



Machine-learning-based forecasting of distributed solar energy production

Lead organisations: **NICTA & ANU**

Project commencement date: **1 January 2013**

Completion date: **30 June 2016**

Date published: **1 July 2016**

Contact name: **Robert C. Williamson**

Title: **Prof Dr**

Email: **bob.williamson@nicta.com.au**

Phone: **+61 2 6218 3712**

Website: **<http://solar.research.nicta.com.au>**



This project was lead by Prof Dr Bob Williamson, NICTA and ANU, and managed by Dr Christfried Webers, NICTA.

The following researchers contributed to the writing of this report:

chapter	contributor
1	Dr Christfried Webers (NICTA)
2	Dr Christfried Webers (NICTA)
3	Dr Nicholas Engerer (NICTA and ANU)
4	Dr Paulette Lieby (ANU)
5	Dr Xinhua Zhang (NICTA), Dr Justin Domke (NICTA)
6	Dr Xinhua Zhang (NICTA), Dr Miaomiao Liu (NICTA)
7	Dr Aditya Menon (NICTA), Dr Mark Reid (ANU)
8	Dr Christfried Webers (NICTA)
9	Dr Christfried Webers (NICTA)
10	Dr Justin Domke (NICTA), Dr Aditya Menon (NICTA)
11	Shaw Kudo (ANU), Dr Nicholas Engerer (ANU)
12	Dr Miaomiao Liu (NICTA)

We are very grateful to have received support from the following partners:

partner	kind of support
Telstra	supporting our project with sim cards for over hundred data loggers and covering the wireless communication cost for the data transmission
SolarHub	providing us with data about the solar energy production from their installations
ActewAGL	providing us insight into their operations and for financial support for two summer scholar students
AEMO	providing us with a better understanding of the Australian energy market



Contents

I Executive Summary	15
1 Executive Summary	17
1.0.1 Subject matter	17
1.0.2 Methods of analysis	17
1.0.3 Findings	18
1.0.4 Conclusions	19
1.0.5 Recommendations	19
1.0.6 Limitations	20
II January 2013 - June 2015	21
2 Data Loggers and Cameras	23
2.1 Overview	23
2.2 Objectives	23
2.3 Evidence of completion	23
2.3.1 Cameras	23
2.3.2 Data loggers	25
2.3.3 Data collection system	25
2.4 Extent to which the objectives were achieved	26
2.5 Highlights, difficulties and breakthroughs	26
2.6 Conclusions and recommendation	26
3 PV measurements	27
3.1 Objective and Overview	27
3.2 Evidence of completion	27
3.2.1 Data Acquisition	27
3.2.2 Dataset Curation	28
3.3 Conclusion	28
4 Sky images	35
4.1 Overview	35
4.2 Objectives	35
4.3 Evidence of completion	35
4.3.1 Creating high dynamic range images	35
4.3.2 Data Portal	54
4.4 Extent to which the objectives were achieved	54
4.4.1 Hardware and data	55
4.4.2 HDR image reconstruction	55
4.4.3 HDR image quality	55
4.5 Conclusion	56



5 Machine learning forecasting	57
5.1 Overview	57
5.2 Objectives	57
5.3 Evidence of completion	58
5.3.1 Identify important features for building predictors: baseline benchmarking	58
5.3.2 Improving forecasting accuracy by using distributed PV network	74
5.3.3 Simulators for more insights	80
5.4 Extent to which the objectives were achieved	83
5.5 Highlights, difficulties and breakthroughs	83
5.6 Conclusions and recommendation	84
6 Forecasting with image features	87
6.1 Overview	87
6.2 Objectives	87
6.3 Evidence of completion	87
6.3.1 Visual feature extraction	87
6.3.2 Forecasting with image features	89
6.4 Extent to which the objectives were achieved	91
6.5 Highlights, difficulties and breakthroughs	91
6.6 Conclusions and recommendations	92
7 Performance measures	93
7.1 Overview	93
7.2 Objectives	93
7.3 Evidence of completion	93
7.3.1 Measures of performance that serve multiple needs	94
7.3.2 Adapting ML methods to different measures of performance	97
7.4 Extent to which the objectives were achieved	100
7.5 Highlights, difficulties and breakthroughs	101
7.6 Conclusions and recommendations	101
III July 2015 - June 2016	103
8 Installation of 100 data loggers in the ACT region	105
8.1 Overview	105
8.2 Objectives	105
8.3 Approval Procedures: Human Ethics, ActewAGL, Telstra	105
8.3.1 Human Ethics application	105
8.3.2 ActewAGL	106
8.3.3 Telstra	106
8.4 Finding and Selecting Residential Homes	107
8.4.1 Finding volunteers to receive a data logger	107
8.4.2 Selection criteria for the optimal data logger locations	107
8.5 Inspection of candidate locations and interviews with home owners	108
8.6 Installation	108
8.7 Highlights	108
8.8 Conclusions	109
9 Collection, Analysis and Publication of Data	111
9.1 Overview	111
9.2 Objectives	111
9.3 Overview of data collection	111
9.4 Data Analysis of data logger data	112
9.4.1 Clear sky day	112



9.4.2 Clear sky day with additional voltage measurements	112
9.4.3 Impact of shadows on the power output	113
9.4.4 Cloudy day	114
9.4.5 Irregular output	114
9.5 Publication of Data	115
9.6 Evidence of completion	115
9.7 Conclusions and recommendations	115
10 Loss-calibrated predictions	117
10.1 Overview	117
10.2 Objectives	117
10.3 Loss-calibrated probabilistic modelling: theory	118
10.3.1 Probabilistic modelling	118
10.3.2 Loss-calibrated predictions	118
10.3.3 Loss-calibrated probabilistic modelling	119
10.4 Loss-calibrated probabilistic modelling: optimisation	119
10.4.1 Asymmetric loss with closed-form a^*	120
10.4.2 Asymmetric loss with semi-closed form a^*	120
10.5 Instantiation for distributed solar prediction	121
10.6 Experiments on loss-calibration	122
10.6.1 Probability models	122
10.6.2 Losses	123
10.6.3 Experimental results	124
10.7 Multi- versus single-site prediction	126
10.7.1 Multi-site loss functions	126
10.7.2 Optimising multi-site loss functions: two broad approaches	131
10.7.3 An equivalence when using linear models and square loss	131
10.8 Experiments on multi- versus single-site prediction	132
10.9 Extent to which the objectives were achieved	133
10.10 Highlights, difficulties, and breakthroughs	134
10.11 Conclusion and recommendations	135
11 Sub-sampled PV Monitoring Data	137
11.1 Overview	137
11.2 Objective	137
11.3 Evidence of Completion	137
11.3.1 Acquiring Zone Sub-station Data from ActewAGL	137
11.3.2 Analysis of Sub-sampled PV Data	139
11.3.3 Analysing Total Solar Energy Production via RPSS version 1	143
11.4 Extent to which the objectives were achieved	144
11.5 Conclusions and Recommendations	145
12 Forecasting from Sky Images with Deep Convolutional Neural Networks	149
12.1 Overview	149
12.2 Objectives	149
12.3 Evidence of completion	149
12.3.1 Learning solar power via DCNN	149
12.3.2 DCNN structure	150
12.3.3 Data preparation and solar power prediction	151
12.4 Extent to which the objectives were achieved	153
12.5 Highlights, difficulties and breakthroughs	153
12.6 Conclusions and recommendations	153



IV Outreach and Knowledge Sharing 155

13 Outreach and knowledge sharing 157
13.1 Conference/journal papers 157
13.2 Conferences presentations 157
13.3 Trade shows 158
13.4 Government and industry visits 158
13.5 Press releases 158
13.6 Student projects related to this project 158



List of Figures

2.1	XIMEA MU9PM-MH (© XIMEA)	24
2.2	Complete camera enclosure (left), dome removed (middle), and disassembled lens and sensor (right). (The remaining space in the enclosure is filled with moisture absorbing SILIKA-Gel to prevent condensation of water inside the dome in the morning.)	24
2.3	Camera locations (Map: © Google Earth)	25
3.1	Total number of sites actively reporting from PVOutput.org over the duration of the project	29
3.2	Total number of sites actively reporting from PVOutput.org as of December 2014	30
3.3	Map denoting target areas for PV data logger deployment	31
3.4	Map showing the location of the ACT Solar Schools data	32
3.5	Example of data transferred via 3G from an Arduino based data-logger, communicating with an SMA inverter via RS485	33
3.6	Randomly fluctuating data, which is deemed to be erroneous by the QCPV algorithm and removed. The blue line shows the clear sky curve, which shows overnight reported PV data and periods during which the clear sky power output is exceeded (which are physically impossible)	33
3.7	Cloud enhancement events are identified using a detection algorithm in the QCPV routine	34
3.8	Determination of the tilt and azimuth of a given PV system, accomplished by minimising the RMSE between the observed and simulated clear sky data.	34
4.1	The camera response function $f : \log(eX) \mapsto b$ for Ximea MU9PC-MH.	36
4.2	Additive response for the 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$; the graph on the right shows a restricted range.	37
4.3	Two additive response functions: desired is the reference response (using the 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$); the other response shown uses the set of exposures $\{2^{15}, 2^{16}, 2^{17}, 2^{18}\} \mu\text{s}$, and the objective function has value 72.0694.	38
4.4	The irradiance detectable range of the set of 20 exposures. Right: images are full sky images; Left: images are from inside an office comprising part of the sky and hence display a larger irradiance range.	38
4.5	Collated results for camera CAM000, 20/10/2014: MBS. The red bar indicates the number of times an MBS occurs in the collated results. The green square dots indicate the value of the objective function. We note that among the most frequent occurring configurations for that day, the configuration $\{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$ had the smallest value for the objective function.	39
4.6	Collated results for camera CAM000, 20/10/2014: value of the objective function for configurations of similar size to $\{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$. The red bar indicates the number of times the configurations occur in the collated results. The green square dots indicate the value of the objective function. We note that the configuration $\{2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$ occurs most often and has the smallest value for the objective function.	39
4.7	Comparing the three construction methods <code>bestp</code> , <code>add</code> , and <code>addw</code> for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$	42



4.8 Comparing the two construction methods <code>bestp</code> and <code>addw</code> for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$	43
4.9 Comparing Mertens' algorithm with the three construction methods <code>bestp</code> , <code>add</code> , and <code>addw</code> for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$	43
4.10 Comparing composite images from recreated exposures and original exposures for the three construction methods <code>bestp</code> , <code>add</code> , and <code>addw</code>	44
4.11 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> and <code>addw</code>	44
4.12 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> , <code>addw</code> , and <code>add</code> ; the composite images were created from the eleven exposures $2^0, 2^{10}, \dots, 2^{19} \mu\text{s}$ (images taken on 14/12/2014).	45
4.13 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> , <code>addw</code> , and <code>add</code> ; the composite images were created from the ten exposures $2^{10}, \dots, 2^{19} \mu\text{s}$ (images taken on 14/12/2014).	46
4.14 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> , <code>addw</code> , and <code>mertens</code>	46
4.15 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> and <code>mertens</code>	47
4.16 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> , <code>addw</code> , and <code>mertens_from_addw</code>	47
4.17 Comparing composite images from recreated exposures and original exposures for <code>bestp</code> and <code>mertens_from_addw</code>	48
4.18 An example of noise in the <code>bestp</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image.	49
4.19 An example of noise in the <code>bestp</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image.	49
4.20 An example of noise in the <code>bestp</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image.	50
4.21 An example of posterisation in the <code>bestp</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image.	50
4.22 An example of posterisation in the <code>bestp</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image. The images are obtained from the $2^1, 2^{10}, \dots, 2^{19} \mu\text{s}$ exposures.	50
4.23 An example of posterisation in the <code>addw</code> image. Left: a <code>bestp</code> irradiance map; Right: an <code>addw</code> irradiance map for the same image.	51
4.24 Left: <code>mertens</code> image corresponding to the image in Figure 4.18. Right: <code>mertens</code> image corresponding to the image in Figure 4.19.	51
4.25 Left: <code>mertens</code> image corresponding to the image in Figure 4.20. Right: <code>mertens</code> image corresponding to the image in Figures 4.21 and 4.22.	52
4.26 <code>mertens</code> image corresponding to the image in Figure 4.23.	52
4.27 Left: composite image for <code>mertens</code> . Right: composite image for <code>mertens_from_addw</code> . Note the posterisation in the <code>mertens_from_addw</code> which is not present in the <code>mertens</code> image.	53
4.28 Left: composite image for <code>mertens</code> . Right: composite image for <code>mertens_from_addw</code> . The images are comparable, although <code>mertens_from_addw</code> shows a slight posterisation (difficult to detect here).	53
4.29 Left: composite image for <code>mertens</code> . Right: composite image for <code>mertens_from_addw</code> . The images are comparable, although <code>mertens_from_addw</code> shows a slight posterisation (difficult to detect here).	54
4.30 A snapshot of the data portal.	55



5.1	253 PV systems (sites) distributed across Canberra. The data is retrieved from PVOutput.org, with sites gradually added since year 2011.	74
5.2	The rated power output for the 29 sites. The height of each bar is proportional to the power of the site.	74
5.3	Scatter plot of RMSE for persistence, local, and joint predictors with horizon being 10 minutes	76
5.4	Scatter plot of RMSE for persistence, local, and joint predictors with horizon being 30 minutes	76
5.5	Scatter plot of RMSE for persistence, local, and joint predictors with horizon being 60 minutes	76
5.6	Mean test RMSE as a function of the number of nearest neighbours for joint predictor.	77
5.7	Mean test RMSE as a function of time lag interval for local and joint predictor.	78
5.8	Mean test RMSE as a function of number of quarters back in history for local and joint predictor.	78
5.9	RMSE in kW _r for different methods (best viewed in colour). From left to right: i (red): the baseline persistency predictor, ii (blue): local predictor, with missing data filled by the mean of other sites, iii (green): joint predictor with missing data filled by the mean of other sites, iv (brown): local predictor with missing data filled with Gaussian, v (black): joint predictor with missing data filled with Gaussian.	79
5.10	Some examples of simulation with random clouds and 500 randomly scattered sites.	80
5.11	Examples of occlusion tracks and simulated power (as a fraction of total site capacity) for a day. Each line corresponds to one site (10 total).	80
5.12	Local regularisation with the simulated data	81
5.13	The benefits of feature expansion of the wind.	82
5.14	Interpolated occlusion maps.	82
5.15	Estimated vs. true wind directions.	82
5.16	Errors with estimated vs. true wind directions.	82
5.17	Online vs. batch training.	84
5.18	RMSE in kW _r for modeling clear-sky and shade (best viewed in color). From left to right: i. ignoring the two flags as we did above (red bar), ii. using the two flags by training four models (blue bar), and iii. training and testing <i>only</i> on clear-sky time steps (green bar). Some sites do not have any clear-sky time in 2011 and 2012, hence the corresponding third bar is not available.	85
6.1	Illustration of computing radiance features. A set of circular regions around the projected sun location in the image are considered, with increasing radius. Then compute the the mean radiance within all circles.	88
6.2	Comparison between the projection of the sun trajectory using the calibrated camera (blue dots) with the ground truth (red dots). In particular, this figure shows projection of the sun trajectory on 30 September, 2014. The comparison shows that our calibration algorithm can produce comparatively accurate projection results.	88
6.3	Compute the optical flow between two consecutive image frames. (a) and (b) show the consecutive image frames. (c) shows the computed optical flow encoded by rgb colours. Note that similar colour denotes the pixels move in the similar direction.	89
6.4	Correlation of global radiance with features under various prediction horizons.	90
7.1	PV output forecast time series A and B.	96
7.2	Incomes of two forecasts with the same absolute error, over a single day. Also shown is the income from an optimal forecast.	97
7.3	Performance of probabilistic forecasts when trained on single day.	99
7.4	Dollar gain from using probabilistic forecasts over expected PV output when bidding.	100
9.1	Data collection with two virtual machines (VM) and one rack mounted 50 TB Dell server with 20 TB extension.	112
9.2	Typical power output reported for a clear sky day.	113
9.3	Typical power output and voltage reported for a clear sky day with the upgraded data loggers.	113
9.4	Typical power output for a location with shadows from neighbouring obstacles.	114
9.5	Typical power output for a day with many patches of clouds.	114
9.6	Example of power output for a switched-off PV system.	115
9.7	Locations of 100 data loggers installed in the ACT region.	116



10.1 Comparison of truncated Gaussian and standard Gaussian distributions for various choices of mean μ and standard deviation σ	123
10.2 Beta distribution for various choices of hyperparameter (α, β)	124
10.3 The asymmetric M-Loss for various values of tuning parameter a	125
10.4 Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss.	126
10.5 Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Gaussian distribution with fixed variance. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.	127
10.6 Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Gaussian distribution with learned variance. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.	128
10.7 Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, truncated Gaussian distribution. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.	129
10.8 Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Beta distribution. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.	130
10.9 Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, M-Loss.	134
10.10 Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, smooth pinball loss.	135
10.11 Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, square loss.	135
11.1 Example Zone Substation Data for 12 Zones in the ActewAGL Network. The blue vertical lines bound a negative ramp event and the black vertical lines, a positive ramp event.	138
11.2 Example ramp event date showing a negative ramp event and a positive ramp event, both shown in Figure 11.1. The black line indicates the clear-sky power output from all monitored PV systems, and the red line the mean observed power output from all monitored PV systems. Units are in kW/kW_p (observed power output divided by installed capacity).	138
11.3 ActewAGL Distribution Network Zone Substations (from Energy Consumer Reference Council Background Information Kit, 2014).	139
11.4 Location of the 100 PV sites utilised in this analysis. Red dots indicated the location of the reporting PV system. Black polygons display the boundaries of Canberra suburbs.	140
11.5 Location of the selected PV sites for analysis filtered according to substation location. Blue dots indicate sites assumed to fall in the chosen substations.	141
11.6 Correlations between filtered quality controlled sites (QC output) and substation power outputs for thirty-minute intervals for 5 March 2014.	141
11.7 Correlations between quality controlled sites and feeder based power outputs for five minute intervals for 5 March 2014.	142
11.8 Correlations observed for five minute interval RPSS_v1 data set for collective ramp event on 5 March 2014.	143
11.9 Correlations observed for five minute interval RPSS_v1 data set for collective ramp event on 5 March 2014.	144
11.10 Correlations observed for subsetted data set for collective ramp event on 5 March 2014.	145



12.1	Architecture of Alex net [?]. C- $\{i\}$ denotes the convolutional layers and Fc- $\{j\}$ denotes the fully-connected layer. Details about the size of the network are referred to Table 12.1.	150
12.2	VGG net architecture. C- $\{i\}$ denotes the convolutional layers and FC denotes the fully-connected layers. Details about the size of the network are referred to Table 12.2.	151



Australian Government
Australian Renewable Energy Agency

ARENA



List of Tables

2.1 Camera locations (http://www.findlatitudeandlongitude.com , precision 1'')	24
5.1 Mean bias error on Radiation data.	64
5.2 Mean absolute error on Radiation data.	65
5.3 Normalized root mean squared error on Radiation data.	66
5.4 Coefficient of determination on Radiation data.	67
5.5 Mean bias error on PV output data.	68
5.6 Mean absolute error on PV output data.	69
5.7 Normalized root mean squared error on PV output data.	70
5.8 Coefficient of determination on PV output data.	71
5.9 Summary of Radiation data.	72
5.10 Summary of PV output data.	73
5.11 Mean RMSE on all times steps of persistence model, local predictor, and joint predictor.	75
5.12 Mean RMSE on non-clear-sky time steps of persistence model, local predictor, and joint predictor.	75
6.1 RMSE (Wm-2) of global radiation at ANU Engineering Building.	90
6.2 RMSE (Wm-2) of direct radiation at ANU Engineering Building.	91
7.1 Summary of recommended performance measures for various market participants.	95
7.2 Performance of probabilistic forecasts when trained on single day.	98
7.3 Comparing the mean absolute error of joint predictor based on squared and absolute error.	100
8.1 Number of data loggers for each disclosure option	108
8.2 Number of data loggers installed in 64 suburbs.	109
10.1 Average loss incurred when training with a given loss for each probabilistic model.	125
10.2 Comparison of direct and indirect optimisation of aggregate statistics.	133
11.1 High level mapping of suburbs by zone substation	140
11.2 Correlation coefficients for analysis of thirty minute intervals for 5 March 2014.	142
11.3 Correlation coefficients for analysis of five minute intervals for 5 March 2014.	142
11.4 Correlation coefficients for analysis of ramp event subset five minute intervals for 5 March 2014.	143
11.5 Correlation coefficients for analysis of five minute intervals RPSS_v1 data for 5 March 2014.	144
11.6 Correlation coefficients for analysis of subset five minute intervals for 5 March 2014 using RPSS.	145
11.7 List of all critical collective ramp events for which correlations were tested during this study	146
12.1 Alex net configuration. The convolutional layer parameters are denoted as conv<receptive field size>-<number of channels>. The fully-connected layers are denoted as Fc-<number of neurons>.	151
12.2 VGG net configuration. The convolutional layer parameters are denoted as conv<receptive field size>-<number of channels>. The fully-connected layers are denoted as Fc-<number of neurons>.	152



- 12.3 RMSE and AbsE (Wm-2) of **global** radiation at ANU Engineering Building using network tailored from ALEX net. We evaluated on three subset, such as Training set, validation set and test set, respectively. 152



Part I

Executive Summary



Chapter 1

Executive Summary

1.0.1 Subject matter

A high uptake of photovoltaic (PV) systems creates instability in the electricity grid due to the intermittency of sun light. If there exists no reliable method to estimate the contribution of distributed residential PV systems to the power grid, this leaves grid operators vulnerable to power quality issues.

This project explores real-time data mining from a few hundred widely distributed residential PV systems combined with image data from inexpensive sky cameras. The forecast horizon was chosen to lie between 5 minutes (the shortest time to place a bid on the energy market) and 60 minutes (lead time for diesel or gas turbines to provide additional power if necessary).

The number of all PV systems is often much larger than the number of systems from which data can be collected. Statistical modelling can return averages which closely describe the behaviour of the whole system without collecting data for each single part of the system. A spatially distributed system of PV generators changes its power output rapidly when clouds move in or out. It therefore has to be exploring how many data sources need to be sampled in which frequency in order for the resulting model to represent the behaviour of all PV systems.

The quality of publicly available real-time PV data varies largely due to the fact that most of the data are created and recorded by individuals. Therefore, this project also chose to create a network of 100 low-cost data loggers installed in residential homes which measure the power output of the PV system in short intervals. The data loggers are distributed over the whole ACT providing high quality measurements of the PV system power.

1.0.2 Methods of analysis

The project collected PV output data from a few hundred residential home in Canberra, ACT, over several years. As this data only represent current and past measurements, we also developed cost-effective sky cameras providing a peek into the future by detecting cloud patterns on the horizon which will later impact the amount of radiation falling onto a PV system.

We then developed machine learning technology to learn a model from both of the data streams and applied the model to predict the future power generation at different time intervals ahead. After time had passed, we measured the real power output and compared it to the prediction in order to evaluate how good the prediction was.

Special consideration was given to explore different types of performance measures as the choice of how one measures errors is intrinsically related to how well the multiple market players (electricity generators and distributors, market operators, financiers, and customers) can make decisions using those predictions.

With the goal of decision making in mind, the project developed state-of-the-art research results to define new performance measures for the joint forecast over distributed sites, and demonstrates the real-world value of these performance measures. We studied several forms of asymmetric performance measure which take into account that over-predicting and under-predicting a target value may incur very different losses, i.e. under-predicting the current in a transmission line may destroy the line if the real current is much bigger than the predicted value; over-predicting would not damage anything.



To find out how well a relative small sub-sample of PV system data can represent the total energy measured at a grid sub-station we approached the local energy provider, ActewAGL. They provided us with sub-station data for several suburbs in Canberra. We then compared the total energy flow with the measurements and predictions modelled with the help of data from a relatively small number of data loggers.

1.0.3 Findings

The predictions based on a distributed network of sensors achieved a significantly higher accuracy compared to localised predictors. The implication of the developed method can be applied to other areas where spatio-temporal data from different sensor types are fused together to create predictions. Interestingly, we found that for noisy and incomplete data, increasing the amount of data for training or using a more complex model does not necessarily result in better prediction results when the model is applied to predict from unseen new data. This is consistent with the theory of statistics and machine learning and necessitates extra care to devise methods (e.g. regularisation) which allow the finding of wanted patterns in the data without overfitting to the noise of the training data.

Our prediction algorithm was trained with data from 253 sites over the whole year 2013. The learned model was then used to predict the solar energy production for 2014 with three different horizons. The predictor outperforms a persistence model by 9.16%, 18.4%, and 31.1% on 10-min, 30-min, and 60-min horizons respectively. It is not surprising that a persistence model is quite competitive for a 10-min horizon, because generally there is only mild change in such a short period. However when the horizon gets longer, the advantage of a joint predictor becomes much more substantial. A predictor without information from surrounding sites provides nearly half as much improvement as a predictor taking information from neighbouring sites into account, which corroborates the importance of aggregating the PV output of other sites for prediction accuracy.

By using statistical features of cloud images and inverter data, our approach bypasses the high cost of sensors, the model complexity and the demand for computing power associated with an exact 3-D reconstruction of the clouds and their movement. The radiance features we extracted from sky images boost the forecasting accuracy, can be computed efficiently and locally at each camera, and can be transmitted very economically as a very low density data stream over existing wireless networks.

Although the camera manufacturer would not provide us with guarantees for the life time of the camera system under the harsh conditions of full sun light, we can report that the addition of a neutral density filter of at least 6 notches has prevented the cameras from becoming damaged. The first installed camera has been operating for more than two years.

We developed a novel loss-calibrated probabilistic modelling view of solar PV prediction. The idea of probabilistic forecasting is by itself relatively novel in the solar forecasting community. We are not aware of any prior work on loss-calibrating such forecasts for PV prediction. We believe our experiments indicate the promise of such approaches.

We also presented a means of easily obtaining loss-calibrated predictions via automatic differentiation. This solves the non-trivial problem of finding model parameters that result in the best optimal actions, where the latter are implicitly defined as the result of a nested optimisation problem.

We showed the value of directly optimising for a joint or aggregate performance measure, such as total PV output in an area, compared to simply optimising for the average of local measures. This further illustrates a point we have made before, namely that if one fixes a performance measure, then optimising for it directly can be beneficial. We showed that in an interesting special case of square loss, the joint and local approaches are however identical.

Our research work demonstrates a suitable method for quantifying the relationship between distributed PV power output and zone sub-station load data, and has determined that 30-minute interval data are generally insufficient for these purposes, and that 5-minute interval data are required. Furthermore, to truly analyse these relationships, regional PV simulations have been demonstrated to be crucial to fully investigating the relationships between distributed PV power output and distribution network loads. Given a robust and accurate distributed PV modelling framework, strong correlations can be observed between distributed PV power output and network loads during critical collective ramping events.

Machine-learning based approaches for prediction rely heavily on designing the right features as input to a model. We explored the application of a Deep Convolutional Neural Network (DCNN) architecture and tailored it for the solar power prediction task. Our primary results show that the DCNN is able to automatically learn



useful visual features from cloud images.

The project installed a network of 100 low-cost data loggers in residential homes distributed over 64 suburbs in the ACT (including a few locations in neighbouring NSW). These data loggers report measurements of the power (and voltage) of each PV system at 5 minute intervals and provide a very valuable source of data for research. Data logger data and cloud images are collected in a robust system designed to survive wireless network and computer outages.

Both PV system data and cloud images, are published respecting the privacy concerns of the participants. About 30% of the home owners chose to not publish their exact location, but a location with noise of maximal 100 metres added.

1.0.4 Conclusions

The project has shown that it is possible to develop a statistical model for solar prediction which outperforms existing methods without building a 3-D cloud model. The machine learning technology employed fuses information from two different data sources collected with inexpensive sensors: power output from PV systems and sky images. The amount of data necessary from each sensor station is so small that it can easily be transmitted via existing wireless network technology.

The inclusion of cloud images providing information about clouds which are still far away can improve the predictive power of the model. Given the high data rate resulting from raw images, our research in automatic feature extraction shows promising results. More effort will be necessary in order to reduce the processing time and transmission rate to a level which is achievable in a low-cost camera system.

Our simulations and analysis of performance measures show that there is considerable benefit from going beyond naïve measures of performance, such as the squared error between predicted and observed forecast. Further progress along this lines would represent a significant aid in the decision-making capabilities of various participants in the energy market.

1.0.5 Recommendations

The availability of distributed data does improve the prediction accuracy for solar power generation. We therefore recommend to design measures to make more data available for research and forecasting. It is very inexpensive to add data loggers to a PV system at installation time, and the communication cost are very low. Often home owners themselves are interested in monitoring their PV system. As the power grid is a shared resource, regulators and the power companies need to apply more effort to collect the data necessary for making informed decisions.

A number of cameras located on the periphery of Canberra would allow the acquisition of earlier signals about changing cloud conditions than the currently installed three cameras which are concentrated in the centre. As the cameras we designed are inexpensive, this could be easily achieved. Based on the prototype camera we built, it will become feasible to create a stand-alone camera system with a small computer and a 4G wireless transmission which is completely solar cell/battery powered.

More joint research bringing together experts in computer vision and solar energy forecasting could result in better predictive features for cloud images reducing the amount of data which has to be recorded or increasing the predictive power of the forecast.

We expect that it is possible to design better probabilistic models for PV, in particular accounting for heteroskedastic variance. Training loss-calibrated versions of such models would be of interest. Future work could incorporate uncertainty in the model parameters due to limited sample sizes in the loss-calibration framework. This would allow for more fully Bayesian inference to be performed in a loss-sensitive way. Future research should investigate other aggregate performance measures, e.g. taking into account the precise geographical layout of sites.

The design and implementation of a realistic market simulator could enable a more fine-grained understanding of the economic impact of mis-predictions and uncertainty for the market participants. Experiments in a simulated environment would allow to do research with well defined scenarios which otherwise could not be found under the very complex conditions of a real market. Furthermore, it could also be used as a planning tool to understand infrastructure and policy changes in the power energy market.



1.0.6 Limitations

This project has collected and processed data from Canberra only. The relatively large number of days in Canberra where there is either clear sky or a dense cloud cover makes it hard to improve on existing methods which just predict the ideal PV system output given on the position of the sun. The developed methods should therefore be even more powerful if applied to data from locations which have more days with changing cloud cover than Canberra.



Part II

January 2013 - June 2015



Chapter 2

Data Loggers and Cameras

2.1 Overview

One of the unique goals of the project was the fusion of different data streams in the form of cloud images and inverter data from distributed sensors. As those data sources were not available when the project was created it was necessary to develop low-cost prototype data loggers and inexpensive sky cameras and install them in locations distributed over the Australian Capital Territory (ACT).

2.2 Objectives

Develop and deploy low-cost prototype data loggers and inexpensive sky cameras to one hundred residential and commercial PV sites in the ACT.

2.3 Evidence of completion

2.3.1 Cameras

The development of a prototype for the camera system was based on the following principles

- Small form factor for the sensor to allow for a compact final system.
- Enough pixel resolution to enable research and experiments with a higher resolution than a camera in a final product would need.
- No mechanical parts in the camera as those would not be able to withstand the temperature variations (below zero in winter and fifties to sixties exposed to the sun light in summer). This rules out lenses with mechanical shutters which are also more expensive than cameras with a rolling shutter.
- Having a system where the sensor is all the time exposed to the sun, the sensor needs to be protected with a strong neutral density filter.
- Having a neutral density filter demands a higher sensitivity from the camera as the reduced radiation can not be arbitrarily compensated for by extending the exposure time of the camera. The reason is that in situations of strong wind where the clouds move fast, exposure times need to be kept short in order to record images of the clouds which are not blurred.

For the above reasons, we chose the **XIMEA MU9PM-MH** mini camera with a size of $15 \times 15 \times 8$ mm and a maximal resolution of 2592×1944 pixels.

The camera system was assembled into an enclosure which is large enough to hold the final system together with a mini-PC like a Raspberry-Pi system. In the prototype, we connected an Intel NUC DC53427HYE to



Figure 2.1: XIMEA MU9PM-MH (© XIMEA).



Figure 2.2: Complete camera enclosure (left), dome removed (middle), and disassembled lens and sensor (right). (The remaining space in the enclosure is filled with moisture absorbing SILIKA-Gel to prevent condensation of water inside the dome in the morning.)

record and transmit the images to a central data collection system. Further details of the data collection system can be found in sub-section 2.3.3.

Three sky cameras (denoted CAM000, CAM001, CAM002) are now installed on ANU and NICTA buildings.

CAM000	School of Engineering, ANU
Coordinates	Lat 35° 16' 34" S, Long 149° 7' 11" E
Installed & Recording started	October 2013
Neutral density filter	8 stops until 26/09/2014, 9 stops since 27/09/2014
CAM001	NICTA Canberra Research Laboratory
Coordinates	Lat 35° 16' 54" S, Long 149° 7' 30" E
Installed & Recording started	April 2014
Neutral density filter	6 stops
CAM002	Fenner School of Environment and Society, ANU
Coordinates	Lat 35° 16' 41" S, Long 149° 6' 55" E
Installed & Recording started	May 2015
Neutral density filter	10 stops

Table 2.1: Camera locations (<http://www.findlatitudeandlongitude.com>, precision 1")

As the manufacturer could not provide us with information about the durability of the sensor, we decided to operate the second camera with a weaker filter than necessary in order to explore how the sensor behaves under stronger radiation.

The images are used for two main purposes:

- the estimation of the amount of radiation falling on some region of the sensor (focus on high radiation regions), and



Figure 2.3: Camera locations (Map: © Google Earth).

- the estimation of cloud flow (low radiation regions).

It turned out that this is not possible without taking multiple exposures and combining them into high dynamic range images. This will be explained in detail in chapter 4.

2.3.2 Data loggers

A major hurdle in collecting data on the output side of the inverter system, is the fact that one would need to interfere with power cables carrying 240 V and/or a power meter in a residential home. Subsequent talks with Canberra's electricity provider ActewAGL convinced us that this was a dead end. We therefore searched for alternatives and found a contactless system using a current clamp provided by IT Suppliers. By regulation in the ACT, all residential solar systems must have an extra power meter measuring the output of the solar system. A current clamp can measure the output current which multiplied by the norm voltage of 240 V provides the current power output. The error is the sum of the error stemming from the voltage fluctuation and the error of the current estimation.

Currently, we have 12 data loggers installed with 28 more data loggers in the pipeline to be installed in the coming weeks.

2.3.3 Data collection system

The data collection system needs to take into account that each data provider (residential home owner, institutions like ANU) wants to preserve the security of their connection to the internet. That implies that the data streams should not force the data provider to open extra ports on their firewall. From this consideration follows:

- Data are pushed from the sensor (camera, data logger) to a publicly available network node using secured authentication methods. We call this network node a ‘collector’.
- The collector serves as a buffer from which data are then accessed and transferred into the final database server(s) sitting behind a firewall.



- Further measures have to be taken if a camera system is not allowed to connect to a wired network because facility regulations do not allow this. The main reason for this is to prevent lightning to destroy wired network infrastructure. For one of our cameras this was the case. As the data rates we wanted to work with did also not allow us to use the campus-wide wireless network, we had to create our own dedicated wireless link from the camera computer to a computer inside of the building which then relayed the data further to the collector node.

All computers in our data collection network are running Ubuntu 14. The rsync utility has been proven to be very important to secure the data flow from one end to the other. Rsync ensures that data are only deleted on the sending side, if it has been successfully transferred to the receiving side.

2.4 Extent to which the objectives were achieved

This project developed three camera systems and installed them onto locations which form a triangle with edge lengths in the range 0.46 . . . 0.97 kilometer.

A prototype 3G data logger, capable of communicating with SMA inverters was created through an engineering project at The ANU. This product was not advanced to the deployment stage. The reason for this, is that other products became commercially available during the course of the project, which were capable of completing the required 3G enabled data logging tasks.

A commercially available contactless data logger system is now being installed on 40 residential locations. We have augmented this by collecting 5 minute interval data from more than 300 sites, of which currently 170 are actively reporting. For further details please see chapter [3](#).

2.5 Highlights, difficulties and breakthroughs

This project developed three camera systems which are now fully functional and provide a data stream of about 23 GB of high-resolution data per camera per day.

The development of our own data loggers has been stopped as a) cheap commercial data loggers became available, and b) the necessary effort to secure legal access and liability when physically accessing inverters have turned out to be enormous.

Circumventing the direct physical access to the inverters, contactless data loggers have been purchased and - again for legal reasons - installed by professional electricians.

Additional effort has been put into reaching a legal agreement with SolarHub to get access to historical data from residential installations. Together with publicly available data from PVoutput.org we got a data set of a few hundred sites distributed over Canberra.

2.6 Conclusions and recommendation

The machine learning research of the project needs data of high quality. When the project was setup, the effort of developing, building and installing data loggers into many residential homes was underestimated. Furthermore, the initial investment into such a large installation is huge and should be offset by a long-term plan to utilise the collected data.

For the camera systems, we successfully developed low-cost camera system prototypes. Initial development was on the Microsoft Windows operating system. It turned out that Linux did not only provide much better support for secure data transmission tools, but also comes as the operating system of choice on many mini-computer systems like Raspberry-Pi. We therefore switched to Linux and developed all software in a portable way to allow easy transfer to those min-computer systems in the future.



Chapter 3

PV measurements

3.1 Objective and Overview

This project established an objective of measuring photovoltaic (PV) system power output at five minute intervals across five distributed suburbs for one year to create a high-quality curated benchmark data set which was to be made freely available to the community. This chapter will explore how and to what extent this objective was achieved.

3.2 Evidence of completion

The collection, analysis and curation of PV data in the Canberra region has two key aspects, which we have broken into into two sub-sections covering data acquisition and data curation.

3.2.1 Data Acquisition

Project data was acquired from several sources: PVOutput.org, the ACT “Solar Schools” program, through cooperation with local solar installer SolarHub and, lastly, through the distribution of our own project-funded data-logger devices. The project also developed a device which may make future PV based data collection simpler.

3.2.1.1 SolarHub and PVOutput.org

Primarily, data were collected from PVOutput.org. In total, 300 sites reported data over the course of the project, with 170 sites actively reporting at the end of 2014 (see Figure 3.1 and 3.2). These sites were well-distributed over the whole of the ACT, with excellent coverage developing through the end of 2014. The majority of these sites (212 total installations) were installed by local solar installer and project partner SolarHub. The data reported to PVOutput.org is generally not suitable for advanced scientific research, due to the uncertainty in PV system characteristics and frequent interruptions in power output reporting. However, since so many of these sites were professionally installed and maintained by SolarHub, we were able to use this data for higher level research. SolarHub was able to provide exact system characteristics, including module and inverter types, exact location and layout and the orientation of the solar arrays in space. Therefore, this partnership with SolarHub allowed us to scale-back our initial target of project funded data-logger deployments (100+), which we re-directed to sparse reporting regions (see Figure 3.3), while simultaneously expanding our project ambitions to record PV system power output data across all of Canberra.

3.2.1.2 ACT Solar Schools

Power output data was also extracted from 70+ sites through two student projects at The ANU. These projects analysed the data recorded by the ACT Solar Schools programs, building an additional database of hourly-interval solar energy system data. Although this data was not integrated into the project forecasting experi-



ments, it generated an additional, high value data-set. This allowed the students to complete some basic PV analytics projects, and report these results in their honours theses. One of these students went on to author a publication on this analysis project [2]. A map of these sites is included in Figure 3.4.

3.2.1.3 Development of a Novel 3G Data-logger

This project also supported two student projects for hardware development, which developed a prototype 3G enabled data-logger. This data-logger was constructed via the Arduino platform, and communicates directly with widely used inverters from SMA Solar Technology AG over its RS485 bus interface before reporting its data to a central server via 3G (Figure 3.5).

3.2.2 Dataset Curation

The aforementioned PV system data collection requires extensive quality control procedures, as there are significant inaccuracies in the reported power output data and uncertainty regarding the system characteristics. To deal with this, one of our researchers authored a comprehensive quality-control algorithm for PV system power output, entitled QCPV. This code is capable of analysing PV system power output to detect the orientation of PV systems in space, and quality control their power output.

3.2.2.1 QCPV

A manuscript which details this algorithm has been submitted to a refereed journal, and will likely be published in late 2015. Given the unpublished nature of this scientific work, only a basic overview of the quality control algorithms will be provided herein. The primary tool for quality controlling the PV data in the QCPV routine is the PV system's clear sky power output curve. The clear sky power output curve provides physical limits for the observed data, and allows for the removal of physically improbable data. Physically improbable data includes most often values which are too large, or reported overnight. The clear sky curve is generated through a series of radiation and PV system models through the methods outlined in [1]. Another important tool used in the QCPV routine is "across-system" comparisons. By comparing the power output recorded over a given time period from several nearby PV systems, it is possible to detect spurious/random behaviour from a given system (See Figure 3.6). This also makes it possible to retain real features in the data, which may appear to be erroneous. A prime example of this are cloud enhancement events, where PV system power output temporarily exceeds the clear sky threshold when sunlight is reflected off the sides of nearby clouds. These data are real and should be retained in the processed dataset. The QCPV algorithm uses across-system testing to determine if cloud enhancement events are occurring at more than one site in the region, in order to decide if they should be eliminated or retained (Figure 3.7).

3.2.2.2 Determining PV System Orientation

Given the extraordinary importance of the clear sky curve in the QCPV algorithm, accurate PV system 'meta-data' is required. This metadata consist of the characteristics of a given PV system, including its location, module and inverter type, layout and orientation in space. In nearly all cases, the metadata provided to this project included very specific information about the location and module and inverter types for each installations. The tilt and azimuth of these systems, however, was not always provided, and in some cases, was incorrect. To amend this, an algorithm was developed to determine the tilt and azimuth of a given PV system, using a time-series of its power output. This was accomplished by extracting the clear sky periods from the reported power output, and then matching it to a tilt and azimuth combination which minimises the difference between the observed and simulated clear sky data. The outcome of this process is shown in Figure 3.8, where the Root Mean Square Error (RMSE), between extracted clear sky periods and simulated clear sky data is minimised for a tilt and azimuth of 12.5 and -20 degrees.

3.3 Conclusion

The project objective was to measure PV system power output at five minute intervals across five distributed suburbs for one year to create a high-quality curated benchmark data set which we will make freely available

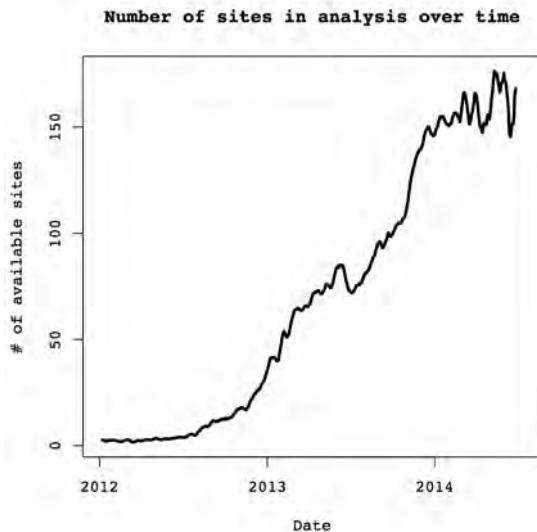


Figure 3.1: Total number of sites actively reporting from PVOutput.org over the duration of the project

to the community. Given the sampling of the existing network in Canberra, and the quality control routines implemented, it is very clear that we have completed this objective. There is a single caveat to this however: the data publication will likely take a few additional months of processing, as we'd like to include the data recorded by the data-loggers which we will soon deploy. We will release a limited dataset at the official conclusion of the project, with an expected follow-up posting of a more comprehensive dataset by the end of 2015.

Furthermore, we are very close to achieving high-density coverage (1-3 sites/suburb) across the entirety of Canberra. Although we initially expected to use our own data logging network to achieve this goal, we were able to accomplish it via our industry partnership with SolarHub. By the end of the project, we were able to obtain five minute interval data from over 300 PV sites, 170 of which are currently reporting. This made all of the required analysis, testing and product development possible. Now, at the conclusion of the project, we have plans to install 40 data-loggers into regions where the existing network under-samples. This will ensure that the entirety of the ACT is monitored at sufficient density for further experimentation. There are plans to maintain this monitoring network for the next year, with The ANU actively pursuing additional ARENA funding (with SolarHub as an industry partner) to extend this timeline further.

Bibliography

- [1] Nicholas A. Engerer and Franklin P. Mills. Kpv: A Clear-Sky Index for Photovoltaics. *Solar Energy*, July(105):679–693, 2014.
- [2] J Tan, Nicholas A. Engerer, and Franklin P. Mills. Estimating Hourly Energy Generation of Distributed Photovoltaic Arrays : a Comparison of Two Methods. In *Proceedings of the 52nd Annual Conference, Australian Solar Energy Society (Australian Solar Council)*, number May, 2014.



