Project 2:

This is a continuation of the previous round. You may find the added sections highlighted.

Data type: time series

Number of features: 463 (the first column includes time)

Number of Samples: 397 (the first row includes feature IDs)

Target IDs for prediction: (3 different accuracies)

Good result: 542236, 67321

mid result: 549295

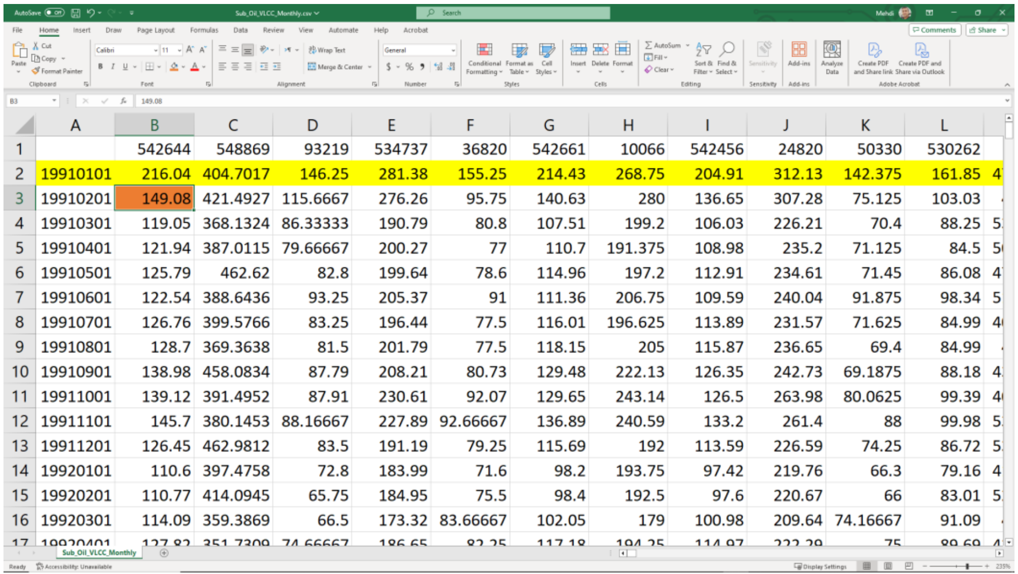
bad result: 41108, 541982

Dear all,

I am excited to present an engaging project involving predictive modeling in the maritime shipping industry using a comprehensive dataset from 1991 to the present day. The ultimate objective is forecasting future values after the last recorded month.

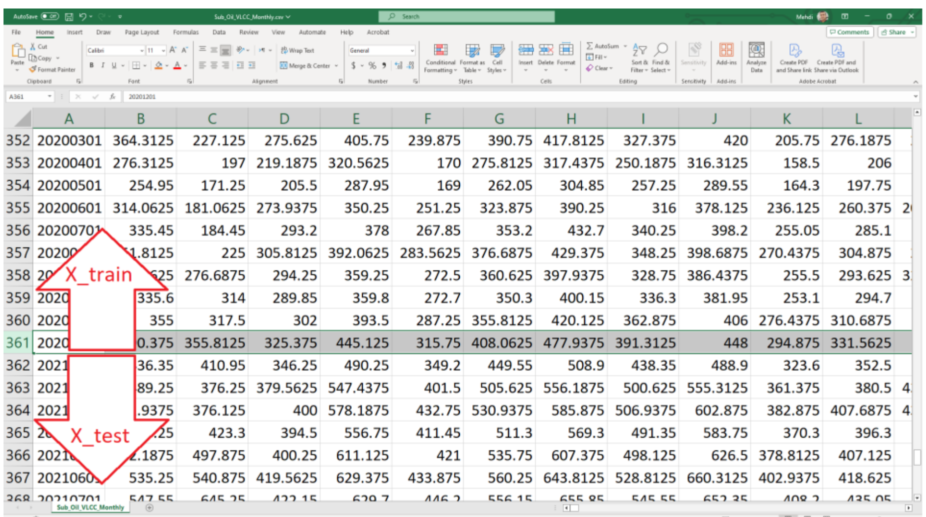
The dataset is a tabular time series with time in the first column. Each sample includes all features of a month as the input to the model (independent variables) and the intended value (in the specified column as a target) for the next month as the output of the model (dependent variable).

To generate X and y, we can consider the entire table as X and copy the target column in a vector as y. Remember that the label of sample t in X is in row t+1 in y.



