Predicting Solar Generation from Weather Forecasts Using Machine Learning

Navin Sharma, Pranshu Sharma, David Irwin, and Prashant Shenoy
Department of Computer Science
University of Massachusetts Amherst
Amherst, Massachusetts 01003
{nksharma,pranshus,irwin,shenoy}@cs.umass.edu

***Abstract*—A key goal of smart grid initiatives is significantly increasing the fraction of grid energy contributed by renewables. One challenge with integrating renewables into the grid is that their power generation is intermittent and uncontrollable. Thus, predicting future renewable generation is important, since the grid must dispatch generators to satisfy demand as generation varies. While manually developing sophisticated prediction models may be feasible for large-scale solar farms, developing them for distributed generation at millions of homes throughout the grid is a challenging problem. To address the problem, in this paper, we explore automatically creating site-specific prediction models for solar power generation from National Weather Service (NWS) weather forecasts using machine learning techniques. We compare multiple regression techniques for generating prediction models, including linear least squares and support vector machines using multiple kernel functions. We evaluate the accuracy of each model using historical NWS forecasts and solar intensity readings from a weather station deployment for nearly a year. Our results show that SVM-based prediction models built using seven distinct weather forecast metrics are 27% more accurate for our site than existing forecast-based models.**

I. Introduction

A key goal of smart grid efforts is to substantially increase the penetration of environmentally-friendly renewable energy sources, such as solar and wind. For example, the Renewables Portfolio Standard targets up to 25% of energy generation from intermittent renewables [1], while Executive Order S-21-09 in California calls for 33% of their generation to come from renewables by 2020 [2]. Substantial grid integration of renewables is challenging, since their power generation is intermittent and uncontrollable. The modern electric grid permits households to consume electricity in essentially arbitrary quantities at any time, and is not currently designed for vast quantities of uncontrollable generation. Instead, the grid constantly monitors the demand for electricity, and dispatches generators to satisfy demand as it rises and falls. Fortunately, electricity demand is highly predictable when aggregating over thousands of buildings and homes. As a result, today's grid is able to accurately plan in advance which generators to dispatch, and when, to satisfy demand.

The problem with substantial renewable integration is that the electricity renewables generate is not easily predictable in advance and varies based on both weather conditions and site-specific conditions. While utilities may take the time to manually develop accurate prediction models for large-scale solar farms that produce multiple megawatts, manually developing specialized models that predict the power output from distributed generation at many small-scale facilities at smart homes and buildings throughout the grid is infeasible. This fact is evident in current net metering laws for most states, which allow consumers to sell energy produced from on-site renewables back to the grid, but typically places low caps on both the total number of participating customers and/or the total amount of energy contributed per customer [3]. As one example, Massachusetts caps the total number of participating customers at 1% of all customers. Utilities restrict the contribution from renewables, since, unlike electricity demand, renewable generation is not easily predictable, and complicates advance planning of the grid's generator dispatch schedule.

To facilitate better planning and lower the barrier to increasing the fraction of renewables in the grid, we focus on the problem of *automatically* generating models that accurately predict renewable generation using National Weather Service (NWS) weather forecasts. Specifically, we experiment with a variety of machine learning techniques to develop prediction models using historical NWS forecast data, and correlate them with generation data from solar panels. Once trained on historical forecast and generation data, our prediction models use NWS forecasts for a small region to predict future generation over several time horizons. Our experiments in this paper use solar intensity as a proxy for solar generation, since it is proportional to solar power harvesting [4]. Importantly, since we generate our models from historical site-specific observational power generation data, they inherently incorporate the effects of local characteristics on each site's capability to generate power, such as shade from surrounding trees. Since local characteristics influence power generation, individual sites must tune prediction models for site-specific characteristics. We view automatic model generation as critical to scaling distributed generation from renewables to millions of homes throughout the grid.

Our goal is to automate generating prediction models for smart homes that include on-site renewables. Both the grid and individual smart homes may use these prediction models for advance planning of electricity generation and consumption. The grid can use the models to plan generator dispatch schedules in advance as the fraction of renewables increases in the grid. Smart homes can use the models to potentially plan

978-1-4577-1702-4/11/$26.00 ©2011 IEEE

their consumption patterns to better match the power that they generate on-site. In both cases, better prediction models are a prerequisite for increasing efficiency and encouraging broader adoption of distributed generation from renewables in the grid and at smart homes. In studying prediction models for solar energy harvesting, we make the following contributions.

