

# Forecasting economic time series using chaotic neural networks

Victor H. Gonçalves<sup>1</sup>, João L. G. Rosa<sup>1</sup>

<sup>1</sup> Instituto de Ciências Matemáticas e de Computação  
Universidade de São Paulo (USP) – São Carlos, SP – Brazil

vihg@usp.br, joaoluis@icmc.usp.br

***Abstract.** This paper describes the application of KIII, a biologically more plausible neural network model, for forecasting economic time series. K-sets are connectionist models based on neural populations and have been used in many machine learning applications. In this paper, this method was applied to IPCA, a Brazilian consumer price index surveyed by IBGE. The values ranged from August 1994 to June 2017. Experiments were performed using four non-parametric models and seven parametric methods. The statistical metric RMSE was used to compare methods performance. Freeman KIII sets worked well as a filter, but it was not a good prediction method. This paper contributes with the use of non-parametrics models for forecasting inflation in a developing country.*

## 1. Introduction

This paper describes the application of a connectionist approach, Freeman K-sets, for predicting Brazilian economic time series. These time series were extracted from the Time Series Management System of the Central Bank - v2.1.

Economic time series are useful for two purposes. The first one is the possibility of using the time series data to make future forecasts. This is the focus of this paper. The second one is to provide a way of presenting the facts associated with the movements of the economic time series [Harvey 1985]. Economic time series also serve for the Central Bank to plan economic policy [Stock and Watson 2005].

The forecast of the consumer price index is a rather complex task because the inflation indices of developing countries are high and volatile. Before the Plano Real, they were extremely high and much more volatile, which made it impossible to predict these indices. Soon after the Plano Real, inflation declined and its volatility too, but both remained high, which is an ideal scenario for neural networks, especially Freeman K-sets, to predict more realistic values than the other methods. Only in recent years has volatility declined further, allowing exponential smoothing methods to perform better in predicting price indices. But even so, this volatility is high enough that Freeman KIII-sets can perform well in this task.

Freeman K-sets form a nested hierarchy of population dynamics models of  $\sim 10^4$  neurons with  $\sim 10^8$  synapses that mediate between the microscopic activity of small neural networks and the macroscopic activity of the whole brain. The topology of the connections is modeled by networks of populations of excitatory and inhibitory neurons. Dynamics are approximated by nonlinear parts of ordinary differential equations (ODE),

with distributed parameters to describe dynamics of neural populations starting from microcolumns to hemispheres [Kozma and Freeman 2001]. In this work we will use a hierarchical approach to spatio-temporal neurodynamics, based on Freeman K-sets.

The objective of this work is to verify if the use of Freeman K-sets contributes to more realistic previsions of economic time series in relation to models that already exist in the scientific literature because these models work well with problems where there are unstable time series and difficult to predict, which is the case of forecasting Brazilian economic time series.

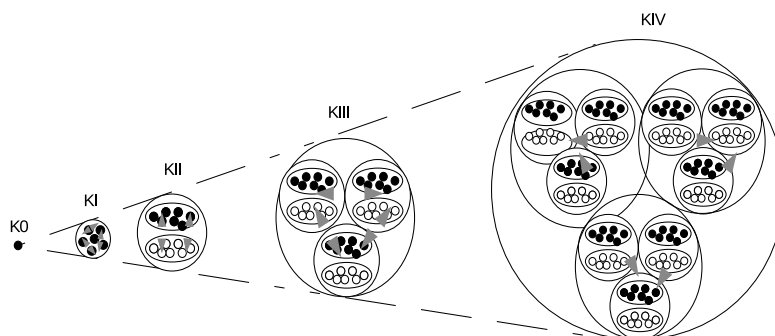
For the comparison, four non-parametric models were used (KIII, continuous kNN, classical ANN, and SVM) and seven parametric methods: ARIMA, SARIMA, Moving Average, SES, Holt, Additive Holt–Winters, and Multiplicative Holt–Winters. KIII is expected to perform better in most experiments. Performance is evaluated using the RMSE metric. It should be noted that K models have already been used successfully for the prediction of time series [Kozma and Beliaev 2004][Li and Kozma 2003].

The remainder of this paper is organized as follows. Section II describes the theoretical background, including a description of K-sets model and IPCA index. Section III introduces the proposed methods. Section IV shows the results obtained with the execution of our experiments. Section V presents concluding remarks.

