Allan Robert da Silva

Markov Chain with Acceptance-Rejection

Recife, 2018

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM BIOMETRIA E ESTATÍSTICA APLICADA

Markov Chain with Acceptance-Rejection

Thesis considered adequate for obtaining the doctoral degree in Biometrics and Applied Statistics, defended and approved unanimously by the Examinination Board on December 18, 2018.

Area of Concentration: Statistical and Computational Modeling

Advisor: Prof. Dr. Borko D. Stosic
Co-Advisor: Profa. Dra. Tatijana Stosic

RECIFE/PE - December -2018.

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas da UFRPE
Biblioteca Central, Recife-PE, Brasil

S586m Silva, Allan Robert da.
Markov Chain with Acceptance-Rejection / Allan Robert da Silva. – Recife, 2018.
61 f.: il.

Orientador: Borko Stosic.
Coorientador: Tatijana Stosic
Tese (Doutorado em Biometria e Estatística Aplicada) – Universidade Federal Rural de Pernambuco, Departamento de Estatística e Informática, Recife, 2018.
Inclui referências e apêndices.

1. Markov Chains 2. Wind speed 3. Precipitation 4. Acceptance-Rejection 5. Synthetic Data I. Borko, Stosic, orient. II. Stosic, Tatijana III. Título

CDD 310

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM BIOMETRIA E ESTATÍSTICA APLICADA

Markov Chain with Acceptance-Rejection

Allan Robert da Silva

Thesis considered adequate for obtaining the doctoral degree in Biometrics and Applied Statistics, defended and approved unanimously by the Examinination Board on December 18, 2018.

Advisor:

Prof. Dr. Borko D. Stosic
UFRPE

Co-Advisor:

Profa. Dra. Tatijana Stosic
UFRPE

Examining Board:

Prof. Dr. Silvio Fernando Alves Xavier Júnior
UEPB

Prof. Dr. Antonio Samuel Alves da Silva
UFRPE

Prof. Dr. Lucian B. Bejan
UFRPE

This work is dedicated to my wife, Maria Aliani, and to my son, Pedro Antônio.

Acknowledgements

First I want to thank God that through mercy and wisdom he gave me the necessary knowledge so that I could develop all this research.

I would also like to thank my mother, Josefa Valentino da Silva, and my father, Joventino Pedro da Silva (deceased a few years), who with great love dedicated themselves to our family. He educated us not only with words, but with our own testimony of life that we should live with love, affection, respect, responsibility and honesty.

To my dear brothers who, from an early age, encouraged me to read and play games and, without realizing it, they also shaped my way of thinking and developing solutions to my daily problems.

My beloved wife, Maria Aliani, who dedicates so much to me and my son. Without it all this work would not make sense, because our mutual love generates in me much happiness and makes me who I really am.

To my son Pedro Antonio who early in the morning makes me stand up in order to liven up my day to day giving me a strength so that I can continue walking and improving more and more as a human being and as a father.

To my dear teachers who have been with me since the first years of study until my Master's degree, because without each of them today I would not be here in this defense of thesis. Especially my dear and late professor of letters Grespan, in which he was the first teacher who encouraged me to seek a better professional qualification.

My undergraduate and master's counselor Dra. Carla Almeida Vivacqua opened the door to scientific knowledge for me and was instrumental in encouraging me to take a postgraduate course.

To my professors of doctorate prof. Dr. Borko Stosic and profa. Dra. Tatijana Stosic, who encouraged me and gave me freedom and great suggestions for the development of this thesis. I would like to point out that without them this work would not have this quality and that I admire them not only as professionals, but also as people.

The coordination, the teachers and the secretaries that are part of the postgraduate program in Biometrics and applied Statistics of UFRPE dedicated to create and develop, in partnership with the Federal University of Sergipe (UFS), the DINTER project that provided this training to UFS teachers.

The availability of the members of this thesis examination board who dedicated themselves to reading and correcting this text and contributing to its improvement of quality.

Thank you to Célia Maria Pereira who welcomed me in the time that I have been developing in person this doctorate in the city of Recife.

To the friends of the doctorate and my department of work (DECAT), especially Samuel Oliveira, Luiz Henrique and Carlos Raphael, who always take time to talk and exchange experiences of life and work.

Wisdom offers protection,
how does money,
but the advantage of knowledge is this:
wisdom preserves the life of the possessor.
(Holy Bible, Eclesiastes 7, 12)

Abstract

This thesis presents a new algorithm to generate synthetic wind speed data using Markov chains and the Acceptance-Rejection Method. Traditionally, to simulate wind speed data using Finite Markov chains, states are represented by classes of values and, after selecting a using the transition matrix, a uniform random number is generated within the boundaries of this class. In the novel algorithm proposed here, each state is randomly generated within the selected class using the Acceptance-Rejection method. It is shown that the generated synthetic data of the proposed algorithm better reproduce the distribution of observed data than the traditional algorithm, for different numbers of states and for transition matrices from first to fifth order. To improve the reproduction of the data over time, the proposed algorithm is used to simulate only the random part series of wind speed, i.e. the observed series with extraction of trend and seasonality. With this, the synthetic data is closer to the original series, as it can be verified in different measures of error and in the behavior of the autocorrelation. Finally, the proposed algorithm is used to generate synthetic data from the Standardized Precipitation Index (SPI) in which it was able to reproduce the statistical characteristics of the sample density even with a relatively small sample. The new algorithm is described in general and can be adapted to other variations of Markov chains, as well as to obtain synthetic series of other natural phenomena.

Keywords: Markov Chain. Acceptance-Rejection. Synthetic Data. Wind Speed. Precipitation.

Resumo

Esta tese tem o objetivo de apresentar um novo algoritmo para gerar dados sintéticos de velocidade do vento usando Cadeias de Markov e o Método da Aceitação-Rejeição. Tradicionalmente para simular dados de velocidade de vento, usando Cadeias de Markov Finitas, cada estado é representado por uma classe de valores e, após a seleção da classe obtida na matriz de transição, é gerado um número aleatório uniforme considerando os limites desta classe. No algoritmo proposto, cada elemento dos estados é gerado usando o método da Aceitação-Rejeição. Os dados sintéticos gerados do algoritmo proposto conseguem reproduzir melhor a distribuição dos dados observados do que o algoritmo tradicional em diferentes números de estados e para matrizes de transição de primeira até a quinta ordem. Para melhorar a reprodução dos dados ao longo do tempo, o algoritmo proposto é usado para simular somente a parte aleatória da série de velocidade do vento, ou seja, a série observada com extração da tendência e sazonalidade. Com isso, os dados sintéticos se aproximam mais da série original, conforme pôde ser verificado em diferentes medidas de erro e no comportamento da autocorrelação. Ao final, o algoritmo proposto é usado para gerar dados sintéticos do Indice de Precipitação Padronizado (SPI), no qual consegue reproduzir bem as características estatísticas da densidade amostral mesmo usando uma amostra relativamente pequena. O novo algoritmo é descrito de maneira geral e pode ser adaptado para outras variações de Cadeias de Markov e na obtenção de séries sintéticas de outras variáveis ambientais.

Palavras-chave: Cadeias de Markov. Aceitação-Rejeição. Dados Sintéticos. Velocidade do Vento. Precipitação.

List of Figures

List of Tables

Summary

1 Introduction

Global cumulative wind power capacity has been growing exponentially over the last decades (SAWYER; DYRHOLM, 2018), due to technological advances that have led to market competitiveness, as well as concerns over global warming partly caused by fossil fuel energy generation. This growth has been accompanied by a growing interest in wind power production modeling and simulation studies, including wind resource quantification, wind speed modeling and prediction, wind power production, and system reliability assessment (YU; TUZUNER, 2008).

In order to plan the effective use of wind energy at a given location, it is necessary to have a solid understanding of the characteristics of the region's wind distribution. Information on wind speed, generally collected through meteorological stations, can give an initial estimate of the potential for using this type of renewable energy.

Many methods have been developed for wind speed forecasting, that may be divided into two main categories: i) physical models that take into account physical characteristics of the region, and ii) statistical models that explore relationships within measured data (LEI et al., 2009). The physical models usually have advantage in long-term forecasting, while the statistical methods do well in short-term prediction (LEI et al., 2009).

Several techniques based on a stochastic approach are used to model wind speed, among which the most widely used methodologies include the autoregressive moving average models (ARMA), Markov chains and wavelet analysis (AKSOY et al., 2004). A variety of density functions to describe wind speed are often cited in the literature (CARTA; RAMIREZ; VELAZQUEZ, 2009). In addition, these techniques are compared among themselves and with other new methodologies (KAMINSKY et al., 1991; SFETSOS, 2000; CARAPELLUCCI; GIORDANO, 2013; SOMAN et al., 2010; TASCIKARAOGLU; UZUNOGLU, 2014; TANG; BROUSTE; TSUI, 2015).

In the applications involving finite Markov chains, wind speed is categorized into intervals (classes) that correspond to states of a finite chain, and, with this, synthetic series are obtained that incorporate information of a current state in order to estimate future states. One of the pioneering publications describing a simulation algorithm of synthetic wind speed series, based on the cumulative transition matrix observed and estimated by maximum likelihood, was written by Sahin e Sen (2001). The authors used a first order Markov chain with eight states to model the wind speed. The states are defined by means of deviations around the sample mean. The quality of the fit was assessed by comparing the distance between the mean and the standard deviation of the observed data and the actual data.

Shamshad et al. (2005) used first and second order matrices taking 1m/s as amplitude of wind speed for the definition of twelve states. The authors use mean, percentiles, standard deviation, and other measures, in addition to the autocorrelation function and spectral density, to compare the original data with the synthetic data. They also compare parametric estimates by fitting a Weibull model to actual data and simulated data. They conclude that the second-order transition matrices are better than the first-order matrices and suggest studies with higher order chains.

The effect of state resolution on the fit quality for synthetic series generation is addressed in the study of Hocaoglu, Gerek e Kurban (2008). For this, the authors compare two models with state amplitude of 1m/s and 0.5m/s, in which they generated 13 and 26 states respectively for the construction of a first order matrix. Descriptive measures and frequency of observed and estimated values for each state were compared, concluding that there is an improvement in the estimates with 26 states, and that it is necessary to verify this effect in higher order chains.

Petre, Rebenciuc e Ciucu (2016) used first-order Markov chains considering the states spaced equally at 1m/s within the turbine operational range. The authors indicate that the method has good predictive result for wind power in the short term and can be used to fill small gaps in the time series.

Seasonality effect has been discussed by authors who work with synthetic data generation of wind speed via Markov chains. For example, Wu et al. (2012) used wind energy data obtained every minute for 9 months to examine the effect of different numbers of states and seasonality. State spaces of size 10, 15, 20 and 100 were observed, in which synthetic data were generated in two ways: i) from a general transition matrix, and ii) using nine different transition matrices, one for each month (all were matrices of first order). For comparison, they used descriptive measures, autocorrelation function and probability density function. They concluded that by selecting an ideal number of states it is possible to generate wind energy series of better quality, especially when seasonality is taken into account (using 9 chains).

Similarly Karatepe e Corscadden (2013) used 9 first and second order chains with data from two stations from different climatic regions collected every 10, 20 and 50 minutes. Transition matrices of one month of each of the four seasons of the year were estimated. Descriptive measures, probability density function and a goodness of fit test were compared. The authors concluded that one month data is sufficient for reproducing general wind speed characteristics associated with seasonality.

Another important discussion is the limitation of autocorrelation reproduction by the Markov chain. Nfaoui, Essiarab e Sayigh (2004) used a 12 state first-order Markov chain and conclude that the comparison between the observed wind speed and the synthetic indicates a good reproduction of wind speed characteristics, but the synthetic data can not reproduce the auto-correlation of the original data. A more successful reproduction of autocorrelation was reported by Pesch et al. (2015), using a second-order transition matrix.

Besides the above mentioned developments, there have been many other publications that used finite Markov chain to generate synthetic wind speed data using the original algorithm proposed by Sahin e Sen (2001). The initial motivation of this study and its primary objective is to propose an enhancement of this algorithm, to achieve a more accurate description of wind speed data in terms of its probability density function (PDF), and consequently of the capacity to reproduce the series. The idea is to generate the synthetic data for each state in accordance with the empirical distribution, using the acceptance-rejection method (a succinct description of accept-reject sampling can be found in Casella, Robert e Wells (2004)).

The data of wind speed are from the municipality of Petrolina (9°24"S, 40°29"W, 370 m) that is located in the Brazilian northeast. The municipality has its economy based on agriculture with irrigated fruit cultivation representing great economic and social importance for the region (MELO; ARAGÃO; CORREIA, 2013). Figure 1 shows the map of Brazil emphasizing the state of Pernambuco in blue, and the municipality of Petrolina, in green. The first object of this study is wind speed collected in 10 minute intervals in the period from January 1 to December 31, 2010, at the height of 50 meters, with 52470 observations measured in meters per second (m/s). The observed data can be obtained from the website of the Brazilian National Institute of Space Research (INPE) available at http://sonda.ccst.inpe.br/.

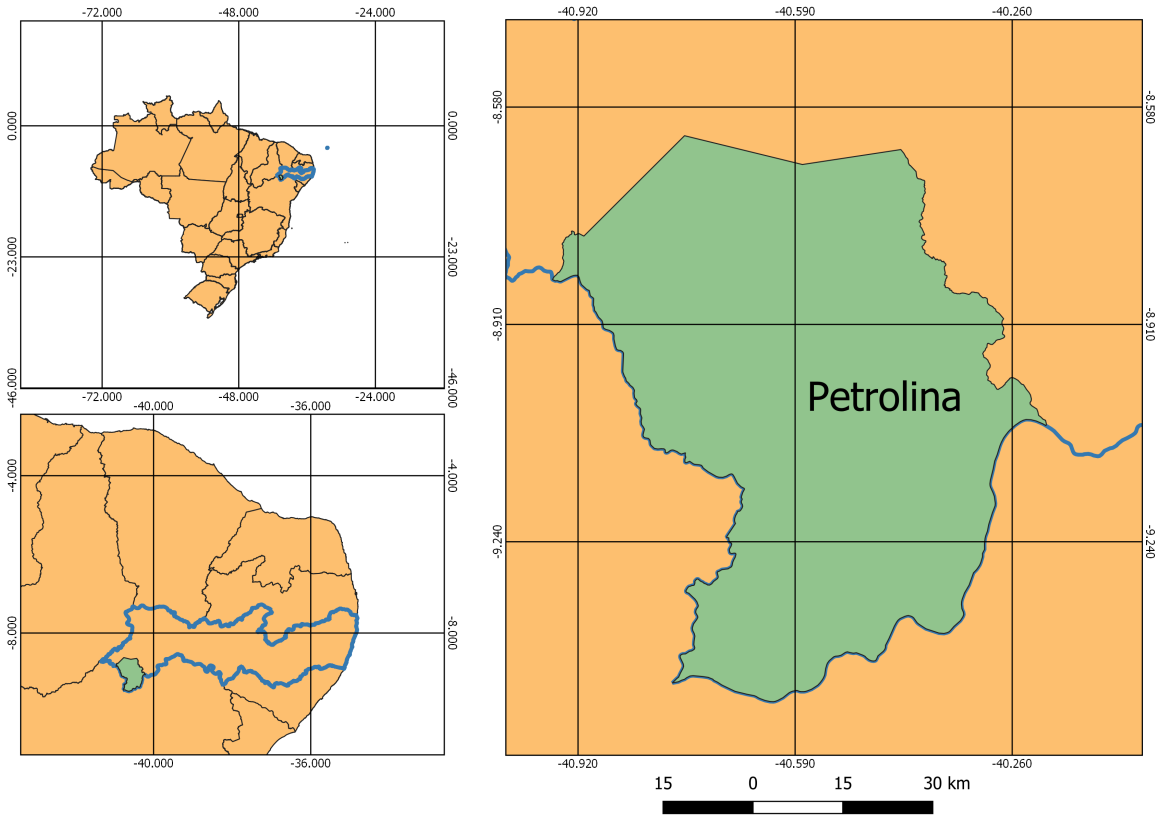

Figure 1 – Map of Brazil with the state of Pernambuco (with outline in blue) and your municipality Petrolina (in green)

Another justification for using data from this municipality is its potential for wind power generation. Carneiro e Carvalho (2015) studied the wind patterns of three Brazilian cities: Maracanaú, Parnaíba, which are part of the coast of the state of Ceará and Piauí respectively, and Petrolina that is part of the interior of Pernambuco. They concluded that the municipality of Petrolina, even though it is not a coastal city, is one of the cities that presents the best potential for wind power due to its high regularity and low variability in wind behavior. This information on the wind potential associated with knowledge about the wind speed distribution of Petrolina is important for future wind energy investment in the region.

The second objective of this work is to use the proposed algorithm to generate synthetic data of the Standardized Precipitation Index (SPI) to monitor the drought in the semi-arid northeast. To generate these synthetic data, monthly precipitation series were used from, January 1950 to December 2012 from the neighboring municipalities of Petrolina and Afrânio (8°30"S, 40°00"W, 522 m).

The SPI proposed by McKee et al. (1993), is the most common index used to monitor droughts, which is easy to co-recharge and can be used at different scales times (GUTTMAN, 1999). As opposed to wind speed, no studies have been identified in the literature with the objective of generating synthetic SPI data using Markov chains. It is worth mentioning that it is common to find high resolution datasets of wind speed series, such as databases containing information per minute, every 10 minutes, etc. However, precipitation data are generally available monthly and this implies that the data sets are small and there is a greater difficulty in estimating the statistical parameters using traditional methods.

In the following chapters, the methodology will be explored in some detail. In Chapters 2 to 4 the proposed algorithm is applied to wind speed. In chapter 2, first-order chains are considered with variations in the number of states and the results will be compared to the traditional method. In chapter 3, higher-order Markov chains will be used for synthetic data generation. The Chapter 4 presents a methodology that suggests the extraction of the trend and seasonality of the wind speed series to obtain a closer approximation of the synthetic series of the observed series. The Chapter 5, the proposed algorithm is used to generate synthetic SPI data. Finally, in chapter 6, a general conclusion is presented to summarize the results obtained and perspectives for future work are discussed. All simulations, tests and graphical visualizations were performed in software R (R Core Team, 2018) (commands in Appendix B).

