

# Il Transfer Learning come modello relazionale di mortalità per la costruzione di tavole di mortalità entity-specific

Mario Marino

Dipartimento di Scienze Economiche, Aziendali, Matematiche e Statistiche “Bruno de Finetti”  
Università di Trieste

Seminario del Comitato Scientifico dell’Ordine degli Attuari

17 giugno 2026

# Introduzione

## Topic

Costruzione di basi tecnico-demografiche per la valutazione di passività *life-contingent*

## Focus

Modellazione della mortalità di uno specifico collettivo, ad es. popolazione assicurata o portafoglio (*entity-specific mortality*)

↔ Possibili difficoltà metodologiche dovute a

- ridotta dimensione del collettivo
- insufficiente profondità storica dell'esperienza di mortalità
- limitata credibilità statistica dell'esperienza osservata

# Introduzione

## Soluzioni comuni nella prassi

- 1 Utilizzo diretto di tavole di mortalità relative alla popolazione nazionale o a livello di mercato  $\rightsquigarrow$  **Basis risk**
- 2 Utilizzo di modelli relazionali di mortalità  $\rightsquigarrow$  **Adeguatezza relazione/aggiustamento**

## Generalizzazione proposta nel seguito

Utilizzo del *Transfer Learning* (TL) per la generalizzazione dei modelli relazionali di mortalità

- Modelli relazionali come caso particolare
- Utilizzo di molteplici informazioni esterne
- Aggiornamento dinamico delle assunzioni demografiche

# Introduzione

## Analisi, risultati e takeaways del progetto di ricerca

*Dynamic pricing of biometric risks in new digital ecosystems*

**Team:** M. Marino (P.I.), F.V. Riccio, G.A. Spedicato

## Agenda

- I. I modelli relazionali di mortalità
- II. Il Transfer Learning (TL)
- III. Applicazione attuariale: il TL come generalizzazione dei modelli relazionali

# I. Modelli relazionali di mortalità

## Scopo di un modello relazionale di mortalità

Modellare la mortalità di una popolazione target sfruttando l'esperienza di mortalità di una popolazione di riferimento

- Esperienza di **mortalità di una popolazione di riferimento**
  - ▷ *Popolazione di riferimento*: nazionale o specifica popolazione di mercato (ad es. percettori di rendita)
  - ▷ *Esperienza di mortalità*: misurazione quantitativa fornita da una variabile demografica d'interesse  $\rightsquigarrow$  nel seguito: forza di mortalità,  $\mu$
- Adozione di un **modello  $f$  che "metta in relazione"** la mortalità di riferimento con quella del collettivo target (ad es. del portafoglio)

$$\underbrace{\text{mortalità d'interesse}}_{\mu} = f\left(\underbrace{\text{mortalità popolazione di riferimento}}_{\mu^{\text{ref}}}\right)$$

# I. Modelli relazionali di mortalità

- **Forma generale di un modello relazionale di mortalità** (Delwarde et al. (2004)):

$$\ln \mu_a = f \left( \ln \mu_a^{\text{ref}} \right), \quad (1)$$

con  $a = a_0, a_1, \dots, a_{\max}$  l'età ed  $f : \mathbb{R} \rightarrow \mathbb{R}$  un'opportuna funzione regolare

- E.g., scegliendo  $f(c) = \alpha + \beta c$ , si ha il noto **modello relazionale di Brass**

$$\ln \mu_a = \alpha + \beta \ln \mu_a^{\text{ref}}, \quad \alpha, \beta \in \mathbb{R}, \quad (2)$$

o equivalentemente

$$\mu_a = e^\alpha \left( \mu_a^{\text{ref}} \right)^\beta. \quad (3)$$

# I. Modelli relazionali di mortalità

- Il **modello di Brass** può esser inglobato all'interno del modello di regressione di Poisson:

$$D_a \sim \text{Poisson}(E_a^c \mu_a), \text{ con } \ln \mu_a = \alpha + \beta \ln \mu_a^{\text{ref}},$$

con  $D_a$  il numero aleatorio di decessi ad età  $a$ , ed  $E_a^c$  il numero centrale di esposti al rischio ad età  $a$ .

- Più in generale, i modelli relazionali di mortalità possono esser estesi per incorporare le **caratteristiche della popolazione target** (oltre alla sola età)

▷ Dati popolazione target:  $\mathcal{S} = \{(\mathbf{x}_i, E_i^c, d_i), i = 1, \dots, n\}$

★  $\mathbf{x}_i = (1, x_{i,1}, \dots, x_{i,q}) \in \mathcal{X}$  caratteristiche a livello individuale o di classe di rischio

★  $d_i \in \mathbb{N}_0$  numero decessi osservati;  $E_i^c \in \mathbb{R}_+$  esposti al rischio

# I. Modelli relazionali di mortalità

- Assunzione:  $D_i \sim \text{Poisson}(E_i^c \mu_i)$
- Funzione di regressione,  $\lambda : \mathbb{R}_+ \times \mathcal{X} \rightarrow \mathbb{R}_+$

$$(E_i^c, \mathbf{x}_i) \mapsto \ln \lambda(E_i^c, \mathbf{x}_i) = \ln E_i^c + \langle \boldsymbol{\beta}, \mathbf{x}_i \rangle$$

dove  $\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle := \sum_{j=0}^q \beta_j x_{i,j}$ .

