

# Time series analysis and forecasting

Integrative analytics for multi-modal, multi-scale neuroscience  
IPN Summer School

Golia Shafiei, PhD candidate

Supervisor: Dr. Bratislav Misic

Network Neuroscience Lab

McConnell Brain Imaging Centre

Montreal Neurological Institute, McGill University



**McGill**  
UNIVERSITY

June 30, 2021

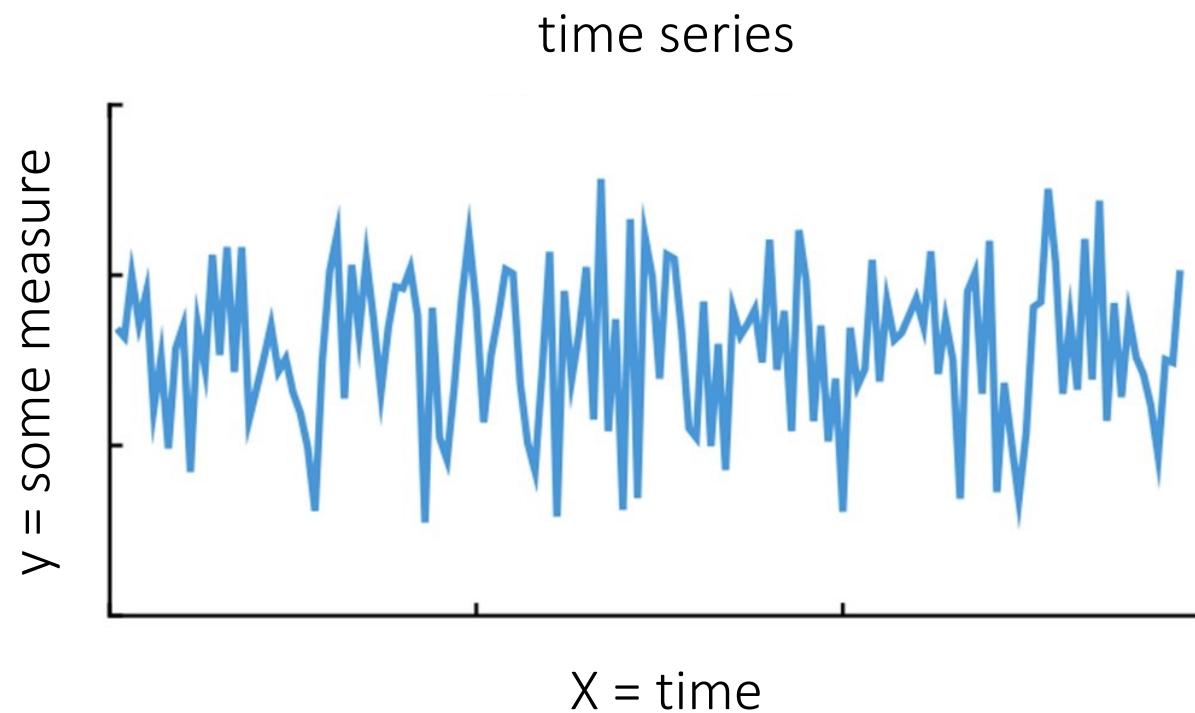


# Content

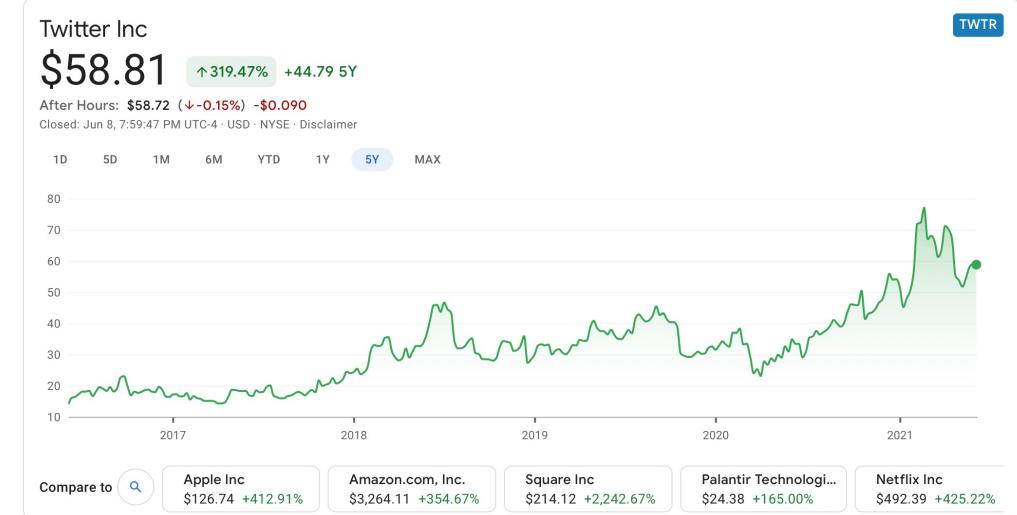
- Definition and characteristics of a time series
- Time series analysis
- Time series forecasting
- Spectral analysis
- Time series phenotyping

# What is a time series?

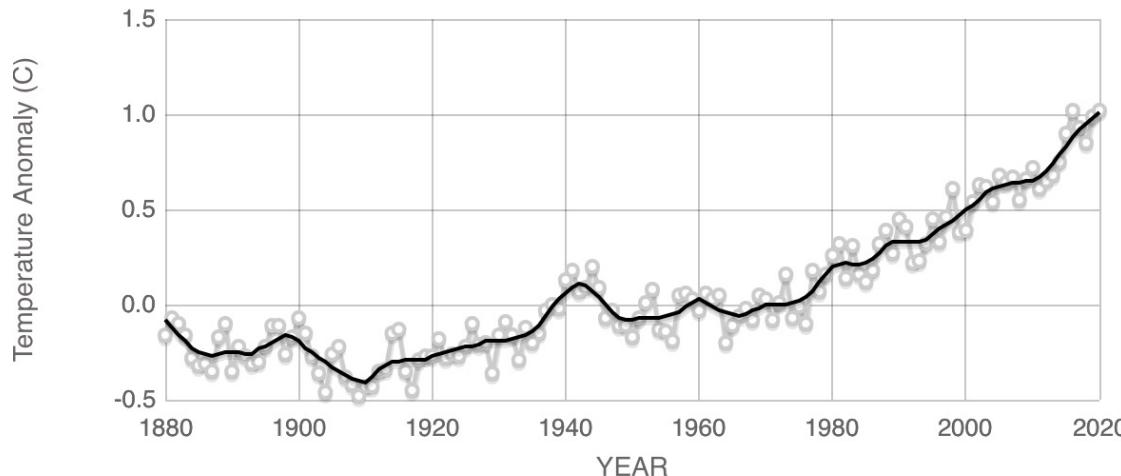
- Repeated measurements of a variable over time



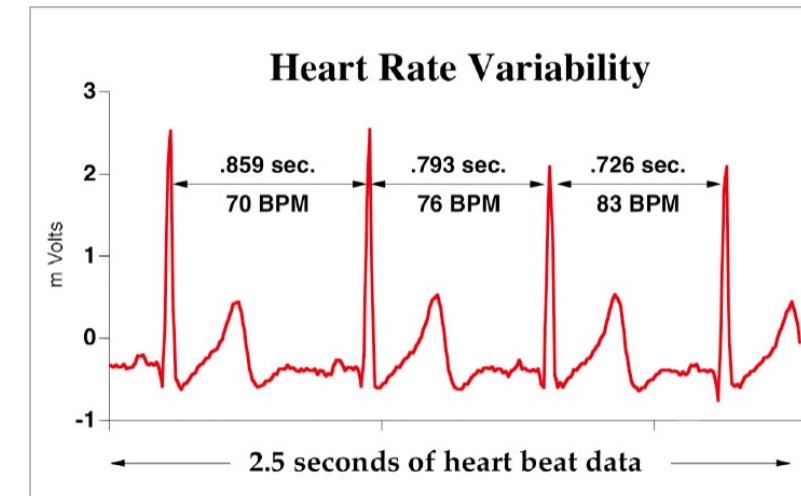
# What is a time series?



Source: <https://www.google.com/finance/>

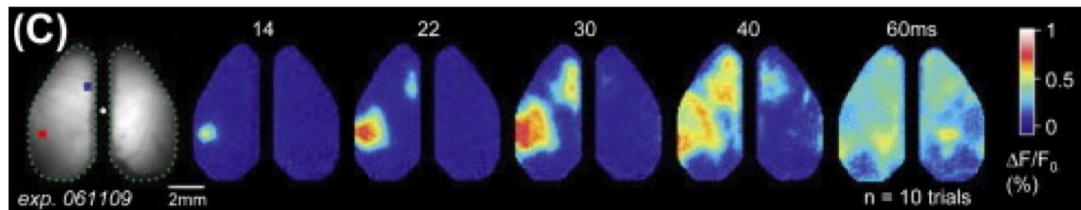
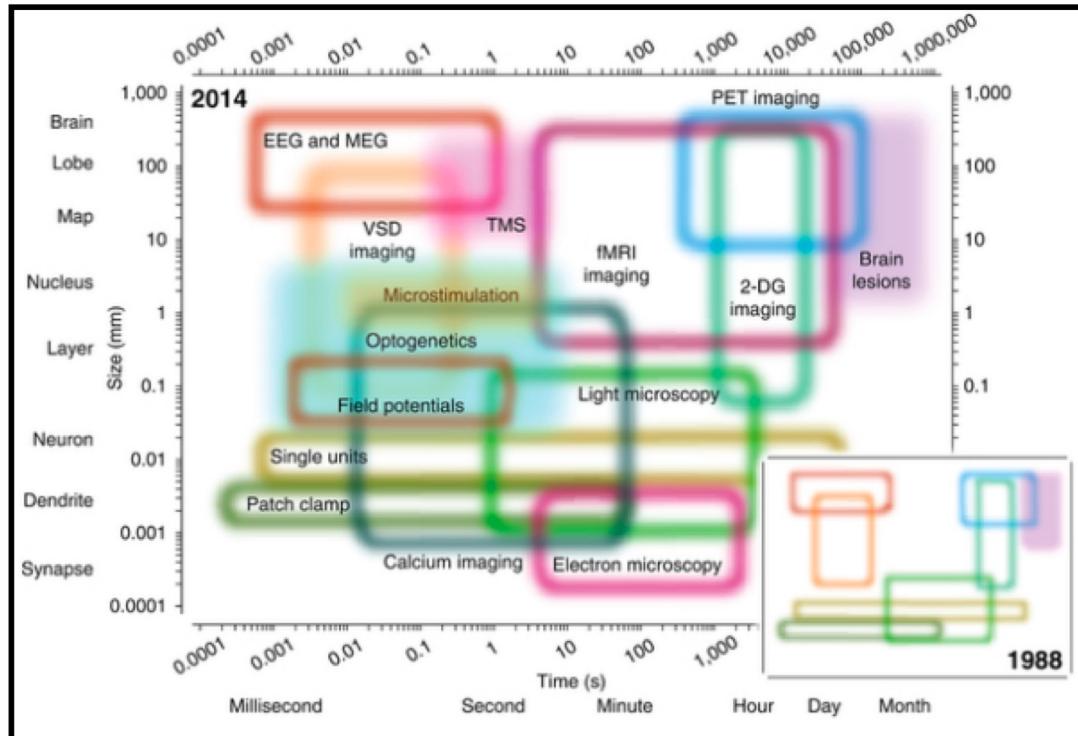


Source: <https://climate.nasa.gov>



# What is a time series?

- Neural time series measured at different spatial and temporal scales



## Electrophysiology:

- > voltage [  $V(t)$  ]
- > currents (ion exchange)
- > **MEG**: magnetic fields

## Optical Imaging: (luminance)

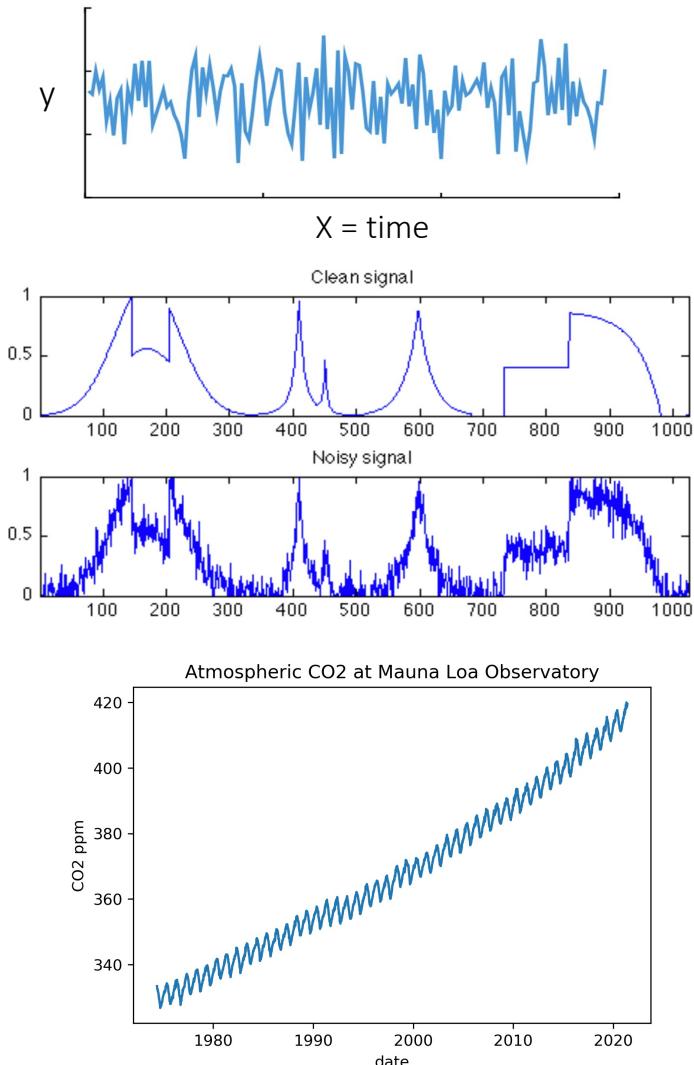
- Concentration of something
- > Calcium
  - > Non-specific charges  
(voltage sensitive dye)