## 2. Theoretical Background

### 2.1. K-sets models

K-sets were introduced by Walter Freeman in the 1970's, and their name came from Aharon Kachalsky, one of the first researchers of neural dynamics [Freeman 1975]. K models use ordinary differential equations with distributed parameters to describe dynamics of neural populations starting from microcolumns to hemispheres. The hierarchy of models is constructed starting from a basic unit (K0). Networks of greater neural complexity are constructed from this base unit (Figure 1). K-sets models represents a mesoscopic level, intermediate between microscopic neurons and macroscopic brain [Kozma et al. 2013].



**Figure 1. Hierarchy of neural network K models, from K0 to KIV - Adapted from [Rosa and Piazzini 2016]**

K0 sets represent the smaller mass of non-interacting neurons and are the more basic building blocks for K-sets [Freeman 1975]. They represent a neural population of about 10,000 neurons and are motivated by cortical microcolumns, the structure of columns of neurons in the brain. The K0 sets are

governed by a point attractor that remains in equilibrium except when disturbed [Kozma et al. 2013][Rosa and Piazzentin 2016][Piazzentin and Rosa 2015].

KI sets are the next level in the hierarchy of models of neuron populations. KI sets are the simplest form of interaction among neuron population and are built by two inhibitory or excitatory K0 sets with positive feedback. The KI dynamics is a convergence to the non-zero fixed point. If a KI has the necessary connection density, then it is able to maintain a non-zero state of background activity by mutual excitation (or inhibition) [Kozma et al. 2013][Rosa and Piazzentin 2016][Piazzentin and Rosa 2015].

KII sets are formed by interactions between KI sets with negative feedback. KII sets have four types of interactions: excitatory-excitatory, inhibitory-inhibitory, excitatory-inhibitory, and inhibitory-excitatory. They give rise to limit cycle oscillations due to their feedback between neural populations. KI sets that form the KII, when active, oscillate at a characteristic frequency in response to an input stimulus. The more interesting behavior of KII arises when it is built by KIs with inverse polarities. The connections of the KII have fixed weights, and KIIs form oscillators when these connections are configured [Kozma et al. 2013][Rosa and Piazzentin 2016][Piazzentin and Rosa 2015].

KIII is the lowest level of behavioral analysis and the first level with practical applications of K-sets neural networks. KIII sets are built of three interacting layers of KII. The first layer corresponds to olfactory bulb (OB) and has the size of the input of the network. The OB bulb then connects to anterior olfactory nuclei (AON) and prepiriform cortex (PC), the second and third layers respectively. AON and PC have connections between themselves and back to the OB. As the output is dispersed in the PC layer of the KIII, a second algorithm is typically used to translate or classify this output. The most commonly used algorithms for this task are the k-nearest neighbors and the linear discriminant analysis. Learning in KIII is done through the Hebbian learning rule [Piazzentin and Rosa 2014]. KIII sets generate chaotic, broadband oscillations as background activity by the negative and positive feedback combined between various populations of KII with incommensurate frequencies [Kozma et al. 2013]. KIII also can serve as an associative memory, encoding input data in spatio-temporal AM patterns [Kozma and Freeman 2001].

## 2.2. IPCA

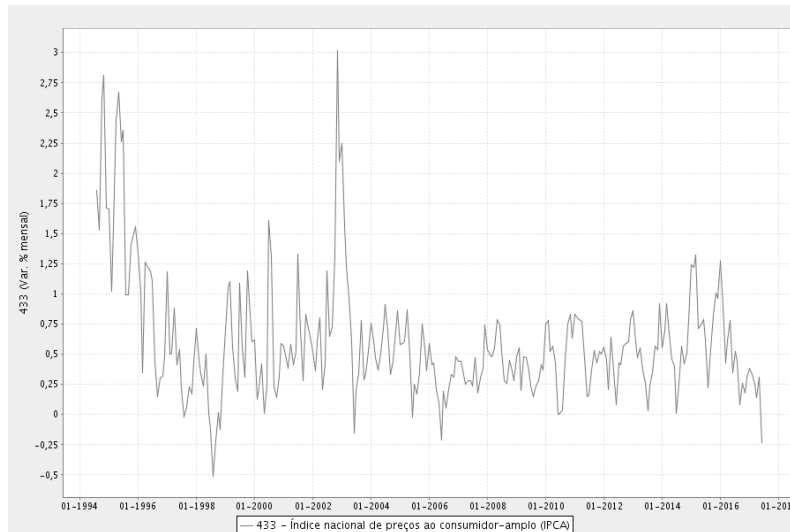
IPCA (Índice de Preços ao Consumidor Amplo - Broad National Consumer Price Index) is a consumer price index surveyed by IBGE (Instituto Brasileiro de Geografia e Estatística - Brazilian Institute of Geography and Statistics), the official statistical institute of the Brazilian government. IPCA is part of SNIPC (Sistema Nacional de Índices de Preços ao Consumidor - National System of Consumer Price Indices) [Instituto Brasileiro de Geografia e Estatística 2017].

