A neural network analyzer for mortality forecast

D. Hainaut\textsuperscript{1}

\textsuperscript{1}ISBA, Université Catholique de Louvain

BNP PARIBAS CARDIFF 21-3-17
Motivation

- Improvement of longevity observed over the last century is a matter of concerns for the insurance industry.

- Reasons: reduction of mortality caused by infectious and chronic diseases at older ages.

- Standard in the industry model of Lee and Carter (LC) (1992) and its extensions:
  - Brouhns et al. (2002): log-likelihood maximization
  - Renshaw and Haberman (2003): multi-factor version
  - Renshaw and Haberman (2006): cohort effect
Motivation

- Hypothesis: a linear dynamic between log-forces of mortality and latent time processes.
- This hypothesis allows to use a Principal Component analysis to estimate parameters.
- PCA: extraction method that attempts to characterize lower-dimensional structure in large multivariate datasets.
- Alternative to PCA:
  - Kramer (1991)/ neural network based generalization of PCA to the nonlinear feature extraction problem (NLPCA). Field of application: chemical engineering (Dong and McAvoy 1996), psychology (Fotheringham and Baddeley 1997), climatic mathematics (Monahan, 2000).
  - Hastie and Stuetzle (1989): principal curves and surfaces (PCS). Equivalent to NLPCA.
Contributions

1) One of the first article that applies the NLPCA to reduce the dimension of the term structure of mortality rates. Sparse literature on this field (Atsalakis et al. (2007), Garko (2015), Puddu and Menotti (2009). Previous attempt use a neural net to model and forecast a unique time serie (not the term structure).

2) Calibration of the network with a genetic algorithm (McNelis (2005)). Given that the number of parameters, gradient descents, are inefficient and the risk of finding a local minimum is important.

3) Numerical illustration with French mortality rates from 1946 to 2014: the Neural Network (NN) approach significantly outperforms the LC model.
Bench mark: the LC model

The Lee-Carter model is used as benchmark to evaluate the NN approach. The death probability:

\[ q(t, x) \approx 1 - \exp(-\mu(t, x)) . \]

In multi-factor LC models, the log-force of mortality is a linear combination of \( d \) time latent factors \( \kappa^i_t \), with covariates that depend on the age:

\[
\ln \mu(t, x) = \alpha_x + \sum_{i=1}^{d} \beta_x^i \kappa_t^i. \tag{1}
\]

\( \alpha_x \in \mathbb{R}^{x_{\text{max}}} \): permanent impact of age. \( \beta_x^{i=1\ldots d} \in \mathbb{R}^{x_{\text{max}}} \): marginal effect of latent factors on mortality at each age.

For identifiability, we impose:

\[
\sum_x \beta_x^i = 1 \quad \sum_t \kappa_t^i = 0 \quad i = 1, \ldots, d. \tag{2}
\]
Benchmark: the LC model

- Dataset of mortality forces \((\mu(t, x))\) range from year \(t_{min}\) to \(t_{max}\) and from age \(x_{min}\) to \(x_{max}\).

- First stage, from the constraint \(\sum_{t=t_{min}}^{t_{max}} \kappa^i_t = 0\),

\[
\alpha_x = \frac{1}{t_{max} - t_{min} + 1} \sum_{t=t_{min}}^{t_{max}} \ln \mu(t, x) \quad x = x_{min}, \ldots, x_{max}.
\]

- The next step consists to perform a PCA on adjusted observations denoted by \(X\):

\[
X := (\ln \mu(t, x) - \alpha(t, x))_{t=t_{min}\ldots t_{max}, x=x_{min}\ldots x_{max}}
\]

to find the \(d\)-plet \(\kappa_t := (\kappa^1_t, \kappa^2_t, \ldots, \kappa^d_t)\) and \((\beta^i_x)_{x=x_{min}\ldots x_{max}}\).
Benchmark: the LC model

