# Statistical Learning for Short-Term Photovoltaic Power Predictions

Björn Wolff[1], Elke Lorenz[2], Oliver Kramer[1]

[1]Department of Computing Science
[2]Institute of Physics, Energy and Semiconductor Research Laboratory
Carl von Ossietzky University in Oldenburg, Germany

**Abstract.** A precise prediction of photovoltaic (PV) power has an important part to play as basis for operation and management strategies for a reliable and economical integration into the power grid. Due to fast changing weather conditions, e.g., clouds and fog, a precise forecast in the minute to hour range can be a difficult undertaking. But the growing IT-infrastructure allows a fine screening of PV power. On the basis of big data sets of PV measurements, we apply methods from statistical learning for a one-hour ahead prediction based on data with hourly resolution. In this work, we employ nearest neighbors regression and support vector regression for PV power predictions based on measurements and numerical weather predictions. We put an emphasis on the analysis of feature combinations based on PV time series and numerical weather predictions.

## 1 Introduction

In Germany, with an installed capacity of more than 32 GW at the end of 2012, PV power prediction services are already an essential part of the grid control. On the local scale, storage management and smart grid applications define a sector with increasing need for PV power forecasting. As the benefit of using a forecast is directly related to the forecast accuracy, continuous research is performed to enhance PV power predictions. Based on large data sets of PV measurements, we employ methods from statistical learning for precise short-term predictions with hourly resolution. Objective of this paper is to analyze the training sets, features and model parameters for predictions based on nearest neighbors and support vector machines for 1-hour ahead forecast on the level of a single PV station and 10 stations with real-world data.

This paper is structured as follows. In Section 2, we present the data sets we employ in our experimental analysis. In Section 3, we shorty introduce the employed methods, i.e., uniform and weighted K-nearest neighbors (KNN) and support vector regression (SVR). An analysis of training sets, features and parameters are presented in Section 4 for KNN and the corresponding analysis for SVR in Section 5. A comparison of both methods is introduced in Section 6, while we summarize the most important results in Section 7.

## 2 Data Sets

Our forecasts are based on the combination of past solar power measurements with numerical weather predictions (NWP). Bacher *et al.* [1] have shown that for forecasts under a two-hour horizon, the most important features are measurements, while for longer horizons, NWPs are appropriate. We combine both types of features for our analysis. The data sets we employ consist of hourly PV power measurements and predictions from an NWP system (a local area model of European Center of Medium Range Weather Forecasts (ECMWF)) in 3-hour steps. Predictions of the NWP model (i.e., irradiance $\phi$ and cloud coverage) are converted to 1-hour data with a simple linear interpolation.

Figure 1 shows the 87 PV stations that are basis of our analysis. The red point marks the location of the station that is basis of our detailed single PV station analysis. The blue points mark the positions that are employed in the 10-stations experiments. Each of these 10 stations generates a forecast and the average is analysed.

For PV prediction, we use the following features:

- $T$: time $t$,
- $P$: relative power in % of maximum power,
- $CS$: clear sky power,
- $Temp$: temperature (from NWP model),
- $I$: irradiance (from NWP model), and
- $CC$: cloud cover (from NWP model).

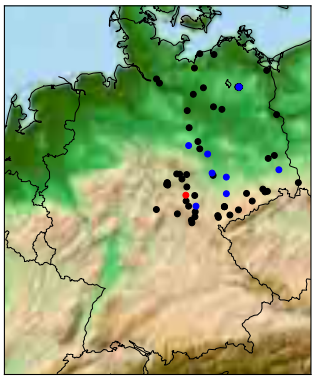


**Fig. 1.** Locations of PV stations in the eastern part of Germany. Blue points mark the stations that are basis of the 10-stations experiments, while the red point is the location of the single station we concentrate on.

The single PV station is located near Erfurt with at latitude 51.006 and longitude 11.016. It employs 70.37 MW of installed power. Different training sets are tested in the experimental settings. The test set consists of three different weeks of 1-hour PV time series, i.e., from 2011-07-13 to 2011-07-19, from 2011-08-20 to 2011-08-26, and from 2011-09-18 to 2011-09-24. In total, the test time series is 504 hours long.

