A deep learning integrated Lee-Carter model

May 8, 2019

Background: Mortality

- Mortality Forecasting
- mortality modelling: statistical tools

But...

- statistical learning techniques are back on stage under the name of **Machine Learning**
- we try to fill the gap between demography and machine learning,

Purpose

- In order to better describe the nonlinear shape of mortality surface,
- we use deep learning techniques to forecast the time index of mortality in the LC model

Purpose

- **Lee-Carter:** classic version of Lee-Carter model, in which we get the estimation of parameters.
- **Neural Network Model:** we obtain $k-NN$, forecast of k parameter using Neural Network
- **Results:** forecasted mortality rate using non-linear $k-NN$ into Lee-Carter Model

Data: HMD.

Model:Lee-Carter

Lee and Carter (1992), using SVD.

$$ln(m_{ij}) = \alpha_i + \beta_i \kappa_j + \varepsilon_{ij} \tag{1}$$

α_i is the general shape of the log-mortality at age i
κ_j represents the time trend index of general mortality level
β_i indicates the sensitivity of the log-mortality at age i to variations in the time index
ε_{ij} is the residual term at age x and time t.
In the traditional LC formulation, κ_t is usually modeled by an ARIMA(0,1,0):

$$\kappa_t = \kappa_{t-1} + \delta + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma_\kappa^2), \tag{2}$$

where δ is the drift parameter and ε_t are the error terms, normally distributed with null mean and variance σ_κ^2.

Model:Lee-Carter

ARIMA(p,d,q) Model

- AR=AutoRegression and p is the order,
- I=Integration, d is the order,
- MA=Moving Average, q is the order.
- So the ARIMA (p,d,q) is a ARMA(p,q) model on the differences of order "d" instead of on the real value.

We calibrate the best ARIMA(p,d,q) according to the Hyndman-Khandakar algorithm.

- In the first round, checks the stationarity of the time series and chooses the differencing order d.
- In a second stage, it selects the best values of auto-regressive and moving average order, respectively p and q
- The algorithm is implemented by the function auto.arima available in the R package forecast.

Model:Lee-Carter

- Although the ARIMA process is widely used in modeling the time indexes of mortality, it has a fixed structure and works well when data satisfies the ARIMA assumptions,
- e.g., the constant variance assumption that is one of the most important features of the integrated models.
- In many cases, demographical data may exhibit volatility changes and this feature does not fit the ARIMA assumption, especially for long time series.

The final aim

So far...

- **Lee Carter:** parameters and forecasting.

- **Aim:** apply the neural network on k parameter using the following model:

$$k_t = f(k_{t-1}) + \varepsilon_t \tag{3}$$

Where $k_{t-1} = (k_{t-1}, k_{t-2}, ..., k_{t-n})$ is a vector containing the values of the series and f is a neural network with n hidden nodes in a m layer.

Model:Neural Network

How does Neural Network work?

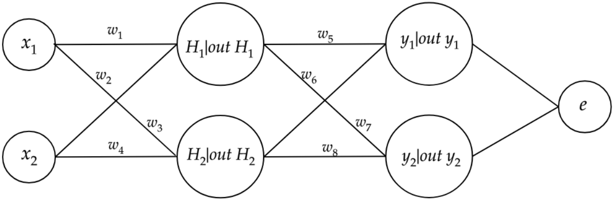
Job interview:

- Interviewer: What' your biggest strength?
- Me: I'm an expert in machine learning.
- Interviewer: What's 9+10?
- Me: Its 3.
- Interviewer: Not even close. It's 19.
- Me: It's 16.
- Interviewer: Wrong. Its still 19.
- Me: it's 19.
- Interviewer: You're hired!

Model:Neural Network

1. FORWARD PROPAGATION: we apply a set of weights to the input data and calculate an output.

1. $H_n = \sum_{j=0}^{n} x_j w_j = w^T x$ 2. $out\ H_n = f(H) = \frac{1}{1+e^{-H_n}}$ 3. $y_m = \sum_{j=0}^{m} out\ H_j w_j$ 4. $out\ y_m = f(y) = \frac{1}{1+e^{-y_m}}$



2. BACK PROPAGATION:
we measure the error of the output and adjust the weights accordingly to decrease the error (gradient descent).

5. $e = (y - \hat{y});\ J = \sum \frac{1}{2}(e)^2$ 6. $\Delta w = -r \nabla J(w)$

3. DERIVATION STEPS:

$for\ w_k;\ k = 5, \dots, 8$

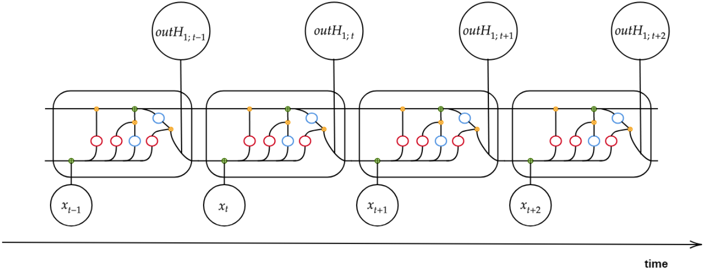
$$\frac{\partial total\ E}{\partial w_k} = \frac{\partial total\ E}{\partial out\ y_k} \frac{\partial out\ y_k}{\partial y_k} \frac{\partial y_k}{\partial w_k}$$

