Intra-hour Photovoltaic Forecasting through a Time-varying Markov Switching Model

Karol Rosen[a,], César Angeles-Camacho[a], Víctor Elvira[b], Servio Tulio Guillén-Burguete[a]

[a]*Instituto de Ingeniería, Universidad Nacional Autónoma de México, México*
[b]*School of Mathematics, University of Edinburgh, United Kingdom.*

Abstract

This work presents a Markov switching model with a time-varying transition matrix to forecast intra-hour photovoltaic (PV) power output, aiming at providing forecasting flexibility. First, the proposed methodology captures images of the sky employing a self-made, low-cost all-sky imager. Second, the goal is to limit exposure problems in those images via the exposure fusion technique. Third, the proposed algorithm identifies groups of pixels forming clouds through a super paramagnetic clustering algorithm. Finally, we model the problem with a homogeneous Poisson process and forecast the cloud location and the shadowed area on a PV plant for the coming minutes. The shadowed area together with meteorological data are the inputs to this model. In the case study, our approach shows better performance than the persistence method, in particular for changing cloud conditions.

Keywords: Photovoltaic energy, forecasting, Markov switching models, all-sky images, clustering
PACS: 0000, 1111
2000 MSC: 62M10, 62M20, 62M30, 62P30

1. Introduction

High variability is one of the main difficulties with renewable energy resources, mainly photovoltaic (PV) power and wind generation. The power output of those energy resources depends essentially on random weather

⋆SPC: Super paramagnetic clustering. SBC: Single board computer.

conditions that, at least for now, humans cannot control. At the present moment, increasing penetration of PV and wind energy resources creates challenges to grid operators. One of those challenges is to guarantee energy supply with zero or minimum load loss. In some cases, grid operators have no other option than to keep backup (traditional) power generation plants working on to replace PV or wind sources when there is not enough generation available from them. The latter typically implies to keep burning fossil fuels.

Thus, accurate PV power output forecast have high relevance to help in the real-time operation of integrated systems with high penetration of distributed energy sources [1]. Furthermore, those forecasts help to reduce battery requirements in PV plants and also because grid codes recently imposed limits in the fluctuations of PV power output [2]. Moreover, accurate forecasting is essential for the real-time energy market decision makers, as well as in the integration of large-scale PV plants. Accurate PV output and load demand forecasts, both as a whole, help to add flexibility and better management in energy systems [3]. All the above, cannot be achieved without accurate PV output forecasts. In the end, all boils down to the economic welfare of nations [4].

The goal of this work is to propose an accurate intra-hour PV power output forecasting method. Another relevant goal is to be able to detect ramps. PV ramps usually are difficult to detect in advance. Markov switching models are beneficial to achieve this goal.

Model flexibility is an important feature to have in any forecasting model. For the Markov switching model of this work, extra flexibility comes from two sources: First, the hidden Markov chain selecting regimes is associated to a time-varying transition matrix. Second, values for this transition matrix depend on another variable that we generate based on nephograms (images of the sky and clouds) captured with an all-sky imager. To the best of our knowledge, those two aspects are novelties in PV power output forecasting methods. Then, the main contribution of this work is the proposal of a novel, flexible and accurate Markov switching model for intra-hour PV output based on nephograms, image clustering and Poisson point processes. It is worth to mention that, by construction, the proposed Markov switching model, generates probabilistic forecasts based on Markovian mixtures of probability distributions. Probabilistic forecasts provide much more information than a point forecast that is just a number.

The proposed framework works as follows. First, it captures nephograms. Then it applies the exposure fusion method [5] to the nephograms to limit under and over exposure. The third step is to identify groups of cloudy, clear-sky as well as pixels representing the sun in those nephograms. Pixel identification works via a super paramagnetic clustering algorithm. The fourth step estimates cloud locations minutes ahead through a homogeneous Poisson process. Finally, the outcome of those steps are combined with extra data in a Markov-switching model to generate intra-hour PV power output forecasts. Parameter estimation is based on the Kim filter, that can be thought of as an analogous form of the Kalman filter for linear Markov switching models. Our work shows that Markov switching models provide accurate PV power output forecasts.

1.1. PV power forecasting methods

Despite the vast existing literature, forecasting accuracy and ramp identification are still a challenge. In [6], [7] and [8] there are reviews of PV power forecasting methods. The reviews of [6] and [8], group forecasting methods in statistical and physical. According to them, any physical method is based on measurements of meteorological variables, including also numerical weather prediction and cloud imagery models. Hybrid methods combine more than one type of method. They also cover for different forecasting horizons, considering suitable methods for variable spatial and temporal resolutions. In [7] there is a review of forecasting methods coming from statistics and machine learning including supervised and unsupervised learning, clustering, and ensemble learning. In [**?**], there is a review about machine learning methods besides artificial neural networks, i.e, support vector machines, random forest among others, that could be also suitable in PV output forecasting. The authors compare the features and performance of those methods.

Among statistical and machine learning methods, [9] present a state-space exponential smoothing method. In [10], moving average and exponential smoothing methods aiming at irradiance day-ahead forecasts. In [11], they present a Gaussian weighted regression process to detect outliers. Another state-space model with continuous dynamic driven by the Wiener process appears in [12] relying on the extended Kalman filter for parameter estimation. A method for irradiance forecasting through probability distributions appears in [13]. In [14], the authors present a combination of artificial deep neural networks and long short-term memory algorithms aiming at intra-hour PV

power output forecasting.

In [15], a technique to generate short-term forecasts down scaling data in space and time is proposed. Input data comes from numerical weather prediction models intended for the medium and long terms. In [16], there is a method combining spatio-temporal correlation between a group of PV plants and satellite imagery. To schedule energy resources in remote microgrids, [17] presents a linear Markov switching model with three regimes based on Fourier expansions and considering irradiance and clear-sky irradiance past data. The last work mentioned here is [18], where we find a Markov switching model on the day-ahead prediction of bi-hourly irradiance.

The rest of the paper is organized as follows. Section 2 introduces a simple dynamic linear model with no change of regime to take as a comparison point for the model from Section 3. In Section 4, forecasting performance is evaluated for the models from Sections 2 and 3. Summary and conclusions of this work are presented in Section 5.

2. First approach: A state-space model

2.1. A simple dynamic linear model

Consider the following linear state-space or dynamic linear model,

$$y_t = F_t\theta_t + \nu_t,\ \nu_t \sim N(0, V_t), \tag{1}$$

$$\theta_t = \theta_{t-1} + \omega_t,\ \omega_t \sim N(0, W_t). \tag{2}$$

Equations (1) and (2) are known as the measurement (observation) and the state (evolution or system) equations, respectively. In this section and beyond, y_t represents the univariate variable PV power output. Parameter θ_t is a 4-dimensional state vector, ν and ω are noise parameters with zero mean and variance V_t and variance-covariance matrix W_t respectively with the appropriate dimensions. Note that $N(\mu, \sigma)$ represents a normal distribution with mean μ and variance σ. $F_t = (1, X_{t_1}, X_{t_2}, X_{t_3})$ represents a vector of covariates as we work with a dynamic regression model. This model is completed with an initial distribution for the state vector, $\theta_0 \sim N(\mu_0, \sigma_0)$. In this particular case, the model parameters are the variances V_t and, W_t plus the parameters of the initial distribution for θ_0, i.e., μ_0 and σ_0. The processes ν_t, ω_t are uncorrelated and uncorrelated between them.

For linear state-space models, the Kalman filter updates the probability distribution of the state vector at time $t+1$, $p(\theta_{t+1}|y_1, \cdots y_{t+1})$, from the probability distribution at time t, $p(\theta_t|y_1, \cdots, y_t)$ plus new incoming data at time $t+1$ [19, 20]. Note that this is possible whenever the model parameters are known. See for instance, [21, 22] and [23] for a review. In the end, the filtering distribution for θ_t at each time step, i.e., $p(\theta_t|y_1, \cdots, y_t)$, is just a normal distribution where its parameters are updated via the Kalman filter equations. The focus is on the filtering distribution of the state vector θ_t to then be able to forecast y_t (PV power output), as new data becomes available at each time step through Equations (1) and (2).

