doi:10.3969/j.issn.1001-4616.2024.03.017

DeephitTM:医学生存分析的时间相关性深度学习模型

张大鹏[1,2],程学亮[2],孙明霞[1]

(1.江苏信息职业技术学院物联网工程学院,江苏 无锡 214153)
(2.燕山大学信息科学与工程学院,河北 秦皇岛 066004)

[**摘要**] 生存分析是医学中经常用到的一种健康预测方法,越来越多的学者开始采用深度学习的方法对生存分析问题进行建模以得到更好的预测结果. 目前已有的方法都假设风险和时间的联合概率是无关联的. 然而生存分析数据的实际结果中却包含时间因素,这就无法保证不同时刻得到的风险概率是无关联的. 本文提出一种带有时间相关性的深度学习模型 DeephitTM,该模型对已有的深度学习模型 Deephit 进行了改进. 实验结果表明,在不同的数据集上,改进后的模型的性能相比于原模型能够提升 1 到 3 个百分点.

[**关键词**] 生存分析,深度学习,时间相关性,神经网络,Deephit 模型
[**中图分类号**]TP391 [**文献标志码**]A [**文章编号**]1001-4616(2024)03-0138-11

DeephitTM:a Time-dependent Deep Learning Model for Medical Survival Analysis

Zhang Dapeng[1,2],Cheng Xueliang[2],Sun Mingxia[1]

(1.School of IoT Engineering,Jiangsu Vocational College of Information Technology,Wuxi 214153,China)
(2.College of information Science and Engineering,Yanshan University,Qinhuangdao 066004,China)

Abstract:Survival analysis is a health prediction method often used in medicine. More and more scholars start to use deep learning method to model survival analysis problems to get better prediction results. Currently, existing methods assume that the joint probability of risk and time is uncorrelated, but the actual results of survival analysis data contain time factors, which cannot guarantee that the risk probability obtained at different times is uncorrelated. This paper proposes a time-dependent deep learning model, DeephitTM, to improve the existing deep learning model Deephit. Experimental results show that the performance of our model can be improved by 1 to 3 percentage points compared with the original model on different data sets.

Key words:survival analysis,deep learning,temporal correlation,neural network,Deephit mode

生存分析问题是一类问题的统称,如股票何时盈利,工厂机器何时运转不正常,甚至一个病人何时死亡,都属于生存分析问题. 当医生需要为病人制定治疗方案时,好的生存分析模型可以帮助医生制定更加合理的治疗方案来提高医疗水平.

生存分析问题存在于医学、经济金融和工程学等多个领域,但其本质是对研究的个体进行时间和风险的联合概率预测. 分析问题的研究首先是建立在样本数据的基础上的,通过对已知的样本记录数据进行处理,得到生存分析模型来预测新的样本未来的风险概率,但是不同的模型因为对样本记录数据的不同处理,进行生存分析的能力是不同的.

生存分析有传统的生存分析模型,也有机器学习模型. 传统模型大多是基于统计学、随机过程和概率

收稿日期:2022-09-20.
基金项目:国家自然科学基金项目(61973261)、江苏省高等学校自然科学研究面上项目(18KJD510011)、江苏省高等职业教育高水平专业群建设项目(苏教职函〔2021〕1 号).
通讯作者:张大鹏,博士,副教授,研究方向:机器学习. E-mail:daniao@ ysu.edu.cn

论的数学模型,如 Cox 模型[1]、Kaplan-Meier 模型[2]和 Aaron 等提出的统计模型[3]等. 机器学习模型如 Faraggi 等设计的神经网络模型[4]、基于随机森林的 RSF 模型[5-6]、Luck 等[7]基于神经网络改进的 Cox 模型、DMGP 模型[8]、Deephit 模型[9]和基于支持向量机的生存分析模型[10-12]等等.

Kaplan-Meier 曲线可以从样本总体上刻画不同时刻样本发生某一事件的概率. 可以直观地看到病人死亡率随着时间的变化情况,但是无法对具体样本进行具体分析. 有的学者认为风险发生时间的概率分布可以看作是隐藏的随机过程的首达时间的概率分布[9],即样本自变量和风险时间之间具有某种概率分布对应关系. Cox 回归模型通过主观假设函数表示形式可以基于样本自变量来预测结果. 相比于 Kaplan-Meier 模型,Cox 回归模型可以根据样本的自变量来分析样本不同时间的风险概率,但该模型是基于主观假设建立的,无法保证该模型的概率分布函数的真实性,且无法用于多个风险存在时的生存分析.

不仅 Cox 回归模型有主观假设的问题,基于 Cox 回归模型改进的深度 Cox 风险模型[13]、Luck 等基于神经网络改进的 Cox 模型和 BecCox 模型[14],以及基于机器学习的 Deep Survival Analysis 模型、DeepSurv[15]模型和 DMGP 模型等,均在不同程度上进行了主观假设. 以 DMGP 模型为例,该模型通过构建多层高斯过程来拟合生存分析中输入特征与风险概率之间的关系,主观假设生存分析问题符合高斯过程,其优点在于加入了竞争风险[16-17],适用于多风险的生存分析问题. 2018 年 Bellot 等[18]提出的 HBM 模型也可以用于存在竞争风险的生存分析,但仍然存在主观假设的问题.

为了解决上述模型中存在的主观假设问题,Deephit 模型采用对任意曲线均有很好的拟合效果的神经网络结构拟合生存分析问题中可能存在的函数形式,并且可以进行多风险生存分析,相比于上述模型能够更全面的进行生存分析. 此外 DRSA[19]和 Liu 等基于 GBDT 方法建立的 Hitboost 模型[20]也能很好的拟合生存分析问题中的函数表示,同时相比于 Deephit 模型具有一定的优势. 如 DRSA 模型考虑到时间因素对生存分析问题性能的影响,采用了循环神经网络进行生存分析,Hitboost 模型采用集成学习方法中的 GBDT 模型来加强对生存分析数据的学习能力,增加了模型的可解释性.

虽然 DRSA 模型和 Hitboost 模型具有一定的优势,但也有不足之处. DRSA 模型的循环神经网络结构在序列数据集上具有更好的适用性,但当数据为非序列数据且考虑时间因素时无法很好的适用,而且当样本的时间范围较大时,如当数据集的生存时间范围包含的时间点为上千个甚至更多时,RNN 模型无法进行训练,而且即使可以训练也会产生训练时间长、计算复杂及梯度如何处理等问题. Hitboost 模型未考虑竞争风险存在的情况,而且存在训练时间长的问题.