2 Markov Chain with Acceptance-Rejection: Variations in the Number of States

In this chapter the new algorithm to generate wind speed synthetic data is presented and compared to the traditional approach. The first-order Chains are considered with variations in the number of states. Initially it was verified whether the successive events of the transition matrices are independent or dependent on each other. The quality of reproduction of the original distribution is verified by comparing the descriptive measures (mean, median, 1^{st} quartile, 3^{rd} quartile and standard deviation) and using determination coefficient (R^2), Chi-square (χ^2) and root mean square error (RMSE) applied to the relative frequencies. The Kolmogorov-Smirnov test is used to verify if the distribution of synthetic data is equal to that of the observed data. The impact of autocorrelation will be verified in both algorithms. Finally agreement measures are used to test the quality of simulations.

2.1 Methodology for generating synthetic data using Markov Chain

In what follows, we first describe the standard Markov chain approach. Next, we address the acceptance-rejection method for generating samples drawn from an arbitrary function on a closed interval, applied in the current proposal for enhancing the Markov chain algorithm. Finally, we describe the commonly used algorithm proposed by Sahin e Sen (2001), together with the current enhanced Markov algorithm proposal to generate wind speed synthetic data.

2.1.1 Markov Chain

The first-order Markov chain is a stochastic process where the next state depends only on the current state, that is, past states do not influence the future (HOEL; PORT; STONE, 1986). This (Markovian) property can be formulated as

$$P(X_{n+1} = j|X_0 = i_0, \ldots, X_n = i,) = P(X_{n+1} = j|X_n = i), \tag{1}$$

where $X_1, X_2, \ldots, X_n, X_{n+1}$ are the values of the random variable X observed at instances $1, 2, \ldots, n, n+1$. If the transition probabilities of the Markov chain are time independent the chain is said to be homogeneous. Also, when the number of states is finite, then the chain is called the finite first-order Markov chain. To simplify the notation, we shall henceforth write

$$p_{ij} \equiv P(X_{n+1} = j|X_n = i)$$

for the probability of transition from state i to state j. The maximum likelihood estimator of p_{ij} is given by

$$\hat{p}_{ij} = \frac{m_{ij}}{\sum_{j=1}^{n} m_{ij}}, \tag{2}$$

where m_{ij} represents the observed number of transitions from state i to state j.

For n states the transition probabilities can be written as a square matrix (the so called transition matrix) P of the form:

$$P = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix},$$

with the restrictions

1. $p_{ij} \geq 0$;
2. $\sum_{j=1}^{n} p_{ij} = 1, \quad \forall i.$

The original algorithm proposed by Sahin e Sen (2001) for simulating wind speed data rests on the above first order Markov chain approach, while more elaborate methods may take into account dependence of the transition on $k > 1$ previous values ($k-th$ order Markov chain).

Some important definitions of Markov chains (will be more discussed in the following chapters):

Definition 1. ***(Accessibility):*** *In a Markov chain, state j is said to be accessible to state i if in "m" steps $P_{ij}^{(m)} > 0$ for some $m \geq 0$. This means that starting from state i, it is possible (with positive probability) to enter state j in finite number of transitions.*

Definition 2. ***(Communicate):*** *State i and state j are said to communicate if state i and state j are accessible from each other ($P_{ij}^{(m)} > 0$ and $P_{ji}^{(m)} > 0$ for some $m \geq 0$).*

Definition 3. ***(Class of states):*** *Two or more states that communicate are said to be in the same class.*

Definition 4. ***(Irreducibly):*** *A Markov chain is said to be irreducible, if all states belong to the same class, i.e. they communicate with each other.*

For the simulation of synthetic data it is important that the estimated transition matrix be irreducible. Otherwise problems will occur, for example if there are more than one class, from a certain moment the synthetic data will be continuously and exclusively generated in this class.

2.1.2 Acceptance-Rejection Method

In the original wind speed simulation Markov chain approach of Sahin e Sen (2001) the next state j is selected on the basis of the current state i, with probability p_{ij}, and then the actual value X_j is drawn uniformly from the bin corresponding to the state j. This amounts to approximating the empirical distribution with the histogram on the scale defined by the chosen

number of bins, which may be considered too coarse in the context of large samples commonly available and used nowadays, in conjunction with the readily available elevated computational capabilities.

In Figure 2a the histogram of the observed data of wind speed in the municipality of Petrolina in 2010 is presented where each class is defined with amplitude of 1m/s, together with the empirical Gaussian kernel density estimate (WAND; JONES, 1994).

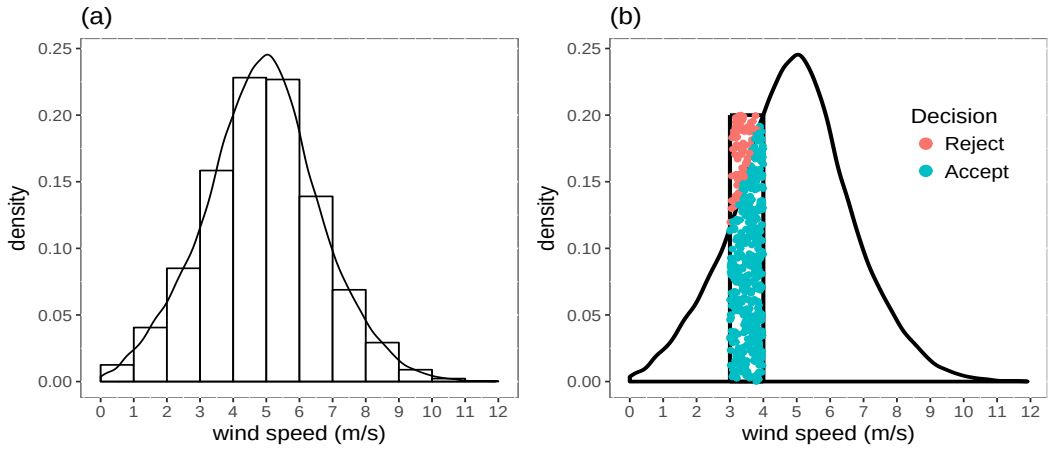

Figure 2 – (a) Histogram of observed wind speed of the municipality of Petrolina in 2010 distributed every 1m/s, and (b) simulating 500 numbers using the acceptance-rejection method at a kernel density range 3m/s to 4m/s of the observed wind speed data, where the red dots represent rejected and the blue dots accepted trials.

Assuming that the bin j corresponding to this particular interval was chosen as the next state of the Markov chain process, the histogram approach would suggest a uniform value for X_j anywhere between 3 and 4, while it is clear from Figure 2 that values close to $X_j = 3$ should be chosen with a much lower probability (almost twice as low) than the values close to $X_j = 4$.

The ARM (Acceptance-Rejection Method) is a simple alternative that takes into account the empirical distribution for choosing X_j, once that bin j has been selected as the Markov chain "next" state. In what follows, the most elementary version of the acceptance-rejection sampling (CASELLA; ROBERT; WELLS, 2004) is implemented for the sake of clarity (a more sophisticated approach by choosing piecewise functional approximations could somewhat enhance performance, sacrificing simplicity, but this appears to be only of academic, and not of practical interest in the current context).

Suppose one needs to generate numbers distributed according to the sample density $f(x)$, in a given interval $[x_k, x_{k+l}]$. The idea of the ARM is to sort out a pair of data (x_0, y_0), in which the value x_0 is obtained uniformly in $[x_k, x_{k+l}]$ and at the same time a value y_0 is drawn uniformly between 0 and the maximum value of the density $f(x)$ in this interval. If the drawn value y_0 is less than $f(x_0)$ then the value x_0 is accepted as a sample of this density function, otherwise it is discarded (rejected). The Figure 3 illustrates the steps for generating a sample in the interval $[x_k, x_{k+l}]$ via ARM. This process is repeated until a desired number of accepted values of x are accumulated.

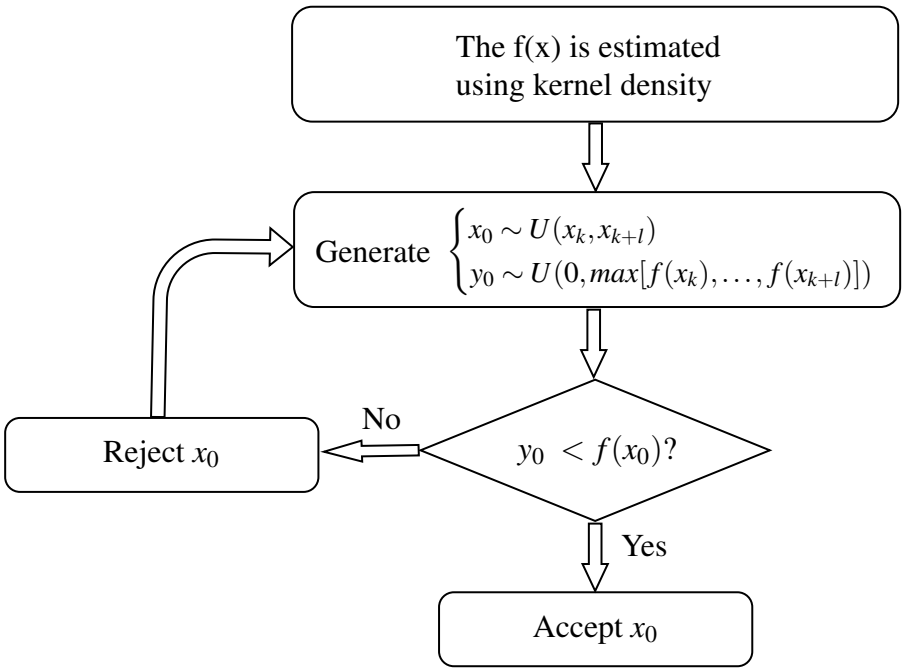

Figure 3 – Flow chart illustrating the Acceptance-Rejection method

An application of this method can be seen in Figure 2b that presents the sample density obtained from a Gaussian kernel of the observed wind speed data, where a rectangle with 500 uniformly distributed random points generated in the range of wind speed from 3 to 4, and density from 0 to 0.20 (maximum of the Gaussian kernel density in this interval). From these points 405 (in blue) were accepted since they are below the estimated density and 95 (in red) were rejected since they are above the estimated density.

Note that the frequency of accepted values gradually increases from $x = 3$ to $x = 4$ representing well the characteristic density curvature on this interval. If the draw had only been uniform in this interval as routinely done in the standard Markov chain approach, all the points presented would have been accepted and consequently the frequency of elements close to $x = 3$ would be equal to the frequency of elements near $x = 4$, ignoring the curvature of the sample density.

2.1.3 Generating synthetic data with the standard and the enhanced algorithm

After classification of each of the observed values of wind speed into one of the classes (states) Sahin e Sen (2001) propose the following first order finite Markov chain algorithm:

1. Find the cumulative probability transition matrix P, with values in each row summing to unity ($\sum_{j=1}^{n} p_{ij} = 1, \quad \forall i$).
2. Choose randomly the initial state i.
3. To generate the next state j from the current step i, a uniform random number u is drawn between 0 and 1, and from the i-th row of the transition matrix j is chosen as the state whose cumulative probability corresponds to u, such that $\sum_{k=1}^{j-1} p_{ik} < u \leq \sum_{k=1}^{j} p_{ik}$.
4. The estimated wind speed in each generated state is determined by another uniform random number for intermediate states, while for the extreme states (the first and the last bin) values are generated by considering a shifted exponential distribution. The smallest values in such exponential distributions are the lowest boundaries of the extreme states.

The enhancement of this algorithm proposed in the current work, modifies only the last step:

4. From the chosen bin j the estimated wind speed is determined by the acceptance-rejection method from the sample kernel density estimate.

As described in Chapter 1, there is no general rule of how states should be chosen, for example, most works suggest using equally spaced states every 1m/s or 0.5m/s, or composed of deviations around the mean. For coherent comparison of the algorithms all simulations were generated here with states equally spaced.

2.2 Results

Simulations were performed considering 8, 12, 16, 20, 24 and 28 states. For each number of states it was checked if the successive events of the transition matrices are independent or dependent on each other. For each combination of state size and algorithm, the mean, median, 1st quartile, 3rd quartile and standard deviation were compared. The quality of approximation of the observed data by the synthetic series was verified using determination coefficient (R^2), Chi-square (χ^2) an root mean square error (RMSE) applied to the relative frequencies (measures commonly used to compare adequacy of probability distributions to observed data (AKPINAR; AKPINAR, 2004; KANTAR; USTA, 2008; KANTAR; USTA, 2015)). Then, it was verified whether the generated series and the observed series come from the same distribution using

Kolmogorov-Smirnov test (WANG; TSANG; MARSAGLIA, 2003) and Anderson-Darling test (SCHOLZ; STEPHENS, 1987). Finally, the synthetic series are compared with the observed series using the error measures: Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE) and Symmetric Mean Absolute Percentage Error (sMAPE), where the latter is a modified version of Mean Absolute Percentage Error (MAPE) suggested by Makridakis (1993) to be used when there are values equal or very close to zero (TASHMAN, 2000; HYNDMAN; KOEHLER, 2006). The formulas of these measures are given by

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|x_i - \hat{x}_i| \quad , \tag{3}$$

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(x_i - \hat{x}_i)^2 \quad , \tag{4}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(x_i - \hat{x}_i)^2} \quad , \tag{5}$$

$$sMAPE = \frac{1}{m}\sum_{i=1}^{m}\frac{|x_i - \hat{x}_i|}{(|x_i| + |\hat{x}_i|)/2} \quad . \tag{6}$$

The Markov chain properties can be tested statistically by checking whether the successive events are independent or dependent on each other (SHAMSHAD et al., 2005). For successive events to be independent, the statistic γ, mathematically defined by:

$$\gamma = 2\sum_{i=1}^{n}\sum_{j=1}^{n} ln\frac{p_{ij}}{p_j} \tag{7}$$

is distributed asymptotically as χ^2 having $(n-1)^2$ degrees of freedom (DF), where n is the total number of states. The marginal probabilities p_j for the jth column of the transition probability matrix are given by

$$p_j = \frac{\sum_{i=1}^{n} n_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} n_{ij}} \tag{8}$$

where n_{ij} is the frequency of state i being followed by state j. The Table 1 presents the results of this test for each number of states using $\alpha = 5\%$ and in all cases it can be concluded that the events are dependent on each other.

Table 1 – Results of test independence each size of states

	Numbers of States					
Values	8	12	16	20	24	28
DF of χ^2	49	121	225	361	529	729
$\chi^2_{(\alpha=5\%)}$	66.34	147.67	260.99	406.3	583.61	792.92
γ	47147.77	85026.26	90681.42	94007.56	95787.58	96910.3

The Figure 4 shows the observed data kernel density estimate together with the density of the data synthesized from the conventional algorithm (FIMCUNI - Finite Markov chain with

Uniform distribution) and the proposed enhanced algorithm (FIMCAR- Finite Markov chain with Acceptance-Rejection) for each choice of the number of states.

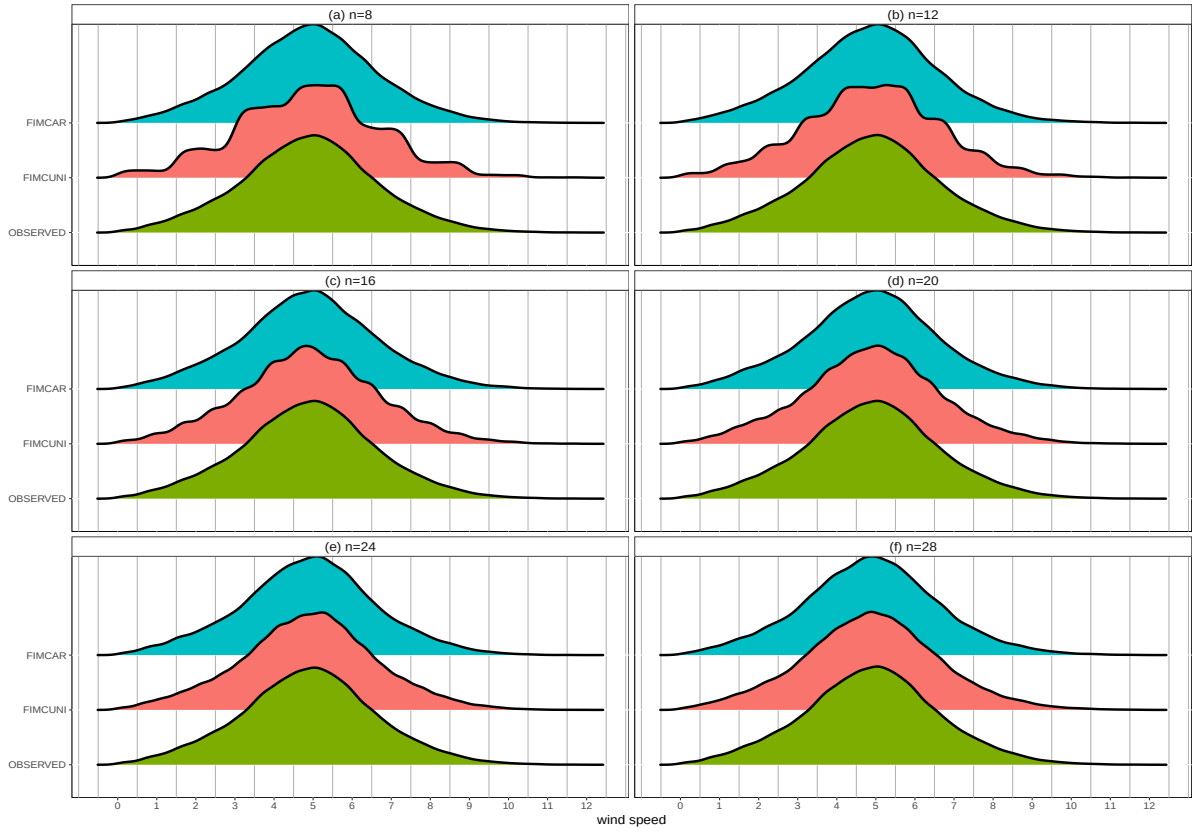

Figure 4 – Density of the wind speed data and density obtained from the synthetic data generated by the algorithms in each number of states

It can be seen that the density obtained by FIMCAR approaches the observed data kernel density for all state sizes, while in the standard FIMCUNI algorithm the approximation visually indiscernible on the scale of the graph from the kernel density estimate occurs only with 20 or more states.

