

Asfendiyarov Kazakh National Medical University

**Тема исследования: Практическое применение
Автоматизированной системы научных исследований в медицине,
здравоохранении и смежных областях**

**Проект: Анализ факторов риска сердечно сосудистых заболеваний
и прогноз исходов лечения при помощи методов Машинного
Обучения**

Автор исследования: Dr. Alexander Wagner (Berlin)





Содержание

Предисловие	6
История создания Автоматизированной Системы Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»	6
Краткая характеристика «АСНИ-МЕД»	6
Цель создание АСНИ	6
Назначение системы	7
Область применения системы.....	7
Задачи решаемые при помощи системы	7
Результаты применения системы	8
Введение.....	9
Цель исследования	10
Материалы и методы	12
Исследовательский анализ данных (EDA)	12
Введение в EDA.....	12
Набор данных для анализа	15
Преобразование категориальной переменной в числовую	17
Таблица 1. Исходные данные для анализа.....	19
Таблица 2. Дескриптивная статистика числовых переменных для всех пациентов	21
Таблица 3. Распределение категориальных переменных по частоте для всех пациентов.....	23
Таблица 4. Основная статистика для переменной HeartDisease='Healthy'.....	23
Таблица 5. Основная статистика для переменной HeartDisease='Heart Disease'	24
График №1. BoxPlot для всех числовых переменных по классу Заболевание (HeartDisease)	24
График №2. BoxPlot для всех числовых переменных	25
Таблица №6. Распределение пациентов по заданным категориям	25
График №3. Распределение числовых переменных по классу пол (Sex)	27
Таблица №7. Распределение пациентов по заданным категориям	28
График №4. Гистограммы распределения для всех числовых переменных по классу Заболевание (HeartDisease).....	30
Таблица №8. Распределение пациентов по заданным категориям	30
График №5. Гистограммы распределения для всех числовых переменных по классу пол (Sex)	33
Таблица №9. Распределение пациентов по заданным категориям	33
График №6. Двумерное распределение переменной 'HeartDisease' по классам Sex и RestingBP	35
Таблица №10. Распределение пациентов по заданным категориям	35
График №7. График распределение переменной Cholesterol по Возрасту (Age).....	37
Таблица №11. Распределение пациентов по заданным категориям	37



График №8. Гистограммы распределения для всех числовых переменных, представленные на одном графике	39
Таблица №12. Распределение пациентов по заданным категориям	39
График №9. Матрица корреляции Пирсона для числовых переменных	45
Таблица №13. Распределение пациентов по заданным категориям	45
График №10. Гистограммы распределения для всех числовых переменных, в виде субграфиков на одной панели.....	49
Таблица №14. Распределение пациентов по заданным категориям	49
График №11. Распределение переменной Cholesterol по классу (Age, Sex, FastingBS) в виде 4 субграфиков на одной панели.....	55
Таблица №15. Распределение пациентов по заданным категориям	55
График №12. Распределение по возрасту (Age) для переменных: 'Sex','ChestPainType','FastingBS','RestingECG','ExerciseAngina','ST_Slope','HeartDisease' в форме Виалин-графиков.....	61
Таблица №16. Распределение пациентов по заданным категориям	61
График №13. Распределение всех числовых переменных в виде субграфиков по классу 'HeartDisease' на одной панели.....	62
Таблица №17. Распределение пациентов по заданным категориям	62
График №14. Распределение всех числовых переменных в виде столбиковых диаграмм как субграфиков по классу 'HeartDisease' на одной панели.....	63
Таблица №18. Распределение пациентов по заданным категориям	63
График №15. Распределение численности пациентов по всем переменным в виде столбиковых диаграмм как субграфиков по классу пол (Sex) на одной панели	65
Таблица №19. Распределение пациентов по заданным категориям	65
График №16. Биполярное распределение переменной Cholesterol по возрасту 'Age'	67
График №17. Биполярное распределение разного графического типа всех численных переменных на одной панели.....	69
График №18. Плотность распределения переменной Cholesterol по классу пол (Sex)	71
График №19. Распределение переменной Age возрастным группам в виде столбиковых диаграм по классу 'HeartDisease'	73
График №20. Распределение переменной Cholesterol по возрастным группам в виде столбиковых диаграм по классу 'HeartDisease'.....	75
График №21. Распределение переменной RestingBP по возрастным группам в виде столбиковых диаграм по классу 'HeartDisease'.....	77
График №22. Распределение переменной MaxHR по возрастным группам в виде столбиковых диаграм по классу 'HeartDisease'	79
График №23. Распределение переменной Oldpeak по возрастным группам в виде столбиковых диаграм по классу 'HeartDisease'	81
График №24. Торт-диаграмма распределения пациентов по класс-переменной HeartDisease	83
График №25. Торт-диаграмма распределения пациентов по класс-переменной Sex	85
График №26. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType','FastingBS','RestingECG','ExerciseAngina', 'ST_Slope','HeartDisease' по категориям	87



График №27. Блок-бокс диаграмма распределения переменных: 'Sex', 'ChestPainType','FastingBS','RestingECG','ExerciseAngina', 'ST_Slope','HeartDisease' в виде субграфиков на одной панели по категориям	89
График №28. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType','FastingBS','RestingECG','ExerciseAngina', 'ST_Slope','HeartDisease' по категориям	91
График №29. Секторная диаграмма распределения переменной ChestPainType	93
График №30. Секторная диаграмма (2-го типа) распределения переменной ST_Slope	95
График №31. Комбинированная диаграмма распределения переменной пол (Sex)	97
График №32. Комбинированная диаграмма распределения переменной HeartDisease	99
График №33. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType','FastingBS','RestingECG','ExerciseAngina', 'ST_Slope' по категориям и классу 'HeartDisease'	101
График №34. Столбиковая диаграмма распределения переменных: 'ChestPainType','FastingBS','RestingECG','ExerciseAngina', 'ST_Slope' по категориям и классу 'Sex'	103
График №35. Столбиковая диаграмма распределения переменных: 'Sex','FastingBS','RestingECG','ExerciseAngina','ST_Slope', 'HeartDisease' по категориям и классу 'ChestPainType'	105
График №36. Комбинированная (столбиковая и секторная) диаграмма распределения переменной пол (Sex).....	107
График №37. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ChestPainType	109
График №38. Комбинированная (столбиковая и секторная) диаграмма распределения переменной RestingECG	111
График №39. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ExerciseAngina	113
График №40. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ST_Slope.....	115
График №41. Комбинированная (столбиковая и секторная) диаграмма распределения переменной Cholesterol_Category	117
График №42. Комбинированная (столбиковая и секторная) диаграмма распределения переменной RestingBP_Category.....	119
График №43. Двойная секторная диаграмма (Sunburst, 6 субграфиков) распределения пар переменных: ['ChestPainType', 'FastingBS'], ['ST_Slope', 'RestingECG'], ['ExerciseAngina', 'ChestPainType'] по классу пол (Sex)	121
График №44. Столбиковая диаграмма распределения переменных: RestingECG, ChestPainType (2 субграфика на одной панели) по классу пол (Sex).....	123
График №45. Столбиковая диаграмма распределения переменных: ST_Slope, ExerciseAngina (2 субграфика на одной панели) по классу пол (Sex).....	125
Заключение (EDA)	127
Исходные данные и их организация.....	129
Предварительный анализ данных.....	131
Моделирование.....	132
Модель: Linear Regression	133



Модель: Logistic Regression.....	135
Модель: Perceptron	137
Модель: Linear SVC	139
Модель: MLPClassifier.....	141
Модель: Decision Tree Classifier 1.....	143
Модель: Stochastic Gradient Decent.....	145
Модель: RidgeClassifier.....	147
Модель: BaggingClassifier.....	149
Модель: AdaBoostClassifier 1	151
Модель: GradientBoostingClassifie	153
Модель: KNeighborsClassifier.....	155
Модель: DecisionTreeClassifier 2.....	157
Модель: RandomForestClassifier.....	159
Модель: XGBClassifier.....	161
Модель: AdaBoostClassifier 2	163
Модель: Naive Bayes	165
Модель: SVC.....	167
Результаты моделирования	170
Оценка моделей и рекомендации	171
Обсуждение и выводы	174
Заключение	175
Литература	176
Приложение	183



Предисловие

Данная работа посвящена проблеме проведения научного исследования и создания научного отчета в медицине и здравоохранении при помощи Автоматизированной Системы Научных Исследований. Краткая характеристика системы приведена в данном разделе. Описание научного исследования, выполнено по действующим международным стандартам (1) приводится последующих разделах данного документа.

История создания Автоматизированной Системы Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

За многие десятилетия работы в роли Биостатистика и Data Scientist(a) в различных фирмах разной величины и масштаба и в разных странах у меня неоднократно возникала идея о необходимости автоматизировать процесс анализа данных. Мне удалось реализовать несколько таких проектов и внедрить их в нескольких организациях. С примером одной из таких систем демонстрировавшейся на научной конференции (2019), можно познакомиться по ссылке:

https://saswiki.de/display/KONFERENZEN/KSFE+2019?preview=19726371/19726410/23_KSFE_2019_Wagner_-_Ein_SAS_basiertes_System_zur_automatisierten_Auswertung_und_Berichterstellung_von_klinischen_Studien.pdf

С учетом накопленного опыта и в условиях появления новых возможностей в Информационных Технологиях, в том числе Web-технологий и Искусственного Интеллекта, возникли хорошие возможности для создания АСНИ собственными силами и в реальные сроки. Считаю своим долгом предложить вашему вниманию концепт этой системы.

Краткая характеристика «АСНИ-МЕД»

Научная работа – это наиболее сложная человеческая деятельность, для её успешного проведения требуются не только личные качества, но и соответствующий инструментарий. Прежде всего это научные данные и средства их анализа. В медико-социальных исследованиях этим инструментарием являются Биостатистика со всем набором современных методов анализа и прогноза данных и ситуаций и современные компьютерные системы, позволяющие это реализовать.

К сожалению, освоение этого технического инструментария требует больших затрат времени и не всем желающим это в силу различных причин удаётся. Эта проблема существует уже долгие годы и на протяжении длительного времени как крупные компании так и небольшие организации испытывают постоянную потребность в квалифицированных Биостатистиках, к сожалению, их поиск довольно трудное дело и не всегда завершается успехом. В то же время, привлечение специализированных организаций и Freelancer(ов) стоит дорого и не гарантируют от неудач.

Цель создание АСНИ

Главная цель создание системы заключается в упрощении и облегчении процесса анализа медико-социальных данных и построение моделей прогноза ситуаций в процессах, происходящих в медицине, здравоохранении и в смежных областях (в медицинском страховании, в фармокондустрии и т.п.)

Цель реализуется через предоставлению следующих возможностей пользователю:

- Простой и беспроблемный доступа к Базе статистических знаний
- Запуск типовых программ Анализа данных без необходимости инсталляции специальных статистических систем на пользовательском компьютере



- Самостоятельное решение своих (научных) проблем по Анализу медицинских данных
- Предоставление других услуг

Назначение системы

Система предназначена для широкого круга пользователя, в том числе:

- Студентов медицинских факультетов
- Аспирантов и докторантов медицинских факультетов
- Научных работников сферы здравоохранения
- Администраторов здравоохранения
- Сотрудников специализированных организаций, занимающихся анализом данных клинических исследований (Clinical Trials, <https://en.wikipedia.org/wiki/ClinicalTrials.gov>).
- Других пользователей

Область применения системы

Данная Система предусмотрена для использования на всех уровнях управления. Это означает, что как на государственном (с расширенным функциональным набором), так и на областных/территориальных уровнях система работает идентично. Привилегированный пользователь может использовать без ограничения все данные и возможности системы.

Другие пользователи имеют доступ только к данным конкретной области. Система построена по модульному принципу и является открытой для развития и расширения.

Задачи решаемые при помощи системы

При помощи АСНИ решаются следующие задачи:

- Визуальный анализ данных при помощи интерактивных графиков и таблиц различного типа и свойства, отображающих в динамическом режиме например, состояние основных показателей здоровья населения территории
- Составление аналитических отчетов о состоянии параметров и показателей управляемого объекта, например, здоровья населения территории
- Научный анализ данных и прогноз при помощи современных средств продвинутой статистики (Advanced Statistics) и методов Искусственного Интеллекта
- Построения моделей оптимизации работы отрасли или организации
- Другие оперативные и стратегические задачи

Методы решения задач АСНИ

Для решения поставленных задач АИС используются методы:

- Агрегации данных и создания многомерных отчетов
- Визуализации данных (динамические графики, карты, пр.)
- Классической прикладной математической статистики
- Методы статистического моделирования, в том числе Монте Карло
- Методы Доказательной медицины (Evidence Based Medicine)
- Методы Экономической медицины (Health Economics)
- Методы Искусственного Интеллекта, в том числе: Машинного обучения(Machine learning), глубокого обучения(Deep learning)
- Математические Методы оптимизации (Operations Research)
- Эвристические методы



Результаты применения системы

Результатами являются:

- Отчеты научно-исследовательских работ
- Статьи и презентации на научно-практических конференциях
- Диссертационные работы
- Отчеты плановых и коммерческих проектных работ
- Другие формы выходных результатов научных и проектных работ



Введение

Цель данного проекта состоит из двух подцелей, в том числе:

- Первая главная подцель – это создание компьютерной системы АСНИ для автоматизированного анализа и прогноза ситуаций и феноменов из области медицины и здравоохранения при помощи современных информационных технологий и методов Машинного обучения, соответственно Искусственного Интеллекта.
- Вторая главная подцель – это непосредственное научное исследование определенной медицинской проблемы (клинический триал). ВА нашем случае это научное исследование влияние факторов риска на заболеваемость и смертность больных сердечно-сосудистыми заболеваниями. Дальнейшее изложение посвящено этой второй подцели.



Цель исследования

За последние несколько десятилетий смертность пациентов с острым коронарным синдромом (ОКС) возросла [1] и стала ведущей причиной смертности во всем мире [2]. По данным Всемирной организации здравоохранения, острый коронарный синдром является основной причиной смерти во всем мире. Ранняя диагностика острого коронарного синдрома и его прогнозирование очень важны для пациентов с заболеваниями сердца. С другой стороны, очень трудно точно предсказать тождественность острого коронарного синдрома по медицинскому набору данных, поскольку он зависит от множества факторов риска.

В 1960-х годах во Фрамингемском исследовании сердца [4] была выдвинута идея острого коронарного синдрома, и модель прогнозирования острого коронарного синдрома была разделена на два метода, а именно методы, основанные на регрессии, и методы, основанные на машинном обучении. Существует множество моделей прогнозирования риска, основанных на регрессиях, но наиболее распространенными моделями прогнозирования риска для раннего прогнозирования и диагностики основных неблагоприятных сердечно-сосудистых событий являются тромболизис при инфаркте миокарда (TIMI) [5] и Глобальный регистр острых коронарных событий (GRACE) [6], которые используются для прогнозирования оценки риска острого коронарного синдрома. Обе модели используют предыдущую медицинскую карту для изучения и прогнозирования тяжести состояния пациентов, но есть и некоторые недостатки этих старых моделей прогнозирования оценки риска, поскольку они были разработаны и внедрены около 10 лет назад. Эти модели используют несколько человек для прогнозирования риска и предсказывают уровень смертности на основе этих предикторов риска. Существует также больше предикторов, которые могут быть использованы для прогнозирования существования серьезных неблагоприятных сердечно-сосудистых событий (МАСЕ), таких как предыдущая медицинская карта и текущее состояние здоровья пациента.

Существует два метода диагностики и прогнозирования случаев острого коронарного синдрома: клинические методы и модель прогнозирования риска для постановки диагноза. Клиническими методами диагностики острого коронарного синдрома являются ангиография, электрокардиограмма (ЭКГ), холтеровское мониторирование, эхокардиограмма, нагрузочный тест, катетеризация сердца, компьютерная томография сердца (КТ) и магнитно-резонансная томография сердца (МРТ) [7]. Другой метод заключается в проектировании и разработке моделей прогнозирования рисков для ранней диагностики и прогнозирования ОКС с использованием алгоритмов статистического анализа и машинного обучения.

Алгоритмы машинного обучения повышают точность прогнозирования сердечно-сосудистых заболеваний и предотвращают ненужное лечение [8]. Методы машинного обучения преодолели проблемы традиционных методов, основанных на регрессии, и популярны для диагностики и прогнозирования возникновения МАСЕ. Кроме того, он устраняет типичные проблемы с данными и устраняет отсутствующие значения и выбросы с помощью методов интеллектуального анализа данных. Методы машинного обучения основаны на нелинейных связях и взаимодействиях между несколькими переменными и имеют дело с различными предикторами риска для точного прогнозирования риска пациентов. В этом исследовании также изучается эффективность методов прогнозирования риска на основе машинного обучения для прогнозирования степени тяжести пациентов с острым коронарным синдромом. Johnson et al. [9] отметили важность алгоритмов машинного обучения для прогнозирования и диагностики сердечно-сосудистых заболеваний. Тем не менее, методы, основанные на машинном обучении, имеют ряд сложных проблем для прогнозирования случаев МАСЕ в группах ИМпСТ и НСТЕМ у пациентов с острым коронарным синдромом, а именно: Во-первых, не существует специфического машинного обучения или ансамблевого подхода, что дает хорошие результаты для прогнозирования и работы с такого рода клиническими наборами данных. Кроме того, необходимо определить указанные предикторы, влияющие на возникновение острого коронарного синдрома и оказывающие большое влияние на МАСЕ. К сожалению, старые модели прогнозирования в основном основаны на регрессии или их точность колеблется от 65 до 84% [10]. Кроме того, эти модели зависят от нескольких факторов риска. Существуют и другие



факторы риска, которые оказывают большее влияние на возникновение острых коронарных синдромов. Кроме того, есть и другие факторы, которые мы должны вывести из других атрибутов и которые оказывают большое влияние на острый коронарный синдром.

Таким образом, в данной работе предлагается ансамблевый классификатор на основе машинного обучения с мягким голосованием, который может заниматься ранней диагностикой и прогнозом у пациентов с острым коронарным синдромом и обеспечить наилучший метод борьбы с возникновением сердечных событий. Основной целью данной работы является разработка модели прогнозирования риска для раннего выявления **случаев MACE в течение двухлетнего наблюдения после выписки из стационара у пациентов с острым коронарным синдромом**. Содержание нашего исследования также можно резюмировать следующим образом.

Во-первых, для экспериментов мы используем набор данных **Корейского регистра острого инфаркта миокарда (KAMIR-NIH)** [11], который разделен на две подгруппы: ИМпST и ИМпNST.

Во-вторых, мы предлагаем классификатор ансамбля мягкого голосования с использованием алгоритмов машинного обучения, таких как случайный лес (RF), дополнительное дерево (ET) и машина градиентного бустинга (GBM), для повышения точности диагностики и прогнозирования случаев **MACE** [12], таких как **сердечная смерть, несердечная смерть, инфаркт миокарда (ИМ), повторное чрескожное коронарное вмешательство (re-PCI) и аортокоронарное шунтирование (АКШ)**.

В-третьих, мы уточним предикторы риска MACE для групп ИМпST и ИМпNST между предыдущими моделями и нашей новой моделью и сравним результаты этих моделей. Наконец, мы сравниваем результаты прогнозирования случаев MACE в группах ИМпST и NSTEMI в течение двухлетнего наблюдения после выписки из больницы между применяемыми методами машинного обучения (RF, ET и GBM) и нашим классификатором ансамбля мягкого голосования с помощью показателей эффективности: **точность, точность, полнота, оценка F1 и площадь под ROC-кривой (AUC)**.

В-четвертых, мы проводим отбор наилучших моделей по критерию **AAAA**



Материалы и методы

Исследовательский анализ данных (EDA)

Введение в EDA

Исследовательский анализ данных (Exploratory data analysis, EDA) используется для анализа и исследования наборов данных и обобщения их основных характеристик

EDA помогает определить, как лучше всего манипулировать источниками данных для получения необходимых ответов, упрощая специалистам по обработке и анализу данных обнаружение закономерностей, выявление аномалий, проверку гипотез или предположений.

EDA в основном используется для того, чтобы увидеть, что данные могут выявить за пределами формального моделирования или проверки гипотез, и обеспечивает лучшее понимание переменных набора данных и отношений между ними. Это также может помочь определить, подходят ли статистические методы, которые вы рассматриваете для анализа данных.

Основная цель EDA — помочь взглянуть на данные, прежде чем делать какие-либо предположения. Это может помочь выявить очевидные ошибки, а также лучше понять закономерности в данных, обнаружить выбросы или аномальные события, найти интересные связи между переменными.

Специалисты по обработке и анализу данных могут использовать исследовательский анализ, чтобы убедиться, что результаты, которые они получают, являются достоверными и применимыми к любым желаемым бизнес-результатам и целям. EDA также помогает заинтересованным сторонам, подтверждая, что они задают правильные вопросы. EDA может помочь ответить на вопросы о стандартных отклонениях, категориальных переменных и доверительных интервалах.

Конкретные статистические функции и методы, которые можно выполнять с помощью инструментов EDA, включают:

Методы кластеризации и уменьшения размерности, которые помогают создавать графические представления многомерных данных, содержащих множество переменных.

Одномерная визуализация каждого поля в необработанном наборе данных со сводной статистикой.

Двумерные визуализации и сводная статистика, которые позволяют оценить связь между каждой переменной в наборе данных и целевой переменной, которую вы просматриваете.

Многомерные визуализации для сопоставления и понимания взаимодействий между различными полями данных.

Кластеризация K-средних — это метод кластеризации в , при котором точки данных распределяются по K-группам, т.е. по количеству кластеров, в зависимости от расстояния от центроида каждой группы. Точки данных, ближайшие к определенному центроиду, будут объединены в одну категорию. Кластеризация K-



средних обычно используется для сегментации рынка, распознавания образов и сжатия изображений.

Прогностические модели, такие как линейная регрессия, используют статистику и данные для прогнозирования результатов.

Существует четыре основных типа EDA:

Одномерный неграфический. Это простейшая форма анализа данных, при которой анализируемые данные состоят всего из одной переменной. Поскольку это одна переменная, она не имеет отношения к причинам или отношениям. Основной целью одномерного анализа является описание данных и поиск закономерностей, которые в них существуют.

Одномерная графика. Неграфические методы не дают полной картины данных. Поэтому требуются графические методы. К распространенным типам одномерной графики относятся:

Штамбовые и листовые графики, которые показывают все значения данных и форму распределения.

Гистограммы, гистограмма, на которой каждый столбец представляет частоту (количество) или долю (количество/общее количество) вариантов для диапазона значений.

Ящичковые диаграммы, которые графически изображают сводку из пяти чисел минимума, первого квартиля, медианы, третьего квартиля и максимума.

Многомерные неграфические: Многомерные данные возникают из нескольких переменных. Многомерные неграфические методы САПР обычно показывают взаимосвязь между двумя или более переменными данных с помощью перекрестной таблицы или статистики.

Многовариантная графика: Многомерные данные используют графику для отображения связей между двумя или более наборами данных. Наиболее часто используемым рисунком является сгруппированная линейчатая диаграмма или линейчатая диаграмма, где каждая группа представляет один уровень одной из переменных, а каждая полоса в группе представляет уровни другой переменной.

К другим распространенным типам многомерной графики относятся:

Точечная диаграмма, которая используется для отображения точек данных по горизонтальной и вертикальной оси, чтобы показать, насколько одна переменная подвержена влиянию другой.

Многомерная диаграмма, представляющая собой графическое представление взаимосвязей между факторами и реакцией.