$for\ w_z;\ k = 1, \dots, 4$

$$\frac{\partial total\ E}{\partial w_z} = \frac{\partial total\ E}{\partial out\ H_n} \frac{\partial out\ H_n}{\partial H_n} \frac{\partial H_n}{\partial w_z}$$

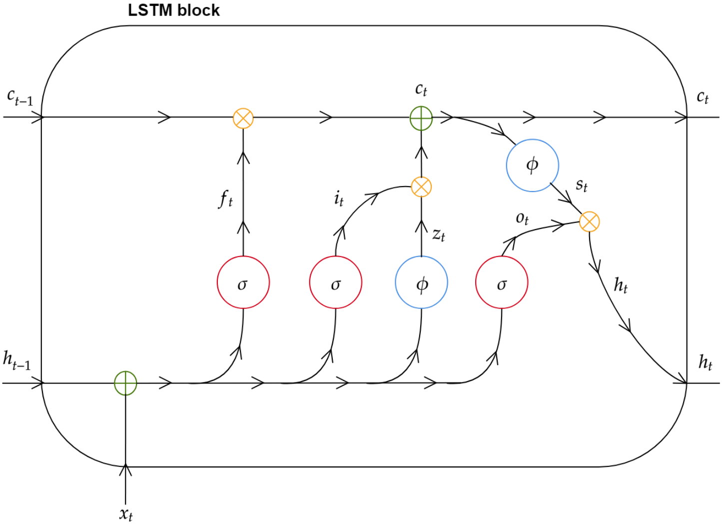
$$\frac{\partial total\ E}{\partial out\ H_n} = \frac{\partial E_1}{\partial out\ H_n} + \frac{\partial E_2}{\partial out\ H_{n+1}}$$

Model:Neural Network



- The main component is it cell state, the horizontal line which pass over the diagram. It works like a transporter belt on which information flows.
- forgetgate layer that decides what information to reject from the cell.
- To store, two layers are used: a sigmoid layer (the input gate layer), which decides which values we will update and the tanh layer that creates a vector of new candidate values.
- We then update $C(t-1)$ to obtain the new candidate values.
- Through the layer sigmoid, we decide which part of the state of the cell we will return as output.

Model:Neural Network



Model:Neural Network (LSTM)

...an appropriate NN approach

Long Short Term Memory: suitable for sequential data scructure.

- **preserve the information** over the time, thus preventing older signals from vanishing during processing.
- **it is capable to capture the noise** of the past mortality trend and repeat it in forecasting trend.

Numerical Application

Train and Test

- Training set is used for supervised learning,
- Testing is used to validate the model.

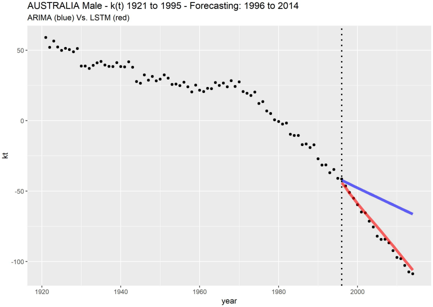
Output		Input	
κ_t	κ_{t-1}	κ_{t-2}	κ_{t-3}
κ_{t+1}	κ_t	κ_{t-1}	κ_{t-2}
...	...	...	...
κ_n	κ_{n-1}	κ_{n-2}	κ_{n-3}

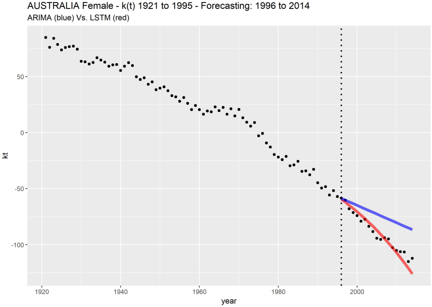
Afterwards network has learned the input-output functional relationship and it should be able to predict future values of k_t using only the input.

Results: Out of Sample

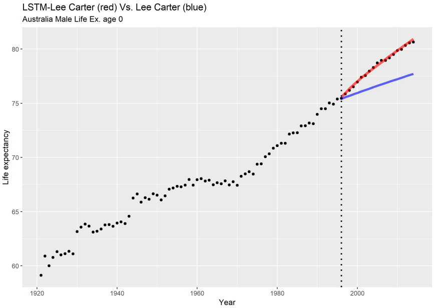
Country	Male		Female	
Australia	*MAE*	*RMSE*	*MAE*	*RMSE*
κ_t Arima	23.36	26.70	12.85	14.43
κ_t LSTM	2.73	3.27	4.23	5.34
Italy	*MAE*	*RMSE*	*MAE*	*RMSE*
κ_t Arima	40.93	48.88	30.15	34.51
κ_t LSTM	7.11	11.06	13.30	15.00
Spain	*MAE*	*RMSE*	*MAE*	*RMSE*
Arima	21.09	25.67	26.59	32.36
κ_t LSTM	4.28	6.44	6.05	7.93
U.S.A.	*MAE*	*RMSE*	*MAE*	*RMSE*
κ_t Arima	8.55	9.66	4.33	4.96
κ_t LSTM	2.13	3.03	3.73	4.42
Japan	*MAE*	*RMSE*	*MAE*	*RMSE*
κ_t Arima	5.11	6.04	10.03	12.71
κ_t LSTM	3.85	4.49	18.75	24.78

