Министерство образования и науки Российской Федерации
Московский физико-технический институт (государственный университет)

Физтех-школа прикладной математики и информатики
Кафедра алгоритмов и технологии программирования

Выпускная квалификационная работа магистра

Нейросетевые модели анализа выживаемости для решения задачи предсказания оттока абонентов

Автор:
Студент М05-313а группы
Батарин Егор Владиславович

Научный руководитель:
Джумакаев Тимур Казбекович

Москва 2025

Аннотация

Нейросетевые модели анализа выживаемости для решения задачи предсказания оттока абонентов

Батарин Егор Владиславович

В работе решается задача прогнозирования оттока абонентов компании Мегафон. Проведен анализ существующих решений из моделей анализа выживаемости, кратко описаны их отличия и проведены сравнения данных моделей на задаче оттока клиентов Мегафона.

Задача рассматривается как многоклассовая классификация, где в качестве меток класса выбраны факты оттока в будущие месяцы и факт отсутствия оттока в эти месяцы. Предлагаются различные подходы к решению задачи, как классические подходы: градиентный бустинг и модель Кокса, так и более современные подходы, связанные с применением методов глубокого обучения в моделях выживаемости. Проводится сравнение различных подходов с точки зрения принятых в работе критериев качества.

В рамках работы была разработана модель, отличающаяся от предыдущих использованием временных рядов векторов признаков. Основная новизна работы заключается в использовании специфической функции потерь. В работе доказывается свойство добавки к функции потерь. Модель, разработанная в данной работе, показала более высокие результаты по сравнению с другими моделями в плане более низкой потери качества на "сжатом"признаковом описании и более высоким C-индексом.

Оглавление

Введение

Данная работа посвящена решению задачи прогнозирования оттока в контексте телекоммуникационной отрасли в рамках проекта компании Мегафон. Данная задача позникает в большом количестве различных индустрий и для ее решения используются разные подходы. [1]. Среди всех таких подходов, включающих в себя разные современные методы машинного обучения (SVM, Random Forest, Gradient Boosting) основным направлением исследований для решения данной задачи был выбран анализ выживаемости.

Классической работой по анализу выживаемости является модель пропорциональных рисков [2]. В настоящее время появилось много новых подходов, так или иначе задействующих глубокое обучение [3]. Эти подходы расширяют классические модели анализа выживаемости, позволяя обойти некоторые предположения о данных, которые на практике редко выполняются, например, предположение о пропорциональности рисков. Все модели анализа выживаемости можно разделить на две категории с зависимости от того, считается ли время непрерывным или дискретным. Поскольку специфика проекта Мегафона требует рассмотрения дискретного времени, то именно такой случай фигурирует в постановке задачи.

Одной из известных дискретных моделей выживаемости является модель DeepHit [4], являюшейся одной из основных для сравнения в дискретной постановке задачи.

Одним из самых распространенных подходов к задаче является градиентый бустинг [5]. В данной работе в качестве базового подхода рассматривается категориальный бустинг от компании Яндекс [6]. Он получил большое распространение на российском рынке, поэтому исследуемые в работе подходы, основанные на анализе выживаемости, сравниваются с CatBoost.

В работе используются внутренние датасеты компании Мегафон, собранные на основе данных в КХД. В роли критериев качества модели

выступают метрики Precision, Recall, F1, вычисленные при различных вероятностных порогах - числах, позволяющих перевести вероятности классов в метки классов.

Глава 1

Постановка задачи

1.1 Общая постановка задачи анализа выживаемости

В дискретном случае время имеет вид $\mathcal{T} = \{0, \ldots, T_{\max}\}$, где $T_{\max}$ - это максимальный горизонт предсказания (например, максимально возможное время жизни абонента). В любой их этих моментов может произойти событие $\mathcal{K} = \{\emptyset, 1, \cdots, K\}$, где все события $\{1, \cdots, K\}$ соответствуют факту оттока по одной из K возможных причин в некоторый момент времени τ, а событие $\emptyset$ означает факт правого цензурирования - информация о том, что отток произойдет не раньше того времени τ, когда произошло цензурирование, но точно неизвестно когда именно. Для каждого момента времени, таким образом, мы можем написать $\tau^i = \min(T^i, C^i)$, где $T^i \in \mathcal{T}$ - это времена наступлений одного из событий $\{1, \cdots, K\}$, а $C^i \in \mathcal{T}$ соответствует право-цензурированным событиям $\emptyset$. Имея в распоряжении информацию о произошедших событиях (включая цензурирования) $\{\tau^i, k^i)\}_{i=1}^N$ и некоторую дополнительную информацию о признаках абонентов, мы хотим научиться предсказывать вероятности наступления события из $\mathcal{K}$ в будущем. В зависимости от того, рассматриваем ли мы признаки абонентов только в один момент времени или рассматриваем для каждого абонента временной ряд соотвествующих ему признаков, будет немного различаться математическая постановка задачи.

Непрерывный случай отличается от дискретного тем, что время представляет из себя отрезок

1.2 C-индекс и его связь с ранжированием абонентов

Специфичным для анализа выживаемости критерием качества является C-индекс (concordante index) [7], являющийся аналогом широкого известного критерия качества AUC. Дадим его строгое определение:

Определение 1. *Пусть для пары абонентов* (i,j) *определены* $\tau_i < \tau_j$ *- моменты времени (возможно, цензурированные),* δ_i *- индекс цензурирования, равный* 0*, если* τ_i *право-цензурировано и* 1 *- в противном случае. Также обозначим через* $\hat{F}_{k,i}$ *и* $\hat{F}_{k,j}$ *вероятности оттока до момента по событию* k *времени* τ_i *для абонентов* i *и* j *соответственно (оцененные моделью функции распределения). Тогда C-индекс определяется следующим образом:*

$$C\text{-}index = \frac{\sum\limits_{i,j} \mathbf{1}_{\tau_i<\tau_j} \cdot \mathbf{1}_{\hat{F}_{k,i}>\hat{F}_{k,j}} \cdot \delta_i}{\sum\limits_{i,j} \mathbf{1}_{\tau_i<\tau_j} \cdot \delta_i}$$

Нам также понадобится определение верно упорядоченной пары абонентов:

Определение 2. *Пара абонентов* (i,j) *считается верно упорядоченной (отранжированной), если для оцененных моделью функций распределений* $\hat{F}_{k,i}$ *и* $\hat{F}_{k,j}$ *выполнено неравенство:*

$$\hat{F}_{k,i}(\tau_i) > \hat{F}_{k,j}(\tau_i)$$

Он численно равен доле пар абонентов, которые модель верно упорядочила.

Этот индекс отражает то разумное требование к моделям выживаемости, которое заключается в том, что если абонент i оттекает раньше абонента j, то оцениваемая моделью вероятность его оттока до момента его фактического оттока будет больше, чем аналогичная вероятность для j абонента, рассматриваемая для того же момента времени - фактического оттока i-ого абонента.

1.3 Постановка с одним вектором-признаком

В данной модели предполагается, что абонент полностью описывается вектором $\mathbf{x} \in X$, соответственно обучающая выборка имеет вид $\mathcal{D} = \{(\mathbf{x}^{(i)},\tau^{(i)},k^{(i)})\}_{i=1}^{N}$, вероятности нецензурированных событий $k^* \neq \emptyset$ имеют вид $P(\tau = \tau^*,k = k^*|\mathbf{x} = \mathbf{x}^*)$, а функция распределения имеет вид:

$$F_{k^*}(t^*|\mathbf{x}^*) = P(\tau \leq t^*,k = k^*|\mathbf{x} = \mathbf{x}^*) = \sum_{\tau^*=0}^{t^*} P(\tau = \tau^*,k = k^*|\mathbf{x} = \mathbf{x}^*). \quad (1.1)$$

1.4 Поставнока в временным рядом векторов-признаков

Эта модель является расширением предыдущей и для каждого абонента i в ней рассматривается уже не единственный вектор-признак $\mathbf{x}_i$, а временной ряд векторов-признаков:

$$\mathcal{X}^i(t) = \{\mathbf{x}^i(t^i_j) : 0 \leq t^i_j \leq t \text{ for } j = 1,\cdots,J^i\}$$

, где $\mathbf{x}^i(t_j)$ - это вектор признак i-ого абонента, замеренный в момент времени t_j и равный $\mathbf{x}^i_j = [x^i_{j,1},\cdots,x^i_{j,d_r}]$. Кроме того, в этой модели для каждого абонента i вводится набор векторов-флагов $\mathrm{M}^i = \{\mathrm{m}^i_1,\cdots,\mathrm{m}^i_{J^i}\}$, $\mathbf{m}^i_j = [m^i_{j,1},\cdots,m^i_{j,d_x}]$, которые сигнализируют о пропущенных значениях: $m^i_{j,d} = 1$ тогда и только тогда, когда $x^i_{j,d}$ пропущено, иначе $m^i_{j,d} = 0$. Напоследок, вводится последовательность векторов временных интервалов между замерами времени $\Delta_i = \{\delta^i_1,\delta^i_2\cdots,\delta^i_{J^i}\}$, где $\delta^i_j = t^i_{j+1} - t^i_j$ для всех $1 \leq j < J^i$ и $\delta^i_{J^i} = 0$. В итоге получается обучающая выборка: $\mathcal{D} = \{(\mathbf{X}^i,\mathbf{M}^i,\Delta^i,\tau^i,k^i)\}_{i=1}^{N}$.

Глава 2

Существующие подходы в анализе выживаемости

2.1 Непрерывные модели выживаемости

2.1.1 Оценка Каплана–Майера

Оценка Каплана–Майера [8] — это непараметрический метод оценки функции выживания $S(t)$ на основе наблюдаемых данных. Пусть для выборки из N объектов известны времена τ^i и индикаторы события $\delta^i \in \{0,1\}$, где $\delta^i = 1$ означает, что событие (например, отток) произошло, и $\delta^i = 0$ — что наблюдение было право-цензурировано. Тогда функция выживания оценивается как

$$\hat{S}(t) = \prod_{t_i \leq t} \left(1 - \frac{d_i}{n_i}\right), \tag{2.1}$$

где:

- t_i — моменты наступления событий в порядке возрастания,
- d_i — число событий (нецензурированных) в момент t_i,
- n_i — число объектов, находившихся под наблюдением непосредственно до t_i.

Эта оценка хорошо работает для визуального анализа, но не использует признаковые вектора $\mathbf{x}^i$ и не позволяет делать индивидуализированные предсказания.

2.1.2 Модель пропорциональных рисков Кокса

Модель Кокса [2] является полупараметрической моделью, в которой предполагается, что функция риска (hazard function) имеет вид:

$$h(t|\mathbf{x}) = h_0(t) \cdot \exp(\mathbf{x}^\top \boldsymbol{\beta}), \tag{2.2}$$

где:

- $h_0(t)$ — базовая (неизвестная) функция риска,
- $\boldsymbol{\beta}$ — вектор коэффициентов, обучаемых по данным.

Функция выживания тогда выражается как:

$$S(t|\mathbf{x}) = S_0(t)^{\exp(\mathbf{x}^\top \boldsymbol{\beta})}, \tag{2.3}$$

где $S_0(t)$ — функция выживания, соответствующая базовой функции риска.

Параметры $\boldsymbol{\beta}$ оцениваются методом максимизации частичной функции правдоподобия:

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i:\delta^i=1} \frac{\exp(\mathbf{x}^{(i)\top}\boldsymbol{\beta})}{\sum_{j:\tau^j \geq \tau^i} \exp(\mathbf{x}^{(j)\top}\boldsymbol{\beta})}. \tag{2.4}$$

Модель хорошо интерпретируема, но накладывает ограничение пропорциональности рисков и плохо применима при высокоразмерных или нелинейных признаках.

2.1.3 DeepSurv

DeepSurv [9] — это нейросетевое расширение модели Кокса. Вместо линейной зависимости $\mathbf{x}^\top \boldsymbol{\beta}$ используется произвольная нелинейная функция, моделируемая нейросетью $f_\theta(\mathbf{x})$:

$$h(t|\mathbf{x}) = h_0(t) \cdot \exp(f_\theta(\mathbf{x})). \tag{2.5}$$

Модель обучается аналогично классической модели Кокса — путём максимизации частичной функции правдоподобия:

$$\mathcal{L}(\theta) = -\sum_{i:\delta^i=1} \left[f_\theta(\mathbf{x}^i) - \log \sum_{j:\tau^j \geq \tau^i} \exp(f_\theta(\mathbf{x}^j)) \right]. \tag{2.6}$$

Таким образом, DeepSurv сохраняет достоинства модели Кокса (работа с цензурированными данными), но благодаря использованию нейросетей способна учитывать сложные нелинейные зависимости между признаками и риском.

2.2 Дискретные модели выживаемости

2.2.1 Nnet-survival

Nnet-survival [10] — это дискретная нейросетевая модель, в которой время разделяется на интервалы $[0, t_1), [t_1, t_2), \ldots, [t_{M-1}, t_M)$, и на каждом из них предсказывается условная вероятность наступления события.