Запустите диаграмму, которая представляет собой линейный график данных, построенный во времени.

Пузырьковая диаграмма, представляющая собой визуализацию данных, отображающую несколько кругов (пузырьков) на двумерном графике.

Тепловая карта, представляющая собой графическое представление данных, где значения изображены цветом.

[101] (Link:)



Набор данных для анализа

Краткое описание переменных анализа

☰ Набор данных для анализа

Age: age of the patient [years]

Sex: sex of the patient [M: Male, F: Female]

ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]

RestingBP: resting blood pressure [mm Hg]

Cholesterol: serum cholesterol [mm/dl]

FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]

RestingECG: resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]

MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]

ExerciseAngina: exercise-induced angina [Y: Yes, N: No]

Oldpeak: oldpeak = ST [Numeric value measured in depression]

ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]

HeartDisease: output class [1: heart disease, 0: Normal]

Наш набор данных содержит в общей сложности 6 числовых переменных:

Age, RestingBP, Cholesterol, MaxHR, Oldpeak, HeartDisease

В нашем наборе данных имеется также 6 категориальных переменных:

Sex, ChestPainType, FastingBS, RestingECG, ExerciseAngina, ST_Slope



Преобразование категориальной переменной в числовую

Категориальные значения — это тип данных, которые могут быть сгруппированы в различные категории, такие как пол, цвет или жанр. Числовые значения — это данные, которые имеют числовое значение, например возраст, рост или цена. Нам необходимо преобразовать категориальные значения в числовые в САПР по нескольким причинам:

Требования к моделированию: Если вы планируете создавать прогнозные модели или модели машинного обучения, необходимы числовые представления. Многие алгоритмы машинного обучения, особенно основанные на математических уравнениях, работают с числовыми входными данными.

Визуализация: Числовые данные часто легче визуализировать и интерпретировать. Графики, диаграммы и другие визуализации обычно используются в САПР для получения аналитических сведений о данных, а числовые представления облегчают создание осмысленных визуализаций данных.

Статистический анализ: Некоторые статистические тесты и анализы предполагают наличие числовых данных. Например, коэффициенты корреляции, регрессионный анализ и другие статистические методы предназначены для работы с числовыми переменными.

Согласованность типов данных: преобразование категориальных переменных в числовые представления помогает поддерживать согласованность типов данных в наборе данных. Такая согласованность может упростить процесс анализа данных и сделать его более понятным.

Совместимость с методами анализа: Многие статистические методы и методы машинного обучения требуют числовых входных данных. Преобразование категориальных переменных в числовые представления позволяет применять к набору данных более широкий спектр методов анализа.

Существует два распространенных метода преобразования категориальных переменных в числовые:

One-Hot Encoding: Этот метод создает двоичные столбцы для каждой категории, представляющие наличие или отсутствие этой категории.

Кодировка метки: Этот метод присваивает каждой категории уникальное целое число.

Однако горячее кодирование более распространено для двоичных категориальных переменных, так как оно явно представляет каждую категорию независимо.

Здесь мы используем метод One-Hot Encoding для преобразования данных.



Таблица 1. Исходные данные для анализа

Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
40	M	ATA	140	289	0	Normal	172	N	0	Up	0
49	F	NAP	160	180	0	Normal	156	N	1	Flat	1
37	M	ATA	130	283	0	ST	98	N	0	Up	0
48	F	ASY	138	214	0	Normal	108	Y	1.5	Flat	1
54	M	NAP	150	195	0	Normal	122	N	0	Up	0
39	M	NAP	120	339	0	Normal	170	N	0	Up	0
45	F	ATA	130	237	0	Normal	170	N	0	Up	0
54	M	ATA	110	208	0	Normal	142	N	0	Up	0
37	M	ASY	140	207	0	Normal	130	Y	1.5	Flat	1
48	F	ATA	120	284	0	Normal	120	N	0	Up	0
37	F	NAP	130	211	0	Normal	142	N	0	Up	0
58	M	ATA	136	164	0	ST	99	Y	2	Flat	1
39	M	ATA	120	204	0	Normal	145	N	0	Up	0
49	M	ASY	140	234	0	Normal	140	Y	1	Flat	1
42	F	NAP	115	211	0	ST	137	N	0	Up	0
54	F	ATA	120	273	0	Normal	150	N	1.5	Flat	0
38	M	ASY	110	196	0	Normal	166	N	0	Flat	1
43	F	ATA	120	201	0	Normal	165	N	0	Up	0
60	M	ASY	100	248	0	Normal	125	N	1	Flat	1
36	M	ATA	120	267	0	Normal	160	N	3	Flat	1
43	F	TA	100	223	0	Normal	142	N	0	Up	0
44	M	ATA	120	184	0	Normal	142	N	1	Flat	0
49	F	ATA	124	201	0	Normal	164	N	0	Up	0
44	M	ATA	150	288	0	Normal	150	Y	3	Flat	1
40	M	NAP	130	215	0	Normal	138	N	0	Up	0
36	M	NAP	130	209	0	Normal	178	N	0	Up	0
53	M	ASY	124	260	0	ST	112	Y	3	Flat	0
52	M	ATA	120	284	0	Normal	118	N	0	Up	0
53	F	ATA	113	468	0	Normal	127	N	0	Up	0
51	M	ATA	125	188	0	Normal	145	N	0	Up	0
53	M	NAP	145	518	0	Normal	130	N	0	Flat	1
56	M	NAP	130	167	0	Normal	114	N	0	Up	0
54	M	ASY	125	224	0	Normal	122	N	2	Flat	1
41	M	ASY	130	172	0	ST	130	N	2	Flat	1
43	F	ATA	150	186	0	Normal	154	N	0	Up	0
32	M	ATA	125	254	0	Normal	155	N	0	Up	0
65	M	ASY	140	306	1	Normal	87	Y	1.5	Flat	1
41	F	ATA	110	250	0	ST	142	N	0	Up	0
48	F	ATA	120	177	1	ST	148	N	0	Up	0
48	F	ASY	150	227	0	Normal	130	Y	1	Flat	0
54	F	ATA	150	230	0	Normal	130	N	0	Up	0
54	F	NAP	130	294	0	ST	100	Y	0	Flat	1
35	M	ATA	150	264	0	Normal	168	N	0	Up	0
52	M	NAP	140	259	0	ST	170	N	0	Up	0
43	M	ASY	120	175	0	Normal	120	Y	1	Flat	1
59	M	NAP	130	318	0	Normal	120	Y	1	Flat	0
37	M	ASY	120	223	0	Normal	168	N	0	Up	0
50	M	ATA	140	216	0	Normal	170	N	0	Up	0
36	M	NAP	112	340	0	Normal	184	N	1	Flat	0
41	M	ASY	110	289	0	Normal	170	N	0	Flat	1
50	M	ASY	130	233	0	Normal	121	Y	2	Flat	1
47	F	ASY	120	205	0	Normal	98	Y	2	Flat	1
45	M	ATA	140	224	1	Normal	122	N	0	Up	0



Таблица 2. Дескриптивная статистика числовых переменных для всех пациентов

Variable	count	mean	std	min	25%	50%	75%	max
Age	918	53.51	9.43	28	47	54	60	77
RestingBP	918	132.4	18.51	0	120	130	140	200
Cholesterol	918	198.8	109.38	0	173.25	223	267	603
FastingBS	918	0.23	0.42	0	0	0	0	1
MaxHR	918	136.81	25.46	60	120	138	156	202
Oldpeak	918	0.89	1.07	-2.6	0	0.6	1.5	6.2
HeartDisease	918	0.55	0.5	0	0	1	1	1



Таблица 3. Распределение категориальных переменных по частоте для всех пациентов

Sex	ChestPainType	FastingBS	RestingECG	ExerciseAngina	ST_Slope	count
M	ASY	0	Normal	Y	Flat	84
M	ATA	0	Normal	N	Up	59
M	NAP	0	Normal	N	Up	40
M	ASY	0	Normal	N	Flat	36
M	ASY	0	Normal	N	Up	35
F	ATA	0	Normal	N	Up	33
M	ASY	0	ST	Y	Flat	31
M	ASY	1	Normal	Y	Flat	25
M	ASY	0	LVH	Y	Flat	25
M	ASY	1	Normal	N	Flat	19
M	ASY	1	ST	Y	Flat	18
M	NAP	0	Normal	Y	Flat	17
F	ASY	0	Normal	Y	Flat	16
F	NAP	0	Normal	N	Up	16
M	ASY	0	Normal	Y	Down	13
M	ASY	1	ST	N	Flat	12
M	ASY	0	LVH	N	Up	11
M	NAP	0	Normal	N	Flat	11
M	ASY	0	Normal	Y	Up	10
M	ATA	0	ST	N	Up	10
M	ASY	0	ST	N	Up	10
M	ASY	0	ST	Y	Up	10
M	ASY	0	LVH	N	Flat	10
F	NAP	0	Normal	N	Flat	10
M	NAP	0	LVH	N	Up	10
M	ASY	0	LVH	Y	Up	9
M	ATA	0	LVH	N	Up	9
M	NAP	0	LVH	N	Flat	9
M	ASY	1	Normal	N	Up	8
F	ATA	0	ST	N	Up	8
F	ASY	0	Normal	N	Up	8
M	NAP	0	ST	Y	Flat	7
M	ASY	1	LVH	Y	Flat	7
F	NAP	0	LVH	N	Up	7
F	ASY	0	LVH	N	Flat	7
M	ATA	0	Normal	N	Flat	6
M	NAP	1	Normal	N	Flat	6
M	NAP	1	Normal	N	Up	6
M	ASY	1	Normal	Y	Down	6
M	ASY	1	Normal	Y	Up	6

Таблица 4. Основная статистика для переменной HeartDisease='Healthy'

Variable	count	mean	std	min	25%	50%	75%	max
Age	410.0	50.55	9.44	28.0	43.0	51.0	57.0	76.0
RestingBP	410.0	130.18	16.5	80.0	120.0	130.0	140.0	190.0
Cholesterol	410.0	227.12	74.63	0.0	197.25	227.0	266.75	564.0
FastingBS	410.0	0.11	0.31	0.0	0.0	0.0	0.0	1.0
MaxHR	410.0	148.15	23.29	69.0	134.0	150.0	165.0	202.0
Oldpeak	410.0	0.41	0.7	-1.1	0.0	0.0	0.6	4.2
HeartDisease	410.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0



**Таблица 5. Основная статистика для переменной
HeartDisease='Heart Disease'**

Variable	count	mean	std	min	25%	50%	75%	max
Age	508.0	55.9	8.73	31.0	51.0	57.0	62.0	77.0
RestingBP	508.0	134.19	19.83	0.0	120.0	132.0	145.0	200.0
Cholesterol	508.0	175.94	126.39	0.0	0.0	217.0	267.0	603.0
FastingBS	508.0	0.33	0.47	0.0	0.0	0.0	1.0	1.0
MaxHR	508.0	127.66	23.39	60.0	112.0	126.0	144.25	195.0
Oldpeak	508.0	1.27	1.15	-2.6	0.0	1.2	2.0	6.2
HeartDisease	508.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0

**График №1. BoxPlot для всех числовых переменных по классу
Заболевание (HeartDisease)**

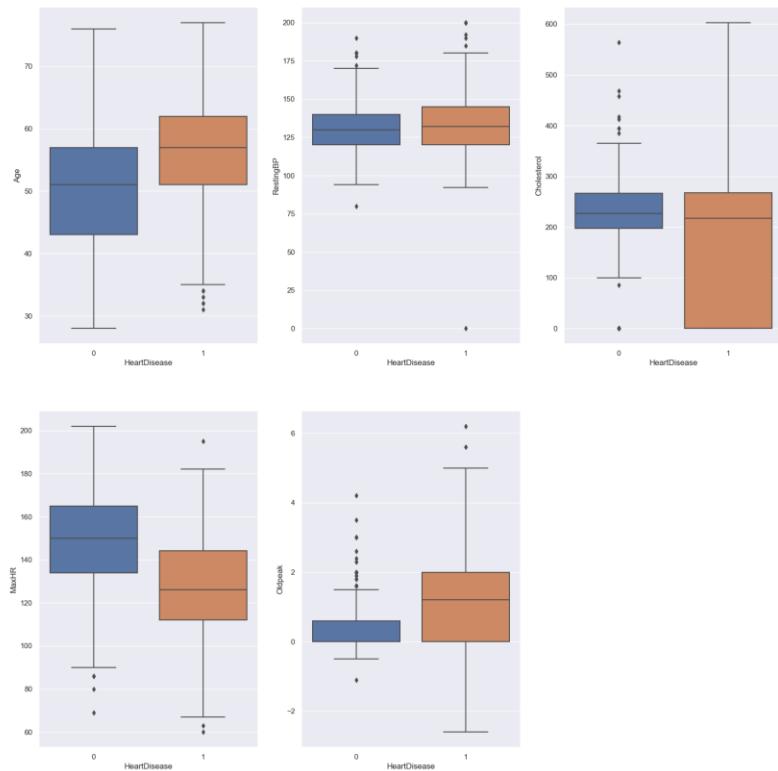




График №2. BoxPlot для всех числовых переменных Boxplots for each variable

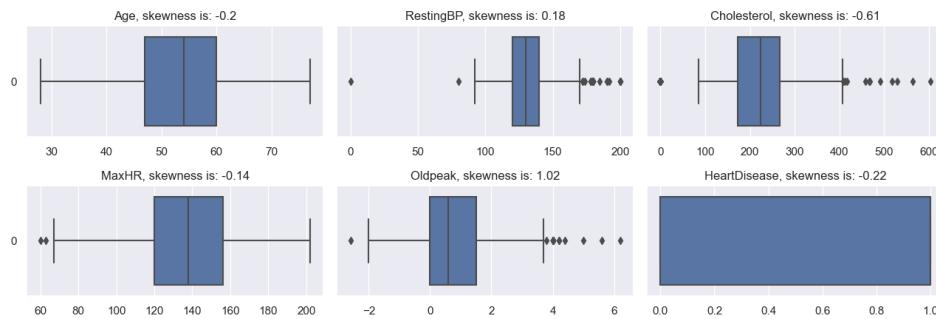


Таблица №6. Распределение пациентов по заданным категориям

Analysis: Oldpeak by Geschlecht, Alter								
		N	Min	Max	Range	Mean	Median	Std
Female	30 - <39 years	19	0	4	4	2	1	1
	40 - <65 years	356	0	6	6	2	2	1
	65 - <75 years	63	0	4	4	2	2	1
	75+ years	4	2	4	2	3	3	1
Male	30 - <39 years	4	1	3	2	1	1	1
	40 - <65 years	77	0	6	6	1	1	1
	65 - <75 years	13	0	3	2	1	1	1
	75+ years	1	1	1	0	1	1	
Total	30 - <39 years	23	0	4	4	2	1	1
	40 - <65 years	433	0	6	6	2	2	1
	65 - <75 years	76	0	4	4	1	2	1
	75+ years	5	1	4	2	2	2	1



График №3. Распределение числовых переменных по классу пол (Sex)

EDA на наборе данных показала нам, как каждая переменная связана с переменной отклика и как мы можем сделать нашу модель эффективной, используя различные методы EDA. Визуализации данных поднимают наше понимание набора данных на более высокий уровень, позволяя нам делать выводы.

Интеграция классификатора дерева принятия решений в наш анализ расширяет прогностические возможности нашей модели. В этом блоге мы не только изучили набор данных с помощью методов EDA, но и сделали еще один шаг вперед, внедрив модель машинного обучения. Такой целостный подход позволяет нам использовать сильные стороны как статистического анализа, так и прогнозного моделирования, способствуя более глубокому пониманию сложной динамики, связанной с экстремальными погодными явлениями.

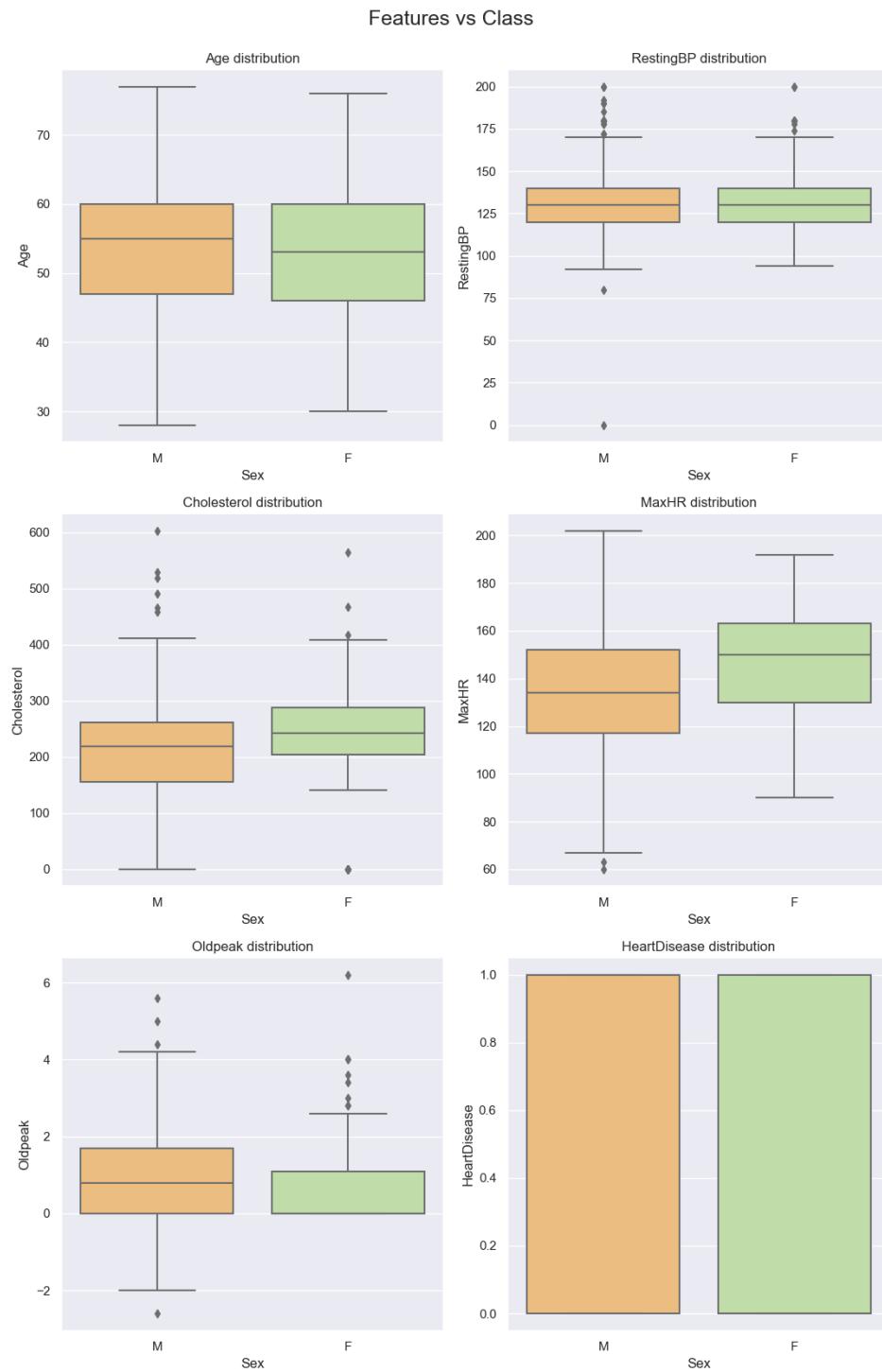


Таблица №7. Распределение пациентов по заданным категориям



Age		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Female	0 - 29 years	28.75	4	100.00			0.00	28.75	4	100.00
	30 - <39 years	36.68	34	59.65	36.17	23	40.35	36.47	57	100.00
	40 - <65 years	51.41	213	36.72	54.73	367	63.28	53.51	580	100.00
	65 - <75 years	67.80	15	19.23	68.10	63	80.77	68.04	78	100.00
	75+ years	75.00	1	16.67	76.00	5	83.33	75.83	6	100.00
	Всего	50.20	267	36.83	55.87	458	63.17	53.78	725	100.00
Male	0 - 29 years									
	30 - <39 years	35.56	16	84.21	36.33	3	15.79	35.68	19	100.00
	40 - <65 years	51.12	112	72.26	56.51	43	27.74	52.61	155	100.00
	65 - <75 years	68.00	14	77.78	67.50	4	22.22	67.89	18	100.00
	75+ years	76.00	1	100.00			0.00	76.00	1	100.00
	Всего	51.20	143	74.09	56.18	50	25.91	52.49	193	100.00
Total	0 - 29 years	28.75	4	100.00			0.00	28.75	4	100.00
	30 - <39 years	36.32	50	65.79	36.19	26	34.21	36.28	76	100.00
	40 - <65 years	51.31	325	44.22	54.92	410	55.78	53.32	735	100.00
	65 - <75 years	67.90	29	30.21	68.06	67	69.79	68.01	96	100.00
	75+ years	75.50	2	28.57	76.00	5	71.43	75.86	7	100.00
	Всего	50.55	410	44.66	55.90	508	55.34	53.51	918	100.00
Всего	0 - 29 years	28.75	8	100.00			0.00	28.75	8	100.00
	30 - <39 years	36.32	100	65.79	36.19	52	34.21	36.28	152	100.00
	40 - <65 years	51.31	650	44.22	54.92	820	55.78	53.32	1470	100.00
	65 - <75 years	67.90	58	30.21	68.06	134	69.79	68.01	192	100.00
	75+ years	75.50	4	28.57	76.00	10	71.43	75.86	14	100.00
	Всего	50.55	820	44.66	55.90	1016	55.34	53.51	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №4. Гистограммы распределения для всех числовых переменных по классу Заболевание (HeartDisease)

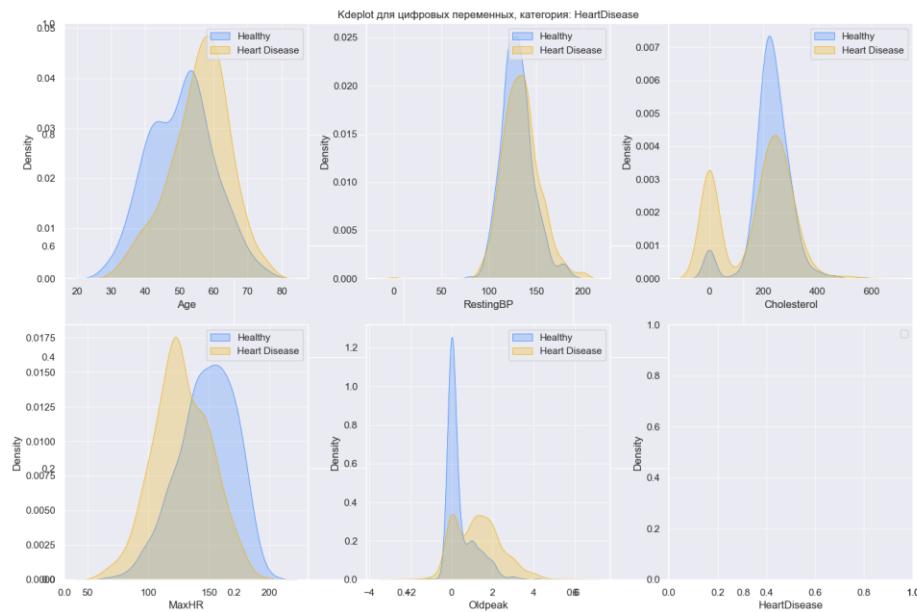


Таблица №8. Распределение пациентов по заданным категориям

Cholesterol		Healthy			Heart Disease			Всего			
		Mean	N	%	Mean	N	%	Mean	N	%	
Female	0 - 29 years	210.50	4	100.00				0.00	210.50	4	100.00
	30 - <39 years	240.91	33	67.35	248.13	16	32.65	243.27	49	100.00	
	40 - <65 years	231.16	199	44.42	249.12	249	55.58	241.14	448	100.00	
	65 - <75 years	239.45	11	19.30	243.33	46	80.70	242.58	57	100.00	
	75+ years	310.00	1	16.67	203.20	5	83.33	221.00	6	100.00	
	Всего	232.81	248	43.97	247.50	316	56.03	241.04	564	100.00	
Male	0 - 29 years										
	30 - <39 years	204.38	16	94.12	246.00	1	5.88	206.82	17	100.00	
	40 - <65 years	251.06	111	75.00	282.92	37	25.00	259.03	148	100.00	
	65 - <75 years	289.21	14	87.50	226.50	2	12.50	281.38	16	100.00	
	75+ years	197.00	1	100.00			0.00	197.00	1	100.00	
	Всего	249.18	142	78.02	279.18	40	21.98	255.77	182	100.00	
Total	0 - 29 years	210.50	4	100.00				0.00	210.50	4	100.00
	30 - <39 years	228.98	49	74.24	248.00	17	25.76	233.88	66	100.00	
	40 - <65 years	238.28	310	52.01	253.50	286	47.99	245.58	596	100.00	
	65 - <75 years	267.32	25	34.25	242.63	48	65.75	251.08	73	100.00	
	75+ years	253.50	2	28.57	203.20	5	71.43	217.57	7	100.00	
	Всего	238.77	390	52.28	251.06	356	47.72	244.64	746	100.00	