↪ Esempio modello di Brass con  $\mathbf{x}_i = (1, a_i, \text{Smk}_i, \ln \mu_{a_i}^{\text{ref}})$

$$\ln \lambda(E_i^c, \mathbf{x}_i) = \ln E_i^c + \beta_0 + \beta_1 a_i + \beta_2 \text{Smk}_i + \beta_3 \ln \mu_{a_i}^{\text{ref}}$$

## Nota:

Se la popolazione target riguarda un portafoglio assicurativo, allora è necessario tener conto dell' "**ecceso di zeri**" relativo al numero di decessi osservati in  $\mathcal{S}$ .

# I. Modelli relazionali di mortalità

- **Assunzione: Zero-Inflated Poisson (ZIP)**  $\rightsquigarrow D_i \sim \text{ZIP}(\pi_i, E_i^c \mu_i)$

$$D_i = \begin{cases} 0 & \sim \pi_i \\ \text{Poisson}(E_i^c \mu_i) & \sim 1 - \pi_i \end{cases}, \quad i = 1, \dots, n.$$

- **Caratteristiche della popolazione target:**  $\mathbf{x}_i = (1, x_{i,1}, \dots, x_{i,q}) \in \mathcal{X}$  e  $\mathbf{z}_i = (1, z_{i,1}, \dots, z_{i,s}) \in \mathcal{Z}$
- **Due distinte funzioni di regressione,**  $\lambda_\mu$  e  $\lambda_\pi$

$$\lambda_\mu : \mathbb{R}_+ \times \mathcal{X} \rightarrow \mathbb{R}_+, \quad (E_i^c, \mathbf{x}_i) \mapsto \ln \lambda_\mu(E_i^c, \mathbf{x}_i) = \ln E_i^c + \langle \boldsymbol{\beta}_\mu, \mathbf{x}_i \rangle$$

$$\lambda_\pi : \mathcal{Z} \rightarrow [0, 1], \quad \mathbf{z}_i \mapsto \text{logit } \lambda_\pi(\mathbf{z}_i) = \langle \boldsymbol{\beta}_\pi, \mathbf{z}_i \rangle$$

# Dai modelli relazionali al Transfer Learning

- L'operatività di un modello relazionale, dunque, si sostanzia nel:
  - ① Considerare un'informazione di mortalità da trasferire,  $\mu^{\text{ref}}$
  - ② Adottare un modello di trasferimento, e.g. modello di Brass, regressione di Poisson o ZIP
  - ③ Predire la mortalità della popolazione target
- Tale operatività richiama il più generale obiettivo modellistico del Transfer Learning

## Generalizzazione del modello relazionale: il Transfer Learning

Il **Transfer Learning** è una famiglia di tecniche di apprendimento aventi lo scopo di migliorare il compito predittivo su un dominio target sfruttando la conoscenza appresa su un altro dominio, detto sorgente.

## II. Transfer Learning

- Concetti chiave: **source vs target** e **domain vs task**
- Source domain e source task, rispettivamente:

$$\mathcal{D}_S = \{\mathcal{X}_S, \mathbb{P}(X_S)\}, \quad \mathcal{T}_S = \{\mathcal{Y}_S, f_S(\cdot)\}$$

↔ Funzione predittiva del source,  $f_S : \mathcal{X}_S \rightarrow \mathcal{Y}_S$ ,  $\mathbf{x}_S \mapsto f_S(\mathbf{x}_S)$

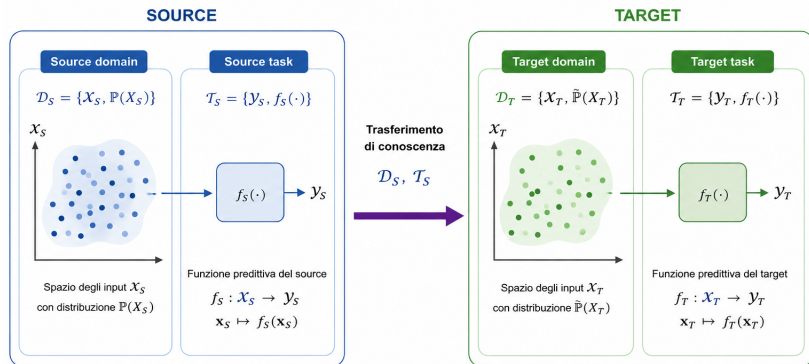
- Target domain e Target task, rispettivamente:

$$\mathcal{D}_T = \{\mathcal{X}_T, \tilde{\mathbb{P}}(X_T)\}, \quad \mathcal{T}_T = \{\mathcal{Y}_T, f_T(\cdot)\}$$