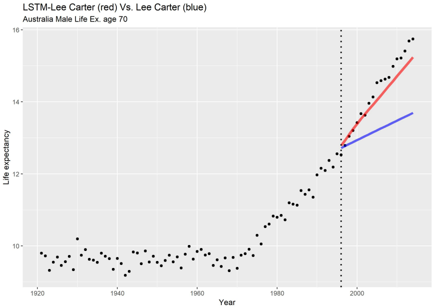
Results: Australia k_t Out of Sample



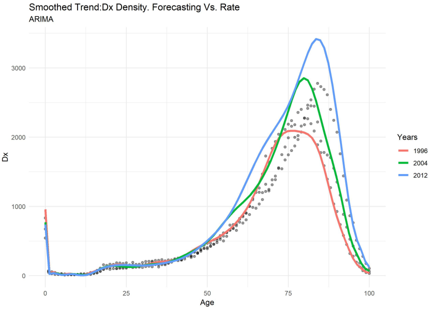


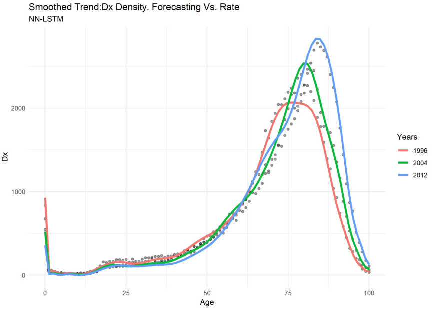
Results: Australia e_0, e_{70} Out of Sample





Results: Australia D_x. ARIMA Vs. LSTM



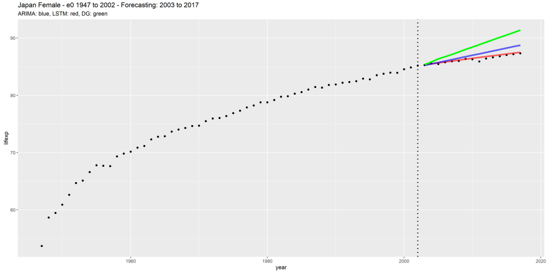


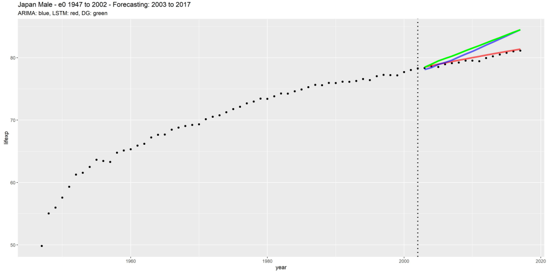
Observations, limitations and future developments

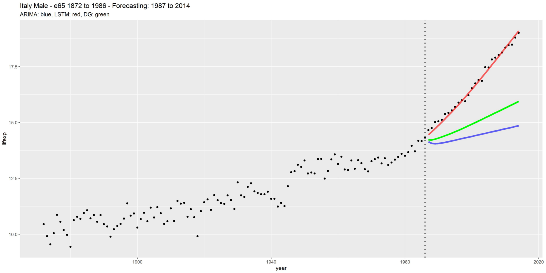
- **Observation.** nonlinear forecasting of k: this characteristic has important consequences on the forecast trend. Could be a more plausible way to understand mortality patterns.
- **Limitation, Future developments & Preliminary results.**
  - Confidence interval for Deep Learning Estimation .

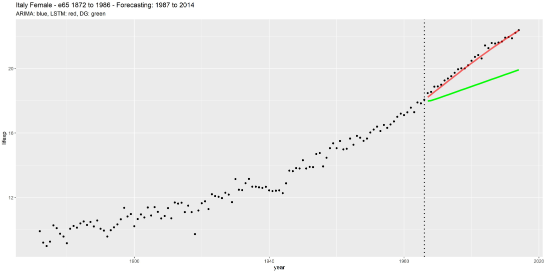
Life Expectancy Investigation...

Life Expectancy LSTM






Italy Male - e65 1872 to 1986 - Forecasting: 1987 to 2014
ARIMA: blue, LSTM: red, DG: green
lifexp
17.5
15.0
12.5
10.0
1900
1940
1980
2020
year


Italy Female - e65 1872 to 1986 - Forecasting: 1987 to 2014
ARIMA: blue, LSTM: red, DG: green
lifexp
20
16
12
1900
1940
1980
2020
year