Cholesterol		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Всего	0 - 29 years	210.50	8	100.00			0.00	210.50	8	100.00
	30 - <39 years	228.98	98	74.24	248.00	34	25.76	233.88	132	100.00
	40 - <65 years	238.28	620	52.01	253.50	572	47.99	245.58	1192	100.00
	65 - <75 years	267.32	50	34.25	242.63	96	65.75	251.08	146	100.00
	75+ years	253.50	4	28.57	203.20	10	71.43	217.57	14	100.00
	Всего	238.77	780	52.28	251.06	712	47.72	244.64	1492	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №5. Гистограммы распределения для всех числовых переменных по классу пол (Sex)

Таблица №9. Распределение пациентов по заданным категориям

RestingBP		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Female	0 - 29 years	130.00	4	100.00			0.00	130.00	4	100.00
	30 - <39 years	128.50	34	59.65	118.43	23	40.35	124.44	57	100.00
	40 - <65 years	130.69	213	36.79	133.52	366	63.21	132.47	579	100.00
	65 - <75 years	138.13	15	19.23	139.94	63	80.77	139.59	78	100.00
	75+ years	160.00	1	16.67	131.80	5	83.33	136.50	6	100.00
	Всего	130.93	267	36.88	133.62	457	63.12	132.63	724	100.00
Male	0 - 29 years									
	30 - <39 years	123.94	16	84.21	105.00	3	15.79	120.95	19	100.00
	40 - <65 years	128.64	112	72.26	142.84	43	27.74	132.58	155	100.00
	65 - <75 years	134.71	14	77.78	160.75	4	22.22	140.50	18	100.00
	75+ years	140.00	1	100.00			0.00	140.00	1	100.00
	Всего	128.79	143	74.09	142.00	50	25.91	132.21	193	100.00
Total	0 - 29 years	130.00	4	100.00			0.00	130.00	4	100.00
	30 - <39 years	127.04	50	65.79	116.88	26	34.21	123.57	76	100.00
	40 - <65 years	129.98	325	44.28	134.50	409	55.72	132.50	734	100.00
	65 - <75 years	136.48	29	30.21	141.18	67	69.79	139.76	96	100.00
	75+ years	150.00	2	28.57	131.80	5	71.43	137.00	7	100.00
	Всего	130.18	410	44.71	134.45	507	55.29	132.54	917	100.00
Всего	0 - 29 years	130.00	8	100.00			0.00	130.00	8	100.00
	30 - <39 years	127.04	100	65.79	116.88	52	34.21	123.57	152	100.00
	40 - <65 years	129.98	650	44.28	134.50	818	55.72	132.50	1468	100.00
	65 - <75 years	136.48	58	30.21	141.18	134	69.79	139.76	192	100.00
	75+ years	150.00	4	28.57	131.80	10	71.43	137.00	14	100.00
	Всего	130.18	820	44.71	134.45	1014	55.29	132.54	1834	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №6. Двумерное распределение переменной 'HeartDisease' по классам Sex и RestingBP

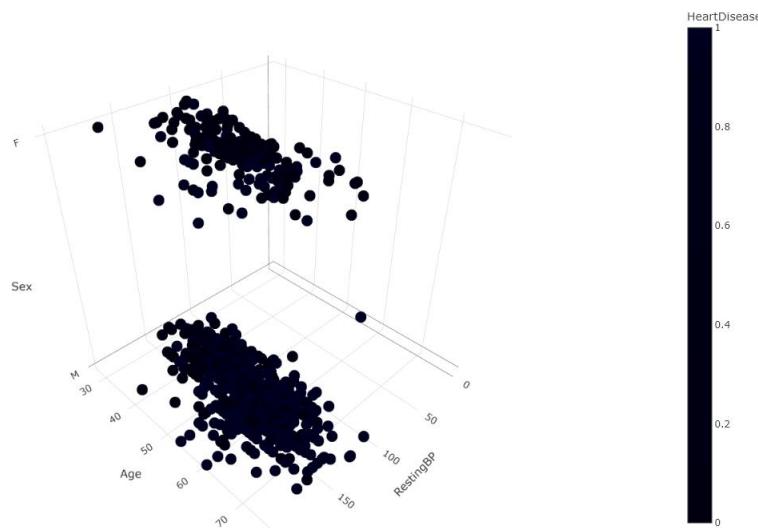


Таблица №10. Распределение пациентов по заданным категориям

MaxHR		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Female	0 - 29 years	179.25	4	100.00			0.00	179.25	4	100.00
	30 - <39 years	159.53	34	59.65	143.91	23	40.35	153.23	57	100.00
	40 - <65 years	146.34	213	36.72	126.53	367	63.28	133.80	580	100.00
	65 - <75 years	133.60	15	19.23	120.65	63	80.77	123.14	78	100.00
	75+ years	112.00	1	16.67	122.40	5	83.33	120.67	6	100.00
	Всего	147.67	267	36.83	126.55	458	63.17	134.33	725	100.00
Male	0 - 29 years									
	30 - <39 years	165.63	16	84.21	157.33	3	15.79	164.32	19	100.00
	40 - <65 years	147.75	112	72.26	137.88	43	27.74	145.01	155	100.00
	65 - <75 years	142.86	14	77.78	122.50	4	22.22	138.33	18	100.00
	75+ years	116.00	1	100.00			0.00	116.00	1	100.00
	Всего	149.05	143	74.09	137.82	50	25.91	146.14	193	100.00
Total	0 - 29 years	179.25	4	100.00			0.00	179.25	4	100.00
	30 - <39 years	161.48	50	65.79	145.46	26	34.21	156.00	76	100.00
	40 - <65 years	146.83	325	44.22	127.72	410	55.78	136.17	735	100.00
	65 - <75 years	138.07	29	30.21	120.76	67	69.79	125.99	96	100.00
	75+ years	114.00	2	28.57	122.40	5	71.43	120.00	7	100.00
	Всего	148.15	410	44.66	127.66	508	55.34	136.81	918	100.00



MaxHR		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Всего	0 - 29 years	179.25	8	100.00			0.00	179.25	8	100.00
	30 - <39 years	161.48	100	65.79	145.46	52	34.21	156.00	152	100.00
	40 - <65 years	146.83	650	44.22	127.72	820	55.78	136.17	1470	100.00
	65 - <75 years	138.07	58	30.21	120.76	134	69.79	125.99	192	100.00
	75+ years	114.00	4	28.57	122.40	10	71.43	120.00	14	100.00
	Всего	148.15	820	44.66	127.66	1016	55.34	136.81	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №7. График распределение переменной Cholesterol по Возрасту (Age)

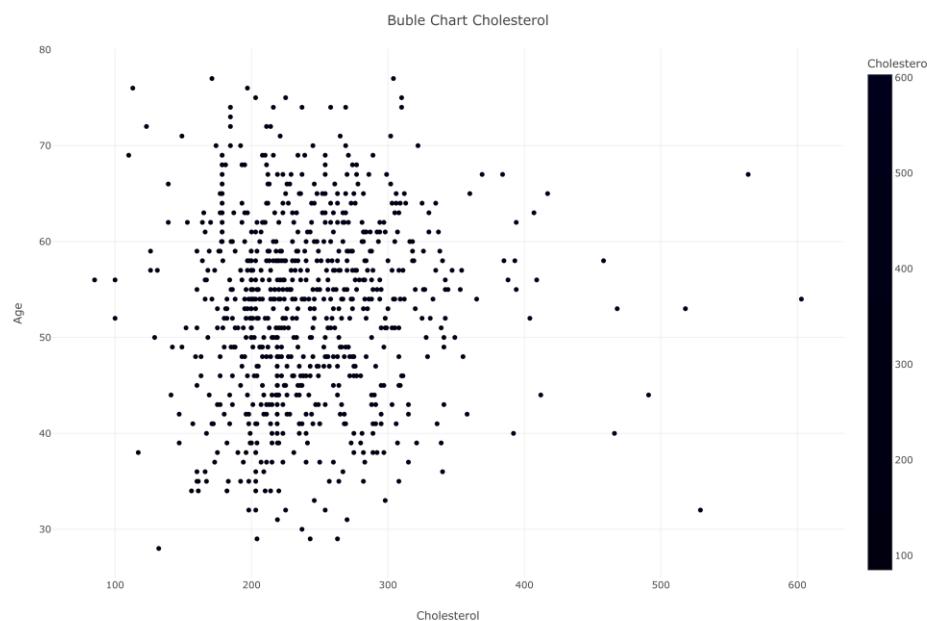


Таблица №11. Распределение пациентов по заданным категориям

Oldpeak		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Female	30 - <39 years	1.70	5	26.32	1.53	14	73.68	1.57	19	100.00
	40 - <65 years	1.04	84	23.60	1.73	272	76.40	1.57	356	100.00
	65 - <75 years	0.76	12	19.05	1.76	51	80.95	1.57	63	100.00
	75+ years	2.00	1	25.00	2.83	3	75.00	2.63	4	100.00
	Всего	1.05	102	23.08	1.74	340	76.92	1.58	442	100.00
Male	30 - <39 years	1.05	2	50.00	1.90	2	50.00	1.47	4	100.00
	40 - <65 years	1.02	46	59.74	1.97	31	40.26	1.40	77	100.00
	65 - <75 years	1.13	11	84.62	1.00	2	15.38	1.11	13	100.00
	75+ years	1.10	1	100.00			0.00	1.10	1	100.00
	Всего	1.04	60	63.16	1.91	35	36.84	1.36	95	100.00
Total	30 - <39 years	1.51	7	30.43	1.58	16	69.57	1.56	23	100.00
	40 - <65 years	1.03	130	30.02	1.76	303	69.98	1.54	433	100.00
	65 - <75 years	0.93	23	30.26	1.73	53	69.74	1.49	76	100.00
	75+ years	1.55	2	40.00	2.83	3	60.00	2.32	5	100.00
	Всего	1.04	162	30.17	1.76	375	69.83	1.54	537	100.00



Oldpeak		Healthy			Heart Disease			Всего		
		Mean	N	%	Mean	N	%	Mean	N	%
Всего	30 - <39 years	1.51	14	30.43	1.58	32	69.57	1.56	46	100.00
	40 - <65 years	1.03	260	30.02	1.76	606	69.98	1.54	866	100.00
	65 - <75 years	0.93	46	30.26	1.73	106	69.74	1.49	152	100.00
	75+ years	1.55	4	40.00	2.83	6	60.00	2.32	10	100.00
	Всего	1.04	324	30.17	1.76	750	69.83	1.54	1074	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №8. Гистограммы распределения для всех числовых переменных, представленные на одном графике

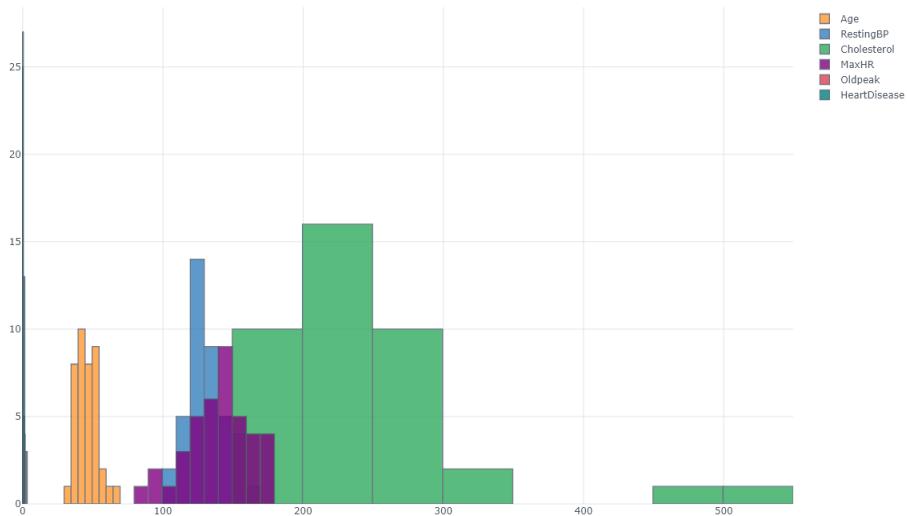


Таблица №12. Распределение пациентов по заданным категориям



ChestPainType			Healthy		Heart		Всего	
			N	%	N	%	N	%
ASY	Female	0 - 29 years						
		30 - <39 years	4	19.05	17	80.95	21	100.00
		40 - <65 years	64	18.39	284	81.61	348	100.00
		65 - <75 years	4	7.69	48	92.31	52	100.00
		75+ years	1	20.00	4	80.00	5	100.00
		Всего	73	17.14	353	82.86	426	100.00
	Male	0 - 29 years						
		30 - <39 years	3	50.00	3	50.00	6	100.00
		40 - <65 years	26	44.07	33	55.93	59	100.00
		65 - <75 years	2	40.00	3	60.00	5	100.00
		75+ years						
		Всего	31	44.29	39	55.71	70	100.00
	Total	0 - 29 years						
		30 - <39 years	7	25.93	20	74.07	27	100.00
		40 - <65 years	90	22.11	317	77.89	407	100.00
		65 - <75 years	6	10.53	51	89.47	57	100.00
		75+ years	1	20.00	4	80.00	5	100.00
		Всего	104	20.97	392	79.03	496	100.00
	Всего	0 - 29 years						
		30 - <39 years	14	25.93	40	74.07	54	100.00
		40 - <65 years	180	22.11	634	77.89	814	100.00
		65 - <75 years	12	10.53	102	89.47	114	100.00
		75+ years	2	20.00	8	80.00	10	100.00
		Всего	208	20.97	784	79.03	992	100.00



ChestPainType			Healthy		Heart		Всего	
			N	%	N	%	N	%
ATA	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	15	88.24	2	11.76	17	100.00
		40 - <65 years	73	82.02	16	17.98	89	100.00
		65 - <75 years	1	33.33	2	66.67	3	100.00
		75+ years						
		Всего	93	82.30	20	17.70	113	100.00
	Male	0 - 29 years						
		30 - <39 years	6	100.00		0.00	6	100.00
		40 - <65 years	48	92.31	4	7.69	52	100.00
		65 - <75 years	2	100.00		0.00	2	100.00
		75+ years						
		Всего	56	93.33	4	6.67	60	100.00
	Total	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	21	91.30	2	8.70	23	100.00
		40 - <65 years	121	85.82	20	14.18	141	100.00
		65 - <75 years	3	60.00	2	40.00	5	100.00
		75+ years						
		Всего	149	86.13	24	13.87	173	100.00
	Всего	0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	42	91.30	4	8.70	46	100.00
		40 - <65 years	242	85.82	40	14.18	282	100.00
		65 - <75 years	6	60.00	4	40.00	10	100.00
		75+ years						
		Всего	298	86.13	48	13.87	346	100.00



ChestPainType			Healthy		Heart		Всего	
			N	%	N	%	N	%
NAP	Female	0 - 29 years						
		30 - <39 years	14	93.33	1	6.67	15	100.00
		40 - <65 years	63	53.39	55	46.61	118	100.00
		65 - <75 years	7	43.75	9	56.25	16	100.00
		75+ years		0.00	1	100.00	1	100.00
		Всего	84	56.00	66	44.00	150	100.00
	Male	0 - 29 years						
		30 - <39 years	5	100.00		0.00	5	100.00
		40 - <65 years	33	86.84	5	13.16	38	100.00
		65 - <75 years	8	88.89	1	11.11	9	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	47	88.68	6	11.32	53	100.00
	Total	0 - 29 years						
		30 - <39 years	19	95.00	1	5.00	20	100.00
		40 - <65 years	96	61.54	60	38.46	156	100.00
		65 - <75 years	15	60.00	10	40.00	25	100.00
		75+ years	1	50.00	1	50.00	2	100.00
		Всего	131	64.53	72	35.47	203	100.00
	Всего	0 - 29 years						
		30 - <39 years	38	95.00	2	5.00	40	100.00
		40 - <65 years	192	61.54	120	38.46	312	100.00
		65 - <75 years	30	60.00	20	40.00	50	100.00
		75+ years	2	50.00	2	50.00	4	100.00
		Всего	262	64.53	144	35.47	406	100.00



ChestPainType			Healthy		Heart		Всего	
			N	%	N	%	N	%
TA	Female	0 - 29 years						
		30 - <39 years	1	25.00	3	75.00	4	100.00
		40 - <65 years	13	52.00	12	48.00	25	100.00
		65 - <75 years	3	42.86	4	57.14	7	100.00
		75+ years						
		Всего	17	47.22	19	52.78	36	100.00
	Male	0 - 29 years						
		30 - <39 years	2	100.00		0.00	2	100.00
		40 - <65 years	5	83.33	1	16.67	6	100.00
		65 - <75 years	2	100.00		0.00	2	100.00
		75+ years						
		Всего	9	90.00	1	10.00	10	100.00
	Total	0 - 29 years						
		30 - <39 years	3	50.00	3	50.00	6	100.00
		40 - <65 years	18	58.06	13	41.94	31	100.00
		65 - <75 years	5	55.56	4	44.44	9	100.00
		75+ years						
		Всего	26	56.52	20	43.48	46	100.00
	Всего	0 - 29 years						
		30 - <39 years	6	50.00	6	50.00	12	100.00
		40 - <65 years	36	58.06	26	41.94	62	100.00
		65 - <75 years	10	55.56	8	44.44	18	100.00
		75+ years						
		Всего	52	56.52	40	43.48	92	100.00



ChestPainType			Healthy		Heart		Всего	
			N	%	N	%	N	%
Всего	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	34	59.65	23	40.35	57	100.00
		40 - <65 years	213	36.72	367	63.28	580	100.00
		65 - <75 years	15	19.23	63	80.77	78	100.00
		75+ years	1	16.67	5	83.33	6	100.00
	Male	Всего	267	36.83	458	63.17	725	100.00
		0 - 29 years						
		30 - <39 years	16	84.21	3	15.79	19	100.00
		40 - <65 years	112	72.26	43	27.74	155	100.00
		65 - <75 years	14	77.78	4	22.22	18	100.00
	Total	75+ years	1	100.00		0.00	1	100.00
		Всего	143	74.09	50	25.91	193	100.00
		0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	50	65.79	26	34.21	76	100.00
		40 - <65 years	325	44.22	410	55.78	735	100.00
	Всего	65 - <75 years	29	30.21	67	69.79	96	100.00
		75+ years	2	28.57	5	71.43	7	100.00
		Всего	410	44.66	508	55.34	918	100.00
		0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	100	65.79	52	34.21	152	100.00
	Всего	40 - <65 years	650	44.22	820	55.78	1470	100.00
		65 - <75 years	58	30.21	134	69.79	192	100.00
		75+ years	4	28.57	10	71.43	14	100.00
		Всего	820	44.66	1016	55.34	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №9. Матрица корреляции Пирсона для числовых переменных

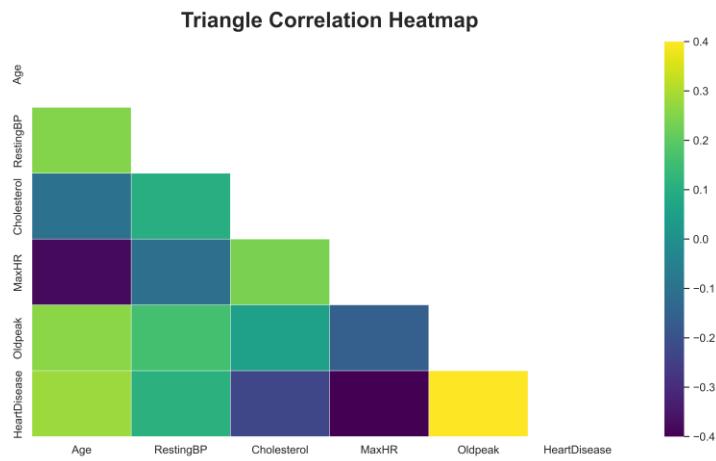


Таблица №13. Распределение пациентов по заданным категориям



ExerciseAngina			Healthy		Heart		Всего	
			N	%	N	%	N	%
N	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	33	75.00	11	25.00	44	100.00
		40 - <65 years	178	57.23	133	42.77	311	100.00
		65 - <75 years	13	36.11	23	63.89	36	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	228	57.43	169	42.57	397	100.00
	Male	0 - 29 years						
		30 - <39 years	16	88.89	2	11.11	18	100.00
		40 - <65 years	97	84.35	18	15.65	115	100.00
		65 - <75 years	13	81.25	3	18.75	16	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	127	84.67	23	15.33	150	100.00
	Total	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	49	79.03	13	20.97	62	100.00
		40 - <65 years	275	64.55	151	35.45	426	100.00
		65 - <75 years	26	50.00	26	50.00	52	100.00
		75+ years	1	33.33	2	66.67	3	100.00
		Всего	355	64.90	192	35.10	547	100.00
	Всего	0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	98	79.03	26	20.97	124	100.00
		40 - <65 years	550	64.55	302	35.45	852	100.00
		65 - <75 years	52	50.00	52	50.00	104	100.00
		75+ years	2	33.33	4	66.67	6	100.00
		Всего	710	64.90	384	35.10	1094	100.00



ExerciseAngina			Healthy		Heart		Всего	
			N	%	N	%	N	%
Y	Female	0 - 29 years						
		30 - <39 years	1	7.69	12	92.31	13	100.00
		40 - <65 years	35	13.01	234	86.99	269	100.00
		65 - <75 years	2	4.76	40	95.24	42	100.00
		75+ years	1	25.00	3	75.00	4	100.00
		Всего	39	11.89	289	88.11	328	100.00
	Male	0 - 29 years						
		30 - <39 years		0.00	1	100.00	1	100.00
		40 - <65 years	15	37.50	25	62.50	40	100.00
		65 - <75 years	1	50.00	1	50.00	2	100.00
		75+ years						
		Всего	16	37.21	27	62.79	43	100.00
	Total	0 - 29 years						
		30 - <39 years	1	7.14	13	92.86	14	100.00
		40 - <65 years	50	16.18	259	83.82	309	100.00
		65 - <75 years	3	6.82	41	93.18	44	100.00
		75+ years	1	25.00	3	75.00	4	100.00
		Всего	55	14.82	316	85.18	371	100.00
	Всего	0 - 29 years						
		30 - <39 years	2	7.14	26	92.86	28	100.00
		40 - <65 years	100	16.18	518	83.82	618	100.00
		65 - <75 years	6	6.82	82	93.18	88	100.00
		75+ years	2	25.00	6	75.00	8	100.00
		Всего	110	14.82	632	85.18	742	100.00



