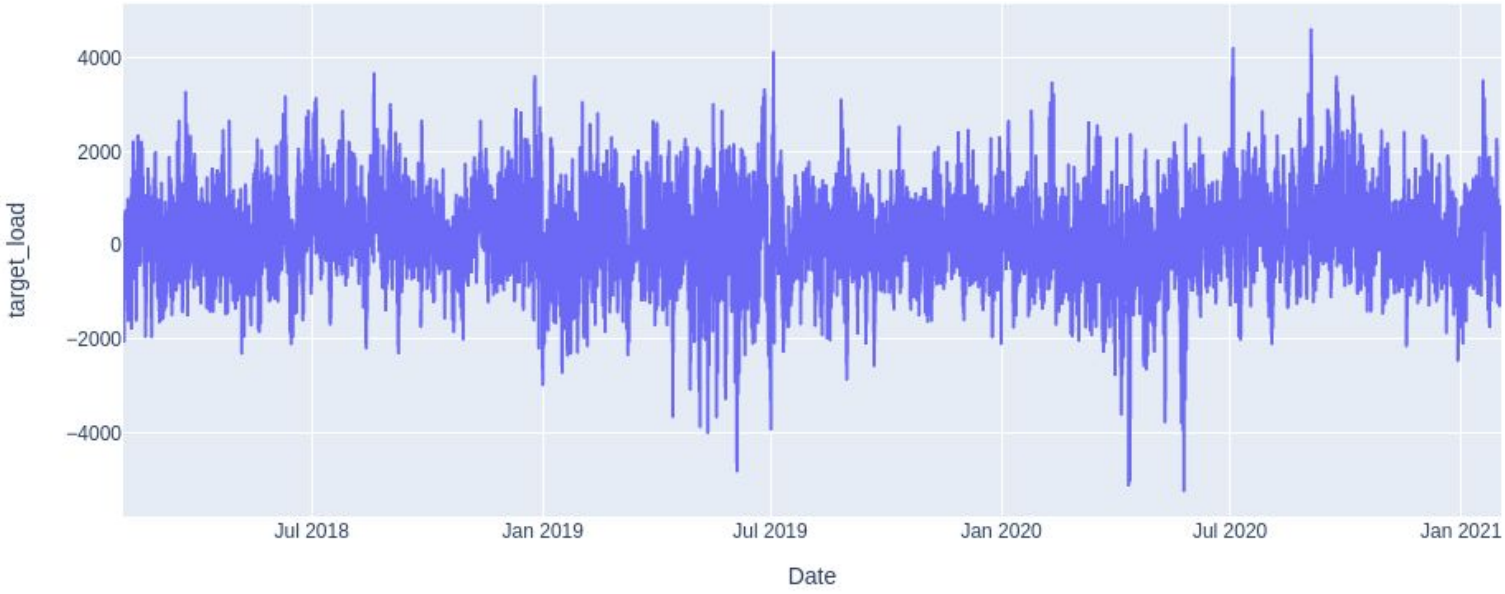
Explanatory data analysis(EDA)

Exploratory data analysis is a method by which data is analyzed, and the main characteristics are summarized. Traditionally, various statistical graphs and data visualization methods are used, EDA is mostly used to observe trends, seasonalities, and different features formal hypothesis testing and modeling fail to do. EDA can provide analysts a more intuitive grasp of the data and allow them to make sense of the data's behavior. Since time series data is always 2-dimensional, a simple plot of the data can prove quite useful.

The most widely used graphical techniques:

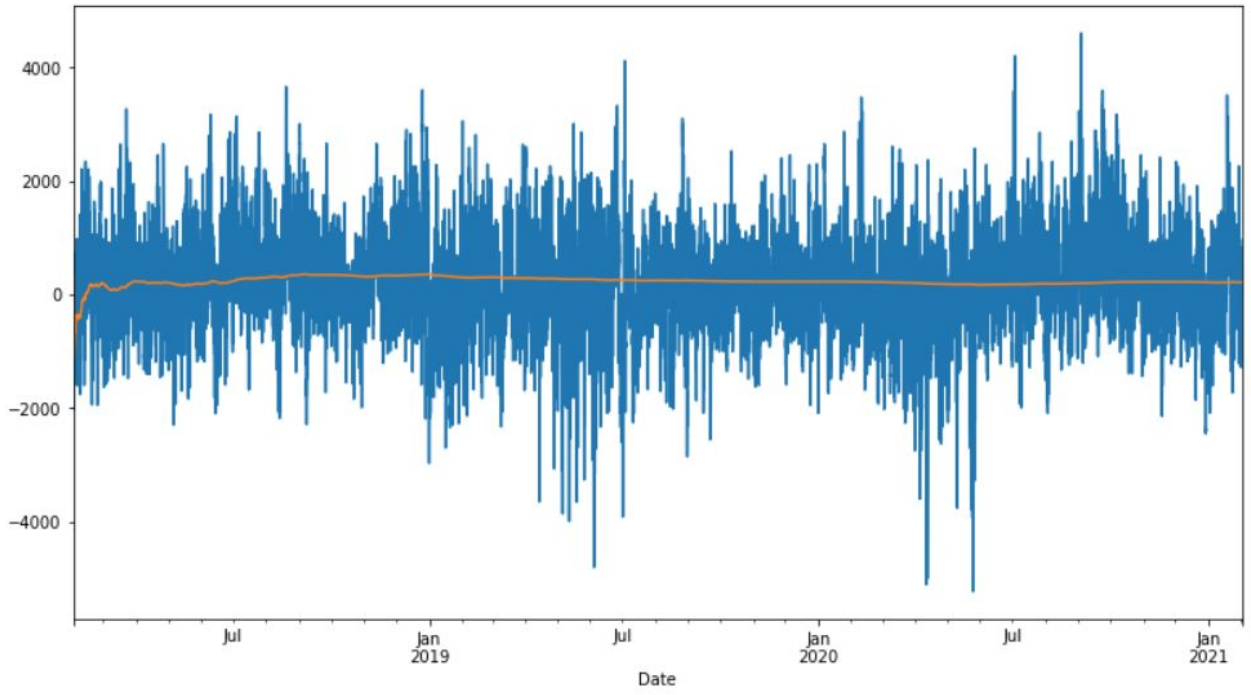
1. Box plot
2. Histogram
3. Scatter plot
4. Line plots