↔ Funzione predittiva del target,  $f_T : \mathcal{X}_T \rightarrow \mathcal{Y}_T$ ,  $\mathbf{x}_T \mapsto f_T(\mathbf{x}_T)$

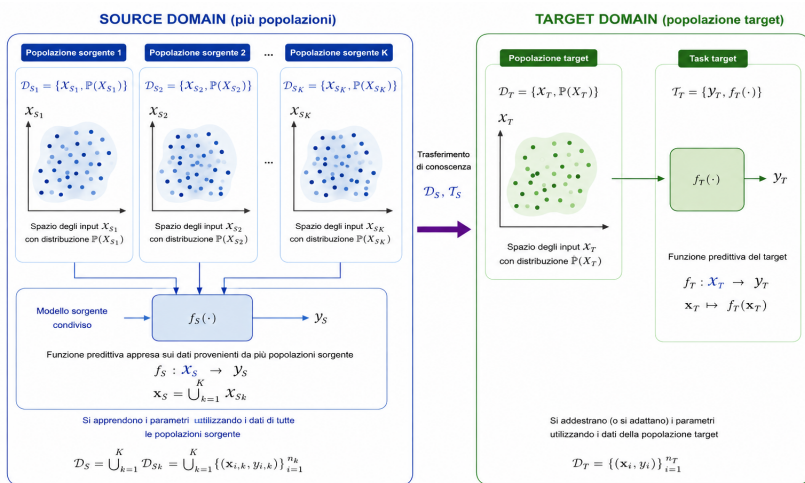
**Il TL mira a migliorare  $f_T(\cdot)$  sfruttando quanto appreso da  $\mathcal{D}_S$  e  $\mathcal{T}_S$ , con  $\mathcal{D}_S \neq \mathcal{D}_T$  e/o  $\mathcal{T}_S \neq \mathcal{T}_T$**

## II. Transfer Learning



- Cosa trasferire?  $\rightsquigarrow$  Categorie di TL
- Come trasferire?  $\rightsquigarrow$  Strategie di TL

## II. Transfer Learning



## II. Transfer Learning

- **Tre categorie principali di TL** (Yosinski et al. (2014)):
  - ▷ Inductive TL:  $\mathcal{D}_S = \mathcal{D}_T \wedge \mathcal{T}_S \neq \mathcal{T}_T$
  - ▷ Transductive TL:  $\mathcal{D}_S \neq \mathcal{D}_T \wedge \mathcal{T}_S = \mathcal{T}_T$
  - ▷ Unsupervised TL:  $\mathcal{D}_S \neq \mathcal{D}_T \wedge \mathcal{T}_S \neq \mathcal{T}_T$
- **Strategie di TL:**
  - ▷ Instance-based
  - ▷ Feature-based
  - ▷ Parameter-based
  - ▷ Relation-based

Di seguito, faremo riferimento al **parameter-based TL utilizzando le feed-forward neural networks (FFN)** (si veda, e.g., Weiss et al. (2016)).

## II. Transfer Learning

**Step 1:** Modellare i dati del source  $D_S = \{(\mathbf{x}_s, \mathbf{y}_s), \mathbf{x}_s \in \mathcal{X}_S, \mathbf{y}_s \in \mathcal{Y}_S\}$

$$\lambda_S : \mathcal{X}_S \rightarrow \mathcal{Y}_S, \quad \mathbf{x}_S \mapsto \lambda_S(\mathbf{x}_S) = \left( \psi_S \circ h_S^{(K_S:1)} \right) (\mathbf{x}_S)$$

dove  $K_S \in \mathbb{N}$ ,  $\psi_S : \mathbb{R}^{q_{K_S}} \rightarrow \mathbb{R}^{n_S}$  e

- $h_S^{(K_S:1)} = \left( h_S^{(K_S)} \circ h_S^{(K_S-1)} \circ \dots \circ h_S^{(1)} \right) (\mathbf{x}_S)$
- $\kappa$ -esimo hidden layer,  $h_S^{(\kappa_S)}(\mathbf{h}) = \left( h_{1,S}^{(\kappa_S)}(\mathbf{h}), h_{2,S}^{(\kappa_S)}(\mathbf{h}), \dots, h_{q_{\kappa_S,S}}^{(\kappa_S)}(\mathbf{h}) \right)$
- $j$ -esimo neurone nel  $\kappa$ -esimo hidden layer, per  $j = 1, \dots, q_{\kappa_S}$ ,

$$h_{j,S}^{(\kappa_S)}(\mathbf{h}) = \phi^{(\kappa_S)} \left( w_{j,0,S}^{(\kappa_S)} + \sum_{\ell=1}^{q_{\kappa_S-1}} w_{j,\ell,S}^{(\kappa_S)} h_\ell \right) = \phi^{(\kappa_S)} \left( \left\langle \mathbf{w}_{j,S}^{(\kappa_S)}, \mathbf{h} \right\rangle \right)$$

