МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

по курсу

«Data Science»

Слушатель Стебунова Рената Вячеславовна

Москва, 2023

Содержание

Введение

Темой данной работы является анализ выживаемости пациентов с раком молочной железы методами машинного обучения на примере датасета METABRIC, представленного на Kaggle.

Актуальность выбора данной темы обусловлена растущим интересом к возможностям искусственного интеллекта в разных областях медицины, особенно в онкологии. Разработки таких компаний как IBM, Google в области искусственного интеллекта для решения медицинских задач, подтверждают наш тезис. Интерес аналитиков в области Data Science характеризует наличие на Kaggle немалого количества проектов, связанных с практическим использованием искусственного интеллекта в онкологии самой разной направленности.

Цифровизация здравоохранения и переход на электронные медицинские карты с накоплением больших данных (Big Data) в медицине является отчасти причиной такого повышенного интереса.

В последние годы большое количество исследований, публикаций, а также законодательных изменений посвящено теме реальной клинической практики: как самих данных, так и проводимых на них исследований; общепринятая англоязычная аббревиатура RWD/RWE (real-world data, real-world evidence).

Исследования реальной клинической практики становятся важным дополнением к золотому стандарту исследований в медицине - рандомизированным клиническим исследованиям.

Вместе с изменением парадигмы меняются и методы исследования: от использования группы внешнего контроля (на примере накопленных данных реальной клинической практики) и добавления propensity score matching (так называемой псевдорандомизации) для устранения влияния конфаундеров в статистический анализ до активного применения методов машинного обучения,

например, автоматическое распознавание медицинских изображений, классификация генетических мутаций и формирование групп повышенного риска с целью оптимизации онкологического скрининга.

Безусловно, очевидный интерес в онкологии представляет анализ выживаемости, который является основным при решении о внедрении тех или иных методов скрининга, диагностики, лечения и других. Несмотря на явный интерес к методам машинного обучения в онкологии, публикаций, посвященных именно анализу выживаемости, не так много в сравнении с работами по диагностике и раннему выявлению при помощи анализа изображений [10].

В связи с чем исследование по теме данной работы является актуальным и представляет научный интерес.

1. Аналитическая часть

1.1. Постановка задачи

Для проведения исследования был выбран датасет METABRIC с Kaggle, содержащий анонимизированные данные 1904 пациентов и представленный одним из контрибьюторов онлайн-платформы.

METABRIC (Molecular Taxonomy of Breast Cancer International Consortium) представляет собой достаточно большой научный наблюдательный проект более 2000 пациентов с раком молочной железы в 5 медицинских центрах Великобритании и Канады 1977-2005 гг. Проект представлен на официальном сайте https://www.cbioportal.org/, по анализу данных пациентов с РМЖ в рамках данного проекта опубликовано много научных работ в ведущих медицинских журналах [11, 12].

Задачей данного исследования является разработка моделей машинного обучения, прогнозирующих выживаемость пациентов в зависимости от

признаков. Также требуется разработать приложение, которое позволит осуществить практическое использование предложенной модели.

Первоначально анализ выживаемости не был исходной задачей при поиске и выборе датасета, но, учитывая изложенное выше в введении, был выбран приоритетной задачей аттестационной работы.

Датасет представляет собой анонимизированные данные 1904 пациентов с раком молочной железы с 693 признаками, из которых 31 признак - это клинические данные, 331 - z-индекс уровней мРНК 331 генов и 175 - типы мутаций в конкретных генах. Клинические признаки представлены в таблице 1.

Таблица 1. Описание клинических признаков датасета

Название	**Тип данных**	**Описание**	**Непустые значения**	**% пропусков**
patient_id	object	ID пациента	1904	
age_at_diagnosis	float	Возраст пациента на момент диагноза	1904	
type_of_breast_surgery	object	Вид операции: 1- МАСТЭКТОМИЯ, операция по удалению всей ткани молочной железы в качестве способа лечения или предотвращения рака молочной железы; 2 - ОРГАНОСОХРАНЯЮЩАЯ операция, при которой удаляется только часть молочной железы, пораженная раком	1882	1,16
cancer_type	object	Вид злокачественной опухоли: 1- Рак молочной железы или 2- Саркома молочной железы	1904	

cancer_type_detailed	object	Детализированная классификация по строению: 1- Протоковый инвазивный рак, 2 - Смешанный протоковый и лобулярный рак, 3 - Дольковый инвазивный рак, 4 - Смешанный инвазивный муцинозный рак, 5 - Метапластический рак	1889	0,79
cellularity	object	Плотность опухолевых клеток после химиотерапии, которая относится к количеству опухолевых клеток в образце и их расположению в кластеры	1850	2,84
chemotherapy	int	Получал ли пациент химиотерапию в качестве лечения (да/нет)	1904	
pam50_+_claudin-low_subtype	object	Pam 50: это тест профиля опухоли, который помогает предсказать, могут ли некоторые эстроген-рецептор-положительные (ER-положительные) и HER2-отрицательные виды рака молочной железы метастазировать (когда рак молочной железы распространяется на другие органы). Подтип рака молочной железы с низким уровнем клаудина определяется характеристиками экспрессии	1904	

		генов, в первую очередь: низкая экспрессия генов межклеточной адгезии, высокая экспрессия генов эпителиально-мезенхимального перехода (ЭМТ) и паттерны экспрессии генов, подобные стволовым клеткам / менее дифференцированные.		
cohort	float	Когорта — это группа субъектов, имеющих общую определяющую характеристику (принимает значения от 1 до 5)	1904	
er_status_measured_by_ihc	float	Оценка экспрессии рецепторов эстрогена на раковых клетках при помощи иммуногистохимии (краситель, используемый в патологии, нацеленный на определенный антиген, если он есть, образец будет окрашен, если его нет, будет окрашена ткань на предметном стекле) (позитивный/негативный)	1874	1,58
er_status	object	Наличие эстрогеновых рецепторов в опухоли - положительный/отрицательный статус	1904	
neoplasm_histologic_grade	int	Степень злокачественности опухолевых клеток, определяемая при	1832	3,78

		гистологическом исследовании (от 1 до 3)		
her2_status_measured_by_snp6	object	Оценка экспрессии рецепторов HER2 с использованием передовых молекулярных методов (секвенирование следующего поколения)	1904	
her2_status	object	Наличие рецепторов HER2 в опухоли - положительный/отрицательный статус	1904	
tumor_other_histologic_subtype	object	Разновидность рака по строению по результатам микроскопического исследования (значения 'Протоковый/неспецифического типа', 'Смешанный', 'Дольковый', 'Тубулярный/крибриформный', 'Муцинозный', 'Медуллярный', 'Другой', 'Метапластический')	1889	0,79
hormone_therapy	int	Проводилась ли гормональная терапия (да/нет)	1904	
inferred_menopausal_state	object	Статус менопаузы (пост/пре)	1904	
integrative_cluster	object	Молекулярный подтип рака, основанный на экспрессии определенного гена (принимает значения из «4ER+», «3», «9», «7», «4ER-», «5», «8», «10», «1», «2», «6»)	1904	

primary_tumor_laterality	object	Локализация - правая или левая молочная железа	1798	5,57
lymph_nodes_examined_positive	float	Количество образцов лимфатических узлов, взятых во время операции и пораженных раков	1904	
mutation_count	float	Количество генов с мутациями	1859	2,36
nottingham_prognostic_index	float	Используется для определения прогноза после операции по поводу рака молочной железы. Его значение рассчитывается с использованием трех критериев: размер опухоли; количество вовлеченных лимфатических узлов; и степень злокачественности опухоли.	1904	
oncotree_code	object	OncoTree — это онтология с открытым исходным кодом, разработанная в Memorial Sloan Kettering Cancer Center (MSK) для стандартизации диагностики типов рака с клинической точки зрения путем присвоения каждому диагнозу уникального кода OncoTree.	1889	0,79
overall_survival_months	float	Продолжительность от момента вмешательства до смерти	1904	
overall_survival	object	Жив пациент или мертв	1904	

pr_status	object	Наличие рецепторов прогестерона в опухоли - положительный/отрицательный статус	1904	
radio_therapy	int	Проводилась ли лучевая терапия (да/нет)	1904	
3-gene_classifier_subtype	object	Подтип классификатора с тремя генами. Он принимает значения из «ER-/HER2-», «ER+/HER2- высокая пролиферация», nan, «ER+/HER2- низкая пролиферация», «HER2+».	1700	10,71
tumor_size	float	Размер опухоли, измеренный методами визуализации	1884	1,05
tumor_stage	float	Стадия рака, основанная на поражении окружающих структур, лимфатических узлов и отдаленном распространении	1403	26,31
death_from_cancer	int	Связана ли смерть пациента с РМЖ (да/нет)	1903	0,05

Неклинические признаки датасета представляют собой z-индекс уровня мРНК 331 генов и мутации 175 генов.

Для данных экспрессии мРНК выполняются расчеты относительной экспрессии отдельного гена и опухоли по отношению к распределению экспрессии гена в эталонной популяции. Эта референтная популяция представляет собой все образцы в исследовании. Возвращаемое значение указывает количество стандартных отклонений от среднего значения экспрессии в эталонной популяции (Z-индекс). Эта мера полезна для определения того, какая

регуляция наблюдается - up или down по сравнению с нормальными образцами или всеми другими образцами опухоли.

Формула расчета: z = (экспрессия в опухолевом образце - средняя экспрессия в референтном образце) / стандартное отклонение экспресии в референтном образце. Таким образом, исходно имеем стандартизованные показатели в генетической части датасета.

Для упрощения анализа и обработки данных разделим датасет на клиническую и генетическую части.

Начнем с анализа клинической части датасета. Выделим 31 клинический признак (с 0 по 30 включительно).

Гистограммы распределения переменных, диаграммы «ящик с усами» и «скрипичного» графика приведены на рисунках 1-3. При визуальном анализе очевидно, что из числовых признаков только для возраста пациента ‘age_at_diagnosis’ характерно нормальное распределение.

Рисунок 1. Гистограмма распределения числовых признаков клинической части датасета Metabric

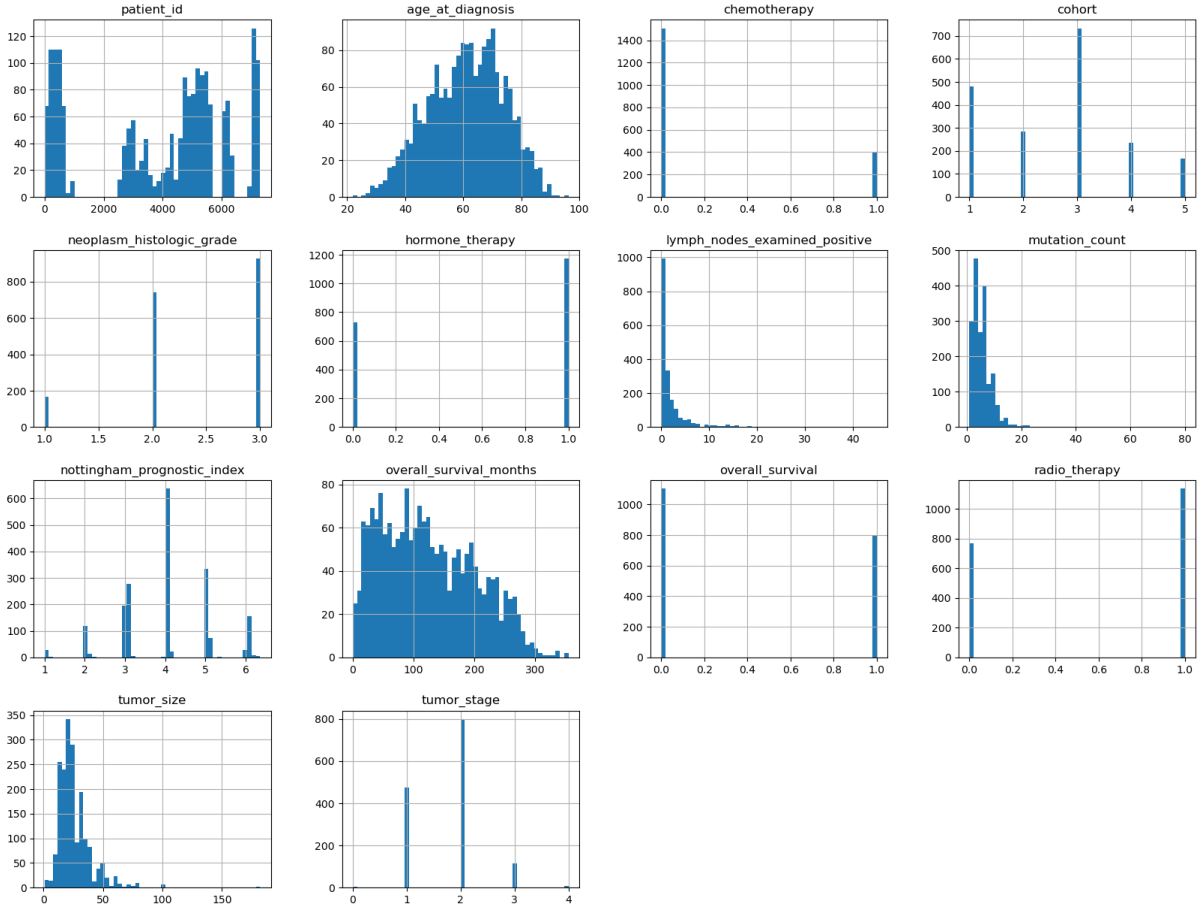


Рисунок 2. Диаграмма "ящик с усами" числовых признаков клинической части датасета Metabric

