1. Problem & Data Introduction

This academic semester my university did not offer the Time Series Analysis course I was planning on taking, for this reason I've chosen to work with time series data. For this project I will be analyzing and attempting to forecast the next closing price of the Nasdaq E-Mini Futures. This data contains the trading volume, opening, high, low, and closing prices for every trading hour beginning in January of 2024 and ending in October of 2025; thus, containing over 10,000 samples.

2. What is Regression & How does it work?

Regression is simply a statistical technique in which we can model the relationship between a dependent variable and one or more independent variables. When only one independent variable is used in regression it is known as Simple Regression, and when used with more than one it is known as Multiple Regression. We can represent the dependent variable (y) as a linear combination of the dependent variables (x_i) plus some error term (ε_i) with β_0 being the y-intercept and the rest of the β_i being the slope of the hyperplane in the direction of x_i:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i + \varepsilon_i$$

Furthermore, to find the Betas β the Ordinary Least Squares method can be used which now involves matrix multiplication with the matrix X of size (number of observations, number of independent variables) and finding its Penrose-Pseudoinverse (more on that later):

$$\beta = (X^T X)^{-1} X^T y$$

3. Data-Understanding & Preprocessing

To gain a better understanding of the data, I begin by assessing if there are any missing values or gaps in time. In this case, there are several time gaps in the data due to non-trading days. Since these are non-trading days, we do not have to attempt to interpolate the missing timeframes as the price resumes the next trading day where it was halted on the previous trading day. Additionally, since this data is composed of the open, high, low, and closing prices it is expected that these features are highly correlated which can be seen in the following pairwise matrix.

	open	high	low	close	volume
open	1.000000	0.999811	0.999785	0.999688	-0.091214
high	0.999811	1.000000	0.999600	0.999808	-0.083198
low	0.999785	0.999600	1.000000	0.999808	-0.101532
close	0.999688	0.999808	0.999808	1.000000	-0.092779
volume	-0.091214	-0.083198	-0.101532	-0.092779	1.000000

4. Experiment 1: Linear Regression with OHLC prices

For the first model, I will be using an Ordinary Least Squares Linear Regression from Scikit-Learn. While not my first choice, it will be interesting to see how it fits the data given that OLS is not particularly good to model time series data, since it will be attempting to model the future state of a feature given that features current state, this is also known as Autocorrelation. Additionally, the features in this dataset are highly correlated as previously seen in the correlation matrix, this will cause the OLS to have much higher errors.

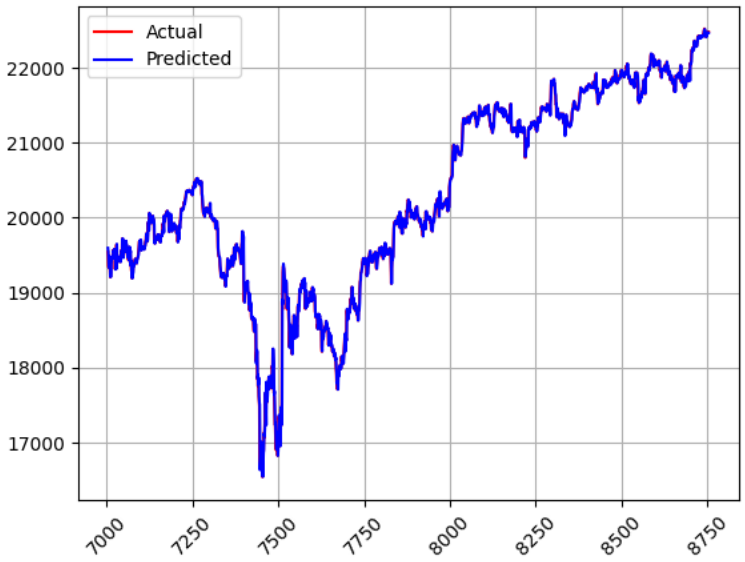
The first experiment begins by shifting the features to create a lag of t-1, t-2 and t-3, so that we can attempt to model the closing price at time (t). I also opted to use Scikit-Learn's TimeSeriesSplit to create 5 cross validation folds so that the model can train on past data and test on future data. For example, the model would use samples from index 0 to 100 to train and samples from index 101 to 150 for testing. This approach also accumulates samples for training until the last fold, this means that the last training set is a superset containing the previous training sets. Furthermore, to evaluate this model I will be using some basic statistical measures, including the Coefficient of Determination, Mean Absolute Error, and Adjusted Coefficient of Determination. The following plot shows the predictions on the test dataset of the linear model using only lagged features. As expected, the model follows the data almost exactly which seems to be due to the autocorrelation of the features within the chosen timeframe (1-hour).