	A	B	C	D	E	F	G	H	I	J	K	L
1		542644	548869	93219	534737	36820	542661	10066	542456	24820	50330	530262
2	19910101	216.04	404.7017	146.25	281.38	155.25	214.43	268.75	204.91	312.13	142.375	161.85
3	19910201	149.08	421.4927	115.6667	276.26	95.75	140.63	280	136.65	307.28	75.125	103.03
4	19910301	119.05	368.1324	86.33333	190.79	80.8	107.51	199.2	106.03	226.21	70.4	88.25
5	19910401	121.94	387.0115	79.66667	200.27	77	110.7	191.375	108.98	235.2	71.125	84.5
6	19910501	125.79	462.62	82.8	199.64	78.6	114.96	197.2	112.91	234.61	71.45	86.08
7	19910601	122.54	388.6436	93.25	205.37	91	111.36	206.75	109.59	240.04	91.875	98.34
8	19910701	126.76	399.5766	83.25	196.44	77.5	116.01	196.625	113.89	231.57	71.625	84.99
9	19910801	128.7	369.3638	81.5	201.79	77.5	118.15	205	115.87	236.65	69.4	84.99
10	19910901	138.98	458.0834	87.79	208.21	80.73	129.48	222.13	126.35	242.73	69.1875	88.18
11	19911001	139.12	391.4952	87.91	230.61	92.07	129.65	243.14	126.5	263.98	80.0625	99.39
12	19911101	145.7	380.1453	88.16667	227.89	92.66667	136.89	240.59	133.2	261.4	88	99.98
13	19911201	126.45	462.9812	83.5	191.19	79.25	115.69	192	113.59	226.59	74.25	86.72
14	19920101	110.6	397.4758	72.8	183.99	71.6	98.2	193.75	97.42	219.76	66.3	79.16
15	19920201	110.77	414.0945	65.75	184.95	75.5	98.4	192.5	97.6	220.67	66	83.01
16	19920301	114.09	359.3869	66.5	173.32	83.66667	102.05	179	100.98	209.64	74.16667	91.09
17	19920401	127.82	351.7309	74.66667	186.65	82.25	117.18	194.25	114.97	222.29	75	89.69

Creating X_train, y_train, X_test, and y_test from X and y is crucial. In time series, we must avoid data leakage, which means seeing a sample between test samples in the train set considering the order of the samples.



	A	B	C	D	E	F	G	H	I	J	K	L
352	20200301	364.3125	227.125	275.625	405.75	239.875	390.75	417.8125	327.375	420	205.75	276.1875
353	20200401	276.3125	197	219.1875	320.5625	170	275.8125	317.4375	250.1875	316.3125	158.5	206
354	20200501	254.95	171.25	205.5	287.95	169	262.05	304.85	257.25	289.55	164.3	197.75
355	20200601	314.0625	181.0625	273.9375	350.25	251.25	323.875	390.25	316	378.125	236.125	260.375
356	20200701	335.45	184.45	293.2	378	267.85	353.2	432.7	340.25	398.2	255.05	285.1
357	20200	1.8125	225	305.8125	392.0625	283.5625	376.6875	429.375	348.25	398.6875	270.4375	304.875
358	20	25	276.6875	294.25	359.25	272.5	360.625	397.9375	328.75	386.4375	255.5	293.625
359	2020	335.6	314	289.85	359.8	272.7	350.3	400.15	336.3	381.95	253.1	294.7
360	2020	355	317.5	302	393.5	287.25	355.8125	420.125	362.875	406	276.4375	310.6875
361	2020	0.375	355.8125	325.375	445.125	315.75	408.0625	477.9375	391.3125	448	294.875	331.5625
362	2021	36.35	410.95	346.25	490.25	349.2	449.55	508.9	438.35	488.9	323.6	352.5
363	2021	89.25	376.25	379.5625	547.4375	401.5	505.625	556.1875	500.625	555.3125	361.375	380.5
364	2021	.9375	376.125	400	578.1875	432.75	530.9375	585.875	506.9375	602.875	382.875	407.6875
365	20	25	423.3	394.5	556.75	411.45	511.3	569.3	491.35	583.75	370.3	396.3
366	2021	2.1875	497.875	400.25	611.125	421	535.75	607.375	498.125	626.5	378.8125	407.125
367	202106	535.25	540.875	419.5625	629.375	433.875	560.25	643.8125	528.8125	660.3125	402.9375	418.625
368	20210701	547.55	645.25	422.15	629.7	446.2	556.15	655.85	545.55	652.35	408.2	435.05