SNIPC is a IBGE's system which consists of a combination of processes designed to produce consumer price indices. The system covers the metropolitan areas of Rio de Janeiro, Porto Alegre, Belo Horizonte, Recife, São Paulo, Belém, Fortaleza, Salvador, Curitiba, and Vitória, as well as Distrito Federal and the municipalities of Goiânia and Campo Grande. The survey began in January 1979 for Rio de Janeiro; June 1979 for Porto Alegre, Belo Horizonte and Recife; January 1980 to São Paulo, Brasília and Belém; October 1980 to Fortaleza, Salvador and Curitiba; January 1991 to Goiânia; and January

2014 to Vitória and Campo Grande [Instituto Brasileiro de Geografia e Estatística 2017].

The integral collection of prices occurs every 30-day period that is segmented, without interruption, in 4 subperiods. Each of them lasts about 7 days with dates defined by the SNIPC Annual Collection Calendar. The prices obtained are those actually charged to the consumer for cash payment. The survey is conducted in commercial establishments, service providers, households, and utility concessionaires [Instituto Brasileiro de Geografia e Estatística 2017].

The target population of the IPCA refers to families living in urban areas of the SNIPC regions with incomes of 1 to 40 minimum wages (\$245.00[cal 2018b] to \$9,800.00[cal 2018a], approximately) regardless of income source [Instituto Brasileiro de Geografia e Estatística 2017]. Figure 2 shows IPCA indices from August 1994 to June 2017.



**Figure 2. Broad National Consumer Price Index (IPCA) - Adapted from [Brazilian Central Bank 2017a]**

### 3. Proposed Working Methods

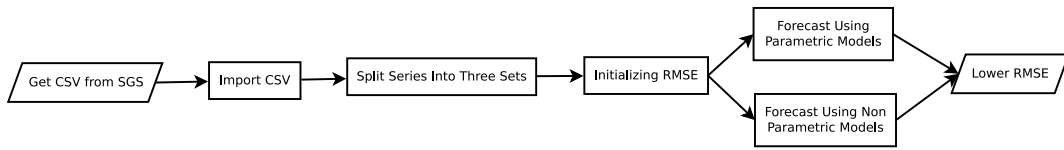
#### 3.1. Datasets

The data used in this work were obtained from [Brazilian Central Bank 2017b]. From that, a CSV file in English and another in Portuguese are downloaded both containing values from August 1994 to June 2017. Values prior to August 1994 were disregarded due to the high magnitude and instability of the index in that period. The CSV file has 277 rows: one header, one footer, and 275 rows with the index data, each corresponding to one month. The data lines have two columns, the month/year and the index value one.

#### 3.2. Overview

An overview of the forecasting process used in this paper is shown in Figure 3 and described below.

Initially, the CSV file in English was read by R and the CSV file in Portuguese was read by Scilab. The values were converted to double type. After conversion, the absolute



**Figure 3. Overview of the Forecasting Process Used in This Paper**

value of the smallest vector element and an extremely small positive number were added for all values to be positive, because some methods do not work with negative inflation. The header, footer, and month/year column were removed.

The series was divided into three subsets: set A, which contains the months from August 1994 to March 2002 and size of 92 months, set B, from April 2002 to October 2009 and size of 91 months, and the set C, with the indices from November 2009 to June 2017 and size of 92 months. Gaps at the end of each series were withdrawn, so that they could be used to compare forecasts with real values.

Finally, we generated the test series, which are nothing more than the data immediately after the series without gaps, until the maximum forecast horizon used in this paper, which was 6. Once the base was created, the program was started. The RMSE of all predictive methods were set to a very large maximum value, in this case  $10^{40}$ .

The KIII method required the creation of training, validation, and reference sets. These sets were created using index matrices, which in turn were generated through index vectors. We also created a set of tests, which is the set of validation transposed.