## II. Transfer Learning

↪ A seguito del training, i pesi della FFN del source sono:

$$\hat{\theta}_S = \left( \hat{w}^{(1)}, \dots, \hat{w}^{(m)}, \hat{w}^{(m+1)}, \dots, \hat{w}^{(K_S)} \right)$$

dove  $\hat{w}^{(\kappa_S)} = \left( w_{j,S}^{(\kappa_S)} \right)_{1 \leq j \leq \kappa_S}$

**Step 2:** Modellare i dati target,  $D_T = \{(\mathbf{x}_T, \mathbf{y}_T), \mathbf{x}_T \in \mathcal{X}_T, \mathbf{y}_T \in \mathcal{Y}_T\}$ , usando i pesi della FFN del source

$$\lambda_T : \mathcal{X}_T \rightarrow \mathcal{Y}_T, \quad \mathbf{x}_T \mapsto \lambda_T(\mathbf{x}_T; \hat{\theta}_S^{\text{TL}}) = \left( \psi_T \circ h_T^{(K_T:1)} \right) (\mathbf{x}_T; \hat{\theta}_S^{\text{TL}})$$

$K_T \in \mathbb{N}$ ,  $\psi_T : \mathbb{R} \rightarrow \mathbb{R}$  and  $h_T^{(K_T:1)} = \left( h_T^{(K_T)} \circ \dots \circ h_T^{(1)} \right) (\mathbf{x}_T; \hat{\theta}_S^{\text{TL}})$

↪  $\hat{\theta}_S^{\text{TL}}$  vettore dei pesi trasferiti da  $\hat{\theta}_S$

## II. Transfer Learning

↪ La funzione  $h_T^{(K_T:1)}$  può esser riscritta come

$$h_T^{(K_T:1)} = \left( h_T^{(K_T:(m+1))} \circ h_T^{(m:1)} \right) (\mathbf{x}_T; \hat{\boldsymbol{\theta}}_S^{\text{TL}})$$

- $h_T^{(m:1)} = \left( h_T^{(m)} \circ \dots \circ h_T^{(1)} \right) (\mathbf{x}_T; \hat{\boldsymbol{\theta}}_S^{\text{TL}})$  è la **transfer function**
- Essa contiene la conoscenza trasferita, ossia  $\hat{\boldsymbol{\theta}}_S^{\text{TL}}$
- E.g., il  $j$ -esimo neurone nel  $\kappa$ -esimo hidden layer in  $h_T^{(m:1)}$ , per  $j = 1, \dots, q_{\kappa_T}$  e  $1 \leq \kappa_T \leq m$

$$h_{j,T}^{(\kappa_T)}(\mathbf{h}) = \phi^{(\kappa_T)} \left( \hat{w}_{j,0,S}^{(\kappa_T)} + \sum_{\ell=1}^{q_{\kappa_T}} \hat{w}_{j,\ell,S}^{(\kappa_T)} h_\ell \right) = \phi^{(\kappa_T)} \left( \left\langle \hat{\mathbf{w}}_{j,S}^{(\kappa_T)}, \mathbf{h} \right\rangle \right)$$

## II. Transfer Learning

**Come sfruttare i pesi  $\hat{\theta}_S^{\text{TL}}$  durante l'addestramento della FFN sul target:**

- Strategia "dell'inizializzazione" (*initialization approach*)
  - ▷  $\hat{\theta}_S^{\text{TL}}$  sono usati come valore di inizializzazione dei pesi  $\theta_T$  della FFN sul target
    - ↪ Il TL agisce solo sul primo passo forward del processo di apprendimento
- Strategia "del congelamento" (*freezing approach*)
  - ▷  $\hat{\theta}_S^{\text{TL}}$  sono considerati come parametri non addestrabili

## II. Transfer Learning

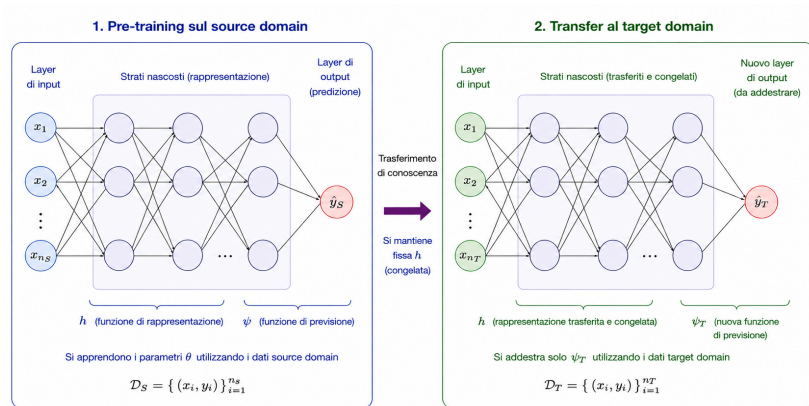


Figure: Rappresentazione grafica semplificata del parameter-based TL tramite il freezing approach.

