CS229 Fall 2013 Final project writeup

Prediction of total daily incoming solar energy based on weather models

Team Members: Jave Kane and Julie Herberg

December 13, 2013

Introduction

Our project responds to a Kaggle challenge posted by EarthRisk Technologies [1], to "discover which statistical and machine learning techniques provide the best short term predictions of solar energy production". Renewable energy sources such as solar and wind offer environmental advantages over fossil fuels for generating electricity, but due to the temporal variability of these sources, utilities must be prepared to make up deficits with power from traditional fossil fuel plants or purchases of power from neighboring utilities. Maintaining the correct mix requires detailed, accurate short-term predictions of wind patterns and solar illumination.

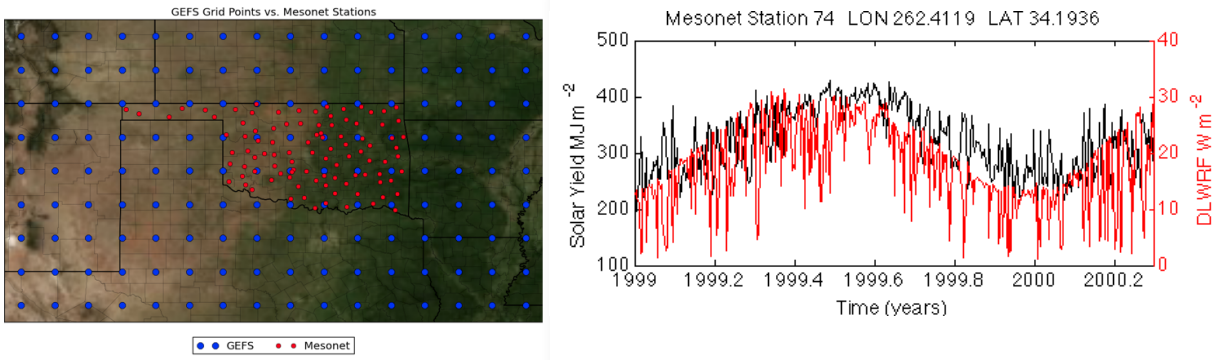

Figure 1. Left: GEFS weather forecast grid sites (blue dots) and Oklahoma Mesonet stations where solar radiation measurements are made (red dots). Right: Plot of training set output (black) at one Mesonet station, and input feature #2, "Downward long-wave radiative flux average at the surface" (red), at nearest GEF site at one time of day. Full time domain of the training set is 14 years (1.3 years are shown.)

Contestants are asked to predict the total daily incoming solar energy at 98 Oklahoma Mesonet ("mesoscale + "network") environmental monitoring stations [2], which serve as 'solar farms' for the contest. Every five minutes these stations record mesoscale weather observations, including incoming solar radiation — the output for the Kaggle challenge. The features are weather predictions from the NOAA/ESRL Global Ensemble Forecast System (GEFS) [3]. Our goal for this project is to set up a framework for this type of research and identify relevant machine learning tools. The winning Kaggle approach was made public Nov. 21, 2013 (we discovered this only within the last week), and turned out to be similar to ours.

Data

The training features come from an ensemble of 11 weather forecasts (one unperturbed, and 10 perturbed using various instrument uncertainties) for five 3-hour intervals during each of the 5113 days from January 1, 1994 through December 31, 2007 inclusive (14 full years), at 16 x 9 GEF sites (a grid of 16 longitudes and 9 latitudes (Figure 1, left panel). There are fifteen forecast features (see [4] for a full list), including, for example, (2) Downward long-wave radiative flux average at the surface (W m^{-2}), (4) Air pressure at mean sea level (Pa), (7) Total cloud cover over the entire depth of the atmosphere, and (9) Maximum Temperature over the past 3 hours at 2 m above the ground (K). Training output is the total daily incoming solar energy in (J m^{-2}), integrated over each 24-hour day at the 98 Mesonet sites. The right panel of Figure 1 plots the output for one Mesonet station over 1.3 years, and one feature from the nearest GEF site. GEF test features are provided for each of the 1796 days from Jan 1, 2008 through Dec. 31, 2012 (5 years). Test output is held privately on the Kaggle website. Model predictions of the output can be submitted to the website, which returns (1) a total error on the entire output, and (2) the ranking of the model output on the leader board [5]. There were 163 contestants by the end of the competition, which closed Nov. 21, 2013.