2.2. Covariates selection

Many factors affect PV power output, environmental, social, PV cells technology, system size, maintenance, and grid interaction [24]. The most important factors in PV cells performance are array degradation or lifetime, cells temperature, irradiance variation, and, shading, soiling, or dust [25]. Additional factors are the characteristics of the inverters, PV system batteries, and charge controllers [26]. PV cells typically work for wavelengths in the range $(0.25, 1.25)\mu m$ overlapping with the ultraviolet (UV) range, $(0.28, 0.4)\mu m$. However, for intra-hour performance, UV radiation only matters indirectly through the cells temperature [27]. UV radiation is significant for PV cells degradation though [28].

The most important covariate is irradiance followed by the PV cells temperature. PV cells temperature, $T_{PVCells}$, could be either measured directly on the cells or estimated. One common way to estimate this temperature is through the nominal operating cell temperature, T_{op}, commonly known as NOCT and the nominal operation irradiance, GHI_{op}. Parameters T_{op} and GHI_{op} are usually part of the specification sheet given by the manufacturer. For many manufacturers, $T_{op} = 45°C$ and $GHI_{op} = 800W/m^2$. The empirical equation to estimate the PV cells temperature is given by $T_{PVCells} = T_{air} + (GHI/GHI_{op})(T_{op} - 20)$, where T_{air} is the air temperature ($°C$) and GHI is the global horizontal irradiance (W/m^2) that a PV cell gets at any given moment.

Relative humidity is another important covariate, see Section 2.3.1.

2.3. Case study

We consider a real-world scenario with data from a PV power plant. There are around 10 PV plants at *Ciudad Universitaria* (CU) campus, *Universidad*

Nacional Autónoma de México (UNAM). Inside the campus, electricity is supplied through an underground distribution network of $23kV$.

A PV plant is at *Unidad de Posgrado* and is the one serving as case study in this work. This plant consists of 270 PV panels ($156mm$ Poly crystalline Silicon Cells) of $250W$ each one, oriented from North to South with a maximum power generation of $67.5kW$. There are no buildings or any other objects blocking sunlight. This system works with three inverters, each one connected to 90 PV panels, conforming 5 arrays in series by 18 in parallel. In this way, each inverter gets a maximum voltage of around $180Vdc$ with a maximum power of $22.5kW$.

2.3.1. Data acquisition

Data recollection started on November 25 and finished on December 21, 2021, measuring different electric variables through a power quality analyzer model "PowerPad III 8336". This analyzer was configured to get data every second registering from the electric board at the output of the inverters.

The *Red Universitaria de Observatorios Atmosféricos* (RUOA) from UNAM [29], kindly shared with us measurements for a bit more than 10 meteorological variables from November 25 to December 21, 2021, registering data every minute. The meteorological variables measured at RUOA come from professional high-end quality instruments, providing high accuracy and little noise in the data.

A correlation analysis between PV power and all those meteorological variables shows that relative humidity has a negative correlation with PV power, especially for values above 40%. Besides that, also air temperature and Global Horizontal Irradiance (GHI) data are required, as explained in Section 2.2. RUOA has a meteorological observatory at CU campus. However, it is located around 2 *km* away from the PV plant at *Unidad de Posgrado*. Results from this model appear in Section 4.

3. A Markov switching model with time-varying transition matrix

This section presents the regime switching model, generalizing the model from Equations (1) and (2). The main advantage is the addition of extra models, one for each regime [30, 31, 32]. Each regime is chosen depending on the value of a hidden or latent (thus not observable) categorical variable S_t at time t. Modeling S_t as a hidden Markov chain. In particular, for sudden changes or ramps in

PV power output, those models seem suitable. Regime switching models have been applied in a wide range of subjects, for example, in sociology, and finance [33], [34].

3.1. Linear Gaussian Markov switching model

Consider the following linear Gaussian Markov switching model with two regimes.

- Regime 1:

$$y_t = m_1 + B_1 x + \theta_t, \tag{3}$$
$$\theta_t = \phi_1 \theta_{t-1} + \omega_t, \tag{4}$$

  where $\omega_t \sim N(0, W)$.

- Regime 2:

$$y_t = m_2 + B_2 x + \theta_t, \tag{5}$$
$$\theta_t = \phi_2 \theta_{t-1} + \omega_t, \tag{6}$$

  where $\omega_t \sim N(0, W)$.

Parameters m_1 and, m_2 are intercepts, $B_1 = (\alpha_1, \beta_1, \gamma_1)$, $B_2 = (\alpha_2, \beta_2, \gamma_2)$ are vectors of regression weights for the variables in the covariates vector x. We consider the same covariates as in Section 2.2. Subscripts 1 and 2 in the parameters refer to each one of the two regimes. Parameter ϕ is regime-specific, there is a heterogeneous variance for the latent parameter θ despite that the noise $\omega_t \sim N(0, W)$ is common in both regimes with constant variance W. Inside each regime, Equations (3) and (5) are known as the measurement or observation. Analogously, Equations (4) and (6) are known as the dynamic equations. A similar model has been applied in [35] on a different subject.

3.1.1. Time-varying transition matrix

The transition matrix P driving the Markov chain S_t over time is

$$P = \begin{pmatrix} \frac{\exp(k_{11})}{\exp(k_{11})+\exp(0)} & \frac{\exp(0)}{\exp(k_{11})+\exp(0)} \\ \frac{\exp(k_{21})}{\exp(k_{21})+\exp(0)} & \frac{\exp(0)}{\exp(k_{21})+\exp(0)} \end{pmatrix}. \tag{7}$$

Parameters k_{11} and, k_{21} represent log-odds instead of transition probabilities. For identification purposes, the second state (column) is the reference state

setting its log-odds to zero as shown in the matrix P above, [35]. To construct a time-varying transition matrix, the log-odds parameters are represented in a regression model depending on variable z_t at time t, i.e., $k_{i1}^t = c_{i1} + d_{i1}^t z_t$, $i = 1, 2$. Thus, $c_{i1}, d_{i1}^t, i = 1, 2$ are additional parameters to estimate. In the end, this model needs the estimation of 15 parametres. To complete the model description, an initial distribution is needed to initiate the two state Markov chain S_t at time $t = 0$.

As explained in Section 3.5, variable z_t is defined as the difference in the measured and estimated shadowed area on the PV plant at time t. The estimated and measured shadowed areas come from processing nephograms, Sections 3.2 to 3.5.

A more compact way to represent Equations (3) to (6) is

$$y_t = m_{S_t} + B_{S_t} x + \theta_t, \tag{8}$$

$$\theta_t = \phi_{S_t} \theta_{t-1} + \omega_t, \ \omega_t \sim N(0, W), \tag{9}$$

where subscript S_t indicates that parameters m, B and ϕ are dependent on the value of the 2-states Markov chain S_t at time t.

As this is a linear regime switching model, parameter estimation is done employing the Kim filter, [36]. This filter is a combination of the Kalman and Hamilton filters with a collapsing procedure to reduce the computational burden. The Kim filter is analogous to the Kalman filter for linear regime switching models. For more elaborated regime switching models, for example, with non-linear dynamic equations, estimation relies on the extended Kim filter and Kim smoother [34].

The Markov switching model as well as the dynamic linear model from Section 2, are probabilistic forecasting methods, designed to work with data from the current day. The only exception could be at sunrise when there is not enough data available to estimate parameters. Then, the methods rely either on artificial data or real data from other days with similar conditions at sunrise. As new data becomes available at each time step, both models update its parameters through the Kim and Kalman filters, respectively. Figure 1 shows the flow chart of the Markov switching model. In this flow chart, we define three main stages, Acquisition, Processing, and Forecast. The first two stages, highlighted in green and blue dashed rectangles, are useful to compute the value of the variable z_t.

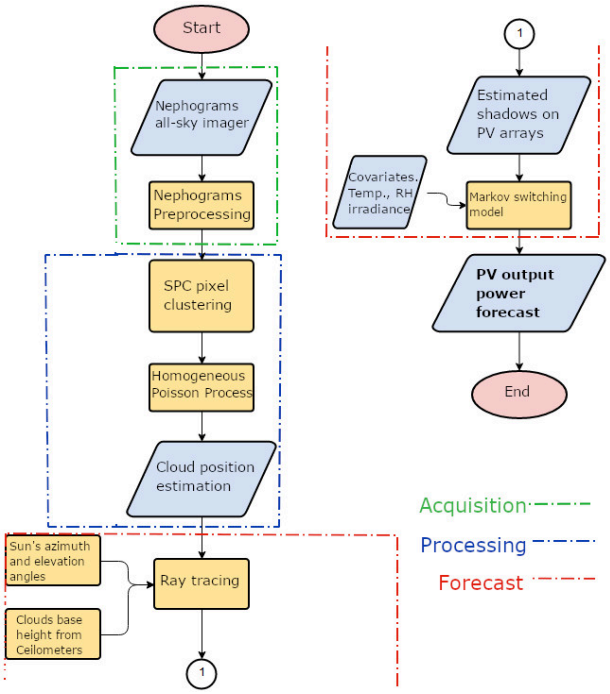

Figure 1: Markov switching model flow chart.

