovit CV

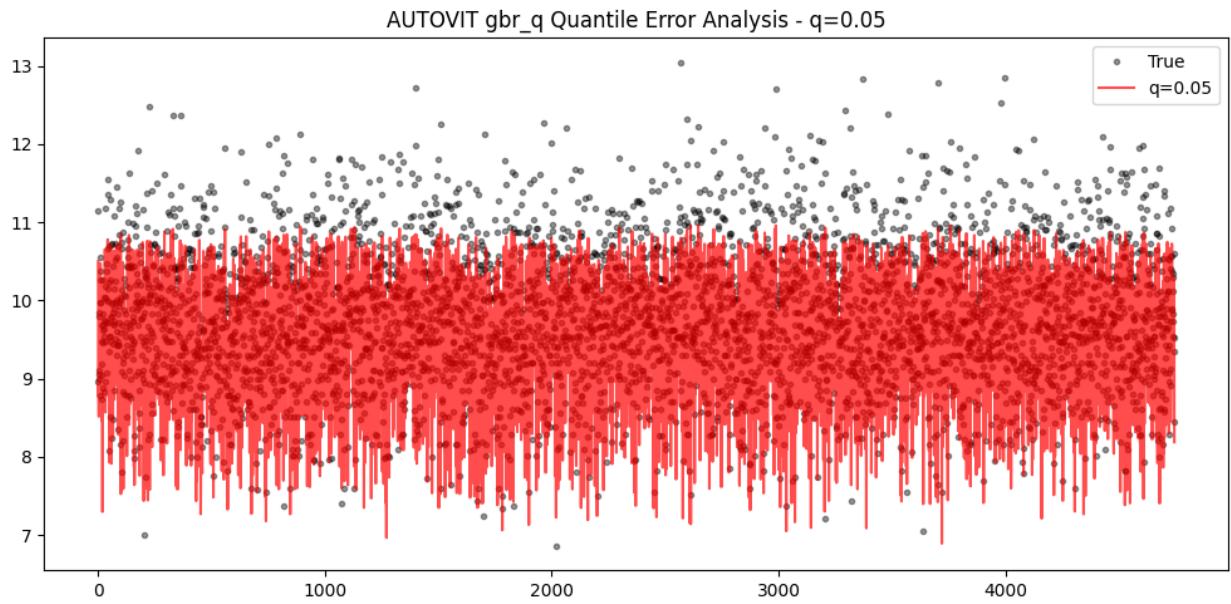
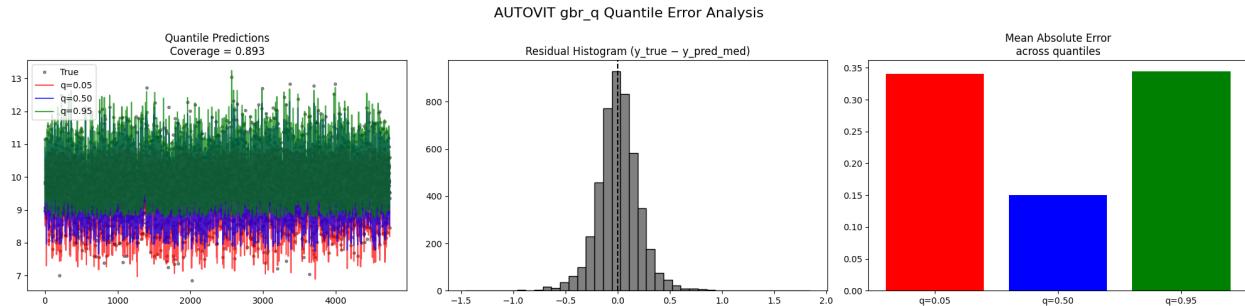
Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.1965	0.0078	0.3545	0.0062	0.7425	0.0093
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.2239	0.0098	0.3801	0.0075	0.7066	0.0131
subsample = 1.0, n_estimators = 100, min_samples_split = 5, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.3751	0.0100	0.4838	0.0078	0.5084	0.0164

