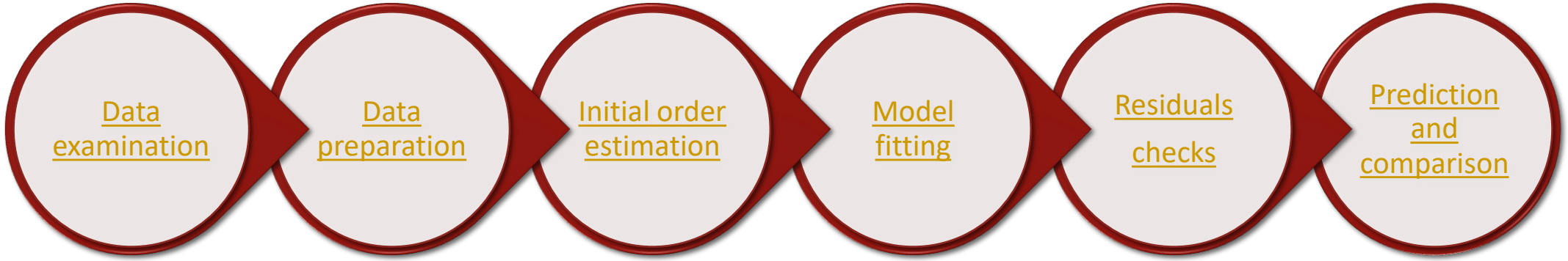
Introduction to Time Series

THE UNIVERSITY OF CHICAGO - MASTERS IN ANALYTICS

ARIMA MODELING PROCEDURE

© NECO, 2021

Model identification procedure



- Exploratory Data Analysis
- Transformation
- Train/test split
- Differencing
- ACF
- PACF
- Parameter estimation
- Rely on R or python
- AIC/BIC
- Plot
- Histogram
- ACF
- Ljung-Box
- MAE/RMSE
- MAPE/sMAPE
- Only step that uses Test Set

© NECO, 2021

Data Examination

- The purpose of this step is to examine the data, for a better understanding. This will help us in building more accurate models and in delivering better insights.
- Qualitative
  - Histograms, box plots, line plots, correlation plots
- Quantitative
  - Check NA's, summary statistics, correlation coefficients
- Verbal summary
  - Verbalize what you found, the importance of it. Any flaws in the data or assumptions that you are making.

© NECO, 2021

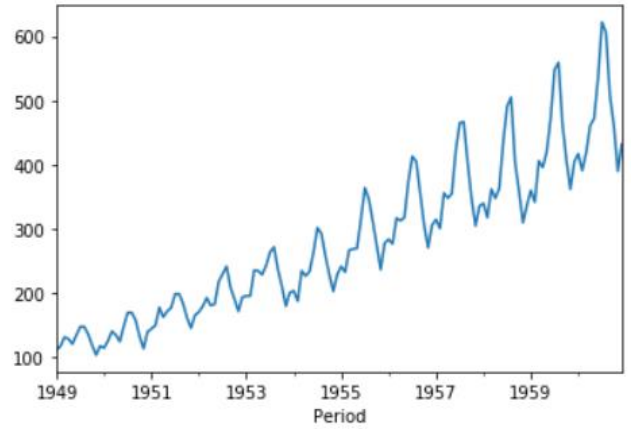
Data Preparation

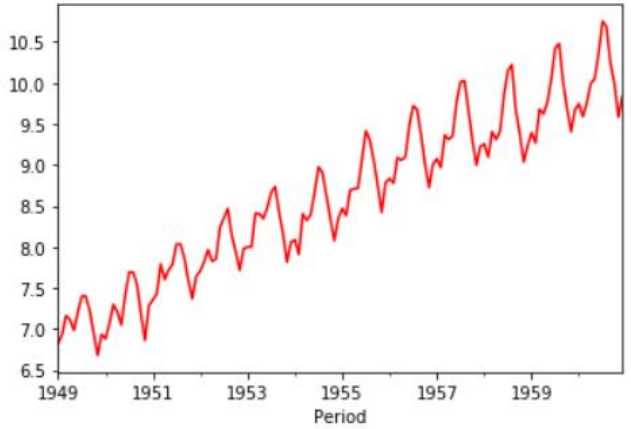
- The purpose of this step is to prepare the data, so that it fits the assumptions of ARIMA modeling. Our data must become stationary
- First, if the variance is not constant, we must use a **log transformation**
- Next, split into train and test. We must use a temporal train test split. I prefer to do the split at this point in the process:
  - Both train and test are already in log transformed units
  - When we examine, difference and fit the best ARIMA model, we do so with the training data only
  - We do not want to let our test set influence any of our modeling decisions
- Finally, if there is a trend or if ADF/KPSS tests tell us it is not stationary, we must take a **difference**