3.2. Additional data

On top of the data from Section 2.3.1, the model needs another variable to incorporate in the Markov switching model of this section, namely, the cloud base height. RUOA also kindly shared data for this extra variable from November 25 to December 21, 2021. Cloud base height was generated by measurements of a Vaisala ceilometer CL31 in 10 minutes averages of the Mixed Layer Height from the gradient method [37]. This ceilometer is also part of the meteorological observatory at CU mentioned in Section 2.3.1. Finally, the proposed model requires nephograms captured at the PV plant of interest, as explained in Section 3.3. Cloud base height data is incorporated for ray tracing, Section 3.5.

3.3. Image acquisition and pre-processing

First, nephograms were captured through a self-made and low-cost all-sky imager placed at 20 *m* from the PV plant. The all-sky imager consists of a Single Board Computer (SBC), a Raspberry Pi 4, equipped with a camera and a fish-eye lens with a vision angle of 220° inside an enclosure with a transparent dome. For most of the data recollection period, this device produced a final image every six

minutes. Every day, image capture started at sunrise and finished at sunset. Figure 2 below, shows the all-sky imager and part of the PV plant in the background.

Figure 2: All-sky imager and a section of the PV plant at *Unidad de Posgrado*.

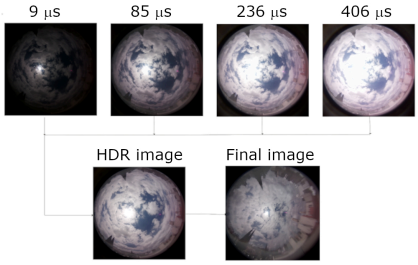

Figure 3: Captured images at four different exposure times, fused (HDR) and final images.

Based on a technique called exposure fusion, [5] the aim is to limit exposure problems in the nephograms. This technique selects regions of pixels from a few images, four in our case, captured at different exposure times to form a fused image. The fused image is similar to a High Dynamic Range (HDR) image.

Thereafter, the distortion introduced by the fish-eye lens is removed to get the final image. Figure 3 shows an example of it. Distortion is more noticeable for pixels that are farther away from the center of the image. Fish-eye lenses follow the pinhole model, i.e., the projection of every straight line is a straight line [38]. Removing distortion is nothing more than a change of coordinates. The all-sky imager can generate a final image at a maximum rate of around one per minute and a half.

Following [38] and through images from a 9x6 checkerboard in different positions and angles, it was possible to estimate the parameters for the change of coordinates to remove distortion from the final images.

All images, captured, fused and final, have a resolution of 1728x1728 pixels. Final images contain the date and time when it was generated as part of its file name. The proposed model works with raw BGR (Blue-Green-Red) images where data is interleaved. This means that blue, green, and red values for a given pixel are grouped together, in that order. Image acquisition and pre-processing was done straight in the all-sky imager through own code in *Python3* and its *openCV* library.

3.4. Pixel clustering and cloud location

Due to unavoidable constraints at *Unidad de Posgrado* plus the fact that the fish-eye lens has a vision angle of 220°, the all-sky imager captured part of a nearby building, the upright of a security camera and even part of the horizon in all final images. None of those elements affect the PV plant and are only appearing in the nephograms. Before proceeding with pixel clustering, each final image was processed as follows.

First, cutting the final images to correspond to a vision angle of 170° through measurements coming from a digital sextant. Second, cleaning the cut images by smoothing the Red-Green-Blue color channels values, using the 3-nearest neighbors, removing the building and the upright of the security camera. Third, identifying the sun center position in the clean images, noting that this position changes only a few tens of pixels at similar times during the whole data recollection period. Thus, to simplify, we consider the sun center position as fixed for the measurement campaign. Choosing as a reference the sun center positions in every image captured on December 3, 2021, we formed a list of 39 elements for different times from sunrise to sunset. Images from different days do not necessarily were generated at the same time. Then, the sun center position was taken from the element of that list that is closest in time to the image we want to work on. Finally, defining the *Sun neighborhood* as a square of size 210 pixels centered at the sun center position. The main reasons to work with the *Sun neighborhood* image are:

- Regions of the sky away from the sun center mainly add diffuse radiation, reducing the amount with its distance to the sun center.
- A smaller image reduces the computational effort for pixel clustering, as there are fewer pixels to classify.

The next step is to process the *Sun neighborhood* images in a second computer to identify groups of pixels representing clouds, the sun and clear-sky.

3.4.1. Pixel clustering through Super Paramagnetic Clustering

A Super Paramagnetic Clustering algorithm (SPC), [39], [40] classifies pixels from the *Sun neighborhood* images. This SPC algorithm has its origins in the ferromagnetic Potts model from Statistical Physics, [41]. More precisely, from interacting particle systems [39]. The goal is to model and artificial ferromagnetic spin-spin interaction.

At low temperatures, the *ferromagnetic phase*, spins are ordered with all spins sharing the same value, spins are aligned. When heating it up, the system moves to the *super paramagnetic phase*, spins tend to form clusters of aligned spins. In this phase, spins in different clusters are independent, but spins inside a cluster are highly correlated. At high temperatures, the *paramagnetic phase*, spins are disordered, and all spin values are independent.

SPC defines an artificial ferromagnetic Potts model to try to take advantage of the super paramagnetic phase to identify clusters in a data set. Given a local interaction function and a fixed number of possible spin values, the SPC algorithm proceeds as follows [39]:

1. Randomly assign a spin to each pixel data from the q available values.
2. Identify the temperature ranges for phase changes, via the susceptibility. Susceptibility is proportional to the variance of the magnetization and is estimated via the Swendsen-Wang Monte Carlo algorithm [42]. The super paramagnetic phase is located between the temperatures where the susceptibility peaks (T_{peaks}) and where it becomes zero at high temperatures (T_{nul}), [40].
3. At any temperature in $[T_{peaks}, T_{nul}]$, estimate the spin-spin correlation [40] to identify clusters. Groups of spins having correlation greater than some predefined fixed value, for example 0.5, are part of the same cluster.

In its simplest form, the interaction intensity between spins depends on the distance between them, see Section 4.2.2. Instead of considering only the distance, we could replace it by a similarity measure incorporating other features relevant for the data set.

3.4.2. Poisson process for cloud location

After forming pixel clusters to identify cloudy regions in the sky, the next step is to define a model for cloud location for the next minutes. Here, we work under the *simplified assumption that the process of cloud forming and its location is a Completely Spatially Random (CSR) process,* i.e., there is no obvious interaction structure in the cloud forming process leading to a homogeneous Poisson process [43], [44]. Another option is to try to check, for example, through distance analysis [44], if a homogeneous Poisson process fits the data set. If that is not the case, move to more general processes as non-homogeneous Poisson or log-Gaussian Cox processes, where the intensity function is not constant, or it is random, respectively. Those more general settings are not considered in this

work.

A homogeneous Poisson process $Z \in \mathbb{R}^d$ with constant intensity function λ, satisfies the next two technical conditions,

1. The number of points inside a set A, $N_Z(A)$, is a Poisson random variable with mean $\lambda \mid A \mid$ for every bounded Borel set $A \subset \mathbb{R}^d$. The d-dimensional volume of set A is denoted by $\mid A \mid$.
2. For any k, disjoint bounded Borel sets $A_1, \cdots, A_k, k \in \mathbb{N}$, the random variables $N_Z(A_1), \cdots, N_Z(A_k)$ are independent.

In this work, $d = 2$ and the intensity function at each time step t is estimated as the ratio of the number of cloudy pixels (N_{cl}) to the total area (A) in the *Sun neighborhood* image, $\lambda_t = N_{cl}/A$. Recall that N_{cl} comes from the analysis in Section 3.4.2.

3.5. Ray tracing

Ray tracing helps to project the sunlight to a point on the surface of the earth. Sunlight projection requires a precise location on the earth, its latitude and longitude geographical coordinates. In addition, the sun position, given by the zenith (or altitude) and azimuth angles for a specific date and time. As the interest lies in projecting the clouds on the PV panels, measurements of cloud the base heights are needed too. Cloud base heights can be obtained with ceilometers as mentioned in Section 3.2. The ray tracing procedure in this work closely follows [45].