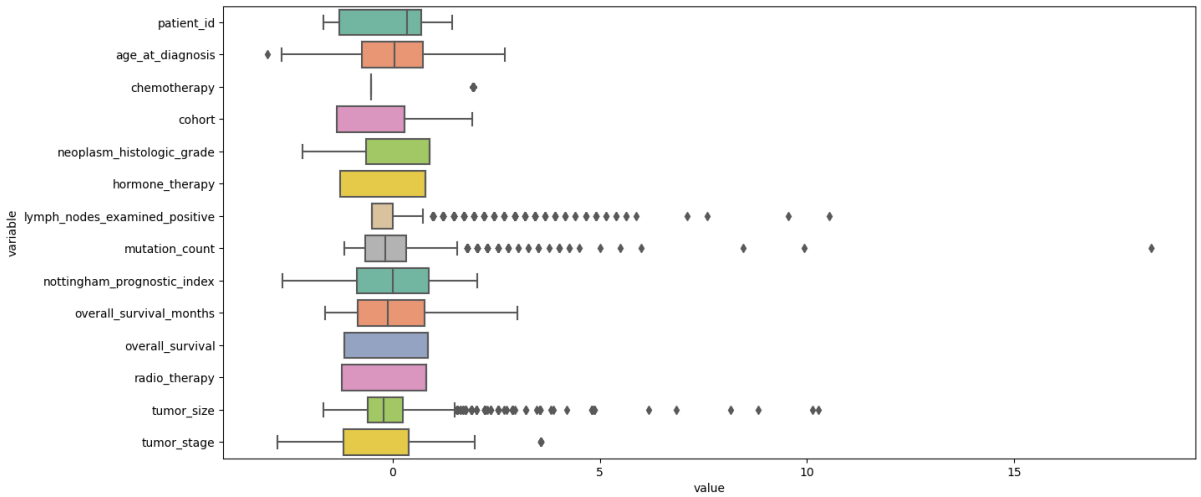
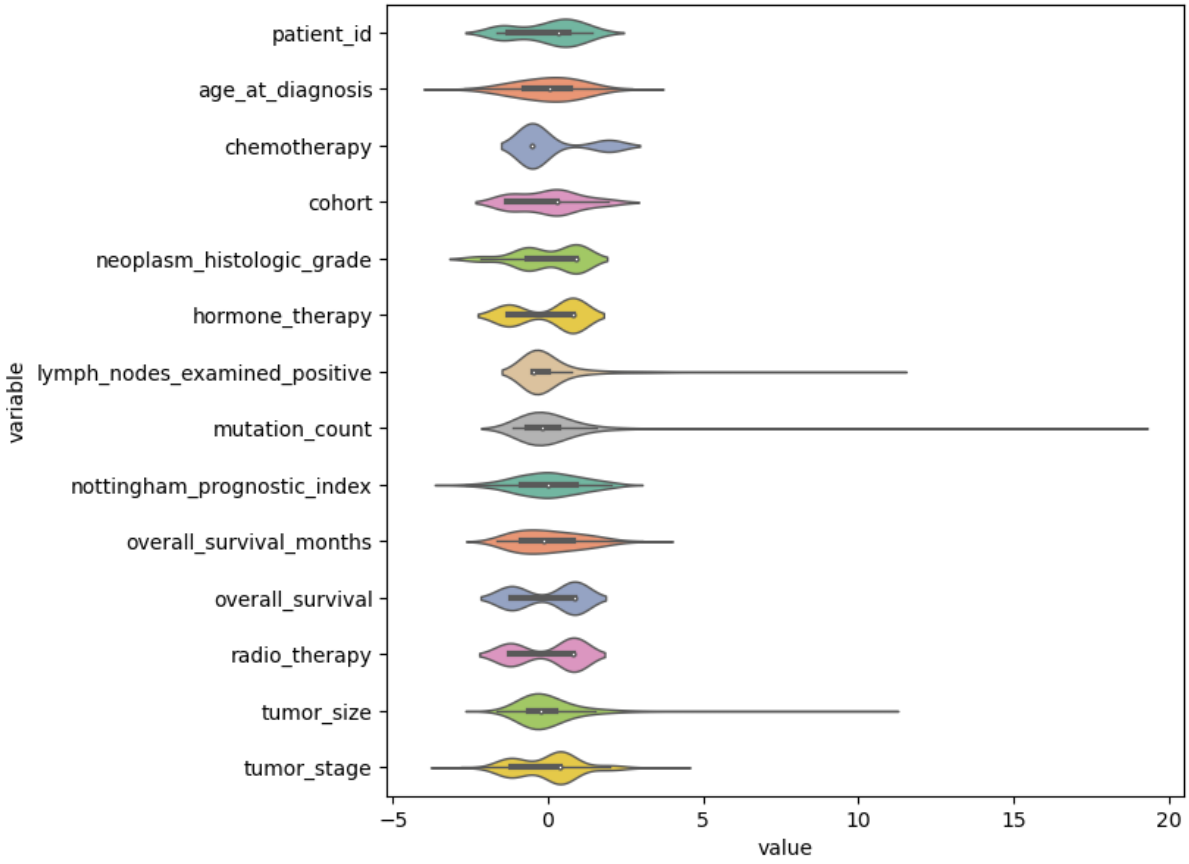


Рисунок 3. "Скрипичная" диаграмма числовых признаков клинической части датасета Metabric



Описательная статистика клинических признаков датасета представлена в таблице 2. Она в численном виде отражает то, что мы видим на гистограммах.

Таблица 2. Описательная статистика числовых признаков клинической части датасета Metabric

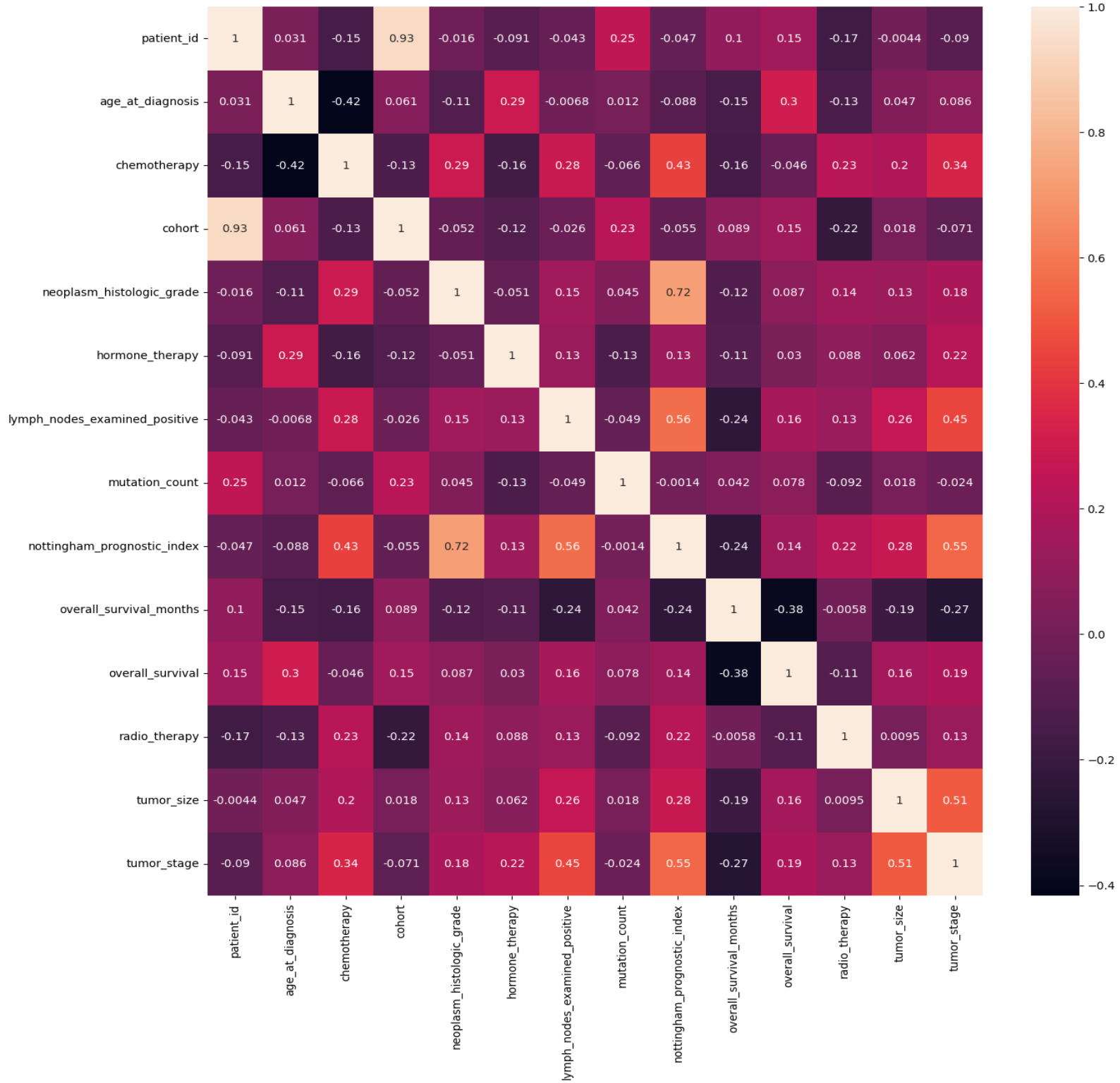
Признак	**Среднее**	**Стандартное отклонение**	**Минимум**	**Максимум**	**Медиана**
age_at_diagnosis	61.09	12.98	21.93	96.29	61.77
chemotherapy	0.21	0.41	0.00	1.00	0.00
cohort	2.64	1.23	1.00	5.00	3.00

neoplasm_histologic_grade	2.42	0.65	1.00	3.00	3.00
hormone_therapy	0.62	0.49	0.00	1.00	1.00
lymph_nodes_examined_positive	2.00	4.08	0.00	45.00	2.00
mutation_count	5.70	4.06	1.00	80.00	5.00
nottingham_prognostic_index	4.03	1.14	1.00	4.04	6.36
overall_survival_months	125.12	76.33	0.00	355.20	115.62
overall_survival	0.42	0.49	0.00	1.00	0.00
radio_therapy	0.60	0.49	0.00	1.00	1.00
tumor_size	26.24	16.16	1.00	182.00	23.00
tumor_stage	1.75	0.63	0.00	4.00	2.00

На графиках мы видим выбросы, явно выраженные в трех признаках - количество лимфоузлов, количество мутаций и размер опухоли. Однако, учитывая характер датасета, считаем эти данные реальными и важными для последующего анализа.

Аналогичной тактики будем придерживаться и с пропущенными данными. Несмотря на большой процент пропущенных данных у признака “tumor_stage” - 26%, данный признак общеизвестно является важным для прогноза на основании многочисленных статистических данных по заболеваемости и смертности, таким образом, решаем выбрать тактику заполнения пропущенных полей в дальнейшей обработке.

Рисунок 4. "Тепловая" карта



1.2 Описание используемых методов

Под методами оценки выживаемости (survival analysis) понимается изучение закономерности появления ожидаемого события у представителей наблюдаемой выборки во времени. Такое событие – не обязательно летальный исход, как можно предположить из названия анализа. Им может быть рецидив заболевания или, наоборот, выздоровление, в общем случае – происхождение определенного события. Точкой отсчета могут быть дата (час) выполнения процедуры, назначения лекарственного препарата, возраст на момент диагноза и т. п. Период времени от начального события (например, постановки диагноза) до итогового (летальный исход, рецидив, выздоровление) называется временем до события (time to event), или временем ожидания.

Исходя из общей постановки задачи, т. е. анализа среднего времени выживания и проведения на его основе, например, оценки эффективности нового метода лечения, казалось бы, можно воспользоваться параметрическими и непараметрическими статистическими методами. В принципе это возможно, но анализ выживаемости имеет важное отличие в способе построения выборки. В то время как для рассмотренных ранее статистических методов объем и структура выборки являются постоянными, в анализе времени до события они могут меняться. Проблема заключается в том, что время до события не обязательно может быть определено для всех пациентов выборки в ходе запланированного срока наблюдения. Значение этого показателя становится определенным только среди тех лиц, у которых произошло интересующее событие. Для всех остальных объектов наблюдения показатель остается неизвестным до наступления события, которое может вообще не произойти за период наблюдения. Кроме того, пациенты могут выбывать из исследования в силу разных обстоятельств (смена места жительства и т. п.), включаться в исследование в его середине или в конце, а также ожидаемое событие может

быть вызвано иной причиной (например, летальный исход не от заболевания, а в результате несчастного случая). Все это приводит к (нерегулярным) качественным и количественным изменениям в анализируемых данных и определяет необходимость применения специальных методов, в которых можно было бы учесть и использовать неполную информацию [2].

Данные, которые содержат неполную информацию, называют цензурированными (censored). С такими выборками приходится иметь дело, когда наблюдаемый параметр является временем до наступления события, а период наблюдения ограничен (например, у пациента рецидив заболевания не обнаружен за 6 мес до того, как он переехал в другой город и дальнейшая информация о нем недоступна). При анализе выживаемости, как и при других методах статистического анализа, вся информация о выборке содержится в соответствующей ей функции распределения вероятности (в данном случае – времени ожидания), но используется она не в виде плотности распределения вероятности значений, а в виде функции выживания (survival function). Кумулятивная функция распределения F(t) времени ожидания отражает вероятность того, что время ожидания события меньше t. Соответственно функция выживания S(t) = 1 - F(t) равна вероятности того, что событие не состоится ранее, чем по истечении времени t.

Частая ошибка специалистов в области Data Science заключается в игнорировании цензурированных пациентов/субъектов. Если нам нужно оценить среднюю продолжительность жизни заданной популяции популяции, а мы решили не включать цензурированных пациентов/субъектов, очевидно, что мы серьезно недооценим истинную среднюю продолжительность жизни.

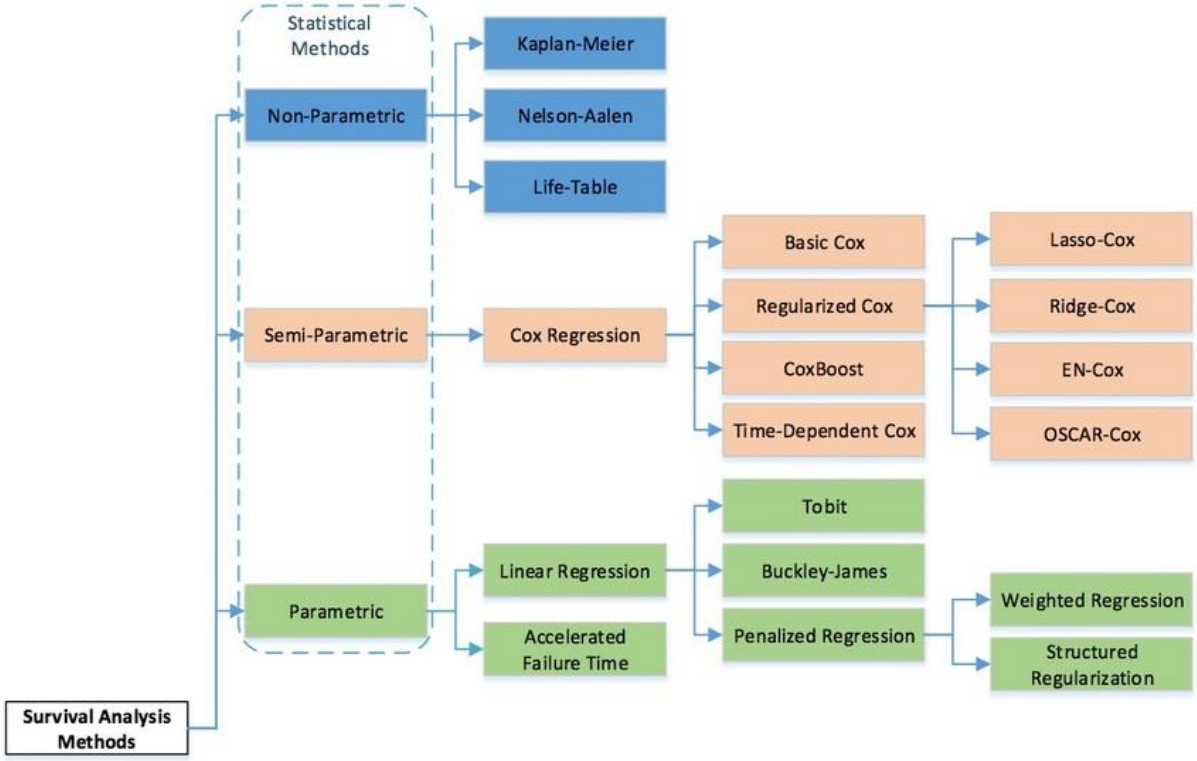
Кроме того, если вместо этого мы просто возьмем среднюю продолжительность жизни, включая цензурированных пациентов, мы все равно недооценим истинную среднюю продолжительность жизни.