### 3.3. Models

Forecasts were started using Moving Averages. The parameter size of seasonal period from 3 to 12, in steps of 2. The input was the series without gaps. Specific test series were required for this method, due to the fact that there were no moving average values at the extremes of the series.

Subsequently, predictions were made using the SES method. This method received as input the time series without gaps, the forecast horizon and the parameter  $\alpha$ . The parameter  $\alpha$  ranged from  $10^{-4}$  (minimum allowed by R) to 1, in steps of 0.05.

In the time series prediction using the linear method of Holt, there are two parameters,  $\alpha$  and  $\beta$ . Both ranged from  $10^{-4}$  to 1, in steps of 0.05. For the method to work,  $\beta$  must be less than or equal to  $\alpha$ . The prediction was made, for each combination of  $\alpha$  and  $\beta$ , passing to the method the time series without the gap, the forecast horizon, and the parameters  $\alpha$  and  $\beta$ .

The predictive mode made by the additive and multiplicative methods of Holt-Winter is quite similar. Both models have three parameters:  $\alpha$ ,  $\beta$ , and  $\gamma$ . What differs from one model to the other is the value of the seasonal variable. In the additive method, this value is configured for additive and in the multiplicative method, for multiplicative. In addition to these parameters, the models also receive the time series with the withdrawn gap and the forecast horizon. The parameters  $\alpha$ ,  $\beta$ , and  $\gamma$  ranged from  $10^{-4}$  to 1, in steps of 0.05. There are many combinations of  $\alpha$ ,  $\beta$ , and  $\gamma$  that prevent prediction. The most trivial combinations were avoided by nested statement structures if. However, more

complex combinations are very difficult to avoid and need to be addressed using the try command. When the prediction gave error, the forecast model returned an object of the class try-error. When the class was different, the prediction was made.

At the end of each forecast, the value obtained is compared with the actual value. The error is calculated and, from it, the RMSE. A smaller RMSE replaces a larger RMSE stored in memory and, at the end, the smaller RMSE obtained is used.

At the beginning of the predictions using non-parametric methods, Root Mean Square Error (RMSE) measurements of each model and set were started. As the goal was to lower the values of this measure, they were initialized with the value  $10^{40}$ . This value was used as a very large value, and it was expected that all RMSE values that appeared throughout the paper were smaller than it. If values greater than this initial measure appeared, this value would be increased so that it was always a maximum value.

The first non-parametric method started was KIII. It is implemented in the object-oriented paradigm. Then, at the beginning of the execution, an object of the class KIII was created, having as parameter the size of the layer of KII-sets. Other configurations were then made and the network started.

The next step was the configuration of the counters to be used in the iterative loop. Within the iterative loop, the parameters varied within a predefined range, based on the work [Parmezan 2016].

The KIII objects were initially trained with the training sets and the results were obtained by filtering the reference sets through the respective KIII sets, generating the matrix kresult. To compare the prediction, we selected the last 15 indexes of the three initial sets, A, B, and C, forming the matrix kprediction. Each column of the test set was traversed by its respective KIII object and the results concatenated in a array kexp. The three matrices, plus the number of nearest neighbors, were passed to kNN, which returned a prediction result.

The process for the single kNN was simpler. The k value was varied from 0 to 10. As parameters to kNN, the reference set, the training set, the matrix kprediction, and the number of nearest neighbors (k) were passed.

Another non-parametric method used for the comparison was the classical artificial neural networks. To prepare the data for the forecast using this model, the raw, training, and test data sets were initially normalized to the interval [0 1]. A variable corresponding to the value to be compared was separated to find out the accuracy of the forecast. All datasets generated in this manner have been resized to the same size. The sizes of the input and output layers followed the sizes of the data sets that went in and out, and therefore only the size of the middle layer was varied. This size ranged from 3 to 11, which is the odd number immediately below 12, size of the seasonal period. Once the network size has been defined, the weights have been initialized and adjusted, the tests performed to define the network accuracy.