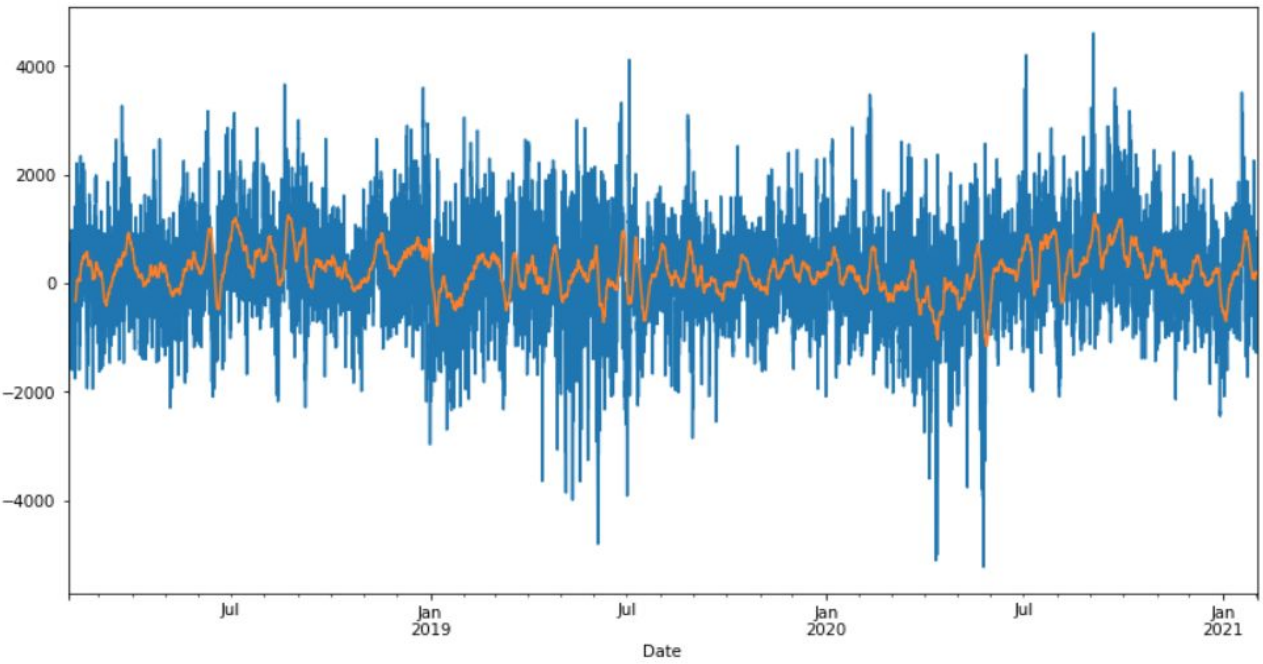
In this section, we will provide an overview of some of the analyses conducted. Firstly, we shall consider the topology of the 'load' variable across time. Please note that our data range is from February-2018 to march 2021; the data has an hourly frequency.



For a better understanding, let's zoom into the data. An interactive version of the plot above is available on our GitHub page.

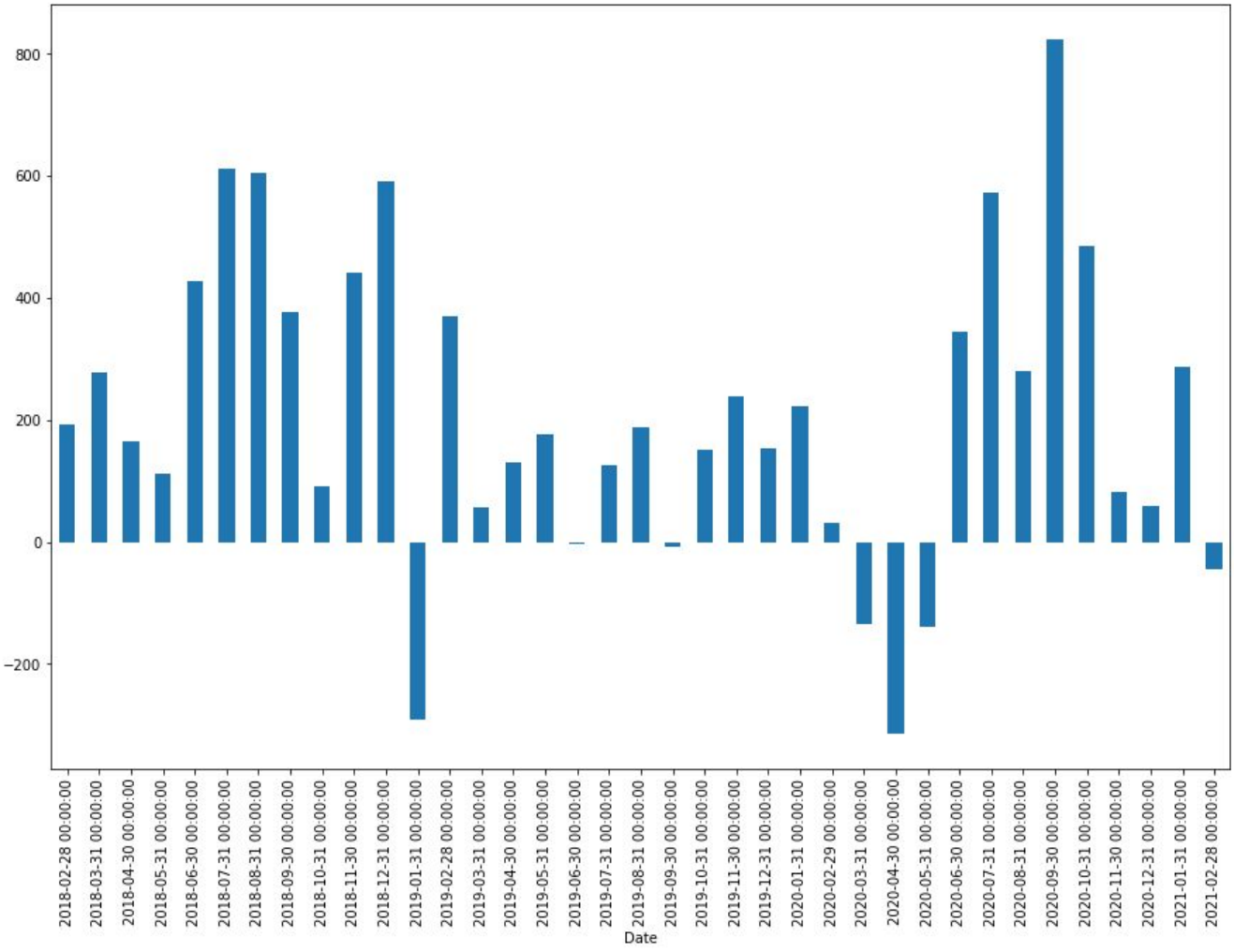
A rolling average is an important graph that allows us to determine whether the time series is above or below the cumulative average at any given point. In our case, we used both cumulative and weakly cumulative averages. The graphs are plotted respectively:





As seen from the graphs above, we observe no trend in the data, i.e the mean and standard deviation are both constants across time. Having stationary data will prove useful in the later sections as ARIMA methods always work on stationary data. In the later sections, we will do a formal analysis to prove the data's stationarity.