从 Deephit 模型的不同方面出发,相关学者进行了一些改进,如通过加入特征选择的预处理模型来对数据进行预处理提升模型的性能[21]、建立 Dynamic-DeepHit 模型[22]以更好的对重复测量数据进行生存分析. 这虽然在一定程度上使改进的模型得到了性能提升,但是对这些模型和 Deephit 模型进行分析,发现限制该模型性能的原因除了数据集相关问题,模型的输出方式也存在一定的问题. 通过构建和训练 Deephit 模型,可以得知 Deephit 模型采用多分类神经网络[23]来进行生存分析问题的研究,该模型适用于对无关联的多个类别实行分类任务. 该模型输出的实际意义为在各个时间点发生某种风险的联合概率,但是生存分析数据的实际结果中包含时间因素,该模型无法保证不同时刻得到的风险概率之间是无关联的. Deephit 模型产生无关联分类主要是因为该模型简单地采用 softmax 激活函数作为输出层的激活函数,一方面会使不同时间和风险的联合概率之间没有任何相关性的约束;另一方面使得整个研究时段的总体风险概率为 1,但是由于数据集本身有部分缺失,假设所研究的时间段内总体风险概率为 1 也是无法得到保证的.

综合相关研究思路,本文将对 Deephit 模型进行改进,通过对上述分析中 Deephit 模型存在的问题进行研究,加入时间的相关性因素,修改模型的损失函数使其更加合理,最终模型可以学习到时间相关的概率分布. 经过实验与 Deephit 模型等其他模型对比来验证修改后的模型的性能提升.

1 技术背景

1.1 Deephit 模型简介

生存分析问题的数据由样本的特征、时间和风险类型组成. 样本数据集含有 N 个样本,可以表示为 $\{x^i,e^i,t^i\}_{i=1}^N$. 其中 x^i 表示第 i 个样本的特征;e^i 表示第 i 个样本发生的风险,$e^i \in \{\phi,e_1,e_2,\cdots,e_k\}$,$\phi$ 表示

数据删失;t^i 表示第 i 个样本发生风险的时间,$t^i \in \{0, T_1, T_2, \cdots, T_{\max}\}$,风险的最大时间为 $T_{\max}$. 对于每个样本,最多发生单个风险,不存在样本在某一时间段内发生两种或者更多风险的情况. 由于数据删失的存在,使得数据集本身存在信息缺失,这是限制模型学习能力的主要原因. 样本发生删失一般指样本直到删失时间未发生任何风险,删失时间之后样本是否经历某种风险无法确定.

Deephit 模型的结构如图 1 所示,其中 x 为研究对象的 m 维的输入特征向量. 该模型包含两部分全连接神经网络,即图 1 所示的 Shared Sub-network 和 Cause-Specific Sub-network. Shared Sub-network 的结果与 x 经过残差连接[24]处理后得到 Z,将 Z 作为 Cause-Specific Sub-network 输入数据. 当竞争风险个数为 n 时,Deephit 模型的 Cause-Specific Sub-network 模块可分为 Cause-Specific Sub-network 1,…,Cause-Specific Sub-network n,得到的输出分别对应各个风险的 $T_{\max}$ 个学习结果. 所有 Cause-Specific Sub-network 的输出结果拼接在一起经过 softmax 激活函数得到最终预测结果.

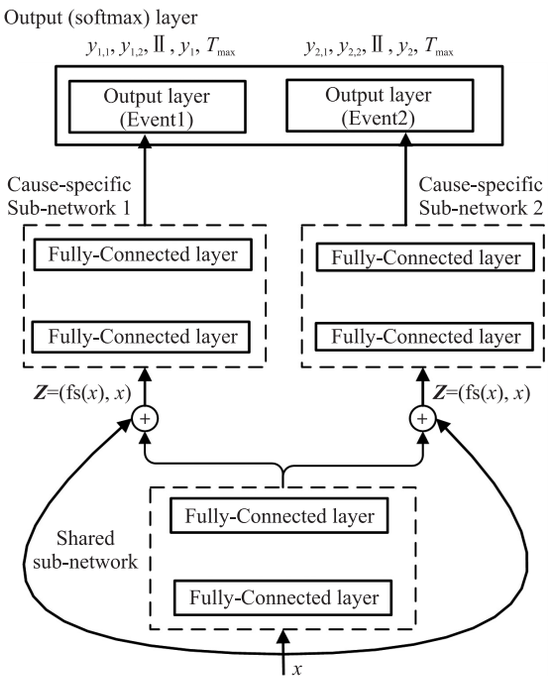


图 1　Deephit 模型结构

Fig. 1　Structure of Deephit Model

含有两种竞争风险时最终的概率预测表示为:$y = \{y_{1,1}, \cdots, y_{1,T_{\max}}, y_{2,1}, \cdots, y_{2,T_{\max}}\}$. Deephit 模型的损失函数包含两部分,其中第一部分损失如式(1)所示:

$$Loss_1 = -\sum_{i=1}^{N}\left[\alpha(e^{(i)} \neq \phi) \times \log(y^{(i)}_{e^{(i)},s^{(i)}}) + \alpha(e^{(i)} = \phi) \times \log 1 - \sum_{e \in \{e_1,e_2\}} \hat{F}_e(s^{(i)} \mid x^{(i)})\right], \tag{1}$$

式中,$\alpha(m)$的值在条件 m 成立时为 1,不成立时为 0. 在该损失函数中,当样本 i 是不含删失数据的样本时,训练的目的是最大化风险 $e^{(i)}$ 下时刻 $s^{(i)}$ 处的预测概率 $y^{(i)}_{e^{(i)},s^{(i)}}$.

Deephit 模型的损失函数第二部分如式(2)所示:

$$Loss_2 = \sum_{e \in \{e_1,e_2\}} \alpha_e \cdot \left(\sum_{i \neq j} A_{e,i,j} \times \eta(\hat{F}_e(s^{(i)} \mid x^{(i)}), \hat{F}_e(s^{(i)} \mid x^{(j)}))\right), \tag{2}$$

式中,$A_{e,i,j}$表示样本 i 和样本 j 是否满足如下条件:e 为样本 i 的实际风险,样本 i 发生风险时的真实时间点 $s^{(i)}$ 是否小于样本 j 发生风险时的真实时间点 $s^{(j)}$,满足该条件时 $A_{e,i,j}=1$,否则 $A_{e,i,j}=0$.