Locations of Reporting Sites Dec 2014



Figure 3.2: Total number of sites actively reporting from PVOutput.org as of December 2014



Locations of Reporting Sites Dec 2014

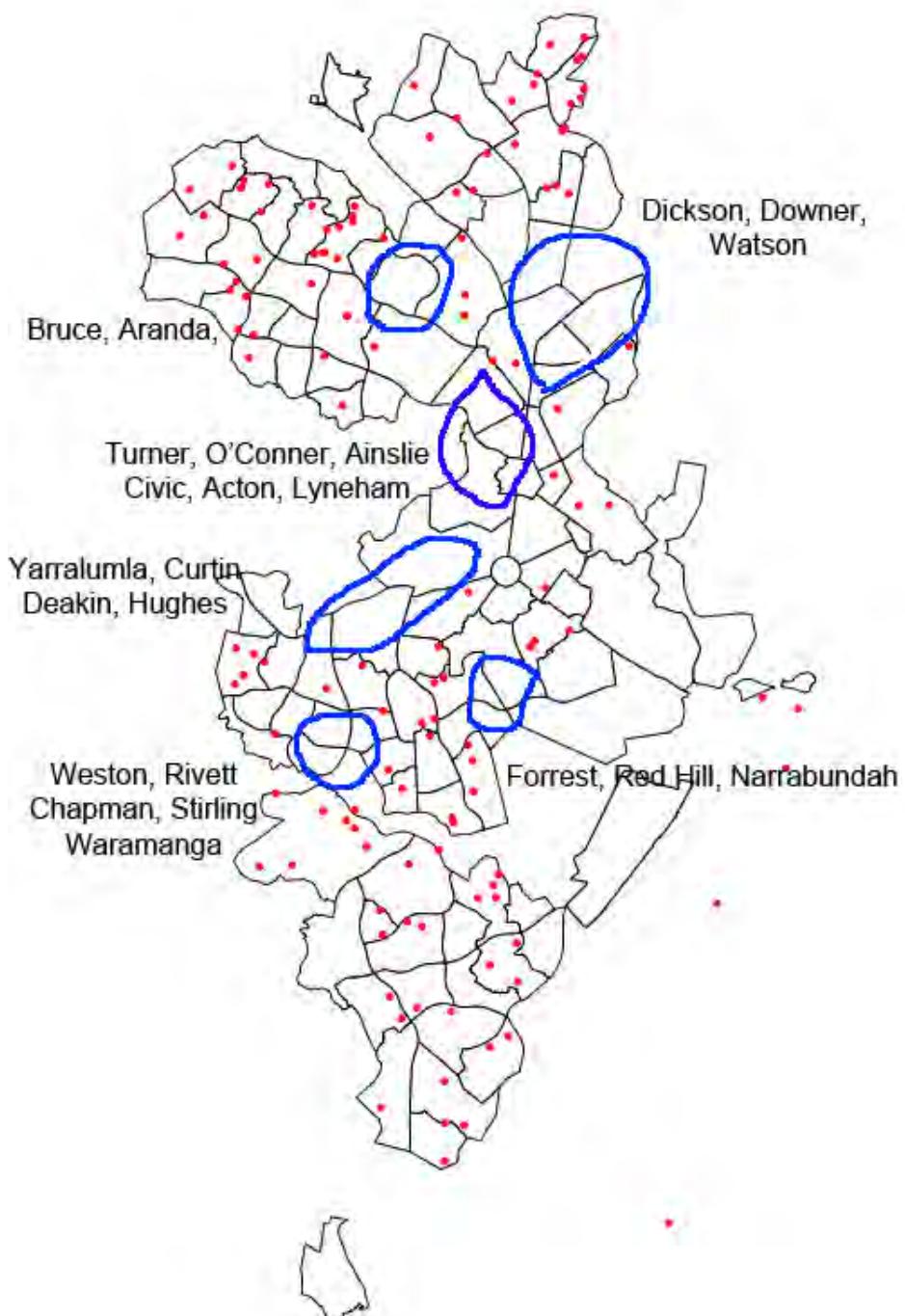


Figure 3.3: Map denoting target areas for PV data logger deployment

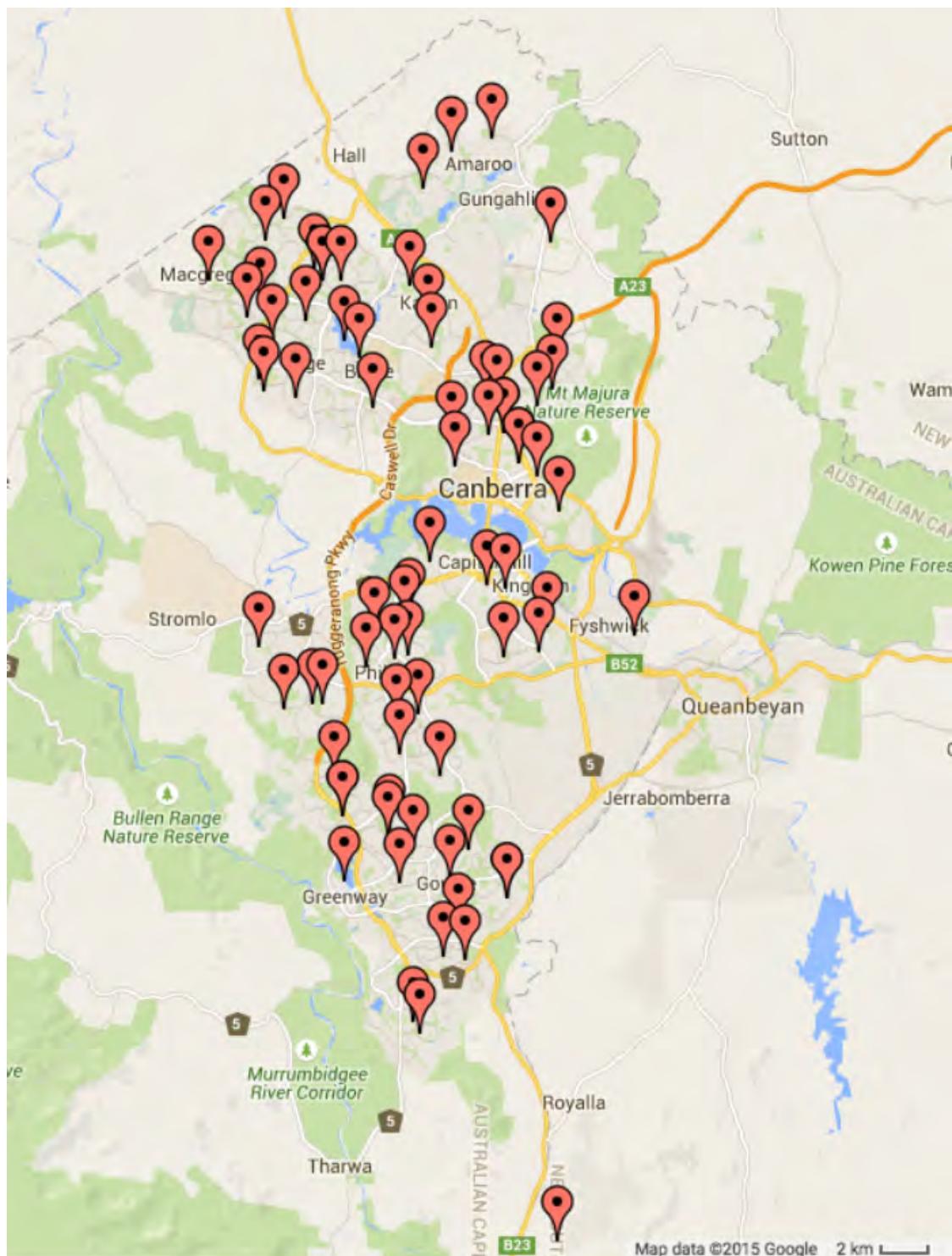


Figure 3.4: Map showing the location of the ACT Solar Schools data

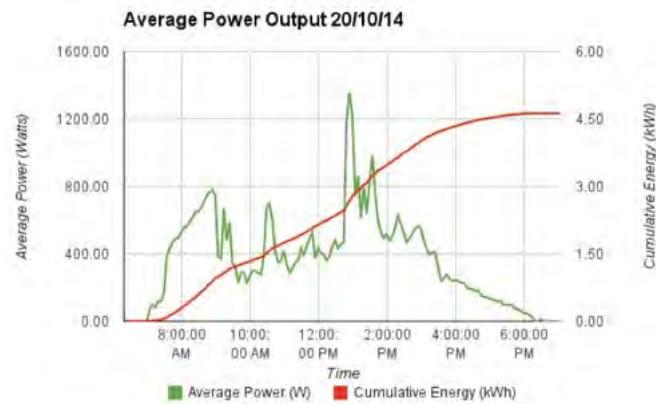


Figure 3.5: Example of data transferred via 3G from an Arduino based data-logger, communicating with an SMA inverter via RS485

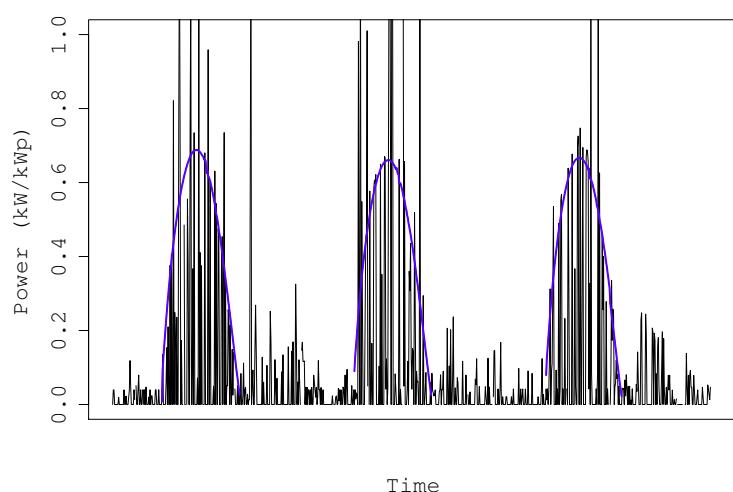


Figure 3.6: Randomly fluctuating data, which is deemed to be erroneous by the QCPV algorithm and removed. The blue line shows the clear sky curve, which shows overnight reported PV data and periods during which the clear sky power output is exceeded (which are physically impossible)

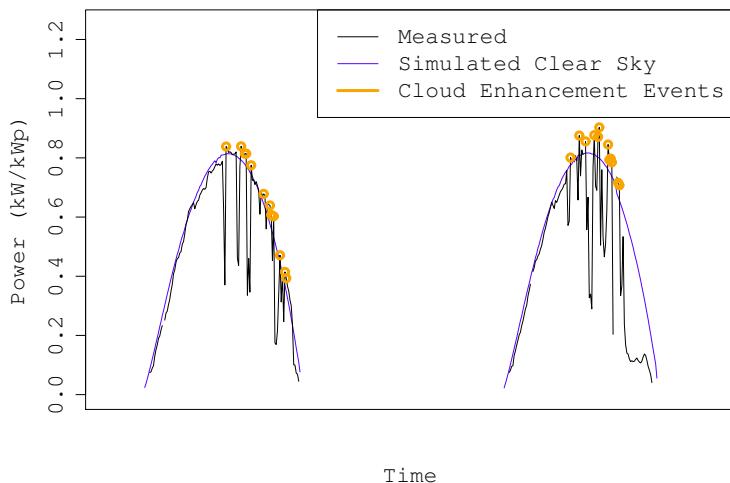


Figure 3.7: Cloud enhancement events are identified using a detection algorithm in the QCPV routine

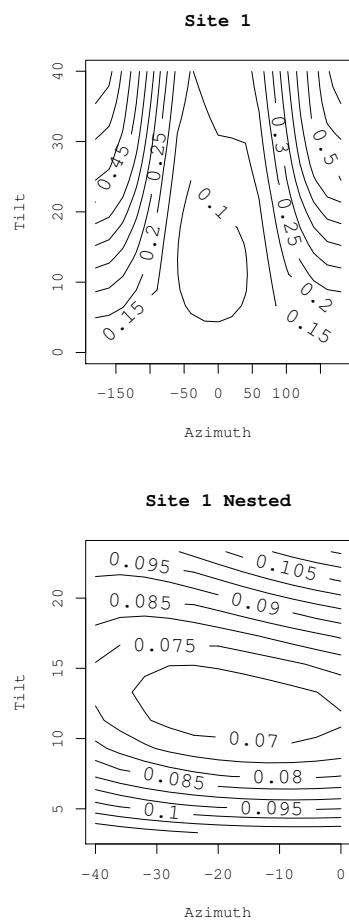


Figure 3.8: Determination of the tilt and azimuth of a given PV system, accomplished by minimising the RMSE between the observed and simulated clear sky data.



Chapter 4

Sky images

4.1 Overview

This section describes the sky image capture system designed and deployed as a complement to the PV sensor network: there are three sky image cameras programmed to capture images until 30 June 2016. See Chapter 2 for complete details about their specifications and locations. The cameras are named CAM000, CAM001, and CAM002 respectively.

Image processing is performed at the time of capture, and images are transferred to a data server. Cameras may be connected wirelessly to the internet (one camera in our case) or via wire (two cameras) to allow for image transfer.

The dynamic range in images that include the sun is much larger than can be captured in a single exposure, so a series of exposures is combined in the image processing to create a larger dynamic range composite image.

The image data is publicly available at <http://solar.research.nicta.com.au>.

4.2 Objectives

Record sky images at five minute intervals at locations selected to complement the PV sensor network, and make the data freely available.

4.3 Evidence of completion

4.3.1 Creating high dynamic range images

The cameras perform sky capture from dawn to dusk, seven days a week. Sky images have a very large dynamic range, a single image may show a very bright spot (the sun) together with dark patches (clouds). This presents the challenge of obtaining an image that retains details across a dynamic range that is larger than the typical 8 bits or 256 intensity levels.

The solution we have chosen is to take a series of exposures of different length of time and of low dynamic range (LDR). Each exposure results in an 8 bit image, which may contain underexposed areas or overexposed areas if the exposure duration for this image was too short or too long, respectively. But as the exposure durations vary over a wide range, each area of the image is faithfully represented in some subset of all images taken with different exposure durations. We then extract the faithful information from the exposures and create a high dynamic range (HDR) image following the radiance format.

Each pixel of a HDR image occupies 32 bits with the following semantics:

- 8 bits mantissa each for the red, green, and blue colour channels,
- 8 bits exponent representing the shared exponent for all three colour channels.

The cameras' exposure durations are software controlled and we use this feature to obtain the desired exposures.

In what follows we describe how we determined the series of exposures to take and the reconstruction algorithms to create the HDR images.

4.3.1.1 Camera response function

In the first instance, we need to determine the camera response function (CRF) for the Ximea MU9PM-MH cameras: the CRF

$$f : eX \mapsto b$$

maps, with a given exposure duration e , irradiance value X (power per unit area on the Earth's surface) to pixel brightness value b . Note that the unit of X is Wm^{-2} and therefore eX represents energy per area as Jm^{-2} .

Figure 4.1 shows the CRF in logarithmic scale for the Ximea MU9PM-MH cameras; it has been obtained from a series of exposures of different length of time using theDebevec [2] algorithm.

Validation of the CRF included a) validation using Robertson's [7] algorithm; b) validation within a same day of image capture; c) validation across two cameras (CAM000 and CAM001); d) validation across several days of image capture (20 to 24 October 2014); and e) validation using images of various nature (images taken in a lab, indoor and outdoor images).

The values of eX are only known up to a scale factor: in Figure 4.1, the exposure interval e is scaled so that an irradiance of 1.0 maps onto a brightness of 128 and e is the unit exposure duration.

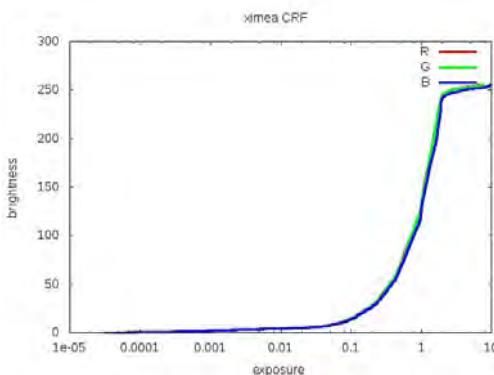


Figure 4.1: The camera response function $f : \log(eX) \mapsto b$ for Ximea MU9PC-MH.

4.3.1.2 Computing a bracketing set of exposures

4.3.1.2.1 Exposures used for computations

For all computations described in this section (objective function and bracketing system) exactly 20 exposures of length $2^i \mu s$, $i \in \{0, \dots, 19\}$, were used. These exposures are set and triggered via software; the largest possible exposure for the Ximea MU9PC-MH is 1s.

In the remainder of the text, we will use exposures to mean either the resulting images, or the duration of the exposures. The meaning will be made clear from the context.

4.3.1.2.2 The additive response function

Following Grossberg [4], we define the additive response function as the sum of the response functions for n exposures

$$h(X) = \sum_{j=1}^n f(e_j X)$$

where e_j is the exposure duration for the j^{th} exposure and X is the irradiance which for each pixel is assumed to be the same for all exposures taken in one set.



Figure 4.2 shows the additive response computed from the 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$, using the CRF shown in Figure 4.1. Note that the function is actually a step function (this is not visible in the graphs shown here due to the lack of resolution).

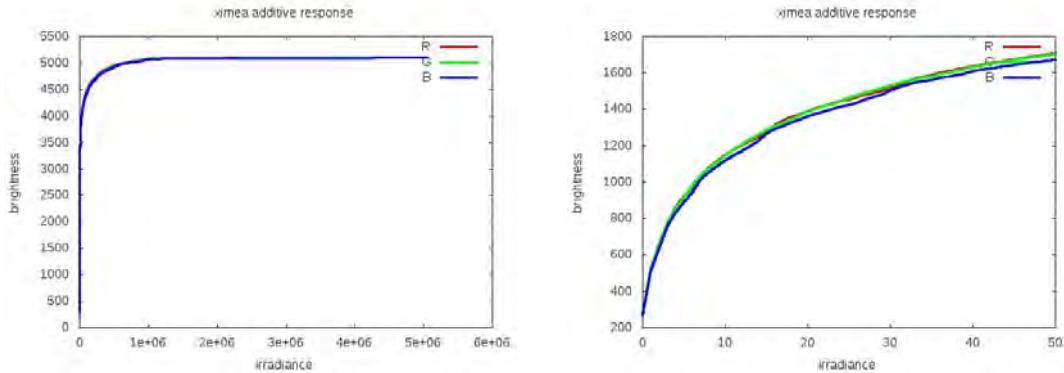


Figure 4.2: Additive response for the 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$; the graph on the right shows a restricted range.

4.3.1.2.3 Evaluating a configuration of exposures

Following Grossberg [4], we introduce the desired additive response function $g(X)$ and the response function $h(X)$ stemming from the camera. The objective is then to minimise the following function

$$\int_{X_{\min}}^{X_{\max}} |g'(X)' - h'(X)|^2 w(X) dX,$$

where $g(X)'$ and $h(X)'$ are the first derivatives of $g(X)$ and $h(X)$, respectively. The integration boundaries are given by the interval $[X_{\min}, X_{\max}]$ where $g(X)'$ is non-zero, and $w(X)$ is a weighting function defined as

$$w(X) = \begin{cases} 0 & \text{when } g'(X) < h'(X), \\ 1 & \text{otherwise.} \end{cases}$$

In this section, whenever we give values for the objective function of h , the desired set of exposures g against which h is compared is the set of 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$: that is, we use the objective function to compare configurations of exposures to the full set of 20 exposures. Figure 4.3 shows an example giving the value of the objective function for exposures $\{2^{15}, 2^{16}, 2^{17}, 2^{18}\} \mu\text{s}$.

4.3.1.2.4 Finding a minimal bracketing set

Following Barakat [1] we compute a minimal bracketing set (MBS): a minimal bracketing set is a set of exposures who together cover the entire detectable irradiance range of the system. Here the detectable range is the irradiance interval covered by the set of 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$. Figure 4.4 shows this range for a sky image and for an image taken from an office. Note that the office image (that includes part of the office's interior and part of the sky) displays a larger irradiance range than the full sky image.

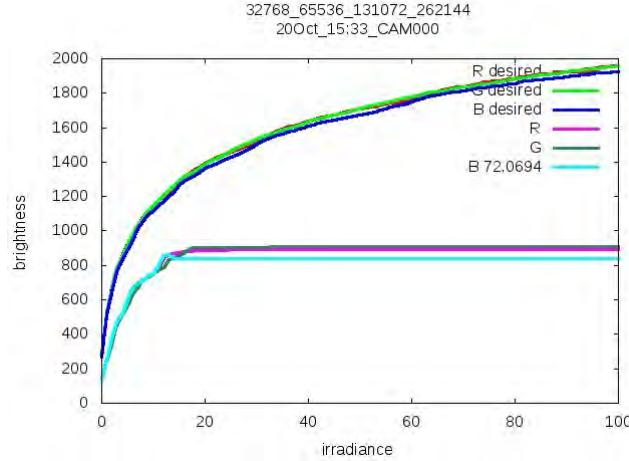


Figure 4.3: Two additive response functions: **desired** is the reference response (using the 20 exposures $2^i \mu\text{s}$, $i \in \{0, \dots, 19\}$); the other response shown uses the set of exposures $\{2^{15}, 2^{16}, 2^{17}, 2^{18}\} \mu\text{s}$, and the objective function has value 72.0694.

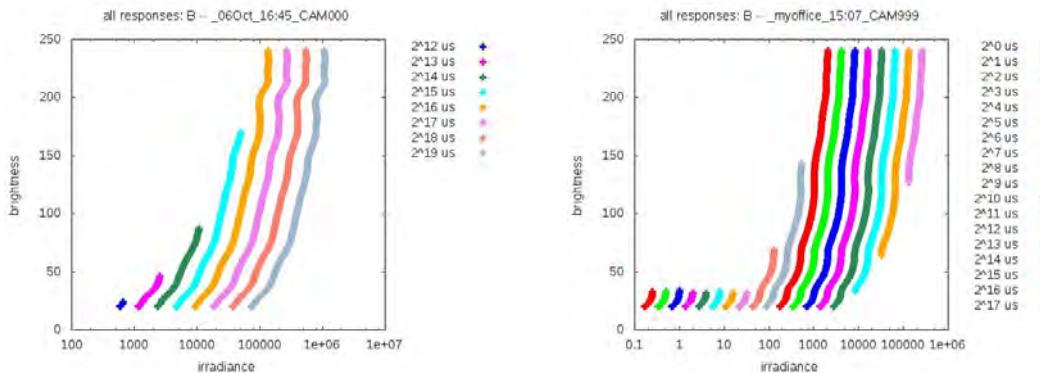


Figure 4.4: The irradiance detectable range of the set of 20 exposures. Right: images are full sky images; Left: images are from inside an office comprising part of the sky and hence display a larger irradiance range.

4.3.1.2.5 Putting it all together

We use the MBS as a guide to find configurations of similar size with possibly smaller objective function value. For each set of 20 exposures taken over five days (20 to 24 October 2014) with cameras CAM000 and CAM001, we compute a) the minimum bracketing set r , and b) the value of the objective function for configurations of similar size to that of r . For each camera the above results are collated over one day: we obtain the number of times a MBS r occurs together with the average value of its objective function for the day, see Figure 4.5 for an example. For the same day results for configurations of similar size to r are collated in the same manner, see Figure 4.6.

At the conclusion of this process, we find that, on average, the most frequently occurring MBS over the period 20-24 October and for both cameras, and with the smallest objective function value, is $\{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$. Let's denote this configuration by R . Of configurations of size 6 to 10, we find that, on average, the set $\{2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$ occurs most often and has smallest objective function value.

We choose the latter configuration as the configuration of choice for camera capture. It is larger than configuration R (hence its smaller objective function value), but being larger, it is better suited to accommodate

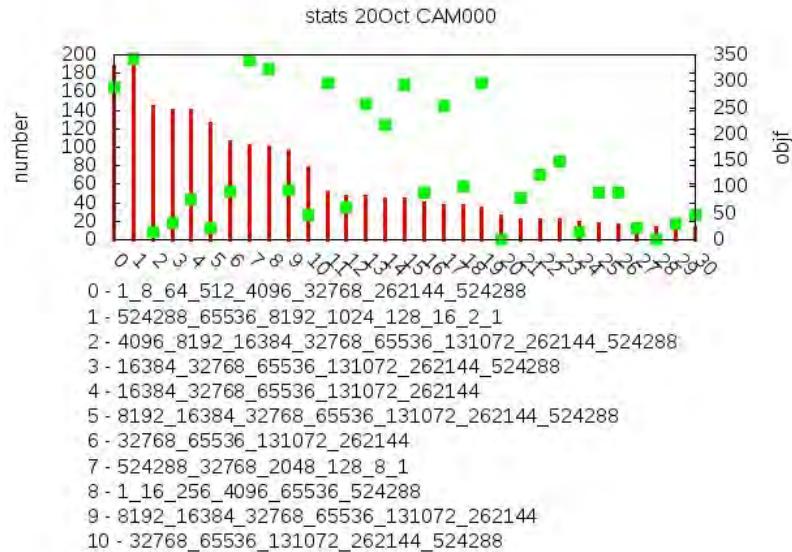


Figure 4.5: Collated results for camera CAM000, 20/10/2014: MBS. The red bar indicates the number of times an MBS occurs in the collated results. The green square dots indicate the value of the objective function. We note that among the most frequent occurring configurations for that day, the configuration $\{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$ had the smallest value for the objective function.

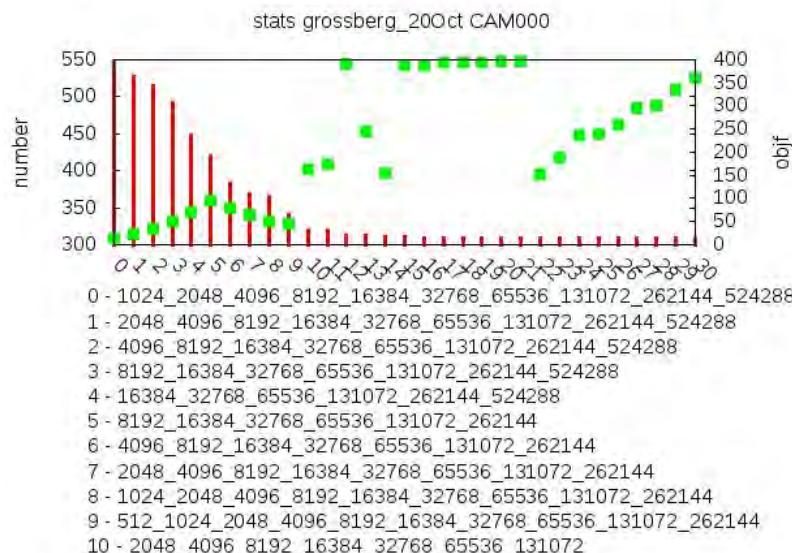


Figure 4.6: Collated results for camera CAM000, 20/10/2014: value of the objective function for configurations of similar size to $\{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$. The red bar indicates the number of times the configurations occur in the collated results. The green square dots indicate the value of the objective function. We note that the configuration $\{2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}, 2^{19}\}$ occurs most often and has the smallest value for the objective function.



possible future irradiance variations, for example, if cameras are fitted with weaker filters. Note that the collated results originate from two cameras who actually have different filter strength (9 and 6 stops for CAM000 and CAM001 respectively).

4.3.1.3 Constructing a HDR image

We have seen in the previous section that a good minimal bracketing set consists of the ten exposures $2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}, 2^{17}, 2^{18}$, and $2^{19} \mu\text{s}$. Camera capture was thus set up so that these ten exposures are taken at each time point. The camera is rigid, and since the total time required to take the ten exposures is less than one second, we may assume that the images resulting from the exposures are registered (that is, there is no significant difference between features from one exposure to the next), and that lighting is constant across the exposures.

4.3.1.3.1 Using the camera response function

We investigated three approaches to construct a HDR image from a series of different exposures of the same scene and the camera response function; the basic idea is to construct an irradiance map.

Let $g : b \mapsto eX$ be the inverse function f^{-1} of the camera response function that maps a brightness b to its inducing exposure X .

For exposure $i = 1, \dots, n$, let p_l^i be the pixel at location l in the image and let b_l^i its brightness value. The corresponding pixel in the composite image is denoted by p_l and its assigned irradiance value by I_l . The three approaches are detailed below.

1. bestp: Reconstruction by taking the pixel carrying the most information

I_l is assigned the value $\frac{g(b_l^m)}{e_m}$, where m denotes the exposure i for which b_l^i is the largest within a given brightness range. The brightness range $[\min_b, \max_b]$ we consider is determined by the values of $b \leq \min_b$ and $b \geq \max_b$ for which $\frac{d}{db} \log f(g(b)) \approx 0$. In our case $\min_b = 20$ and $\max_b = 240$.

For a given pixel p_l in the composite image, the value I_l is obtained as follows:

$$I_l = \begin{cases} 0 & \text{if } \max_i \{b_l^i\} < \min_b; \\ \frac{g(255)}{e_1} & \text{where } e_1 = \min_i \{e_i\}, \text{ if } \min_i \{b_l^i\} > \max_b; \\ \frac{g(b_l^m)}{e_m} & \text{where } b_l^m = \max_i \{b_l^i\}, \text{ otherwise.} \end{cases}$$

That is, a pixel whose brightness across all exposures is smaller (larger) than \min_b (\max_b) in all exposures is set/clamped to zero (largest irradiance), in other words noise (saturated). Otherwise, the brightness conveying most information is chosen. As Mann [5] and others point out, this method is likely to result in posterisation as there is no smooth transition between pixel values in the composite image as neighbouring pixels may be assigned values resulting from different exposures. A better approach is to use an averaged sum, as explained in the next section.

2. add: Reconstruction by a simple summation of exposures

The value assigned to I_l is the averaged sum $\frac{1}{n} \sum_{i=1}^n \frac{g(b_l^i)}{e_i}$. While the averaged sum mitigates posterisation artifacts it will retain noise and saturated pixels in the composite image, as all exposures are given equal weight in the sum.

3. addw: Reconstruction by a weighted summation of exposures

I_l is assigned the averaged weighted sum of irradiances $\frac{1}{\sum_i w(b_l^i)} \sum_i w(b_l^i) \frac{g(b_l^i)}{e_i}$.

This is the Mann algorithm [5] where the weighting function is given by the derivative of the log of the camera response function. Thus $w(b_l^i) = \frac{d}{db} \log f(g(b_l^i))$.

Weighting each pixel contribution by the value of the derivative naturally discards noise and saturation: the steeper the derivative, the more detail that can be recovered; the smaller the derivative, the less detail that can be



Australian Government

Australian Renewable Energy Agency

ARENA

recovered as there is little difference in brightnesses for a small change of irradiance. Derivatives are computed in logarithmic scale so that quantisation errors are perceptually uniform.

4.3.1.3.2 Using computer vision techniques

A different approach to construct HDR images is Mertens' algorithm [6] which constructs a low dynamic range image composite image from the exposures without any knowledge of their duration, nor of the camera response function. The particularity of the algorithm resides in that it merges the exposures solely based on image quality measures like saturation, contrast, and well-exposedness, and as a consequence the resulting image retains an optimal number of features present in the exposures, while removing noise and saturation.

4.3.1.3.3 Performance measurement

To gain an understanding of how well our construction algorithms perform we have computed several error functions. They all make use of the exposures recreated from the composite image. These functions test two scenarios:

1. how the exposures recreated from the composite image compare to the original exposures;
2. how the composite image obtained from the recreated exposures compares with the one obtained from the original data.

These two tests are not identical, and one HDR construction approach might perform poorly in one test, and better in the other, as we will see below. The first test tells us to what extent the original data can be recreated from the composite image; the second test tells us to what extent the composite image is ‘stable’ after recreating the exposures.

Image quality is a difficult notion to define here: on top of preserving all data present in all the exposures, we would like the composite image to be fit for computer vision processing. In particular, the image should be ‘smooth’, that is, there should be no abrupt change in value from one pixel to a neighbouring one; that is, we want to minimise posterisation artifacts due to coarse quantisation errors.

None of the tests described here address this point, more on this in Section 4.3.1.3.5.

Recovering the original exposures

When recovering the original exposures, we always assume that the number and duration of the original exposures contributing to the original image are known. That is, we do not attempt to recover an exposure of duration e_i when it was not used in the construction of the HDR image.

Exposures are recovered as follows: Given a pixel p_l in the composite image with value irradiance I_l , each exposure i with duration e_i is recreated by setting its pixels to $b_l^i = f(I_l e_i)$.

Measures

In comparing a recreated (exposure or composite) image \hat{i}_k with the original (exposure or composite) image i_k we compute two quantities from the absolute difference image $di_k = |i_k - \hat{i}_k|$. They are the mean μ_k and standard deviation σ_k of di_k , and the relative L2 norm $\frac{\|i_k - \hat{i}_k\|_{L_2}}{\|i_k\|_{L_2}}$. More precisely, these quantities are computed for each colour channel, and the mean, standard deviation, and relative L2 norm for di_k are then obtained as an average across the channels.

When testing for the recreated exposures, we'll give one single measure for the corresponding composite image: we take the average of the above quantities over the exposures. That is, if an image has been obtained from n exposures with exposure durations e_k , $k = 1, \dots, n$, we assign it two error measures:

1. the mean error $\mu = \frac{1}{n} \sum_k \mu_k$ and the standard deviation $\sigma = \frac{1}{n} \sum_k \sigma_k$,
2. the relative L2 norm $\frac{1}{n} \sum_k \frac{\|e_k - \hat{e}_k\|_{L_2}}{\|e_k\|_{L_2}}$.



4.3.1.3.4 Results

1. Comparing recreated exposures with original exposures

From the description of the three construction methods (Section 4.3.1.3.1) and the exposure recovery algorithm, one expects that on average, **bestp** shows the smallest error since the two other methods involve an addition of images. This is indeed verified by the graphs in Figure 4.7. We see that **add** shows poor results; but again, this readily follows from the fact that all brightness values (that is, including noise and saturation) are given equal weight.

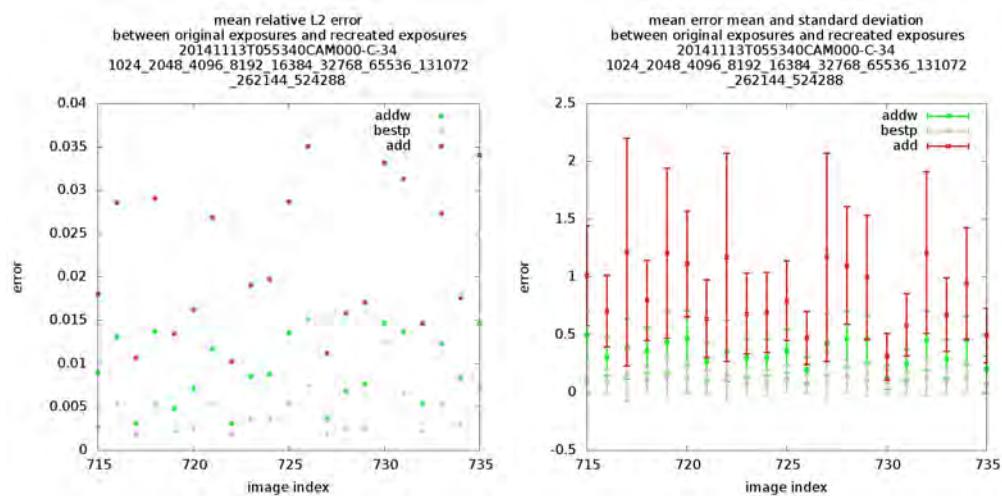


Figure 4.7: Comparing the three construction methods **bestp**, **add**, and **addw** for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$.

Since the relative error range shown by **add** is so large let us compare only **bestp** and **addw** to get a clearer view. In Figure 4.8 we see clearly that **bestp** outperforms **addw** for the error measures considered. But note that in both cases the errors remain tiny.

Intuitively, one does not expect Mertens' algorithm to perform well for this measure, as the formation of the composite image is cognisant nor of the camera response function nor of the duration of the exposures, data that underlies the exposure recovery algorithm described here. Indeed, Figure 4.9 confirms this.

1.1. Comments

Figures 4.7, 4.8, and 4.9 display results for approximately one hundred images from a single day and for one camera; the images were taken in the afternoon. Similar results were obtained over tens of thousands of images covering a range of six days (from sunny to cloudy), for three different cameras, two taking sky images, and the other indoor scenes.

Moreover, these measurements were repeated while considering the colour channels independently, and considering different values of n and e_k . The results were similar to the ones shown here and are thus not presented here.

In all the graphs presented thereafter, the results shown correspond to the same subset of images as the one used in Figures 4.7, 4.8, and 4.9, except where otherwise specified. As usual, the composite images are created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$.

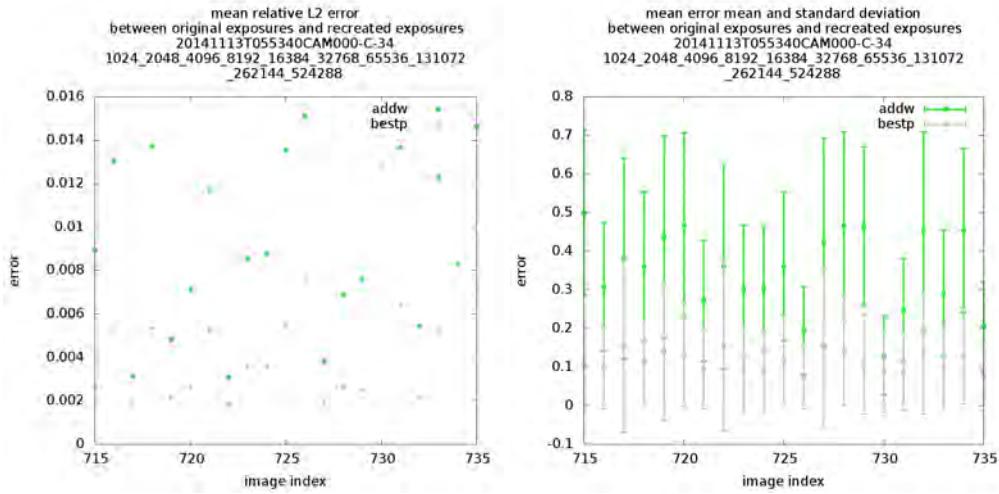


Figure 4.8: Comparing the two construction methods `bestp` and `addw` for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$.

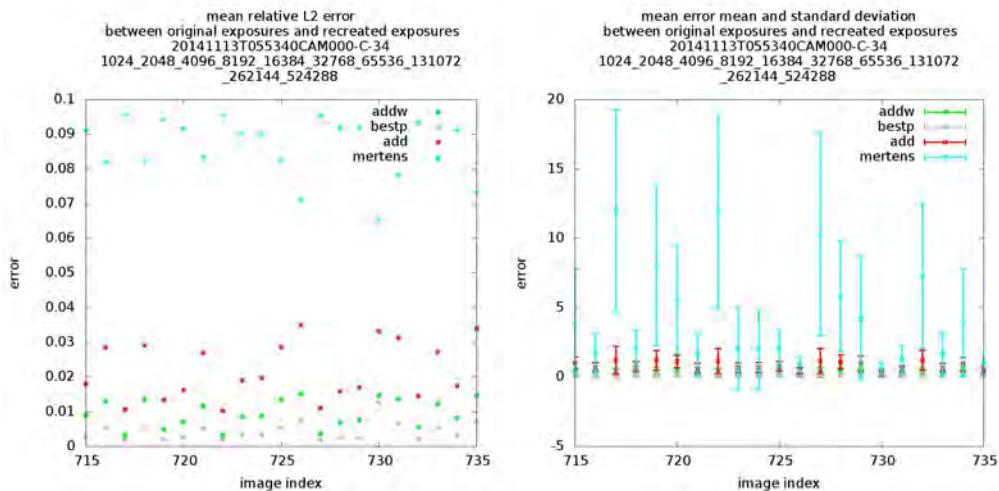


Figure 4.9: Comparing Mertens' algorithm with the three construction methods `bestp`, `add`, and `addw` for a subset of images taken on 13/11/2014. The x axis gives the image index i in the sequence of images taken during the day. The composite image was created from the exposures $2^{10}, \dots, 2^{19} \mu\text{s}$.



2. Comparing composite images from recreated exposures and from original exposures

Figure 4.10 compares the composite images obtained from the recreated exposures and from the original images for the reconstruction methods that compute an irradiance map. We find a situation similar to that of Figure 4.7: the add methods performs poorly, but again, note that the quantities are very small.

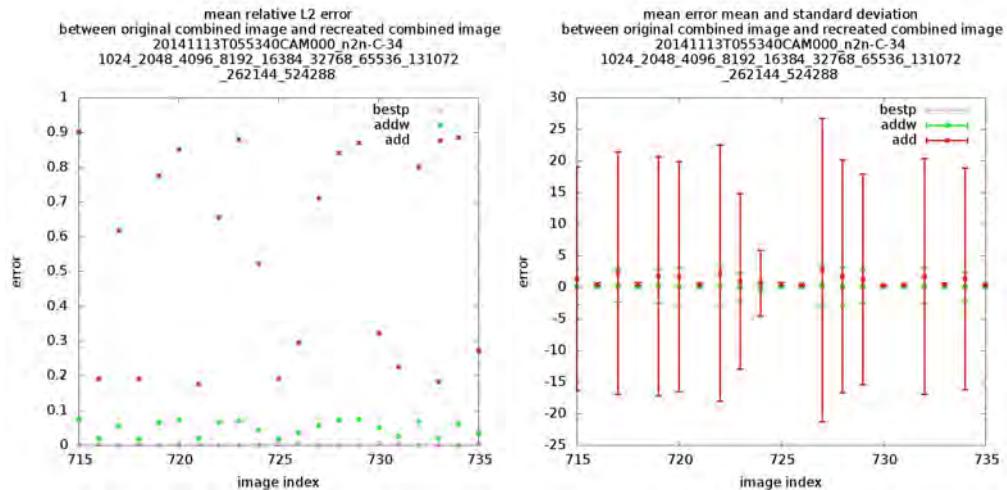


Figure 4.10: Comparing composite images from recreated exposures and original exposures for the three construction methods **bestp**, **add**, and **addw**.

Figure 4.11 shows the same results for methods **bestp** and **addw** only: again, **bestp** outperforms **addw**.

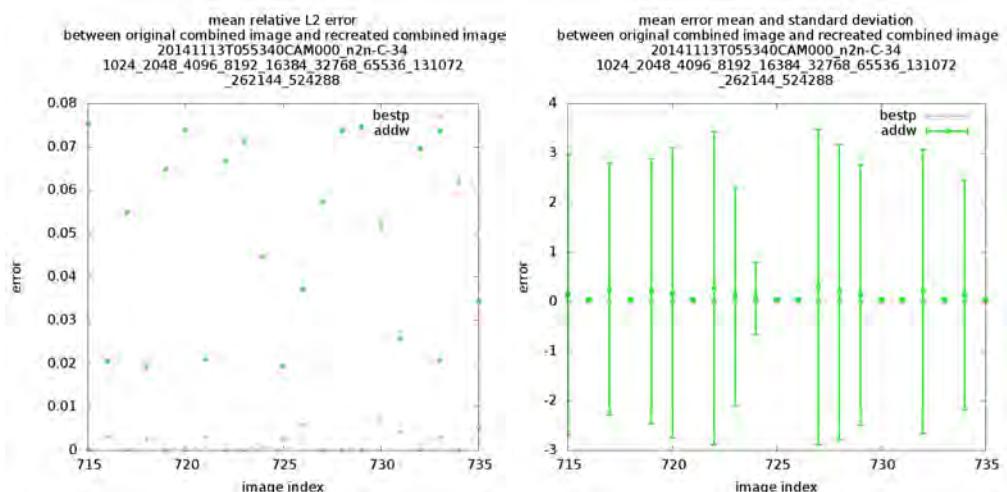


Figure 4.11: Comparing composite images from recreated exposures and original exposures for **bestp** and **addw**.

2.1. Comments

Similar results to those shown here have been obtained for tens of thousands of images covering a range of six days and for three different cameras. We noticed however a sensitivity of the results to the kind of exposures used to create the composite image. For example, Figure 4.12 show results where the composite image has been



created using the eleven exposures $2^0, 2^{10}, \dots, 2^{19}$ while for Figure 4.13 the composite image was created from the ten exposures $2^{10}, \dots, 2^{19}$ (images taken on 14/12/2014). We note that the errors are much larger in the first figure, presumably because the $1\ \mu s$ exposure mostly/only consists of noise (the $1\ \mu s$ exposures have the largest recovery error).

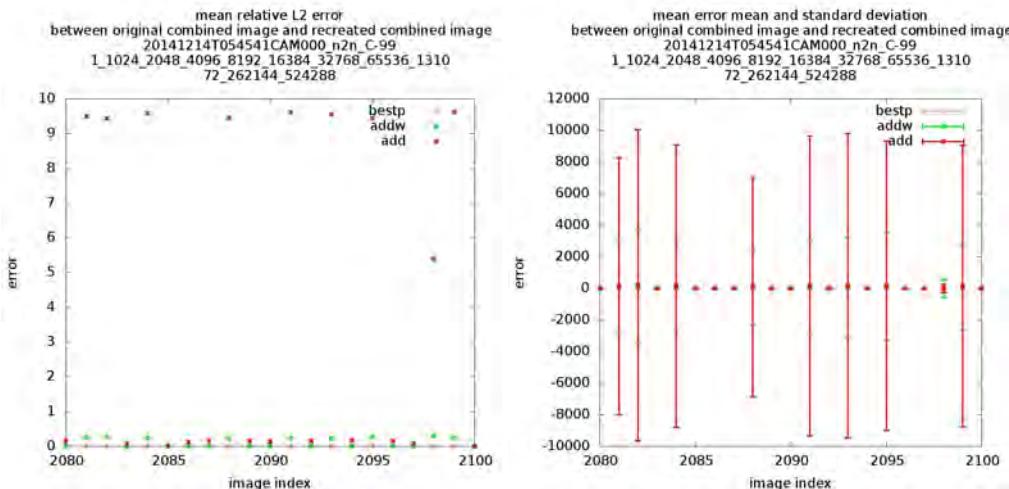


Figure 4.12: Comparing composite images from recreated exposures and original exposures for bestp, addw, and add; the composite images were created from the eleven exposures $2^0, 2^{10}, \dots, 2^{19}\ \mu s$ (images taken on 14/12/2014).

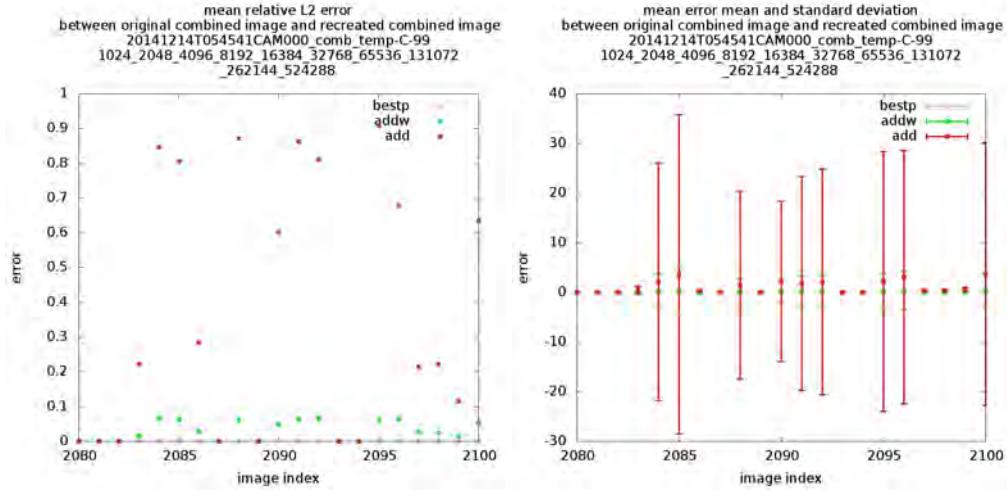


Figure 4.13: Comparing composite images from recreated exposures and original exposures for `bestp`, `addw`, and `add`; the composite images were created from the ten exposures $2^{10}, \dots, 2^{19} \mu\text{s}$ (images taken on 14/12/2014).

3. Evaluating Mertens' algorithm

Combining the results shown so far in this section with the ones obtained for the `mertens` approach, we obtain Figure 4.14. The relative L2 norm measure for `mertens` fares poorly, but this reflects the fact that exposures cannot be recovered properly in the first instance. Nevertheless the `mertens` image obtained from the recovered exposures is quite similar to the original `mertens` composite image (it shows a good result compared to `addw` for mean and standard deviation –but recall that `mertens` is a LDR image as opposed to the rest which are HDR images): this should not surprise us as image composition is driven by computer vision features. This tells us that these features are preserved when recovering the exposures and combining them again. For clarity, Figure 4.15 compares `bestp` and `mertens`.

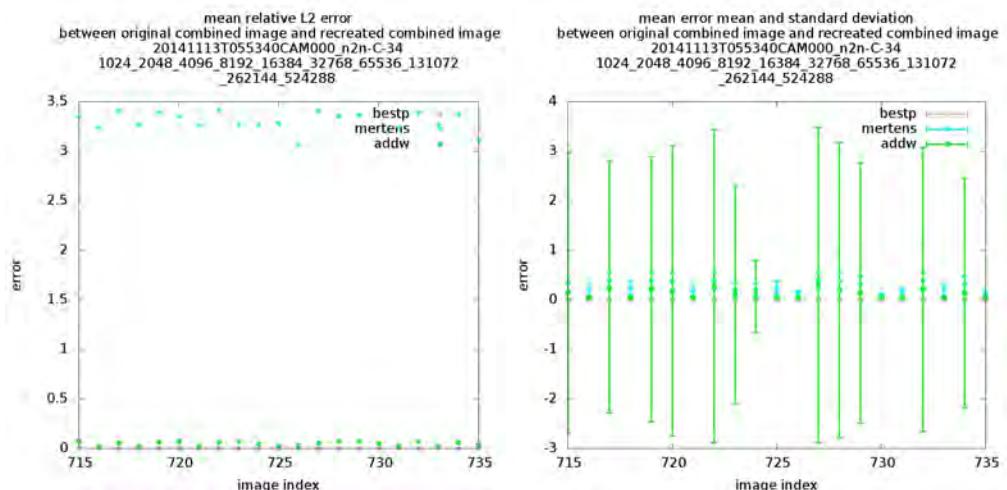


Figure 4.14: Comparing composite images from recreated exposures and original exposures for `bestp`, `addw`, and `mertens`.

With Mertens, another measure is worthy of interest: that of comparing the composite images obtained

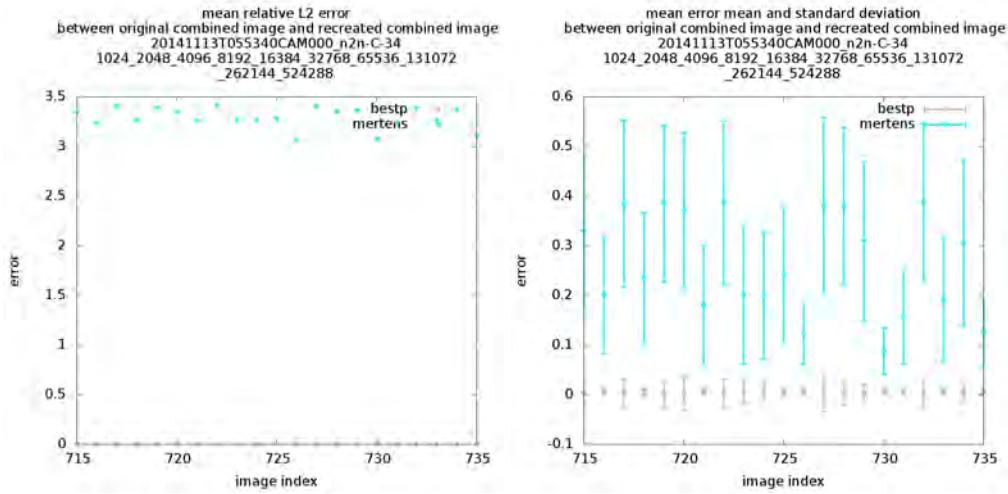


Figure 4.15: Comparing composite images from recreated exposures and original exposures for `bestp` and `mertens`.

using Mertens' algorithm applied to the original exposures and applied to the exposures recovered from the `addw` composite image –indeed, recall that recovered exposures from `addw` are much more faithful to the originals than when recovered from `mertens`. This is shown in Figures 4.16 and 4.17: `mertens_from_addw` denotes the Mertens image obtained from the recovered exposures from `addw`. Note that the errors for `mertens_from_addw` are very small indeed.

However, as already noted, when comparing mean and standard deviation measures for `mertens` and `mertens_from_addw` to the same measures for `bestp` and `addw` one must be mindful that the `mertens` and `mertens_from_addw` have a low dynamic range (8 bits or 256) compared to the `bestp` and `addw` images.

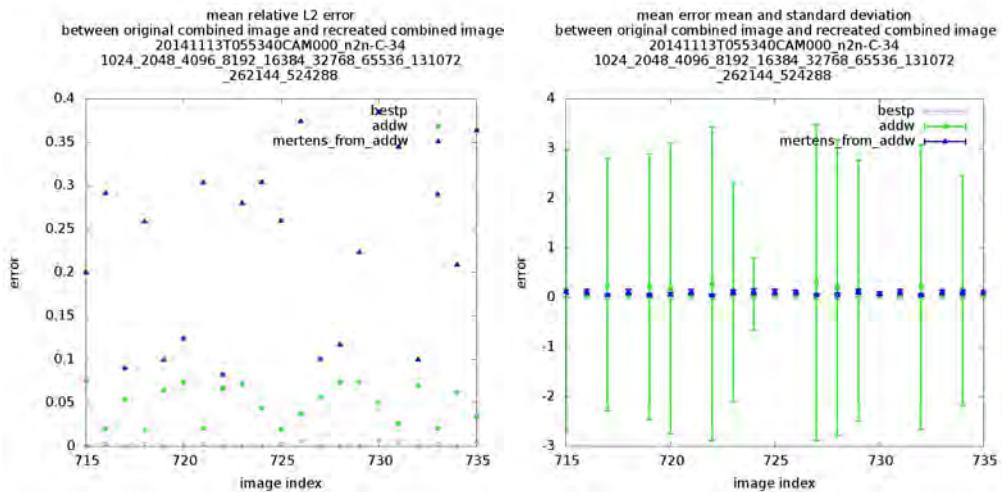


Figure 4.16: Comparing composite images from recreated exposures and original exposures for `bestp`, `addw`, and `mertens_from_addw`.

