|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stage** | **Aims** | **Steps** | **Data** | **Notes** |
| Data preparation | * Identifying the extent of missing data in each time series * MCAR,MAR,MNAR test | a. What is the distribution of gaps in the missing data without Saturdays (may still include holidays)  b. Check for holidays  c. Count missing days not including saturdays  d. check for evidence of highly variable time gaps in your data, create a histogram of the time differences between adjacent data points | Original | - For the number of missing days, they are not likely to all be holidays (>100 per year) so reflect actually missing data. That should be considered as reducing the amount of relevant data points.  - However for 5 years of 2-minute interval data, it is unlikely that will have highly variable time gaps between data points. This is because the 2-minute interval is small enough to capture most of the variation in the data, and the regularity of the interval reduces the likelihood of gaps.  - if you do see evidence of highly variable time gaps in your data, it could indicate that there are issues with the data collection process. there may be issues with the equipment or software used to collect and store the data.  - For interval seasonal models like arima, equally spaced time interval data is required, so will need to impute for the missing time intervals |
|  | ii. Handle missing data (NaN values) by Deciding on an appropriate imputation method (e.g., forward fill, interpolation, or dropping missing data) | 1. Fill forward all Saturdays and holidays 2. Use separate imputation for short gaps and long gaps 3. Choose imputation based on result of MCAR MAR MNAR 4. Moving av – can introd bias 5. Exp smoothing – if trend or seasonal | Original to imputed | Assume that volatility is not too significant, but would use Kalman filter or ml methods(k means, rf) to be more robust to volatility  Assume Saturdays and holidays are price stable so ffil or bfil  All else impute appropriately.  Since don’t have market condition information for comparison with market movement over holidays, not use regression methods  iii. Choices to impute or remove.  a. Removing data points: If you remove these data points, you may lose valuable information, and the remaining data might not be representative of the original time series. This can lead to biased estimates and affect the accuracy of your analysis. However, if the missing data points are not crucial to your analysis or if they represent a small proportion of the total data, removing them might be a reasonable option.  b. Imputing data points: Imputing the missing data points can help preserve the structure of the time series and maintain the continuity of the data. However, the accuracy of the imputed values depends on the imputation method used and the nature of the missing data. Some common imputation methods include linear interpolation, moving average, and more advanced techniques like state-space models with Kalman filtering.  Including non trading days since models is not focussing on short term price movements where non trading days would not provide much information for intra day prediction  Not using other features like technical idicators which are not considering non trading days, so removing non trading days would reduce the amount of features/data available  Model not considering any unique evnts that would exclude non trading days |
| Data exploration | Seasonality | Perform seasonality and trend analysis | Lin Imputed | Using imputed data so that it is easier to observe seasonality or cyclicity |
|  | Statistical analysis  Perform exploratory statistical analysis on the data. Data is likely distributed non parametrically so quartile tests for outliers will not be very effective. Results from EDA will hopefully inform how to deal with outliers/noise missing values. High frequency data. | - Transform log return  a. Visualize the time series data (ts1 and ts2) to understand their behavior over time  b. Calculate summary statistics (mean, median, standard deviation, etc.) for each time series c. Investigate the correlation between ts1 and ts2  Calculate summary statistics (mean, median, standard deviation, skewness and kurtosis, coefficient of variation.) for each time series . Assess the presence of volatility clustering or heteroskedasticity in the data  Testing for stationarity using statistical tests | log returned orignal | Use unimputed data so that do not have any bias introduced  By analyzing the raw data, you can get a more accurate representation of the underlying distribution and identify any potential outliers or anomalies that might be due to data errors or other issues. |
|  | Outlier analysis  Point outliers: These are individual data points that are significantly different from the majority of the data. In the context of time series, point outliers can be caused by sudden, unexpected events, such as market crashes, earnings surprises, or geopolitical events.  Collective outliers: These are groups of data points that exhibit unusual behaviour when considered collectively, even if each individual data point may not be an outlier. In time series, collective outliers can be caused by structural breaks, regime shifts, or periods of increased volatility. | - Tukey's fences:  Peaks Over Threshold (POT) method  Hampel Identifier  Visualise outliers vs non outliers data, inspect extreme values | Log returned original | Rare events: Outliers can be caused by rare events that are not part of the typical behavior of the data. While these events may be important to consider in some analyses, they can also distort the results of models that assume a more stable underlying process.  Explain why keeping outliers  model is intended to take extreme market events into consideration. Keeping the outliers in this case is a reasonable decision, as it allows your model to account for the full range of market conditions, including extreme events like the 2008 market crash.  do not need to clean the outliers but include them in your analysis. Since you're dealing with a heavy-tailed distribution, extreme events are an inherent part of the data, and it's essential to model them accurately. |