对于满足 $A_{e,i,j}=1$ 的样本对(i,j),可以知道,样本 j 发生风险的真实时间点在样本 i 发生风险之前,因此在风险 $e=e^{(i)}$ 下,可以推测出样本 j 直到时刻 $s^{(i)}$ 预测的累积概率应当小于样本 i 直到时刻 $s^{(i)}$ 预测的累积概率这一事实(记为 R).

第二部分损失中,$\eta(\hat{F}_e(s^{(i)} \mid x^{(i)}), \hat{F}_e(s^{(i)} \mid x^{(j)}))$是用于度量在样本对$(i,j)$处的预测累积概率与事实 R 的符合程度的函数表达式,其中 $\hat{F}_e(s^{(i)} \mid x^{(i)})$和 $\hat{F}_e(s^{(i)} \mid x^{(j)})$分别为 i 和 j 在风险 k 下直到时刻 $s^{(i)}$ 预测的累积风险概率,该部分函数可以表示为 $\eta(x,y) = \exp(-(x-y)/\beta)$. $Loss_2$ 有两部分超参数,分别为 $\alpha_e, e \in (e_1,e_2)$及 β,依照原论文设置,训练模型时使 $\alpha_{e1}=\alpha_{e2}=\alpha$,选取合适的 α 和 β 得到具有最优性能的模型.

值得注意的是,$Loss_2$ 虽然在一定程度上可以量化满足 R 的样本对(i,j)的预测概率的损失,但是 $Loss_2$ 的函数表示形式属于主观假设,缺乏理论支持,因此会产生损失误差.

1.2　生存分析度量标准

C^{td}指数[9,18]是生存分析的度量标准[17,25-26],可以用式(3)表示:

$$C^{td} = P(\hat{F}_e(s^{(i)} \mid x^{(i)}) > \hat{F}_e(s^{(i)} \mid x^{(j)}) \mid s^{(i)} < s^{(j)}) \approx \frac{\sum_{i \neq j} A_{e,i,j} \times \alpha(\hat{F}_e(s^{(i)} \mid x^{(i)}) > \hat{F}_e(s^{(i)} \mid x^{(j)}))}{\sum_{i \neq j} A_{e,i,j}}. \tag{3}$$

此度量标准表示:在样本 i 真实发生风险为 $e^{(i)}$ 的时刻 $s^{(i)}$ 小于另一个样本 j 发生任何风险的时刻 $s^{(j)}$,在风险 $e^{(i)}$ 下,直到样本 i 风险发生时刻 $s^{(i)}$,样本 i 的累积风险概率应当大于样本 j 到时刻 $s^{(j)}$ 的累积风险概率,即对于样本对(i,j),当满足条件 $e=e^{(i)}$ 且 $s^{(i)}<s^{(j)}$ 时,累积风险概率 $F_e(s^{(i)}|x^{(i)})>F_e(s^{(i)}|x^{(j)})$. C^{td} 衡量实际符合上述依据的样本对的预测结果同样符合上述依据的样本对的占比情况.

当 C^{td} 趋向于 0.5 时,模型得到的输出相当于随机输出;当 C^{td} 趋向于 1 时,模型输出越趋近于实际输出结果.

2 模型分析

Deephit 模型相比于其他生存分析模型考虑的问题比较全面,如竞争分析、生存分析中概率函数的拟合方式、如何建立样本特征与生存分析结果之间的模型结构等. 因此 Deephit 模型是一种比较有代表性的生存分析模型. 目前大部分的生存分析模型研究样本与风险和时间之间的绝对概率,本文考虑采用条件概率研究样本与风险发生时间的关系. 以 Deephit 模型为例对比绝对概率和条件概率在生存分析研究中的区别,主要分为以下三部分.

(1)结合 Deephit 论文中的介绍和输出层的 softmax 激活函数可知样本各时刻的累积概率和为 1. 即如式(4)所示.

$$\sum_{e=1}^{K}\sum_{t=1}^{T_{\max}} y_{t,e}=1. \tag{4}$$

假设各时刻的条件概率表示如式(5)所示,则基于条件概率的累积概率和如式(6)所示.

$$Y=\{Y_{1,1},\cdots,Y_{1,T_{\max}},\cdots,Y_{K,1},\cdots,Y_{K,T_{\max}}\}, \tag{5}$$

$$P=\sum_{e=1}^{K}\sum_{t=1}^{T_{\max}} Y_{k,t}\cdot\prod_{i=1}^{t-1}\prod_{j=1}^{K}(1-Y_{j,i}). \tag{6}$$

只有当某一时刻的条件概率为 1 时,$P=1$,否则 $P\neq 1$. 基于绝对概率的概率和为 1,而基于条件概率的概率和不一定为 1. 由于数据集的时间范围不一定可以代表样本风险概率的所有时间范围,且无法用有限的数据集代表所有真实样本的风险发生情况,所以强制研究时间范围内的概率和为 1 不合理,因此基于条件概率的生存分析研究更合适.

(2)以最大化真实风险发生时刻的概率定义损失函数(即 Deephit 模型的 $Loss_1$),基于绝对概率和条件概率的损失函数分别如式(7)和(8)所示.

$$Loss_y=-\sum_{i=1}^{n}\log(y^{(i)}_{e^{(i)},t^{(i)}}), \tag{7}$$

$$Loss_Y=-\sum_{i=1}^{n}\left(\log(Y^i_{e^i,t^i})+\sum_{t=1}^{t^i-1}\sum_{e=1}^{K}\log(1-Y^i_{e,t})\right), \tag{8}$$

基于式(7)可以推导出式(9),过程如下.

$$\min-\sum_{i=1}^{N}\log(y^{(i)}_{e^{(i)},s^{(i)}})]\Leftrightarrow\min-\sum_{i=1}^{N}\log\left(1-\sum_{t\neq e^{(i)},e\neq s^{(i)}}y^{(i)}_{t,e}\right)]\Leftrightarrow\min-\sum_{i=1}^{N}\log\left(1-\sum_{e\neq s^{(i)}}\sum_{t=0}^{t^{(i)}-1}y^{(i)}_{t,e}+\sum_{t=t^{(i)}+1}^{T_{\max}}y^{(i)}_{t,e}\right)]. \tag{9}$$

由式(9)可知,在最大化真实风险发生时刻的绝对概率式会最小化其他时刻的风险概率,即认为真实风险之后的所有时刻的风险概率同样影响该时刻的风险发生情况,由于该时刻风险发生情况已经确定,显然无法受到之后各时刻风险情况的影响.