- **Data Analysis**. We analyze extensive traces of historical data from a weather station, as well as the corresponding NWS weather forecasts, to correlate the weather metrics present in the forecast with the solar intensity, in watts per m^2, recorded by the weather station. Our analysis quantifies how each forecast parameter affects each other and the solar intensity. For solar energy harvesting, we find that sky cover, relative humidity, and precipitation are highly correlated with each other and with solar intensity, while temperature, dew point, and wind speed are only partially correlated with each other and with solar intensity.
- **Model Generation**. We apply multiple machine learning techniques to derive prediction models for solar intensity using multiple forecast metrics, and then analyze the prediction accuracy of each model. We use machine learning on a training data set of historical solar intensity observations and forecasts to derive a function that computes future solar intensity for a given time horizon from a set of forecasted weather metrics. We formulate models based on linear least squares regression, as well as support vector machines (SVM). We find that SVM with radial basis function kernels built using historical data from seven weather metrics is 27% more accurate than existing forecast-based models that use only sky condition for predictions [4] and is 51% better than simple approaches that only use the past to predict the future.

In Section 2 we analyze forecast metrics and explore how they affect each other, as well as how they affect solar intensity, while in Section 4 we describe and evaluate multiple machine learning strategies for generating prediction models using our weather station data and NWS forecasts. Finally, Section 5 discusses related work and Section 6 concludes.

II. Data Analysis

We collect weather forecast data and observational solar intensity data for 10 months starting from January 2010. We obtain historical forecast data from the NWS at http://www.weather.gov, which we have been collecting for the past 2 years. The NWS provides historical textual forecasts for small city-size regions throughout the U.S., which include multiple weather metrics for every hour of every day for the last few years. Each forecast includes predictions of each metric every 1 hour from 1 hour to 6 days into the future. Examples of weather metrics include temperature, dew point, wind speed, sky cover, probability of precipitation, and relative humidity. Sky cover is an estimate of the percentage (0%-100%) of cloud coverage in the atmosphere. In addition to making historical forecasts available, the NWS also operates a real-time web service that enables applications to retrieve

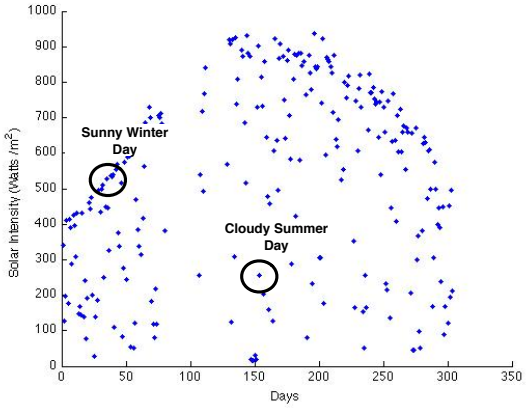
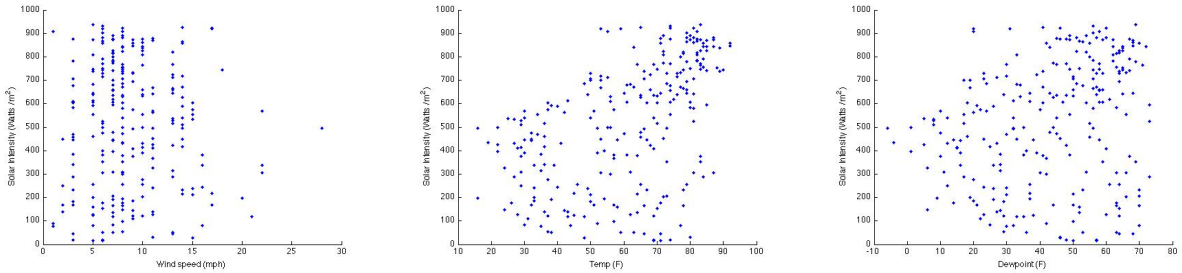


Fig. 1. Solar intensity shows seasonal variation with days of a year, although daily weather conditions also have a significant impact.

forecasts programmatically as they become available. In addition to these metrics, we include the specific day of the year and time of the day as metrics, since daylight influences solar intensity and varies throughout the year for a given location.