### III. TL come generalizzazione dei modelli relazionali

Come generalizzare i modelli relazionali di mortalità:

- **Informazione di mortalità trasferita:** (ref = popolazione del dominio sorgente)
  - ▷ Senza TL:  $\mu^{\text{ref}}$
  - ▷ Con TL: parametri e funzione predittiva di  $\mu^{\text{ref}}$
- **Modello di trasferimento:**
  - ▷ Senza TL: stabilire a priori il modello e le relative assunzioni (Brass-like, Poisson GLM, ZIP, etc.)
  - ▷ Con TL: utilizzo di una pre-addestrata funzione di regressione, potendo rappresentare come casi particolari gli usuali modelli relazionali (Brass-like, Poisson GLM, ZIP, etc.)

### III. TL come generalizzazione dei modelli relazionali

- Dati popolazione source:  $D_S = \{(\mathbf{x}_{i,S}, d_{i,S}), i = 1, \dots, n_S\}$
- Dati popolazione target:  $D_T = \{(\mathbf{x}_{i,T}, d_{i,T}), i = 1, \dots, n_T\}$
- Sia  $E_{i,S}^c$  che  $E_{i,T}^c$  come variabile esplicativa
- **Zero-Inflated Poisson (ZIP)** per modellare i decessi sia per il dominio sorgente che per il dominio target

$$D_{i,\star} \sim \text{ZIP}(\lambda_{\pi,\star}(\mathbf{x}_{i,\star}), \lambda_{\mu,\star}(\mathbf{x}_{i,\star})), \quad \star = S, T$$

- Funzione di regressione FFN per i dati della popolazione source

$$\lambda_{\mu,S} : \mathcal{X}_S \rightarrow \mathbb{R}_+, \quad \mathbf{x}_S \mapsto \ln \lambda_{\mu,S}(\mathbf{x}_S) = \left\langle \mathbf{w}_{\mu,S}^{(K_S+1)}, h_{\mu,S}^{(K_S)}(\mathbf{x}_S) \right\rangle$$

$$\lambda_{\pi,S} : \mathcal{X}_S \rightarrow [0, 1], \quad \mathbf{x}_S \mapsto \text{logit } \lambda_{\pi,S}(\mathbf{x}_S) = \left\langle \mathbf{w}_{\pi,S}^{(K_S+1)}, h_{\pi,S}^{(K_S)}(\mathbf{x}_S) \right\rangle$$

### III. TL come generalizzazione dei modelli relazionali

- Funzione di regressione FFN per i dati della popolazione target

$$\lambda_{\mu,T} : \mathcal{X}_T \rightarrow \mathbb{R}_+, \quad \mathbf{x}_T \mapsto \ln \lambda_{\mu,T}(\mathbf{x}_T; \hat{\theta}_S^{\text{TL}}) = \left\langle \mathbf{w}_{\mu,T}^{(K_T+1)}, h_{\mu,T}^{(K_T)}(\mathbf{x}_T; \hat{\theta}_S^{\text{TL}}) \right\rangle$$

$$\lambda_{\pi,T} : \mathcal{X}_T \rightarrow [0, 1], \quad \mathbf{x}_T \mapsto \text{logit } \lambda_{\pi,T}(\mathbf{x}_T) = \left\langle \mathbf{w}_{\pi,T}^{(K_T+1)}, h_{\pi,T}^{(K_T)}(\mathbf{x}_T) \right\rangle$$

- Le precedenti funzioni di regressioni sono adottate per applicare sia l'initialization che il freezing approach
- La loss function adottata per il processo di apprendimento è la negative log-likelihood per la distribuzione ZIP

## Applicazione numerica

- **Dataset:** Individual Life Experience Committee study on mortality 2012-2019 (Society of Actuaries)
  - ▷ Contiene i dati sull'esperienza di mortalità del mercato vita USA
  - ▷ Circa 4.5 mln di record disponibili
- Selezionati random 1 milione di record, successivamente divisi in source data e target data
- **Source dataset** composto da 900k record, di cui 897.8k presentano zero decessi
- **Target dataset** composto da 93.3k record, di cui 92.6k presentano zero decessi
- Sia source che target dataset sono suddivisi in **train-validation-test** sets secondo la proporzione **70%-10%-20%** (record randomicamente allocati)

# Applicazione numerica

- **Dataset: variabile risposta ed esplicative** (sia per source e target)

Risposta $y_i$	Descrizione	Tipo
Numero decessi, $d$	Num. aggregato di decessi per esposizione	Numerica