ExerciseAngina			Healthy		Heart		Всего	
			N	%	N	%	N	%
Всего	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	34	59.65	23	40.35	57	100.00
		40 - <65 years	213	36.72	367	63.28	580	100.00
		65 - <75 years	15	19.23	63	80.77	78	100.00
		75+ years	1	16.67	5	83.33	6	100.00
	Male	Всего	267	36.83	458	63.17	725	100.00
		0 - 29 years						
		30 - <39 years	16	84.21	3	15.79	19	100.00
		40 - <65 years	112	72.26	43	27.74	155	100.00
		65 - <75 years	14	77.78	4	22.22	18	100.00
	Total	75+ years	1	100.00		0.00	1	100.00
		Всего	143	74.09	50	25.91	193	100.00
		0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	50	65.79	26	34.21	76	100.00
		40 - <65 years	325	44.22	410	55.78	735	100.00
	Всего	65 - <75 years	29	30.21	67	69.79	96	100.00
		75+ years	2	28.57	5	71.43	7	100.00
		Всего	410	44.66	508	55.34	918	100.00
		0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	100	65.79	52	34.21	152	100.00
	Всего	40 - <65 years	650	44.22	820	55.78	1470	100.00
		65 - <75 years	58	30.21	134	69.79	192	100.00
		75+ years	4	28.57	10	71.43	14	100.00
		Всего	820	44.66	1016	55.34	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №10. Гистограммы распределения для всех числовых переменных, в виде субграфиков на одной панели

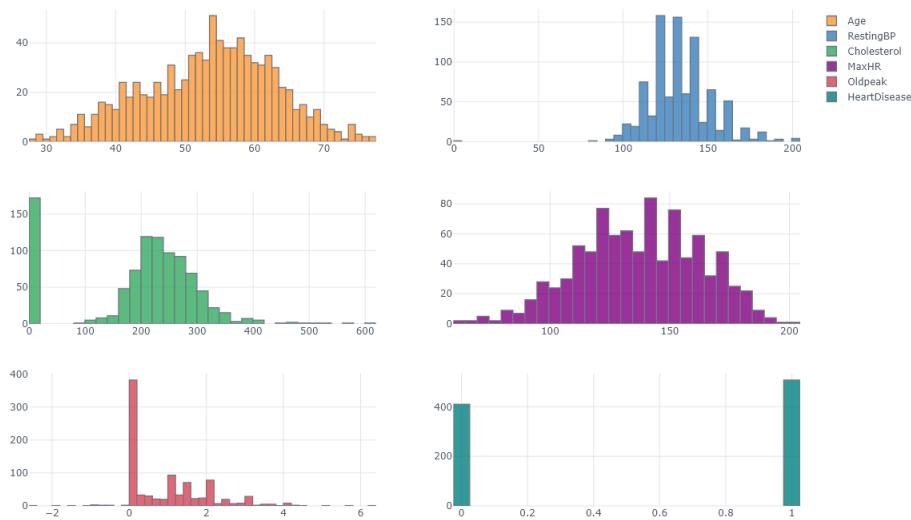


Таблица №14. Распределение пациентов по заданным категориям



RestingECG			Healthy		Heart		Всего	
			N	%	N	%	N	%
LVH	Female	0 - 29 years	2	100.00		0.00	2	100.00
		30 - <39 years	4	80.00	1	20.00	5	100.00
		40 - <65 years	36	32.73	74	67.27	110	100.00
		65 - <75 years	7	31.82	15	68.18	22	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	49	34.75	92	65.25	141	100.00
	Male	0 - 29 years						
		30 - <39 years						
		40 - <65 years	26	66.67	13	33.33	39	100.00
		65 - <75 years	7	87.50	1	12.50	8	100.00
		75+ years						
		Всего	33	70.21	14	29.79	47	100.00
	Total	0 - 29 years	2	100.00		0.00	2	100.00
		30 - <39 years	4	80.00	1	20.00	5	100.00
		40 - <65 years	62	41.61	87	58.39	149	100.00
		65 - <75 years	14	46.67	16	53.33	30	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	82	43.62	106	56.38	188	100.00
	Всего	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	8	80.00	2	20.00	10	100.00
		40 - <65 years	124	41.61	174	58.39	298	100.00
		65 - <75 years	28	46.67	32	53.33	60	100.00
		75+ years		0.00	4	100.00	4	100.00
		Всего	164	43.62	212	56.38	376	100.00



RestingECG			Healthy		Heart		Всего	
			N	%	N	%	N	%
Normal	Female	0 - 29 years	2	100.00		0.00	2	100.00
		30 - <39 years	26	54.17	22	45.83	48	100.00
		40 - <65 years	143	41.33	203	58.67	346	100.00
		65 - <75 years	6	16.67	30	83.33	36	100.00
		75+ years	1	50.00	1	50.00	2	100.00
		Всего	178	41.01	256	58.99	434	100.00
	Male	0 - 29 years						
		30 - <39 years	11	78.57	3	21.43	14	100.00
		40 - <65 years	71	74.74	24	25.26	95	100.00
		65 - <75 years	7	77.78	2	22.22	9	100.00
		75+ years						
		Всего	89	75.42	29	24.58	118	100.00
	Total	0 - 29 years	2	100.00		0.00	2	100.00
		30 - <39 years	37	59.68	25	40.32	62	100.00
		40 - <65 years	214	48.53	227	51.47	441	100.00
		65 - <75 years	13	28.89	32	71.11	45	100.00
		75+ years	1	50.00	1	50.00	2	100.00
		Всего	267	48.37	285	51.63	552	100.00
	Всего	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	74	59.68	50	40.32	124	100.00
		40 - <65 years	428	48.53	454	51.47	882	100.00
		65 - <75 years	26	28.89	64	71.11	90	100.00
		75+ years	2	50.00	2	50.00	4	100.00
		Всего	534	48.37	570	51.63	1104	100.00



RestingECG			Healthy		Heart		Всего	
			N	%	N	%	N	%
ST	Female	0 - 29 years						
		30 - <39 years	4	100.00		0.00	4	100.00
		40 - <65 years	34	27.42	90	72.58	124	100.00
		65 - <75 years	2	10.00	18	90.00	20	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	40	26.67	110	73.33	150	100.00
	Male	0 - 29 years						
		30 - <39 years	5	100.00		0.00	5	100.00
		40 - <65 years	15	71.43	6	28.57	21	100.00
		65 - <75 years		0.00	1	100.00	1	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	21	75.00	7	25.00	28	100.00
	Total	0 - 29 years						
		30 - <39 years	9	100.00		0.00	9	100.00
		40 - <65 years	49	33.79	96	66.21	145	100.00
		65 - <75 years	2	9.52	19	90.48	21	100.00
		75+ years	1	33.33	2	66.67	3	100.00
		Всего	61	34.27	117	65.73	178	100.00
	Всего	0 - 29 years						
		30 - <39 years	18	100.00		0.00	18	100.00
		40 - <65 years	98	33.79	192	66.21	290	100.00
		65 - <75 years	4	9.52	38	90.48	42	100.00
		75+ years	2	33.33	4	66.67	6	100.00
		Всего	122	34.27	234	65.73	356	100.00



RestingECG			Healthy		Heart		Всего	
			N	%	N	%	N	%
Всего	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	34	59.65	23	40.35	57	100.00
		40 - <65 years	213	36.72	367	63.28	580	100.00
		65 - <75 years	15	19.23	63	80.77	78	100.00
		75+ years	1	16.67	5	83.33	6	100.00
		Всего	267	36.83	458	63.17	725	100.00
		0 - 29 years						
	Male	30 - <39 years	16	84.21	3	15.79	19	100.00
		40 - <65 years	112	72.26	43	27.74	155	100.00
		65 - <75 years	14	77.78	4	22.22	18	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	143	74.09	50	25.91	193	100.00
		0 - 29 years						
		Total	0 - 29 years	4	100.00		0.00	4
	Всего	30 - <39 years	50	65.79	26	34.21	76	100.00
		40 - <65 years	325	44.22	410	55.78	735	100.00
		65 - <75 years	29	30.21	67	69.79	96	100.00
		75+ years	2	28.57	5	71.43	7	100.00
		Всего	410	44.66	508	55.34	918	100.00
		0 - 29 years						
		0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	100	65.79	52	34.21	152	100.00
		40 - <65 years	650	44.22	820	55.78	1470	100.00
		65 - <75 years	58	30.21	134	69.79	192	100.00
		75+ years	4	28.57	10	71.43	14	100.00
		Всего	820	44.66	1016	55.34	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №11. Распределение переменной Cholesterol по классу (Age, Sex, FastingBS) в виде 4 субграфиков на одной панели

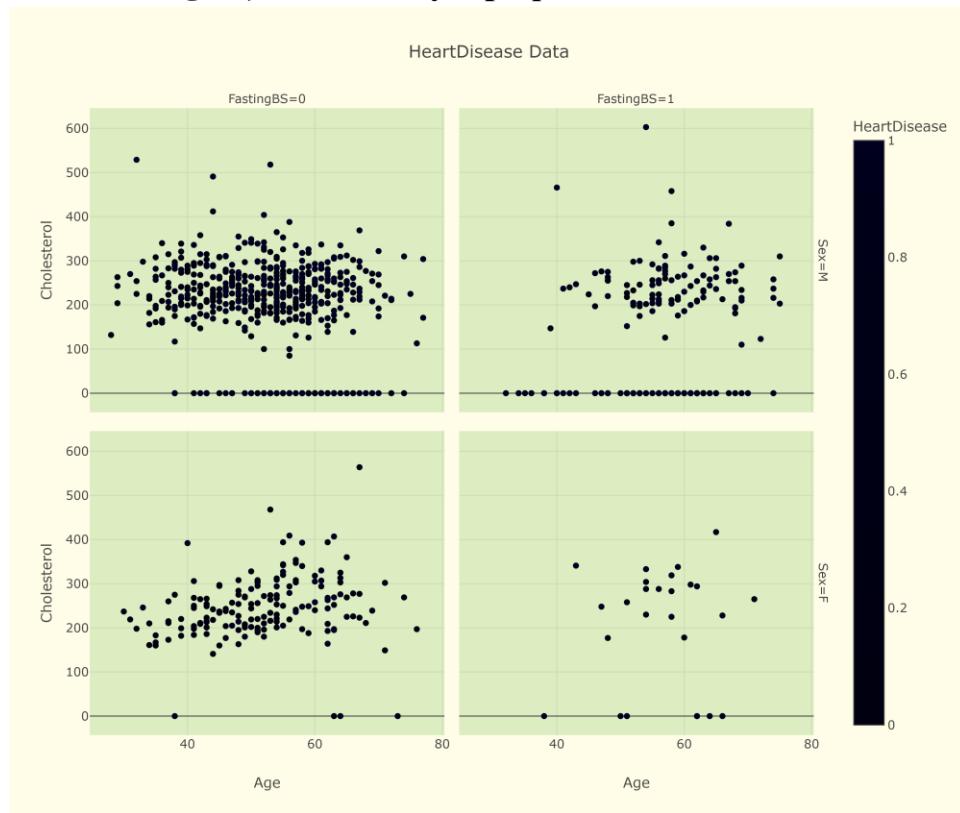


Таблица №15. Распределение пациентов по заданным категориям



ST_Slope			Healthy		Heart		Всего	
			N	%	N	%	N	%
Down	Female	0 - 29 years						
		30 - <39 years	1	50.00	1	50.00	2	100.00
		40 - <65 years	10	22.22	35	77.78	45	100.00
		65 - <75 years		0.00	7	100.00	7	100.00
		75+ years	1	50.00	1	50.00	2	100.00
		Всего	12	21.43	44	78.57	56	100.00
	Male	0 - 29 years						
		30 - <39 years						
		40 - <65 years	1	16.67	5	83.33	6	100.00
		65 - <75 years	1	100.00		0.00	1	100.00
		75+ years						
		Всего	2	28.57	5	71.43	7	100.00
	Total	0 - 29 years						
		30 - <39 years	1	50.00	1	50.00	2	100.00
		40 - <65 years	11	21.57	40	78.43	51	100.00
		65 - <75 years	1	12.50	7	87.50	8	100.00
		75+ years	1	50.00	1	50.00	2	100.00
		Всего	14	22.22	49	77.78	63	100.00
	Всего	0 - 29 years						
		30 - <39 years	2	50.00	2	50.00	4	100.00
		40 - <65 years	22	21.57	80	78.43	102	100.00
		65 - <75 years	2	12.50	14	87.50	16	100.00
		75+ years	2	50.00	2	50.00	4	100.00
		Всего	28	22.22	98	77.78	126	100.00



ST_Slope			Healthy		Heart		Всего	
			N	%	N	%	N	%
Flat	Female	0 - 29 years						
		30 - <39 years	3	13.64	19	86.36	22	100.00
		40 - <65 years	36	11.69	272	88.31	308	100.00
		65 - <75 years	4	7.55	49	92.45	53	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	43	11.17	342	88.83	385	100.00
	Male	0 - 29 years						
		30 - <39 years	1	33.33	2	66.67	3	100.00
		40 - <65 years	30	46.88	34	53.13	64	100.00
		65 - <75 years	4	57.14	3	42.86	7	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	36	48.00	39	52.00	75	100.00
	Total	0 - 29 years						
		30 - <39 years	4	16.00	21	84.00	25	100.00
		40 - <65 years	66	17.74	306	82.26	372	100.00
		65 - <75 years	8	13.33	52	86.67	60	100.00
		75+ years	1	33.33	2	66.67	3	100.00
		Всего	79	17.17	381	82.83	460	100.00
	Всего	0 - 29 years						
		30 - <39 years	8	16.00	42	84.00	50	100.00
		40 - <65 years	132	17.74	612	82.26	744	100.00
		65 - <75 years	16	13.33	104	86.67	120	100.00
		75+ years	2	33.33	4	66.67	6	100.00
		Всего	158	17.17	762	82.83	920	100.00



ST_Slope			Healthy		Heart		Всего	
			N	%	N	%	N	%
Up	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	30	90.91	3	9.09	33	100.00
		40 - <65 years	167	73.57	60	26.43	227	100.00
		65 - <75 years	11	61.11	7	38.89	18	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	212	74.65	72	25.35	284	100.00
	Male	0 - 29 years						
		30 - <39 years	15	93.75	1	6.25	16	100.00
		40 - <65 years	81	95.29	4	4.71	85	100.00
		65 - <75 years	9	90.00	1	10.00	10	100.00
		75+ years						
		Всего	105	94.59	6	5.41	111	100.00
	Total	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	45	91.84	4	8.16	49	100.00
		40 - <65 years	248	79.49	64	20.51	312	100.00
		65 - <75 years	20	71.43	8	28.57	28	100.00
		75+ years		0.00	2	100.00	2	100.00
		Всего	317	80.25	78	19.75	395	100.00
	Всего	0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	90	91.84	8	8.16	98	100.00
		40 - <65 years	496	79.49	128	20.51	624	100.00
		65 - <75 years	40	71.43	16	28.57	56	100.00
		75+ years		0.00	4	100.00	4	100.00
		Всего	634	80.25	156	19.75	790	100.00



ST_Slope			Healthy		Heart		Всего	
			N	%	N	%	N	%
Всего	Female	0 - 29 years	4	100.00		0.00	4	100.00
		30 - <39 years	34	59.65	23	40.35	57	100.00
		40 - <65 years	213	36.72	367	63.28	580	100.00
		65 - <75 years	15	19.23	63	80.77	78	100.00
		75+ years	1	16.67	5	83.33	6	100.00
		Всего	267	36.83	458	63.17	725	100.00
		0 - 29 years						
	Male	30 - <39 years	16	84.21	3	15.79	19	100.00
		40 - <65 years	112	72.26	43	27.74	155	100.00
		65 - <75 years	14	77.78	4	22.22	18	100.00
		75+ years	1	100.00		0.00	1	100.00
		Всего	143	74.09	50	25.91	193	100.00
		0 - 29 years						
		Total	0 - 29 years	4	100.00		0.00	4
	Всего	30 - <39 years	50	65.79	26	34.21	76	100.00
		40 - <65 years	325	44.22	410	55.78	735	100.00
		65 - <75 years	29	30.21	67	69.79	96	100.00
		75+ years	2	28.57	5	71.43	7	100.00
		Всего	410	44.66	508	55.34	918	100.00
		0 - 29 years						
		0 - 29 years	8	100.00		0.00	8	100.00
		30 - <39 years	100	65.79	52	34.21	152	100.00
		40 - <65 years	650	44.22	820	55.78	1470	100.00
		65 - <75 years	58	30.21	134	69.79	192	100.00
		75+ years	4	28.57	10	71.43	14	100.00
		Всего	820	44.66	1016	55.34	1836	100.00

© Dr. Alexander Wagner. Все права охраняются законом



График №12. Распределение по возрасту (Age) для переменных: 'Sex','ChestPainType','FastingBS','RestingECG','ExerciseAngina','ST_Slope','HeartDisease' в форме Виалин-графиков

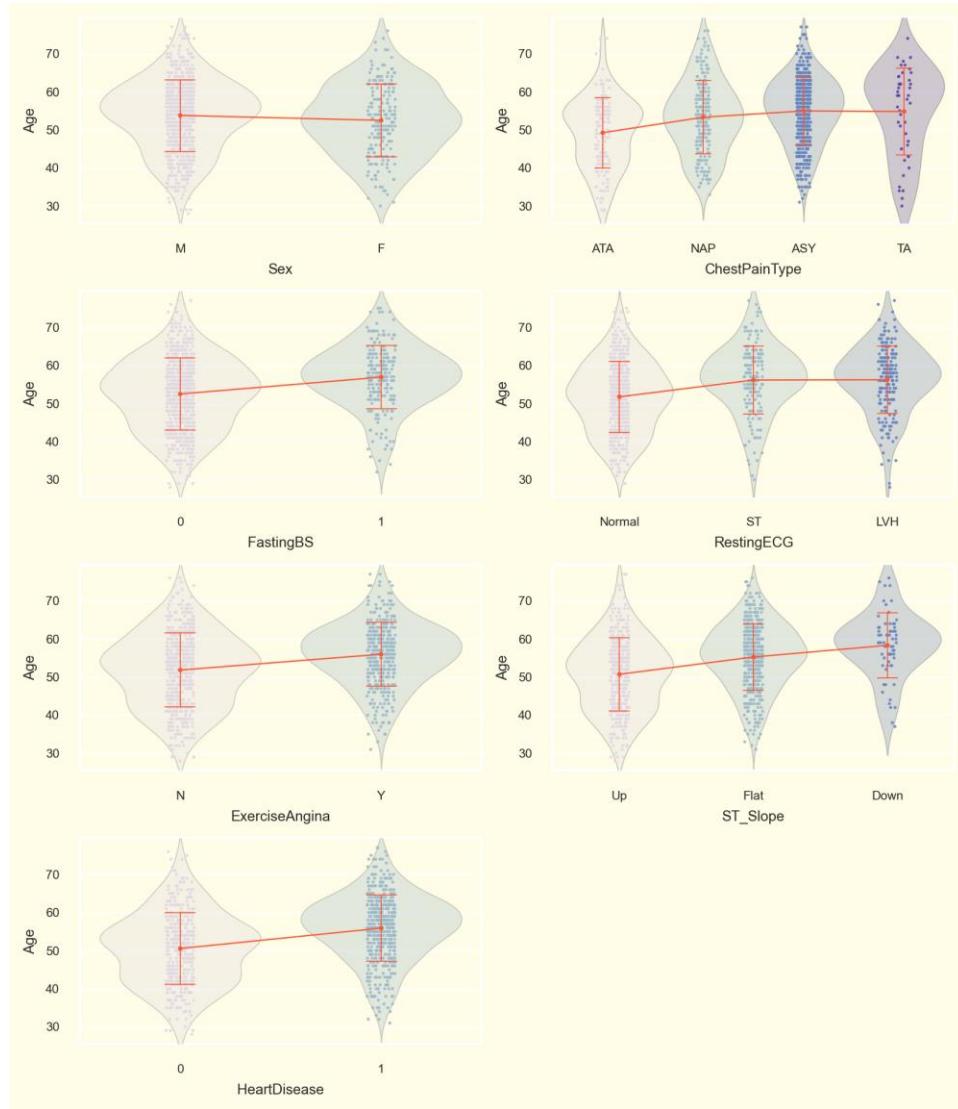


Таблица №16. Распределение пациентов по заданным категориям



График №13. Распределение всех числовых переменных в виде субграфиков по классу 'HeartDisease' на одной панели

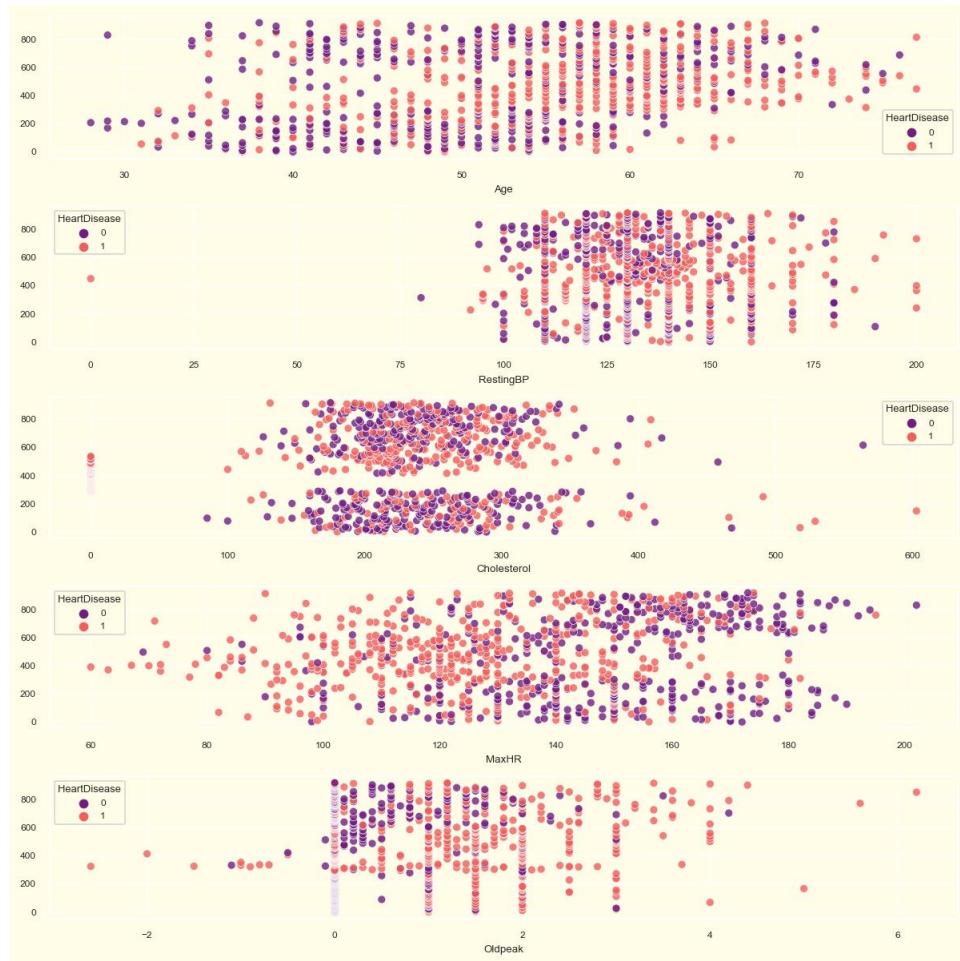


Таблица №17. Распределение пациентов по заданным категориям



График №14. Распределение всех числовых переменных в виде столбиковых диаграмм как субграфиков по классу 'HeartDisease' на одной панели

