SDA-CaseStudy-TimeSeries

Usage of TimeSeriesHandler.py

```
usage: TimeSeriesHandler.py [-h] [--input <filename>] [--output <filename>] [--
plot] [--iqr] [--std] [--s <s>] [--log]

optional arguments:
  -h, --help           show this help message and exit
  --input <filename>   Specify the path to the input-file
  --output <filename>  Specify the path to the output-file
  --plot               Show Plot (default: disabled)
  --iqr                Use IQR for outlier removal (default: enabled)
  --std                Use Z-Score for outlier removal (default: disabled)
  --s <s>              Z-Score for outlier detection (default: 3)
  --log                Show detailed logs (default: disabled)
```

Examples

1. Example how to start TimeSeriesHandler.py with Input-file `input.log`, Output-file `output.log`, deactivated plot (plot.png will be saved in the same directory), Interquartile range `--iqr` for outlier removal is activated in default:

```
python.exe TimeSeriesHandler.py --input input.log --output output.log
```

2. Example how to start TimeSeriesHandler.py with Input-file `input.log`, Output-file `output.log`, activated plot (plot.png will also be saved in the same directory), activated Standard deviation with 2 Standard deviations `--std 2` and detailed logs `--log`:

```
python.exe TimeSeriesHandler.py --input input.log --output output.log --plot --std
--s 2 --log
```

