# **Data wrangling: Predicting energy consumption at an hourly frequency for San Diego region**

## **Importing data**

The hourly energy consumption data for the 4 utilities- PGE (Pacific Gas and Electric), SCE (Southern California Edison), SDGE (San Diego Gas and Electric) and VEA (Valley Electric Association, which actually covers some parts of Nevada)- under the CAISO (California Independent System Operator) is available on CAISO’s website in the form of .csv files. The most recent data for past 3 months and 2018 is available on<http://www.caiso.com/planning/Pages/ReliabilityRequirements/Default.aspx> and previous data upto 2014 is available in the arhive folder here:<http://www.caiso.com/Pages/documentsbygroup.aspx?GroupID=8879C382-6EA8-4357-B752-D4F571388958>.

Python’s *requests* and *html* packages were used to extract the *.csv* files directly from the websites and stored in dictionaries before concatenating them into a single dataframe.

## **Data cleaning**

The data was checked for any errors and after looking at the head and tail of each year’s datasets and also after calculating the count of rows for each year in the dataset it was found that the-

1. ‘Dates’ column of 2017 and 2018 were a bit different than the other years' *‘Dates’* column. This was solved by converting all the rows in *'Dates'* column data which were in the format *'201x-xx-xx yy:59:59.992'* to *'201x-xx-xx yy:00:00'* by using dt.floor.

2. Also, the last few entries in the *‘Dates’* column of the year 2018 had 2017 as the year instead of 2018. Ideally the count of rows in each years’ datasets should have been 8760 for each year except 2016 because it was a leap year. So, the number of rows in 2014,2015,2017 and 2018 should have been 8760 each and that in 2016 should have been 8784. But, the count of rows in each years’ datasets showed that this was not the case, and in fact 2018’s dataset had some entries for year 2017. The plausible explanation to this discrepancy was that somewhere in the data, 24 hours of year 2018 weren't recorded and the data from 2017 was used to fill in the missing values. This left the *‘Dates’* column with the year as 2017 instead of 2018. The incorrect year was replaced with the correct one by using *Dateoffset* method on the dataframe.

3. The energy consumption values should have been numeric but they were found to be of *object* type. So, they were converted to float using *pd.to\_numeric*.

4. The dataset had few missing values for the energy consumption columns. Since this is an energy consumption data, we need to keep in mind that the energy consumption of an area (including cities and rural) changes depending on the hour of the day, day of the week, month of the year, season and the year itself. So, the actual day of the month won’t matter as much as the month, day of the week and hour of the day. Keeping this is mind the missing values were filled using *.fillna()* method by the average values from the same hour, weekday and month taken from rest of the data.

5. The *.info()* and *.describe()* were used to check if the data had any outliers and no outliers were found in the dataset after carrying out the above cleaning operations.

## **Next steps**

* Exploring the data using the *matplotlib* library to verify it more thoroughly and also jot down basic overall observations from the dataset.