Mortality Forecasting via Multi-Task Neural Networks

Luca De Mori[1], Steven Haberman[2], Pietro Millossovich[3], and Rui Zhu[4]

[1,2,3,4]Bayes Business School
[3]DEAMS: Università degli Studi di Trieste

January 4, 2025

Abstract

In recent decades, analysing the progression of mortality rates has become very important for both public and private pension schemes, as well as for the life insurance branch of insurance companies. Traditionally, the tools used in this field were based on stochastic and deterministic approaches that allow extrapolating mortality rates beyond the last year of observation. More recently, new techniques based on machine learning have been introduced as alternatives to traditional models, giving practitioners new opportunities. Among these, neural networks play an important role due to their computation power and flexibility to treat the data without any probabilistic assumption. In this paper, we apply multi-task neural networks, whose approach is based on leveraging useful information contained in multiple related tasks to help improve the generalized performance of all the tasks, to forecast mortality rates. Finally, we compare the performance of multi-task neural networks to that of existing single-task neural networks and traditional stochastic models on mortality data from seventeen different countries.

1 Introduction

Artificial neural networks, abbreviated as neural networks (NNs), are a subfield of machine learning, commonly referred to as deep learning, that have been applied to demography in recent years for analyzing and predicting mortality rates and other mortality-related metrics. Generally speaking, an NN can be seen as a universal

function approximator, i.e., a mapping that, once properly structured and trained, can approximate any function that links a series of inputs to outputs, see Hornik et al. (1989). Focusing on mortality forecasting, there are two advantages of using NNs instead of traditional stochastic models such as the Lee-Carter Model and its extensions, see Lee and Carter (1992). Firstly, they simplify the model definition and free us from specifying how variables, such as age and calendar year, interact. Secondly, they allow us to consider the mortality experience of several populations simultaneously. Among the most important contributions to NNs applied to mortality forecasting, the following studies are among the ones that stand out: Richman and Wüthrich (2021), and Perla and Scognamiglio (2023) exploit feedforward NNs; Nigri et al. (2019), Chen and Khaliq (2022), Lindholm and Palmborg (2022), and Euthum et al. (2024) use long short-term memory NNs; Perla et al. (2021), Wang et al. (2021), and Schnürch and Korn (2022) utilize convolutional NNs; and Hainaut (2018) as well as Scognamiglio (2022) apply hybrid models.

In this paper we focus on simultaneously forecasting the mortality rates of a given set of countries. In order to do that, we implement a methodology called multi-task NNs, consisting of several NNs that share a certain number of parameters. In the past years, multi-task deep learning has been applied with promising results in several fields, such as computer vision, see Girshick (2015), natural language processing, see Collobert and Weston (2008), speech recognition, see Deng et al. (2013), and insurance, see Lindholm et al. (2023). Finally, we recommend Zhang and Yang (2021) for a theoretical overview of multi-task NNs.

Specifically, we propose a hierarchical network structure for multi-population mortality forecasting. The lower hidden layers of these multi-task NNs, i.e. those closer to the input layer, are shared across all countries, capturing the general properties of mortality trends, while the higher hidden layers, i.e. those closer to the output layer, are country-specific or shared only within clusters of countries with more similar past mortality trends. The clusters are obtained by applying the k-means clustering machine learning technique to past data for some key mortality metrics, i.e. life expectancy and lifetime standard deviation. Finally, each country has its own layer to learn its distinct property.

In this paper, we quantitatively compare multi-task NNs with pre-existing single-task NNs and stochastic models considering mortality data of seventeen different countries. The comparison is based of mortality rates, life expectancy and lifetime standard deviation forecasting errors. With multi-task NNs, we expect to improve the performance of NNs at country-specific level dedicating more parameters to single countries.

Our main conclusions are that multi-task NNs performance compared to single-task

NNs and stochastic models depends on the metric, age range, and training period considered. Overall, single-task NNs gives the best results in terms of mortality rates forecasting error, while multi-task NNs and stochastic models have the lowest forecasting error respectively for life expectancy and lifetime standard deviation. Furthermore, implementing a weighting scheme in their training improves the multi-task NNs performance, especially for life expectancy and lifetime standard deviation when considering wider age ranges.
The remainder of this paper is organized as follows. Section 2 contains a general theoretical framework for feedforward NNs, followed by a practical application to mortality rates forecasting. In Section 3, we introduce feedforward multi-task NNs and present the NNs proposed by us. In Section 4, the data used in the empirical analysis and settings for the training of the NNs are reported. In Section 5, the numerical results are presented and discussed. In Section 6, we draw the conclusion and propose some future outlooks.

2 Feedforward neural networks

Feedforward neural networks (FNNs) are the most basic type of NN. Information flows in one direction, from input neurons through hidden layers to output neurons, see Schmidhuber (2015). Cycles and loops are not present in this type of NN. They are generally used for classification, regression, and pattern recognition, and, in particular, they can be applied to mortality forecasting. In this context, FNNs are especially useful for mortality forecasting when the focus is on modelling the relationship between input features (age, calendar year, cohort year, etc.) and mortality rates.

2.1 Notation and terminology

Given a set of L input variables $\mathbf{X} = (X_1, \ldots, X_L)$ that can be numerical or categorical, or a combination of them, and the corresponding output Y, we have to focus on the hyperparameters of the NNs, i.e. those settings that have to be set before the parameters are learnt in the training process, see Goldberg (2017) and Prince (2023). These hyperparameters are:

- N: number of hidden layers in the NN.
- L_1,...,L_N: numbers of neurons for each layer.

- $f^{(1)}$,..., $f^{(N+1)}$: activation functions of the NN. Notice: $f^{(1)}$ will be the activation function of the first hidden layer, while $f^{(N+1)}$ will be the activation function of the output layer. Some popular activation functions, that are also used in this paper, are Sigmoid (also called Logistic), Hyperbolic Tangent (tanh), and Rectified Linear Unit (ReLU), see Dubey et al. (2022).

Once we have specified these hyperparameters, it is possible to estimate the parameters $\mathbf{B}^{(1)} \in \mathbb{R}^{L_1 \times L}$, $\mathbf{B}^{(2)} \in \mathbb{R}^{L_2 \times L_1}$, ..., $\mathbf{B}^{(N)} \in \mathbb{R}^{L_N \times L_{N-1}}$, $\mathbf{B}^{(N+1)} \in \mathbb{R}^{1 \times L_N}$, and $\mathbf{c}_1 \in \mathbb{R}^{L_1}$, $\mathbf{c}_2 \in \mathbb{R}^{L_2}$, ..., $\mathbf{c}_N \in \mathbb{R}^{L_N}$, $c_{N+1} \in \mathbb{R}$, that represent respectively weight matrices and intercept vectors. These are the parameters that are learned during the training of the network.
The layers will be so computed, using matrix notation:

$$\mathbf{Z}^{(1)} = f^{(1)}(\mathbf{c}_1 + \mathbf{B}^{(1)}\mathbf{X}) \in \mathbb{R}^{L_1}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^L$ is the input vector,

$$\mathbf{Z}^{(j)} = f^{(j)}(\mathbf{c}_j + \mathbf{B}^{(j)}\mathbf{Z}^{(j-1)}) \in \mathbb{R}^{L_j}, \ j = 2, \ldots, N. \tag{2}$$

Finally, for the output layer:

$$\hat{Y} = Z^{(N+1)} = f^{(N+1)}(c_{N+1} + \mathbf{B}^{(N+1)}\mathbf{Z}^{(N)}) \in \mathbb{R}. \tag{3}$$

We now discuss the training of the NN during which all the weight matrices and intercept vectors are estimated through a process called backpropagation. In order to do that, the additional hyperparameters reported below have to be specified, see Prince (2023).

- The **loss function** is the criterion through which, starting from the observed value of the outputs and the predicted output of the network, we calculate the quantity that has to be minimized when we train the NN. In the remainder of this paper, the mean squared error (MSE) is used as loss function:

$$MSE = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2, \tag{4}$$

  where n is the number of observations, Y_i are the observed values of the output, $\hat{Y}_i$ are the values predicted by the NN as in equation (3).

- The **optimizer** is the algorithm used during the training phase to adjust the parameters of the neural network in order to minimize the loss. In the remainder of this paper, we will utilize the Adam optimizer (Adaptive Moment

Estimation), a gradient-based optimization algorithm that leverages first-order (gradient) and second-order (squared gradient) moment estimates to adapt the learning rate for each parameter, see Kingma and Ba (2014).

- The **number of epochs** is the amount of times the optimizer runs on the training set.
- The **validation set** is a subset of the available data used to provide an unbiased evaluation of a model fit identifying eventual overfitting while the training set is used to tune the NN parameters.
- The **batch size** defines the number of training samples processed simultaneously before the model's weights are updated. It determines how many samples are passed through the network in each forward and backward pass during training.
- The **learning rate** controls the size of the steps taken during the optimization process when adjusting the weights of the model.

2.2 Feedforward single-task neural network applied to mortality forecasting

In this subsection, we are going to provide a framework for forecasting of mortality rates with feedforward single-task NNs based on the paper of Richman and Wüthrich (2021). The input variables considered in the NNs are calendar year t, age x, gender g, and country p, $\tilde{\mathbf{X}} = (t, x, g, p)$, and they will be treated as categorical with the single exception of calendar year, which will be treated as numerical, while the output, Y, is the central mortality rate $m_{x,t}^{(g,p)}$ at age x, year t, gender g and population p. In order to treat the categorical variables in the input layer, embedding layers are used, see Mikolov et al. (2013). An embedding layer, from a mathematical point of view, is a function that maps discrete data into continuous vector representations. So, given a categorical variable with b distinct categories or levels (e.g., the categories "male" and "female" for the variable "gender", the different countries for the variable "country", etc.), and a dimension d, which represents the size of the continuous embedding space (e.g., each categorical level will be represented by a vector in $\mathbb{R}^d$), the embedding layer performs the mapping

$$f : \{0, 1, \ldots, b-1\} \to \mathbb{R}^d. \tag{5}$$

In a NN, the embedding layer can be identified with a parametrized matrix belonging to $\mathbb{R}^{b \times d}$. The parameters of the embedding layers, similarly to the parameters of

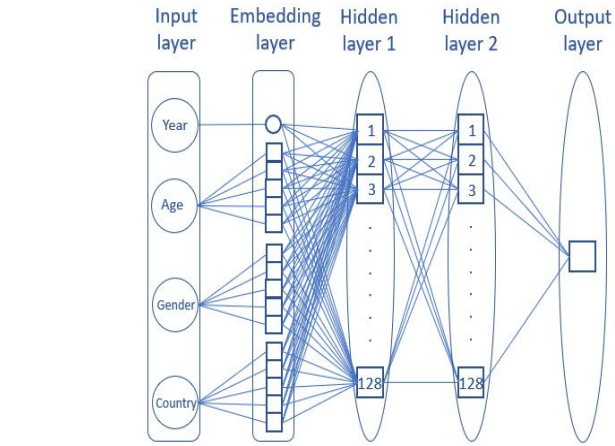
the hidden layers, are learned as the network is trained. Notice that if $d = 1$, the embedding layer becomes equivalent to the classical treatment of categorical variables in regression models: each level of the variable is coded with a specific value. Following Richman and Wüthrich (2021), d is set equal to 5 for all three categorical variables, so: $x \to \mathbf{x} \in \mathbb{R}^5$, $g \to \mathbf{g} \in \mathbb{R}^5$, and $p \to \mathbf{p} \in \mathbb{R}^5$. Once embedding vectors ($\mathbf{x}$, $\mathbf{g}$ and $\mathbf{p}$) have been created, we have the vector $\mathbf{X} = (t, \mathbf{x}, \mathbf{g}, \mathbf{p}) \in \mathbb{R}^{16}$ that represents the actual input that will be passed to the first hidden layer of the NN. The number of hidden layers here considered differs by the NN considered, $N = 2$ or 5; the number of neurons in each hidden layer is equal to 128 neurons, $L_1 = \cdots = L_N = 128$; the output layer that represents the mortality rate for the gender g in the country p at age x in year t has one neuron, $L_{N+1} = 1$, with sigmoid activation function,

$$m_{x,t}^{(g,p)} = Z^{(N+1)} = \frac{1}{1 + e^{-(c_{N+1} + \mathbf{B}^{(N+1)} \mathbf{Z}^{(N)})}}. \tag{6}$$

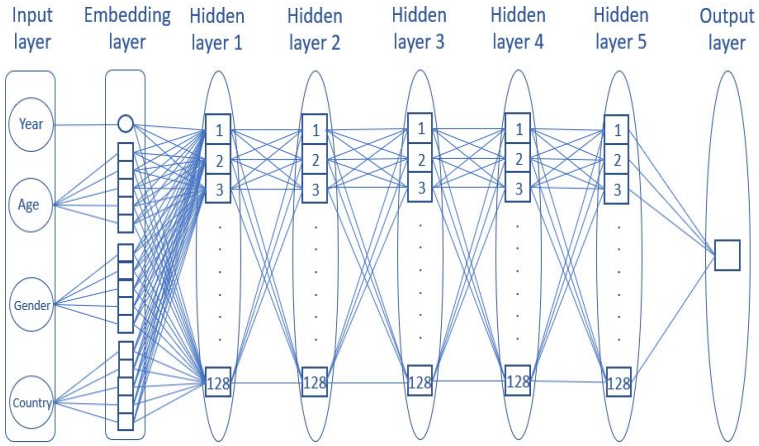
The NNs also differ among themselves by the type of activation function in the hidden layers, $f^{(1)} = \cdots = f^{(N)} = \tanh$ or $f^{(1)} = \cdots = f^{(N)} =$ReLU, and by the presence or not of a direct connection, called a skip connection, between the embedding layer and the last hidden layer. These NNs are referred as DEEPi, $i = 1, \ldots, 6$, and the details about their architecture are reported in Table 1 and in Figure 1.