## Medical Imaging:

- > fMRI: magnetic spin
- > PET: radiation (positron)

# Characteristics of time series data

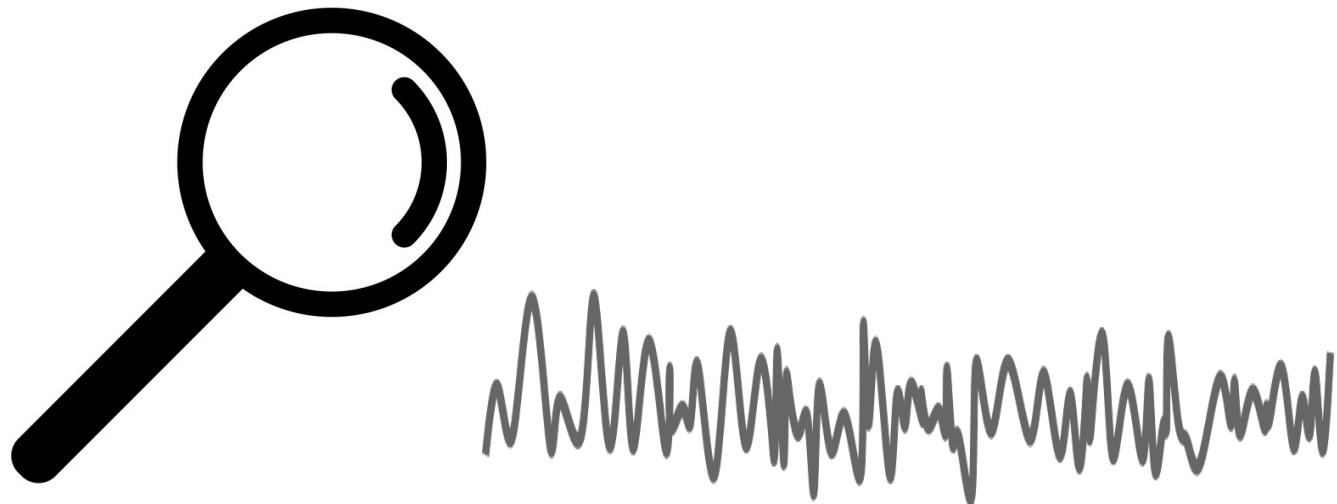
- An ordered sequence of values of a variable measured over time
  - Time is a monotonically increasing axis
  - Time series vs randomly sampled data
- Structure and behavior of data
  - Signal and noise components
  - Stationarity vs non-stationarity
  - Trend
  - Seasonality or periodicity
  - Autocorrelation and temporal dependency
- Stochastic behavior of time series
  - Change points
  - Volatile behavior



# Objectives of time series analysis

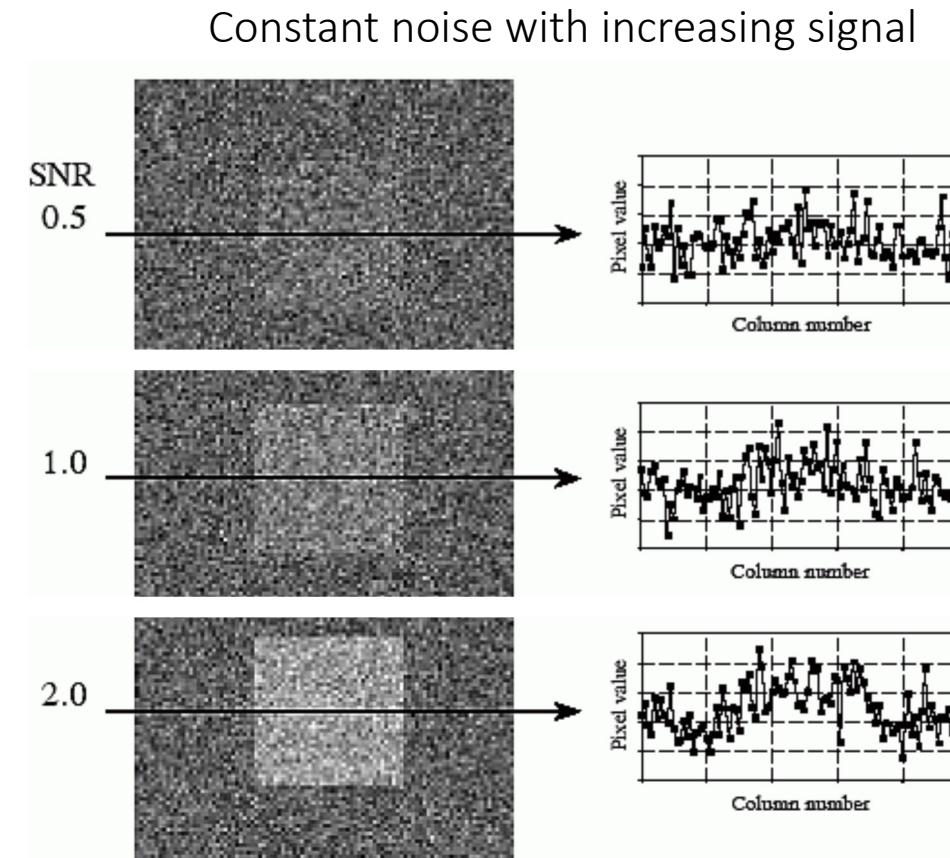
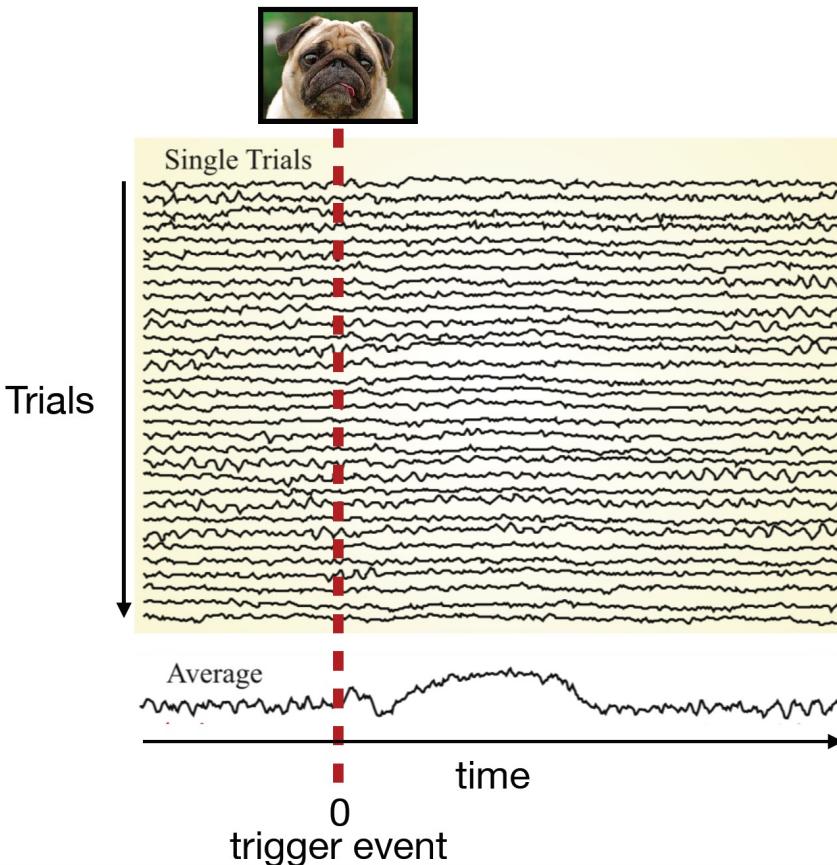
- Exploratory data analysis
  - How does data behave over time? Trend, seasonality, stationarity
  - Understanding the nature of processes generating the data
- Predictive data analysis
  - Estimate past, present, and future of data
  - Time series forecasting
  - Time series classification
  - Anomaly prediction

# Explore the data!



# Signal and noise components

- Real world time-series are noisy.
- Signal-to-noise ratio (SNR) quantifies how noisy a signal is.



Root mean square (rms)

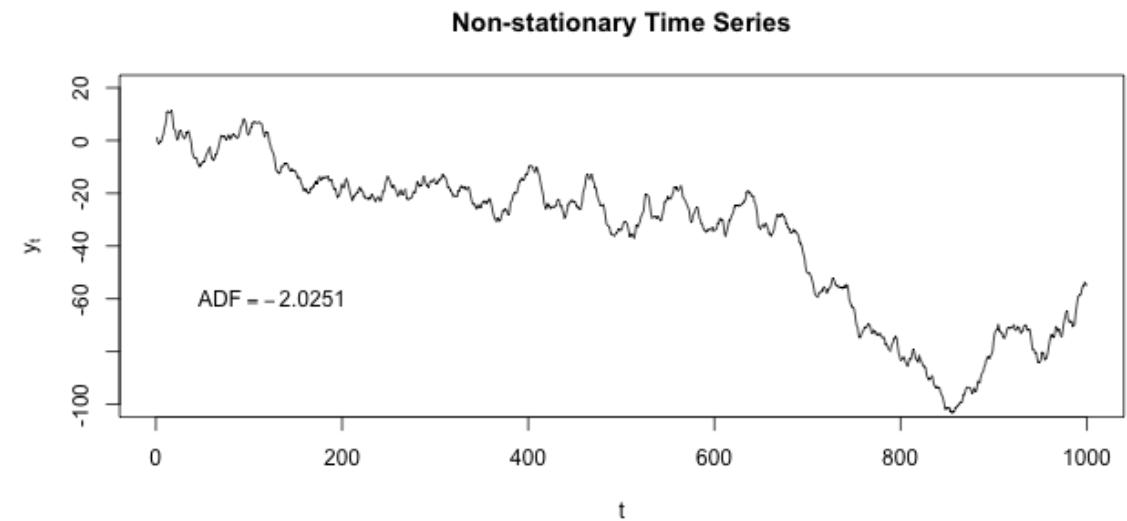
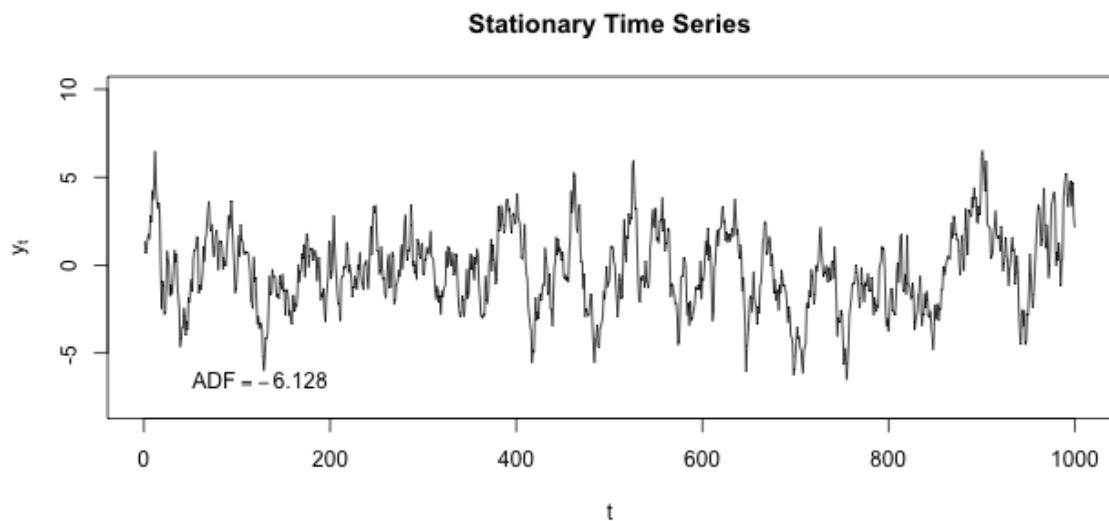
$$rms = \sqrt{\frac{1}{N} \sum_{i=1}^n x_i^2}$$