© NECO, 2021

Log Transformation part 1

- We can use BoxCox transformation to control the variance and bring the data closer to a normal distribution
- In the blue plot (original) we can see how the variance changes as the y value increases
- In the red plot (log transformed) we can see how the variance is steady throughout the time series
- See Week 2: 0.BoxCox Transformation.ipynb

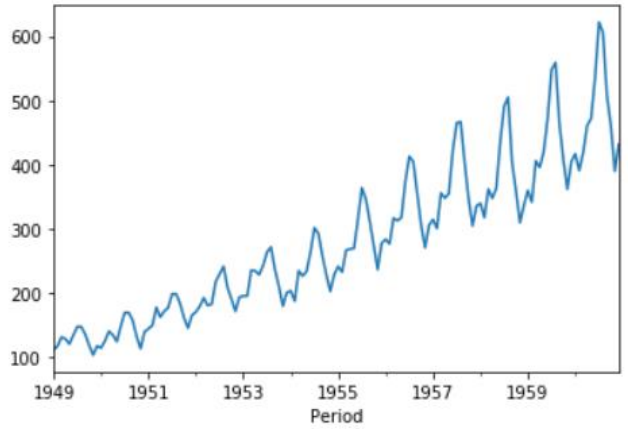


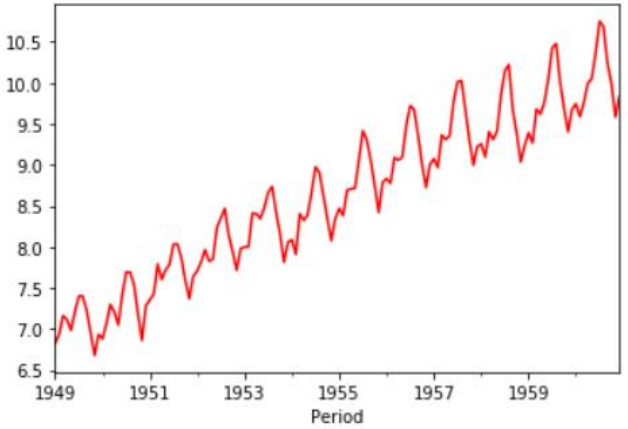


© NECO, 2021

Log Transformation part 2

- **We must use caution when using log transformation**
- When we hand over our final results, we must transform the data back into it's original units. ARIMA will **NOT** handle this transformation, so we must do this manually.
- Second, when comparing error metrics (sMAPE) across different models (ARIMA vs prophet) we must make sure that we are calculating sMAPE on the same units for both models.
- **Be careful not to compare sMAPE on log transformed data in ARIMA against original data in prophet.**





© NECO, 2021

Differencing

- If we see seasonality in the plot, we must take the seasonal difference first. If not, proceed to the stationarity testing
  - Be sure to drop the na values
  - Seasonal_diff_timeseries = timeseries.diff(12).dropna
- We can use the ADF and KPSS tests to determine if the training set is stationary
  - ADF H0: Non-stationary – we want to see a small p-value to reject this
  - KPSS H0: Stationary – we want to see a large p-value to “accept” this
- If our time series is non stationary, we should take a one period difference. Then check ADF/KPSS tests again. We can do another differencing if required to achieve stationarity. Be sure to drop na values