For training of the regression models that aim at predicting the PV power at time $t$ for time step $t + 1$, we can only employ the information we have at time $t$. This is, the time information, the power at time $t$, as well as the the NWP predictions (clear sky, temperature, irradiance and cloud coverage) for time $t + 1$. For the construction of a training pattern, the corresponding label is the PV power at time $t + 1$. In the test setting, this is the target value we want to predict. As we employ a maximum of six features, the patterns have a dimensionality of $d \leq 6$. In Sections 4.2 and 5.2, we concentrate on the selection of optimal feature subsets that result in an optimal forecast. To measure the

quality of a forecast, the root mean squared error (RMSE) is used, which is defined as

$$RMSE(\mathbf{x}, \mathbf{x}') = \frac{1}{\sqrt{N}} \sqrt{\sum_{i=1}^{N} (x_i - x_i')^2}, \tag{1}$$

with $\mathbf{x}$ being the PV measurement and $\mathbf{x}'$ the corresponding prediction.

## 3 Statistical Learning Methods

In this section, we introduce the machine learning methods KNN and SVR that are basis of our experimental analysis.

### 3.1 Nearest Neighbors

Nearest neighbor predictions are based on the labels of the $K$-nearest patterns in data space. Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$ be a training set, then the prediction of the uniform KNN regression model is defined as $\mathbf{f}_{KNN}(\mathbf{x}') = \frac{1}{K} \sum_{i \in \mathcal{N}_K(\mathbf{x}')} \mathbf{y}_i$ with set $\mathcal{N}_K(\mathbf{x}')$ containing the indices of the K-nearest neighbors of $\mathbf{x}'$. A distance-weighted variant has been introduced by Bailey and Jain [2] to smooth the prediction function weighting the prediction with the similarity $\Delta(\mathbf{x}', \mathbf{x}_i)$ of the nearest patterns $\mathbf{x}_i$ with $i \in \mathcal{N}_K(\mathbf{x}')$ to the target $\mathbf{x}'$

$$\mathbf{f}_{wKNN}(\mathbf{x}') = \sum_{i \in \mathcal{N}_K(\mathbf{x}')} \frac{\Delta(\mathbf{x}', \mathbf{x}_i)}{\sum_{j \in \mathcal{N}_K(\mathbf{x}')} \Delta(\mathbf{x}', \mathbf{x}_j)} \mathbf{y}_i. \tag{2}$$

Patterns close to the target should contribute more to the prediction than patterns that are further away. Similarity can be defined with the distance between patterns, e.g. by

$$\Delta(\mathbf{x}', \mathbf{x}_i) = 1/\|\mathbf{x}' - \mathbf{x}_i\|^2. \tag{3}$$

Model $\mathbf{f}_{wKNN}$ introduces a continuous output.

### 3.2 Support Vector Regression

For prediction, we employ one of the most popular tools in the field of machine learning, i.e., support vector machines, which can be used for classification, regression, and a variety of other learning settings [6, 10, 11]. In general, the basis for training appropriate models is a set $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\} \subset \mathbb{R}^d \times \mathcal{Y}$ consisting of labeled patterns. For classification settings, the space $\mathcal{Y}$ of labels is discrete (e. g., $\mathcal{Y} = \{-1, +1\}$ for the binary case). For regression scenarios, the space $\mathcal{Y}$ is given by $\mathbb{R}$; here, the goal of the learning process consists in finding a prediction function $f : \mathcal{X} \to \mathbb{R}$ that maps unseen patterns $\mathbf{x} \in \mathcal{X}$ to reasonable real-valued labels. These models can be seen as a special instance of problems having the form

$$\inf_{f \in \mathcal{H}, b \in \mathbb{R}} \frac{1}{N} \sum_{i=1}^{n} L\big(y_i, f(\mathbf{x}_i + b)\big) + \lambda ||f||_{\mathcal{H}}^2, \tag{4}$$

where $\lambda > 0$ is a user-defined real-valued parameter, $L : \mathbb{R} \times \mathbb{R} \to [0, \infty)$ a loss function and $||f||^2_{\mathcal{H}}$ the squared norm in a so-called *reproducing kernel Hilbert space* $\mathcal{H} \subseteq \mathbb{R}^{\mathcal{X}} = \{f : \mathcal{X} \to \mathbb{R}\}$ induced by an associated *kernel function* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. The space $\mathcal{H}$ contains all considered models, and the term $||f||^2_{\mathcal{H}}$ is a measure for the *complexity* of a particular model $f$ [6]. Forecasts based on machine learning models require large data sets and advanced knowledge about the underlying data mining process chain. Important aspects are model selection, data preprocessing, definition of learning sets, and parameter tuning.

