The Neural Network Lee–Carter Model with Parameter Uncertainty: The Case of Italy

Mario Marino and Susanna Levantesi

Abstract One of the main challenges for life actuaries is modeling and predicting the future mortality evolution. To this end, several stochastic mortality models have been proposed in literature, starting from the pivotal approach of the Lee–Carter model. These models essentially use the ARIMA processes to forecast the future mortality trends. Recently, some research works have shown the adequacy of the deep learning techniques to improve mortality modeling, obtaining competitive and outperforming forecasts compared to the ARIMA. The present work focuses on the application of a recurrent neural network, the Long Short-Term Memory (LSTM), in the Lee–Carter model framework. The LSTM has an architecture specifically designed to model and predict sequential data, such as time series, well capturing hidden patterns within data related to events that may be far from each other. In mortality modeling, this means that the forecasted mortality rates take into account the hidden features of the past phenomenon not always adequately captured by the ARIMA. We extend the approach proposed in Nigriet al. (Risks 7(1), 33 (2019)), performing a point forecasting of the Lee–Carter time-index through LSTM and deriving the related prediction interval representing the LSTM's parameter uncertainty.

Keywords Mortality forecasting · Lee-Carter model · Deep neural networks · Parameter uncertainty

1 Introduction

Mortality forecasting is a relevant topic for actuaries and demographers. It has gained prominence in recent decades due to the rapidity of the life expectancy increases. The need of accurate forecasting to address longevity risk and adequately pricing the annuities products has led actuaries towards more sophisticated extrapolative meth-

M. Marino · S. Levantesi (✉)
Department of Statistics, Sapienza University of Rome, Rome, Italy
e-mail: susanna.levantesi@uniroma1.it

M. Marino
e-mail: m.marino@uniroma1.it

ods in a stochastic environment. Most of these models essentially refer to ARIMA processes, that can be considered the most popular linear models in describing the future evolution of mortality. Though these processes are flexible tools in representing a time series, their linear structure can sometimes constitute a limit, providing estimates not consistent with the real mortality shape. Recently, some research works have shown the power of deep learning techniques in improving the accuracy of mortality forecasting. For example, Hainaut [1] used a neural network to identify the latent factors of mortality, then forecasted by a random walk with drift. Richman and Wüthrich [2] apply neural networks to extend the Lee–Carter model to multiple populations, achieving competitive out-of-sample forecasting performance. Nigri et al. [3] use a recurrent neural network with Long Short-Term Memory (LSTM) architecture to predict the time-index of the Lee–Carter model, achieving more accurate predictions compared to the ARIMA models in several countries and for both genders. The LSTM is designed to model and predict sequential data, such as time series, capturing hidden patterns (both linear and nonlinear) within data. In the field of mortality forecasting, this means predicting mortality rates over time taking into account features of the past, more or less recent. The present work extends the paper of Nigri et al. [3], providing an estimation of parameter uncertainty of the LSTM, beside the point estimate, in the Lee–Carter model. Develop a prediction interval for neural networks is still a big challenge. At the current state of art, the main contributions on this topic are from Koshravi [4], Keren et al. [5] and Petneházi [6], but not in the field of neural networks applied to mortality time series. Our paper provides a new contribution in this research area. The predictive power of LSTM point estimates is measured by Root Mean Square Error (RMSE) and Mean Average Error (MAE) in an out-of-sample test. While the prediction interval resulting from the LSTM parameter uncertainty is compared to that from ARIMA by a graphical analysis, including the observed mortality data. The case study refers to Italy.