基于条件概率的损失函数只和真实风险时刻之前的风险概率有关,更符合不同时刻的风险影响情况,因此更适合用于生存分析.

(3)虽然绝对概率和条件概率可以相互转换,但是能够相互转换的前提是具有正确的绝对概率和条件概率,当模型的预测方式、损失函数表示方式及模型结构不同时,预测结果也会有所不同.

除了概率表示方式的不同,模型结构在一定程度上也会影响训练速度和效果,下面对 Deephit 模型结构进行分析. 当竞争风险多时,在 Deephit 模型中 Cause-Specific Sub-network 部分含有多个并行部分,会增加模型的复杂度和计算量,为了对比含有多个并行 Cause-Specific Sub-network 和单个 Cause-Specific Sub-

network 的区别,得到实验的结果如图 2(a)(b)所示.

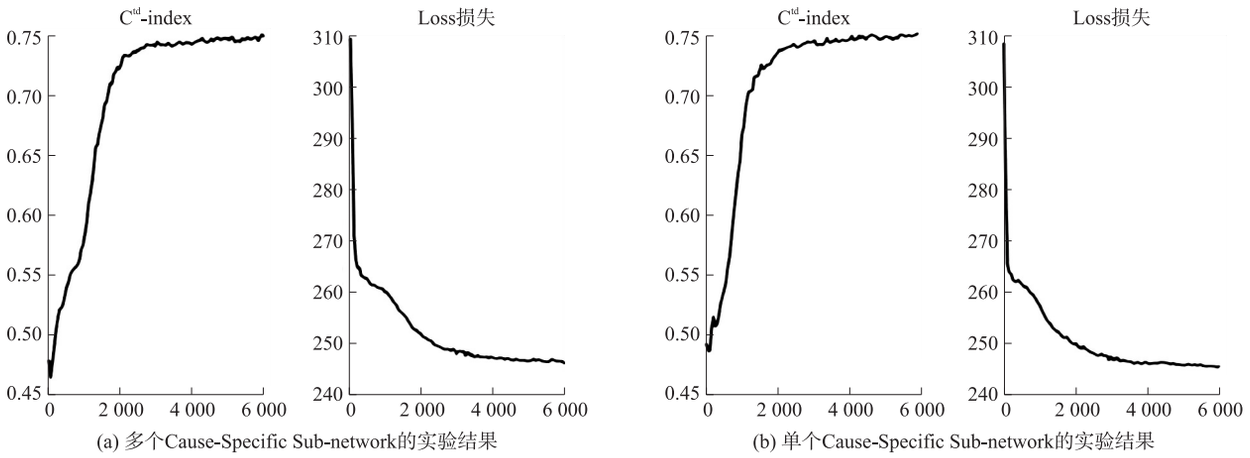


图 2　不同结构下 Deephit 模型实验分析

Fig. 2　Experimental analysis of Deephit model under different structures

图 2 是在保持模型其他结构不变(详见 Deephit 模型结构),损失函数相同(即 Deephit 模型的损失函数及相关设置)的情况下,不同 Cause-Specific Sub-network 结构的 Deephit 模型在 Synthetic 数据集下的训练结果,该数据集是多风险数据集,有关 Synthetic 数据集的详细介绍参见第 4 节. 图 2(a)是基于多个 Cause-Specific Sub-network 并行运行的原始 Deephit 模型在 Synthetic 数据集上训练过程中性能变化和损失函数变化情况,图 2(b)是 Deephit 模型采用单个 Cause-Specific Sub-network 时在相同数据集上训练过程中性能变化和损失函数变化情况,该图由于采用单个 Cause-Specific Sub-network,为了得到合适的输出,将该 Cause-Specific Sub-network 模块的最后一层神经元的个数设置为风险个数与 $T_{\max}$ 的乘积.

由图 2 可知,在整个模型训练过程中,两种方式下的性能变化过程和预测结果基本一致,但是多个 Cause-Specific Sub-network 并行运行的神经元数量会增加,进而带来更大的计算负担,因此存在多竞争风险时,采用具有合适的神经元个数的单个 Cause-Specific Sub-network 更为合适.

3　DeephitTM 模型

在 Deephit 模型中并没有体现不同时刻概率之间的关联性,更倾向于进行多个无关联的分类任务的学习. 结合第 2 节中的分析采用条件概率研究生存分析中的概率分布,可以学习到合理的时间相关性规律. 基于条件概率的 DeephitTM 模型的具体结构如图 3 所示.

DeephitTM 模型中的 Sub-network1 和 Sub-network2 是两个全连接神经网络结构. 首先样本特征向量 $\boldsymbol{x}$ 作为模型输入经过 Sub-network1, Sub-network1 的输出值 $f(x)$ 与输入 x 通过残差连接的方式拼接成 Input, Input 作为 Sub-network2 的输入,经过 Sub-network2 输出结果为一个 N 行、$K*T_{\max}$ 列的矩阵,可以表示为如式(10)所示的方式,其中 N 是一个超参数,K 为风险个数,$T_{\max}$ 为最大风险时间.

$$\begin{bmatrix} o_{1,1}^{1} & \cdots & o_{1,T_{\max}}^{1} & \cdots & o_{K,1}^{1} & \cdots & o_{K,T_{\max}}^{1} \\ & \vdots & & \ddots & & \vdots & \\ o_{1,1}^{N} & \cdots & o_{1,T_{\max}}^{N} & \cdots & o_{K,1}^{N} & \cdots & o_{K,T_{\max}}^{N} \end{bmatrix}. \quad (10)$$

Output mark 是一个 N 行、$K*T_{\max}$ 列的自定义矩阵,在风险 k 下如式(11)所示.

