statsmodels.tsa.seasonal.seasonal_decompose

statsmodels.tsa.seasonal.seasonal_decompose(*x*, *model='additive'*, *filt=None*, *period=None*, *two_sided=True*, *extrapolate_trend=0*)[source] [../_modules/statsmodels/tsa/seasonal.html#seasonal_decompose]

Seasonal decomposition using moving averages.

Parameters:

x : array_like [https://numpy.org/doc/stable/glossary.html#term-array_like]

Time series. If 2d, individual series are in columns. x must contain 2 complete cycles.

model : {"additive", "multiplicative"}, `optional`

Type of seasonal component. Abbreviations are accepted.

filt : array_like [https://numpy.org/doc/stable/glossary.html#term-array_like], `optional`

The filter coefficients for filtering out the seasonal component. The concrete moving average method used in filtering is determined by two_sided.

period : `int` [https://docs.python.org/3/library/functions.html#int], `optional`

Period of the series. Must be used if x is not a pandas object or if the index of x does not have a frequency. Overrides default periodicity of x if x is a pandas object with a timeseries index.

two_sided : bool [https://docs.python.org/3/library/stdtypes.html#bltin-boolean-values], `optional`

The moving average method used in filtering. If True (default), a centered moving average is computed using the filt. If False, the filter coefficients are for past values only.

extrapolate_trend : `int` [https://docs.python.org/3/library/functions.html#int] or 'freq', `optional`

If set to > 0, the trend resulting from the convolution is linear least-squares extrapolated on both ends (or the single one if two_sided is False) considering this many (+1) closest points. If set to 'freq', use *freq* closest points. Setting this parameter results in no NaN values in trend or resid components.

Returns:

`DecomposeResult` [statsmodels.tsa.seasonal.DecomposeResult.html#statsmodels.tsa.seasonal.DecomposeResult]

A object with seasonal, trend, and resid attributes.

See also

`statsmodels.tsa.filters.bk_filter.bkfilter` [statsmodels.tsa.filters.bk_filter.bkfilter.html#statsmodels.tsa.filters.bk_filter.bkfilter]

Baxter-King filter.

`statsmodels.tsa.filters.cf_filter.cffilter` [statsmodels.tsa.filters.cf_filter.cffilter.html#statsmodels.tsa.filters.cf_filter.cffilter]

Christiano-Fitzgerald asymmetric, random walk filter.

`statsmodels.tsa.filters.hp_filter.hpfilter` [statsmodels.tsa.filters.hp_filter.hpfilter.html#statsmodels.tsa.filters.hp_filter.hpfilter]

Hodrick-Prescott filter.

`statsmodels.tsa.filters.convolution_filter`

> Linear filtering via convolution.
>
> `statsmodels.tsa.seasonal.STL` [statsmodels.tsa.seasonal.STL.html#statsmodels.tsa.seasonal.STL]
>
> Season-Trend decomposition using LOESS.

Notes

This is a naive decomposition. More sophisticated methods should be preferred.

The additive model is Y[t] = T[t] + S[t] + e[t]

The multiplicative model is Y[t] = T[t] * S[t] * e[t]

The results are obtained by first estimating the trend by applying a convolution filter to the data. The trend is then removed from the series and the average of this de-trended series for each period is the returned seasonal component.