Анализ нашего датасета подтверждает этот тезис. Медиана продолжительности жизни (общепринятым является использованием именно медианы, а не среднего применительно к продолжительности жизни) согласно вышепроведенному описательному анализу составила 115,62 месяца. Тогда как медиана продолжительности жизни согласно кривой Каплана-Мейера значительно выше (см. дальше по тексту).

Поэтому очевидно, что обычные методы не подходят и требуется специальные методы анализа и библиотеки. Для таких целей и был разработан анализ выживаемости.

Наиболее распространенными описательными методами исследования цензурированных данных являются построение таблиц дожития (mortality table) и метод Каплана–Мейера (Kaplan–Meier method). Для анализа используют несколько подходов, из которых мы остановимся на лог-ранк- тесте (логарифмический ранговый тест; англ, log-rank test) и модели пропорциональных интенсивностей Кокса (или модель пропорциональных рисков Кокса; англ. Cox Proportional Hazards Model).

Рисунок 5. Стандартные методы анализа выживаемости [16]



1.2.1 Метод Каплана-Майера

Метод Каплана–Мейера является описательным методом и используется для оценки доли объектов наблюдения (пациентов), у которых событие не произошло (функция выживания, выживаемость), для любого момента времени в течение всего периода наблюдения.

Оценка функции выживания в методе Каплана–Мейера представляет собой произведение выживаемости в данный момент времени на выживаемость в следующий момент времени, когда событие произошло.

Как и таблицы дожития, метод Каплана–Мейера полностью применим к цензурированным данным. Для расчетов используется истинное количество объектов, у которых событие еще не произошло в любой момент времени, для которого производится оценка.

Графическое представление метода Каплана–Мейера заключается в построении кривой выживаемости, отражающей пропорцию пациентов, у которых ожидаемое событие не произошло к определенному моменту времени. Временные интервалы определяются либо периодичностью контрольных обследований, либо временем до события в реальном масштабе (если известен момент происхождения события). Когда у объекта наблюдения происходит ожидаемое событие, производят перерасчет пропорции оставшихся в исследовании объектов, у которых событие не произошло, что отображается "ступенькой" вниз на кривой.

Кривые, построенные с помощью метода Каплана-Мейера, часто используются для оценки собственно выживаемости или безрецидивной выживаемости онкологических больных [2].

Для построения кривых Каплана-Майера использовались две библиотеки - lifelines и scikit-survival [4, 13]. Для работы с моделями машинного обучения в дальнейшем использовалась библиотека scikit-survival.

Scikit-survival — это модуль Python для анализа выживаемости, созданный поверх scikit-learn, что позволяет выполнять анализ выживаемости, используя возможности scikit-learn, например, для предобработки или кросс-валидации [13].

Для использования пакета необходимо импортировать kaplan_meier_estimator из sksurv.nonparametric.

Чтобы быть полностью совместимым с scikit-learn, Status и Survival time (в случае нашего датасета - это overall_survival и overall_survival_months) должны храниться в виде структурированного массива с первым полем, указывающим, зафиксировано ли фактическое время выживаемости или оно цензурировано, иными словами - произошло заданное событие или нет, а вторым полем, обозначающим наблюдаемое время выживаемости, которое рассчитано до даты

смерти (если Статус == 'умер', 1, True) или время до последнего контакта с этим человеком (если Статус == 'жив', 0, False).

Можно построить несколько кривых Каплана-Мейера на одном графике, разделив датасет на подгруппы (см. ниже пример в практической части). Однако, учитывая описательный характер метода, для оценки различий между группами необходимо использовать лог-ранговый тест.

Если же мы хотим оценить влияние нескольких переменных, необходимо обратиться к линейной модели пропорциональных рисков Кокса.

1.2.2 Модель пропорциональных рисков Кокса

Регрессия Кокса, часто называемая в литературе "пропорциональной моделью Кокса" (ПРК), – наиболее используемый в современных публикациях и рекомендуемый инструмент анализа данных выживаемости. В ее основе лежит метод множественной регрессии, и в качестве выходного параметра модель возвращает значение отношения рисков и его доверительный интервал. Отношение рисков (hazard ratio – HR) – это оценка отношения интенсивностей (показателей, уровней, функции) риска в экспериментальной и контрольной группах, рассчитанная для любого момента времени наблюдения. Модель предполагает, что HR у членов экспериментальной и контрольной групп остается неизменным в течение всего периода наблюдения (предположение о пропорциональности, proportionality assumption). Интенсивность риска представляет собой вероятность того, что событие, не произошедшее к определенному моменту времени, случится в следующий интервал времени, отнесенную к продолжительности этого интервала. Временной интервал может быть установлен очень коротким, поэтому оценку можно делать для любого момента времени. Говоря другими словами и применительно к клиническому испытанию, в котором ожидаемым результатом является, например, выздоровление пациента, HR отражает относительную вероятность

быстрейшего выздоровления у больных, получающих лечение, по отношению к пациентам контрольной группы для любого момента времени [2].

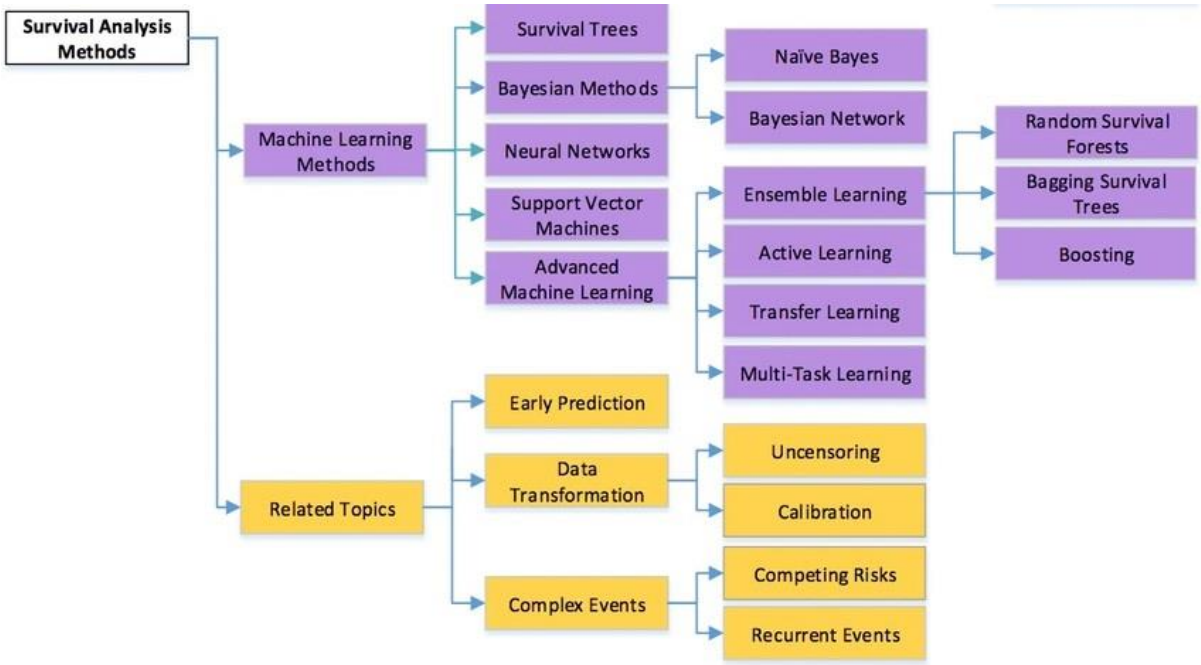
Данная модель позволяет включать в исследование всех интересующих нас пациентов, невзирая на цензурирование (частичную неполноту данных), поскольку использует базисное допущение о том, что пациенты выбывают случайным образом и с одинаковой вероятностью как в изучаемой, так и в контрольной группе. Кроме того, изначально предполагается, что пациенты, у которых произойдет или не произойдет событие, выбывают из исследования с одинаковой вероятностью (правила пропорциональности модели). Для работы с методом необходимо импортировать CoxPHSurvivalAnalysis из sksurv.linear_model.

Модели выживаемости в scikit-survival следуют тем же правилам, что и оценщики в scikit-learn, то есть у них есть метод fit, который ожидает матрицу данных и структурированный массив времени до события и бинарных индикаторов событий [13].

Результатом является вектор коэффициентов, по одному для каждой переменной, где каждое значение соответствует логарифму отношения рисков. Используя подобранную модель, мы можем предсказать функцию выживаемости для конкретного пациента, передав соответствующую матрицу данных методу Predict_survival_function, где каждое значение соответствует логарифму отношение рисков. Подобно kaplan_meier_estimator, метод predict_survival_function возвращает последовательность ступенчатых функций, которую мы можем построить.

Модель пропорциональных рисков Кокса на сегодняшний день является самой популярной моделью выживаемости, потому что после обучения ее легко интерпретировать [3]. Однако, если главной целью является качество прогнозирования, более сложные, нелинейные или ансамблевые модели могут привести к лучшим результатам.

Рисунок 6. Методы машинного обучения, используемые для анализа выживаемости [16]



1.2.3 Лассо (LASSO), гребневая (Ridge), эластичная (Elactic Net) регрессии

Модель пропорциональных рисков Кокса часто является привлекательной моделью, поскольку ее коэффициенты можно интерпретировать с точки зрения отношения рисков, что часто дает ценную информацию. Однако, если мы хотим оценить коэффициенты многих признаков, стандартная модель Кокса разваливается, потому что внутри она пытается инвертировать матрицу, которая становится невырожденной из-за корреляций между признаками.

Например, для данных, содержащих информацию об экспрессии генов, рекомендовано использование регуляризованных линейных моделей (гребневая регрессия, регрессия Лассо, эластическая сеть), реализующих различные способы ограничения весов.

Вышеуказанные регуляризованные регрессионные модели основаны на модели пропорциональных рисков Кокса. Идея состоит в том, чтобы добавить

регуляризацию в функцию частичного правдоподобия Кокса и управлять переобучением [8, 15].

1.2.4 Случайный лес выживаемости (Random Survival Forests)

Как популярные аналоги для классификации и регрессии, Random Survival Forest представляет собой ансамбль из деревья решений. Случайный лес выживаемости гарантирует, что отдельные деревья “декоррелированы” за счет 1) построения каждого дерева на разной бутстрэп-выборке исходных обучающих данных и 2) в каждом узле только оценка критерия разделения для случайно выбранного подмножества признаков и порогов. Прогнозы формируются путем агрегирования прогнозов отдельных деревьев в ансамбль [1, 6, 13].

Для работы с методом необходимо импортировать RandomSurvivalForest из sksurv.ensemble.

1.2.5 Градиентный бустинг (Gradient Boosting Survival Analysis)

Градиентный бустинг относится не к одной конкретной модели, а к универсальной структуре для оптимизации многих функций потерь. Он объединяет прогнозы нескольких базовых алгоритмов для получения более мощной общей модели. Базовые алгоритмы часто представляют собой очень простые модели, которые лишь немногим лучше, чем случайное угадывание, поэтому их также называют “слабыми” [6].

Градиентный бустинг похож на Random Survival Forest в том смысле, что он полагается на несколько базовых алгоритмов для получения общего прогноза, но отличается тем, как они комбинируются. В то время как случайный лес выживаемости соответствует набору деревьев выживаемости независимо, а затем усредняет их прогнозы, модель градиентного бустинга строится последовательно поэтапным способом, позволяя оптимизировать произвольную дифференцируемую функцию потерь.

Для работы с методом необходимо импортировать GradientBoostingSurvivalAnalysis из sksurv.ensemble [13].

1.2.6 Метод опорных векторов для анализа выживаемости (Survival Support Vector Machines)

Главное преимущество метода заключается в том, что он может учитывать сложные нелинейные отношения между признаками и выживаемостью с помощью, так называемого, “ядерного трюка”. Функция ядра неявно отображает входные признаки в многомерные пространства признаков, где выживаемость может быть описана гиперплоскостью. Это делает метод опорных векторов выживаемости чрезвычайно универсальными и применимыми к широкому спектру данных. Популярным примером такой функции ядра является радиальная базисная функция.

Анализ выживания в контексте машин опорных векторов можно описать двумя разными способами:

1. В качестве проблемы ранжирования: модель учится присваивать выборкам с более коротким временем до события более низкий ранг, рассматривая все возможные пары выборок в обучающем сете.
2. В качестве проблемы регрессии: модель учится напрямую предсказывать (логарифмическое) время до события.

В обоих случаях недостатком является то, что прогнозы не могут быть легко связаны со стандартными величинами в анализе выживаемости, а именно, с функцией выживания и кумулятивной функцией риска. Более того, они должны сохранять копию обучающих данных, чтобы делать прогнозы.

Для использования метода необходимо импортировать FastSurvivalSVM и FastKernelSurvivalSVM из sksurv.svm [13].

1.2.7 Нейронная сеть (PyCox, PyTorch)

В 1995 году Фарагги и Саймон представили первое приложение нейронных сетей для анализа выживаемости. В отличие от стандартной модели пропорциональных рисков Кокса, в этой работе использовалась нейронная сеть прямого распространения для выяснения взаимосвязи ковариат с функцией риска [5]. Несколько лет назад Katzman et al. [7] в 2018 повторили этот подход, использовав более сложную архитектуру сети, названную DeepSurv, и функции потерь.

Архитектура DeepSurv показала улучшение модели ПРК и показателей производительности при работе с нелинейными данными. Эта архитектура также смогла справиться с основным ограничением модели ПРК.

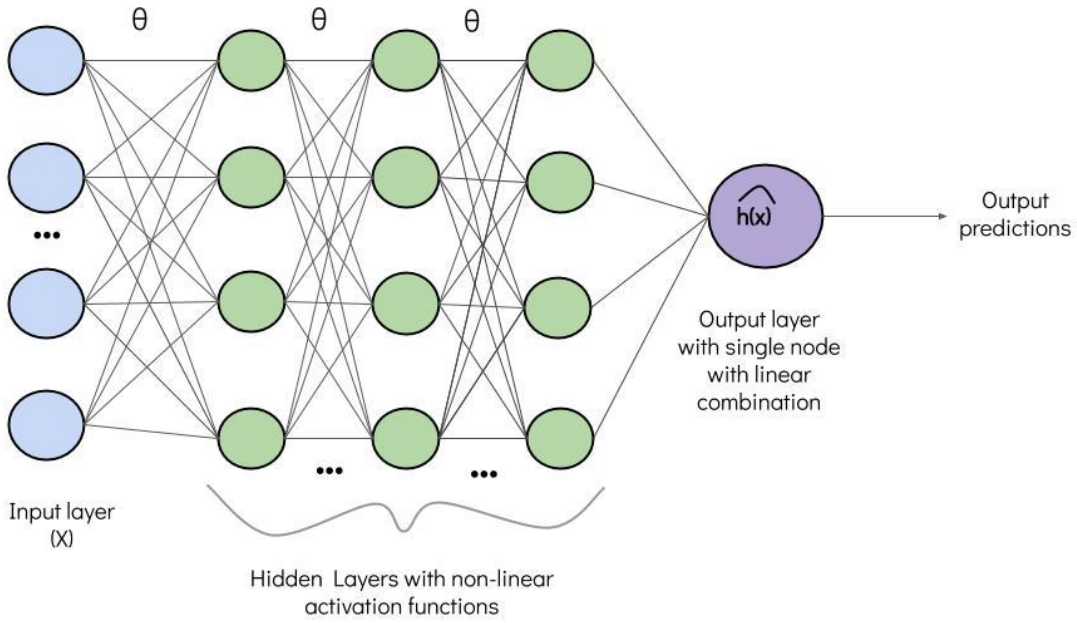
DeepSurv — это нейронная сеть прямого распространения, которая оценивает влияние данных каждого пациента на уровень риска с учетом параметризованных весов сети θ. В целом, структура этой нейронной сети довольно проста. По сравнению с сетью Саймона-Фарраги, DeepSurv имеет несколько настраиваемых скрытых слоев.