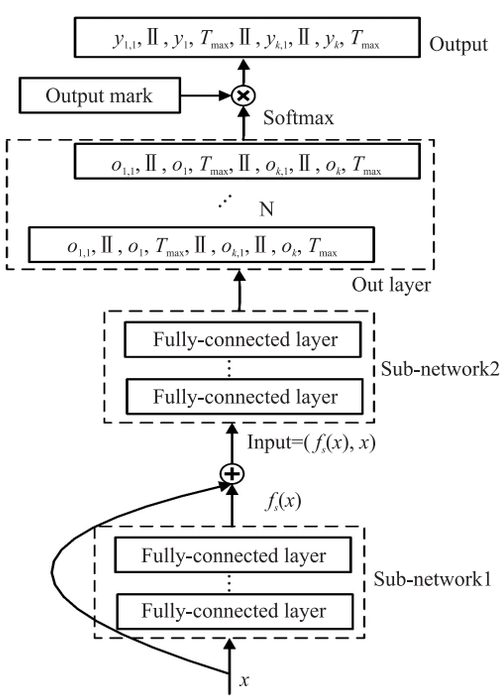


图 3　DeephitTM 模型结构

Fig. 3　Structure of DeephitTM Model

$$\begin{bmatrix} m_{1,1} & \cdots & m_{1,T_{\max}} \\ \vdots & \ddots & \vdots \\ m_{N,1} & \cdots & m_{N,T_{\max}} \end{bmatrix}. \tag{11}$$

Sub-network2 的输出矩阵按列通过 softmax 激活函数处理后与 Output mark 共同生成最终的预测结果,对于风险 k 在时刻 t 处的条件概率的具体实现如式(12)及(13)所示.

$$\mathrm{softmax}\left(\begin{bmatrix} o_{k,t}^{1} \\ \vdots \\ o_{k,t}^{N} \end{bmatrix}\right) = \begin{bmatrix} O_{k,t}^{1} \\ \vdots \\ O_{k,t}^{N} \end{bmatrix}, \tag{12}$$

$$y_{k,t} = \sum_{i=1}^{N} m_{i,(k-1)*T_{\max}+t} * O_{k,t}^{i}. \tag{13}$$

下面讨论 Output mark 的定义方式,主要分为三种:

(1)当每一列设置为固定值时,如 $N=2$ 时,设置矩阵第一行元素全为 1,第二行全为 0. 当 $N>2$ 时,设置前 m 行元素为 1($0<m<N$),剩余($N-m$)行全为 0. 针对此情况下 Output mark 的定义方式,本次实验主要设置 $N=2$ 和 $m=1$.

(2)当每一列设置为变化值,如每一列采用 Kaplan-Meier 曲线下的条件概率作为参考值时,设置 $N=3$,矩阵第一行元素全为 1,第二行元素分别为对应风险和时刻由 Kaplan-Meier 曲线得到的条件概率,第三行元素全为 0.

(3)当 $N=1$ 时,设置矩阵元素全为 1,式(12)中的激活函数改为 sigmoid 函数.

三种 Output mark 的定义方式主要用于筛选合适的 DeephitTM 模型作为最优模型,按照 Output mark 的定义方式的不同,相应的 DeephitTM 模型分别标记为 DeephitTM_FR、DeephitTM_KM 及 DeephitTM_SIG 模型.

基于条件概率的样本 i 在时刻 t 发生风险 $e(i)$ 的概率 $P_{e^{(i)},t}$ 的大小依赖于直到上一时刻该样本未发生风险 $e(i)$ 的概率,因此在样本 i 上的 $P_{e^{(i)},t}$ 如式(14)所示.

$$P_{e^{(i)},s^{(i)}}^{i} = P(t=s^{(i)} \mid e=e^{(i)}, x=x^{(i)}, t\geqslant s^{(i)}) \times \prod_{T<s^{(i)}} (1-P(t=T \mid e=e^{(i)}, x=x^{(i)}, t\geqslant T)), \tag{14}$$

式中,$P(t=s^{(i)} \mid e=e^{(i)}, x=x^{(i)}, t\geqslant s^{(i)})$ 表示样本 i 存活到时刻 $s(i)$ 的情况下,在时刻 $s(i)$ 发生风险 $e(i)$ 的概率 $y_{s^{(i)},s^{(i)}}^{i}$.

该模型的损失函数分为三部分,第一部分损失如式(15)所示.

$$Loss_1 = -\sum_{i=1}^{N} \left[\alpha(k^{(i)} \neq \phi) \times \log(P_{k^{(i)},s^{(i)}}^{(i)}) + \alpha(k^{(i)} = \phi) \times \left(\sum_{e\in\{e_1,e_2\}} \left(\sum_{t=1}^{s^{(i)}} \log(1 - y_{e,t}^{(i)})\right)\right)\right], \tag{15}$$

该式中的 $\alpha(m)$ 同样表示一个判断函数,当 m 表示的条件为真时返回值为 1,否则返回 0. i 代表第 i 个样本,$i\in 1,2,\cdots,N$. e 表示风险,$e\in\{e_1,e_2\}$. $s(i)$ 为第 i 个样本实际发生风险或者数据删失时的真实时刻. Loss1 表示当样本未发生删失时,使模型在实际风险发生时间的预测概率尽可能增大,否则使删失时刻之后的风险尽可能增大. 每个时刻发生风险的概率是由该时刻与之前时刻的条件风险概率共同决定的,因此使不同的时间风险预测概率产生了时间上的关联性.

第二部分损失如式(16)所示:

$$Loss_2 = -\sum_{i=1}^{N} \left(\sum_{j\neq i} A_{e,i,j} \times (\log\hat{F}_e(s^{(i)} \mid x^{(i)}) + \log(1 - \hat{F}_e(s^{(i)} \mid x^{(j)})))\right). \tag{16}$$

通过 $A_{e,i,j}$ 筛选出满足相应条件的样本对(i,j),最大化样本 i 发生风险的累计概率,最小化样本 j 发生风险的累计概率. 该损失函数的作用是调整累计风险概率,使模型的预测结果符合 C^{td} 指数. 该损失函数与 Deephit 模型 Loss2 不同之处在于,Loss2 目的是通过扩大两个累计概率的差异达到符合 C^{td} 指数的目的,但是无法合理的衡量两个累计概率之间差异的大小,Loss2 通过分别衡量样本 i 和样本 j 的累积概率达到符合 C^{td} 指数的目的,避免产生类似 Loss2 的问题.