The efficiency of the two algorithms in reproducing descriptive measures for different number of states is presented in Table 2, together with the p-values of the Kolmogorov-Smirnov test and Anderson-Darling test. While the mean, median, first and third quartile, and standard deviation are rather similar to the empirical values in all cases, the proposed algorithm presents a greater value of R^2 and smaller values of χ^2 and RMSE than the traditional algorithm, with more pronounced difference for sizes smaller or equal to 16 states. In addition, the p-values for the K-S and A-D statistic at 5% level demonstrate that for the FIMCUNI algorithm equivalence is not rejected only for 20 or more states, while the FIMCAR algorithm passes the rejection test for all state numbers.

Table 2 – Descriptive measures obtained from the algorithms in each variation of state size

FIMCUNI							
Statistics	Observed	Number of States					
		8	12	16	20	24	28
Mean	4.888	4.895	4.883	4.898	4.884	4.890	4.875
Median	4.897	4.919	4.902	4.885	4.906	4.909	4.874
1^{st} Quartile	3.756	3.649	3.730	3.761	3.757	3.765	3.734
3^{rd} Quartile	5.993	6.024	5.991	6.040	5.993	5.991	5.982
Stand. Dev	1.740	1.825	1.756	1.782	1.746	1.744	1.757
R^2		0.973	0.996	0.998	0.998	0.999	0.999
χ^2		$2x10^{-4}$	$2.4x10^{-5}$	$4.3x10^{-6}$	$3.2x10^{-6}$	$9.6x10^{-7}$	$6.9x10^{-7}$
RMSE		0.014	0.005	0.002	0.002	0.001	$8x10^{-4}$
A-D test p-value		$3.1x10^{-13}$	$5.4x10^{-7}$	0.001	0.412	0.447	0.172
K-S test p-value		$1.1x10^{-10}$	$7.2x10^{-7}$	$5x10^{-4}$	0.416	0.200	0.134
FIMCAR							
Statistics	Observed	8	12	16	20	24	28
Mean	4.888	4.897	4.883	4.899	4.883	4.890	4.876
Median	4.897	4.905	4.909	4.899	4.902	4.909	4.876
1^{st} Quartile	3.756	3.771	3.770	3.763	3.770	3.780	3.745
3^{rd} Quartile	5.993	6.006	5.980	6.023	5.984	5.986	5.984
Stand. Dev	1.740	1.728	1.715	1.755	1.735	1.732	1.743
R^2		0.999	0.999	0.999	0.999	0.999	0.999
χ^2		$3.7x10^{-7}$	$1.2x10^{-6}$	$7.9x10^{-7}$	$6.6x10^{-7}$	$7.3x10^{-7}$	$4.2x10^{-7}$
RMSE		$6x10^{-4}$	0.001	$9x10^{-4}$	$8x10^{-4}$	$9x10^{-4}$	$7x10^{-4}$
A-D test p-value		0.504	0.301	0.201	0.572	0.328	0.261
K-S test p-value		0.676	0.286	0.115	0.787	0.384	0.243

The two algorithms present the same relative frequency within the classes, because the proposed algorithm only changes the way of generating the data in the class itself. To use measures R^2, χ^2 and RMSE the number of classes was doubled by dividing them in half, and relative frequencies of these sub-classes were compared. Table 3 presents observed and estimated relative frequencies and measures for eight state simulation results, divided into sixteen sub-classes. While FIMCUNI by construction exhibits equal relative frequency in sub-class pairs (corresponding to the original classes), the proposed FIMCAR approach follows closely the observed data sub-class frequencies.

The Figure 5 shows the autocorrelation for both algorithms. Note that there is no difference between the autocorrelation values using the FUNCUNI and FUMCAR algorithm, which indicates that the algorithms do not influence autocorrelation. There is a small improvement when the number of states is increased, but it still does not reflect well the autocorrelation the observed data. It is worth noting that the autocorrelation decays faster due to the loss of memory inherent to the first-order Markov chain methodology.

Table 3 – Relative frequency and measures for eight states

States	Observed	FIMCUNI	FIMCAR
1	0.00699	0.01407	0.00734
	0.02049	0.01372	0.02045
2	0.04082	0.05451	0.04033
	0.06751	0.05377	0.06794
3	0.10839	0.13210	0.10856
	0.15388	0.12992	0.15346
4	0.18085	0.17454	0.18125
	0.16451	0.17218	0.16547
5	0.11466	0.09076	0.11412
	0.06926	0.09413	0.07077
6	0.03998	0.02914	0.03926
	0.02013	0.02966	0.01954
7	0.00820	0.00534	0.00741
	0.00313	0.00494	0.00286
8	0.00099	0.00065	0.00097
	0.00023	0.00059	0.00027
R^2		0.9734	0.999
χ^2		$2x10^{-4}$	$3.7x10^{-7}$
RSME		0.014	$6x10^{-4}$

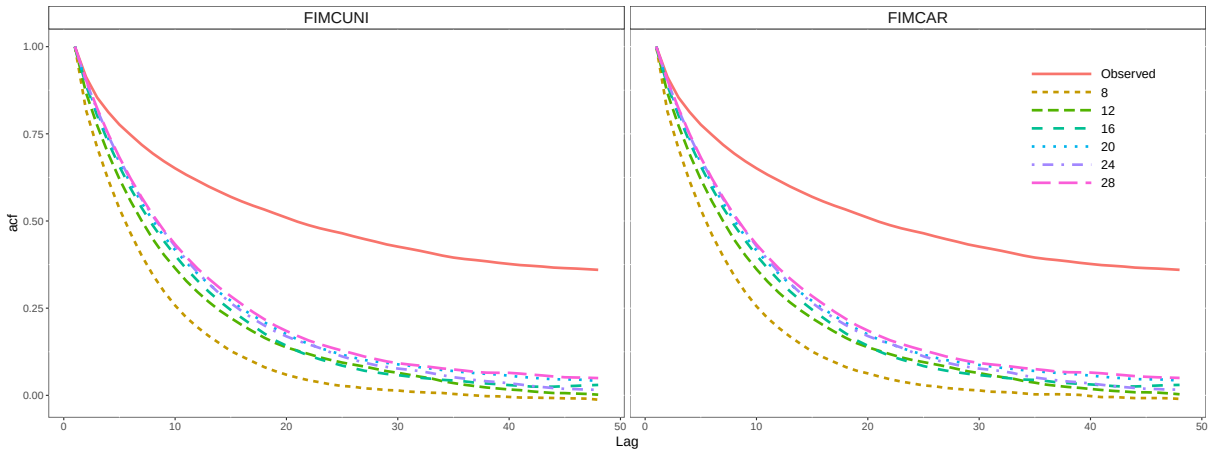

Figure 5 – Autocorrelation observed and estimated for each algorithm and size of states

Each error measurements (see appendix A) was normalized subtracting each value from the mean and dividing by the standard deviation. Figure 6 which shows the parallel coordinate of the synthetic data from each algorithm and each number of states. Note that for all state numbers error measurements indicate that the data simulated by the proposed algorithm present smaller distance from the real data (lower values on the graph), especially for less than 20 states.

Figure 6 – Graph of parallel coordinates of the error measurements applied in the synthetic data obtained from the algorithms for each size of states (lower is better for all measures)

2.3 Conclusions and Discussions

According to the simulation results, the synthetic data generated by the new algorithm are closer to the observed wind speed data of municipality of Petrolina than the conventional algorithm, both from the point of view of the reproduction capacity of the series, and its density. This improvement is more pronounced when the number of states is below 20, which is quite common in various cited studies.

The main advantage of the new algorithm is to make the choice of the number of states more flexible (since it produces good results regardless of the number of states), permitting the use of smaller samples for adequate characterization of the distribution, independent of their form.

An alternative possibility to this method would be to find a suitable probability distribution to the wind speed data and then generate truncated data considering this adjusted distribution. Several authors have discussed different distributions of probability to apply to wind speed data (CARTA; RAMIREZ; VELAZQUEZ, 2009; WANG; HU; MA, 2016; KANTAR et al., 2016; MASSERAN et al., 2013; SOHONI; GUPTA; NEMA, 2016). However the proposed method has a clear advantage because its own sample density is used and it is not necessary to find an ideal distribution to fit the data.

The algorithm proposed in this paper has been applied to finite first-order Markov chains, but can still be adapted to other Markov chain variations such as, for example, higher-order Markov chains and/or non-homogeneous Markov chains.

3 Markov Chain with Acceptance-Rejection: Exploring Higher Order Markov Chains

In this chapter, the algorithm proposed to generate synthetic wind speed data will be applied in higher order Markov chains. The problems of using higher-order Markov chains when increasing the number of states and how the use of this algorithm may make its use more flexible will be discussed. Markov chains will be simulated from first to fifth order with different sample sizes using the algorithms FIMCUNI and FIMCAR. The Kolmogorov-Smirnov test is used to verify if the distribution of synthetic data is equal to that of the observed data. Finally, autocorrelation and error measures are used to test the quality of the simulations.

3.1 Introduction

Markov chain has been commonly applied to wind speed and wind power using only first order chains and presents a satisfactory result in the reproduction of the characteristics of the wind speed distribution, being able to be used in the short term forecast (SAHIN; SEN, 2001; PETRE; REBENCIUC; CIUCU, 2016; SHAMSHAD et al., 2005). Selecting the correct number of states (usually greater than or equal to 12) generates better wind speed synthetic data (HOCAOGLU; GEREK; KURBAN, 2008; WU et al., 2012). Some studies compare first-order Markov chains to other new techniques aiming to improve especially the autocorrelation function (CARAPELLUCCI; GIORDANO, 2013; AKSOY et al., 2004; PESCH et al., 2015; XIE et al., 2017). Other authors apply first and second order Markov chains at the same time and conclude that there is an improvement of the accuracy by using the second-order model (KAMINSKY et al., 1991; KARATEPE; CORSCADDEN, 2013; SHAMSHAD et al., 2005). Markov chains from first to third order are explored only in wind power with similar conclusions regarding the difficulty in reproducing the autocorrelation function (PAPAEFTHYMIOU; KLOCKL, 2008; BROKISH; KIRTLEY, 2009).

Initially, the computational problems arising from simulations using higher order Markov chains will be discussed. Then, different situations involving Markov chains with different number of states and order are simulated using the FIMCUNI and FIMCAR algorithms.

3.2 Problems and solutions when using higher order Markov chains

Many authors indicate the use of higher-order chains, but there are no discussions in the literature regarding the difficulties presented in computational simulations in this type of study.

From a practical illustration, two problems will be discussed: the problem of using many states and the problem of the last sequence of states.

3.2.1 Increasing the number of states

Generalizing the Equation 1, in Markov chain of order $k \geq 1$ the probability of a state in time $t+1$ depends of k previous states, which can be described as:

$$\begin{aligned} &P(X_{t+1} = x_{t+1} | X_0 = x_0, \ldots, X_{t-1} = x_{t-1}, X_t = x_t) = \\ &P(X_{t+1} = x_{t+1} | X_{t-k+1} = x_{t-k+1}, \ldots, X_{t-1} = x_{t-1}, X_t = x_t) \end{aligned} \tag{9}$$

When $k = 1$, this usual fist order Markov chain. For $k = 2$ returns a second order Markov chain, mathematically rewritten as:

$$P(X_{t+1} = x_{t+1} | X_0 = x_0, \ldots, X_{t-1} = x_{t-1}, X_t = x_t) = P(X_{t+1} = x_{t+1} | X_{t-1} = x_{t-1}, X_t = x_t) \tag{10}$$

If there are n states with a matrix of order k, then for computational purposes, the transition matrix can be written with $n^k n$ elements, with n^k rows and n columns, where the sum of rows must add 1. For an example, consider $n = 3$ and a string of order 2, then the transition matrix P will have this characteristic:

$$\begin{array}{c} \\ 11 \\ 12 \\ 13 \\ 21 \\ 22 \\ 23 \\ 31 \\ 32 \\ 33 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{pmatrix} p_{111} & p_{112} & p_{113} \\ p_{121} & p_{122} & p_{123} \\ p_{131} & p_{132} & p_{133} \\ p_{211} & p_{212} & p_{213} \\ p_{221} & p_{222} & p_{223} \\ p_{231} & p_{232} & p_{233} \\ p_{311} & p_{312} & p_{313} \\ p_{321} & p_{322} & p_{323} \\ p_{331} & p_{332} & p_{333} \end{pmatrix} \end{array}$$

If, for example, 10 states are used considering a second-order Markov chain, then the transition matrix will have 100 rows and 10 columns, making a total of 1000 elements to be filled. If one wants to keep the 10 states and use a third-order matrix, there will now be 10,000 elements to fill. A large data set is required to correctly estimate all transitions. In practice, the authors suggest 12 or more states and suggest doing higher-order chains, which is only possible if there is large sample to estimate probabilities of transitions. This difficulty also induces a high computational cost and requires an advanced programming level.

3.2.2 The problem of last observed sequence of states

Regardless of the order or number of states to generate synthetic data using Markov chains, it is necessary that the transition matrix be irreducible, e.g. all states must communicate. Otherwise, the process will stop each time it arrives in a state that does not communicate with any other. This makes the simulation tiring and biased because at each stop it will be necessary to restart the simulation from a new state without considering what has already been simulated. In practice, this will only be a problem when the last sequence is unprecedented in the data set. Which tends to occur when dealing with higher order transition matrices and many states.

To illustrate this problem, consider the following sequence for four states:

1232324323112443311212334433221111212122321121223321122312332134 1

where the transition matrix of the first-order this sequence of states is given by:

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.37 & 0.58 & 0.05 & 0.00 \\ 0.36 & 0.19 & 0.36 & 0.09 \\ 0.18 & 0.41 & 0.29 & 0.12 \\ 0.17 & 0.00 & 0.50 & 0.33 \end{pmatrix} \end{array}$$

As the matrix of this example is irreducible there will be no problems in generating synthetic data for wind speed. Consider now the second-order transition matrix of this same example below:

$$\begin{array}{c} \\ 11 \\ 12 \\ 13 \\ 14 \\ 21 \\ 22 \\ 23 \\ 24 \\ 31 \\ 32 \\ 33 \\ 34 \\ 41 \\ 42 \\ 43 \\ 44 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.29 & 0.71 & 0.00 & 0.00 \\ 0.36 & 0.27 & 0.27 & 0.09 \\ 0.00 & 0.00 & 0.00 & 1.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \\ 0.38 & 0.50 & 0.12 & 0.00 \\ 0.25 & 0.00 & 0.75 & 0.00 \\ 0.25 & 0.38 & 0.37 & 0.00 \\ 0.00 & 0.00 & 0.50 & 0.50 \\ 0.67 & 0.31 & 0.00 & 0.00 \\ 0.43 & 0.14 & 0.29 & 0.14 \\ 0.20 & 0.60 & 0.00 & 0.20 \\ 0.50 & 0.00 & 0.00 & 0.50 \\ 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.33 & 0.67 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 \end{pmatrix} \end{array}$$

This transition matrix is not irreducible since there is a non-zero probability of arriving at state 41 (in red) with a 50% chance after the sequence 34 (in blue), but since there is no probability to go to another state and not even remain in it, this leads to a problem in the simulation of synthetic data. With a larger sample it is possible this problem does not occur, but one may have problems with a chain higher order. If one more state was observed and it was, for example, equal to 2 (two) this problem would have been solved. Now the last sequence would be 12 and would not be unpublished. Consequently, the sequence 41 would have a nonzero probability to go to sequence 12, so there would be no problem in using a second-order Markov chain. However, if there is interest in third-order matrices, the last sequence will be unpublished and the problem of irreducibly would occur again.

The two problems were cited separately, but they tend to be correlated. The greater the number of states and the order of the chain, the greater the chance of the last sequence observed to be unpublished as seen in the illustration above.

3.2.3 A possible solution using the FIMCAR algorithm

Several studies which cite the traditional algorithm of Sahin e Sen (2001) indicate that the higher the number of states the better the generation of synthetic data (HOCAOGLU; GEREK; KURBAN, 2008; WU et al., 2012). This creates a drawback for the use of larger order transition matrices since the increase of states and the use of chains of higher order generate several problems already mentioned. A possible solution would be the use of the proposed algorithm FIMCAR, because it can reproduce well the characteristics of the wind speed distribution with a small number of states. This makes it more convenient for generating wind speed synthetic data using higher order Markov chains, especially if one has a relatively large data set.

The wind speed data set used here is once again the data from Petrolina in 2010. Synthetic wind speed data will be generated using first-order to fifth-order Markov chains considering 4, 6, 8 and 10 states from the FIMCUNI algorithms and FIMCAR. Although it has been verified that FIMCUNI cannot represent well the distribution characteristics with few states, this result is specific for first order Markov chains. In addition, the limitations themselves discussed in the previous section show that it is not possible to have a large number of states when one wants to use higher order chains. The distribution equality between the synthetic data and the observed data will be compared using the Kolmogorov-Smirnov test and the proximity of the synthetic series to that observed is calculated using the error measures MAE, MSE, RMSE and sMAPE, already presented in the previous chapter.