Figure 4 shows that the cloud base radius $R = H \tan(90^\circ - A)$. Here, H is the cloud base height and A is the vision angle of the fish-eye lens in the all-sky imager. If this angle is greater or equal than 180°, only the region corresponding to an angle slightly less than 180° is considered. We set $A = 170^\circ$ as explained in Section 3.4. A pixel size in a selected unit of length, is $P_{size} = R/r_{img}$, where $r_{img} = Res/2$. Here, Res is the resolution of the *Sun neighborhood* images given in pixels. For us, the resolution of those images is 210x210 pixels, therefore, $r_{img} = 105$ pixels.

Solar position angles, zenith and azimuth, are easily obtained, for example, through the SOLPOS calculator [46] given a date, time, and a location on the earth. Another required asset is an overhead image of the place of interest to be able to locate the shadows on PV panels.

Ray tracing geometry is represented in Figure 5. The bottom layer is an overhead image of the place of interest, and the top layer is an image with the estimated future cloud location at that same place. The goal is to project the top layer on the bottom layer. The list below presents the ray tracing steps, refer also to Figure 4.

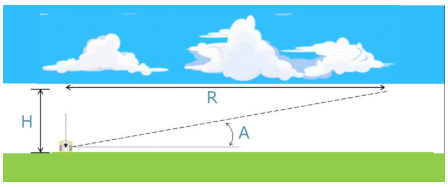

Figure 4: Cloud base radius.

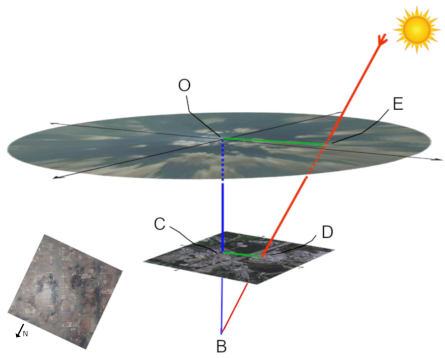

Figure 5: Ray tracing geometry.

1. Point C represents the location where we place the all-sky imager and determines the origin of the coordinate system. Point O, the center of top layer, is aligned with point C.
2. Project a point E from the top layer on the bottom layer. Identify the distance in pixels between points O and E in the top layer. The real distance equals that distance in pixels multiplied by P_{size}, see Figure 4.
3. Identify the distance between points C and D in the bottom layer. $CD = OE - OC\tan(\angle OBE)$. This expression comes after combining the next three equations. First note that $CD = CB\tan(\angle OBE)$ and $CB = OB - OC$, where OC is known as it is the cloud base height. On the other hand, $OB = \frac{OE}{\tan(\angle OBE)}$. Observe that the angle $\angle OBE$ is the zenith angle for a given latitude, longitude, date, and time. Get the value of the zenith angle, for example, from the SOLPOS calculator.
4. Coordinates of point D are, (r, θ) in polar coordinates, and $(r\sin\theta, r\cos\theta)$ in Cartesian coordinates. Here, $r = CD$ and θ is the azimuth angle for a given latitude, longitude, date, and time. Get the value of the azimuth angle from the SOLPOS calculator.

Taking Figure 5 as reference, projecting in this way all points from the top layer (estimated near future cloud position layer) on the bottom layer (region containing PV panels). Thus, computing the shadowed area on those PV panels.

4. Results

The interest is in intra-hour forecasting of PV power output. To have a flexible model, we estimate new model parameters *every day*, also updating it as new data arrives along the day for both models, from Sections 2 and Sections 3. Forecasting results of the dynamic linear and linear Gaussian Markov switching models appear in Sections 4.1 and 4.2 respectively. We are convinced that there is no universal way to measure forecasting accuracy, in particular in PV power output. Instead, anyone interested in this topic needs to define the most relevant aspects to then be able to measure forecasting accuracy, even with tailor-made metrics if necessary.

As an example, the Root Mean Square Error (RMSE) is a common way to measure forecasting accuracy in general. However, the RMSE scale, is dependent on the scale of the data. Outliers affect it more than other accuracy metrics, as the Mean Absolute Error (MAE) [47]. Percentage errors are also common to measure forecasting accuracy, for example, the Mean Absolute Percentage Error (MAPE). The scale of percentage error metrics is not dependent on the scale of the data. Although, are only suitable for data having an absolute zero, i.e., data is in a ratio scale instead of only an interval scale, [47]. The scale of relative errors is not dependent on the scale of the data, but could have even infinite variance.

In [47], the authors generalize the scaling of relative errors to define new scaled error metrics which scale is not dependent on the scale of the data. The Mean Absolute Scaled Error (MASE) is considered by those authors as the best available forecast accuracy metric. Particularly, when working with data in different scales and with data sets with negative or close to zero values. Assuming there are N data points, y_t is a measurement and $\hat{y}_t$ its forecast, each forecast error, $e_t = y_t - \hat{y}_t$, is scaled as

$$q_t = \frac{e_t}{\frac{1}{N-1}\sum_{t=2}^{N} |y_t - y_{t-1}|} \tag{10}$$

Thus, MASE is the mean of all scaled errors $|q_t|$ given by Equation (10). In MASE, all errors are scaled with respect to the MAE of the naive forecast, i.e., $\hat{y}_t = y_{t-1}$. In MASE, the naive forecast is typically taken as the reference forecast. Extensions of the RMSE and other common error metrics, not dependent on the scale of the data, can be defined similarly. In this work, we evaluate forecasting accuracy through scaled error metrics as MASE. Note that

the persistence method, widely applied in PV output forecasting, is non-other than the naive forecast. Thus, the persistence method will serve as a benchmark forecast method.

In the results appearing below, both models from Sections 2 and 3 are tested on December 18, 2021, as that day was the one showing the highest variability in PV power output for the case study of this work, see Section 2.3. Days with high variability in PV output are the most difficult to forecast and are usually linked to rapid changing conditions in clouds. Both models, dynamic linear and Markov switching, are implemented in *R version 4.2.2 (Innocent and Trusting)* in a personal computer with processor Intel(R) Core(TM) i5-3337U CPU @ 1.80GHz.

Sections 4.2.1 to 4.2.3 are here to explain how to obtain the values of covariate z at each time step. Recall that this covariate drives the time-varying transition matrix of Markov chain S_t as part of the proposed Markov switching model, see Section 3.1.

4.1. Dynamic linear model

On December 18, 2021, there are 642 data points registered every minute from sunrise (7:10 hours) to sunset (17:51 hours). Measured variables are PV power output, relative humidity, GHI and PV cells temperature as described in Section 2.2. However, we only consider the same 44 data points as in Section 4.2. The plan is to test both models with the same data set. Parameters of the model from Section 2.1 are estimated through the *dlm (dynamic linear model)* package for *R* [20]. The *dlm* package estimates model parameters for dynamic linear models through the Kalman filter that is already part of that package.

The initial covariance matrix W_0 is a diagonal matrix with values $(0.4, 0.2, 0.4, 0.2)$ and variance $V = 1.3$, both were estimated from data in the period November 26 to December 9, 2021. The initial distribution for $\theta_0 \sim N(\mu_0, \sigma_0)$ with $\mu_0 = (0, 0, 0)$ and σ_0 is a diagonal matrix with values $(10, 10, 10, 10)$.

Figure 6 shows two interesting points: 1) a drop in PV power output between 13:00-15:00 hours, a big discrepancy in measurements and forecast, 2) at 11:36 hours, $1.9kW$ (measured) and 13.5 kW (forecast) and, 3) at 15:32 hours, $14.7kW$ (measured) and $23.9kW$ (forecast).

The first point above is explained as between 13:00-15:00 hours, the sky started to be covered by clouds, even with a bit of rain. This drop in PV power output is captured by the model. We confirmed in the data a drop in GHI, also founding that the sky was covered with gray clouds in the nephograms that we captured in that range of time.

For the second point above, we have no clear explanation for the drop of PV power output. This drop lasted close to two minutes to then recover its previous level. Data shows almost no change in the covariates, in particular GHI, which was almost constant at $650W/m^2$ from 11:30-11:42 hours. Nephograms show no clouds blocking the sun. Perhaps a short duration failure that was detected by the inverter protection system or an external agent, for example, a bird or some leafs moved by the wind, were the source of the dropping of PV power at this particular time.