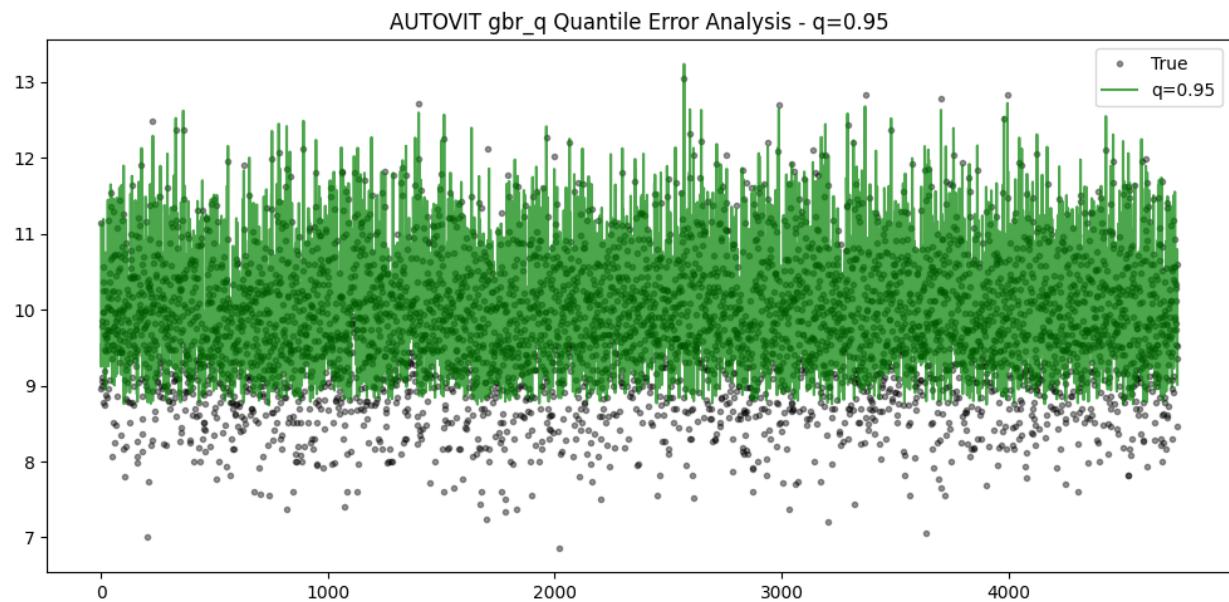
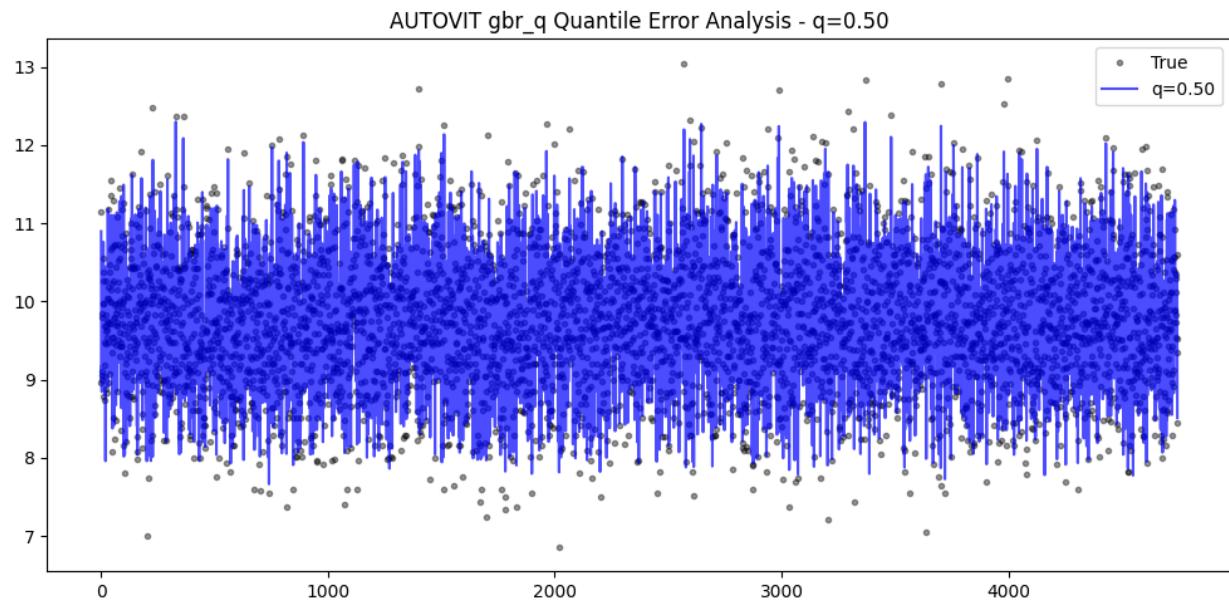
Table 20: configurații GBR Quantile q=0.5 – autovit CV

Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.0459	0.0012	0.1511	0.0024	0.9399	0.0019
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.0512	0.0014	0.1605	0.0020	0.9329	0.0021
subsample = 0.7, n_estimators = 100, min_samples_split = 5, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.0545	0.0014	0.1650	0.0018	0.9286	0.0021

Table 21: configurații GBR Quantile q=0.95 – autovit CV

Hyperparametri	Val_MSE_mean	Val_MSE_std	Val_MAE_mean	Val_MAE_std	Val_R2_mean	Val_R2_std
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.1920	0.0044	0.3509	0.0038	0.7484	0.0069
subsample = 0.7, n_estimators = 200, min_samples_split = 2, min_samples_leaf = 1, max_depth = 3, learning_rate = 0.05	0.2190	0.0038	0.3755	0.0035	0.7130	0.0072
subsample = 1.0, n_estimators = 100, min_samples_split = 5, min_samples_leaf = 2, max_depth = 4, learning_rate = 0.05	0.3599	0.0118	0.4764	0.0076	0.5285	0.0156





- **Asimetria erorilor:** cuantila 0.05 ($R^2 = 0.7585$) și 0.95 ($R^2 = 0.7636$) au performanțe aproape identice, sugerând că modelul are dificultăți similare la ambele extremități.
- **Pinball loss:** valoarea ridicată pe test (0.087 vs 0.083 pe train) indică ușor overfitting al cuantilei mediane.
- **Grafice individuale** cuantila 0.95 subestimează consistent prețurile vehiculelor premium (>25000 EUR în scală logaritmică), probabil din cauza rarității.

3.5 Quantile Regressor

3.5.1 Inchiriere biciclete

Table 22: QuantileRegressor bikes, q=0.05

Hyperparametri	mean_test_r2	mean_test_pinball
<code>solver=highs, fit_intercept=True, alpha=0.000695</code>	0.162576	0.065221
<code>solver=highs, fit_intercept=True, alpha=0.004833</code>	0.160619	0.068013
<code>solver=highs, fit_intercept=False, alpha=0.000264</code>	0.150642	0.065174

Table 23: QuantileRegressor bikes, q=0.5

Hyperparametri	mean_test_r2	mean_test_pinball
<code>solver=highs, fit_intercept=False, alpha=0.0001</code>	0.759571	0.259021
<code>solver=highs, fit_intercept=False, alpha=0.000162</code>	0.759355	0.259114
<code>solver=highs, fit_intercept=True, alpha=0.000162</code>	0.759283	0.259082