3.3 Results

The Figure 7 shows the histogram of each combination of number of states (columns) and chain order (rows) of the synthetic wind speed data using the FIMCAR algorithm. Note that regardless of the number of states and the string order the density of the synthetic data are similar. The Table 4, shows the value associated with the Kolmogorov-Smirnov test, in which it can be verified that in the FIMCAR algorithm independently of the number of states and the order of the chain one can conclude, at the level of 5% of significance that it is not rejected that the distribution of generated data is equal to the distribution of the observed data. On the other hand, for the algorithm FIMCUNI all cases have equal distribution rejected with the same significance. Since there were no good results again when using the FUMCUNI algorithm, the next results will be discussed only for the FIMCAR algorithm.

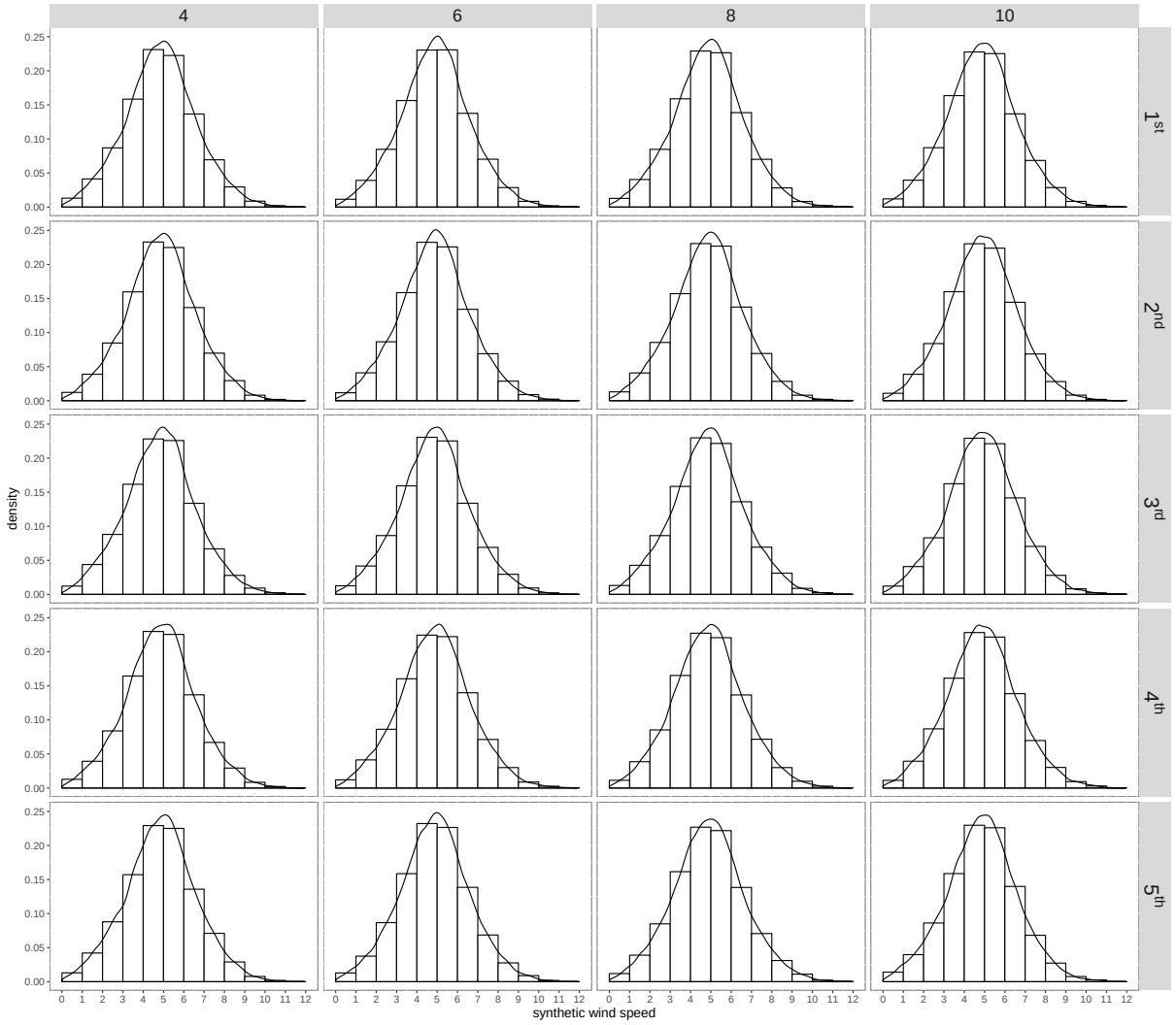

Figure 7 – Histograms with the distribution of the synthetic data of each combination of number of states (in the columns) and chain order (in the lines) the synthetic wind speed data using the FIMCAR algorithm

Table 4 – P-value of test Kolmogorov-Smirnov comparing the distribution of the synthetic data and the observed data

	FIMCAR				FIMCUNI			
	Number of States							
Order	4	6	8	10	4	6	8	10
1^{st}	0.157	0.100	0.943	0.146	0.000	0.000	0.000	0.000
2^{nd}	0.677	0.205	0.513	0.277	0.000	0.000	0.000	0.000
3^{rd}	0.227	0.207	0.290	0.533	0.000	0.000	0.000	0.000
4^{th}	0.184	0.362	0.302	0.408	0.000	0.000	0.000	0.000
5^{th}	0.313	0.630	0.187	0.433	0.000	0.000	0.000	0.000

Figure 8 shows the error measures (see appendix A) using the FIMCAR algorithm, in which it can be seen that these measures are not influenced by the variation of the chain order and the number of states addressed in this study. It can be observed, for example, that the first order Markov chain presented smaller error measures when there were six and ten states, whereas for four states it had worse performance and for eight it had a regular performance.

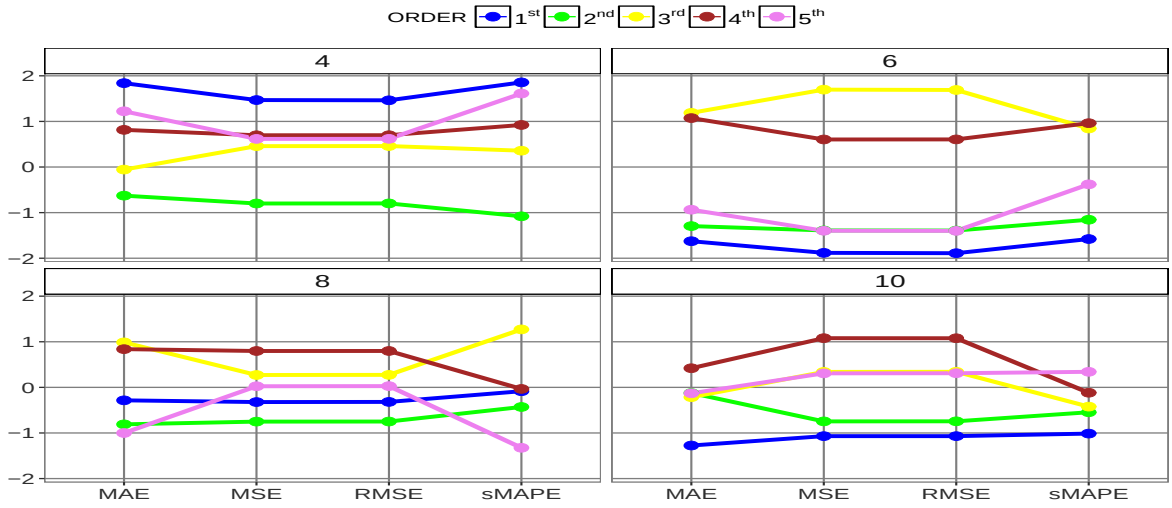

Figure 8 – Graph of parallel coordinates of the error measures in the synthetic data from the algorithm FIMCAR for each size of states and order of Markov chain

The Figure 9 presents the original series and the other synthetic series from the first to the fifth order of the FIMCAR algorithm. Note that the original data appear to have a trend and seasonality and that the first order chain exhibits a random behavior, but when the chain order is increased the behavior of the synthetic series tends to be closer to the original series. Even with the improvement, it can be noted that the trend and seasonality are not fully captured even in the higher order chains. Extracting these components from the original series can be a solution to improve synthetic data generation.

Figure 9 – Observed series and synthetic series of Markov chains from first through fifth order using the algorithm FIMCAR for eight states

Figure 10 shows that for the first lags the estimated autocorrelation is closer to that observed with a greater number of states (n = 10) and using higher order chains of these studies (4^{th} and 5^{th} order). For greater lags a slight improvement is observed when using higher order chains. However all cases present a significant distance from the observed autocorrelation.

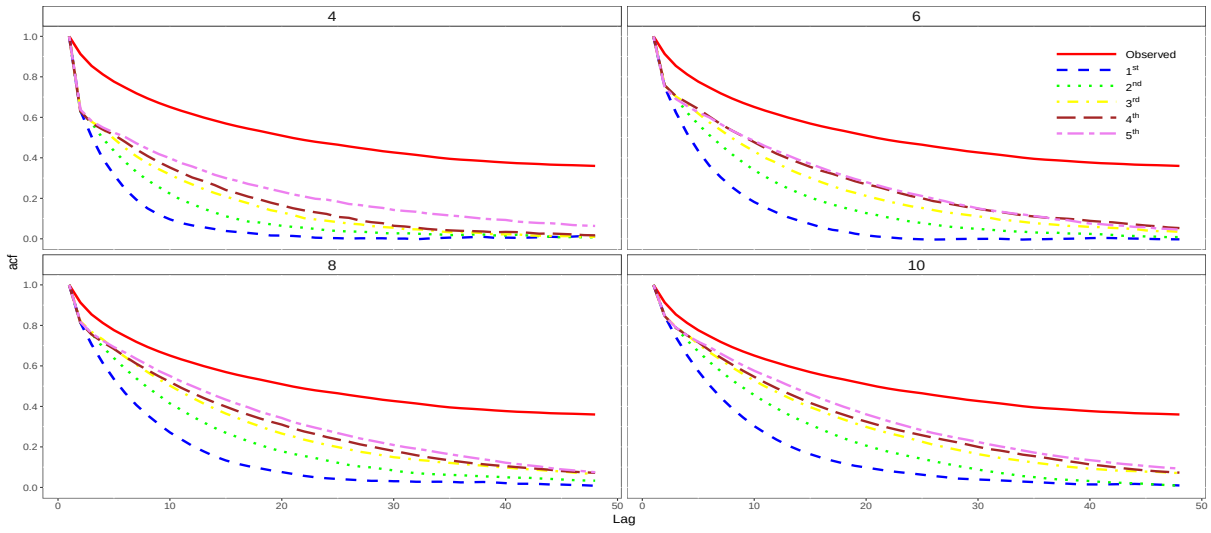

Figure 10 – Autocorrelation observed and estimated for each order and size of states

3.4 Conclusions and Discussions

The FIMCAR algorithm can preserve the characteristics of the observed wind speed data distribution for the state and order numbers of the observed Markov chains, whereas the traditional FIMCUNI algorithm cannot reproduce the original distribution in any of the situations explored in this chapter. This extends the results obtained in the previous chapter and it can now be concluded that the FIMCUNI algorithm cannot reproduce the behavior of the wind speed series distribution using few states considering Markov chains from first to fifth order.

In the literature the question of generating synthetic data of wind speed using higher order Markov chains was addressed only by authors Papaefthymiou e Klockl (2008), Brokish e Kirtley (2009), who used chains up to the third order. The use of larger chain orders to generate wind speed synthetic data has been restricted due to the technical and computational limitations discussed throughout this text. The work described in this chapter is pioneering, as it is the first time that 4^{th} and 5^{th} order Markov chains are used for this objective.

Visually, it is possible to suspect that the trend and seasonality of the original series is best reproduced in higher order chains, but the observed error measures do not indicate this improvement. Another interesting result is that the autocorrelation of the synthetic data has a small improvement when the order and the number of states is increased, but all cases present a distant autocorrelation of the observed one.

A possible solution to improve the generation of synthetic data in relation to the reproduction of the series over time and its autocorrelation is to simulate only the random part of the original series. The next chapter will explore this point by re-evaluating the impact of higher order chains using the FIMCAR algorithm.

4 Markov Chain with Acceptance-Rejection: Extracting Trend and Seasonality

This chapter will discuss the impact of trend and seasonality on synthetic wind speed data generation. Then, a proposed methodology is described suggesting the extraction of these components from the observed series using the FIMCAR algorithm. The resulting random distribution is compared to the simulated series using chains from first-order up to the fifth order. Finally, the trend and the seasonality are regrouped with the observed and simulated random part for the appropriate comparisons. These results will be compared to the results of the previous chapter to verify if there was improvement in the generation of wind speed synthetic data.

4.1 Introduction

Some authors argue that to better reproduce the wind speed series over time it is necessary to extract the trend and the seasonality of the data (SHAMSHAD et al., 2005; ETTOUMI; SAUVAGEOT; ADANE, 2003; MUSELLI et al., 2001). Of course a higher order chain tends to have a lower loss than a low order chain because it contains more previous information to generate a future state. However, as seen in the previous chapter, increasing the order of the chain does not cause the error measures to decrease. This indicates that there is no significant improvement in the approximation of the synthetic data to the actual data as the chain order increases.

Wind speed series already presented in previous chapters will be used, however, the trend and seasonality will be determined considering the fixed size of 8 (eight) states using transition matrices from the first to the fifth order. The simulated data will correspond to the random values of the series (without trend and seasonality). First, it will be verified if the simulated data has the same distribution of the random observed using the Kolmogorov-Smirnov test. Then, the trend and seasonality will be regrouped both in the random part of observed series and in the series estimated to be compared with the error measures and autocorrelation. In the end, these results will be compared to the results from the previous chapter for this number of states.

4.1.1 Extracting trend and seasonality of observed series

From the data of the observed series corresponding to only one year, it is not possible to verify the seasonality of the series in the traditional way when one has annual information, but the daily seasonality can be extracted (PESCH et al., 2015). The trend was estimated using a moving-average using 144 observations corresponding to the information of one day of the

wind speed series. From this series, without trend, the mean per hour and month will be obtained to estimate the seasonality. These means are shown in Figure 11, in which it can be seen that the wind speed structure has a drop between 1h and 9h (stronger in the winter months) and then from 9h to 12h there there is rapid growth (with peak in the winter months) and, finally, between 13 and 24 hours there are two different behaviors: in the months referring to autumn and spring there is a quadratic behavior and in the summer and winter months there is a decreasing behavior (with different intensities). Although the behaviors are similar considering each month and its season, the intensities are different. Therefore, the combination of hour and month will be used as an estimate of the seasonality in this period. The random series to be simulated will be the series of the original data taken from the series without trend less and the seasonal component (mean per hour and month) which is presented in Figure 12 along with the observed series together with the estimated trend in which from now on be named raw series.

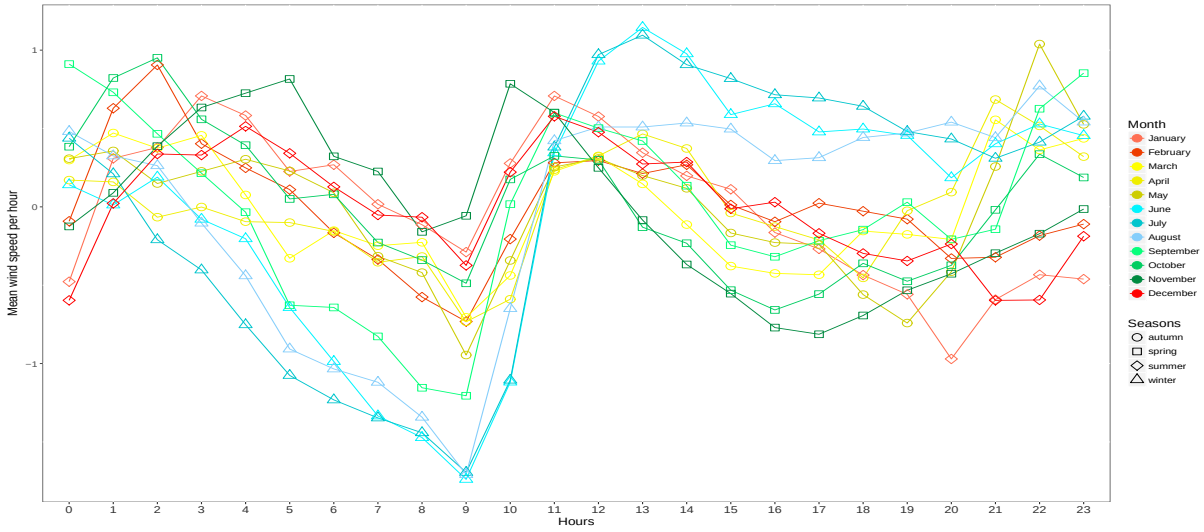

Figure 11 – Mean per month and hour of the wind speed series without trend (similar color scale indicate same meteorological season)

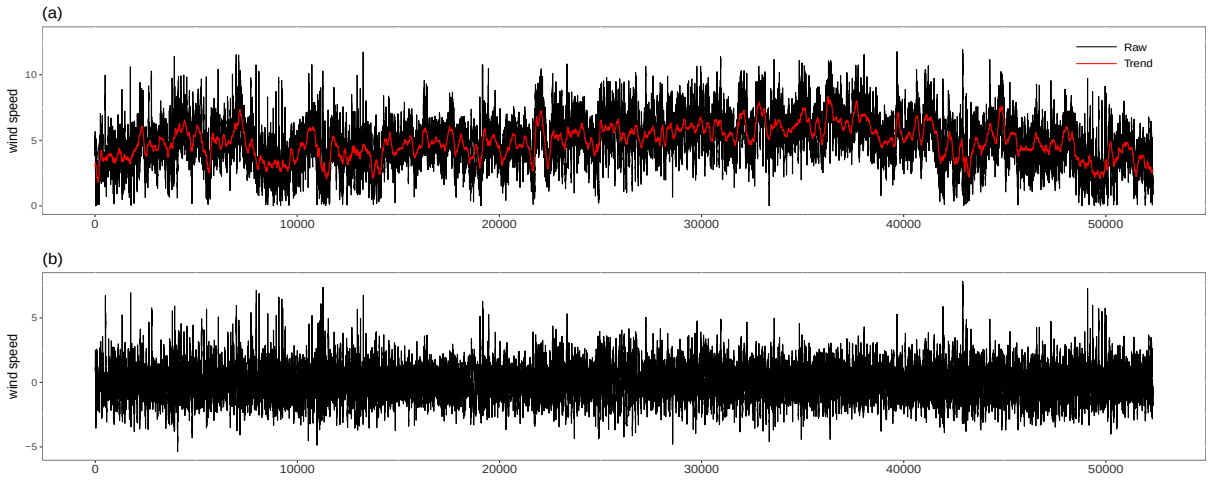

Figure 12 – (a) Raw series (Observed series) with the estimated trend and (b) random series

4.2 Results

The Figure 13 shows the distribution the random part of the wind speed data and the synthetic data generated from 8 (eight) states with Markov chains from first through fifth order. Each part referring to the simulated data present the p-value of the Kolmogorov test, in which it can be concluded that independently of the order of the Markov chain the FIMCAR algorithm was able to reproduce the density of the random part of the data.