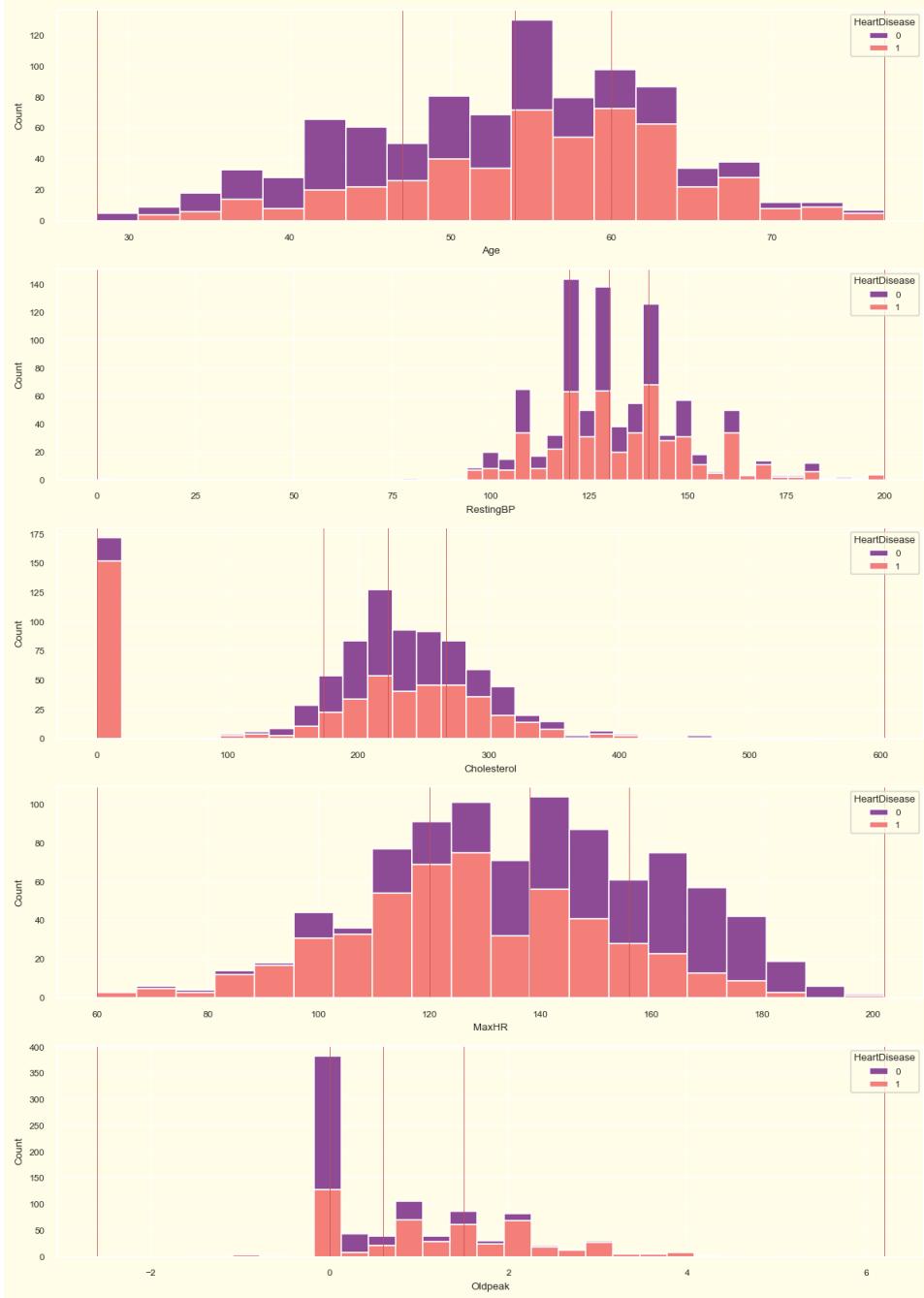


Таблица №18. Распределение пациентов по заданным категориям



График №15. Распределение численности пациентов по всем переменным в виде столбиковых диаграмм как субграфиков по классу пол (Sex) на одной панели

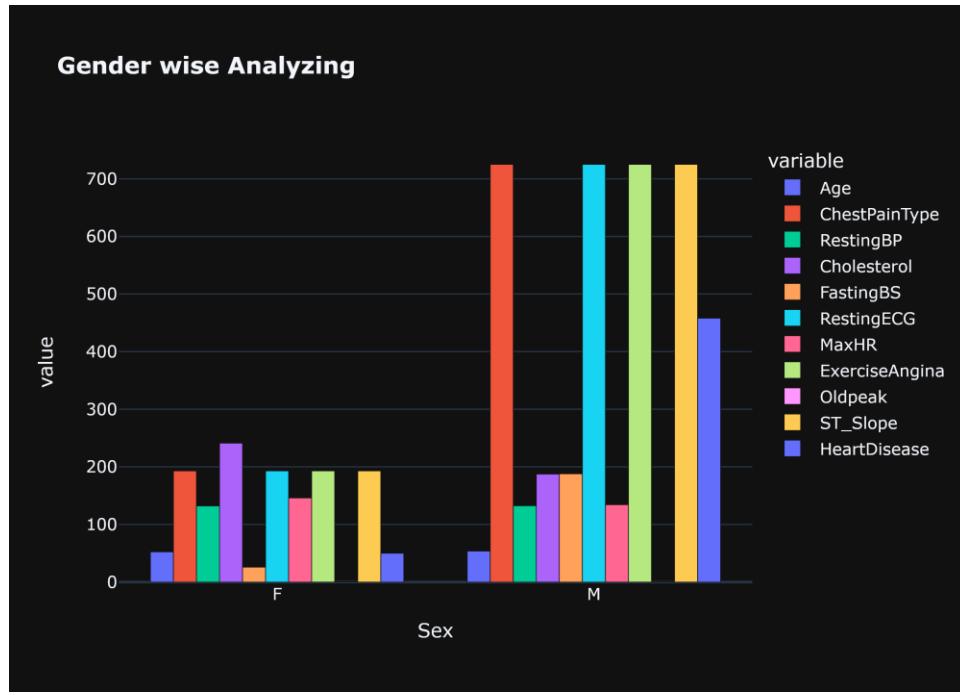


Таблица №19. Распределение пациентов по заданным категориям



График №16. Биполярное распределение переменной Cholesterol по возрасту 'Age'

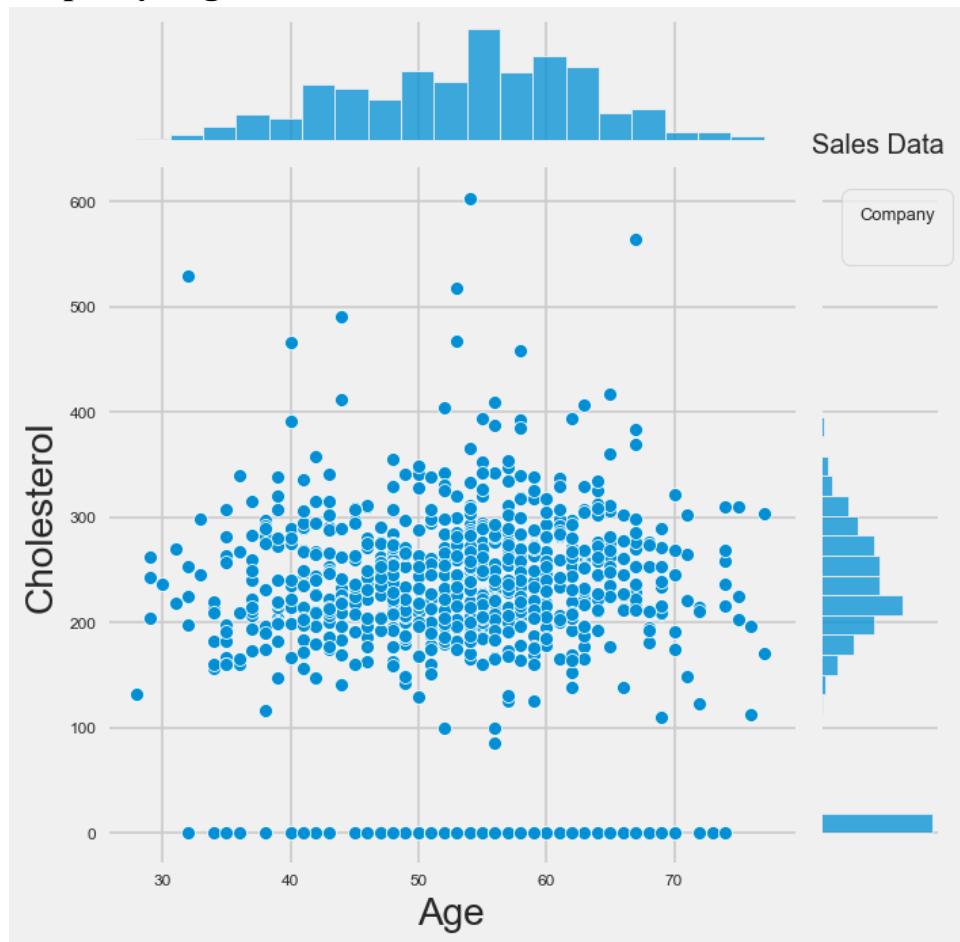




График №17. Биполярное распределение разного графического типа всех численных переменных на одной панели

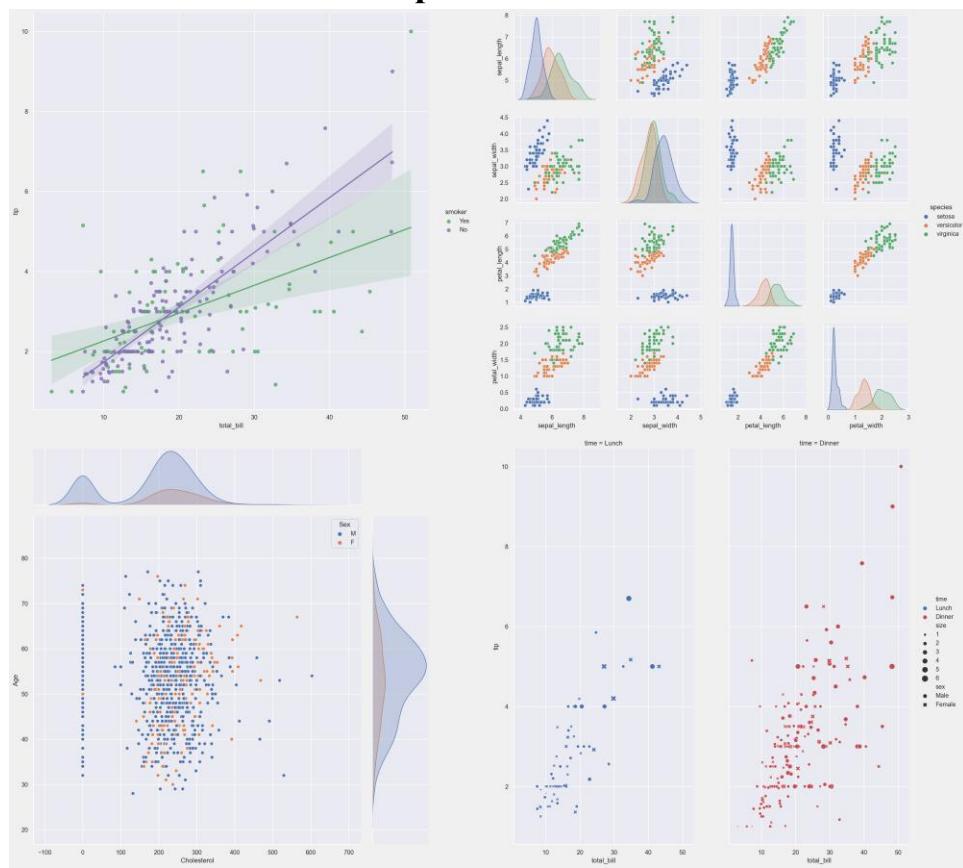




График №18. Плотность распределения переменной Cholesterol по классу пол (Sex)

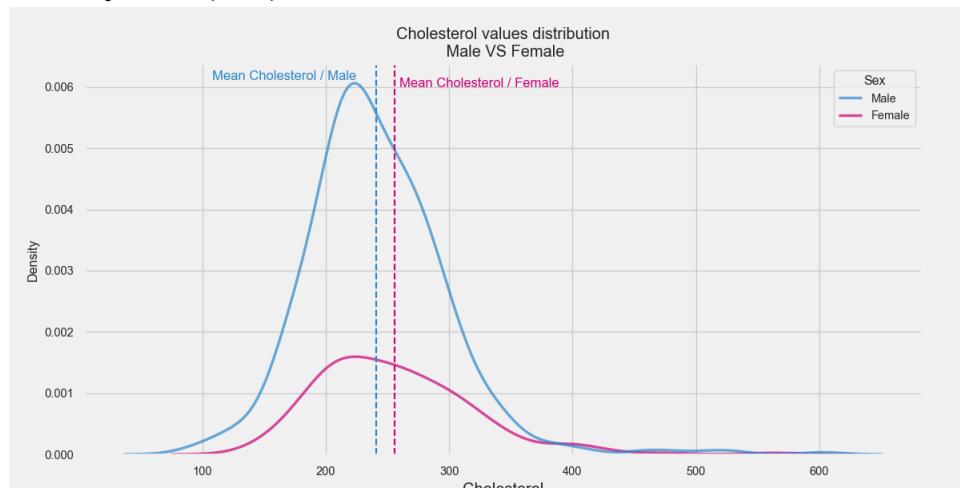




График №19. Распределение переменной Age возрастным группам в виде столбиковых диаграмм по классу 'HeartDisease'

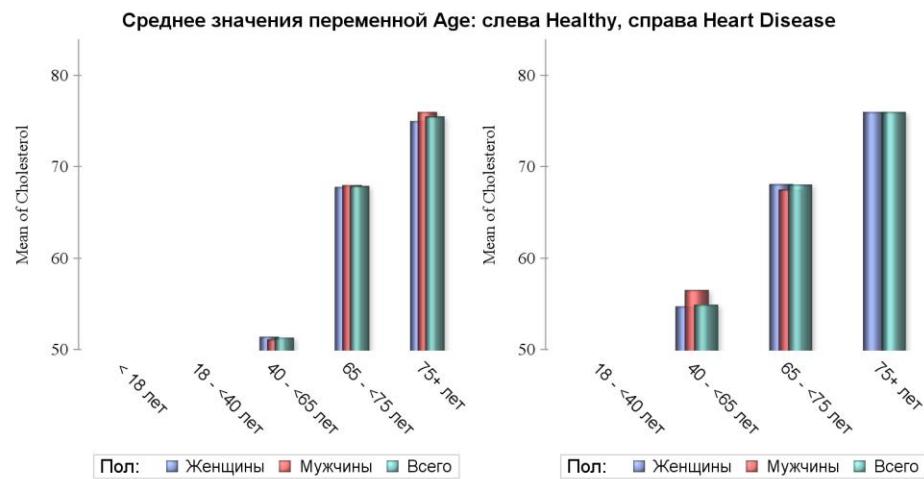




График №20. Распределение переменной Cholesterol по возрастным группам в виде столбиковых диаграмм по классу 'HeartDisease'

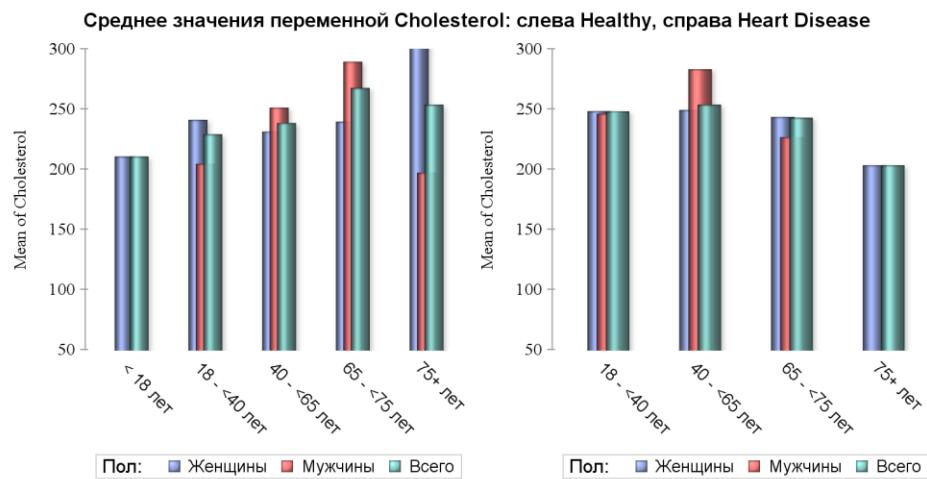




График №21. Распределение переменной RestingBP по возрастным группам в виде столбиковых диаграмм по классу 'HeartDisease'

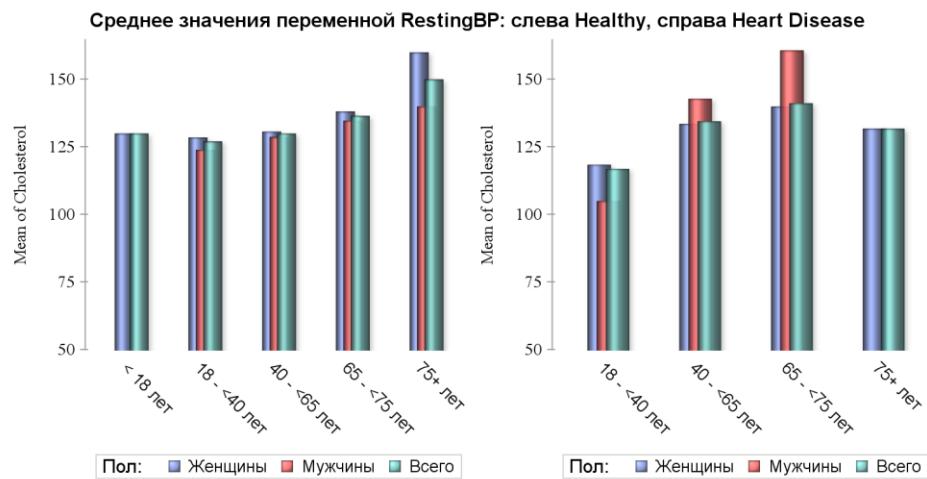




График №22. Распределение переменной MaxHR по возрастным группам в виде столбиковых диаграмм по классу 'HeartDisease'

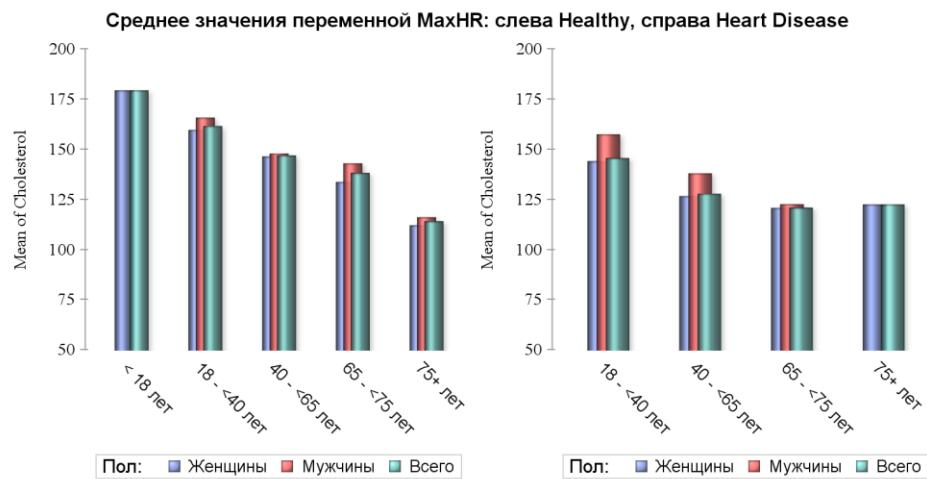




График №23. Распределение переменной Oldpeak по возрастным группам в виде столбиковых диаграмм по классу 'HeartDisease'





График №24. Торт-диаграмма распределения пациентов по классу переменной HeartDisease