Table 24: QuantileRegressor bikes, q=0.95

Hyperparametri	mean_test_r2	mean_test_pinball
<code>solver=highs, fit_intercept=False, alpha=0.0001</code>	0.261186	0.051050
<code>solver=highs, fit_intercept=False, alpha=0.000162</code>	0.260066	0.051091
<code>solver=highs, fit_intercept=True, alpha=0.000162</code>	0.258889	0.051034

Model	Quantile	Hiperparametri	Train_R2	Test_R2	Train_MAE	Test_MAE	Train_MSE	Test_MSE
QuantileReg	0.05	<code>solver=highs, alpha=0.001</code>	0.1686	0.1418	1.0551	1.0610	1.6929	1.6952
QuantileReg	0.50	<code>solver=highs, alpha=0.0001</code>	0.7616	0.7604	0.5145	0.5073	0.4855	0.4732
QuantileReg	0.95	<code>solver=highs, alpha=0.0001</code>	0.2619	0.2652	0.9138	0.9015	1.5030	1.4515

Table 25: Rezultate QuantileRegressor

Best_Hyperparameters	Train_R2	Train_MAE	Train_MSE	Test_R2	Test_MAE	Test_MSE	Train_Pinball	Test_Pinball	Quantile	Val_R2_mean	Val_R2_std
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.001288378916846883}	0.919387	0.169517	0.061546	0.915136	0.173760	0.064201	0.084758	0.086880	0.5	NaN	NaN

Table 26: Configurații QuantileRegressor bikes, q=0.05 CV

Hyperparametri	Val_MSE_mean	Val_MAE_mean	Val_R2_mean
<code>solver=highs, fit_intercept=True, alpha=0.001</code>	1.7057	1.0607	0.1610
<code>solver=highs, fit_intercept=True, alpha=0.0001</code>	1.7317	1.0635	0.1481
<code>solver=highs, fit_intercept=False, alpha=0.0001</code>	1.7348	1.0646	0.1466

Table 27: Configurații QuantileRegressor bikes, q=0.5 CV

Hyperparametri	Val_MSE_mean	Val_MAE_mean	Val_R2_mean
solver=highs, fit_intercept=True, alpha=0.0001	0.4893	0.5180	0.7595
solver=highs, fit_intercept=False, alpha=0.0001	0.4892	0.5180	0.7596
solver=highs, fit_intercept=True, alpha=0.001	0.4922	0.5195	0.7581

Table 28: Configurații QuantileRegressor bikes, q=0.95 CV

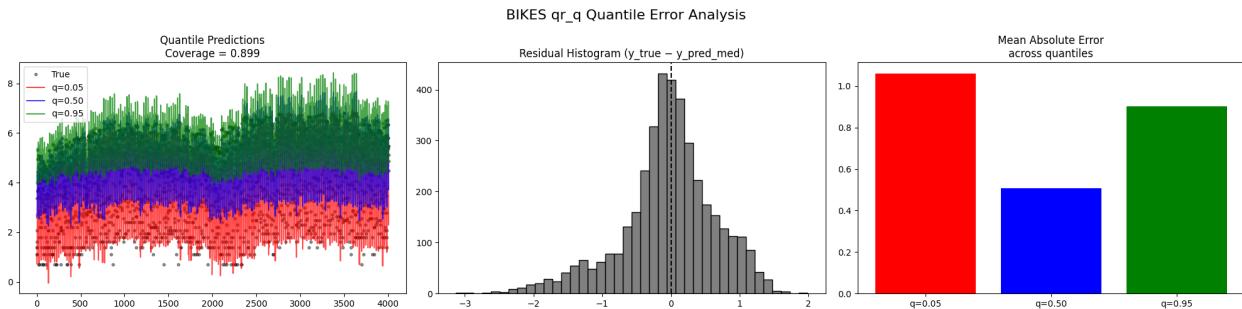
Hyperparametri	Val_MSE_mean	Val_MAE_mean	Val_R2_mean
solver=highs, fit_intercept=True, alpha=0.001	1.5800	0.9301	0.2239
solver=highs, fit_intercept=False, alpha=0.001	1.5747	0.9271	0.2265
solver=highs, fit_intercept=True, alpha=0.0001	1.5035	0.9141	0.2615