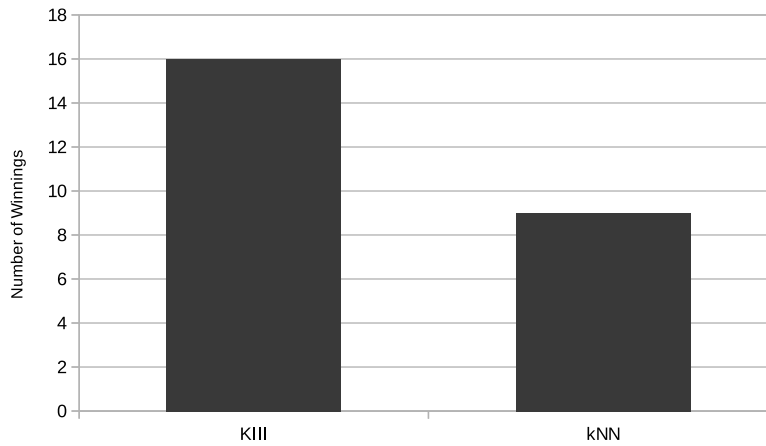
The last non-parametric model used was the SVM. The type of SVM used was the nu-SVR with core-based radial function [Chang and Lin 2016]. In the use of this model, two parameters were considered: the regularization parameter, which varied from a very small positive number, close to zero, to 1, with a step of 0.25, and the gaussian

width of the radial base function, which varied 0.005 to 0.25, in a step of 0.05. To make predictions, first the models were trained using a vector with random elements and normal distribution, the standard training data already used in the forecast with classical neural networks, the type of SVM, the type of core function, the normalization parameter and the Gaussian width of the radial basis function. Next, the predictions were calculated by moving to the Scilab function corresponding to the vector with the random elements, the vector with the data and the object with the trained models.

At end of each forecast, the RMSE was calculated. The calculated value was compared to the current value. If the value obtained was less than the value stored in memory, the value obtained became the new value of RMSE. Otherwise, the RMSE remained as it was.

#### 4. Experiments and Results

Figure 5 shows the overall performance of the methods used in the comparison. Figure 6 shows the performance of the methods in the experiments made using only the non-parametric methods and Figure 7 shows the performance considering only the parametric methods. The results represented in the figures are those obtained by the best combinations of parameters in each method. The non-parametric methods used were KIII Set with kNN (KIII), continuous kNN (kNN), Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The parametric methods used were AutoRegressive Integrated Moving Average (ARIMA), Seasonal AutoRegressive Integrated Moving Average (SARIMA), Moving Average (MA), Simple Exponential Smoothing (SES), Holt Linear Method (Holt), Additive Holt-Winters (AHW) and Multiplicative Holt-Winters (MHW).

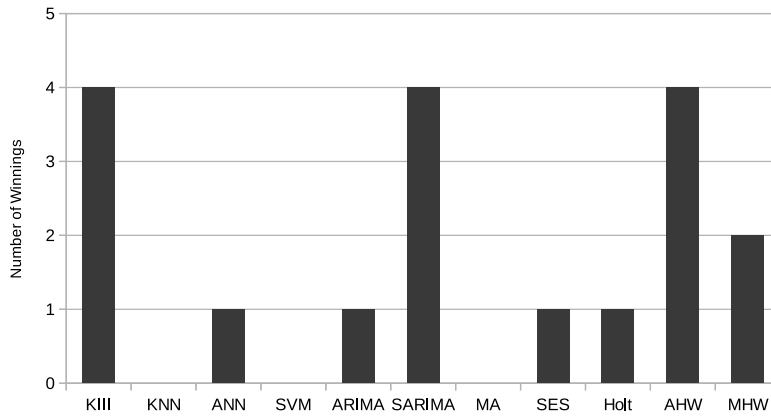


**Figure 4. Comparison between the performance of KIII and kNN**

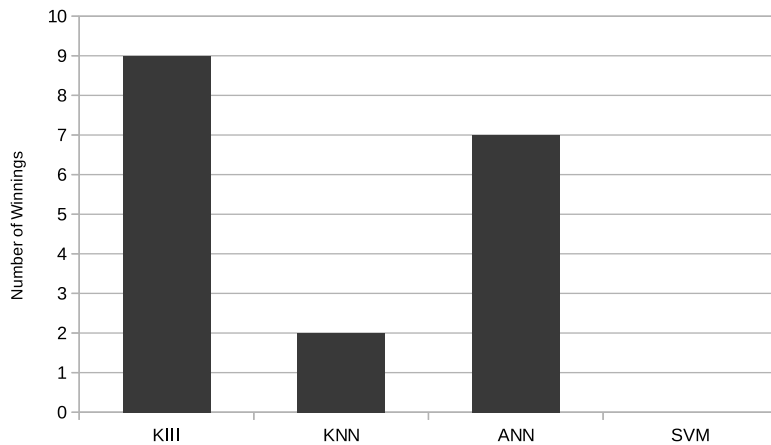
The comparison measure used was the Root Mean Squared Error (RMSE). It was chosen because the square root smoothes the values of the MSE measure, making it easier to be understood by human readers, without losing the ability to compare performance. The equation of the measure can be seen just below (Equation 1).