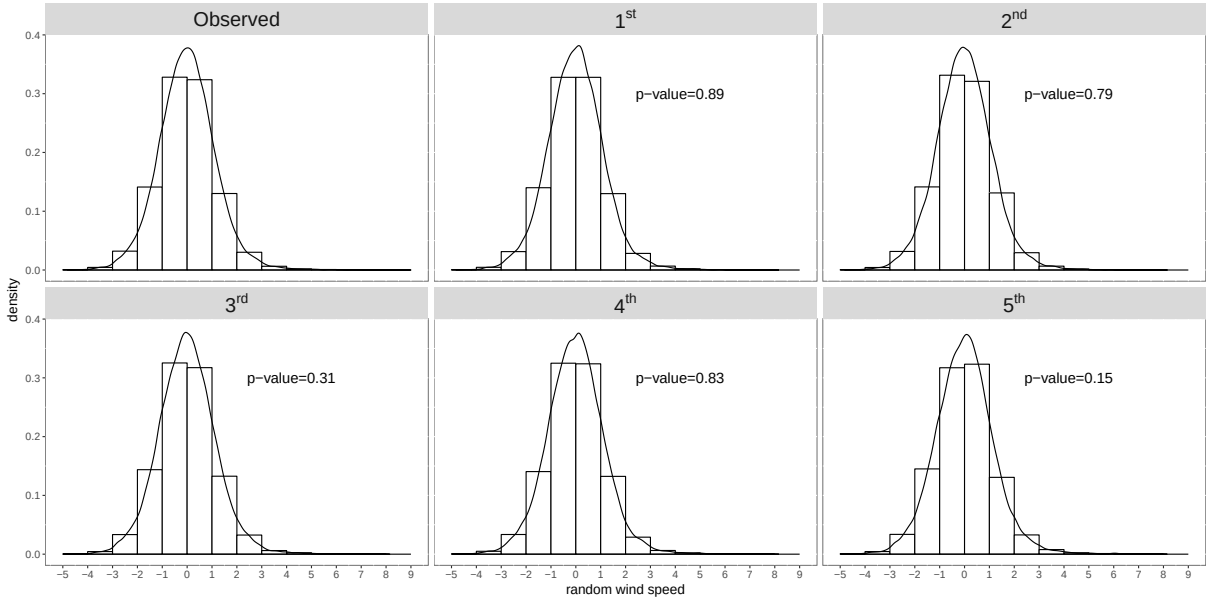

Figure 13 – Density of random part of the wind speed data and the synthetic data generated from 8 (eight) states with Markov chains from first to fifth order

The estimated trend and seasonality of the original series were incorporated in the random (returning the original series) and incorporated in the synthetic series to verify if they present approximation in terms of descriptive measures and if they present similar behavior over time. Descriptive measures of mean, median and standard deviation are presented in the Table 5, in which it can be observed that the synthetic series with trend and seasonality have measures of central trends very similar to the original series, but the estimated standard deviation is slightly underestimated.

Table 5 – Descriptive measures of the synthetic series added to the trend and the seasonality of the observed series

		Order				
Statistics	Observed	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}
Mean	4.894	4.893	4.888	4.899	4.893	4.906
Median	4.902	4.897	4.877	4.889	4.883	4.896
1^{st} Quartile	3.763	3.797	3.785	3.777	3.769	3.762
3^{rd} Quartile	5.996	5.985	5.962	6.005	5.988	5.998
Stand. dev	1.737	1.636	1.642	1.661	1.675	1.680

In order to verify if there was improvement in the reproduction of the synthetic series using this methodology, we compared the error measures of the original series and the simulated series plus trends and seasonality. Figure 14 shows these comparisons together with the results of the previous chapter using 8 (eight) states, in which it can be seen that in all cases the observed series is best reproduced when only its random part is used in red) than when using the raw data (blue line above) and that in both cases the order of the chain does not seem to influence the improvement of the reproduction of the original series (see appendix A).

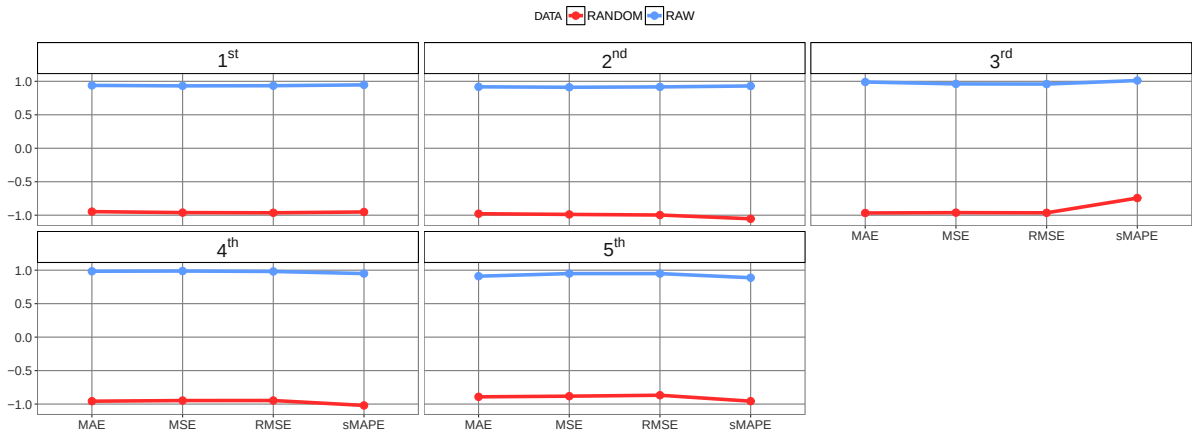

Figure 14 – Graph of parallel coordinates of the error measures in the synthetic data from the algorithm FIMCAR for raw data and random data

In Figure 15 it is possible to verify that the observed series and the synthetic series (estimate via the random series with fifth-order Markov chain with added the estimated seasonality and seasonality) show a very similar behavior over time.

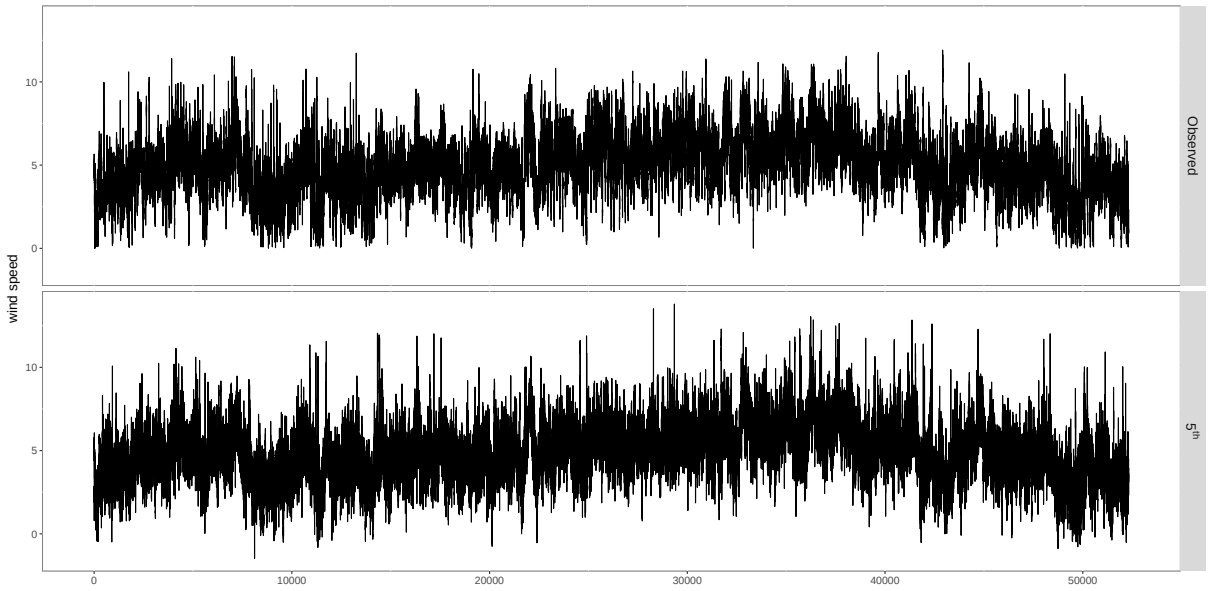

Figure 15 – Observed series and synthetic series using a fifth-order Markov chain

The figure 16 shows the autocorrelation for raw and random data, and it can be verified that simulating only the random data improve the autocorrelation estimates for any order of the chain. Observing the autocorrelation generated by simulating only random data there is a slight improvement in approximation when using 4^{th} and 5^{th} order chains.

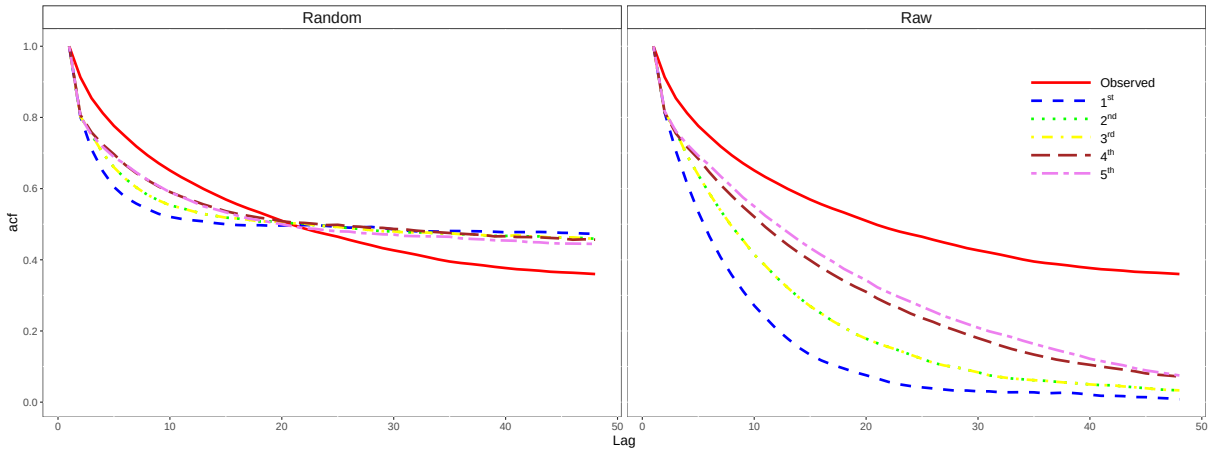

Figure 16 – Autocorrelation observed and estimated for each algorithm and size of states

4.3 Conclusions and Discussions

The extraction of the trend and the seasonality allowed to obtain a random series over time. The synthetic series generated from the FIMCAR algorithm were able to reproduce well the characteristics of the random series. With the incorporation of trend and seasonality in the observed and estimated series, it was verified that the central measurements are well reproduced and that only the variability is slightly underestimated.

When comparing the error measures of this chapter with the previous one (for eight states with raw data) it can be observed that there was an improvement in the generation of synthetic data using only the random wind speed series and that the order of the chain did not seem to have influence in both cases.

The simulated data only of the random part present an autocorrelation function much closer to the real if compared to the data using only raw data. In both cases, the 4^{th} and 5^{th} order chains show a subtle improvement in autocorrelation reproduction when compared to the other chain orders.

All the simulations of this chapter were using the algorithm FIMCAR, because a size of 8 (eight) states was used that did not have good results for the algorithm FIMCUNI. We did not verify the effect of this methodology for the FIMCUNI and FIMCAR algorithms using lower order chains, which may be a future study.

5 Markov Chain with Acceptance-Rejection: Generating SPI Synthetic Data

In the previous chapters the proposed algorithm was applied to wind speed, but in this chapter it will be applied to another natural phenomenon: precipitation. In this chapter, the FIMCAR and FIMCUNI algorithm will be used to generate synthetic data from the Standardized Precipitation Index (SPI). A series of 63 years of precipitation (756 observations) will be used from the neighboring municipalities of Afrânio and Petrolina. For both municipalities the adequacy of the distribution of the observed and synthetic SPI data will be verified using descriptive measures minimum, 1^{st} quartile, median, 3^{rd} quartile, maximum, mean and standard deviation and the Kolmogorov-Smirnov test.

5.1 Introduction

Good knowledge of influential meteorological phenomena, especially drought, is essential for the management and planning of water resources in a region. Drought is a natural phenomenon popularly known as the cause of various damages and affects a significant number of people in the world.

To monitor the drought it is necessary to know the local and regional characteristics. Analyzing the behavior of a hydrological time series is important to create a suitable mathematical model that allows a consistent forecast. More than 100 local and regional drought indices have been developed for this purpose (ZARGAR et al., 2011).

Developed by McKee et al. (1993) the Standardized Precipitation Index (SPI) is the most commonly used, which is easy to co-recharge and can be used at different scales times (GUTTMAN, 1999). In 2009, with the participation of 22 countries, the Interregional Workshop on indices and early warning systems for drought was held. The consensus of the participants was to recommended the use of SPI in all national meteorological services, and to provide this information on their websites (HAYES et al., 2011).

Autoregressive stochastic models are applied to rainfall and drought events. The Markov models along with SPI are often proposed to estimate the probabilities of drought. Sanusi et al. (2015) used first-order Markov Chains to predict and monitor drought in Malaysia. The authors used monthly SPI data to estimate the mean residence time, the mean recurrence time, and the average time of the first passage of the drought classes.

Factors that may influence the forecasting capability of Markov chains were studied by Banimahd e Khalili (2013) using SPI, RDI, EDI and SPEI in different climatic zones.

Steinemann (2003) used six classes, relative to the PDSI and the SPI applied Markov chain to characterize the probabilities for drought class transition and to identify the time of duration the classes.

Teixeira-Gandra, Damé e Silva (2017) estimated the SPI using Markov chain to simulate the occurrence of rain with the gamma distribution to predict monthly precipitation. They concluded that the statistical characteristics of the dry and wet days series were maintained, but extreme drought events were underestimated.

Estimate drought probabilities and drought predictions via non-homogeneous Markov chain model were studied by Rahmat, Jayasuriya e Bhuiyan (2017) who concluded that model can predict fairly well drought situations for a month ahead.

A comparison between Markov chain (MC) models and Network-Based (NB) models is discussed by Avilés et al. (2016). The results indicate that MC based models predict better wet and dry periods, while BN based models generate more accurate predictions of severe droughts.

Different from wind speed data, no workshave been identified with a proposal to generate synthetic SPI data via Markov chains. This is probably due to a limited number of samples to generate reliable data with traditional methods. In this chapter, the FIMCAR algorithm will be used to generate monthly synthetic SPI data for the municipalities of Afrânio (8°30"S, 40°00"W, 522 m) and Petrolina (9°24"S, 40°29"W, 370 m) which are neighboring and located in the Brazilian northeast as can be seen in Figure 17 below:

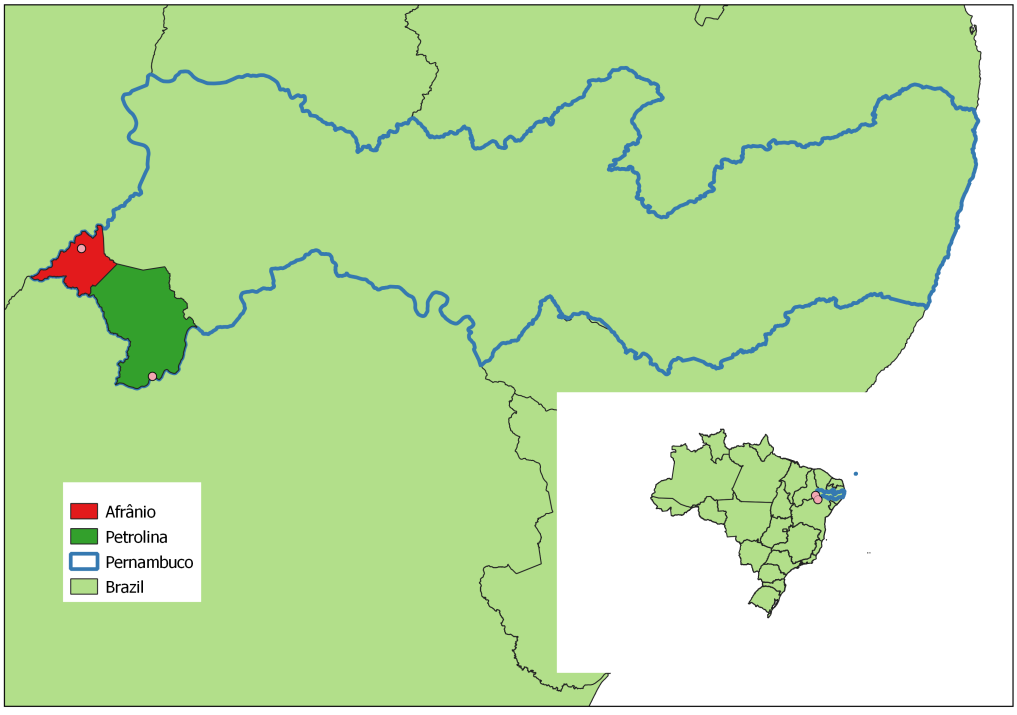

Figure 17 – Map of Brazil with state of Pernambuco and the cities of Petrolina and Afrânio

According to the climate classification of Köppen-Geiger the municipality of Afrânio is classified as *BSk*, i.e, semi-arid hot, while Petrolina is classified with *BSh*, i.e., semi-arid cold (PEEL; FINLAYSON; MCMAHON, 2007). Although it has low rainfall considering other climates, the Brazilian semiarid is one of the rainiest in the world presenting a mean annual precipitation around 750mm (ZANELLA, 2014). The Figure 18 shows the total annual precipitation series of Afrânio and Petrolina between 1950 to 2012.