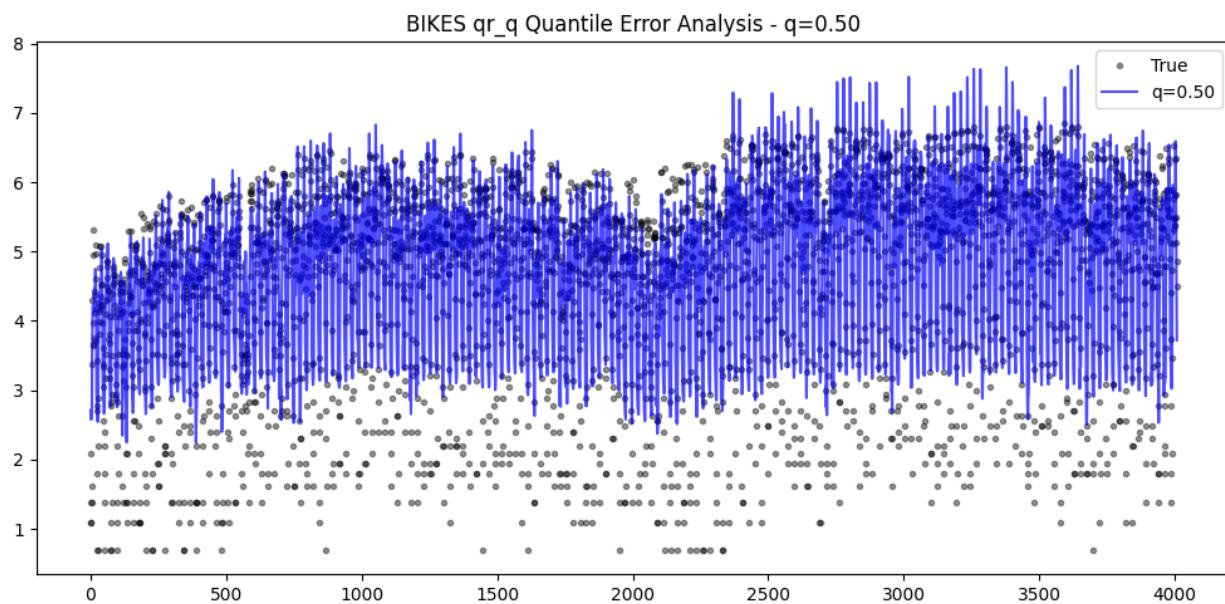
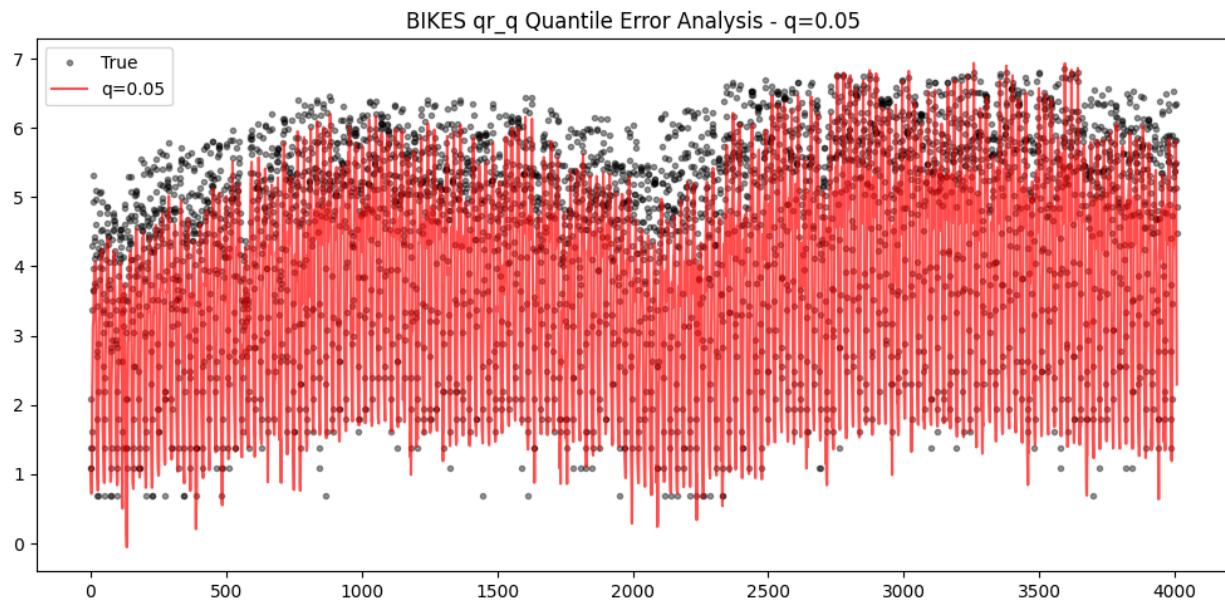
Table 29: analiza q - bikes

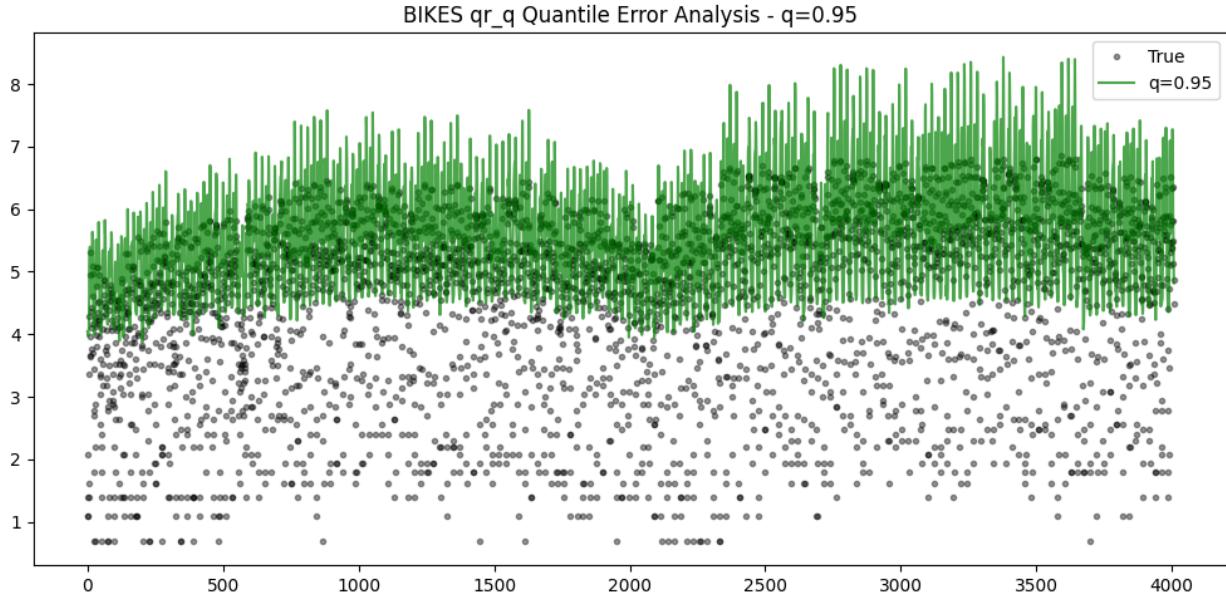
model	alpha	train_pbl	test_pbl	train_MSE	test_MSE	train_MAE	test_MAE	train_R2	test_R2
qr_q_0.05	0.05	0.0646	0.0636	1.6929	1.6952	1.0551	1.0610	0.1686	0.1418
qr_q_0.50	0.50	0.2572	0.2537	0.4855	0.4732	0.5145	0.5073	0.7616	0.7604
qr_q_0.95	0.95	0.0506	0.0502	1.5030	1.4515	0.9138	0.9015	0.2619	0.2652