2 Neural Networks and Parameters Uncertainty

Consider a set of data $\{\mathbf{y}, \mathbf{x}\}$, where y is the target variable and $\mathbf{x}$ the n-dimension vector of independent variables. We aim to identify the relationship between $\mathbf{y}$ and $\mathbf{x}$ through a neural network (NN). In a NN, the neuron represents the basic unit receiving the input that transforms into output. This transformation takes place through the use of a set of weights, $\mathbf{w} = \{w_1, w_2, ...\}$, and activation functions, $\phi = \{\phi_1, \phi_2, ...\}$. Therefore, denoting f_{NN} the NN function, we assume that: $y = f_{NN}(\mathbf{x}; \mathbf{w}; \phi) + \epsilon$, where ϵ is the error term assumed to be i.i.d. with an expected value of zero, $E(\epsilon_i) = 0 \ \forall i$ and $E(\epsilon_i \epsilon_j) = 0 \ \forall i \neq j$. The optimal weights, $\hat{\mathbf{w}}$, are obtained by minimizing a given loss function, $L(\mathbf{y}, \hat{\mathbf{y}})$, with $\hat{\mathbf{y}} = f_{NN}(\mathbf{x}; \hat{\mathbf{w}}; \phi)$, during the training phase of the NN. The resulting estimate $\hat{\mathbf{y}}$ represents a point estimation of the mean in the regression problem and depends on the realization $\mathbf{x}$ to which the estimate of weights, $\hat{\mathbf{w}}$, is linked. Therefore, it does not provide any information on the uncertainty given by $\hat{\mathbf{w}}$. However, by considering the input as a random variable, whose possible realizations determine the weights in the NN, we can measure the uncertainty of the NN estimator around the point estimation. The uncertainty of the NN estimator and the related prediction intervals, is obtained by bootstrap, a well-known technique following an ensemble scheme, so that a set of NNs is able to produce a point estimate that is generally less biased respect of a single NN ([4] and [6]). The scheme is the following:

a. We train the NN to identify the best performing hyperparameter configuration for the input data. This represents the best NN architecture.
b. From the training set, we generate B bootstrap samples, $\left\{\mathbf{y}^{(b)}, \mathbf{x}^{(b)}\right\}$,where $b = 1, \ldots, B$.
c. For each bootstrap sample b, using the same NN architecture as at step a, we obtain B point estimates characterized by new weights but same hyperparameters: $\hat{\mathbf{y}}^{(b)} = f_{NN}\left(\mathbf{x}^{(b)}; \hat{\mathbf{w}}^{(b)}; \phi\right),\ b = 1, \ldots, B$. We build the prediction intervals on these estimates.

3 The Neural Network Lee–Carter Model

In the Lee–Carter (LC) model, as proposed by Brouhns et al. [7], the logarithm of the central death rate, $m_{x,t}$ is the central death rate at age x in year t, is described by:

$$log(m_{x,t}) = \alpha_x + \beta_x \kappa_t \tag{1}$$

where α_x is the average age-specific pattern of mortality, β_x is the pattern of deviations from the age profile as κ_t varies, andκ_t is the time index describing mortality trend.

The model assumes a Poisson distribution for the numbers of deaths and is subject to the constraints, $\sum_x \beta_x = 1$ and $\sum_t \kappa_t = 0$.

In the traditional LC formulation κ_t is usually modeled by an ARIMA process. In this paper, we use a one-order autoregressive approach and describe κ_t by a LSTM neural network (see Nigri et al. [3] for a detailed description of LSTM applied to κ_t). Denoting by f_{LSTM} the LSTM, the Lee–Carter time-index is modeled by:

$$\kappa_t = f_{LSTM}(\kappa_{t-1}; \mathbf{w}; \phi) + \epsilon_t \tag{2}$$

The NN training, as described in Sect. 2, produce the following estimate:

$$\hat{\kappa}_t = f_{LSTM}(\kappa_{t-1}; \hat{\mathbf{w}}; \phi) \quad for \ \ t = t_1, \ldots, t_\tau \tag{3}$$

where the optimal weights $\hat{\mathbf{w}}$ are obtained by minimizing the Mean Squared Error (MSE) loss function, $MSE = \frac{1}{t_\tau - t_1} \sum_{t=t_1}^{t_\tau} \left(\kappa_t - \hat{\kappa}_t\right)^2$. The forecasted values of $\hat{k}_t$ are then estimated by:

$$\hat{\kappa}_t = f_{LSTM}(\hat{\kappa}_{t-1}; \hat{\mathbf{w}}; \phi) \quad for \ \ t = t_{\tau+1}, \ldots, t_s$$

Therefore, the time series $\left\{\hat{\kappa}_{t_{\tau+h}}\right\}_{h=1}^{s-\tau}$ is the point prediction provided by the LSTM. The accuracy of prediction is measured according to the $RMSE = \sqrt{\frac{\sum_{T=t_\tau+1}^{t_s-t_\tau}(\kappa_T-\hat{\kappa}_T)^2}{t_s-t_\tau}}$ and $MAE = \frac{\sum_{T=t_\tau+1}^{t_s-t_\tau}|\kappa_T-\hat{\kappa}_T|}{t_s-t_\tau}$.