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^n (\hat{y}_t - y_t)^2}{n}} \quad (1)$$

where  $\hat{y}_t$  is the predicted value for time t,  $y_t$  is the actual value of time t and n is the number of predictions.



**Figure 5. Winning Methods**



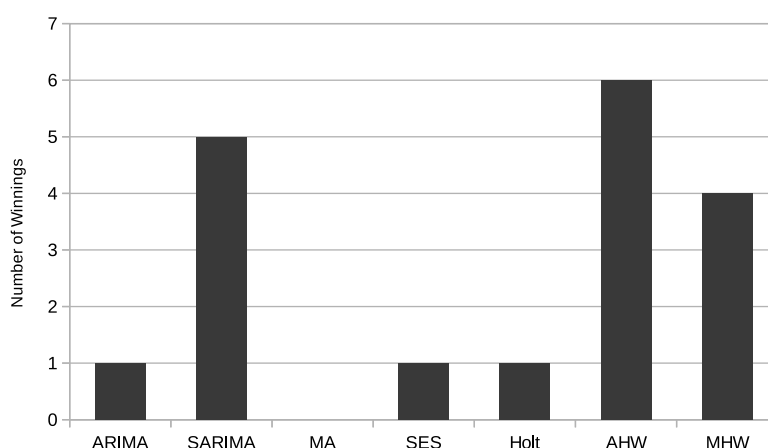
**Figure 6. Non-parametric Methods Winners**

Our initial objective and hypothesis was that the application of a KIII filter would improve the performance in the forecast of economic time series. From the results reported here, this hypothesis was partially confirmed, since the performance of KIII with kNN exceeded the performance of predictions made only with kNN, with two exceptions (Figure 4), demonstrating that it is a good filter. However, the KIII overcame the other methods only in four situations, tying with SARIMA and Additive Method of Holt–Winters (Figure 5).

Among the non-parametric methods, the KIII performed well, winning in nine situations, surpassing all other methods (Figure 6). The single continuous kNN, in most cases did not perform well, surpassing the other non-parametric methods in only two situations (Figure 6). RNAs and KIII had the best performance in the oldest indices, showing that inflation was more unstable during this period. In the more recent indexes, other methods had a greater chance, due to the less instability of the series, however the neural networks continued to dominate.

In the parametric methods, the Holt–Winters method was the winner because the time series has a linear tendency and seasonality. It is unclear whether this seasonality is additive or multiplicative, since in some cases the additive Holt–Winters won, and in others it was the multiplying Holt–Winters who won. The (S)ARIMA methods also





**Figure 7. Parametric Methods Winners**

performed well, especially the seasonal version, a fact that occurred due to the presence of seasonality and the presence of autocorrelation among the elements of the series.

In the set A, among the non-parametric methods, the KIII filter had better performance in three of the prediction horizons, the classic neural networks in two and the kNN in one. Both the classical neural networks and the KIII filter are neural networks and are more robust to noise and volatility. Since this data set contains the period with the first months immediately after the Plano Real, IPCA volatility was still very high at that time. Among the parametric methods, the additive method of Holt–Winters gained in 2 prediction horizons, whereas the ARIMA, Holt linear, SARIMA and Multiplicative Holt–Winters methods gained in one horizon each.

In set B, among the non-parametric methods, classical neural networks performed better in four prediction horizons and KIII in two horizons. In this set, with respect to parametric methods, the multiplicative methods of Holt–Winters, additive of Holt–Winters, and SARIMA have gained in two horizons each. This makes it clear that the series has seasonality. However, it is not clear whether seasonality is additive or multiplicative.

In the set C, among the non-parametric methods, the biggest winner was the KIII, which won all the first four forecast horizons. The classical neural networks overcame horizon 5, more difficult. The kNN overcame horizon 6, very difficult and randomly predicted. In the parametric methods, there was the victory of the method of Holt–Winters, being 2 horizons for the additive and one for the multiplicative. SARIMA won on two horizons and SES on horizon six. The SES method won in horizon 5. The SARIMA method won in horizon 6, the most difficult to predict among all the horizons of this research.