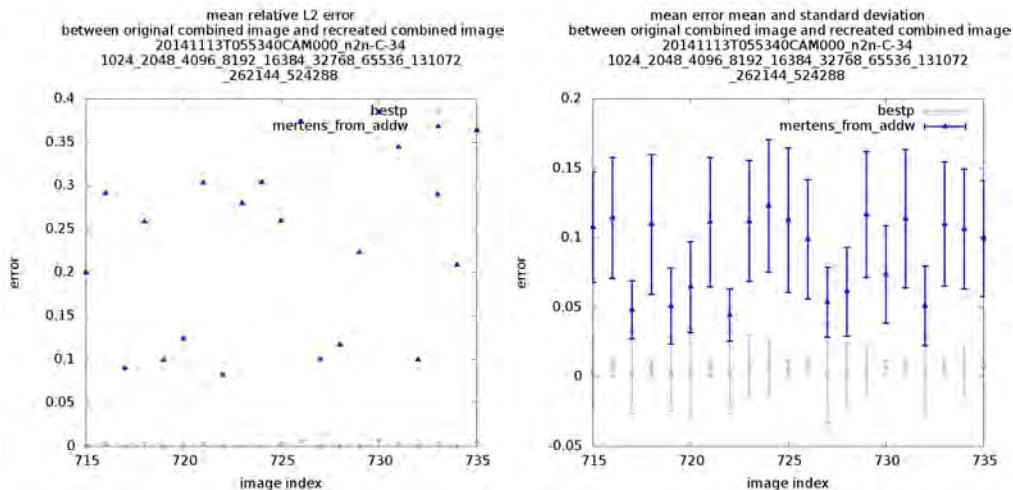


Figure 4.17: Comparing composite images from recreated exposures and original exposures for `bestp` and `mertens_from_addw`.

4.3.1.3.5 Discussion: Which algorithm to choose?

The results show in Section 4.3.1.3.4 are only indicative of the relative performance of the different approaches tested. As already mentioned, the tests cannot tell us anything about the ‘quality’ of the composite image. Rather they measure how faithfully the original LDR exposures may be recovered from the constructed HDR image. With respect to this measure, we see that `bestp` is an optimal choice, with `addw` a close second. In both cases, the errors are very small. Even though `bestp` is optimal, we posit that `addw` should be our preferred method, the main reason being that the `bestp` approach induces posterisation and is prone to retain noise in the composite image. Some examples are given below, Section 4.3.1.3.6.

Finally, we advocate the use of Mertens’ algorithm, in addition of the `addw` approach. Interestingly, we see from Comment 3. Page 46 that an approach would be to create the Mertens’ image from the exposures recovered from the `addw` image as a post-processing step. But from the note in that section with respect to the `mertens` image dynamic range we might not always obtain an optimal result.

4.3.1.3.6 Examples

1. Example composite images for `bestp` and `addw`

All figures here show, for the original set of exposures, a `bestp` composite image on the left-hand side and a `addw` composite image on the right-hand side. These images were taken on various days in November and December 2014. All images were built from the ten exposures $2^{10}, \dots, 2^{19} \mu\text{s}$, except the ones shown in Figure 4.22 which were built from the eleven $2^1, 2^{10}, \dots, 2^{19} \mu\text{s}$ exposures. A gamma correction of 2.2 was applied when viewing the HDR images shown below, but no tone mapping. Instead an appropriate dynamic range was chosen for the purpose of rendering them in this document.

Figures 4.18, 4.19, and 4.20 are examples where the `bestp` image retains noise. This is a consequence of clamping pixel values to zero (see Section 4.3.1.3.1).

Another artifact is posterisation; Figures 4.21, 4.22, and 4.23 are some examples. Although on average the `bestp` image is more prone to posterisation, the `addw` may also exhibit this feature (Figure 4.23). When images are built from ten exposures, posterisation is hard to detect visually; it is more prominent when images are built from eleven exposures as seen in Figure 4.22. This is explained by the fact that the dynamic range of the eleven exposures image is larger than for the ten exposures image, resulting in a better visualisation of the artifacts.



Finally there is another artifact that can be seen in the `bestp` images, and which might be present in `addw` image, but to a lesser degree. Looking closely at Figure 4.19 we see some yellowish horizontal lines streaking across the `bestp` image. They are the result of clamping only some colour channels and is dependent on the brightness range chosen $[\min_b, \max_b]$. The larger the range, the stronger the artifacts as more pixels will see some (but not all) of their channels clamped (or to express this differently: the smaller the range, the more pixels will have all the three channels clamped, the weaker these artifacts –but more noise). These lines are apparent (more or less faintly depending on which exposures one is looking at) in the original exposures –these were captured at dawn and as a consequence there is a lot of noise present.

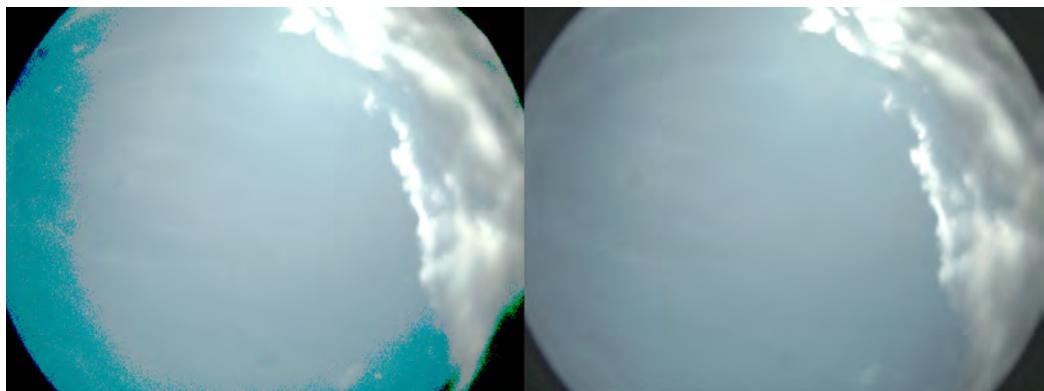


Figure 4.18: An example of noise in the `bestp` image. Left: a `bestp` irradiance map; Right: an `addw` irradiance map for the same image.



Figure 4.19: An example of noise in the `bestp` image. Left: a `bestp` irradiance map; Right: an `addw` irradiance map for the same image.

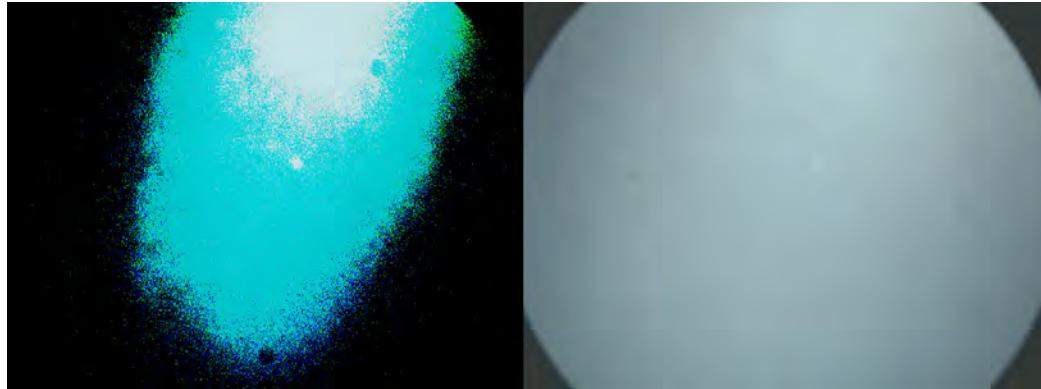


Figure 4.20: An example of noise in the **bestp** image. Left: a **bestp** irradiance map; Right: an **addw** irradiance map for the same image.



Figure 4.21: An example of posterisation in the **bestp** image. Left: a **bestp** irradiance map; Right: an **addw** irradiance map for the same image.

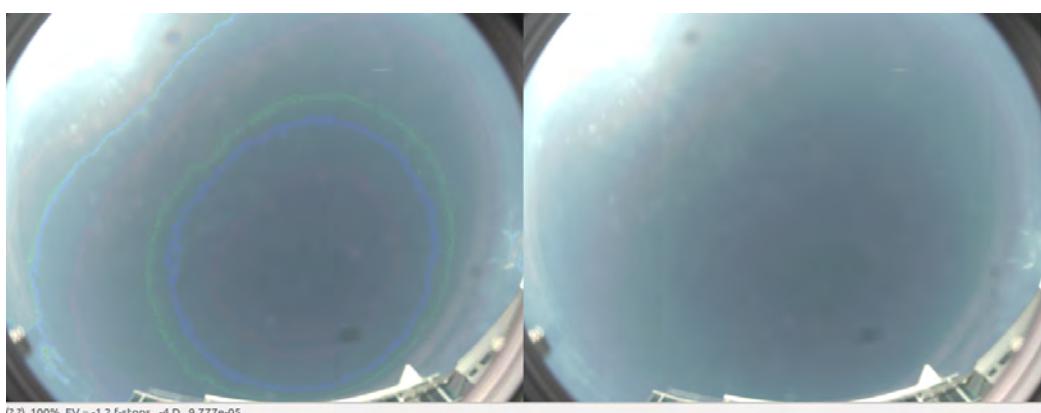


Figure 4.22: An example of posterisation in the **bestp** image. Left: a **bestp** irradiance map; Right: an **addw** irradiance map for the same image. The images are obtained from the $2^1, 2^{10}, \dots, 2^{19} \mu\text{s}$ exposures.

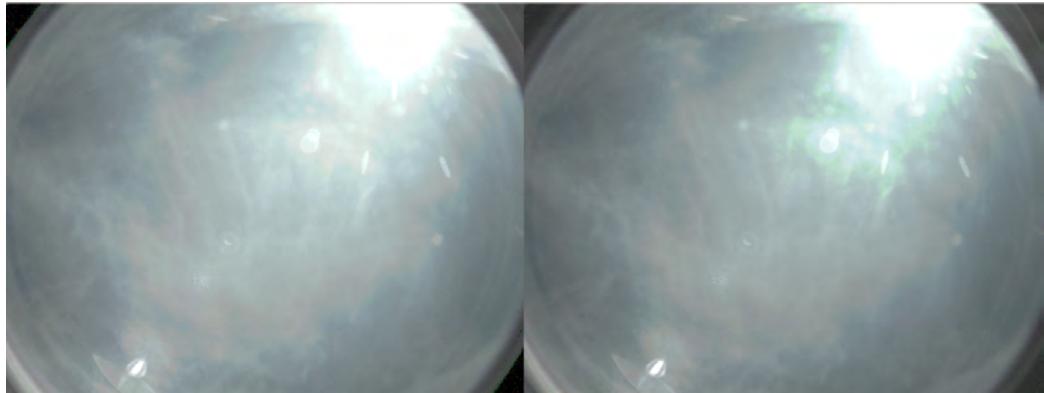


Figure 4.23: An example of posterisation in the `addw` image. Left: a `bestp` irradiance map; Right: an `addw` irradiance map for the same image.

2. Example composite images for mertens

For each image shown above, we show the corresponding `mertens` composite image.



Figure 4.24: Left: `mertens` image corresponding to the image in Figure 4.18. Right: `mertens` image corresponding to the image in Figure 4.19.

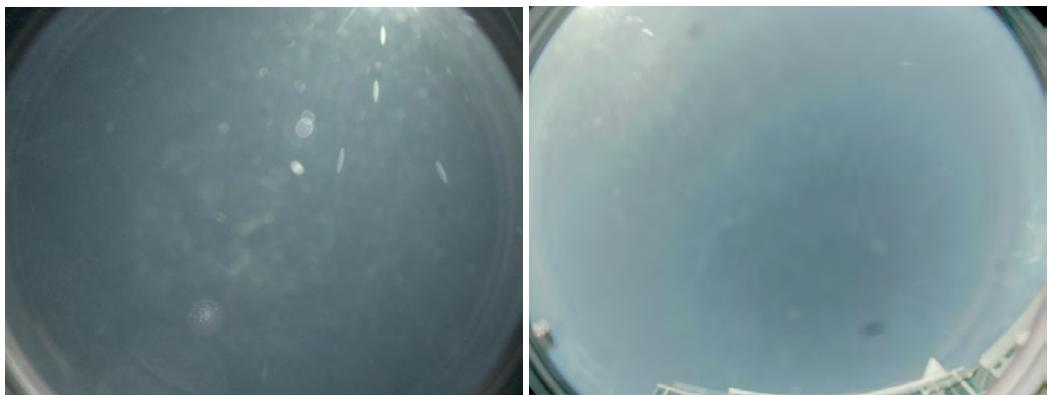


Figure 4.25: Left: mertens image corresponding to the image in Figure 4.20. Right: mertens image corresponding to the image in Figures 4.21 and 4.22.

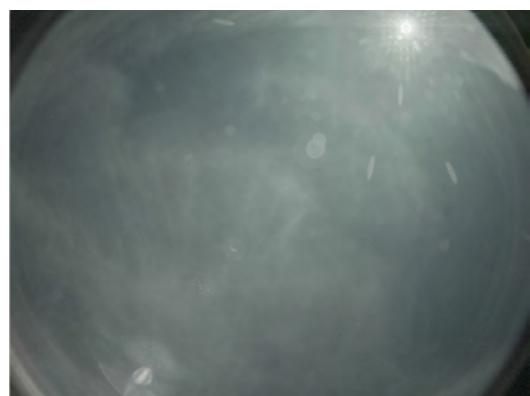


Figure 4.26: mertens image corresponding to the image in Figure 4.23.



3. Comparing images for mertens and mertens_from_addw

The next three figures compare composite images for `mertens` and `mertens_from_addw`. Figure 4.27 is an example where `mertens_from_addw` compares poorly, while in Figures 4.28 and 4.29 the two images are comparable.



Figure 4.27: Left: composite image for `mertens`. Right: composite image for `mertens_from_addw`. Note the posterisation in the `mertens_from_addw` which is not present in the `mertens` image.

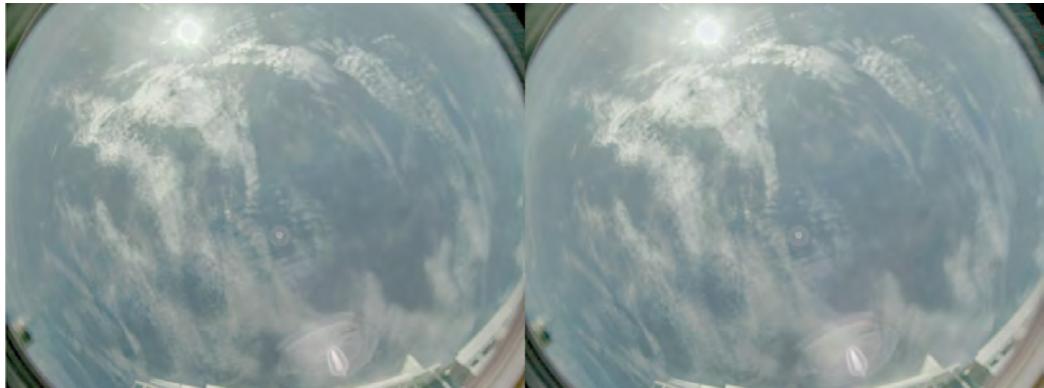


Figure 4.28: Left: composite image for `mertens`. Right: composite image for `mertens_from_addw`. The images are comparable, although `mertens_from_addw` shows a slight posterisation (difficult to detect here).

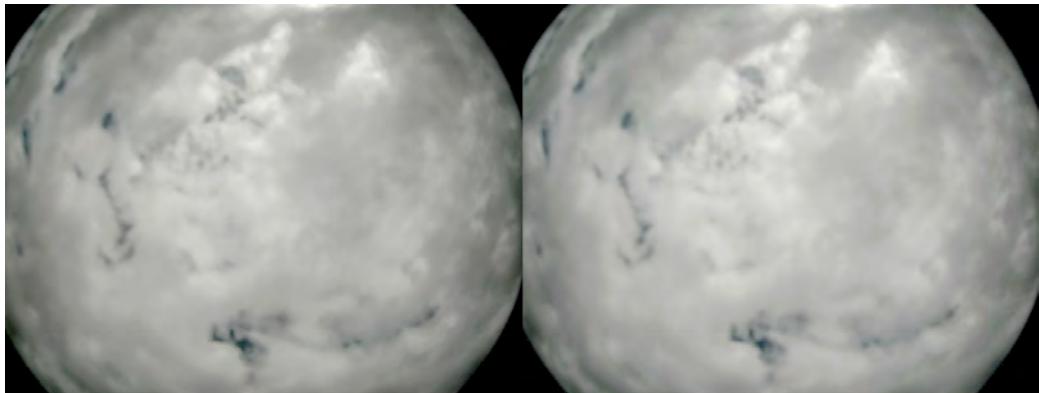


Figure 4.29: Left: composite image for `mertens`. Right: composite image for `mertens_from_addw`. The images are comparable, although `mertens_from_addw` shows a slight posterisation (difficult to detect here).

4.3.1.3.7 Conclusion

In conclusion we see that the `bestp` and `addw` methods perform well in the recovery (exposures and composite image) tests. But `bestp`'s intrinsic weakness (clamping and lack of smooth pixel transition) does not render it suitable as a faithful representation of the irradiance map. Thus our choice will be `addw` (the Mann [5] algorithm), the simplest of the HDR algorithm constructing an HDR irradiance map from a linear combination of LDR exposures.

Because Mertens' algorithm [6] relies on image quality metrics which are not addressed in our tests, and because it has the advantage of constructing a low dynamic range image, we have implemented this algorithm as well in our operational system. Of interest would be to compare the reconstructed images according to the quality measures described in [6]. But even there comparison of the resulting composite image with existing tone mapping techniques is done visually, as we did in Section 4.3.1.3.6.

4.3.2 Data Portal

The sky images are publicly available from <http://solar.research.nicta.com.au> (see Figure 4.30).

Users may choose the camera(s) and the reconstruction algorithm(s) they are interested in. They will also specify the time period of interest as well as the image frequency. Since the data in question is quite large (approximately 1M per image), a summary of the requested download is first presented to the user after which they may decide to proceed or modify their request.

Images captured by the cameras have a resolution of 2592x1944 pixels, but only 4-downsampled images (648x486 pixels) are available through the portal –this being due to the unmanageable size of the larger images. The higher resolution images can be requested from the maintainer of the site. Images are in the radiance format (with extension .hdr).

Images are available for the period 24/09/2014 (21/05/2015 for CAM002) to 30/06/2016, that is, image recording and processing is planned to be maintained until 30/06/2016. Image frequency is 10s (60s for the full resolution 2592x1944 images).

4.4 Extent to which the objectives were achieved

Sky images were recorded at 10 seconds intervals at three locations that complement the PV and solar radiation sensor networks. A coherent set of image data is available from September 2014 for cameras CAM000 and CAM001, and from May 2015 for camera CAM002. Data collection is ongoing, and the collected data are available from <http://solar.research.nicta.com.au>. This work fulfils the objectives for this part of the project.

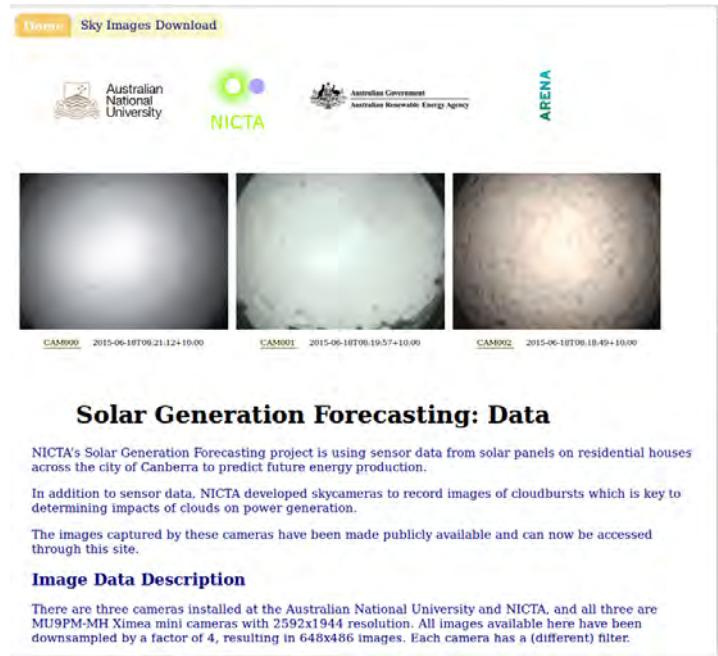


Figure 4.30: A snapshot of the data portal.

4.4.1 Hardware and data

Three cameras have been deployed on or close to the Australian National University campus. They capture low dynamic range sky images which are then combined into high dynamic range images and transferred to the central data server. Full resolution (2592×1944 pixels) and down-sampled low resolution (648×486 pixels) images are captured every 60s and 10s, respectively. The data is publicly available from <http://solar.research.nicta.com.au>.

We intended to deploy a few single-board RaspberryPie computers with cameras attached, but shortage of time didn't allow us to proceed. The process for obtaining images with the single-board cameras should be the same as for the existing cameras. The developed software can be easily ported, especially if the underlying operating system is also Linux (Ubuntu). Some time would be required to recover the camera response function, identify appropriate exposures and evaluate the quality of the data.

4.4.2 HDR image reconstruction

There are a few algorithms for reconstructing the HDR image using a weighted sum of the exposures ([2, 3, 5, 7]). In addition to Mann [5], we tried Debevec and Robertson [2, 7] but the results were not convincing; lack of time prevented further investigation in this space.

We believe that [3] deserves a closer look, as they present a weighting function that claims to account for all noise sources. But we have seen that the simple Mann [5] algorithm performs well and it has the advantage of being extremely simple and fast.

Finally, we note that the Ximea MU9PM-MH camera can capture raw images of up to 12bits. Of interest would be to analyse the quality of such exposures and determine whether the HDR image composition could thus be simplified.

4.4.3 HDR image quality

As mentioned in Section 4.3.1.3.3 we did not test the HDR reconstruction algorithms against an image quality measure like smoothness, contrast, etc... The first difficulty is to be able to define such a measure. As in [4] we could have compared composite images by comparing the density of their iso-brightnesses curves. But this



is a fraught exercise as the result is highly dependent on the resolution of the curves and could only apply in a controlled environment: for example, a heavily posterised image could show a larger curve density than a corresponding ‘smooth’ image.

We suggest establishing a quality measure in the manner of [6] in order to get a more meaningful comparison between composed images.

4.5 Conclusion

In this chapter we described how to combine a set of low dynamic range images typically recorded from a low cost camera system into a high dynamic range (HDR) irradiance maps. Each irradiance map represents two important types of information: a) very bright details from the sky representing the major part of the irradiance falling onto the camera (and any PV system nearby), and b) quite dark details of the cloud flow representing information about how the sky will change in the future.

The main steps of the chosen approach are

- the recovery of the camera response function for each colour channel,
- the determination of an optimal set of low dynamic range exposures from which a high dynamic range image will be created, and
- the determination of two simple and fast algorithms which combine exposures of different duration into the wanted HDR irradiance map.

Three sky cameras are now continuously producing HDR irradiance maps using the described approach. Further work could use correlations in the overlapping area between different cameras in order to enhance the quality of the HDR irradiance maps.

Bibliography

- [1] Neil Barakat, A Nicholas Hone, and Thomas E Darcie. Minimal-bracketing sets for high-dynamic-range image capture. *Image Processing, IEEE Transactions on*, 17(10):1864–1875, 2008.
- [2] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 Classes*, SIGGRAPH ’08, pages 31:1–31:10, New York, NY, USA, 2008. ACM.
- [3] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H.-P. Seidel, and H. Lensch. Optimal hdr reconstruction with linear digital cameras. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 215–222, June 2010.
- [4] Michael D Grossberg and Shree K Nayar. High dynamic range from multiple images: Which exposures to combine. In *Proc. ICCV Workshop on Color and Photometric Methods in Computer Vision (CPMCV)*, 2003.
- [5] S. Mann and R.W. Picard. On being ‘undigital’ with digital cameras: Extending dynamic range by combining differenwntly exposed pictures. Technical Report 323, M.I.T. Media Lab Perceptual Computing Section, Boston, Mass., 1994. Also appears, IS&T’s 48th annual conference Cambridge, Mass., May 1995.
- [6] Tom Mertens, Jan Kautz, and Frank Van Reeth. Exposure fusion. *Pacific Graphics*, 2007.
- [7] Mark A. Robertson, Sean Borman, and Robert L. Stevenson. Estimation-theoretic approach to dynamic range enhancement using multiple exposures. *Journal of Electronic Imaging*, 12(2):219–228, 2003.



Chapter 5

Machine learning forecasting

5.1 Overview

This section presents the final report on the use of machine learning techniques for accurate predictions of solar power. It is based on large-scale PV datasets in Canberra retrieved from PVOutput.org over years 2012 to 2014. A variety of methods have been explored with varied parameter settings, and our new method clearly outperforms the state of the art. A simulator was also developed which allowed us to perform controlled experiments and to derive insights from simulation. We will describe the initial objectives, the extent to which those objectives were achieved, the key challenges and findings, and recommendations for how to build on the outcomes.

The code for the experiments described in this chapter is available upon request by sending an e-mail to christfried.webers@nicta.com.au.

5.2 Objectives

The objective of this module as defined in the Project Plan was to

Create novel machine-learning forecasting techniques and compare them against existing techniques drawn from the published literature, and test their performance for distributed PV prediction on our collected data sets.

The primary objective was to develop a methodology that compares favourably against existing methods in forecasting PV output. Although accuracy is one of the key aspects for comparison, computational and data complexity is also important in practice. Therefore, we addressed the overall objective by pursuing three sub-goals.

(a) Identify important features for building predictors

A lot of features are conceived valuable for making predictions. For example, the output of a panel a few minutes/hours ago, exogenous features such as temperature and wind speed, side information such as the theoretical PV output if the sky is clear, *etc.* A variety of off-the-shelf preprocessing techniques are also available such as normalisation and taking logarithm. However many features can be redundant in the presence of other features, and their value may vary under different prediction horizons and error measurements. So combinations of feature engineering options and forecasting scenarios lead to exponentially many experiment settings which pose significant computational challenges for large-scale data. Our first goal, therefore, is to confirm the efficacy of a variety of feature engineering options by experimenting on a *single* PV system, over a range of standard forecasting algorithms and scenarios.

(b) Improving forecasting accuracy by using distributed PV network

Equipped with the insights on feature engineering drawn from a single site, we next develop novel approaches that further improve the forecasting accuracy by aggregating the output of a *distributed PV network*.



The growth in the source of data, however, raises a new challenge of missing observations, because it also becomes more likely that some sites break down and cease providing data. Furthermore, the computational cost also grows at least linearly in the number of sites, necessitating more refined data reduction. We will elaborate on some new techniques developed to resolve these issues.

(c) Simulators for more insights

Any insights build on data from a real system will be constrained by the realities of that system, and corruptions present in real data. Accordingly, we have built a simple simulation of generated PV power that moves simulated clouds over an area and simulates the generated power. This simulation has two major advantages. First, since the data is entirely free of "corruptions" in real data, we can disambiguate problems due to the core predictor, and problems due to inaccurate location/angle information on real systems, incorrect dates, etc. A successful predictor on simulated data can then be translated to real data with the knowledge that any degradation in performance is due to these factors. The second advantage of simulation is that one can test the impact of changes in the system design. We want to understand how effective a distributed PV network could be with a given number of sites, spread in a given fashion, and with or without additional measurements that might be derived from visual data. Clearly, simulation gives a much simpler approach to testing these scenarios than physically implementing a huge variety of different systems.

5.3 Evidence of completion

The completion of the overall objective is demonstrated in the following three subsections that correspond to the above three sub-goals.

5.3.1 Identify important features for building predictors: baseline benchmarking

In order to identify valuable features and to gauge the current state of the art in solar forecasting, we have designed and executed a series of baseline experiments that are representative of the current capabilities of the field. Using both PV power output, solar radiation data, and weather observations from local meteorological observation stations, we produced several datasets for benchmarking analysis. These consist of several years of data each.

Hourly, 30 minute and 10 minute averaging of the data were completed in order to produce hourly, sub-hourly and near minute level forecasts. In each experiment, forecasts were produced for the $t + 1$, $t + 2$ and $t + 3$ forecast horizons, using the previous five measurements, along with the previous day's output for that same time. In addition, we experimented with including exogenous weather data.

Four different forecasting methods were tested. Three of the methods are machine-learning (ML) based methods, including two linear methods, with squared and absolute error¹, respectively, and a multi-layer perceptron neural network. The fourth was a naïve persistence predictor, for comparison to the machine learning methods. In addition to the solar radiation and PV power output measurements, a normalisation of these two fields by their clear-sky component was also undertaken, in order to remove yearly, seasonal and diurnal cycles from the data. These normalised fields were then subjected to the same experimental routine as the unnormalised data. To evaluate their performance, mean bias, mean absolute error, root mean square error, and the coefficient of determination were computed. Finally, we tested how much more difficult the problem becomes when one needs to predict multiple intervals of time ahead.

Results are summarised for radiation data in Table 5.9 and for PV data in Table 5.10. All predictors used their respective optimal parameters. They generally indicate that it is difficult to improve on a baseline measure of simply predicting a value equal to the previous value observed. However, linear methods trained with an l_1 or l_2 loss generally do so when carefully regularised to prevent overfitting. It should be noted that the weather pattern exhibited in Canberra results in a large number of cloudless days per year and an additional number of days with almost constant cloud cover. Both cases result in a very regular solar energy output from PV systems which only depends on the angle of the sun. For all of those days, short-term predictions based on the previous measurement does already quite well and extra effort has to be spent in beating this benchmark.

We now describe all the aspects of the experiment in more detail.

¹We will refer to them as l_2 and l_1 loss respectively.



5.3.1.1 Literature Review

To develop a baseline for assessing performance, we selected a few typical papers from the solar prediction literature.

Marquez et al. [11] use neural networks to predict radiation measurements (global horizontal irradiance) at time horizons of 30 to 120 minutes. Along with the previous radiation measurements, data from satellite images is included.

Huang et al. [8] use an autoregressive model to predict the difference of a measured radiation model from a model based on Fourier series (essentially modelling seasonality and time of day). Hybrid models incorporating neural networks are also considered.

Reikard [13] compares a variety of methods for predicting radiation, including linear models, autoregressive models, neural networks, and hybrids, all with and without transfer functions intended to transforms the measurements to make prediction simpler.

Dazhi et al. [17] predict hourly radiation data using an autoregressive model. They consider adding a variety of meteorological measurement to aid prediction, including cloud cover, pressure, wind speed, and relative humidity.

Izgi et al. [9] use neural networks to directly predict the photovoltaic output of a solar panel, with varying time horizons. They find relatively accurate prediction for up to 5 minutes, rising significantly up to an hour.

Voyant et al. [15] use neural networks to predict radiation data (horizontal global irradiance) using a variety of exogenous inputs, including pressure, temperature, wind information, humidity, and precipitation data. They find the addition of these exogenous data somewhat improve predictive performance.

5.3.1.2 Datasets

There were two major datasets used in this benchmarking study. First, global horizontal irradiance measurements were obtained from the Australian Bureau of Meteorology's (BoM) 1-minute radiation dataset (<http://reg.bom.gov.au/climate/reg/oneminsolar/index.shtml>), which have recently been made available via support from the Australian Solar Institute (now part of the Australian Renewable ENergy Agency - ARENA). The Wagga Wagga site (station # 71250) was selected, due to its proximity to the Canberra region (approximately 160 km to west).

The photovoltaic (PV) power output data was gathered from PVOutput.org, a website to which PV system owners in Canberra publicly report their power output data. A 30kW system in central Canberra with high quality power output data was chosen (e.g. absence of shading, regular reporting, well documented layout of PV array). This data was available at five-minute intervals and input as the fractional performance ratio (kW_r) - which is the measured power output divided by the rated power output for the array.

Along with the solar data, exogenous meteorological data was included. These were available from weather observations taken at the Wagga site and the Canberra Airport (another BoM measurement site, #70351). The fields chosen have been suggested to be useful in previous work [6, 10, 15] and are as follows:

- Dew Point, Station Level Pressure, Temperature, Wind Direction, Wind Speed

All these features were divided by the max (or median) of the value of the feature in the whole dataset (training + test).

In order to prepare these data for the benchmarking experiments, all of the fields were averaged to 10, 30, 60 minutes. So in the sequence, the time steps $t + 1$ and t are apart by 10, 30, or 60 minutes respectively.

These datasets were then further divided into training and test sets, where the last two years were used for testing, and the rest of the data for training. The PV data was trained on the period from April 2010 through the end of 2011 and then tested on 2012 through May 2013. The radiation data was trained on 2007-2009 data, and then tested over 2010-2011.

We standardised all features to zero mean and unit variance, and found that doing so regularly improved the performance. We also tried taking the log of all intrinsic feature (i.e. features that are not exogenous), but in general it does not improve – and can even harm – the performance. Predictions are always converted back to the raw measurements to measure performance.



5.3.1.3 Clear-Sky Normalisation

By nature, the time series data from solar radiation and PV power output are non-stationary, as they have both daily, seasonal and even yearly cyclical trends. This non-stationarity, however, is based on well-understood physical processes driven primarily by the position of the sun in the sky, as given by the zenith θ_z and azimuth angles ϕ_a . Therefore, previous solar forecasting projects have had success in removing these cycles by normalising (dividing) the given radiation time series by its theoretical clear sky component as given by a model [6, 15]. The understanding is, that with the cyclical behaviour removed, the learning algorithms do not have to spend time adjusting to compensate to them, and the overall problem of predicting the time series is reduced to determining the quantity of the deviation from clear sky.

To create the normalised form of the global solar radiation, denoted by K_c , the measured values were divided by the clear sky estimates as generated by the REST2 clear sky radiation model [7]. This model was selected based on a separate study which gauges the performance of clear sky radiation models in Southeastern Australia [4]. The clear sky PV output estimates were generated by a series of radiation and PV performance models, and designed to specifically represent the 30kW system in the Canberra dataset. Normalisation of the measured output by this clear sky estimate gives the clearness index for photovoltaics, K_{PV} . A full description of this process can be found in [3].

Given normalised and unnormalised data, we are able to consider two forms of prediction, one for each data format. The models are developed for this normalised data, exactly as they are for the original data, but with one extra step. After making predictions, these are again transformed using the clear-sky model to give error measurements in the original (unnormalised) space.

5.3.1.4 Prediction Methods

We now describe the four prediction methods. Our basic methodology is to train a function $f(x)$ which will take a set of “features” x and predict the PV or radiation data for the new time. If exogenous data is excluded, then x is a vector of length 7 consisting of the following values:

- A constant of 1 (used to allow the model $f(x)$ to have an offset from zero at $x = 0$)
- The measurement at the same time 24 hours previously
- The measurement one/two/.../five unit(s) of time previously

If exogenous data is included, then x is a vector of length 12, consisting of the previous 7 values, plus the five meteorological measurements discussed in Section 5.3.1.2.

The models are trained by forming the feature vectors x^k as above for all the times k in the training set, along with the corresponding true value y^k for the measurement being predicted. Then, the model was trained through “empirical risk minimisation”. Namely, the function f was selected to minimise

$$c \sum_k L(f(x^k), y^k) + h(f),$$

where L is a “loss function” that measures the error associated with a given prediction. Here $h(f)$ is a “regulariser” that penalises more complex functions f . The constant c can be adjusted to trade off between emphasising the loss function and the regulariser.

Linear with Squared Error

The classic method for regression is to fit a linear function under a squared loss. That is, to use the function

$$f(x^k) = w \cdot x^k,$$

where \cdot denotes the inner product. Along with this, the loss function is simply the squared difference

$$L(f(x^k), y^k) = (f(x^k) - y^k)^2,$$

meaning that the learning problem is to find the vector w that minimises

$$\sum_k (w \cdot x^k - y^k)^2.$$



This can very efficiently be done by reduction to a simple linear system. Here, we slightly generalise this to an ϵ -insensitive squared loss, where errors are measured as

$$L(f(x^k), y^k) = \max(0, |f(x^k) - y^k| - \epsilon)^2,$$

for an $\epsilon \in \{10^{-3}, 10^{-2}, 10^{-1}\}$, which was automatically set to optimize test-set performance.

Here, a regulariser of the form

$$\|w\|^2$$

was used. The constant $c \in \{0.01, 0.1, 1, 10, 100\}$ automatically selected to optimize test-set performance.

Linear with Absolute Error

Because errors are sometimes measured with the absolute error, rather than the squared error, we also consider again using a linear function $f(x^k) = w \cdot x^k$, but trained to minimise the ϵ -insensitive absolute loss

$$L(f(x^k), y^k) = \max(0, |f(x^k) - y^k| - \epsilon) \quad (5.1)$$

instead of the squared loss. The same regularisation was used as above, again with ϵ and c automatically set to optimize test-set performance.

Both this model and the previous model are fit using the LIBLINEAR package, which performs coordinate ascent in a dual reformulation of the optimisation problem [5].

Neural Networks (Multi-Layer Perceptron)

Another standard model to use is the neural networks [6, 15]. Here, we use the most common variant, consisting of a single hidden-layer. The final function which is learned is of the form

$$f(x^k) = w \cdot \sigma(Ux^k),$$

for a “sigmoid” function $\sigma(s) = \frac{1}{1+\exp(-s)}$. We overload the notation by defining σ applied to a vector to mean applying it to each component of the vector. Here, U is a matrix of size either $H \times 7$ or $H \times 12$, depending on if exogenous features are used, where $H \in \{10, 20, 40\}$ the number of hidden units, again set automatically to optimize test-set performance. This can be thought of as first mapping x^k to the nonlinear “feature space” $\sigma(Ux^k)$, and then performing linear regression on this space. The number of training epochs (passes through the whole training set) has an upper bound of 5.

Here, we found regularisation to be relatively unimportant, and so simply used a squared regulariser with a small constant.

A downside of neural networks, compared to linear methods, is that the problem of finding the optimal parameters (w, U) is non-convex, meaning that the globally optimal parameters might not be recovered. A total of five passes of stochastic gradient descent was used to find parameters.

Baseline Persistence Model

It is common practice in the solar forecasting literature to compare the more advanced forecast methods to an extremely naïve “predictor” that simply predicts that the PV output or radiation data will remain the same over into the next forecast interval. For unnormalised data, the current measurements is assumed to remain constant over the next 10 minutes or 30 minutes. In the case of hourly data, the power output from the previous 24 hours is used e.g [1, 15]. In the case of the normalised data, the current value is always used for the next time-step estimate, as the diurnal cycle has been removed.

5.3.1.5 Experimental Design

The benchmarking study employs the four different prediction methods outlined in Section 4, trained and tested on the four data types (unnormalised and normalised solar radiation and PV time series), which were selected to replicate the current state of the art in machine learning based solar forecasting. Each of these experiments is then also tested on three time horizons, $t + 1$, $t + 2$, or $t + 3$, on the three averaged data intervals (10 minutes, 30 minutes, or an hour). These intervals were chosen to be consistent with the ongoing research in the field, hourly [2, 8, 12, 16], sub-hourly [15] and minute level [13, 18].



Performance measures

Four performance measures were selected to analyse the performance of these models in accordance to those most commonly used in solar radiation/PV power output modelling research [4]. To define these measures, we define the true sequence of values as a vector $\bar{y} = (\bar{y}_1, \dots, \bar{y}_n)$ and the predicted values are then $y = (y_1, \dots, y_n)$. To compare \bar{y} against y , we use the following:

The first is the **mean bias error**,

$$MBE = \frac{1}{n} \sum_{i=1}^n (\bar{y}_i - y_i).$$

This measures the bias of the predictor – if it tends to predict higher values or lower values. Note that a very poor predictor could still have zero mean bias error (e.g. simply predicting the average of \bar{y}).

The second performance measure is the **mean absolute error**,

$$MAE = \frac{1}{n} \sum_{i=1}^n |\bar{y}_i - y_i|,$$

measuring the mean difference between the predicted value and the true value.

A similar error metric is the **root mean squared error** (RMSE)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\bar{y}_i - y_i)^2}.$$

The main difference is that the RMSE is more sensitive to “outliers” than the mean absolute error: Given a set of predictions, most of which are very accurate, but with a few very inaccurate predictions, the RMSE will tend to be high since the large errors are squared. But the mean absolute error will “average out” these large errors.

Finally, the **coefficient of determination** is

$$R^2 = 1 - \frac{\sum_{i=1}^n (\bar{y}_i - y_i)^2}{\sum_{i=1}^n (\bar{y}_i - \bar{Y}_i)^2} \quad (5.2)$$

which is a measure of the correlation between the true sequence and predicted values (where \bar{Y}_i is the overall mean of the true sequence). Unlike the other error measures, a higher coefficient of determination is better. The measure is suggested for use by another benchmark oriented study [12].

5.3.1.6 Results

Experimental results are presented in Tables 3-10. Overall, there were four forecast methods (Baseline persistence, Linear L1, Linear L2, neural network) tested on the two major datasets, global horizontal solar radiation from Wagga, and photovoltaic power output data from a 30 kW rooftop installation in Canberra, respectively. These data were broken into three different data intervals - 10 minute, 30 minute and 60 minute (hourly), and then all three ML methods were trained and tested on $t + 1$, $t + 2$, $t + 3$ forecast horizons. This sequence of experimentation was then repeated for both datasets using exogenous meteorological data inputs, and clear sky normalised data. The results from these experiments are broken down into discussion by the type of data being forecast in the time series.

Solar Radiation Forecasts

In general, the three forecasts methods outperformed the baseline persistence model. There were limited exceptions to this, most of which occur in the 10 minute interval experiments. Over short time horizons, persistence forecasts have a much stronger chance of being accurate, unless the cloud cover is fluctuating quickly. In the case of the unnormalised data, the persistence forecasts lost accuracy much more quickly than the normalised data, as the diurnal cycle is not removed. Quite interestingly, the normalised persistence model has lower MAE values than either of the three ML methods in forecasts of up to approximately 1 hour. However, a review of the nRMSE and R^2 values suggest the baseline persistence model is really only competitive on very short, 10-20 minute forecast horizons, once all error measures are jointly considered.



Amongst the ML models tested, the results are varied across the four error measures. The neural network's performance is unreliable. Its MAE and nRMSE values increase with the forecast horizon in the 10 minute interval forecasts, but do the opposite with the 30 minute data. It also experiences increases in error measures with normalisation of the data. Interestingly, the neural network delivers the most accurate forecast for the 10 minute forecast horizon (10 minute data, $t+1$) while also giving the most accurate 3 hour forecasts (hourly data, $t+3$). This result is consistent with similar findings in other solar forecasting research using neural networks [9]. In the linear methods, the L1 loss Support Vector Machine (SVM) either outperforms or is equivalent to the L2 loss SVM method at timescales less than 60 minutes, and has much better accuracy at time horizons greater than one hour. Interestingly, the normalisation of the data tends to make the performance of the models more similar. It generally decreases their accuracy at forecast horizons of less than one hour, but the opposite is true for the hourly data forecast (60, 120, 180 minutes). The overall trend can be seen best in Tables 5.10 and 5.9, which summarised the results from the previous 8 Tables, giving the value from the most accurate method. To date, we are unaware of any studies who have investigated the benefits of time series normalisation on solar forecasting performance. As such, it is recommended that this be further explored.

Overall, the inclusion of exogenous data confuse the results, and consistently deliver less accurate performance, in most cases increasing error by an order of magnitude. This is in conflict with work in [15] who suggested their inclusion would increase accuracy marginally, but their work was focused on daily intervals and may not be relevant in this context. It is however supported by the research undertaken in [6] who tested the benefit of including temperature, dew point, relative humidity, windspeed and precipitable water in an MLP neural network and concluded that none of these exogenous values contributed to better forecast results.

Photovoltaic Power Output Forecasts

The baseline persistence forecast for the PV power output data follow the same trend as that of the radiation data, but with slightly higher overall error values. Here, normalisation of the data further improved the performance of the baseline forecasts for 10-60 minute horizons, but was detrimental to those on hourly timescales. Coincidentally, at short forecast horizons, the persistence model outperforms the L2 loss SVM and the neural networks, receiving lower nRMSE and MAE values. This relationship changes as the forecast horizon is increased, with all ML methods eventually surpassing the baseline in accuracy. Overall, the L1 loss SVM has the lowest nRMSE and MAE values, but its R^2 scores seem to indicate poor performance for longer term forecasts (60+ minute) forecasts. Again, as with the solar radiation experiments, the exogenous data inclusion is detrimental to the results in all methods at all time intervals. Of the studies concerned herein, only [9] directly ran experiments with photovoltaic data. They report RMSE (R^2) values of 40W (0.94) at 10 minute forecast horizons, climbing (falling) to approximately 55 W (0.85) at 40 minutes using an ANN on unnormalised data, which are slightly more proficient than the results we obtained with an ANN (R^2 values of 0.881 - 0.874 in this time frame).