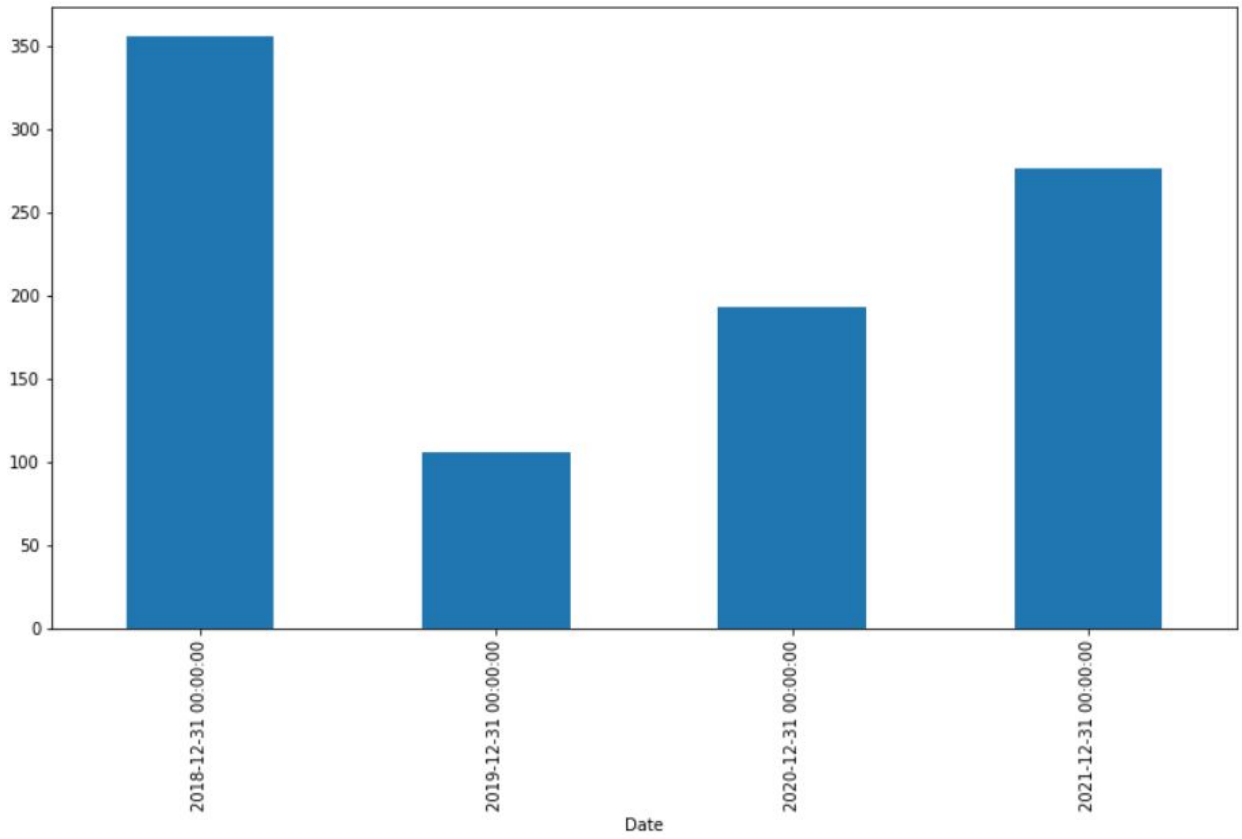
For a data set, the mean, conceptually known as the expected value is by far the most widely used statistic. The mean of a dataset will provide us with a geometric understanding of how the data behaves. The following graph will illustrate the monthly averages of the 'net_load' variable. We made use of the resampling method provided by the Pandas library to conduct this analysis.



Since the graph might be hard to read we will tabulate our statistics in the following table:

Date	
2018-02-28	192.103065
2018-03-31	277.676855
2018-04-30	164.548500
2018-05-31	113.112137
2018-06-30	427.933722
2018-07-31	611.801989
2018-08-31	604.492339
2018-09-30	377.594556
2018-10-31	90.264234
2018-11-30	441.941222
2018-12-31	591.133642
2019-01-31	-292.022823
2019-02-28	371.057202
2019-03-31	55.947634
2019-04-30	131.395792
2019-05-31	176.267366
2019-06-30	-2.254667
2019-07-31	127.082796
2019-08-31	188.767917
2019-09-30	-8.159861
2019-10-31	150.632245
2019-11-30	238.842806
2019-12-31	153.349315
2020-01-31	221.984987
2020-02-29	30.938937
2020-03-31	-133.864180
2020-04-30	-314.091944
2020-05-31	-138.257419
2020-06-30	344.568472
2020-07-31	572.857661
2020-08-31	279.846277
2020-09-30	824.665250
2020-10-31	485.068427
2020-11-30	82.749292
2020-12-31	58.189651
2021-01-31	286.283226
2021-02-28	-45.507083

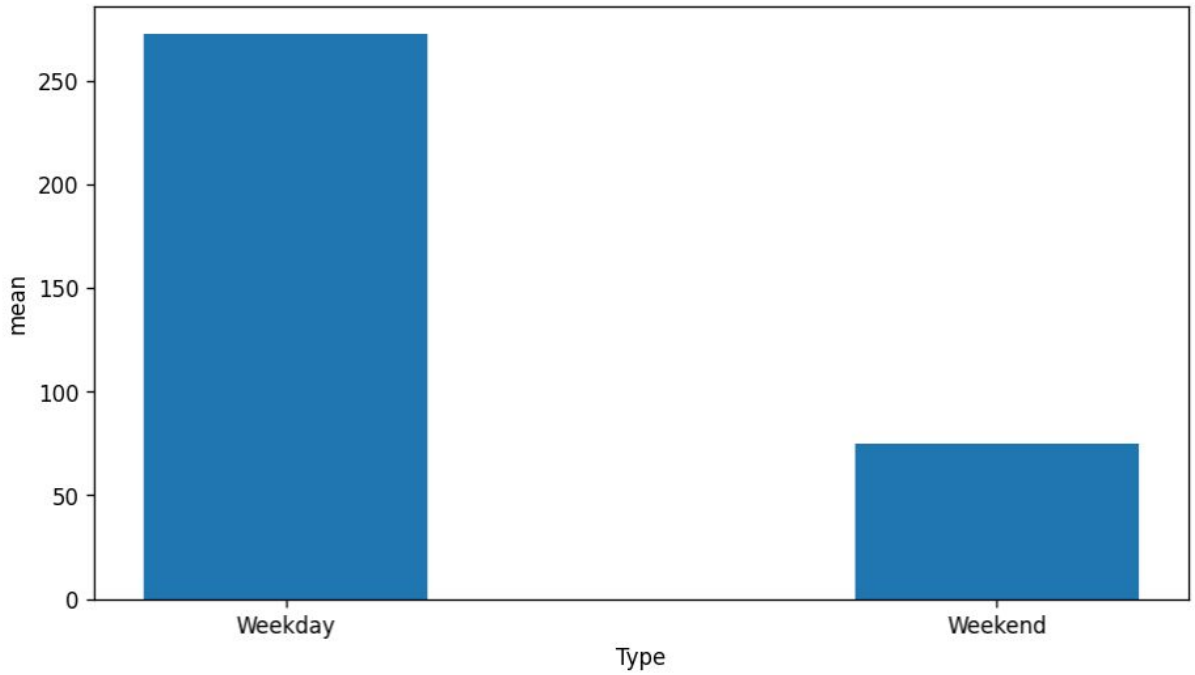
We also looked at the yearly averages and plotted their values:



2018-12-31	355.336356 kW
2019-12-31	105.602963 kW
2020-12-31	193.318439 kW
2021-12-31	275.914779 kW