## 5. Conclusion

The main objective of this paper was to demonstrate that a biologically plausible connectionist approach, Freeman K-sets, more specifically the KIII level, provides more realistic predictions for the economic time series of a developing country, more specifically a Brazilian time series of economic activity, a consumer price index, the Broad Consumer Price Index, collected and released monthly by the Brazilian Institute of Geography and

Statistics.

Throughout the development of this paper, predictions were made using KIII, which were compared with other state-of-the-art techniques, based on [Parmezan 2016] and [Gooijer and Hyndman 2006]. At the end of the research, the hypothesis was partially verified, because the KIII sets worked well as a filter to improve the performance of the continuous kNN. However, they outperformed other techniques tested in only a few results, indicating that this is not a good regression model.

As a future work, it is proposed to use the KIII sets as a filter not only for kNN but also for other techniques in order to confirm if the low performance of KIII was due to the continuous kNN regression algorithm or to the KIII filter itself. Another possibility of continuation of the work would be the use of other similarity prediction methods, such as kNN-TSPI [Parmezan and Batista 2015] and kNN-Time Series Prediction (kNN-TSP) [Ferrero 2009]. We intend to remake the experiments, considering more advanced semi-parametric and non-parametric methods, that are used in the economy for the forecasting of time series, and to compare them with the forecast made using KIII.

It is also intended to use techniques for parameter estimation, such as holdout validation, cross validation and Box-Jenkins method, to obtain more optimized parameters and better performance. An interesting future work would be a more comprehensive empirical assessment, based on the data sets available in the ICMC-USP Time Series Prediction Repository [Parmezan and Batista 2014]. An interesting research question would be the in-depth study of K-sets, especially the KIII level, with the aim of creating a new, better-performing variant based on Freeman K-sets.

## Acknowledgment

V. H. Gonçalves is grateful to the Brazilian Institute of Geography and Statistics and Sociedade Ibgeana de Assistência e Seguridade for his income during the research.

J. L. G. Rosa is grateful to the Brazilian agency São Paulo Research Foundation (FAPESP – process 2016/02555-8) for the financial support.

## References

- (2018a). 38160 reais =. Dollar values obtained by conversion of Brazilian real to US dollar made by Google calculator on 07/03/2018.
- (2018b). 954 reais =. Dollar values obtained by conversion of Brazilian real to US dollar made by Google calculator on 07/03/2017.
- Brazilian Central Bank (2017a). Broad national consumer price index (ipca). Time Series Management System - v2.1.
- Brazilian Central Bank (2017b). Time series management system - v2.1.
- Chang, C.-C. and Lin, C.-J. (2016). Libsvm – a library for support vector machines.
- Ferrero, C. A. (2009). Algoritmo knn para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia. Master's thesis, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

- Freeman, W. J. (1975). *Mass Action in the Nervous System*. Academic Press New York, San Francisco, London.
- Gooijer, J. G. D. and Hyndman, R. J. (2006). 25 years of time series forecasting. *International Journal of Forecasting*, 22(3):443–473.
- Harvey, A. (1985). Trends and cycles in macroeconomic time series. *Journal of Business & Economic Statistics*, 3(3):216–227.
- Instituto Brasileiro de Geografia e Estatística (2017). Sistema nacional de Índices de preços ao consumidor ipca e inpc.
- Kozma, R. and Beliaev, I. (2004). Time series prediction using chaotic neural networks: case study of ijcnncats benchmark test. In *Proc. 2004 IEEE international joint conference on neural networks*, volume 2, pages 1609–1613.
- Kozma, R. and Freeman, W. J. (2001). Chaotic resonance - methods and applications for robust classification of noisy and variable patterns. *International Journal of Bifurcation and Chaos*, 11(6):1607–1629.
- Kozma, R., Piazzentin, D. R. M., and Rosa, J. L. G. (2013). Cognitive clustering algorithm for efficient cybersecurity applications. In *2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.
- Li, H. and Kozma, R. (2003). A dynamic neural network method for time series prediction using the kiii model. In *Proc. 2003 IEEE International Joint Conference on Neural Networks*, pages 347–352.
- Parmezan, A. R. S. (2016). Similarity-based time series prediction. Master's thesis, Universidade de São Paulo, São Carlos.
- Parmezan, A. R. S. and Batista, G. E. (2014). Icmc-usp time series prediction repository.
- Parmezan, A. R. S. and Batista, G. E. (2015). A study of the use of complexity measures in the similarity search process adopted by knn algorithm for time series prediction. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 45–51. IEEE.
- Piazzentin, D. R. M. and Rosa, J. L. G. (2014). Motor imagery classification for brain-computer interfaces through a chaotic neural network. In *2014 International Joint Conference on Neural Networks (IJCNN)*, number 3, pages 4103–4103.
- Piazzentin, D. R. M. and Rosa, J. L. G. (2015). A simulator for Freeman k-sets in java. In *2015 International Joint Conference on Neural Networks (IJCNN)*, number 3, pages 1–8.
- Rosa, J. L. G. and Piazzentin, D. R. M. (2016). A new cognitive filtering approach based on Freeman k3 neural networks. *Applied Intelligence*, 45(2):363–382.
- Stock, J. H. and Watson, M. W. (2005). Implications of dynamic factor models for var analysis. working paper.