Table 1: Summary of the NNs DEEPi, $i = 1, \ldots, 6$, architectures.

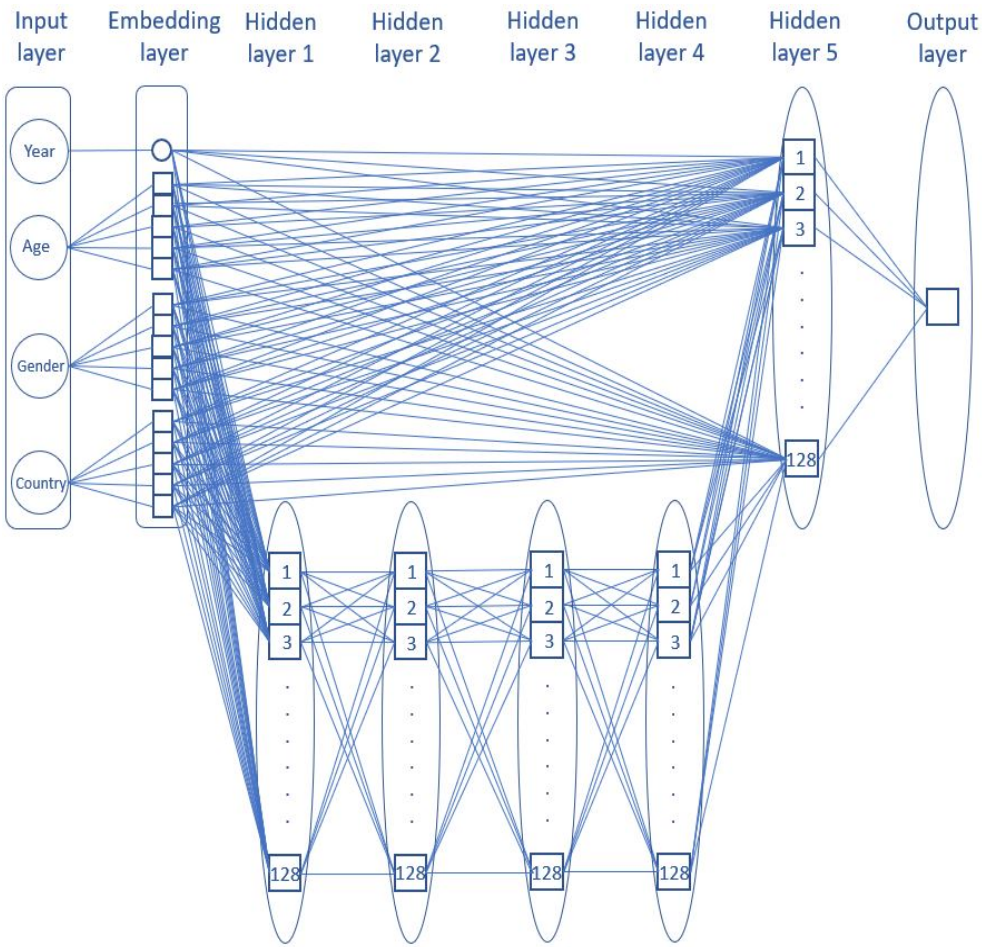
model	# hidden layers	activation function	skip connection
DEEP1	2	ReLU	No
DEEP2	2	tanh	No
DEEP3	5	ReLU	No
DEEP4	5	tanh	No
DEEP5	5	ReLU	Yes
DEEP6	5	tanh	Yes



(a) DEEP1 and DEEP2



(b) DEEP3 and DEEP4



(c) DEEP5 and DEEP6

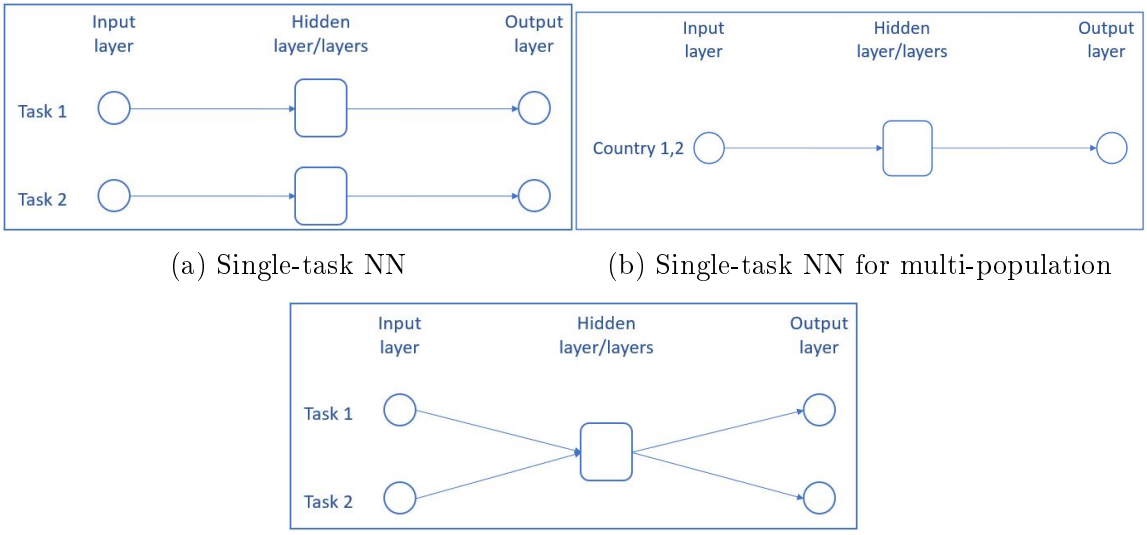
Figure 1: Architecture of the NNs DEEPi, $i = 1, \ldots, 6$ as described in Table 1.

3 Multi-task neural networks

Generally speaking, multi-task deep learning consists of different NNs (one for each task) that share at least one layer. The shared part of the NNs can be the input layer, one or more hidden layers, or a combination of them. It is relevant to notice that the output layer cannot be shared, as we must have one output neuron for each task. Given that we have P different datasets, each corresponding to a distinct country, this paper employs multi-task NNs with a multi-input, multi-output structure, see Menet et al. (2023). Specifically, these NNs share hidden layers across tasks while maintaining P separate input and output layers.

Let us now consider $P > 1$ countries and the following P tasks: $T_p =$ "forecasting the mortality rates for p^{th} country", $p = 1, \ldots, P$. If we want to forecast the mortality rates of the P countries using a feed-forward NN, then we have three different options. The first consists of using P different NNs with their own input layer, hidden layers, and output layer, with the p^{th} of them to predict the mortality rates of the p^{th} country. This solution can be called single-task NNs approach and is graphically represented in Figure 2(a). The second option is to use one single-task NN like those presented in Section 2 (see Figure 2(b)). The third option is to consider the P NNs sharing one or more of their hidden layers, and in this way we will have a multi-task NN, see Figure 2(c). Generally, a multi-task NN has three main advantages compared to using P different single-task NNs. Firstly, it noticeably improves the training time as we optimize just one NN rather than P different NNs. Secondly, as the countries are likely to share some common behaviors in their mortality evolution, such as the long-term trend of improving mortality, there will likely be mutual benefits for all the P tasks by training them together, see Crawshaw (2020). Thirdly, the multi-task neural network will operate on a single large dataset rather than P smaller datasets, thereby capturing a greater amount of information and leading to more robust predictions.

At this point, we pose a different question: what is the advantage of using a multi-task NN (see Figure 2(c)) compared to a single-task NN, as presented in Section 2 (see Figure 2(b))? The primary advantage is that a multi-task NN not only shares knowledge across related tasks through shared layers (as in single-task NNs) but also enables task-specific specialization via country-specific layers. For instance, when a single-task NN is trained on a large set of countries, it can become dominated by the majority countries—those with similar mortality trends—while minority countries, such as the US and Japan, which exhibit distinct mortality patterns, tend to be under-represented. This imbalance often leads to poorer predictions for the minority countries. In the Results Section, among other things, we will evaluate whether



(a) Single-task NN

(b) Single-task NN for multi-population

(c) Multi-task NN

Figure 2: Illustrations of single and multi-task NNs for mortality prediction.

the multi-task structure in 2(c), with its country-specific layers designed to capture unique mortality patterns, can address this issue effectively.

3.1 Architecture of the multi-task NNs for mortality forecasting

Similarly to the NNs discussed in Section 2, the multi-task NNs will be of the feedforward type. They will have P input layers, one for each country, where the variables are calendar year, age, country, and gender. There are then P different embedding layers where each categorical variable, i.e. age, country and gender, is transformed into a vector belonging to $\mathbb{R}^5$ as explained in Section 2.2. These embedding layers are fully connected to two hidden layers with 128 neurons and tanh activation function, following Richman (2022). The second of these intermediate layers is then fully connected to a third hidden layer with 64 neurons and tanh activation function. From the third hidden layer, there are ramifications with P country-specific hidden layers having 32 neurons and tanh activation function. Finally, these P layers are connected to P output layers where the activation function is of Sigmoid type. Figure 3 reports a graphical representation of the just-described NN, which will be referred

to as MT1 in the remainder of this paper.

In more formal terms, the layers of MT1 will be computed as follows:

- Input layer:
$$\tilde{\mathbf{X}}_p = (t, x, g, p), \quad p = 1, \ldots, P. \tag{7}$$

- Embedding layer:
$$\mathbf{X}_p = (t, \mathbf{x}, \mathbf{g}, \mathbf{p}) \in \mathbb{R}^{16}, \quad p = 1, \ldots, P, \tag{8}$$
$$\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_P) \in \mathbb{R}^{16 \times P}. \tag{9}$$

- Hidden layer 1:
$$\mathbf{Z}^{(1)} = f^{(1)}(\mathbf{c}^{(1)} + \mathbf{B}^{(1)}\mathbf{X}) \in \mathbb{R}^{128}, \tag{10}$$
where $\mathbf{c}^{(1)} \in \mathbb{R}^{128}$, $\mathbf{B}^{(1)} \in \mathbb{R}^{128 \times 16 \times P}$, and $f^{(1)} = \tanh$.

- Hidden layer 2:
$$\mathbf{Z}^{(2)} = f^{(2)}(\mathbf{c}^{(2)} + \mathbf{B}^{(2)}\mathbf{Z}^{(1)}) \in \mathbb{R}^{128}, \tag{11}$$
where $\mathbf{c}^{(2)} \in \mathbb{R}^{128}$, $\mathbf{B}^{(2)} \in \mathbb{R}^{128 \times 128}$, and $f^{(2)} = \tanh$.