Модель предсказывает условную вероятность $q_m = P(T \in [t_{m-1}, t_m) \mid T \geq t_{m-1}, \mathbf{x})$ с помощью softmax-выхода нейросети. Далее рассчитывается функция выживания:

$$\hat{S}(t_m|\mathbf{x}) = \prod_{l=1}^{m}(1 - q_l). \tag{2.7}$$

Функция распределения, в свою очередь:

$$\hat{F}(t_m|\mathbf{x}) = 1 - \hat{S}(t_m|\mathbf{x}). \tag{2.8}$$

Функция потерь имеет вид отрицательного логарифма правдоподобия:

$$\mathcal{L} = -\sum_{i=1}^{N} \log P(\tau^i, \delta^i|\mathbf{x}^i), \tag{2.9}$$

где в зависимости от того, цензурировано наблюдение или нет, вероятность имеет разный вид (используется либо q_m, либо $\hat{S}(t_m|\mathbf{x}^i)$).

Модель является более гибкой, чем классические методы, и хорошо работает при большом числе наблюдений и цензурировании, однако требует выбора разбиения времени.

2.2.2 PMF (Probability Mass Function)

PMF-модель подходит для многоклассовой дискретной предсказательной задачи. Модель предсказывает полную функцию распределения вероятностей событий по времени:

$$\mathbf{p} = [p_1, p_2, \ldots, p_T], \quad \sum_{t=1}^{T} p_t = 1, \tag{2.10}$$

где $p_t = P(T = t \mid \mathbf{x})$ — вероятность наступления события в момент t. Функция выживания тогда записывается как

$$\hat{S}(t|\mathbf{x}) = \sum_{j=t+1}^{T} p_j. \tag{2.11}$$

А функция распределения:

$$\hat{F}(t|\mathbf{x}) = \sum_{j=1}^{t} p_j. \tag{2.12}$$

Функция потерь — логарифмическое правдоподобие, аналогично Nnet-survival, но с прямым использованием вероятностей p_t:

$$\mathcal{L} = -\sum_{i=1}^{N} \left[\delta^i \cdot \log p_{\tau^i}^{(i)} + (1 - \delta^i) \cdot \log \hat{S}(\tau^i|\mathbf{x}^i) \right]. \tag{2.13}$$

Преимуществом модели является простота и интерпретируемость, а также прямая оценка распределения времени до события. Однако, как и в Nnet-survival, требуется дискретизация времени, что может снизить точность при неудачном разбиении.

2.2.3 Модель DeepHit

Поскольку теоретическая функция распределения неизвестна, то рассматривается ее оценка

$$\hat{F}_{k^*}(\tau^*|\mathbf{x}^*) = \sum_{m=0}^{\tau^*} o_{k,m}^* \tag{2.14}$$

на основе ответов модели DeepHit: $\mathbf{o} = [o_{1,1}, \cdots, o_{1,T_{\max}}, \cdots, o_{K,1}, \cdots, o_{K,T_{\max}}]$, где $o_{k,\tau}$ - оценка моделью DeepHit вероятности того, что событие оттока k произойдет в момент времени τ.

В качестве функции потерь используется сумма двух слагаемых $\mathcal{L}_{\text{Total}} = \mathcal{L}_1 + \mathcal{L}_2$, в которой первое слагаемое имеет вид:

$$\mathcal{L}_1 = -\sum_{i=1}^{N} \left[\mathbb{1}(k^{(i)} \neq \emptyset) \cdot \log \left(y^{(i)}_{k^{(i)},\tau^{(i)}} \right) + \mathbb{1}(k^{(i)} = \emptyset) \cdot \log \left(1 - \sum_{k=1}^{K} \hat{F}_k(\tau^{(i)}|\mathbf{x}^{(i)}) \right) \right] \tag{2.15}$$

и оно отвечает за логарифмическое правдоподобие, а второе слагаемое имеет вид:

$$\mathcal{L}_2 = \sum_{k=1}^{K} \alpha_k \cdot \sum_{i \neq j} A_{k,i,j} \cdot \eta \left(\hat{F}_k(\tau^{(i)}|\mathbf{x}^{(i)}), \hat{F}_k(\tau^{(i)}|\mathbf{x}^{(j)}) \right) \tag{2.16}$$

где $A_{k,i,j} = \mathbb{I}(k^{(i)} = k, \tau^{(i)} < \tau^{(j)})$ - индикатор того, что событие k наступает для j-ого абонента позже, чем для i-ого и функция $\eta(x,y) = \exp\left(\frac{-(x-y)}{\sigma}\right)$.

Глава 3

Описание подхода, разработанного в данной работе

3.1 Решение задачи для временного ряда векторов-признаков

Теоретическая функция распределения в нашей модели принимает вид:

$$\begin{aligned} F_{k^*}(\tau^*|\mathcal{X}^*) &= P(T \leq \tau^*, k = k^*|\mathcal{X}^*, T > t^*_{J^*}) = \\ &= \sum_{\tau \leq \tau^*} P(T = \tau, k = k^*|\mathcal{X}^*, T > t^*_{J^*}). \end{aligned} \tag{3.1}$$

Теоретическая функция выживания вычисляется следующим образом:

$$\begin{aligned} S(\tau^*|\mathcal{X}^*) &= P(T > \tau^*|\mathcal{X}^*, T > t^*_{J^*}) = \\ &= 1 - \sum_{k \neq \emptyset} F_k(\tau^*|\mathcal{X}^*) \end{aligned} \tag{3.2}$$

Поскольку теоретические функции неизвестны, мы можем пользоваться только оценочными. Оценочная функция распределения выражается через ответы модели:

$$\hat{F}_{k^*}(\tau^*|\mathcal{X}^*) = \frac{\sum_{t^*_{J^*} < \tau \leq \tau^*} o^*_{k^*,\tau}}{1 - \sum_{k \neq \emptyset} \sum_{n \leq t^*_{J^*}} o^*_{k,n}} \tag{3.3}$$

Функция потерь состоит из трех частей: $\mathcal{L}_{\text{Total}} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3$. Слагаемые $\mathcal{L}_1$ и $\mathcal{L}_2$ аналогичны соответствующим слагаемым из модели DeepHit и имеют вид:

$$\mathcal{L}_1 = -\sum_{i=1}^{N}\left[\mathbb{1}(k^i \neq \emptyset)\cdot\log\left(\frac{o^i_{k^i,\tau^i}}{1-\sum_{k\neq\emptyset}\sum_{n\leq t^i_{Ji}} o^i_{k,n}}\right)\right.$$
$$\left. + \mathbb{1}(k^i = \emptyset)\cdot\log\left(1-\sum_{k\neq\emptyset}\hat{F}_k(\tau^i|\mathcal{X}^i)\right)\right] \tag{3.4}$$

$$\mathcal{L}_2 = \sum_{k=1}^{K}\alpha_k\sum_{i\neq j}A_{kij}\cdot\eta\left(\hat{F}_k(s^i+t^i_{J^i}|\mathcal{X}^i),\hat{F}_k(s^i+t^j_{Jj}|\mathcal{X}^j)\right) \tag{3.5}$$

, где $s^i = \tau^i - t^i_{J^i}$, $A_{kij} = \mathbb{1}(k^i = k, s^i < s^j)$, $\eta(a,b) = \exp\left(-\frac{a-b}{\sigma}\right)$

Третье слагаемое в общей функции потерь является новым и отвечает за регуляризацию временных рядов:

$$\mathcal{L}_3 = \beta\cdot\sum_{i=1}^{N}\sum_{j=0}^{J^i-1}\sum_{d\in\mathcal{I}}(1-m^i_{j+1,d})\cdot\zeta(x^i_{j+1,d}, y^i_{j,d}) \tag{3.6}$$

Здесь $\mathcal{I}$ определяет подмножество зависящих от времени признаков абонентов, по которым мы хотим провести регуляризацию.

3.2 Описание архитектуры модели

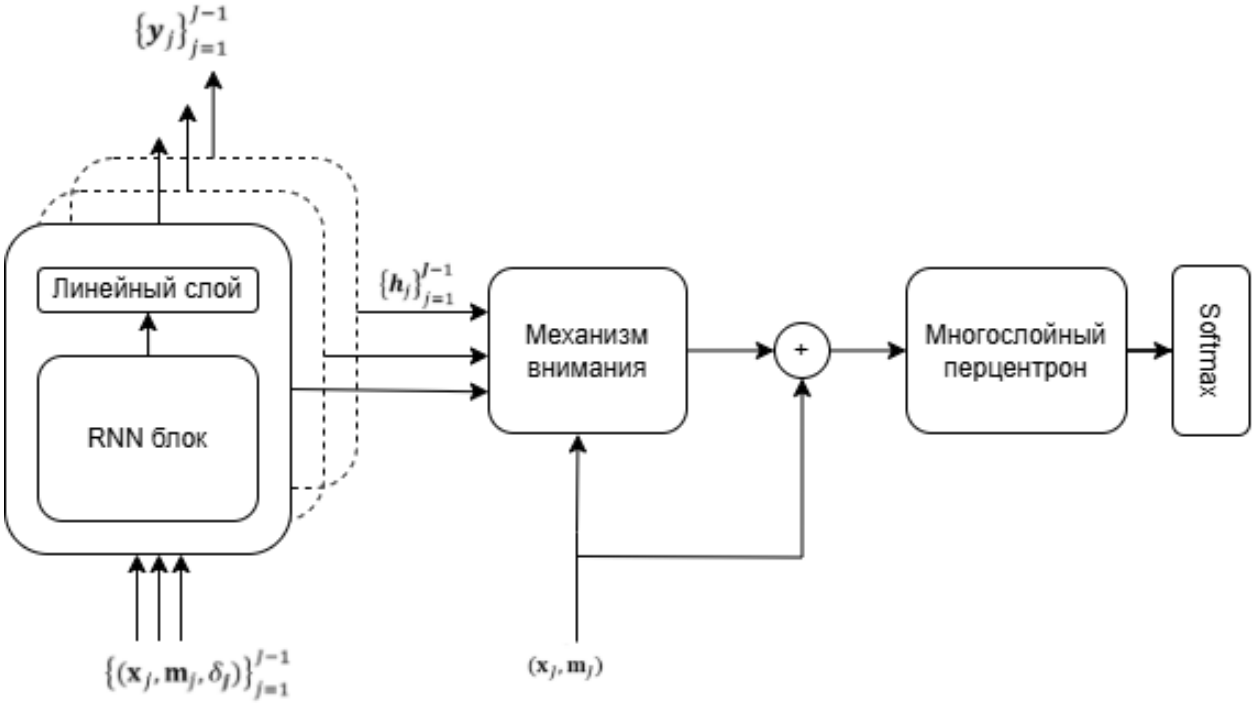

Рис. 3.1: Архитектура модели

Наша модель состоит из трех частей:

1. RNN блок - обрабатывает входящую последовательно векторов признаков на разных временных срезах

2. Механизм внимания выделяет ключевую информацию из последовательности

3. Многослойный перцептрон с Softmax агрегирует информацию в вектор вероятностей

Подобная архитектура используется во многих моделях выживаемости и хорошо себя зарекомендовала. Наш результат, таким образом, состоит не в новой архитектуре, а новой функции потерь. Мы далее покажем одно из основных свойств добавки $\mathcal{L}_2$ этой функции потерь.

3.3 Свойства добавки $\mathcal{L}_2$

Выше мы упомянули, что $\mathcal{L}_2$ обладает ранжирующим свойством с точки зрения C-индекса. Сформулируем строго это утверждение и докажем его для двух вышеописанных постановок задач. Прежде введем понятие отступа, которое аналогично по смыслу понятию отступа в задаче классификации:

Определение 3. *Пусть* (i,j) *- пара абонентов, для которых произошло событие оттока* k *в моменты времени* $\tau^{(i)}$ *и* $\tau^{(j)}$ *соответственно, причем* $\tau^{(i)} < \tau^{(j)}$*. Тогда в случае постановки с одним вектором-признаком отступ* $M_{k,i,j}$ *определяется как:*

$$M_{k,i,j} = \hat{F}_k(\tau^{(i)}|\mathbf{x}^{(i)}) - \hat{F}_k(\tau^{(i)}|\mathbf{x}^{(j)})$$

С случае постановки для временного ряда векторов-признаков отступ $M_{k,i,j}$ *определяется как:*

$$M_{k,i,j} = \hat{F}_k(s^i + t^i_{J^i}|\mathcal{X}^i) - \hat{F}_k(s^i + t^j_{J^j}|\mathcal{X}^j)$$

В задаче классификации отступ является мерой того, насколько уверенно модель правильно классифицирует объекты. В нашей постановке анализа выживаемости отступ является мерой того, насколько модель уверенно правильно упорядочивает абонентов. Теоремы ниже являются точным выражением того, что добавка $\mathcal{L}_2$ тем ниже, чем выше отступы $M_{k,i,j}$.