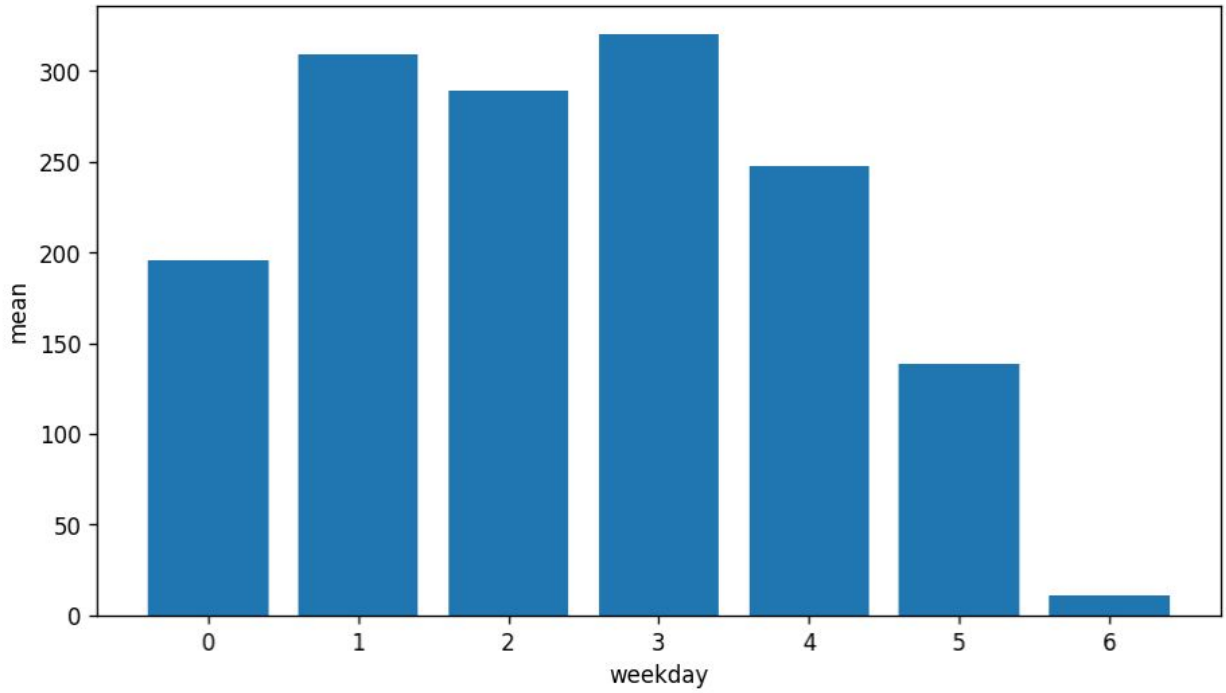
We can clearly observe that 2019 showed a trend towards down-regulation.

Another important statistic is to study the difference between weekdays and weekends. The following plot will illustrate the difference in the means between weekdays and weekends.



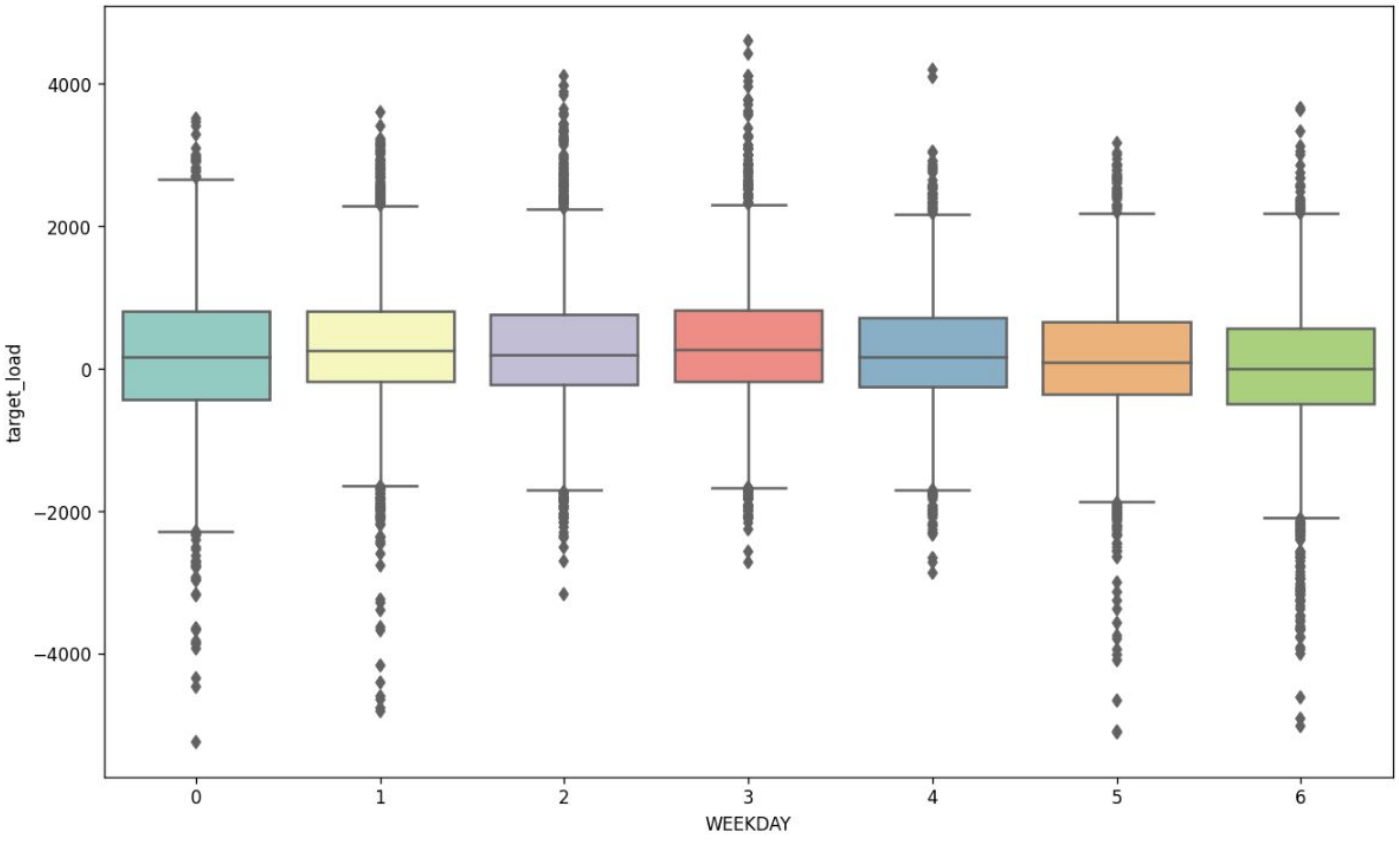
Weekday mean	272.37 kW
Weekend mean	74.97 kW

Now we will look at the days individually and report the daily mean statistics. Please note that the zeroth day is Monday.