- Hidden layer 3:
$$\mathbf{Z}^{(3)} = f^{(3)}(\mathbf{c}^{(3)} + \mathbf{B}^{(3)}\mathbf{Z}^{(2)}) \in \mathbb{R}^{64}, \tag{12}$$
where $\mathbf{c}^{(3)} \in \mathbb{R}^{64}$, $\mathbf{B}^{(3)} \in \mathbb{R}^{64 \times 128}$, and $f^{(3)} = \tanh$.

- Country specific layers:
$$\mathbf{Z}_p^{(4)} = f^{(4)}(\mathbf{c}_p^{(4)} + \mathbf{B}_p^{(4)}\mathbf{Z}^{(3)}) \in \mathbb{R}^{32}, \quad p = 1, \ldots, P, \tag{13}$$
where $\mathbf{c}_p^{(4)} \in \mathbb{R}^{32}$, $\mathbf{B}_p^{(4)} \in \mathbb{R}^{32 \times 64}$, and $f^{(4)} = \tanh$.

- Output layers:
$$Z_p^{(5)} = f^{(5)}(c_p^{(5)} + \mathbf{B}_p^{(5)}\mathbf{Z}_p^{(4)}) \in \mathbb{R}, \quad p = 1, \ldots, P, , \tag{14}$$
where $\mathbf{c}_p^{(5)} \in \mathbb{R}$, $\mathbf{B}_p^{(5)} \in \mathbb{R}^{32}$, and $f^{(5)} = \text{sigmoid}$.

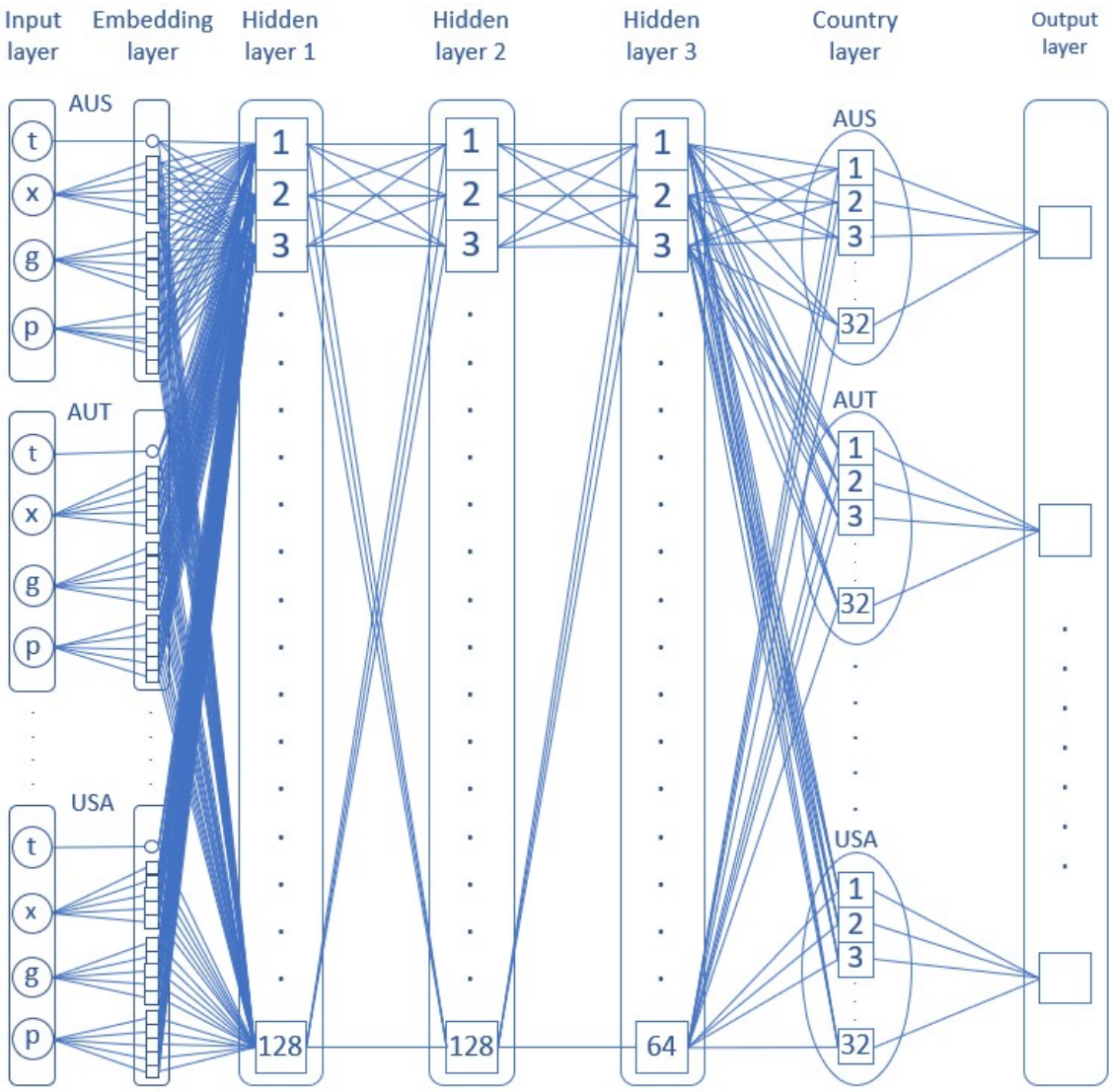


Figure 3: Graphical representation of the multi-task NN MT1.

3.2 Clustering of the third hidden layer

When considering a group of countries, it is natural that some share similar mortality trends while differing from others. These similarities and differences can stem from various social, economic, and geographical factors. For example, the Scandinavian countries, characterised by high wealth levels, extensive social welfare, and geographical proximity, will likely exhibit similar mortality evolutions. To enhance the performance of the multi-task network, we propose clustering the third hidden layer. This approach allows clusters of countries with similar mortality trends to share additional parameters, creating a hierarchical network structure. In this design, lower layers (i.e., hidden layers 1 and 2) capture the overall mortality trend across all countries, while the higher layer (i.e., hidden layer 3) extracts patterns shared by clusters of countries with similar trends. Finally, each country-specific layer learns the distinct mortality pattern for its respective country. To identify countries with similar survival patterns effectively, we can analyze historical mortality data using specific techniques that group countries into homogeneous sets known as clusters. For relevant studies on clustering techniques in the context of mortality forecasting, see Danesi et al. (2015), Nandini and Sanjjushri (2023), and Carracedo et al. (2018).

Having regard to the above discussion, we aim to assess the advantages of clustering the P countries based on their past mortality experiences and construct a new NN architecture that incorporates this clustering. To achieve this, we implement a two-step procedure for each $K = 2, 3$, where K denotes the number of clusters:

1. We use K-means clustering for grouping the P countries into K groups, see Scitovski et al. (2021). In order to do that, we consider the observed changes, in a chosen training period, of the following metrics:

   - Life expectancy for a newborn, truncated at age 90, see Dickson et al. (2019):

   $$\mathring{e}_{0:\overline{90}|,t} = \sum_{x=1}^{90} {}_{x-1}p_{0,t}\Big(1 - \frac{1}{2}q_{0+x-1,t}\Big), \tag{15}$$

   where $q_{x,t}$ and ${}_hp_{x,t}$ are respectively the 1 year probability of death at age x in year t and the probability of surviving for h years for an individual aged x in year t. These quantities can be derived from the mortality rates $m_{x,t}$ using the following formulas:

   $$q_{x,t} = \frac{m_{x,t}}{1 + \frac{1}{2}m_{x,t}}, \tag{16}$$

$$ {}_hp_{x,t} = \prod_{j=1}^{h}(1 - q_{x+j-1,t}). \tag{17} $$

- Standard deviation of the lifetime of a newborn, truncated at age 90:

$$ \mathrm{SD}_{0:\overline{90}|,t} = \sqrt{\sum_{x=0}^{89} {}_{x|1}q_{0,t}\ (x - \mathring{e}_{0:\overline{90}|,t})^2 + {}_{90}p_{0,t}(90 - \mathring{e}_{0:\overline{90}|,t})^2}, \tag{18} $$

where ${}_{h|1}q_{x,t}$ represents the deferred 1 year probability of death between ages $x+h$ and $x+h+1$ for an individual of age x in year t, and is given by

$$ {}_{h|1}q_{x,t} = {}_hp_{x,t}q_{x+h,t}. \tag{19} $$

2. For each K, we build the NN MTK, similar to MT1 but with K clustered hidden layers instead of hidden layer 3. These clustered hidden layers have 64 neurons and a tanh activation function, and are fully connected to hidden layer 2. Furthermore, they are connected with the country-specific layers based on the following rule: if a country is in cluster k, with $k = 1, \ldots, K$, then its country-specific layer is fully connected with cluster layer k. For the remaining parts of the NN, i.e. input layers, embedding layers, hidden layer 1, hidden layer 2, country-specific hidden layers, and output layers, they are specified as in MT1. Formulas for calculating input layer, embedding layer and hidden layers 1 and 2 are the same of (7)-(11). For the cluster layers, we have

$$ \mathbf{Z}_k^{(3)} = f^{(3)}(\mathbf{c}_k^{(3)} + \mathbf{B}_k^{(3)}\mathbf{Z}^{(2)}) \in \mathbb{R}^{64}, \quad k = 1, \ldots, K, \tag{20} $$

where $\mathbf{c}_k^{(3)} \in \mathbb{R}^{64}$, $\mathbf{B}_k^{(3)} \in \mathbb{R}^{64\times 128}$, and $f^{(3)} = \tanh$. For the country specific layers, we have

$$ \mathbf{Z}_p^{(4)} = f^{(4)}(\mathbf{c}_p^{(4)} + \sum_{k=1}^{K} I_{p,k}\mathbf{B}_p^{(4)}\mathbf{Z}_k^{(3)}) \in \mathbb{R}^{32}, \quad p = 1, \ldots, P, \tag{21} $$

where $\mathbf{c}_p^{(4)} \in \mathbb{R}^{32}$, $\mathbf{B}_p^{(4)} \in \mathbb{R}^{32\times 64}$, $f^{(4)} = \tanh$, and $I_{p,k} = 1$ if country p belongs to cluster k and $I_{p,k} = 0$ otherwise. Finally, the formula for the output layer is the same as (14).

The architectures of MT2 and MT3 can be found respectively in Figures 4 and 5.