### 3.3 Related Work

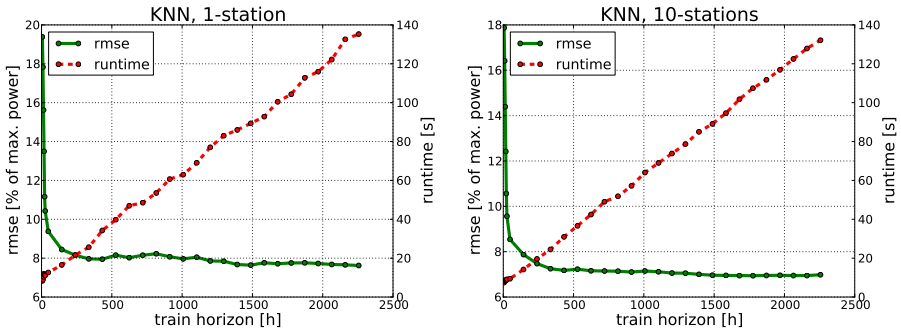
An early approach for PV power prediction is the autoregressive integrated moving average (ARIMA) model [4]. A related comparison of models is given by Reikard [9]. Popular prediction models are based on neural networks, see [8]. Similar to our approach, [3] employs forecasts from numerical weather predictions as input of their models. Fonseca *et al.* [5] employ SVR for prediction of the power of a 1-MW photovoltaic power plant in Kitakyushu, Japan. Many approaches are based on cloud motion vectors and numerical weather predictions. For a comprehensive survey, we refer the interested reader to the overview by Lorenz and Heinemann [7].

## 4 Nearest Neighbor PV Predictions

In the following, we present the result of our experimental study. We try to answer the questions, (1) how the training sets should look like, (2) which features are important, and (3) which KNN model to choose, i.e., uniform and distance-weighted KNN and the choice of $K$ for a 1-hour ahead prediction of PV power.

### 4.1 Training Set

First, we concentrate on the choice of a proper training set size. For this sake, we choose uniform KNN with $K = 5$ and the features time $T$, power $P$, clear sky $CS$, temperature $Temp$, irradiance $I$ and cloud coverage $CC$. For a single station, Figure 2 shows the runtime and the corresponding RMSE for increasing training set size. The runtime describes the average time needed to build the regression models and to perform the predictions of one week (168 hours). We can observe that the runtime is linearly increasing, while the error is decreasing exponentially. Table 1 shows the learning results of KNN for the single station w.r.t. two different training set sizes. The results show that the RMSE can be reduced for increasing training set sizes. Further, the table shows the maximum error, which is the largest deviation between time series and prediction w.r.t. the maximum power that has been generated. Our analysis leads to the choices $N = 1,488$ (1-station) and $N = 1,776$ (10-stations) yielding acceptable RMSE values, corresponding to 62 and 74 training days.




(a) (b)

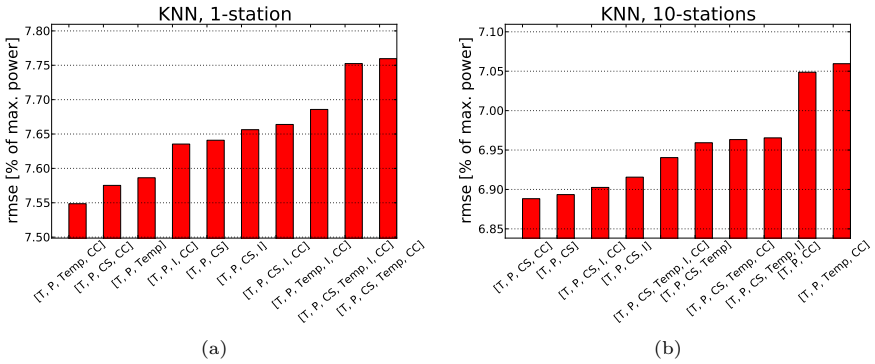
**Fig. 2.** RMSE and runtime w.r.t. an increasing training set size of KNN. The runtime is linearly increasing, while the error is decreasing exponentially. Acceptable values are achieved with $N = 1,488$ (1-station) and $N = 1,776$ (10-stations).