Теорема 1. *Пусть* $\mathcal{L}_2 = \sum_{k=1}^{K} \alpha_k \cdot \sum_{i \neq j} A_{k,i,j} \cdot \eta \left(\hat{F}_k(\tau^{(i)}|\mathbf{x}^{(i)}), \hat{F}_k(\tau^{(i)}|\mathbf{x}^{(j)}) \right)$ *- добавка к функции потерь в смысле постановки с одним вектором-признаком. Тогда* $\mathcal{L}_2$ *является убывающей от отступов функцией.*

Теорема 2. *Пусть* $\mathcal{L}_2 = \sum_{k=1}^{K} \alpha_k \cdot \sum_{i \neq j} A_{k,i,j} \cdot \eta \left(\hat{F}_k(s^i + t^i_{J^i}|\mathcal{X}^i), \hat{F}_k(s^i + t^j_{J^j}|\mathcal{X}^j) \right)$ *- добавка к функции потерь в смысле постановки для временного ряда векторов-признаков. Тогда* $\mathcal{L}_2$ *является убывающей от отступов функцией.*

Доказательство. В обеих постановках добавка принимает вид:

$$\mathcal{L}_2 = \sum_{k=1}^{K} \alpha_k \cdot \sum_{i \neq j} A_{k,i,j} \cdot \exp\left(-\frac{M_{k,i,j}}{\sigma} \right)$$

поэтому выражения для производных принимают вид:

$$\frac{\partial \mathcal{L}_2}{\partial M_{k,i,j}} = -\frac{1}{\sigma} \sum_{k=1}^{K} \alpha_k \cdot A_{k,i,j} \cdot \exp\left(-\frac{M_{k,i,j}}{\sigma} \right) < 0$$

откуда следуют утверждения теорем. □

Таким образом, смысл добавки $\mathcal{L}_2$ сводится к штрафованию модели появлению маленьких отступов, которые связаны с нарушением верной упорядоченности абонентов и понижением C-индекса.

Глава 4

Численные эксперименты

4.1 Описание постановки задачи

В численных экспериментах рассматривался частный случай общей поставноки задачи при котором $K = 1$ - события оттока не различаются между собой. При этом производительность моделей сравнивалась на задачах бинарной и многоклассовой классификации:

1. В задаче бинарной классификации предсказываются вероятности двух классов - сохранение абонента в следующий месяц (таргет $q = 0$) и отток абонента в следующий месяц (таргет $q = 1$)

2. В задаче многоклассовой классификации предсказываются вероятности следующих четырех классов: отток в конец текущего месяца (таргет $q = 0$), отток в конец следующего месяца (таргет $q = 1$), отток в конец послеследующего месяца (таргет $q = 3$) и сохранение абонента на конец послеследующего месяца (таргет $q = 4$). Новые признаки, генерируемые моделью, представляют собой вектор $(x_0,x_1,x_2,x_3,x_4) \in \mathbb{R}^5$, где x_i - вероятность выживания в i-ый месяц, где за 0 месяц берется июнь 2024

4.2 Описание данных

Эксперименты проводились на данных компании Мегафон, собранных с начала апреля 2024 до конца октября 2024.

Обучающая выборка состоит из 2.7 миллионов примеров, валидационная и тестовая выборки содержат по 600 и 800 тыс. обучающих примеров соответственно.

Для модели DeepHit обучающая выборка предварительно была нормирована. Для CatBoost нормировка не проводилась. Для подбора гиперпараметров CatBoost была использована библиотека optuna. Для архитектуры DeepHit был использован многослойный перцептрон.

4.3 Критерии качества

Для анализа качества была использована функция reports из модуля scoring, которая строит отчеты-таблицы по моделям. Отчет-таблица демонстирует, насколько хорошо модель отделяет каждый из классов от всех остальных (One-vs-All подход). Опишем структуру этого отчета. Для начала введем понятие топ перцентиля:

Определение 4. *Топ-p перцентиль для некоторого класса q определяется как квантиль уровня $1 - p$ для предсказанных моделью вероятностей класса q.*

В рамках One-vs-All подхода, по определению будем считать, что если предсказанная моделью вероятность класса q больше порога в топ-p перцентиль для некоторого заранее фиксированного p, то ответом модели будет класс q, иначе ответом будут все классы, кроме q.

В случае задачи бинарной классификации рассматриваются только топ перцентили для класса $q = 1$, соответствующему оттоку абонента, а в случае многоклассовой классификации - топ-перцентили для всех классов. При этом в многоклассовой постановке каждому классу соответствует свой отчет, показывающий, как хорошо данный класс отделяется от всех остальных.

Стоблцы таблицы соответствуют широко распространенным метрикам классификации - Precision, Recall, F1 и AUC. Строки таблицы соответствуют различным топ-p перцентилям для данного класса. Таким образом можно сравнивать метрики в зависимости от различных порогов вероятностей.

4.4 Метод подбора гиперпараметров и шага обучения

4.4.1 Grid Search

Для подбора гиперпараметров использовался известный метод Grid Search. Его сущность заключается в следующем:

1. Определение сетки гиперпараметров: для каждого гиперпараметра задаются конкретные значения или диапазон значений.

2. Перебор всех комбинаций:алгоритм последовательно проверяет каждую возможную комбинацию гиперпараметров, создавая новую модель для каждой комбинации и оценивая её качество на заранее выбранном метричном показателе.

3. Оценка качества моделей: качество каждой модели оценивается на отдельной тестовой выборке.

4. Выбор лучшей комбинации: после проверки всех комбинаций выбирается та комбинация гиперпараметров, которая дала наилучшее значение целевой метрики качества.

4.4.2 Learning Rate Finder

Процедура Learning Rate Finder позволяет найти оптимальное значение шага обучения. Она работает следующим образом:

1. Выбор начальных и конечных границ: выбирается интервал скоростей обучения, в пределах которого будет проводиться исследование. Скорость обучения изменяется экспоненциально между начальной и конечной точкой. Это значит, что каждое следующее значение увеличивается по отношению к предыдущему на постоянный коэффициент.

2. Быстрое обучение и сбор статистики: модель быстро обучается на небольшом числе эпох или батчей, используя различные значения скорости обучения, вычисляемые на каждом шаге согласно формуле выше. На каждом этапе фиксируются потери модели.

3. Анализ результатов: полученная зависимость потерь от скорости обучения визуально отображается графиком. Оптимальной скоростью обучения считается точка, где потеря резко снижается, но ещё не начала увеличивать-

ся снова. Этот момент соответствует точке максимальной эффективности градиентного спуска.

4.5 DeepHit в задаче бинарной классификации

4.5.1 Число эпох обучения = 3

Для начала было использовано три эпохи обучения. График функции потерь имеет вид:

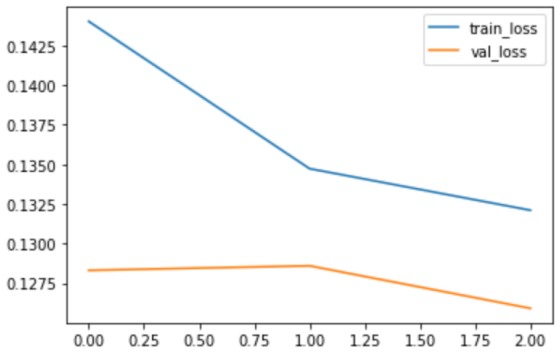

Рис. 4.1: Функция потерь

Ниже поедставлено сравнение метрик на ТОП 10% для DeepHit и CatBoost.

	metrics	precision	recall	f1	roc-auc
score date	percentile				
1	0.500	0.986	0.010	0.019	0.505
	1.000	0.982	0.019	0.038	0.510
	2.000	0.977	0.039	0.074	0.519
	3.000	0.975	0.058	0.109	0.528
	4.000	0.972	0.077	0.142	0.537
	5.000	0.969	0.096	0.174	0.546
	10.000	0.959	0.190	0.317	0.591

Рис. 4.2: Результаты DeepHit

	metrics	precision	recall	f1	roc-auc
score date	percentile				
1	0.500	0.997	0.010	0.020	0.505
	1.000	0.994	0.020	0.039	0.510
	2.000	0.992	0.039	0.076	0.519
	3.000	0.990	0.059	0.111	0.529
	4.000	0.988	0.078	0.145	0.539
	5.000	0.987	0.098	0.178	0.548
	10.000	0.979	0.194	0.323	0.595

Рис. 4.3: Результаты CatBoost

4.5.2 Число эпох обучения = 30

На этом этапе мы заметно увеличили число эпох обучения DeepHit. Такое число эпох оказалось оптимальным - исследования показали, что после 30 эпох происходит переобучение и функция потерь на валидационной выборке начинает вести себя нестабильно. Поэтому было принято решение установить число эпох не более 30.