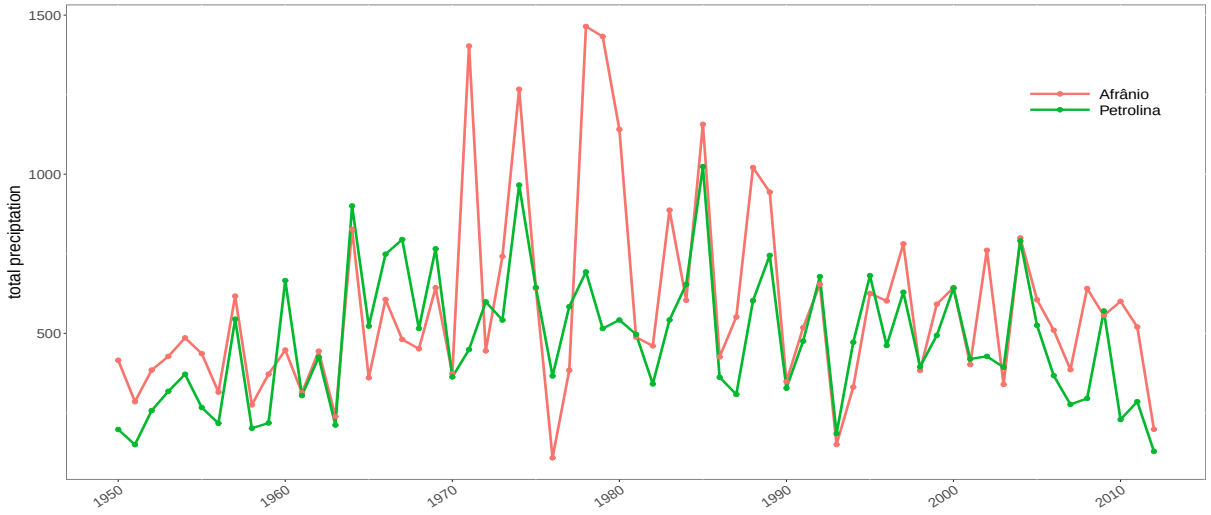

Figure 18 – Total annual precipitation in Afrânio and Petrolina between 1950 to 2012

5.2 Material and methods

The observed series that presents the monthly precipitation from January 1950 to December 2012 was available by Silva (2015) who obtained the data originally from the Institute of Technology of Pernambuco (Itep) and filled in the missing information using the trend surface analysis method, after testing different interpolation techniques.

In order to calculate the SPI, it is necessary to adjust a Probability Density Function (PDF) in the rainfall totals of a region, in which several authors discuss the ideal distribution (SVENSSON; HANNAFORD; PROSDOCIMI, 2017; BLAIN, 2011; STAGGE et al., 2015; BLAIN; MESCHIATTI, 2015). In this work the gamma distribution was used, which is the most used to fit precipitation time series that has PDF given by:

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta} \tag{11}$$

where $\alpha > 0$ is a shape parameter and $\beta > 0$ is a scale parameter.

$$\Gamma(\alpha) = \int_0^{\infty} y^{\alpha-1} e^{-y} \tag{12}$$

were Γ is the gamma function.

Thom (1966) obtained the following optimum solutions for the maximum likelihood estimators of α e β:

$$\hat{\alpha} = \frac{1}{4A}\left(1+\sqrt{1+\frac{4A}{3}}\right) \quad , \tag{13}$$

$$\hat{\beta} = \frac{\bar{x}}{\alpha} \quad , \tag{14}$$

$$A = ln\bar{x} - \frac{\sum lnx}{n} \quad . \tag{15}$$

where n is the number of precipitations observations.

Letting $t = x/\beta$ the cumulative probability the gamma distribution is given by:

$$F(x) = \int_0^x f(x)dx = \frac{1}{\Gamma(\alpha)}\int_0^x t^{\alpha-1}e^{-t}dt \tag{16}$$

The gamma function is undefined for $x = 0$ and naturally precipitation data contain zeros, then cumulative probability becomes:

$$H(x) = q + (1-q)F(x) \tag{17}$$

where q is the probability of a zero in date of precipitation. If m represents the number of zeros in a precipitation series, Thom (1966) indicates that q can be estimated by m/n. Finally, the SPI is generated by standardizing the values obtained in $H(x)$ based on the following equations proposed by Abramowitz e Stegun (1964):

$$\begin{cases} SPI = -\left(t - \frac{c_0+c_1t+c_2t^2}{1+d_1t+d_2t^2+d_3t^3}\right), & \text{for } 0 < H(x) \leq 0.5 \\ SPI = +\left(t - \frac{c_0+c_1t+c_2t^2}{1+d_1t+d_2t^2+d_3t^3}\right), & \text{for } 0.5 < H(x) \leq 1 \end{cases} \tag{18}$$

where

$$\begin{cases} t = \sqrt{ln\left(\frac{1}{H(x)^2}\right)}, & \text{for } 0 < H(x) \leq 0.5 \\ t = \sqrt{ln\left(\frac{1}{(1-H(x))^2}\right)}, & \text{for } 0.5 < H(x) \leq 1 \end{cases} \tag{19}$$

The values of c_0, c_1, c_2, d_1, d_2 and d_3 are, respectively, 2.515517, 0.802853, 0.010328, 1.432788, 0.189269, 0.001308. After determination of the monthly SPI for each of the stations of the two municipalities, the frequencies of SPI were obtained for the classification according to Agnew (2000) proposal, as presented in Table 6 bellow:

Table 6 – Classes of SPI suggested by Agnew (2000)

Code	SPI value	Drought and wet classes
1	SPI≤ -1.65	extreme drought
2	$-1.65 <$SPI≤ -1.28	sereve drought
3	$-1.28 <$SPI≤ -0.84	moderate drought
4	$-0.84 <$SPI≤ 0.84	normal
5	$0.84 <$SPI≤ 1.28	moderate wet
6	$1.28 <$SPI≤ 1.65	severe wet
7	$1.65 <$SPI	extreme wet

The categories will be used to derive the boundaries of first order transition matrices using the FIMCUNI and FIMCAR algorithms. For each municipality it is verified if the synthetic data of SPI are similar to the observed values comparing their observed and synthetic frequencies. Then, descriptive measures of minimum, 1^{st} Quartile, median, 3^{rd} Quartile, maximum, mean and standard deviation of the series will be compared. At the end, the Kolmogorov-Smirnov test will be used to verify if the observed and synthetic series have the same distribution.

5.3 Results

The Figure 19 shows the sample distribution of the SPI using a violin plot (with an internal box plot). The violin plot is a type of graph proposed by Hintze e Nelson (1998) that combines box plot with estimated density. One of the main advantages with regards to the box plot is that it is possible to visualize more than one mode in the data set, such as the SPI distribution of the Afrânio that is bimodal. In Petrolina the box plot in conjunction with the violin plot improves the visualization of the SPI distribution indicating a slight positive skewness.

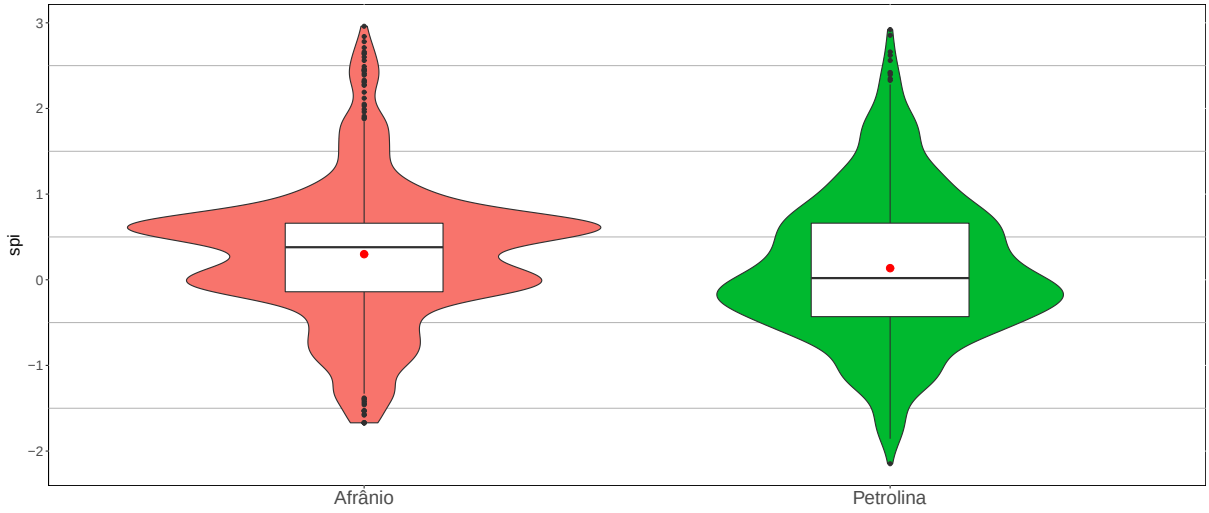

Figure 19 – Violin plot with box plot of SPI of the Afrânio and Petrolina, in which the red dot represents the sample mean of each municipality

After generating the monthly SPI the first-order transition matrices estimated for Afrânio ($\hat{P}_A$) and Petrolina ($\hat{P}_P$) were obtained by maximum likelihood, as shown below:

$$\hat{P}_A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{pmatrix} 0.00 & 0.17 & 0.50 & 0.17 & 0.00 & 0.00 & 0.17 \\ 0.04 & 0.08 & 0.08 & 0.58 & 0.12 & 0.04 & 0.04 \\ 0.00 & 0.03 & 0.06 & 0.77 & 0.03 & 0.03 & 0.09 \\ 0.01 & 0.03 & 0.04 & 0.77 & 0.10 & 0.03 & 0.03 \\ 0.03 & 0.03 & 0.03 & 0.76 & 0.09 & 0.01 & 0.04 \\ 0.00 & 0.00 & 0.00 & 0.73 & 0.05 & 0.00 & 0.23 \\ 0.00 & 0.07 & 0.07 & 0.44 & 0.05 & 0.12 & 0.26 \end{pmatrix} \end{array} \quad \hat{P}_P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{pmatrix} 0.08 & 0.00 & 0.25 & 0.33 & 0.08 & 0.08 & 0.17 \\ 0.11 & 0.00 & 0.00 & 0.67 & 0.11 & 0.00 & 0.11 \\ 0.02 & 0.03 & 0.12 & 0.66 & 0.09 & 0.07 & 0.02 \\ 0.01 & 0.01 & 0.07 & 0.74 & 0.08 & 0.03 & 0.06 \\ 0.03 & 0.02 & 0.08 & 0.61 & 0.14 & 0.06 & 0.08 \\ 0.00 & 0.03 & 0.06 & 0.71 & 0.10 & 0.03 & 0.06 \\ 0.00 & 0.00 & 0.07 & 0.68 & 0.09 & 0.09 & 0.07 \end{pmatrix} \end{array}$$

According to the $\hat{P}_a$ and $\hat{P}_p$ for Afrânio if the current month is classified as "extremely drought" the chance of next month to be classified again as "extreme drought" is zero, but the chance of being classified in another drought state is 67% (sereve drought and moderate drought), while for Petrolina this chance is only 33%. In other states for both Afrânio and Petrolina the chance of next month to be classified a “normal” state is high and varies between 44% to 77% chance.

The Figure 20 shows the frequency of the observed series and the synthetic series generated by the Markov chain for the municipalities of Afrânio and Petrolina. It is possible to notice that, for both municipalities, the observed and generated frequencies are quite close. This indicates that the use of Markov chains are able to reproduce the frequency of series with significant quality.

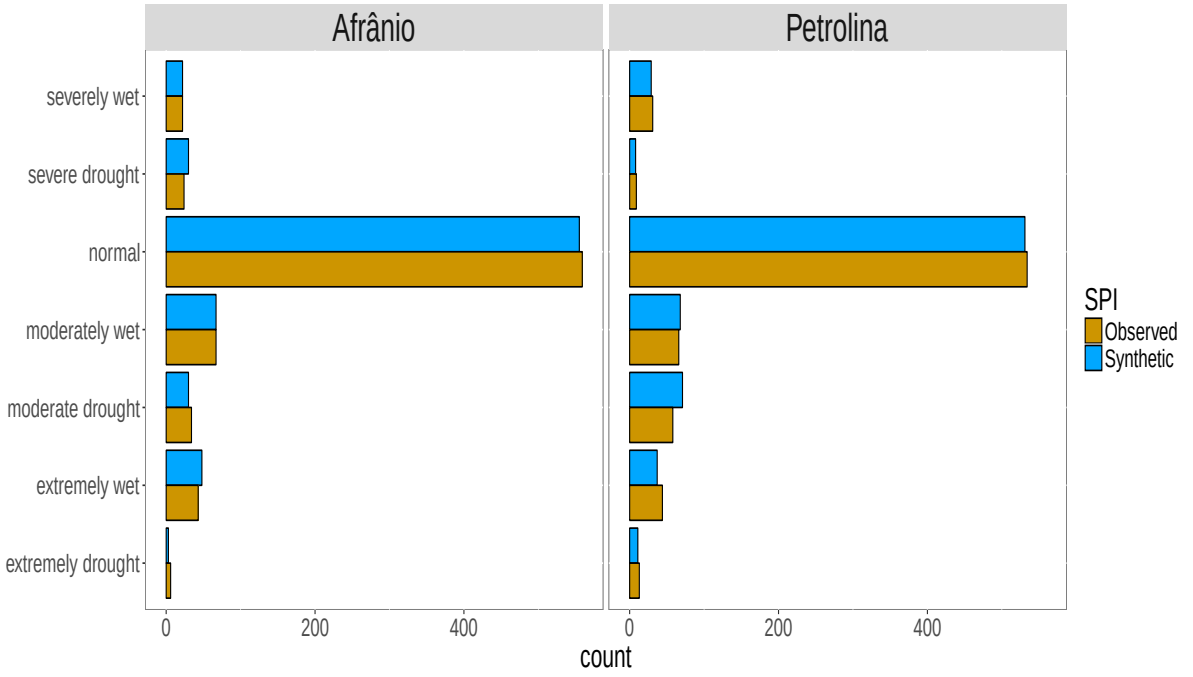

Figure 20 – Observed and synthetic frequency of SPI by category of the municipalities of Afrânio and Petrolina between 1950 and 2012

The observed and synthetic descriptive measures of SPI of the studied municipalities are presented in Table 7. The results indicate that, in both municipalities, the data generated by the FIMCAR algorithm are better approximated to the observed SPI data than those generated by FIMCUNI.

Table 7 – Descriptive measures of observed and generated SPI via FIMCUNI and FIMCAR algorithms for Afrânio and Petrolina

	Afrânio			Petrolina		
Statistics	OBSERVED	FIMCUNI	FINCAR	OBSERVED	FINCUNI	FIMCAR
Minimum	-1.6700	-1.6611	-1.6601	-2.1480	-2.1347	-2.1096
1^{st} Quartile	-0.1400	-0.4379	-0.1466	-0.4303	-0.5213	-0.3944
Median	0.3800	0.1104	0.3274	0.0195	0.0792	0.0504
3^{rd} Quartile	0.6600	0.6947	0.7044	0.6606	0.6743	0.6560
Maximum	2.9600	2.9399	2.8423	2.9215	2.8787	2.8059
Mean	0.2988	0.1809	0.2958	0.1354	0.1382	0.1477
Stand. Dev	0.8064	0.8831	0.8190	0.8474	0.9386	0.8246

The observed density of SPI and the density generated by FIMCUNI and FIMCAR algorithms are shown in Figure 21 for the municipalities studied. For both municipalities the observed p-value of the K-S test, considering 5% of significance, indicates that the synthetic data of SPI generated by FIMCAR algorithm are of the same distribution of the observed data, unlike data generated via FIMCUNI that are rejected by the K-S test.