Models

We used (1) linear regression via the Normal equations, (2) Minimum Absolute Error (MAE) using MATLAB 'fminsearch' (finds minimum of unconstrained multivariable function using derivative-free method) [6], with parameters initialized from linear regression, and (3) Gradient Boosted Regression (GBR), a predictive model typically used in models associated with Decision Tree Learning (DTL). DTL utilizes a stepwise approach to evaluate a large dataset and evaluates the results at different nodes. Likewise, GBR builds a prediction model with the ensemble of weak prediction models, such as decision trees [7–8]. We 'black boxed' GBR in the sense that it was not clear to us how to characterize an overall error function that GBR is minimizing. (We simply evaluated the MAE of the GBR outputs.) We also tested k=2 linear regression, but consistently encountered poorly conditioned feature matrices, so we did not pursue this. For a given model family we generated one set of parameters (or GBR model) per Mesonet station, *i.e.* we did not consider possible correlations between stations.

For preliminary model assessments, we performed holdout cross validation and k=10-fold cross validation. We also considered options for feature aggregation including: using only the nearest GEF site to a Mesonet station, the mean or distance-weighted mean of the features at nearest four GEF sites, and all features at those four sites. We considered aggregating the five daily GEF measurements, using either the middle one, or averaging over all five. We wound up mostly using all five three-hour measurements as separate features.

Even for k=1 linear regression, some matrices at some Mesonet stations were poorly conditioned, apparently due to (surprising) near-dependence of two features: (4) Air Pressure and (6) Specific Humidity — for simplicity we preempted this issue by using matrix pseudo-inverse in the Normal equations. A preliminary look at the features using Principal Component Analysis (PCA) suggested at least 10 of 15 components are needed to cover 99% of the variance. To make progress, we mostly used only the unperturbed forecast of the GEF ensemble. For some Kaggle submissions we averaged outputs over all 11 forecasts. We investigated subtracting the annual Fourier component (see Figure 1) from the data, and in the process we discovered further structure that might be analyzed by *e.g*. wavelets (not shown). Ultimately we simply included time-of-year as an added feature. For simplicity we did not include GEF site elevation, latitude or longitude as features.

We used Gradient Boosted Regression (GBR) code from the Open MATLAB Forum [9], with simply the default exponential loss function. The GBR training step was very slow. Therefore it was difficult to use cross-validation to choose the number of trees, number of leaves, and learning constant. A series of models at single Mesonet stations suggested that using 128 trees and 8 leaves gave good errors without obvious overfitting. When averaging the four nearest GEF sites, a learning constant υ of 0.01 (*i.e.* 1% of information discarded) appeared optimal, whereas when using *all* features at the four nearest GEF sites, υ = 0.1 seemed better. Speculatively, this could be due to correlation between the features at the 4 GEF sites, *i.e.* possibly the algorithm works better when discarding redundant information more quickly.

Results and Discussion

As Figure 2 shows, PCA suggests at least 10 fairly distinct components. Since the spread between models (and Kaggle scores) was not large, a few percent of error mattered, so we did not pursue PCA further. Figure 3 shows MAE at a single Mesonet station for selected models, using only the unperturbed forecast from the GEF ensemble. Figure 4 show errors for selected holdout cross-validation and k=10-fold cross-validation errors for models at all Mesonet stations using the unperturbed GEF forecasts; the figure shows that using the four nearest GEF sites, instead of the nearest site, decreased all errors.