Finally, for the third point above, the forecast identifies the peak of the measurements, however, with a higher value. Data shows variability in GHI and nephograms also show changing cloud conditions around 15:32 hours.

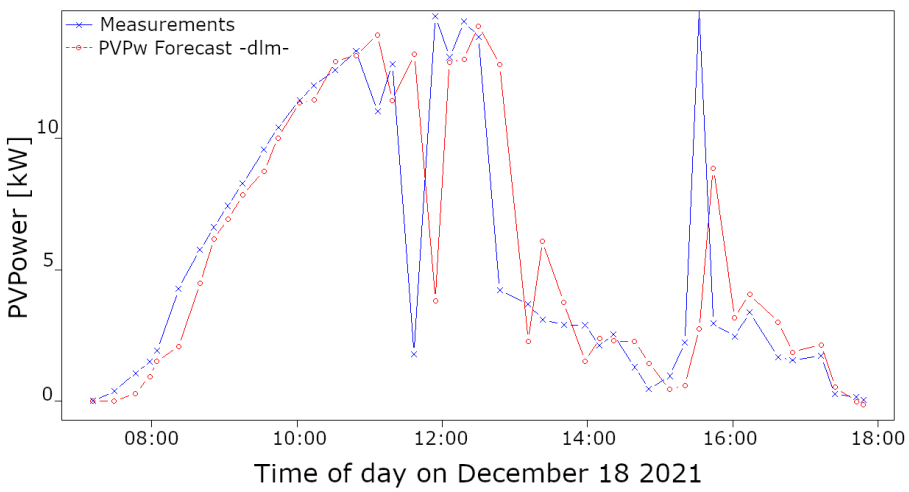

Figure 6: PV output measurements (blue) and forecast (red) from the dynamic linear model of Section 2.

Global forecasting accuracy metrics, as MASE, are not necessarily the best option when the focus is to detect ramps in PV power output. Another option is

to rely on scaled forecast errors, for example, the scaled errors from Equation (10). There are two main reasons supporting this claim. First, scaled errors are not dependent on the scale of the data. Second, the performance of a forecasting method can be easily seen on a simple plot of the scaled errors against time. Particularly useful to evaluate ramp detection in a forecasting model. The perfect forecast is a horizontal line placed at zero.

Figure 7 presents the scaled forecast errors, q_t, from Equation 10. As in Figure 6, higher errors are located around noon, 13:00 and 15:32 hours. At the rest of the times, the scaled errors are near zero. Forecasts values less than measurements appear as scaled errors greater than zero. Forecasts values greater than measurements appear as scaled errors less than zero.

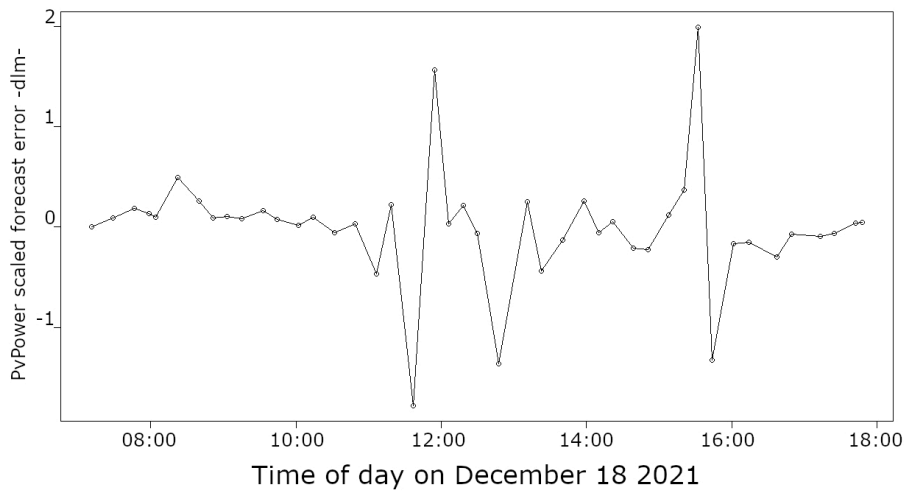

Figure 7: Scaled forecast errors q_t. Dynamic linear model from Section 2.

4.2. *Linear Gaussian Markov switching model*

We decided to consider data every 15 minutes, only 44 data points from the 110 available, as we want to reproduce a real operation forecasting procedure given the computational power at our disposal. The whole process at each time step takes between 10 and 13 minutes given the constraints described just before Section 4.1. Thus, considering as if data were registered approximately every 15 minutes from sunrise (7:10 hours) to sunset (17:51 hours) on December 18, 2021. Measured variables PV power output, relative humidity, GHI and PV

cells temperature as described in Section 2.2. Note that in Section 4.1 there are 642 data points available instead, data registers every minute. For this Markov switching model, we incorporate data from ceilometers as well as the nephograms from Sections 3.2 and 3.3. As a downside, this

Parameter estimation for this Markov switching model, see Section 3.1 was done through the *dynr (dynamic modeling in R)* package. The *dynr* package estimates model parameters for regime switching models via the Kim filter that is already part of that package. We set generic initial values for all model parameters as in Section 4.1. All 15 parameters are re-estimated when new data arrives.

Discrepancies between measurements and forecasts also appear in Figure 8 at 11:36 and at 15:32 hours. As explained on p. 15-16, the data set (GHI, nephograms and covariates) does not correspond with the drop in PV power output at 11:36 hours. Interestingly, at 15:32 hours, the forecast is closer to the measurement than it is in Figure 6. Opposite to Figure 6, in this case the forecast is closer to the measurement. The rest look pretty similar on both figures.

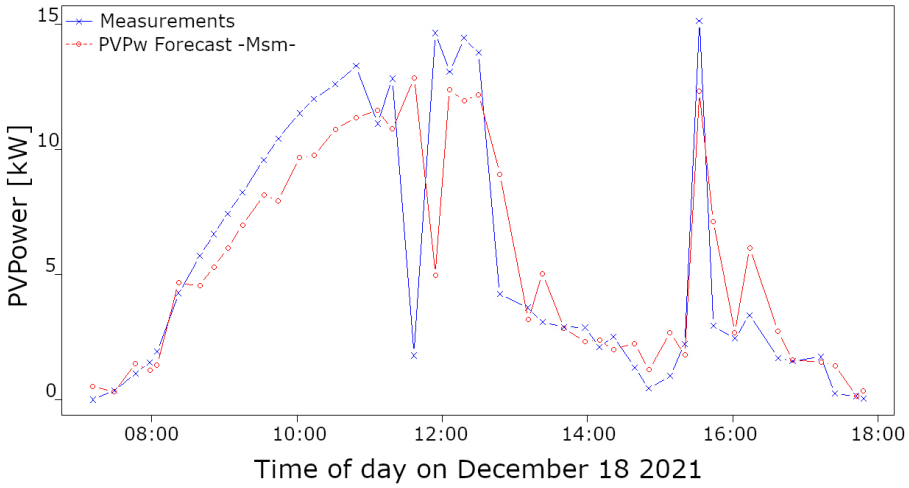

Figure 8: PV output measurements (blue) and forecast (red) from the linear Gaussian Markov switching model of Section 3.

From the evolution of the latent variable θ at the top of Figure 9, we clearly identify a close correspondence in the drops and rises from Figure 8 with the change of regimes from Figure 9. This is actually a good sign, as the goal is to model low PV power output with one regime and high PV power output with the other one. Parameter estimates, at the end of the day, excluding the time-varying transition matrix of the Markov chain S_t, appear at the bottom right of Figure 9.