Входные данные X представлены набором изучаемых признаков (ковариат).

Скрытые слои в этой модели представляют собой полносвязные нелинейные слои активации с не обязательно одинаковым количеством узлов в каждом из них, за которыми следуют dropout слои.

Выходной слой имеет только один узел с линейной функцией активации, которая дает выход $\hat{h}\ \theta$ (логарифмическая оценка рисков).

Рисунок 7. Структура нейронной сети DeepSurv



Функция потерь для этой сети представляет собой отрицательный логарифм частичного правдоподобия Lc (β) из ПРК с дополнительной регуляризацией:

$$l(\theta) = -\frac{1}{Ne=1}\sum i : ei = 1(\hat{h\beta}(xi) - log\sum j \in R(ti)(e^{\hat{h\beta}(x_j)})) + \lambda * ||\theta||_2^2,$$

где λ — параметр регуляризации l2, а N(e = 1) — множество пациентов с наблюдаемым событием.

Чтобы минимизировать функцию потерь при такой регуляризации, необходимо максимизировать часть в больших скобках. По каждому пациенту i, с наступившим событием, мы увеличиваем фактор риска и цензурированный пациент j, у которого до времени ti событие не наступило, должен иметь минимальный риск.

Для работы с нейронной сетью устанавливаем пакет pycox (https://github.com/havakv/pycox), основанный на среде PyTorch.

1.3 Разведочный анализ данных

Проведем разведочный анализ данных для определения характеристик датасета, связей между признаками, а также выбора методов для создания моделей машинного обучения.

На “тепловой” карте (Рис.4) мы видим слабую корреляцию между клиническими признаками за исключением когорты и ID пациента, что, очевидно, относится к особенностям кодирования и анонимизации, и не является клиническим признаком по сути, а также Ноттингемского прогностического индекса, количества пораженных лимфоузлов, степени злокачественности, стадии, размера опухоли. Однако, эти признаками взаимосвязаны по определению - Ноттингемский индекс рассчитывается с использованием трех признаков: размер опухоли, количество вовлеченных лимфатических узлов, и степень злокачественности опухоли. А стадия опухоли оценивается, исходя из размера опухоли, вовлечения лимфатических узлов и наличия отдаленного метастазирования (Табл. 1).

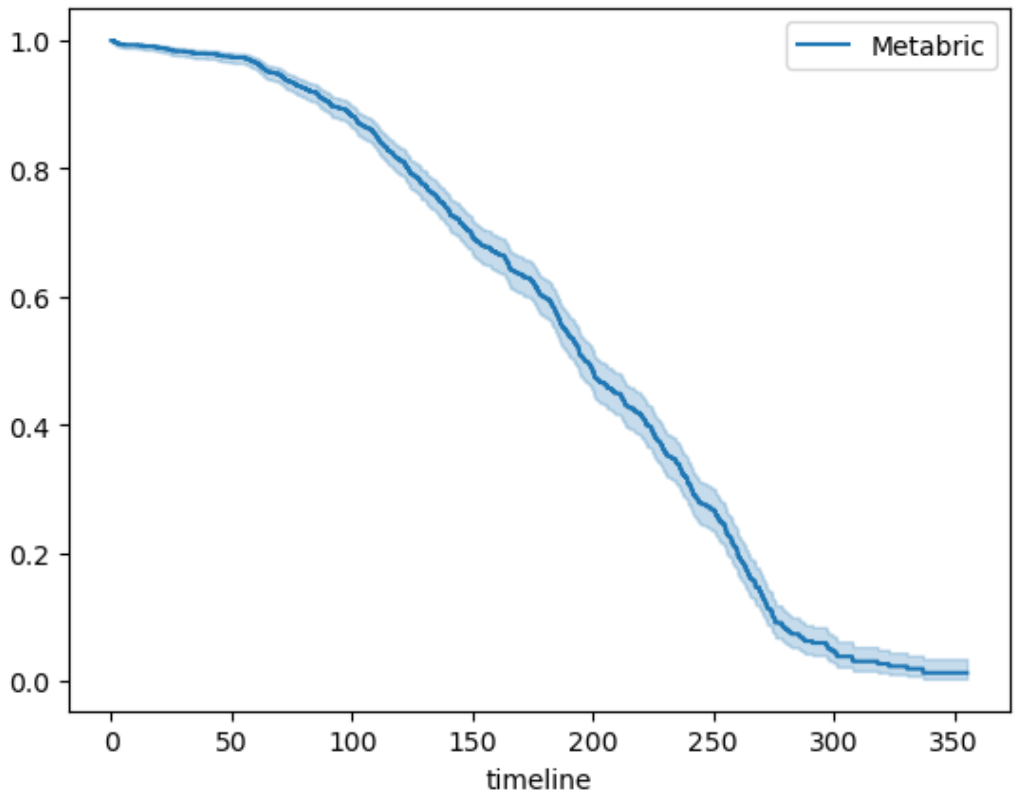
У наших целевых переменных (overall_survival и overall_survival_months) не выражена корреляция с другими числовыми признаками, за исключением слабой корреляции между собой. Несмотря на известную связь между продолжительностью жизни и стадией заболевания, которую можно проверить в любом статистическом сборнике по онкологии, при использовании “тепловой” карты мы обнаруживаем лишь слабую корреляцию между этими признаками.

Учитывая цензурированность данных, мы будем работать с инструментами, позволяющими строить модели, учитывающие цензурирование.

Для описательных целей построим кривые Каплана-Мейера (КМ). А для анализа влияния многих признаков - модель пропорциональных рисков

(Регрессия Кокса). При анализе кривой КМ (достаточно плавный спуск вниз и высокий показатель медианы = 196,87 месяцев) появляются сомнения в валидности данных в датасете. Обратимся к датасету и официальному сайту проекта. Вне зависимости от выбранной библиотеки (lifelines или scikit-survival) очевидно, что в выбранном датасете с Каггла неверно расставлены значения с точки зрения произошедших событий: 0 - умер, 1 - живой. Тогда как в действительности должно быть иначе: 1 - есть событие, т.е. смерть в данном случае, 0 - события нет (это и есть цензурирование).

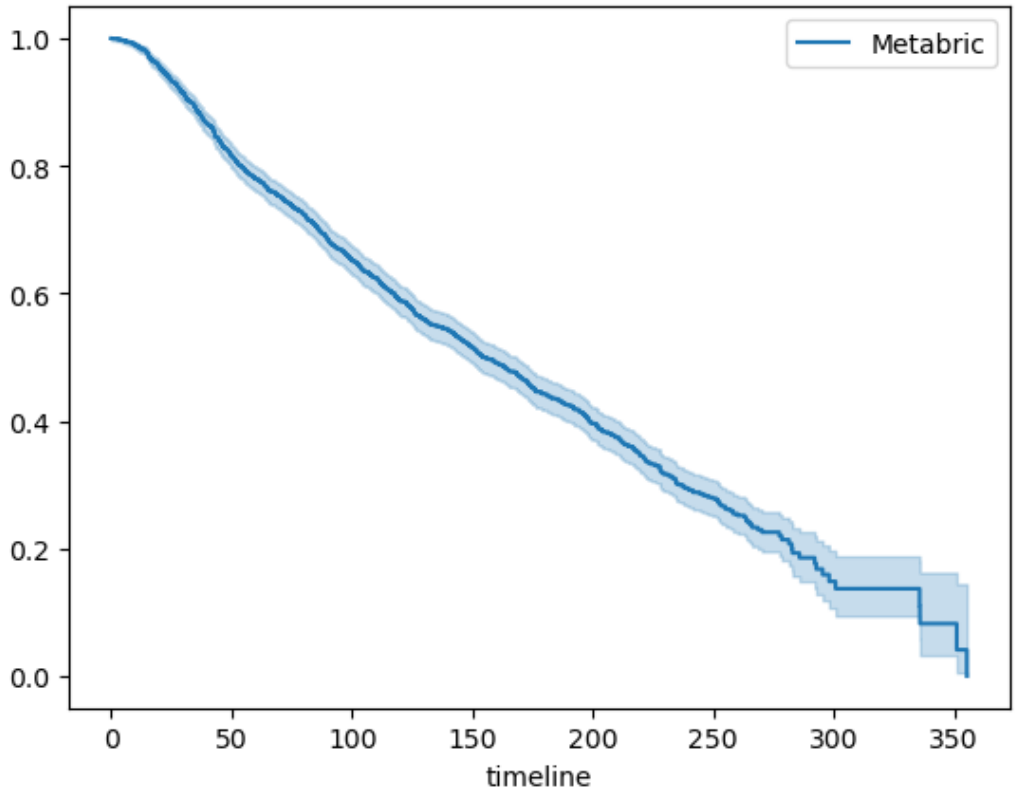
Рисунок 8. Кривая Каплана-Мейера с исходными данными датасета



Данные с сайта проекта www.cbioportal.org подтверждают наше предположение.

Меняем 0 и 1 местами для целевой переменной overall_survival и строим новую кривую КМ.

Рисунок 9. Кривая Каплана-Мейера со скорректированными данными переменной overall_survival

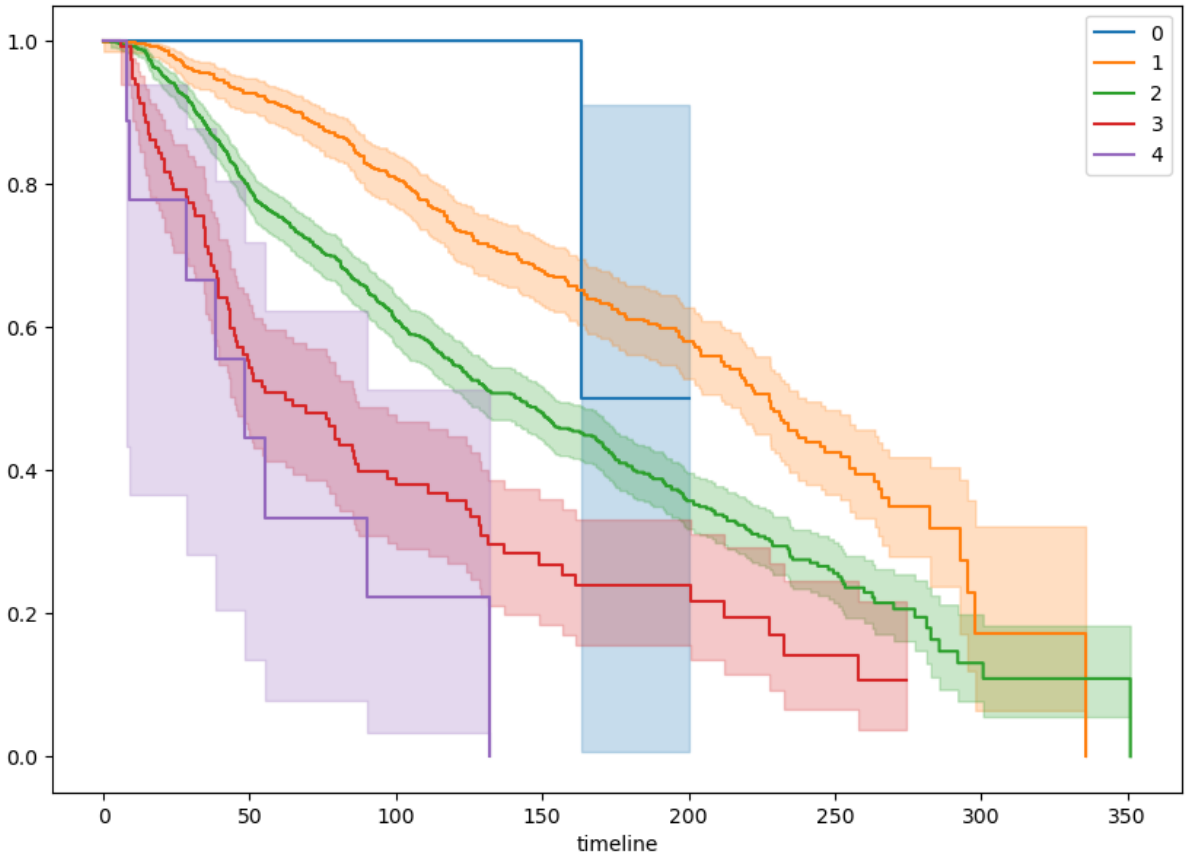


Медиана в данном случае составляет 154,5 месяцев. Возвращаясь к проведенному ранее анализу без поправки на цензурирование, мы видим значительную недооценку медианы (115,62 месяца) в сравнении расчетами по кривой Каплана-Мейера (154,5 месяцев).

Построим кривые КМ в зависимости от стадии заболевания. На рисунке 10 мы можем визуально оценить зависимость времени до события от стадия заболевания - большая стадия заболевания соответствует худшей выживаемости (для определения достоверности отличий необходимо провести лог-ранговый тест, однако мы в данном случае ограничимся визуальным анализом, учитывая его наглядность). Возвращаясь к корреляционному анализу и построенной

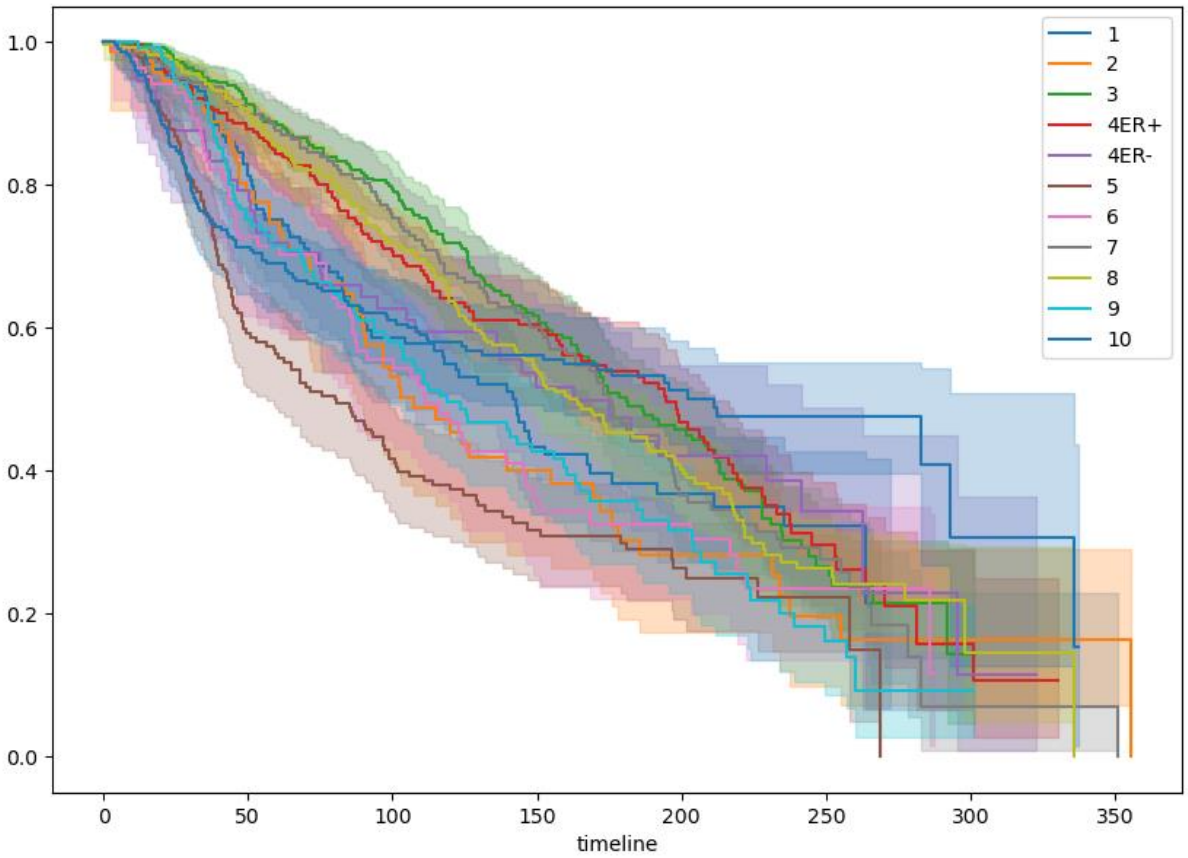
“тепловой” карте без учета цензурированных данных, мы можем видеть недостаточную информативность стандартных методов анализа.