Loss1 和 Loss2 由平衡权重的参数 θ 联系在一起,参数 θ 的取值范围为[0,1]. Loss1 和 Loss2 用式(17)联合表示.

$$Loss_{1+2}=\theta\times Loss_1+(1-\theta)\times Loss_2. \tag{17}$$

第三部分损失采用 Deephit 模型的 Loss2 损失函数,即 $Loss_3=Loss_2$,主要是为了保留该损失函数可能带来的性能提升. 最终的损失函数如式(18)所示:

$$Loss=Loss_{1+2}+Loss_3. \tag{18}$$

4　数据集介绍

本次实验用到四个数据集:Synthetic 数据集[27]、Support 数据集[9]、Nwtco 数据集[9]和 Metabric 数据集[9]. 这些数据集包含风险和时间在内的多种信息,其中 Synthetic 是一个人工合成的数据集,该数据集含有两个风险,样本总量为 30 000,其中删失样本占 50%. Support 是关于 9 000 多个重病患者的单风险数据集,通过随机采样选取 8 500 个样本进行本次实验. 该数据集包括性别、年龄、种族、心率和体温等 14 项特征信息. 选取的 8 500 个样本中,发生删失的样本有 2 706 个,占所有选取样本的 31.84%. Nwtco 数据集含有 4 000 个样本,删失样本占 85.75%,占比为四个数据集中最高的. 该数据集是单风险数据集,数据集记录的特征包括组织学信息、疾病分期阶段和年龄等. Metabric 用于乳腺癌病人风险研究,其中包含 2 000 个样本,删失样本占 55.25%,其中包含性别、年龄、血压和基因信息等 21 个特征.

5　实验与结果分析

本次实验中的对比实验包括 Cox 及其改进模型[9-27]、PMF[9]、Nnet-survival[15]、N-MTLR[28]、RSF 和 Deephit 等多种模型.

为了使 DeephitTM 模型与 Deephit 模型的性能对比更有说服力,本文模型神经元的设置与 Deephit 模型中神经元大致相同. 设置 Sub-network1 为单层神经网络,神经元个数是输入维度的 3 倍,激活函数为 relu 函数. Sub-network2 为三层神经网络,各层神经元的个数设置为输入维度的 5 倍、5 倍和 3 倍,激活函数为 relu 函数,其他参数设置如表 1 所示. 其中 batch-size 为每次训练模型输入的样本数量,一般数据集训练时设置 dropout 来防止模型过拟合,根据不同数据集的具体情况增减 dropout,模型参数初始化方式为 Xavier 初始化,优化算法采用 AdamOptimizer 进行梯度下降,学习率设为 le-3 或者 1e-4 来调整模型训练速度. 最后采用 C^{td} 指数作为性能评估指标.

表 1　参数选取

Table 1　Parameter selection

参数	值或范围
Batch_size	50
Dropout	0.5/0.6/0.7
模型参数初始化	Xavier 初始化
优化算法	AdamOptimizer
Learning_rate	1e-3/1e-4
θ	(0,1)

表 1 中的初始化算法和梯度优化算法采用 Deephit 模型相关设置,其他参数可基于模型实验进行类似网格搜索的超参数微调. 根据模型训练过程过拟合或欠拟合程度可选取合适的超参数 batch_size 和 Dropout 防止过拟合和欠拟合现象,根据模型是否因为步长过大产生模型震荡或者步长过小导致模型训练缓慢选取合适的学习率. θ 是衡量不同损失函数重要性的参数,取值范围为 0~1,可通过验证集来确定取值.

5.1　合成数据集下的实验结果对比

本文采用合成数据集 Synthetic 比较多个竞争风险下模型的性能,Synthetic 数据集含有两个竞争风险 event1 和 event2. 在这个数据集下,DeephitTM 模型与其他模型的 C^{td} 对比如表 2 所示.

表 2　Synthetic 数据集下不同模型的性能对比

Table 2　The performance comparison of different models under Synthetic dataset

模型	Event1	Event2	模型	Event1	Event2
Cox	0.571	0.597	Deephit($\alpha=0$)	0.746	0.742
CoxTime	0.732	0.720	Deephit	0.751	0.747
CoxCc	0.733	0.727	DeephitTM_KM($\alpha=0$)	0.748	0.747
Deepsurv	0.744	0.744	DeephitTM_KM	0.751	0.750
PCHazard	0.636	0.616	DeephitTM_FR($\alpha=0$)	0.751	0.749
PMF	0.732	0.733	DeephitTM_FR	0.754	0.752
Nnet-survival	0.739	0.737	DeephitTM_SIG($\alpha=0$)	0.752	0.750
N-MTLR	0.735	0.733	DeephitTM_SIG	0.756	0.751
RSF	0.702	0.700			

由表 2 可知,在 Synthetic 数据集的风险 Event1 和 Event2 下,三种 DeephitTM 模型相比于 Deephit 模型在内的其他模型性能均有提升,说明 DeephitTM 模型具有较强的拟合生存分析潜在概率分布的能力. 由于该数据集为合成数据集,相比于真实数据集,可能存在数据设计方面的缺陷导致,所以相比于 Deephit 模型的性能提升较小,大约提升了 0.5%.

5.2 单风险数据集下的实验结果对比

使用 Support、Metabric 和 Nwtco 等数据集研究单风险数据集. 各个模型在不同的数据集下训练得到的实验结果对比如表 3 所示.

首先对比 DeephitTM 与 Deephit 的性能,在 Support、Nwtco 及 Metabric 数据集上,两种模型均在 $\alpha \neq 0$ 时得到最佳性能,此时 DeephitTM_SIG 相比于 Deephit 性能提升分别为 0.80%、2.30%和 2.20%. Support 数据集上各种模型的性能普遍偏低,说明该数据集可能存在较多的噪声,模型可学习的生存分析信息少,因此在该数据集上 DeephitTM 模型性能的提升较小. Support 和 Metabric 数据集下 RSF 模型比 Deephit 性能强,DeephitTM_SIG 相比 RSF 模型性能的提升分别为 0.60%和 0.90%. 综合对比各个模型的性能可知,DeephitTM 的性能在三个数据集上的生存分析性能有 0.6%到 2.30%的不同程度的提升.