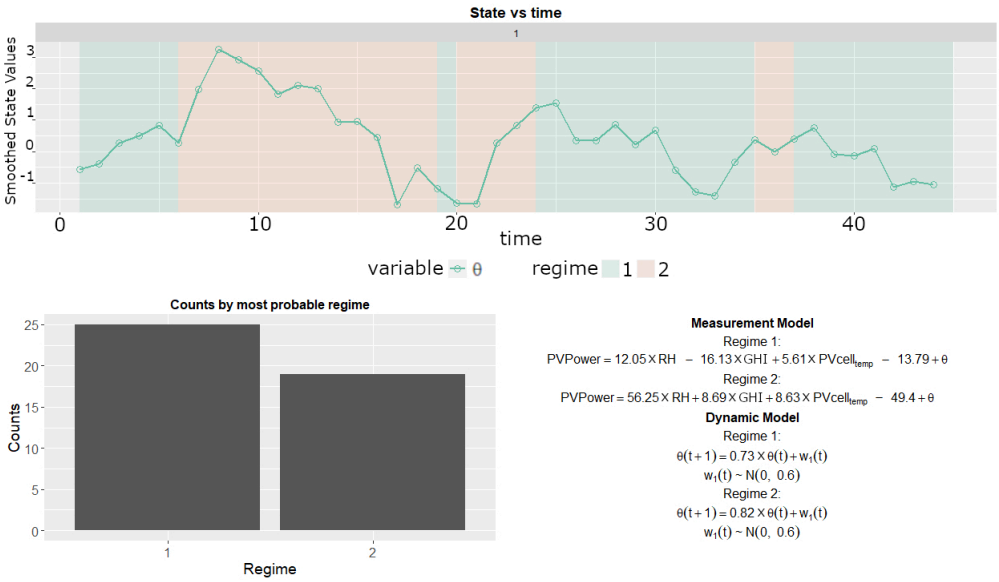

Figure 9: (Top) Evolution of the latent variable θ, indicating regimes. (Bottom left) Histogram with the number of times that the model was in each regime. (Bottom right) Linear Gaussian Markov switching model of Section 3 with parameter estimates at sunset.

Comparison between Figures 7 and 10 shows similar performance in the scaled forecast errors. The scaling follows Equation (10). Only at 11:36 hours, the scaled error is around two times higher in Figure 10 than in Figure 7, see p. 18.

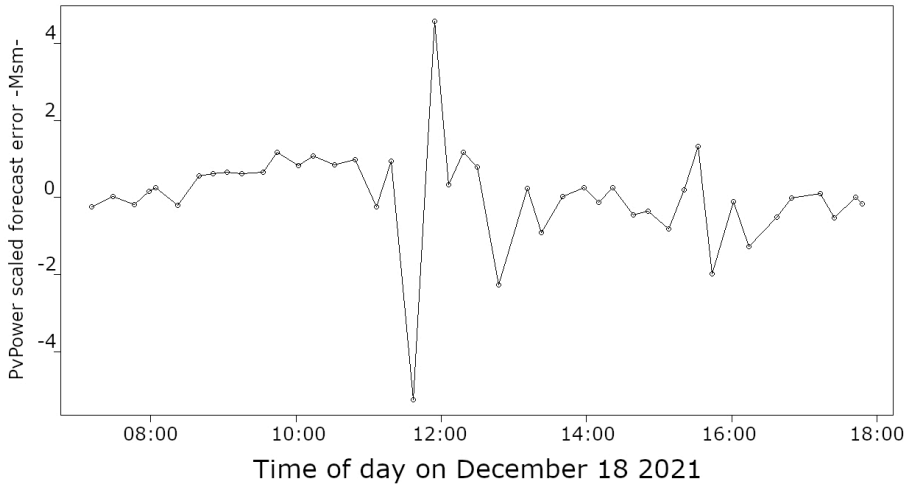

Figure 10: Scaled forecast errors q_t. Linear Gaussian Markov switching model of Section 3.

Global forecasting performance is easy to interpret for scaled error metrics. Values less than 1 indicate better (smaller) performance errors on average in the proposed forecast method than in the reference forecast. Values greater than 1, indicate worse (bigger) performance errors on average in the proposed forecast method than in the reference forecast.

Table 1 shows the MASE forecast error metric for both proposed models. The first two rows take the naive or persistence forecast as reference. The third and fourth rows take the dynamic linear model forecast as reference. According to MASE, the dynamic linear and the Markov switching models perform better on average than the naive forecast. The Markov switching model also performs better on average than the dynamic linear model forecast.

Error metric	Dynamic linear model (dlm)	Markov switching model	dlm [a]
MASE [b]	0.861	0.779	1.167
RMSSE	0.007	0.036	0.013
MASE [c]	1	0.905	0.284
RMSSE	1	4.817	0.618

Table 1: Forecasting errors metrics performance on December 18 2021.

[a] Considering all 642 available data points.
[b] Rows 1 and 2 take the naive forecast as reference.
[c] Rows 3 and 4 take the dlm forecast as reference.

For completeness, in Table 1 we add the performance of the dynamic linear model considering all 642 data points and the Root Mean Squared Scaled Error (RMSSE) for the tested models. Under RMSSE, both proposed models also perform better on average than the naive forecast. However, under RMSSE, the Markov switching model performs worst on average than the dynamic linear model.

The RMSSE equals the square root of the mean of all scaled errors q_t^2. Assuming N data points, y_t a measurement and $\hat{y}_t$ its forecast, each forecast error, $e_t = y_t - \hat{y}_t$, is scaled as

$$q_t = \frac{e_t}{\sqrt{\frac{1}{N-1}\sum_{t=2}^{N}(y_t - y_{t-1})^2}}. \tag{11}$$

Note that the scaling is just the RMSE of the naive forecast. The naive forecast is known the persistence forecast in PV forecasting.

The next subsections aim at obtaining the value of covariate z, that drives the time-varying transition matrix of the Markov chain S_t from Section 3.1.1. S_t chooses the regime of the linear Gaussian Markov switching process model at each time step. The implementation was done in *R version 4.2.2* through own code and a few of its packages.

4.2.1. Image acquisition and pre-processing

Image acquisition and pre-processing is explained in Section 3.4 showing in Figure 11 a sample of the final, cut, and *Sun neighborhood* images at 11:18:53 hours. The final image appears on the left side as it comes from the all-sky imager. In the center, is the cut image now corresponding to a vision angle of 170° plus having removed the interfering objects. The *Sun neighborhood* is at the right and is the image to cluster pixels from.

4.2.2. Pixel clustering and cloud location

Pixel clustering follows the steps of Section 3.4.1 in the *Sun neighborhood* image of 210x210 pixels. Defining $q = 10$ (10 different spin values) to randomly assign a spin value to each pixel. The spin interaction, in this case, only considers the distance between pixels. The susceptibility was estimated through Swendsen-Wang Monte Carlo, it was of the order of 10^{-7} after 25 iterations with $T_{nul} = 0.13$ and $T_{peaks} = 0.01$. Then, the super paramagnetic region is in the range [0.01,0.13]. Any temperature in this range will move the system to the

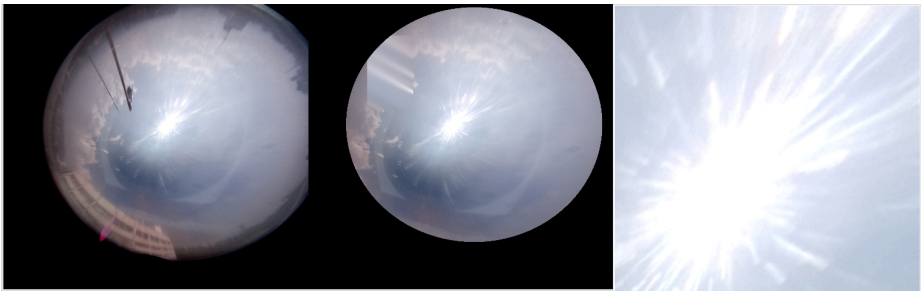

Figure 11: December 18, 2021 11:18:53 hours. (Left) final image, (Center) cut and cleaned image, (Right) *Sun neighborhood*.

super paramagnetic region and would in theory work as the cluster temperature, and thus we chose a temperature of 0.07 as the cluster temperature.

Estimation of the spin-spin correlation at the cluster temperature, is done again employing the Swendsen-Wang Monte Carlo algorithm, forming pixel clusters if their correlation is above 0.5. In the *Sun neighborhood* image from Figure 11, we found 13% of cloudy pixels. Implementation in *R version 4.2.2* through own code.

The number of cloudy pixels gives the intensity of the homogeneous Poisson process, helping to estimate cloud location for the subsequent minutes. Defining a grid on the area of interest, we can easily estimate the number of cloudy pixels at each subregion on the grid. For homogeneous processes, the intensity at each subregion is proportional to its area, see Section 3.4.2. Implementation via *spatstat (Spatial Point Pattern Analysis, Model-Fitting, Simulation, Tests)* package for *R*.

4.2.3. Ray tracing

By applying the steps from Section 3.5 to the real and estimated cloud layers, we obtain the shadows from clouds on a region on the surface of the earth. Recall that the sun zenith and azimuth angles are needed, as well as the cloud base height, given a date and a time. Sun position angles come from the SOLPOS calculator. See Sections 3.5 and 3.2 respectively.