5.3.1.7 Conclusions

- The models tested do better than the persistence model by only using data for prediction at single sites, but without visual data.
- Normalisation helps in some cases, but interpreting the results requires care.
- Neural Networks are a bad choice to solve the problem as the performance of those models are unreliable for different time horizons.
- Different error measures applied to the training of the models make a difference, but note that the two considered error measures are both symmetric.
- As per background from machine learning, proper regularisation is important to avoid overfitting and preserve the generalisation power of the trained model.
- Exogenous features are generally not helpful for the current models.
- These baseline results will be used to compare with our future more elaborate forecasts methods that exploit a refined set of measurements together with camera data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.000	-0.000	-0.000	-0.000	-0.000	-0.000	-0.000	-0.000	-0.000
	L_1 loss SVM	-0.000	-0.000	-0.000	-0.000	-0.001	-0.004	-0.002	-0.004	-0.004
	L_2 loss SVM	-0.001	-0.001	-0.002	-0.007	-0.008	-0.009	-0.008	-0.011	-0.013
Yes	Neural net	-0.003	0.040	-0.137	-0.050	-0.121	0.278	-0.106	-0.005	0.639
	Baseline	0.002	0.003	0.004	0.005	0.006	0.006	0.002	0.004	-0.000
	L_1 loss SVM	-0.022	-0.023	-0.023	-0.024	-0.024	-0.024	-0.025	-0.019	-0.015
	L_2 loss SVM	-0.022	-0.022	-0.022	-0.023	-0.023	-0.023	-0.024	-0.024	-0.024
	Neural net	0.000	0.326	-1.590	1.035	-2.469	0.421	-0.328	0.030	-0.001
	With exogenous features									
No	L_1 loss SVM	0.002	0.000	0.003	-0.000	-0.002	-0.001	-0.001	-0.002	-0.000
	L_2 loss SVM	0.002	0.004	0.007	-0.011	-0.007	-0.005	-0.011	-0.010	-0.009
	Neural net	-0.079	-0.003	0.011	-0.028	-0.079	-0.053	-0.007	0.039	-0.032
Yes	L_1 loss SVM	0.012	0.007	-0.001	-0.001	0.001	-0.001	-0.009	0.004	0.002
	L_2 loss SVM	-0.143	0.004	0.093	-0.012	0.051	0.076	0.028	0.052	0.010
	Neural net	-0.493	-0.097	0.208	-0.098	-0.119	-0.103	-0.068	2.569	-0.101

Table 5.1: Mean bias error on Radiation data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		Prediction horizon			Prediction horizon			Prediction horizon		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.040	0.047	0.055	0.062	0.084	0.106	0.085	0.120	0.158
	L_1 loss SVM	0.033	0.039	0.045	0.037	0.055	0.077	0.052	0.087	0.121
	L_2 loss SVM	0.046	0.054	0.061	0.111	0.125	0.139	0.123	0.144	0.162
Yes	Neural net	0.069	0.106	0.145	0.080	0.125	0.310	0.114	0.095	0.648
	Baseline	0.032	0.035	0.038	0.038	0.048	0.057	0.065	0.076	0.091
	L_1 loss SVM	0.068	0.069	0.069	0.066	0.066	0.066	0.065	0.065	0.065
With exogenous features										
No	L_1 loss SVM	0.036	0.042	0.048	0.038	0.057	0.078	0.055	0.088	0.122
	L_2 loss SVM	0.048	0.056	0.063	0.107	0.123	0.139	0.117	0.141	0.162
	Neural net	0.120	0.044	0.083	0.087	0.154	0.163	0.092	0.121	0.122
Yes	L_1 loss SVM	0.149	0.162	0.116	0.118	0.118	0.116	0.120	0.122	0.120
	L_2 loss SVM	0.764	0.732	0.875	0.465	0.508	0.505	0.352	0.346	0.315
	Neural net	0.875	0.102	0.358	0.151	0.152	0.192	1.261	2.573	0.109

Table 5.2: Mean absolute error on Radiation data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		Prediction horizon			Prediction horizon			Prediction horizon		
		$t+1$	$t+2$	$t+3$	$t+1$	$t+2$	$t+3$	$t+1$	$t+2$	$t+3$
Without exogenous features										
No	Baseline	0.016	0.017	0.019	0.014	0.017	0.020	0.080	0.101	0.125
	L_1 loss SVM	0.013	0.015	0.016	0.010	0.012	0.014	0.045	0.066	0.086
	L_2 loss SVM	0.014	0.016	0.017	0.016	0.018	0.020	0.081	0.095	0.107
Yes	Neural net	0.018	0.024	0.028	0.016	0.023	0.035	0.076	0.066	0.340
	Baseline	0.023	0.037	0.039	0.023	0.034	0.045	0.095	0.163	0.230
	L_1 loss SVM	0.022	0.022	0.022	0.014	0.014	0.014	0.061	0.061	0.061
	L_2 loss SVM	0.022	0.022	0.022	0.014	0.014	0.014	0.061	0.061	0.061
	Neural net	0.018	0.093	0.453	0.180	0.426	0.073	0.273	0.061	0.055
	With exogenous features									
No	L_1 loss SVM	0.013	0.015	0.016	0.010	0.012	0.014	0.045	0.066	0.087
	L_2 loss SVM	0.014	0.015	0.017	0.016	0.018	0.020	0.081	0.095	0.107
	Neural net	0.027	0.014	0.019	0.013	0.023	0.025	0.064	0.075	0.086
Yes	L_1 loss SVM	0.052	0.056	0.042	0.025	0.026	0.026	0.124	0.126	0.122
	L_2 loss SVM	0.263	0.256	0.286	0.098	0.103	0.105	0.353	0.360	0.328
	Neural net	0.294	0.033	0.121	0.030	0.035	0.040	1.235	2.125	0.104

Table 5.3: Normalized root mean squared error on Radiation data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		Prediction horizon			Prediction horizon			Prediction horizon		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.886	0.867	0.837	0.767	0.656	0.518	0.640	0.426	0.114
	L_1 loss SVM	0.928	0.902	0.880	0.869	0.814	0.750	0.885	0.755	0.586
	L_2 loss SVM	0.912	0.889	0.866	0.673	0.595	0.509	0.627	0.488	0.349
Yes	Neural net	0.848	0.741	0.645	0.679	0.327	0.000	0.671	0.754	0.000
	Baseline	0.753	0.386	0.298	0.350	0.000	0.000	0.488	0.000	0.000
	L_1 loss SVM	0.785	0.778	0.778	0.763	0.763	0.763	0.789	0.789	0.790
With exogenous features										
No	L_1 loss SVM	0.927	0.903	0.883	0.869	0.814	0.749	0.884	0.758	0.573
	L_2 loss SVM	0.912	0.891	0.870	0.670	0.595	0.513	0.632	0.496	0.359
	Neural net	0.666	0.906	0.837	0.786	0.342	0.243	0.770	0.682	0.579
Yes	L_1 loss SVM	0.000	0.000	0.194	0.225	0.185	0.198	0.140	0.101	0.166
	L_2 loss SVM	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	Neural net	0.000	0.497	0.000	0.000	0.000	0.000	0.000	0.000	0.396

Table 5.4: Coefficient of determination on Radiation data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.000	-0.000	-0.000	0.000	0.000	0.000	0.001	0.001	0.001
	L_1 loss SVM	-0.000	-0.001	-0.000	0.002	-0.007	-0.009	-0.004	-0.010	-0.010
	L_2 loss SVM	-0.002	-0.002	-0.003	-0.005	-0.011	-0.015	-0.011	-0.020	-0.030
Yes	Neural net	0.043	-0.000	-0.000	0.004	-0.002	-0.014	0.003	-0.026	-0.009
	Baseline	0.001	0.001	0.001	0.005	0.007	0.008	0.009	0.021	0.026
	L_1 loss SVM	-0.002	-0.005	-0.007	-0.004	-0.019	-0.021	-0.010	-0.009	-0.011
	L_2 loss SVM	-0.032	-0.036	-0.039	-0.037	-0.043	-0.046	-0.049	-0.052	-0.055
	Neural net	-0.066	0.000	-0.041	0.003	0.007	-0.036	-0.362	-0.054	-0.030
	With exogenous features									
No	L_1 loss SVM	0.001	-0.000	-0.003	0.000	-0.006	-0.008	-0.005	-0.010	-0.004
	L_2 loss SVM	0.001	0.000	-0.001	-0.001	-0.003	-0.008	-0.007	-0.017	-0.025
	Neural net	-0.000	-0.000	-0.001	0.003	0.009	0.010	-0.004	-0.010	0.009
Yes	L_1 loss SVM	0.001	-0.002	-0.000	-0.002	-0.005	-0.015	-0.008	-0.011	-0.013
	L_2 loss SVM	-0.005	-0.009	-0.013	-0.016	-0.024	-0.031	-0.004	-0.015	-0.025
	Neural net	0.009	0.063	0.008	0.013	-0.004	-0.054	-0.148	-0.031	-0.041

Table 5.5: Mean bias error on PV output data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		Prediction horizon			Prediction horizon			Prediction horizon		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.041	0.049	0.057	0.065	0.089	0.113	0.087	0.126	0.167
	L_1 loss SVM	0.029	0.040	0.047	0.038	0.060	0.079	0.051	0.090	0.129
	L_2 loss SVM	0.037	0.047	0.054	0.042	0.064	0.084	0.056	0.095	0.128
Yes	Neural net	0.072	0.040	0.047	0.040	0.053	0.085	0.051	0.071	0.108
	Baseline	0.034	0.038	0.042	0.043	0.056	0.069	0.072	0.096	0.123
	L_1 loss SVM	0.028	0.035	0.039	0.035	0.052	0.060	0.055	0.071	0.086
	L_2 loss SVM	0.051	0.057	0.060	0.059	0.068	0.072	0.073	0.078	0.083
	Neural net	0.071	0.036	0.085	0.031	0.039	0.064	0.378	0.075	0.075
	With exogenous features									
No	L_1 loss SVM	0.033	0.042	0.051	0.039	0.060	0.079	0.051	0.090	0.128
	L_2 loss SVM	0.039	0.051	0.060	0.043	0.064	0.084	0.056	0.095	0.129
	Neural net	0.031	0.041	0.048	0.038	0.062	0.079	0.046	0.080	0.103
Yes	L_1 loss SVM	0.028	0.034	0.037	0.035	0.049	0.058	0.051	0.070	0.086
	L_2 loss SVM	0.045	0.050	0.054	0.054	0.063	0.069	0.067	0.073	0.078
	Neural net	0.122	0.112	0.069	0.042	0.071	0.116	0.212	0.071	0.092

Table 5.6: Mean absolute error on PV output data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.087	0.097	0.106	0.120	0.158	0.195	0.167	0.220	0.276
	L_1 loss SVM	0.070	0.083	0.091	0.075	0.101	0.127	0.087	0.145	0.201
	L_2 loss SVM	0.069	0.081	0.090	0.074	0.101	0.126	0.087	0.140	0.185
Yes	Neural net	0.082	0.078	0.085	0.071	0.091	0.136	0.080	0.122	0.158
	Baseline	0.084	0.089	0.094	0.118	0.179	0.247	0.178	0.375	0.610
	L_1 loss SVM	0.066	0.075	0.079	0.073	0.102	0.123	0.113	0.185	0.249
	L_2 loss SVM	0.090	0.098	0.103	0.112	0.128	0.137	0.143	0.154	0.164
	Neural net	0.120	0.073	0.147	0.068	0.082	0.123	0.685	0.149	0.164
	With exogenous features									
No	L_1 loss SVM	0.070	0.083	0.091	0.075	0.101	0.127	0.087	0.144	0.200
	L_2 loss SVM	0.069	0.082	0.091	0.075	0.101	0.126	0.087	0.140	0.185
	Neural net	0.066	0.078	0.085	0.071	0.094	0.114	0.080	0.120	0.145
Yes	L_1 loss SVM	0.067	0.074	0.078	0.074	0.099	0.123	0.108	0.183	0.267
	L_2 loss SVM	0.083	0.090	0.094	0.105	0.123	0.133	0.137	0.151	0.162
	Neural net	0.244	0.194	0.128	0.089	0.142	0.231	0.416	0.148	0.200

Table 5.7: Normalized root mean squared error on PV output data.



Normalization	Prediction algorithm	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		Prediction horizon			Prediction horizon			Prediction horizon		
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$
Without exogenous features										
No	Baseline	0.866	0.836	0.804	0.792	0.640	0.455	0.610	0.326	0.000
	L_1 loss SVM	0.913	0.880	0.855	0.920	0.854	0.768	0.895	0.706	0.432
	L_2 loss SVM	0.916	0.884	0.857	0.920	0.853	0.771	0.893	0.725	0.520
Yes	Neural net	0.881	0.894	0.874	0.927	0.882	0.736	0.912	0.790	0.650
	Baseline	0.876	0.861	0.844	0.801	0.541	0.123	0.557	0.000	0.000
	L_1 loss SVM	0.924	0.902	0.891	0.922	0.852	0.784	0.822	0.520	0.134
	L_2 loss SVM	0.858	0.831	0.813	0.819	0.763	0.730	0.714	0.670	0.625
	Neural net	0.746	0.906	0.622	0.934	0.903	0.783	0.000	0.691	0.625
	With exogenous features									
No	L_1 loss SVM	0.914	0.880	0.855	0.920	0.854	0.768	0.895	0.711	0.440
	L_2 loss SVM	0.916	0.882	0.854	0.920	0.853	0.772	0.894	0.726	0.520
	Neural net	0.923	0.894	0.874	0.928	0.872	0.812	0.910	0.799	0.704
Yes	L_1 loss SVM	0.922	0.903	0.892	0.921	0.861	0.784	0.836	0.533	0.002
	L_2 loss SVM	0.879	0.859	0.844	0.842	0.785	0.745	0.736	0.682	0.633
	Neural net	0.000	0.340	0.711	0.887	0.709	0.235	0.000	0.695	0.442

Table 5.8: Coefficient of determination on PV output data.



Performance Measure	Normalization	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes		
		t + 1	t + 2	t + 3	t + 1	t + 2	t + 3	t + 1	t + 2	t + 3
Without exogenous features										
Mean bias error	No	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Mean absolute error	No	0.033	0.039	0.045	0.037	0.055	0.077	0.052	0.087	0.121
Normalized root mean squared error	No	0.013	0.015	0.016	0.010	0.012	0.014	0.045	0.066	0.086
Coefficient of determination	No	0.928	0.902	0.880	0.869	0.814	0.750	0.885	0.755	0.586
Mean bias error	Yes	0.855	0.779	0.779	0.765	0.765	0.765	0.791	0.792	0.829
With exogenous features										
Mean bias error	No	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Mean absolute error	No	0.036	0.042	0.048	0.038	0.057	0.078	0.055	0.088	0.122
Normalized root mean squared error	No	0.013	0.014	0.016	0.010	0.012	0.014	0.045	0.066	0.086
Coefficient of determination	No	0.927	0.906	0.883	0.869	0.814	0.749	0.884	0.758	0.579
Coefficient of determination	Yes	0.753	0.497	0.298	0.350	0.185	0.198	0.488	0.101	0.396

Table 5.9: Summary of Radiation data.



Performance Measure	Normalization	Interval = 10 minutes			Interval = 30 minutes			Interval = 60 minutes			
		$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	$t + 1$	$t + 2$	$t + 3$	
Without exogenous features											
With exogenous features											
Mean bias error	No	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.001	0.001	
	Yes	0.001	0.000	0.001	0.003	0.007	0.008	0.009	0.009	0.011	
Mean absolute error	No	0.029	0.040	0.047	0.038	0.053	0.079	0.051	0.071	0.108	
	Yes	0.028	0.035	0.039	0.031	0.039	0.060	0.055	0.071	0.075	
Normalized root mean squared error	No	0.069	0.078	0.085	0.071	0.091	0.126	0.080	0.122	0.158	
	Yes	0.066	0.073	0.079	0.068	0.082	0.123	0.113	0.149	0.164	
Coefficient of determination	No	0.916	0.894	0.874	0.927	0.882	0.771	0.912	0.790	0.650	
	Yes	0.924	0.906	0.891	0.934	0.903	0.784	0.822	0.691	0.625	

Table 5.10: Summary of PV output data.

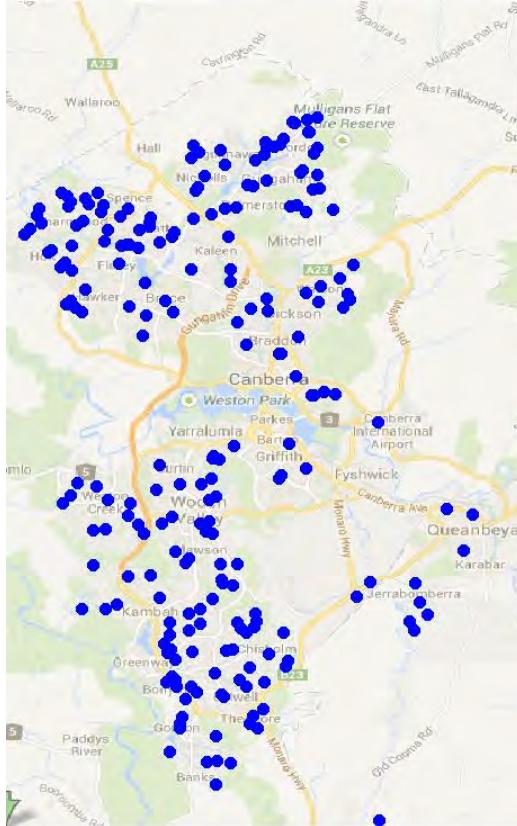


Figure 5.1: 253 PV systems (sites) distributed across Canberra. The data is retrieved from PVOutput.org, with sites gradually added since year 2011.

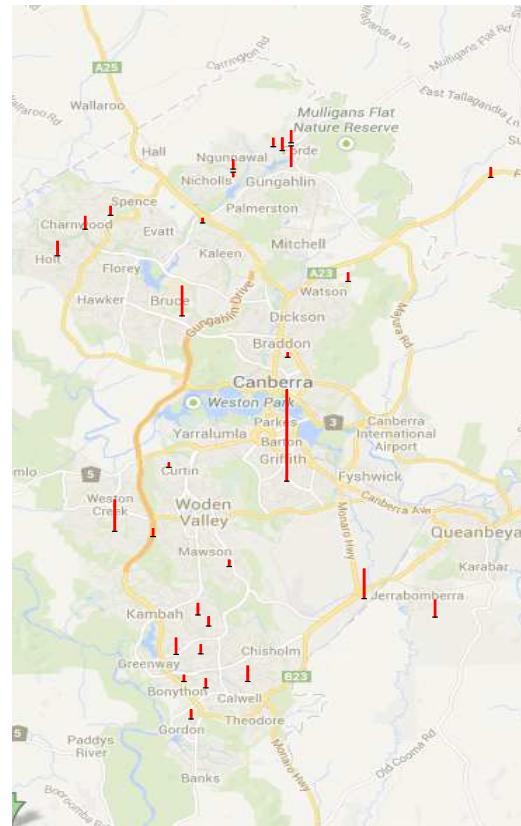


Figure 5.2: The rated power output for the 29 sites. The height of each bar is proportional to the power of the site.

5.3.2 Improving forecasting accuracy by using distributed PV network

We next developed novel approaches that further improve the forecasting accuracy by aggregating the output of a *distributed PV network*. The data we used was gathered from PVOutput.org, a website to which PV system owners in Canberra publicly report their power output. It contains 253 sites and their geographical distribution is shown in Figure 5.1. We first highlight the key issues under study and our major discoveries:

- **Local predictor versus joint predictor.** In Section 5.3.1, a site makes predictions locally, *i.e.* based on its own past measurements. However in practice the output of PV systems in a distributed network clearly exhibit temporal and spatial correlations. Therefore it can be highly beneficial to base the prediction on the past measurement of all (neighbouring) PV systems. We refer to this scheme as *joint predictor*. Our first key discovery is that this significantly improves accuracy upon the persistence baseline (*e.g.* by 31.1% for 60-min horizon).
- **Optimal range of neighbourhood for joint predictor.** A key challenge arising from joint predictor is the increased cost in communication and computation. To retain efficiency, we explored various data reduction techniques by utilizing saturation of information. For example, measurement of sites that are far away could be of little help. Our key discovery here is that for 10-minute prediction, only 30 neighbouring sites will be needed for achieving the best accuracy, while for 30 or 60 minute horizon more sites will be necessary.
- **Optimal time lag interval for training.** In Section 5.3.1, we used the output of the past 5 time steps for local predictor. In the case of joint predictor, however, such information could be redundant as the output of nearby sites are available. Indeed, we found that using only 1 time lag is enough to produce the best accuracy.



- **Optimal window of history for training.** Solar radiation exhibits yearly, seasonal and diurnal periodicity. Therefore it is conceivable that there is no need to train on many years' historical data. This was confirmed in our experiment, where one year seemed sufficient.
- **Learning with missing data.** In reality, data measurements can be unavailable at any time for various reasons. The PV system may break down, or the logger transmission component can go faulty, etc. The situation is exacerbated when we work on a distributed system of PV panels. In our data there is actually *no* time point when all sites have a meaningful reading. Therefore it will be important to address missing data without sacrificing the prediction accuracy. Our key contribution here is to find a highly effective and efficient strategy of imputation: use the mean output of available sites at the same time step.

Based on the lessons drawn from single site experiments, we fixed the model to linear predictor trained with least-squared error, and evaluated the performance by using RMSE on fractional performance ratio (kWr)—the measured power output divided by the rated power output for the array. We did check other performance measures such as mean absolute error and coefficient of correlation. The comparative results remain the same.

The dataset consists of mean kWr measured over 10-minute intervals (which we call one time step). To allow for more detailed illustration, we also experimented on a subset of 29 PV systems, with different rated power outputs as shown in Figure 5.2. The highest-power site (located in the Griffith area) has a rated power output of 29.7kW, and was used in Section 5.3.1. The lowest-power site has a rating of 1.5kW.

5.3.2.1 Local predictor versus joint predictor

We first present a summary of results in Table 5.11. It compares three methods: persistence, local predictor and joint predictor. The latter two methods were trained on the data of year 2013, with parameters set to optimal (see below). All three methods were tested on year 2014, and three horizons were studied: 10 minutes, 30 minutes, and 60 minutes. In Table 5.11, the numbers are the mean RMSE averaged over all the 253 sites and their respective time steps with meaningful measurement. Also provided in the table is the relative improvement upon persistence model in percentage. In Table 5.12, the performance is measured only on the time steps between 7 am and 5 pm every day and are *not* considered as clear-sky by [14]. The purpose here is to focus on meaningful and *hard* scenarios of prediction.

Method	Horizon = 10 minutes		Horizon = 30 minutes		Horizon = 60 minutes	
	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence
Persistence	0.0784	-	0.113	-	0.153	-
Local predictor	0.0746	4.91%	0.104	8.12%	0.128	16.2%
Joint predictor	0.0713	9.16%	0.0925	18.4%	0.106	31.1%

Table 5.11: Mean RMSE on all times steps of persistence model, local predictor, and joint predictor.

Method	Horizon = 10 minutes		Horizon = 30 minutes		Horizon = 60 minutes	
	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence
Persistence	0.09	-	0.128	-	0.17	-
Local predictor	0.0861	4.4%	0.119	7.15%	0.147	13.5%
Joint predictor	0.0807	10.4%	0.104	18.9%	0.121	29%

Table 5.12: Mean RMSE on **non-clear-sky** time steps of persistence model, local predictor, and joint predictor.

In Table 5.11, the joint predictor outperforms persistence model by 9.16%, 18.4%, and 31.1% on 10-min, 30-min, and 60-min horizons respectively. It is not surprising that persistence model is quite competitive for 10-min horizon, because generally there is only mild change in such a short period. However when the horizon gets longer, the advantage of joint predictor becomes much more substantial. The local predictor provides nearly half as much improvement as joint predictor, which corroborates the importance of aggregating the PV output of other sites for prediction accuracy.



Comparing Table 5.11 and 5.12, the test performance of all methods only deteriorates slightly when the prediction is restricted to non-clear-sky steps. The relative reduction over persistence model changes even less. This suggests that the error lies evenly in clear-sky and non-clear-sky moments. For this reason, our subsequent results are based on all time steps, *i.e.* like Table 5.11.

To illustrate the results in more detail, Figures 5.3 to 5.5 compare the three methods in scatter plots for the three horizons. Each dot corresponds to a site, and for robustness only sites with over 7000 test steps are included (about 2 months' measurements). This ends up with 234 sites/dots, while the rest 100 sites were included in Table 5.11. Clearly most dots lie below the diagonal line, which is consistent with the comparative results in Table 5.11.

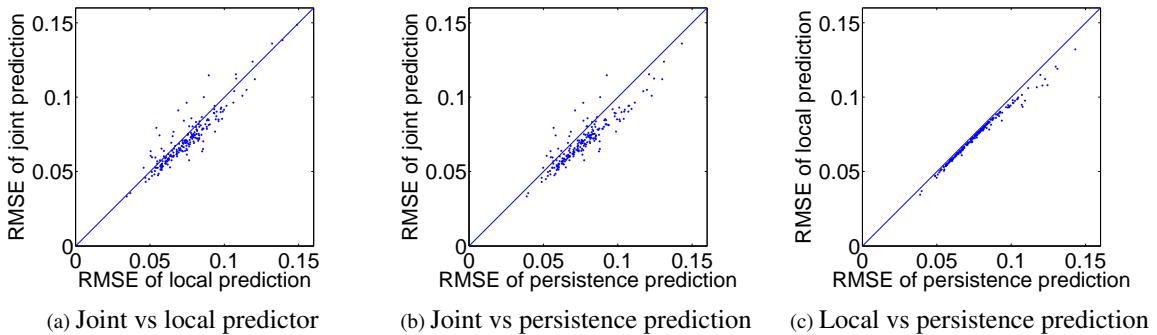


Figure 5.3: Scatter plot of RMSE for persistence, local, and joint predictors with **horizon being 10 minutes**.

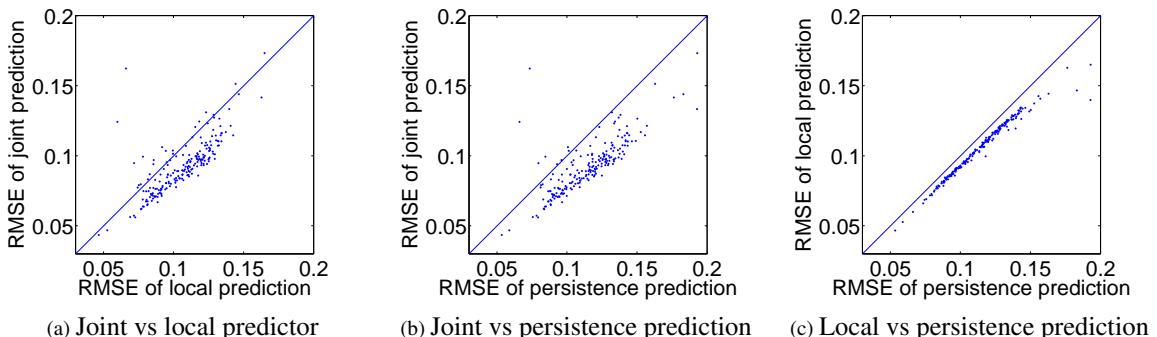


Figure 5.4: Scatter plot of RMSE for persistence, local, and joint predictors with **horizon being 30 minutes**.

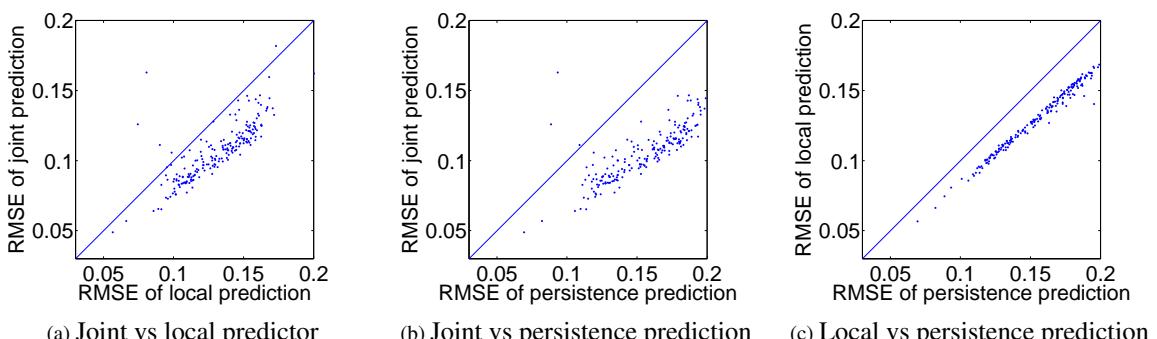


Figure 5.5: Scatter plot of RMSE for persistence, local, and joint predictors with **horizon being 60 minutes**.

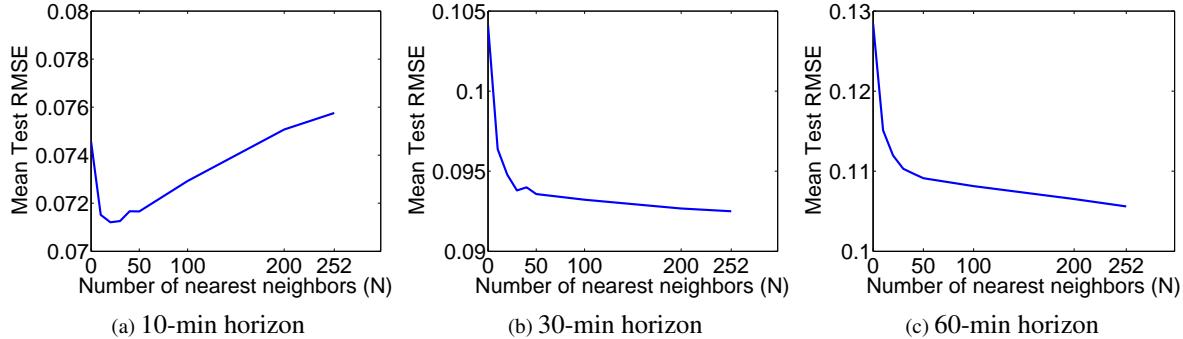


Figure 5.6: Mean test RMSE as a function of the number of nearest neighbours for joint predictor.

5.3.2.2 Optimal range of neighbourhood for joint predictor

We next study the influence of model parameters on the performance. The first factor of interest is how many nearest neighbouring sites need to be taken into account in order to make accurate predictions for each individual site. This also has direct impact on the cost of computation and wireless communication. To keep the notion of neighbourhood invariant to the geographical scale, we did not vary the radius of proximity around each site. Instead we varied the number of nearest neighbours N in 0 (no neighbor), 10, 20, 30, 50, 100, 200, and 252 (all other sites).

Figure 5.6 shows how the test RMSE (averaged over all 253 sites) depends on N , the number of neighbours used for training at each site. Interestingly in Figure 5.6a where the horizon is 10 minutes, the mean test RMSE is minimised when the number of neighbor is 30, and including more sites will in fact *impair* the performance. This phenomenon is actually reasonable. The initial decline of RMSE is consistent with our motivation that fusing data from multiple sources can improve prediction. However sites that are far away should not be helpful, especially those whose distance is more than the normal displacement of cloud over 10 minutes. In fact including them in the predictor can be even detrimental because they constitute noisy and confounding information. More training examples will be required to figure out their irrelevance, and given the limited availability of training data in this experiment, it is not surprising that the RMSE grows when N is excessively large.

When the horizon is 30 or 60 minutes as in Figure 5.6b and 5.6c, the test RMSE keeps declining for a longer range of N . This is presumably because cloud moves much longer distance in 30-60 minutes, making a larger number of nearby sites to be relevant to the prediction. The slope at the tail suggests that for 60-min horizon, one may keep benefiting from having more sites, while that gain levels off for 30-min horizon.

5.3.2.3 Optimal time lag interval for training

The input features (*i.e.* elements in the feature vector f^i) used for training are the kW_r measurements of the site at the predicted time of the *last* day, at the current time step, and at the preceding #lag time steps. To test the influence of #lag, *i.e.* the time lag interval, we varied #lag from 0 (not using any previous measurement), 1 (using the measurement of 10 minutes before), 2, ..., up to 4. Therefore joint predictors use $(2 + \text{#lag}) \times 253$ features for training, while local predictors use $2 + \text{#lag}$ features.

Figure 5.7 shows the average test RMSE over 253 sites in year 2014 as a function of #lag. When the horizon is 10 minutes as in Figure 5.8a, adding one lag reduces the mean RMSE from 0.0660 to 0.0652. But further incorporating more lags can only reduce RMSE very slightly, and is generally not worth the cost in computation. Note local predictor exhibits similar trend of dependency on time lag interval as the joint predictor.

The case of 30-min and 60-min horizon is different, where RMSE is almost invariant to the time lag window size. This is probably because 30-60 minutes is too long for earlier measurements to be helpful.

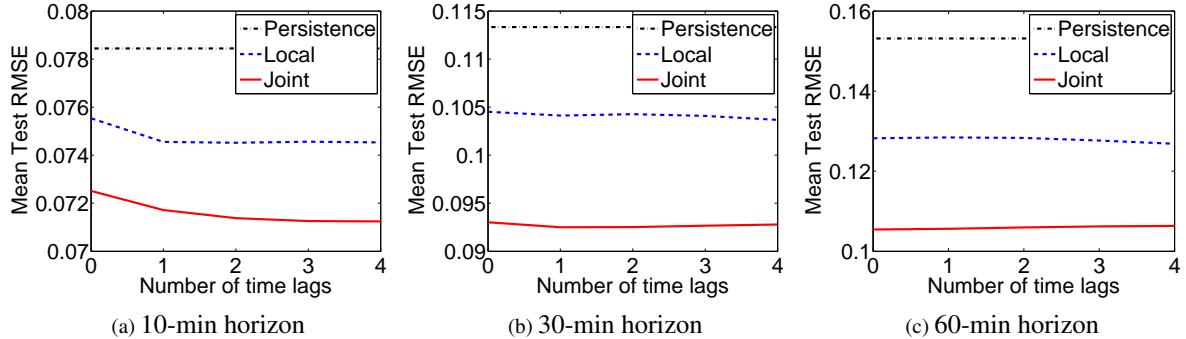


Figure 5.7: Mean test RMSE as a function of time lag interval for local and joint predictor.

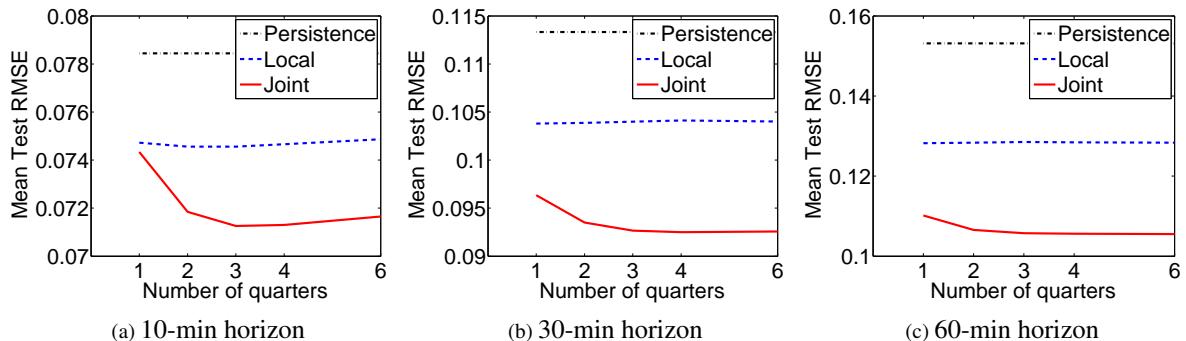


Figure 5.8: Mean test RMSE as a function of number of quarters back in history for local and joint predictor.

5.3.2.4 Optimal window of history for training

Although a few years' data has been accumulated at PVOutput.org, it is conceivable that data from the most recent years will be most valuable. Including data far back in the past will cost more computation, while the improvement on prediction accuracy could be marginal. In this section, we study the diminishing return of using old data.

In the light of the results from last section, we included the measurement of #lag=1 steps as features (in addition to the current measurement and the day before). We trained on the last q quarters in year 2012 and 2013 where $q = \{1, 2, 3, 4, 6\}$. $q = 3$ means April to December 2013, while $q = 6$ means July 2012 to December 2013. Figure 5.8 shows the evolution of mean test RMSE as q increases. For all horizons, the optimal duration of history to train on is around 3-4 quarters. Using 6 quarters will actually harm the test accuracy slightly when the horizon is 10 minutes. Therefore limiting the historical span not only saves computation, but also benefits accuracy. This is not surprising, because sites are gradually added to PVOutput.org over the years. Since the missing measurements are imputed by using the mean of other sites, this heuristic is much more inaccurate for older data when fewer sites were available.

5.3.2.5 Learning with missing data

In practice, many features can be missing. For example, at no time point do all the 29 sites have an output measurement. In the previous subsections, we have filled the missing data by using the average output of all other sites at the same point of time. In machine learning, a typical approach to this problem is to estimate a probabilistic model on $z^i := (f^i, y^i)$, where y^i is the target PV output for prediction. The most commonly used model assumes that z^i has a Gaussian distribution with mean μ and variance Σ : $z^i \sim \mathcal{N}(\mu, \Sigma)$. Let the observed components of z^i be called z_{obs}^i . Then we learn μ and Σ by maximising the likelihood

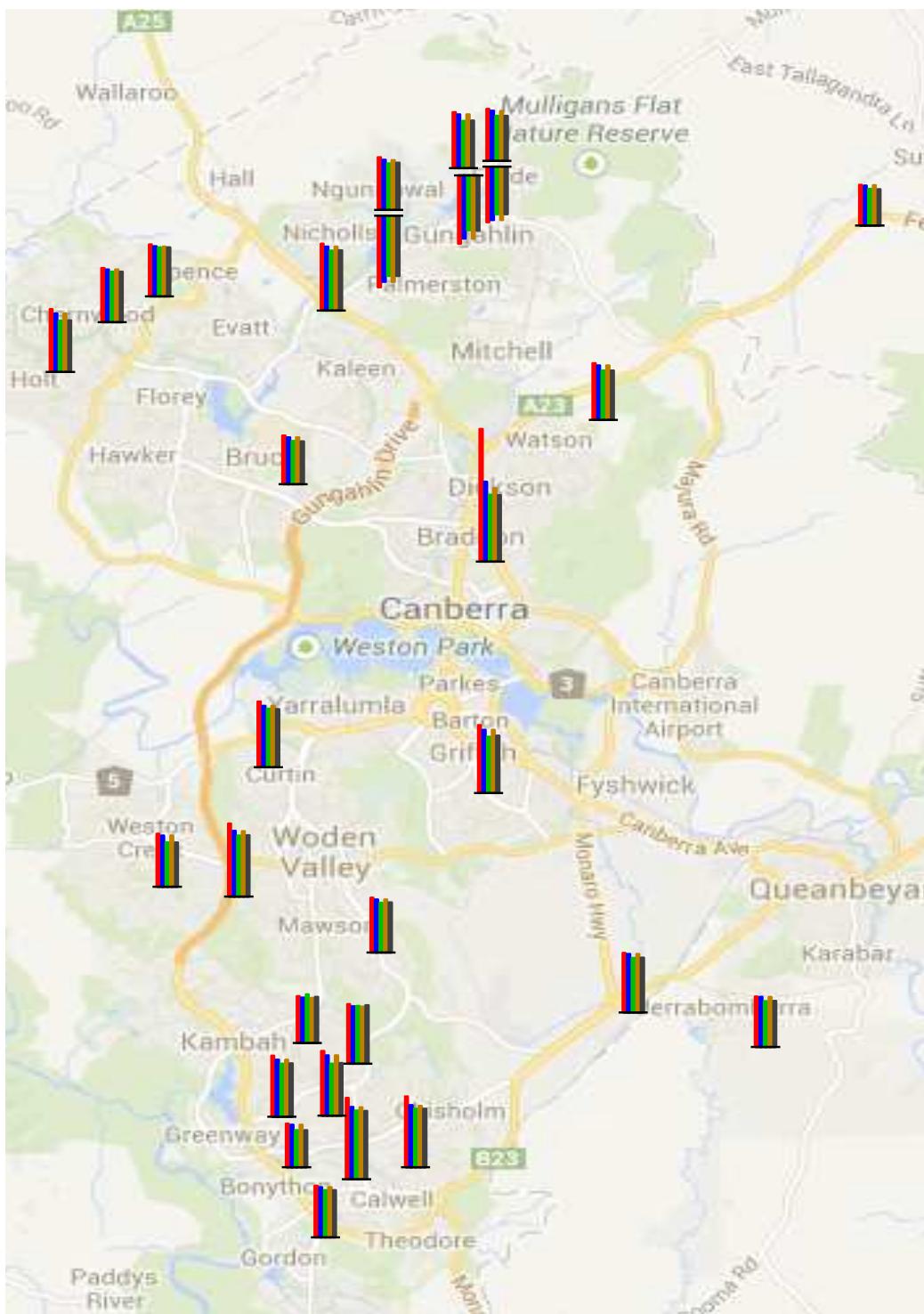


Figure 5.9: RMSE in kW^r for different methods (best viewed in colour). From left to right: i (red): the baseline persistency predictor, ii (blue): local predictor, with missing data filled by the mean of other sites, iii (green): joint predictor with missing data filled by the mean of other sites, iv (brown): local predictor with missing data filled with Gaussian, v (black): joint predictor with missing data filled with Gaussian.

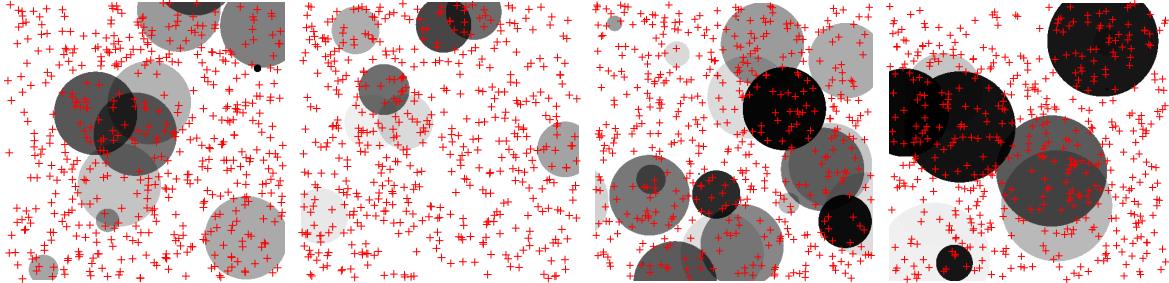


Figure 5.10: Some examples of simulation with random clouds and 500 randomly scattered sites.

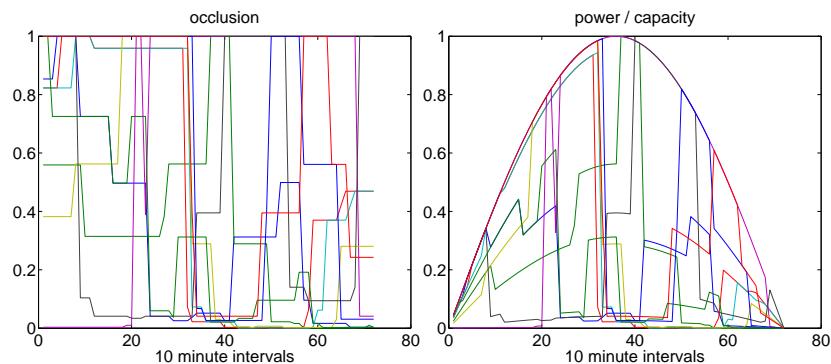


Figure 5.11: Examples of occlusion tracks and simulated power (as a fraction of total site capacity) for a day. Each line corresponds to one site (10 total).

$$\max_{\mu, \Sigma} \prod_i \Pr(z_{\text{obs}}^i; \mu, \Sigma),$$

where $\Pr(z_{\text{obs}}^i; \mu, \Sigma)$ is the marginal distribution of z_{obs}^i . At test time, we simply use μ and Σ to compute the marginal distribution y given a test feature f , and make a prediction based on its mode.

We applied this probabilistic fill-in to the 29 site dataset, and Figure 5.9 presents the result of the two imputation methods applied in conjunction with local and joint predictors. Clearly the probabilistic filling yields almost the same prediction accuracy as filling by the mean of other sites, with slight improvement at several sites. This is not surprising as Gaussian is a very coarse approximation of the joint distribution of (f^i, y^i) . As imputation by the mean is obviously more efficient in computation, we used it in all experiments.

5.3.3 Simulators for more insights

We briefly describe the simulator, which is easiest to do procedurally. First, some number of sites are randomly scattered on the unit square. For each day, a cloud creation probability is chosen uniformly on $[0, 1]$ and an overall cloud movement uniformly chosen as (δ_x, δ_y) where δ_x and δ_y are randomly chosen from $[-0.02, 0.02]$. After each time step, representing 10 minutes, a new cloud is created with this probability. Then a maximum radius r_{max} for clouds is chosen from a Poisson distribution. Next, a set of clouds are generated uniformly over the square, with sizes chosen as $\min(r^4, r_{\text{max}})$ where r is chosen uniformly on $[0, 1]$. The opacity of each cloud is chosen uniformly on $[0, 1]$. Figure 5.10 shows some example sites and cloud covers.

After each time step, the cloud centres are advanced by (δ_x, δ_y) and each dimension of the cloud position is corrupted by a “jitter” of a Gaussian distribution with standard deviation 0.005. Finally, the radius is recomputed as .95 times the old radius, and .05 times a new radius randomly chosen from the same Poisson distribution.

For each site, an “occlusion” constant is computed by multiplying the opacity of each cloud that is overhead (Figure 5.11, left). Finally, the power is generated from a 10 hour sinusoid multiplied by the occlusion (Figure

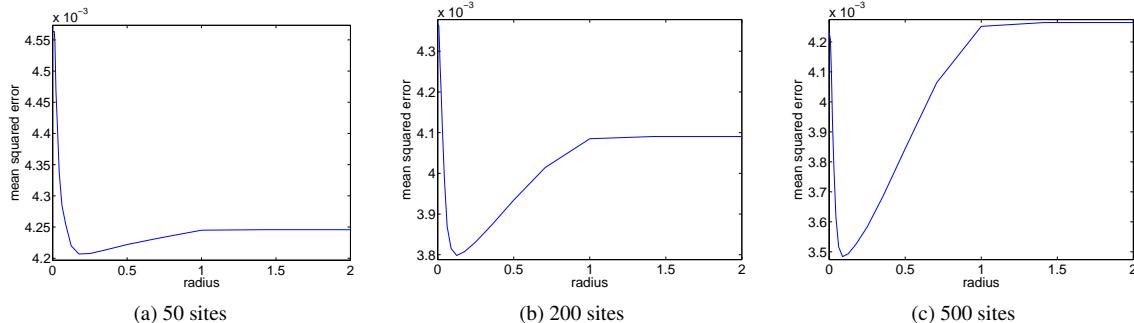


Figure 5.12: Local regularisation with the simulated data

5.11, right). All the experiments described here use one year of data for training and one year for testing.

Now, what insights were derived from the simulation? A first experiment tested the concept of "local regularisation" in connection with the total number of available sites. The idea is that faraway sites, since they are likely to be under different cloud cover, contain less predictive information than nearby sites. If overfitting were no issue, this wouldn't be a problem, but our experiments show that the amount of available data is quite critical. Thus, we experimented with building a standard linear predictor with the single change that the predictor for one site would only have access to information from other sites within a given radius. This experiment was repeated with various numbers of total sites, and is shown in Figure 5.12. The conclusions are clear: with a small number of sites, local regularisation has little impact. With a large number of sites, it has a huge impact— to the degree that with 500 sites, joint prediction is no better than local prediction unless local regularisation is used. Further, each of these curves has a minimum test error at a radius of around 0.1. Notice that this is much larger than the cloud movement, which is in the range on $(-0.02, -0.02)$ to $(0.02, 0.02)$. The reason for this, presumably, is that the clouds themselves are quite large, and thus somewhat distant sites can still provide valuable information. Nevertheless, with more years of training data, this step would not be necessary.

A second insight from the simulation was how to best make use of the geometry of wind and site locations. If we disregard training for a moment, and think of how a test-time predictor might take advantage of a known wind direction, it is easy to imagine a situation like the following: There are a set of different weight vectors, one for each of a mesh of possible wind directions. Given the particular known wind direction, the weights for the current situation are selected by a linear interpolation of the weights in the nearest grid points.

While training a system as described above might seem difficult, it is easy to do using the standard duality between feature-spaces and parameter-spaces. Namely, rather than interpolating weights, we expand the feature space by a factor of the number of mesh points. To create the features for a given point in time a "reverse-interpolation" is used. Namely, one calculates the weights that the point would receive from each of the neighbouring mesh points. Then, each of the corresponding feature spaces is given the original features, multiplied by the corresponding weight. Training this model with a fixed set of weights and "reverse-interpolated features" is exactly equivalent to training a model of the above type with an interpolated set of weights.

To test this in the simplest setting, we considered a very low-order expansion, where there are 3 grid points for the wind in each direction, for a total of 9. Thus, the feature space is expanded by a factor of nine. Because of the success of local regularisation above, and because we now have an even larger feature space, this expansion is used in conjunction with the weights being constrained to a radius of 0.1, as this was nearly optimal in all the experiments above. The grid points are chosen as $-0.02, 0$, or 0.02 , since this is the range of the wind. The results are shown in Figure 5.13. One can see that wind expansion creates a substantial improvement, which increases with the number of sites available.

In practice, of course, one will not typically have access to the exact known wind direction. An interesting question is if this can be estimated from simulated data alone. After some experimentation, we found a very simple strategy for this. Firstly, at each time, a nonlinear interpolation is used to estimate the current occlusion map on a discretized regular grid of the image. See Figure 5.14 for an example. Next, a simple image transfor-

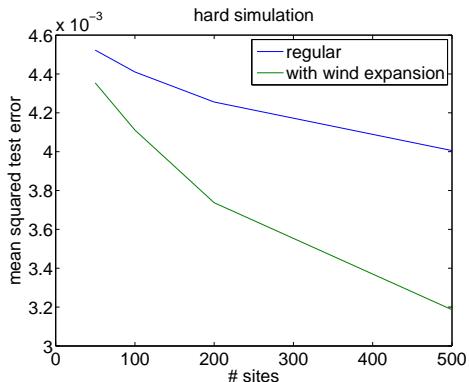


Figure 5.13: The benefits of feature expansion of the wind.

mation algorithm is used to estimate the "shift" between the two images which is equivalent (after a change of coordinates) to an estimate of the wind. Figure 5.15 shows a scatter-plot of the estimated vs. true wind using this algorithm. One can see that, while far from perfect, this generally produces an estimate close to the true one. Finally, we take this estimated wind direction and use it in the final predictor in place of the true wind. This is shown in Figure 5.16. Remarkably, there is essentially zero difference between using the estimated wind direction and the true one. This suggests that, given a sufficiently large number of sites, the wind can be estimated accurately enough to improve the predictor. This makes sense, given that the wind is still only being used in a somewhat crude way above, via wind expansion.



Figure 5.14: Interpolated occlusion maps.

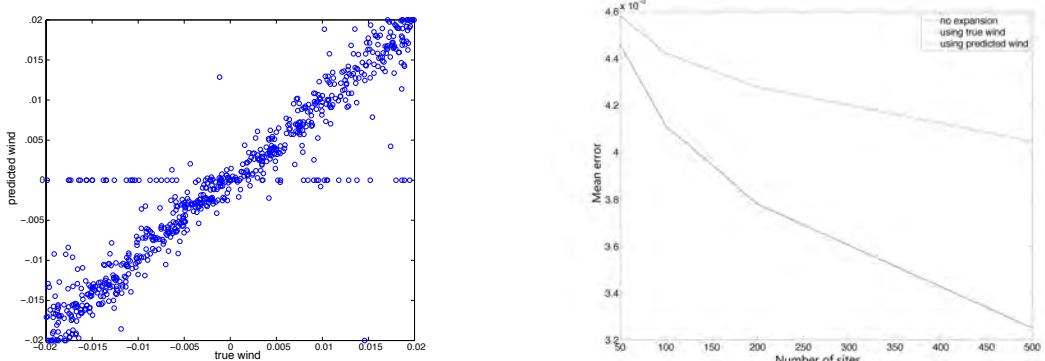


Figure 5.15: Estimated vs. true wind directions. Figure 5.16: Errors with estimated vs. true wind directions.



5.4 Extent to which the objectives were achieved

We have successfully developed and employed new algorithms to tackle the challenge of forecasting the PV output in the 10-60 minute horizon. By aggregating the data from a distributed network of PV systems, we have successfully improved the prediction accuracy by 9.2-31.1% compared with the state-of-the-art methods that are based on single sites. Furthermore by exhaustive experiments we identified the set of features that are key to this success. In particular, the appropriate choice of historical window and neighbourhood range are observed to be crucial for achieving the highest accuracy. Computationally, they also underpin the efficiency that is critical for the deployment of real systems.

We believe that the objectives of the module have been largely achieved. It should be borne in mind that the results and conclusions are derived from limited historical data because of the intrinsic availability constraints. This situation can be ameliorated when an increasing number of sites make available their measurements (through PVOutput.org). Clearly, this takes time, and collecting several years of data will take a corresponding amount of time. With the development of sensor technology, more data will become available with less noise, higher reliability, and denser geographical distribution. All these might change the influence of various factors, along with their interactions. Therefore future research will be needed to keep pace with the evolution of data collection technologies.

5.5 Highlights, difficulties and breakthroughs

To the best of our knowledge, the achievement of forecasting PV output based on a distributed network and obtaining significantly higher accuracy than localised predictors represents a world first and a significant breakthrough. The implications of such a method go far beyond the solar energy sector. The approach could have far-reaching implications for a broad range of forecasting tasks where the measurement of different time/location/part exhibit temporal and spatial correlations, e.g. wind energy prediction.

A highlight of the experimental results is that filling missing data by averaging over all other available sites performed surprisingly well in practice. It was discovered by trial and error, and we did try out more mathematically sophisticated methods such as Bayes nets, Markov random fields, and joint Gaussian filling. It turned out none of these methods managed to deliver higher accuracy than such a simple and efficient heuristic.