SNR in decibels (dB)

$$SNR = 20\log_{10} \frac{rms_{signal}}{rms_{noise}}$$

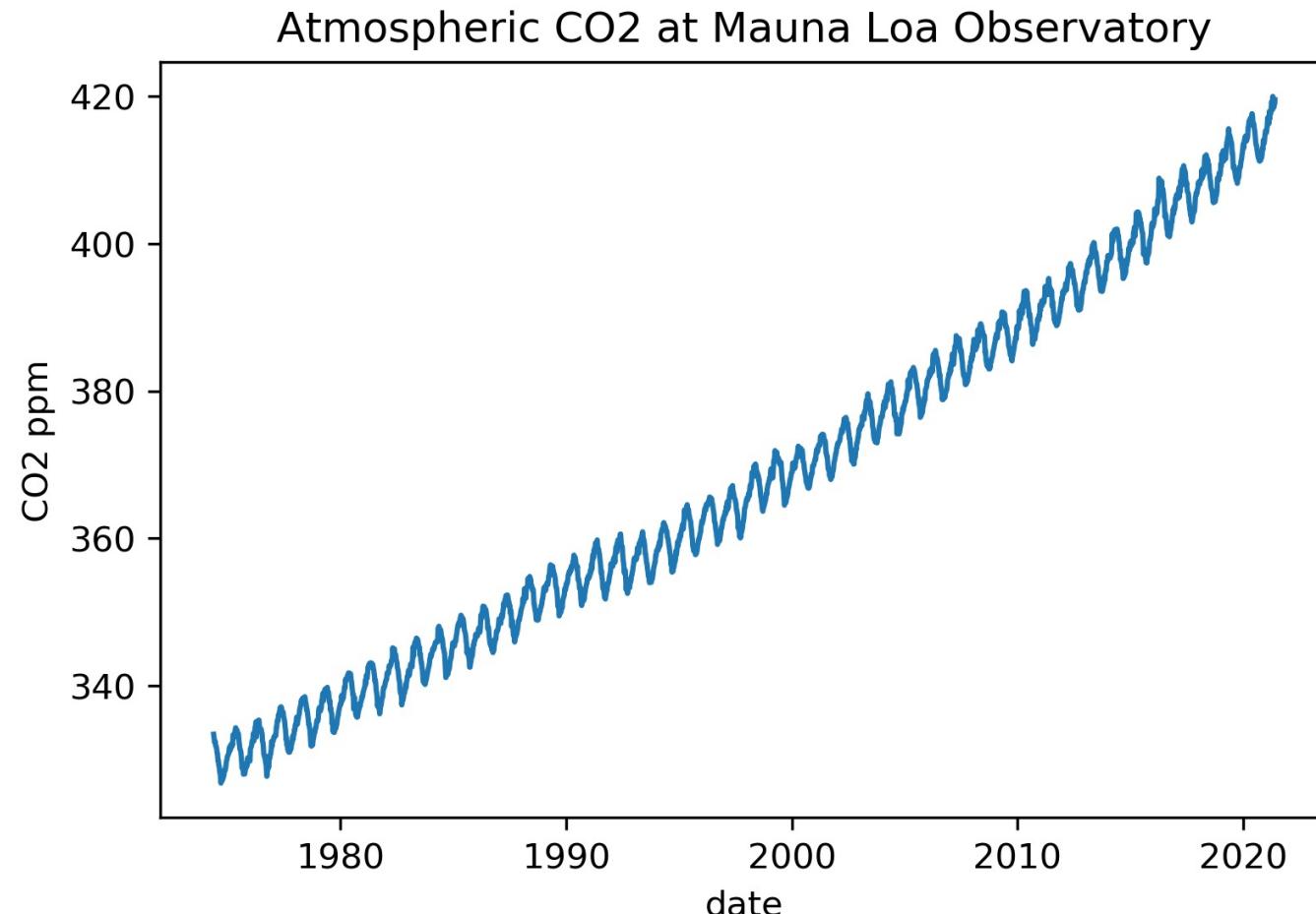
# Stationary vs non-stationary time series

- Stationary time series
  - Mean, standard deviation, and structure of autocorrelation function are constant over time



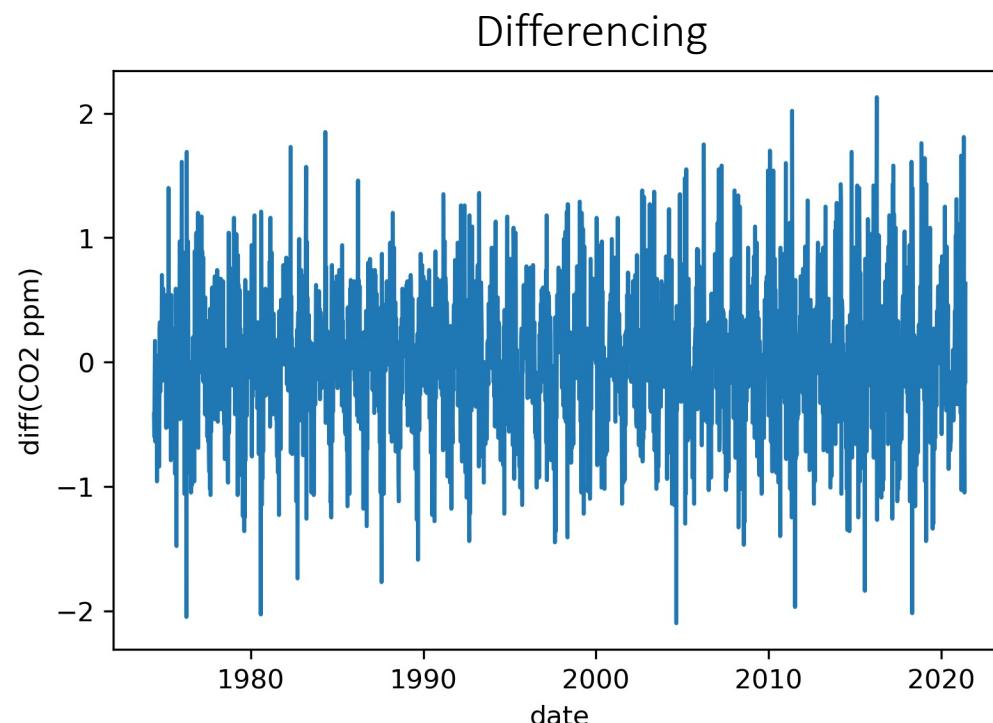
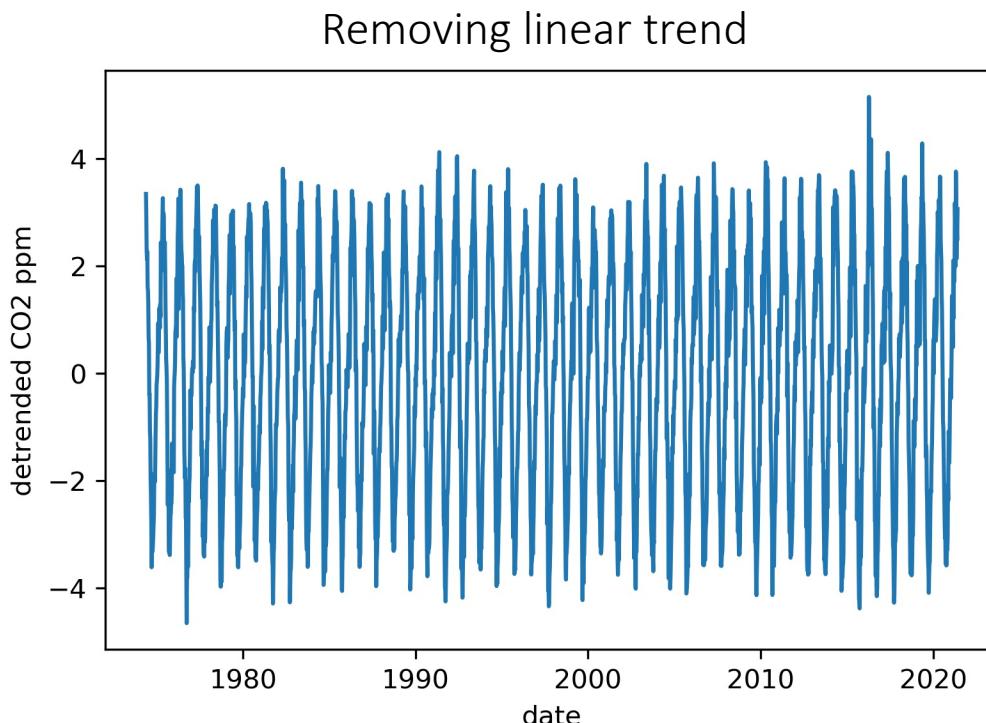
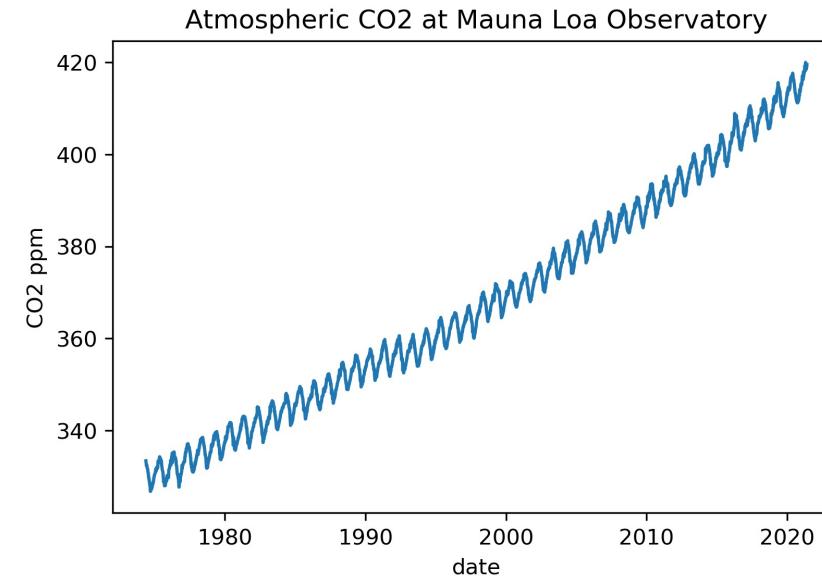
# Example dataset

- Atmospheric CO<sub>2</sub> at Mauna Loa Observatory
- Trend: change in the mean



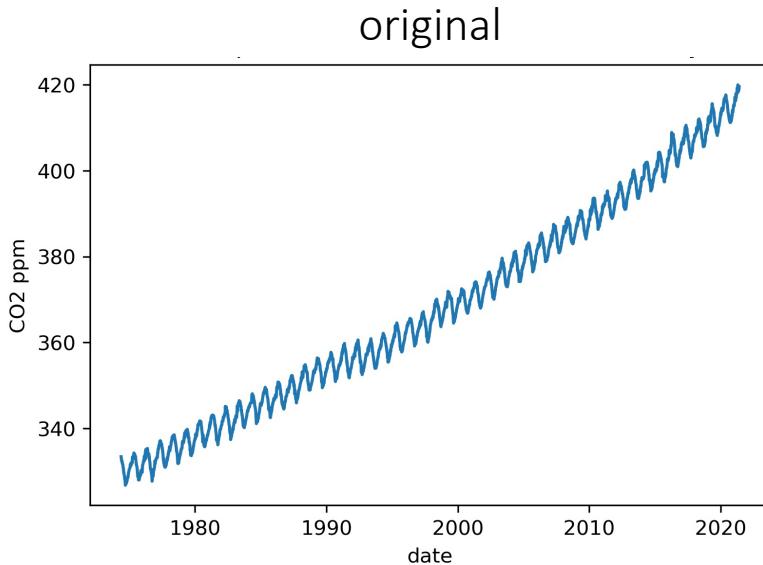
# Example dataset

- Detrending
  - Remove the linear trend
  - Differencing

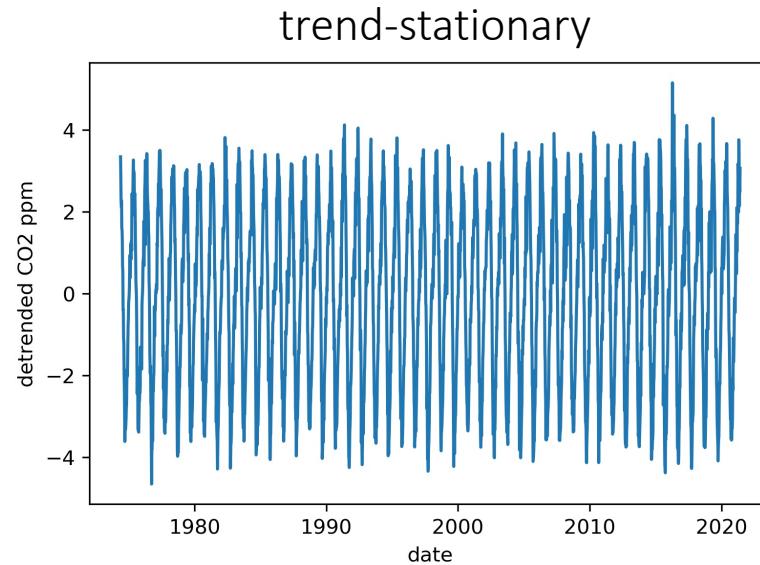


# Example dataset

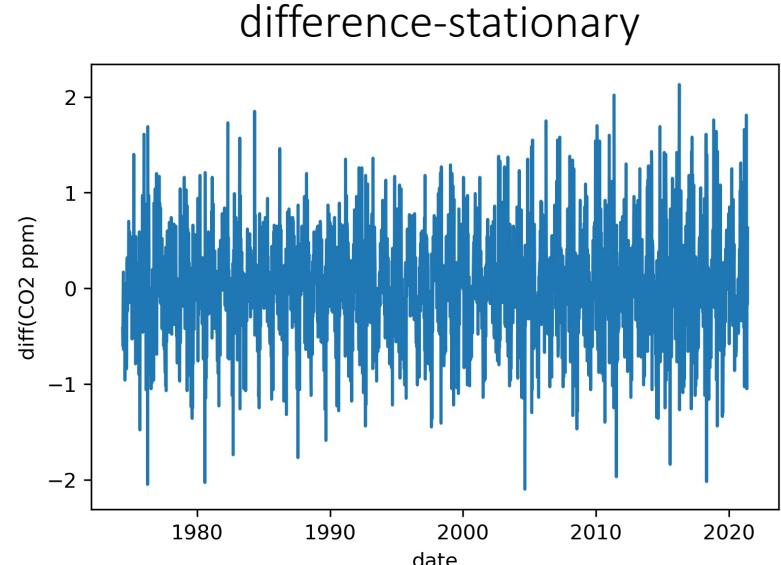
- Stationarity tests:
  - ADF: Augmented Dickey Fuller test
  - KPSS: Kwiatkowski-Phillips-Schmidt-Shin test