|  | Identity and classify noise | Check if noise is due to measurement errors or inherently part of the data  If the noise is consistent across the entire dataset and does not exhibit any unusual patterns or extreme values, it's more likely to be inherent noise.  If the noise appears to be localized to specific data points or time periods, it might be due to measurement errors or other issues. | Log returned original | differentiate between noise that is an inherent part of the data-generating process and noise that is due to measurement errors or other issues. If the noise is an inherent part of the data, you should include it in your analysis and choose a model that can handle noisy data, such as state space models or robust statistical methods. On the other hand, if the noise is due to measurement errors or other issues, you should consider cleaning or adjusting the data before training your model.  based on summary statistic and effect on stationarity or significantly more outliers introduced. Minimise changing these significantly, but acknowledge that would be likely underlying in the data if true values were present |
| (Optional Sensitivity analysis) |  | Compare the results of your EDA using both the original data with missing values and the imputed data. This will help you understand the impact of the imputation on your analysis and identify potential biases introduced by the imputation method.  Use different imputation for each series.  Evaluate imputation method based on summary stats and effect on stationarity or outliers introduced. Minimise changing these significantly, but acknowledge that would be likely underlying in the data if true values were present |  | Ts1 simple linear interpolation works ost likely because missing values are sread throughout series  Ts2 needs to be handled more carefully since lin doesn’t work as missing values may be more clustered  Checks:   * Visual * Preserve summary stats * Evaluate method using test splits |
| Pre-processing | Cleaning | Apply any smoothing or resampling or outlier removal |  | To reduce noise: If your data has a high frequency and contains a lot of noise, you might want to downsample it to a lower frequency to smooth out short-term fluctuations and focus on the underlying trend or seasonality.  To improve computational efficiency: Working with high-frequency data can be computationally expensive, especially when fitting complex models. Resampling to a lower frequency can help reduce the computational burden and speed up the analysis.  To align with the forecasting horizon: If your goal is to forecast at a specific time horizon (e.g., daily, weekly, or monthly), you may need to resample your data to match that frequency. |
| modelling | Univariate | Fit ARIMA best fit  Plot reisduals, acf,pcf and qq prob of residuals  Plot squared residuals, acf,pcf,qq prob  Fit GARCH |  | The ARCH (Autoregressive Conditional Heteroskedasticity) LM test is typically applied to the residuals of a model fitted on the original returns, not the squared returns or the original returns themselves. The main purpose of the ARCH LM test is to determine if there is any remaining ARCH effect (i.e., time-varying volatility) in the residuals of the fitted model.  To perform the ARCH LM test:  Fit a model (such as an autoregressive model) to the original returns of your time series data.  Obtain the residuals from the fitted model.  Square the residuals.  Apply the LM test to the squared residuals to check for autocorrelation, which would indicate the presence of ARCH effects. |
|  |  | Split into training and holdout validation  Training can use rolling window fo all models  Ml models use lagged features  Evaluation is between the training data and the prediction on the training data |  | 1. Start with 5 years of data and create a holdout test set of the last 30 days. This set will be used only for final evaluation. 2. Preprocess the remaining data (5 years minus the 30 days) by creating lagged features to capture the temporal dependencies in the data. 3. Use the preprocessed data (with lagged features) for training, evaluation, cross-validation, and optimization of your model. a. Perform a rolling window approach for training and evaluation. b. Use cross-validation techniques, such as time series cross-validation, to fine-tune your model's hyperparameters. c. Optimize your model based on the evaluation metrics. 4. Analyze the residuals of the models (e.g., check for autocorrelation, normality, and homoscedasticity) and perform diagnostic tests. 5. Select the best model based on a combination of performance metrics, robustness, and model diagnostics. 6. Preprocess the entire non-holdout dataset (5 years minus the 30 days) by creating lagged features. Then, train the selected model on this preprocessed dataset. 7. Preprocess the holdout set (the last 30 days) by creating lagged features using the same approach as before. Finally, test the model on this preprocessed holdout set to see how well it performs on unseen data. |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |