# Trend

pattern that shows upward and downward movement over time.

# Seasonlity

Pattern that repeat after specific period such as day or hour

# Cyclical

* Pattern that occur after year or more.
* variations in the series that repeat with some regularity but of unknown and changing period.

# White noise

* When there is no predictable pattern or trend in the data
* Like zero mean, constant variance, no auto correlation.

# Autocorrelation

* Degree to which a time series in correlated with past values.
* It help to identify the underneath pattern in the data.
* We can use .shift to show the lagged data.
* Acf graph red shading indicates whether the correlations are significantly different froom zero.

# ACF

## Shadded portion

* This represent 95% confidence intervals.

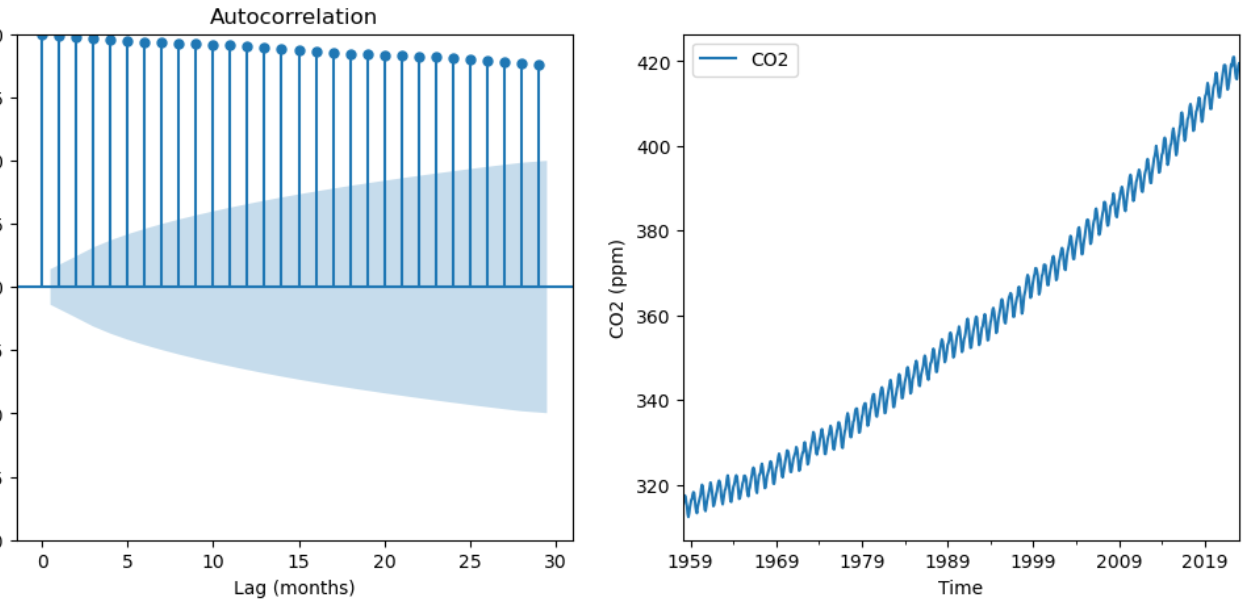
## Vertical line

* This represent different lags like at first line lag one and second line represent lag2.

## Vertical Height

* each line represent correlation coefficients.
* If the vertical lines extends outside the shadded portion it mean that lag is significantly important.
* That lag can be used for identify pattern.
* Positive autocorrelation indicate the presence of a trend or seasonality
* Negative auto correlation indicate the presence of a cyclical effect in the data.
* Both pa and na in same graph means that complex pattern need better model like ARIMA.
* PA indicates that current values of the time series are likely to be similar to past values
* NA indicates that current values are not likely to be similar to past values.