ADF Statistic: 0.46  
p-value: 0.98



ADF Statistic: -25.39  
p-value: 0.0



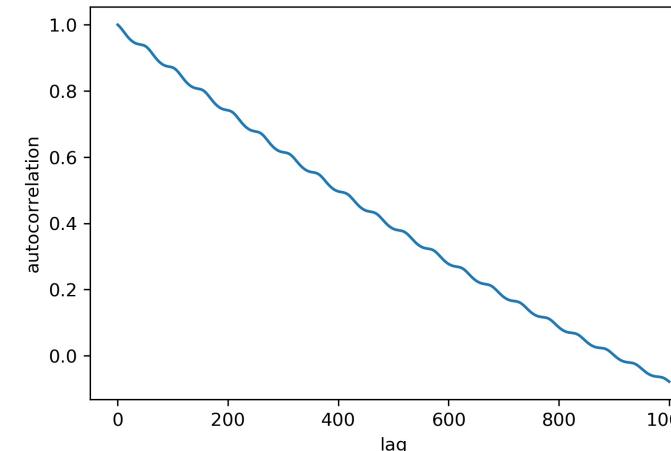
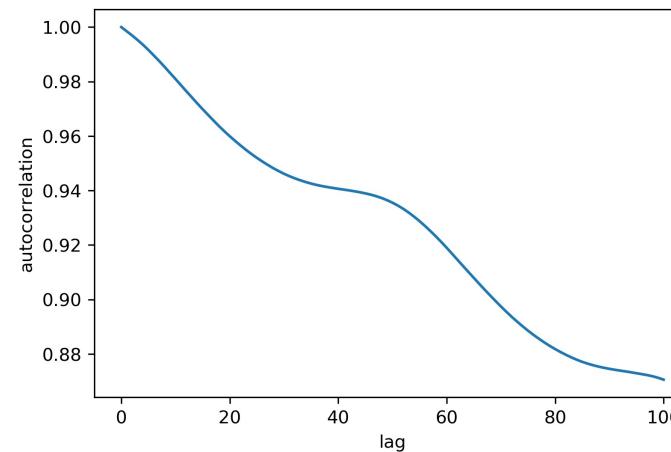
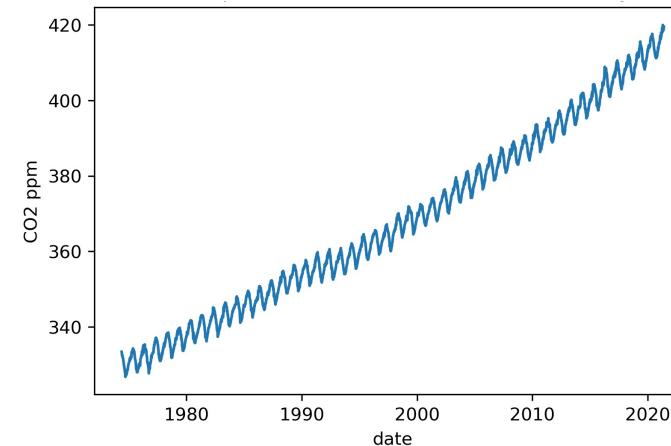
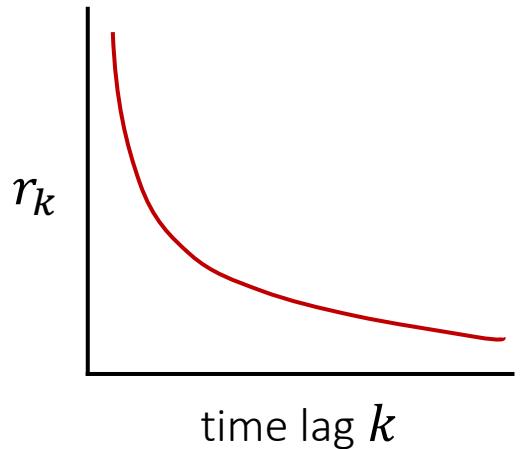
ADF Statistic: -15.92  
p-value: 7.89e-29

# Example dataset

- Seasonality and periodicity
- Autocorrelation
  - Non-randomness in the data
  - Model selection for data fitting

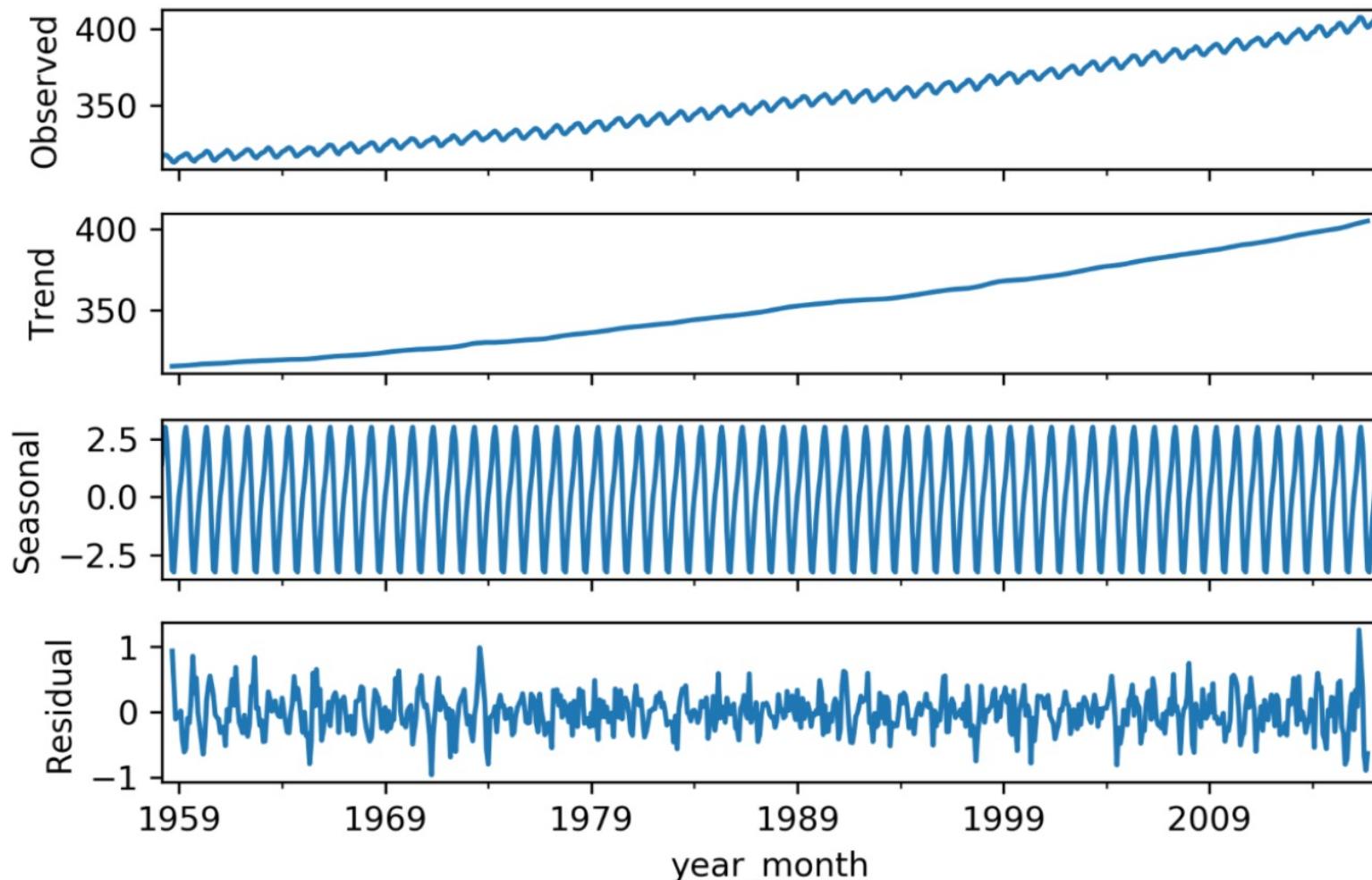
Autocorrelation function

$$r_k = \frac{\sum_{i=1}^{N-k} (Y_i - \bar{Y})(Y_{i+k} - \bar{Y})}{\sum_{i=1}^N (Y_i - \bar{Y})^2}$$



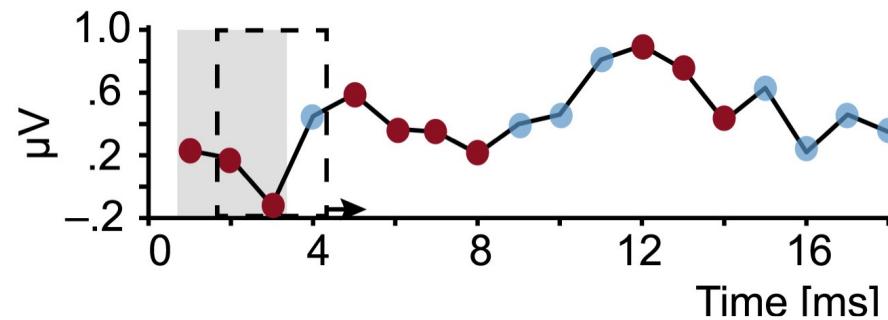
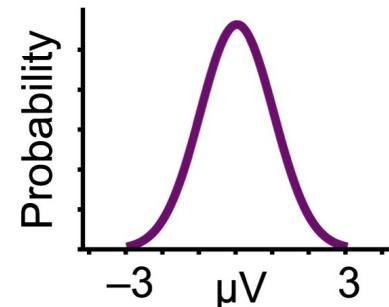
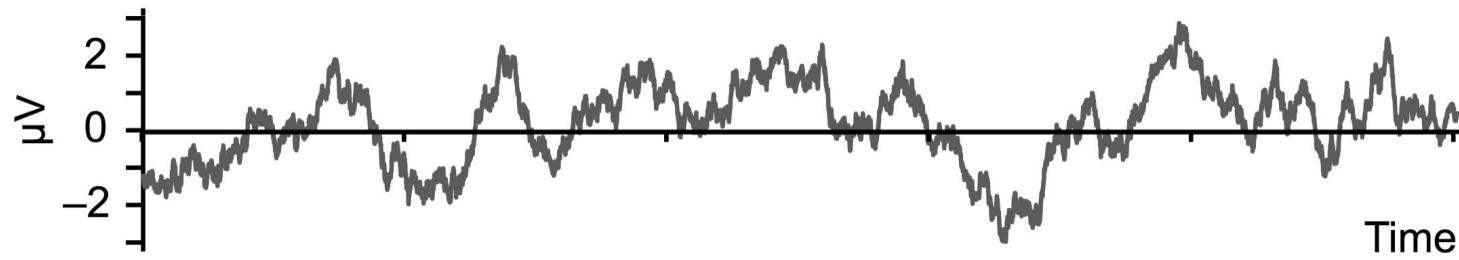
# Example dataset

- Decompose signal into different components



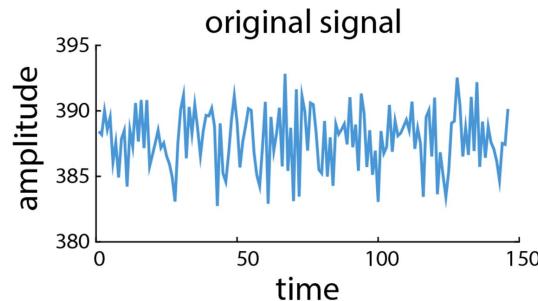
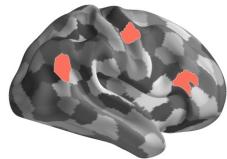
# Measures of time series

- Stationarity
- Distribution: mean, variance, skewness, kurtosis
- Temporal dependencies: autocorrelation, temporal entropy



# Measures of time series

time-series



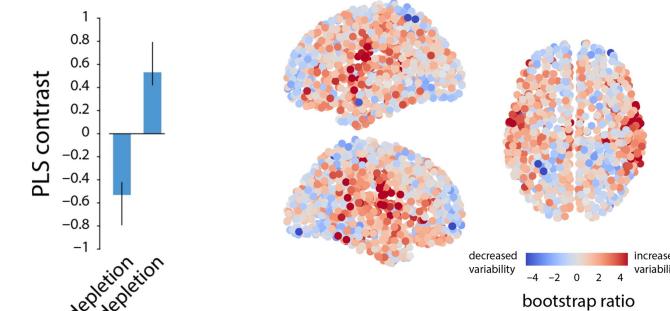
$$SE(m, r, N) = \ln \frac{\sum_{i=1}^{N-m} n_i^m}{\sum_{i=1}^{N-m} n_i^{m+1}}$$

$$MSSD = \sqrt{\frac{\sum_{i=1}^{n-1} (x_{i+1} - x_i)^2}{n - 1}}$$

variability

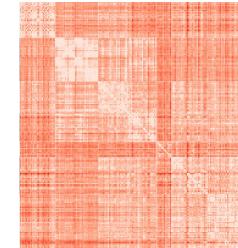
region

$p = 0.014$

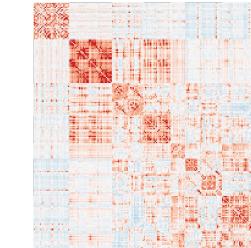


Shafiei et al., 2019, *Cerebral cortex*

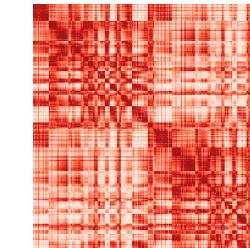
MSSD Similarity



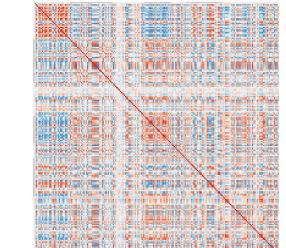
functional connectivity



distance

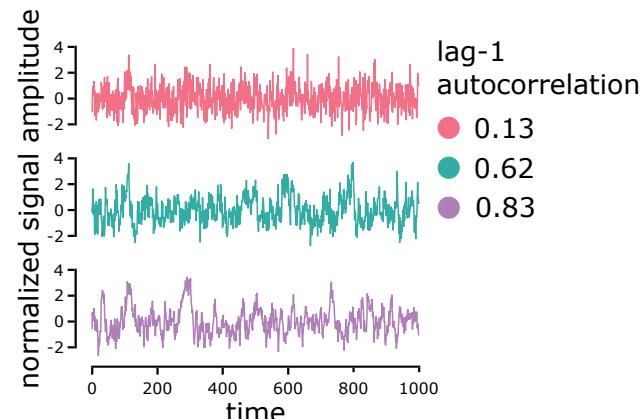
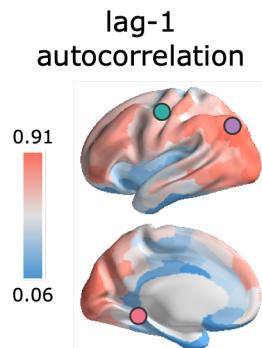


grey matter similarity



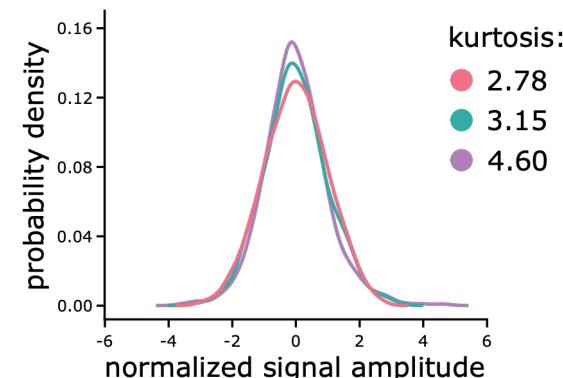
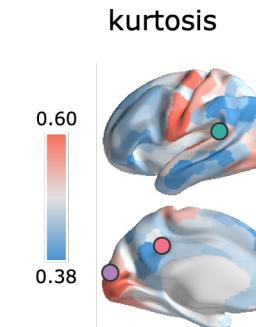
Baracchini et al., 2021, *NeuroImage*

Autocorrelation



Shafiei et al., 2020, *eLife*

Data distribution



# Time series forecasting

- **Objective:** Predict time-series values for future time points from past values.
- AutoRegressive models: AR( $p$ )

$$y_t = B_0 + B_1 y_{t-1} + B_2 y_{t-2} + \cdots + B_p y_{t-p} + e_t$$

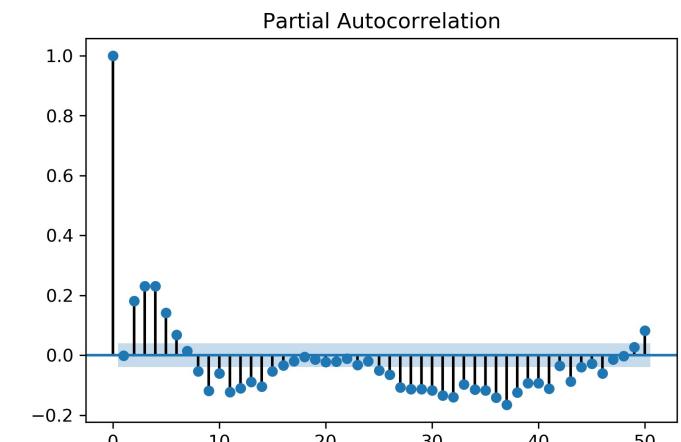
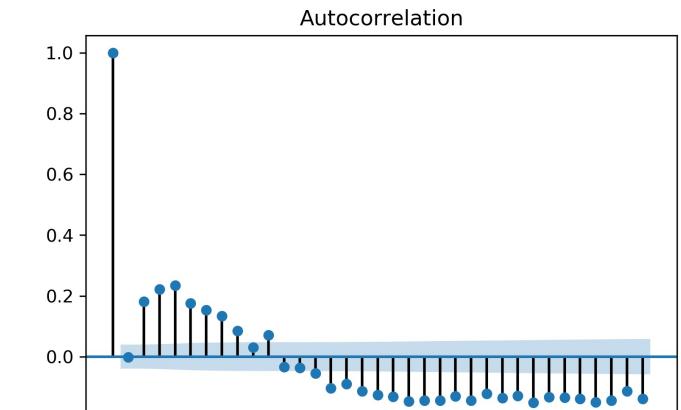
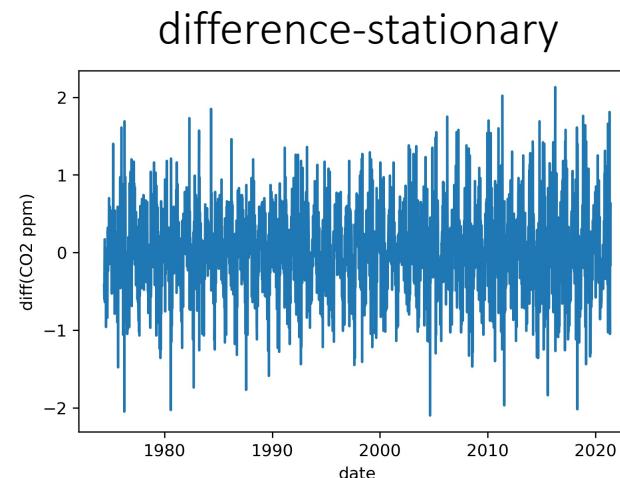
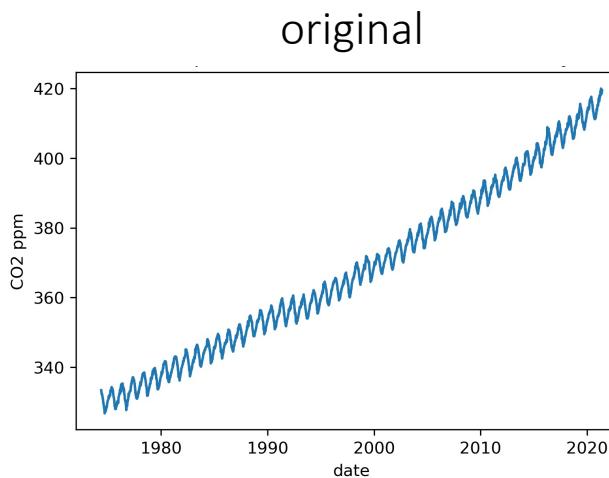
- Moving average models: MA( $q$ )

$$y_t = \mu + e_t + C_1 e_{t-1} + C_2 e_{t-2} + \cdots + C_q e_{t-q}$$

- Autoregressive Moving Average models: ARMA( $p,q$ )
- Autoregressive Integrated Moving Average models: ARIMA( $p,d,q$ )
- Seasonal ARIMA models: SARIMA
- ...

# Time series forecasting

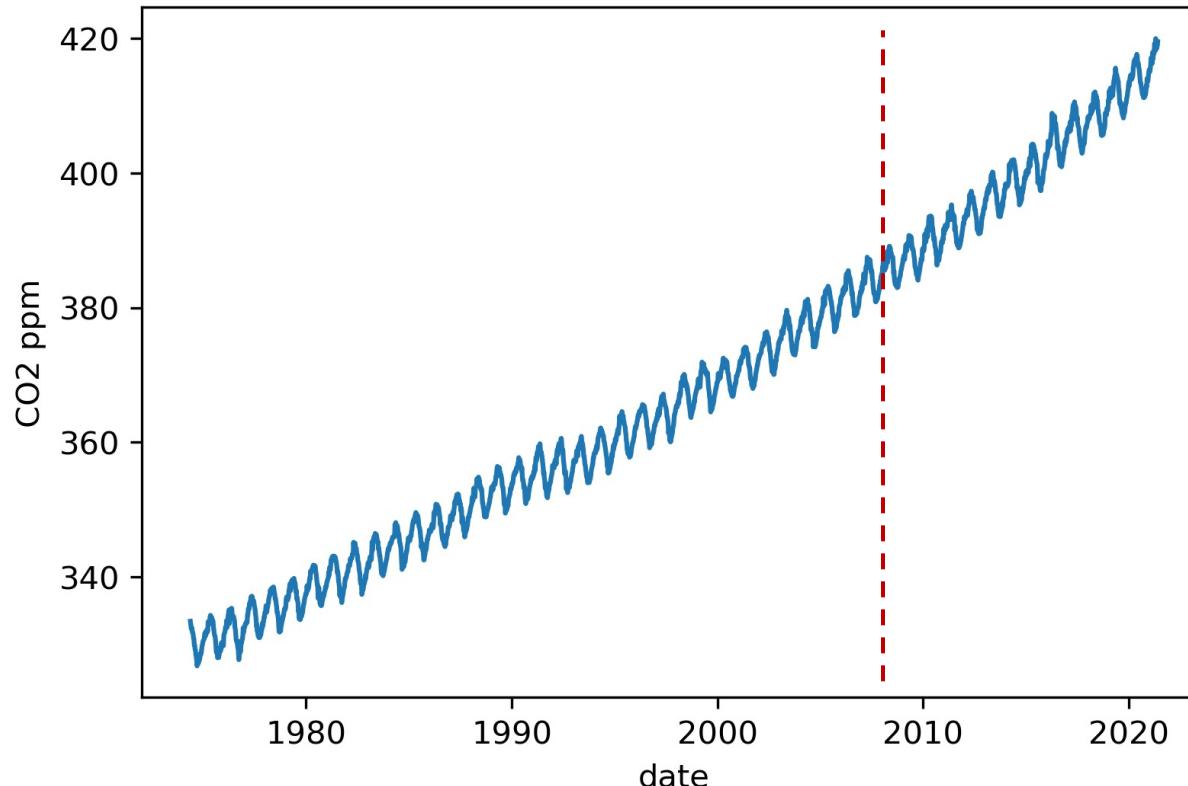
- Parameter selection:
  - MA model parameter: Autocorrelation function (ACF)
  - AR model parameter: Partial autocorrelation function (PACF)
  - Grid search



# Time series forecasting

- Forecasting using AutoRegressive model from Python's *statsmodels*

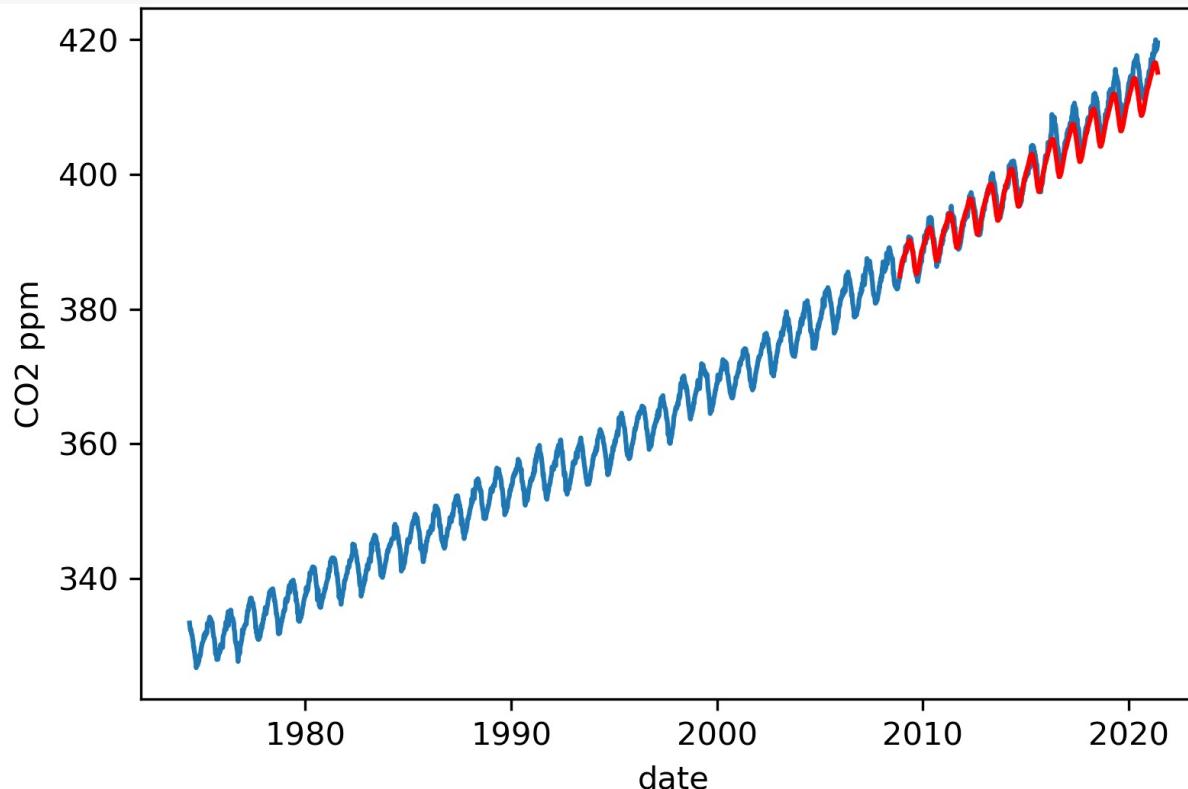
```
from statsmodels.tsa import ar_model
# train AutoRegressive model using training set
ar_model = ar_model.AR(train)
ar_model_fit = ar_model.fit(maxlag=50)
```



# Time series forecasting

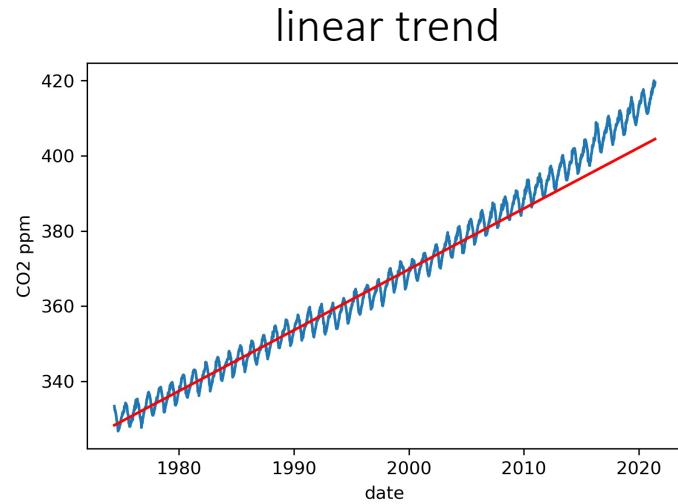
- Forecasting using AutoRegressive model from Python's *statsmodels*

```
# predict and plot time-series values for test data using test set index
predicted_values_ar = ar_model_fit.predict(test.index[0], test.index[-1])
plt.plot(ppm)
plt.plot(predicted_values_ar, 'red')
plt.xlabel('date')
plt.ylabel('CO2 ppm')
```



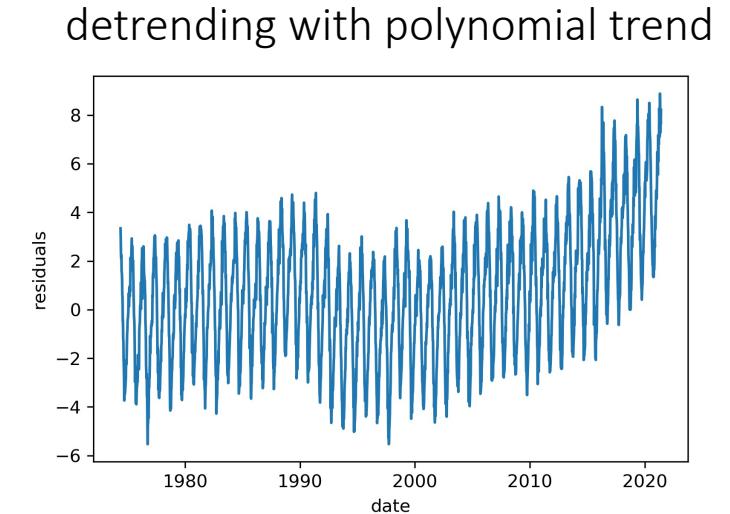
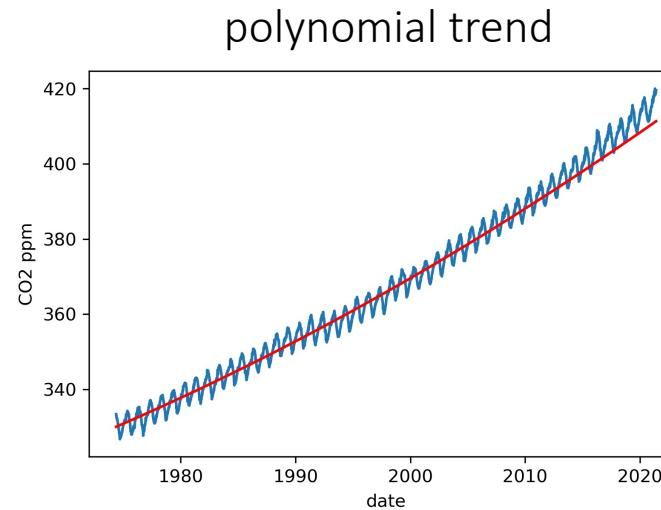
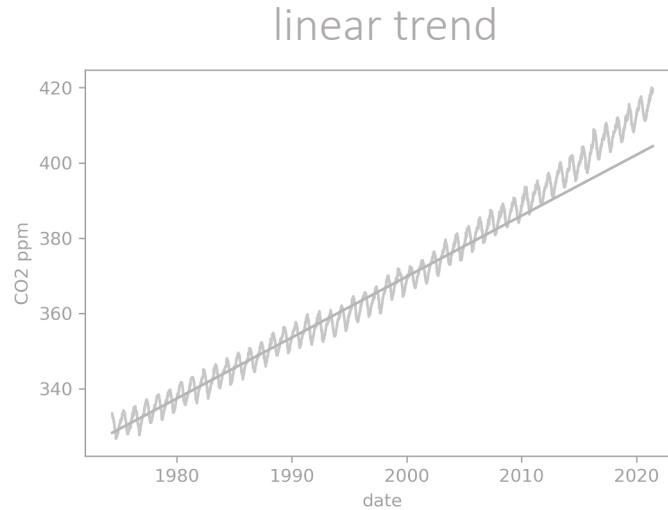
# Time series forecasting

- Forecasting using regression analysis and AR model



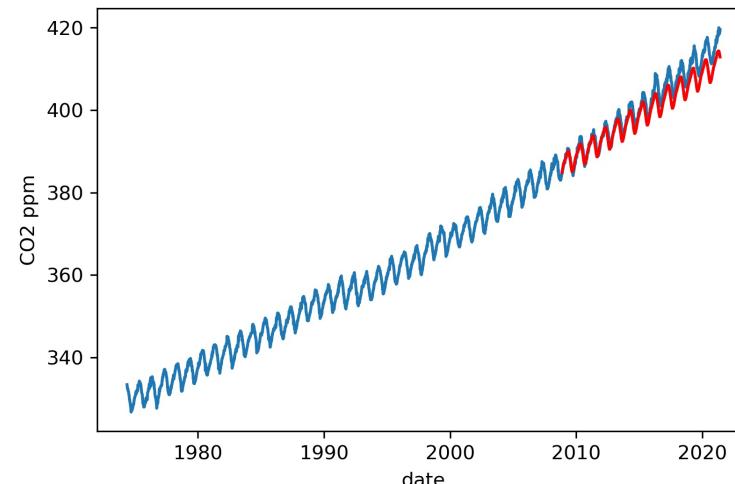
# Time series forecasting

- Forecasting using regression analysis and AR model



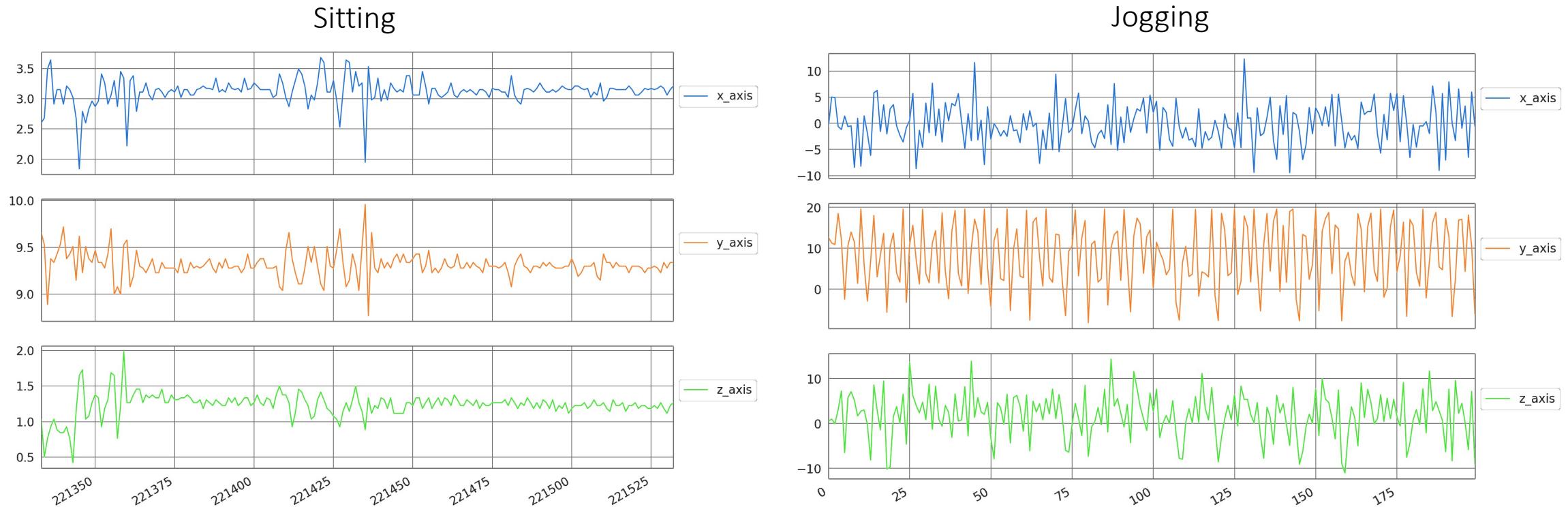
AR model on  
detrended data

A large grey arrow points from the bottom left towards the final time series plot, indicating the flow from the detrending process to the final forecast.



# Time series classification

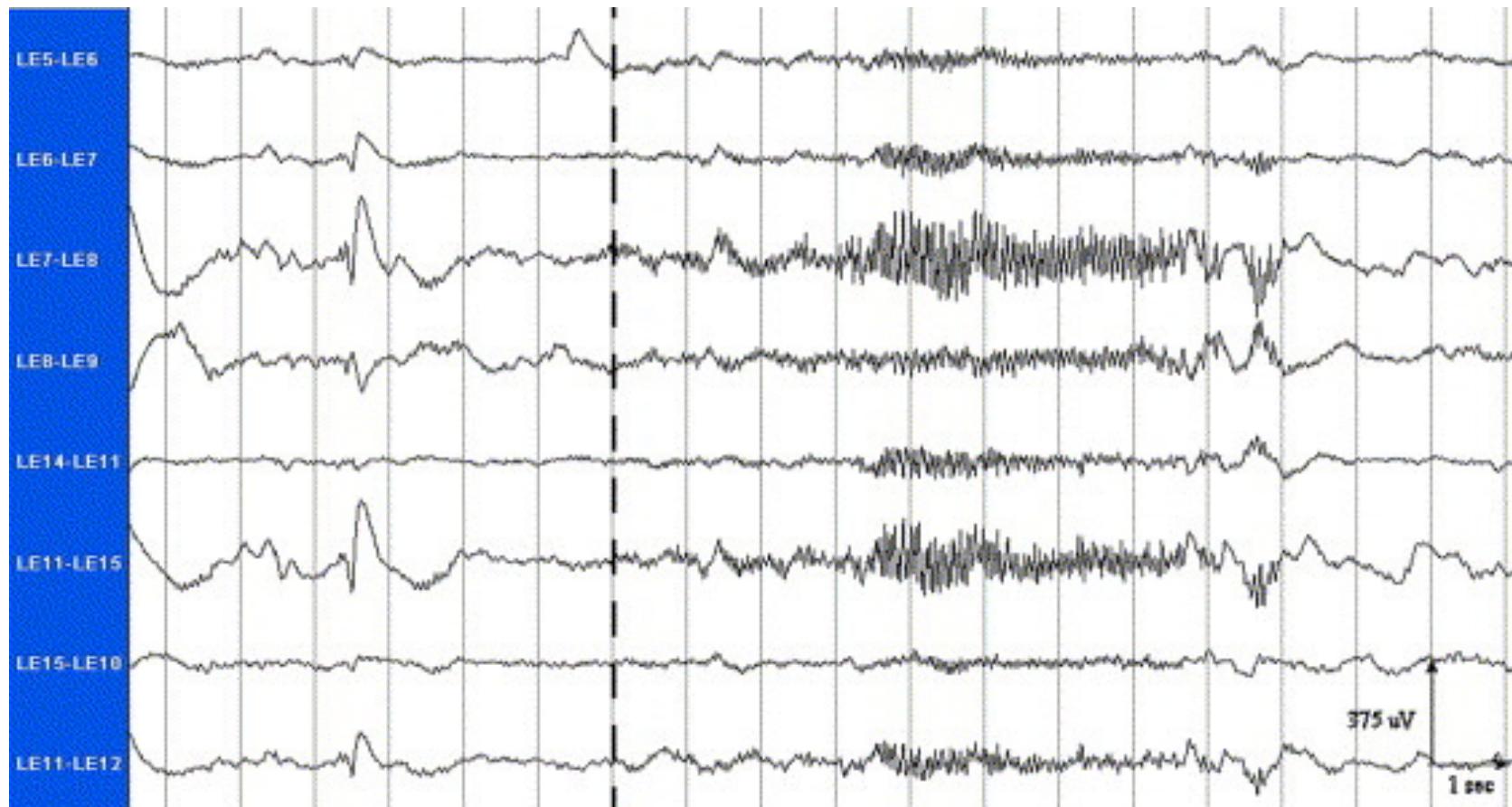
- Classify time series into different categories