Esplicativa $x_i$	Descrizione	Tipo
Esposizione, $E^c$	Num. aggregato di polizze esposte	Numerica
Issue age	Età policyholder alla data di emissione	Numerica
Attained age	Età raggiunta dal policyholder	Numerica
Observation year	Anno in cui $d$ ed $E$ sono misurati	Numerica
Duration	Durata polizza	Numerica
Gender	Genere del policyholder	Categorica
Smoker status	Fumatore o meno	Categorica
Insurance plan	Tipo piano assicurativo	Categorica
Face amount band	Intervallo per somma assicurata in polizza	Categorica
Post level term indicator	Se la durata della polizza è nel Guaranteed Level Term period	Categorica
Guaranteed level term period	Se la polizza ha un fixed-premium term period	Categorica
Anticipate level term period	Se la polizza ha un anticipated term level period	Categorica

# Applicazione numerica

## Data pre-processing, fine-tuning, training

- Min-max scaling delle variabili numeriche
- Per variabili categoriche, data la cardinalità  $\mathcal{C}$ :
  - ▷ Se  $\mathcal{C} < 5$ , adottato one-hot encoding
  - ▷ Se  $\mathcal{C} \geq 5$ , adottato un embedding layer

$$e : \Psi_{\mathbf{x}_k} \rightarrow \mathbb{R}^k, \quad \mathbf{x}_k \mapsto e(\mathbf{x}_k) \in \mathbb{R}^k$$

- Selezione degli iper-parametri delle FFN via Bayesian optimization (Tree-Structured Parzen estimator, Optuna library, Python)
- Processo di apprendimento: backpropagation e gradient descent algorithm
- Profondità delle FFNs prefissata:  $K_S = K_T = 4$

# Applicazione numerica: Modelli di trasferimento

## Senza TL

- ① **FFN from scratch:** utilizzo diretto della FFN

$$\ln \lambda_{\mu,T}(\mathbf{x}_T) = \left\langle \mathbf{w}_{\mu,T}^{(K_T+1)}, h_{\mu,T}^{(K_T)}(\mathbf{x}_T) \right\rangle, \quad \text{logit } \lambda_{\pi,T}(\mathbf{x}_T) = \left\langle \mathbf{w}_{\pi,T}^{(K_T+1)}, h_{\pi,T}^{(K_T)}(\mathbf{x}_T) \right\rangle$$

- ② **Zero-inflated Brass**

$$\ln \lambda_{\mu,T} = \ln E_T^C + \alpha + \beta \ln d_S, \quad \text{logit } \lambda_{\pi,T}(\mathbf{x}_T) = \langle \boldsymbol{\beta}_{\pi,T}, \mathbf{x}_T \rangle$$

- ③ **Credibility-like FFN:** combinazione lineare (convessa) della mortalità del source e quella estrapolata dal target

$$\ln \lambda_{\mu,T}(\mathbf{x}_T) = \alpha \ln d_S + (1 - \alpha) \left\langle \mathbf{w}_{\mu,T}^{(K_T+1)}, h_{\mu,T}^{(K_T)}(\mathbf{x}_T) \right\rangle, \quad \alpha \in [0, 1].$$

# Applicazione numerica: Modelli di trasferimento

## Con TL

- $\ln \lambda_{\mu,T}(\mathbf{x}_T; \hat{\theta}_S^{\text{TL}}) = \left\langle \mathbf{w}_{\mu,T}^{(K_T+1)}, h_{\mu,T}^{(K_T)}(\mathbf{x}_T; \hat{\theta}_S^{\text{TL}}) \right\rangle$ 
  - 1 "No freeze" (Initialization approach)
  - 2 "Freeze 1" (Freezing approach: solo primo layer)
  - 3 "Freeze 2" (Freezing approach: primi due layer)
  - 4 "Freeze 3" (Freezing approach: primi tre layer)
  - 5 "Freeze 4" (Freezing approach: tutti e 4 i layer)

## Applicazione numerica: Risultati out-of-sample

Out-of-sample loss per ogni modello implementato

<b>Modello</b>	<b>Loss</b>
Scratch	0.125869
ZI-Brass	0.122203
Credibility-FFN	<b>0.120941</b>
TL "no freeze"	0.121921
TL "freeze 1"	0.121672
TL "freeze 2"	0.121398
TL "freeze 3"	0.121820
TL "freeze 4"	0.121127

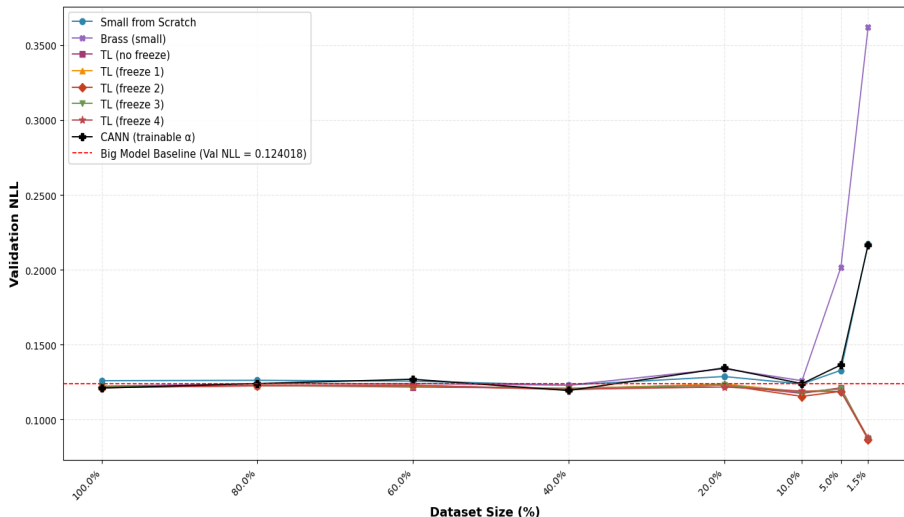
## Applicazione numerica: Risultati out-of-sample