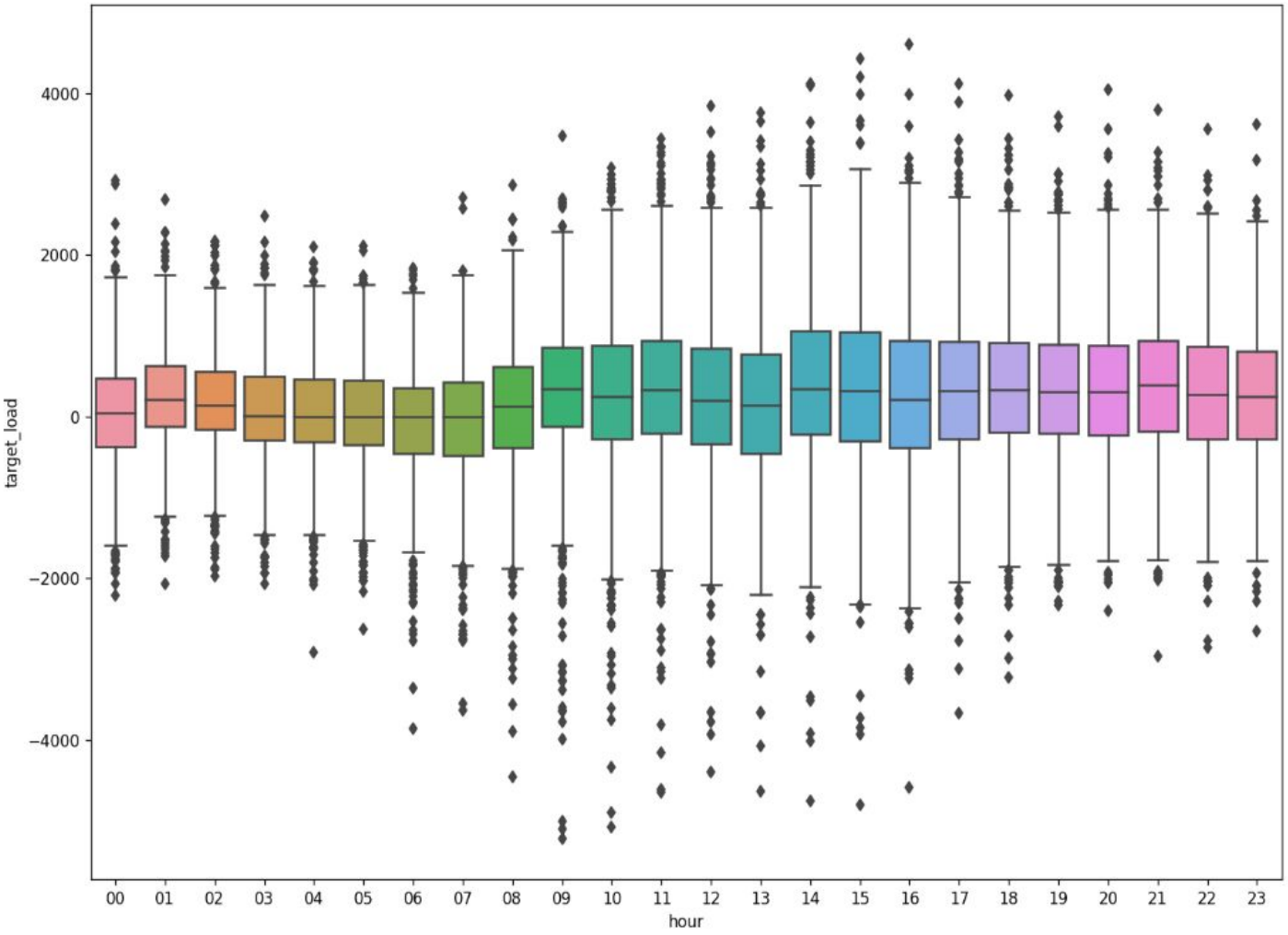


Monday	195.902 kW
Tuesday	309.522 kW
Wednesday	288.995 kW
Thursday	320.230 kW
Friday	247.559 kW
Saturday	138.727 kW
Sunday	11.225 kW

A box plot or boxplot is a tool for graphically representing clusters of numerical data by their quartiles in descriptive statistics. Lines extending from the boxes indicate variability beyond the upper and lower quartiles in box plots. The following graph will illustrate the data on a daily basis.

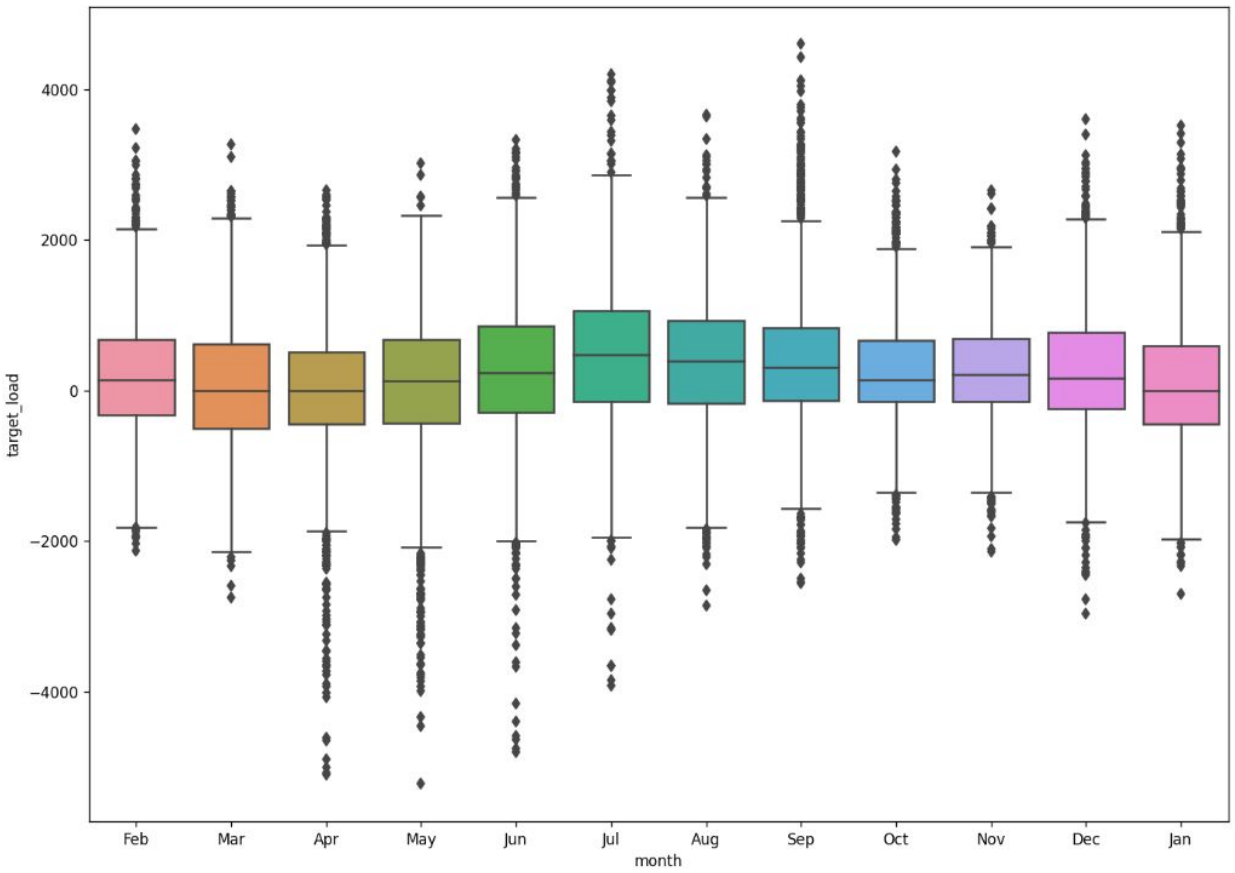


As of now, we have not looked at the data from an hourly perspective. The following graph will indicate the hourly averages in the data.