4 Estimation of Prediction Intervals with Parameter Uncertainty

The parameter uncertainty is incorporated in the forecasting using the Poisson bootstrap proposed by Brouhns et al. [8]. Therefore, following the step b in Sect. 2, we generate B bootstrap sample of the number of deaths from a Poisson distribution with parameter equal to the observed number of deaths: $D_{x,t}^{(b)} \sim Poisson(D_{x,t})$, with $b = 1, \ldots, B$, where, according to the Lee–Carter model, $D_{x,t} = E_{x,t} \cdot \left(\hat{\alpha_x} + \hat{\beta}_x\hat{\kappa}_t\right)$, with $E_{x,t}$ are the exposure to risk. For each bootstrap sample, we obtain the parameters' estimate $\hat{\alpha}_x^{(b)}$, $\hat{\beta}_x^{(b)}$ and $\hat{\kappa}_t^{(b)}$, where $\hat{\kappa}_{t-1}^{(b)}$ represent the LSTM input and $\hat{\kappa}_t^{(b)}$ the associate target. Hence, the latter is linked to the former by the LSTM function, f_{LSTM}, for each bootstrap sample.

LSTM prediction intervals

Each b bootstrap sample of $\hat{\kappa}_t^{(b)}$ is modeled by a LSTM using the best architecture identified during the training and the same loss function, and re-optimizing the weights:

$$\hat{\kappa}_t^{(b)} = f_{LSTM}(\kappa_{t-1}^{(b)}; \hat{\mathbf{w}}^{(b)}; \phi) \quad for \quad t = t_1, \ldots, t_\tau \tag{4}$$

While, the forecasted values are:

$$\hat{\kappa}_t^{(b)} = f_{LSTM}(\hat{\kappa}_{t-1}^{(b)}; \hat{\mathbf{w}}^{(b)}; \phi) \quad for \quad t = t_{\tau+1}, \ldots, t_s \tag{5}$$

Therefore, the time series $\left\{\hat{\kappa}_{t_{\tau+h}}^{(b)}\right\}_{h=1}^{s-\tau}$ is the b point prediction provided by the LSTM, for $b = 1, \ldots, B$. The prediction interval is then defined by setting an $\alpha\%$ percentile on the bootstrap distribution:

$$\left[f_{LSTM}^{Low}, f_{LSTM}^{Up}\right] : Prob\left(f_{LSTM}^{Low} \leq k_t \leq f_{LSTM}^{Up}\right) = 1 - \alpha$$

where $f_{LSTM}^{Low} = f_{LSTM}\left(\hat{\kappa}_{t_{\tau+h}}^{(b)}; \hat{\mathbf{w}}^{(b)}; \phi\right)^{(\alpha)}$ and $f_{LSTM}^{Up} = f_{LSTM}\left(\hat{\kappa}_{t_{\tau+h}}^{(b)}; \hat{\mathbf{w}}^{(b)}; \phi\right)^{(1-\alpha)}$ are the lower and upper bound of the interval, respectively.

ARIMA prediction intervals

Assuming that the time series $\hat{\kappa}_t^{(b)}$ is modeled by a generic ARIMA(p, d, q), where p is the autoregressive order, d the degree of differencing, and q the order of the moving-average model:

$$\nabla^{d^\star} \kappa_t^{(b)} = \delta^{(b)} + \sum_{i=1}^{p^\star} \phi_i^{(b)} \nabla^{d^\star} \kappa_{t-1}^{(b)} + \epsilon_t + \sum_{j=1}^{q^\star} \theta_j^{(b)} \epsilon_{t-j}$$

The parameters $\delta^{(b)}$, $\phi^{(b)}$ and $\theta^{(b)}$ are estimated for each b bootstrap sample, without changing the ARIMA model (p, d, q). In this case, $\left(\hat{\kappa}_{t_{\tau+h}}^{(b)}\right)^{(\alpha)}$ and $\left(\hat{\kappa}_{t_{\tau+h}}^{(b)}\right)^{(1-\alpha)}$ are the lower and upper bound of the interval, respectively.