Out-of-sample loss per ogni modello e per differenti dimensioni del target dataset

Size (%)	ZI-Brass	Cr-FFN	Scratch	TL "no freeze"	TL "freeze 1"	TL "freeze 2"	TL "freeze 3"	TL "freeze 4"
100%	0.122203	<b>0.120941</b>	0.125869	0.121921	0.121672	0.121398	0.121820	0.121127
80%	0.123842	0.123930	0.126116	0.122645	0.122611	<b>0.122364</b>	0.122536	0.122546
60%	0.123876	0.126797	0.125529	<b>0.121629</b>	0.122187	0.122678	0.122523	0.122745
40%	0.122868	<b>0.119281</b>	0.122931	0.120752	0.120534	0.119868	0.120488	0.120083
20%	0.133851	0.134358	0.128624	0.123009	0.123612	0.122952	0.123065	<b>0.121629</b>
10%	0.125834	0.123867	0.123632	0.117365	0.118369	<b>0.115344</b>	0.118060	0.118898
5%	0.201561	0.136388	0.132779	0.121121	0.120366	0.118760	0.120258	<b>0.118590</b>
1.5%	0.362002	0.216590	0.217134	0.086766	0.087412	<b>0.086526</b>	0.087356	0.087532

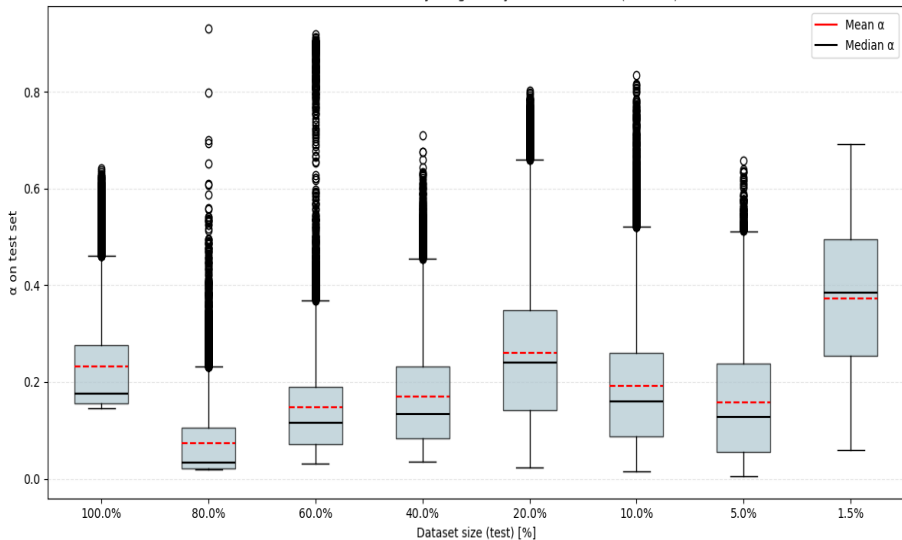
# Applicazione numerica: Risultati out-of-sample

**Sensitivity Analysis: Val NLL vs Dataset Size**  
(Transfer Learning vs From Scratch + Brass + CANN)



# Applicazione numerica: Risultati out-of-sample

Distribution of credibility weight  $\alpha$  by dataset fraction (test set)



# Takeaways

- Il TL è un framework di apprendimento che permette di generalizzare gli esistenti modelli relazioni di mortalità
- Esso richiede la disponibilità di dati relativamente a:
  - ▷ La popolazione di riferimento (source dataset)
  - ▷ La popolazione d'interesse (target dataset)
- Il TL è particolarmente utile qualora la popolazione d'interesse sia caratterizzata da scarsa esperienza di mortalità (dimensione popolazione e/o insufficiente profondità storica)

# Sull'incertezza di previsione nel TL

- Diverse fonti d'incertezza:
  - ▷ Incertezza intrinseca: distribuzione del numero di decessi
  - ▷ Incertezza di modello: risente dell'incertezza relativa sia alla funzione predittiva source, e ai suoi parametri, sia dell'incertezza associata alla funzione predittiva target e ai suoi parametri
  - ▷ Incertezza di trasferimento: dovuta a  $\mathbb{P} \neq \tilde{\mathbb{P}}$
- Costruzione di un'intervallo di previsione per l'incertezza complessiva  
↪ **Conformal prediction** (CF)
- Nella CF, data una sequenza di punti  $(X_1, Y_1), \dots, (X_n, Y_n)$ , l'obiettivo è fare una previsione per un nuovo punto  $(X_{n+1}, Y_{n+1})$ , dove  $Y_{n+1}$  non è noto