表 3 单风险数据集下模型性能对比

Table 3 Comparison of model performance under single risk datasets

模型	Support	Nwtco	Metabric	模型	Support	Nwtco	Metabric
Cox	0.559	0.703	0.650	Deephit($\alpha=0$)	0.614	0.725	0.654
CoxTime	0.618	0.687	0.644	Deephit	0.625	0.734	0.663
CoxCc	0.596	0.688	0.629	DeephitTM_KM($\alpha=0$)	0.623	0.723	0.650
Deepsurv	0.593	0.701	0.655	DeephitTM_KM	0.629	0.746	0.663
PCHazard	0.599	0.678	0.668	DeephitTM_FR($\alpha=0$)	0.618	0.723	0.658
PMF	0.520	0.632	0.659	DeephitTM_FR	0.630	0.747	0.669
Nnet-survival	0.537	0.622	0.665	DeephitTM_SIG($\alpha=0$)	0.617	0.756	0.680
N-MTLR	0.522	0.658	0.669	DeephitTM_SIG	0.633	0.757	0.685
RSF	0.627	0.696	0.676				

5.3 DeephitTM_SIG 与次优模型的性能对比

为了直观的对比 DeephitTM_SIG 模型和 RSF 模型性能差异和增加说服力,对两种模型在 Support 数据集和 Metabric 数据集下分别进行 100 次重复实验. DeephitTM_SIG 模型与 RSF 模型结果对比如图 4 和图 5 所示.

由于 RSF 模型每次的实验结果为固定值,因此以 RSF 模型性能作为基准线(即图中纵坐标为 0 的直线),红色散点分别表示每次 DeephitTM_SIG 模型实验结果相对于 RSF 模型性能的高低程度,当散点越高时性能越优于 RSF 模型,越低甚至在直线下方时,性能越差. 因此从图 4 和图 5 可知,DeephitTM 模型实验结果整体优于 RSF 模型.

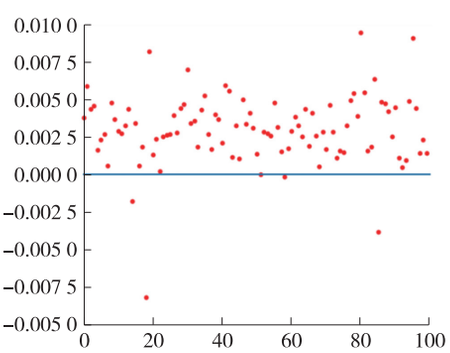


图 4 Support 数据集下 100 次实验结果对比

Fig. 4 Comparison of 100 experimental results in Support dataset

为了直观对比 Nwtco 数据集上 DeephitTM_SIG 模型与 Deephit 模型的 100 次实验结果,通过将两种模型的实验数据分别按性能从小到大排序并分别标号 1~100,然后每次以 Deephit 模型的结果作为横坐标,相同标号的 DeephitTM 模型作为纵坐标,得到如图 6 所示的连续曲线.

由上述曲线的生成方式可知,当 DeephitTM 模型的性能优于 Deephit 模型时,相应的横坐标值应当小于纵坐标值,曲线总体斜率大于 1;当 DeephitTM 模型的性能弱于 Deephit 模型时,相应的横坐标值应当大于纵坐标值,曲线总体斜率小于 1;当两种模型性能相近时,相应的横坐标值与纵坐标值应当基本一致,斜率接近于 1,即接近图中直线. 而图 6 中可以直观得到该曲线斜率高于斜率为 1 的直线,即 DeephitTM 模型具有更好的性能. 以 Deephit 模型的性能为基准,将不同模型的性能转化为相对于 Deephit 模型性能的百分比值得到如图 7 所示结果.

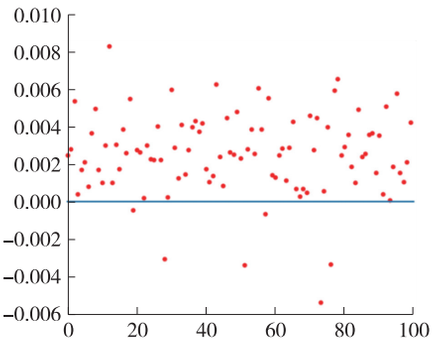


图 5　Metabric 数据集下 100 次实验结果对比

Fig. 5　Comparison of results of 100 experiments in Metabric dataset

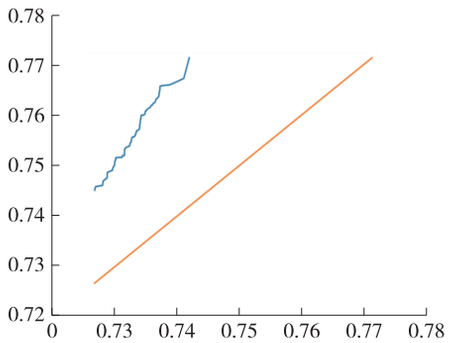
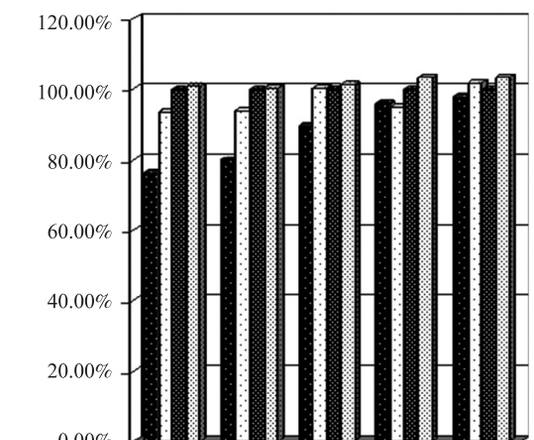


图 6　Nwtco 数据集下 100 次实验结果对比

Fig. 6　Comparison of results of 100 experiments in Nwtco dataset



	Synth_event1	Synth_event2	Support	Nwtco	Metabric
Cox	76.03%	79.92%	89.44%	95.78%	98.04%
RSF	93.48%	93.71%	100.32	94.82%	101.96%
Deephit	100%	100%	100%	100%	100%
DephitTM	100.67	100.54	101.28	103.13	103.32

图 7　各个模型在不同数据集下的比较

Fig. 7　Comparison of various models in different data sets

通过在几种医学数据集上的多种模型的实验,结合图 10 中的柱状图及数据表 1、表 2、表 3 以及图 4 到图 7 的分析,说明 DeephitTM 性能高于其他模型的性能,通过改进模型学习生存分析问题中的时间相关性确实提升了其在生存分析问题上的性能.

