ChronoXtract: A python library for time series feature extraction

Aman Kumar

Abstract

Contents

1 Introduction

Time series or sequential data are ubiquitous in modern applications, covering finance, healthcare, climate science, astrophysics, particle physics and the Internet of Things (IoT). Extracting meaningful features from such data is critical for anomaly detection, forecasting, and classification tasks. Such meaningful features are hard to identify that can summarise the long-term and short-term dynamics contained in the time series/ sequential data.

Statistical feature extraction serves as a foundational step in transforming raw time series data into structured, interpretable representations that reveal underlying patterns, trends, and anomalies. These features act as concise numerical summaries of complex temporal dynamics, enabling downstream tasks such as classification, forecasting, anomaly detection, and model training. For instance, basic statistical measures—such as the mean , median , and mode —capture central tendencies and distributional properties, while metrics like fractional variability (the ratio of the range to the mean) quantify volatility or irregularity in the data. Such metrics are indispensable for identifying outliers, assessing stability, or comparing datasets across domains. Beyond descriptive statistics, spectral features derived from methods like Fourier analysis provide insights into periodicity and cyclical behavior, which are critical for applications such as signal processing, climate modeling, and sensor data analysis. Fourier coefficients, for example, decompose time series into constituent frequencies, enabling the detection of hidden periodic patterns or noise components. Together, these statistical and spectral features form a versatile toolkit for distilling actionable insights from raw data, bridging the gap between raw observations and higher-level analytical tasks. In the next section, we describe the details about the statistical features included in the *ChronoXtract* library.

2 Statistical Features

Here, we discuss the list of statistical features extracted by the ChronoXtract package.

2.1 Mean

The mean (or average) represents the central tendency of the time series data. It provides a single value summarizing the overall magnitude of the dataset. Mean is given by:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

Algorithm:

1. Sum all elements in the time series vector.
2. Divide the sum by the length of the vector.

2.2 Median

The median is the middle value of a sorted dataset. It is robust to outliers and provides a measure of central tendency that is less sensitive to extreme values than the mean.

$$\text{Median} = \begin{cases} \frac{n+1}{2}, & \text{if } n \text{ is odd} \\ \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2}, & \text{if } n \text{ is even} \end{cases} \tag{2}$$

Algorithm:

1. Sort the vector.
2. Compute the median based on whether the length of the vector is odd or even.

2.3 Mode

The mode is the most frequently occurring value in the dataset. It is useful for identifying dominant patterns or trends in categorical or discrete numerical data.

$$\text{Mode} = \text{argmax}\{\text{count}(x_i)\} \tag{3}$$

Algorithm:

1. Use a HashMap to count occurrences of each value in the time series.
2. Iterate through the HashMap to find the key with the maximum count.

2.4 Variance

Variance measures the spread or dispersion of the data around the mean. A higher variance indicates greater variability in the dataset.

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2 \tag{4}$$

here μ is the mean of the dataset.

Algorithm:

1. Compute the mean of the time series.
2. Calculate the squared difference between each element and the mean.
3. Sum the squared differences and divide by the length of the vector.

2.5 Standard Deviation

Standard deviation is the square root of the variance. It provides a measure of the dispersion of the data around the mean. A higher standard deviation indicates greater variability in the dataset.

$$\sigma = \sqrt{\sigma^2} \tag{5}$$

Algorithm:

1. Compute the variance of the time series.
2. Take the square root of the variance.

2.6 Skewness

Skewness measures the asymmetry of the dataset. A positive skewness indicates a longer tail on the right side of the distribution, while a negative skewness indicates a longer tail on the left side.

$$\text{Skewness} = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^3}{\sigma^3} \tag{6}$$

Algorithm:

1. Compute the mean and standard deviation of the time series.
2. Calculate the skewness using the formula above.

2.7 Kurtosis

Kurtosis measures the tailedness of the dataset. A higher kurtosis indicates heavier tails in the distribution, while a lower kurtosis indicates lighter tails.

$$\text{Kurtosis} = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^4}{\sigma^4} \tag{7}$$

Algorithm:

1. Compute the mean and standard deviation of the time series.

2. Calculate the kurtosis using the formula above.

2.8 Min Max Range

The min-max range is the difference between the maximum and minimum values in the dataset. It provides a measure of the spread of the data.

$$\text{min-max range} = \max(x) - \min(x) \tag{8}$$

Algorithm:

1. Find the maximum and minimum values in the time series.
2. Calculate the difference between the maximum and minimum values.

2.9 Quantiles

Quantiles divide the dataset into equal parts. The median is the 50th percentile, while the quartiles divide the dataset into four equal parts. Quantiles provide a measure of the spread and distribution of the data.

Algorithm:

1. Sort the time series.
2. Calculate the quantiles based on the desired percentage.

2.10 Sum

The sum of the dataset provides a measure of the total magnitude of the data.

$$\text{Sum} = \sum_{i=1}^{N} x_i \tag{9}$$

Algorithm:

1. Sum all elements in the time series vector.

2.11 Absolute Energy

Absolute energy is the sum of the squared values in the dataset. It provides a measure of the total energy in the data.

$$\text{Absolute Energy} = \sum_{i=1}^{N} x_i^2 \tag{10}$$

Algorithm:

1. Square each element in the time series.
2. Sum the squared values.

References