A number of difficulties were encountered and overcome during the development and application of the machine learning algorithm. This includes many of the normal difficulties of developing and implementing new software. Identifying and securing time on an appropriate cluster computing facility, while not strictly difficult, took time. However, the most noteworthy trap is the illusion that “the more data we use for training, the more accurate predictions can we make”. As is shown in the previous sections of this chapter, this is definitely not true, especially when real data is laden with noise and incompleteness. This phenomenon is consistent with the general philosophy of statistics, which has developed a large body of recipes such as regularisation. Our methodology has carefully incorporated these principles through, e.g., simple linear models as opposed to complicated neural network, l_2 norm penalties in least-square training, and feature selections via reduced lag interval, neighbourhood range, and window of history for training.

There is also a supply of “smarter” (usually fancier) training schemes that appear advantageous, but turn out ineffective statistically. We next mention two explorations of this kind that were attempted in the project.

Online versus batch training

One interesting question is how well a forecasting system performs when trained in a "batch" vs. "online" fashion. We tested this by taking the aforementioned 29 site data, and training both joint and single-site predictors, using various amounts of data. One year (2013) is used as test data, while parts of 2011 and 2012 are used for training. With the "batch" predictor, the most recent parts of 2011 and 2012 are used to train, and this predictor is used for all times in 2013. With the "online" predictor, however, for each time, a separate predictor is trained using the most recent weeks from 2013. This experiment tests how much performance might be improved by continuously retraining a system, rather than just using a fixed training set. The results are shown in Figure 5.17. It can be seen that online training does provide a modest benefit when the total number of data are few, but this tends to disappear with more data.

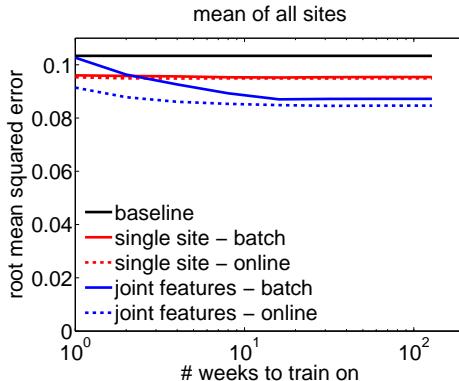


Figure 5.17: Online vs. batch training.

Making predictions for all sites versus directly predicting their sum

Another issue to consider is at what level of granularity one might make predictions. Specifically, one might only be interested in the *mean* power over an entire region, rather than the error at a site-by-site level. We investigated this by training a predictor with a single target of the mean energy (over all sites), and compared this to predicting each individual site, and then averaging. With full joint features used, the difference was essentially zero. For this reason, we pursued site-by-site prediction, since it is more amenable to ideas such a local regularisation discussed above.

Use of shading and clear-sky flags

One could conceive that prediction may benefit from side information such as clear-sky and shading flags, which can be computed using the formulae in [14]. Both flags are represented as boolean variables, which gives four combinations in total. So we divided all training and testing instances of 29 sites into four groups, and trained on them separately. In order to test the effectiveness of this scheme, we considered an overly optimistic setting where the flags used are at 10 minutes into the future, *not* at the current step.

We focused on predicting the output of each single site using its own features. From Figure 5.18, it is clear that training four models according to the two flags does not yield noticeable reduction in RMSE, compared with completely ignoring these flags. Therefore, simply dividing the task into four learning problems does not seem adequate, and more careful modelling within each scenario is necessary. For sanity check, we also tried with training and testing on clear-sky time steps *only*. Clearly, this problem is much easier than the overall task, and one can achieve much lower RMSE.

5.6 Conclusions and recommendation

To summarise, we have made a substantial breakthrough by developing machine learning algorithms that achieve state-of-the-art accuracy in predicting solar outputs by aggregating the measurement of PV panels in a distributed network. Predictive features are identified and their influence are studied in detail. This leads to a method that is efficient in computation and communication. A simulator is also developed, providing useful insights that are not available from given datasets.

To build on our success, we make the following specific recommendations:

- That PV output data for more sites should be made available for research and forecasting, because the availability of distributed solar data can significantly improve prediction accuracy. In fact, the communication cost is relatively low, with just a single real number per minute. The transmission also occurs regularly, allowing channel usage to be subscribed a priori.
- Domain knowledge be applied to cleanse the data because loggers could break down and record incorrect measurement. Besides, regular maintenance is desirable to ensure that the loggers are in good working state for measuring and sending the data.

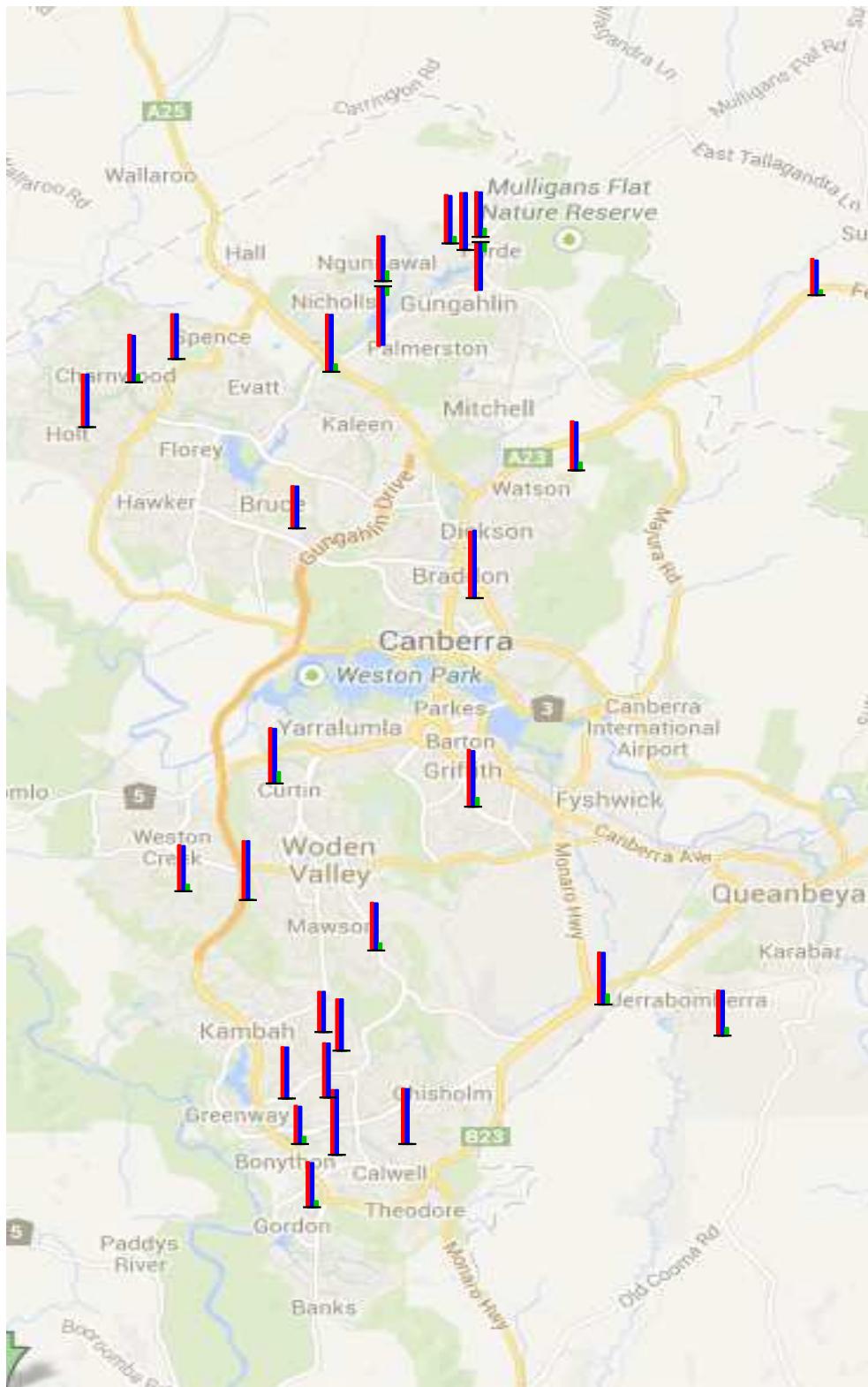


Figure 5.18: RMSE in kW h for modeling clear-sky and shade (best viewed in color). From left to right: i. ignoring the two flags as we did above (red bar), ii. using the two flags by training four models (blue bar), and iii. training and testing *only* on clear-sky time steps (green bar). Some sites do not have any clear-sky time in 2011 and 2012, hence the corresponding third bar is not available.



Bibliography

- [1] M.G. Abraha and M.J. Savage. Comparison of estimates of daily solar radiation from air temperature range for application in crop simulations. *Agricultural and Forest Meteorology*, 148(3):401–416, March 2008.
- [2] Ming Ding, Lei Wang, and Rui Bi. An ANN-based Approach for Forecasting the Power Output of Photovoltaic System. *Procedia Environmental Sciences*, 11:1308–1315, January 2011.
- [3] Nicholas A. Engerer and Franklin P. Mills. Kpv: A Clear-Sky Index for Photovoltaics. *Solar Energy*, July(105):679–693, 2014.
- [4] Nicholas A. Engerer and Franklin P. Mills. Validating Nine Clear-Sky Radiation Models in Australia. *Solar Energy (Accepted)*, 2015.
- [5] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, June 2008.
- [6] F.J. Batlles G. Lopez and J. Tovar-Pescador. Selection of input parameters to model direct solar irradiance by using artificial neural networks. *Energy*, 30:1675–1684, 2005.
- [7] C.A. Gueymard. REST2: High-performance solar radiation model for cloudless-sky irradiance, illuminance, and photosynthetically active radiation: Validation with a benchmark dataset. *Solar Energy*, 82(3):272–285, March 2008.
- [8] Jing Huang, Małgorzata Korolkiewicz, Manju R Agrawal, and John W Boland. Forecasting solar radiation on an hourly time scale using a coupled autoregressive and dynamical system (CARDS) model. *Solar Energy*, 87:136–149, 2012.
- [9] Ercan İzgia, Ahmet Öztopalb, Bihter Yerlib, Mustafa Kemal Kaymakb, and Ahmet Duran Şahinb. Short-mid-term solar power prediction by using artificial neural networks. *Solar Energy*, 86:725–733, 2012.
- [10] Tamer Khatib, Azah Mohamed, K. Sopian, and M. Mahmoud. Assessment of Artificial Neural Networks for Hourly Solar Radiation Prediction. *International Journal of Photoenergy*, pages 1–7, 2012.
- [11] R. Marquez, H.T.C. Pedro, and C.F.M. Coimbra. Hybrid solar forecasting method uses satellite imaging and ground telemetry as inputs to ANNs. *Solar Energy*, 92:172–188, 2013.
- [12] Hugo T.C. Pedro and Carlos F.M. Coimbra. Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy*, 86(7):2017–2028, July 2012.
- [13] Gordon Reikard. Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy*, 83:342–349, 2009.
- [14] Matthew J. Reno, Clifford W. Hansen, and Joshua S. Stein. Global horizontal irradiance clear sky models: Implementation and analysis. Technical Report SAND2012-2389, Sandia National Laboratories, 2012.
- [15] Ayril Voyant, Marc Muselli, Christophe Paoli, and Marie-Laure Nivet. Optimization of an artificial neural network dedicated to the multivariate forecasting of fairly global radiation. *Energy*, 36:348–359, 2011.
- [16] Dazhi Yang, Panida Jirutitijaroen, and Wilfred M. Walsh. Hourly solar irradiance time series forecasting using cloud cover index. *Solar Energy*, 86(12):3531–3543, December 2012.
- [17] Dazhi Yanga, Panida Jirutitijaroena, and Wilfred M. Walshb. Hourly solar irradiance time series forecasting using cloud cover index. *Solar Energy*, 86:3531–3543, 2012.
- [18] L. Zhang and Yun Fei Bai. Genetic algorithm-trained radial basis function neural networks for modelling photovoltaic panels. *Engineering Applications of Artificial Intelligence*, 18(7):833–844, October 2005.



Chapter 6

Forecasting with image features

6.1 Overview

This section presents the final report on the use of sky images for accurate predictions of solar power. The major source of data is the camera installed on ANU Engineering building, and high quality images have been collected since September 2014. Several efficient algorithms for extracting predictive features from images have been developed, leveraging the cutting-edge techniques from computer vision. These features were then seamlessly incorporated into the machine learning models, achieving significant improvement in solar prediction. We will describe in detail the objectives, how they are achieved, the key breakthroughs and challenges, and recommendations for future work.

6.2 Objectives

The objective of this module as defined in the Project Plan was to

Develop machine learning and computer vision analysis techniques to infer spatial and temporal solar radiation variations from PV array output and sky images, and also from numerical weather prediction models, meteorological surface observations and satellite images.

The primary objective was to develop a methodology that allows sky images to improve the PV forecasting accuracy. It breaks down into two sub-goals:

- Extract visual features that capture salient information in the image that has predictive power for PV forecasting.
- Incorporate these features effectively into machine learning models to achieve higher prediction accuracy.

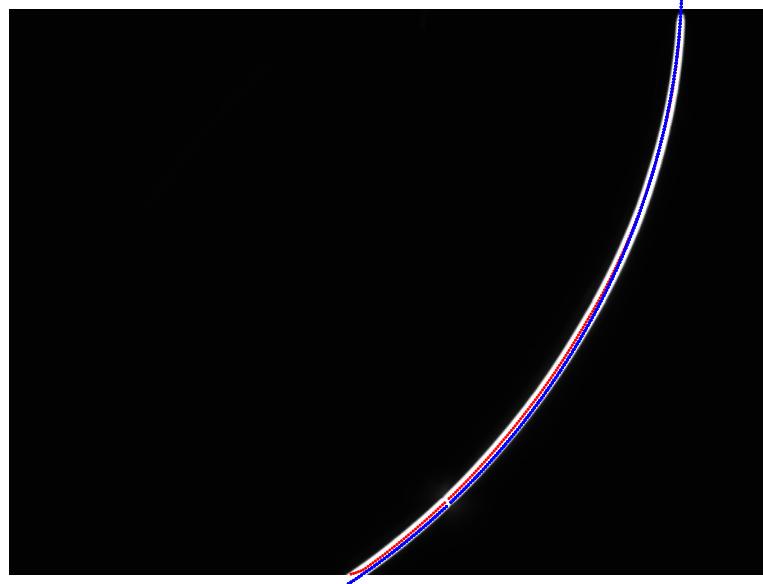
6.3 Evidence of completion

We demonstrate completion of the above two sub-goals in the following two subsections.

6.3.1 Visual feature extraction

We first describe the process of extracting visual features from radiance images. Here, the radiance images were obtained from several sky images taken under multiple exposure times (see Chapter 4 for details). We mainly extracted two types of features, including

- features from radiance image (radiance features),
- features from optical flow image (optical flow features).



Sun Trajectory Sep.30th 2014 Cam0

Figure 6.1: Illustration of computing radiance features. A set of circular regions around the projected sun location in the image are considered, with increasing radius. Then compute the mean radiance within all circles.

Figure 6.2: Comparison between the projection of the sun trajectory using the calibrated camera (blue dots) with the ground truth (red dots). In particular, this figure shows projection of the sun trajectory on 30 September, 2014. The comparison shows that our calibration algorithm can produce comparatively accurate projection results.

Here, the radiance image provides the true incident illumination values. The optical flow image describes the motion of the cloud and the sun across consecutive frames, which also encodes the illumination changes.

As for the radiance features, we mainly considered the radiance values around the projection of the sun in the image. Given an image, we extracted the time information from the image logger, which was then used to compute the zenith and azimuth angle in the world coordinate system. A calibration algorithm was developed to retrieve the correspondence between the 3D points and the image pixels. Therefore, given the calibration information, we were able to obtain the projected sun position in the image. The radiance features were then extracted by concatenating the mean radiance value of a set of circular regions around the projected sun location in the image with varying radius. See Figure 6.1 for an illustration.

We further considered two consecutive radiance frames and computed the optical flow. Note that the optical flow image encodes the motion of the 3D scene in the 2D image, which therefore describes 2D motion fields. We then extracted the optical flow features by computing the statistics of the optical flow vectors for the entire image. More details are available in the following sections, where we first describe our developed calibration algorithm, and then provide the technique details for the extraction of the two type features.

6.3.1.1 Camera calibration and radiance feature extraction

The main target of the camera calibration procedure is to obtain the projection correspondence between the points in the 3D scene and the pixels in the 2D image. This correspondence information will allow us to obtain the projected sun position in the image. Our calibration algorithm is developed based on the omnidirectional camera calibration algorithm [2]. We first retrieved the distortion centre and the coefficients for the distortion model using the OCamCalib toolbox from [2]. We then obtained the relative rotation matrix and translation vector between the world and camera coordinate system by solving a non-linear optimisation problem using the levenberg-marquardt method. Figure 6.2 shows the comparison between the projection of sun trajectory in the image using the calibrated camera model (blue dots) with the ground truth (red dots). It suggests high accuracy of our calibration method.

Given the calibration information, we obtained the radiance features by concatenating the mean radiance

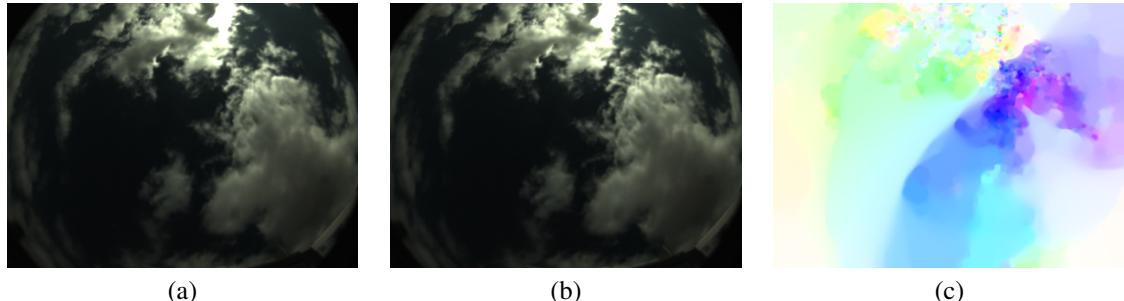


Figure 6.3: Compute the optical flow between two consecutive image frames. (a) and (b) show the consecutive image frames. (c) shows the computed optical flow encoded by RGB colours. Note that similar colour denotes the pixels move in the similar direction.

value of a circular region around the projected sun position in the image with 12 different radii. This resulted in 36 features with radiance encoded by RGB.

6.3.1.2 Optical flow feature extraction

The optical flow describes the motion of the scene across two consecutive image frames in pixels. Therefore, the optical flow for each pixel is described by a 2D vector in the image plane. We adopted the optical flow computation algorithm [1] and computed the optical flow for every two consecutive image frames. Figure 6.3 shows the computed optical flow image. We used the RGB colour to encode the optical flow direction and magnitude. Note that pixels in similar colour perform similar motion.

Given the computed optical flow, we extracted the optical flow features by computing the statistics of the optical flow vectors for the entire image. In particular, the optical flow feature for each image is defined by concatenating the following 365 features, which are

- mean optical flow over the entire image (two real numbers);
- the mean magnitude and mean (normalised) direction over the optical flow vectors in the entire image (three real numbers).
- the histogram of the direction of the optical flow in 360 bins.

6.3.2 Forecasting with image features

We next evaluate the predictive power of image features for solar forecasting. Since only the camera on ANU Engineering Building has been working for a reasonable amount of time, we focus on this site only. It would be ideal to investigate the value of image features for predicting the output of a PV system that is *collocated* with the camera. Owing to the lack of such a PV system at ANU/NICTA, we resorted to predicting the global and direct radiation¹ (Wm^{-2}) which are collected every **minute** at the Engineering Building of ANU. In total we have access to the collocated image and radiation data from September 2014 to May 2015, and we used half of them for training and half for testing.

The predictor we considered is again the simple linear model. We varied the prediction horizon in 10 and 30 minutes. We also considered the “imagined scenario” where the image is used to “predict” the radiation at the current time (*i.e.* horizon being 0). As a baseline competitor, we used the persistence model again.

For sanity check, we first computed the correlation of global radiation with all features, and the results are shown in Figure 6.4. The first three bars correspond to the RGB radiation in the circle closest to the sun, and the next three bars correspond to a larger radius. Naturally, the features closer to the sun are more correlated with the global radiation. There is strong correlation (around 0.8) when the prediction horizon is 0, and it naturally decays when the horizon grows, staying around 0.6 for 30-min horizon. The B (blue) feature seems to have a higher correlation than R and G features, because the sky is mostly blue and white.

¹Not to be confused with the radiance features extracted from images.

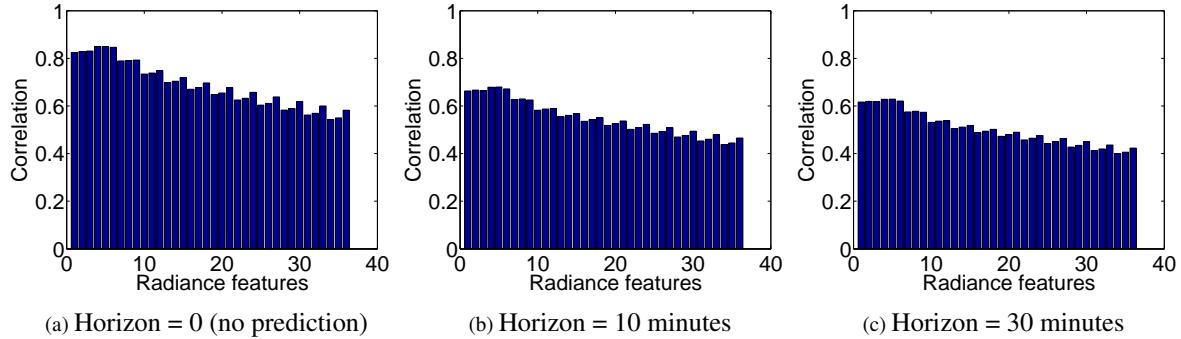


Figure 6.4: Correlation of global radiance with features under various prediction horizons.

Method	Horizon = 0 minutes	Horizon = 10 minutes		Horizon = 30 minutes	
	Mean RMSE	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence
Persistence	0	222.615	-	252.432	-
time series only	-	203.157	8.74%	231.305	8.4%
radiance only	141.84	217.717	2.20%	233.516	7.5%
optical flow only	298.13	306.073	-37.5%	310.408	-23.0%
time series + radiance	-	192.891	13.4%	210.285	16.7%
time series + optical flow	-	199.523	10.4%	221.96	12.1%
all three	-	194.741	12.5%	210.587	16.6%

Table 6.1: RMSE (Wm-2) of **global** radiation at ANU Engineering Building.

We next present our key results in Table 6.1 and 6.2, which summarise the RMSE of various predictors for global and direct radiation respectively. The time series features refer to the radiation values of the preceding 5 minutes. Several observations are in order.

- Radiance features alone do not have as much predictive power as time series features, but they do help significantly when used in conjunction with time series features. For example, when predicting global radiation with 30-min horizon, time series features can reduce RMSE over persistence by only 8.4%, while radiance features further boost the reduction to 16.7%.
- Optical flow features almost always leads to higher RMSE, and at best do not harm much when used in conjunction with radiance features.
- The benefit of using radiance features is more evident when the horizon is longer (30 minutes). This is consistent with the results in Section 5.3.2 where persistence model performs well for short horizons.
- The comparative results for global and direct radiation are very similar.



Method	Horizon = 0 minutes	Horizon = 10 minutes		Horizon = 30 minutes	
	Mean RMSE	Mean RMSE	Reduction over Persistence	Mean RMSE	Reduction over Persistence
Persistence	0	272.21	-	314.57	-
time series only	-	249.65	8.29%	288.61	8.25%
radiance only	184.42	281.53	3.43%	307.01	2.40%
optical flow only	318.84	352.63	-29.5%	360.65	-14.7%
time series + radiance	-	243.31	10.6%	277.49	11.8%
time series + optical flow	-	253.81	6.76%	288.28	8.36%
all three	-	249.67	8.28%	282.87	10.1%

Table 6.2: RMSE (Wm-2) of **direct** radiation at ANU Engineering Building.

6.4 Extent to which the objectives were achieved

We have successfully developed novel algorithms to extract predictive features from images, which when incorporated into machine learning forecasters, significantly reduces prediction errors of radiation compared with not using image features. The algorithm for extracting radiance features is efficient and scalable to large images, and the features can be readily incorporated to any standard machine learning algorithm. Importantly, these features can be computed locally at the cameras, and the results are just 36 real numbers per image, easy for transmission on wireless networks.

We believe that the objectives of the module have been largely achieved. The results are currently restricted to one camera and the radiation at its location. We have *not* yet investigated the value of image features for *joint predictors* introduced in Section 5.3.2. This is because only three cameras have been installed so far and only one of them has gathered a reasonable amount of images. Besides, we have not yet experimented with PV outputs because no PV panel is collocated with the camera. Satellite images were not used either because there is no collocated PV or radiation data. Installing hardware and gathering data take time, and more studies can be performed after they become available. That said, the positive results presented above strongly suggest that image features will considerably improve prediction accuracy for a variety of forecasting tasks.

6.5 Highlights, difficulties and breakthroughs

To the best of our knowledge, using sky images to significantly reduce errors in solar prediction is highly innovative and is a breakthrough. Analogous to the distributed machine learning forecasting techniques, this approach can be applied to a broader range of forecasting tasks where useful side information can be retrieved from images.

A highlight of our method is that we directly extract features from current and past images to facilitate solar prediction. This is different from most prevalent approaches of utilising sky images, where computer vision techniques are applied to build geometric models of the sky, simulate and extrapolate its configuration into the future, and then relate it to the quantity of interest. Instead our approach bypasses the challenging step of 3D sky reconstruction, and resorts to machine learning as the engine of prediction. Images are used as a valuable source of information, in parallel to time series from data loggers. Eventually, information from multiple (inexpensive) sources are fused in the unified framework of machine learning, which constitutes an economic and accurate prediction framework.

A highlight of the radiance features we discovered is that they exhibit all the following advantages

- They boost the forecasting accuracy,
- They can be computed efficiently and locally at each camera,



- They can be transmitted economically,
- They only require a fixed amount of storage that can be determined a priori.

The major difficulty encountered in our research is the lack of domain knowledge on what feature to extract from images such that all the four properties described above can be satisfied simultaneously. We have tried many other features, which similar to the optical flow features, did not help much. Since images are generally much more expensive to process than time series from data loggers, experimenting with new features is generally expensive in both computation and human power. For example, it took 5 seconds to compute the optical flow for each image, and processing 64,800 images collected from September 2014 to May 2015 cost a lot of computational power. Unfortunately, these features turned out not very helpful for prediction.

It is conceivable that we may start from low-resolution images and test on a small subset of the data to estimate the value of these candidate features. However, this is tricky because the computation of optical flow necessitates high resolution in images (we used 486x648 for height-x-width), and the forecasting performance based on small subsets of data can deviate significantly from if trained on a large set of data. Furthermore, it is hard to see a priori how much a feature can be useful in the context of other features such as time series and radiance. At present we can only test by trial and error, and it will be helpful to have more expert guidance.

6.6 Conclusions and recommendations

We have made a substantial breakthrough by discovering an efficient way to extract predictive features from sky images. They provide significant improvement in solar prediction, and is efficient for both computation and communication. To build on our success, we make the following specific recommendations.

- More cameras should be installed in dispersed locations in Canberra as soon as possible. The availability of distributed image information will allow for more comprehensive data analysis for distributed solar prediction.
- Experts in computer vision and solar energy should collaborate more to engineer predictive features from images. Incorporating such domain knowledge into a machine learning model can be much more effective and efficient than automatically synthesising them from data, or manually constructing them by trial and error.

Bibliography

- [1] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision*, volume 3024, pages 25–36, May 2004.
- [2] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5695–5701. IEEE, 2006.



Chapter 7

Performance measures

7.1 Overview

This section summarises the outcomes of the project component concerned with the development of performance measures for PV output prediction, and their impact on the design and training of machine learning driven forecasters. It describes the Project Agreement objectives, details the extent to which these objectives were met, highlights the challenges that were encountered, and provides recommendations for leveraging the outcomes of our analysis and experiments.

7.2 Objectives

As outlined in previous sections, the central idea of the project was the use of learning methods to make forecasts of PV output at some future time horizon. The *aim* of these forecasts is to aid the decision making of the participants in the energy market: for example, the market operator can use a forecast of low supply to request alternative generators ahead of time, while energy-conscious consumers can use a forecast of high supply in the future to decide to use some high consumption device at a later time.

The decision making capabilities of a market participant based on forecasts are constrained by at least two factors. The first factor is *how reliable* the forecasts are, according to some suitable performance measure. Ideally, such a measure reflects the participants' underlying utility (e.g. the dollar cost incurred from mispredictions); further, forecasts should ideally be produced so as to *optimise* such a reference performance measure. The second factor is the precise *nature of information* provided in these forecasts. For example, information about *uncertainty* in a forecast can allow for more sophisticated decision making (as shall be discussed subsequently).

Thus, to aid the decision making of market participants, suitably designed performance measures, and forecasts that are optimised against these measures, are of interest. This served as the focus of the present project component, whose objectives, as defined in the Project Agreement, were to:

- Develop measures of performance that serve the needs of the multiple market players involved, e.g. electricity distributors, generators, market operators, financiers, and customers;
- Develop methods whereby the machine learning (ML) methods can be adapted to the different measures of performance.

7.3 Evidence of completion

We briefly summarise the outcomes for this project component, as matched against the above objectives. On the first objective, we have carefully surveyed the existing literature in solar and wind forecasting, and conducted discussions with some market participants, to summarise some recommendations as to suitable performance measures for the various participants; further, we have demonstrated via simulations how these measures can more accurately reflect the usefulness of forecasts, compared to standard measures such as square error.



On the second objective, we have demonstrated via simulations how established statistical techniques can endow the machine learning models considered in the project with quantifiably meaningful measures of uncertainty; we also demonstrated via simulations the feasibility of adapting learning methods to optimise various performance measures, such as e.g. maximising expected revenue from bidding in the energy market.

We now describe in more detail the outcomes for each of the above objectives separately.

7.3.1 Measures of performance that serve multiple needs

To address the first objective, we pursued two lines of inquiry.

- (1) To leverage existing work on designing performance measures for energy forecasts, we conducted a thorough survey of the academic literature on both solar and wind forecasting. The latter is a more mature field, which provided several insights (detailed below). We assessed that largely, the current state-of-the-art in both fields with regards to evaluating performance is to use some variant of square or absolute error. We did however find a small, but not insignificant, body of work in wind forecasting that quantifies economic costs of mispredictions to generators that bid in the energy market. This is done by designing a simulator of the energy market, an issue that we shall return to subsequently.
- (2) To better understand the needs of market participants, we had discussions with two key participants – the market operator, AEMO, and a distributor, ActewAGL – to understand the value they obtain from forecasting. Both participants indicated that they used either absolute or square error to measure the quality of their internal forecasts, but acknowledged the limitations of these measures.

Based on these inquiries, we identified several key issues that affect the various market players.

- *Security of supply.* A critical responsibility of the market operator (AEMO) is ensuring stability of the overall energy network, and in particular ensuring that sufficient energy can be dispatched to meet supply. Generally, energy may be dispatched from the various generators bidding in the market.

However, in periods of anomalously low supply (or high demand), the market operator needs to schedule energy from alternative sources, such as emergency generators. These sources have startup and operational costs. Thus, being able to predict such events beforehand is of interest. According to this criterion, a *bad* forecaster is one that either does not warn the operator of an impending ramp (or spike) in demand, or one whose predictions significantly depart from reality.

We can thus recommend as concrete measures:

- the *maximal deviation* between the forecast and observed output over some time-horizon. This pessimistic measure is more stringent than square error, as it implicitly considers a single bad prediction to outweigh several good predictions. (In wind forecasting, this measure has been recommended previously for system operators; see e.g. [4].)
- *imbalanced* and *cost-sensitive* measures, where one treats the existence of a ramp in PV supply as a binary variable that is to be detected by the forecaster. For example, one can compute the *balanced error* of a forecaster, which measures its ability to detect ramps in supply, while also not triggering many false alarms. (In wind forecasting, such measures have been recommended previously; see e.g. [3].)
- *Distribution reserve capacity.* Distributors (such as ActewAGL) primarily use forecasts of demand to ensure there is sufficient capacity to meet consumer demand. If peak demand exceeds supply, then procedures such as load shedding may be necessary. Of particular interest is the occurrence of upward ramps in demand (or downward ramps in supply). Therefore, as for the system operators, we can recommend as concrete measures the imbalanced and cost-sensitive measures for detection of ramp events.
- *Profitable bidding strategies.* Generators participate in the market by placing bids at regular intervals. The generators' primary interest is in maximising income from participation in the market. Achieving this requires a strategy for placing bids, e.g. deciding the appropriate amount to bid, based on expected future supply, and the current market price.

The literature on wind forecasting has established how probabilistic forecasts may be used to devise profitable bidding strategies. We can thus recommend as concrete measures:



- the dollar income resulting from optimal bidding using a probabilistic forecast, using a model of the energy market. Even when such a market simulator is highly simplified, it can be more informative than using more naïve measures such as square error.
- *Household energy bills.* Our discussion thus far has focussed on the aggregate PV supply from the *entire* distributed network of panels. However, for an individual consumer, the aggregate supply is likely not of immediate interest; rather, the interest is likely to be in reducing household energy bills, and possibly maintaining a low carbon footprint.

For the average consumer, we recommend square or absolute error as an adequate performance measure. This tells them how much energy can expect to use from the grid. Each unit of absolute error in prediction has a linear relationship with the cost as seen by the consumer in their energy bill.

Table 7.1 summarises the recommended performance measures for various market participants.

Participant	Value of forecast	Recommended measures
Market operator	Ensure security of supply	Maximal deviation
		Balanced error from ramp detection
Distributor	Ensure system reliability	Balanced error from ramp detection
Generator	Maximise return from bid	Dollar income (from simulator)
Household	Lower energy bills	Square error (for own panel output)
	Lower carbon footprint	

Table 7.1: Summary of recommended performance measures for various market participants.

To illustrate the viability of these measures in reflecting participants' underlying utilities, we conducted several simulations. We summarise some experimental findings below.

Example where square error does not reflect utility. The root mean square error (RMSE) of a forecast is an intuitive, easy to compute measure of performance. However, it may not always reflect the actual efficacy of a forecaster. We will now illustrate this in the context of detecting ramp events. We now consider the scenario where one has to forecast the PV output for a particular site on a day containing spikes in the output, which we shall consider to be ramp events. We show how square error will result in a potentially misleading statement about the relative efficacy of forecasts.

We compare two forecasts of PV output for a 24 hour period on a single site, on a particular day of the year. Our first forecast is derived as follows: we consider each of the other 364 day-long series for the same site, and compute the RMSE against the series for the day in question. We then find the series with the lowest RMSE, and consider that as being our forecast. Our second forecast is derived as follows: we consider the time series for a different day, derived using a procedure that we shall detail shortly. We shall call the resulting time series A and B, respectively.

Figure 5 illustrates the results of this simple procedure. It is visually apparent that the “forecast” time series A has a small squared error compared to the reference time series, much more so than the forecast time series B. Quantitatively:

$$\begin{aligned} \text{Forecast A RMSE} &= 0.8997 \\ \text{Forecast B RMSE} &= 1.5897 \end{aligned}$$

This is a close to two-fold difference in performance. Judged solely by RMSE, then, the choice between these forecasts is clear: even if one were to allow for some fluctuations in error due to randomness, it seems incontrovertible that forecast A is to be preferred.

However, a closer examination of the forecasts reveals a different picture. Recall that the reference time series is characterised by a number of spikes in output. Forecast A, despite being a good overall fit to the

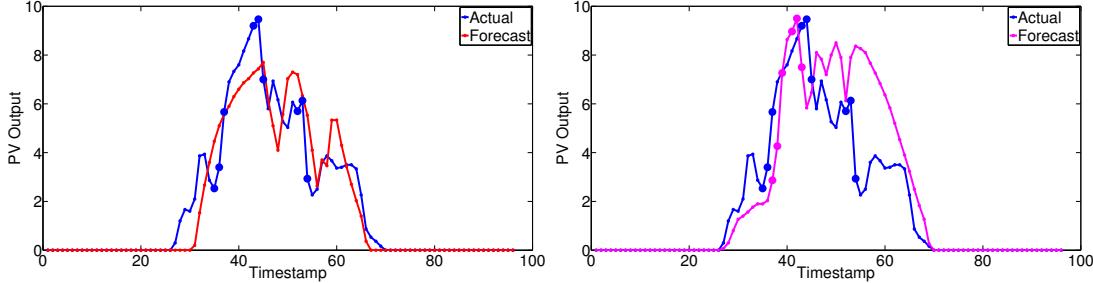


Figure 7.1: PV output forecast time series A and B.

reference series, does not possess such spikes. Indeed, to capture the sudden surge in output between timestamps 25 and 40, the forecast series takes the approach of gradually increasing the output till it reaches a level commensurate with that of the reference series. A similar trend is observed for the downward spikes. But this means that if planning is based on this forecast, the network operators will be unprepared for surges on the day in question. In contrast, forecast B is much better at capturing the ramps: in particular, it correctly identifies the existence of an up-ramp around timestamp 30, and a down-ramp around timestamp 40. Subsequently, however, it fails to capture the final down-ramp, and consequently incurs a heavy penalty in terms of RMSE.

In fact, we can now reveal that the procedure for selecting forecast B was minimising the balanced error in terms of ramp detection. For a reference energy signal y over a period of T time steps, and prediction \hat{y} , this measure is computed as

$$\text{BER}(y, \hat{y}) = \frac{\sum_{t \in [T]} \mathbf{1}[y_t - y_{t-1} > \tau, \hat{y}_t - \hat{y}_{t-1} < \tau]}{2 \sum_{t \in [T]} \mathbf{1}[y_t - y_{t-1} > \tau]} + \frac{\sum_{t \in [T]} \mathbf{1}[y_t - y_{t-1} < \tau, \hat{y}_t - \hat{y}_{t-1} > \tau]}{2 \sum_{t \in [T]} \mathbf{1}[y_t - y_{t-1} < \tau]}$$

where τ is a predefined threshold on difference in energy which indicates the occurrence of a ramp, and $\mathbf{1}[\cdot]$ is the indicator function. Essentially, the measure assesses whether one predicts a ramp when there is one; and does not predict a ramp when there isn't one. In terms of this measure, we see that

$$\begin{aligned} \text{Forecast A Balanced Error} &= 0.5000 \\ \text{Forecast B Balanced Error} &= 0.3960 \end{aligned}$$

indicating forecast B is to be preferred. This illustrates how such measures can lead to selection of more useful forecasts, from the perspective of underlying utility (in this case, ensuring security of supply.)

Relationship between numerical and dollar value penalty. We now illustrate how one can estimate dollar revenues (or losses) for a generator who bids in the energy market. Following [5], we consider a highly simplified, but we believe nonetheless illustrative, model of the energy market: we assume that all bids are accepted, and that the income awarded to a bidder who promises generation of \hat{y} energy units, but delivers y energy units, is

$$\text{Income}(\hat{y}, y) = p \cdot \hat{y} + p_{\text{down}} \cdot (y - \hat{y}) - (p_{\text{down}} - p_{\text{up}}) \cdot \max(0, y - \hat{y})$$

where p is the spot-price, and $p_{\text{down}}, p_{\text{up}}$ are the down- and up-regulation prices. Clearly, the maximal achievable income is $p \cdot y$. We can define the “lost income” to be the difference between this maximal income, and the achieved income.

Observe that in this setup, over- and under-predictions are costed differently. If $y > \hat{y}$ – that is, the bidder under predicts their generation capacity – then, the “lost income” is $(p - p_{\text{up}}) \cdot (y - \hat{y})$. By contrast, if $y < \hat{y}$ – that is, the bidder over predicts their generation capacity – then, the “lost income” is $-(p_{\text{down}} - p) \cdot (y - \hat{y})$. Thus, when $p \neq \frac{p_{\text{up}} + p_{\text{down}}}{2}$, the lost income in both cases is different.

To illustrate the asymmetry in over- and under-prediction, we consider a scenario where we have two forecasts of PV output. Both forecasts achieve the *same absolute error*, but dramatically different incomes. The example proceeds as follows: let y be the generated PV from a particular bidder over a day, and let \hat{y} be the forecast PV for the same period. Now consider a second forecaster who predicts $2y - \hat{y}$. Then, the absolute errors of the two forecasters are trivially identical.



On the other hand, their resulting incomes are very different. Figure 7.2 plots, at each time period the income over the preceding 5-min time window for the two forecasters, on the same example site as the previous subsection. For this plot, we choose $p = \$35$, being the average annual spot price in NSW for 2014 – 2015; for simplicity, we choose $p_{\text{down}} = 2p$, and $p_{\text{up}} = \frac{p}{4}$. We find that, over the entire day,

$$\begin{aligned}\text{Forecast A Income} &= \$8177.75 \\ \text{Forecast B Income} &= \$4676.00\end{aligned}$$

That is, while the forecasts have the *same absolute error*, the first one is *nearly twice as profitable!*

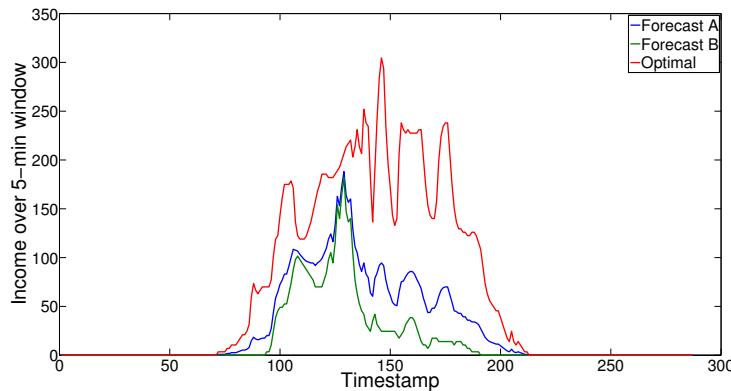


Figure 7.2: Incomes of two forecasts with the same absolute error, over a single day. Also shown is the income from an optimal forecast.

7.3.2 Adapting ML methods to different measures of performance

To address the second objective, we conducted simulations to investigate the feasibility of simple extensions to the learning methods considered into project, so as to endow them with favourable performance according to the various measures listed above. (As these measures paint a different picture of performance – as we also demonstrated in our simulations – we could not devise one single extension for all the measures.) We focussed on two major directions:

- (1) The major focus was in the generation of probabilistic predictions. These allow generators to devise sophisticated bidding strategies. Additionally, they allow other decision makers to take into account their risk aversion when taking actions.

On this front, we demonstrated via simulations how to use the *bootstrap* to endow forecasters with measures of uncertainty. This approach has the benefit of being mathematically simple to implement, and not requiring modifications to the underlying learning algorithm. This is in contrast to other sophisticated techniques, such as Bayesian forecasting methods, which typically require fundamental changes to the way the forecaster is trained.

- (2) Our other focus was the modification of standard learning objectives to adapt to different performance measures. For example, we demonstrated how to make forecasters optimise the expected return to a bidder in the energy market. We illustrated how this can result in selecting forecasts that are significantly better – in terms of returns – than using a naïve measure like square error, which results in predicting the expected PV output (under the uncertainty distribution).

We summarise some experimental findings below.

Probabilistic forecasts with the bootstrap. In our first experiment, we compare three methods: linear regression, Bayesian linear regression, and the bootstrap. The latter two methods produce probability distributions, rather than a single point forecast. We train all methods on a single day's worth of data, and evaluate them on a (separate) single day's worth of data. We evaluate all methods based on the RMSE against the testing day's data, which measures the deterministic (mean) output's quality.



We also measure probabilistic performance by assessing 95% coverage, 95% sharpness, and the average bias. These measures are defined as follows. Suppose a probabilistic forecaster outputs a predicted density \hat{p}_t over the target variable at time t . The *quantiles* of the predicted distribution, for any $\alpha \in [0, 1]$ are

$$q_t(\alpha) = (F_t)^{-1}(\alpha),$$

where F_t denotes the cumulative distribution function (CDF) derived from \hat{p}_t . Another important quantity is the $100 \cdot (1 - \alpha)\%$ confidence interval of the mean,

$$C_t(\alpha) = [a, b],$$

where $\mathbb{E}_{Y \sim \hat{p}_t}[Y] \in [a, b]$, and $F_t(b) - F_t(a) = 1 - \alpha$. Then,

$$\begin{aligned}\text{Bias} &= \alpha - \frac{1}{T} \sum_{t \in [T]} \mathbf{1}[y_t < q_t(\alpha)] \\ \text{Sharpness} &= \frac{1}{T} \sum_{t \in [T]} (q_t(1 - \alpha) - q_t(\alpha)) \\ \text{Coverage} &= \frac{1}{T} \sum_{t \in [T]} \mathbf{1}[y_t \in C_t(\alpha)].\end{aligned}$$

Intuitively, these measures assess whether or not the predicted distribution is not overly vague (assigning equal probability to all possible energy outputs), while also being sufficiently concentrated around the observed value (not being overly confident about incorrect values).

Table 7.2 summarises the performance of the methods. As their mean predictions are the same (modulo the noise injected in the bootstrap), their RMSEs are essentially identical. However, as further illustration that RMSE is only part of the story, we see that the methods can be distinguished along other metrics that measure fidelity of the predicted distributions. For example, linear regression has the worst coverage of the three, which is unsurprising, given that it does not account for any uncertainty in the estimation of the conditional mean. Further, linear regression tends to be overly confident about these predictions, as reflected by its lower sharpness score. (A sharpness of zero indicates a point-mass, perfectly confident distribution.) This is also unsurprising, given that the bootstrap and Bayesian linear regression additionally report uncertainty due to the estimation of the conditional mean.

Focussing on the bootstrap and Bayesian linear regression, there is little to choose amongst the two methods. The bootstrap's predictions are slightly worse in terms of sharpness, but are slightly better in terms of coverage and bias. The bootstrap's relative simplicity of implementation, however, may make it an appealing solution, especially when employing more complicated forecasting methods.

Method	RMSE	Coverage	Bias	Sharpness
Linear regression	1.0688	0.7465	0.1292	0.7748
Bayesian linear regression	1.0688	0.8229	0.1277	1.1052
Bootstrapped linear regression	1.0749	0.8611	0.1203	1.1322

Table 7.2: Performance of probabilistic forecasts when trained on single day.

The above suggests that RMSE provides only one perspective on prediction. It also suggests that the bootstrap is a sensible and simple approach, if one's interest is in attaining good 95% prediction intervals (in the sense of being sharp and attaining good coverage). Strictly more information is present in the actual graphs of the predictive distributions, which we summarise in Figure 7.3. The shaded regions around the predicted time series represent the 95%, 75%, 50% and 25% predictive intervals, estimated using the appropriate predictive uncertainty. The bootstrap appears to quite successfully mimic the predictive distribution of Bayesian linear regression. (We do observe slightly less concentrated predictions, as reflected in the bootstrap's worse sharpness score.) These visualisations confirm also that the measures of probabilistic performance match our intuition as to the quality of the produced probability distributions.

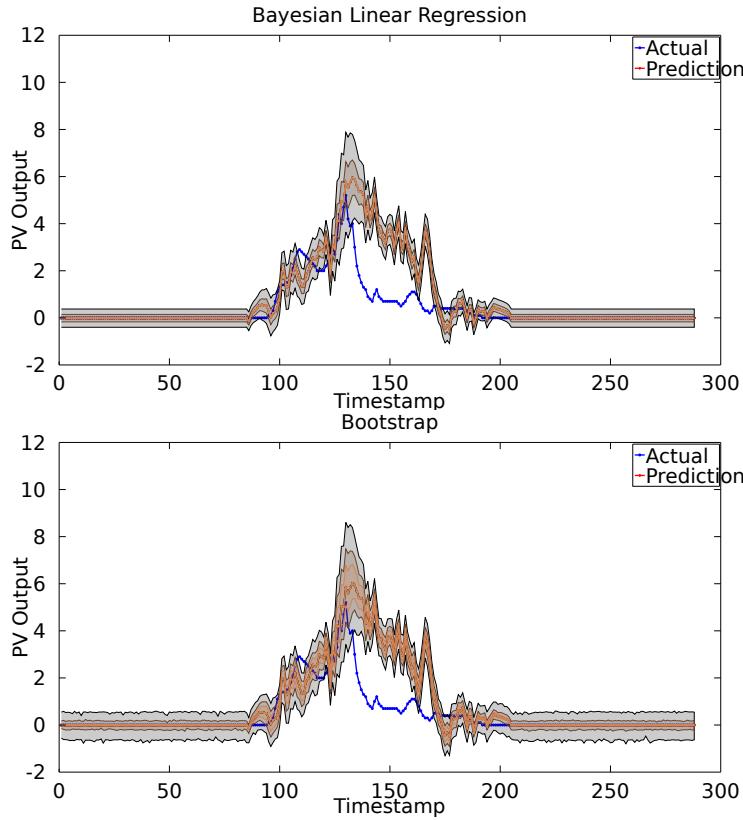


Figure 7.3: Performance of probabilistic forecasts when trained on single day.

Adapting learning methods to various measures. We illustrate how one can tune forecasts to directly optimise a market participant’s utility. We do so by considering probabilistic forecasts of PV output, and showing how they may be used to design profitable bidding strategies for a generator, as per [5].

We consider the same simplified energy market scenario as earlier. We compare the income received by two forecasters. The first forecaster simply predicts, for each time of day, the average PV output observed at that time of day, over all training days. Recall that this is the optimal forecast if one prescribes squared error as the performance measure. The second forecaster instead constructs the *distribution* of PV outputs observed at that time of day. Then, to make a single prediction, the forecaster chooses to *maximise the expected return*,

$$\text{ExpReturn}(\hat{y}) = \sum_y \mathbb{P}(y) \cdot \text{Income}(\hat{y}, y),$$

where $\mathbb{P}(y)$ denotes the forecasters’ (estimated) probability of generating PV output of level y at the current time of day.