**Table 1.** Analysis of training set size w.r.t. RMSE, maximal error and runtime of KNN. The settings we choose for the following experiments are shown in bold.

| KNN, 1-station | | | |
|---|---|---|---|
| train horizon [h] | RMSE [%] | max error [%] | runtime [s] |
| 2256 | 7.6220 | 32.8909 | 135.2831 |
| **1488** | **7.6412** | **35.1116** | **89.3406** |
| 2160 | 7.6564 | 32.8900 | 132.5754 |
| KNN, 10-stations | | | |
| train horizon [h] | RMSE [%] | max error [%] | runtime [s] |
| **1776** | **6.9404** | **32.1766** | **107.4191** |
| 1680 | 6.9461 | 32.3517 | 101.7518 |

### 4.2 Features

Now, we analyze the features that are useful for PV predictions. For this sake, we employ the training set size from the last experiment, and test all $2^6 - 1 = 63$ feature combinations. Figure 3 shows the ten best feature combinations for PV predictions sorted w.r.t. the resulting RMSE of (a) one station and (b) ten stations. We can observe that the lowest RMSE is achieved with the feature combination $T, P, Temp, CC$ in both settings. Table 2 shows the RMSE with maximum error and runtime for two feature combinations for the single station experiment (upper part) and the 10-stations setting (lower part). The optimal feature set in case of the single station experiment achieves an RMSE that is not only lower in average, but also w.r.t. the maximum error achieved, here in comparison to the feature set $T, P, CS, CC$, which achieved the second best RMSE, and $TP, P, CS, CC$. All accuracies could be improved in comparison to the previous experiments.




(a) (b)

**Fig. 3.** The ten best feature combinations for PV prediction with KNN sorted w.r.t. resulting RMSE for the 1-station and the 10-stations experiment.

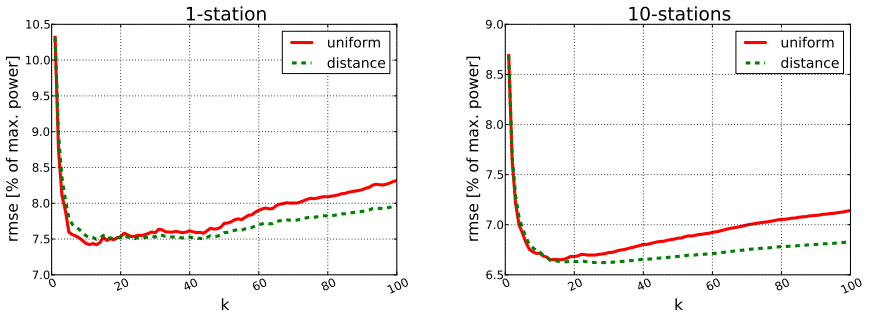
**Table 2.** Feature combinations: RMSE, maximum error and runtime for two feature combinations of KNN for the 1- and the 10-stations experiment.

| KNN, 1-station | | | |
|---|---|---|---|
| pattern | RMSE [%] | max error [%] | runtime [s] |
| **T, P, Temp, CC** | **7.5486** | **28.1711** | **64.9916** |
| T, P, CS, CC | 7.5753 | 32.9269 | 64.3939 |
| KNN, 10-stations | | | |
| pattern | RMSE [%] | max error [%] | runtime [s] |
| **T, P, CS, CC** | **6.8883** | **32.1027** | **66.6126** |
| T, P, CS | 6.8935 | 32.1847 | 49.8370 |

### 4.3 Model Parameters

As last part of our KNN study, we analyze the regression model employed. We compare uniform to distance-weighted KNN and various settings for $K$. Figure 4 shows the influence of the neighborhood size $K$ for uniform and distance-weighted KNN on the RMSE for (a) the 1-station experiment and (b) the 10-station experiment. Both figures show a similar result. Small choices for $K$ lead to large errors, while settings around 10 to 20 lead to good results. Then, the error is increasing again, faster for the uniform variant. This is probably due to the missing ability of uniform KNN to take the similarity of patterns into account. The choice of $K = 13$ led to optimal results in average in both cases.