# Time series classification

- Classify time series into different categories

Epileptic EEG recording



# Time series classification

- Classify time series into different categories
- Scikit-learn classification algorithms



The image is a screenshot of the official Scikit-learn website. At the top, there's a navigation bar with the title "scikit-learn" and "Machine Learning in Python", followed by links for "Getting Started", "Release Highlights for 0.24", and "GitHub". To the right of the navigation bar is a yellow box containing a bulleted list of features:

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

The main content area is divided into three sections: "Classification", "Regression", and "Clustering".

- Classification:** A brief description: "Identifying which category an object belongs to." It lists "Applications: Spam detection, image recognition." and "Algorithms: SVM, nearest neighbors, random forest, and more...". Below this is a grid of nine small plots showing various classification models like K-Means and Decision Trees applied to datasets like Iris and Wine.
- Regression:** A brief description: "Predicting a continuous-valued attribute associated with an object." It lists "Applications: Drug response, Stock prices." and "Algorithms: SVR, nearest neighbors, random forest, and more...". Below this is a line plot titled "Boosted Decision Tree Regression" showing a red line representing the model's predictions against training samples (black dots) over a range of data values from 0 to 6.
- Clustering:** A brief description: "Automatic grouping of similar objects into sets." It lists "Applications: Customer segmentation, Grouping experiment outcomes" and "Algorithms: k-Means, spectral clustering, mean-shift, and more...". Below this is a scatter plot titled "K-means clustering on the digits dataset (PCA-reduced data)" showing a 2D PCA-reduced digit dataset with centroids marked by white crosses and colored clusters.

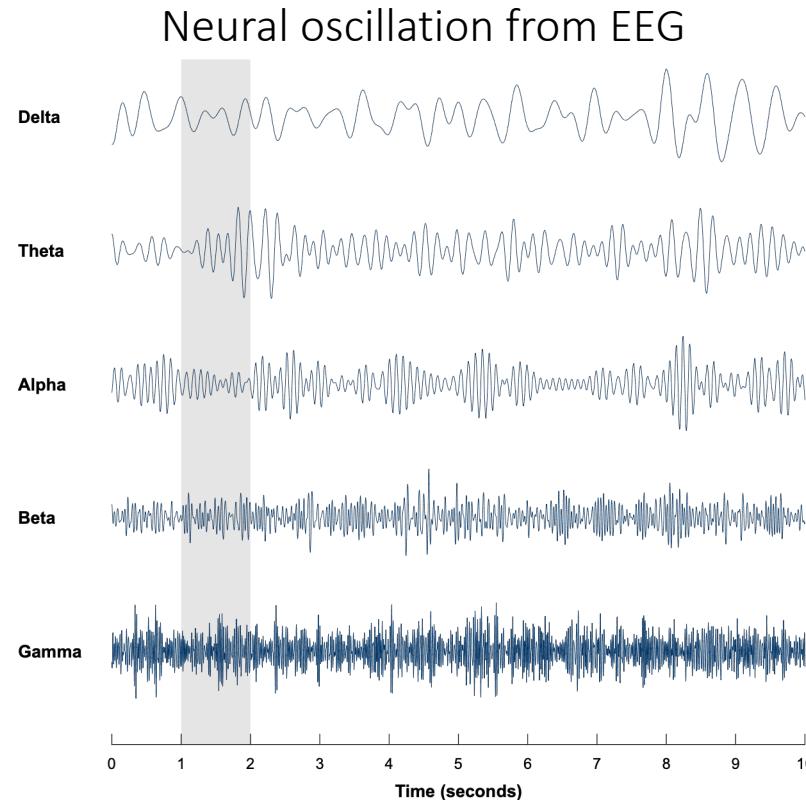
At the bottom of each section is a blue button labeled "Examples".

# Spectral analysis

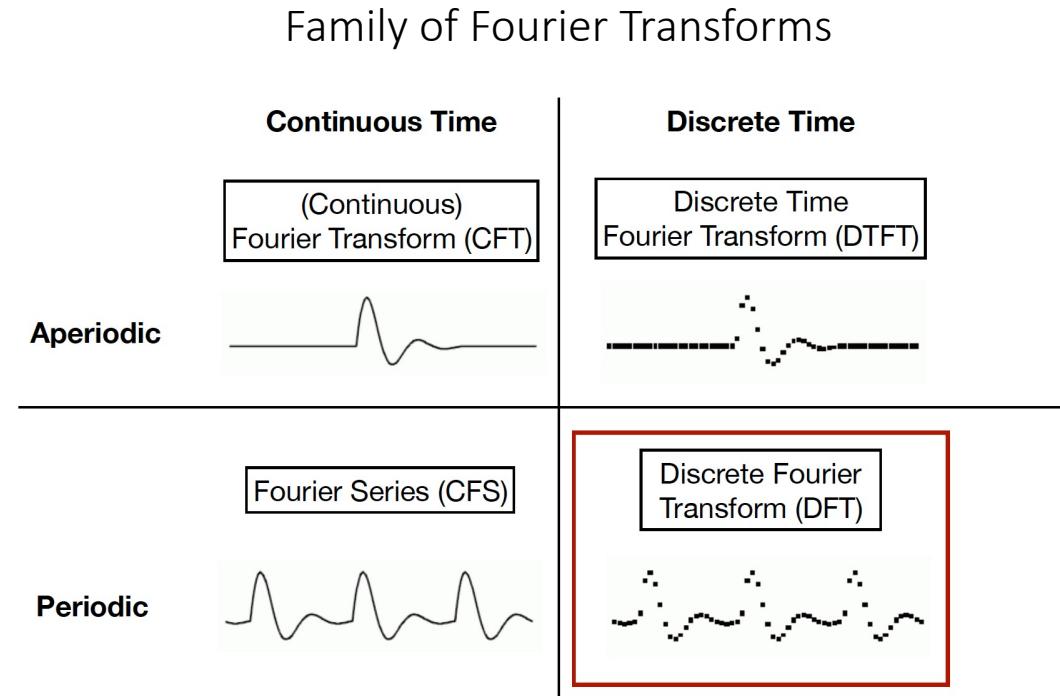
- Fourier transform



“Fourier transform is a mathematical transform that decomposes functions depending on space or time into functions depending on spatial or temporal frequency.”



[https://en.wikipedia.org/wiki/Neural\\_oscillation](https://en.wikipedia.org/wiki/Neural_oscillation)



# Spectral analysis

- Discrete Fourier transform (DFT) transforms a sequence of  $N$  complex numbers  $\{x_n\}$  to another sequence of complex numbers  $\{X_k\}$ :

Seq1:  $\{x_n\} := x_0, x_1, \dots, x_{N-1}$       ***n* is time**

DFT  
→

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N} kn}$$

Euler's formula  
 $e^{ix} = \cos(x) + i \sin(x)$

Seq2:  $\{X_k\} := X_0, X_1, \dots, X_{N-1}$       ***k* is frequency**

$$= \sum_{n=0}^{N-1} x_n \cdot \left[ \cos\left(\frac{2\pi}{N} kn\right) - i \cdot \sin\left(\frac{2\pi}{N} kn\right) \right]$$

- Both  $k$  and  $n$  range between  $0$  and  $N-1$ .
- As many frequency points as time points!

- Parseval's theorem: Sum (or integral) of the square of a function is equal to the sum (or integral) of the square of its Fourier Transform. For DFT:

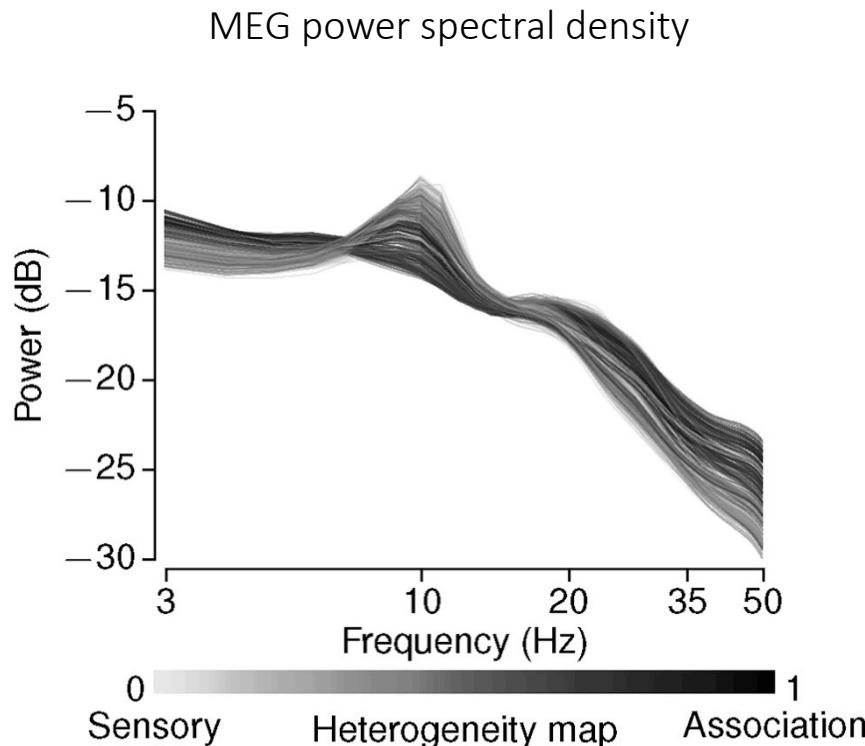
$$\sum_{n=0}^{N-1} |x_n|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X_k|^2$$

- If  $x_t$  measures voltage of a source in time  $t$ , power of  $x_t$  will be  $(x_t)^2$ .
- Thus, power is conserved in time and frequency domain!

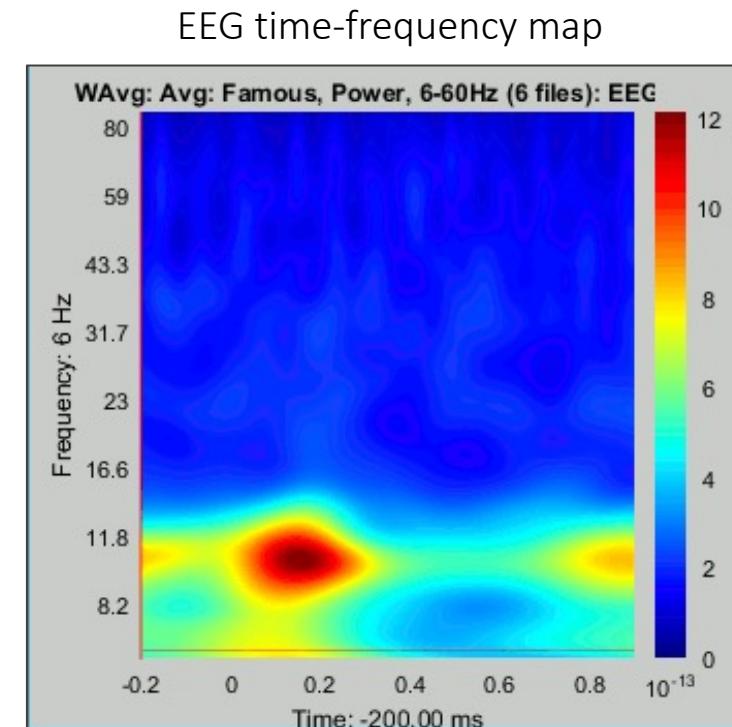
- We can study power of a signal in frequency domain using Fourier Transform! (which is Power spectral analysis!)