As established earlier, the Income derived is an asymmetric function of the discrepancy between the predicted and observed PV output. Therefore, one can expect directly optimising for this to yield improved returns. Figure 7.4 confirms this is so for the majority of timestamps. We also see that, in general, the gains from bidding based on probabilistic forecasts are to be had when there is high (estimated) variance in the PV output. Over the entire day, we find that,

Forecast A Income = \$15389.50

Forecast B Income = \$14511.30

so that *probabilistic forecasts can yield significant revenue increases over naïve mean predictions.*

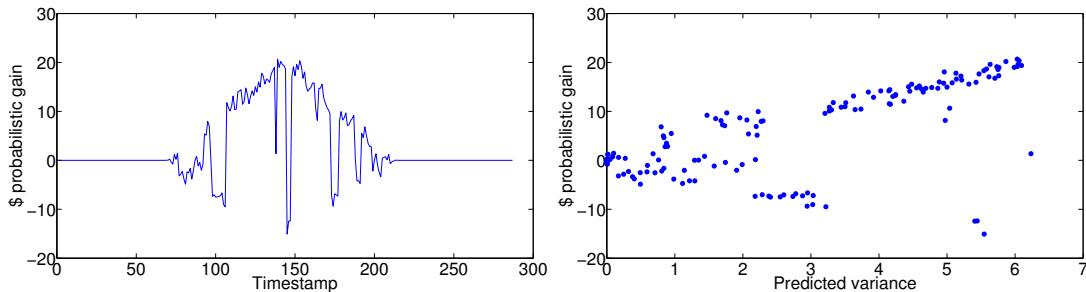


Figure 7.4: Dollar gain from using probabilistic forecasts over expected PV output when bidding.

7.4 Extent to which the objectives were achieved

The outcomes for this component indicate how decision making can be refined by suitable choice of performance measures, and how such objectives can be incorporated into the process of training a forecaster. For the first objective, we showed how specific performance measures, such as those that measure detection of ramp events, can be used to aid decision making. For the second objective, we illustrated how to optimise forecasts for bidding by estimating the uncertainty in predictions.

However, our outcomes do fall short of the Agreement Objectives outlined previously. For the first objective, our analysis and results, while illustrative, were not refined to the extent that they demonstrated *operational* value to market participants. It would be ideal, for example, to have demonstrated that our proposed measures of generator utility correlated strongly with the actual gains observed by a generator in the course of market bidding. (We shall discuss in the next section the difficulties we encountered in refining our analysis to this extent.)

As for the second objective, most studies in the Chapters 5 and 6 were based on root mean squared error, with mean absolute error used sporadically in Section 5.3.1. This is because the objective therein was to select the important and predictive features for the forecasting model, and to construct effective and efficient learning algorithms with improved accuracy. In many cases, the conclusions related to feature selection are relatively stable under various performance measures, and learning with squared error is most favorable because of its computational efficiency (about an order of magnitude faster than using absolute error). However we did observe that, fixing the features, the learning algorithm needs to be customised with the performance measure taken into account. For example, using the joint predictor on the 253 site PV data with all parameters tuned to optimum, Table 7.3 shows that training with absolute error as in equation (5.1) (with $\epsilon = 0$) leads to lower mean absolute error than using squared error.

	Horizon = 10 minutes	Horizon = 30 minutes	Horizon = 60 minutes
Joint predictor with squared error	0.039	0.055	0.066
Joint predictor with absolute error	0.035	0.052	0.063

Table 7.3: Comparing the mean absolute error of joint predictor based on squared and absolute error.

The practical implication is that, given the significant advantage of using joint prediction and image feature as demonstrated in the context of squared error, the benefit of machine learning can be extended to a broader range of performance measures, insofar as the training objective is reformulated based on the measure.

We emphasise that one must consider the achieved outcomes in context of the state-of-the-art in both industry and academia. In industry, the current state-of-the-art (based on our discussions with several market participants, discussed earlier) is to produce for each future time step a single predicted PV value (a *point* forecast), which is evaluated by computing either the absolute or square error of this point forecast to the observed value. While not entirely unreasonable, as our simulations illustrate, this approach offers limited decision making capabilities. Similarly, in academia, the current state-of-the-art (based on our extensive literature review) in solar forecasting is to use a point forecast, with some variants on the standard absolute or square error to measure performance. In the field of solar forecasting, our progress thus represents a good first step.



7.5 Highlights, difficulties and breakthroughs

There are three main highlights from our work in this project component.

- (1) First, we leveraged insights from the (comparatively) mature field of wind forecasting with regards to the design of suitable performance measures. Many of the issues we have discussed above – the ability to use uncertainty in predictions to devise profitable bidding strategies, the need to design ramp-sensitive measures, and so on – have their genesis in the wind forecasting literature, but have been relatively under-explored in the solar forecasting community. Our simulations have shown that, unsurprisingly, many of these ideas do carry over and are valuable from the perspective of more closely aligning with a participants' utility.
- (2) Second, we illustrated the mismatch between standard performance measures, such as square error, and our model participants' utility. For example, we quantified the dollar cost from mispredictions to generators bidding in the market, and contrasted this to the square error of predictions. This highlights the potential perils of using naïve performance measures to make decisions, e.g. changing bidding strategies based on square error rather than dollar cost.
- (3) Third, we demonstrated the viability of a simple technique, the bootstrap, in generating uncertainty estimates. The ease of implementation of this technique reduces the barrier of adoption for implementation in industrial systems.

We faced three main difficulties in conducting the research for this component.

- (1) First, the utilities for each of the market participants are very different, which required separate analysis and research as to each of their operations and incentives in the market. As detailed earlier, there is no single “right” performance measure that satisfies all, or even most market participants. Indeed, the notions of utility for different entities in an energy market may in fact be in competition with each other [1].
- (2) Second, the utilities for each of the market participants is non-trivial to elicit. In our discussions with ActewAGL and AEMO, we discovered that there was limited internal assignment of costs to errors in their forecasting tools. Further, there was no systematic understanding of the range of possible costs that could be incurred from a misprediction. We did not find much relevant academic literature on the problem of eliciting utility from the consumer of a forecast; indeed, most works seek to optimise some assumed measure of utility, with little discussion on how this is derived.
- (3) Third, even with some understanding of costs involved in mispredictions, a holistic picture of the aggregate effect of a misprediction requires a simulator of the energy market. This is a separate undertaking in itself, and requires detailed study and care to be realistic. Some simulators for the Australian market have recently been developed [2]; leveraging these for future studies seems an important line of inquiry.

7.6 Conclusions and recommendations

We have described the outcomes of our analysis on the design of suitable performance measures for PV output forecasts. Our simulations and analysis indicate that there is considerable potential benefit from going beyond naïve measures of performance, such as the square error of predicted and observed forecast. Further progress along these lines would represent a significant aid in the decision-making capabilities of various participants in the energy market.

To build on our efforts, we make the following specific recommendations:

- Design a realistic market simulator, and use this to get a more fine-grained understanding of the economic impact of mispredictions on e.g. energy generators.
- Work with generators to assess the viability of using probabilistic forecasts to improve bidding strategies.
- Compare performance of industrially deployed forecasters based on proposed measures, and match against performance according to traditional measures e.g. square error.



Bibliography

- [1] R. J. Bessa, V. Miranda, A. Botterud, and J. Wang. “Good” or “bad” wind power forecasts: a relative concept. *Wind Energy*, 14(5):625–636, 2011.
- [2] Ben Elliston, Mark Diesendorf, and Iain MacGill. Simulations of scenarios with 100% renewable electricity in the Australian National Electricity Market. *Energy Policy*, 45(0):606 – 613, 2012.
- [3] C. Ferreira, J. Gama, L. Matias, A. Botterud, and J. Wang. A survey on wind power ramp forecasting. Technical Report ANL/DIS-10-13, Argonne National Laboratory, 2011. <http://www.dis.anl.gov/pubs/69166.pdf>.
- [4] UCTE NetWork of experts on Wind Power. Wind power in the ucte interconnected system. https://www.entsoe.eu/fileadmin/user_upload/_library/publications/ce/otherreports/Wind_Power_20041125.pdf, 2004. UCTE Position Paper.
- [5] Julio Usaola and Jorge Luis Angarita. Bidding wind energy under uncertainty. In *International Conference on Clean Electrical Power, 2007 (ICCEP)*, pages 754–759, 2007.



Part III

July 2015 - June 2016



Chapter 8

Installation of 100 data loggers in the ACT region

8.1 Overview

This chapter summarises the outcomes of the project concerned with installing 100 data loggers in residential homes. It also describes the challenges encountered, and provides recommendations regarding similar undertakings in the future.

8.2 Objectives

The objectives of this module as defined in the Project Plan (Deed of Variation 2) were:

- Installation of 100 data loggers in the ACT region.

In this Chapter, we discuss:

1. The necessary approval procedures before installing equipment into power meter boxes of residential homes.
2. How we approached the task to find and select volunteers who receive a data logger.
3. Challenges faced at installation time.

8.3 Approval Procedures: Human Ethics, ActewAGL, Telstra

In the next subsection, we describe formal approval processes necessary for the installation of data loggers in residential homes. This involve not only a “Human Ethics application” as we interact with members of the public, but also technical approvals to install data loggers in power meter boxes (ActewAGL), and permissions to use Telstra’s wireless machine to machine (M2M) infrastructure with our equipment.

8.3.1 Human Ethics application

Before interacting with the public by installing data loggers in residential homes, an application with the “Human Research Ethics Committee” of the ANU had to be filed for approval. This application contained a detailed description of the planned data logger installations, all project members involved and samples of the following documents intended for each home owner potentially participating in the data collection.

- **Project Description** outlining the scope and purpose of the data collection and the goal of the project.



- **Participant Questionnaire** asking each participant for technical details of the PV system installed in the residential home.
- **Participant Information Sheet** formalises all aspects of participating in this project as a volunteer.
 - Project Title and Brief Description and Methodology
 - Incentives, financial compensation, and financial costs of participation
 - Risks
 - Liability
 - Participant Involvement (Voluntary Participation & Withdrawal, What will participants have to do?, Location and Duration, Data to be collected, Implications of Participation)
 - General Outline of the Project (Participants, Use of Data and Feedback, Project Funding)
 - Confidentiality
 - Data Storage
 - Intellectual Property rights
 - Lead Researchers
 - Queries and Concerns (Contact Details for More Information)
 - Ethics Committee Clearance
- **Participant Consent Form** which has to be signed by each volunteer receiving a data logger. Not knowing the reaction of the home owners with regard to data disclosure to the public, we prepared four options:
 1. Full disclosure of location to the public.
 2. Disclosure of location masked with noise of up to 100m.
 3. Masked location and dividing the rated output of the system by a factor effectively hiding the real size of the PV system.
 4. No disclosure outside of the research team.

8.3.2 ActewAGL

Installation of equipment into power meter boxes follows strict regulations. If representatives of the power company who read the power consumption/production in the power meter box find foreign equipment, they are encouraged to remove this equipment because it has the potential (intended or unintended) to interfere with the correct energy measurements. For this reason, we contacted the local utility provider, ActewAGL, and established a protocol to inform them about all our data logger installations. This serves to protect our installed equipment from the accidental removal by a ActewAGL representative reading the energy consumption/production.

8.3.3 Telstra

Our data loggers use wireless 3G/4G data connections via Telstra's M2M network to send data in intervals of 5 minutes to a central data collection server. Research equipment like our data loggers are not usually sold in large quantities like mobile phones, but custom-build and therefore need special approval to be connected to a public wireless network. We discussed this with our sponsor, Telstra, and received permission to connect our research equipment to their network.



8.4 Finding and Selecting Residential Homes

8.4.1 Finding volunteers to receive a data logger

Early in the project, we setup a web page asking for volunteers to sign-up to receive a data logger. In about six months, this led to about 20 residential home owners signalling their interest in the project, resulting in 12 to finally agree to an installation.

As this was quite unsatisfactory, we asked NICTA's and ANU's Press Offices for help. The resulting press release was published in many daily newspapers, weekly magazines and electronic media in Canberra. The highest impact however came from an interview with ABC Radio Canberra in the "Breakfast with Philip Clark" show at prime time 8.40 am, 17 November, 2015. We subsequently learned that many people listen to this show while driving to work. In a few days, we had hundreds of volunteers, after three weeks we reached 400.

This now presented a new problem: How to choose the best locations for 100 data loggers out of the 400 volunteers? In the following, we describe the set of criteria we used to optimise the selection of locations.

8.4.2 Selection criteria for the optimal data logger locations

8.4.2.1 Locations outside of the ACT

We had a few volunteers from outside of the ACT, two from Queanbeyan, and each one from Murrumbateman, Wamboin and Hall (NSW). Data loggers in those locations outside of the ACT provide important early signals for changing weather and cloud patterns in the directions from North to Southeast. We therefore decided to include them.

8.4.2.2 Coverage of the ACT, outwards boundaries of suburbs

The next goal was to achieve a good coverage of the ACT by having data loggers in each suburb of the territory. If a suburb shared a border with the ACT border, we preferred data logger locations close to the ACT border. The goal is again to receive a signal as early as possible when clouds enter the ACT via that border.

Technical challenges arise from the fact that those locations on the ACT boundary are often far away from a wireless connection point. In many cases, multiple visits to the location were necessary in order to optimise the antenna position for the wireless transmission based on data recorded over the previous days.

8.4.2.3 Availability of PVoutput.org data feed

Some home owners send their data from the inverter directly to a data collection site like PVoutput.org. We installed a few data loggers in these locations in order to compare the output of our system with the reported data.

8.4.2.4 Preferred Disclosure options

We tried to convince home owners who had chosen restrictive data disclosure options (2. masked location; 3. masked location and masked actual rating of the system; 4. no public disclosure) to allow for a less restrictive choice for the public disclosure of their data. If they insisted on a restrictive disclosure option, we only included them when data from their location was deemed to be very valuable for research.

8.4.2.5 Availability of residual-current devices (RCD)

A residual-current device (RCD), or residual-current circuit breaker (RCCB), is a device to quickly disconnect current to prevent serious harm from an ongoing electric shock. When installing a power point into the power meter box of a residential home, it is mandatory to also install a RCD if the line is not already secured with an RCD. The cost for an RCD alone is \$60, but in some cases a more extensive upgrade resulting in considerable extra cost would have been necessary. In general, we therefore tried to avoid houses which did not yet have RCD devices installed.



Disclosure option	data loggers
Full disclosure of location to the public	67
Disclosure of location masked with noise of up to 100m	30
Masked location and dividing the rated output of the system by a factor effectively hiding the real size of the PV system	3
No disclosure outside of the research team	0

Table 8.1: Number of data loggers for each disclosure option

8.4.2.6 New data loggers with additional voltage measurements

It turned out that an additional voltage measurement in our distributed system does not only increase the precision to which the power can be estimated, but is also of great interest to the utility companies. Given the full coverage of the ACT by our data loggers, we decided to upgrade as many data loggers as possible and add a voltage measurement. This was achieved without incurring extra costs. About 50% of the data loggers are now also reporting the voltage in addition to the current. The locations for the improved model were chosen with the goal of having a wide coverage of the ACT.

8.5 Inspection of candidate locations and interviews with home owners

The selection of candidate locations turned out to be a very time consuming process. Our team member, Dr Frank Mills, ANU,

- contacted hundreds of home owners,
- visited many of their homes and inspected the power meter box,
- talked to the home owners explaining the project goal and the documents from the Human Ethics Approval process, and
- finally – if they were still interested in participating – recorded their consent and their public disclosure option.

8.6 Installation

The installation of each data logger has to be performed by a registered electrician. Although the data logger itself runs on low voltage DC, both the installation of a power point into the power meter box and the installation of the contactless current clamp around one of the hot wires is only allowed to be done by professional electricians. Any changes to the power meter box have also to be reported to an ACT agency and a fee must be paid.

8.7 Highlights

As the data logger installation involved interaction with the public and exploration of many PV systems not available at the start of the project, design decisions and technical setups were very much influenced by the specific situation with regard to wireless connectivity, available space in the power meter boxes, and willingness of the home owners to fully disclose their location/PV system information.

The overwhelming response of the ACT public resulting in 400 applications to receive a data logger has been very encouraging. It shows a great interest in quantifying/measuring the performance of PV systems. The resulting effort to find optimal locations for the data logger placement was much higher than anticipated when the project was initially established.



suburb	count	suburb	count	suburb	count	suburb	count
Ainslie	4	Dickson	2	Hawker	2	Ngunnawal	2
Amaroo	1	Downer	4	Holt	2	Nicholls	3
Banks	2	Duffy	2	Hughes	1	North	1
Bonner	2	Dunlop	1	Isaacs	1	O'Connor	2
Bonython	1	Evatt	1	Isabella	2	Phillip	1
Calwell	1	Fadden	1	Kaleen	4	Queanbeyan	1
Campbell	1	Farrer	2	Kambah	7	Red	1
Casey	1	Florey	2	Karabar	1	Spence	1
Chapman	1	Forde	1	Kingston	1	Torrens	1
Chifley	1	Forrest	1	Latham	1	Wallaroo	1
Chisholm	1	Franklin	1	Lyneham	1	Wamboin	1
Conder	1	Fraser	2	Macgregor	2	Wanniassa	1
Cook	1	Garran	1	Macquarie	1	Watson	1
Coombs	1	Giralang	3	Melba	1	Weetangera	1
Curtin	3	Gordon	3	Monash	1	Weston	1
Deakin	1	Gowrie	1	Murrumbateman	1	Yarralumla	1

Table 8.2: Number of data loggers installed in 64 suburbs.

8.8 Conclusions

Measuring the performance of PV systems in short intervals does serve many purposes:

- monitoring the health of the PV system (degradation, soiling),
- comparison of different systems,
- forecasting PV performance,
- observing the interaction of PV systems with the electricity network, and
- integration of distributed generation, storage and load control.

It is therefore very desirable to replace the current ad hoc (and often incompatible) solutions with a more principled approach by developing standards for data collection and data communication.

The project largely underestimated the difficulties in installing 100 data loggers in residential homes. This led to several delays and did bind personal resources which were intended for the technical support and scientific exploration. For future projects, it is recommended to partner with a larger installation company to get easier access to data from residential homes.

On the positive side, we have now a data collection system in place which receives PV system data from all over the ACT. This is valuable data, and there is a great interest to continue to collect and use this data. We are currently discussing with Dr Nick Engerer how to utilise the ongoing data collection in the ARENA funded project “Real-time Operational PV Simulations for Distribution Network Service Providers”.





Chapter 9

Collection, Analysis and Publication of Data

9.1 Overview

This chapter presents an overview of the data collection, data analysis, and the distribution of the data to the interested public. At inception, we did not know which data streams would provide us sufficient information. Therefore we decided to collect visual data (cloud images), data logger data (power and voltage output), and additionally weather data from an ANU weather station and the Bureau of Meteorology (BOM).

9.2 Objectives

The objective of this module is

- to describe the technical setup of the data collection system.
- to provide reasons why the arrangement of servers was chosen in this specific way.
- to provide an analysis of the data received from the data loggers.
- to show how the public can access the collected data.

9.3 Overview of data collection

Data are collected from several sensors in varying intervals. When the recording interval is smaller than the transmission interval, data are locally accumulated and sent as a batch.

- 24 hours interval: data from the Bureau of Meteorology are collected for weather stations around Canberra (recording interval: 30 minutes)
- 6 hours interval: the weather station at the ANU sends direct radiation data, global radiation data, wind data and temperature data (recording interval: 1 minute)
- 5 minute interval: each of the 100 data loggers sends scalar values representing the current from the inverter of the PV system measured with a contactless current clamp. Additionally, half of the data loggers also measure the voltage and therefore send a second scalar value.
- 1 minute interval: each sky camera sends a high resolution image (2592×1944 RGB)
- 10 seconds interval: each sky camera sends a high resolution image (648×486 RGB)

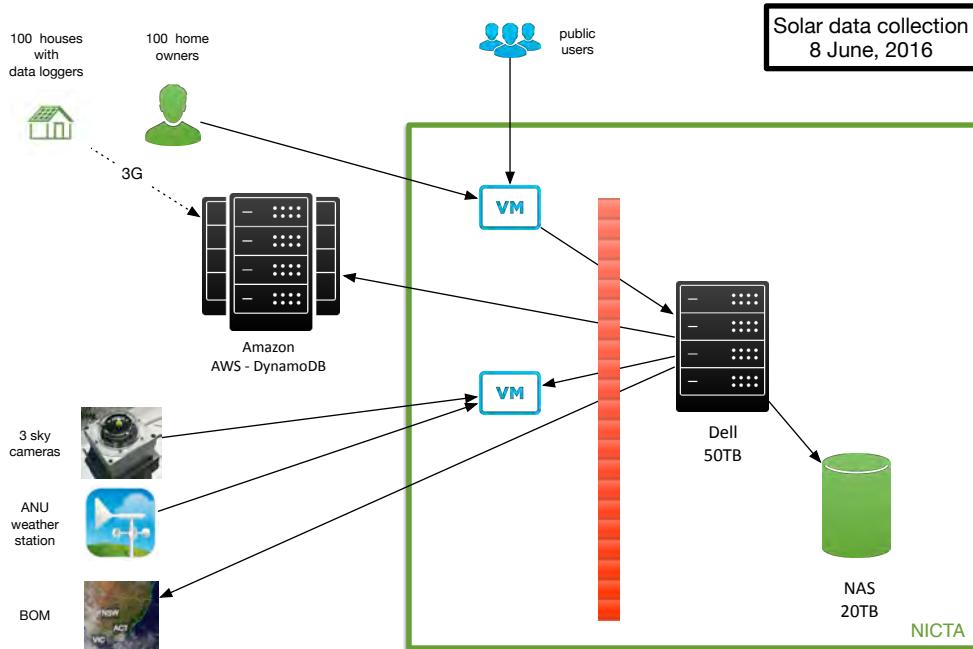


Figure 9.1: Data collection with two virtual machines (VM) and one rack mounted 50 TB Dell server with 20 TB extension.

For all sensor sources which actively send data, a virtual machine in the “demilitarised zone” (DMZ) has been set up. This machine operates outside of NICTA’s secured internal network and allows access from registered machines connected to the sensors. This push technology avoids the creation of special ‘holes’ in the firewall of the organisation hosting the sensors (like ANU). At the same time, it isolates the secure internal network where the data collection server is located. Data transfer from the VM is initiated from the data collection server inside of the secure network and therefore the firewall in NICTA does not need to be opened for external access.

9.4 Data Analysis of data logger data

9.4.1 Clear sky day

On a clear sky day, the power follows a trajectory which is defined by the relative angles of the sun and the orientation of the solar panels on the roof of a house. Power is reported as the average current sampled with high frequency over an interval of 5 minutes. This way, even sudden changes in the power are integrated into the average reported after 5 minutes (Figure 9.2).

9.4.2 Clear sky day with additional voltage measurements

Voltage measurements are reported in the same 5 minute interval as the power measurements. In contrast to the power measurements, the voltage is measured at one point only as integrating over a 5 minute interval would remove interesting spikes which otherwise may appear in the reported data (Figure 9.3).

The voltage follows a path typical for a residential area: It starts from a high value in the night because the load on the grid is minimal while most people are at sleep. It then falls to a minimum around 7 am when people switch on heatings and prepare for breakfast. It then increases until noon when input from the solar systems is at a maximum and many people are absent because they are at work. It then falls again with the reduction of sun light and the returned home owners using heating and other appliances like TV, computers, etc. The voltage then rises again in the late evening until it reaches the high value in the early morning of the next day.

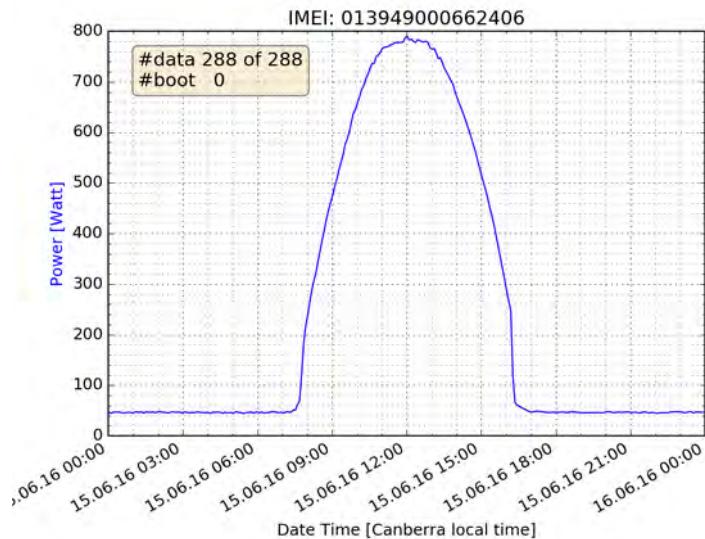


Figure 9.2: Typical power output reported for a clear sky day.

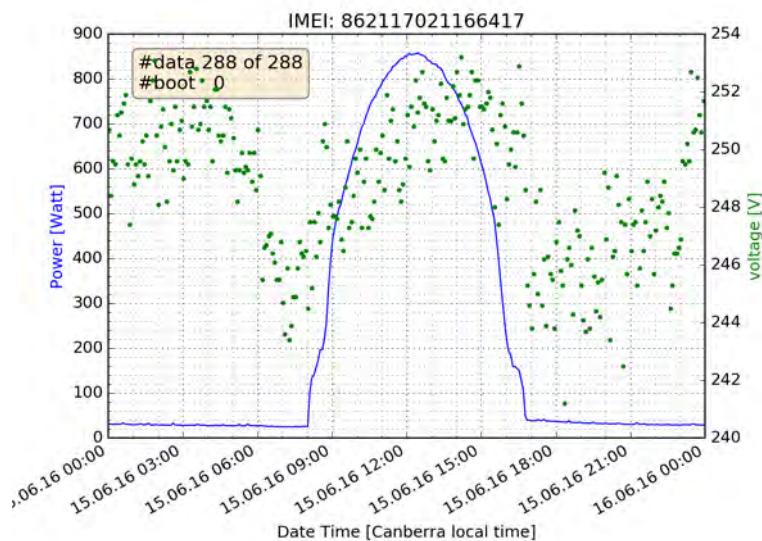


Figure 9.3: Typical power output and voltage reported for a clear sky day with the upgraded data loggers.

9.4.3 Impact of shadows on the power output

Shadows originating from obstacles close to the PV system like trees and high buildings modulate the power output and need to be taken into account when estimating the energy production of a PV system at a given time. In Figure 9.4 we show a typical example recorded on the same clear sky day as the Figures 9.2 and 9.3.

In the morning around 9 am, an obstacle prevents the sun from shining onto the PV system. Suddenly around both 10 am and 12 am, the power output increases dramatically in a very short time because the sun is no longer blocked from reaching the solar panel. Shortly after mid-day, the power output drops considerably again, albeit slower. And then at 4 pm, it suddenly drops to a very low level indicating that another obstacle is shadowing the panel.

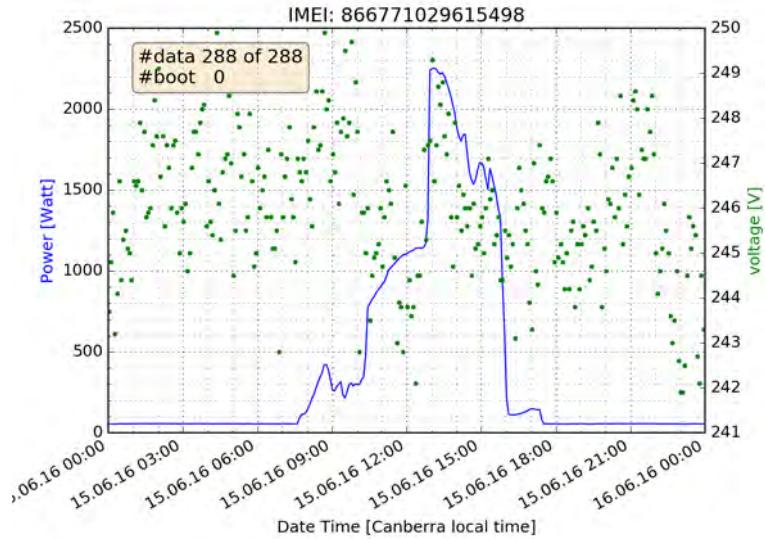


Figure 9.4: Typical power output for a location with shadows from neighbouring obstacles.

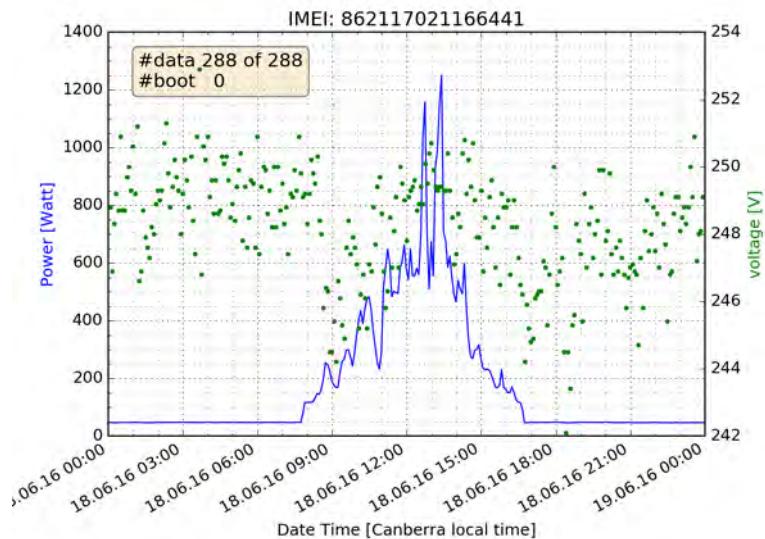


Figure 9.5: Typical power output for a day with many patches of clouds.

9.4.4 Cloudy day

The impact of the intermittency of clouds can be clearly seen in Figure 9.5 recorded on a cloudy day with patches of blue sky between the clouds. Sudden decreases/increases of the power by a factor of 2 represent the types of problems the power grid has to deal with when a large number of PV systems is installed in some area.

9.4.5 Irregular output

We detected a number of houses which showed an unexpected output as for example seen in Figure 9.6. The very untypical spikes in the curve together with a very low power output raised our suspicion. And indeed, about 6 month ago this home owner had repair done on a safety switch for the solar system. For unknown reasons, the safety switch was never turned on again. We had about a dozen cases where we detected problems

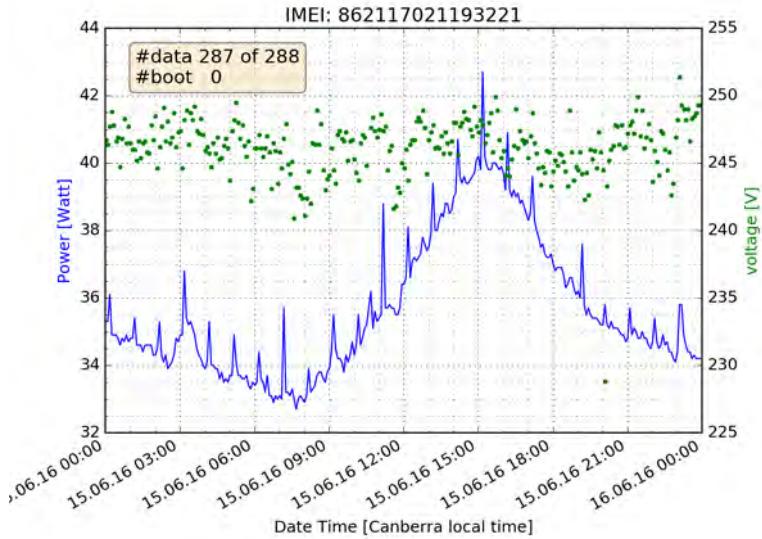


Figure 9.6: Example of power output for a switched-off PV system.

in the setup of the solar system or in the wiring information available to us. After contacting the home owners and the electrician who did the installation, these problems were rectified.

9.5 Publication of Data

There are three different types of data which are published on the web site <http://solar.research.nicta.com.au>

- Power data for all or a range of data loggers for a given start to end date. For the newer version of the data loggers installed on about 50% of the houses, additional data for the voltage will be provided.
- Location information respecting the privacy option chosen by the home owner. For 67 data loggers, full disclosure is possible. For 30 locations, random noise of up to 100 m is added. And for 3 location, the rated output of the system is not available.
- Cloud camera data

9.6 Evidence of completion

The following map shows the distribution of data loggers over the ACT region (Figure 11.1). Examples of data recordings from several data loggers have been presented in the Figures 9.2, 9.3, 9.4, 9.5, and 9.6.

9.7 Conclusions and recommendations

In this chapter, we described the collection, analysis and publication of the collected data.

The guiding principle for the design of the collection system was to minimally interfere with the technical and security setup of the participating institutions (e.g. ANU, NICTA). Furthermore, each data stream pushing data to the collection system was designed with local buffer space allowing the central data collection system to be offline for several days without incurring any data loss.

As the data collection from the data loggers only started in May 2016, the value of the collected data will steadily increase as more data become available. For machine learning purposes, it is very much desired to have at least one year of continuously collected data available. We therefore aim to continue the data collection. We

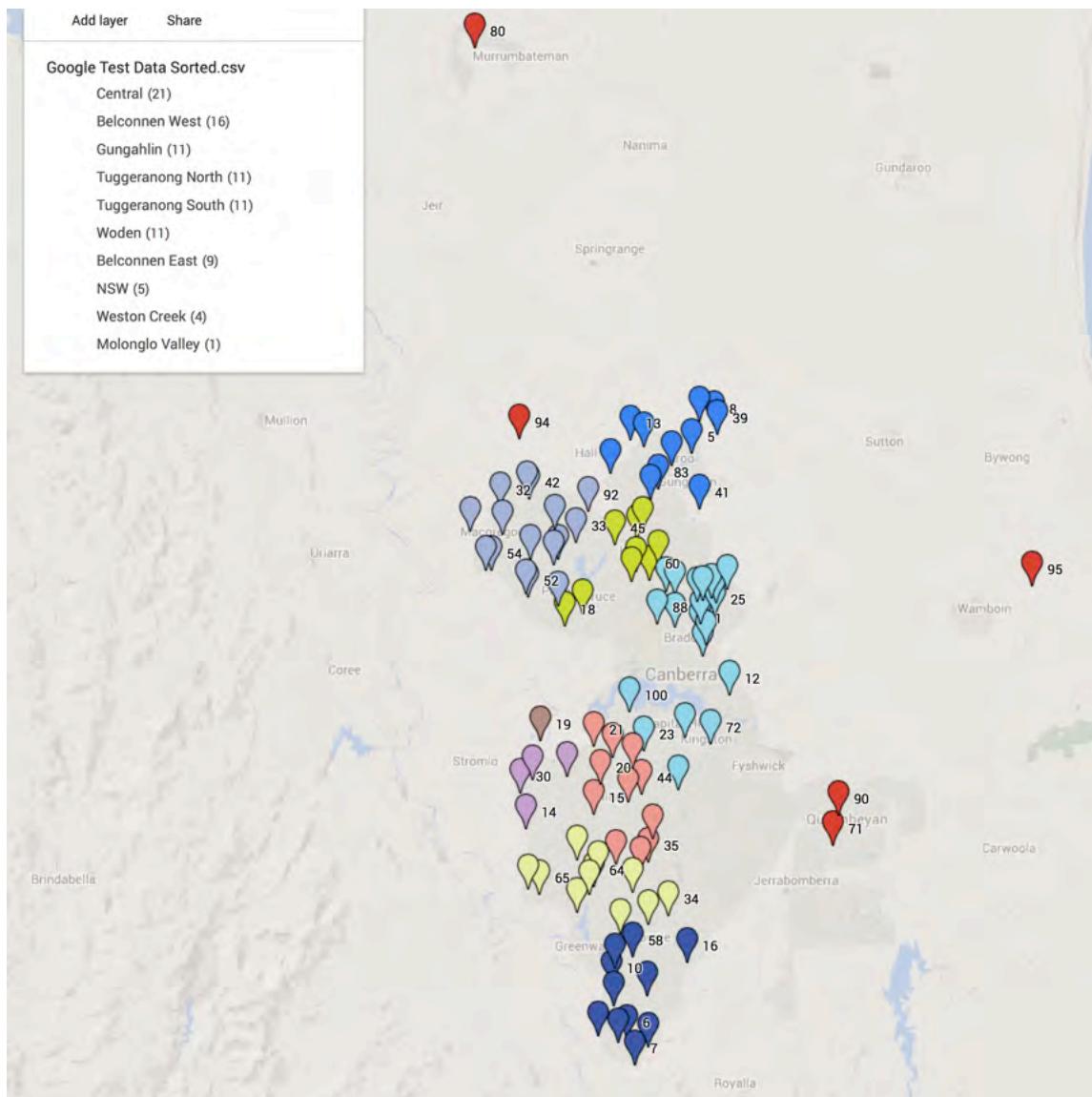


Figure 9.7: Locations of 100 data loggers installed in the ACT region.

are currently talking to our sponsor Telstra about their continuing support for sim cards and communication costs.



Chapter 10

Loss-calibrated predictions

10.1 Overview

This Chapter summarises the outcomes of the project concerned with fusing loss functions and probabilistic modelling in both single- and distributed-site regimes. It describes the Project Agreement objectives, details the extent to which these objectives were met, highlights the challenges that were encountered, and provides recommendations for leveraging the outcomes of our analysis and experiments.

10.2 Objectives

The objectives of this module as defined in the Project Plan (Deed of Variation 2) were to:

- Integrate the loss functions developed in earlier milestones into final predictors and extend these to produce probabilistic outputs that are calibrated to the needs of individual users.
- Design new losses for the joint forecasts over all sites, and extend loss-calibrated probabilistic predictors to this regime.
- Demonstrate the real-world value of loss-tuned probabilistic solar forecasts to best reflect the needs of users.

Our goals in this Chapter are to investigate precisely these issues. We have already discussed in Chapter 7 the value of producing *probabilistic* forecasts, as opposed to single *point* forecasts. In this Chapter, we build on this work and discuss:

1. The value of *loss-calibrated predictions* where, given a probabilistic forecast, we pick a single prediction to minimise the expected user loss when a single point “action” is chosen. This might allow, e.g., for users to explicitly say that over-prediction is worse than under-prediction.
2. The value of *loss-calibrated probabilistic learning*. Here, one learns a probabilistic forecast. However, rather than optimising a generic measure like a likelihood, we measure the average quality of the loss-calibrated prediction above. Note here that the user loss is being used in two places. First, it is used to determine the mapping from a forecast to an action. Second, it determines the mapping from an action (observed value) to a measure of accuracy. When learning, parameters determining the probabilistic forecast are adjusted so that the average value of this final measure of accuracy is as good as possible, taking into account the entire prediction pipeline.
3. The value of “joint” vs “single-site” prediction. Some market participants may not be concerned about the errors of predictions at single sites, but only the mean errors over some area. (Imagine a utility producing power for a city.) It is natural to conjecture that some of the errors in predictions at individual sites average out, and thus directly predicting the mean value would be easier than predicting individual values. We investigate this using several different loss functions.



10.3 Loss-calibrated probabilistic modelling: theory

We describe a simple approach to fitting a probabilistic model so as to be calibrated for a loss function. We begin with our basic probabilistic modelling framework.

10.3.1 Probabilistic modelling

To present our approach in the greatest generality, we shall suppose we are interested in predicting some signal $y \in \mathcal{Y}$ given input $\mathbf{x} \in \mathcal{X}$. (In a solar prediction context, the signal is simply the PV output at a site, while the input some feature representation relevant to that site, e.g. cloud images.) Typically, we have $\mathcal{Y} \subseteq \mathbb{R}$, and $\mathcal{X} \subseteq \mathbb{R}^d$. To do so, we construct a *parametric probabilistic model* $p(y | \mathbf{x}; \theta)$, where $\theta \in \Theta$ denotes the parameters to the model.

Example 1. In linear regression, we have a Gaussian model for $y | \mathbf{x}$ with mean $\mathbf{w}^T \mathbf{x}$ and variance σ^2 :

$$p(y | \mathbf{x}; \theta) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-(y - \mathbf{w}^T \mathbf{x})^2 / (2\sigma^2)}$$

Here, $\Theta = \{\mathbf{w}, \sigma\}$. Note that for the purposes of this section, we do not consider additional uncertainty in w due to limited training samples, though these can be incorporated as well.

In an abuse of notation, let $\mathbb{P}_\theta(\mathbf{x})$ denote the conditional distribution over \mathcal{Y} i.e. $p(\cdot | \mathbf{x}; \theta)$ for a given $x \in \mathcal{X}$ and $\theta \in \Theta$. For example, for a Gaussian $p(y | \mathbf{x}; \theta)$, this is simply a normal distribution with mean $\mathbf{w}^T \mathbf{x}$ and variance σ^2 .

We now consider two questions relevant to using such a model:

- how do we use it to make predictions?
- how do we estimate its parameters?

10.3.2 Loss-calibrated predictions

Suppose we have some parameter estimate θ . (Precisely how this is derived is deferred to the next section.) We can then use our probabilistic model to make predictions about the signal. A natural prediction for the signal is simply the *expected value* or average signal under our model,

$$a(\mathbf{x}; \theta) = \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [\mathbf{Y}] . \quad (10.1)$$

Of course, the above is not the only prediction one could make. For example, one might equally justify intuitively predicting the median of the signal under our model. How can we choose amongst different predictions? One way is to impose a *loss function* $\ell: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ with which we measure the quality of our predictions. We can then seek the *loss-calibrated prediction*, where we minimise the *expected loss* under our probability model:

$$a^*(\mathbf{x}; \theta) = \operatorname{argmin}_{a \in \mathcal{Y}} \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [\ell(\mathbf{Y}, a)] . \quad (10.2)$$

Example 2. Suppose $\mathcal{Y} \subseteq \mathbb{R}$, and we use the *square loss*

$$\ell(y, \hat{y}) = (y - \hat{y})^2.$$

Then,

$$\begin{aligned} a^*(\mathbf{x}; \theta) &= \operatorname{argmin}_{a \in \mathcal{Y}} \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [(Y - a)^2] \\ &= \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [Y], \end{aligned}$$



viz. the standard mean prediction of Equation 10.1.

Example 3. Suppose $\mathcal{Y} \subseteq \mathbb{R}$, and we use the *least absolute deviation loss*

$$\ell(y, \hat{y}) = |y - \hat{y}|.$$

Then,

$$\begin{aligned} a^*(\mathbf{x}; \theta) &= \operatorname{argmin}_{a \in \mathcal{Y}} \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [|Y - a|] \\ &= \text{median}(\mathbb{P}_\theta(\mathbf{x})). \end{aligned}$$

10.3.3 Loss-calibrated probabilistic modelling

We would like to fit θ to observed samples $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ of instances and target signals. To do so, the natural way is to simply find the maximum likelihood estimate of θ given the samples, which corresponds to minimising the *negative log-likelihood* of θ :

$$\mathcal{L}(\theta) = \sum_{i=1}^n -\log p(y_i \mid \mathbf{x}_i; \theta).$$

There is much to commend about this approach: it is simple, well-understood, and when the probability model is a member of the exponential family, involves a convex optimisation.

However, an obvious limitation of the above is that it does not incorporate any knowledge of the loss function used to evaluate predictions. For example, suppose we are going to be evaluated on least absolute deviation error. Then, intuitively it makes sense to ensure that the median of $\mathbb{P}_\theta(\mathbf{x})$ is accurately modelled for every $x \in \mathcal{X}$, possibly at the expense of accurate modelling of the tails of the distribution.

Remark 1. Note that if our model were well-specified, and we had a large enough number of samples, this issue would be irrelevant, as we would be able to exactly recover $\mathbb{P}_\theta(\mathbf{x})$, and thus any summary statistic thereof. However, in practice, neither condition holds.

To rectify this issue, a natural approach is to directly minimise

$$F(\theta) = \sum_{i=1}^n \ell(y_i, a^*(\mathbf{x}_i; \theta)) \tag{10.3}$$

where $a^*(\mathbf{x}; \theta)$ is as per Equation 10.2. We call this approach *loss-calibrated probabilistic modelling*. This approach certainly overcomes the limitation of standard maximum likelihood estimation. However, it comes at a price: how do we efficiently compute gradients of the above, since a^* depends on θ via a non-trivial argmin? This is the focus of the next section.

10.4 Loss-calibrated probabilistic modelling: optimisation

The basic approach to optimising $F(\theta)$ as in Equation 10.3 is to use automatic differentiation. This powerful toolkit lets us differentiate F with complexity exactly that required to actually compute the function. Concretely, then, all that is required from us is a specification of how to compute $a^*(\mathbf{x}; \theta)$. We have seen earlier that for the special case of square loss, the minimisation has a closed form solution. We now discuss how to compute $a^*(\mathbf{x}; \theta)$ for more complicated, asymmetric losses. In what follows, for fixed $x \in \mathcal{X}$, let

$$\begin{aligned} L(a; \mathbf{x}, \theta) &= \mathbb{E}_{\mathbf{Y} \sim \mathbb{P}_\theta(\mathbf{x})} [\ell(Y, a)] \\ &= \int_{\mathcal{Y}} p(y \mid \mathbf{x}, \theta) \cdot \ell(y, a). \end{aligned}$$

Evidently,

$$a^*(\mathbf{x}; \theta) = \operatorname{argmin}_a L(a; \mathbf{x}, \theta).$$



10.4.1 Asymmetric loss with closed-form a^*

Let $\ell(y, \hat{y}) = \phi(\hat{y} - y)$ where

$$\phi(v) = v + \exp(-v). \quad (10.4)$$

This asymmetric loss is seen to penalise under-predictions much more than over-predictions. Fortunately, under this loss there is a simple closed form solution for the optimal action a^* , thus permitting automatic differentiation with minimal effort.

Theorem 1. *For the loss ϕ given in Equation 10.4, we have*

$$a^*(\mathbf{x}; \theta) = \log \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \exp(y) dy.$$

Proof. We have

$$\begin{aligned} L'(a; \mathbf{x}, \theta) &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \ell'(y, a) dy \\ &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \phi'(a - y) dy \\ &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot (1 - \exp(y - a)) dy. \end{aligned}$$

Thus, at optimality,

$$\int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) dy = \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \exp(y) \cdot \exp(-a^*) dy,$$

or

$$a^* = \log \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \exp(y) dy,$$

since the LHS must be 1 by definition. \square

Now, the integral above may be difficult to compute in general. We can however approximate it by sampling. Suppose we pick grid points $y^{(1)}, \dots, y^{(G)}$, e.g. uniformly spaced on the interval $[-B, B]$ for some sufficiently large B such that we capture most probability mass. Then,

$$a^* = \log \sum_{g=1}^G p(y^{(g)} | \mathbf{x}, \theta) \cdot \exp(y^{(g)}) - \log \sum_{g=1}^G p(y^{(g)} | \mathbf{x}, \theta),$$

where the denominator appears because with sampling, the approximation to the LHS is no longer 1.

10.4.2 Asymmetric loss with semi-closed form a^*

Let $\ell(y, \hat{y}) = \phi(\hat{y} - y)$ where

$$\phi(v) = \begin{cases} \frac{1}{2}\alpha v^2 & \text{if } v < 0 \\ \frac{1}{2}\beta v^2 & \text{if } v > 0 \end{cases} \quad (10.5)$$

for some $\beta > \alpha$. This asymmetric loss is seen to penalise under-predictions much more than over-predictions. This loss does not have a closed-form solution for a^* . But, it can nonetheless be efficiently computed.

Theorem 2. *For the loss ϕ given in Equation 10.5, we have*

$$a^*(\mathbf{x}; \theta) = \frac{\alpha \cdot \int_{\mathcal{Y}: y > a^*} y \cdot p(y | \mathbf{x}, \theta) + \beta \cdot \int_{\mathcal{Y}: y < a^*} y \cdot p(y | \mathbf{x}, \theta)}{\alpha \cdot \int_{\mathcal{Y}: y > a^*} p(y | \mathbf{x}, \theta) + \beta \cdot \int_{\mathcal{Y}: y < a^*} p(y | \mathbf{x}, \theta)}.$$



Proof. Observe that

$$\begin{aligned} L'(a; \mathbf{x}, \theta) &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \ell'(y, a) \\ &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \phi'(a - y) \\ &= \int_{\mathcal{Y}} p(y | \mathbf{x}, \theta) \cdot \begin{cases} \alpha \cdot (a - y) & \text{if } a < y \\ \beta \cdot (a - y) & \text{if } a > y \end{cases} \end{aligned}$$

Thus, at optimality,

$$\alpha \cdot \int_{\mathcal{Y}: y > a^*} (a^* - y) \cdot p(y | \mathbf{x}, \theta) = -\beta \cdot \int_{\mathcal{Y}: y < a^*} (a^* - y) \cdot p(y | \mathbf{x}, \theta)$$

or

$$a^* = \frac{\alpha \cdot \int_{\mathcal{Y}: y > a^*} y \cdot p(y | \mathbf{x}, \theta) + \beta \cdot \int_{\mathcal{Y}: y < a^*} y \cdot p(y | \mathbf{x}, \theta)}{\alpha \cdot \int_{\mathcal{Y}: y > a^*} p(y | \mathbf{x}, \theta) + \beta \cdot \int_{\mathcal{Y}: y < a^*} p(y | \mathbf{x}, \theta)}.$$

□

Notice the right hand side above depends on a , and thus the above does not strictly constitute a closed-form solution. We can efficiently find a solution via linear search, however. Suppose we use a finite grid as before. Then, the above is

$$a^* = \frac{\alpha \cdot \sum_{g=1}^{G'} p(y^{(g)} | \mathbf{x}, \theta) \cdot y^{(g)} + \beta \cdot \sum_{g=G'+1}^G p(y^{(g)} | \mathbf{x}, \theta) \cdot y^{(g)}}{\alpha \cdot \sum_{g=1}^{G'} p(y^{(g)} | \mathbf{x}, \theta) + \beta \cdot \sum_{g=G'+1}^G p(y^{(g)} | \mathbf{x}, \theta)}$$

for some unknown grid point G' . This grid point can be determined by a simple linear scan to find the transition point where the RHS goes from being negative to positive, as this determines the crossing point of the derivative of $L(a; \mathbf{x}, \theta)$.