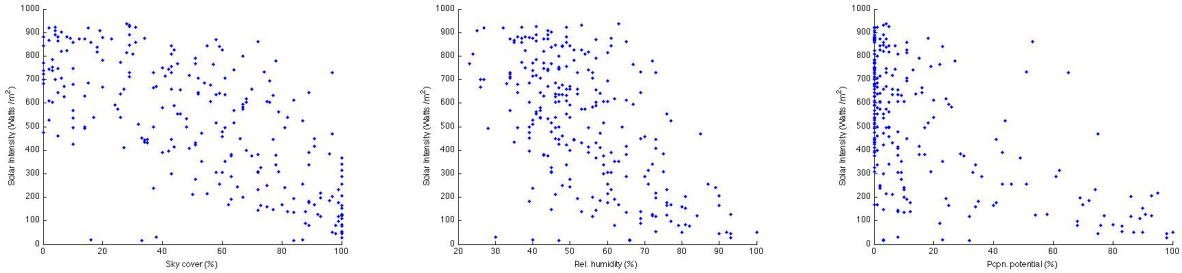
We use observational solar intensity data from an extended weather station deployment on the roof of the Computer Science Building at the University of Massachusetts Amherst. The weather station reports solar intensity in watts/m^2 every 5 minutes of every day. Traces from the weather station are available at http://traces.cs.umass.edu. As we show in previous work, power generation from solar panels is directly proportional to solar intensity [4]; in general, solar panel inefficiencies result in power output that is a fixed percentage decrease from the raw solar intensity readings at the same location. We use NWS forecasts for Amherst, Massachusetts. In this section, we study how solar intensity varies with individual forecast parameters and how these forecast parameters are related to each other. The purpose of our data analysis is to provide intuition into how solar intensity and solar panel power generation depends on a combination of multiple weather metrics, and is not easily predictable from a single weather metric. The complexity in predicting solar intensity from one or more weather metrics motivates our study of automatically generating prediction models using machine learning techniques in the next section.

Fig. 1 shows how the day of the year affects solar intensity by charting the average solar intensity reading at noon per day over our 10 month monitoring period, where day zero is January 1st, 2010. As expected, the graph shows that the solar intensity is lowest in January near the winter solstice and increases into the summer before decreasing after the vernal equinox. Additionally, the graph also implies that other conditions also have a significant impact on solar intensity, since many days throughout the spring and summer have low solar intensity readings. The graph shows that solar intensity and the day of the year are roughly correlated: most of the time, but not always, a summer day will have a higher solar intensity than a winter day. However, other factors must contribute to the solar intensity, since there are clearly some sunny winter days that record higher solar intensity readings than some



(a) (b) (c)

Fig. 2. Solar intensity and wind speed show little correlation (a). Solar intensity shows some correlation with temperature at high temperatures (b) and with dew point at high dew points (c).



(a) (b) (c)

Fig. 3. Solar intensity generally decreases with increasing values of sky cover (a), relative humidity (b), and precipitation potential (c).

cloudy summer days. To better understand correlations with other weather metrics, we model similar relationships for the other forecast metrics.

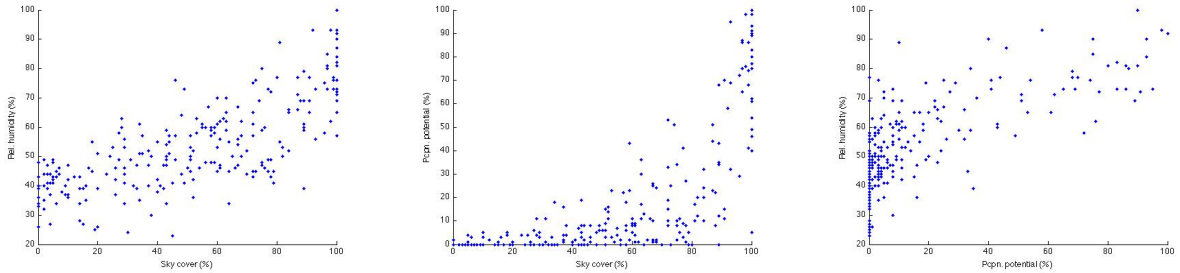
For example, Fig. 2 shows that wind speed, dew point, and temperature are not highly correlated with solar intensity. Solar intensity varies almost uniformly from lower to higher values at any value of wind speed (a). Thus, wind speed has nearly zero correlation with solar intensity and its value is not indicative of the solar intensity or solar panel power generation. Both temperature (b) and dew point (c) correlate with solar intensity at higher values: if the temperature or dew point is high, then the solar intensity is likely to be high. However, if the temperature or dew point is low, the solar intensity exhibits a more significant variation between high and low values. The results are intuitive. For example, in the summer a high temperature is often dependent on sunlight, while in the winter sunlight contributes less in raising the ambient temperature.