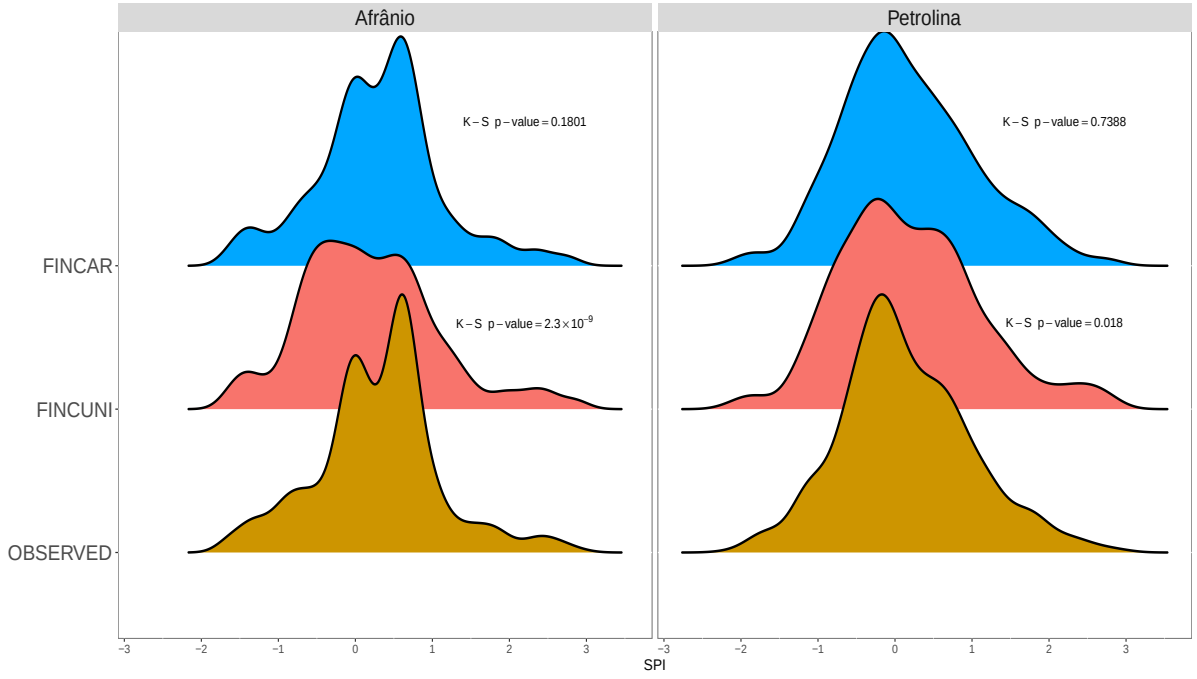

Figure 21 – Density SPI observed and synthetic of the municipalities of Afrânio and Petrolina using FIMCAR and FIMCUNI algorithms

5.4 Conclusions and discussions

It has been observed that Afrânio has twice the chance of remaining in an "extreme drought" situation than Petrolina. The municipality of Petrolina has a 33% chance of remaining in some drought state, which requires special attention due to its economic importance. It is worth noting that Petrolina has been outstanding in the economic area due to the growth in the agricultural exports of the last decades (ARAÚJO; SILVA, 2013).

The results of the simulations indicate that The FIMCAR algorithm was superior to the FIMCUNI algorithm in reproducing the characteristics of the SPI distribution for the two municipalities of this study. It is worth noting that the level of dryness and humidity were represented by 7 (seven) categories, and that only 756 observations were used in each municipality. It is worth mentioning that the observed density of Afrânio is bimodal, which makes it difficult to generate synthetic data by other traditional methods, especially in a closed interval.

6 General conclusions and discussions

According to the simulation results the synthetic data generated by the novel algorithm, proposed in this work, are closer to the observed than the conventional algorithm, both from the point of view of the reproduction capacity of the series, and its density. This improvement is more pronounced when the number of states is small, which is quite common in various cited studies.

This work is the first time that 4^{th} and 5^{th} order Markov chains are used to model wind speed. The relation between number of states and the order of the chain was little explored due to computational limitations discussed in Chapter 2. The increase in the number of states makes it impracticable to use higher order chains in the traditional algorithm. However this problem is minimized with the use of the FIMCAR algorithm because it can reproduce well the observed series density, using a small number of states, independently of the order of the chain.

In Chapter 4 the use of chains using the FIMCAR algorithm was explored for the observed series without trend and seasonality, using eight states. The random series were well reproduced in terms of density, and by restoring the trend and the seasonality of the observed series it was verified that the central measurements were well estimated. It was shown that this methodology presents a significant decrease in error measures when comparing with those obtained in Chapter 3 that used the raw data.

The simulated data only of the random part present an autocorrelation function much closer to the real if compared to the data using only raw data in Chapter 3. In both cases, the 4^{th} and 5^{th} order chains show a subtle improvement in autocorrelation reproduction when compared to the other chain orders. The trend and seasonality extraction can still be studied and compared with more states for the FIMCUNI and FIMCAR algorithms using lower order Markov chains.

The main advantage of the new algorithm is to make the choice of the number of states more flexible (since it produces good results regardless of the number of states), permitting use of smaller samples for adequate characterization of the distribution, independent of their form.

The algorithm proposed in this paper has been applied to finite first-order Markov chains, but can still be adapted to other Markov chain variations. Although the object of study has been wind speed, the proposed algorithm can be used to generate synthetic data of other natural phenomena, especially those that in the literature are categorized in a few classes such as, in chapter 5 in which it was used to generate synthetic SPI data. Application in other natural phenomena can still be studied, such as solar radiation (TUSHAR et al., 2014).

Bibliography

ABRAMOWITZ, M.; STEGUN, I. Handbook of mathematical functions with Formulas, Graphs, and Mathematical Tables (Applied Mathematics Series 55). National Bureau of Standards, Washington, DC, 1964. 31

AGNEW, C. Using the SPI to identify drought. Drought Network News (1994-2001), 2000. xii, 31, 32

AKPINAR, E.; AKPINAR, S. Statistical analysis of wind energy potential on the basis of the Weibull and Rayleigh distributions for Agin-Elazig, Turkey. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, SAGE Publications Sage UK: London, England, v. 218, n. 8, p. 557–565, 2004. 9

AKSOY, H.; TOPRAK, Z. F.; AYTEK, A.; ÜNAL, N. E. Stochastic generation of hourly mean wind speed data. *Renewable energy*, Elsevier, v. 29, n. 14, p. 2111–2131, 2004. 1, 15

ARAÚJO, G. J. F.; SILVA, M. M. Crescimento econômico no semiárido brasileiro: o caso do polo frutícola Petrolina/Juazeiro. *Caminhos de Geografia*, v. 14, n. 46, 2013. 35

AVILÉS, A.; CÉLLERI, R.; SOLERA, A.; PAREDES, J. Probabilistic forecasting of drought events using markov chain-and bayesian network-based models: A case study of an andean regulated river basin. *Water*, Multidisciplinary Digital Publishing Institute, v. 8, n. 2, p. 37, 2016. 29

BANIMAHD, S. A.; KHALILI, D. Factors influencing Markov chains predictability characteristics, utilizing SPI, RDI, EDI and SPEI drought indices in different climatic zones. *Water resources management*, Springer, v. 27, n. 11, p. 3911–3928, 2013. 28

BLAIN, G. C. Standardized precipitation index based on Pearson type iii distribution. *Revista Brasileira de Meteorologia*, SciELO Brasil, v. 26, n. 2, p. 167–180, 2011. 30

BLAIN, G. C.; MESCHIATTI, M. C. Inadequacy of the gamma distribution to calculate the Standardized Precipitation Index. *Revista Brasileira de Engenharia Agrícola e Ambiental*, SciELO Brasil, v. 19, n. 12, p. 1129–1135, 2015. 30

BROKISH, K.; KIRTLEY, J. Pitfalls of modeling wind power using Markov chains. In: IEEE. *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*. [S.l.], 2009. p. 1–6. 15, 22

CARAPELLUCCI, R.; GIORDANO, L. A new approach for synthetically generating wind speeds: A comparison with the Markov chains method. *Energy*, Elsevier, v. 49, p. 298–305, 2013. 1, 15

CARNEIRO, T. C.; CARVALHO, P. C. M. de. Caracterização de potencial eólico: estudo de caso para Maracanaú (CE), Petrolina (PE) e Parnaíba (PI). *Revista Brasileira de Energia Solar*, v. 6, n. 1, 2015. 4

CARTA, J. A.; RAMIREZ, P.; VELAZQUEZ, S. A review of wind speed probability distributions used in wind energy analysis: Case studies in the Canary Islands. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 13, n. 5, p. 933–955, 2009. 1, 14

CASELLA, G.; ROBERT, C. P.; WELLS, M. T. Generalized accept-reject sampling schemes. *Lecture Notes-Monograph Series*, JSTOR, v. 45, p. 342–347, 2004. 3, 7

ETTOUMI, F. Y.; SAUVAGEOT, H.; ADANE, A.-E.-H. Statistical bivariate modelling of wind using first-order Markov chain and Weibull distribution. *Renewable energy*, Elsevier, v. 28, n. 11, p. 1787–1802, 2003. 23

GUTTMAN, N. B. Accepting the standardized precipitation index: a calculation algorithm. *JAWRA Journal of the American Water Resources Association*, Wiley Online Library, v. 35, n. 2, p. 311–322, 1999. 4, 28

HAYES, M.; SVOBODA, M.; WALL, N.; WIDHALM, M. The lincoln declaration on drought indices: universal meteorological drought index recommended. *Bulletin of the American Meteorological Society*, American Meteorological Society, v. 92, n. 4, p. 485–488, 2011. 28

HINTZE, J. L.; NELSON, R. D. Violin plots: a box plot-density trace synergism. *The American Statistician*, Taylor & Francis, v. 52, n. 2, p. 181–184, 1998. 32

HOCAOGLU, F.; GEREK, O.; KURBAN, M. The effect of Markov chain state size for synthetic wind speed generation. In: IEEE. *Probabilistic Methods Applied to Power Systems, 2008. PMAPS'08. Proceedings of the 10th International Conference on*. [S.l.], 2008. p. 1–4. 2, 15, 18

HOEL, P. G.; PORT, S. C.; STONE, C. J. *Introduction to stochastic processes*. [S.l.]: Waveland Press, 1986. 5

HYNDMAN, R. J.; KOEHLER, A. B. Another look at measures of forecast accuracy. *International journal of forecasting*, Elsevier, v. 22, n. 4, p. 679–688, 2006. 10

KAMINSKY, F.; KIRCHHOFF, R.; SYU, C.; MANWELL, J. A comparison of alternative approaches for the synthetic generation of a wind speed time series. *Journal of solar energy engineering*, American Society of Mechanical Engineers, v. 113, n. 4, p. 280–289, 1991. 1, 15

KANTAR, Y. M.; ILHAN, U.; YENİLMEZ, I.; IBRAHIM, A. A study on estimation of wind speed distribution by using the Modified Weibull distribution. *Bilişim Teknolojileri Dergisi*, v. 9, n. 2, p. 63, 2016. 14

KANTAR, Y. M.; USTA, I. Analysis of wind speed distributions: Wind distribution function derived from minimum cross entropy principles as better alternative to Weibull function. *Energy Conversion and Management*, Elsevier, v. 49, n. 5, p. 962–973, 2008. 9

KANTAR, Y. M.; USTA, I. Analysis of the upper-truncated Weibull distribution for wind speed. *Energy Conversion and Management*, Elsevier, v. 96, p. 81–88, 2015. 9

KARATEPE, S.; CORSCADDEN, K. W. Wind speed estimation: Incorporating seasonal data using Markov chain models. *ISRN Renewable Energy*, Hindawi Publishing Corporation, v. 2013, p. 1–9, 2013. 2, 15

LEI, M.; SHIYAN, L.; CHUANWEN, J.; HONGLING, L.; YAN, Z. A review on the forecasting of wind speed and generated power. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 13, n. 4, p. 915–920, 2009. 1

MAKRIDAKIS, S. Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, Elsevier, v. 9, n. 4, p. 527–529, 1993. 10

MASSERAN, N.; RAZALI, A. M.; IBRAHIM, K.; ZAHARIM, A.; SOPIAN, K. The probability distribution model of wind speed over East Malaysia. *Research Journal of Applied Sciences, Engineering and Technology*, Maxwell Scientific Publications, v. 6, n. 10, p. 1774–1779, 2013. 14

MCKEE, T. B.; DOESKEN, N. J.; KLEIST, J. et al. The relationship of drought frequency and duration to time scales. In: AMERICAN METEOROLOGICAL SOCIETY BOSTON, MA. *Proceedings of the 8th Conference on Applied Climatology*. [S.l.], 1993. v. 17, n. 22, p. 179–183. 4, 28

MELO, E.; ARAGÃO, M. S.; CORREIA, M. Regimes do vento à superfície na área de Petrolina, Submédio do Rio São Francisco. *Revista Brasileira de Meteorologia*, v. 28, n. 3, p. 229–241, 2013. 3

MUSELLI, M.; POGGI, P.; NOTTON, G.; LOUCHE, A. First order Markov chain model for generating synthetic "typical days" series of global irradiation in order to design photovoltaic stand alone systems. *Energy Conversion and Management*, Elsevier, v. 42, n. 6, p. 675–687, 2001. 23

NFAOUI, H.; ESSIARAB, H.; SAYIGH, A. A stochastic Markov chain model for simulating wind speed time series at Tangiers, Morocco. *Renewable Energy*, Elsevier, v. 29, n. 8, p. 1407–1418, 2004. 2

PAPAEFTHYMIOU, G.; KLOCKL, B. MCMC for wind power simulation. *IEEE transactions on energy conversion*, IEEE, v. 23, n. 1, p. 234–240, 2008. 15, 22

PEEL, M. C.; FINLAYSON, B. L.; MCMAHON, T. A. Updated world map of the Köppen-Geiger climate classification. *Hydrology and earth system sciences discussions*, v. 4, n. 2, p. 439–473, 2007. 30

PESCH, T.; SCHRÖDERS, S.; ALLELEIN, H.; HAKE, J. A new Markov-chain-related statistical approach for modelling synthetic wind power time series. *New journal of physics*, IOP Publishing, v. 17, n. 5, p. 055001, 2015. 2, 15, 23

PETRE, I. A.; REBENCIUC, M.; CIUCU, S. C. The use of Markov chains in forecasting wind speed: Matlab source code and applied case study. *Computational Methods in Social Sciences*, Nicolae Titulescu University Editorial House, v. 4, n. 2, p. 44, 2016. 2, 15

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2018. Disponível em: <https://www.R-project.org/>. 4

RAHMAT, S. N.; JAYASURIYA, N.; BHUIYAN, M. A. Short-term droughts forecast using Markov chain model in Victoria, Australia. *Theoretical and Applied Climatology*, Springer, v. 129, n. 1-2, p. 445–457, 2017. 29

SAHIN, A.; SEN, Z. First order Markov chain approach to wind speed modeling. *Wind Engineering and Industrial Aerodynamic*, v. 89, n. 3-4, p. 263–270, 2001. 1, 3, 5, 6, 9, 15, 18

SANUSI, W.; JEMAIN, A. A.; ZIN, W. Z. W.; ZAHARI, M. The drought characteristics using the first-order homogeneous Markov chain of monthly rainfall data in peninsular Malaysia. *Water resources management*, Springer, v. 29, n. 5, p. 1523–1539, 2015. 28

SAWYER, S.; DYRHOLM, M. *Global Wind Report: Annual Market Update 2017*. [S.l.], 2018. 1

SCHOLZ, F. W.; STEPHENS, M. A. K-sample Anderson–Darling tests. *Journal of the American Statistical Association*, Taylor & Francis Group, v. 82, n. 399, p. 918–924, 1987. 10

SFETSOS, A. A comparison of various forecasting techniques applied to mean hourly wind speed time series. *Renewable energy*, Elsevier, v. 21, n. 1, p. 23–35, 2000. 1

SHAMSHAD, A.; BAWADI, M.; HUSSIN, W. W.; MAJID, T.; SANUSI, S. First and second order Markov chain models for synthetic generation of wind speed time series. *Energy*, Elsevier, v. 30, n. 5, p. 693–708, 2005. 2, 10, 15, 23

SILVA, A. S. *Ferramentas para modelagem e interpolação de dados ambientais em escala regional*. Tese (Doutorado) — Rural Federal University of Pernambuco, Recife-PE, 7 2015. 30

SOHONI, V.; GUPTA, S.; NEMA, R. A comparative analysis of wind speed probability distributions for wind power assessment of four sites. *Turkish Journal of Electrical Engineering & Computer Sciences*, The Scientific and Technological Research Council of Turkey, v. 24, n. 6, p. 4724–4735, 2016. 14

SOMAN, S. S.; ZAREIPOUR, H.; MALIK, O.; MANDAL, P. A review of wind power and wind speed forecasting methods with different time horizons. p. 1 – 8, 10 2010. 1

STAGGE, J. H.; TALLAKSEN, L. M.; GUDMUNDSSON, L.; LOON, A. F. V.; STAHL, K. Candidate distributions for climatological drought indices (SPI and SPEI). *International Journal of Climatology*, Wiley Online Library, v. 35, n. 13, p. 4027–4040, 2015. 30

STEINEMANN, A. Drought indicators and triggers: a stochastic approach to evaluation 1. *JAWRA Journal of the American Water Resources Association*, Wiley Online Library, v. 39, n. 5, p. 1217–1233, 2003. 29

SVENSSON, C.; HANNAFORD, J.; PROSDOCIMI, I. Statistical distributions for monthly aggregations of precipitation and streamflow in drought indicator applications. *Water Resources Research*, Wiley Online Library, v. 53, n. 2, p. 999–1018, 2017. 30

TANG, J.; BROUSTE, A.; TSUI, K. L. Some improvements of wind speed Markov chain modeling. *Renewable Energy*, Elsevier, v. 81, p. 52–56, 2015. 1

TASCIKARAOGLU, A.; UZUNOGLU, M. A review of combined approaches for prediction of short-term wind speed and power. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 34, p. 243–254, 2014. 1

TASHMAN, L. J. Out-of-sample tests of forecasting accuracy: an analysis and review. *International journal of forecasting*, Elsevier, v. 16, n. 4, p. 437–450, 2000. 10

TEIXEIRA-GANDRA, C. F. A.; DAMÉ, R. d. C. F.; SILVA, G. M. da. Stochastic modeling using Markov chain on the forecast standardized precipitation index. *Científica*, v. 45, n. 2, p. 137–144, 2017. 29

THOM, H. Some methods of climatological analysis. *WMO technics/note number*, n. 81, p. 16–22, 1966. 31

TUSHAR, W.; HUANG, S.; YUEN, C.; ZHANG, J. A.; SMITH, D. B. Synthetic generation of solar states for smart grid: A multiple segment Markov chain approach. In: IEEE. *Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES*. [S.l.], 2014. p. 1–6. 36

WAND, M. P.; JONES, M. C. *Kernel smoothing*. [S.l.]: Chapman and Hall/CRC, 1994. 7

WANG, J.; HU, J.; MA, K. Wind speed probability distribution estimation and wind energy assessment. *Renewable and sustainable energy reviews*, Elsevier, v. 60, p. 881–899, 2016. 14

WANG, J.; TSANG, W. W.; MARSAGLIA, G. Evaluating Kolmogorov's distribution. *Journal of Statistical Software*, American Statistical Association, v. 8, n. 18, 2003. 10

