Санкт-Петербургский государственный университет
Математическое обеспечение и администрирование информационных систем
Кафедра информационно-аналитических систем

Дэлгэр Батзориг

Прогнозирование риска сердечной недостаточности с использованием методов машинного обучения

Учебная практика 2

Научный руководитель:
кандидат. ф.-м. н., доцент Н. Г. Графеева

Санкт-Петербург
2022

Оглавление

Введение

В настоящее время сердечно-сосудистые заболевания (ССЗ) являются одной из самых распространенных причин смерти во всем мире. Обнаружить заболевание сердца не всегда легко, так как для его раннего прогнозирования требуются квалифицированные знания или опыт в области симптомов сердечных заболеваний. На любом этапе жизни люди могут столкнуться с любыми симптомами ССЗ. Часто для выявления болезней в медицинской науке собирается огромное количество информации. Вся эта информация не является полезной, но жизненно важной для принятия правильного решения [3].

Большинство ССЗ можно предотвратить путем устранения поведенческих факторов риска, таких как употребление табака, нездоровое питание и ожирение, недостаточная физическая активность и вредное употребление алкоголя, используя общепопуляционные стратегии[6].

Люди с ССЗ или с высоким сердечно-сосудистым риском (из-за наличия одного или нескольких факторов риска, таких как гипертония, диабет, гиперлипидемия или уже установленное заболевание) нуждаются в раннем выявлении и лечении, и здесь большую помощь может оказать модель машинного обучения.

В данной работе рассматриваются разные методы машинного обучения для прогнозирования риска сердечной недостаточности. Набор данных, который использован для исследования содержит 12 признаков (возраст, пол, гемоглобин, сахарный диабет, наличие гипертензии у пациента, тромбоциты в крови, уровень креатинина в крови и т.д) и был взят из kaggle. В качестве методов рассмотрены следующие самые распространенные методы машинного обучения: линейный и квадратичный дискриминантный анализ, логистическая регрессия, случайный лес, метод опорных векторов, метод k-ближайших соседей.

Постановка задачи

Целью данной работы является прогнозирование риска сердечной недостаточности и сравнение эффективности разных методов машинного обучения. Для решения ставятся следующие задачи:

1. Рассматривать теоретические основы методов классификации машинного обучения.
2. Сделать первичную статистическую обработку данных для анализа.
3. Провести классификацию на основе теории по разным методам и получить аккуратность.
4. Сравнить результаты методов и определить лучший метод классификации для прогноза.

Глава 1

Основные понятия и теории методов машинного обучения

1.1. Дискриминантный анализ для классификации

1.1.1. Общий подход к классификации

Дискриминантный анализ – это метод классификации с обучением. Что означает, известно групповая принадлежность часть данных (тренировочной выборки). На основе анализа тренировочной выборки будет известно как должны распределяться по группам остальные, неклассифицированные данные.

Общий подход к классификации состоит в том, что строятся классифицирующие функции f_i, такие что в качестве этих функций берут вероятность принадлежности к i-му классу. Пусть ξ – дискретная с.в., которая принимает значения $\{A_i\}$, $i = 1 \dots k$, $\mathcal{P}(\eta|\xi = A_i) = \mathcal{P}_i$ и имеет плотность $p_i(x)$. Тогда $f_i = p_i$. Если известно из какого класса индивид, то можем учесть это априорное знание вероятности. Пусть $C_i = \{\xi = A_i\}$ – класс. Тогда $\pi_i = P(C_i)$ априорная вероятность того, что наблюдение принадлежит к i-му классу. Тогда апостериорная вероятность вычисляется по формуле Байеса:

$$P(C_i|x) = \frac{P(x|C_i)\pi_i}{\sum_{j=1}^{k} P(x|C_j)\pi_j}.$$

В качестве классифицирующих функций можно взять

$$f_i(x) = \frac{p_i(x)\pi_i}{\sum_{j=1}^{k} p_j(x)\pi_j}.$$

У всех f_i знаменатель одинаковый, поэтому отбросим его и получаем итоговые классифицирующие функции: $f_i(x) = P(x|C_i)\pi_i = p_i(x)\pi_i$.