- Last stage: a time series model is specified for the latent processes.
- Most of authors use an AR(1) model or a random walk with drift.
- In this paper, we opt for the second choice and assume that increments of $\kappa^i_t$ are Gaussian random variables with a mean $\gamma_i$ and a variance $\sigma^2_i$:

$$\kappa^i_t - \kappa^i_{t-1} = \gamma_i + \sigma_i \epsilon_t \quad i = 1, \ldots, d$$

(3)

where $\epsilon_t$ is a standard normal random variable.
The neural net analyzer

- Most neurons receive input signals throughout their dendritic trees (sometimes many thousands of input signals).
- Whether or not a neuron is excited into firing an impulse depends on the sum of all of the excitatory and inhibitory signals.
- If the neuron does end up firing, the nerve impulse, or action potential, is conducted down the axon.
The neural net analyzer

The signals (e.g., $x_0$) interact multiplicatively with the dendrites of the other neuron based through synaptic weights (e.g., $w_0 x_0$). The dendrites carry input signals to the cell body, where they all are summed. If the final sum is above a threshold, the neuron fires, sending a spike along its axon.
The activation function $f$ is a sigmoid:

$$a = \text{tansig}(n) = \frac{2}{1 + e^{-2n}} - 1$$
The neural net analyzer

- Input signals = centered log-forces of mortality:

\[ X(t) := \begin{pmatrix} \ln \mu(t, x_{\min}) - \alpha x_{\min} \\ \vdots \\ \ln \mu(t, x) - \alpha x \\ \vdots \\ \ln \mu(t, x_{\max}) - \alpha x_{\max} \end{pmatrix} \quad t = t_{\min}, \ldots, t_{\max} \]

- We aim to calibrate the weights \( (w_i)_{i=1}^l \) defining an encoding and a decoding functions: \( f^{enc} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^d \) and \( f^{dec} : \mathbb{R}^d \rightarrow \mathbb{R}^{n_x} \).
The neural net analyzer

- Architecture of the neural net:

![Neural network diagram with nodes and edges representing the flow of data from input to output through various layers and functions.](image-url)
The neural net analyzer

- The encoding function, $f^{enc}(.)$, projects curves of mortality rates at time $t$, $X(t) \in \mathbb{R}^{nx}$ in $\mathbb{R}^d$. As in the LC model, projections in $\mathbb{R}^d$ are in a $d$-plet $\kappa^{nn}_t := (\kappa^{nn,1}_t, ..., \kappa^{nn,d}_t)$:

$$\kappa^{nn}_t := f^{enc}(X(t)) \quad t = t_{\text{min}}, ..., t_{\text{max}}.$$ 

- The decoding function $f^{dec}(.)$ uses $\kappa^{nn}_t$ to build an approximation $\hat{X}(t) \in \mathbb{R}^{nx}$ of the initial curve of log-mortality rates:

$$\hat{X}(t) := f^{dec}(\kappa^{nn}_t).$$

- Compared to the original LC model, the linear relation $\beta_x \kappa_t$ is replaced by a non-linear function:

$$\ln \mu(t, x) = \alpha_x + f^{dec}(x, \kappa^{nn}_t) \quad (4)$$

- Calibration of weights by minimization of the quad. error:

$$\arg \min_{\text{weights}} \sum_{t=t_{\text{min}}}^{t_{\text{max}}} \|X(t) - \hat{X}(t)\|_2^2.$$
The neural net analyzer

Initial population generation

Selection

Crossover

Mutation

Evaluation

Done?

Yes

Over!
The neural net analyzer

- Create a population of candidate parameters (weights that multiply input signals) (100 individuals).

- **Selection**: choose randomly two pairs of individuals from the population. The two winning individuals \( P_1, P_2 \in \Omega \) are retained for “breeding” purposes.

- **Crossover** of \( P_1 \) and \( P_2 \) (3 types: Shuffle, Arithmetic, Single-point), 2 children: \( C_1 \) and \( C_2 \).