In contrast, Fig. 3 shows that sky cover, relative humidity, and chance of precipitation have high negative correlations with solar intensity. In each case, as the value of the metric increases, the solar intensity reading generally decreases. However, as with the day of the year, there must be other factors that contribute to the solar intensity reading, since there are some days with a high sky cover, relative humidity, and precipitation probability, but a high solar intensity reading and vice versa. In addition to exhibiting complex relationships with solar intensity, each weather metric also exhibits a complex relationship with other weather metrics. For example, Fig. 4 shows that relative humidity (a) and chance of precipitation (b) exhibit strong, but not perfect correlations, with sky cover, while relative humidity is strongly correlated with chance of precipitation (c). In all three cases, the metrics rise in tandem, although the relationship is noisy due to the value of other weather metrics.

Table 1 shows correlation coefficients for each weather metric using the Pearson product-moment correlation coefficient, which divides the covariance of the two variables by the product of their standard deviations. The higher the absolute value of the correlation coefficient, the stronger the correlation between the two weather metrics—a positive correlation indicates an increasing linear relationship, while a negative correlation indicates a decreasing linear relationship. The complex relationships between weather metrics and solar intensity shown in this table motivate our study of automated prediction models using machine learning techniques in the next section.

III. Prediction Models

We represent both observational and forecast weather metrics as a time-series that changes due to changing weather patterns and seasons. As the previous section shows, solar intensity depends on multiple weather metrics, which complicates the task of developing an accurate prediction model. The high dimensionality of the time-series data motivates our study of regression methods to develop solar intensity prediction models. To generate each model we provide eight months of training data (January to August) as input, which includes solar intensity readings as well as NWS forecasts for 6 weather metrics. The machine learning techniques automatically output



(a) (b) (c)

Fig. 4. Relative humidity (a) and precipitation % (b) positively correlate with sky cover. Relative humidity also increases with increasing precipitation % (c).

TABLE I
CORRELATION MATRIX SHOWING CORRELATION BETWEEN DIFFERENT FORECAST PARAMETERS.

	Day	**Temp.**	**Dew**	**Wind**	**Sky cover**	**Pcpn.**	**Humidity**
Day	1.0000	0.7022	0.7007	-0.0711	-0.1034	-0.0571	0.0645
Temp.	0.7022	1.0000	0.9212	-0.1994	-0.2582	-0.1279	-0.0791
Dew point	0.7007	0.9212	1.0000	-0.2251	0.0455	0.1491	0.3081
Wind	-0.0711	-0.1994	-0.2251	1.0000	-0.0192	0.0340	-0.1025
Sky cover	-0.1034	-0.2582	0.0455	-0.0192	1.0000	0.7067	0.7525
Precipitation	-0.0571	-0.1279	0.1491	0.0340	0.7067	1.0000	0.7475
Humidity	0.0645	-0.0791	0.3081	-0.1025	0.7525	0.7475	1.0000

a function that computes solar intensity from the 6 weather metrics, as well as the day of the year. We use the remaining 2 months of our data set to test the model's accuracy. One benefit of using machine learning to automatically generate prediction models is that, in general, the more training data that is available, the more accurate the model.