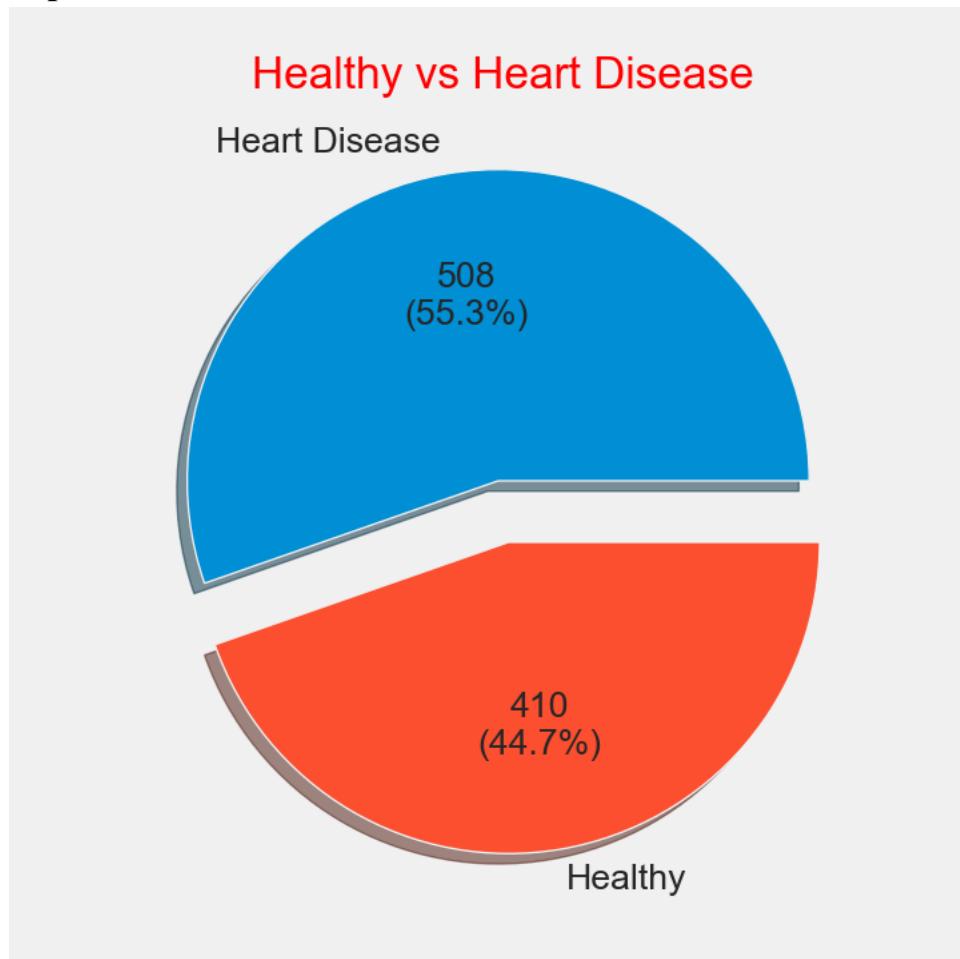




График №25. Торт-диаграмма распределения пациентов по класс-переменной Sex

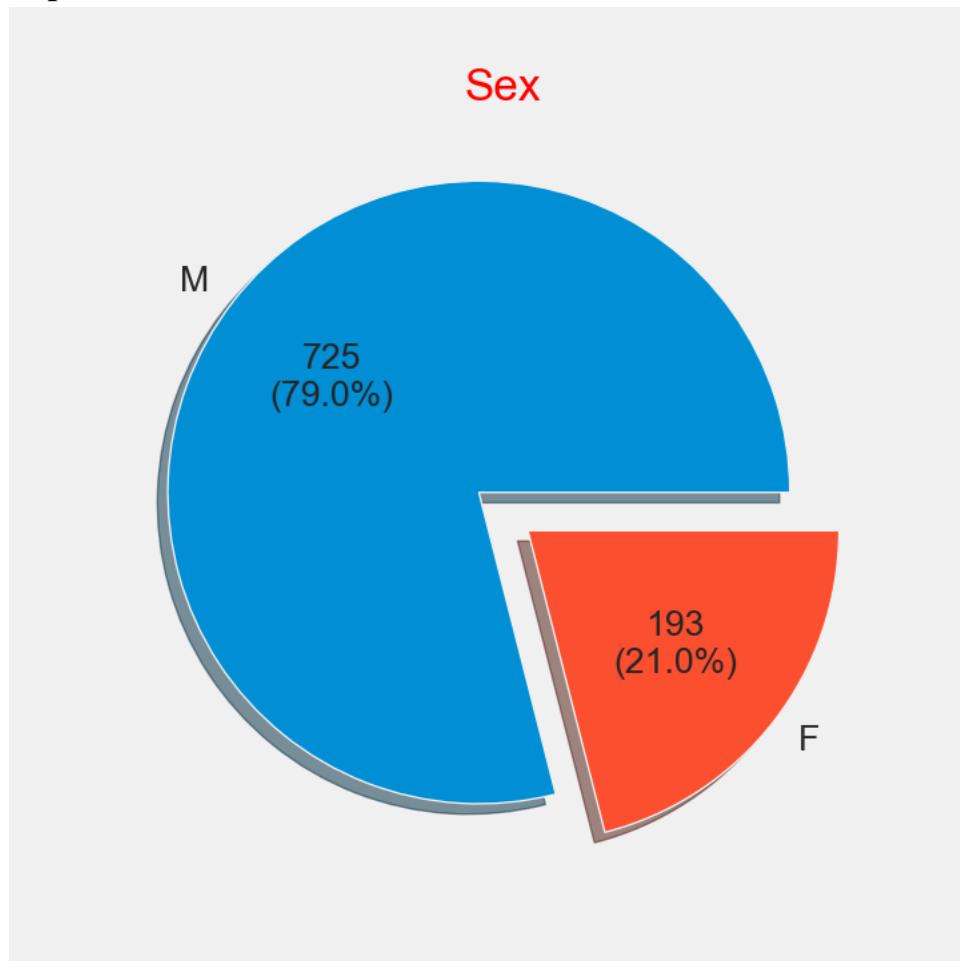




График №26. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'ST_Slope', 'HeartDisease' по категориям

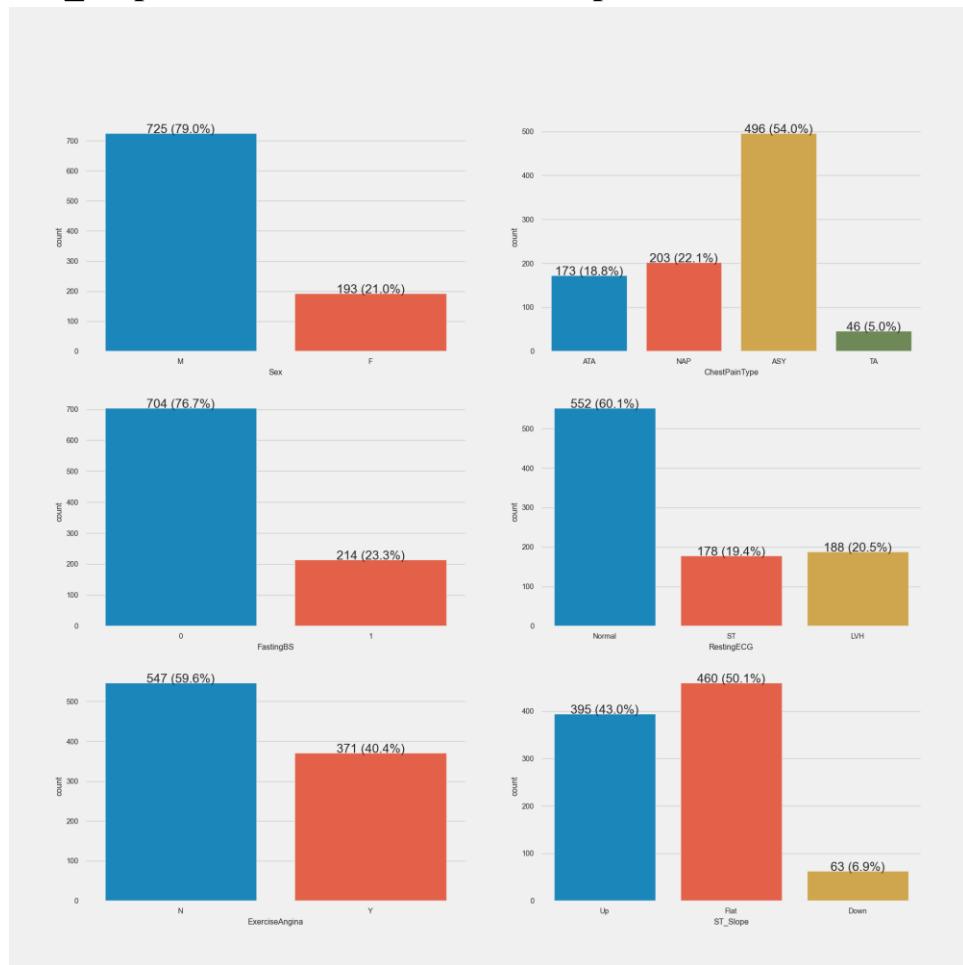




График №27. Блок-бокс диаграмма распределения переменных: 'Sex', 'ChestPainType', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'ST_Slope', 'HeartDisease' в виде субграфиков на одной панели по категориям

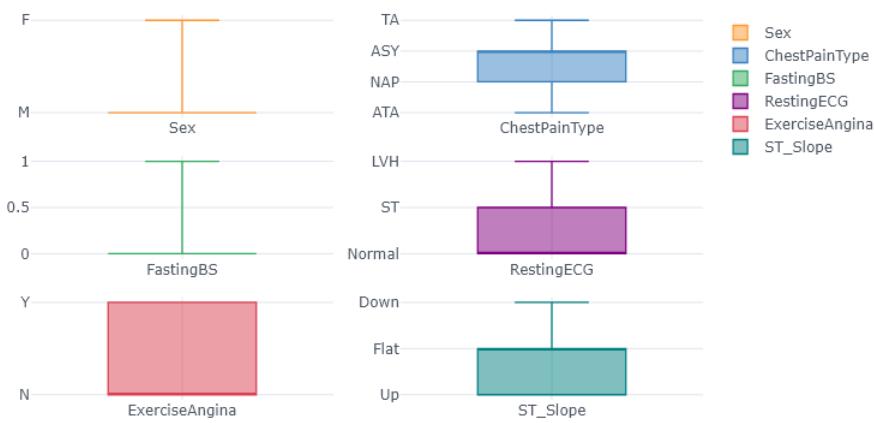




График №28. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'ST_Slope', 'HeartDisease' по категориям

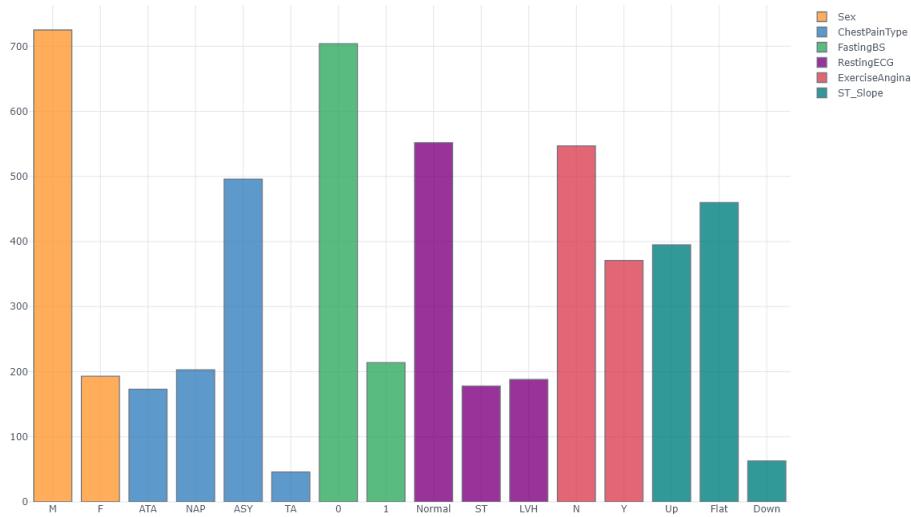




График №29. Секторная диаграмма распределения переменной ChestPainType

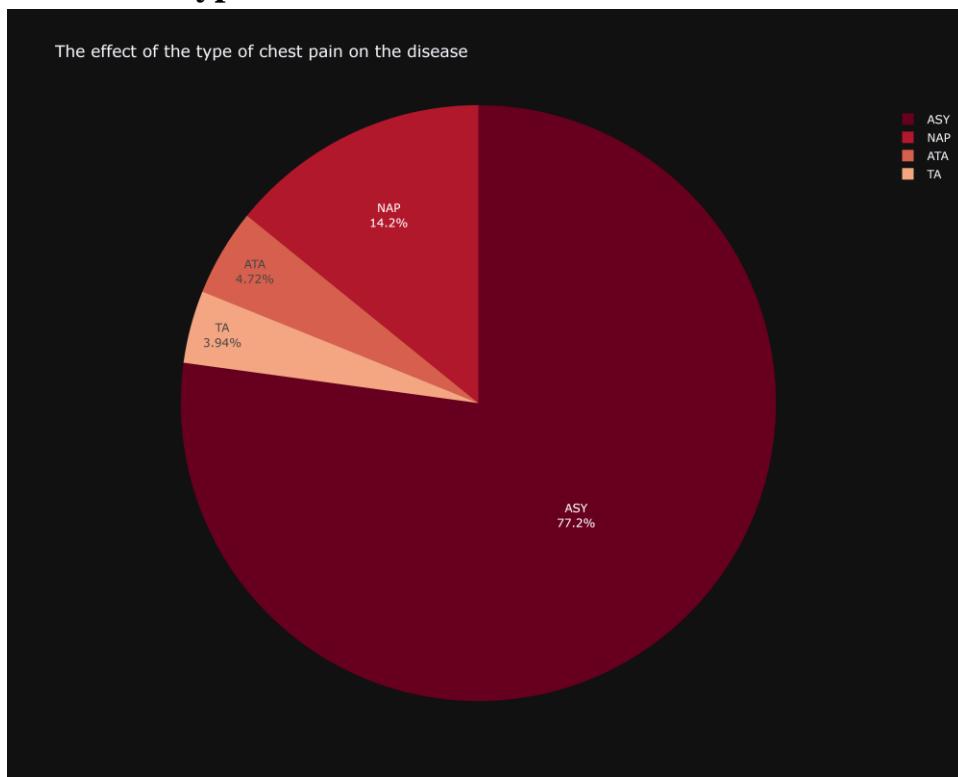




График №30. Секторная диаграмма (2-го типа) распределения переменной ST_Slope

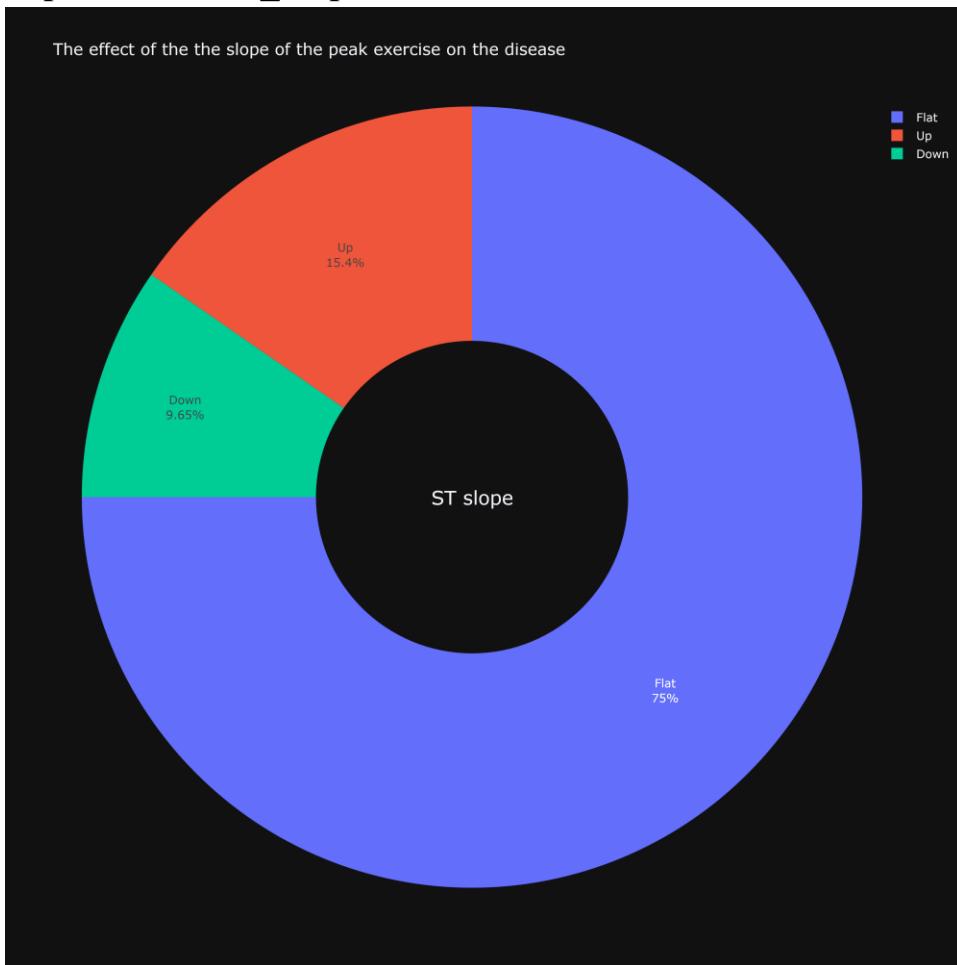




График №31. Комбинированная диаграмма распределения переменной пол (Sex)

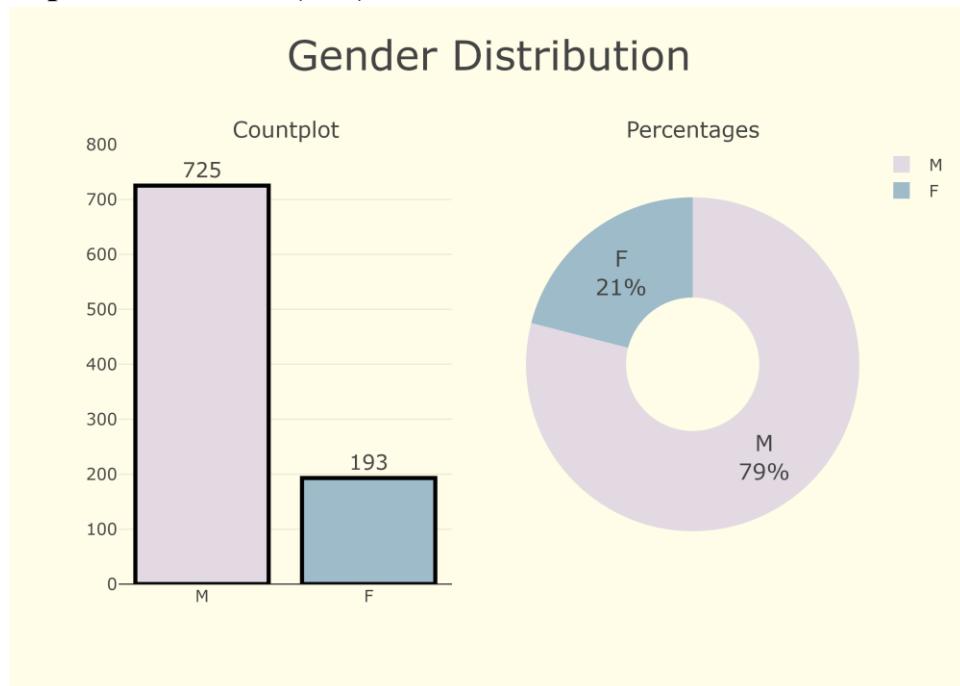




График №32. Комбинированная диаграмма распределения переменной HeartDisease

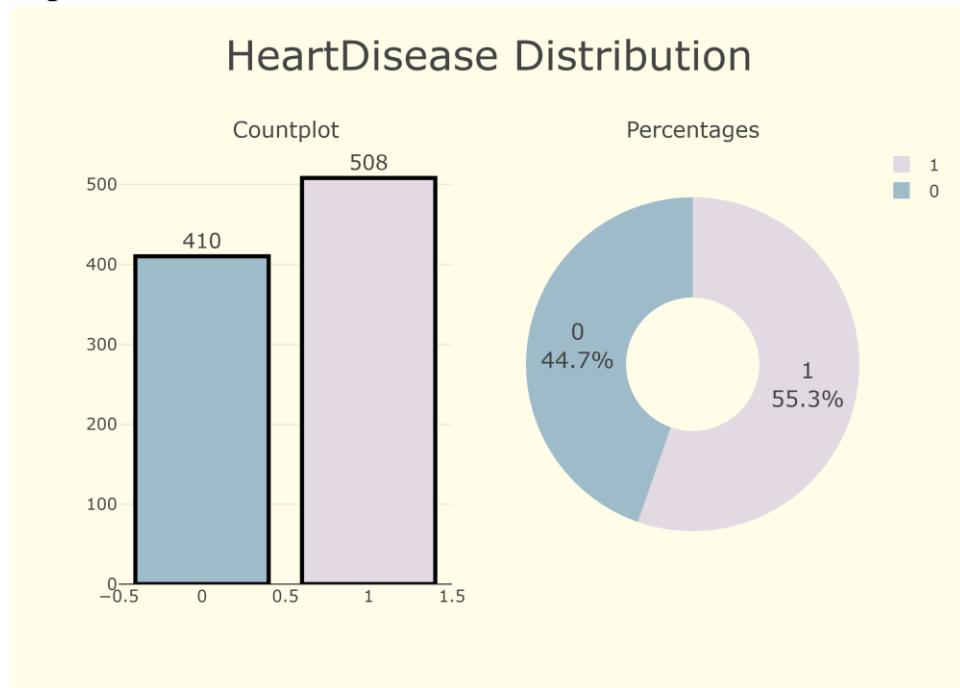




График №33. Столбиковая диаграмма распределения переменных: 'Sex', 'ChestPainType', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'ST_Slope' по категориям и классу 'HeartDisease'

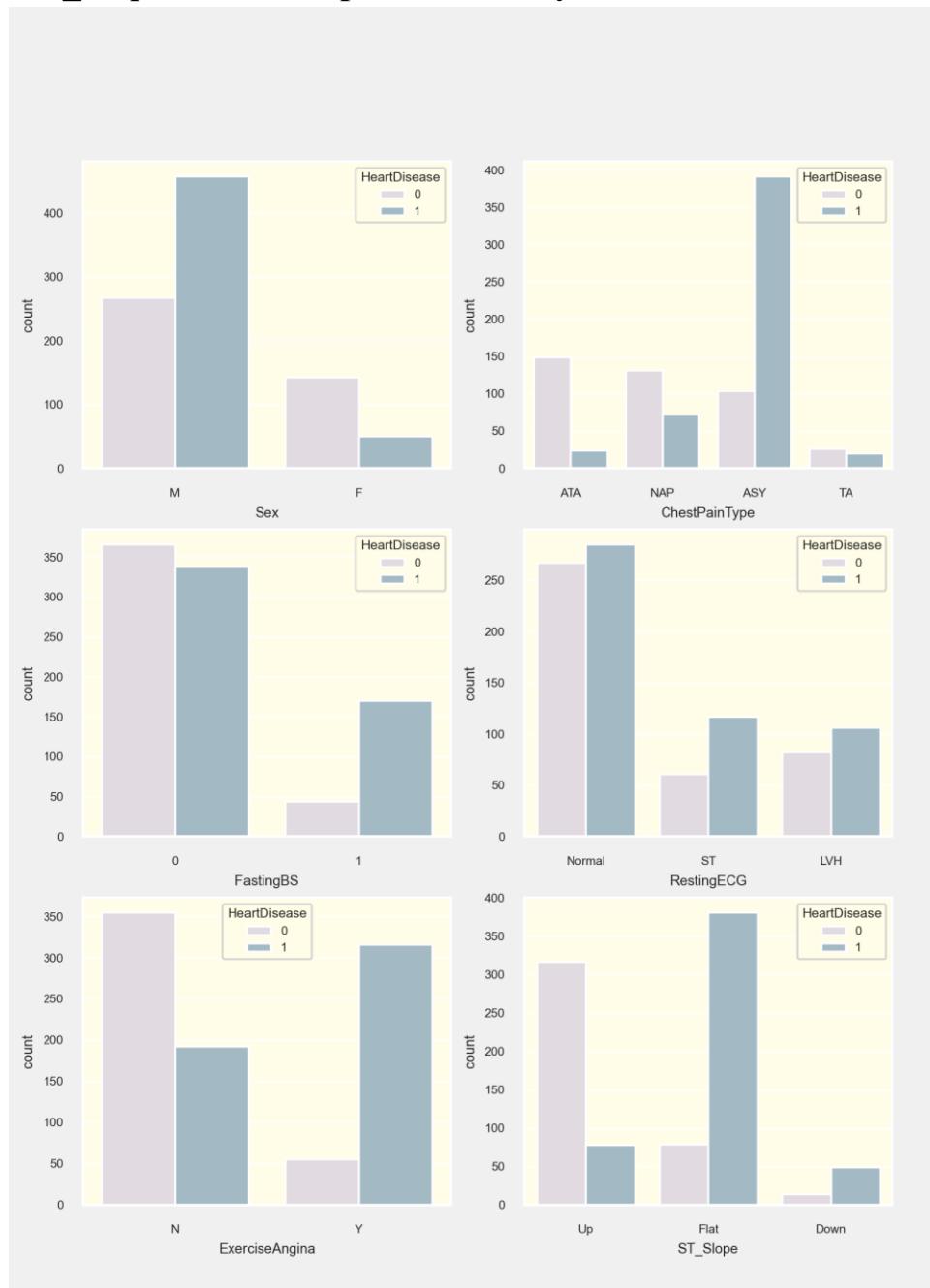




График №34. Столбиковая диаграмма распределения переменных: 'ChestPainType', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'ST_Slope' по категориям и классу 'Sex'