10.5 Instantiation for distributed solar prediction

The above presentation was abstract in order to establish the generality of the proposed approach. We now make explicit how to translate it to our problem of distributed solar prediction. In distributed solar prediction, we are interested in predicting the PV output $y_{it} \in \mathcal{Y}$ for a number of sites $i \in \{1, 2, \dots, N\}$ over a number of timestamps $t \in \{1, 2, \dots, T\}$. We use as features $\mathbf{x}_{it} \in \mathcal{X}_i$, thus allowing a per-site feature representation. (This may be useful if for example each site is represented using measurements from all neighbouring sites within a certain distance of it.) Further, we use a per-site probabilistic model $p(y_i | \mathbf{x}_i; \theta_i)$, with distinct parameters $\{\theta_i\}_{i=1}^N$ for each site.

Notice we can effectively decompose the training set into a number of independent training sets, one per each site:

$$\begin{aligned} \mathcal{S} &= \bigcup_{i=1}^N \mathcal{S}_i \\ \mathcal{S}_i &= \{(\mathbf{x}_{it}, y_{it})\}_{t=1}^T. \end{aligned}$$

The preceding analysis can thus be applied to each \mathcal{S}_i , since there is no sharing of *parameters* amongst the various models, although there may be sharing of *features* as the following illustrates.

Example 4. Suppose we use as features recordings (e.g. time-lagged PV, cloud images) for *all* N sites. In this case, there is the same ostensible feature representation used for all sites at a given timestamp. We

then have

$$\mathcal{S} = \bigcup_{i=1}^N \mathcal{S}_i$$

$$\mathcal{S}_i = \{(\mathbf{x}_t, y_{it})\}_{t=1}^T.$$

When the features for the i th site only comprise its nearest neighbours, we can preserve same global feature representation, and think of the procedure as a per-site local regularisation.

Note finally that our loss from the previous section is more explicitly

$$F(\boldsymbol{\theta}) = \sum_{i=1}^N F_i(\theta_i) \quad (10.6)$$

$$F_i(\theta_i) = \frac{1}{T} \sum_{t=1}^T \ell(y_{it}, a^*(\mathbf{x}_{it}; \theta_i))$$

where $\boldsymbol{\theta} = [\theta_1; \theta_2; \dots, \theta_N]$. Thus, all analyses from the previous section may be applied to each per-site objective $F_i(\theta_i)$.

10.6 Experiments on loss-calibration

We now present experiments illustrating that the value of our loss-calibrated probabilistic modelling framework for a range of different losses and probability models.

10.6.1 Probability models

We consider the following probability models for $p(y | \mathbf{x}; \theta)$:

- **FIXED VARIANCE GAUSSIAN.** Here, we have a Gaussian model

$$p(y | \mathbf{x}; \theta) = \frac{1}{\sqrt{2\pi}\sigma_0} \cdot e^{-(y - \mathbf{w}^T \mathbf{x})^2 / (2\sigma_0^2)}$$

defined for $y \in \mathbb{R}$, where $\Theta = \{\mathbf{w}\}$ and $\sigma_0 \in \mathbb{R}_+$ is some *fixed* parameter. This is the implicit probabilistic model used when fitting ordinary least squares.

- **LEARNED VARIANCE GAUSSIAN.** Here, we have a Gaussian model

$$p(y | \mathbf{x}; \theta) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-(y - \mathbf{w}^T \mathbf{x})^2 / (2\sigma^2)}$$

defined for $y \in \mathbb{R}$, where $\Theta = \{\mathbf{w}, \sigma\}$, so that the variance is also estimated from data. Fitting such a model is thus roughly equivalent to performing ordinary least squares but with estimation of the variance from the data.

- **TRUNCATED GAUSSIAN.** Here, we have a *truncated* Gaussian model

$$p(y | \mathbf{x}; \theta) = \frac{C}{\sqrt{2\pi}\sigma} \cdot e^{-(y - \mathbf{w}^T \mathbf{x})^2 / (2\sigma^2)}$$

defined for $y \in [0, 1]$, where $\Theta = \{\mathbf{w}, \sigma\}$, and C is a constant ensuring the above is a valid distribution. As the name suggests, the above is simply a standard Gaussian distribution where we explicitly constrain the range of allowed values. Figure 10.1 compares the two distributions.

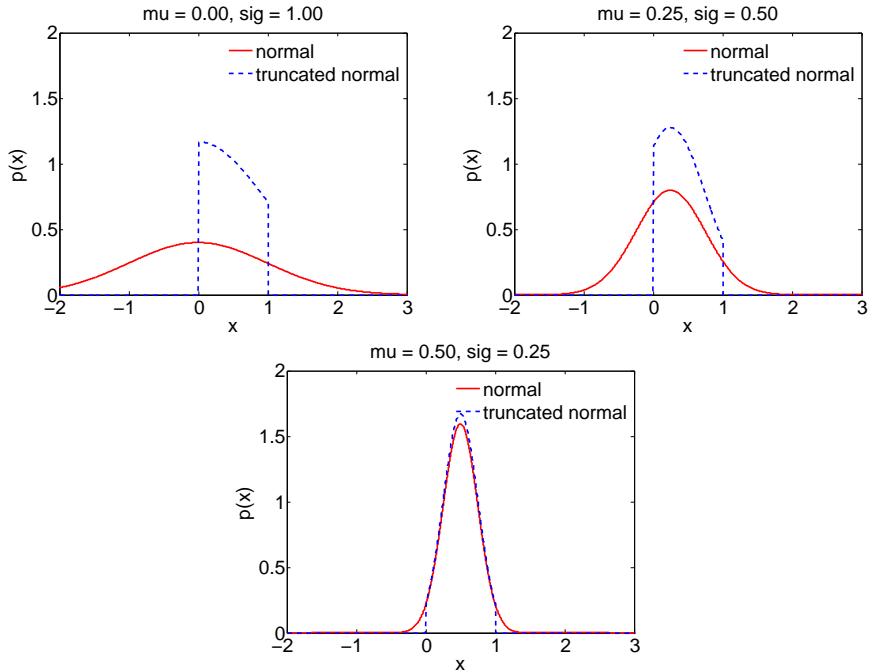


Figure 10.1: Comparison of truncated Gaussian and standard Gaussian distributions for various choices of mean μ and standard deviation σ .

- **BETA.** Here, we have a Beta model

$$p(y | \mathbf{x}; \theta) = C \cdot y^\alpha \cdot (1 - y)^\beta$$

defined for $y \in [0, 1]$, where $\Theta = \{\mathbf{w}\}$, C is a constant ensuring the above is a valid distribution, and

$$\begin{aligned} \alpha &= \frac{1 - \mu(\mathbf{x})}{\mu(\mathbf{x})} \cdot \beta \\ \mu(\mathbf{x}) &= f(\mathbf{w}^T \mathbf{x}) \end{aligned}$$

for some link function $f: \mathbb{R} \rightarrow [0, 1]$. We considered for simplicity a capped sigmoid $f(z) = a + (1 - a)/(1 + e^{-z})$. Figure 10.2 shows some examples of the Beta distribution for various fixed choices of α, β .

Note that PV targets in our experiments are guaranteed in $[0, 1]$. Thus, the latter two distributions are arguably more appropriate than the former two. Note however that the latter two have the disadvantage of encouraging predictions to be in $(0, 1)$ i.e. away from the boundaries, which can be an issue since the PV is often exactly zero e.g. during nighttime.

10.6.2 Losses

We consider the following losses $\ell(y, \hat{y})$ to measure the performance of any candidate predictor:

- **SQUARED**, which is squared loss,

$$\begin{aligned} \ell(y, \hat{y}) &= \phi(y - \hat{y}) \\ \phi(v) &= \frac{1}{2}v^2. \end{aligned}$$

As noted earlier, this is the loss function employed when using ordinary least squares.

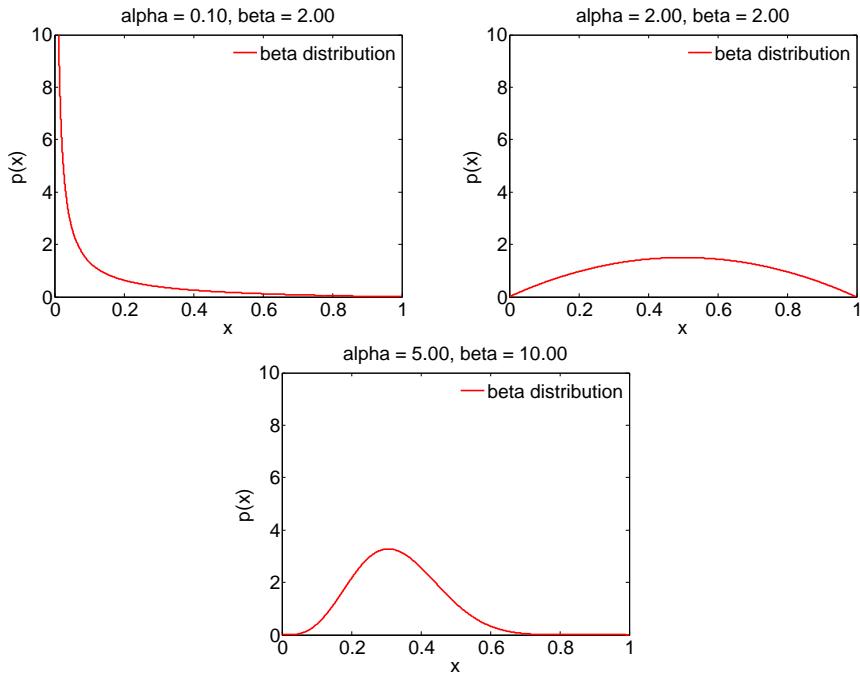


Figure 10.2: Beta distribution for various choices of hyperparameter (α, β) .

- M-LOSS, the novel asymmetric loss of §10.4.1, given by

$$\begin{aligned}\ell(y, \hat{y}) &= \phi(y - \hat{y}) \\ \phi(v) &= v + (e^{-a \cdot v} - 1)/a\end{aligned}$$

for tuning parameter $a > 0$. This loss is plotted for various values of a in Figure 10.3. It is seen to more strongly penalise under-predictions compared to over-predictions.

10.6.3 Experimental results

We aim to demonstrate with our experiments that:

- it is feasible to train a probabilistic model to be calibrated to a given loss, and
- performing such training yields better results with respect to the given loss, than when training with a generic loss.

To illustrate (a), we implemented loss-calibration for each of the probability models and loss functions above. Using an automatic differentiation library, all that was required for this step was to implement the procedure to find $a^*(x; \theta)$ efficiently, which is trivial for squared loss and discussed in §10.4.1 for the M-Loss. We ran experiments on simulated data as described in our previous Chapters. Specifically, we generated a year of data using 50 sites in the “hard” setting where each day has a randomly generated wind direction, and clouds move with extra randomness in each 10 minute timestep added to position, opacity, and size. For results on single-site prediction, we use the first site, where the normalized power 10 minutes ahead is predicted using the power at the most recent 5 timesteps, as well as the power 24 hours ago. For multi-site prediction, we predict the power 10 minutes ahead using the 3 most recent measurements at all sites.

To illustrate (b), we first report for each probability distribution the difference in predictions produced when training to optimise square loss, versus when training to optimise the M-Loss. This is shown in Figure 10.4. The results are intuitive: we find that in general, training with squared loss results in predictions that are systematically *larger* than those produced when training with the M-Loss. This is precisely what we expect, since

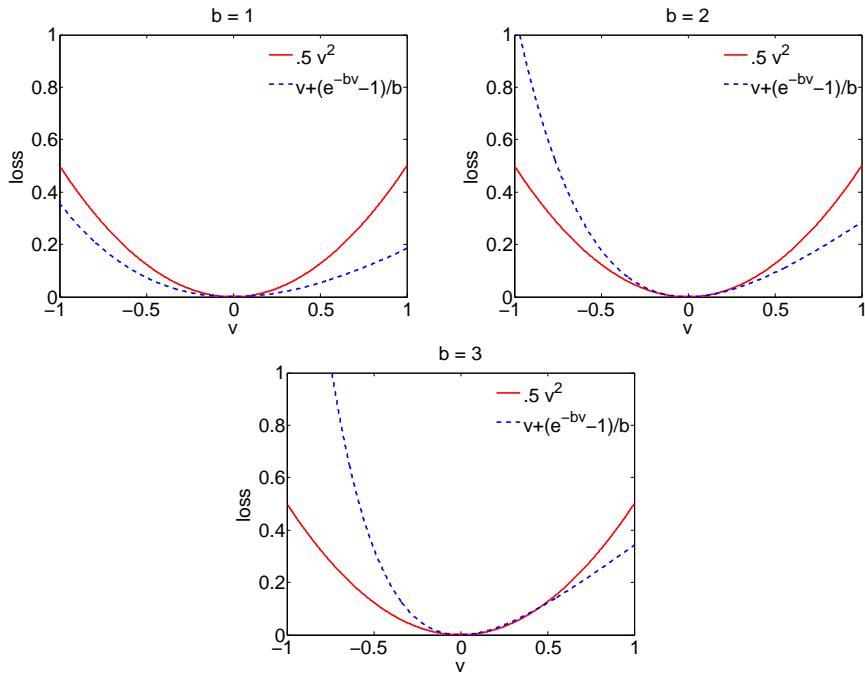


Figure 10.3: The asymmetric M-Loss for various values of tuning parameter a .

the latter loss strongly penalises under- rather than over-predictions. It is notable that the basic trend is evident for each of the distributions, despite them possessing markedly different characteristics. This illustrates the power of our generic loss-calibration framework, and indeed the value of separating the concerns of modelling a probability distribution and making loss-calibrated actions.

To further evince this trend, Figures 10.5 – 10.8 show a subsample of the predicted PV time series produced when using each of underlying probability distributions, and training with each of the losses. The plots show the ground truth time series as a dashed black line, the loss-calibrated prediction $a^*(\mathbf{x}; \theta)$ as a blue line, and the 5% and 95% quantiles as a shaded blue boundary around the $a^*(\mathbf{x}; \theta)$ curve. We find that, consistent with the previous plots, training when using the M-Loss results in a slight but consistent tendency to under-predict the PV signal when compared to training with the square loss.

Finally, we present a table of the average evaluation loss when training for a given loss, for all possible combinations of losses. This numerically confirms that loss-calibrated training indeed results in predictions with slightly lower average error than when training with some other loss.

Distribution	Train Loss	Square Loss Error	M-Loss Error
Gaussian fixed variance	Square	0.00488	0.00252
	M-Loss	0.00491	0.00251
Gaussian learned variance	Square	0.00488	0.00252
	M-Loss	0.00491	0.00251
Truncated Gaussian	Square	0.01110	0.00574
	M-Loss	0.01112	0.00573
Beta	Square	0.01045	0.00538
	M-Loss	0.01046	0.00537

Table 10.1: Average loss incurred when training with a given loss for each probabilistic model.

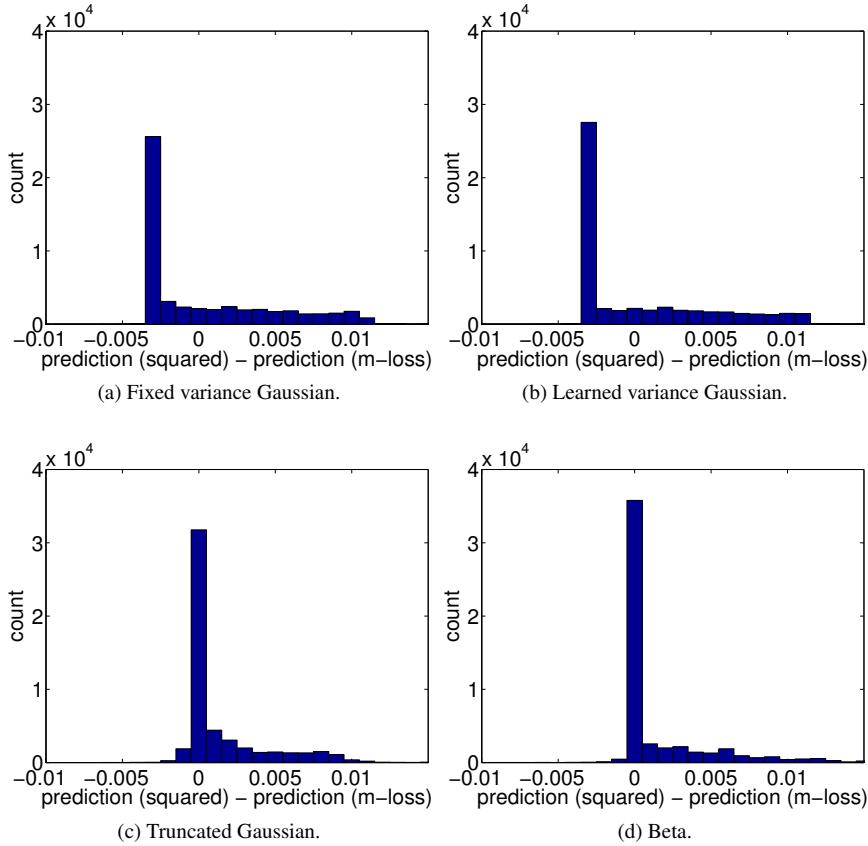


Figure 10.4: Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss.

10.7 Multi- versus single-site prediction

The preceding discussion focussed on modelling the PV at a single site, with multiple sites accommodated by simple averaging of per-site models. In this project, we are centrally interested in *distributed* solar prediction, and so it is natural to consider performance measures with a more holistic consideration of the predictions on individual sites. In this section, we present examples of multi-site loss functions, and discuss how to directly optimise such losses.

10.7.1 Multi-site loss functions

The previous section considered predictions for a single target $\mathcal{Y} \subseteq \mathbb{R}$, which in a solar context represents the PV output for a single site. We allowed for multiple sites by simply considering each site in isolation, and averaging the prediction error on each. However, one can conceive of more complex measures that do not necessarily decompose into per-site measures.

A simple yet generic example of such a loss is the following. Given PV values (real or predicted) for N sites, we consider a generic *aggregation function* $\text{Agg}: \mathcal{Y}^N \rightarrow \mathcal{Y}$ which compresses the observations into a single summary statistic. We then combine this with a standard (single site) loss $\ell: \mathcal{Y} \rightarrow \mathcal{Y}$ to give the *multisite aggregate loss*

$$\ell_{\text{agg}}(\mathbf{y}, \hat{\mathbf{y}}) = \ell(\text{Agg}(\mathbf{y}), \hat{\mathbf{y}}_{\text{Agg}}). \quad (10.7)$$

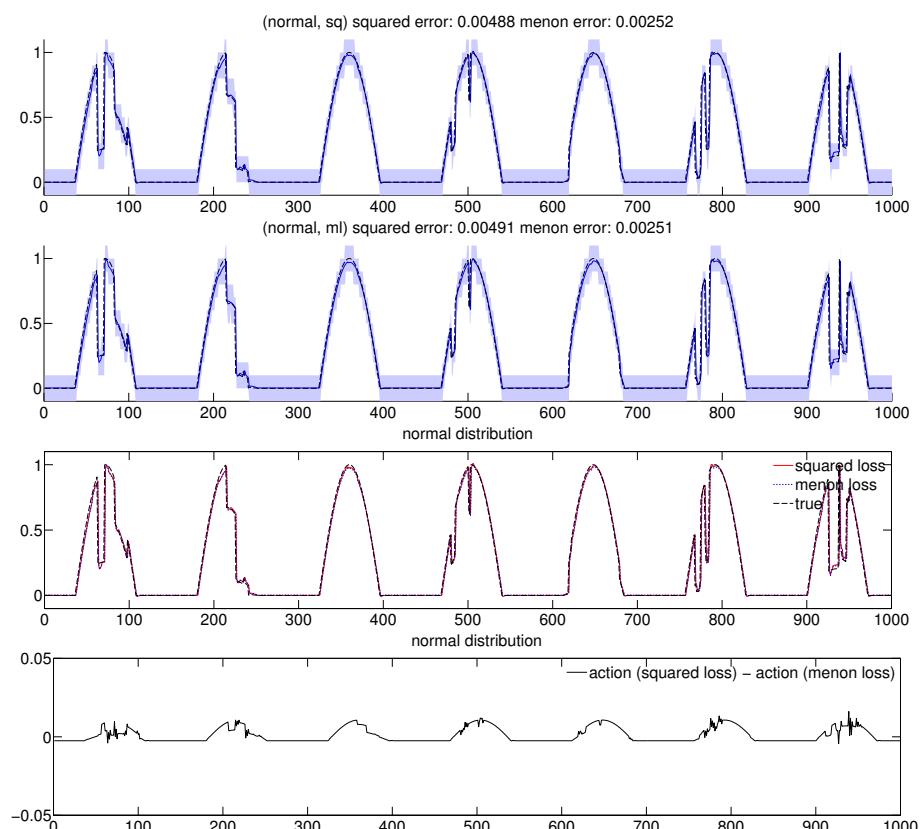


Figure 10.5: Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Gaussian distribution with fixed variance. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.

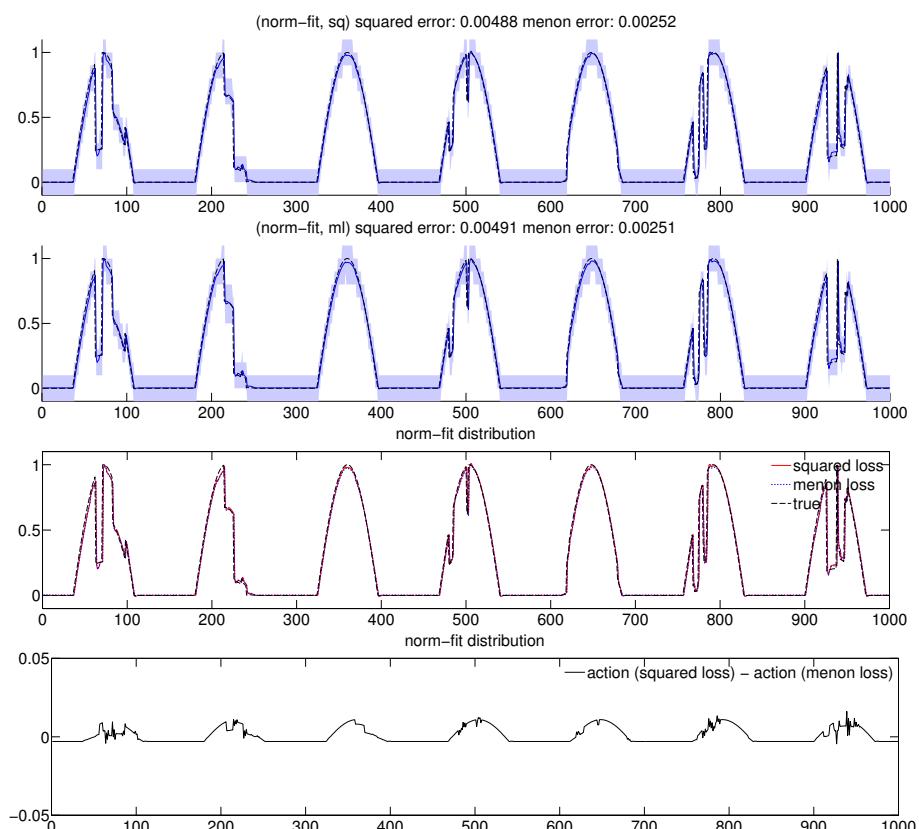


Figure 10.6: Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Gaussian distribution with learned variance. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.

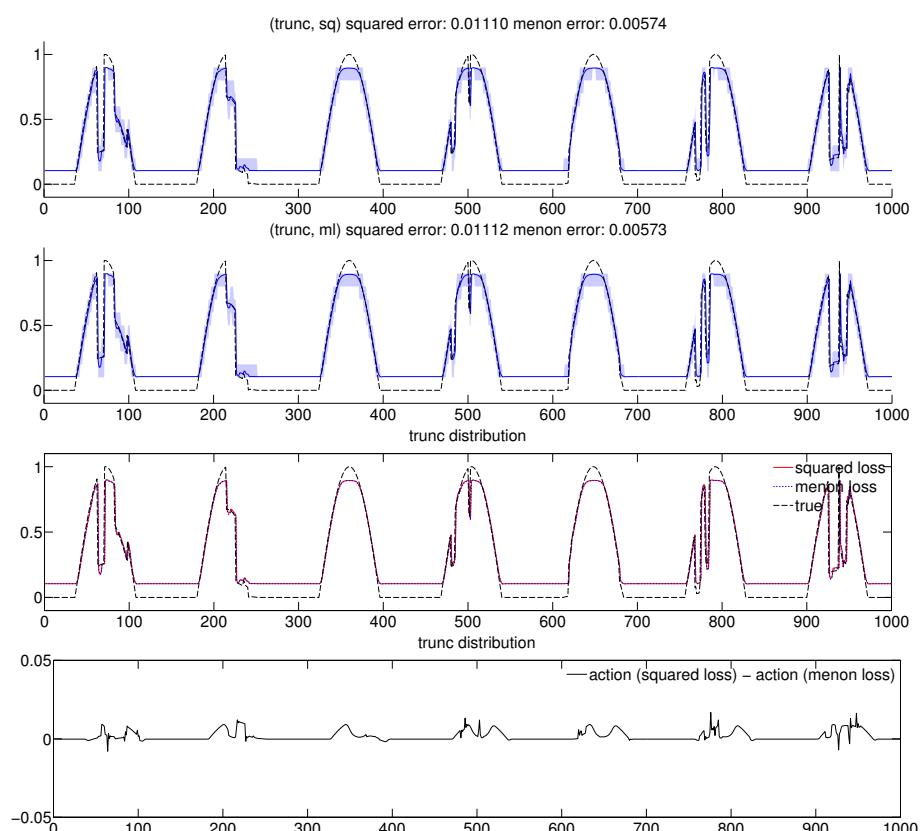


Figure 10.7: Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, truncated Gaussian distribution. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.

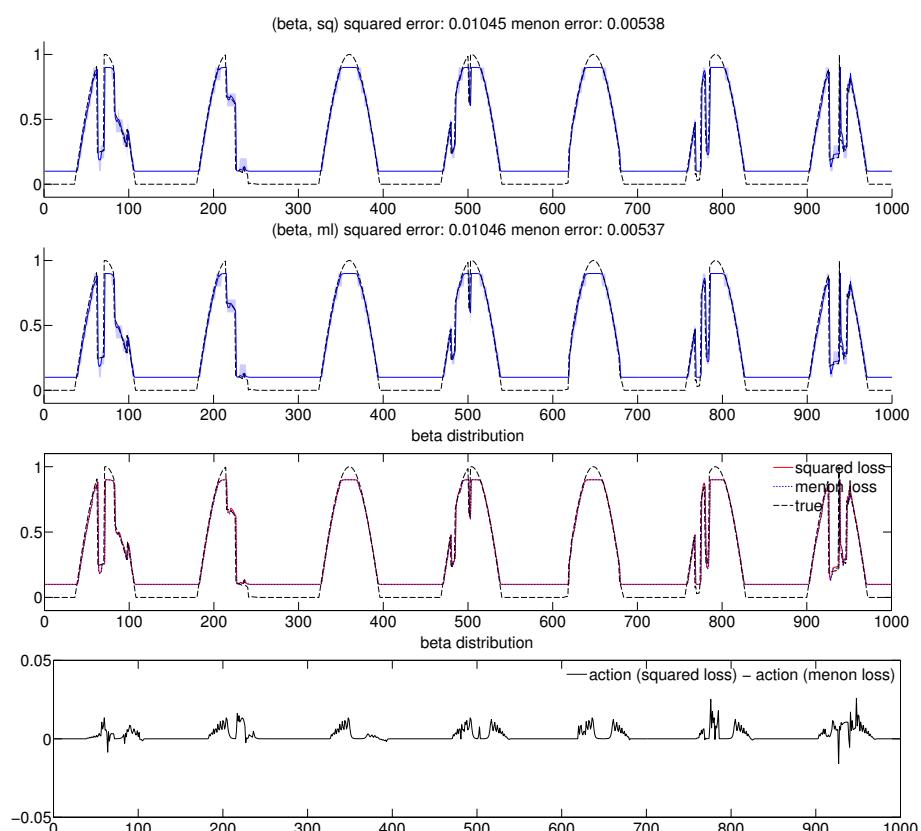


Figure 10.8: Comparison of predictions when training various probabilistic models to optimise square loss versus the M-Loss, Beta distribution. First plot shows prediction and quantile when using squared loss, Second plot when using M-Loss, Third plot an overlay of the predictions for the two losses, and Fourth plot the difference between the predictions for the two losses.



Example 5. Let ℓ be any single site loss, and $\text{Agg}(\mathbf{y}) = \mathbf{1}^T \mathbf{y}$. Then, ℓ_{agg} measures the quality of predictions for the *aggregate* PV output.

10.7.2 Optimising multi-site loss functions: two broad approaches

There are two ways of optimising a multi-site loss, such as the aggregate loss of Equation 10.7.

- We can indirectly compute the aggregate by first computing per-site PV predictions, and then aggregate them *post-hoc*. Formally, suppose we compute $\hat{\mathbf{y}} \in \mathcal{Y}^N$ using e.g. the methods of the previous section. Then, we can compute

$$\hat{y}_{\text{Agg}} = \text{Agg}(\hat{\mathbf{y}}).$$

- We can directly model the aggregate, completely skipping any intermediate predictions on individual sites. One can consider probabilistic models for the aggregate, and thus considering $\text{Agg}(\mathbf{y})$ as the signal for a conceptual “pseudo-site”, one can apply the methods of the previous section.

Example 6. Let ℓ be any single site loss, and $\text{Agg}(\mathbf{y}) = \mathbf{1}^T \mathbf{y}$. Then, in the first approach above, we would first model the individual PV outputs on each site, and then post-hoc compute the sum of all predicted PVs. The second approach by contrast would attempt to directly model the aggregate PV. This is intuitively a much simpler problem, as we do not need to consider the idiosyncrasies of each sites’ PV pattern.

Remark 2. In general, the aggregate of the loss on individual sites is not necessarily the same as the loss of the aggregate prediction over all sites. That is,

$$\text{Agg}(\ell(\mathbf{y}, \hat{\mathbf{y}})) \neq \ell(\text{Agg}(\mathbf{y}), \text{Agg}(\hat{\mathbf{y}})).$$

Intuitively, we expect the latter approach to perform better, as it directly optimises the performance measure of interest, but furthermore involves solving a simpler problem. However, in an important special case, we show that the two approaches are *identical*.

10.7.3 An equivalence when using linear models and square loss

A natural first attempt at optimising multi-site loss functions is to consider squared loss as the base predictor,

$$\ell(y, \hat{y}) = (y - \hat{y})^2.$$

In the following, we shall assume the use of shared features for every site, so that our training sample may be compactly represented as

$$\mathbf{S} = (\mathbf{X}, \mathbf{Y})$$

where $\mathbf{X}_{:t} \in \mathbb{R}^d$ comprises the shared d -dimensional feature representation for all sites at time t , and $\mathbf{Y}_{ti} \in \mathcal{Y} \subseteq \mathbb{R}$ comprises the PV observation for site i at time t . In the simplest case, $\mathbf{X}_{:t}$ comprises the time-lagged PV observations at all N sites, so that $d = N$ and $\mathbf{X} \in \mathbb{R}^{T \times N}$, $\mathbf{Y} \in \mathbb{R}^{T \times N}$.

We now consider the two approaches outlined in the previous section.

- Recall the first approach is to learn per-site models, and aggregate the predictions *post-hoc*. Suppose we make predictions using an ordinary least squares model, so that

$$\hat{y}_i(\mathbf{x}; \theta_i) = \mathbf{w}_i^T \mathbf{x},$$

where $\Theta = \{\mathbf{w}_1; \dots; \mathbf{w}_N\}$. We have optimal solution

$$\mathbf{w}_i^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}_{:i}.$$

The matrix of weights for each site is thus

$$\mathbf{W}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}.$$

The matrix of predictions for each site is then

$$\hat{\mathbf{Y}} = \mathbf{X} \mathbf{W}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}$$

The resulting aggregate prediction is then

$$\begin{aligned}\hat{y}_{\text{Agg}} &= \hat{\mathbf{Y}} \mathbf{1} \\ &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \mathbf{1}.\end{aligned}$$

- (b) Recall the second approach is to learn directly the aggregate statistic. Suppose we model this with a linear model,

$$\hat{y}_{\text{Agg}}(\mathbf{x}; \theta) = \mathbf{w}^T \mathbf{x},$$

where $\Theta = \{\mathbf{w}\}$. We will find \mathbf{w} to optimise the square loss to the observed aggregate PV on each site, i.e. we will minimise

$$F(\theta) = \|\mathbf{X} \mathbf{w} - \mathbf{Y} \mathbf{1}\|_2^2.$$

Evidently, the optimal weight is

$$\mathbf{w}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \mathbf{1},$$

so that the resulting aggregate prediction is then

$$\begin{aligned}\hat{y}_{\text{Agg}} &= \hat{\mathbf{Y}} \mathbf{w}^* \\ &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \mathbf{1}.\end{aligned}$$

Thus, in both cases, we find the predicted aggregate to be the same. We emphasise that such a result is specific to the use of square loss. For more general losses, we expect directly optimising for the aggregate to be beneficial.

10.8 Experiments on multi- versus single-site prediction

We now present experiments illustrating that optimising directly for the multi-site aggregate loss can be beneficial over *post-hoc* aggregating the results of optimising a per-site loss. We consider predicting the total PV using the two approaches of the previous section. Both approaches are trained with a number of losses:

- SQUARED, the squared loss,

$$\begin{aligned}\ell(y, \hat{y}) &= \phi(y - \hat{y}) \\ \phi(v) &= \frac{1}{2}v^2.\end{aligned}$$

- M-LOSS, the asymmetric loss of §10.4.1, given by

$$\begin{aligned}\ell(y, \hat{y}) &= \phi(y - \hat{y}) \\ \phi(v) &= v + (e^{-a \cdot v} - 1)/a\end{aligned}$$

for tuning parameter $a > 0$.

- SMOOTH PINBALL, a smoothed version of the pinball loss given by

$$\begin{aligned}\ell(y, \hat{y}) &= \phi(y - \hat{y}) \\ \phi(v) &= \log(e^v + e^{-av})\end{aligned}$$

for tuning parameter $a > 0$. When $a = 1$, this is simply $\phi(v) = \log \cosh(v)$, a standard differentiable alternative to the absolute value function.



Loss function	Direct optimisation	Indirect optimisation
M-Loss ($a = 1$)	.000182	.000187
M-Loss ($a = 2$)	.000364	.000407
M-Loss ($a = 3$)	.000547	.000711
Smooth pinball ($a = 1$)	.693329	.693329
Smooth pinball ($a = 2$)	.671615	.671721
Smooth pinball ($a = 5$)	.598195	.598346
Squared	.000182	.000182

Table 10.2: Comparison of direct and indirect optimisation of aggregate statistics.

Table 10.2 summarises the results of direct and indirect optimisation of total PV for each loss. We find that in general, direct optimisation yields slight improvements over indirect optimisation. (In keeping with our theoretical analysis, the square loss sees no difference between the two approaches.) We find a significant difference with skewed version of the M-Loss, which strongly penalise under-predictions.

To further illustrate these differences, Figures 10.9 – 10.11 present histograms of the differences in predictions using direct and indirect optimisation for each of the losses. In keeping with the results of Table 10.2, we see that for the asymmetric M-Loss the direct optimisation approach tends to slightly underpredict the aggregate PV compared to the indirect approach. This shows that, despite the fact that the latter attempts to ensure that individual sites have slight underprediction in PV, directly optimising for the aggregate statistic can produce more finely loss-calibrated results.

10.9 Extent to which the objectives were achieved

We believe we have largely met the objectives laid out in the Project Plan. In detail:

- A general theoretical framework for producing loss-calibrated predictions has been provided. We further showed how this framework can be instantiated with (tractable variants of) the cost-sensitive losses introduced in Chapter 7, which are of interest when over-prediction and under-prediction are not treated equally.
- A range of experiments has been presented confirming that loss-calibrated methods do indeed yield consistent benefits, in keeping with the theory. We did so with a range of probabilistic models and loss functions, including a new family of asymmetric losses (the M-Loss). This gives credence that for a generic utility or loss (e.g. a favouring of over- rather than under-prediction), our framework can yield benefits to the end user of a forecast.

One limitation of our work in this component was that we did not consider losses suitable when detection of ramp events are of interest. Such losses fall comfortably in our framework, and so in principle should not pose any problems. A careful empirical assessment of the usage of these losses would be an important piece of future work.

- A general class of losses for measuring the discrepancy between any the actual and predicted aggregate statistic of local PV outputs has been provided. We then argued theoretically and showed empirically that directly optimising such a loss can yield better performance than merely optimising local losses. It has been shown that in a special case, the two approaches may be identical, which is an important finding that needs to be considered when experimenting with such an approach.

One limitation of our work for this component was that we did not consider more complicated joint-site measures, such as those taking into account sensor location. While there is no conceptual difficulty in doing so, and indeed it is trivial to accomodate our framework to allow for arbitrary joint-site losses, a more careful exploration of the implications for learning in such scenarios would be of interest.

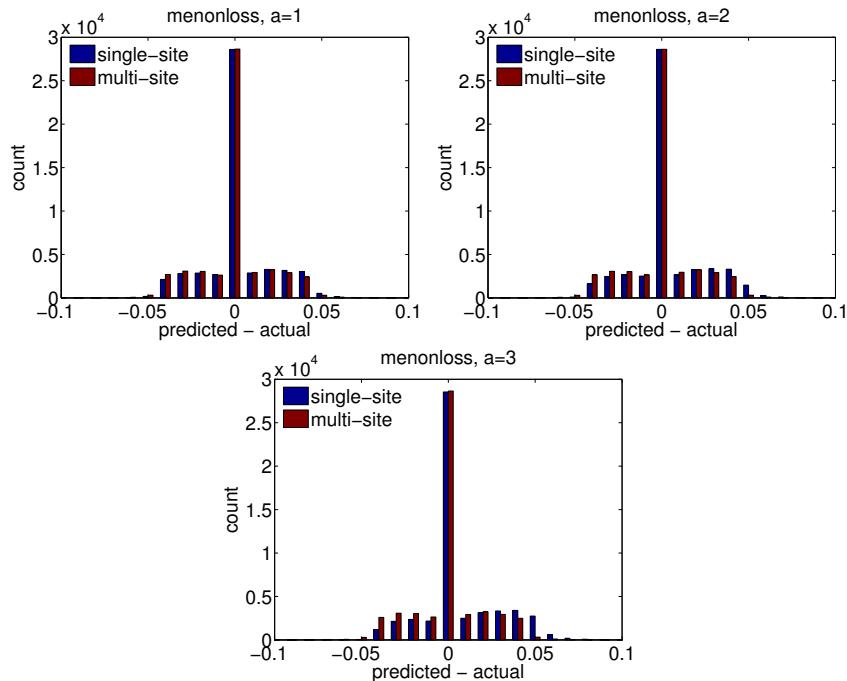


Figure 10.9: Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, M-Loss.

10.10 Highlights, difficulties, and breakthroughs

There are three main highlights in this module.

- (1) A novel *loss-calibrated probabilistic modelling* view of solar PV prediction was presented. As noted in earlier chapters, the idea of probabilistic forecasting is by itself relatively novel in the solar forecasting community. We are not aware of any prior work on loss-calibrating such forecasts for PV prediction. We believe our experiments indicate the promise of such approaches.
- (2) We presented a means of easily obtaining loss-calibrated predictions via automatic differentiation. This solves the non-trivial problem of finding model parameters that result in the best optimal actions, where the latter are implicitly defined as the result of a nested optimisation problem.
- (3) We showed the value of directly optimising for a joint or aggregate performance measure, such as total PV output in an area, compared to simply optimising for the average of local measures. This further illustrates a point we have made before, namely that if one fixes a performance measure, then optimising for it directly can be beneficial. We showed that in an interesting special case of square loss, the joint and local approaches are however identical.

Three main difficulties were encountered in this module.

- (1) First, as indicated in our experiments, while loss-calibration consistently yields positive improvement, it does not always yield *significant* improvement. Our initial assessment of this approach was thus clouded by the apparently lack-lustre empirical performance. It was only after some inspection that we discovered that the choice of loss function is crucial in providing a noticeable benefit for loss-calibration (in particular, variants of the M-Loss showed conclusive benefits of the approach).
- (2) Second, we found initial implementation of loss-calibration to be non-trivial. This is because it was not immediately obvious how to optimise the model parameters given an implicitly defined $a^*(\mathbf{x}; \theta)$. Viewing the problem in terms of automatic differentiation was a breakthrough that facilitated easy experimentation.

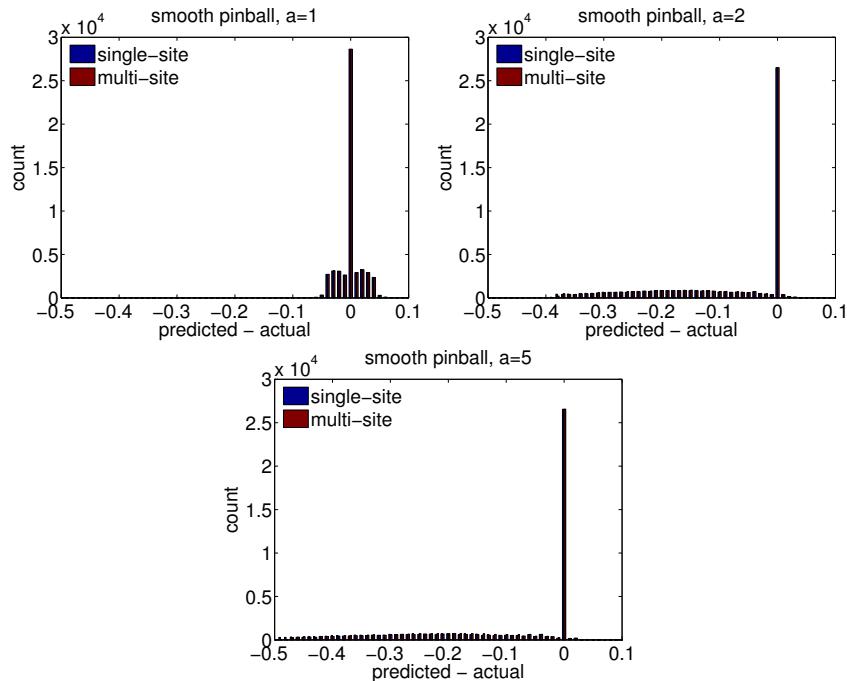


Figure 10.10: Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, smooth pinball loss.

(3) Third, we found initial implementation of loss-calibration with learning of the variance of a normal distribution to be very fragile. On the other hand, we found that fixing the variance to sometimes yield sub-optimal results. The key ingredient that let us overcome these was the use of the persistence forecaster’s variance as the “origin” against which other candidate measures were compared.

10.11 Conclusion and recommendations

We have described the outcomes of our work on designing probabilistic forecasters whose predictions are calibrated to a specific loss function. We also discussed the use of losses that consider the distributed nature of PV prediction, using as an example the problem of predicting the aggregate PV for a given solar array. Our experiments confirm that both these ideas can yield consistent benefits for forecasting.

To build on our work, we make the following specific recommendations:

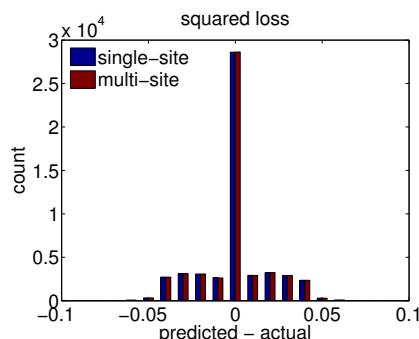


Figure 10.11: Histogram of differences in aggregate PV predictions to ground truth using direct and indirect optimisation, square loss.



- we expect that it is possible to design better probabilistic models of the PV, in particular accounting for heteroskedastic variance. Training loss-calibrated versions of such models would be of interest.
- incorporating uncertainty in the model parameters due to limited sample sizes in the loss-calibration framework would allow for more fully Bayesian inference to be performed in a loss-sensitive way.
- exploration of other aggregate performance measures (e.g. taking into account the precise geographical layout of sites) would be of interest.



Chapter 11

Sub-sampled PV Monitoring Data

11.1 Overview

This section of the final report describes the work completed to acquire relevant sub-station data from ActewAGL and the utilisation of this data to quantify the relationship between sampled solar energy data for 100 PV sites and 30-minute and 5-minute voltage/current measurements from Canberra’s electricity distribution network. We undertake this analysis at three Canberra substations: Belconnen, City East and Wanniassa. Additionally, we also explore the relationships between the distribution network monitoring data and simulated collective PV power output from the Regional PV Simulation System (RPSS, v1). Finally, we make some recommendations for future work in this area, based on our findings.

11.2 Objective

Our primary objective, with respect to the Project Plan was to:

- Acquire sub-station data from ActewAGL; compare those data with sub-sampled data logger measurements in order to quantify how well the sampled data represent the total solar energy production in the given area.

This objective can be broken into two key components. First, sub-station data from ActewAGL must be acquired, processed and prepared for analysis. Second, available PV monitoring data must be analysed with respect to this sub-station data, to determine how representative these data are of total solar energy production in the region concerned. Note that the second objective contains a key caveat in the statement “total solar energy production”, as there is a disconnect between the two data sources. Neither the substation data, nor the monitored PV data can give us an estimate of total solar energy production. This clearly highlights the need for a method which can quantify the total solar energy production in a given region, and for this reason, we employ the RPSS [4] in order to estimate total solar energy production.

11.3 Evidence of Completion

The completion of the overall objective is demonstrated in the following three subsections, which correspond to the above two key components and exploration of the “total solar energy production” caveat.

11.3.1 Acquiring Zone Sub-station Data from ActewAGL

In order to accomplish the first key task, data for 12 zone substations were acquired from ActewAGL for the 2012-2015 period. These data were originally organised into .csv format, and contain load data (MVA) in 30-minute intervals. An R code was written to break data into individual days in the format YYYY-MM-DD in preparation for analysis within the *anusolar* R package environment [3]. Example data for these 12 zone substations is shown in Figure 11.1, from two ramp events observed on 5 March 2014 as shown in Figure 11.2.

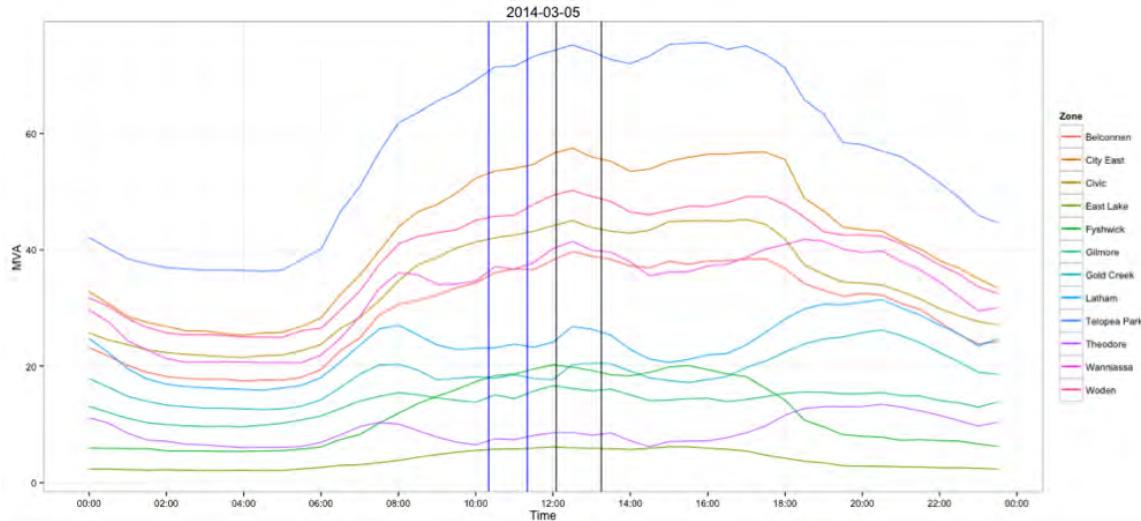


Figure 11.1: Example Zone Substation Data for 12 Zones in the ActewAGL Network. The blue vertical lines bound a negative ramp event and the black vertical lines, a positive ramp event.