Finally, the total area of the PV plant is 439.26 m^2 as this PV plant is formed by 270 PV panels. Each PV panel has an area of, 1640x992 mm^2.

Table 2 shows an example of the data requirements to work with the proposed models. Relative humidity (RH) and the PV areas are dimensionless variables, expressed as ratios, not percentages. The other three variables have the kilo and deci prefixes from the International System of Units (SI). The former is attached to PV Power and irradiance (GHI) and the latter to the PV cells temperature (T_{cells}).

Date Time	RH	GHI (kW/m^2)	T_{cells} ($d°C$)	PV Power (kW)	PV area [a]	PV area est. [b]
Dec 18 2021 11:18:53	0.443	0.616	3.763	12.824	0.09	0.07
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2: Required data to work with the proposed models. PV areas are not needed for the dlm model from Section 2. Data from 11:18:53 hours.

[a] Shadowed area on PV panels computed from nephograms. Section 3.4

[b] Shadowed area on PV panels computed from cloud location estimates. Sections 3.4 and 3.5

According to us, the main constraints of the proposed models are related to 1) parameter estimation at the beginning of the day and 2) scarcity of data from ceilometers. For the former, at the beginning of the day, there is not enough data to estimate model parameters. For the latter, ceilometers are typically placed at big meteorological stations or airports, not necessarily close to the PV plant of interest and they are relatively expensive instruments. As possible solutions to overcome the constraints above, we propose to 1) consider either artificial data or real data from other days having similar conditions at sunrise and 2) to compute the difference in the measured and estimated shadowed areas on the *Sun neighborhood* image instead of the PV plant as appears in the model, skipping the Ray tracing step.

5. Summary and conclusions

The goal of this work is to propose a forecasting method capable to produce accurate intra-hour probabilistic PV power output forecasts. This model works with a new set of parameters every day that is updated at each time step during a day. Particularly, to try to detect ramps that are difficult to detect in advance. Markov switching models are suitable to achieve this goal. The highlight is that Markov switching models could have a good performance to model PV power output data. Then, we compare two similar and simple models: 1) A dynamic

linear model, see Section 2. 2) A linear Gaussian Markov switching model with time-varying transition matrix, see Section 3, depending on covariate z that we generate after processing nephograms, see Sections 3.3 to 3.5. This transition matrix induces a Markovian dependence in the modelling of the PV Power output data. For our goal, applying scaled forecast error metrics is suitable. In particular, the Mean Absolute Scaled Error (MASE) to measure global forecast accuracy in PV power output. Scaled error metrics are not dependent on the scale of the data and with data sets having negative or close to zero values. The latter is common when working with PV power output data. To evaluate ramp detection, scaled forecast errors are preferable. Figures 7 and 10 show a better performance, according to the scaled forecast errors, of the Markov switching model against the dynamic linear model on December 18, 2021.

The case study is the PV plant at *Unidad de Posgrado*, UNAM. Data recollection started on November 25 finishing on December 21, 2021, registering data from sunrise to sunset every day. Data includes PV power output, a few meteorological variables and nephograms captured with our self-made, low-cost all-sky imager. Models from Sections 2 and 3 are tested with data from December 18, 2021, as that day shows high variability in PV power output during the data recollection period. Days with high variability are the most difficult to forecast. On those days, ramps are common too. Table 1 shows a better performance of the Markov switching model against both, the dynamic linear and the persistence or naive forecasting model.

To get the values of covariate z_t at time step t, the model proceeds as follows, see Figure 1:

1. Capture nephograms through the all-sky imager, generating a final image at a maximum rate of approximately one per minute and a half. In the all-sky imager, we apply the exposure fusion technique to limit exposure problems. Removing distortion generated by fish-eye lens with an appropriate change of coordinates. Section 3.3.
2. In a second computer, different from the all-sky imager, we remove interfering objects and cut the final image to obtain what we call the *Sun neighborhood* image. We apply a SPC clustering algorithm to the *Sun neighborhood* image, aiming at grouping cloudy and clear-sky pixels as well as pixels representing the sun. Section 3.4.
3. Through a homogeneous Poisson process and considering the cloudy pixels from 2), we estimate cloud location minutes ahead. 3.4.2.

4. Through ray tracing, we project the shadows on the PV plant knowing its latitude and longitude at a specific time given the sun position, zenith and azimuth angles, and the cloud base height at each time step. We do it for the *Sun neighborhood* image as well as for the estimated cloud location layer we got from 3), Section 3.5. Thus, we compute the percentage of the PV plant area that is covered by shadows in both cases, for the *Sun neighborhood* image (cl) and for the estimated cloud location layer (cl_{est}) from 3). Then computing cl and, cl_{est} respectively.
5. Define $z = cl - cl_{est}$ at each time step. Covariate z then modifies the transition matrix of Markov chain S_t. Section 3.1.

Data were registered at different sample rates due to constraints related to measurement instruments or storage capacity. PV power output every second, meteorological variables every minute except for ceilometer data that was registered every 10 minutes and nephograms were saved every five minutes for most of the days. For the linear Gaussian Markov switching process, we decided to consider data every 15 minutes because we want to reproduce a real operation forecasting procedure with the computational power at our disposal. The whole process at each time step takes between 10 and 13 minutes. This is actually the bottleneck of our method in terms of processing time. With more computational power, the processing time can be reduced.

5.1. Other considerations and extensions for future work

In duration-dependent regime-switching models, it is possible to add other choices for the duration distribution of the regimes. In our case, the duration distribution is inherited by the assumptions of the model and follow a geometric distribution.

The clustering algorithm from 2) above is the most computational intensive. As an alternative, we could classify pixels with a simpler algorithm as the Red-to-Blue Ratio (RBR), as it was done in other works [45].

In 3) above, instead of a homogeneous Poisson process, we could work with models with non-constant or even random intensity functions as non-homogeneous Poisson processes or the more general log-Gaussian Cox processes. Adding other covariates to drive the intensity function in those models as wind speed and direction for example seem reasonable.

A simple variant avoiding 4) above without the need of data from ceilometers or stereographic images to project shadows on the PV plant is to just focus on the *Sun neighborhood* image. We just compute the percentage of the area in the *Sun neighborhood* image that is covered by clouds, as well as in the layer with the cloud location estimates. Those values would become our new cl and cl_{est} respectively.

Ceilometers are relatively expensive instruments, there is a possibility to estimate the cloud base height via stereographic images too [48].

6. Data availability

Data will be made available on request.

7. Acknowledgements

This work has been done thanks to the support of the Mexican Consejo Nacional de Ciencia y Tecnología (CONACyT), CVU 173581. We thank Dr. Michel Grutter and Dr. Alejandro Bezanilla from *Red Universitaria de Observatorios Atmosféricos* (RUOA) for kindly sharing meteorological data, including data from ceilometers. Thanks to Dr. Enrique Acha for his valuable comments. We also thank Alejandro Villalobos and Roberto Fuentes for the easiness to get access to the PV plant at *Unidad de Posgrado*. Both entities are part of *Universidad Nacional Autónoma de México* (UNAM). Finally, big thanks to José Fuentes for the help to connect the Power Quality analyzer and to enable a remote access method to get the captured nephograms.