# Power spectral density

- There are different methods to measure power spectral density of a signal:
  - Welch's method: segments of time series with overlapping windows
  - Bartlet's method: segments of time series with non-overlapping windows



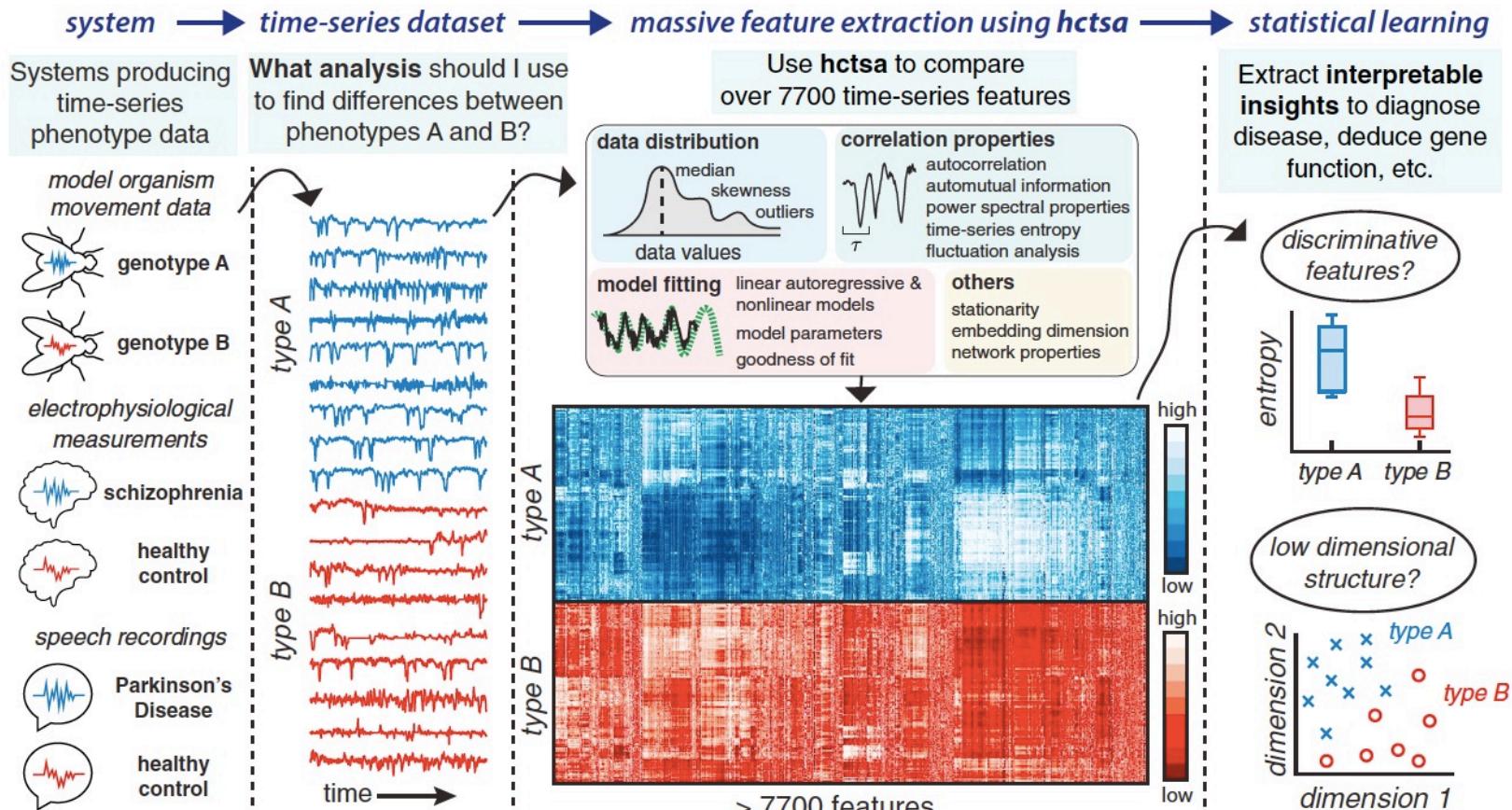
Demirtas et al., 2019, *Neuron*



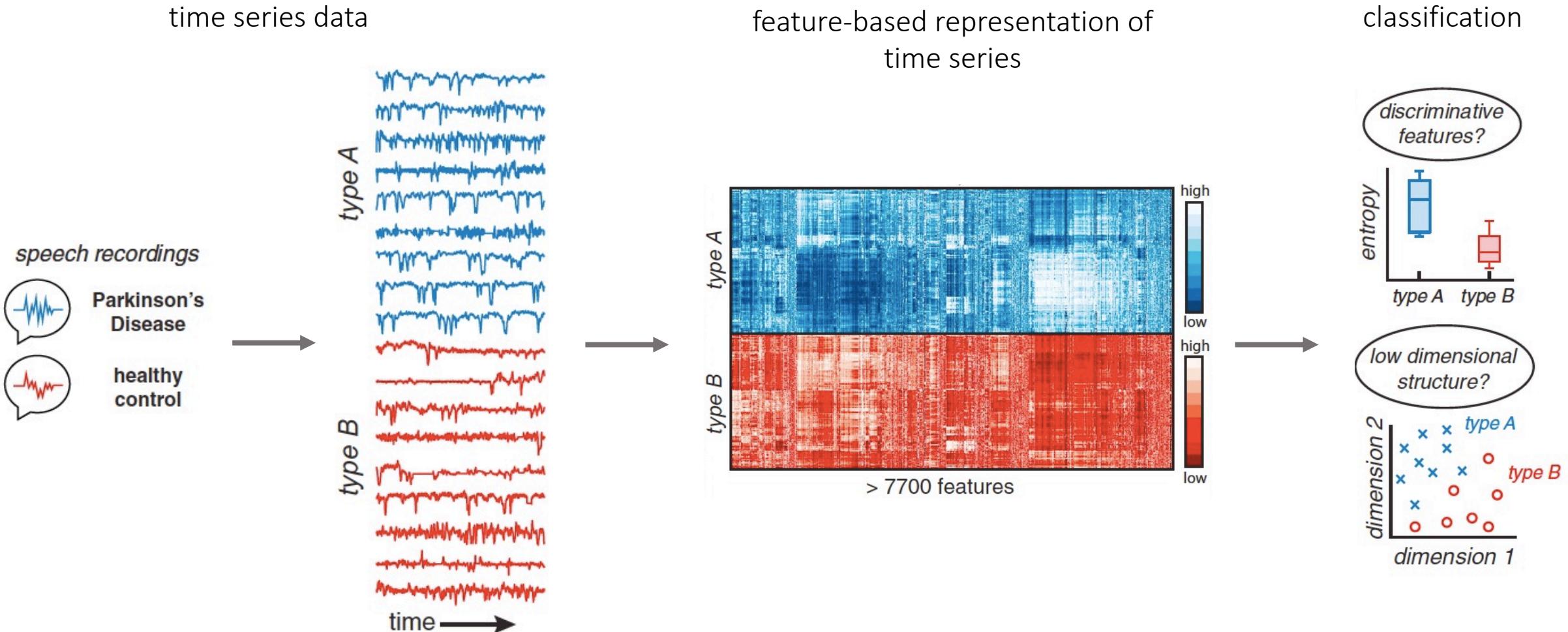
Brainstorm software package  
<https://neuroimage.usc.edu/brainstorm/>

# Time series phenotyping

- Feature-based representation of time series
- Extract several features from a time series
- Highly comparative time series analysis toolbox (hctsa)

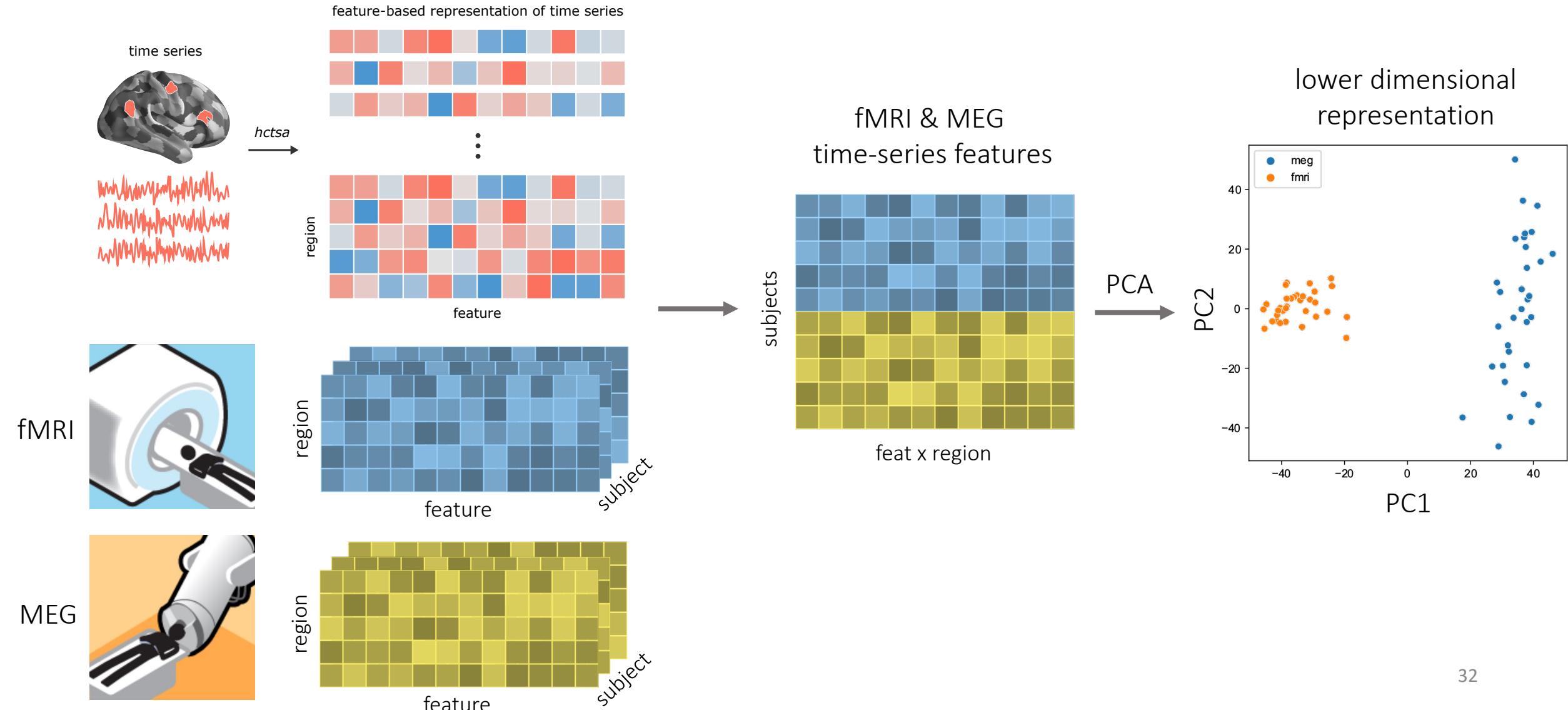


# Time series classification with features



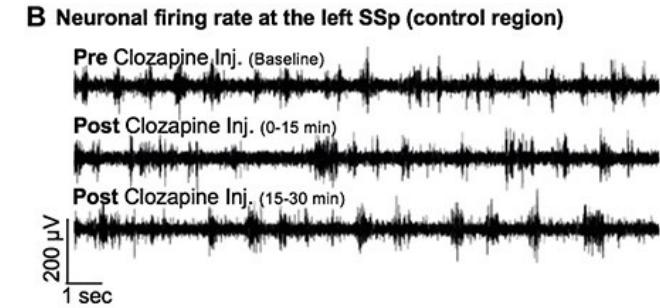
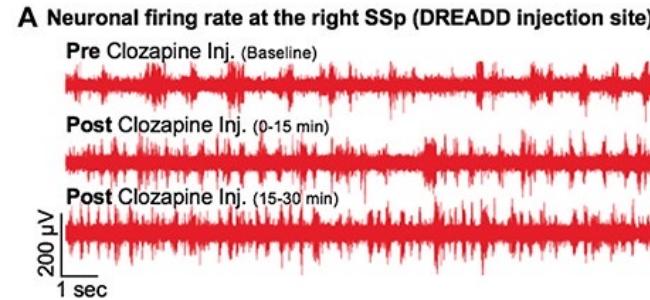
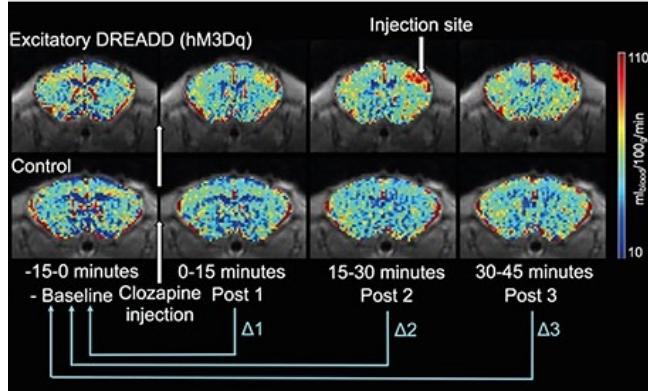
# Time series classification with features

- Feature-based representation of time series from two imaging modalities



# Time series classification with features

- Cortical E:I imbalance with designer receptor exclusively activated by designer drug (DREADD) injection



feature-based time-series representation