Рисунок 10. Кривые Каплана-Мейера для признака tumor_stage (стадия заболевания)



Расчет медианы и визуальный анализ применим и для категориальных признаков без предварительной предобработки. Например, признак integrative_cluster, см. рисунок 11.

Рисунок 10. Кривые Каплана-Мейера для признака integrative_cluster (классификация в зависимости от молекулярного подтипа)



Учитывая большое количество клинических признаков, от продолжения построения кривых КМ решено воздержаться и перейти к выбору признаков.

1.3.1 Выбор признаков

Наши целевые переменные - overall_survival и overall_survival_months.

Для упрощения дальнейшей работы с клиническими признаками удалим неинформативные - patient_id, cohort, cancer_type (1903 из 1904 это один тип), oncotree_code (он дублирует признак cancer_type_detailed). Также удаляем

death_from_cancer, как очевидно самый информативный признак для выбранных переменных, далее в тексте поясним причину такого решения.

Для работы с генетическими признаками - выбираем столбцы с 31 по 520, которые содержат z-индексы мРНК 331 генов. Столбцы с конкретными 175 мутациями полностью исключаем из анализа.

1.3.2 Ход решения задачи

В рамках решения задачи предусмотрено разделить данные на тренировочную и тестовую выборку. На тестирование решено оставить 20%. Для валидации использовать метод K-fold.

Провести предобработку данных, обучить модели, работающие с анализом выживаемости - как стандартные методами, так и методами машинного обучения. Подобрать гиперпараметры с помощью поиска по сетке. Получить предсказания моделей на тестовой выборке, сравнить метрики и сделать выводы.

1.3.3 Препроцессинг данных

Препроцессинг рекомендуется проводить после разделения на тренировочную и тестовую выборки. Однако в ходе работы, учитывая сложность работы с целевыми переменными, была выбрана тактика препроцессинга до разделения на выборки.

Для числовых признаков проводили стандартизацию при помощи StandardScaler; для категориальных признаков использовали OrdinalEncoder.

Предварительная обработка была реализована при помощи ColumnTransformer. Препроцессинг затем нужно будет повторить в приложении.

Целевые переменные преобразовываем в структурированный массив, согласно требованию библиотеки scikitsurv. Другой обработки с целевыми переменными не проводилось.

1.3.4 Метрики качества модели

В этой работе использовался индекс конкордации Харелла. После обучения модели, мы обычно хотим оценить, насколько хорошо модель может на самом деле предсказать выживаемость. Наша тестовая выборка также цензурирована, поэтому такие показатели, как корень среднеквадратичной ошибки, не подходят. Вместо этого используют обобщение площади под кривой ROC, называемое индексом соответствия/конкордации Харрелла, или c-индексом.

Интерпретация идентична традиционной площади под метрикой кривой ROC для бинарной классификации: - значение 0,5 обозначает случайную модель, - значение 1,0 идеальную модель, - значение 0,0 совершенно неправильную модель.

Индекс определяется как отношение правильно упорядоченных (согласованных) пар к сравниваемым парам. Два образца i и j сопоставимы, если образец с меньшим временем наблюдения Y достигает события, т. е. если $Yj > Yi$, и $\delta i = 1$, где δi является бинарным индикатором событий.

Сопоставимая пара (i, j) является конкордантной, если оцениваемый риск f в модели выше у субъектов с меньшим временем до события, иначе пара считается дискордантной.

2. Практическая часть

Для анализа выживаемости выбираем следующие модели:

- регрессия Кокса (модель пропорциональных рисков Кокса, раздел 1.2.2);

- регуляризованные модели Кокса (penalized Cox Models, включающие гребневую регрессию, регрессию Лассо, эластичную сеть, раздел 1.2.3);
- случайный лес выживаемости (Random Survival Forests, раздел 1.2.4);
- градиентный бустинг (Gradient Boosting Survival Analysis, раздел 1.2.5);
- метод опорных векторов для анализа выживаемости (Survival Support Vector Machine, раздел 1.2.6);
- нейронные сети (DeepSurv, раздел 1.2.7).

2.1. Стандартные методы анализа выживаемости

Модели выживаемости в scikit-survival следуют тем же правилам, что и алгоритмы в scikit-learn, то есть у них есть метод обучения (fit), который ожидает матрицу данных и структурированный массив времени до события и бинарного индикатора события (1 - событие есть, 0 - события нет).

Результатом примененной регрессии Кокса является вектор коэффициентов, по одному для каждой переменной, где каждое значение соответствует логарифмическому коэффициенту риска.

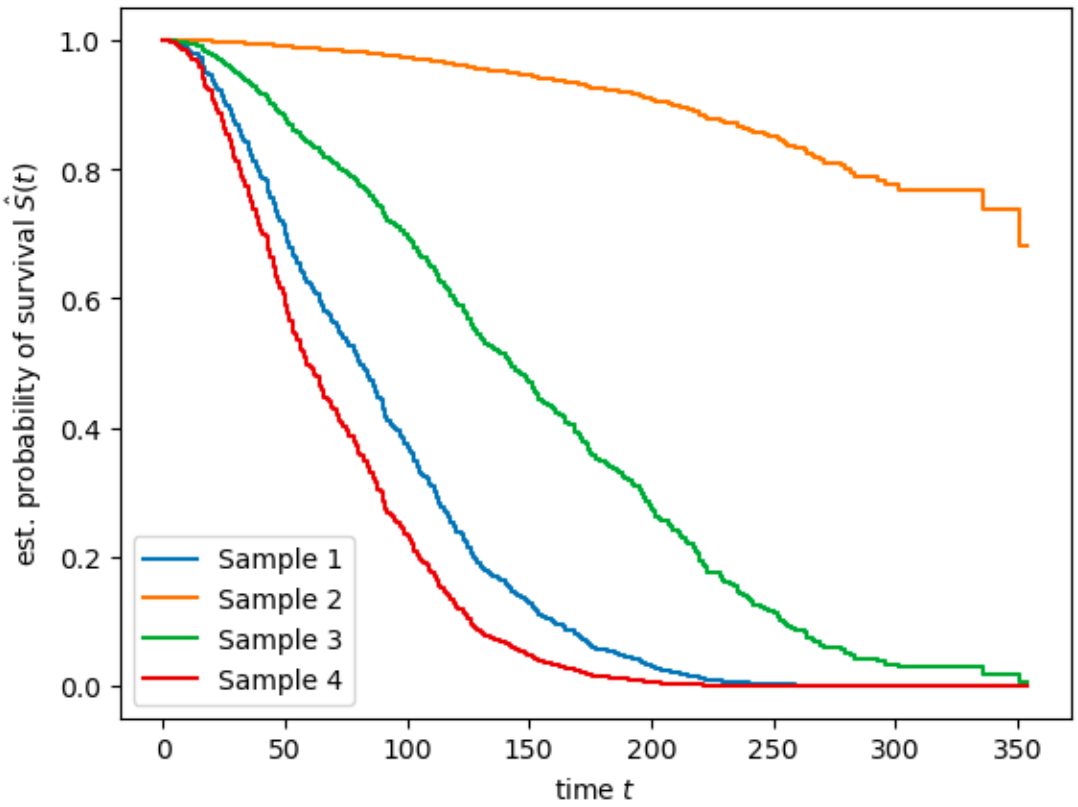
Рисунок 11. Коэффициенты для 25 признаков клинической части датасета

```
Out[27]: CoxPHSurvivalAnalysis()

In [28]: pd.Series(estimator.coef_, index=X_train.columns)

Out[28]: num__age_at_diagnosis                  0.485499
         num__chemotherapy                      0.130871
         num__neoplasm_histologic_grade        -0.008600
         num__hormone_therapy                   0.031980
         num__lymph_nodes_examined_positive     0.070330
         num__mutation_count                    0.012590
         num__nottingham_prognostic_index       0.036380
         num__radio_therapy                    -0.079629
         num__tumor_size                        0.050466
         num__tumor_stage                       0.115715
         cat__type_of_breast_surgery           -0.023190
         cat__cancer_type_detailed              0.098371
         cat__cellularity                      -0.046094
         cat__pam50_+_claudin-low_subtype      -0.034079
         cat__er_status_measured_by_ihc        -0.100438
         cat__er_status                        -0.599246
         cat__her2_status_measured_by_snp6     -0.184157
         cat__her2_status                       0.050821
         cat__tumor_other_histologic_subtype   -0.108593
         cat__inferred_menopausal_state         0.328219
         cat__integrative_cluster              -0.012303
         cat__primary_tumor_laterality         -0.033006
         cat__pr_status                        -0.144112
         cat__3-gene_classifier_subtype        -0.089480
         cat__death_from_cancer                -1.604850
         dtype: float64
```

Используя обученную модель, мы можем предсказать функцию выживания для конкретного пациента, передав соответствующую матрицу данных методу Predict_survival_function.

Вводим данные для 4 синтетических пациентов и при помощи вышеуказанного метода получаем последовательность ступенчатых функций, которую мы можем отразить на графике.

Рисунок 12. Функции выживания для 4 синтетических пациентов



Индекс конкордантности Харелла составил 0,84, что очень хорошо для этой модели.

Отберем признаки, которые являются лучшим предиктором риска для данной модели.

Рисунок 13. C-индекс по каждому признаку

```
Out[34]: cat__death_from_cancer                  0.813584
         num__nottingham_prognostic_index        0.653552
         num__tumor_size                         0.626544
         num__lymph_nodes_examined_positive      0.619861
         num__age_at_diagnosis                   0.601891
         num__tumor_stage                        0.590635
         cat__type_of_breast_surgery             0.569659
         num__neoplasm_histologic_grade          0.569173
         cat__primary_tumor_laterality           0.544821
         cat__pr_status                          0.544082
         num__chemotherapy                       0.538274
         cat__pam50_+_claudin-low_subtype        0.537048
         cat__integrative_cluster                0.536231
         cat__her2_status_measured_by_snp6       0.535803
         cat__er_status                          0.529537
         cat__er_status_measured_by_ihc          0.529294
         num__hormone_therapy                    0.522386
         cat__inferred_menopausal_state          0.520256
         cat__tumor_other_histologic_subtype     0.515751
         num__mutation_count                     0.514720
         cat__her2_status                        0.512609
         cat__cellularity                        0.505273
         num__radio_therapy                      0.504670
         cat__3-gene_classifier_subtype          0.503950
         cat__cancer_type_detailed               0.486457
         dtype: float64
```

Самый высокий показатель у переменной cat__death_from_cancer = 0,81. Очевидно, что данная переменная отчасти дублирует наши целевые переменные, поэтому мы убираем ее из X (оставим в итоге 24 признака) и обучаем модель заново.

Коэффициенты и функции выживания для синтетических пациентов с 24 переменными не привожу, они представлены в Jupiter ноутбуке Metabric_regression_2.

Индекс конкордантности после наших преобразований с датасетом в виде исключения очевидно связанной переменной - 0,69.

Если мы хотим построить более “компактную” модель, исключив ненужные переменные, можно использовать ранжирование по убыванию, но необходимо определить, каким должно быть оптимальное отсечение. К счастью, scikit-learn имеет встроенный поиск по сетке (GridSearch). Нам нужно определить

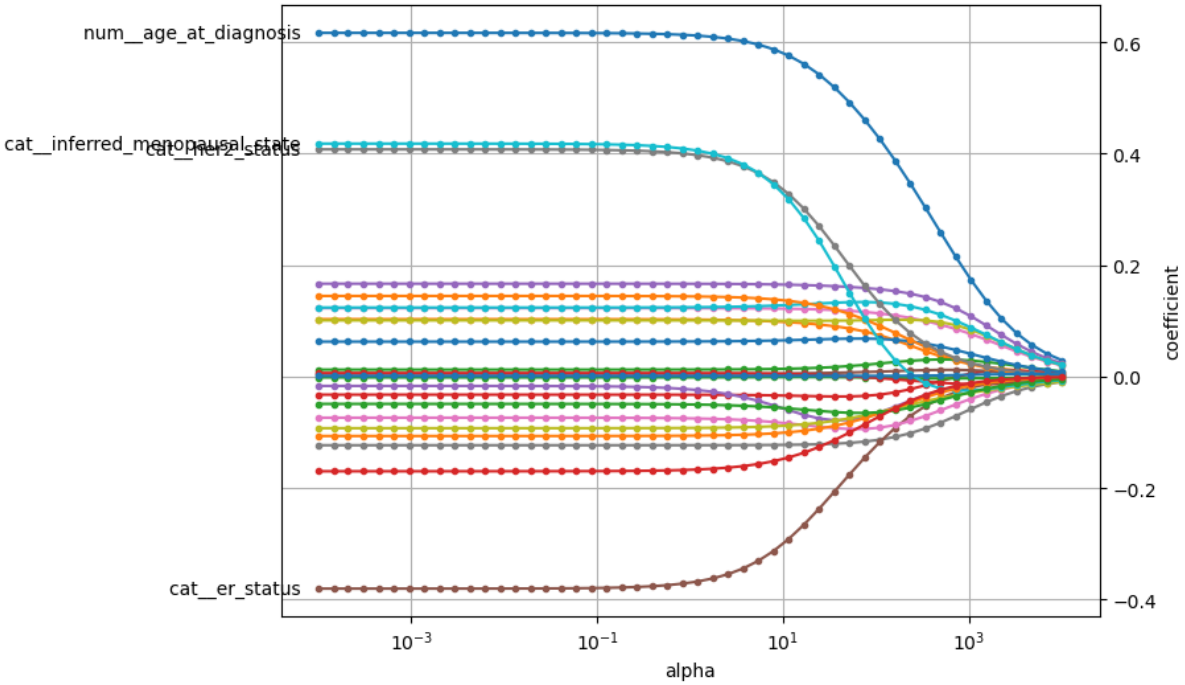
диапазон параметров, которые мы хотим исследовать во время поиска по сетке. Мы хотим оптимизировать параметр k класса SelectKBest и позволить k варьироваться от 1 до 24 признаков.