© NECO, 2021

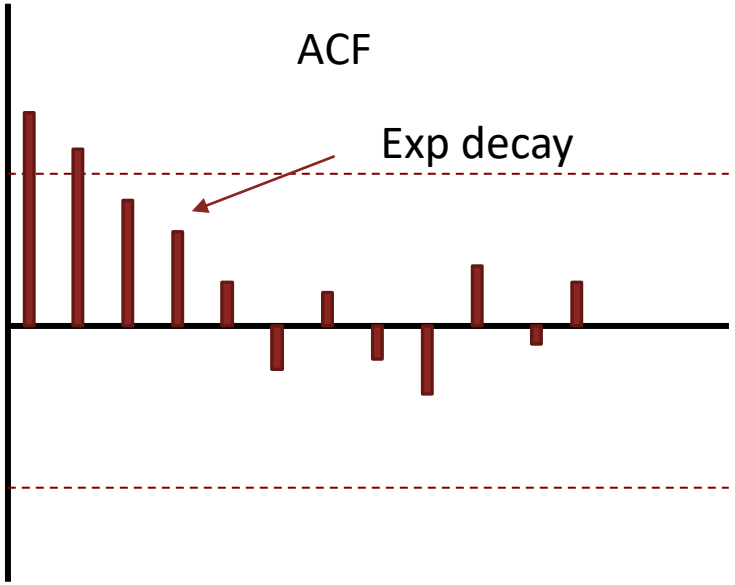
Initial order estimation

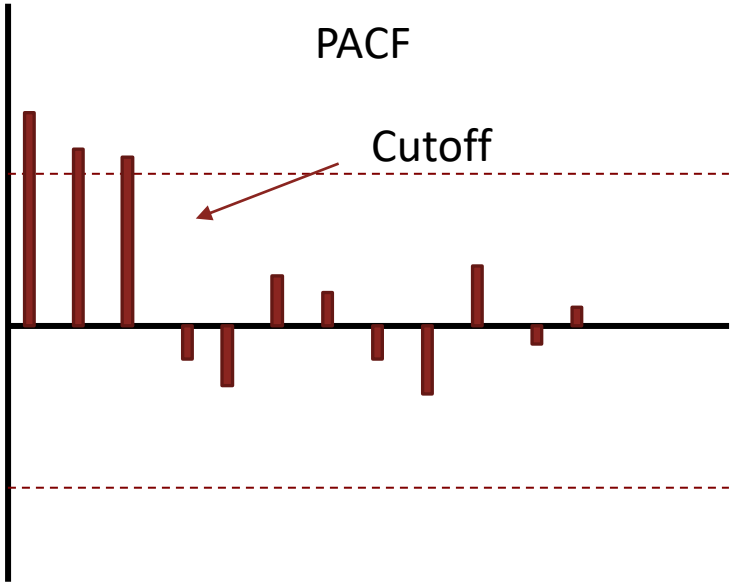
- Now that we have our stationary training dataset, we can use this to determine the best order for our ARIMA model.
- This part is a mix of “art and science”, be prepared for no straight answers
- Also, you will need to try multiple orders in order to find the best ARIMA fit. One and done does not work

© NECO, 2021

ACF and PACF patterns

- Combination of patterns can ONLY tell us about a pure AR(p) or MA(q) process, NOT an ARMA(p,q) process
- ARIMA(p,d,0) if ACF is exp decaying or sinusoidal; there is a significant lag in PACF but none after p
- ARIMA(0,d,q) if PACF is exponentially decaying or sinusoidal; significant spike at lag q in ACF, but none after q





Estimated order
(3,0)

© NECO, 2021

EACF patterns

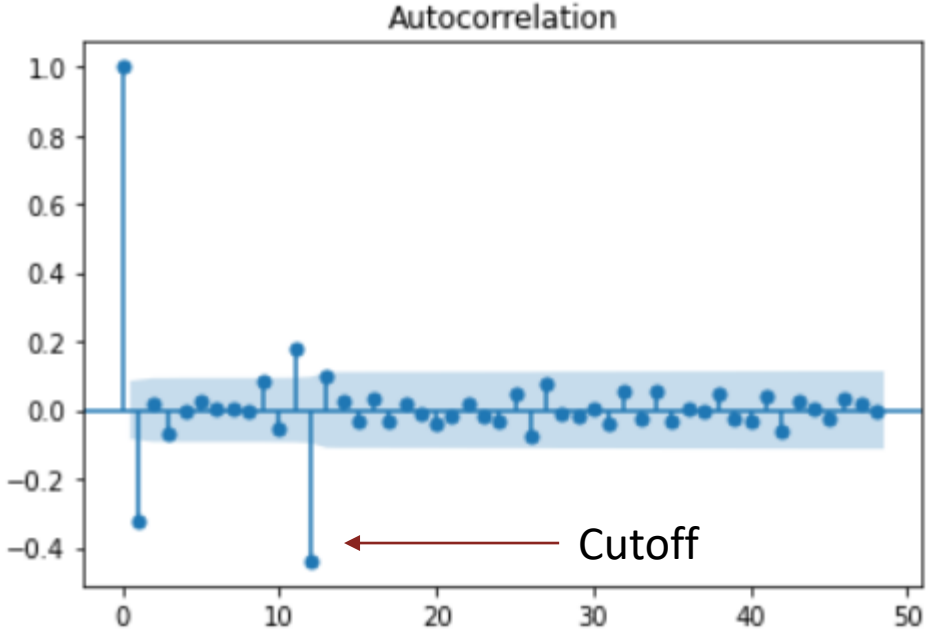
- ACF/PACF patterns can only tell us about a pure AR(p) or a pure MA(q) process
- EACF is used when we have a mixed ARMA(p,q) process
- EACF is not available in python, only in R
- All three are used to give us an initial order estimate. This may or may not be the best fitting model.
- It should **definitely not** be the only model we fit