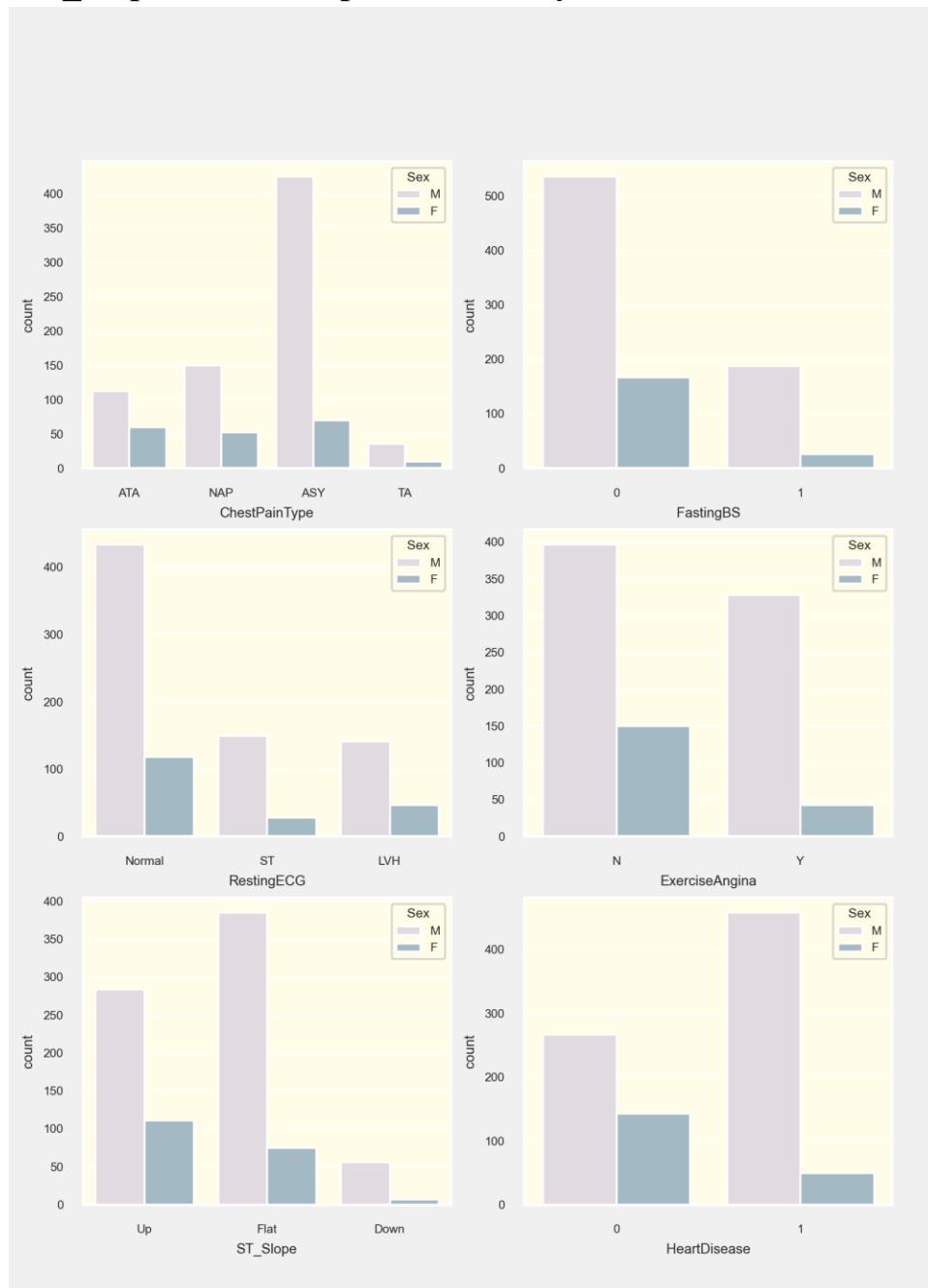




График №35. Столбиковая диаграмма распределения переменных: 'Sex','FastingBS','RestingECG','ExerciseAngina','ST_Slope', 'HeartDisease' по категориям и классу 'ChestPainType'

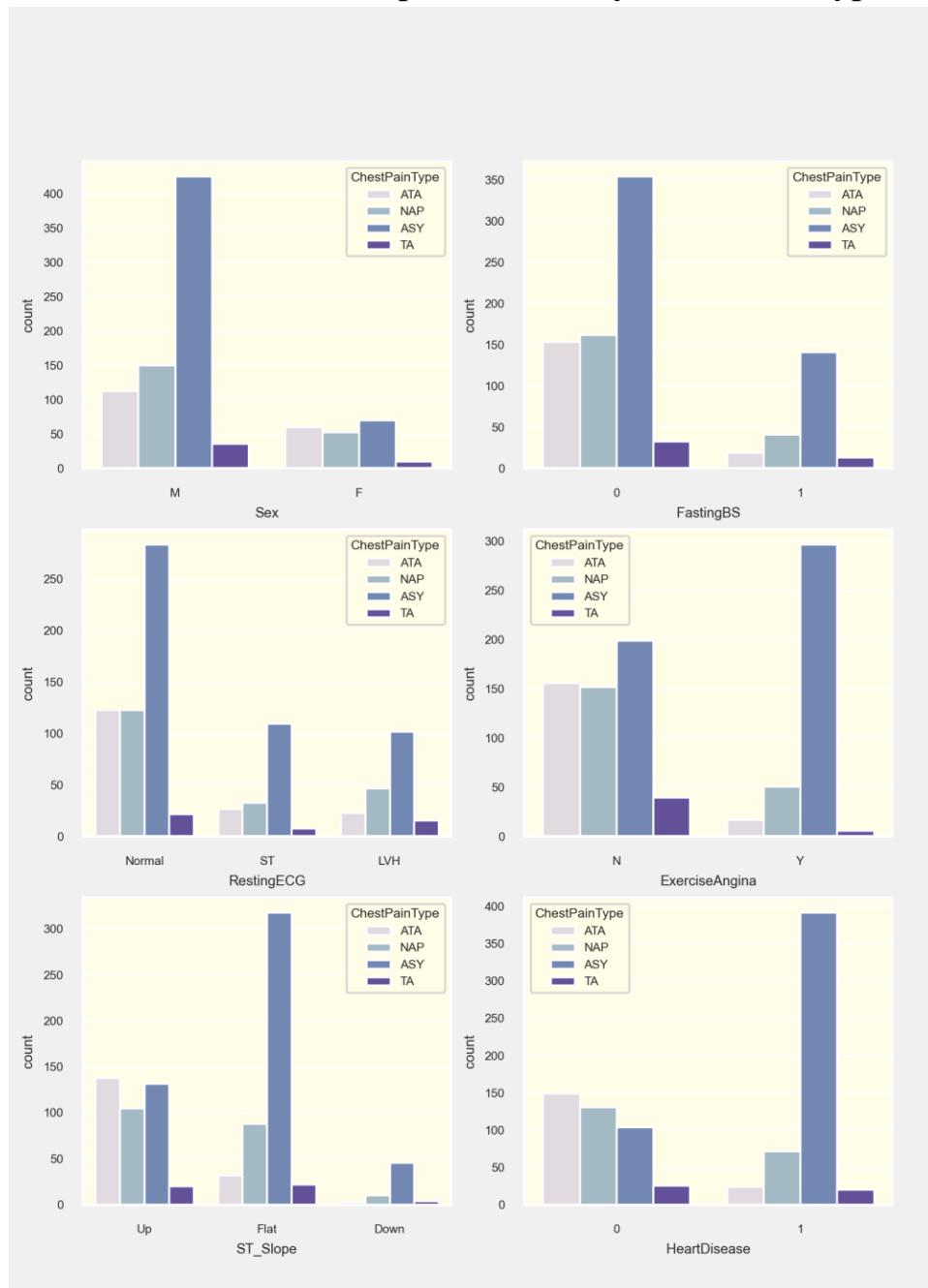




График №36. Комбинированная (столбиковая и секторная) диаграмма распределения переменной пол (Sex)

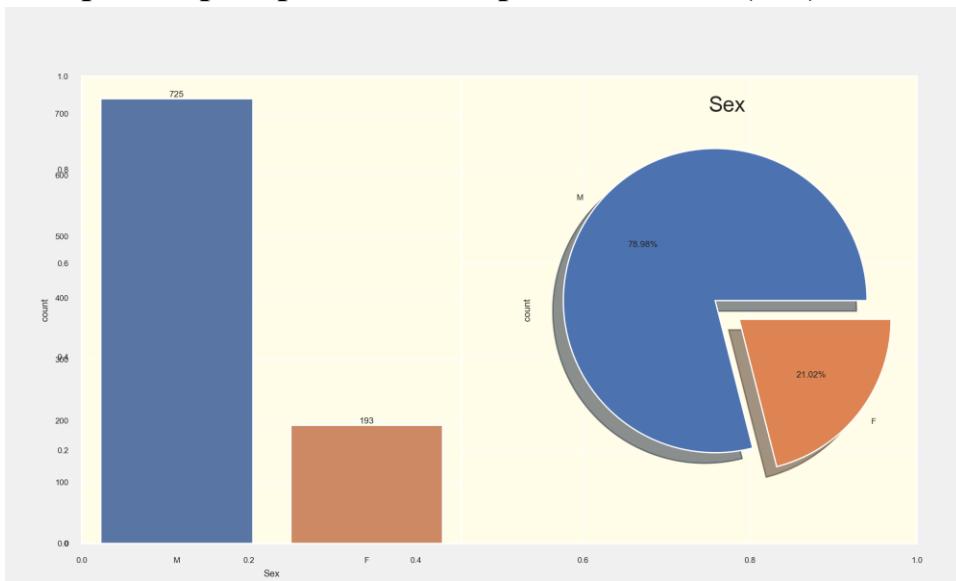




График №37. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ChestPainType

EDA на

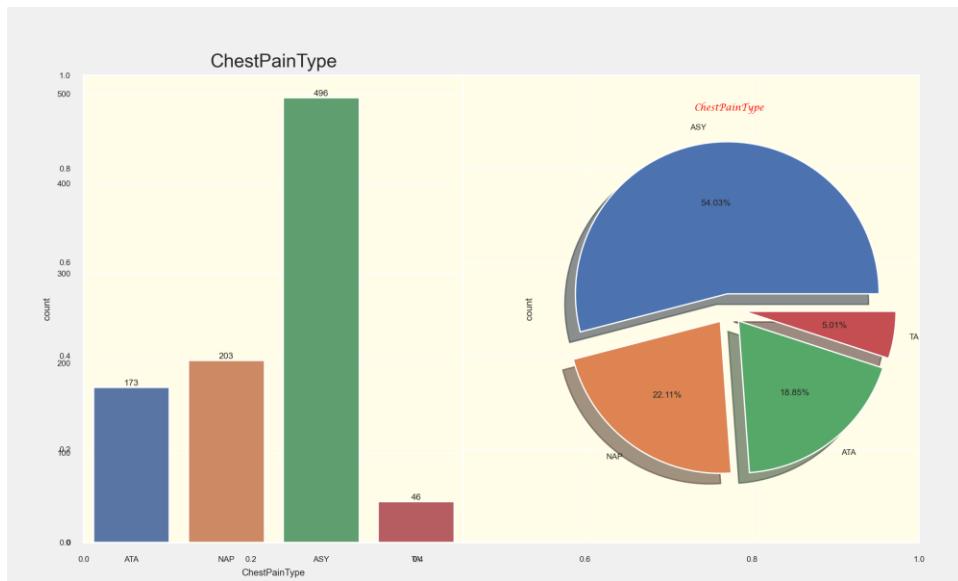




График №38. Комбинированная (столбиковая и секторная) диаграмма распределения переменной RestingECG

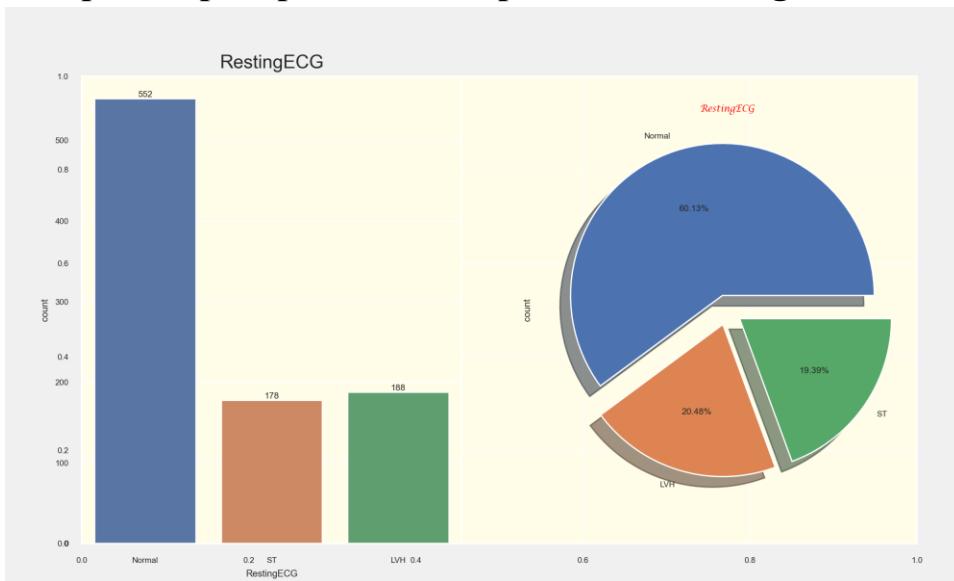




График №39. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ExerciseAngina

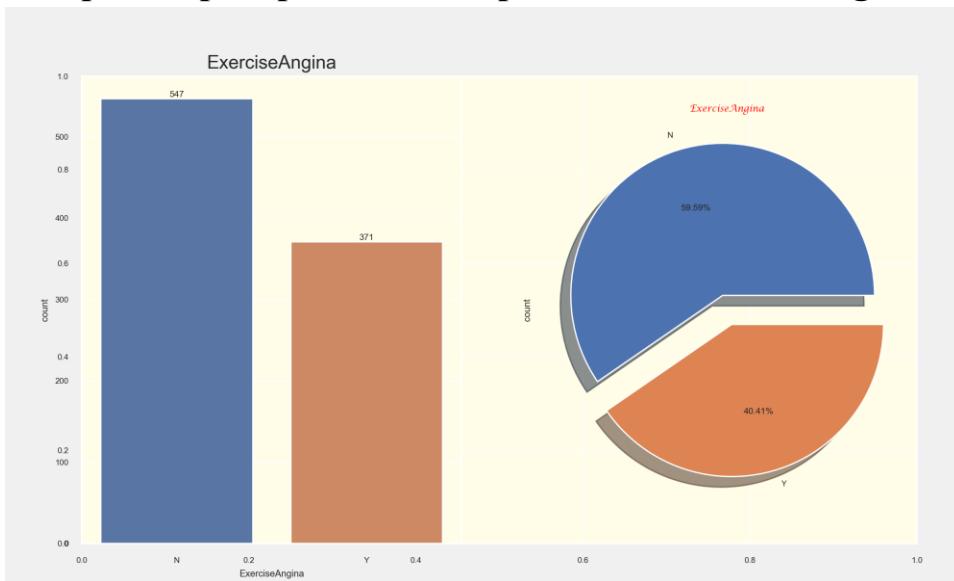




График №40. Комбинированная (столбиковая и секторная) диаграмма распределения переменной ST_Slope

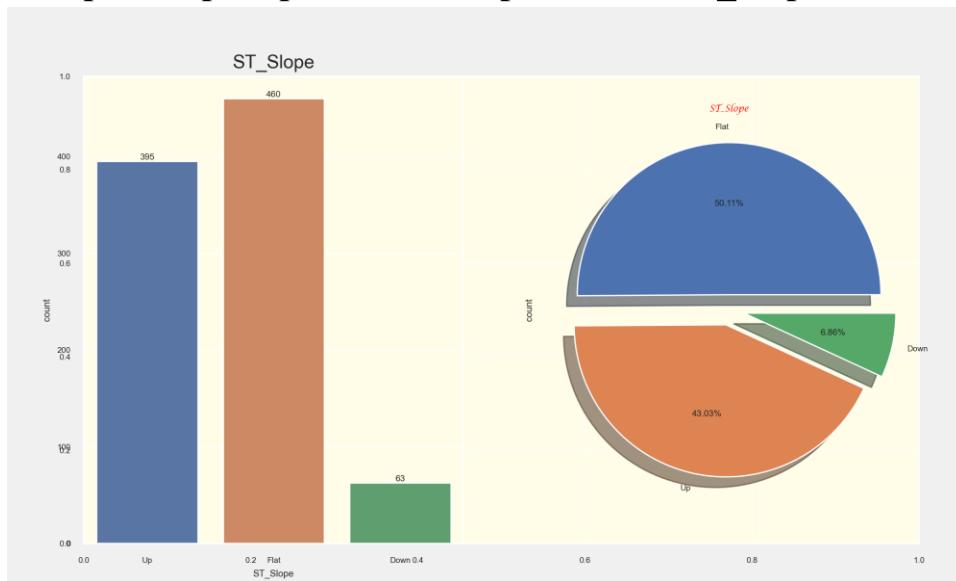




График №41. Комбинированная (столбиковая и секторная) диаграмма распределения переменной Cholesterol_Category

EDA на наборе данных показала нам, как каждая переменная связана с переменной отклика и как мы можем сделать нашу модель эффективной, используя различные методы EDA. Визуализации данных поднимают наше понимание набора данных на более высокий уровень, позволяя нам делать выводы.

Интеграция классификатора дерева принятия решений в наш анализ расширяет прогностические возможности нашей модели. В этом блоге мы не только изучили набор данных с помощью методов EDA, но и сделали еще один шаг вперед, внедрив модель машинного обучения. Такой целостный подход позволяет нам использовать сильные стороны как статистического анализа, так и прогнозного моделирования, способствуя более глубокому пониманию сложной динамики, связанной с экстремальными погодными явлениями.

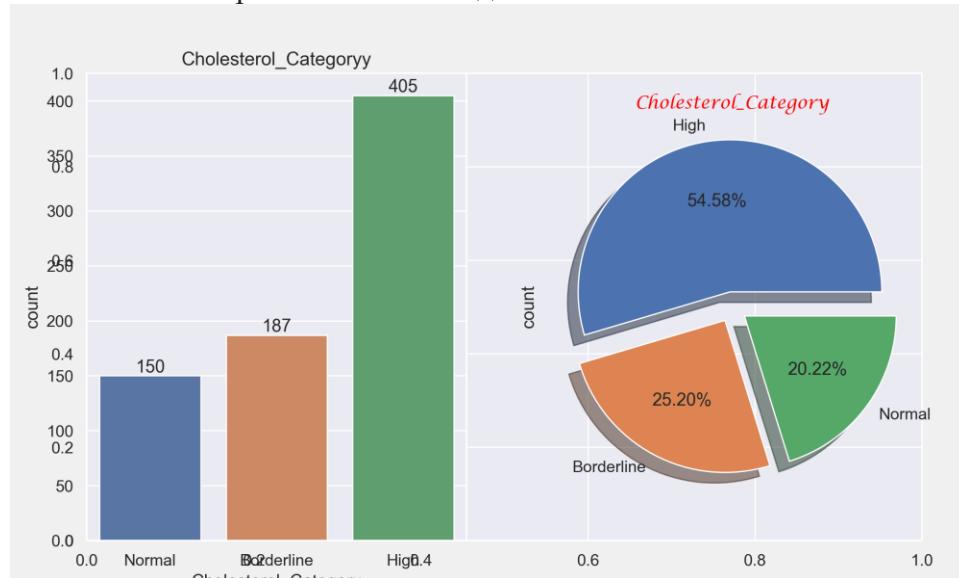




График №42. Комбинированная (столбиковая и секторная) диаграмма распределения переменной RestingBP_Category

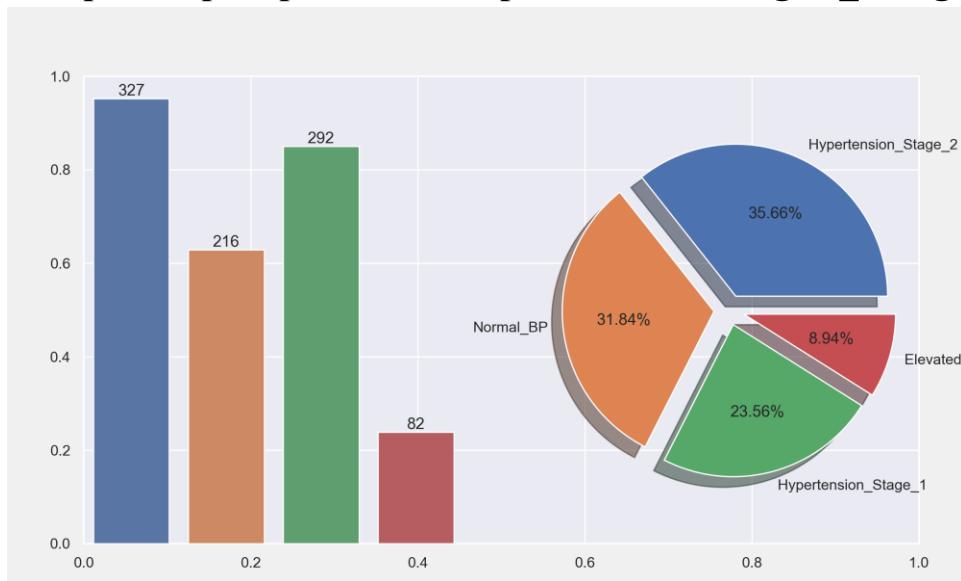




График №43. Двойная секторная диаграмма (Sunburst, 6 субграфиков) распределения пар переменных: ['ChestPainType', 'FastingBS'], ['ST_Slope', 'RestingECG'], ['ExerciseAngina', 'ChestPainType'] по классу пол (Sex)

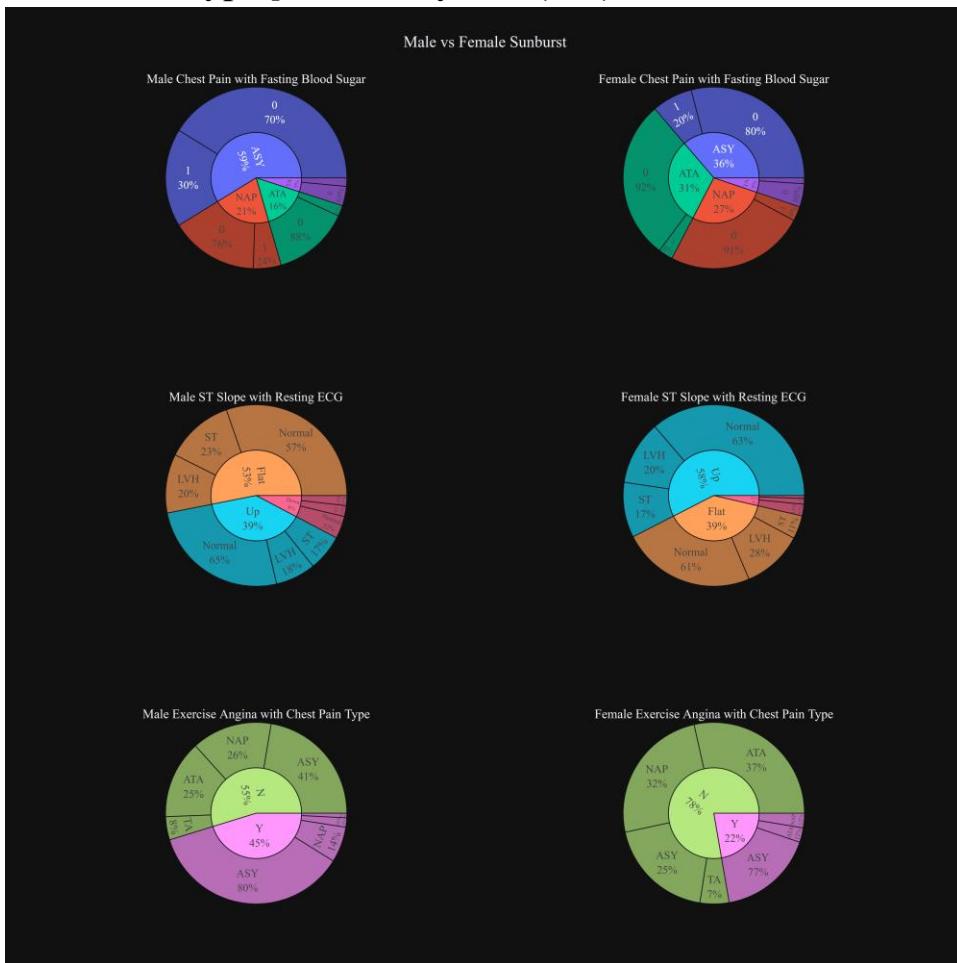




График №44. Столбиковая диаграмма распределения переменных: RestingECG, ChestPainType (2 субграфика на одной панели) по классу пол (Sex)