The upper part of Table 3 compares uniform and distance weighted KNN w.r.t. RMSE, maximum error and runtime in seconds, for the 1-station setting, the lower part shows the same comparison for the 10-stations setting. In three settings, the optimal choice for the neighborhood size was $K = 13$. For the 10-stations experiment, $K = 28$ achieved the best results. The best learning results are again marked in bold, i.e., in case of the 1-station experiment uniform KNN was superior, while distance-weighted KNN achieved the best results on the 10-stations experiment.




(a) (b)

**Fig. 4.** Analysis of neighborhood size $K$ in terms of RMSE for uniform and distance-weighted KNN on the 10-stations setting.

**Table 3.** Comparison of models (uniform vs. weighted) and choice of neighborhood size $K$ w.r.t. RMSE, maximal error and runtime in seconds. The best results are marked in bold.

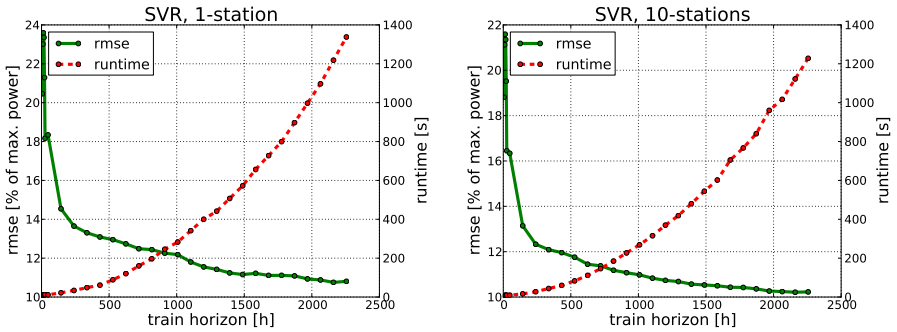
| KNN, 1-station | | | | |
|---|---|---|---|---|
| weight | K | RMSE [%] | max error [%] | runtime [s] |
| **uniform** | **13** | **7.421** | **29.7085** | **65.1795** |
| distance | 13 | 7.4968 | 29.8822 | 60.8313 |
| KNN, 10-stations | | | | |
| weight | K | RMSE [%] | max error [%] | runtime [s] |
| uniform | 13 | 6.6477 | 30.6246 | 66.0068 |
| **distance** | **28** | **6.6224** | **30.3252** | **66.7710** |

## 5 Support Vector Regression PV Predictions

In the following, we analyze the same line of experimental analysis for SVR, i.e., we concentrate on optimal training set sizes, feature combination and model parameters for a 1-hour ahead PV prediction.

### 5.1 Training Set

Again, we start with the training set size experimentally. Figure 5 shows RMSE and runtime of SVR w.r.t. increasing training set sizes. The SVR employs and RBF kernel with $C = 100$ and $\gamma = 0.0$, as well as $\epsilon = 0.1$ for the $\epsilon$-insensitive loss. All features are used. The left plot shows the results for the 1-station experiments, the right part shows the corresponding results for the 10-stations experiment. Similar to KNN, we can observe that the accuracy increases dramatically at the beginning with an increasing training set size. But the runtime is not increasing linearly, but exponentially with the training set size. This increase can be problematic in case of very large PV data set sizes. Table 4 shows RMSE, maximum error and runtime for selected settings for the 1-station setting




(a) (b)

**Fig. 5.** RMSE and runtime w.r.t. an increasing training set size of SVR. The runtime is linearly increasing, while the error is decreasing exponentially. An acceptable value is the choice $N = 1,680$ in both cases.

in the upper part and the 10-stations setting in the lower part. Finally, we chose the size $N = 1,680$ for the following experiments. The best results are shown in bold. Without further feature selection and model parameter tuning, the SVR results are worse than the results of KNN.