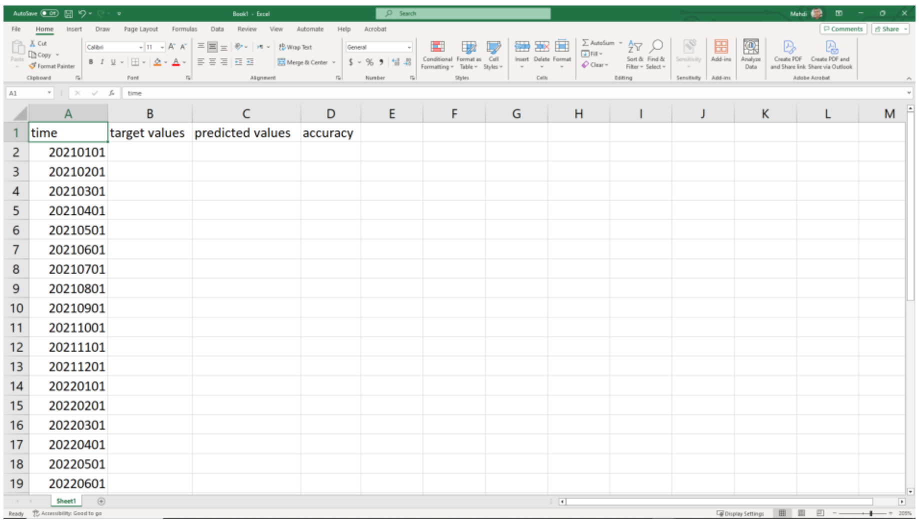
Due to the temporal nature of time series data, we aim to assess the model's accuracy over the last three years available in the dataset. To achieve this, consider training the model on all samples from the beginning up to n-36, and then test it on the last 36 samples (n is the total number of samples excluding the last one that doesn't include a label).

Your task is to train a regression model on the training set X_train and evaluate its performance on a designated test set X_test. It is highly recommended that you test your code by printing a sample input and output to the model to ensure true data preparation.

Finally, an Excel file comprising the data from the last 36 months will be created for reporting and analysis. The first column should represent time, the second column the true target values, the third column the predicted values, and the fourth column the calculated accuracy using the formula:

Accuracy = 100*(1-abs((actual-prediction)/actual))

Ultimately, the average accuracy across the 36 predictions will be calculated.



time	target values	predicted values	accuracy
20210101			
20210201			
20210301			
20210401			
20210501			
20210601			
20210701			
20210801			
20210901			
20211001			
20211101			
20211201			
20220101			
20220201			
20220301			
20220401			
20220501			
20220601			

This project offers a hands-on opportunity to apply regression modeling techniques to real-world data, emphasizing the challenges and nuances of forecasting in the dynamic maritime shipping industry. I encourage you to explore different regression models, fine-tune parameters, and critically evaluate the model's performance.

Please submit your code named your “group name” Version 1, an Excel file of the results, and a Word file of the report. I will add some tasks in the next rounds to the initial core I'm sharing now. You may modify or complete your code and update the results and the report every time.

There are some suggestions that may or may not improve the results. You can give them a shot to observe and report their results:

1. You can try other regression models to find the best match for the problem in hand. Here are some of them:

- Linear Regression
- Lasso Regression
- Ridge Regression
- Decision Tree Regression (DTR)
- Random Forest (RF)
- Support Vector Regression (SVR)
- XGBoost
- MLP

- What else?

2. Normalization is a technique for bringing all the input features to the same range
   - Normalization is best done after generating X and y and before separating them into training and test data.
   - Min-Max normalization: Min-max normalization is one of the most common ways to normalize data. For every feature, the minimum value of that feature gets transformed into a 0, the maximum value gets transformed into a 1, and every other value gets transformed into a decimal between 0 and 1.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

   - It is suggested to apply normalization only to X and not to y, because y is the output, and if we transform it, we need to reverse the transformation after prediction. Furthermore, if the y value is close to zero, the error rate would be huge, which may affect the accuracy.

3. In the preprocessing phase, each row signifies information from a specific month (lag 0), and the target value for the subsequent month is considered (lead 1). The lag parameter reflects the delay in input information, while the lead indicates the prediction period ahead.
   - You may want to try more lags as the input of each sample. For example, you can concat two rows of X to generate lag 1, and the output of this sample will be the target value for the upcoming month. For implementation, you must delete the two first rows of y and the two last rows of X to have the output in the same row as the input.

(you don't need to apply lag for this phase)

Finally, you must report everything you did. The best way to recap the results is by summarizing the results in a table:

Id	model	accuracy
1	Linear Regression	75%
2	Lasso Regression	88%
3	Lasso Regression + normalization	**91%**
4	Lasso Regression + normalization + lag 1	85%
…	…	…

Maybe some efforts can or cannot improve the accuracy. Please report all of them and finally bold the best result in the table.

Best of luck, and I look forward to your insightful analyses.

Regards,

Mehdi.