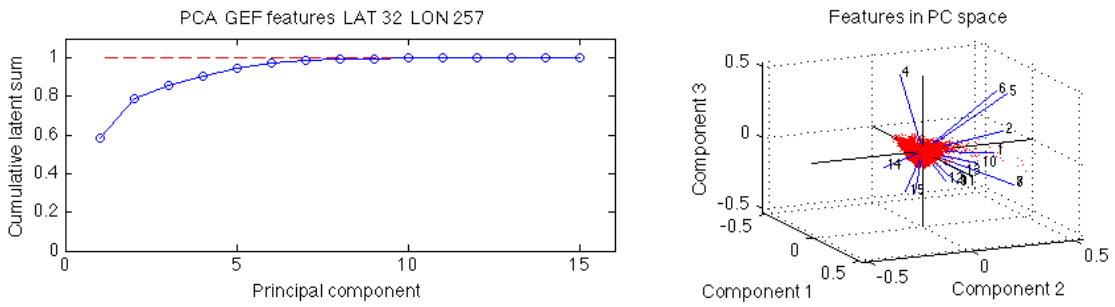

Figure 2 PCA GEF features LAT 32 LON 257. Right: Cumulative latency. Left: features in PC space.

#	Model	MAE (MJ m^{-2})
1	Linear regression, nearest GEF site, middle time	3.49
2	Linear regression, nearest GEF site, middle time of day, plus time of year	3.44
3	Linear regression, mean of 4 nearest GEFs sites, middle time of day, time of year	2.92
4	Linear regression, 4 nearest GEFs sites, middle time of day, time of year	2.58
5	MAE/fminsearch on output of #4	2.54
6	GBR(128 trees, 8 leaves, ν=**0.01**), 4 nearest GEF sites, all times of day, all features, time of year	2.18

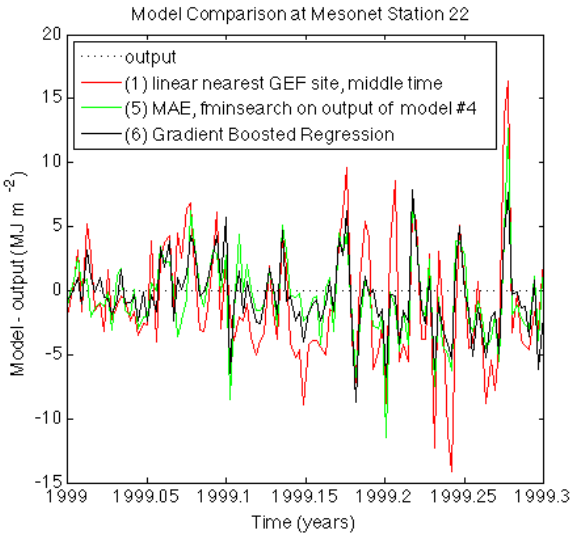

Figure 3 Predictions of selected models run with the unperturbed GEF forecast (ensemble forecast 1/11). Left side: MAE. Right side: Model minus output for a 1.3 years of the training time domain.