## Sull'incertezza di previsione nel TL

- Invece che produrre la previsione puntuale  $\hat{Y}_{n+1}$ , la CF produce un insieme di previsioni  $\mathcal{C}(X_{n+1}) \subseteq \mathcal{Y}$  tale che

$$\mathbb{P}(Y_{n+1} \in \mathcal{C}(X_{n+1})) \geq 1 - \alpha \quad (4)$$

con  $\alpha \in (0, 1)$  detto *nominal coverage level*.

- La costruzione dell'insieme  $\mathcal{C}(X_{n+1})$  richiede:
  - ▷ la scelta di un **nonconformity score**,  $s : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ , tale che  $s(x, y)$  misura quanto è non conforme la coppia  $(x, y)$  rispetto ai training data e al modello di apprendimento
    - ↪ Log-score:  $s(x, y) = |\log(y + 1) - \log(\hat{\mu}(x) + 1)|$   
con  $\hat{\mu}(x) = (1 - \hat{\pi}(x))\hat{\lambda}(x)$
  - ▷ la scelta di un algoritmo di CF ↪ **Split Conformal Prediction**

# Sull'incertezza di previsione nel TL

---

## Algorithm : Split Conformal Prediction

---

- 1: **Splittare** i dati in un training set  $\mathcal{D}_{\text{train}}$  e un calibration set  $\mathcal{D}_{\text{cal}} = \{(X_i, Y_i)\}_{i=1}^m$
- 2: **Addestrare** il modello  $\hat{f}$  su  $\mathcal{D}_{\text{train}}$
- 3: **Calcolare** i nonconformity scores sul calibration set:

$$s_i = s(X_i, Y_i; \hat{f}), \quad i = 1, \dots, m$$

- 4: **Calcolare** il quantile empirico a livello  $(1 - \alpha)$

$$\hat{q} = \text{Quantile}\left(\{s_1, \dots, s_m\}, \frac{\lceil (m+1)(1-\alpha) \rceil}{m}\right)$$

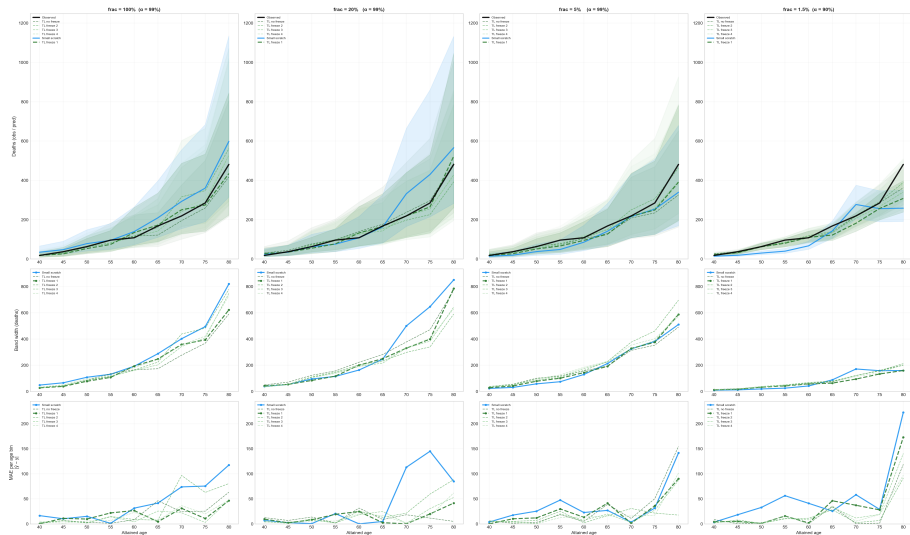
- 5: **Construct** the prediction set for test point  $X_{n+1}$ :

$$\mathcal{C}(X_{n+1}) = \{y \in \mathcal{Y} : s(X_{n+1}, y; \hat{f}) \leq \hat{q}\}$$

---

# Sull'incertezza di previsione nel TL

Aggregated Mortality Curve with Conformal Prediction Bands by Training Fraction  
(age 40-80, Log score)



Grazie per la vostra attenzione

Contact: [mario.marino@deams.units.it](mailto:mario.marino@deams.units.it)

## References

- Delwarde, A., Kachachidze, D., Olie, L., and Denuit, M. (2004). Modeles lineaires et additifs generalises, maximum de vraisemblance local et methodes relationnelles en assurance sur la vie. *Bulletin Francais d'Actuariat*, 6:77–102.
- Weiss, K., Khoshgoftaar, T. M., and Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1).
- Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.