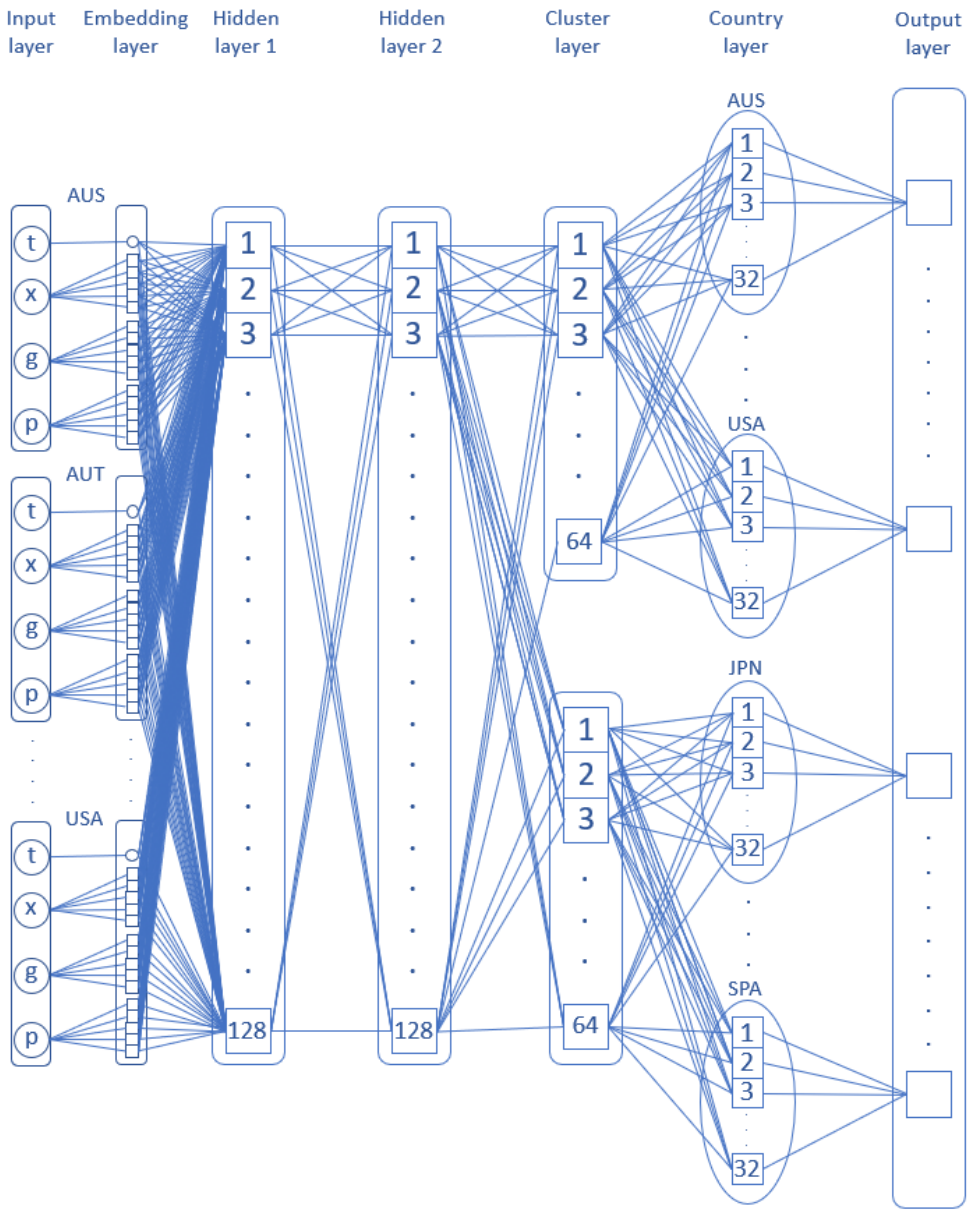


Figure 4: Graphical representation of the multi-task NN MT2.

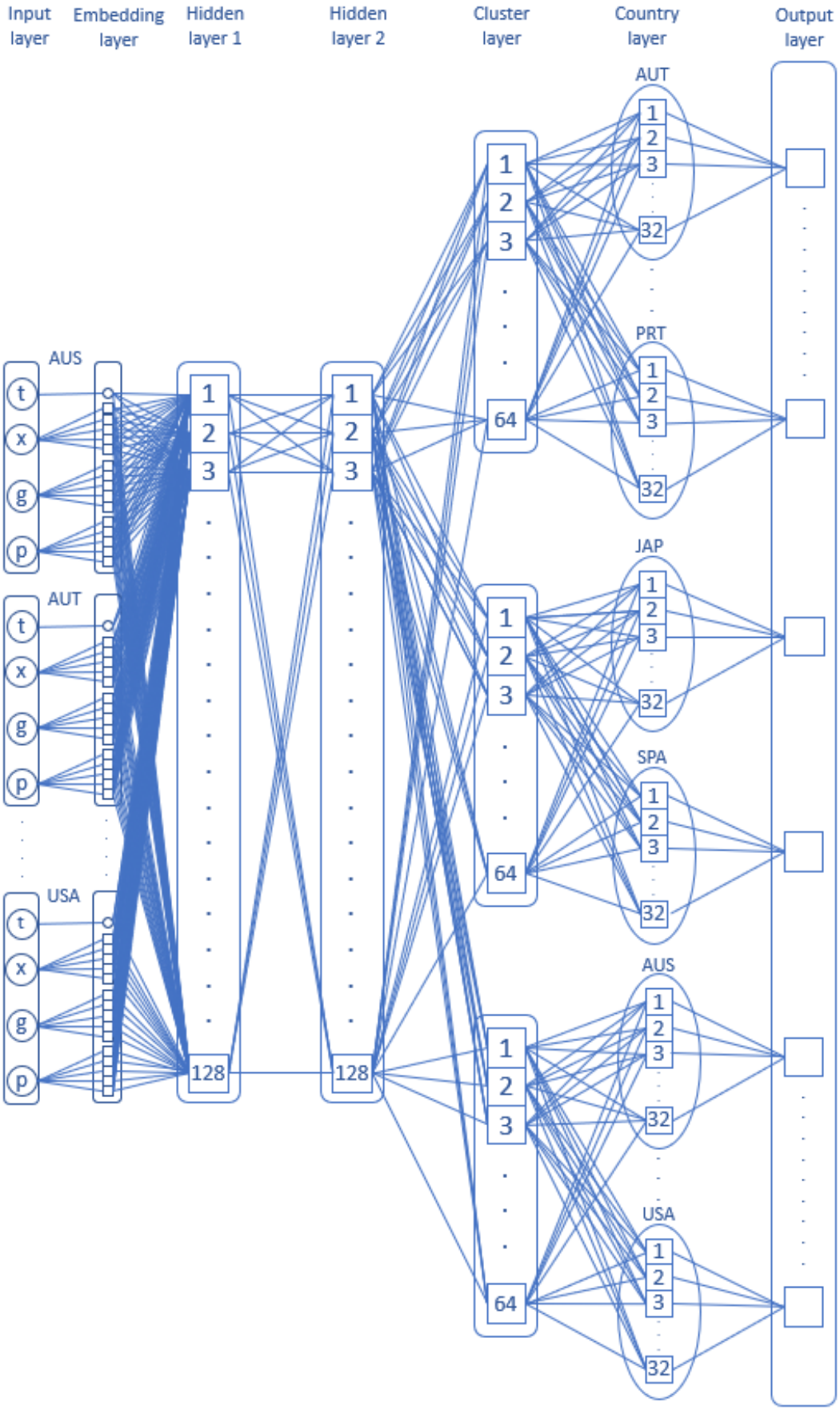


Figure 5: Graphical representation of the multi-task NN MT3.

4 Data, clustering and training

The choice of the countries we consider in the quantitative analysis is based on three factors: firstly, they must have data available in the HMD[1]. Secondly, historical data series for these countries must be complete from the year 1950 onwards. Thirdly, each selected country must have had a population of at least 3 million in 1950. In light of this, we consider historical mortality data for males and females from $P = 17$ countries: Australia, Austria, Belgium, Canada, Denmark, England & Wales, Finland, France, Italy, Japan, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland, and the US. For these countries, we consider the yearly central mortality rates obtained from HMD in three different age bands: 0-89, 20-89, and 55-89 to test the sensitivity of the different approaches with respect to the age band. Regarding the choice of the time interval, we considered, following Richman and Wüthrich (2021), a 50-years training period (1950-1999) and a 20-years test period (2000-2019). Finally, in order to study how the performance of the models varies based on the length of the training period, we also considered the following training sets: 1955-1999, 1960-1999, 1965-1999, 1970-1999, 1975-1999, and 1980-1999 (using 20-89 as reference age range).

The results of clustering using the approach described in Section 3.1 are reported in Table 2. Looking at the composition of the clusters obtained, we notice that Japan and Spain are clustered together in both cases. We also have a group of European countries, i.e. Austria, Belgium, Finland, France, Italy and Portugal that are clustered together, and another one with Australia, Canada, Denmark, England & Wales, Netherlands, Norway, Sweden, Switzerland, and the USA.

Regarding the training of the multi-task NNs, we used the following hyperparameters: 150 epochs when using 55-89 age band, and 250 epochs when using 20-89 and 0-89 age bands, batch size equal to 32, learning rate equal to 0.0005, Adam optimizer, and mean squared error as loss function:

$$L = \sum_{p=1}^{P} \frac{1}{n_p} \sum_{j=1}^{n_p} w_j^{(p)} (m_j^{(p)} - \hat{m}_j^{(p)})^2 \tag{22}$$

where n_p is the total number of observations for country p, and $w_j^{(p)}$ is the relative weight. We set $w_j^{(p)} = 1$ in the unweighted case and $w_j^{(p)} = \frac{1}{m_j^{(p)}}$ in the weighted case. In the following, results obtained using multi-task NNs are denoted by MT1, MT2, and MT3, in the unweighted case, and by MT1 w, MT2 w, and MT3 w, in

[1]Human Mortality Database: www.mortality.org.

the weighted case. Finally, we repeated the training of each NN 10 times in order to ensure robustness towards the effects of randomness in the training process.

Table 2: Results of clustering.

MT1		MT2		MT3	
Country	**Cluster**	**Country**	**Cluster**	**Country**	**Cluster**
Australia		Japan		Australia	
Austria		Portugal	1	Canada	
Belgium		Spain		Denmark	
Canada		Australia		England & Wales	
Denmark		Austria		Netherlands	1
England & Wales		Belgium		Norway	
Finland		Canada		Sweden	
France		Denmark		Switzerland	
Italy	1	England & Wales		USA	
Japan		Finland	2	Japan	2
Netherlands		France		Spain	
Norway		Italy		Austria	
Portugal		Netherlands		Belgium	
Spain		Norway		Finland	3
Sweden		Sweden		France	
Switzerland		Switzerland		Italy	
USA		USA		Portugal	

5 Results

In this section, we compare goodness of the forecasts obtained using multi-task NNs, MT1, MT2 and MT3 (both in the weighted and unweighed case), the single-task NNs DEEP1, DEEP2, DEEP3, DEEP4, DEEP5 and DEEP6, and 3 widely-used stochastic mortality models from the literature - the single population version of the LC model, see Lee and Carter (1992),

$$\ln(m_{x,t}^{(g,p)}) = \alpha_x^{(g,p)} + \beta_x^{(g,p)}\kappa_t^{(g,p)}, \tag{23}$$

the single population version of the CBD model[2], see Cairns et al. (2006),

$$\ln(m_{x,t}^{(g,p)}) = \kappa_t^{(1,g,p)} + \kappa_t^{(2,g,p)}(x - \bar{x}), \tag{24}$$

and a version of the ACF model, see Chen and Millossovich (2018), used for modelling simultaneously both gender and a set of different countries,

$$\ln(m_{x,t}^{(g,p)}) = \alpha_x^{(g,p)} + B_x K_t + \beta_x^{(g)} k_t^{(g)} + \beta_x^{(g,p)} k_t^{(g,p)}. \tag{25}$$

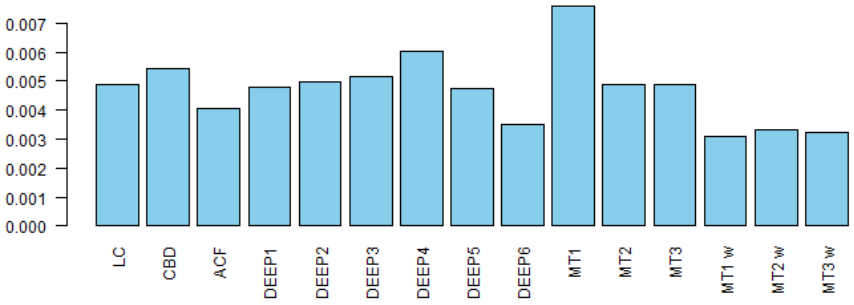
[2]CBD model is only tested in the 55-89 age range, for which it was designed.

Here $\alpha_x^{(g,p)}$, $\beta_x^{(g)}$, $\beta_x^{(g,p)}$, and B_x are age-dependent parameters, $\bar{x} = 72$, the average over the population age range 55-89, while $k_t^{(g)}$, $k_t^{(g,p)}$, $k_t^{(1,g,p)}$, $k_t^{(2,g,p)}$ and K_t are time-dependent stochastic factors. Here $k_t^{(g,p)}$ (in the LC model), $k_t^{(1,g,p)}$, $k_t^{(2,g,p)}$ and K_t are modelled as a random walk with drift, while $k_t^{(g)}$ and $k_t^{(g,p)}$ (in the ACF model) are modelled as an $AR(1)$. Notice that unlike the ACF model, the LC and CBD models treat different countries independently. Both the fitting and the forecasting of these three models are obtained using the StMoMo package, see Villegas et al. (2018).