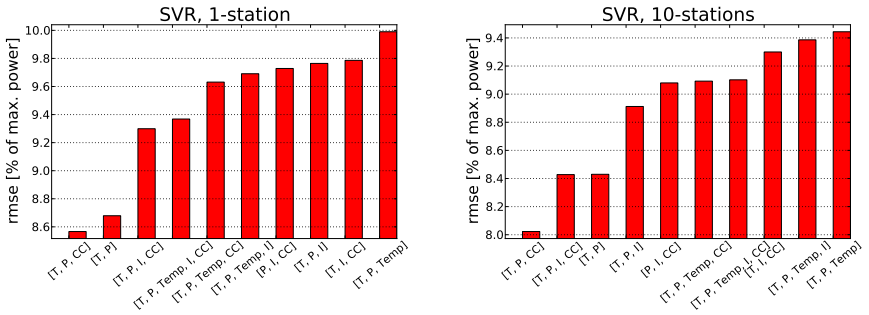
**Table 4.** Analysis of training set size w.r.t. RMSE, maximal error and runtime of SVR. The training set size we choose for the following experiments are shown in bold.

| SVR, 1-station | | | |
|---|---|---|---|
| train horizon [h] | RMSE [%] | max error [%] | runtime [s] |
| 2160 | 10.7541 | 35.5728 | 1218.3753 |
| 2064 | 10.8784 | 36.0253 | 1096.9466 |
| 1872 | 11.0882 | 36.2599 | 897.5565 |
| **1680** | **11.1119** | **37.0376** | **727.5991** |
| 1776 | 11.1142 | 36.7121 | 799.5511 |
| SVR, 10-stations | | | |
| train horizon [h] | RMSE [%] | max error [%] | runtime [s] |
| 2160 | 10.2115 | 36.1512 | 1122.1244 |
| 2064 | 10.2416 | 36.2138 | 1016.8459 |
| 1872 | 10.3610 | 36.3251 | 840.4265 |
| **1680** | **10.4314** | **36.5362** | **705.1353** |
| 1584 | 10.4976 | 36.7964 | 601.9663 |

### 5.2 Features

Now, we analyze, which feature combination lead to the best results for SVR-based predictions. Figure 6 shows the ten best feature combinations for the SVR-

based prediction. The combination $T, P, CC$ achieves the best results in the 1- and the 10-stations setting. Table 5 shows the achieved results and highlights the best combinations. The best value we achieved was an RMSE of 8.5665, which is still worse than the KNN results. Also a higher maximum error was achieved.




(a) (b)

**Fig. 6.** The ten best feature combinations for PV prediction with SVR sorted w.r.t. resulting RMSE for the 1-station and the 10-stations experiment.

**Table 5.** Feature combinations: RMSE, maximum error and runtime for two feature combinations of SVR for the 1- and the 10-stations experiment.

| SVR, 1-station | | | |
|---|---|---|---|
| pattern | RMSE [%] | max error [%] | runtime [s] |
| **T, P, CC** | **8.5665** | **37.1568** | **286.9199** |
| T, P | 8.6790 | 37.9553 | 201.9782 |
| SVR, 10-stations | | | |
| pattern | RMSE [%] | max error [%] | runtime [s] |
| **T, P, CC** | **8.0228** | **35.6517** | **302.9437** |
| T, P, I, CC | 8.4277 | 39.2239 | 534.5487 |

### 5.3 Model Parameters

In case of the SVR regression model, we have to tune three parameters. An analysis of the $\epsilon$-insensitive loss led to an optimal setting of $\epsilon = 0.11$ in case of the 1-station experiment and $\epsilon = 0.20$ for the ten stations. The analysis of $\epsilon$ is not presented here due to a lack of space. Figure 7 shows the RMSE depending on the SVR parameters $C$ and $\gamma$. The RMSE space employs large plateaus of high errors for bad parameter settings, but good results for $C$ in the range of $10^5$ to $10^{10}$ and $\gamma = 1.0$ to $10^{-10}$ for both experimental settings. The optimal

settings are $C = 5.5 \cdot 10^2$ and $\gamma = 5.0 \cdot 10^{-3}$ for the 1-station experiment and $C = 8.25 \cdot 10^2$ and $\gamma = 4.0 \cdot 10^{-3}$ for the 10-stations experiment. These settings are employed in the model comparison of the following section.