We focus our study on short-term forecasts three hours in the future. For our experiments, we develop models that determine a relationship at any time t between the solar intensity and the forecast weather metrics three hours in the past $(t-3)$. Note that we are able to apply our techniques to forecasts of any length; we choose three hours as a simple illustration. Using our models and the three hour forecast, we are able to compute a prediction for the solar intensity three hours in the future. The models we generate are simple functions, of the form below, that compute solar intensity from multiple weather metrics including the day of the year. We could also add time of the day as an additional metric, but for ease of exposition our experiments focus on predictions at noon. We compare the accuracy of our models with each other, as well as against a simple model we developed in prior work [4] based solely on the sky condition metric and against a simple past-predicts-future model. Our previous model multiplies the maximum power a solar panel is able to generate at a given time (of the day and year) by (1-SkyCover), since sky cover represents an estimate of the percentage of the atmosphere the sun is covering.

SolarIntensity = F(Day, Temperature, DewPoint, WindSpeed, SkyCover, Precipitation, Humidity)

F is the function that we determine using different regression methods. We preserve the units of each metric: we represent each day as a value between 0 and 365, temperature in degrees Fahrenheit, wind speed in miles per hour, sky cover in percentage between 0% and 100%, precipitation potential in percentage between 0% and 100%, and humidity in percentage between 0% and 100%. However, before applying any regression techniques below we normalize all feature data to have zero mean and unit variance. To quantify the accuracy of each model, we use the Root Mean Squared Error (RMS-Error) between our predicted solar intensity at any time and the actual solar intensity observed. RMS-Error is a well-known statistical measure of the accuracy of values predicted by a time-series model with respect to the observed values. An RMS-Error of zero indicates that the model exactly predicts solar intensity three hours in the future. The closer the RMS-Error is to zero the more accurate the model's predictions.

A. Linear Least Squares Regression

We first apply a linear least squares regression method to predict solar intensity. Linear least squares regression is a simple and commonly-used technique to estimate the relationship between a dependent or response variable, e.g., solar intensity, and a set of independent variables or predictors. The regression minimizes the sum of the squared differences between the observed solar intensity and the solar intensity predicted by a linear approximation of the forecast weather metrics. Applying the linear least squares method to the eight months of training data yields the prediction model below, with coefficients for each metric.

*SolarIntensity = 1.18*Day + 77.9*Temp + 33.11*DewPoint + 22.8*WindSpeed - 96.9*SkyCover - 49.15*Precipitation - 43.4*Humidity*

We verify the prediction accuracy using our test dataset for the remaining months of the year. We observe the cross

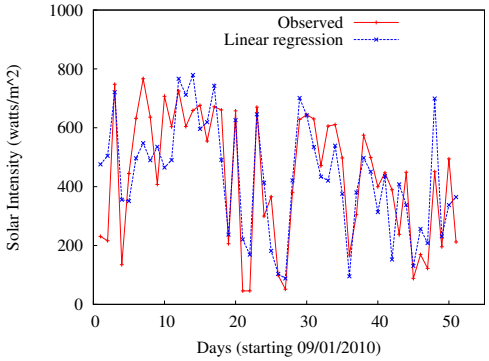


Fig. 5. Observed and predicted solar intensity using linear least squares regression for September and October 2010.

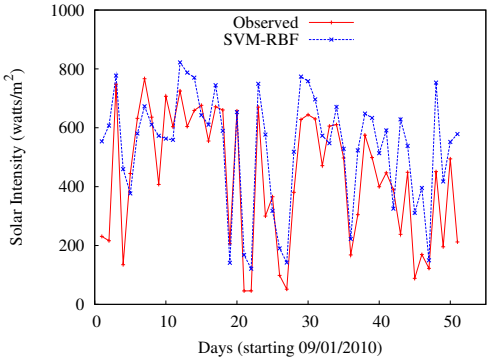


Fig. 6. Observed and predicted solar intensity, using SVM regression with an RBF kernel, for the months of September and October 2010.

validation RMS-Error and prediction RMS-Error in the solar intensity as 165 watts/m^2 and 130 watts/m^2, respectively. We cross validate the regression model with the training dataset (from Jan. to Aug. 2010) and verify its prediction accuracy using the testing dataset (Sept. and Oct. 2010). The cross validation RMS-Error quantifies how well the model predicts values in the training data set, while the prediction RMS-Error predicts how well the model predicts values in the testing data set. Fig. 5 shows the observed and predicted solar intensity for September and October 2010. As the figure shows, the model tracks the solar intensity prediction reasonably accurately, albeit with a few deviations.