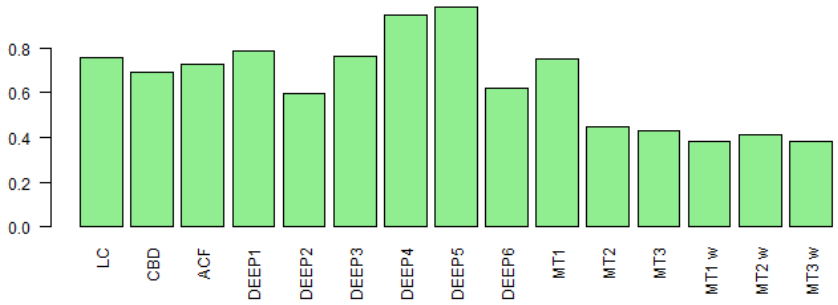
The comparison of models results is based on three metrics: mean absolute forecasting error (MAFE) for individual yearly death rates, for life expectancy and for standard deviation of the lifetime. Figures 6, 7, and 8 report the three metrics respectively for the 55-89, 20-89, and 0-89 age ranges while using 1950-1999 as training period and 2000-2019 as test period. Figures 10-18 extend this analysis by showing the three metrics for individual countries. Figure 9 shows the evolution of the three metrics using different lengths for the training period using 20-89 as age range. Finally, Table 3 summarises the total number of parameters in each approach for the three age ranges considered here.[3]

We observe that there is variability in the different approaches performance based on the metric and age range considered. In Figure 6, we notice how multi-task NNs show noticeable results in the 55-89 age range both with and without the weighting scheme. Indeed, they outperform all other approaches for life expectancy MAFE. Multi-task NNs with weighting scheme results appear to be the best ones for mortality rates MAFE, while in standard deviation MAFE they are outperformed only by Lee-Carter and ACF models.

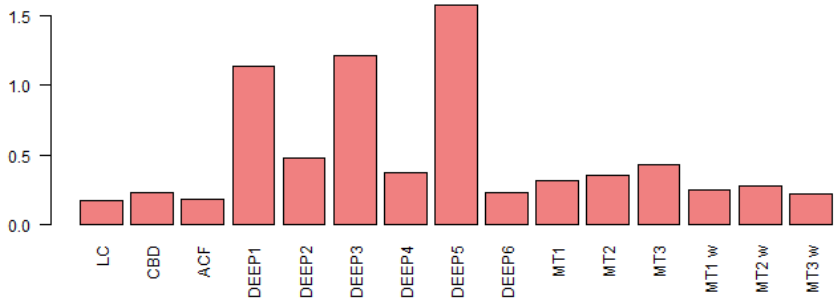
[3]Notice that MT1 w, MT2 w, and MT3 w have the same number of parameters of respectively MT1, MT2, and MT3.



(a) MAFE - Mortality Rates



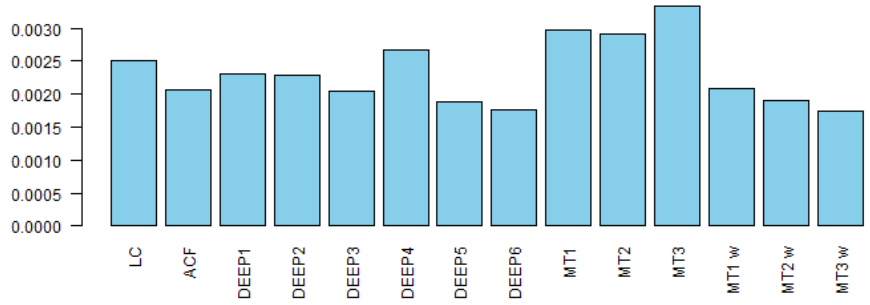
(b) MAFE - Life Expectancy



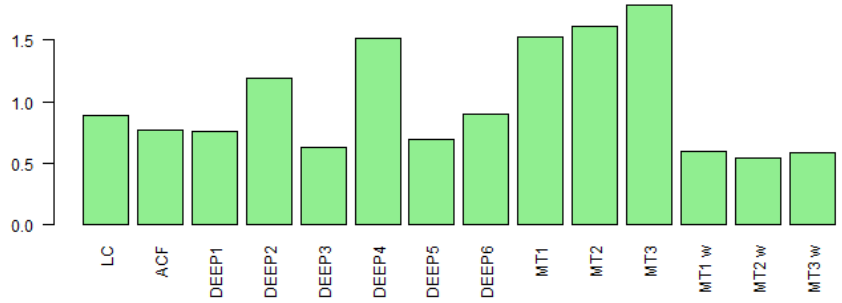
(c) MAFE - Standard Deviation

Figure 6: Comparison of MAFE metrics for Mortality Rates, Life Expectancy, and Standard Deviation. Age range: 55-89.

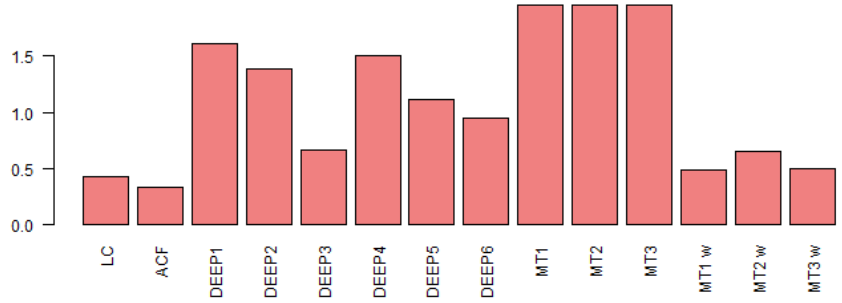
Focusing on 20-89 age range in Figure 7, we notice a big difference with respect to the 55-89 age range case. Firstly, multi-task NNs without a weighting scheme turn out to be the worst ones for all three the metrics considered here. In contrast, multi-task NNs with a weighting scheme still show good results. Indeed, they are among the best ones for mortality rate MAFE, alongside DEEP5 and DEEP6, for life expectancy MAFE, alongside DEEP3, and for standard deviation, where they are outperformed only by Lee-Carter and ACF models. The reason for such a big difference between the performance of multi-task NNs with and without weighting scheme, especially for life expectancy and standard deviation, is likely due to the training of the NNs. Indeed, mortality rates at lower ages are underestimated, due to their lower magnitude, during the training process which tends to place more emphasis on observations with higher magnitude, such as the ones at older ages. As a consequence, the NNs will produce poor forecast for lower ages mortality rates and, as a consequence, bigger errors for life expectancy and standard deviation which are heavily influenced by mortality at early ages.



(a) MAFE - Mortality Rates



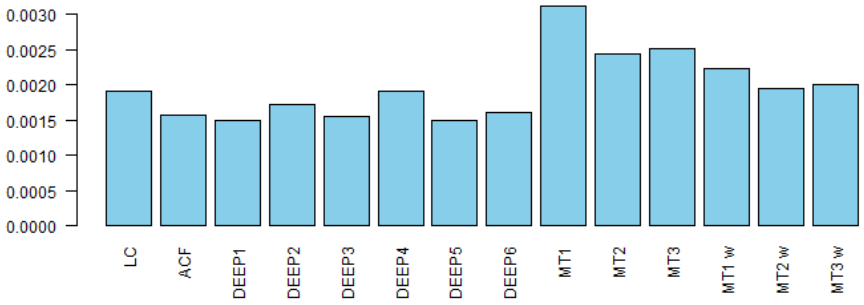
(b) MAFE - Life Expectancy



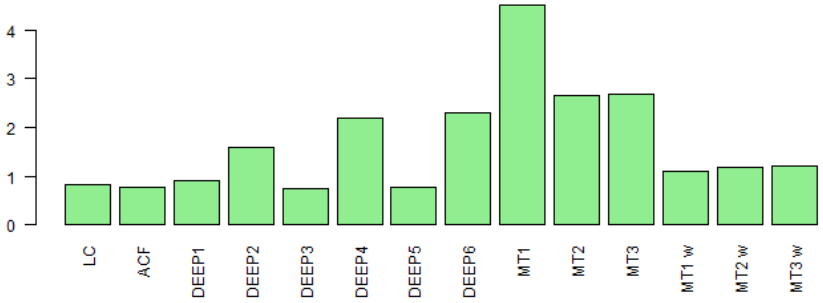
(c) MAFE - Standard Deviation

Figure 7: Comparison of MAFE metrics for Mortality Rates, Life Expectancy, and Standard Deviation. Age range: 20-89.

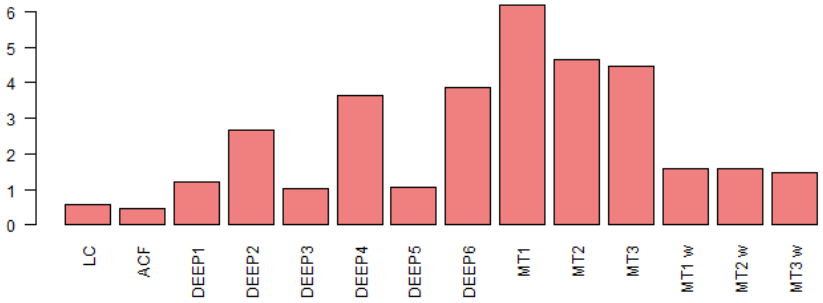
Finally, focusing on Figure 8, we observe a similar pattern also when considering the 0-89 age range. Indeed, the multi-task NNs trained without a weighting scheme result in the poorest performance for all the three metrics considered. Also multi-task NNs with a weighting scheme have a weaker performance compared to the other two cases considered previously. In fact, we notice how Lee-Carter and ACF models, and DEEP1, DEEP3, and DEEP5 single-task NNs have better performance compared to them. Nevertheless, the advantage of using a weighting scheme is still important for multi-task NNs, especially when considering life expectancy and standard deviation. Finally, here we can notice the benefit of clustering the third hidden layer in multi-task NNs (notice that the clustering is based on historical values of life expectancy and standard deviation in the age range 0-89). Indeed, both MT2 and MT3 show an improvement to MT1 in all the metrics considered here. We also observe that when introducing a weighting scheme in the training of the NNs this benefit tends to disappear.



(a) MAFE - Mortality Rates



(b) MAFE - Life Expectancy



(c) MAFE - Standard Deviation

Figure 8: Comparison of MAFE metrics for Mortality Rates, Life Expectancy, and Standard Deviation. Age range: 0-89.

In Figures 10-18, the MAFEs by country and approach are shown. Specifically, the filled black dot (and the corresponding vertical black line) represents the global MAFE (i.e. the same metric showed in Figures 6-8), while the coloured dots represent the MAFE for individual countries. Focusing on the US and Japan, i.e. two countries with particular pattern in the evolution of mortality in the last decades, we notice how the multi-task NNs without a weighting scheme provide good results, the MAFEs are lower than the average across countries, when the age range is large (20-89 an 0-89). On the other hand, notice how the best performing ST NNs tend to have a notably higher MAFE for the US when considering life expectancy and standard deviation. When introducing a weighting scheme in the training of the multi-task NNs, the benefits on the two countries tend to disappear, with their MAFEs being often above the global MAFE. Overall when considering all countries, we notice how in the 55-89 age range case, multi-task NNs tend to have lower dispersion across the countries' MAFEs compared to single-task NNs.

In Figure 9, the minimum MAFE by approach, training period (while keeping fixed the age range 20-89), and metric is reported. The minimum MAFEs for single-task NNs, multi-task NNs without weighting scheme, multi-task NNs with weighting scheme, and stochastic models are obtained as the minimum among DEEP1-DEEP6, MT1-MT3, MT1 w-MT3 w, and LC-ACF, respectively.
We can observe how for mortality rates MAFE, single-task NNs generally give the best results with only two exceptions. Although the gap with stochastic models narrows when considering shorter training periods, multi-task NNs performance appears to be less steady, and results to be the best one only in one case. Nevertheless, the benefit of using a weighting scheme is noticeable in all the training periods considered here.
Finally, the stochastic models outperform NNs in all cases considered while focusing on lifetime standard deviation. They always have the lowest minimum MAFE, despite the gap with NNs getting more narrow in the longest training period, showing a similar trend also found in the mortality rates MAFE. With regards to the NNs, single-task and multi-task alternate each other in terms of best performance with a consistent result that using a weighting scheme improves the forecasting accuracy of multi-task NNs.