- **Mutation** of children. With a decreasing probability,

\[
\tilde{C}_{i,k} = \begin{cases} 
C_{i,k} + s \left( 1 - r_2^{(1 - \frac{G}{G^*})^b} \right) & \text{if } r_1 > 0.5 \\
C_{i,k} - s \left( 1 - r_2^{(1 - \frac{G}{G^*})^b} \right) & \text{if } r_1 \leq 0.5
\end{cases}
\]

where \( r_1 \) and \( r_2 \) are r.v. on \([0, 1]\) and \( s \) is a standard normal r.v.

- The “family” \((P_1, P_2, C_1, C_2)\) engage in a **tournament**. The two vectors with the best goodness of fit, whether parents or children, survive and pass to the next generation.
Application

- We work with raw mortality rates observed for the French population over the period 1946 to 2014 (HMD.org).

- Calibration: from 1946 up to 2000. To compare the predictive capability, projection by simulation over fourteen years (10 000 simulations).

- Comparison with mortality over the period 2001-2014.
## Application

<table>
<thead>
<tr>
<th>Dim.</th>
<th>Coef.</th>
<th>Avg.</th>
<th>$|X - \hat{X}|_2$</th>
<th>$n_l$</th>
<th>$n_d$</th>
<th>Coef.</th>
<th>Avg.</th>
<th>$|X - \hat{X}|_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>180</td>
<td>0.0024</td>
<td></td>
<td>2</td>
<td>2</td>
<td>368</td>
<td>0.0010</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>270</td>
<td>0.0023</td>
<td></td>
<td>3</td>
<td>2</td>
<td>55</td>
<td><strong>0.0009</strong></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>360</td>
<td>0.0023</td>
<td></td>
<td>3</td>
<td>3</td>
<td>558</td>
<td>0.0009</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>450</td>
<td>0.0023</td>
<td></td>
<td>4</td>
<td>3</td>
<td>744</td>
<td>0.0008</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>540</td>
<td>0.0023</td>
<td></td>
<td>4</td>
<td>4</td>
<td>752</td>
<td>0.0008</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>630</td>
<td>0.0023</td>
<td></td>
<td>5</td>
<td>4</td>
<td>940</td>
<td>0.0008</td>
<td></td>
</tr>
</tbody>
</table>

Application
Jarque Bera test applied to increments of latent factors over the period 1970-2000, for neural analyzer with three neurons in input / output layers and two neurons in the intermediate layer.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Normality</th>
<th>p-value</th>
<th>JB statistic</th>
<th>Critical Value 5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\kappa_{t-1}^{nn,1} - \kappa_{t-1}^{nn,1}$</td>
<td>Accept</td>
<td>0.5000</td>
<td>0.0762</td>
<td>4.4496</td>
</tr>
<tr>
<td>$\kappa_{t}^{nn,2} - \kappa_{t-1}^{nn,2}$</td>
<td>Accept</td>
<td>0.3698</td>
<td>1.2296</td>
<td>4.4496</td>
</tr>
</tbody>
</table>
### Application

<table>
<thead>
<tr>
<th>Dim.</th>
<th>Coef.</th>
<th>Avg.</th>
<th>$|X - \hat{X}|_2$</th>
<th>$n_l$</th>
<th>$n_d$</th>
<th>Coef.</th>
<th>Avg.</th>
<th>$|X - \hat{X}|_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>180</td>
<td>0.0049</td>
<td></td>
<td>2</td>
<td>2</td>
<td>368</td>
<td>0.0030</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>270</td>
<td>0.0049</td>
<td></td>
<td>3</td>
<td>2</td>
<td>55</td>
<td><strong>0.0032</strong></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>360</td>
<td>0.0049</td>
<td></td>
<td>3</td>
<td>3</td>
<td>558</td>
<td>0.0034</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>450</td>
<td>0.0049</td>
<td></td>
<td>4</td>
<td>3</td>
<td>744</td>
<td>0.0035</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>540</td>
<td>0.0049</td>
<td></td>
<td>4</td>
<td>4</td>
<td>752</td>
<td>0.0038</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>630</td>
<td>0.0049</td>
<td></td>
<td>5</td>
<td>4</td>
<td>940</td>
<td>0.0042</td>
<td></td>
</tr>
</tbody>
</table>