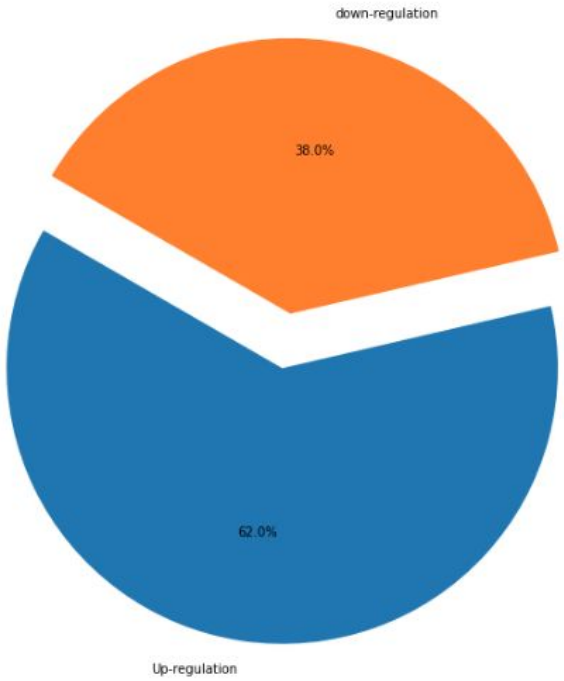


As seen in the data we can see that working hours have higher averages and the up-regulation peaks at 14:00

In our next illustration, we will group the data by months and display their respective box-plots

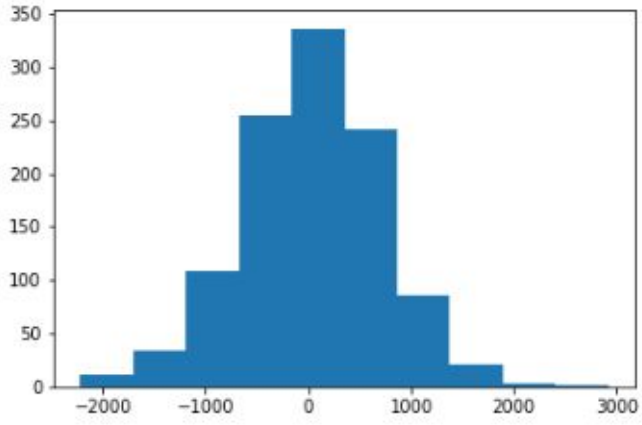


Since we also performed classification tasks, it is always a good idea to look at the distribution of the data by looking at the net_load variable's sign. The following pie-chart will illustrate the data's distribution.

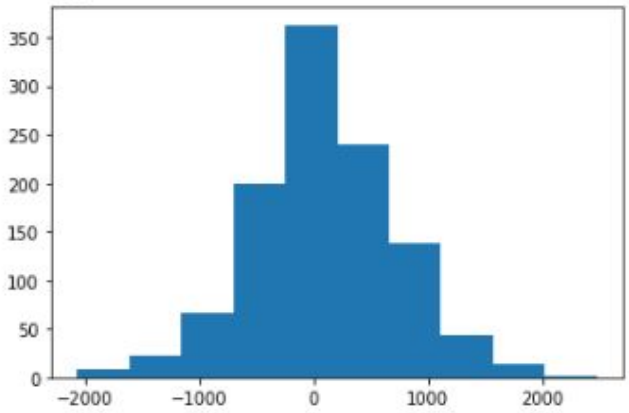


A histogram is a plot that lets you discover, and show, the underlying frequency distribution (shape) of a set of continuous data. This allows the inspection of the data for its underlying distribution (e.g., normal distribution), outliers, skewness. Since our analysis will be conducted on an hourly basis, we will start by showing the histograms of each hour. For the sake of brevity, we will not include all of our histograms. Monthly, weekday type and other plots will be provided on the jupyterLab file on Github