Согласно проведенному анализу наилучшим выбором являются 24 параметра, что совершенно не помогает в построении “компактной” модели.

Гребневая регрессия.

Начнем с подгонки модели Кокса с регуляризацией к различным значениям α, используя sksurv.linear_model.CoxPHSurvivalAnalysis и записывая коэффициенты, которые мы получили для каждого α. На рисунке 14 мы можем наблюдать за изменением коэффициентов при изменении α.

Рисунок 14. Гребневая регрессия для клинических признаков



Мы видим, что если α имеет большой вес (справа), то все коэффициенты уменьшаются почти до нуля. По мере уменьшения веса α значение коэффициентов увеличивается. Мы также можем наблюдать, что num__age_at_diagnosis, cat__inferred_menopausal_state, cat__her2_status и cat__er_status быстро отделяются от остальных коэффициентов, что указывает на то, что эти конкретные признаки являются важными прогностическими

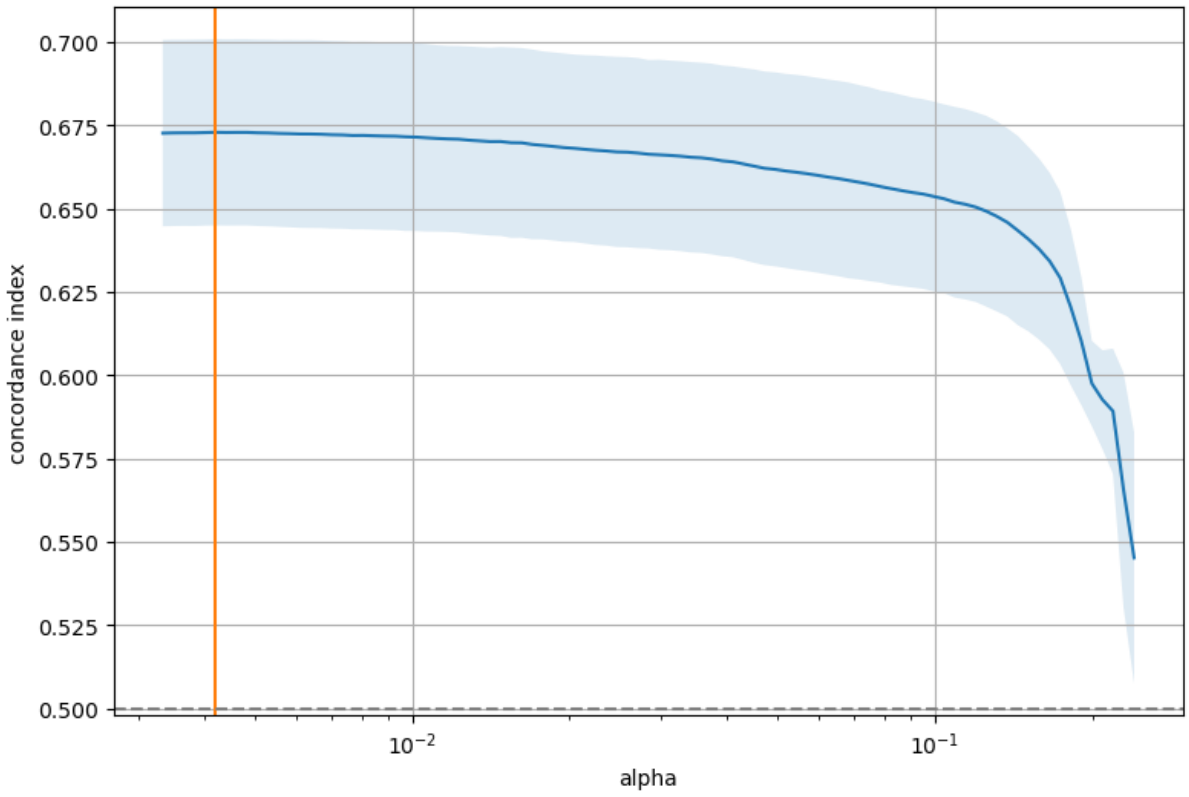
факторами для времени до события (смерти). Регрессия Лассо и эластичная сеть выдают аналогичные результаты.

Однако для предсказания нам нужно выбрать один конкретный α, и подмножество признаков. Здесь мы собираемся использовать перекрестную проверку, чтобы определить, какое подмножество и α обобщаются лучше всего.

Прежде чем мы сможем использовать GridSearchCV, нам нужно определить набор α, которые мы хотим оценить. Для этого мы обучаем регуляризованную модель Кокса на всех данных и получаем предполагаемый набор α.

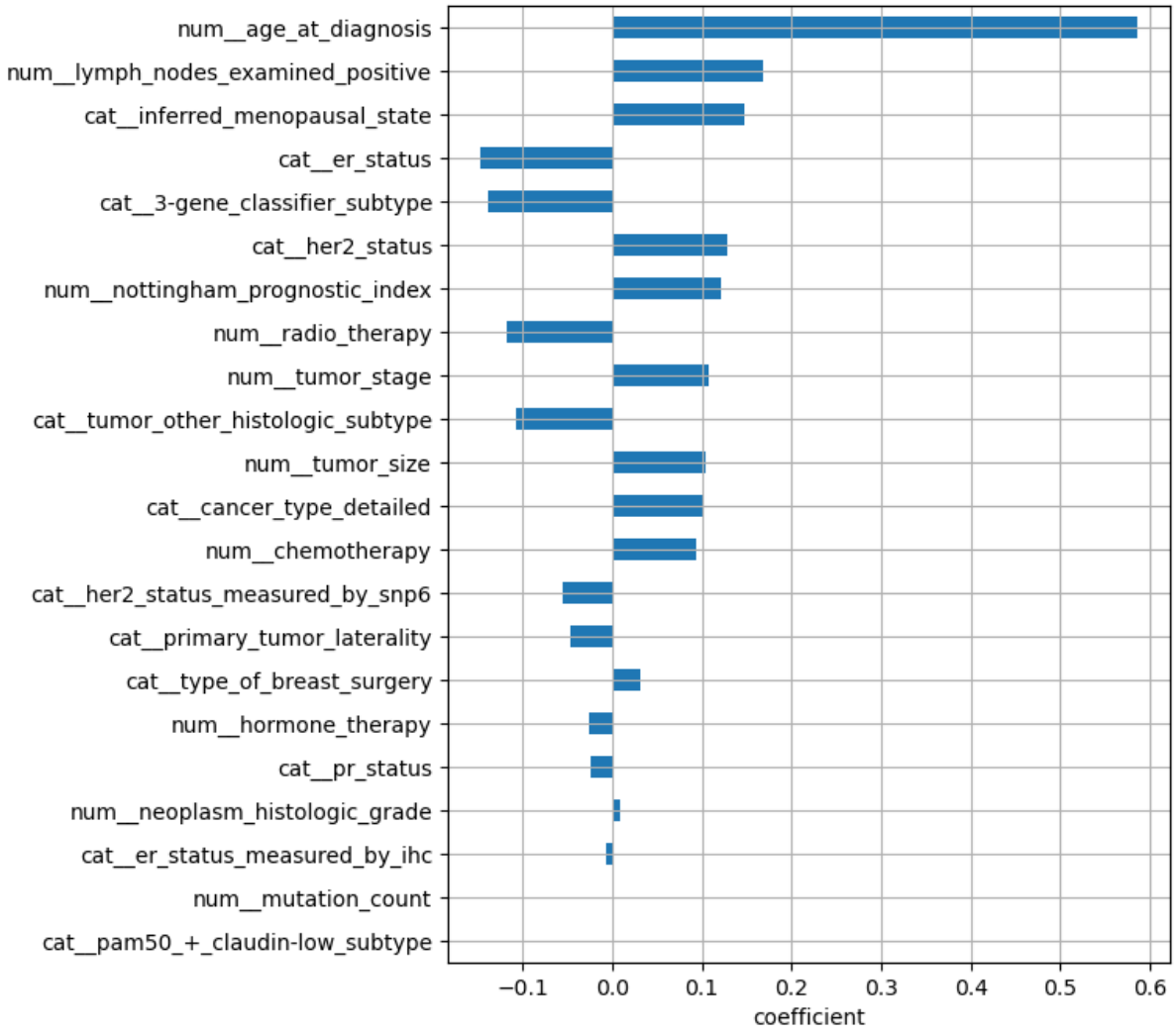
Используя предполагаемый набор α, мы выполняем 5-кратную перекрестную проверку, чтобы оценить производительность — с точки зрения индекса конкордантности для каждого α. Результаты представлены на рисунке 15.

Рисунок 15. Выбор наилучшего α регуляризованной регрессии для клинических признаков



На рисунке видно, что существует диапазон для α вправо, где он слишком велик, и обнуляет все коэффициенты, на что указывает индекс соответствия 0,5 чисто случайной модели. В другой крайности, если α становится слишком маленьким, в модель входит слишком много признаков, и производительность в нашем случае является наилучшей. Лучшая точка (оранжевая линия) как раз слева. Давайте проверим эту модель. На рисунке 16 мы видим, что модель выбрала 1 признак - возраст на момент диагноза как наиболее прогностический важный.

Рисунок 16. Выбор клинических признаков при наилучшем α



Выбрав конкретный α, мы можем выполнить прогноз для конкретного пациента либо с точки зрения оценки показателя риска, используя predict

function, либо с точки зрения функции выживания или функции кумулятивного риска.

Генетические признаки.

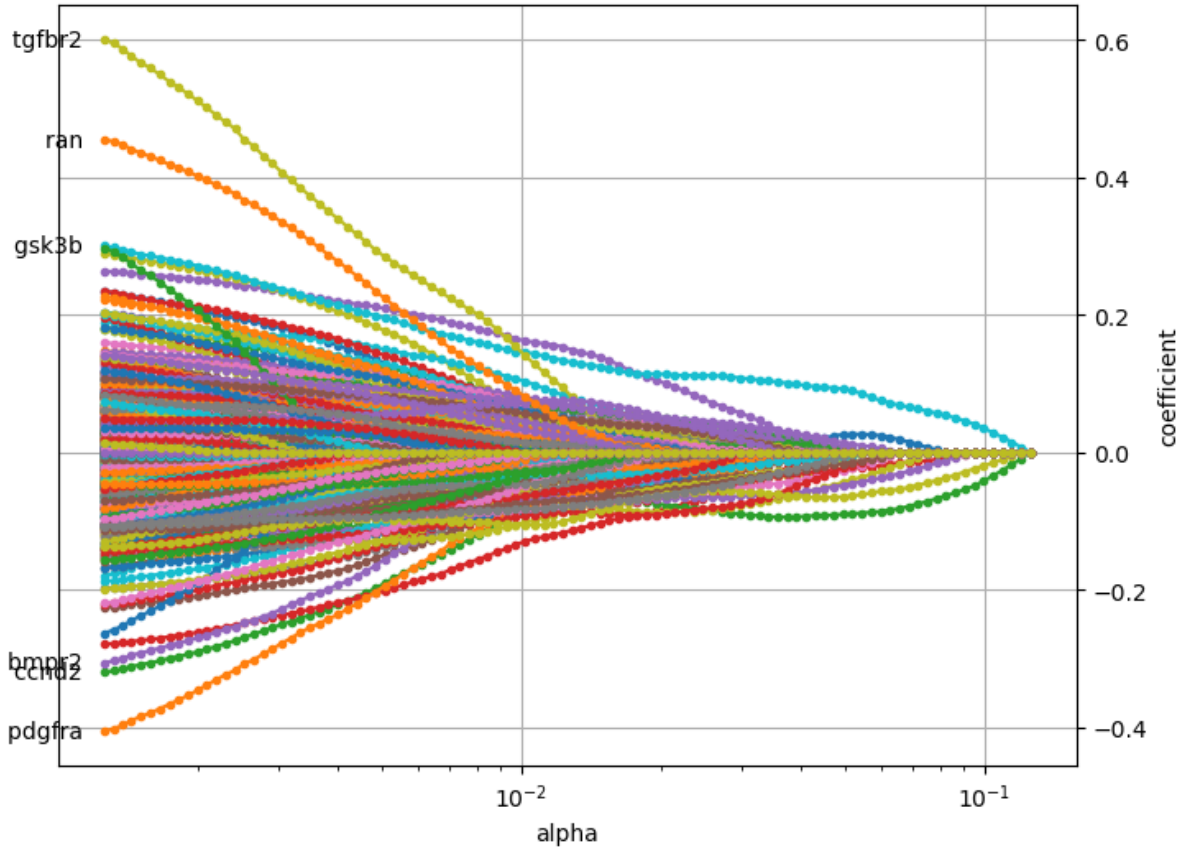
От полного описания проведенного анализа регрессии Кокса и регуляризованной регрессии для генетических признаков решено воздержаться и остановиться лишь на нескольких моментах.

Важно отметить, что индекс конкордантности для регрессии Кокса составил 0,59, что лишь немногим выше случайной модели.

Приступим к регудяризованной регрессии. Нам необходимо измерить уровни экспрессии всех 489 генов, чтобы сделать прогноз. В идеале мы хотели бы выбрать небольшое подмножество признаков, которые наиболее предсказуемы и игнорируют оставшиеся уровни экспрессии генов. Это именно то, что делает регрессия LASSO (Least Absolute Shrinkage and Selection Operator). Вместо сокращения коэффициентов до нуля она выполняет тип непрерывного выбора подмножества, когда подмножество коэффициентов устанавливается равным нулю и эффективно исключается. Это уменьшает количество признаков, которые нам нужно было бы включить в предсказание.

Основная проблема заключается в том, что мы не можем напрямую контролировать количество выбранных признаков, но величина α неявно определяет их количество. Таким образом, нам нужен управляемый данными способ выбора подходящей α и получения “компактной” модели. Мы можем сделать это, сначала вычислив α, которая будет игнорировать все признаки (все коэффициенты равны нулю), а затем постепенно уменьшать ее значение до тех пор, пока мы не достигнем 1% от исходного значения. Этот подход реализован в sksurv.linear_model.CoxnetSurvivalAnalysis путем указания L1_ratio=1.0 для использования регуляризации LASSO и alpha_min_ratio=0.01 для поиска 100α значений до 1% от предполагаемого максимума.