WU, T.; AI, X.; LIN, W.; WEN, J.; WEIHUA, L. Markov chain Monte Carlo method for the modeling of wind power time series. In: IEEE. *Innovative Smart Grid Technologies-Asia (ISGT Asia), 2012 IEEE*. [S.l.], 2012. p. 1–6. 2, 15, 18

XIE, K.; LIAO, Q.; TAI, H.-M.; HU, B. Non-Homogeneous Markov Wind Speed Time Series Model Considering Daily and Seasonal Variation Characteristics. *IEEE Transactions on Sustainable Energy*, IEEE, v. 8, n. 3, p. 1281–1290, 2017. 15

YU, Z.; TUZUNER, A. Wind speed modeling and energy production simulation with Weibull sampling. In: IEEE. *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*. [S.l.], 2008. p. 1–6. 1

ZANELLA, M. E. Considerações sobre o clima e os recursos hídricos do semiárido nordestino. *Caderno Prudentino de Geografia*, n. 36, p. 126–142, 2014. 30

ZARGAR, A.; SADIQ, R.; NASER, B.; KHAN, F. I. A review of drought indices. *Environmental Reviews*, NRC Research Press, v. 19, n. NA, p. 333–349, 2011. 28

APPENDIX A – Referring to the chapters

A.1 Error measures in ch. 2

Table 8 – Error measures of simulations using FINCUNI and FINCAR in Figure 6

	FIMCUNI				FIMCAR			
	Error measures							
Nº States	MAE	MSE	RMSE	sMAPE	MAE	MSE	RMSE	sMAPE
8	2.006	6.347	2.519	0.111	1.946	5.989	2.447	0.107
12	1.965	6.084	2.467	0.108	1.942	5.946	2.438	0.107
16	1.975	6.197	2.489	0.108	1.960	6.105	2.471	0.107
20	1.994	6.268	2.504	0.110	1.985	6.214	2.493	0.109
24	1.955	5.985	2.446	0.108	1.948	5.943	2.438	0.107
28	1.956	6.050	2.460	0.108	1.951	6.015	2.453	0.107

A.2 Error Measures in ch. 3

Table 9 – Error measures of simulations using FIMCAR algorithm in Figure 8

	Numbers of States							
	4				6			
Order	MAE	MSE	RMSE	sMAPE	MAE	MSE	RMSE	sMAPE
1^{st}	1.983	6.182	2.486	0.109	1.938	5.909	2.431	0.106
2^{nd}	1.951	5.997	2.449	0.107	1.942	5.949	2.439	0.107
3^{rd}	1.959	6.100	2.470	0.108	1.975	6.201	2.490	0.108
4^{th}	1.970	6.119	2.474	0.108	1.973	6.112	2.472	0.108
5^{th}	1.975	6.113	2.472	0.109	1.947	5.948	2.439	0.107
	8				10			
1^{st}	1.956	6.036	2.457	0.107	1.943	5.975	2.444	0.107
2^{nd}	1.949	6.001	2.450	0.107	1.958	6.002	2.450	0.107
3^{rd}	1.972	6.085	2.467	0.109	1.956	6.090	2.468	0.107
4^{th}	1.970	6.128	2.475	0.108	1.965	6.151	2.480	0.107
5^{th}	1.946	6.065	2.463	0.107	1.958	6.088	2.467	0.108

A.3 Histogram for algorithm FIMCUNI in ch.3

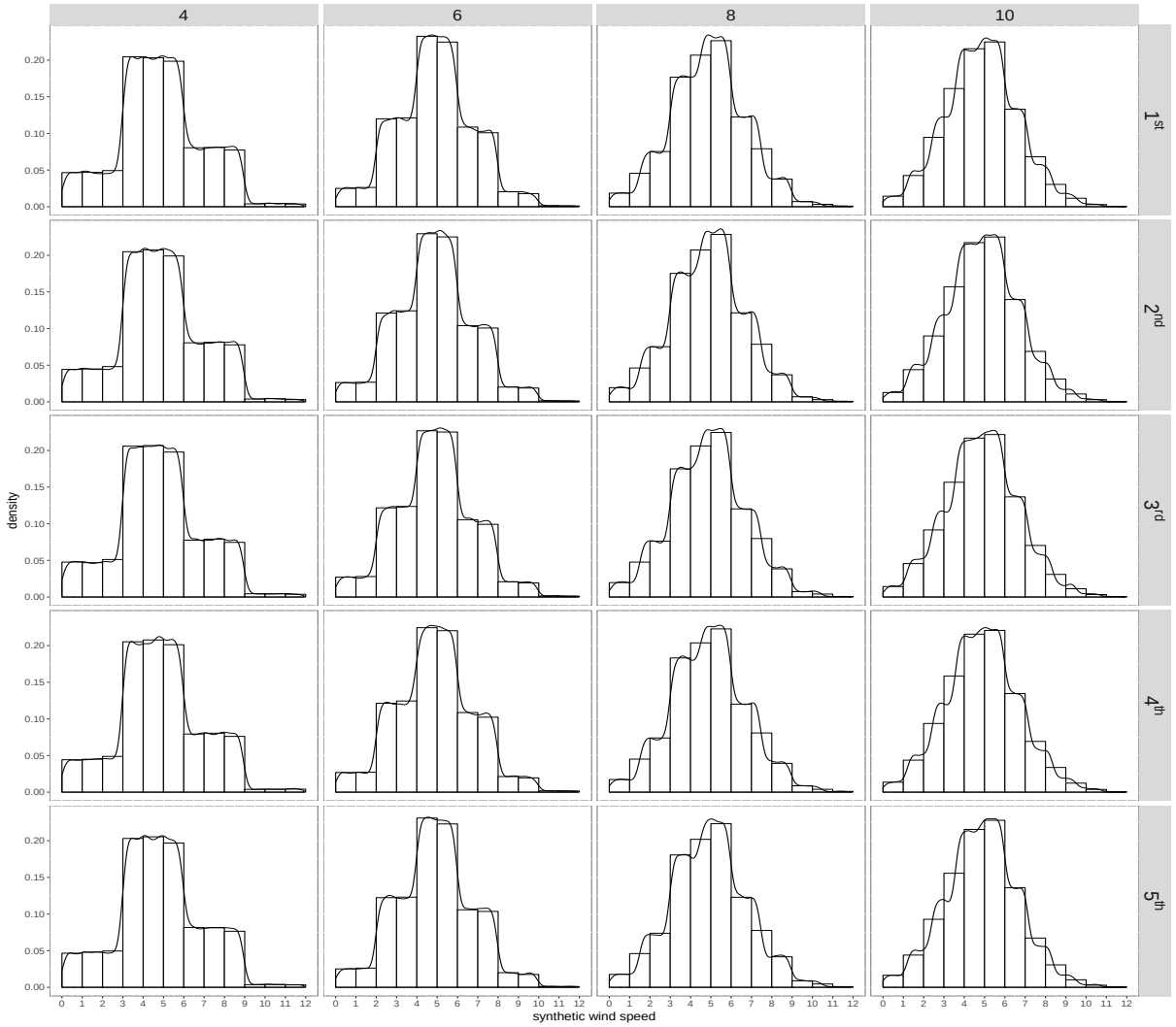

Figure 22 – Histograms with the distribution of the synthetic data of each combination of number of states (in the columns) and chain order (in the lines) the synthetic wind speed data using the FIMCUNI algorithm

A.4 Error Measures in ch.4

Table 10 – Error measures of simulations using FIMCAR algorithm in Figure 14

	Error measures							
	Raw				Random			
Order	MAE	MSE	RMSE	sMAPE	MAE	MSE	RMSE	sMAPE
1^{st}	1.956	6.036	2.457	0.107	1.336	2.888	1.699	0.078
2^{nd}	1.949	6.001	2.450	0.107	1.326	2.843	1.686	0.076
3^{rd}	1.972	6.085	2.467	0.109	1.329	2.888	1.699	0.081
4^{th}	1.970	6.128	2.475	0.108	1.332	2.911	1.706	0.076
5^{th}	1.946	6.065	2.463	0.107	1.354	3.020	1.738	0.077

APPENDIX B – Commands in R

```
# Function that generates states
estados=function(dados,seqest){
                              est=NULL
                              k=seq(1,(length(seqest)-1),1)
                              for(i in 1:length(dados)){
                                                        est[i]=max(1,k[seqest<=dados[i]],1)
                                                        }
                              return(est)
                              }
sequencia=function(x) seq(min(dados),max(dados)+0.001,length=x) # generating sequences
# Generating states using the sequence function within the states function
estger=estados(dados,sequencia(n+1)) #  "n" is number of states

# Function generating synthetic wind speed data via Finite MC FIMCAR and FIMCUNI
# Input (in this order): data, generated states, chain order (from 1st to 5th) and state boundaries.
# Output (in this order and in list form): Synthetic data generated by FIMCUNI, Synthetic data generated by FIMCAR,
# Generated States, Probability Transition Matrix and Count Transition Matrix.


FIMC=function(dados,x,ord,ampli){
                        den=density(dados)
                        nc=length(ampli)-1
                        p <-  array(0,dim=rep(nc,ord+1))

if(ord==1) for (tx in 1:(length(x) - ord)) p[x[tx+0],x[tx+1]]=
p[x[tx+0],x[tx+1]]+1
if(ord==2) for (tx in 1:(length(x) - ord)) p[x[tx+0],x[tx+1],x[tx+2]]=
p[x[tx+0],x[tx+1],x[tx+2]]+1
if(ord==3) for (tx in 1:(length(x) - ord)) p[x[tx+0],x[tx+1],x[tx+2],x[tx+3]]=
p[x[tx+0],x[tx+1],x[tx+2],x[tx+3]]+1
if(ord==4) for (tx in 1:(length(x) - ord)) p[x[tx+0],x[tx+1],x[tx+2],x[tx+3],x[tx+4]]=
p[x[tx+0],x[tx+1],x[tx+2],x[tx+3],x[tx+4]]+1
if(ord==5) for (tx in 1:(length(x) - ord)) p[x[tx+0],x[tx+1],x[tx+2],x[tx+3],x[tx+4],x[tx+5]]=
p[x[tx+0],x[tx+1],x[tx+2],x[tx+3],x[tx+4],x[tx+5]]+1

        np=matrix(p,ncol=nc)
        tl=rowSums(np)
        mattra=np/tl
        mattra[is.nan(mattra)] = 0
# renomando as linhas
        if(ord==1) rownames(mattra)=paste(1:nc,sep="-")
        if(ord==2) rownames(mattra)=paste(rep(1:nc,nc^1),rep(1:nc,1,each=nc),sep="-")
        if(ord==3) rownames(mattra)=paste(rep(1:nc,nc^2),rep(1:nc,1,each=nc),rep(1:nc,1,each=nc^2),sep="-")
        if(ord==4) rownames(mattra)=paste(rep(1:nc,nc^3),rep(1:nc,nc^2,each=nc),rep(1:nc,nc,each=nc^2),
rep(1:nc,1,each=nc^3),sep="-")# renomando as linhas
        if(ord==5) rownames(mattra)=paste(rep(1:nc,nc^4),rep(1:nc,nc^3,each=nc),rep(1:nc,nc^2,each=nc^2),
rep(1:nc,nc,each=nc^3),rep(1:nc,1,each=nc^4),sep="-")# renomando as linhas
        matac=t(apply(mattra, 1, cumsum)) # matriz acumulada

k=seq(1:nc)
        E1=NULL;pvv=as.character(x[1:ord])
        if(ord==1) E1[1]=paste(unlist(strsplit(pvv[1], "[-]")),sep="-")
        if(ord==2) E1[1]=paste(unlist(strsplit(pvv[1], "[-]")),unlist(strsplit(pvv[2], "[-]")),sep="-")
        if(ord==3) E1[1]=paste(unlist(strsplit(pvv[1], "[-]")),unlist(strsplit(pvv[2], "[-]")),
```

```
unlist(strsplit(pvv[3], "[-]")),sep="-")
        if(ord==4) E1[1]=paste(unlist(strsplit(pvv[1], "[-]")),unlist(strsplit(pvv[2], "[-]")),
unlist(strsplit(pvv[3], "[-]")),unlist(strsplit(pvv[4], "[-]")),sep="-")
        if(ord==5) E1[1]=paste(unlist(strsplit(pvv[1], "[-]")),unlist(strsplit(pvv[2], "[-]")),
unlist(strsplit(pvv[3], "[-]")),unlist(strsplit(pvv[4], "[-]")),unlist(strsplit(pvv[5], "[-]")),sep="-")

  E2=NULL;E2[1]=min(k[matac[E1[1],]>runif(1)])
        vel1=vel2=NULL;vel1[1]=vel2[1]=runif(1,ampli[E2[1]],ampli[E2[1]+1])
        for(i in 1:(length(x)-ord-1)){
                    ale=runif(1) # gera um número entre zero e um
                    if(ord==1) E1[i+1]=E2[i]
                    if(ord==2) E1[i+1]=paste(unlist(strsplit(E1[i], "[-]"))[2],E2[i],sep="-")
                    if(ord==3) E1[i+1]=paste(unlist(strsplit(E1[i], "[-]"))[2],
 unlist(strsplit(E1[i], "[-]"))[3],E2[i],sep="-")
                    if(ord==4) E1[i+1]=paste(unlist(strsplit(E1[i], "[-]"))[2],unlist(strsplit(E1[i], "[-]"))[3],
 unlist(strsplit(E1[i], "[-]"))[4],E2[i],sep="-")
                    if(ord==5) E1[i+1]=paste(unlist(strsplit(E1[i], "[-]"))[2],unlist(strsplit(E1[i], "[-]"))[3],
 unlist(strsplit(E1[i], "[-]"))[4],unlist(strsplit(E1[i], "[-]"))[5],E2[i],sep="-")
                    E2[i+1]= min(k[matac[E1[i+1],]>ale])
 # FIMCUNI
          vel1[i+1]=runif(1,ampli[E2[i+1]],ampli[E2[i+1]+ 1])
 # FIMCAR
 pxy=c(0,0)
 repeat{
pxy=c(runif(1,ampli[E2[i+1]],ampli[E2[i+1]+1]), runif(1,0,max(den$y[den$x>=ampli[E2[i+1]]],
den$y[den$x<=ampli[E2[i+1]+1]])))
vel2[i+1]=pxy[1] # termina a rotina antes pausar ok
if(pxy[2]<=den$y[min(which(den$x>=pxy[1]))]) break
       }
}
                                L=list(vel1,vel2,E2,mattra,np)
return(L)
                                   }

# Function that calculates error measures
M_A=function(esti,obs) {
MAE=mean(abs(obs-esti))
MSE=sum(abs(obs-esti)^2)/length(obs)
RMSE=sqrt(mean((obs-esti)^2))
sMAPE=mean(abs(obs-esti)/(abs(obs)+abs(esti))/2)
L=data.frame(MAE,MSE,RMSE,sMAPE)
return(L)
}

# Function that checks if the events of an MC are independent (Enter with count transition matrix)
Indmc=function(M){
P=round(M/rowSums(M),4)
k=ncol(P)
alps=matrix(rep(0,k*k),ncol=k)
for(i in 1:k){
for(j in 1:k){
alps[i,j]=M[i,j]*log(P[i,j]/(colSums(M)[j]/sum(M)))  } }
alps[is.nan(alps)] = 0
alp=sum(alps)*2 ; chi=qchisq(0.95,(k-1)^2) ; pval=1-pchisq(alp,(k-1)^2)
return(list(paste("Alpha =",round(alp,2),sep=" "), paste("Quantil a 5% = ",round(chi,2), sep=" "),
            paste("P_valor = ",pval,sep=" ")))
             }
```

```
############################ Function that generates monthly SPI for 4 distributions ####################
## Enter with data.frame of two variables: 1st A numerical sequence of months, 2nd precipitation data
## Distributions used gamma (default), lognormal, logistic and Inverse Gaussian.
## Required "fitdistrplus" package for use of the "fitdist" function.

library(fitdistrplus)

SPI_men=function(dado,distr="gamma"){
myspi=NULL
for(k in 1:12){
x=dado[dado[,1]==k,] #
x=x[,2] # selecting precipitation data
xsz=x[x>0] # removing zeros
q=(length(x)-length(xsz))/length(x)  # proportion of zeros

if(distr=="gamma"){
A=log(mean(xsz))-sum(log(xsz))/length(xsz)
alpha=(1/(4*A))*(1+sqrt(1+(4*A/3)))
Beta=mean(xsz)/alpha
hx=q+(1-q)*pgamma(x, alpha, 1/Beta) # scale=1/Beta is a gamma parameter
}

if(distr=="lognormal"){
ajln=fitdist(xsz,"lnorm")
hx=q+(1-q)*plnorm(x, ajln$estimate[1],ajln$estimate[2]) # perfeito #
}

if(distr=="weibull"){
ajw=fitdist(xsz,"weibull")
hx=q+(1-q)*pweibull(x, ajw$estimate[1],ajw$estimate[2]) # perfeito
}

if(distr=="invgauss"){
mxsz=mean(xsz);sdxsz=sd(xsz);vxsz=var(xsz)
fitig=fitdist(xsz, "invgauss",start=list(mxsz,sdxsz))
hx=q+(1-q)*pinvgauss(x, fitig$estimate[1],fitig$estimate[2]) # perfeito
}

t1=ifelse(hx<=0.5, sqrt(log(1/(hx)^2)), sqrt(log(1/(1-hx)^2)) )
tempspi=NULL
for (i in 1:length(x)){
if(hx[i]<=0.5) tempspi[i]=-1*(t1[i]-(2.515517+0.802853*t1[i]+0.010328*(t1[i])^2)/
(1+1.432788*t1[i]+0.189269*(t1[i])^2+0.001308*(t1[i])^3))
if(hx[i]>0.5) tempspi[i]=t1[i]-(2.515517+0.802853*t1[i]+0.010328*(t1[i])^2)/
(1+1.432788*t1[i]+0.189269*(t1[i])^2+0.001308*(t1[i])^3)
}
myspi[seq(k,dim(data)[1],12)]=tempspi
                        }
return(myspi)
}
```