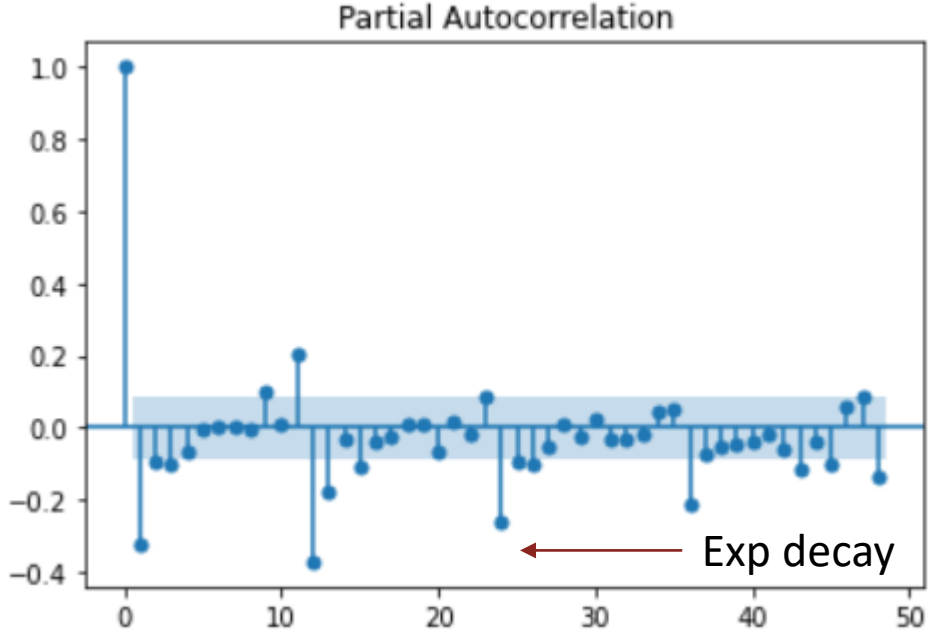
AR/MA	0	1	2	3	4	5	6	7	8	9	10
0	x	x	x	x	x	x	x	x	x	x	x
1	x	0	0	0	0	0	0	0	0	0	0
2	x	x	0	0	0	0	0	0	0	0	0
3	x	x	x	0	0	0	0	0	0	0	0
4	x	x	x	x	0	0	0	0	0	0	0
5	x	x	x	x	x	0	0	0	0	0	0
6	x	x	x	x	x	x	0	0	0	0	0
7	x	x	x	x	x	x	x	0	0	0	0

© NECO, 2021

Seasonal ACF/PACF patterns

- Here we look at the seasonal patterns.
- In ACF, we see significant lag at 12, with an abrupt cutoff, while in the PACF we see exp decay in lags 12, 24, 36, 48, etc.





Estimated seasonal order (0,1)

Estimated non-seasonal order (0,1)

© NECO, 2021

Model fitting

- Once we have determined an order, we can use our training data to fit our parameters for our first model.
- We are trying to find the coefficients that solve this equation for our training dataset

$$\underset{\uparrow}{(1-\phi_1 B-\cdots-\phi_p B^p)}\quad \underset{\uparrow}{(1-B)^d y_t}\quad =\quad c+\underset{\uparrow}{(1+\theta_1 B+\cdots+\theta_q B^q)}\varepsilon_t$$

$$\text{AR}(p) \qquad d \text{ differences} \qquad \text{MA}(q)$$

- The ARIMA function can handle the differencing for us. **Use your original data** (transformed if necessary) that has no differencing. Instead feed the ARIMA function the appropriate orders for your non-seasonal and seasonal (if necessary) orders.