1.1.2. Линейный дискриминантный анализ

Линейный дискриминантный анализ (LDA) применяется для нахождения линейных комбинаций признаков, которые наилучшим образом разделяют класс объектов.

Иными словами, на основании линейных комбинаций значений признаков создаются функции, которые определяют к какому классу принадлежит данная особь.

Преимущество метода в том, что функции могут быть построены, зная только статистические характеристики каждого класса. Нужно рассчитать значение каждой разделяющей функции для классификации объекта. Причем функция, которая дает максимальное значение, определяет класс объекта. LDA дает лучший результат, если параметры класса из нормального распределения и имеют одинаковые ковариационные матрицы [4].

Пусть ξ – дискретная с.в., которая принимает значения $\{A_i\}$, $i = 1 \dots k$, $\mathcal{P}(\eta|\xi = A_i) = \mathcal{N}(\mu_i, \Sigma)$. Тогда плотность в точке x

$$p_i(x) = p(x|\xi = A_i) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(x-\mu_i)^{\mathrm{T}}\Sigma^{-1}(x-\mu_i)\right).$$

и классифицирующая функция $f_i(x) = \pi_i p(x|\xi = A_i)$, где π_i — априорная вероятность. Можно переписать классифицирующую функцию через возрастающее монотонное преобразование как

$$g_i(x) = \log f_i(x) = \log \pi_i - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x-\mu_i)^{\mathrm{T}}\Sigma^{-1}(x-\mu_i).$$

Можем сократить часть, которую не зависит от номера класса

$$h_i(x) = -\frac{1}{2}\mu_i^{\mathrm{T}}\Sigma^{-1}\mu_i + \mu_i^{\mathrm{T}}\Sigma^{-1}x + \log \pi_i$$

.

1.1.3. Квадратичный дискриминантный анализ

Квадратичный дискриминантный анализ (QDA) представляет собой нелинейное обобщение метода LDA, в котором не используется предположение об однородной ковариационной матрице. Другими словами, считаем, что ковариационные матрицы разные [4].

QDA эффективен, когда разделяющая поверхность между классами имеет сильный нелинейный характер.

Пусть ξ – дискретная с.в., которая принимает значения $\{A_i\}$, $i = 1 \dots k$, $\mathcal{P}(\eta|\xi = A_i) = \mathcal{N}(\mu_i, \Sigma_i)$. Тогда плотность в точке x

$$p_i(x) = p(x|\xi = A_i) = \frac{1}{(2\pi)^{p/2}|\Sigma_i|^{1/2}}\, exp\left(-\frac{1}{2}(x-\mu_i)^{\mathrm{T}}{\Sigma_i}^{-1}(x-\mu_i)\right).$$

и классифицирующая функция $f_i(x) = \pi_i p(x|\xi = A_i)$. Применяем возрастающее монотонное преобразование и оставляем в классифицирующей функции только члены, отличающиеся в разных группах

$$g_i(x) = \log f_i(x) = \log\pi_i - \frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(x-\mu_i)^{\mathrm{T}}{\Sigma_i}^{-1}(x-\mu_i),$$

получаем квадратично зависящую от x классифицирующую функцию [**?**].

1.2. Случайный лес (Random Forest)

Случайный лес — один из самых известных алгоритмов машинного обучения, придуманные Лео Брейманом и Адель Катлер. Известность заключается в том, что он подходит для многих задач классификации, регрессии и кластеризации. Также он реализуются в различных пакетах программного обеспечения, в то же время легко переобучается [2].