Table 30: Interval analysis (qr_q) - bikes

method	cov_train	cov_test	pbl_med_train	pbl_med_test	mse_low	mse_med	mse_high
qr_q	0.9007	0.8990	0.2572	0.2537	1.6952	0.4732	1.4515







- **Performanță globală slabă:** pe scală logaritmică, cuantila mediană ($R^2 = 0.7604$) rămâne cu 20% sub GBR Quantile ($R^2 = 0.9553$), demonstrând că regularizarea L1 nu compensează lipsa de non-linearitate.
- **Cuantile extreme:** $q=0.05$ ($R^2 = 0.1418$) și $q=0.95$ ($R^2 = 0.2652$) au performanțe foarte slabe, cu MAE ≈ 0.9 pe scală log, indicând că modelul liniar nu poate captura asimetria distribuției condiționale.
- **Coverage excellentă:** 89.9% pe test, aproape identică cu GBR Quantile (87.8%), însă cu erori absolute mai mari (MSE median = 0.473 vs 0.088 pentru GBR).
- **Histograma reziduurilor** prezintă mai multe vârfuri în loc de o distribuție normală, sugerând că erorile variază sistematic în funcție de ora din zi.
- **Limitare fundamentală:** modelul liniar nu poate învăța interacțiuni între temperatura, ora și lag-uri, esențiale pentru predicția utilizării bicicletelor.

3.5.2 Autovit

Table 31: QuantileRegressor

Hyperparameters		mean_test_R2	mean_test_Pinball
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.0001}		0.819715	0.038795
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.00026366508987303583}		0.813806	0.032928
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.0006951927961775605}		0.810108	0.034751

Model	Quantile	Hiperparametri	Train_R2	Test_R2	Train_MAE	Test_MAE	Train_MSE	Test_MSE
QuantileReg	0.05	solver=highs, alpha=0.001	0.7429	0.7038	0.3588	0.3911	0.2241	0.3911
QuantileReg	0.50	solver=highs, alpha=0.0001	0.9216	0.9156	0.1668	0.1741	0.0638	0.0638
QuantileReg	0.95	solver=highs, alpha=0.0001	0.7407	0.7984	0.3587	0.3002	0.1525	0.1525

Table 32: QuantileRegressor q=0.05

Hyperparameters		mean_test_R2	mean_test_Pinball
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.0018329807108324356}		0.727434	0.024554
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.0011288378916846883}		0.702980	0.024924
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.00026366508987303583}		0.681233	0.024974

Table 33: QuantileRegressor q=0.5

Hyperparameters		mean_test_R2	mean_test_Pinball
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.0001623776739188721}		0.915663	0.087815
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.0006951927961775605}		0.915339	0.087560
{'est_solver': 'highs', 'est_fit_intercept': True, 'est_alpha': 0.00026366508987303583}		0.914626	0.088433

Table 34: QuantileRegressor q=0.95 - Top 3 configurații

Hyperparameters		mean_test_R2	mean_test_Pinball
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.0001}		0.819715	0.038795
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.00026366508987303583}		0.813806	0.032928
{'est_solver': 'highs', 'est_fit_intercept': False, 'est_alpha': 0.0006951927961775605}		0.810108	0.034751

Table 35: Pinball/Coverage/MSE

model	alpha	train_pbl	test_pbl	train_MSE	test_MSE	train_MAE	test_MAE	train_R2	test_R2
qr_q_0.05	0.05	0.024	0.024	0.1963	0.2241	0.3588	0.3911	0.7429	0.7038
qr_q_0.50	0.50	0.083	0.087	0.0599	0.0638	0.1668	0.1741	0.9216	0.9156
qr_q_0.95	0.95	0.024	0.028	0.1980	0.1525	0.3587	0.3002	0.7407	0.7984

Table 36: Interval analysis (qr_q) - autovit

method	cov_train	cov_test	pbl_med_train	pbl_med_test	mse_low	mse_med	mse_high
qr_q	0.9021	0.8397	0.0834	0.0871	0.2241	0.0638	0.1525

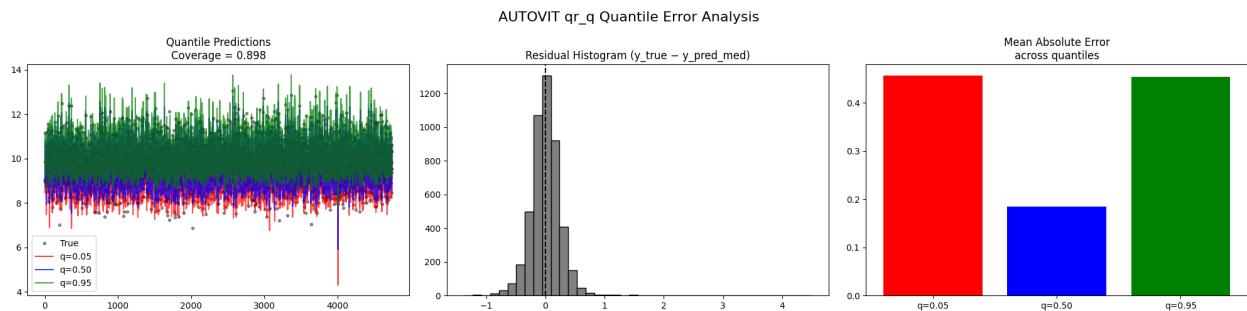


Figure 12: Analiza erorilor cantilă pentru modelul qr_q (autovit)