B. Support Vector Machines

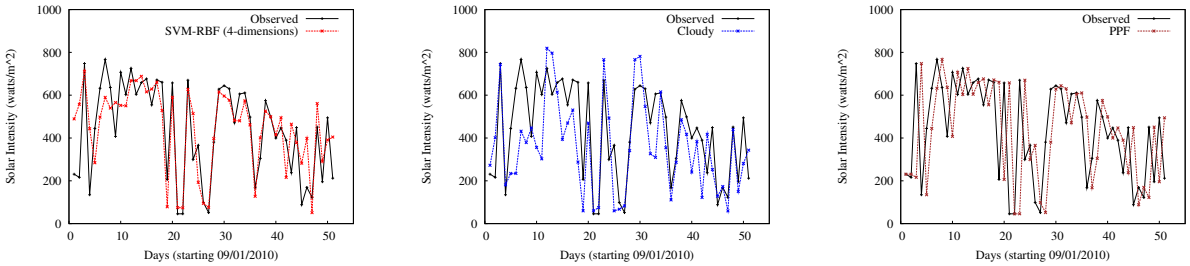
We next look at multiple classes of supervised learning methods using Support Vector Machines (SVM) [5]. SVMs, which construct hyperplanes in a multidimensional space, have recently gained popularity for classification and regression analysis. The accuracy of SVM regression depends on the selection of an appropriate kernel function and parameters. In our work, we studied three distinct SVM kernel functions: a Linear Kernel, a Polynomial Kernel, and a Radial Basis Function (RBF) kernel. An SVM uses the kernel function to transform data from the input space to the high-dimensional feature space. We chose SVMs over other supervised learning methods due to its sparsity property and its ability to handle non-linearity in the data. We use the LibSVM library, which includes a multitude of SVM regression techniques, to implement SVM regression with the linear kernel function on our training data set [6]. We found that both the linear and polynomial kernel performed poorly with RMS-Errors of 201 watts/m^2 and 228 watts/m^2, both of which were worse than the linear least squares approach above. As a result, we focus on results using the RBF kernel.

We tested the RBF kernel and SVM using the LibSVM library on our eight months of training data. In order to find the optimal parameters for the RBF kernel we ran a grid search tool from the LibSVM library on the training dataset. We found the optimal parameters of the RBF kernel to be $cost = 256$, $\gamma = 0.015625$, and $\epsilon = 0.001953125$. Using these parameters, we found that the RBF kernel using all seven dimensions for the seven weather metrics provides a better regression model than the other regression methods, as indicated by its low cross validation and prediction errors, both of which are significantly lower than either the linear or the polynomial kernels. Fig. 6 also reflects this fact in the time-series graph of observed and predicted values. The results show that the RBF kernel also performs better than the linear least squares method, having a slightly lower cross validation RMS-error (164 watts/m^2) and a slightly higher prediction RMS-Error (163 watts/m^2).

C. Eliminating Redundant Information

As we show in the previous section, many weather metrics show a strong correlation with each other. As a result, our SVM regression models contain redundant information, which often decreases the prediction accuracy of each model. Principal component analysis (PCA) is a popular method for removing redundant informations from an input dataset, thereby reducing its dimensionality [7]. Thus, we use the principal component analysis algorithm to remove redundant informations from our feature dataset. The PCA algorithm uses an orthogonal transformation to convert a set of, potentially correlated, input variables into a set of uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables. The first principal component has the maximum possible variance, and the second principal component has the maximum possible variance under the constraint that it be orthogonal to the first component, etc.

We choose the first four principal components corresponding to first four (highest) eigenvalues and run the RBF SVM regression method on the reduced feature-set. The results (Fig. 7(a)) show that the RBF kernel performs better after PCA analysis than when using the full feature-set with a cross validation RMS-Error of 159 watts/m^2 and a prediction RMS-error of 128 watts/m^2, both of which outperform the linear least squares model. We also ran experiments that further reduced the dimensionality of the feature set from 4 to 2. However, we found that all three SVM regression techniques performed worse compared to the 4-dimentional feature set. The performance degradation is the result of the additional



(a) SVM-RBF Kernel (b) Cloudy-FPF (c) PPF

Fig. 7. Observed and predicted solar intensity, using three different prediction techniques — (a) SVM-RBF kernel with 4 dimensions, (b) cloudy computing model using sky condition forecast, (c) past predicts future prediction model — for the months of September and October 2010.

reduction in dimensionality eliminating information that aids in prediction and is not redundant.