MAE: 24.774627767525967

RMSE: 38.88753399113553

R_Squared: 0.99724442164693254

R_Squared_adj: 0.9972394841564507



5. Experiment 2: Adding Time Based Features

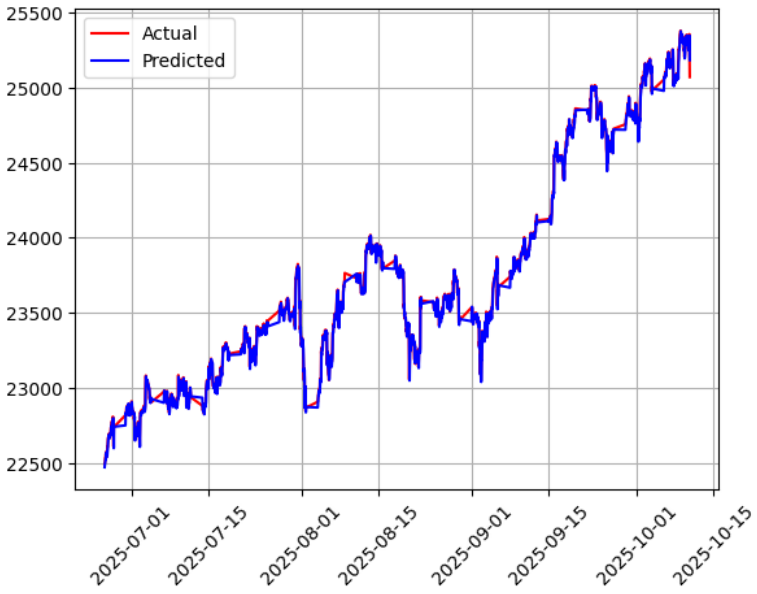
The second experiment consists of extracting time-based features from the current data. Fortunately, with Python and Pandas we can easily extract these features by converting the date string into a datetime object and assigning each component as a separate feature. However, this can create bias since dates happen in cycles this could imply that the model could bias towards days or months with higher values, therefore they must be transformed so that the differences are always the same. The common method for cyclical encoding is using a 2-dimensional transformation which takes the sin and cosine of the value, mapping it to a value on a unit circle. The following plot shows the predicted values on the test set, after applying the cyclical encoding to the original dataset. Looking at the evaluation metrics, the errors have increased slightly over the previous experiment with only lagged features. The Coefficient of Determination has decreased and the Adjusted R_Squared has also decreased suggesting that the added features are statistically insignificant.

MAE: 25.26043027665734

RMSE: 39.150033371473896

R_Squared: 0.9972068866147338

R_Squared_adj: 0.9971982411875411



6. Experiment 3: Seasonal and Trend Decomposition

The last experiment consisted of removing the Seasonal and Trend components from the time series. For this section I will be using the Statsmodels library with the STL (LOESS) model. Finally, to test for stationarity I will be using the Augmented Dickey-Fuller Test.

7. Impact

This project should not be used to make any financial decisions, rather it can be a means of formulating additional research questions. This project can be misleading by only looking at the Coefficient of Determination as it is over 99%, this implies that this simple model can "explain 99% of the variation" in the closing price. The high autocorrelation of the variables in the dataset makes any results based solely on these variables unreasonable and not applicable to real world scenarios.

8. Conclusion

This project is a learning experience for me as I have not yet taken an official Time Series Analysis course although I have interests in exploring this topic further. Although the results I achieved from this project do not seem promising there is still value in analyzing OHLC data such as this one. The transformations I performed did not improve the model as expected, I believe this kind of dataset should be used to complement more in-depth financial data and is not to be used solely.

Sources

Younes, E. (2025, October 10). *NASDAQ-CME-future-NQ*. Kaggle. https://www.kaggle.com/datasets/youneseloiarm/nasdaq-cme-future-nq/data