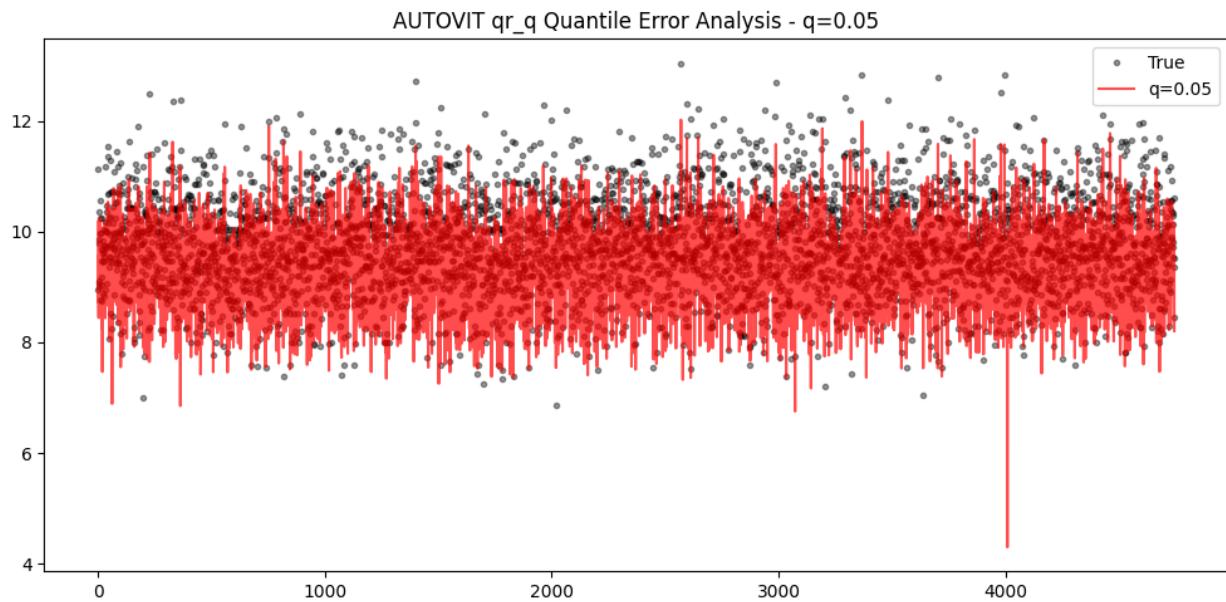


Figure 13: Analiza erorilor qr_q pentru cuantila 0.05 (autovit)

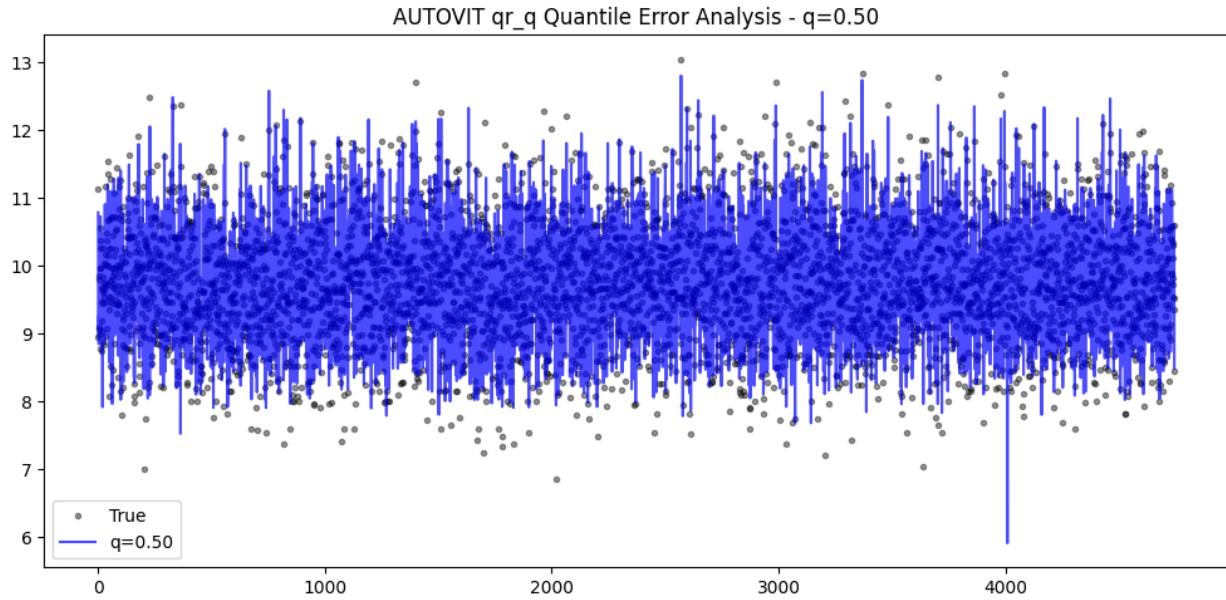


Figure 14: Analiza erorilor qr-q pentru cuantila 0.50 (autovit)