References

[1] Wang, L., Mao, M., Xie, J., Liao, Z., Zhang, H. & Li, H. Accurate solar PV power prediction interval method based on frequency-domain decomposition and LSTM model. *Energy*. **262** (2023)

[2] Cirés, E., Marcos, J., Parra, I., García, M. & Marroyo, L. The potential of forecasting in reducing the LCOE in PV plants under ramp-rate restrictions. *Energy*. **188** (2019)

[3] Mansouri, S., Rezaee Jordehi, A., Marzband, M., Tostado-Véliz, M., Jurado, F. & Aguado, J. An IoT-enabled hierarchical decentralized framework for multi-energy microgrids market management in the presence of smart prosumers using a deep learning-based forecaster. *Applied Energy*. **333** pp. 120560 (2023)

[4] Arcos-Vargas, A., Nuñez, F. & Román-Collado, R. Short-term effects of PV integration on global welfare and CO2 emissions. An application to the Iberian electricity market. *Energy*. **200** (2020)

[5] Mertens, T., Kautz, J. & Van Reeth, F. Exposure Fusion: A Simple and Practical Alternative to High Dynamic Range Photography. *Computer Graphics Forum*. **28**, 161-171 (2009)
[6] Chatuverdi, D. & Isha Solar Power Forecasting: A Review. *International Journal Of Computer Applications*. **14**, 28-50 (2016)
[7] Antonanzas, J., Osorio, N., Escobar, R., Urraca, R., Pisón, F. & Antonanzas-Torres, F. Review of Photovoltaic Power Forecasting. *Solar Energy*. **136** pp. 78-111 (2016)
[8] Voyant, C., Notton, G., Kalogirou, S., Nivet, M., Paoli, C., Motte, F. & Fouilloy, A. Machine Learning Methods for Solar Radiation Forecasting: A Review. *Renewable Energy*. **105** pp. 569-582 (2017)
[9] Dong, Z., Yang, D., Reindl, T. & Walsh, W. Short-term solar irradiance forecasting using exponential smoothing state space model. *Energy*. **55** pp. 1104-1113 (2013)
[10] Prema, V. & Uma Rao, K. Development of Statistical Time Series models for Solar Power Prediction. *Renewable Energy*. **83** pp. 100-109 (2015)
[11] Sheng, H., Xiao, J., Cheng, Y., Ni, Q. & Wang, S. Short-Term Solar Power Forecasting Based on Weighted Gaussian Process Regression. *IEEE Transactions Of Industrial Electronics*. **65**, 300-308 (2017)
[12] Iversen, E., Morales, J., Møller, J. & H, M. Probabilistic Forecast of Solar Irradiance using Stochastic Differential Equations. *IEEE Transactions Of Industrial Electronics*. **25**, 152-164 (2014)
[13] Dukkipati, S., Sankar, V. & Varma, P. Forecasting of Solar Irradiance using Probability Distributions for a PV System: A Case Study. *International Journal Of Renewable Energy Research*. **9** (2019)
[14] Khan, W., Walker, S. & Zeiler, W. Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach. *Energy*. **240** (2022)
[15] Lorenz, E., Hurka, J., Heinemann, D. & Beyer, H. Irradiance Forecasting for the Power Prediction of Grid Connected Photovoltaic Systems. *IEEE J. Of Selected Topics In Applied Earth Observations And Remote Sensing*. **2**, 2-10 (2009)
[16] Wang, F., Lu, X., Mei, S., Su, Y., Zhen, Z. & Zou, Z. A satellite image data based ultra-short-term solar PV power forecasting method considering cloud information from neighboring plant. *Energy*. **238** (2022)
[17] Shakya, A., Michael, S. & Saunders, C. Solar Irradiance Forecasting in Remote Microgrids Using Markov Switching Model. *IEEE Transactions On Sustainable Energy*. **8**, 895-905 (2017)
[18] Jiang, Y., Long, H., Zhang, Z. & Song, Z. Day-Ahead Prediction of Bi-hourly Solar Radiance With a Markov Switch Approach. *IEEE Transactions On Sustainable Energy*. **8**, 1536-1547 (2017)
[19] West, M. & Harrison, J. Finite mixture and Markov switching models. (Springer,1997)
[20] Campagnoli, P., Petrone, S. & Petris, G. Dynamic Linear Models with R. (Springer,2009)
[21] Andrieu, C., Doucet, A. & Holenstein, R. Particle markov chain monte carlo methods. *Journal Of The Royal Statistical Society: Series B (Statistical Methodology)*. **72**, 269-342 (2010)
[22] Elvira, V. & Chouzenoux, É. Graphical Inference in Linear-Gaussian State-Space Models. *IEEE Transactions On Signal Processing*. **70** pp. 4757-4771 (2022)
[23] Särkkä, S. & Svensson, L. Bayesian filtering and smoothing. (Cambridge university press,2023)
[24] Rao, R., Mani, M. & PC, R. An updated review on factors and their inter-linked influences

on photovoltaic system performance. *Helyion*. **4**, 1-34 (2018)
[25] Vidyanandan, K. An Overview of Factors Affecting the Performance of Solar PV Systems. *Energy Scan*. **22** pp. 2-8 (2017)
[26] Meral, M. & Dinçer, F. A review of the factors affecting operation and efficiency of photovoltaic based electricity generation systems. *Renewable And Sustainable Energy Reviews*. **15**, 2176-2184 (2011)
[27] Perrakis, G. & Tasolamprou, A. Ultraviolet radiation impact on the efficiency of commercial crystalline silicon-based photovoltaics: A theoretical thermal-electrical study in realistic device architectures. *OSA Continuum*. **3**, 1436-1444 (2020)
[28] Patel, J. & Tiwana, P. Effect of Ultraviolet Radiation on Organic Photovoltaic Materials and Devices. *ACS Applied Materials And Interfaces*. **11** pp. 21543-21551 (2019)
[29] RUOA Red Unversitaria de Observatorios Atmosféricos. (Instituto de Ciencias de la Atmósfera y Cambio Climático, Universidad Nacional Autónoma de México,0), https://www.ruoa.unam.mx/, [Last accessed on 19 May 2023]
[30] Hamilton, J. Regime switching models. *Macroeconometrics And Time Series Analysis*. pp. 202-209 (2010)
[31] El-Laham, Y., Yang, L., Djurić, P. & Bugallo, M. Particle filtering under general regime switching. *2020 28th European Signal Processing Conference (EUSIPCO)*. pp. 2378-2382 (2021)
[32] Li, W., Chen, X., Wang, W., Elvira, V. & Li, Y. Differentiable Bootstrap Particle Filters for Regime-Switching Models. *ArXiv Preprint ArXiv:2302.10319*. (2023)
[33] Frühwirth-Schnatter, S. Finite mixture and Markov switching models. (Springer,2006)
[34] Chow, S. & Zhang, G. Nonlinear Regime-Switching State-Space (RSSS) Models. *Psychometrika: Application Reviews And Case Studies*. **78**, 742-768 (2013)
[35] Manshu, Y. & Chow, S. Using State-Space Model with Regime Switching to Represent the Dynamics of Facial Electromyography (EMG) Data. *Psychometrika: Application And Case Studies*. **74**, 744-771 (2010)
[36] Kim, C. & Nelson, C. State-space models with regime switching: classical and Gibbs-sampling approaches with applications. (MIT Press,1999)
[37] García-Franco, J., Stremme, W. & Bezanilla, A. Variability of the Mixed-Layer Height Over Mexico City. Boundary-Layer. *Meteorol.*. **167** pp. 493-507 (2018)
[38] Devernay, F. & Faugeras, O. Straight lines have to be straight. *Machine Vision And Applications*. **13** pp. 14-24 (2001)
[39] Blatt, M., Wiseman, S. & Domany, E. Clustering data through an analogy to the Potts model. *Advances In Neural Information Processing Systems*. **8** pp. 416-422 (1996)
[40] Blatt, M., Wiseman, S. & Domany, E. Data Clustering using a granular magnet. *Neural Computation*. **9** pp. 1805-1842 (1997)
[41] Wu, F. The Potts Model. *Reviews Of Modern Physics*. **54**, 235-268 (1982)
[42] Wang, S. & Swendsen, R. Cluster Monte Carlo Algorithms. *Physica A: Statistical Mechanics And Its Applications*. **167**, 565-579 (1990)
[43] Banerjee, S., Carlin, B. & Gelfand, A. Hierarchical Modeling and Analysis for Spatial Data, 2nd ed. (CRC Press,2015)
[44] Diggle, P. Statistical Analysis of Spatial and Spatio-Temporal Point Patterns, 3rd ed. (CRC Press,2015)
[45] Richardson Jr, W., Krishnaswami, H., Vega, R. & Cervantes, M. A Low Cost, Edge Computing, All-Sky Imager for Cloud Tracking and Intra-Hour Irradiance Forecasting. *Sustain-*

ability. **9**, 482-498 (2017)

[46] SOLPOS Solar Position and Intensity Calculator. (National Renewable Energy Laboratory,0), https://midcdmz.nrel.gov/solpos/solpos.html, [Last accessed on 13 March 2022]

[47] Hyndman, L. & Koehler, A. Another look at measures of forecast accuracy. International Journal of Forecasting. *International Journal Of Forecasting*. **22**, 679-688 (2006)

[48] Andreev, M., Chulichkov, A., Medvedev, A. & Postylyakov, O. Estimation of cloud base height using ground-based stereo photography: method and first results. *Proc. SPIE, Remote Sensing Of Clouds And The Atmosphere XIX; And Optics In Atmospheric Propagation And Adaptive Systems XVII*. **9242**, 335-341 (2017)