Brief Reference for Python Functions and Techniques

This document brings together all Python functions, libraries, custom methods, and programming techniques utilized in the given scripts. It acts as a resource for developers seeking to comprehend and repurpose these components.

**General Libraries**

The following libraries are used across the scripts for signal processing, data visualization, and machine learning:

- os: Provides functions to interact with the operating system.

- numpy: Used for numerical operations and handling arrays.

- matplotlib.pyplot: For plotting and visualizing data.

- sklearn.neural\_network.MLPClassifier: Implements the multi-layer perceptron (MLP) classifier.

- joblib: For saving and loading trained models.

## Custom Functions

Here are the custom functions established within the scripts, including their objectives and parameters:

### despiking\_rolling\_window

Detects spikes in data using a rolling window and thresholds based on mean and standard deviation.

### despiking\_overlapping\_window

Detects spikes using overlapping windows and thresholds based on mean and standard deviation.

### despiking\_median\_iqr

Purpose: Detects spikes using median and interquartile range (IQR).

### interpolation

Interpolates over detected spikes to smooth the signal.

### data\_fitting

Fits a given function to the data using a sliding window.

### z\_score\_filter

Filters data using z-scores to identify and remove outliers.

### data\_visualization

Visualization of the data . Type of plots generated : timeseries, scatter, histogram, boxplot.

## Programming Techniques

The scripts utilize various programming techniques for signal processing, data cleaning, and machine learning. Below are some notable methods:

- Using sliding windows for rolling statistics and spike detection.

- Applying machine learning models (MLPClassifier and MLPRegressor) for classification and regression tasks.

- Visualizing data with matplotlib to analyze signals and results.

- Interpolating missing or anomalous values for signal smoothing.

**Signal Processing Functions**

**1. Test Signal Creation**

Generates a test signal consisting of a sine wave with added noise and random spikes.

Key Functions:

- np.arange(): Creates a sequence of time samples.

- np.sin(2 \* np.pi \* t / 500): Generates a sine wave signal.

- np.random.randn(): Adds Gaussian noise to the signal.

- np.random.permutation() Selects random indices for spikes.

**2. Spike Detection**

EXAMPLE OF THE MOST EFFECTIVE METHOD:

### despiking\_median\_iqr function:

Detects spikes in the signal using a sliding window to compute the median and interquartile range (IQR).

Key Functions:

- np.median(): Calculates the median of the sliding window.

- np.percentile(): Computes the 25th and 75th percentiles to determine the IQR.

- np.delete(): Removes detected spikes from the signal.

**3. Data Visualization**

Visualizes the original signal, detected spikes, and cleaned data.

Key Functions:

- plt.plot() : plots the data

-data\_visualization : see custom functions

## Neural Network Functions

### 1. Training the Model

Trains an MLPClassifier to distinguish between spikes and clean signals.

Key Functions and Parameters:

- np.vstack(): Combines signal values and time values into a single input array.

- np.hstack(): Concatenates spike and clean labels into a single output array.

- MLPClassifier(): Initializes the neural network with specific architecture.

- DNN.fit(input\_data, output\_data): Trains the neural network using the prepared input and output data.

### 2. Saving and Loading the Model

Saves and loads the trained neural network model using joblib.

- joblib.dump(): Saves the model to 'something.pkl'.

- something = joblib.load(): Loads the saved model.

### 3. Making Predictions

Uses the trained model to classify signal values and identify spikes.

Key Functions and Steps:

- R.predict(input\_data): Predicts whether each data point is a spike or clean signal.

“[More data beats clever algorithms, but better data beats more data.](https://www.azquotes.com/quote/1020712)”

-[**Peter Norvig**](https://www.azquotes.com/author/49745-Peter_Norvig)