© NECO, 2021

Information criterion

- Once we fit one ARIMA model, our job is not done.
- Alter the p's and q's in your order to fit alternative ARIMA models.
- Use AICc, AIC and BIC to determine which model has the best fit.
- If you tried different orders of differencing, information criteria is not valid across different orders. This is why you look to find the best differencing order in prior steps

© NECO, 2021

Auto-Arima

- Auto arima can help with the model fitting process as it follows a stepwise pattern to try many different orders.
- ** Auto Arima is **NOT** magic. Relying on Auto Arima will not necessarily return the best model**
- First, you need to understand all the steps in this process to use Auto Arima, any mistake in log transformation, differencing, etc. can lead to violations of assumptions
- Additionally, you may want to fit a model with a lower order despite have a slightly higher AIC.
- The risks and benefits of using a simpler/complex model must be decided by you.

© NECO, 2021

Residual check part 1

- This is done on the residuals of your fitted model. Compare the fitted values to the training dataset. Be sure that you are in the same units if you had done a log transformation.
- If you allow ARIMA to handle the differencing, you will be back in the original data units, in regards to differencing. It will **NOT** handle log transformations for you.

© NECO, 2021

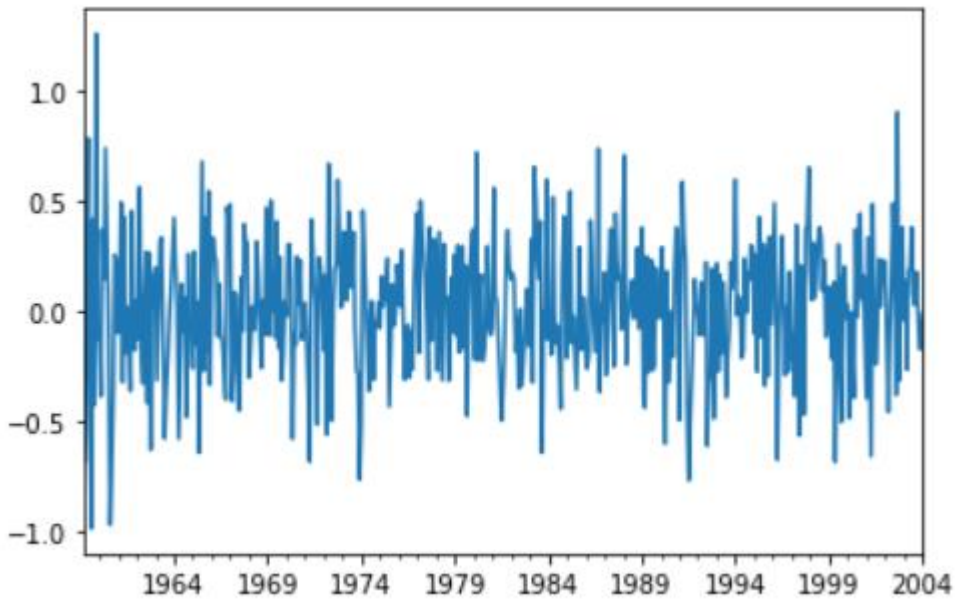
Residual check part 2

- It is recommended to look at the residuals in a plot, histogram, ACF and run LB test
- It is recommended to run each individually, as you have control over parameters such as # of lags in ACF or LB test

© NECO, 2021

Residuals plot

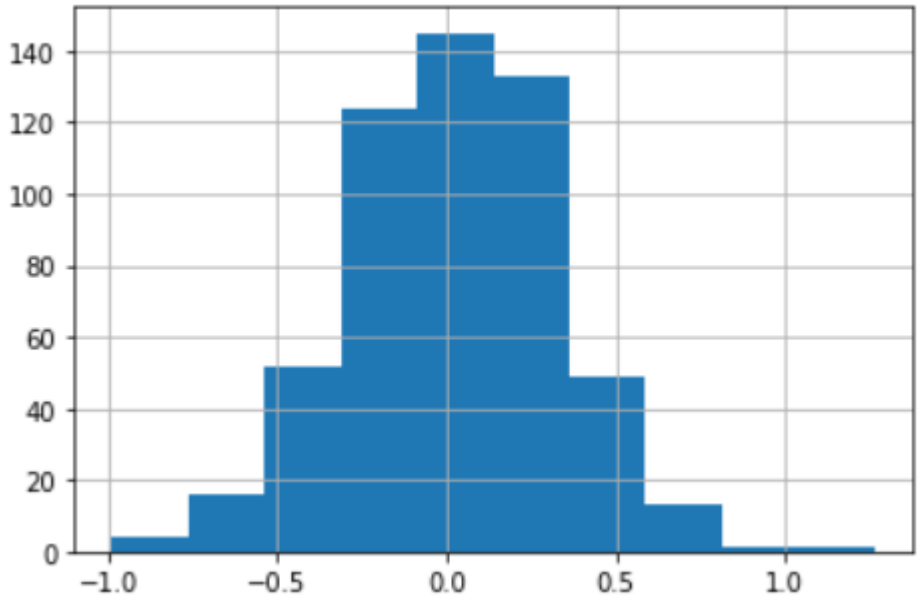
- Plot gives us a view of the residuals.
- We want to see a mean of zero, otherwise there is a bias that our model is not accounting for.
- We want to see constant variance, if not, ARIMA may not be the appropriate model
- We want to be aware of outliers, find out why there was an outlier, do we have a data source problem? Will there be more outliers when our model is in production?