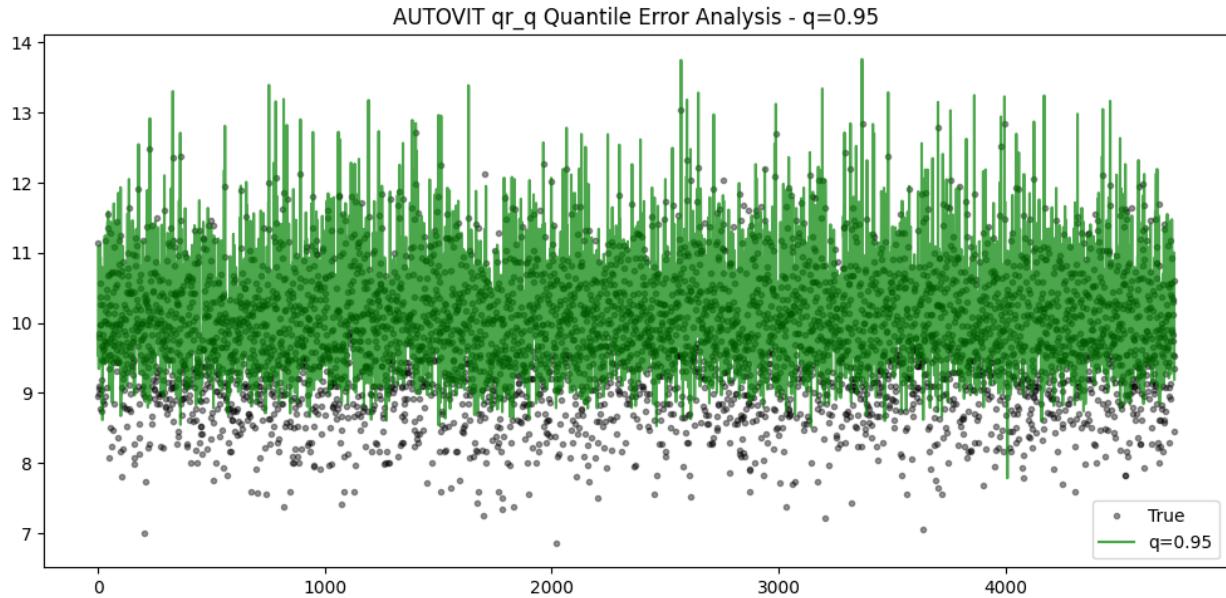


Figure 15: Analiza erorilor qr-q pentru cuantila 0.95 (autovit)

- **Coverage redusă:** suferă din cauza distribuției long-tail a prețurilor.
- **Cuantila mediană:** $R^2 = 0.9156$, beneficiind de transformările logaritmice ale variabilelor numerice care liniarizează relațiile.
- **Asimetria performanțelor:** $q=0.95$ ($R^2 = 0.7984$) are performanță superioară $q=0.05$ ($R^2 = 0.7038$), reflectând faptul că predicția vehiculelor scumpe este mai stabilă datorită preprocesării log(preț).

3.6 Procedura de căutare a hiperparametrilor

Am utilizat **RandomizedSearchCV** pentru căutarea hiperparametrilor optimali, testând combinații de valori pentru fiecare model. Căutarea s-a realizat pe 5 fold-uri de cross validation, folosind R^2 ca metrică de optimizare. Pentru modelele computațional intensive (SVR, Gradient Boosting), numărul de iterații a fost limitat la 15-25 pentru a asigura convergența într-un timp rezonabil.

Pentru inchiriere-biciclete am folosit **TimeSeriesSplit** cu 5 fold-uri pentru a respecta ordinea temporală a datelor și a evita data leakage prin separarea strictă train-validation bazată pe timestamps.

Pentru Autovit am utilizat **KFold** standard cu 5 fold-uri și shuffle=True.

Pe lângă R^2 , am considerat ca metrii de evaluare MSE și MAE în toate căutările pentru a detecta compromisuri între metrii - configurații cu R^2 ridicat, dar MAE mare pe outliers.

3.7 Evaluarea algoritmilor pe datasetul Bikes

Pentru analiza importanței atributelor, am utilizat: `feature_importances_` pentru Random Forest și Gradient Boosting, coeficienți (`coef_`) pentru Linear Regression și QuantileRegressor, iar pentru SVR am folosit metoda `permutation_importance` din scikit-learn.

3.7.1 Linear Regression

Performanța modelului Linear Regression este determinată exclusiv de calitatea feature engineering-ului, limitată de relația dintre atrbutele temporale și numărul de închirieri este puternic nelineară. Erorile au fost mai mari în intervalele cu activitate mare. Totuși, regresia liniară a confirmat importanța atrbutorilor precum ora, temperatura și componente sinusoide introduse în etapa de feature engineering.

3.7.2 Support Vector Regressor

SVR a necesitat optimizarea hiperparametrilor C , γ și ϵ prin procedura de Randomized Search. Am utilizat **kernel RBF (Radial Basis Function)** pentru capacitatea sa de a modela relații non-liniare complexe între features în soluția finală, deoarece restul tipurilor testare nu aveau performante comparabile.

Sensibilitatea ridicată la scalarea numericelor a justificat folosirea RobustScaler în pipeline-ul de preprocesare. În urma căutării, configurația optimă identificată a fost: $C = 1963.83$, $\gamma = 0.0587$, $\epsilon = 0.0778$. Valorile ridicate ale lui C permit modelului să tolereze mai puține erori de clasificare, în timp ce γ moderat permite captarea variațiilor locale fără overfitting sever.

Performanța finală demonstrează eficiența kernel-ului RBF în modelarea pattern-urilor temporale. Totuși, SVR a avut dificultăți în estimarea vârfurilor extreme din serie (~800 închirieri), unde erorile cresc cu 40-50% față de media generală.