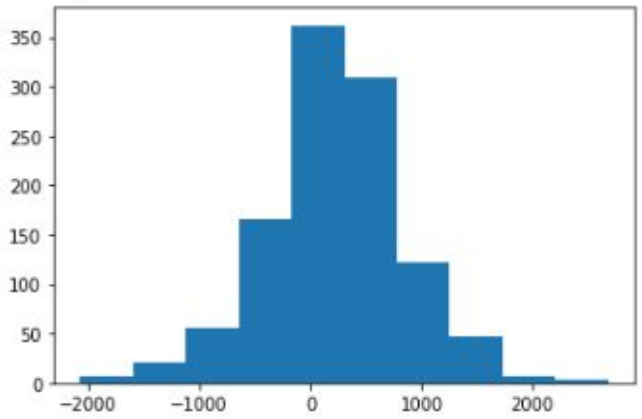
histogram of data at time: 00



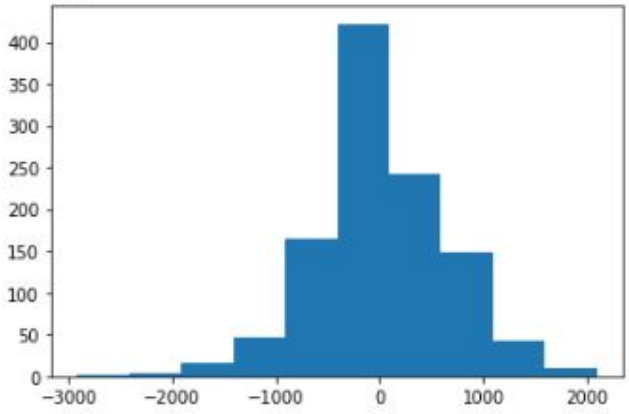
histogram of data at time: 03



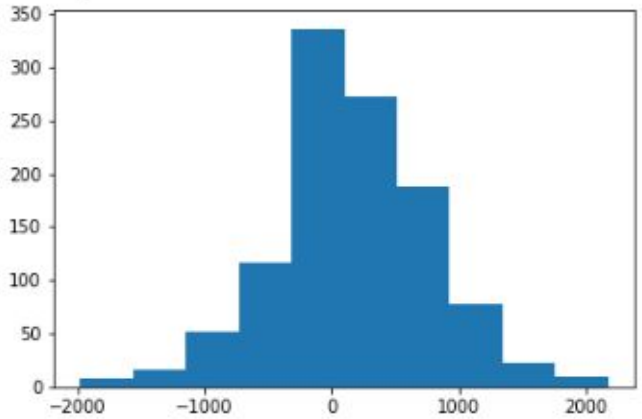
histogram of data at time: 01



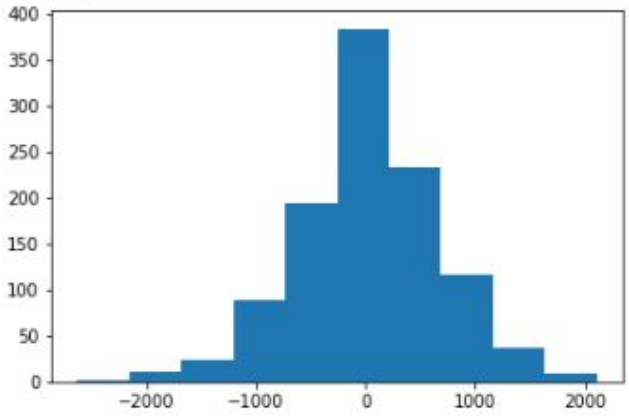
histogram of data at time: 04



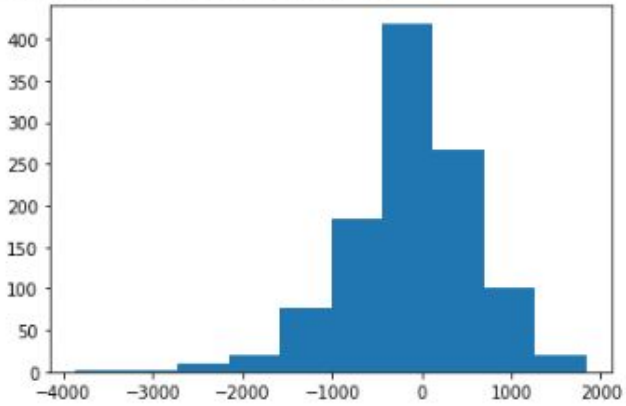
histogram of data at time: 02



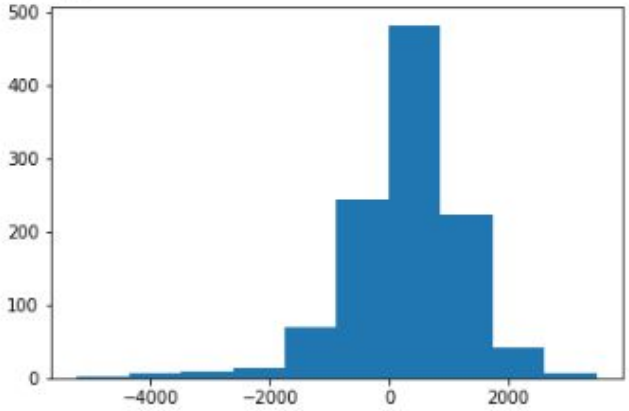
histogram of data at time: 05



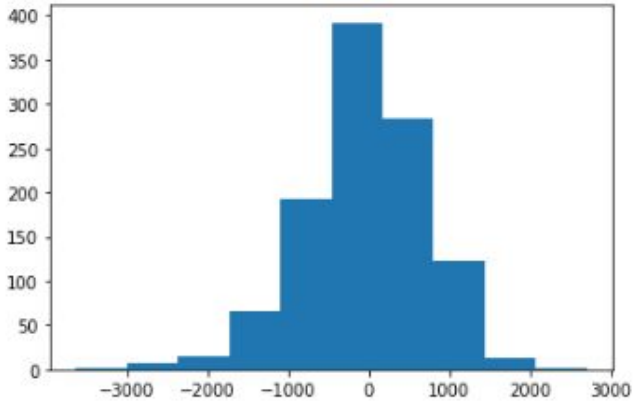
histogram of data at time: 06



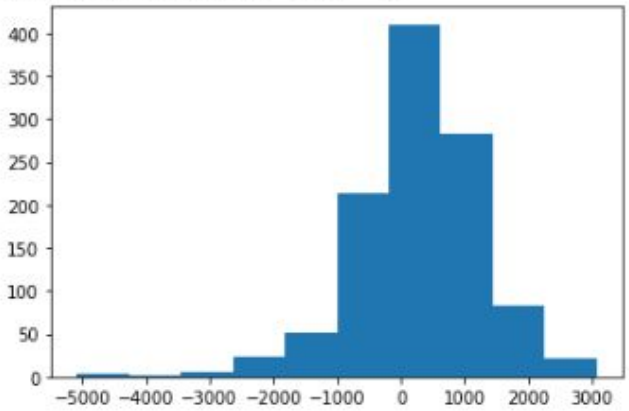
histogram of data at time: 09



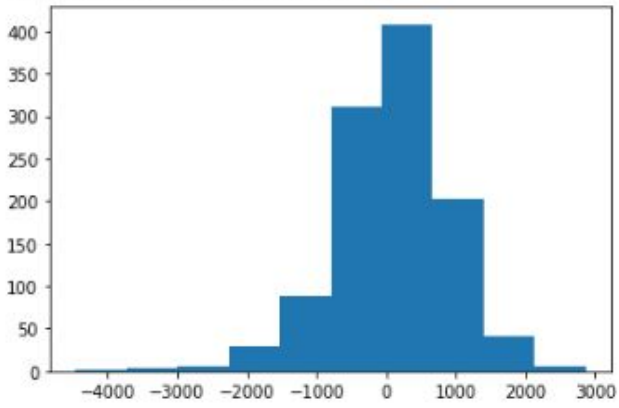
histogram of data at time: 07



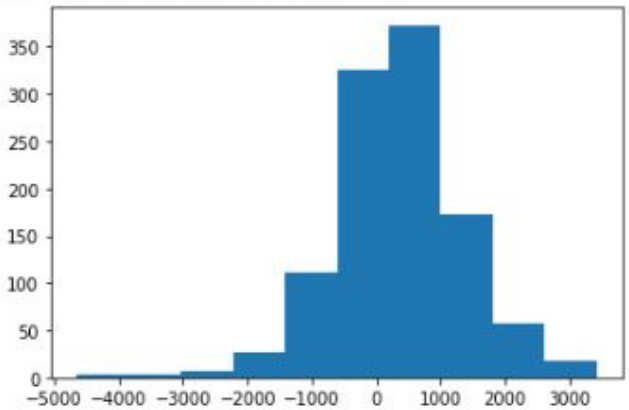
histogram of data at time: 10



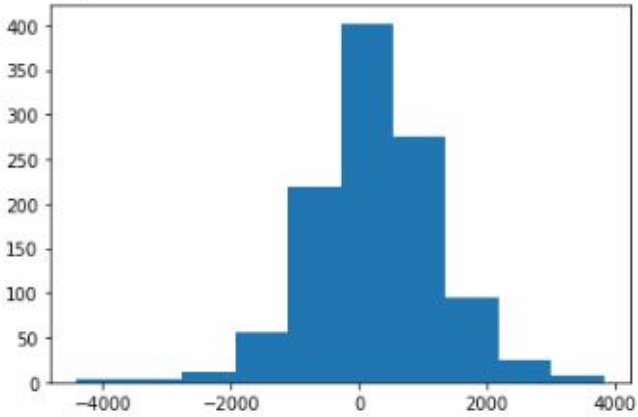
histogram of data at time: 08