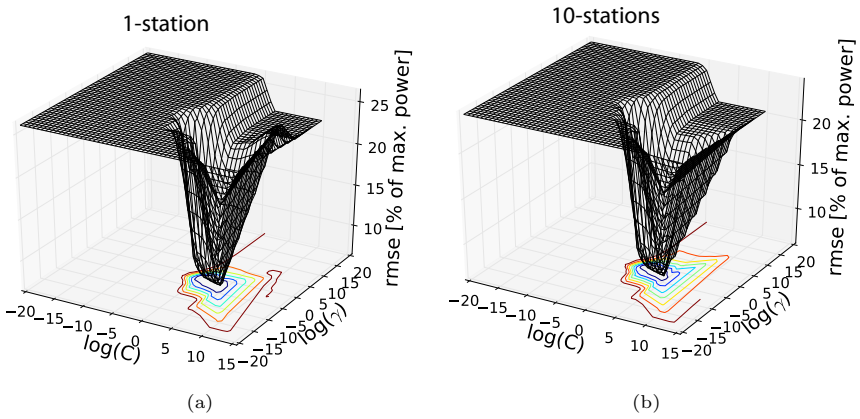
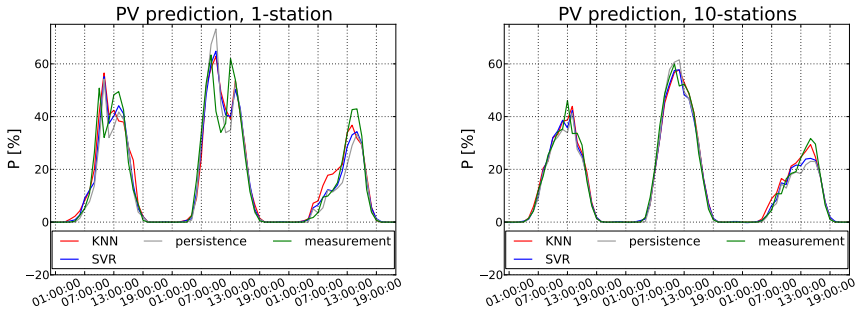



**Fig. 7.** SVR parameter study of $C$ and $\gamma$ for (a) the 1-station and (b) the 10-stations experiment.

## 6 Comparison of Models

After the analysis of training sets, features and parameterizations of the models, we compare KNN and SVR on two test time series, i.e., one week and one month in 2011, with optimal settings. We further compare to a comparatively simple model called persistence that combines the last power value with the clear sky ratio, i.e., $P(t) = P(t-1) \cdot CS(t)/CS(t-1)$. Figure 8 shows the real PV time series (green), the KNN prediction (red), the SVR prediction (blue) and the persistence model (grey). We can observe that all predictions capture the main curvature of the PV energy, while the persistence model tends to overestimate the power, while KNN and SVR are rather conservative at the areas of the highest peaks. Further, KNN tends to overestimate the upswing part of the power curve.

Table 6 helps to estimate the quality of the predictions showing RMSE and maximum error for all three models. For the 1-station setting, the SVR achieves the best results w.r.t. RMSE and also the maximum error. But KNN achieves better results w.r.t. the maximum error in the one-month experiments. Further, we performed a forecast for all 87 stations. Here, the models are trained for each of the 87 stations and the forecasts are aggregated to a single prediction in each step. Also in the 87-stations experiments, the SVR was the best model for the one-week experiment in terms of RMSE and maximum error, but only achieved the lowest RMSE in case of the one-month experiment. But the maximum error was lower in case of the persistence model.




(a) (b)

**Fig. 8.** Comparison between real PV time series, the KNN and SVR prediction with optimal parameters and the persistence model (2011-08-05 – 2011-08-07).

