GENERAL ASSEMBLY

DSI - 7 SF

CAPSTONE: CALIFORNIA ELECTRICITY PRICE PREDICTION

Manu Kalia

PROBLEM STATEMENT

In the Calif. Electricity grid operator's footprint (CAISO), predict Day-ahead Market and Hourly Market prices for wholesale electricity, using system operator demand forecasts, weather observations, and reservoir water levels.

DATA COLLECTION

There are four types of data that come from three different sources.

1. Electricity wholesale prices. The California Independent System Operator (CAISO) is a non-profit entity that is tasked with operating the electrical "grid" as a public good. The primary concern for CAISO is operations without any outages, and to always match supply with demand (also called "load") for energy, which cannot be realistically stored in large amounts.

   CAISo maintains a publicly accessible repository of many types of information, in the interests of transparency. The open access system is known as OASIS, and can be queried via API for downloads of information. 2 requests per second, and no more than 31 days of info can be made to CAISO. Any more and the requester IP address will be blocked.

   Wrote an automated loop to request both price and load (demand) forecast info from CAISO... automatically generating the correct string format url + query syntax, and building in a 5 second delay between successive monthly data requests going back 40 months to Jan. 1, 2016. Data is on hourly, five-minute, and fifteen minute intervals, depending on the particular datum in question.

2. California Dept. of Water Resources (CA DWR) maintains a database of statewide metrics relating to reservoirs, inflow/outflow, water levels, snowpack water content, and more. As one feature, I included an hourly sum of the 47 reservoir's water content, measured in acre-feet. There was no date range restriction on the api queries to CADWR.

   Total water capacity may be an excellent feature to explain some electricity price movement, since water levels dictate hydroelectric power production. A good rainfall season like 2018-2019 can lower electricity prices well into the fall because far less gas-fired generation needs to be dispatched.

3. NOAA hourly weather station historical data. Very large zip tar files are available by year for download. These files unzip into 50 GB directories of many .csv files, each named with an 11-digit numerical code identifying a particular weather station or buoy. Selected four particular weather stations in California: San Diego, Riverside, Redding, and Fresno for inclusion of 4 weather related statistics. The four locations were chosen for their completeness of data, and for their geographic spread in California.

   The four extracted statistics are: surface temperature, wind speed, cloud ceiling height, and horizontal visibility. These are all metrics that affect either the demand for, or supply of, electricity. Temperature motivates electricity usage for air-conditioning and electric heating, wind speed indicates wind farm productivity, and cloud ceiling + visibility may indicate the amount of potential solar generation available.

DATA INSPECTION and EDA

Missing values. CAISO's data set of prices and load forecasts is very complete with no missing values, but I needed to drop one day per 31 day month to keep the API from rejecting my requests during daylight savings time, when the extra hour (relative to UTC) causes the 31-day limit to trip by one hour. Left these as gaps in the datetime index. Downsampled 5- and 15-minute interval data to hourly means. Most data was already in hourly granularity, so that made the most sense to keep.

NOAA dataset was quite a bit messy, with missing values (encoded with '9999', repeated datetime index values, and lots of duplicates. I left joined the four locations' dataframes together after slicing out the relevant statistics from long tuples of comma separated string values. On a case by case basis, I employed a forward fill method to fill Nan values, following

the logic that a missing observation will be most likely to be the same as the previous measured value. Graphed all columns to check for anomalies and outliers/errors.

Correlations. Used a color heatmap of Pearson correlations to assess how predictive the features might be. Did not see very much predictive power, so this is likely a very difficult problem to attack.

TWO TARGETS, 27 FEATURES

Altogether, there are 16 weather features, one water level feature, four datetime features, four electricity demand forecast features, one realtime spot settlement price feature, and two target variables: DAM (Day Ahead Market) and HASP (Hour Ahead Scheduling Process). All regressions will be pairs of models. Each target variable will take the other target as a feature.