If mean remain same with oscilation then there will be no acf oscillation



Estimating the trend cycle

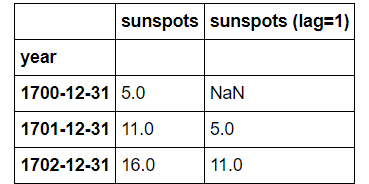
* Curve fitting : statmodels.tsa.tsatools.detrends()
* Moving average : .rolling()

nan : appear at both end of the rolling data

# lag 1 updated code

sun["sunspots (lag=1)"] = sun["sunspots"].shift(1)

sun



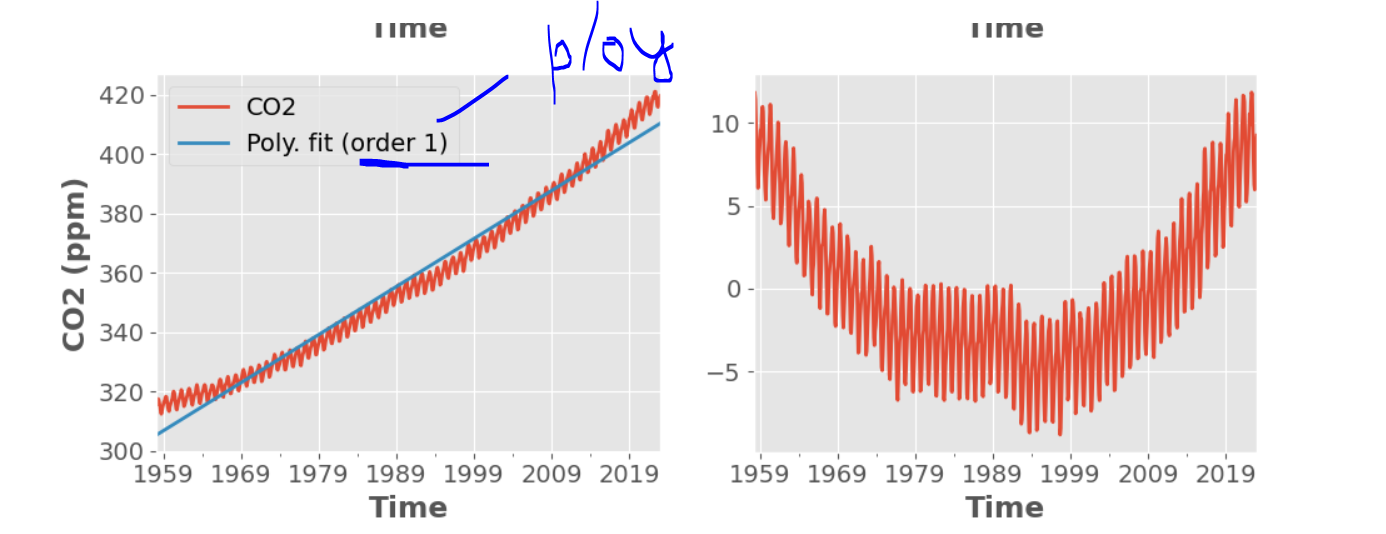
# Detrend

There are two main approaches to estimating the trend component of a time series (if it exists):

* Curve fitting
* Moving average

The function statsmodels.tsa.tsatools.detrend() can help us easily calculate and remove a simple polynomial curve from our data

Ploynomial of order 1 and left side detrended graph:



## Moving average

* It takes average of a fixed number of consecutive observations over time

Pandas rolling is used for this purpsoe.

window\_size = 3

df = pd.DataFrame([2, 6, 4, 2, 0, 1, 5, 3],

index=["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug"])

df.**rolling**(window=window\_size, center=**True**).mean()

* It is useful for smoothing out short-term fluctuations
* window size is equal to seasonality if the data have seasonality

if not then then there could abe issue: centered moving average?

* Curve fitting being a parametric method and moving average being a non-parametric method.
* Curve fitting is more appropriate when the trend component is expected to follow a specific functional form.
* Moving average is more appropriate when the trend component is expected to be smoother and less predictable

# Estimating seasonality

To compute the seasonal component we can simply:

* Remove the trend-cycle component from the data
* Estimate the seasonal component by averaging