Алгоритм метода основан на построении ансамбля деревьев решений. Создается множество деревьев принятия решений, а потом результат их предсказания усредняется. В случайных лесах корреляция между деревьями понижается случайным образом по индивидам и по признакам. По индивидам: по бутстрапированной подвыборке обучается каждое дерево. По признакам: в каждой вершине разбиение ищется по подмножеству признаков. Разделение вершин происходит последовательно до тех пор, пока не будет достигнуто идеальное качество на обучении. Каждая вершина разделяет выборку по одному из признаков относительно некоторого порога. В случайных лесах разбиение производится по признаку выбран не из всех возможных признаков, а лишь из их случайного подмножества размера m.

Для классификации случайный лес получает классовый голос из каждого дерева, а затем классифицирует с использованием большинства голосов. При использовании для регрессии, предсказания от каждого дерева в целевой точке x просто усредняются. Кроме того, рекомендуется брать:

- Для классификации значение по умолчанию для $m - \sqrt{p}$, а минимальный размер узла - единица.

- Для регрессии значение по умолчанию для $m - p/3$, а минимальный размер узла - пять.

где p – число признаков.

Алгоритм случайного леса

Пусть $X = (x_i, y_i), i = 1 \dots N$ – конечная выборка и $X_1, \dots, X_N$ – подвыборки. Обучим по каждой из них и получим базовые алгоритмы $b_1(x), \dots, b_N(x)$. $\tilde{X}$ – генерированная случайная подвыборка с помощью бутстрапа. Тогда:

1. Для $n = 1, \dots, N$, сгенерировать выборку $\tilde{X}_n$ с помощью бутстрэпа
2. Построить решаюотражаюее дерево $b_n(x)$ по выборке $\tilde{X}_n$:
   - дерево строится, пока в каждом листе не окажется не более n_{min} объектов
   - при каждом разбиении сначала выбирается m случайных признаков из p, и оптимальное разделение ищется только среди них
3. Вернуть композицию $a_N(x) = \frac{1}{N} \sum_{n=1}^{N} b_n(x)$ [].

1.3. Логистическая регрессия

Логистической регрессией называется метод обучения, который оценивает вероятность принадлежности объекта к каждому из классов [5].

Пусть в каждой точке пространства индивидов $x \in X$ задана вероятность $p(y == +1|x)$ того, что объект x будет принадлежать классу $+1$. Значения $p(y == +1|x)$ будут находиться в диапазоне от 0 до 1. Если, например, $p(y == +1|x) > 0.5$, то есть больше вероятность, что данный индивид принадлежит к классу $+1$.

Рассмотрим простую модель множественной линейной регрессии:

$$p(X) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p, \tag{1.1}$$

где $X = (X_1, \dots, X_p)$ – предикторы. Значения данной модели могут быть не в диапазоне от 0 до 1, чтобы избежать этой проблемы, нужно моделировать $p(X)$ с помощью функции, которая на выходе дает значения между 0 и 1 для всех X. В логистической регрессии используется логистическая функция:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}. \tag{1.2}$$

Для оценки коэффициентов $\beta_0, \ldots, \beta_p$ модели (1.2) используется метод максимального правдоподобия. Делая некоторые переобразования получим:

$$\frac{p(X)}{1-p(X)} = e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}. \tag{1.3}$$

Прологарифмируем и получаем выражение:

$$log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p. \tag{1.4}$$

Левую сторону (1.4) называют логистическим или логит-преобразованием. Теоретически (1.2) может принимать любое значение. Поскольку логит-преобразование решает проблему об ограничении на 0-1 границы. Видим, что логистическая регрессия – это преобразование обычной линейной регрессии.

1.4. Метод опорных векторов

Рассмотрим линейные классификаторы вида

$$a(x) = sign(\langle w, x\rangle + b), \qquad w \in \mathbf{R}^d,\ b \in \mathbf{R}.$$

Будем считать, что существуют такие параметры $w*$ и $b*$, что соответствующий им классификатор $a(x)$ не допускает ни одной ошибки на обучающей выборке. В этом случае говорят, что выборка линейно разделима.