6　结论

本文主要是通过研究已有的生存分析方法,通过对现有模型进行改进,得到 DeephitTM 模型,该模型可以学习医学生存分析问题中的时间相关性,提升生存分析性能,为相关领域进行生存分析提供性能更好的生存分析模型.

改进后的模型确实实现了性能的提升,但是在实验过程中发现,神经网络参数的初始化对模型性能也有一定的影响,虽然 Xavier 初始化算法相比于其他算法具有一定的性能提升,但是每次 Xavier 初始化得到的参数是不同的,初始化后参数的值会影响模型最终性能好坏,当固定初始化参数的值时,无法确定模型是否会得到最好的性能,虽然本次试验通过上百次实验可以防止可能出现的异常性能测试结果影响实验结果,但是如何在 Xavier 初始化时选取合适的模型初始化随机种子,进而得到稳定的最优性能是将要解决的问题.

[参考文献]

[1] COX D. Regression models and life tables[J]. Journal of the Royal Statistical Society series b:statistical methodology,1972,34(2):187-220.

[2] KAPLAN E L,MEIER P. Nonparametric estimation from in-complete observations[J]. Journal of the American Statal Association,1957,53(282):457-481.

[3] AARON S D,STEPHENSON A L,CAMERON D W,et al. A statistical model to predict one-year risk of death in patients with cystic fibrosis[J]. Journal of clinical epidemiology,2015,68(11):1336-1345.

[4] FARAGGI D,SIMON R. A neural network model for survival data[J]. Statistics in Medicine,2010,14(1):73-82.

[5] ISHWARAN H, KOGALUR U B, BLACKSTONE EUGENE H, et al. Random survival forests[J]. Journal of Thoracic Oncology Official Publication of the International Association for the Study of Lung Cancer,2008,2(12):841-860.

[6] ISHWARAN H,KOGALUR U B,BLACKSTONE EUGENE H,et al. Random survival forests for R[J]. Annals of applied statistics,2007,2(3):25-31.

[7] LUCK M,SYLVAIN T,CARDINAL H,et al. Deep learning for patient-specific kidney graft survival analysis[J/OL]. arXiv Preprint arXiv:1705.10245,2017.

[8] ALAA A M,van der SCHAAR M. Deep multi-task Gaussian processes for survival analysis with competing risks[C]//31st Annual Conference on Newral Information Processing System. Long Beach,CA:NIPS,2017,30:2326-2334.

[9] CHANG LEE, WILLIAM R ZAME, JINSUNG YOON, et al. DeepHit: a deep learning approach to survival analysis with competing risks[C]//The Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans,LA:AAAI,2018:2314-2321.

[10] PLSTERL S, NAVAB N, KATOUZIAN A. Fast training of support vector machines for survival analysis[C]//Machine Learning and Knowledge Discovery in Databases:European Conference,ECML PKDD. Switzerland:Springer,2015:243-259.

[11] PLSTERL S,NAVAB N,KATOUZIAN A. An efficient training algorithm for kernel survival support vector machines[J/OL]. arXiv Preprint arXiv:1611.07054,2016.

[12] PLSTERL S, GUPTA P, WANG L, et al. Heterogeneous ensembles for predicting survival of metastatic, castrate-resistant prostate cancer patients[J]. F1000research,2016,5(2676):1-29.

[13] KATZMAN J,SHAHAM U,BATES J,et al. Deep Survival:a deep cox proportional hazards network[J/OL]. arXiv Preprint arXiv:1606.00931,2016.

[14] LIU P,FU B,YANG S X,et al. Optimizing survival analysis of XGBoost for ties to predict disease progression of breast cancer[J]. IEEE transactions on biomedical engineering,2020,68(1):148-160.

[15] KATZMAN J L, SHAHAM U, CLONINGER A, et al. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network[J]. BMC medical research methodology,2018,18(1):24.

[16] LIM H J,ZHANG X,DYCK R,et al. Methods of competing risks analysis of end-stage renal disease and mortality among people with diabetes[J]. BMC medical research methodology,2010,10(97):1-9.

[17] LAMBERT P C,DICKMAN P W,NELSON C P,et al. Estimating the crude probability of death due to cancer and other causes using relative survival models[J]. Statistics in medicine,2010,29(7/8):885-895.

[18] BELLOT A, SCHAAR M. Tree-based bayesian mixture model for competing risks[C]//21st International Conference on Arttfical Intelligence and Statistics Lanzarote,Spain:Microcome Publishing,2018:910-918.

[19] REN K, QIN J, ZHENG L, et al. Deep recurrent survival analysis[J]. Proceedings of the AAAI conference on artificial intelligence,2019,33:4798-4805.

[20] LIU P,FU B,YANG S X. HitBoost:survival analysis via a multi-output gradient boosting decision tree method[J]. IEEE Access,2019,7:56785-56795.

[21] RIETSCHEL C,YOON J,MIHAELA V. Feature selection for survival analysis with competing risks using deep learning[J/OL]. arXiv Preprint arXiv:1811.09317,2018.

[22] LEE C,YOON J,SCHAAR M. Dynamic-DeepHit:a deep learning approach for dynamic survival analysis with competing risks based on longitudinal data[J]. IEEE transactions on biomedical engineering,2020,67(1):122-133.

[23] COLLOBERT R, WESTON J. A unified architecture for natural language processing: deep neural networks with multitask learning[C]//Machine Learning,Proceedings of the Twenty-Fifth International Conference(ICML 2008),Helsinki,Finland:ACM,2008:160-167.

[24] HE K,ZHANG X,REN S,et al. Deep residual learning for image recognition[C]//IEEE Conference on Computer Vision & Pattern Recognition. IEEE Computer Society,Piscataway,NJ:IEEE,2016:770-778.

[25] HARRELL,FRANK E. Evaluating the yield of medical tests[J]. The journal of the American Medical Association,1982,247(18):2543-2546.

[26] ANTOLINI L,BORACCHI P,BIGANZOLI E. A time-dependent dis-crimination index for survival data[J]. Statistics in medicine,2005,24(24):3927-3944.

[27] KVAMME H,BORGAN R,SCHEEL I. Time-to-event prediction with neural networks and cox regression[J]. Journal of machine learning research,2019,20(129):1-30.

[28] KVAMME H,BORGAN R. Continuous and discrete-time survival prediction with neural networks[J/OL]. arXiv Preprint arXiv:1910.06724,2019.

[责任编辑:杜忆忱]