Рисунок 17. Регрессия LASSO для генетических признаков



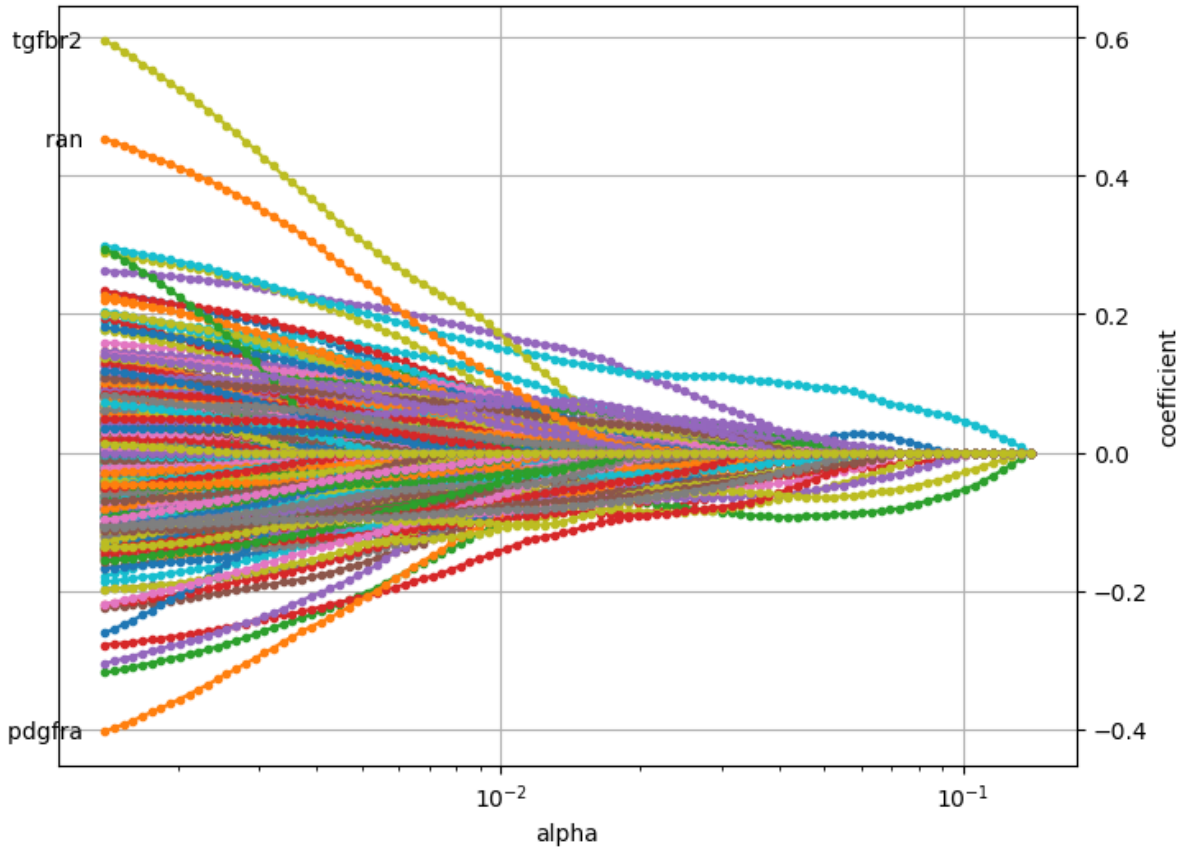
На рисунке 17 показано, что регуляризация LASSO действительно выбирает небольшое подмножество признаков для больших значений α (справа), только три признака (голубая, зеленая и желтая линии) не равны нулю. По мере уменьшения α, все больше и больше признаков становятся активными и им присваивается ненулевой коэффициент до тех пор, пока не будет использован весь набор признаков (слева). Как и на графике выше для гребневого регрессии, выделяются tgfbr2, ran, pdgfra, что указывает на их важность при раке молочной железы. Наиболее важным признаком, по-видимому, является tgfbr2.

LASSO - отличный инструмент для выбора подмножества отличительных признаков, но у него есть два основных недостатка. Во-первых, он не может выбрать больше признаков, чем количество образцов(пациентов в нашем случае) в обучающих данных, что проблематично при работе с данными очень большой

размерности. Во-вторых, если данные содержат группу признаков с высокой степенью корреляции, регуляризация LASSO будет случайным образом выбирать один признак из этой группы.

Эластичная сеть с двумя регуляризаторами сочетает в себе свойство выбора подмножества LASSO с силой регуляризации гребневой регрессии. Это приводит к лучшей стабильности по сравнению со моделью LASSO. Для группы сильно коррелированных признаков LASSO выбирает один признак случайным образом, в то время как эластичная сеть стремится выбрать все. Обычно достаточно задать L2 небольшой вес, чтобы улучшить стабильность LASSO, например, установив r=0,9.

Рисунок 18. Эластичная сеть для генетических признаков



По аналогии с клиническими признаками определяем наилучший α и наилучшие прогностические признаки из 61 (график доступен в Jupiter ноутбуке

Metabric_regression_2). Наилучший из признаков - gsk3b (см. голубой путь на рисунке 17).

2.2. Методы машинного обучения в анализе выживаемости

Случайный лес выживаемости.

Недостатком модели пропорциональных рисков Кокса является то, что она может предсказывать только оценку риска, не зависящую от времени (из-за встроенного предположения о пропорциональных рисках). Таким образом, одна прогнозируемая оценка риска должна хорошо работать для каждой временной точки. Напротив, случайный лес выживаемости не имеет этого ограничения.

В прошлом было предложено несколько разделяющих критериев, но наиболее распространенный из них основан на тесте лог-ранк, который используется для сравнения кривыч выживаемости между двумя или более группами. Используя обучающую выборку, обучаем модель, состоящую из 1000 деревьев.

Индекс конкордантности для клинических признаков - 0,71. Результат неплохой и немного превосходит регрессию Кокса. Для генетических признаков - 0,63.

Для прогнозирования образец передается каждому дереву в лесу, пока не достигнет конечного узла. Данные в каждом терминале используются для непараметрической оценки функции выживания и кумулятивного риска с использованием оценок Каплана-Мейера и Нельсона-Аалена соответственно. Кроме того, может быть вычислена величина риска, которая представляет ожидаемое количество событий для одного конкретного конечного узла. Предсказание ансамбля — это просто среднее значение по всем деревьям в лесу.

Отфильтруем выборку по возрасту и Ноттингемскому индексу и построим предсказание для 6 пациентов. Предсказанные показатели риска значительно выше для последних пациентов (2-5, см. Jupiter ноутбук Metabric_RSF_3).

Важность перестановочного признака.

Проведем оценку важности признаков при помощи метода permutation_importance function библиотеки scikit-learn. Результат анализа показал, что возраст на момент диагноза является наиболее важным признаком. Если убрать его связь со временем до события (случайным образом), индекс конкордации по тестовым данным падает в среднем на 0.065319 балла.

Анализ по генетическим признакам представлены в Jupiter ноутбук Metabric_RSF_3.

Градиентный бустинг.

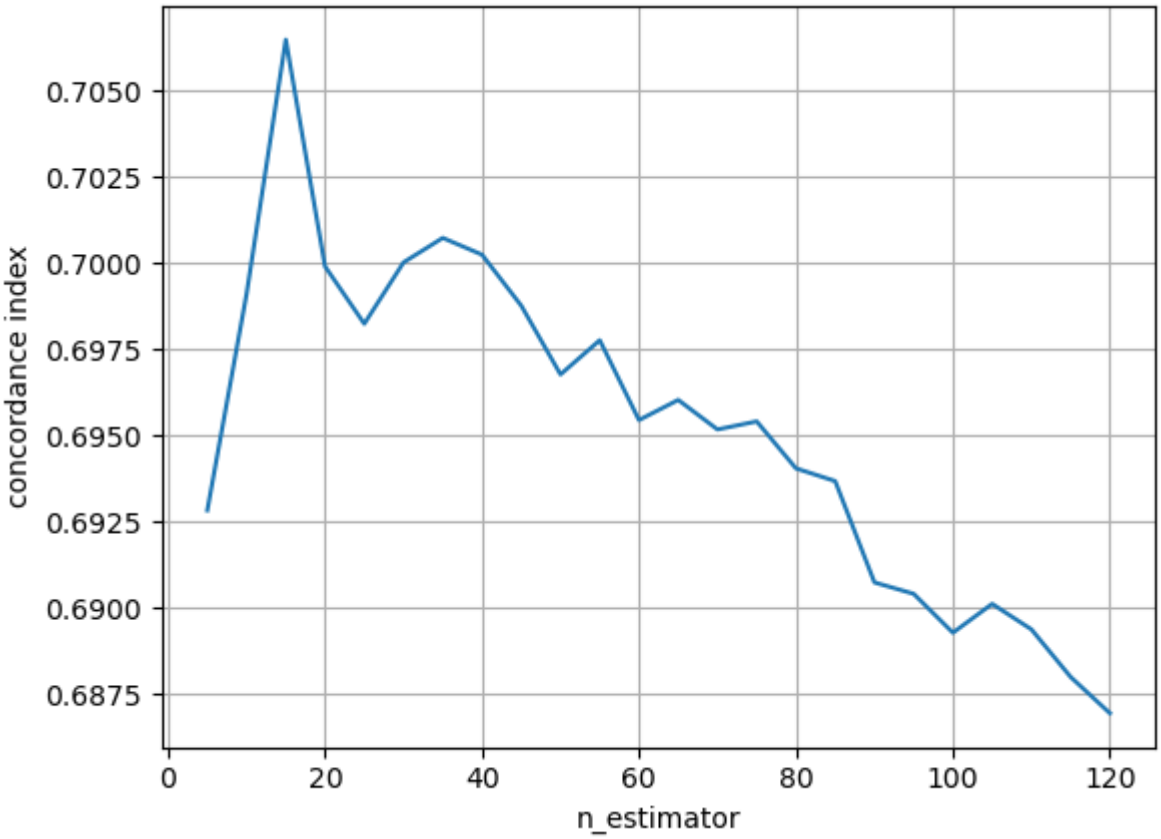
В зависимости от минимизируемой функции потерь и используемого базового обучающего алгоритма можно использовать разные модели. sksurv.ensemble.GradientBoostingSurvivalAnalysis реализует повышение градиента с помощью базового обучающего алгоритма деревьев регрессии, а sksurv.ensemble.ComponentwiseGradientBoostingSurvivalAnalysis использует покомпонентный метод наименьших квадратов в качестве базового обучающего алгоритма. Первый очень универсален и может объяснить сложные нелинейные отношения между функциями и временем до события. При использовании покомпонентного метода наименьших квадратов в качестве базового алгоритма окончательная модель будет линейной моделью, но будет выбрано только небольшое подмножество функций, аналогично модели Кокса с регуляризацией LASSO.

Функция потери может быть указана с помощью аргумента потери; функция потерь по умолчанию представляет собой потерю частичной вероятности регрессии Кокса (coxph).

Мы используем градиентный бустинг на частичном правдоподобии Кокса с базовыми обучаемыми деревьями регрессии, которые мы ограничиваем использованием только одного разделения (так называемые пни).

Модель достигает индекса конкордантности 0,71 на тестовых данных. На рисунке 19 мы видим, как производительность теста меняется с размером ансамбля (n_estimators).

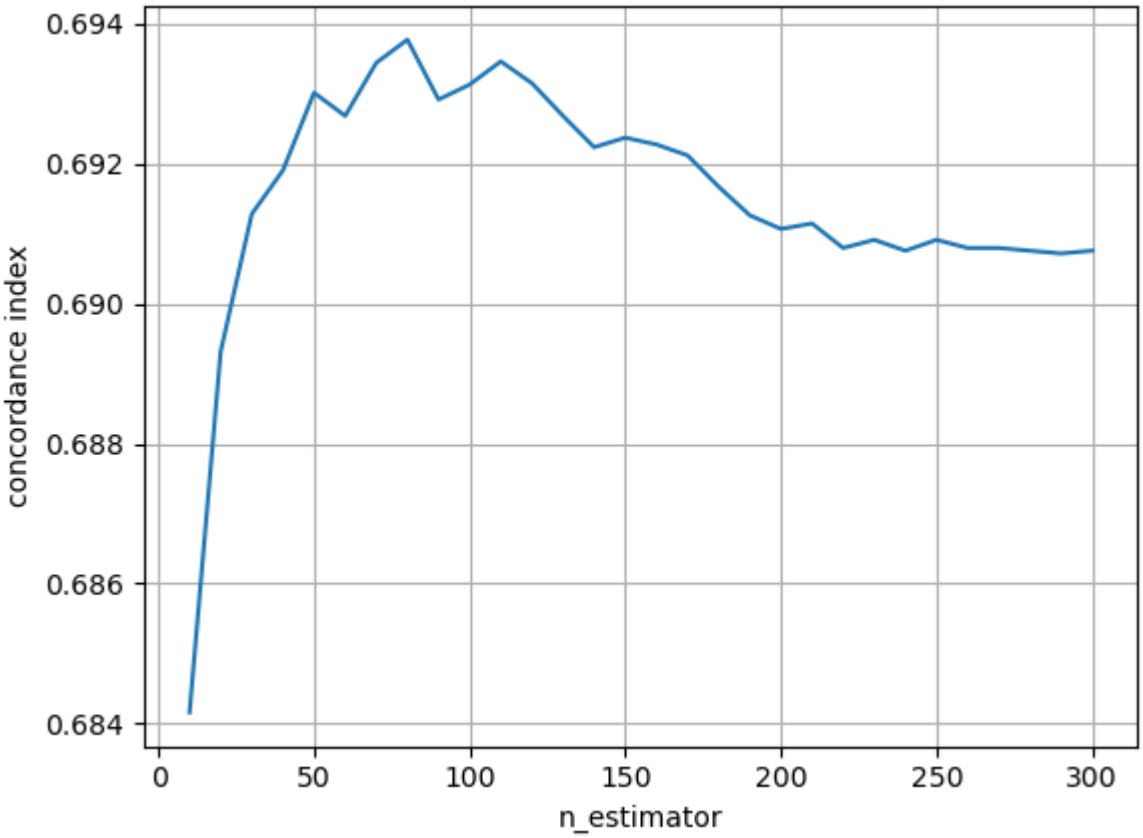
Рисунок 19. Производительность бустинга в зависимости от размера ансамбля



Производительность максимальна при небольшоем размере и стремительно начинает падать при увеличении размера ансамбля.

Повторим анализ, используя покомпонентный метод наименьших квадратов в качестве базового алгоритма (Рис.20).

Рисунок 20. Производительность бустинга при использования покомпонентного метода наименьших квадратов



Прирост производительности здесь намного медленнее, и его максимальная производительность ниже, чем у ансамбля на основе деревьев. Это неудивительно, потому что покомпонентный метод наименьших квадратов в качестве базового алгоритма представляет собой линейную модель, тогда как в случае алгоритмов на основе деревьев - нелинейная модель.

Метод опорных векторов.

Анализ выживаемости методом опорных векторов можно описать двумя разными способами:

- в качестве проблемы классификации: модель учится присваивать выборкам с более коротким временем до события более низкий ранг, рассматривая все возможные пары выборок в обучающих данных.

- в качестве проблемы регрессии: модель учится напрямую предсказывать (логарифмическое) время выживания.

В обоих случаях недостатком является то, что прогнозы не могут быть легко связаны со стандартными понятиями в анализе выживаемости, а именно с функцией выживания и кумулятивной функцией риска. Более того, они должны сохранять копию обучающих данных, чтобы делать прогнозы.

Сфокусируемся на линейной модели выживаемости методом опорных векторов, которая не позволяет выбирать специальные ядерные функции, но гораздо быстрее обучается.