Пусть задан некоторый классификатор $a(x) = sign(\langle w, x\rangle + b)$. Заметим, что если одновременно умножить параметры w и b на одну и ту же положительную константу, то классификатор не изменится. Распорядимся этой свободой выбора и отнормируем параметры так, что

$$\min_{x\in X} |\langle w, x\rangle + b| = 1 \tag{1.5}$$

Можно показать, что расстояние от произвольной точки $x_0 \in \mathbf{R}^d$ до гиперплоскости, определяемой данным классификатором, равно

$$\rho(x_0, a) = \frac{|\langle w, x\rangle + b|}{||w||}$$

Тогда расстояние от гиперплоскости до ближайшего объекта обучающей выборки равно

$$\min_{x \in X^l} \frac{|\langle w, x \rangle + b|}{||w||} = \frac{1}{||w||} \min_{x \in X} |\langle w, x \rangle + b| = \frac{1}{||w||}$$

Данная величина также называется отступом (margin).

Таким образом, если классификатор без ошибок разделяет обучающую выборку, то ширина его разделяющей полосы равна $\frac{2}{||w||}$. Известно, что максимизация ширины разделяющей полосы приводит к повышению обобщающей способности классификатора [1]. Вспомним также, что на повышение обобщающей способности направлена и регуляризация, которая штрафует большую норму весов — а чем больше норма весов, тем меньше ширина разделяющей полосы. Итак, требуется построить классификатор, идеально разделяющий обучающую выборку, и при этом имеющий максимальный отступ. Запишем соответствующую оптимизационную задачу, которая и будет определять метод опорных векторов для линейно разделимой выборки (hard margin support vector machine):

$$\begin{cases} \frac{1}{2}||w||^2 \to \min\limits_{w,b} \\ y_i(\langle w, x \rangle + b) \geq 1, \quad i = 1, \ldots, l. \end{cases} \tag{1.6}$$

Здесь мы воспользовались тем, что линейный классификатор дает правильный ответ на объекте x_i тогда и только тогда, когда $y_i(\langle w, x \rangle + b) > 0$. Более того, из условия нормировки (1.5) следует, что $y_i(\langle w, x \rangle + b) > 1$. В данной задаче функционал является строго выпуклым, а ограничения линейными, поэтому сама задача является выпуклой и имеет единственное решение [2].

1.5. Метод k-ближайших соседей

Метод ближайших соседей — простейший метрический классификатор, основанный на оценивании сходства объектов. Классифицируемый объект относится к тому классу, которому принадлежат ближайшие к нему объекты обучающей выборки.

Пусть $X = (X_1, \ldots, X_p)$ – вектор признаков, а Y – ответы. Тогда, k-ближайших соседей для $\hat{Y}$ определяется следующим образом:

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x) y_i,}$$

где $N_k(x)$ – это окрестность x, определяемая k ближайшими точками x_i в обучающей выборке. В качестве меры расстояния между объектами обычно используется расстояние Евклида. Проще говоря, мы находим k наблюдений с x_i, ближайшими к x в пространстве входных данных, и усредняем их ответы [4].

1.6. Ошибки классификации

Classification matrix – показатель качества классификации (доля неправильно классифицированных объектов). Иными словами, данный метод классификации отнес некоторые числа объектов не в тот класс, к которому они были изначально причислены. Строки таблицы – классы, к которым объекты изначально принадлежали. Столбцы – классы, к которым они принадлежат по результатам классификации. Также в матрице классификации количество соответствующих объектов указано. Например, в первом столбце стоит доля правильно классифицированных объектов (для всех объектов и отдельно по классам). На основе матрицы классификации выясняется классы, которые плохо различаются между собой.