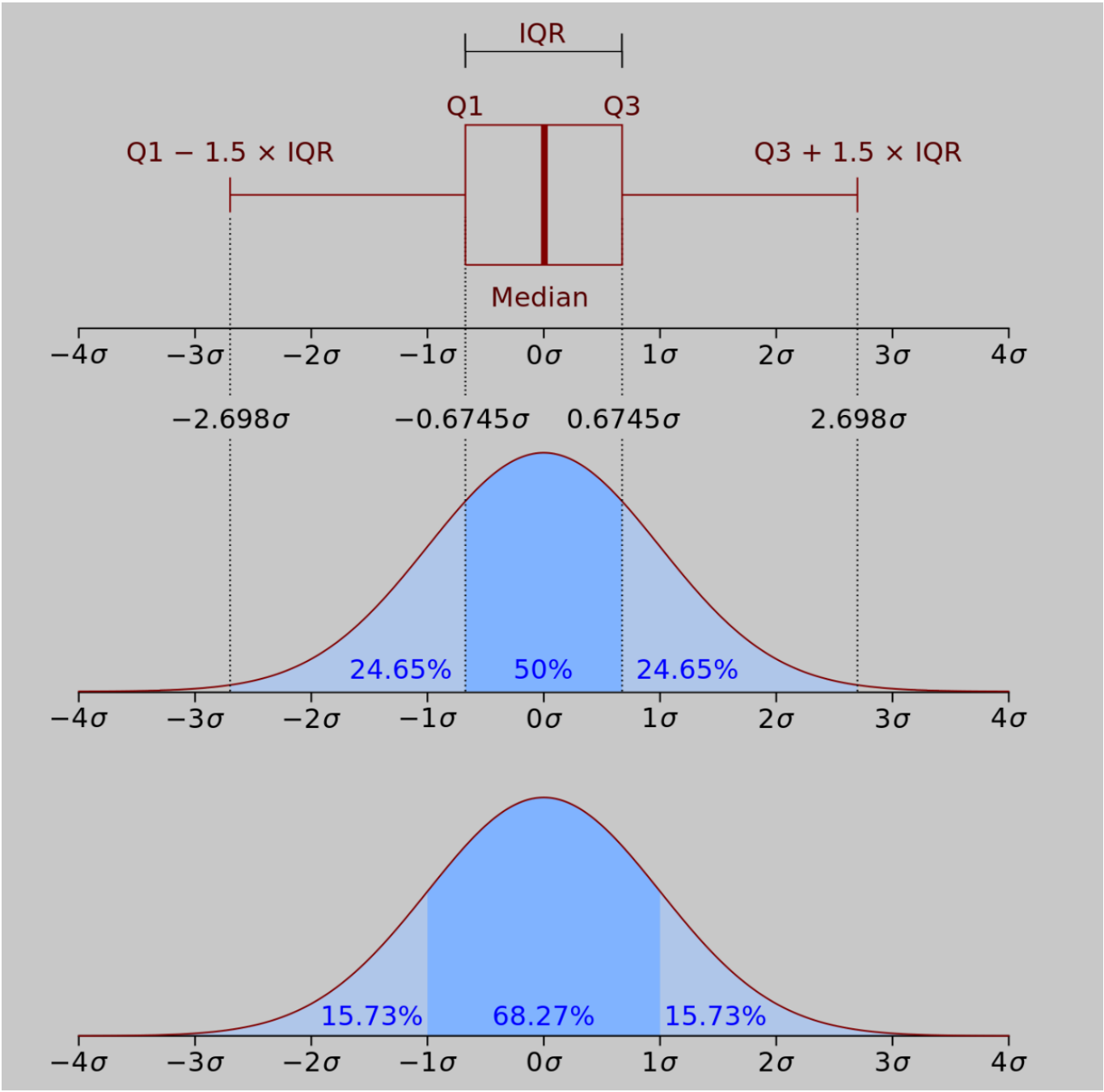
Overview Methods

1. **open_file():** Creates Dataframe from the csv- or log-file in the specified path.
2. **rename_columns():** Renames the columns in the dataframe.
3. **create_datetime():** Creates pandas Datetime in new column. Drops columns Date and Time.
4. **get_first_valid_timestamp():** Gets the first valid timestamp of the Dataframe.
5. **get_last_valid_timestamp():** Gets the last valid timestamp of the Dataframe.
6. **calculate_mean_timegap():** Calculates the mean timegap between timestamps.
7. **check_valid_date():** Checks if dates are valid. Changes invalid dates to NaT.

8. **replace_nat():** Checks the dataframe for NaT. Replaces all NaT / invalid timestamps. Uses the mean timegap for calculations.
9. **format_data_columns():** Replacing Strings in Temp and Hum. Drops column TO. Converts values to float. Replaces empty string with np.nan. Creates NaN Index.
10. **check_valid_value():** Checks if the values of Temp and Hum are in a valid range. Invalid values are replaced with NaN.
11. **interpolate_nan():** Interpolates NaN values of Temp and Hum.
12. **remove_outliers():** Identifies and removes outliers. Works for Standard deviation (Z-Score) and for Interquatrile Range.
13. **plot_data():** Creates Boxplots and Lineplots for Time series Temp and Hum. For a better data comparison two dataframes are compared to each other (before and after outlier removal).
14. **export_file():** Exports Dateframe to File in the specified path.

Statistical Background: IQR, SD and Z-Score

Boxplot (with an interquartile range) and a probability density function (pdf) of a Normal N(0,σ2) Population:



Interquartile Range

In descriptive statistics, the interquartile range (IQR) is a measure of statistical dispersion, which is the spread of the data. It is defined as the difference between the 75th and 25th percentiles of the data. These quartiles are denoted by Q1 (also called the lower quartile), Q2 (the median), and Q3 (also called the upper quartile). The lower quartile corresponds with the 25th percentile and the upper quartile corresponds with the 75th percentile, so IQR = Q3 − Q1. Following steps have to be followed:

- Find the first quartile, `Q1`.
- Find the third quartile, `Q3`.
- Calculate the IQR. `IQR = Q3 - Q1`.
- Define the normal data range with lower limit as `Q1 - 1.5 * IQR` and upper limit as `Q3 + 1.5 * IQR`.
- Any data point outside this range is considered as outlier and should be removed for further analysis.
- In boxplot, this IQR method is implemented to detect any extreme data points where the maximum point (the end of high whisker) is `Q3 + 1.5 * IQR` and the minimum point (the start of low whisker) is `Q1 - 1.5 * IQR`.

Standard deviation

Standard deviation method is similar to IQR procedure. Depending on the set limit either at 2 times stdev or 3 times stdev, we can detect and remove outliers from the dataset.

$$ Upperlimit = { mean + 3 * stdev } $$

$$ Lowerlimit = { mean - 3 * stdev } $$

Z-score is used to convert the data into another dataset with mean = 0. Here, $\bar x$ is the mean value and s is standard deviation. Once the data is converted, the center becomes 0 and the z-score corresponding to each data point represents the distance from the center in terms of standard deviation. For example, a z-score of 2.5 indicates that the data point is 2.5 standard deviation away from the mean. Usually z-score = 3 is considered as a cut-off value to set the limit. Therefore, any z-score greater than +3 or less than -3 is considered as outlier which is pretty much similar to standard deviation method:

$$ Z = {x_{i} - \bar x \over s} $$