6 Conclusion

The results show that using a 50-year training period, the performance of multi-task NNs compared to single-task NNs and traditional stochastic models depends on the metric considered and, especially, on the age range. More specifically, the out-of-

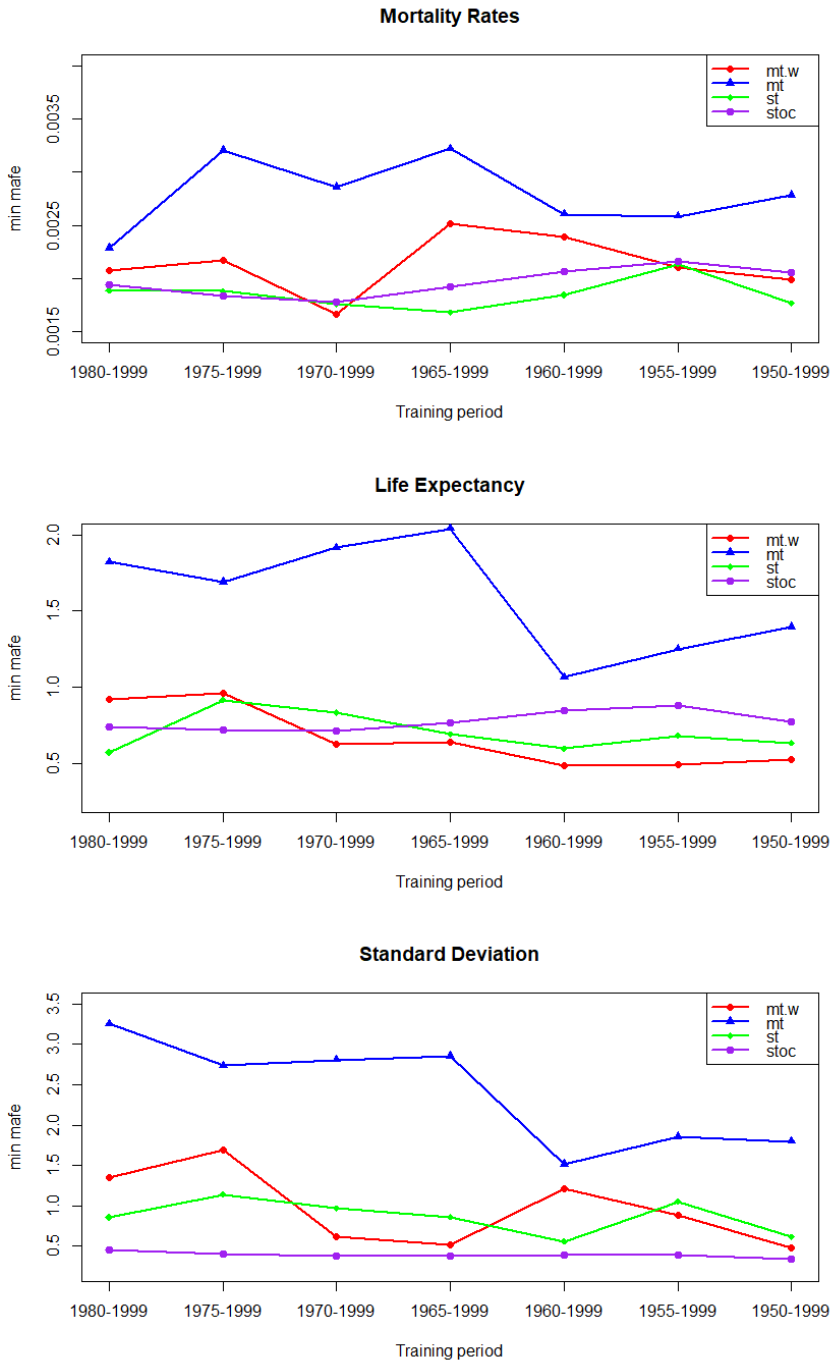


Figure 9: Minimum MAFE for single-task NNs, multi-task NNs and stochastic models by training period and metric considered.

Table 3: Number of parameters by approach and age range.

	0-89	20-89	55-89
LC model	7,820	6,460	4,080
CBD model	-	-	3,400
ACF model	8,240	6,820	4,335
DEEP1	20,386	20,286	20,111
DEEP2	20,386	20,286	20,111
DEEP3	71,458	71,358	71,183
DEEP4	71,458	71,358	71,183
DEEP5	73,506	73,406	73,231
DEEP6	73,506	73,406	73,231
MT1	108,354	106,654	103,679
MT2	116,866	115,166	112,191
MT3	125,378	123,678	120,703

sample precision of multi-task NNs is good with a shorter age range but tends to deteriorate when this age range is increased. This is likely due to the underestimation of lower ages mortality rates that happen in the training period. Adding a weighting scheme to multi-task NNs, markedly improves their performance, especially for life expectancy and standard deviation. Finally, it is noticeable that multi-task NNs with clustering based on past life expectancy and standard deviation show better results only when a weighting scheme is not considered.

When testing the models on shorter training periods, we arrive at a similar conclusion with respect to the 1950-1999 training period case. What is worthy to point out is that traditional stochastic models tend to perform relatively better compared to NNs when considering a short training period.

In terms of future research on multi-task NNs, at least five straightforward developments could be considered: firstly, implementing multi-task NNs where the generic task is based on a categorical variable such as age and gender, rather than, or alongside, country. Secondly, using a different machine learning technique, such as one specifically designed for time series, to cluster the countries based on past mortality experience. Thirdly, a penalization could be added to the loss function of the NNs to ensure that female and male mortality do not diverge, or even that the mortality of different populations does not diverge, achieving some degree of coherence. Fourthly, forecasting mortality rates in a different context, such as by cause of death within a single population. Fifthly, further analysis aiming to study the question of explainability of multi-task NNs could be conducted, see Perla et al. (2024).

References

Cairns, A. J., Blake, D., and Dowd, K. (2006). A two-factor model for stochastic mortality with parameter uncertainty: theory and calibration. *Journal of Risk and Insurance*, 73(4):687–718.

Carracedo, P., Debón, A., Iftimi, A., and Montes, F. (2018). Detecting spatio-temporal mortality clusters of european countries by sex and age. *International Journal for Equity in Health*, 17(1):1–19.

Chen, R. Y. and Millossovich, P. (2018). Sex-specific mortality forecasting for UK countries: a coherent approach. *European Actuarial Journal*, 8:69–95.

Chen, Y. and Khaliq, A. Q. (2022). Comparative study of mortality rate prediction using data-driven recurrent neural networks and the Lee–Carter model. *Big Data and Cognitive Computing*, 6(4):134.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Crawshaw, M. (2020). Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*.

Danesi, I. L., Haberman, S., and Millossovich, P. (2015). Forecasting mortality in subpopulations using Lee–Carter type models: A comparison. *Insurance: Mathematics and Economics*, 62:151–161.

Deng, L., Hinton, G., and Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An overview. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8599–8603. IEEE.

Dickson, D. C., Hardy, M. R., and Waters, H. R. (2019). *Actuarial mathematics for life contingent risks*. Cambridge University Press.

Dubey, S. R., Singh, S. K., and Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 503:92–108.

Euthum, M., Scherer, M., and Ungolo, F. (2024). A neural network approach for the mortality analysis of multiple populations: a case study on data of the italian population. *European Actuarial Journal*, pages 1–30.

Girshick, R. (2015). Fast R-CNN in proceedings of the IEEE International Conference on Computer Vision (pp. 1440–1448). *Piscataway, NJ: IEEE.*, 2.

Goldberg, Y. (2017). *Neural network methods for natural language processing*. Morgan & Claypool.

Hainaut, D. (2018). A neural-network analyzer for mortality forecast. *ASTIN Bulletin: The Journal of the IAA*, 48(2):481–508.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lee, R. D. and Carter, L. R. (1992). Modeling and forecasting US mortality. *Journal of the American Statistical Association*, 87(419):659–671.

Lindholm, M. and Palmborg, L. (2022). Efficient use of data for LSTM mortality forecasting. *European Actuarial Journal*, 12(2):749–778.

Lindholm, M., Richman, R., Tsanakas, A., and Wüthrich, M. V. (2023). A multi-task network approach for calculating discrimination-free insurance prices. *European Actuarial Journal*, pages 1–41.

Menet, N., Hersche, M., Karunaratne, G., Benini, L., Sebastian, A., and Rahimi, A. (2023). Mimonets: Multiple-input-multiple-output neural networks exploiting computation in superposition. *Advances in Neural Information Processing Systems*, 36:39553–39565.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

Nandini, S. and Sanjjushri, V. R. (2023). Estimating countries with similar maternal mortality rate using cluster analysis and pairing countries with identical MMR. *arXiv preprint arXiv:2312.04275*.

Nigri, A., Levantesi, S., Marino, M., Scognamiglio, S., and Perla, F. (2019). A deep learning integrated Lee–Carter model. *Risks*, 7(1):33.

Perla, F., Richman, R., Scognamiglio, S., and Wüthrich, M. V. (2021). Time-series forecasting of mortality rates using deep learning. *Scandinavian Actuarial Journal*, 2021(7):572–598.

Perla, F., Richman, R., Scognamiglio, S., and Wüthrich, M. V. (2024). Accurate and explainable mortality forecasting with the localglmnet. *Scandinavian Actuarial Journal*, pages 1–23.

Perla, F. and Scognamiglio, S. (2023). Locally-coherent multi-population mortality modelling via neural networks. *Decisions in Economics and Finance*, 46(1):157–176.

Prince, S. J. (2023). *Understanding Deep Learning*. The MIT Press.

Richman, R. (2022). Mind the gap–safely incorporating deep learning models into the actuarial toolkit. *British Actuarial Journal*, 27:e21.

Richman, R. and Wüthrich, M. V. (2021). A neural network extension of the Lee–Carter model to multiple populations. *Annals of Actuarial Science*, 15(2):346–366.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117.

Schnürch, S. and Korn, R. (2022). Point and interval forecasts of death rates using neural networks. *ASTIN Bulletin: The Journal of the IAA*, 52(1):333–360.

Scitovski, R., Sabo, K., Martínez-Álvarez, F., and Ungar, Š. (2021). *Cluster analysis and applications*. Springer.

Scognamiglio, S. (2022). Calibrating the Lee-Carter and the Poisson Lee-Carter models via neural networks. *ASTIN Bulletin: The Journal of the IAA*, 52(2):519–561.

Villegas, A. M., Kaishev, V. K., and Millossovich, P. (2018). Stmomo: An R package for stochastic mortality modeling. *Journal of Statistical Software*, 84:1–38.

Wang, C.-W., Zhang, J., and Zhu, W. (2021). Neighbouring prediction for mortality. *ASTIN Bulletin: The Journal of the IAA*, 51(3):689–718.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*.