Глава 2

Результаты применения методов классификации

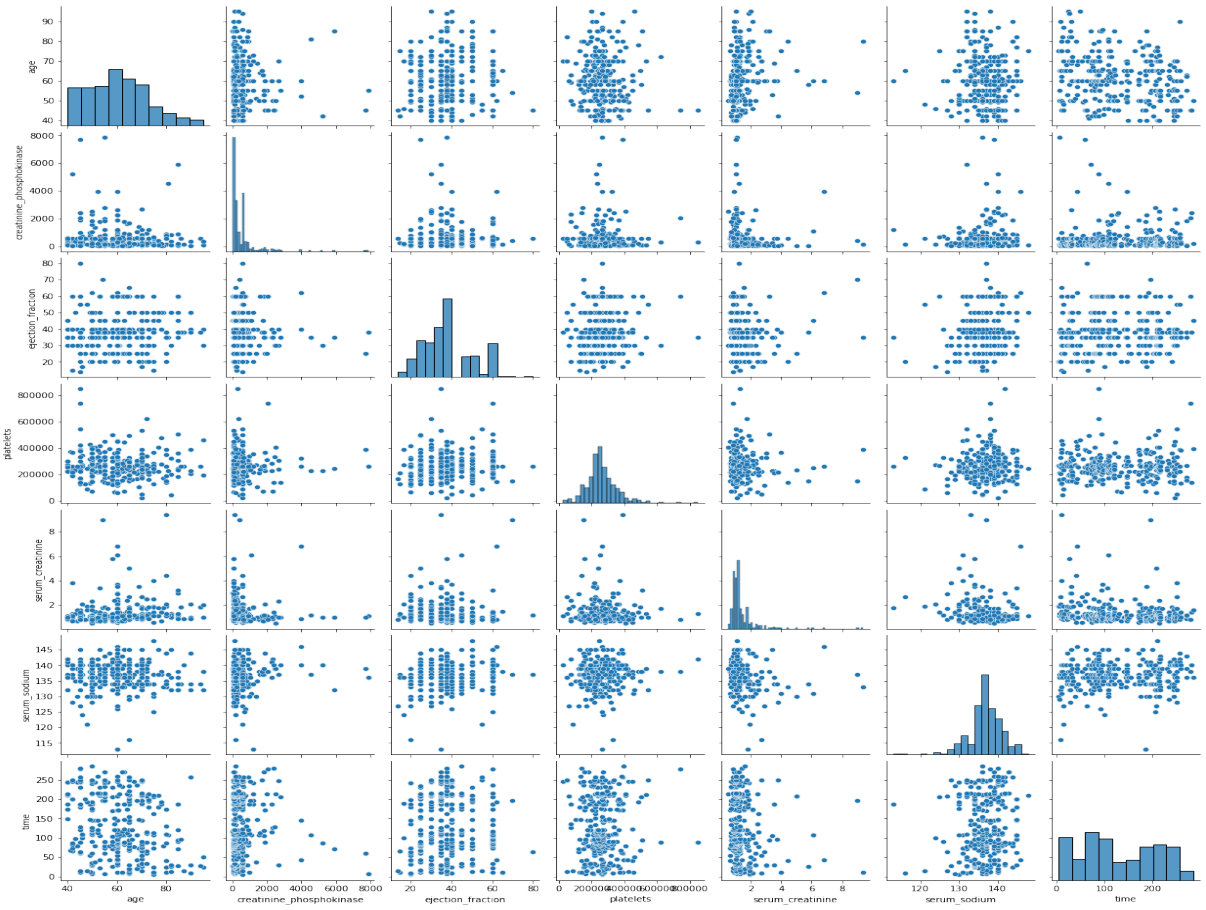
2.1. Описание набора данных для анализа

Данный набор данных был взят из сайта организации конкурсов по исследованию данных Kaggle [1]. Для классификации использован набор данных по сердечной недостаточности. Этот набор данных состоит из 299 индивидов и 13 признаков (возраст, пол, гемоглобин, сахарный диабет, наличие гипертензии у пациента, тромбоциты в крови, уровень креатинина в крови и т.д), из них 6 количественных и 7 категориальных признаков. Классифицирующий фактор – сердечная недостаточность: 0 – нет, 1 – да. С помощью программной среды Python была написана программа для реализации данных методов.