H = 1											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	0,0160000	0,0970000	0,0021658	0,5978732	0,1220474	0,0038439	0,1298373	0,3555048	0,0261817	<b>0,0003479</b>	0,0051310
2	0,0300000	0,0890000	0,0004215	0,4812814	0,0004574	<b>0,0000066</b>	0,0700000	0,0031104	0,0021529	0,0004384	0,0002528
3	0,0400000	0,3400571	0,1180266	0,6484109	0,0006590	0,0073849	0,0407499	0,0086385	0,0011023	<b>0,0000155</b>	0,0019197
H = 2											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	0,0013333	0,0020000	0,0044409	0,7635358	0,0483692	0,0504705	0,5699903	0,0504574	<b>0,0006193</b>	0,1892072	0,0016745
2	0,0024000	0,0093333	<b>0,0001022</b>	0,4472257	0,0845224	0,1126701	0,0600000	0,0726099	0,0224155	0,0092358	0,0056748
3	0,0160000	0,2100000	0,2048598	0,7000238	0,0004846	<b>0,0000538</b>	0,0049399	0,0182939	0,0116061	0,0091006	0,0005136
H = 3											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	<b>2,220E-16</b>	0,0053333	0,0111459	0,8770559	0,0031479	0,0351964	0,0135533	0,0105508	0,0007147	0,0002643	0,0012533
2	0,0080000	0,0240000	0,0016697	0,6868843	0,2047733	0,2156811	0,0500000	0,1926099	0,0552653	<b>0,0004678</b>	0,0059208
3	<b>2,220E-16</b>	0,1000000	0,2096667	0,8790697	0,1032957	0,0039183	0,0167583	0,0178234	0,0071104	0,0006878	0,0734860
H = 4											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	0,0012000	0,0008000	0,0033901	0,2039829	<b>0,0005065</b>	0,0368862	0,1399903	0,0144951	0,0448095	0,0037516	0,3146787
2	0,1060000	0,2050000	0,0023299	0,0053654	0,2933041	0,1300000	0,0366433	0,2826099	0,0266770	<b>0,0010083</b>	0,0038394
3	0,0140000	0,2210000	0,1882608	0,0699648	0,2182344	0,0013951	0,0058812	0,0586385	0,0441788	0,1130154	<b>0,0000153</b>
H = 5											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	0,0005666	0,0300000	0,0035916	0,7653237	0,1010652	<b>0,0000000</b>	0,1183606	0,1155048	0,0011275	0,0948402	0,0462471
2	<b>0,0000000</b>	0,0240000	0,0001754	0,5723659	0,1995173	0,0004853	0,1564554	0,1926099	0,0713806	0,1400928	0,0900865
3	0,1160000	0,3210000	0,0922634	0,5970874	0,3513492	<b>0,0026639</b>	0,0258530	0,1686385	0,1518710	0,1145268	0,0197270
H = 6											
	KIII	KNN	ANN	SVM	ARIMA	SARIMA	MA	SES	Holt	AHW	MHW
1	0,0943333	0,2080000	0,0069655	0,7123337	0,2571433	0,0374895	0,0111151	0,2755048	0,0163146	0,1691680	<b>0,0023422</b>
2	<b>0,0000000</b>	0,0385000	0,0004977	0,6675241	0,1563772	0,1180324	0,1700000	0,1526099	0,1194382	0,0016472	0,0001998
3	0,0065000	0,0031429	0,1781777	0,6924037	0,1966414	0,0374863	0,0030527	<b>0,0013614</b>	0,0204368	0,3316610	0,2071911

Table 1. Results of each method for each forecast horizon and data set considering the best combination of parameters.