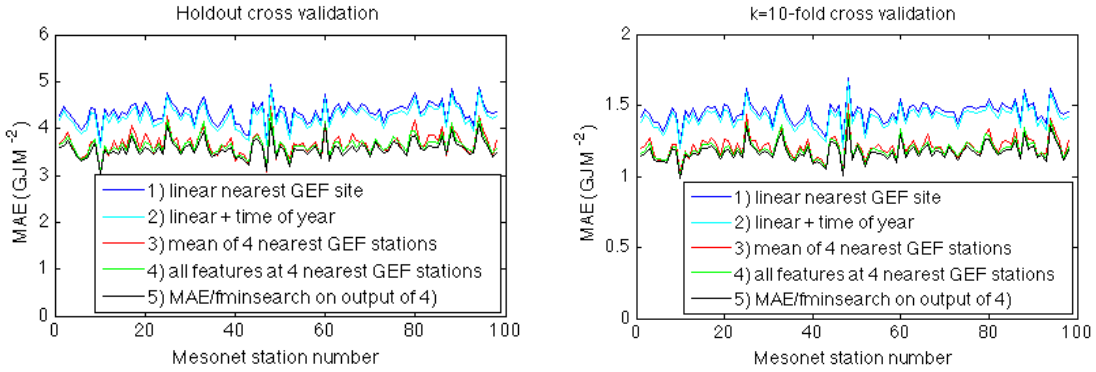

Figure 4 Cross validation generalization error for selected models run with unperturbed GEF forecast. Left side: holdout cross validation (GJ m^{-2}). Right side: *k*=10-fold cross validation (different scaling).

These error assessments illustrate how a series of changes to the feature set incrementally improved the predictions. Based on these results, and on focused evaluations using the complete 11-forecast GEF ensemble (not shown), we chose a set of models that were feasible to train, and ran them on the full set of 98 Mesonet sites using the GEF test features. We submitted the outputs to Kaggle for scoring against the (hidden) test output. Figure 5 shows some of our Kaggle errors and rankings. For model #1, the full GEF forecast ensemble was used; for all others, only the unperturbed forecast in the GEF ensemble was used. Notably, GBR performs very differently when we change the learning constant from 0.01 to 0.1 (with the number of trees and number of leaves fixed); it is not obvious to us why this change made such a large difference. We note that a large improvement in rank (if not error) was achieved by running MAE/fminsearch on the output of linear regression.

#	Description of our model	Kaggle rank out of 164	Kaggle error (MJ m^{-2})	Kaggle error / winning error
1	MAE/fminsearch on output of linear regression, averaged over GEF ensemble	66	2.48	1.18
2	GBR(128 trees, 8 leaves, v=**0.1**) 4 nearest GEF sites, all features, time-of-year	69	2.50	1.19
3	MAE/fminsearch on output of linear regression (#4).	73	2.51	1.19
4	Linear regression on nearest GEF site	120	2.62	1.24
5	GBR(128 trees, 8 leaves, v=**0.01**) 4 nearest GEF sites, all features, time-of-year	133	2.89	1.37

Figure 5 Kaggle rank and error for selected models

Conclusions

We addressed a real-world problem that is significant in its own right, but the data and methods are of broader interest, for example to airborne hyperspectral sensing [10]. The data was 'right-sized', allowing both decent statistics and evaluation of a range of models. Feature selection and aggregation required some thought. As physicists, we spent less time thinking about the content of the features than we expected to, since the machine learning models accepted most sets of features without issue. Our basic skills from CS229 took us within 30% of the winning Kaggle score. However, the winning approach was very similar to ours, suggesting the extra effort to reach #1 follows an '80/20' rule. It appears using GBR requires some experience. Efficiently tuning and using GBR probably requires parallelization; this is difficult with the MATLAB license structure on the machines we could easily use; GBR should probably be run outside MATLAB. (The Kaggle winner used "R".)

References

[1] Kaggle website http://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest
[2] Mesonet Website (2013). http://www.mesonet.org/
[3] U.S. Department of Commerce, National Oceanic & Atmospheric Administration website (2013). http://esrl.noaa.gov/psd/forecasts/reforecast2/
[4] Kaggle website – Data (2013). http://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest/data
[5] Kaggle website – Leadership board (2013). http://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest/leaderboard
[6] Lagarias, J.C., *et. Al.*, *SIAM Journal of Optimization*, Vol. 9 Number 1, pp. 112-147, 1998.
[7] Friedman, J. H. "Greedy Function Approximation: A Gradient Boosting Machine." (February 1999).
[8] Friedman, J. H. "Stochastic Gradient Boosting." (March 1999).
[9] Hara, Kota, Boosted Binary Regression Trees (06 August 2013), http://www.mathworks.com/matlabcentral/fileexchange/42130-boosted-binary-regression-trees
[10] Hyperspectral sensing. http://www.csr.utexas.edu/projects/rs/hrs/hyper.html