A Results by country

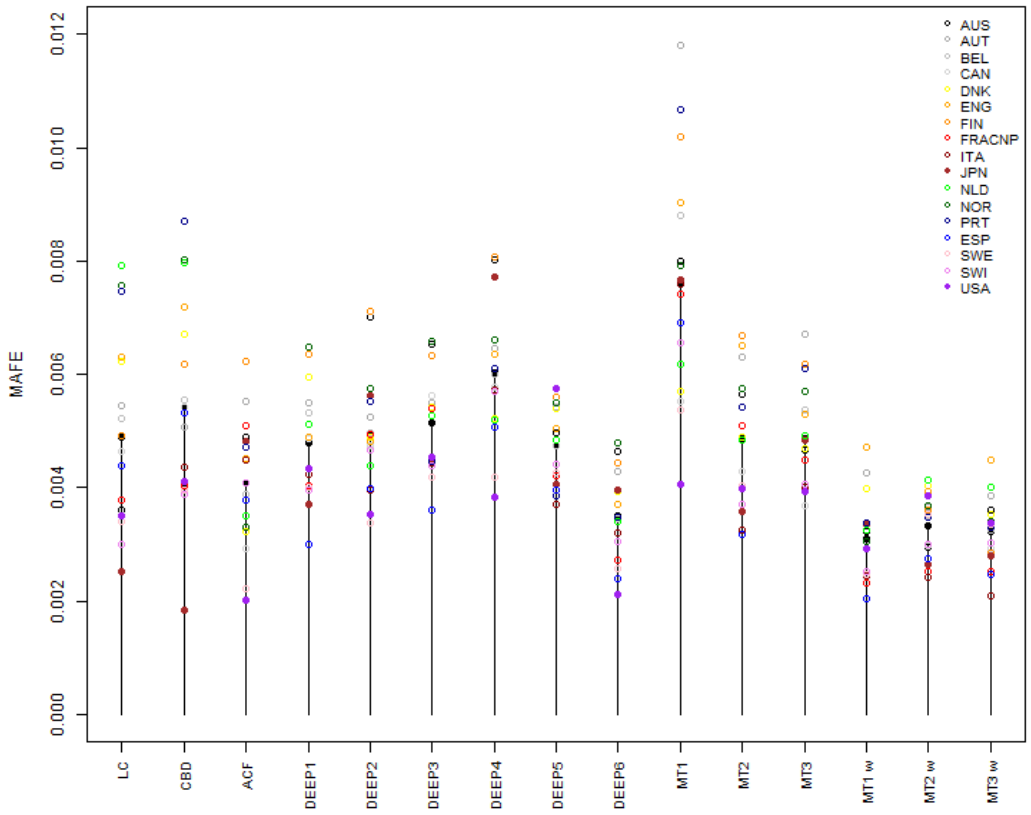


Figure 10: Mean absolute forecasting error by country and approach. Metric considered: mortality rate. Age range: 55-89.

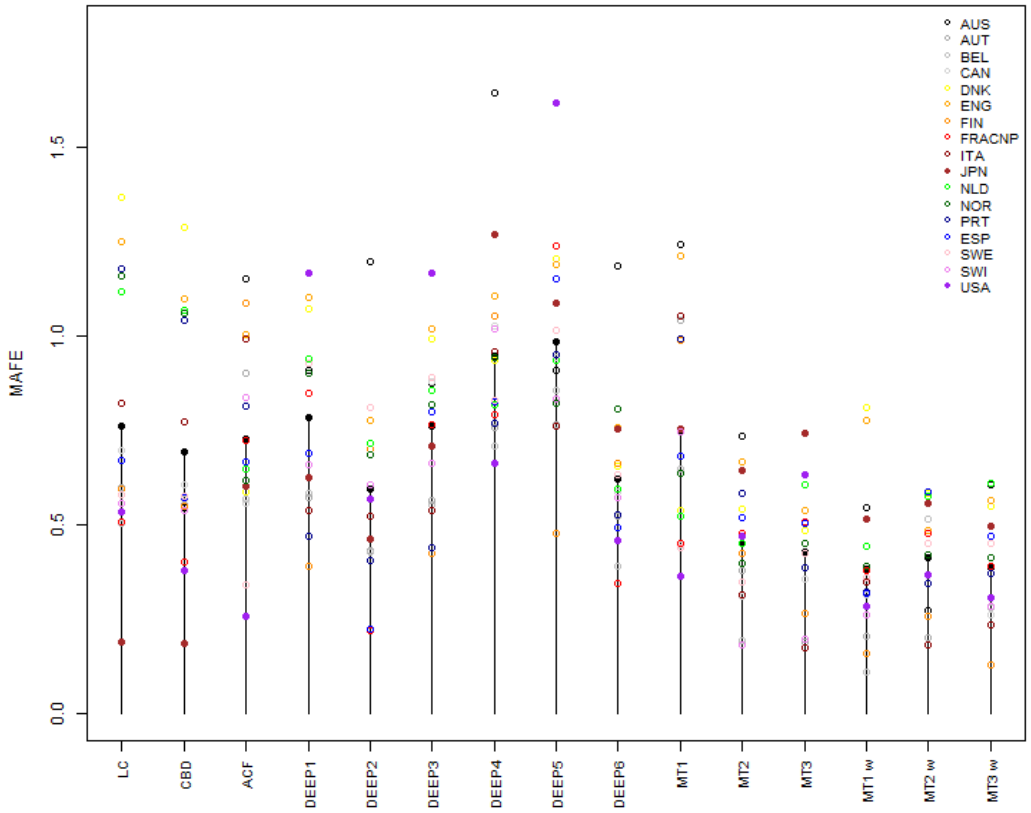


Figure 11: Mean absolute forecasting error by country and approach. Metric considered: life expectancy. Age range: 55-89.

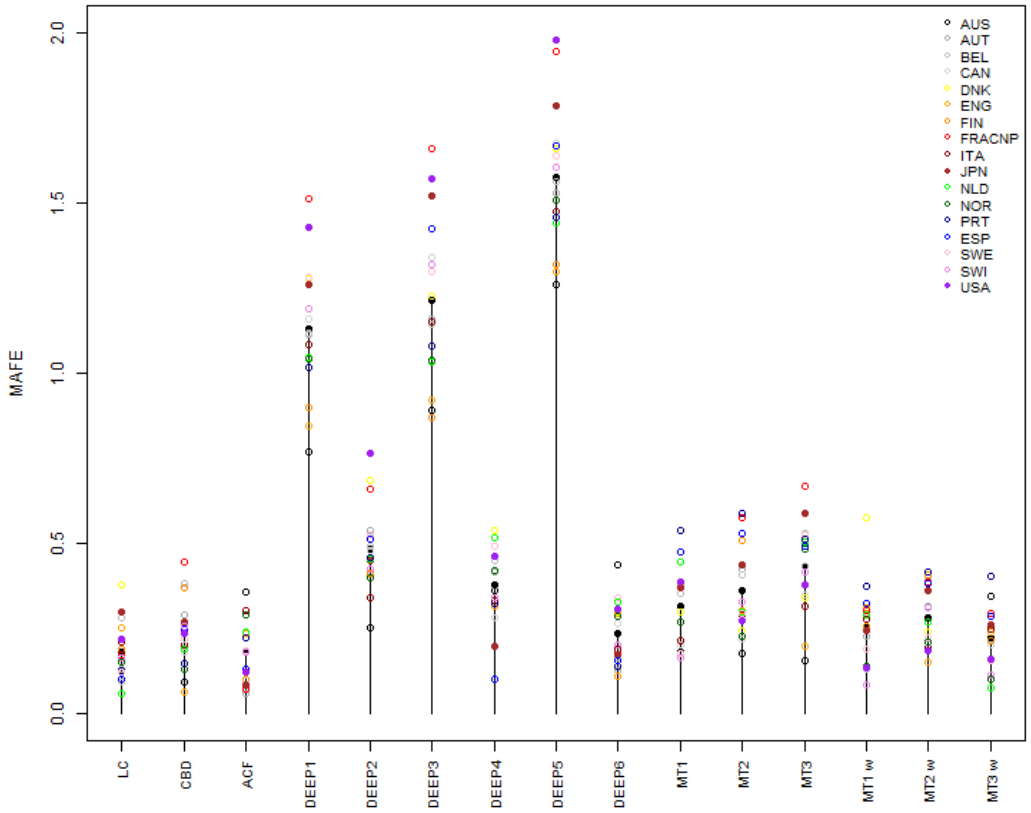


Figure 12: Mean absolute forecasting error by country and approach. Metric considered: standard deviation. Age range: 55-89.

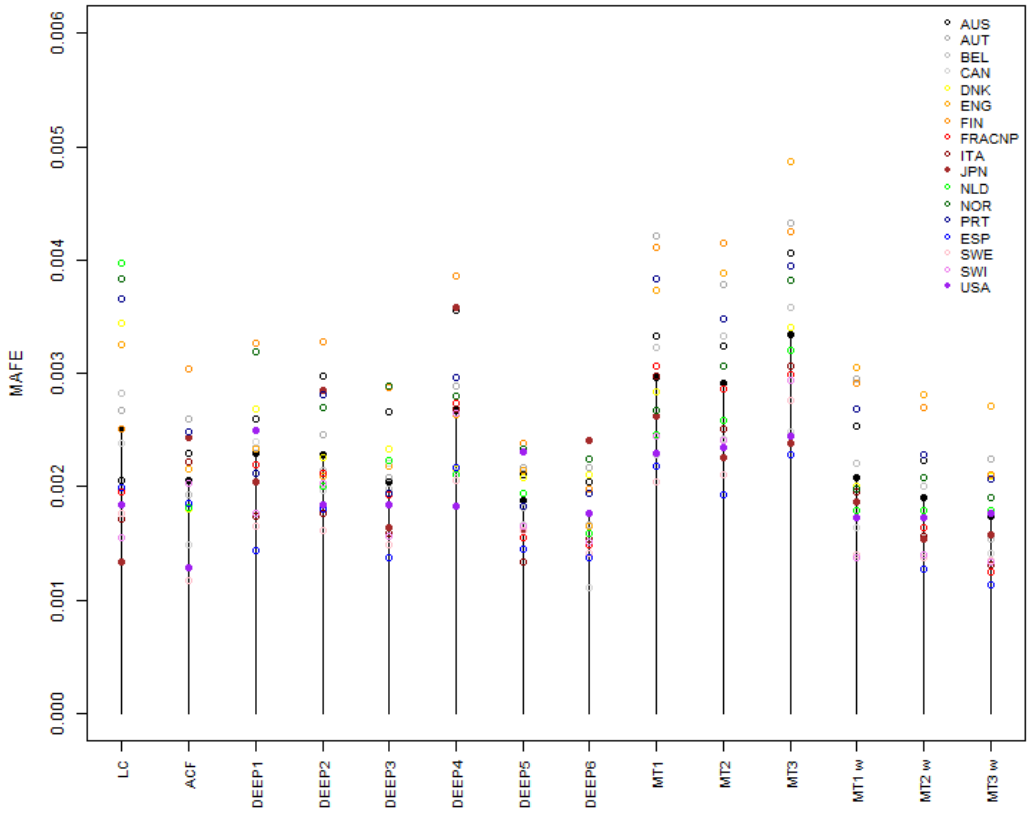


Figure 13: Mean absolute forecasting error by country and approach. Metric considered: mortality rate. Age range: 20-89.

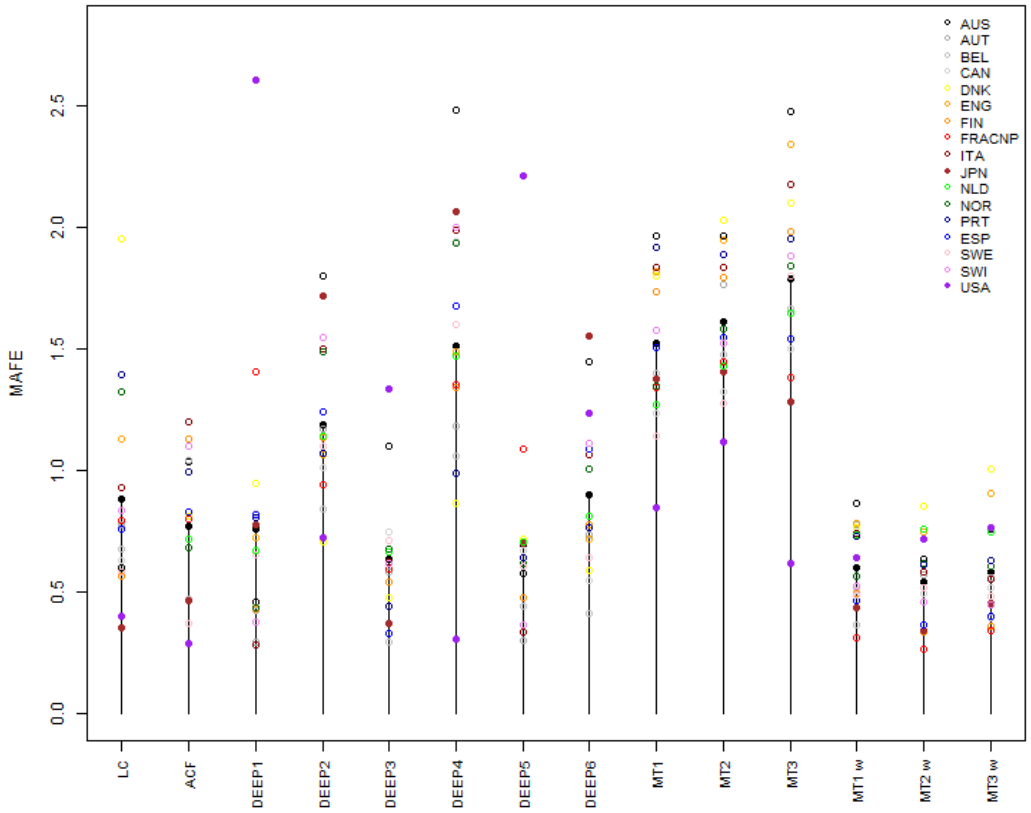


Figure 14: Mean absolute forecasting error by country and approach. Metric considered: life expectancy. Age range: 20-89.