5 Numerical Application

The model is implemented for Italy considering the period 1950–2014 and ages 0–100. Data are taken by the Human Mortality Database. The dataset is split into training and testing set, following the rule 80% train (in-sample: 1950–2001) and 20% test (out-of-sample: 2002–2014). The model accuracy is assessed through an out-of-sample test. The parameters of the best ARIMA models are $(p, d, q) = (0, 2, 3)$ for males (AIC = 240.43 and BIC = 248.08) and $(p, d, q) = (0, 1, 1)$ for females (AIC = 277.73 and BIC = 283.53). The LSTM hyperparameters stemming from a fine tuning procedure. A grid search algorithm is implemented investigating a subset of the hyperparameter space, discretized and bounded coherently with the sample size. Table 1 shows the best tuned values of LSTM hyperparameters.

Table 2 shows the performance of LSTM and ARIMA in terms of RMSE and MAE for Italy, by gender.

Figure 1 shows the prediction intervals for κ_t obtained by LSTM and ARIMA, using $B = 1000$ bootstrap samples.

In conclusion, we observe a higher capacity of the LSTM to capture nonlinear trends respect to the ARIMA models. This is particularly evident in the case of Italian

Table 1 LSTM hyperparameters

Hyperparameter	Males	Females
Activation function	ReLu	ReLu
Recurrent function	TanH	TanH
N. Hidden Layer	1	1
N. Units	97	162
Epochs	2	1

Table 2 RMSE and MAE Italy

Measure	Males		Females	
	LSTM	ARIMA	LSTM	ARIMA
RMSE	4.264	6.526	4.189	4.020
MAE	3.707	6.022	3.315	3.412

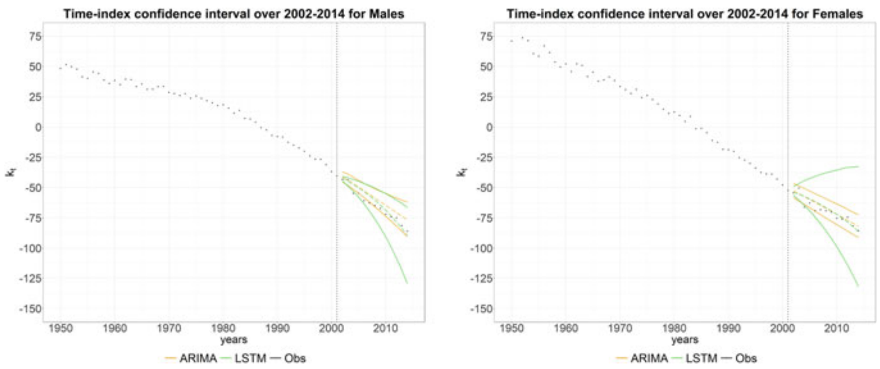

Fig. 1 κ_t with 5 and 95% prediction intervals, Italian males (left panel) and females (right panel): LSTM vs ARIMA

males. Our results highlight the high accuracy provided by LSTM in terms of point forecasting and, at the same time, the greater parameter uncertainty respect to the case of ARIMA, especially for Italian females. This evidence, stemming from the bias-variance tradeoff principle, lays the foundation to continue investigating NNs parameter uncertainty in the field of mortality forecasting.

References

1. Hainaut, D.: A neural-network analyzer for mortality forecast. Astin Bull. **48**, 481–508 (2018)
2. Richman, R., Wüthrich, M.: A neural network extension of the Lee–Carter model to multiple populations. Ann. Actuar. Sci. 1–21 (2019)
3. Nigri, A., Levantesi, S., Marino, M., Scognamiglio, S., Perla, F.: A deep learning integrated Lee–Carter model. Risks **7**(1), 33 (2019). ISSN: 2227-9091
4. Khosravi, A., Creighton, D.: Comprehensive review of neural network-based prediction intervals and new advances. IEEE Trans. Neural Netw. **22**(9), (2011)
5. Keren, G., Cummins, N., Schuller, B.: Calibrated prediction intervals for neural network regressors. IEEE Access **6**, 54033–54041 (2018)
6. Petneházi, G.: Recurrent neural network for time series forecasting (2019). arXiv:1901.00069v1
7. Brouhns, N., Denuit, M., Vermunt, J.K.: A Poisson log-bilinear regression approach to the construction of projected lifetables. Insur.: Math. Econ. **31**(3), 373–393 (2002)
8. Brouhns, N., Denuit, M., Van Keilegom, I.: Bootstrapping the poisson log-bilinear model for mortality forecasting. Scand. Actuar. J. **3**, 212–224 (2005)