French population, 2001-2014, forecast
Comparison of real log-mortality rates in 2014/2001 to the average log-forces of mortality simulated with the 3-2-3 neural analyzer
### Application

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{LC}(t)$</th>
<th>$e_{40}^{LC}(t)$</th>
<th>$e_{60}^{LC}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>60.919</td>
<td>41.762</td>
<td>24.077</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{Obs}(t) - e_{20}^{LC}(t)$</th>
<th>$e_{40}^{Obs}(t) - e_{40}^{LC}(t)$</th>
<th>$e_{60}^{Obs}(t) - e_{60}^{LC}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>2.4362</td>
<td>2.1789</td>
<td>1.7942</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{NN}(t)$</th>
<th>$e_{40}^{NN}(t)$</th>
<th>$e_{60}^{NN}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>62.231</td>
<td>42.992</td>
<td>25.163</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{Obs}(t) - e_{20}^{NN}(t)$</th>
<th>$e_{40}^{Obs}(t) - e_{40}^{NN}(t)$</th>
<th>$e_{60}^{Obs}(t) - e_{60}^{NN}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>1.1243</td>
<td>0.9485</td>
<td>0.7076</td>
</tr>
</tbody>
</table>
Breakdown of means and standard deviations of relative average errors of calibration, per age. We compare the 3 Dimensions Lee Carter model to the 3-2-3 Neural network.
Application

Comparison of simulated densities for $\ln \mu(2010, 40)$, yield by the LC and Neural models.
Predicted cross-sectional lifetime expectancies with the LC and neural models, over the period 2001-2100.
### Application

Long term predictions of cross-sectional life expectancies, computed by simulations with the LC and Neural models, over the period 2001-2100.

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{NN}(t)$</th>
<th>$e_{40}^{NN}(t)$</th>
<th>$e_{60}^{NN}(t)$</th>
<th>$e_{80}^{NN}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001</td>
<td>60.36</td>
<td>41.23</td>
<td>23.62</td>
<td>8.982</td>
</tr>
<tr>
<td>2025</td>
<td>63.59</td>
<td>44.29</td>
<td>26.32</td>
<td>10.76</td>
</tr>
<tr>
<td>2050</td>
<td>65.81</td>
<td>46.42</td>
<td>28.25</td>
<td>12.25</td>
</tr>
<tr>
<td><strong>2100</strong></td>
<td><strong>68.04</strong></td>
<td><strong>48.58</strong></td>
<td><strong>30.28</strong></td>
<td><strong>14.08</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$e_{20}^{LC}(t)$</th>
<th>$e_{40}^{LC}(t)$</th>
<th>$e_{60}^{LC}(t)$</th>
<th>$e_{80}^{LC}(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001</td>
<td>60.25</td>
<td>41.12</td>
<td>23.53</td>
<td>8.925</td>
</tr>
<tr>
<td>2025</td>
<td>61.47</td>
<td>42.29</td>
<td>24.53</td>
<td>9.485</td>
</tr>
<tr>
<td>2050</td>
<td>62.71</td>
<td>43.48</td>
<td>25.54</td>
<td>10.06</td>
</tr>
<tr>
<td><strong>2100</strong></td>
<td><strong>65.05</strong></td>
<td><strong>45.73</strong></td>
<td><strong>27.48</strong></td>
<td><strong>11.20</strong></td>
</tr>
</tbody>
</table>
Conclusion

- Totally new approach based on a neural network to predict and simulate the future human mortality.
- The neural analyzer outperforms the LC model and its multi-factor extensions. The average error calibration error is three times lower than the error for the 3 dimension LC model.
- Excellent predictive power of the neural approach. However, the number of neurons must be chosen carefully to avoid over-parameterization.
- On the other hand, the longevity risk predicted by the neural analyzer is three times bigger than the one forecast by the LC model.
- Neural networks are promising for other applications in actuarial sciences (loss reserving)