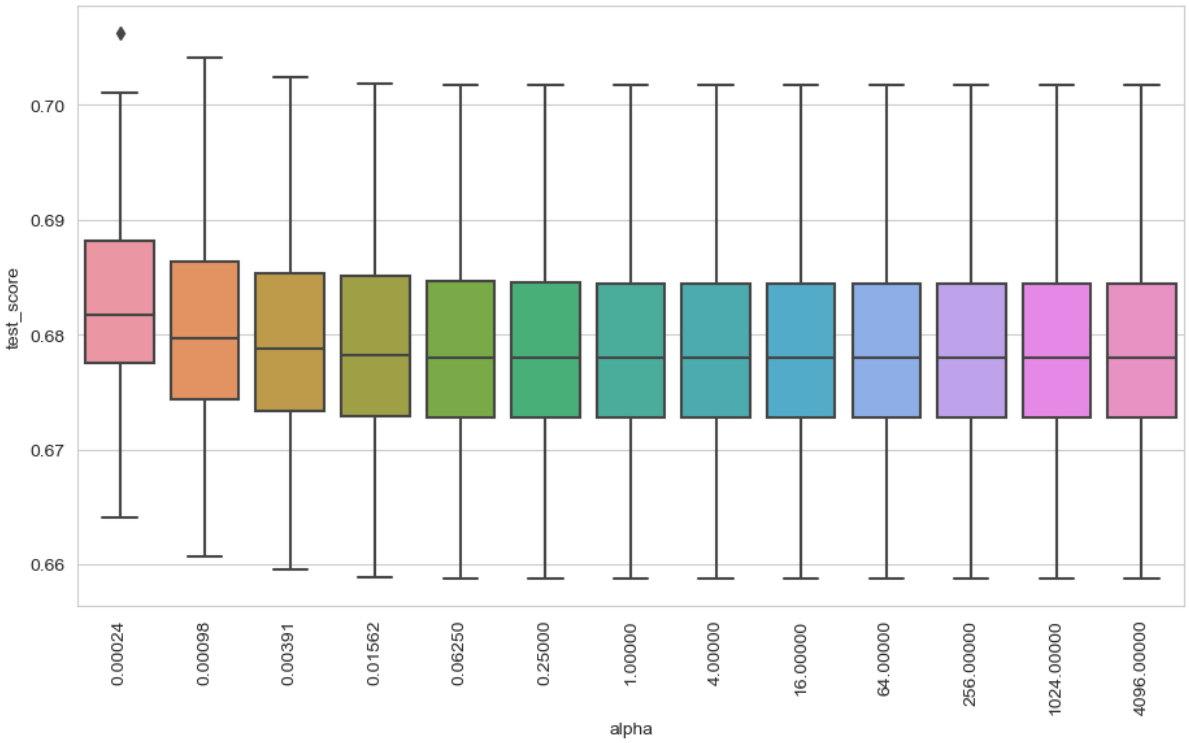
Гиперпараметр $\alpha > 0$ определяет объем применяемой регуляризации: меньшее значение увеличивает количество регуляризации, а более высокое значение уменьшает количество регуляризации. Гиперпараметр $r \in [0,1]$ определяет компромисс между целью классификации и целью регрессии. r=1 сводит к цели классификации, r=0 - к цели регрессии.

Класс sksurv.svm.FastSurvivalSVM поддерживает интерфейсы, используемые в scikit-learn, поэтому его можно комбинировать со вспомогательными классами и функциями из scikit-learn. В этом примере мы собираемся использовать классификацию (r=1) и GridSearchCV, чтобы определить наилучшую настройку для гиперпараметра α.

Создадим исходную модель с параметрами по умолчанию, которые затем передадим в поиск по сетке. Для оценки используем индекс конкордантности.

В результате получаем наилучшую среднюю производительность среди 100 полученных нами случайных разделений train/test и гиперпараметров - 0,682.

Рисунок 21. Выбор наилучшего гиперпараметра



Обучим модель с наилучшим α. Важно помнить, что при r=1, прогнозы обозначают оценку риска - более высокое прогнозное значение указывает на более короткое время до события, а более низкое значение — на более длительную выживаемость.

```
pred = estimator.predict(X_cat.iloc[9:11])
print(np.round(pred, 3))
print(y[9:11])
```

```
[1.624 1.448]
    overall_survival_months  overall_survival
9                 36.266667              True
10               132.033333              True
```

Если используется регрессия, то семантика отличается, потому что теперь прогнозы находятся на временной шкале, и более низкие прогнозируемые значения указывают на более короткую выживаемость, более высокие значения — на более длительную выживаемость. Более того, из приведенной выше гистограммы наблюдаемого времени мы увидели, что распределение искажено,

поэтому время выживания/цензурирования будет логарифмически преобразовано FastSurvivalSVM внутри при использовании r < 1.

Нейронная сеть.

Построим самую простую нейронную сеть во фреймворке PyTorch.

Pycox построен на основе PyTorch и torchtuples, где последний представляет собой простой способ обучения нейронных сетей.

Импортируем LogisticHazard метод и EvalSurv для последующей оценки.

Для простой сети мы можем использовать MLPVanilla, обучаемую torchtuples. Построим сеть с 2 скрытыми слоями по 32 нейрона, активационная функция ReLu.

Для обучения модели нам нужно выбрать оптимизатор. Здесь мы будем использовать оптимизатор Adam со скоростью обучения 0,01.

Далее мы устанавливаем batch size в размере 256 и количество эпох обучения в размере 100.

Мы также добавляем раннюю остановку, чтобы остановить обучение, когда метрики на валидационной выборке перестанут улучшаться.

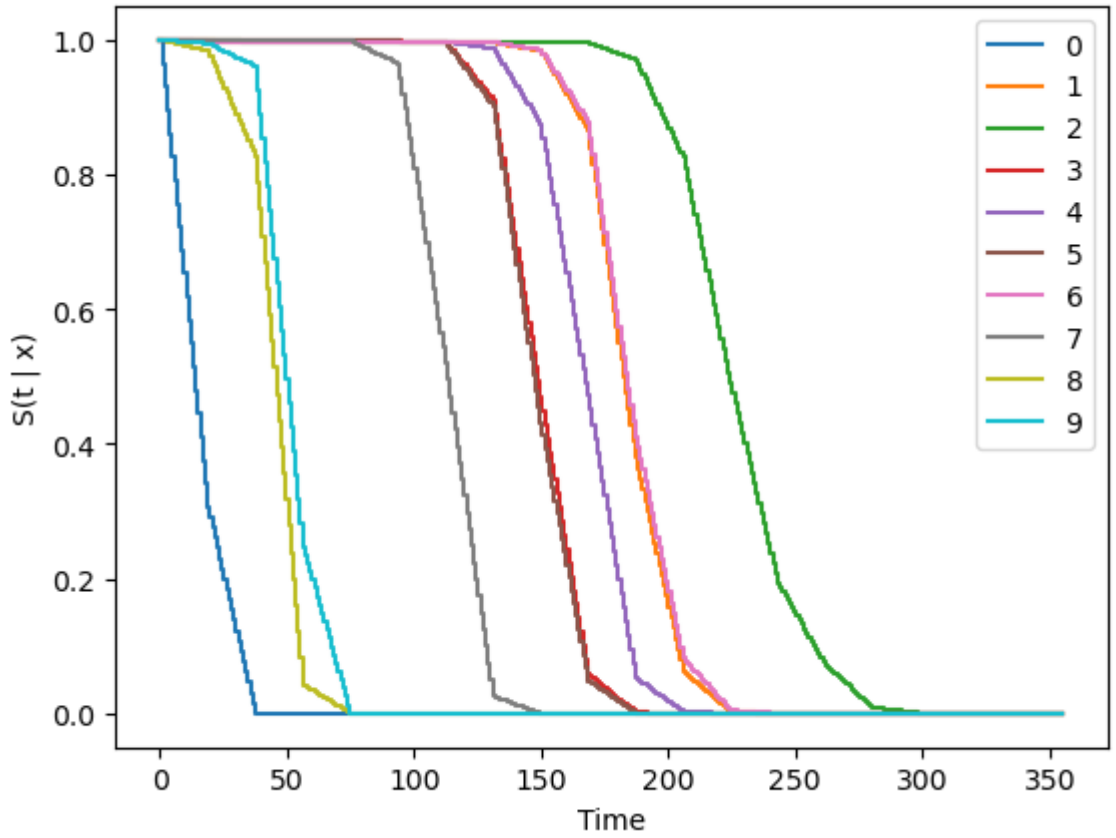
После завершения выгружаем наиболее эффективную модель (с точки зрения метрики валидационной выборки).

```
In [168]: log.to_pandas().val_loss.min()
Out[168]: 0.4623577296733856

In [169]: model.score_in_batches(val)
Out[169]: {'loss': 0.4623577296733856}
```

Построим кривые выживаемости для первых 10 пациентов. Шкала времени корректная, потому что мы установили model.duration_index в качестве точек сетки. Однако мы определили оценки выживаемости только для 20 раз в нашей сетке дискретизации, поэтому оценки выживаемости представляют собой ступенчатую функцию.

Можно интерполировать оценки выживаемости. Линейная интерполяция (интерполяция с постоянной плотностью) может быть выполнена методом интерполяции. Нам также нужно выбрать, сколько точек мы хотим заменить каждой точкой сетки. Выберем 20.

Рисунок 22. Линейная интерполяция кривых выживаемости



.

Класс EvalSurv содержит некоторые полезные критерии оценки для прогнозирования времени до события. Мы устанавливаем censor_surv = 'km', чтобы указать, что мы хотим использовать Каплана-Мейера для оценки распределения цензурирования.

Используем индекс конкордантности Антолини:

```
In [175]: ev.concordance_td('antolini')
Out[175]: 0.9953782383419689
```

2.3. Разработка приложения

Разработано веб-приложение с помощью языка Python, фреймворка Flask и шаблонизатора Jinja.

2.4. Создание удаленного репозитория

Для размещения материалов и результатов данного исследования был создан удаленный репозиторий на GitHub, который находится по адресу https://github.com/Stebunova/Metabric. Загружены Jupiter notebook-и, код приложения, данные в формате csv.

Заключение

В ходе выполнения данный работы пройдена большая часть операций и задач, которую выполняет специалист по работе с данными на примере изучения анализа выживаемости.

Стоит отметить, что подход к разведочному анализу данных, data mining в виде извлечения и анализа признаков, и в особенности выбор модели оказался несколько иным по сравнению с пройденным в рамках курса.

Одной из находок для меня стал тот факт, что на Каггле можно скачать данные с некорректно расставленными статусами пациентов, что не все специалисты по работе с данными понимают специфику цензурированных данных и работают с ними стандартными методами. Тогда как цензурированные данные предполагают выбор моделей, которые с такими данными работают. И для этого есть специальные библиотеки - такие как lifelines и sci-kit survival, а также фреймворк pycox в PyTorch для построения нейронной сети.

Безусловно, есть и другие возможности для анализа выживаемости в том числе и в Keras, в PyTorch [9], но в рамках данной работы я ставила своей целью познакомиться с методами в целом и попробовать несколько из них, доступных

мне с учетом моего опыта, знаний и вычислительной мощности моего персонального компьютера.

Определенная сложность была связана и с большим количеством признаков в датасете. 489 генетических признаков стали сложной задачей с точки зрения вычислительной реализации многими используемыми методами. Предполагаю, что мне стоило углубиться в методы отбора генетических признаков.

Недостаточный опыт и знания не позволили критично оценить результаты, полученные при построении нейронной сети, найти причины неправдоподобно высокого результата при прогнозировании, учитывая средние результаты регрессии Кокса и случайного леса выживаемости, и сделать надлежащие выводы.

Отдельно стоит отметить, что приложение, рассчитывающее прогнозную величину риска наступления смерти, будет совершенно неинформативным для того, кто его будет использовать [14]. Целесообразно в дальнейшем будет предусмотреть пересчет в показатели, например, проценты с учетом конкретного временного интервала (в виде вероятности умереть в % в течение 5 лет, как пример).

Библиографический список

1. Лётчиков А.В., Матвеев Р.Ю., Широбокова М.А. Решение проблемы цензурированных данных при моделировании оценки индивидуального кредитного риска // Вестник Удмуртского университета. Серия «Экономика и право». 2019. №1. URL: https://cyberleninka.ru/article/n/reshenie-problemy-tsenzurirovannyh-dannyh-pri-modelirovanii-otsenki-individualnogo-kreditnogo-riska (дата обращения: 28.04.2023).

2. Румянцев П.О., Саенко В.А., Румянцева У.В., Чекин С.Ю. Статистические методы анализа в клинической практике. Часть. 2. Анализ выживаемости и многомерная статистика. Проблемы Эндокринологии. 2009;55(6):48-56. https://doi.org/10.14341/probl200955648-56

3. Austin PC, Harrell FE, Steyerberg EW. Predictive performance of machine and statistical learning methods: Impact of data-generating processes on external validity in the “large N, small p” setting. Statistical Methods in Medical Research. 2021;30(6):1465-1483. doi:10.1177/09622802211002867

4. Davidson-Pilon, (2019). lifelines: survival analysis in Python. Journal of Open Source Software, 4(40), 1317, https://doi.org/10.21105/joss.01317

5. Faraggi, D., and Simon, R. 1995. A neural network model for survival data. Statistics in Medicine 14:73–82.

6. Hemant Ishwaran. Udaya B. Kogalur. Eugene H. Blackstone. Michael S. Lauer. "Random survival forests." Ann. Appl. Stat. 2 (3) 841 - 860, September 2008. https://doi.org/10.1214/08-AOAS169

7. Katzman JL, Shaham U, Cloninger A, Bates J, Jiang T, Kluger Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC Med Res Methodol. 2018 Feb 26;18(1):24.

8. L. Waldron, M. Pintilie, M.-S. Tsao, F. A. Shepherd, C. Huttenhower, and I. Jurisica, “Optimized application of penalized regression methods to diverse genomic data,” Bioinformatics, vol. 27, no. 24, pp. 3399–3406, 2011.

9. Lee, C., Zame, W., Yoon, J., & van der Schaar, M. (2018). DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1). https://doi.org/10.1609/aaai.v32i1.11842

10. Machine Learning in Oncology: Methods, Applications, and Challenges. Dimitris Bertsimas and Holly Wiberg. JCO Clinical Cancer Informatics 2020 :4, 885-894.

11. Mucaki EJ, Baranova K, Pham HQ, Rezaeian I, Angelov D, Ngom A, Rueda L, Rogan PK. Predicting Outcomes of Hormone and Chemotherapy in the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) Study by Biochemically-inspired Machine Learning. F1000 Res. 2016 Aug 31;5:2124. doi: 10.12688/f1000research.9417.3. PMID: 28620450; PMCID: PMC5461908.

12. Mukherjee, A., Russell, R., Chin, SF. et al. Associations between genomic stratification of breast cancer and centrally reviewed tumour pathology in the METABRIC cohort. npj Breast Cancer 4, 5 (2018). https://doi.org/10.1038/s41523-018-0056-8

13. S. Pölsterl, “scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn,” Journal of Machine Learning Research, vol. 21, no. 212, pp. 1–6, 2020. ‘scikit-survival’ package - https://scikit-survival.readthedocs.io/en/latest/

14. Van Calster B, McLernon DJ, van Smeden M, Wynants L, Steyerberg EW; Topic Group ‘Evaluating diagnostic tests and prediction models’ of the STRATOS initiative. Calibration: the Achilles heel of predictive analytics. BMC Med. 2019 Dec 16;17(1):230. doi: 10.1186/s12916-019-1466-7. PMID: 31842878; PMCID: PMC6912996.

15. Yeuntyng Lai, Morihiro Hayashida, Tatsuya Akutsu, "Survival Analysis by Penalized Regression and Matrix Factorization", The Scientific World Journal, vol. 2013, Article ID 632030, 11 pages, 2013. https://doi.org/10.1155/2013/632030

16. https://humboldt-wi.github.io/blog/research/information_systems_1920/group2_survivalanalysis