DATA DICTIONARY

Index	Date time in Pacific time zone. Starts at 1:00am 01-01-2016, ending at 23:00 04-24-2019 with some gaps, for a total 29,067 rows
Train set	First 75% of data... 21,800 rows
Test set	Last 25% of data... 7,267 rows
dam_price_per_mwh	Day ahead mkt price, in MWh (1 of 2 targets) at Bay Shore SF node
hasp_price_per_mwh	Hour ahead price, in MWh (2 of 2 targets) at Bay Shore SF node
7da_load_fcast	Tot. (in MW) CA elec demand forecast, rolling 7 days in advance
2da_load_fcast	Tot. (in MW) CA elec demand forecast, rolling 2 days in advance
dam_load_fcast	Tot. (in MW) CA elec demand forecast, rolling one day ahead
rtm_load_fcast	Tot. (in MW) CA elec demand forecast, realtime for next hour
water_acre_feet	Total of 47 CA reservoirs' water content, in acre-feet
sand_temp	San Diego weather stn temp (-0932 to +0618) in deg C, scaling factor 10, NaN=9999
sand_wind	San Diego weather stn wind speed (0-900) in m/s,

	scaling factor 10, NaN=9999
sand_vis	San Diego weather stn
sand_ceil	San Diego weather stn (0-22000) in m, scaling factor 1, NaN=99999
rive_temp	Riverside weather stn temp (-0932 to +0618) in deg C, scaling factor 10, NaN=9999
rive_wind	Riverside weather stn wind speed (0-900) in m/s, scaling factor 10, NaN=9999
rive_vis	Riverside weather stn
rive_ceil	Riverside weather stn (0-22000) in m, scaling factor 1, NaN=99999
redd_temp	Redding weather stn temp (-0932 to +0618) in deg C, scaling factor 10, NaN=9999
redd_wind	Redding weather stn wind speed (0-900) in m/s, scaling factor 10, NaN=9999
redd_vis	San Diego weather stn
redd_ceil	Redding weather stn (0-22000) in m, scaling factor 1, NaN=99999
fres_temp	Fresno weather stn temp (-0932 to +0618) in deg C, scaling factor 10, NaN=9999
fres_wind	Fresno weather stn wind speed (0-900) in m/s, scaling factor 10, NaN=9999
fres_vis	Fresno weather stn
fres_ceil	Fresno weather stn (0-22000) in m, scaling factor 1, NaN=99999
year	2016 - 2019
month	1 - 12
day	1 - 31
hour	0 - 23

THREE REGRESSION MODELING ESTIMATORS

1. ARIMA Auto regressive integrated moving average

2. SARIMAX ARIMA with Seasonal effect, and eXogenous variables
3. RNN Recurrent Neural Network

Looking to explore the relative strengths and weaknesses of these three methods, both in predictive performance, as well as in ease of implementation, and computational demand.

CONCLUSIONS AND ASSESSMENT

The hour ahead market proved to be more difficult for these methods to predict than the day ahead market. ARIMA showed surprisingly good performance, matching that of the recurrent neural networks. The RNNs were much faster to fit than ARIMA, and MUCH faster than SARIMAX, which was the worst performer overall.

ARIMA & SARIMAX

a) Performance. ARIMA showed a surprisingly strong performance (DA market), given that no exogenous features go into the prediction. Electricity usage is highly patterned, and costs do not change drastically, so this makes sense. SARIMAX was a worse performer, and given the addition of seasonality and exogenous features, this should not be the case. Both models (as well as RNNs) fared poorly for the HA market.

b) Deficiencies. The Statsmodel package is not intuitive, and difficult to set up for train/test splits to get predictions on a test set fitted on a train set. Extremely poor computational efficiencies, especially for SARIMAX.

RNNs

The Recurrent Neural Networks took effort to configure, with a number of parameters and architectural choice having meaningful effect on predictive performance, including:
- number of perceptrons per layer
- number of hidden layers
- number of "lookback" periods (recurrent aspect of RNN)
- learning rate
- regularization (early stopping, dropout layers, L1, and L2)

With that, however, the fit speed was quite fast relative to the compute intensity of ARIMA, and most especially SARIMAX. Performance equaled that of ARIMA, and was easier to work

with to extract predictions and apply scoring like MSE and r-squared. Given that RNNs are still young with room for improvement, this appears to be the far richer direction to explore.

The Day Ahead market appears to show promise for prediction, more so than the Hour Ahead market. While SARIMAX should perform better than ARIMA, it fails to do so for the parameters attempted, and the long fit times make it unattractive. RNNs show great promise, matching ARIMA results, but faster fit times, and more easily extractable predictions. Overall, a successful first cut at this problem, with a number of rich options to pursue for future refinement.

FUTURE WORK ITEMS

1. Expand data set back more years to get more train-test sets, fill in missing hours by going back to the CAISO API (skipped some days to come in under CAISO 31-day hard restriction per query)

2. Extend ARIMA grid-search, run SARIMAX grid search

3. Do a systematic exploration of all RNN node/layer/lr/look-bk combinations

4. Try the Facebook Prophet time-series ml tool

5. Cast the dataset into "tabular" format and remove the time index to allow for other estimators like random forest, etc.