2.2. Первичный анализ данных сердечной недостаточности

Для классификации необходимо провести первичный анализ данных, в том числе: посмотреть на данные и переобразовать если необходимо, прологарифмировать несимметричных распределений, анализировать пропущенных значений и так далее.

Сначала имеет смысл посмотреть на данные:



Прологарифмируем признаки age, creatinine phosphokinase, serum creatinine так как у них несимметричные распределения (хвосты справа и слева в гистограмме). Посмотрим на графики после логарифмирования:

Видим, что распределения признаков стали более симметрично. Также, на графике заметно, что признаки плохо коррелированы между собой.

2.3. Применение методов для прогнозирования

Данные были разбыты на обучаемые и тестовые выборки (соотношение 80/20) для контроля переобучения. Использован метод определения самых значимых признаков (feature importance) для каждого метода классификации. Были выбраны самые значимые признаки для каждой модели, также проведен анализ выбросов для них. Найдены некоторые выбросы и были удалены для улучшения качества классификации. Благодаря выбору самых значимых признаков и удалению выбросов точность классификации были улучшены в среднем с 70% до 85% для каждого метода классификации.

2.3.1. Метод LDA

Рассмотрим на матрицу классификации (обучаемые и тестовые выборки):

tr	0	1	te	0	1
0	119	15	0	58	9
1	19	50	1	7	14

Доля правильных классификации на обучаемые и тестовые: 0.8325, 0.8181

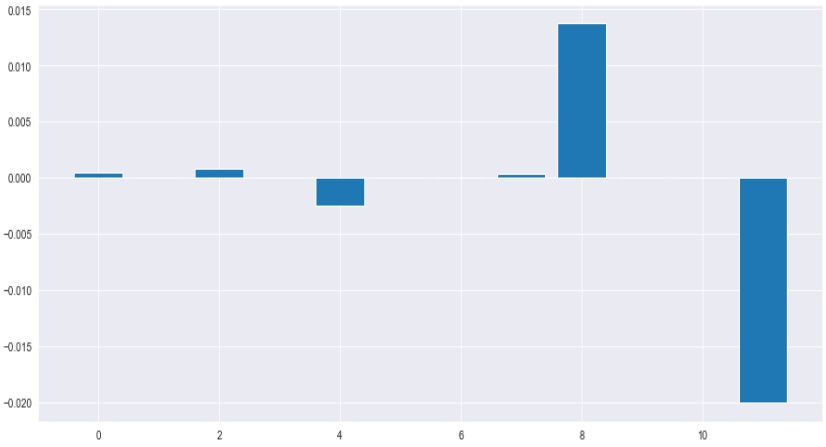
2.3.2. Метод QDA

tr	0	1	te	0	1
0	120	14	0	59	8
1	21	48	1	6	15

Доля правильных классификации на обучаемые и тестовые: 0.82, 0.84

2.3.3. Метод логестической регрессии

Feature immportance:



Рассмотрим на матрицу классификации (обучаемые и тестовые выборки):