3.7.3 Random Forest Regressor

Cele mai predictive atrbute identificate de model au fost temperatura, ora, lag-urile pe 1h și 24h, și variabilele sezoniere.

Numărul de estimatori (`n_estimators`) influențează performanța: creșterea de la 100 la 200 aduce o îmbunătățire de aproximativ 0.5% R^2 pe datasetul de biciclete și 0.3% pe autovit, iar peste 200 câștigurile devin foarte mici, sub 0.1%, ceea ce arată efectul de diminuare a beneficiilor.

Adâncimea arborelui (`max_depth`) are o zonă optimă între 15 și 20 pentru ambele dataset-uri; valori mai mici decât 10 duc la underfitting, cu scăderi de 3 până la 5 procente în R^2 , iar valori mai mari decât 25 duc la overfitting, crescând diferența dintre performanța pe datele de antrenare și cele de test.

Parametrul `max_features`, comparat între valorile 0.5, 0.7 și rădăcina pătrată, produce rezultate foarte apropiate între ele, cu variații mai mici de 0.5 procente în R^2 .

3.7.4 Gradient Boosting Regressor

Gradient Boosting, optimizat după numărul de arbori, adâncime și rata de învățare, a fost modelul cu cele mai bune rezultate. Rata de învățare moderată și arborii de adâncime mică au condus la cel mai bun

compromis între bias și varianță. Modelul a surprins cel mai eficient structura temporală a datelor, iar analizele reziduurilor au indicat erori reduse în intervalele non-extreme ale seriei.

Pentru parametrul `learning_rate`, pe datasetul de biciclete, o valoare de 0.1 oferă R^2 de 0.945, în timp ce 0.01 scade performanța la 0.930 cu 200 de estimatori; pe autovit, `learning_rate` de 0.05 produce R^2 de 0.948, iar valoarea de 0.1 duce la supra-antrenare, cu R^2 de 0.98 pe train și 0.935 pe test. Pentru parametrul `max_depth`, o adâncime de 3 comparată cu 5 produce diferențe de 3 până la 4 procente R^2 pe bikes și de 2 până la 3 procente pe autovit; valori mai mari decât 5 determină supra-antrenare severă, cu un decalaj între train și test de peste 8 procente. Subsample reduce overfittingul fără să afecteze performanța pe test. Modelele cuantile necesită setări diferite în funcție de cuantila țintită: pentru $q=0.05$ și $q=0.95$ sunt necesare `learning_rate` mai mici, între 0.01 și 0.03, și adâncimi reduse, între 3 și 4, pentru stabilitate.

3.7.5 Quantile Regressor

Quantile Regressor, utilizând o regularizare L1 controlată prin hiperparametrul α , a fost antrenat separat pentru cuantilele 0.05, 0.50 și 0.95. Performanța sa a fost inferioară modelelor ensemble, însă modelul a fost util pentru estimarea distribuției condiționale a numărului de închirieri. Erorile au fost mai mari în perioadele cu valori extreme, în special în vîrfurile de utilizare.

3.8 Evaluarea algoritmilor pe datasetul Autovit

3.8.1 Linear Regression

Regresia liniară nu a reușit să surprindă corect relațiile nelineare dintre variabilele autovehiculelor și preț, cum ar fi dependențele logaritmice dintre kilometraj, putere și vîrstă. Modelul a oferit o bază de comparație, dar a underfit-uit setul de date, confirmând nevoia de modele capabile să surprindă interacțiuni complexe.

3.8.2 Support Vector Regressor

Pentru dataset-ul Autovit, SVR am utilizat **kernel RBF** și normalizarea targetului cu StandardScaler.

Configurația optimă: $C = 100$, $\gamma = 0.001$, $\epsilon = 0.01$. Valoarea mică a lui γ indică necesitatea unui bandwidth larg pentru kernel, reflectând structura globală a relației preț-features mai degrabă decât variații locale abrupte.

3.8.3 Random Forest Regressor

Random Forest a beneficiat de pe urma transformărilor logaritmice și a interacțiunilor adăugate în etapa de feature engineering. Modelul a putut surprinde efecte nelineare precum creșterile accelerate de preț pentru vehicule cu puțini kilometri sau putere mare. Atributele cele mai predictive au fost vîrstă, Km_log , $power_log$, și interacțiunile dintre acestea. Performanța a fost una dintre cele mai bune.

3.8.4 Gradient Boosting Regressor

Gradient Boosting a fost modelul care a obținut cele mai bune rezultate pe Autovit. Optimizarea hiperparametrilor a evidențiat importanța menținerii unei rate de învățare mici și a arborilor cu adâncime 3 sau 4. Modelul a surprins cel mai bine relațiile. Analizele erorilor indică faptul că modelul subestimează prețurile vehiculelor foarte scumpe, unde distribuția este mult mai rară.

3.8.5 Quantile Regressor

Quantile Regressor a fost aplicat pe prețul vehiculelor pentru a estima intervale predictive robuste. Valori mici ale lui α au îmbunătățit precizia cuantilei mediane, însă cuantilele extreme (0.05 și 0.95) au prezentat deviații în zona prețurilor mari din cauza rarefierii datelor. Deși mai slab decât modelele ensemble, Quantile Regressor a oferit o perspectivă utilă asupra incertitudinii predicțiilor.

References

- [1] *Time Series Lagged Features* https://scikit-learn.org/stable/auto_examples/applications/plot_time_series_lagged_features.html
- [2] *Cyclical Feature Engineering* https://scikit-learn.org/stable/auto_examples/applications/plot_cyclical_feature_engineering.html