4.6 DeepHit в задаче многоклассовой классификации

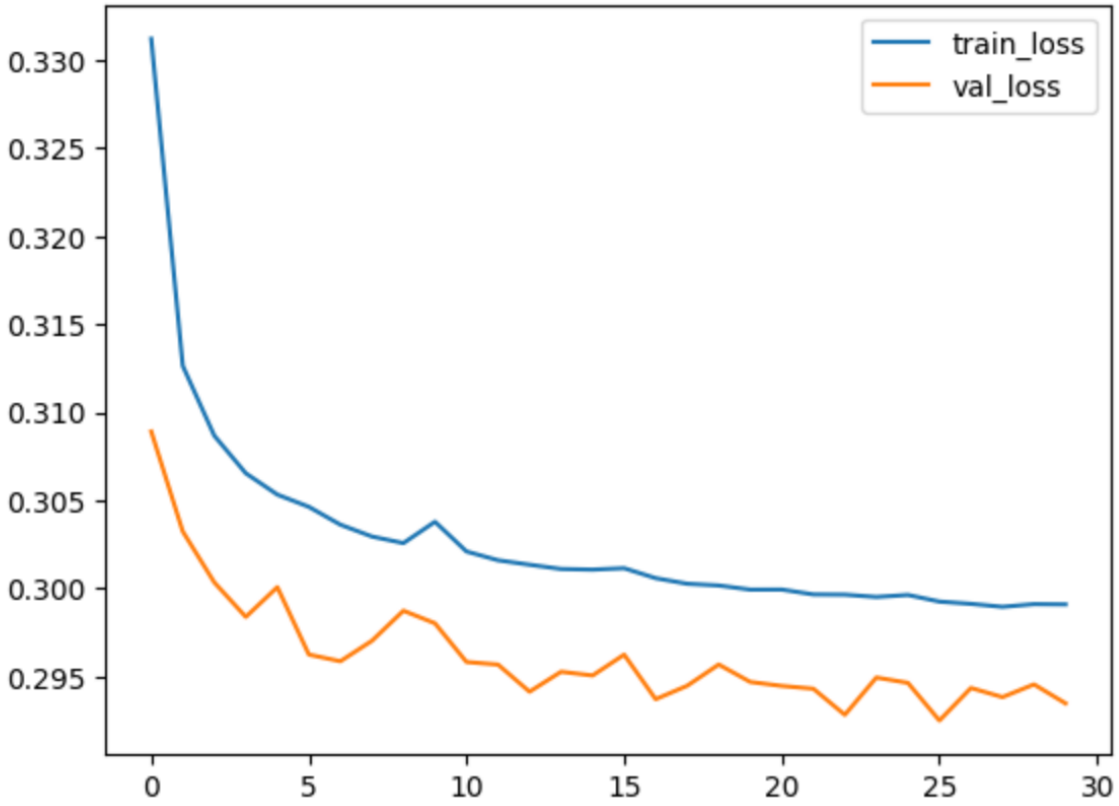

Рис. 4.4: Функция потерь для модели DeepHit

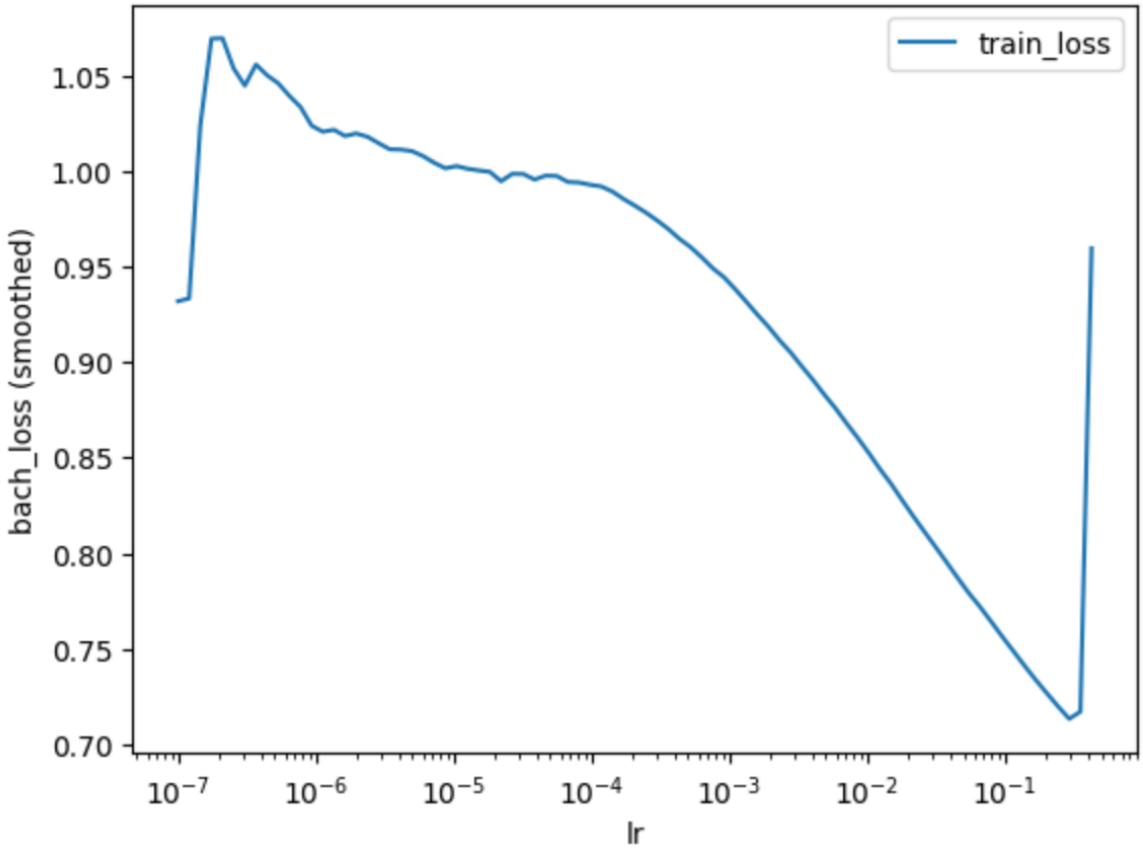

Рис. 4.5: Выбор оптимального шага обучения для модели DeepHit

4.6.1 Вклад признаков моделей

Точность на исходных 50 признаках: 73.1%

Точность на 5 признаках модели: 80.6%

Потеря в точности: 7.5%

4.6.2 Значение C-индекса модели

Значение C индекса: 0.74

4.6.3 Интерпретация вероятностей выживания

Ниже приведены вероятности выживания данной модели для разных месяцев:

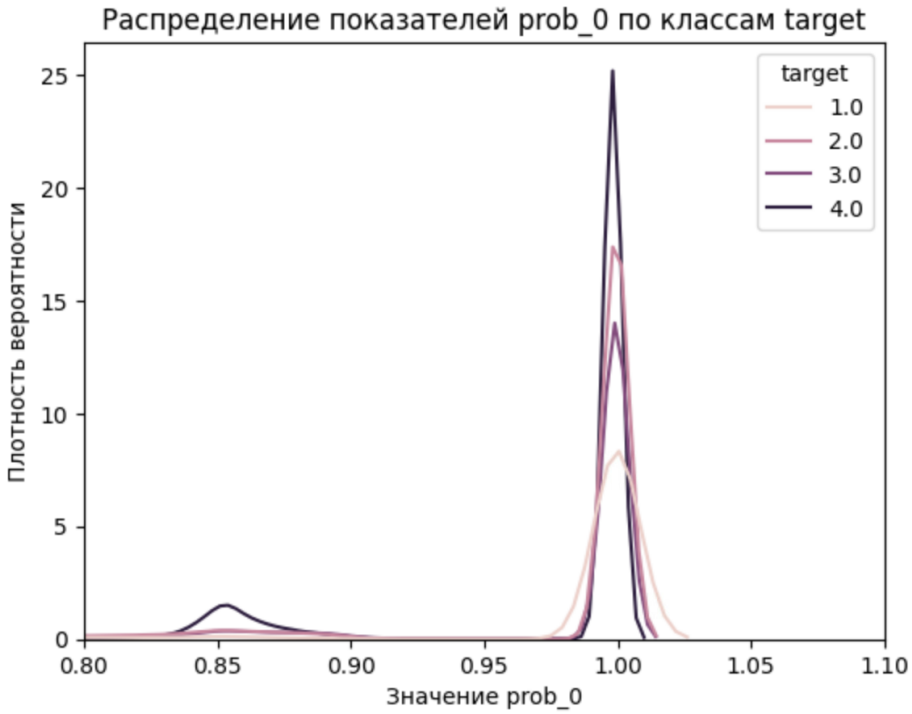

Рис. 4.6: Вероятность выживания в июнь 2024 для модели DeepHit

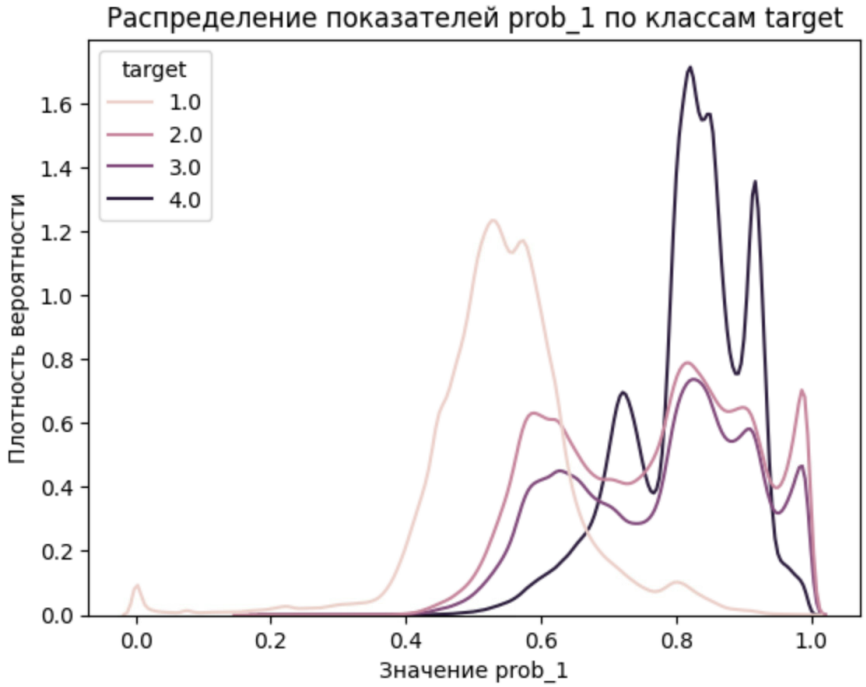

Рис. 4.7: Вероятность выживания в июль 2024 для модели DeepHit

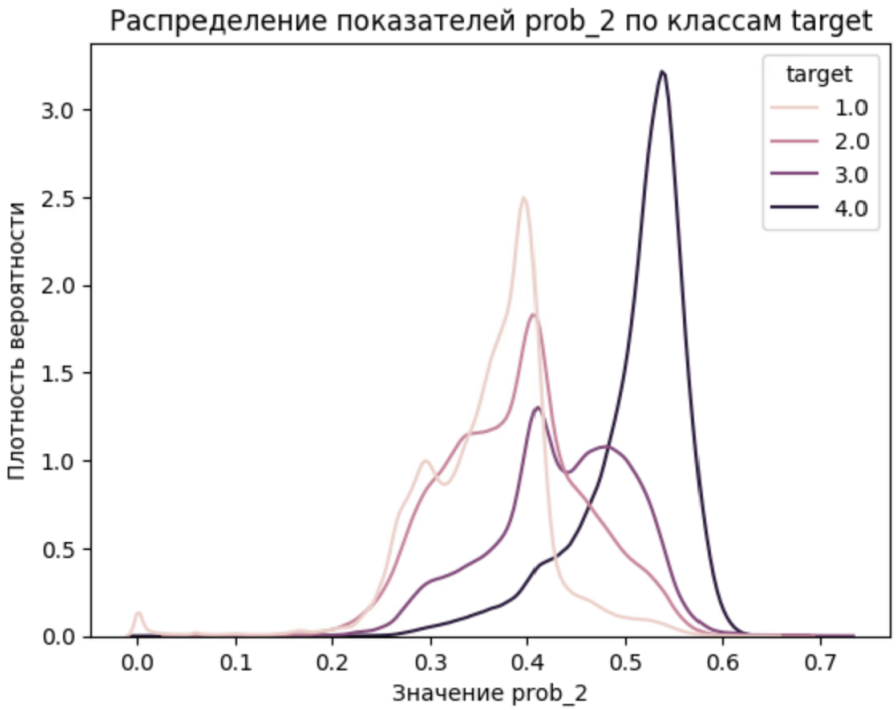

Рис. 4.8: Вероятность выживания в август 2024 для модели DeepHit

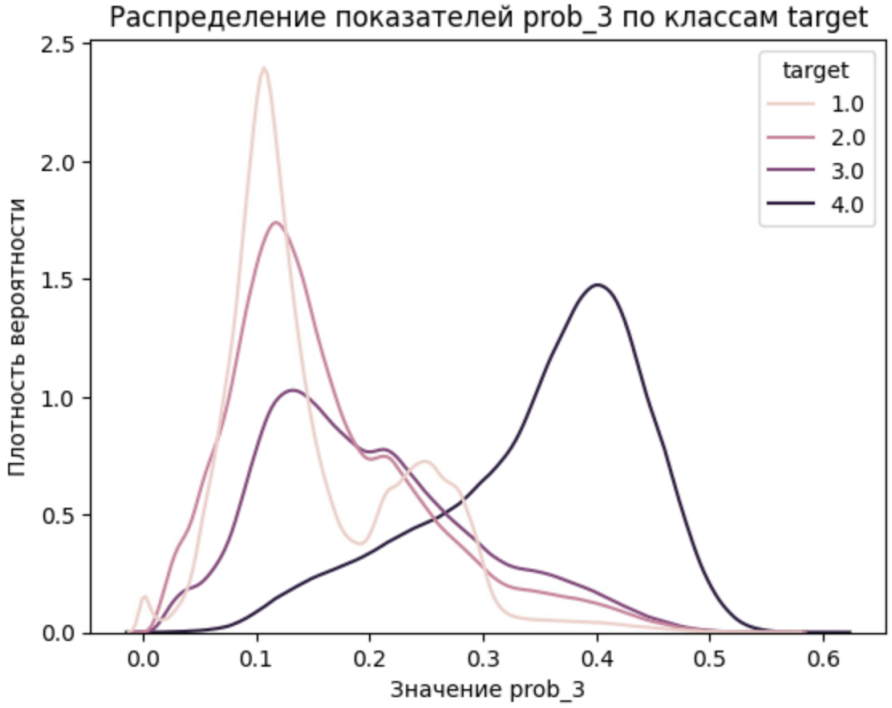

Рис. 4.9: Вероятность выживания в сентябрь 2024 для модели DeepHit

Видим логичную закономерность: чем более поздний месяц мы рассматриваем, тем ниже будет вероятность.

Видно, что модель хорошо отделяет таргет 4 (абоненты, которые не

оттекли) от оттекших абонентов (таргеты 1, 2 и 3), однако плохо отличает оттекших абонентов между собой.

4.7 PMF в задаче многоклассовой классификации

Ниже приведены вероятности выживания данной модели для разных месяцев:

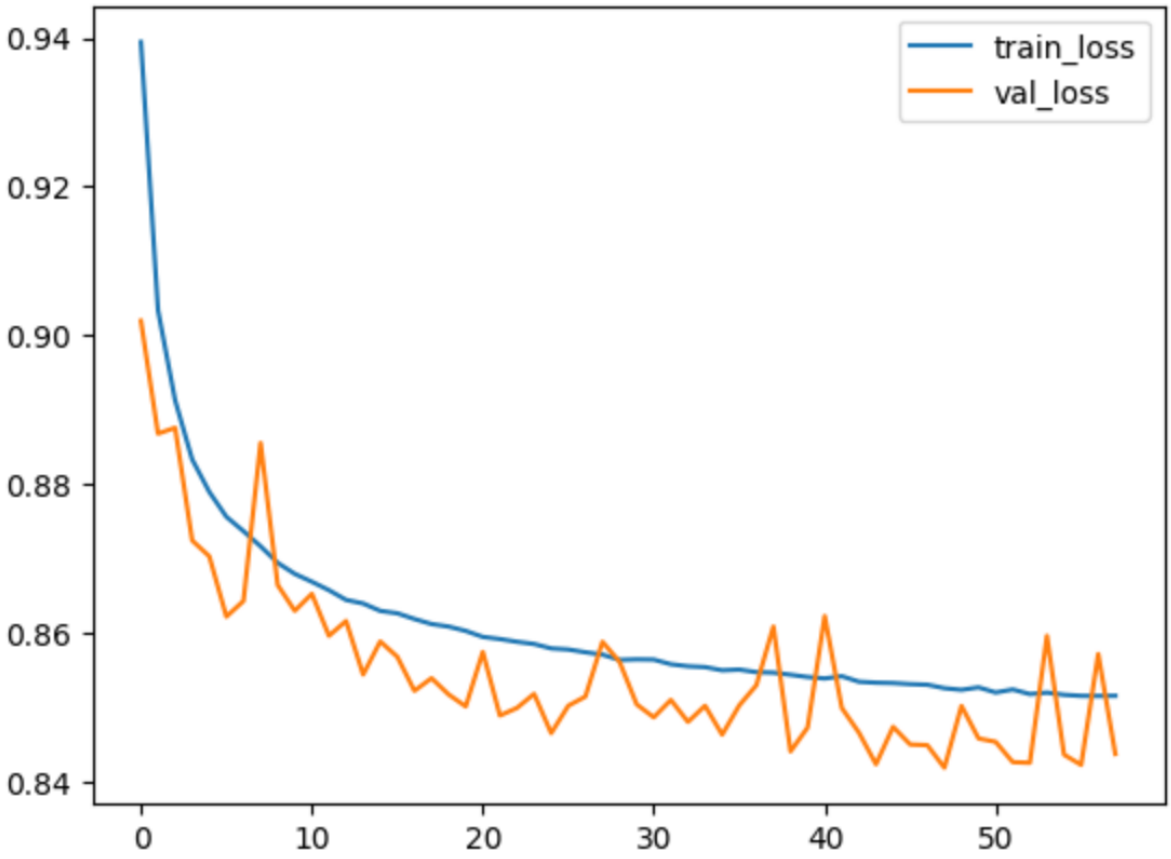

Рис. 4.10: Функция потерь для модели PMF

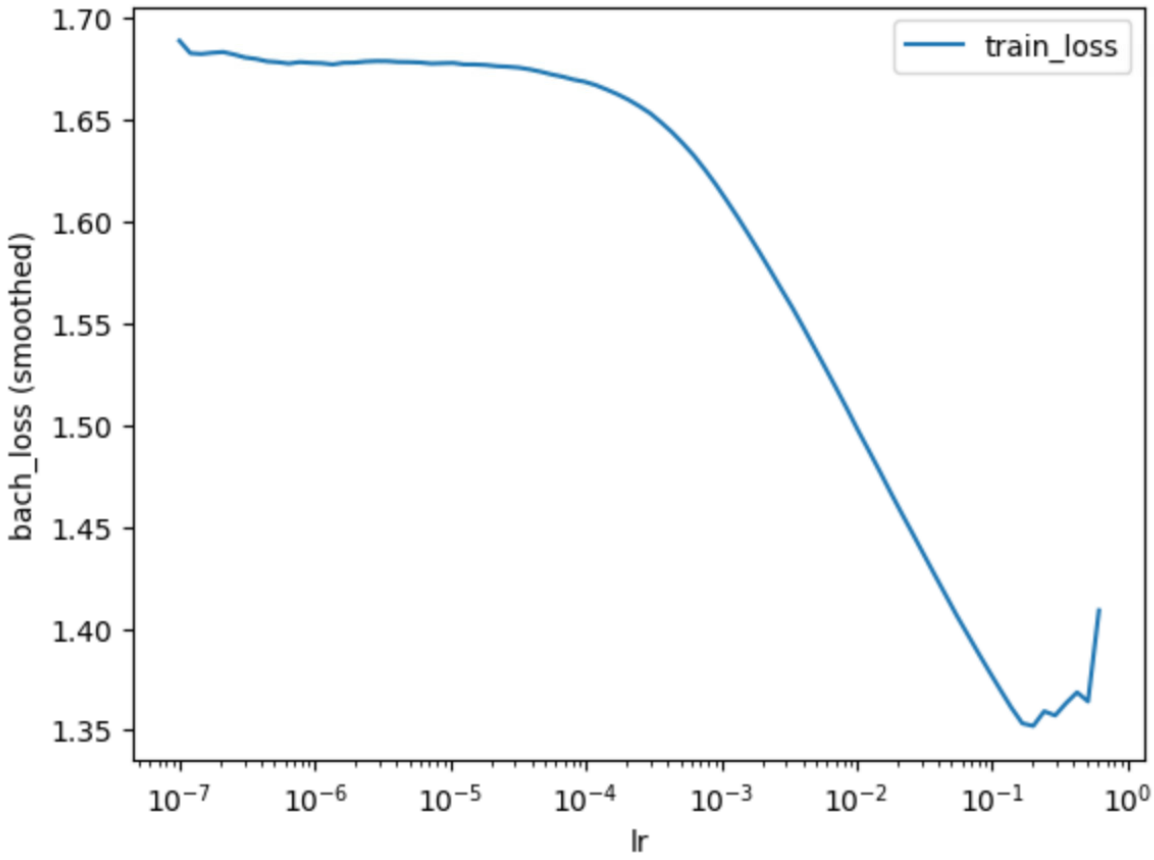

Рис. 4.11: Выбор оптимального шага обучения для модели PMF

4.7.1 Вклад признаков моделей

Точность на исходных 50 признаках: 75.2%

Точность на 5 признаках модели: 80.6%

Потеря в точности: 5.4%

4.7.2 Значение C-индекса модели

Значение C индекса: 0.75

4.7.3 Интерпретация вероятностей выживания

Ниже приведены вероятности выживания данной модели для разных месяцев:

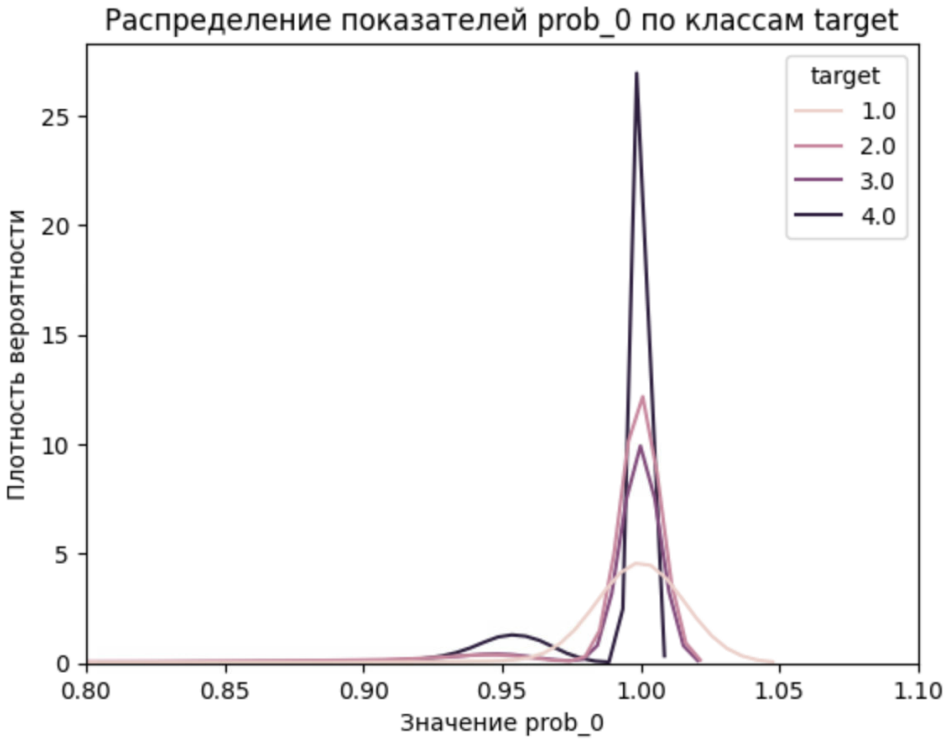

Рис. 4.12: Вероятность выживания в июнь 2024 для модели PMF

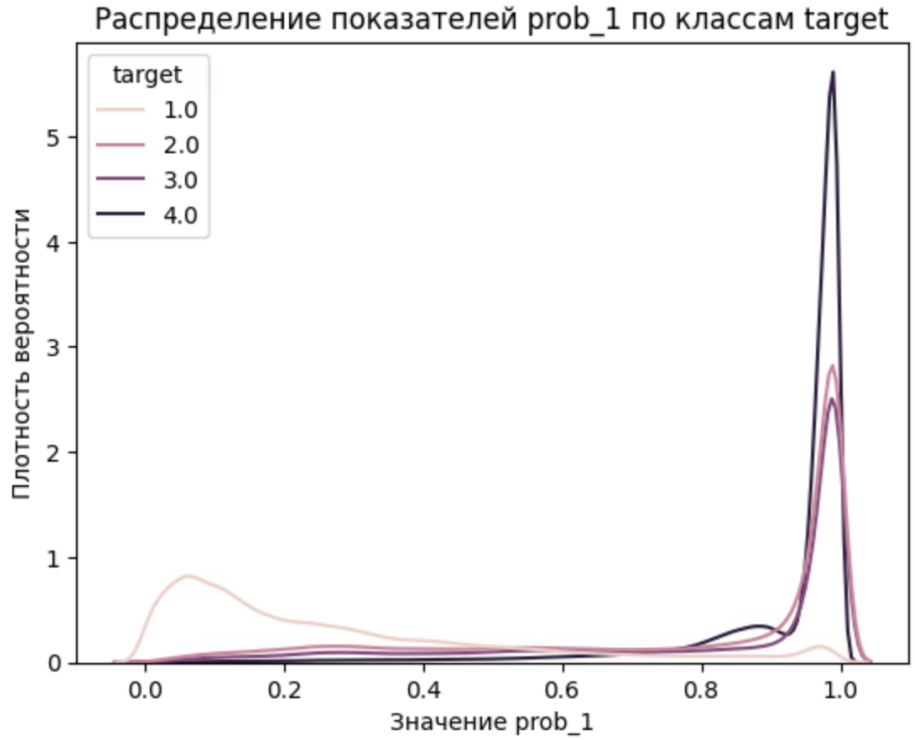

Рис. 4.13: Вероятность выживания в июль 2024 для модели PMF

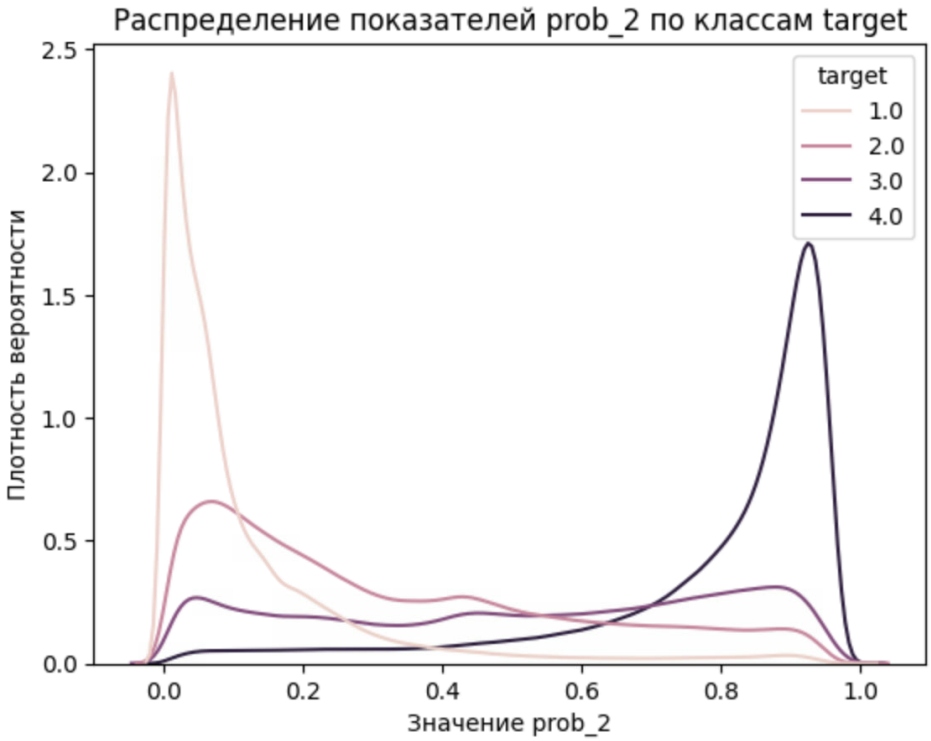

Рис. 4.14: Вероятность выживания в август 2024 для модели PMF

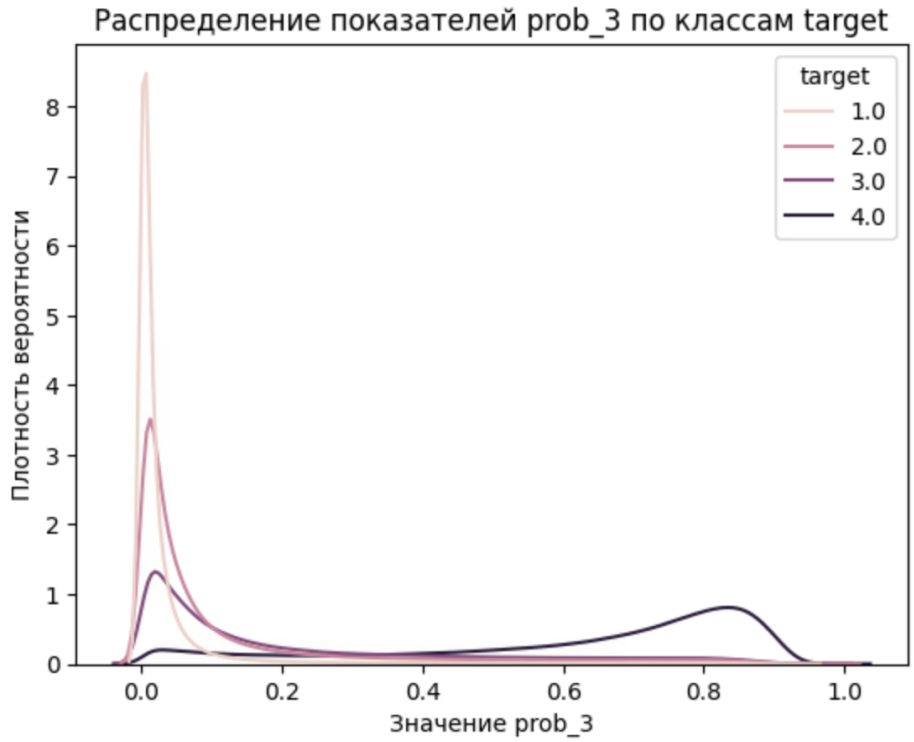

Рис. 4.15: Вероятность выживания в сентябрь 2024 для модели PMF

Модель хорошо отделяет таргет 4 (абоненты, которые не оттекли) от оттекших абонентов (таргеты 1, 2 и 3), однако плохо отличает оттекших абонентов между собой.