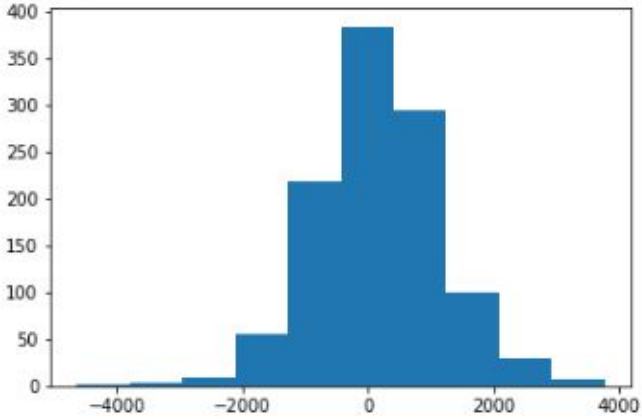
histogram of data at time: 11



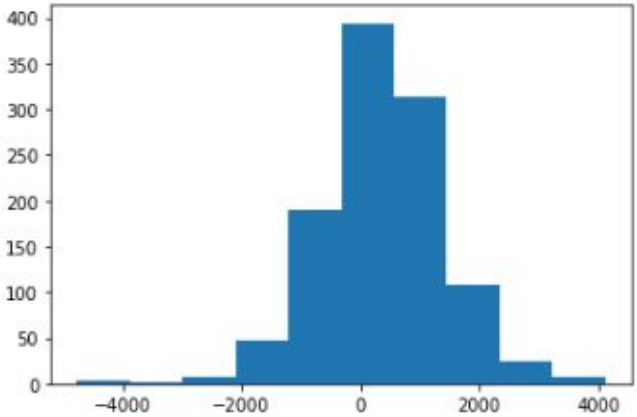
histogram of data at time: 12



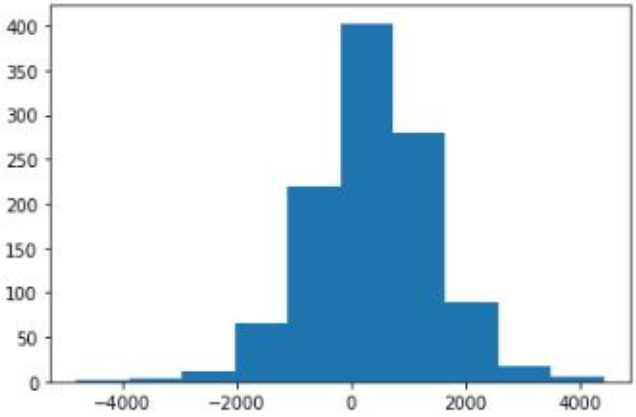
histogram of data at time: 13



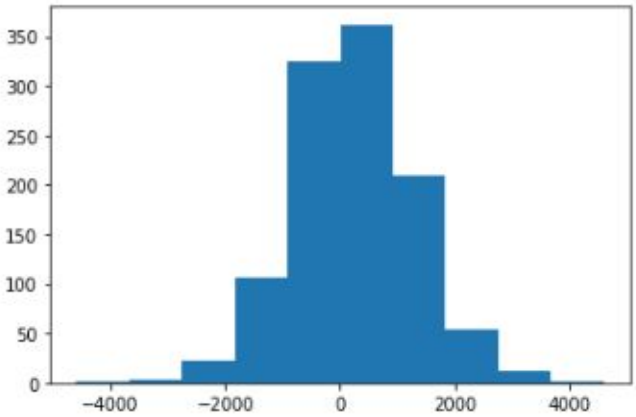
histogram of data at time: 14



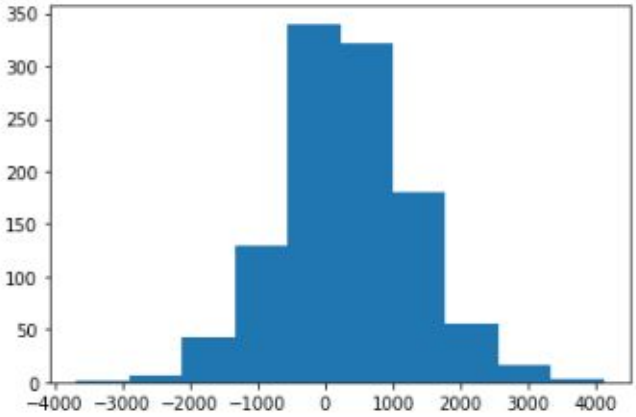
histogram of data at time: 15



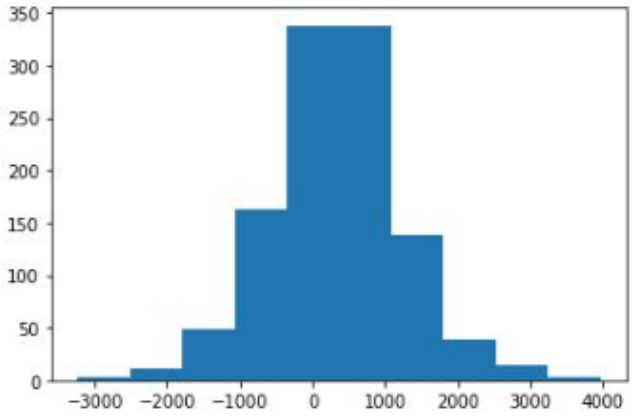
histogram of data at time: 16



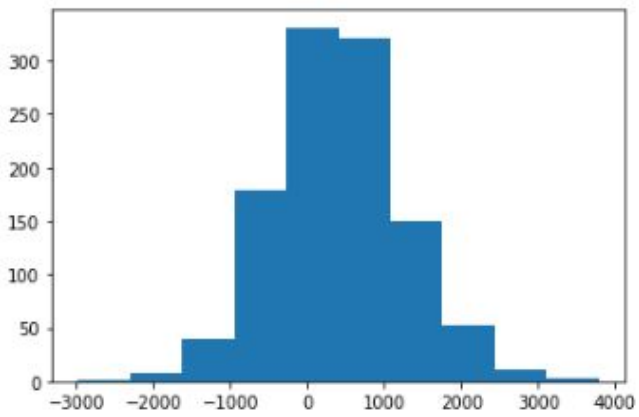
histogram of data at time: 17



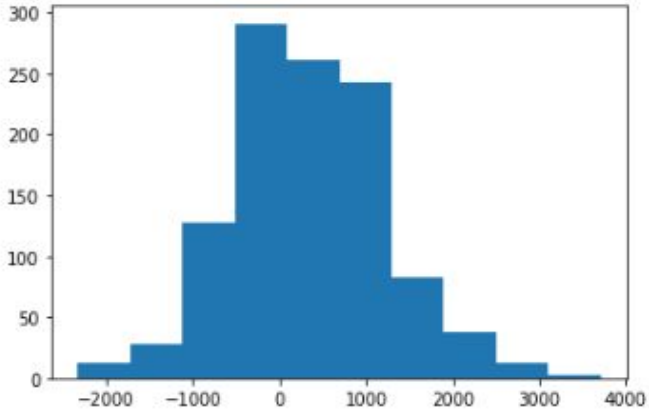
histogram of data at time: 18



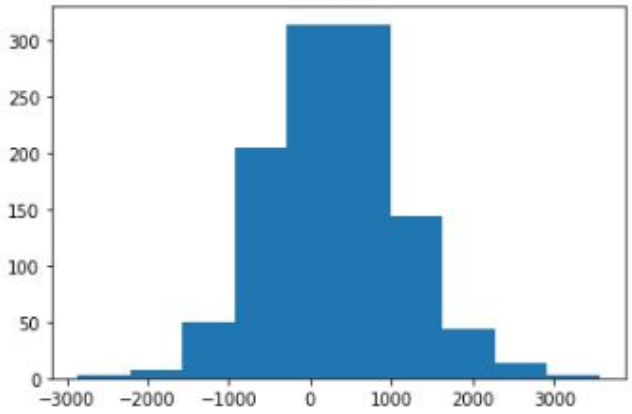
histogram of data at time: 21



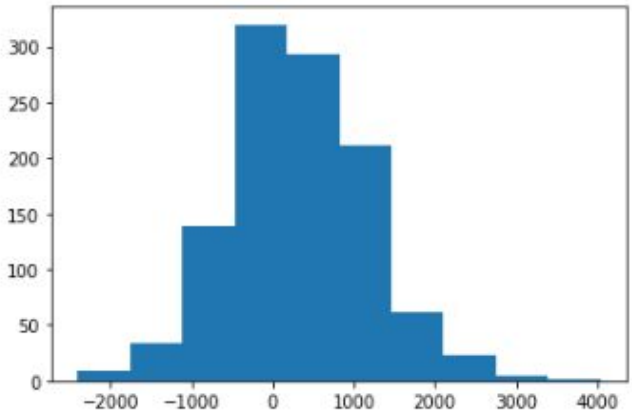
histogram of data at time: 19



histogram of data at time: 22



histogram of data at time: 20



histogram of data at time: 23