D. Comparing with Existing Models

Finally, we compare our regression-based prediction models with existing models. First, we compare with a *past-predicts-future* model (PPF), which uses the previous day solar intensity to predict the next day solar intensity. Past predicts future models are typically used when forecasts are not available, since the past is a reasonably good indicator of the future if the weather does not change. While they are highly accurate if weather conditions do not change, the models are not able to predict drastic changes in the weather. Second, we compare with a simple model that uses only the sky condition as a basis for prediction, called *cloudy*, which we developed in prior work [4]. We have shown that the *cloudy* model is more accurate than existing variants of PPF in the literature [8]. While the model is able to predict changes in weather, it does not incorporate information from multiple weather metrics and their impact on solar intensity.

Fig. 7(b) and (c) show how well cloudy and PPF predict weather in our testing data set, respectively. The results show that while the cloudy model follows the general trend of the weather it frequently exhibits wrong predictions. As expected, PPF's results are inaccurate whenever weather changes, which happens nearly every day. By contrast, Fig. 7(a) shows that SVM-RBF with the reduced feature provides a much more accurate model. The RMS-Errors for each model highlight this result: the RMS-Error for SVM-RBF with four dimensions is 128 watts/m^2, while the RMS-Error for cloudy and PPF is 175 and 261, respectively. Thus, SVM-RBF with four dimensions is 27% more accurate than the simple cloudy model and 51% more accurate than the PPF model.

IV. Conclusion

Prior prediction models for solar energy harvesting have been based primarily on the immediate past [9], [10], [11]. Unfortunately, these methods are unable to predict changes in weather patterns in advance. Since weather forecasts from the NWS are based on aggregations of multiple data sources from across the country, they are able to provide advance warning. The NWS generates forecasts from multiple sophisticated forecast models that synthesize a multitude of observational data. We show that the relationship between these forecast weather metrics and solar intensity is complex. Thus, we automatically derive prediction models from historical solar intensity and forecast data using machine learning techniques.

Our results indicate that automatically generating accurate models that predict solar intensity, and hence energy harvesting of solar arrays, from weather forecasts is a promising area. We find that models derived using SVMs with RBF kernels and linear least squares outperform a past-predicts-future models and a simple model based on sky condition forecasts from prior work [4] and is a promising area for increasing the accuracy of solar power generation prediction, which is essential to increasing the fraction of renewables in the grid. Moving forward, we plan on using our prediction models to better match renewable generation to consumption in both smart homes and data centers that utilize on-site solar arrays to generate power.

Acknowledgements. This work was supported by the National Science Foundation under grants CNS-0855128, CNS-0834243, CNS-0916577, and EEC-0313747.

References

[1] "Database of State Incentives for Renewables and Efficiency," http://www.dsireusa.org, 2010.
[2] "State of California Executive Order S-21-09," http://gov.ca.gov/executive-order/13269, 2009.
[3] "Freeing the Grid: Best and Worst Practices in State Net Metering Policies and Interconnection Procedures," http://www.newenergychoices.org/uploads/FreeingTheGrid2009.pdf, 2009.
[4] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy, "Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems," in *SECON*, June 2010.
[5] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.
[6] "LibSVM: A Library for Support Vector Machines," http://www.csie.ntu.edu.tw/ cjlin/libsvm/, 2011.
[7] I. Jolliffe, *Principal Component Analysis*. Springer Series in Statistics, 2002.
[8] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy, "Leveraging Weather Forecasts in Energy Harvesting Systems," University of Massachusetts Amherst, Tech. Rep., September 2011.
[9] A. Kansal, J. Hsu, S. Zahedi, and M. Srivastava, "Power Management in Energy Harvesting Sensor Networks," in *Transactions on Embedded Computing Systems*, September 2007.
[10] D. Noh, L. Wang, Y. Yang, H. Le, and T. Abdelzaher, "Minimum Variance Energy Allocation for a Solar-powered Sensor System," in *DCSS*, June 2009.
[11] C. Moser, "Power Management in Energy Harvesting Embedded Systems," Ph.D. Thesis, ETH Zurich, March 2009.