**Table 6.** Comparison of RMSE and maximum error for persistence model (pers.), KNN and SVR prediction with optimal settings for one station, ten stations and all available 87 stations.

| | 1 week (2011-08-01 – 2011-08-07) | | | | | |
|---|---|---|---|---|---|---|
| | 1 station | | 10 stations | | 87 stations | |
| | RMSE [%] | max [%] | RMSE [%] | max [%] | RMSE [%] | max [%] |
| pers. | 6.8850 | 31.2146 | 3.0073 | 12.1833 | 3.0568 | 14.6546 |
| KNN | 6.6323 | 27.0242 | 2.8598 | 14.5074 | 3.0196 | 12.5647 |
| SVR | **6.2506** | **25.5280** | **2.8093** | **11.7109** | **2.9329** | **11.6726** |
| | 1 month (2011-09-25 – 2011-10-24) | | | | | |
| | single station | | 10 stations | | 87 stations | |
| | RMSE [%] | max [%] | RMSE [%] | max [%] | RMSE [%] | max [%] |
| pers. | 5.3578 | 28.5739 | 2.4657 | 12.2225 | 2.6370 | **11.7102** |
| KNN | 6.1641 | **26.9346** | 2.7729 | **11.4225** | 3.0437 | 12.5947 |
| SVR | **4.5610** | 27.4008 | **2.3261** | 12.0418 | **2.5387** | 12.1567 |

The RMSE is very low, in particular for the 10-stations setting and the experiment with 87 PV stations, in particular in comparison to the 1-station setting. This also holds for the maximum deviation, which is comparatively good, in particular under 20%, which is a rule-of-thumb value for good predictions in practice. The good values in comparison to the single station experiments are probably due to an averaging effect.

## 7 Conclusions

The prediction of PV power has an important part to play in smart grids. Our experiments have shown that the often employed persistence model based on the clear sky index already achieves good results. But we could improve the PV forecasts with statistical learning methods. In particular, SVR turned out

to show competitive results for a single PV station. The quality of the predictions significantly depends on the proper choice of training sets, features and model parameters. We analyzed the predictions with this regard and identified recommendations, in particular for model parameters $K$ for KNN and $C, \gamma$ and $\epsilon$ in case of SVR with $\epsilon$-insensitive loss. These recommendations are tailored to the employed data sets and may vary for other settings and scenarios. But they turned to be comparatively robust in our experimental line of research. In summary, SVR turned out to be the best model in the experiments for the single station. As future work, we plan to enhance the model to a hybrid classifier that takes advantage of the capabilities of more than one regression model.

**Acknowledgements**

We thank meteocontrol - energy & weather services (http://www.meteocontrol.com/) for providing the PV station measurements and the ECMWF (http://www.ecmwf.int/) for the numerical weather predictions that are basis of our experimental analysis.

## References

1. P. Bacher, H. Madsen, and H. A.Nielsen. Online short-term solar power forecasting. *Solar Energy*, 83(10):1772–1783, 2009.
2. T. Bailey and A. Jain. A note on distance-weighted k-nearest neighbor rules. *IEEE Transaction on Systems, Man and Cybernetics*, 8(4):311–313, 1978.
3. J. Cao and S. Cao. Study of forecasting solar irradiance using neural networks with preprocessing sample data by wavelet analysis. *Energy*, 31(15):3435–3445, 2006.
4. B. Chowdhury. Short-term prediction of solar irradiance using time-series analysis. *Energy Sources*, 12(2):199–219, 1990.
5. J. da Silva Fonseca, T. Oozeki, T. Takashima, G. Koshimizu, Y. Uchida, and K. Ogimoto. Photovoltaic power production forecasts with support vector regression: A study on the forecast horizon. In *Photovoltaic Specialists Conference (PVSC), 2011 37th IEEE*, pages 002579–002583, 2011.
6. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, Berlin, 2009.
7. E. Lorenz and D. Heinemann, editors. *Prediction of Solar Irradiance and Photovoltaic Power*, volume 1 of *Comprehensive Renewable Energy*. Springer, 2012.
8. A. Mellit. Artificial intelligence technique for modelling and forecasting of solar radiation data - a review. *Int. J. Artif. Intell. Soft Comput.*, 1(1):52–76, 2008.
9. G. Reikard. Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy*, 83(3):342–349, 2009.
10. B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.
11. A. J. Smola and B. Schölkopf. *A tutorial on support vector regression*. Statistics and Computing, 14:199–222, Kluwer Academic Publishers, Hingham, MA, USA, 2004.