© NECO, 2021

Residuals histogram

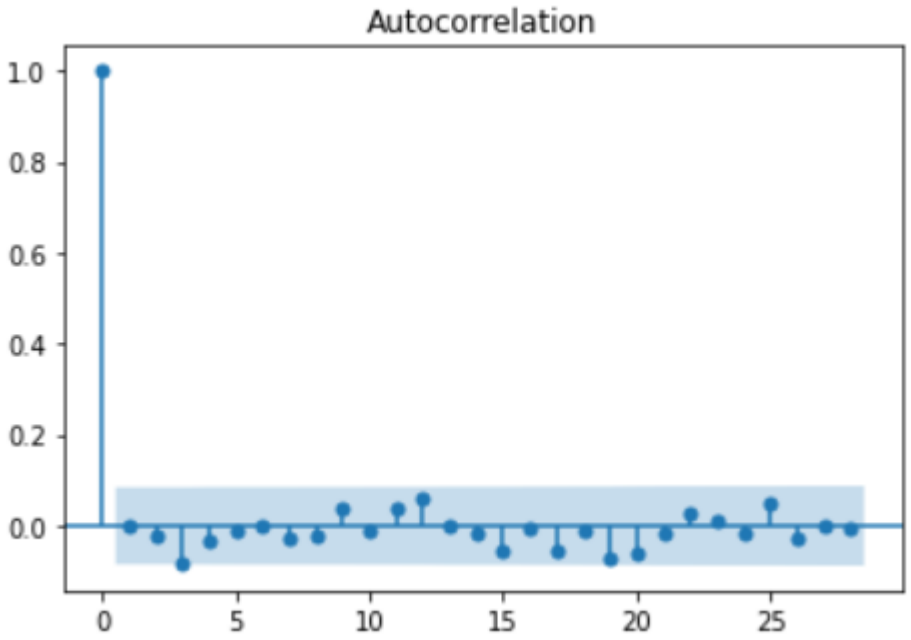
- Histogram will show us the distribution of our residuals.
- We want to see a normal distribution.
- Is this necessary? No, but we will have to understand that our forecasts and forecast intervals will not be as reliable and accurate as if we had normal data
- Our model is based on the assumption of normality and this assumption may be bent, if the user understands the consequences.



© NECO, 2021

Residuals ACF

- ACF will look for serial correlation in our residuals.
- You should not see significant lags
- If you do see autocorrelation, you have not accounted for all serial correlation in your model.
- Fit a new model with a different order



© NECO, 2021

Residuals Ljung-Box test

- Statistical test that looks for correlation at multiple lags
- ** If you have monthly data with yearly seasonality, you **must** check for lags up to 12**
- LB H0: The model is good…you want a large p-value
- If your p-value is small, then your residuals **do not** resemble white noise

	lb_stat	lb_pvalue	bp_stat	bp_pvalue
10	6.107735	0.806132	6.036041	0.812226

© NECO, 2021

Prediction and comparison

- Finally, we use our test set. If you have done a log transformation, make sure your train and test sets align in units.
- Predict using your model. Compare these forecasts to your test set visually as well as statistically
- MSE, MAE, RMSE, MAPE, sMAPE and MASE are all metrics we have used. They all have strengths and weaknesses. Best practice is to return all of them.
- Use these error metrics to compare this model to other models (prophet, VAR, Holt Winters, etc)

© NECO, 2021

Final words

- Time series is a very unique field of statistics and machine learning
- It takes repetition and exposure to learn how to make the best decisions with your modeling
- This process is for using the Box-Jenkins (ARIMA) family of models.
- In the real world, you will need to fit many different models and decide for yourself which model to use. This is reliant on many different reasons, such as your company/client's business case, computational resources, data quality, use of forecasts etc
- Time series is applicable in many industries. As ML and DL become more accessible, the accuracy and application of time series modeling will only grow in importance.

© NECO, 2021