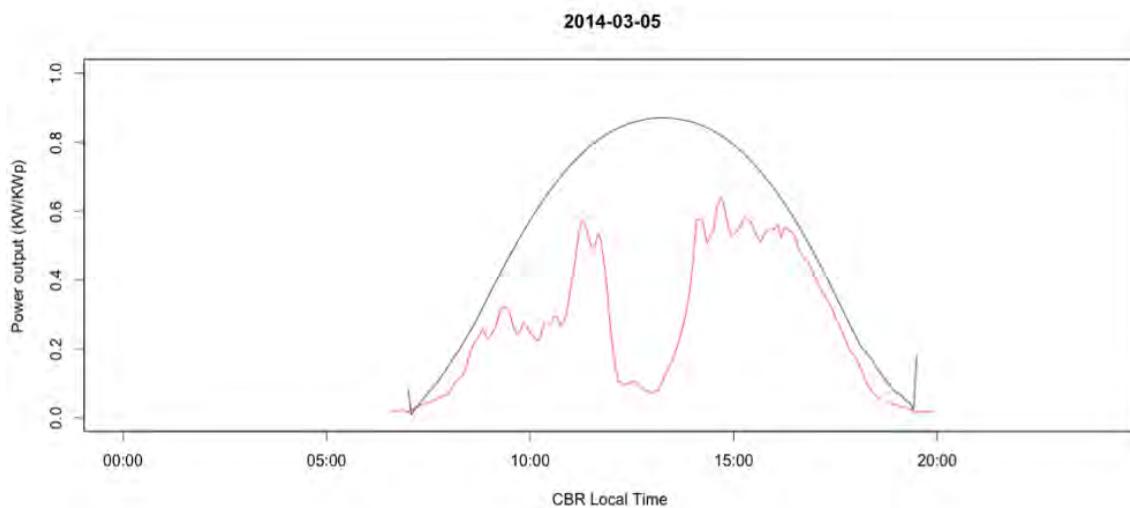


Figure 11.2: Example ramp event date showing a negative ramp event and a positive ramp event, both shown in Figure 11.1. The black line indicates the clear-sky power output from all monitored PV systems, and the red line the mean observed power output from all monitored PV systems. Units are in kW/kW_p (observed power output divided by installed capacity).

11.3.1.1 Additional Data Acquisition

In addition to the required project data, through our work with ActewAGL, we were able to obtain 5-minute interval SCADA data for the 2013-2015 period for 3 zone substations: Belconnen, City-East and Wanniassa sites. Again, an R code was written to process the data from .csv files to a format which will work in the manusolar R package environment. These substations were selected based on geographic spread and data availability within ActewAGL's SCADA system. The location of these substations can be identified in Figure 11.3. Using substation location data from ActewAGL, an algorithm was developed for grouping monitored PV sites according to their appropriate upstream zone substation.



Figure 11.3: ActewAGL Distribution Network Zone Substations (from Energy Consumer Reference Council Background Information Kit, 2014).

11.3.2 Analysis of Sub-sampled PV Data

The second key task required a comparison of the zone substation data with sub-sampled data logger measurements in order to quantify how well the sampled data represent the total solar energy production in the given area. It is hypothesized that solar power output and monitored grid power output would have an inverse relationship. More specifically, that the grid's output will be reactive to solar PV outputs — for each negative solar ramp event, the grid power output will have a positive ramp reaction to balance the energy lost (and vice versa).

To accomplish this task, we utilised monitored power output data from 100 PV sites distributed across Canberra, that recorded power output measurements over the 2013-2015 calendar years. These data were extracted from PVOutput.org, before being processed through a rigorous quality control algorithm [2]. Next, power output data was grouped by assumed upstream zone substations based on high-level suburb based grouping details provided by ActewAGL. These groupings of the chosen zone substations are shown in Table 11.1. It is important to recognise that at the time of research, PV installations were not mapped to feeders or substations precisely. The locations of the chosen quality controlled sites that were instead grouped to suitable substations by hand, and are displayed in Figure 11.5.

Preliminary analysis focused on the comparison of the grouped power output data with 30-minute zone

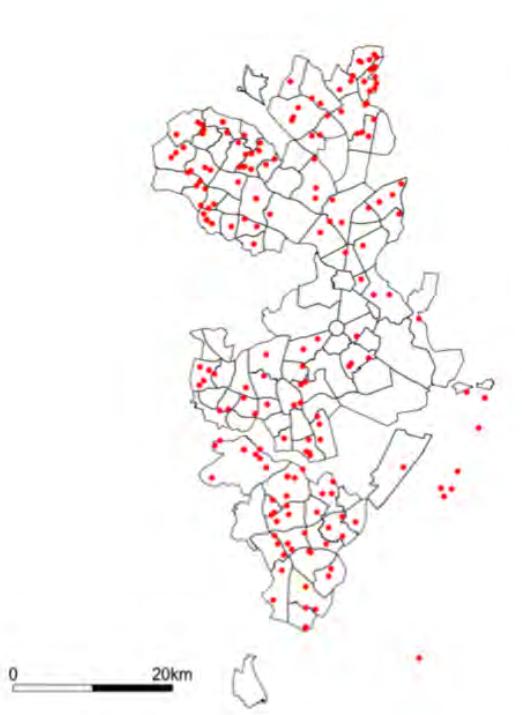


Figure 11.4: Location of the 100 PV sites utilised in this analysis. Red dots indicated the location of the reporting PV system. Black polygons display the boundaries of Canberra suburbs.

substation data using known critical collective ramp event dates. Critical collective ramp events are defined by a $\geq 60\%$ change in collective power output in ≥ 60 minutes, and were explicitly linked to key weather event categories for Canberra[1, 6]. Additional ramp events were detected through processing of quality-controlled data using an R script developed in [5]. The complete list of ramp events used within this study is available in the Appendix.

Using these ramping events, we undertook a linear correlation analysis utilizing the Pearson correlation coefficient. The Pearson correlation coefficient is a measure of the linear correlation between two variables, and gives a value between +1 (positive correlation) and -1 (negative correlation) where 0 represents no correlation. As outlined above, it is expected that negative correlations will be observed between solar PV output data and zone substation data.

Zone Substation	Suburbs
Belconnen	Hall, Evatt, McKeller, Giralang, Kaleen, Lawson, Belconnen, Bruce
City East	Lyneham, Dickson, Downer, Ainslie, Hackett, Braddon
Wanniassa	Pearce, Torrens, Mawson, Isaacs, Farrer, Kambah, Wanniassa, Oxley, Monash, Bonython, Greenway

Table 11.1: High level mapping of suburbs by zone substation

11.3.2.1 30 minute zone substation data

Using the monitored PV power output data, the monitored PV sites estimated to be connected to each of the zone substations were compared against the interval data from the substations. Using each of the ramp events days, we calculated the correlation coefficient between the quality controlled power output and the zone substation measurements. The results demonstrated surprisingly weak correlations. We use a day with multiple

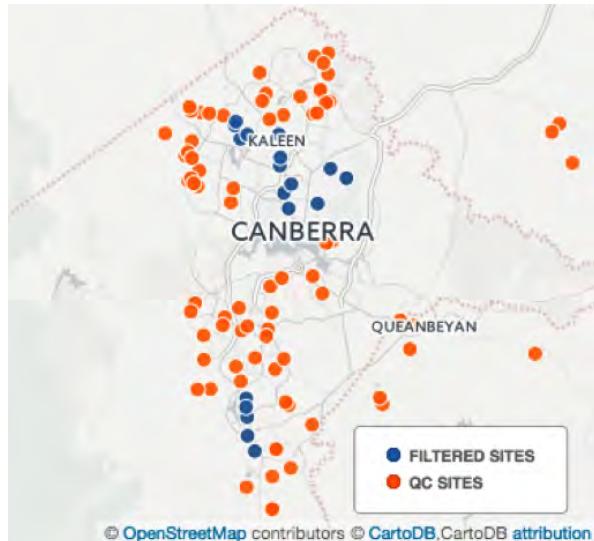


Figure 11.5: Location of the selected PV sites for analysis filtered according to substation location. Blue dots indicate sites assumed to fall in the chosen substations.

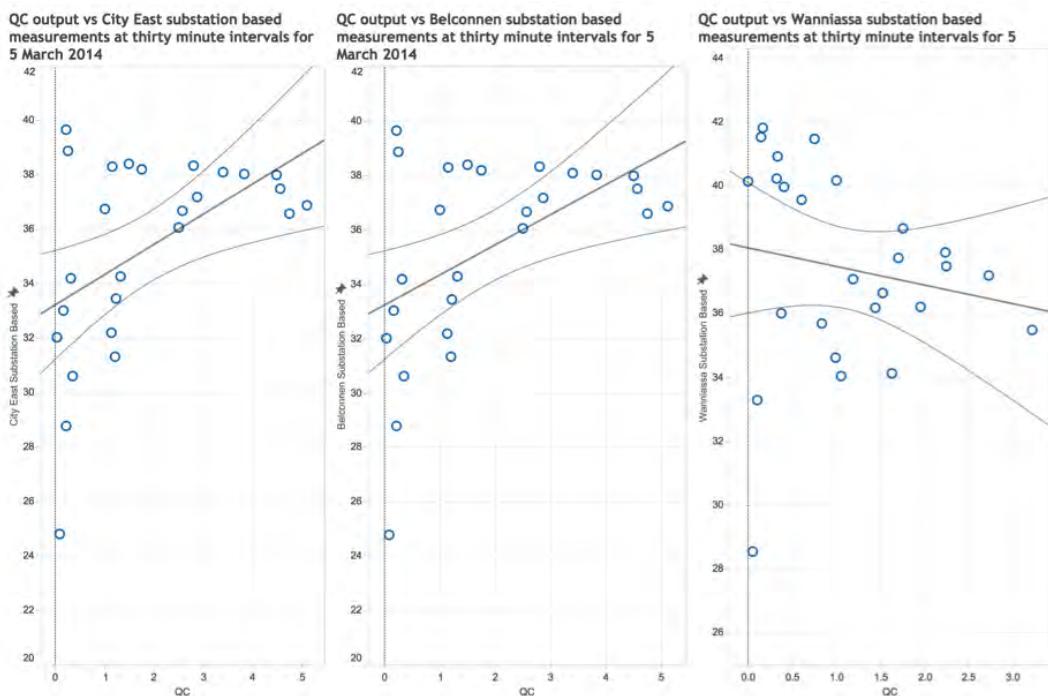


Figure 11.6: Correlations between filtered quality controlled sites (QC output) and substation power outputs for thirty-minute intervals for 5 March 2014.

ramp events, 5 March 2014, to illustrate these results (see Figure 11.6). Positive correlations are observed for City East and Belconnen whilst a negative correlation is observed for Wanniassa. The corresponding correlation coefficients are outlined in Table 11.2. Generally, coefficients greater than 0.8 are considered to be strong correlations, while those smaller than 0.5 suggest no correlation at all. Here the results show a very weak correlation at City-East and no correlation at Belconnen and Wanniassa. Similarly, uncorrelated results were observed for all other ramp event dates (Table 11.7).



Zone Substation	Correlation coefficient
Belconnen	0.476
City East	0.523
Wanniassa	-0.167

Table 11.2: Correlation coefficients for analysis of thirty minute intervals for 5 March 2014.

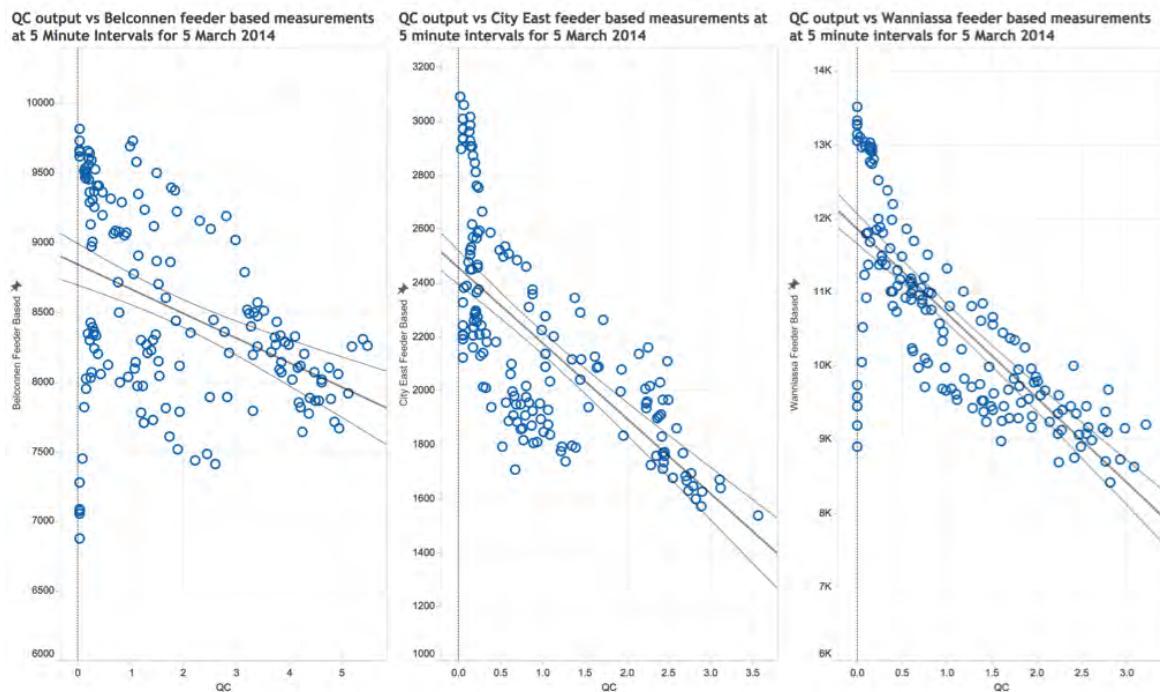


Figure 11.7: Correlations between quality controlled sites and feeder based power outputs for five minute intervals for 5 March 2014.

11.3.2.2 5 minute zone substation data

Given the poor correlations observed in the zone substation 30-minute interval data, we worked with ActewAGL to obtain 5-minute interval data from three zone substations — Wanniassa, Belconnen, and City East — and re-completed the ramp event analysis. For comparison to the 30-minute data in the previous section, we present the observed results from 5 March 2014 in Figure 11.7 and Table 11.6. There are increases in the value of the correlation coefficients when compared to 30-minute interval data, with the results in Table 11.6 showing weak correlations between QC power output data and the City East and Wanniassa substations. However, these results were inconsistent across many of the remaining ramp event dates.

Zone Substation	Correlation coefficient
Belconnen	-0.413
City East	-0.680
Wanniassa	-0.778

Table 11.3: Correlation coefficients for analysis of five minute intervals for 5 March 2014.

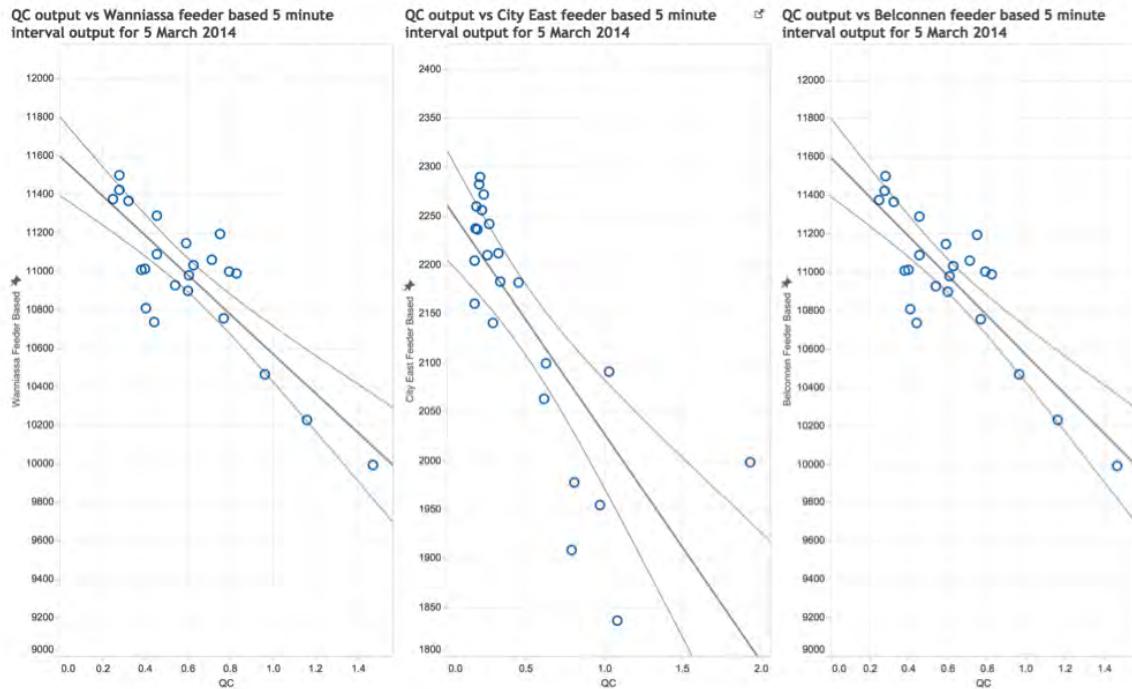


Figure 11.8: Correlations observed for five minute interval RPSS_v1 data set for collective ramp event on 5 March 2014.

11.3.2.3 Sub-setting ramp events

Given the lack of any definitive correlations in the data when applied in the manner discussed above, we further limited our analysis to the periods bounded by critical collective ramp event start and end times. This produced a sub-set of data from periods of the strongest changes in collective PV power output and re-applied our correlation analysis using the 5-minute interval data.

This sub-setting approach brought the observed correlations above an average value of 0.6 across the ramp event dates, with the results from 5 March 2014 provided in Figure 11.8 and 4 demonstrating strong correlations (coefficient $\geq |0.8|$) at the Belconnen and Wanniassa substations.

Zone Substation	Correlation coefficient
Belconnen	-0.871
City East	-0.780
Wanniassa	-0.841

Table 11.4: Correlation coefficients for analysis of ramp event subset five minute intervals for 5 March 2014.

11.3.3 Analysing Total Solar Energy Production via RPSS version 1

One shortcoming of the PV data sub-sampling analysis above, is that it does not truly compare network time series data against total solar energy production, which is specified within our overall objective. While Canberra has over 16,000 PV systems installed, our PV monitoring dataset only contains 100 sites. This leaves out some of the more complicated aspects of quantifying distributed solar impacts, such as shading or how power output is changing between our monitored PV sites.

In order to attempt to remove some of these considerations, we employ the Regional PV Simulation System (RPSS, version 1), which is a distributed solar modeling tool. The RPSS uses monitored PV power output

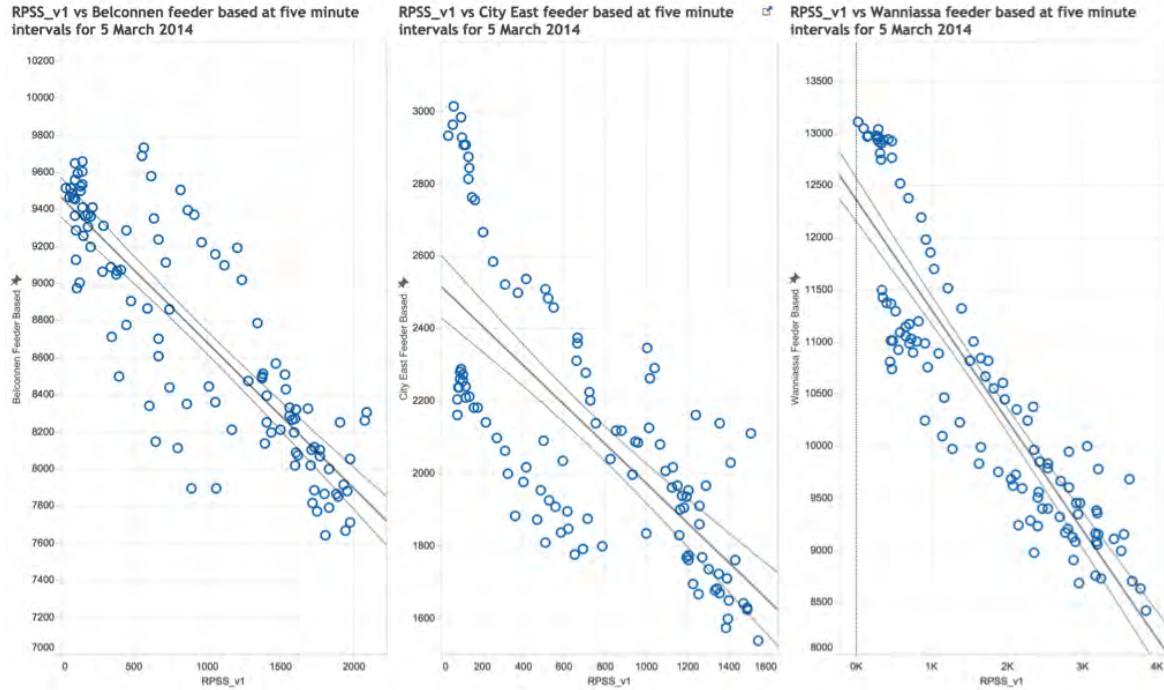


Figure 11.9: Correlations observed for five minute interval RPSS_v1 data set for collective ramp event on 5 March 2014.

from the 100 site dataset to generate power output estimates for the many thousands of PV systems installed across Canberra [2]. In this version of the RPSS, the collective power output from 12,000+ PV systems on ActewAGL's network is simulated during the critical ramp event periods used in the previous analyses.

We then compared the RPSS output against 5-minute time interval SCADA readings at the zone substations, grouping the simulated PV systems by their assumed upstream zone substation. Interestingly, as shown in Figure 11.9 and Table 11.6 the results improved significantly without sub-setting for the ramp events. Particularly for Belconnen and Wanniassa, these coefficients are strong suggesting a strong negative correlation for the entire daytime period (5 March 2014).

Zone Substation	Correlation coefficient
Belconnen	-0.843
City East	-0.694
Wanniassa	-0.843

Table 11.5: Correlation coefficients for analysis of five minute intervals RPSS_v1 data for 5 March 2014.

Finally, we apply the ramp event sub-setting method to the RPSS output and found strong negative correlations across all ramp events (values > 0.9 for 30 March 2014, and 13 October 2013 events). Results from the 5 March 2014 event are presented in Figure 11.10 and Table 11.6 below, for cross-comparison to previous sections.

11.4 Extent to which the objectives were achieved

We have successfully obtained interval data from ActewAGL's zone sub-stations at 30-minute and 5-minute resolutions. Using these data, we have analysed their relationship with power output data from a distributed network of PV systems which sub-sample the total solar energy production across the Canberra region. Through

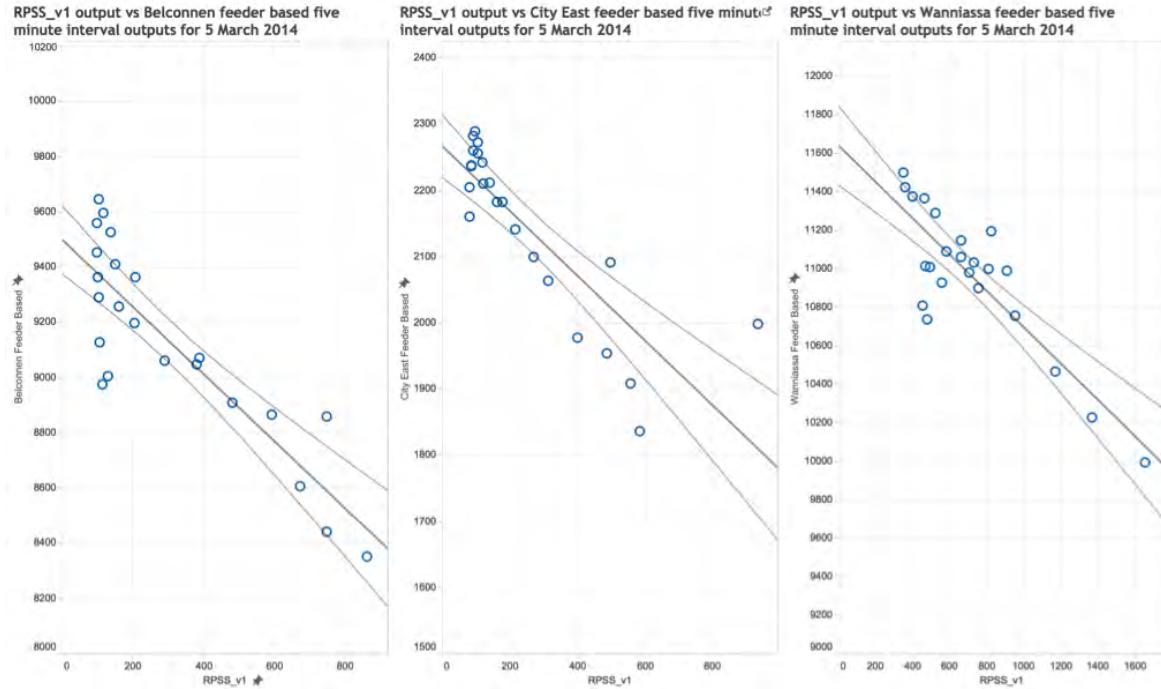


Figure 11.10: Correlations observed for subsetted data set for collective ramp event on 5 March 2014.

Zone Substation	Correlation coefficient
Belconnen	-0.897
City East	-0.862
Wanniassa	-0.872

Table 11.6: Correlation coefficients for analysis of subset five minute intervals for 5 March 2014 using RPSS.

application of the Regional PV Simulation System, we have also been able to compare the zone substation interval data to the simulated total solar energy production of all PV systems connected to these substations. Using these various combinations of data, we have computed correlations between the sub-station and PV data (observed and simulated), thereby quantifying ‘how well’ the observed or simulated data represent total solar energy production in the selected regions. Therefore, it is clear that the objective of this investigation has been largely achieved

11.5 Conclusions and Recommendations

Our research work herein demonstrates a suitable method for quantifying the relationship between distributed PV power output and zone substation load data, and has determined that 30-minute interval data are generally insufficient for these purposes, and that 5-minute interval data are required. Furthermore, to truly analyse these relationships, regional PV simulations have been demonstrated to be crucial to fully investigating the relationships between distributed PV power output and distribution network loads. Given a robust and accurate distributed PV modelling framework (such as the RPSS), strong correlations can be observed between distributed PV power output and network loads during critical collective ramping events. This is a unique and novel finding.

We recommend that future work be undertaken to further investigate the benefits of more accurate PV mapping to distribution networks, such as that now available in more advanced versions of the RPSS (version 2, [4]). Such distributed PV modelling efforts should also be joined with satellite based solar forecasting



methods, to attempt to forecast critical ramp events and predict their impact on the distribution network ahead of their arrival.

Date	Change in Collective Power Output
2012-01-06	-79.08
2012-02-08	-60.24
2012-08-24	68.57
2012-08-31	66.09
2012-09-21	63.95
2012-09-24	60.91
2012-09-24	-73.12
2012-11-18	63.03
2013-07-09	-60.29
2013-10-13	-78.16
2013-11-11	-73.21
2013-11-13	-66.89
2013-11-15	-78.38
2013-11-16	-68.53
2013-11-28	72..77
2013-12-05	73.89
2013-12-09	-67.96
2013-12-11	-69.25
2013-12-27	-77.96
2014-01-06	-69.43
2014-02-11	62.57
2014-02-19	-64.73
2014-03-05	-66.66
2014-03-10	70.10
2014-03-30	-83.00
2014-04-27	60.57
2014-04-30	-70.41
2014-07-16	-71.71
2014-10-23	-61.59
2014-12-05	-83.30
2014-12-22	-80.06
2014-12-29	-80.90
2015-01-06	-75.90
2015-01-07	-67.69
2015-01-24	-63.35
2015-03-01	-61.00

Table 11.7: List of all critical collective ramp events for which correlations were tested during this study

Bibliography

- [1] James Bill. Characterising collective ramp events in Sydney, Australia. Master's thesis, College of Engineering and Computer Sciences, The Australian National University, 2015.
- [2] Nicholas A. Engerer. *City-wide Simulations of Distributed Photovoltaic Array Power Output*. PhD thesis, The Australian National University, 2015.
- [3] Nicholas A. Engerer. *anusolar* R package, 2016.



-
- [4] Nicholas A. Engerer and James Hansard. Real-time simulations of 15,000+ distributed PV arrays at sub-grid level using the regional PV simulation system (RPSS). In *2015 Solar World Congress, International Solar Energy Society*, 2015.
 - [5] Shaw Kudo. Characterising collective ramp of distributed PV systems in Brisbane, Australia. Master's thesis, College of Engineering and Computer Sciences, The Australian National University, 2015.
 - [6] Sonya Wellby and Nicholas A. Engerer. Categorizing the meteorological origins of critical ramp events in collective photovoltaic array output. In *Journal of Applied Meteorology and Climatology*. 2016.





Chapter 12

Forecasting from Sky Images with Deep Convolutional Neural Networks

12.1 Overview

This chapter presents the results of our exploration of using deep convolutional neural networks (DCNN) to predict solar power from sky images. The major source of data are the images captured by the camera installed on the ANU Engineering building. In the following sections, we will describe the project objectives in detail, how they are achieved, the key challenges and breakthroughs, and provide recommendations for future work.

12.2 Objectives

The objective of this module as defined in the Project Plan was to explore the solar power prediction from sky images directly using DCNN. In chapter 6, we have shown that using the cutting-edge computer vision techniques are helpful to extract visual features. Combining such features with other data modalities as input to a linear or non-linear regressor (machine learning models) can lead to better solar power prediction. Note that those visual features are manually designed. In this work, we mainly explore whether the features can be *automatically* learned from the sky images using a DCNN in order to achieve good prediction. Such a framework allows us to perform an end-to-end training to avoid the failure in unexpected ways.

The primary objective was to show that DCNN can be helpful to predict solar power from sky images. The accuracy is one of the key aspects to show its effectiveness. Therefore, we address the overall objectives by pursuing the following two sub-goals,

- Explore different DCNN network architectures and tailor them for the solar power prediction task at hand.
- Predict the solar power for the current time slot t , for $t + 10$ minutes, and for $t + 30$ minutes ahead.

12.3 Evidence of completion

We demonstrate the completion of the overall objectives in the following subsections that corresponds to the above two sub-goals.

12.3.1 Learning solar power via DCNN

Recall that our goal is to train a DCNN that given an input sky image I_s predicts the solar power y_s . This can be formalized as minimizing the following objective:

$$\theta^* = \operatorname{argmin}_{\theta} \sum_{<I_s, y_s> \in \mathcal{D}} \|y_s - g(I_s, \theta)\|_p, \quad (12.1)$$

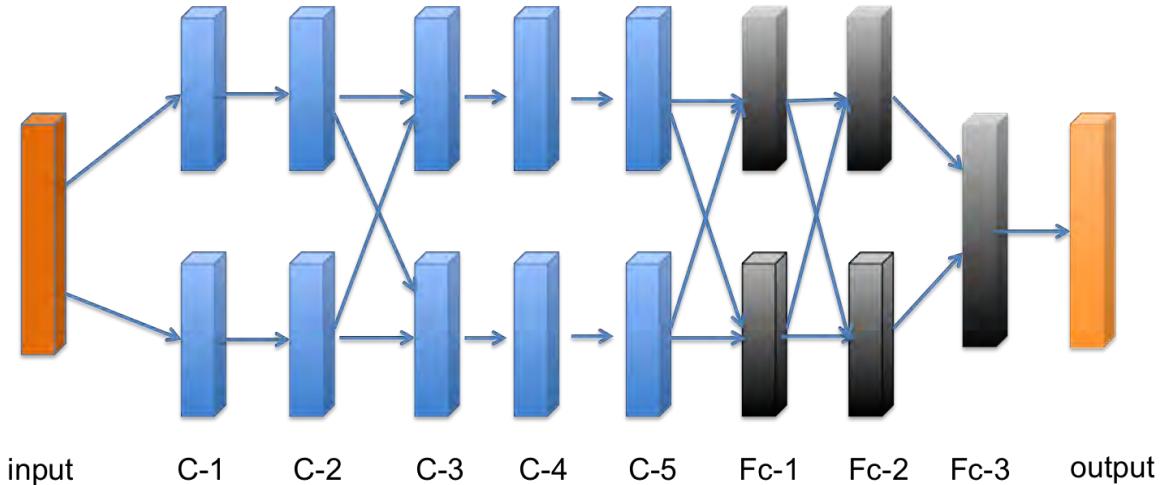


Figure 12.1: Architecture of Alex net [1]. $C-\{i\}$ denotes the convolutional layers and $Fc-\{j\}$ denotes the fully-connected layer. Details about the size of the network are referred to Table 12.1.

where \mathcal{D} is the set of training pairs, $g(\cdot, \theta)$ refers to the nonlinear function implicitly represented by DCNN whose weights, θ , we wish to optimize. $\|\cdot\|_p$ denotes the L_p norm¹.

12.3.2 DCNN structure

In the recent three years, DCNN have achieved the state-of-the-art performance in several computer vision research fields, especially in high level vision, such as image recognition, scene understanding and object detection. Recently, DCNN show promising performance in predicting geometric primitives such as depth maps and surface normals, which are continuous functions. Note that quite a few popular DCNN architectures are mainly adapted from Alex net [1] or VGG net [2], which are designed for image classification. We similarly tailor those basic network structures for the solar power prediction task. In the following two sections, we will first introduce their network structures designed for image recognition. We then show our adaptation to the solar power prediction task.

12.3.2.1 Alex net

An Alex net [1] contains five convolutional layers and three fully-connected layers. The structure is shown in Figure 12.1. In particular, $C-\{i\}$, $i \in \{1, 2, \dots, 5\}$, denotes the convolutional layers, $Fc-\{j\}$, $j \in \{1, 2, 3\}$, are the fully-connected layers. Moreover, the network also includes max pooling and normalization layers in order to realise a large receptive field and representative capacity. The size of the convolutional filters for each layer and the number of neurons for the fully-connected layers are shown in Table 12.1. Note that the network is composed of two parallel streams. The two streams only communicate in a few layers. It thus allows the implementation of the framework in parallel on two graphic computing units (GPUs).

Furthermore, they show that the depth of the network matters. Removing any one of the convolutional or fully-connected layers will result in inferior results. Recall that an Alex net is designed to address an image classification problem. In Table 12.1, the network will produce a 1000×1 probability vector for classifying the image to one of 1000 classes. Here, we are addressing solar power prediction which corresponds to a regression problem. The output will be a single scalar, namely the solar power value. We replace the $Fc-1000$ layer with $Fc-1$. The softmax layer is removed accordingly. We also change the loss function employed in the training process to the mean square error between the prediction and the ground truth value which reflects the energy function in Equation 12.1.

¹We use $p = 2$ in our current exploration, it could be changed to $p = 1$ as well.

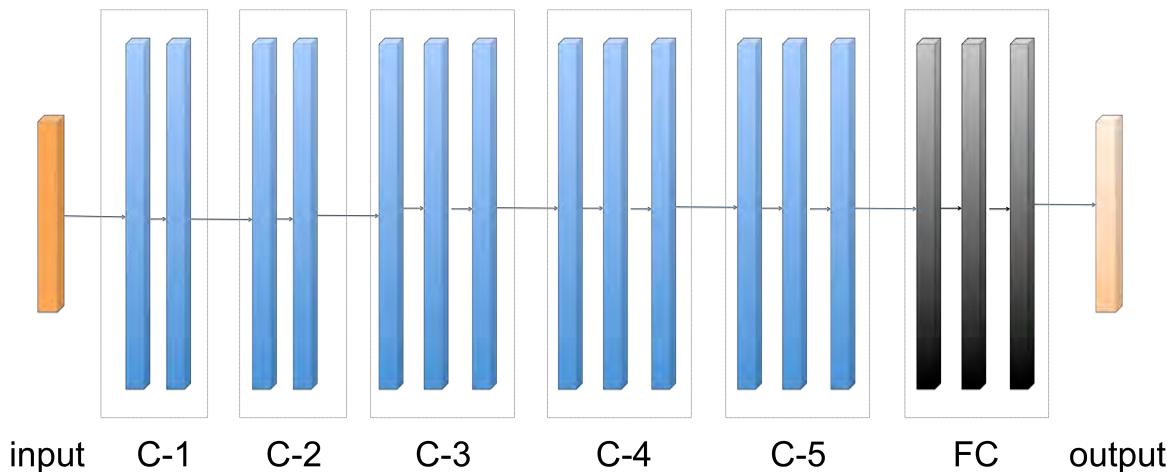


Figure 12.2: VGG net architecture. $C-\{i\}$ denotes the convolutional layers and FC denotes the fully-connected layers. Details about the size of the network are referred to Table 12.2.

12.3.2.2 VGG net

A VGG net [2] has a greater depth than an Alex net. Its architecture is shown in Figure 12.2. It consists of five groups of convolutional layers and one group of three fully-connected layers. The filter for each convolutional layer is of size 3×3 , which is the smallest size to capture the notion of left/right, up/down, center. The readers are referred to Table 12.2 for the details about the size of the filters and the number of neurons for each layer. Such a network performs better than an Alex net on the image classification problem.

Similar to an Alex net, an VGG net is also designed for image classification. In order to achieve the solar power prediction task, we do a similar net surgery as for the Alex net. We replace the Fc-1000 layer with Fc-1. The softmax layer is removed accordingly. We also employ the mean square error as the loss function.

ConvNet Configuration	
Input $224 \times 224 \times 3$	
Stream one	Stream two
C-1	conv11-48
	maxpool
C-2	conv3-128
	maxpool
C-3	conv3-192
C-4	conv3-192
C-5	conv3-128
Fc-1	Fc-2048
Fc-2	Fc-2048
Fc-3	Fc-1000
output	softmax

Table 12.1: Alex net configuration. The convolutional layer parameters are denoted as conv<receptive field size>-<number of channels>. The fully-connected layers are denoted as Fc-<number of neurons>.

12.3.3 Data preparation and solar power prediction

We evaluate the predictive power of the two different networks for solar forecasting. We test their performance on the images captured by the camera installed on the ANU Engineering Building. Owing to the lack of such a PV system at ANU/NICTA, we resorted to predicting the global radiations which are collected every **minute**.



	ConvNet Configuration
	Input $224 \times 224 \times 3$
C-1	conv3-64 conv3-64
	maxpool
C-2	conv3-128 conv3-128
	maxpool
C-3	conv3-256 conv3-256 conv3-256
	maxpool
C-4	conv3-512 conv3-512 conv3-512
	maxpool
C-5	conv3-512 conv3-512 conv3-512
	maxpool
FC	Fc-4096 Fc-4096 Fc-1000
output	softmax

Table 12.2: VGG net configuration. The convolutional layer parameters are denoted as conv<receptive field size>-<number of channels>. The fully-connected layers are denoted as Fc-<number of neurons>.

Dataset	t + 0 minutes		t+10 minutes		t+ 30 minutes	
	RMSE	AbsE	RMSE	AbsE	RMSE	AbsE
Train	178.8	139.2	179.0	137.4	294.7	246.8
Validation	111.1	85.3	129.6	104.6	208.2	183.0
Test	121.1	97.4	158.8	125.3	189.2	166.3

Table 12.3: RMSE and AbsE (Wm-2) of **global** radiation at ANU Engineering Building using network tailored from ALEX net. We evaluated on three subset, such as Training set, validation set and test set, respectively.

Note that the collection of the global radiation and the image is not synchronised. We thus match the pair based on the difference of their collection time less than a threshold, **20 seconds** in the current experiments. These pairs form the data for training, validation and test set of our networks at t . Similarly, the ground truth pair of data can be formed for $t + 10$ and $t + 30$ **minutes** in the similar manner. Due to the limited memory in our current GPU, we choose a set of data collected in 2015 for our evaluation. In particular, we collect 8296 valid pairs from March 1st, to April 30th for training, 3720 from May for validation and 3391 from June for test. Our evaluation is based on two error metrics, namely, root mean square error (RMSE) and mean of the absolute error (AbsE). We trained both the networks adapted based on ALEX net and VGG net. Due to the large amount of parameters of our network adapted based on VGG net (**134 million**), we cannot achieve a successfully training from random initialisations. Moreover, we found that it will produce a constant value when we fine-tune our network from a model trained on the large ImageNet, which is a collection of images for natural scenario. The failure is mainly due to images from Imagenet having different properties than sky images which mainly include the sky and clouds.

Since the network adapted from ALEX net requires a comparably small number of weights, we thus can successfully train the network from random initializations using the sky image data. The results are reported in Table 12.3. Compared with the results reported in Chapter 6, the network can successfully forecast the solar



energy in the same error magnitude. Note that our current network only trained from images without adding the time series and considering temporal information. In Table 12.3, the prediction errors for the training, validation and test set are reported to show that the network is not over-fitting on the training set. Based on the results, we can prove that the DCNN has promising performance in forecasting solar energy.

12.4 Extent to which the objectives were achieved

We have successfully proved that DCNN can be helpful to forecast solar power from sky images. The trained network can automatically learn useful features to predict the solar energy. We currently only test one single site (with one camera). We haven't included the temporal information to help forecasting the solar power for the future due to the DCNN is currently only for a single image not for a sequence. We believe that the objectives of the module have been largely achieved. However, the promising results presented above strongly suggest that DCNN are worthy to be further explored with the goal of improving the prediction of solar energy output from PV systems.

12.5 Highlights, difficulties and breakthroughs

In chapter 6, we have shown that the extracted visual features from sky images are helpful in solar power predictions. Note that those features are manually designed. In fact, we have tried a few designed features. Some of them demonstrate their usefulness when they are fused with other data modalities for the machine learning model. However, meanwhile, some designed features can fail in an unpredicted manner. In order to solve the problem, we propose to adopt the DCNN to let the network **automatically** learn the useful features from the images. Our primary results show that DCNN is helpful and allows the end-to-end training network. In addition, the DCNN can be expected to have multiple input data sources. This allows the inclusion of multi-modality data sources into one deep neural network in a seamless manner.

The major difficulty in our current approach is that the collection and training use a large amount of data in the DCNN. Moreover, ideally, the network should accommodate changes of the camera view point, which may rely on our camera calibration techniques proposed in 6. In addition, we need to further explore using time sequenced images into the network for solar power forecasting.

12.6 Conclusions and recommendations

In this chapter, we explore the DCNN architectures and tailor them for the solar power prediction. Our primary results show that DCNN is useful and helpful for solar energy forecasting. In the DCNN model, the network automatically learned useful visual features for the final prediction. It is an end-to-end network. The concerns from how to design the visual features rendered in 6 are resolved by using DCNN.

However, the DCNN we adopted cannot make use of image sequences for prediction, which is proven quite useful in traditional machine learning model. In addition, we currently work on a single site with the data captured from one camera. More training data is required to train a DCNN to be invariant to camera pose and locations. Therefore, new DCNN should be designed in order to solve the above listed problems.

Bibliography

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.



Australian Government
Australian Renewable Energy Agency

ARENA



Part IV

Outreach and Knowledge Sharing



Chapter 13

Outreach and knowledge sharing

13.1 Conference/journal papers

- F. P. Mills, N. Engerer, P. Duthie, M. Dennis, I. Skryabin, T. Steele, S. Fahey, J. Sullivan, and C. Jones, “Design of a solar radiation measurement network in Canberra, Australia”, International Conference Energy and Meteorology, June 2013, Toulouse, France.
- N. A. Engerer, “Minute resolution estimates of the diffuse fraction of global irradiance for southeastern Australia”, Solar Energy 116 (2015), 215-237.
- N. Ratner and S. Gould and R. Williamson, “Sky Imagery Prediction for Solar Power Production Applications”, preprint submitted to Solar Energy, April 2014.
- N.A. Engerer and F. P. Mills, “Validating Nine Clear-Sky Radiation Models in Australia”, Solar Energy, Accepted June 2015.
- N.A. Engerer “QCPV: A Quality Control Algorithm for Distributed Photovoltaic Array Power Output”. Submitted to IEEE Transactions on Sustainable Energy. July 2015.
- N. A. Engerer and F. P. Mills, “KPV: A Clear-Sky Index for Photovoltaics”, Solar Energy 105 (2014), 679-693, Elsevier, Amsterdam. J. Tan and N. A. Engerer and F. P. Mills, “Estimating Hourly Energy Generation of Distributed Photovoltaic Arrays: a Comparison of Two Methods”, Solar2014: The 52nd Annual Conference of the Australian Solar Council, 8-9 May 2014, Melbourne.
- F. P. Mills, N. A. Engerer and T. Steel, “Design considerations for an urban solar radiation measure network”, Solar2014: The 52nd Annual Conference of the Australian Solar Council, 8-9 May 2014, Melbourne.
- J. Domke, N. Engerer, A. Menon and C. Webers , “Distributed Solar Prediction with Wind Velocity”, IEEE Photovoltaic Specialists Conference 2016, Portland.

13.2 Conferences presentations

- 25 June 2013: Nick Engerer presented “Short-term Machine Learning Based Power Output Forecasts for Collectives of Rooftop Photovoltaics” at The International Conference on Energy Meteorology 2013, Toulouse, France, 25 - 28 June, 2013
- 17 October 2013: ANU PhD Expo — Nick Engerer (Best presentation award)
- 20 November 2013: Canberra Institute Technology Lecture
- 13 December 2013: ANU Solar Forecasting Lecture



- 25 – 28 June 2013: 2nd International Conference Energy and Meteorology (ICEM13), Toulouse, France
- 8 – 11 July 2013: Intersolar North America Conference, San Francisco

13.3 Trade shows

- 13 February 2013: Project presentation at “NICTA TechFest 2013”
- 28 May 2013: Project presentation at CeBit 2013
- 24 July 2013: NICTA Environmental Analytics Showcase, Brisbane, Queensland
- 30 July 2013: NICTA Environmental Analytics Showcase, Albury-Wodonga, NSW
- 21 August 2013: NICTA Smart Cities Innovation Challenge, Sydney, NSW
- 18 September 2013: NICTA Environmental Analytics Showcase, Melbourne, Victoria

13.4 Government and industry visits

- 9 December 2013: Visit of Jeremy Hanson, MLA, and Alistair Coe, MLA, to NICTA Canberra Research Laboratory
- 29 April 2014: Meeting with representatives from SolarHub, Armada Solar, Laros Technologies, ActewAGL, Infigen Energy, Telstra and AEMO
- 7 February 2014: Visit by of Suzanne Campbell, CEO, Australian Information Industry Association (AIIA)
- 19 February 2014: Presentation to ACT Government - NICTA Quarterly Meeting
- 28 February 2014: Visit to ActewAGL
- 9 April 2014: Visit to Telstra
- 12 March 2014: Visit to AEMO in Melbourne
- 22 May 2014: Visit to The Australian Institute of Marine Science (AIMS), Townsville
- 1 June 2014: Tour of the ANU Solar Facilities and explanation of project activities for a group of Scouts

13.5 Press releases

- 17.11.2015: ANU press release: “Volunteers wanted for solar energy study”
- 17.11.2015: ABC News press release: “Solar panels installed on homes across ACT needed by researchers for local power”

13.6 Student projects related to this project

1. Mr. Junyan Tan has been tasked with analyzing the SolarHub database and compiling information, which is useful for our project. Of particular importance is the exact metadata for each site — specifically the tilt, azimuth and system layout. This information allows for the development of automated routines for the curation and quality control of PV system data.
2. Thanard Kurutach, MIT, worked for three months on this project mainly supervised by Xinhua Zhang. He worked on “Probabilistic graphical models for distributed solar predictions with missing measurements”. Missing measurements or missing data can occur for a multitude of reasons, e.g.



- Failure in sensor hardware,
- Transmission failure,
- Recording failure, and
- Wrong manual data entry (detected by verification processes).

The missing data often occur together in a set with other valid measurements. Instead of throwing away the whole set of data (lots of information lost), or replacing the missing data by a predefined value (which one to choose?), machine learning tries to impute a reasonable data value which makes most sense in the context of the other valid measurements. Such a cleaned-up data set can then be used as input to other machine learning technology which relies on the assumption that all data values are valid.

3. Sonya Wellby, ANU, studied so called ramp events which lead to a rapid change in the solar PV production on a time-scale of minutes up to one hour. Only weather events that affect the whole of the A.C.T. (rather than individual suburbs) were analysed, as these have the greatest potential for disrupting the electricity network. Between January 2012 and July 2014, 35 of those events were identified. Positive ramp events (when solar PV output increases rapidly) were often morning events, and occurred most often when north-west cloud-bands moved away from the ACT, or morning fog dissipation was experienced. Negative ramp events (when PV output decreases rapidly) were often afternoon events, caused most frequently by thunderstorms or passing cold fronts. Most of these events last times of 12 and more hours and are well forecast by modern weather modelling technology. Efforts should be made to integrate weather forecast with solar PV forecasts, so that the electricity network can plan for these occurrences.
4. Lim Wei Lun Alan, ANU, studied three aspects related to solar energy generation. The first aspect is the different techniques used in solar forecasting. The second aspect is how the use of different time scales affects the measurement of variability of solar energy generated in an energy plant. This is done through ramp rate analysis and deriving coherence of data achieved for different time scales. The final aspect is how geographic smoothing can help reduce the variability in a solar energy plant, achieved by designing a large solar energy collecting area. This is done through wavelet analysis and fluctuation power indices. The results show that the data derived using large time scales is not reliable for measuring variability of solar insolation and that geographical smoothing can indeed be achieved through the use of a large collecting area.
5. Anthony Leung, ANU, explored the viability of cloud forecasting methods used by Bosch [1] on the ANU campus network. The lack of spatial density of solar measurement instruments on campus is the largest limiting factor and rules out Bosch's most correlated pair method that needs 8 spatially close sensors but the other suggested linear cloud edge method might still be feasible if clouds are sufficiently large. Solar irradiation data from two new trackers are also analysed. Direct, diffuse and global irradiation measurements have proved to be consistent between the trackers with an acceptable level of difference. However, possible calibration errors causing severe inconsistencies between global, direct and diffuse measurements have been found.
6. Shao Qi Lim and Anthony Chi Hin Leung: Solar radiation measurement – Analysis of intercomparison data from two co-located solar measurement stations installed on the ANU campus each monitoring diffuse, direct, and global surface irradiance.
7. Alan Lim: Solar energy: Methods of solar forecasting, irradiance variability due to clouds, and geographic smoothing – A review of “High-frequency irradiance fluctuations and geographic smoothing” (Lave et al, Solar Energy 86:2190, 2012) and “Solar forecasting methods for renewable energy integration” (Inman et al, Progress in Energy and Combustion Science 39:535, 2013)
8. Shaw Kudo worked on the comparison of PV energy prediction from 100 PV sites with the relevant sub-station data acquired from ActewAGL at three Canberra substations: Belconnen, City East and Wanniassa. Further comparisons are made with simulations from the REgional PV Simulation System (see chapter 11).