4.8 Nnet-survival в задаче многоклассовой классификации

Ниже приведены вероятности выживания данной модели для разных месяцев:

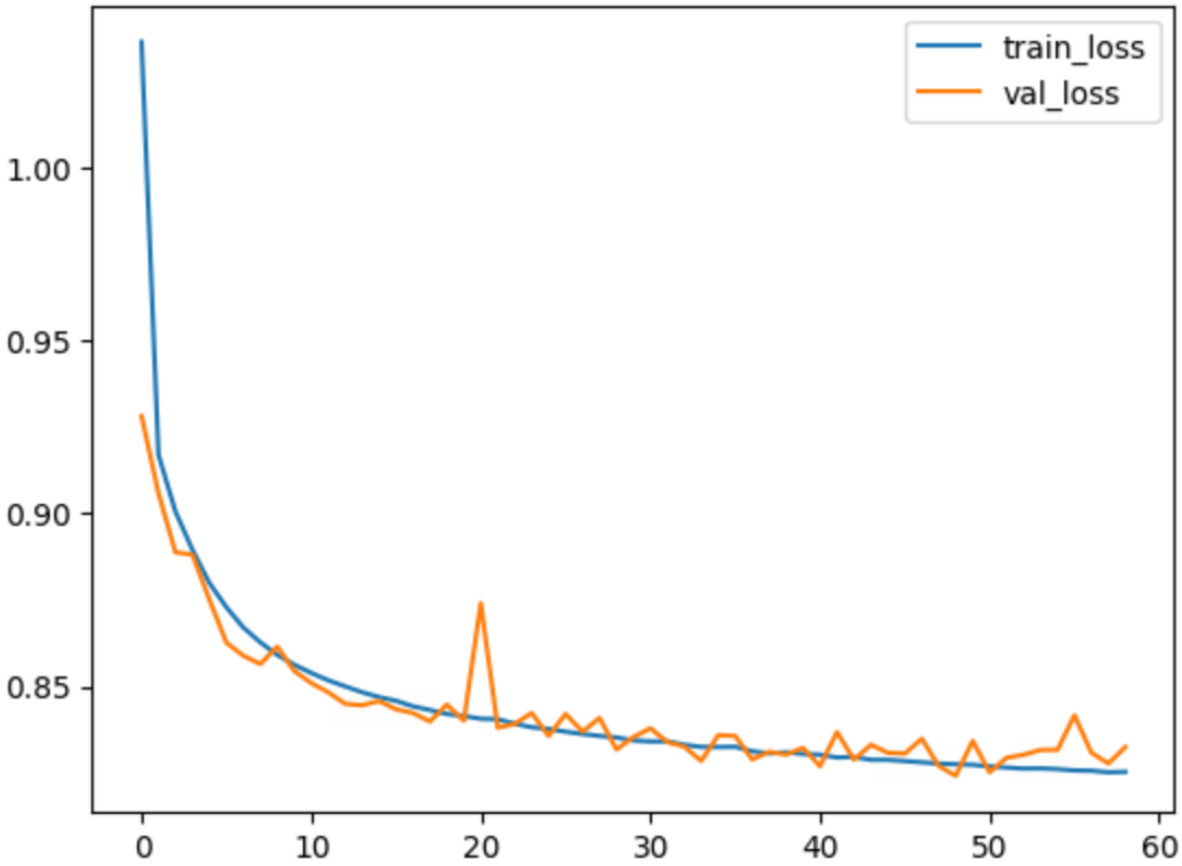

Рис. 4.16: Функция потерь для модели Nnet-survival

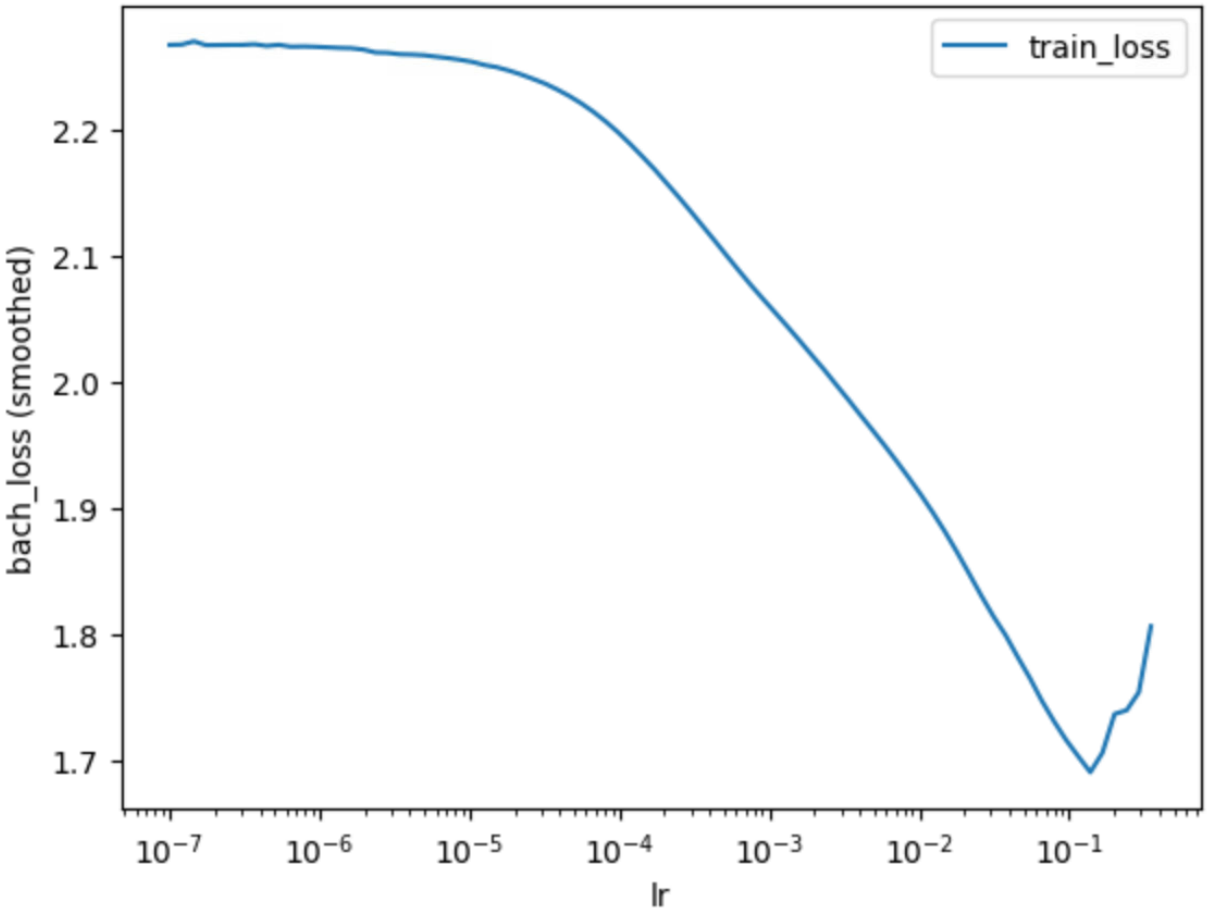

Рис. 4.17: Выбор оптимального шага обучения для модели Nnet-survival

4.8.1 Вклад признаков моделей

Точность на исходных 50 признаках: 74.8%

Точность на 5 признаках модели: 80.6%

Потеря в точности: 5.8%

4.8.2 Значение C-индекса модели

Значение C индекса: 0.77

4.8.3 Интерпретация вероятностей выживания

Ниже приведены вероятности выживания данной модели для разных месяцев:

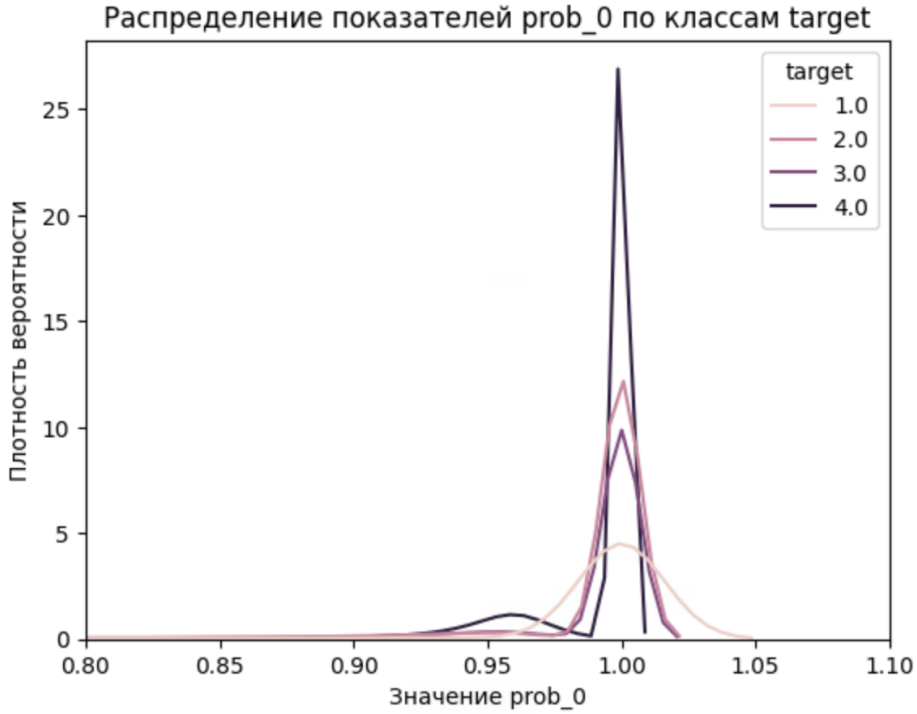

Рис. 4.18: Вероятность выживания в июнь 2024 для модели Nnet-survival

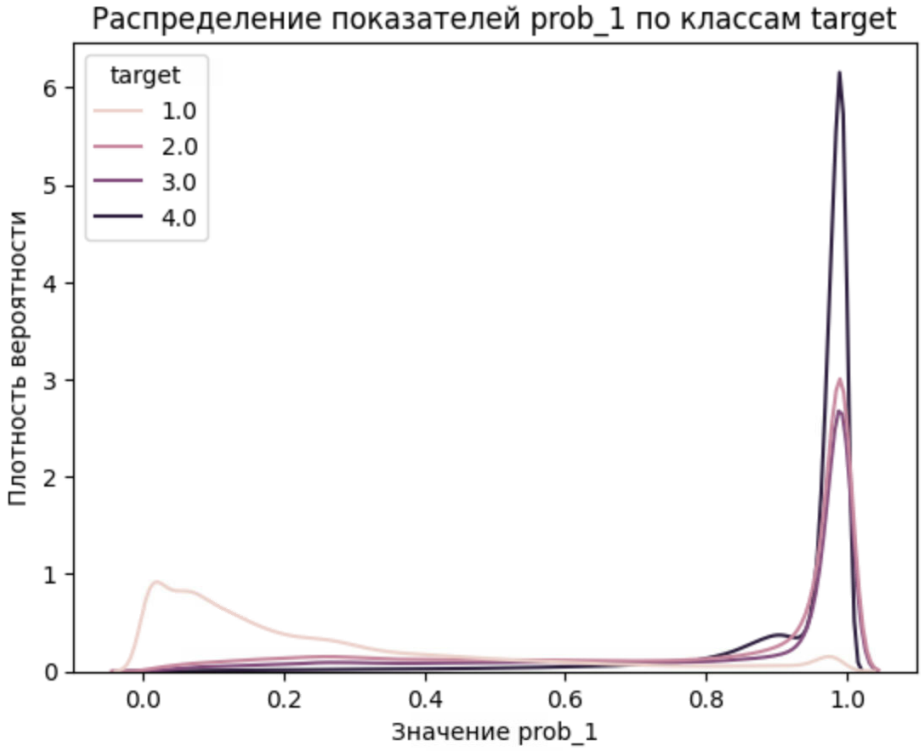

Рис. 4.19: Вероятность выживания в июль 2024 для модели Nnet-survival

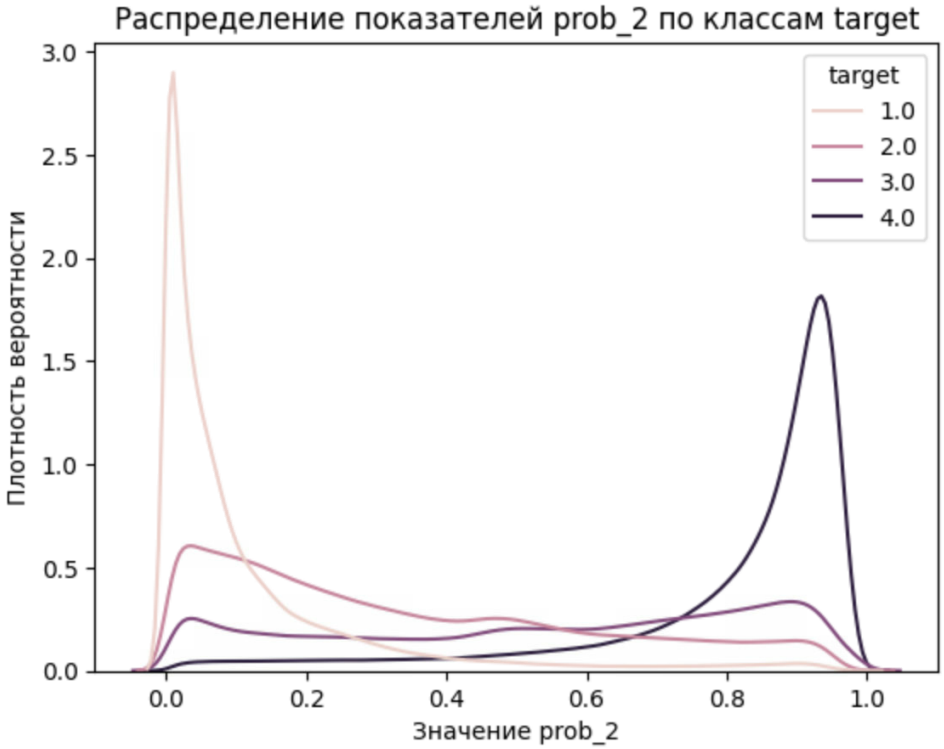

Рис. 4.20: Вероятность выживания в август 2024 для модели Nnet-survival

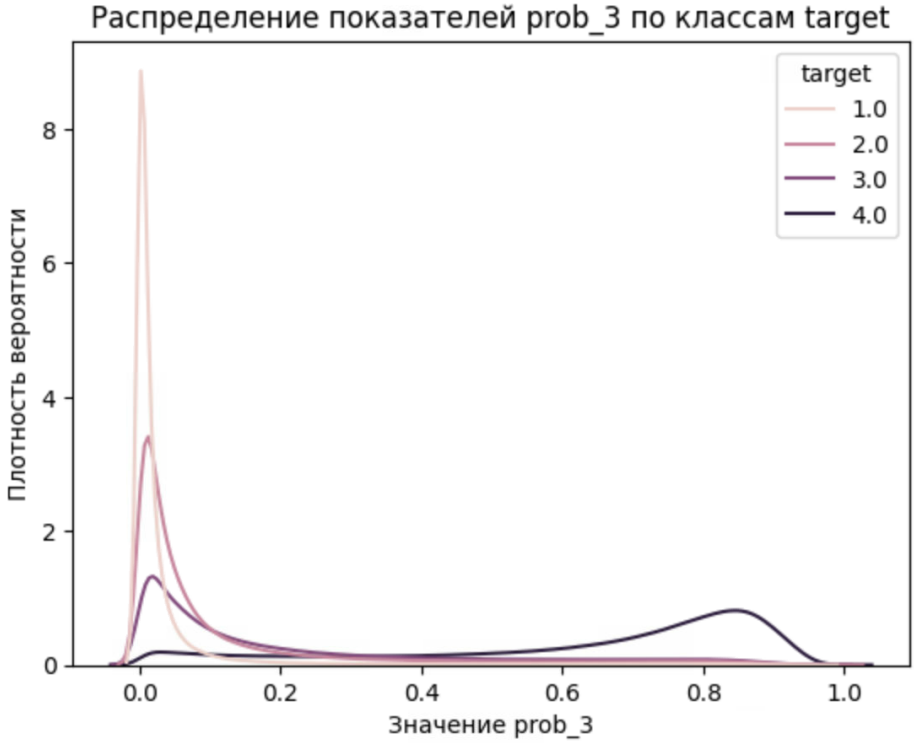

Рис. 4.21: Вероятность выживания в сентябрь 2024 для модели Nnet-survival

Модель хорошо отделяет таргет 4 (абоненты, которые не оттекли) от

оттекших абонентов (таргеты 1, 2 и 3), однако плохо отличает оттекших абонентов между собой.

4.9 Наша модель в задаче многоклассовой классификации

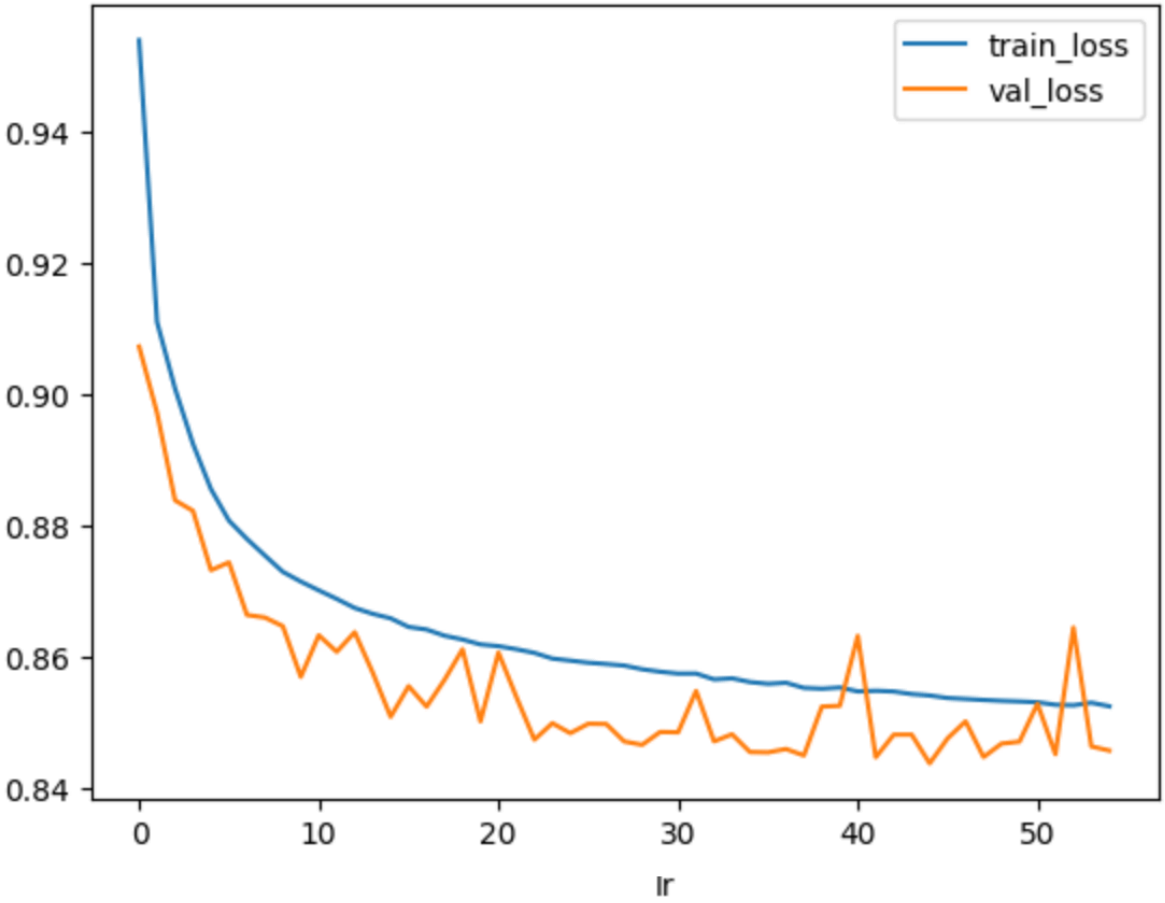

Рис. 4.22: Функция потерь для нашей модели

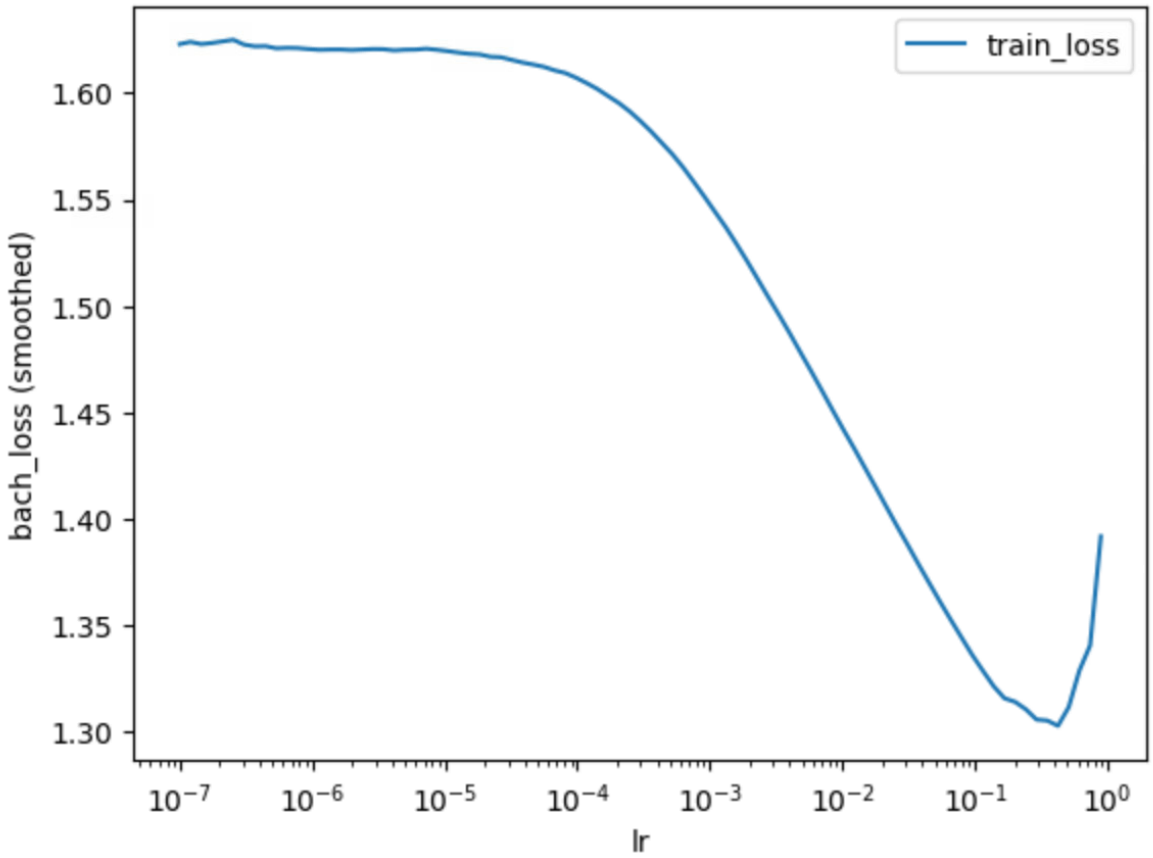

Рис. 4.23: Выбор оптимального шага обучения для нашей модели

4.9.1 Вклад признаков моделей

Точность на исходных 50 признаках: 78.5%

Точность на 5 признаках модели: 80.6%

Потеря в точности: 2.1%

4.9.2 Значение C-индекса модели

Значение C индекса: 0.81

4.9.3 Интерпретация вероятностей выживания

Ниже приведены вероятности выживания данной модели для разных месяцев:

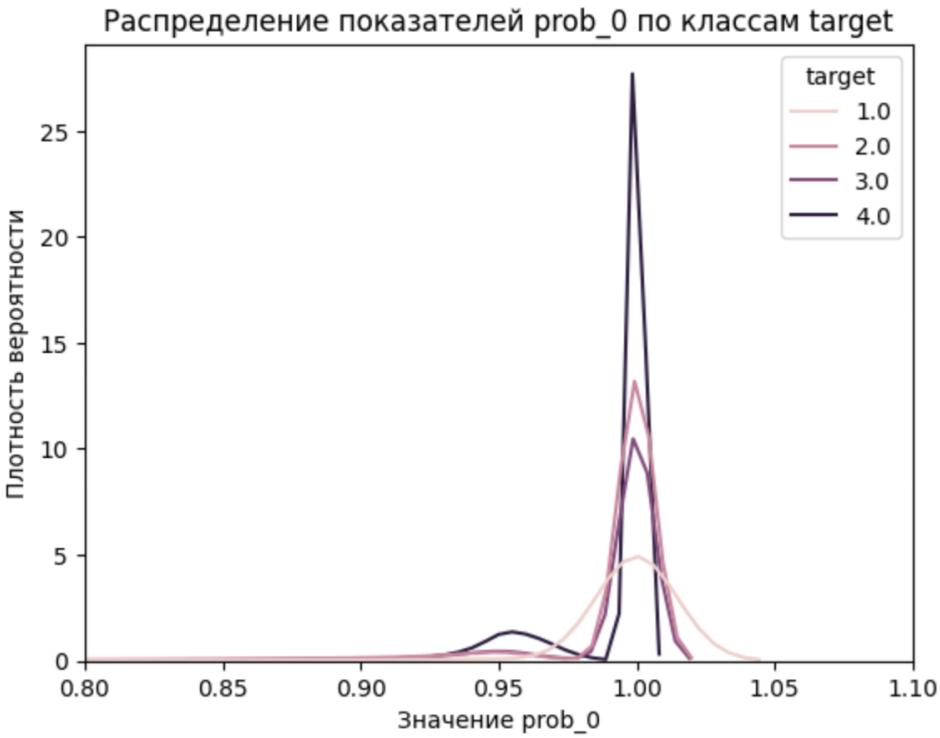

Рис. 4.24: Вероятность выживания в июнь 2024 для нашей модели

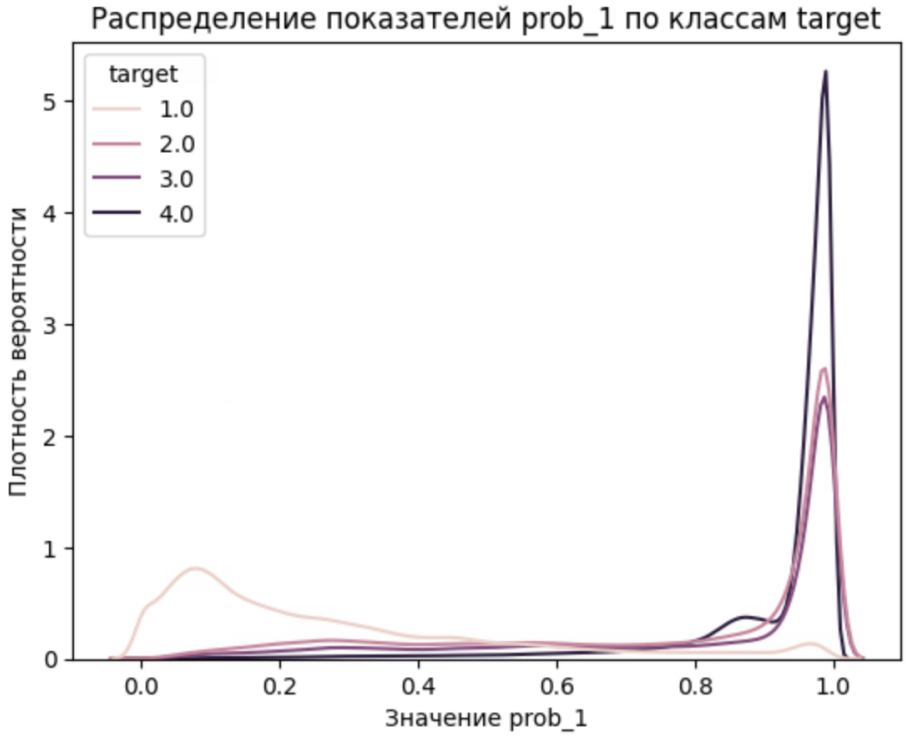

Рис. 4.25: Вероятность выживания в июль 2024 для нашей модели

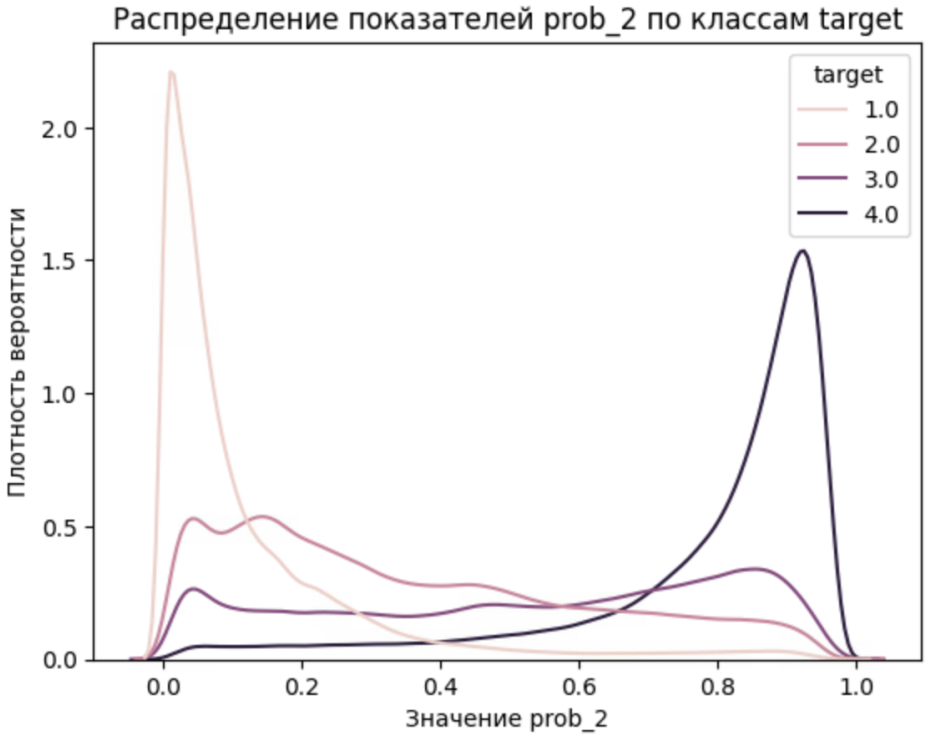

Рис. 4.26: Вероятность выживания в август 2024 для нашей модели

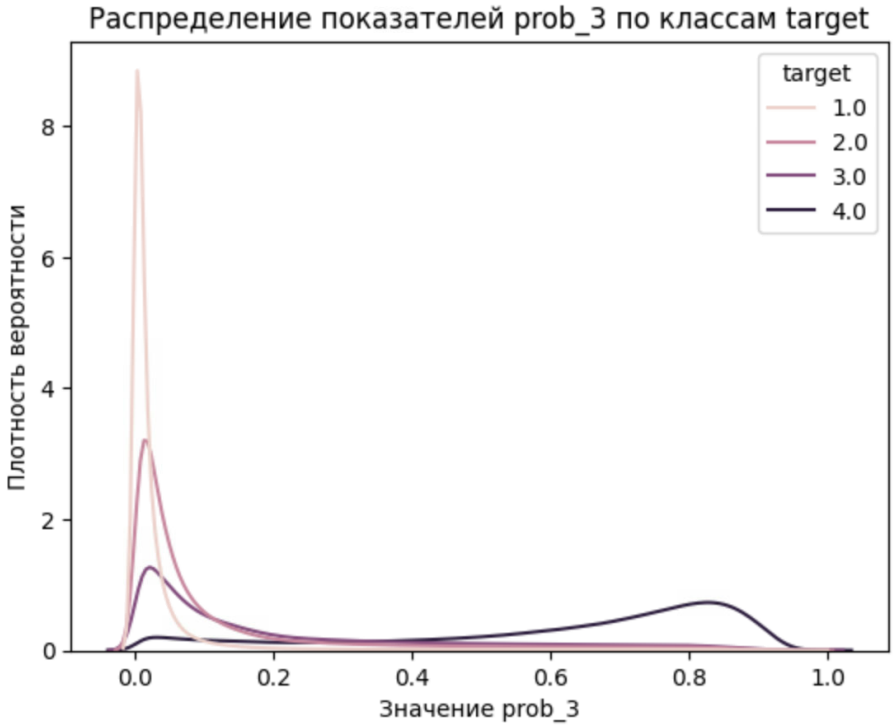

Рис. 4.27: Вероятность выживания в сентябрь 2024 для нашей модели

Увы, но та же проблема переносится и на нашу модель: оттекшие абоненты имеют схожие графики функций плотностей распределений, поэтому их модель не может адекватно отличить.

4.10 Сравнение различных моделей

4.10.1 Сравнение метрик

Суммируем все наши данные по метрикам в таблицу:

Название модели	Исходная точность	Точность на 5 признаках	Потеря точности	C-индекс
DeepHit	80.6%	73.1%	7.5%	0.74
PMF		75.2%	5.4%	0.75
Nnet-survival		74.8%	5.8%	0.77
Наша модель		78.5%	2.1%	0.81

Как видно, наша модель превосходит остальные модели.

4.10.2 Вывод по интерпретации вероятностей выживания

Что касается результатов отдения отточников между собой - тут результаты более удручающие. По результатам обсуждения данного результата внутри Мегафона было выяснено, что данная причина связана с низкой изменчивостью клиентских данных за месячный период. Из за этого задача классификации и отделения отточников, которые оттекают в соседние месяцы является в Мегафоне тяжелой задачей, даже для хорошо зарекомендовавших себя моделей градиентного бустинга.

Заключение

В работе проведен анализ существующих решений из моделей анализа выживаемости, кратко описаны их отличия и проведены сравнения данных моделей на задаче оттока клиентов Мегафона.

В рамках работы была разработана модель, отличающаяся от предыдущих использованием временных рядов векторов признаков. Архитектура модели состоит из RNN блока, механизма внимания и перцептрона и является известным и хорошо зарекомендовавшим себя решением в моделях выживания. Основная новизна работы заключается в использовании специфической функции потерь. В работе доказывается свойство добавки к функции потерь.

Модель, разработанная в данной работе, показала более высокие результаты по сравнению с другими моделями в плане более низкой потери качества на "сжатом"признаковом описании и более высоким C-индексом. К сожалению, наша модель не смогла решить проблему с классификацией отточников, хотя она и хорощо отделяет отточников от сохраняющихся абонентов, как и остальные модели.

Основным направлением дальнейших исследований можно выделить решение проблемы с классификацией отточников. Здесь автор видит основное решение в более продвинутом использовании Feature Engineering, так как, как показали исследования, корень проблем кроется в медленном изменении клиентских данных. Благо анализ существующих работ позволил выявить множество идей признаков, которые хорошо показали себя у других исследователей и которые могут дать весовый прирост на качество моделей Мегафона.

Литература

[1] A Survey on Churn Analysis in Various Business Domains / Jaehyun Ahn, Junsik Hwang, Doyoung Kim et al. // *IEEE Access*. — 2020. — Vol. 8. — Pp. 220816–220839.

[2] *Cox, D. R.* Regression Models and Life-Tables / D. R. Cox // *Journal of the Royal Statistical Society Series B: Statistical Methodology*. — 1972. — . — Vol. 34, no. 2. — Pp. 187–202.

[3] Deep learning for survival analysis: a review / Simon Wiegrebe, Philipp Kopper, Raphael Sonabend et al. // *Artificial Intelligence Review*. — 2024. — . — Vol. 57, no. 3.

[4] DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks / Changhee Lee, William Zame, Jinsung Yoon, Mihaela Van der Schaar // *Proceedings of the AAAI Conference on Artificial Intelligence*. — 2018. — . — Vol. 32, no. 1.

[5] *Ahmad, Abdelrahim Kasem.* Customer churn prediction in telecom using machine learning in big data platform / Abdelrahim Kasem Ahmad, Assef Jafar, Kadan Aljoumaa // *Journal of Big Data*. — 2019. — . — Vol. 6, no. 1.

[6] *Dorogush, Anna Veronika.* CatBoost: gradient boosting with categorical features support. — 2018.

[7] The Concordance Index decomposition: A measure for a deeper understanding of survival prediction models / Abdallah Alabdallah, Mattias Ohlsson, Sepideh Pashami, Thorsteinn Rögnvaldsson. — 2022.

[8] *Kaplan, E. L.* Nonparametric Estimation from Incomplete Observations / E. L. Kaplan, Paul Meier // *Journal of the American Statistical Association.* — 1958. — . — Vol. 53, no. 282. — Pp. 457–481.

[9] DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network / Jared L. Katzman, Uri Shaham, Alexander Cloninger et al. // *BMC Medical Research Methodology.* — 2018. — . — Vol. 18, no. 1.

[10] *Gensheimer, Michael F.* A scalable discrete-time survival model for neural networks / Michael F. Gensheimer, Balasubramanian Narasimhan // *PeerJ.* — 2019. — . — Vol. 7. — P. e6257.

[11] *Lee, Changhee.* Dynamic-DeepHit: A Deep Learning Approach for Dynamic Survival Analysis With Competing Risks Based on Longitudinal Data / Changhee Lee, Jinsung Yoon, Mihaela van der Schaar // *IEEE Transactions on Biomedical Engineering.* — 2020. — . — Vol. 67, no. 1. — Pp. 122–133.