tr	0	1	te	0	1
0	146	15	0	35	5
1	23	48	1	5	14

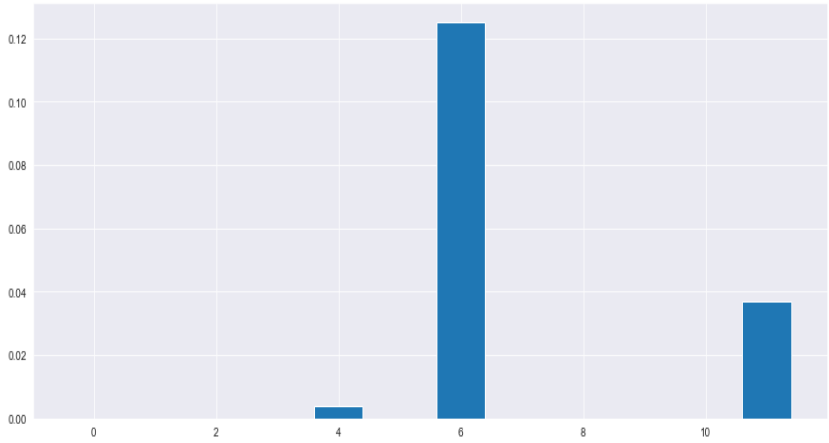
Доля правильных классификации на обучаемые и тестовые: 0.8362, 0.835

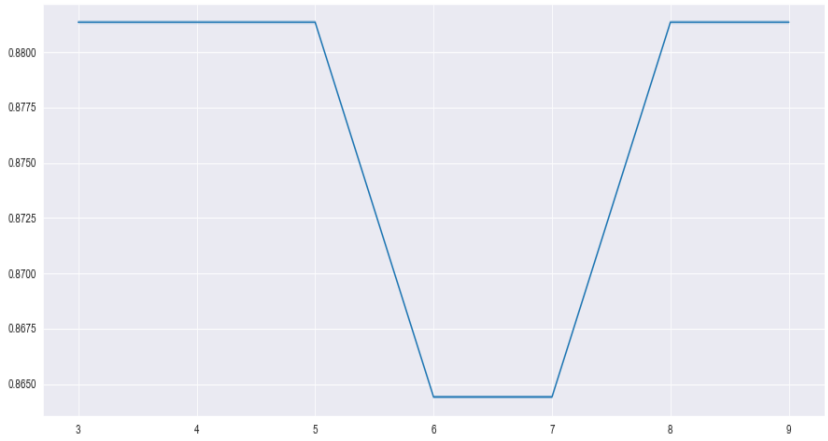
2.3.4. Метод k-ближайших соседей

Feature importance:

Number of optimal neighbors:

Рассмотрим на матрицу классификации (обучаемые и тестовые выборки):





tr	0	1	te	0	1
0	129	5	0	59	8
1	12	57	1	5	16

Доля правильных классификации на обучаемые и тестовые: 0.9163, 0.8522

2.3.5. Метод SVC

Рассмотрим на матрицу классификации (обучаемые и тестовые выборки):

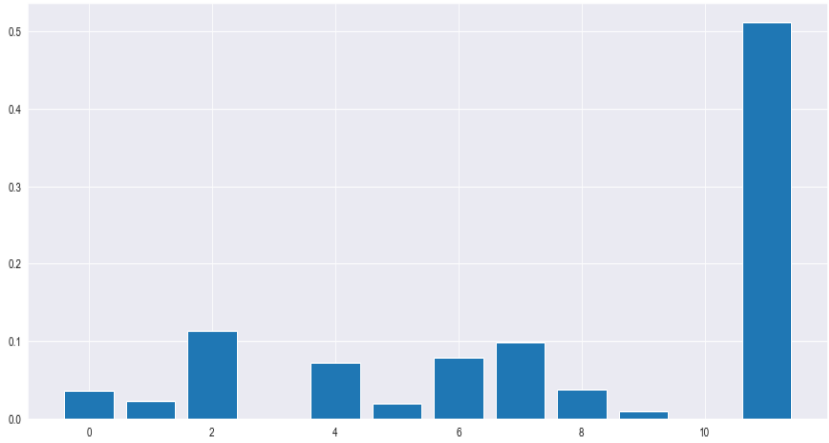
tr	0	1	te	0	1
0	130	4	0	62	5
1	24	45	1	8	13