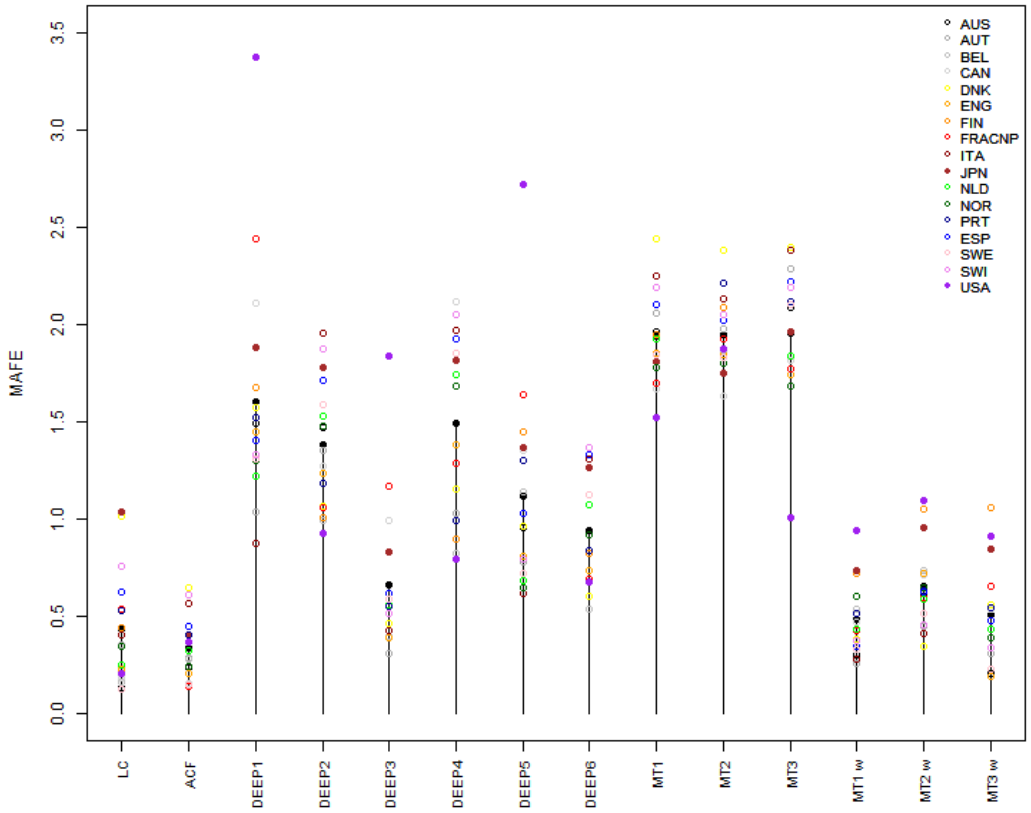


Figure 15: Mean absolute forecasting error by country and approach. Metric considered: standard deviation. Age range: 20-89.

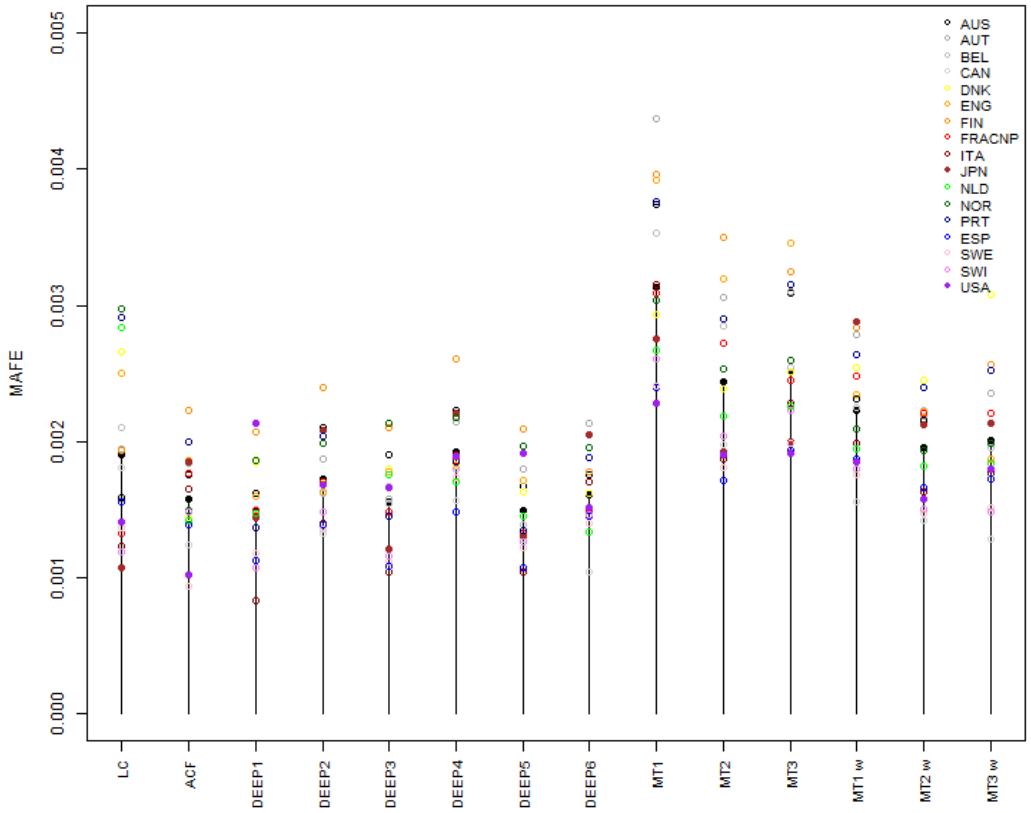


Figure 16: Mean absolute forecasting error by country and approach. Metric considered: mortality rate. Age range: 0-89.

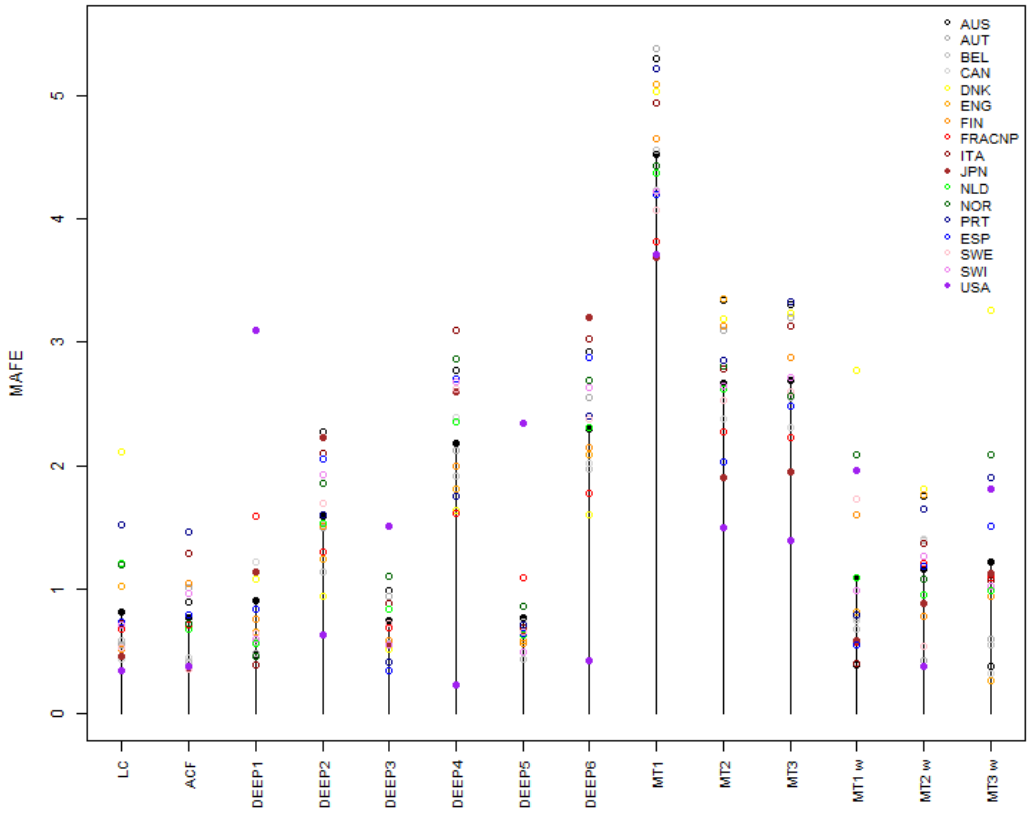


Figure 17: Mean absolute forecasting error by country and approach. Metric considered: life expectancy. Age range: 0-89.

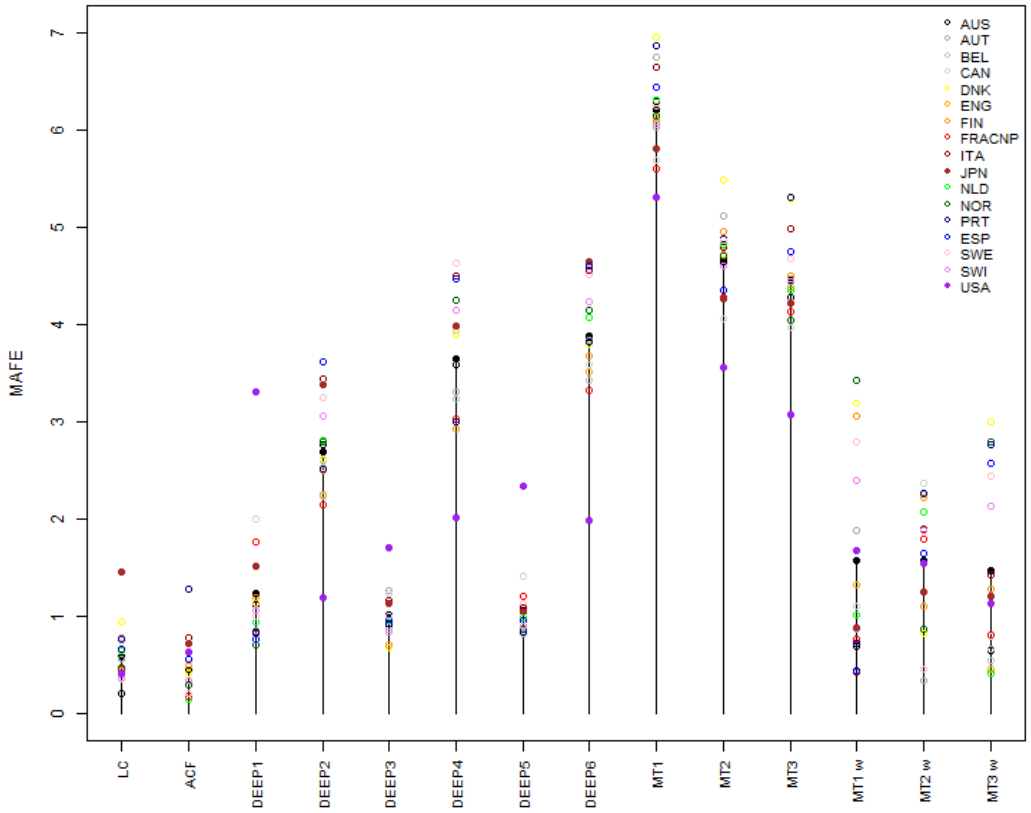


Figure 18: Mean absolute forecasting error by country and approach. Metric considered: standard deviation. Age range: 0-89.