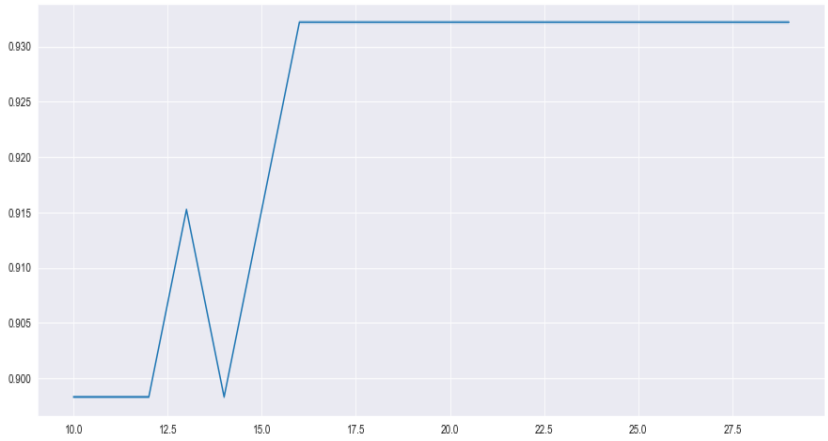
Доля правильных классификации на обучаемые и тестовые: 0.862, 0.8522

2.3.6. Метод random forest

Feature importance:

Number of optimal estimators:





Рассмотрим на матрицу классификации (обучаемые и тестовые выборки):

tr	0	1	te	0	1
0	156	0	0	44	1
1	1	75	1	3	11

Доля правильных классификации на обучаемые и тестовые: 0.9956, 0.9322

2.4. Сравнение результатов классификации

Таблица 2.1. Точность классификации

	LDA	QDA	LogReg	Knn	SVC	RF
train	83.2	82	83.6	91.6	86.2	99.5
test	81.8	84	83.5	85.2	85.2	93.2

Из таблицы видно, что методы дискриминантного анализа (LDA, QDA) дают точности хуже остальных методов. А лучше всех классифицируется метод случайный лес.

Также, разница аккуратности классификаций на обучаемые и тестовые выборки для всех методов получились небольшие.

Заключение

Цель данной работы заключалась в прогнозировании риска сердечной недостаточности и сравнение эффективности разных методов машинного обучения. Для решения ставились следующие задачи:

1. Рассматривать теоретические основы методов классификации машинного обучения.
2. Сделать первичную статистическую обработку данных для анализа.
3. Провести классификацию на основе теории по разным методам и получить аккуратность.
4. Сравнить результаты методов и определить лучший метод классификации для прогноза.

В ходе работе были выполнены все выше упомянутые задачи в том, числе рассмотрены необходимые теоретические основы разных методов классификаций, такие как линейный и квадратичный дискриминантный анализ, случайный лес, SVC, k-ближайших соседей, также их применения для прогнозирования риска сердечной недостаточности. Для классификации использован набор данных по сердечным заболеваниям из сайта Kaggle. Были получены результаты классификаций по этим методам и ошибки классификации. Использованы методы для улучшения качества классификации, такие как анализ выбросов и выбор самых значимых признаков. Были сравнены результаты методов классификации и лучшую точность нам дал метод случайный лес с аккуратностью на тестовые выборки 93.2%.

Список литературы

1. Heart failure prediction. urlhttps://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-d Accessed: 2022-03-20.
2. Соколов Е.А. *Введение в машинное обучение.* ФКН ВШЭ, 2016.
3. SMM Hasan, MA Mamun, MP Uddin, and MA Hossain. Comparative analysis of classification approaches for heart disease prediction. In *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*, pages 1–4. IEEE, 2018.
4. Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.
5. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.
6. C Beulah Christalin Latha and S Carolin Jeeva. Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. *Informatics in Medicine Unlocked*, 16:100203, 2019.
7. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning.* MIT press, 2018.