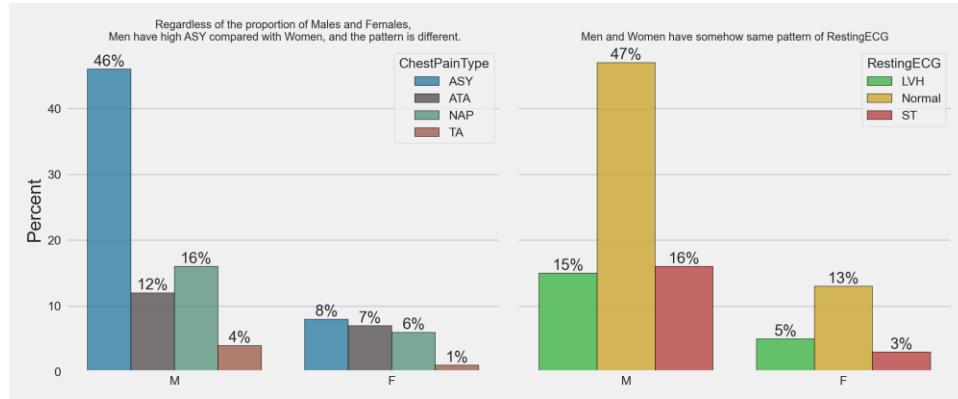
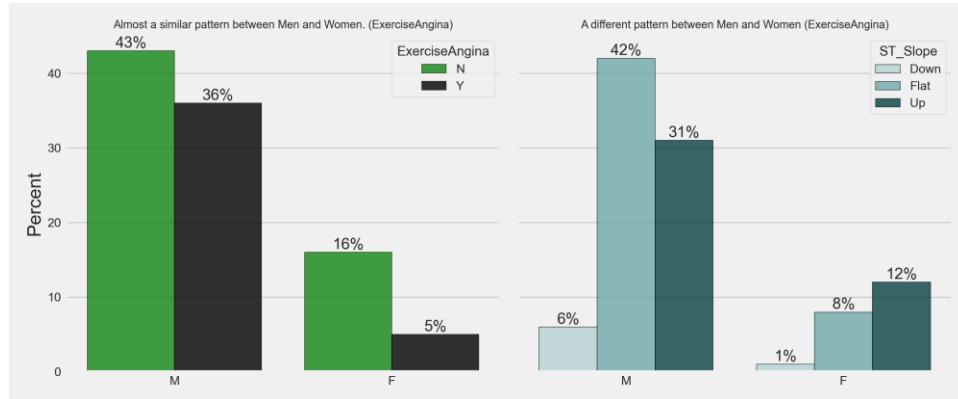




График №45. Столбиковая диаграмма распределения переменных: ST_Slope, ExerciseAngina (2 субграфика на одной панели) по классу пол (Sex)





Автоматизированная Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

Заключение (EDA)





Исходные данные и их организация

Основные этапы предлагаемой нами модели раннего прогнозирования и диагностики основных неблагоприятных сердечно-сосудистых событий показаны в Рис 1. Первым этапом предлагаемой модели является предварительная обработка данных набора данных KAMIR-NIH. На этапе предварительной обработки мы прошли через метод выбора признаков [25], в котором мы удалили неважные признаки из исходного набора данных и использовали наиболее важные признаки в качестве основного вклада в эту модель прогнозирования. Мы применили однократное горячее кодирование и кодирование меток [26] к выбранным признакам и подготовили наш предварительно обработанный набор данных для реализации модели. Предварительно обработанные данные разбиваются на обучающий набор данных (70 %) и проверочный набор данных (30 %) для обучения и тестирования модели соответственно. Вторым шагом предлагаемой нами модели является обучение модели прогнозирования на основе машинного обучения с использованием предварительно обработанного набора данных. На этом этапе обучения мы применили три различные модели машинного обучения в качестве моделей прогнозирования, например, случайный лес, дополнительное дерево и машину градиентного бустинга, и объединили их для создания ансамблевой модели для наилучшего прогнозирования и диагностики основных неблагоприятных сердечно-сосудистых событий. В предложенном нами классификаторе ансамбля мягкого голосования мы использовали алгоритмы машинного обучения случайного леса, дополнительного дерева и градиентного бустинга в качестве базовых классификаторов и скорректировали гиперпараметры с помощью алгоритма поиска по сетке для обучения этой модели, а затем оценили с помощью 5-кратной стратифицированной перекрестной проверки. Для обучения предложенной модели мы скорректировали веса этих классификаторов, т.к. этот классификатор голосования показал наилучшие результаты по удельному значению веса. Кроме того, мы использовали мягкое голосование для нашей модели. Мы настроили допуск, проверочную фракцию, вес и другие гиперпараметры в предложенной нами модели. Настройка гиперпараметров проиллюстрирована в Разделе 4.4. После обучения нашей ансамблевой модели на основе машинного обучения был применен тестовый набор данных (30%) для проверки производительности разработанной нами модели. После оценки модели на тестовых данных были извлечены лучшие значения гиперпараметров и доработана наилучшая модель прогнозирования путем корректировки гиперпараметров. Наконец, лучшие результаты модели прогнозирования будут извлечены и сравнены с результатами других моделей машинного обучения.

Для экспериментальной работы мы выбрали три алгоритма машинного обучения, названные случайным лесом [27, 28], дополнительным деревом [29] и машиной градиентного бустинга [30], и на основе этих трех базовых моделей разработали классификатор ансамбля мягкого голосования. По сравнению с другими алгоритмами машинного обучения, точность этих алгоритмов была сравнительно высокой, и это были лучше прогнозируемые модели для раннего прогнозирования и диагностики острого коронарного синдрома.

Разработанный нами классификатор ансамбля мягкого голосования представляет собой комбинацию нескольких классификаторов, в которых решения принимаются на основе отдельных решений, которые объединяются на основе значений вероятности, чтобы указать, что данные принадлежат к определенному классу. В ансамбле мягкого голосования прогнозы взвешиваются на основе важности классификатора и объединяются для получения суммы взвешенных вероятностей. Целевая метка с наибольшей суммой взвешенных вероятностей выбирается потому, что она имеет наибольшее значение для голосования (Рис 2). Пользовательские весовые коэффициенты также могут использоваться для вычисления средневзвешенного значения, чтобы придать большую важность и вовлеченность какой-либо конкретной модели обучения (базового классификатора). В отличие от жесткого голосования, мягкое голосование дает лучший результат и производительность, поскольку в нем используется усреднение вероятностей [31]. Классификатор ансамбля мягкого голосования скрывает слабость отдельных базовых классификаторов и превосходит общие результаты за счет агрегирования



Автоматизированная Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

нескольких моделей прогнозирования. Основная цель ансамблевых методов – уменьшить систематическую ошибку и дисперсию.



Предварительный анализ данных

Мы используем датасет по острому коронарному синдрому под названием Korea Acute Myocardial Infarction Registry (KAMIR-NIH) [11], который зарегистрирован в 52 больницах Кореи и содержит данные обо всех пациентах с ноября 2011 года по декабрь 2019 года. В своей исследовательской работе мы используем двухлетний набор данных, содержащий 551 различных атрибутов и медицинские карты 13 104 пациентов с двухлетним наблюдением после выписки из стационара. Однако существует ограничение на передачу этих данных, поскольку данные являются конфиденциальными и недоступны публично. Подробная информация о реестре размещена на сайте КАМИР (<http://www.kamir.or.kr>). В нашей выборке данных у нас есть вся основная медицинская информация о пациентах, такая как возраст, артериальное давление, частота сердечных сокращений, рост, вес, другие заболевания, а также предыдущая медицинская карта пациентов, страдающих каким-либо другим заболеванием или уже имеющих сердечную недостаточность, или какова степень тяжести состояния пациента. У нас также есть полные записи о приеме лекарств для пациентов с сердечными заболеваниями с двухлетним наблюдением. В данной работе основные неблагоприятные сердечно-сосудистые события (MACE) определяются как сердечная смерть (БК), несердечная смерть (НИЗ), инфаркт миокарда (ИМ), повторное чрескожное коронарное вмешательство (ре-ЧКВ) и аортокоронарное шунтирование (АКШ).

2.3 Извлечение данных

Извлечение данных — это процесс извлечения или извлечения данных из неструктурированных или полуструктурных источников данных для дальнейшей обработки данных для достижения требуемых результатов. В случае с набором данных KAMIR-NIH он находится в необработанном виде и содержит противоречивые, зашумленные и неполные данные. Он также содержит избыточность данных и выбросы. Чтобы решить эти проблемы, мы предварительно обрабатываем эти данные в понятном формате, чтобы получить больше полезной информации. Мы должны применять методы извлечения данных для извлечения и манипулирования важными признаками и записями из всего набора данных. Прежде всего, мы удалили атрибуты даты из набора данных KAMIR-NIH, поскольку эти атрибуты не влияют на раннюю диагностику и прогноз основных неблагоприятных сердечно-сосудистых событий. Во-вторых, из датасета были исключены все атрибуты, содержащие информацию о препаратах для пациентов, поскольку эти атрибуты не являются обязательными для требуемых результатов и содержат более 70% нулевых значений. В-третьих, все атрибуты, содержащие более 70% значений NULL, удаляются из набора данных. Удалив всю эту ненужную информацию из датасета, мы извлекли важные данные из датасета. Извлечение данных проиллюстрировано в Рис 3 в котором мы использовали набор данных KAMIR-NIH ($N = 13\,104$) и исключили всех пациентов, умерших в больнице во время госпитализации (Excluded $N = 504$). Мы также исключили пациентов, которые не проводили двухлетнее наблюдение (исключено $N = 1,411$). После исключения всех ненужных данных из набора данных KAMIR-NIH у нас были пациенты с острым коронарным синдромом, которые были живы в течение двухлетнего наблюдения после выписки из больницы ($N = 11\,189$). Затем этот набор данных был разделен на наборы данных «Инфаркт миокарда с подъемом сегмента ST» (STEMI) ($N = 5,389$) и «Инфаркт миокарда без подъема сегмента ST» (NST) ($N = 5,800$), а затем разделен на обучающие данные (70%) и данные тестирования (30%). Полные процессы извлечения данных проиллюстрированы в Рис 3.



Моделирование

Исследовательский анализ включает в себя изучение данных и поиск связей между переменными, которые ранее были неизвестны. Вот что вам нужно знать:

- EDA помогает обнаруживать взаимосвязи между показателями в данных, которые не являются доказательством существования корреляции, как обозначается фразой «Корреляция не подразумевает причинно-следственную связь».
- Это полезно для обнаружения новых связей и формирования гипотез. Он управляет планированием проектирования и сбором данных.

Пример исследовательского анализа

Изменение климата становится все более важной темой, поскольку глобальная температура постепенно повышается на протяжении многих лет. Один из примеров исследовательского анализа данных об изменении климата включает в себя повышение температуры за период с 1950 по 2020 год, а также рост человеческой деятельности и индустриализацию, чтобы найти взаимосвязи на основе данных. Например, вы можете увеличить количество заводов, автомобилей на дорогах и самолетов, чтобы увидеть, как это коррелирует с повышением температуры.

Исследовательский анализ исследует данные для поиска взаимосвязей между показателями без выявления причины. Это наиболее полезно при формулировании гипотез.

Для предварительной обработки набора данных KAMIR-NIH мы классифицировали все атрибутивные признаки по различным категориям, например, категориальные признаки, непрерывные признаки и дискретные признаки. Мы определили различные правила предварительной обработки для этих различных типов атрибутов. Для категориальных переменных мы применили кодирование меток [32], а также одно горячее кодирование [26] для предварительной обработки этих переменных. Для непрерывных атрибутов мы классифицировали набор данных по диапазонам, а затем применили кодировку меток для этих определенных подклассов. Для некоторых категориальных и непрерывных переменных, содержащих несколько значений, мы применили одно горячее кодирование, чтобы упростить управление значениями этих атрибутов. Одно горячее кодирование является одним из лучших решений для управления несколькими значениями и предварительной обработки тех атрибутов, которые содержат более одного параметра. В нашем наборе данных также есть атрибуты с двоичным значением. Для таких атрибутов, содержащих ровно два значения, мы преобразовали их в двоичную форму (0 и 1), обозначив 0 как No, 1 как Yes.

В нашем датасете было много атрибутов, которые не нужны для применения различных алгоритмов. Чтобы сделать наши данные более конкретными и безошибочными, мы удалили эти атрибуты из нашего набора данных. Например, для некоторых атрибутов типа даты, содержащих дату и время, нет необходимости использовать эти атрибуты в обучающих моделях. Поэтому мы удалили эти атрибуты. В случае с нашим набором данных некоторые атрибуты отсутствовали в нашем наборе данных, и они очень важны для моделей прогнозирования. На основе нашего текущего набора данных мы вывели эти атрибуты, используя другие атрибуты, и классифицировали их для использования этих атрибутов. Например, артериальное давление (АД), индекс массы тела (ИМТ), общий холестерин и частота сердечных сокращений (ЧСС) отсутствовали в наборе данных, но они были необходимы для моделей прогнозирования. Мы вывели эти атрибуты из других атрибутов и классифицировали их соответствующим образом.

Мы также следовали рекомендациям Корейского общества гипертонии [33, 34] по категоризации артериального давления, а затем применяли кодирование меток для преобразования данных. ИМТ рассчитывается как выражение кг/м² от веса (кг) и роста пациента (м), а затем применили корейские стандарты [35] для категоризации значений ИМТ. Мы применили Национальные рекомендации по лечению холестерина [36] для классификации липопротеинов низкой плотности (ЛПНП), липопротеинов высокой плотности (ЛПВП) и общего холестерина у корейских пациентов. Предпочтительный уровень триглицеридов — менее 150 мг/дл (1,7 ммоль/л), повышенный пограничный — 150–199 мг/дл (1,7–2,2 ммоль/л), повышенный уровень — 200–499 мг/дл (2,3–5,6 ммоль/л), а очень высокий уровень триглицеридов — 500 мг/дл



(5,6 ммоль/л) или выше [37]. Вне отечественных стандартов, согласно критериям ВОЗ, окружность бедер и талии является индикатором для диагностики абдоминального ожирения [38] и указывает на абдоминальное ожирение при $WHR > 0,9$ для мужчин и $> 0,85$ для женщин. С-реактивный белок (высокочувствительный СРБ, hs-СРБ) использовался в качестве предиктора сердечно-сосудистого риска у здоровых взрослых [39]. Люди с высокими значениями hs-CRP имеют высокий риск развития острого коронарного синдрома, а люди с низкими значениями имеют низкий риск. У людей с более высокими результатами hs-CRP в верхнем диапазоне нормы риск сердечного приступа примерно в 1,5-4 раза выше, чем у людей с более низкой частотой. Корейское общество диагностической радиологии использует те же критерии, что и Американская кардиологическая ассоциация [39, 40] и Центры по контролю и профилактике заболеваний США. Эти значения являются частью общего процесса оценки острого коронарного синдрома. Высокий уровень сахара в крови означает в основном диабет. Тем не менее, многие заболевания и системные состояния, кроме диабета, могут повышать уровень сахара в крови. Ниже приведена краткая информация о значении каждого результата теста. Это резюме основано на данных Американской диабетической ассоциации и классифицируется по нормальному уровню глюкозы натощак, преддиабетической стадии и диабету [38]. Концентрация креатинина в сыворотке крови повышается при нарушении функции почек. Аномальный диапазон для мужчин составляет $> 1,2$ мг/дл и $> 1,0$ мг/дл для женщин [41].

В нашем наборе данных мы также разобрались с отсутствующими значениями. В медицинском наборе данных очень сложно работать с пропущенными значениями, особенно когда данные очень чувствительны. Неправильное и неадекватное обращение с пропущенными значениями приведет к низкому прогнозированию фактора риска и наоборот [42]. Когда мы применяли алгоритмы машинного обучения для прогнозирования риска, ранней диагностики и прогнозирования острого коронарного синдрома, мы использовали различные методы импутации для нормализации данных, например, импутирование среднего значения [43, 44] и вменение k-ближайших соседей (k-NN) [43].

В ходе предварительной обработки данных мы выяснили, что некоторые пациенты перенесли множественные сердечные события. Таким образом, мы классифицировали пациентов, перенесших несколько сердечных событий, в одно сердечное событие в зависимости от тяжести, осложнений и эффективности этого события. Например, пациент уже сделал АКШ, а потом умер из-за сердечно-сосудистых заболеваний, мы внесли этого пациента в КК, а не в АКШ.

Модель: Linear Regression

Linear Regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression. Reference Wikipedia.

Таблица классификации

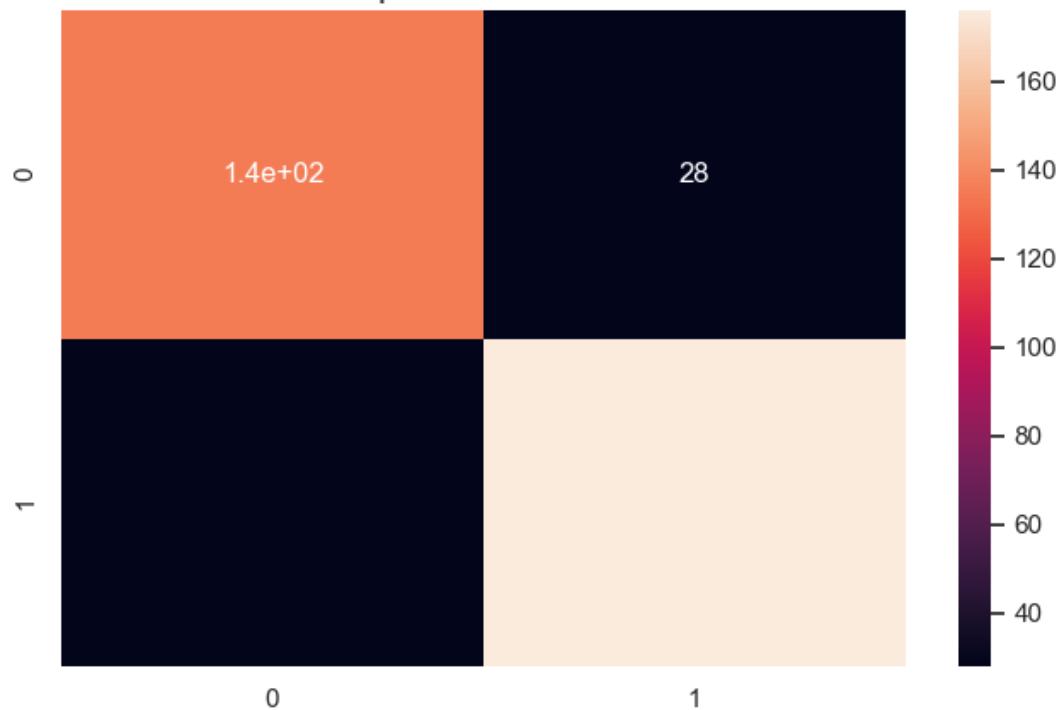
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.849	0.833	0.841	162.0	0.913
class 1	0.87	0.883	0.877	205.0	0.913
accuracy	0.861	0.861	0.861	0.861	0.913
macro avg	0.86	0.858	0.859	367.0	0.913
weighted avg	0.861	0.861	0.861	367.0	0.913

© Dr. Alexander Wagner. Все права охраняются законом

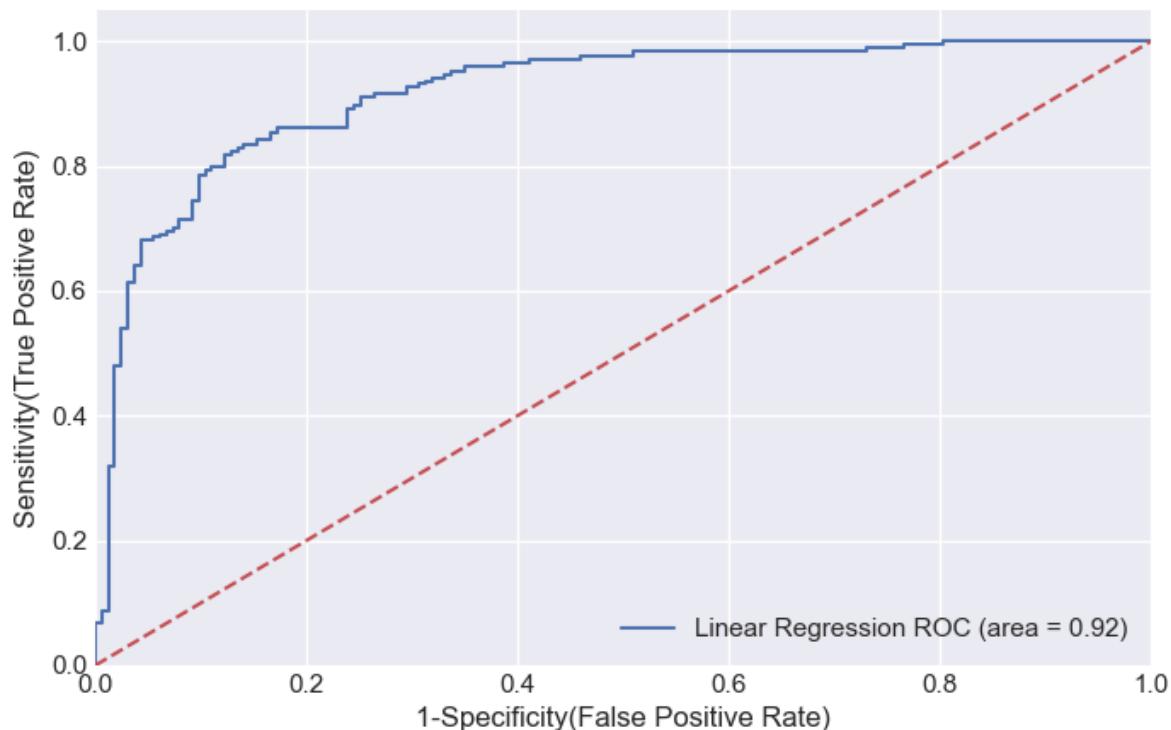
Confusion Matrix



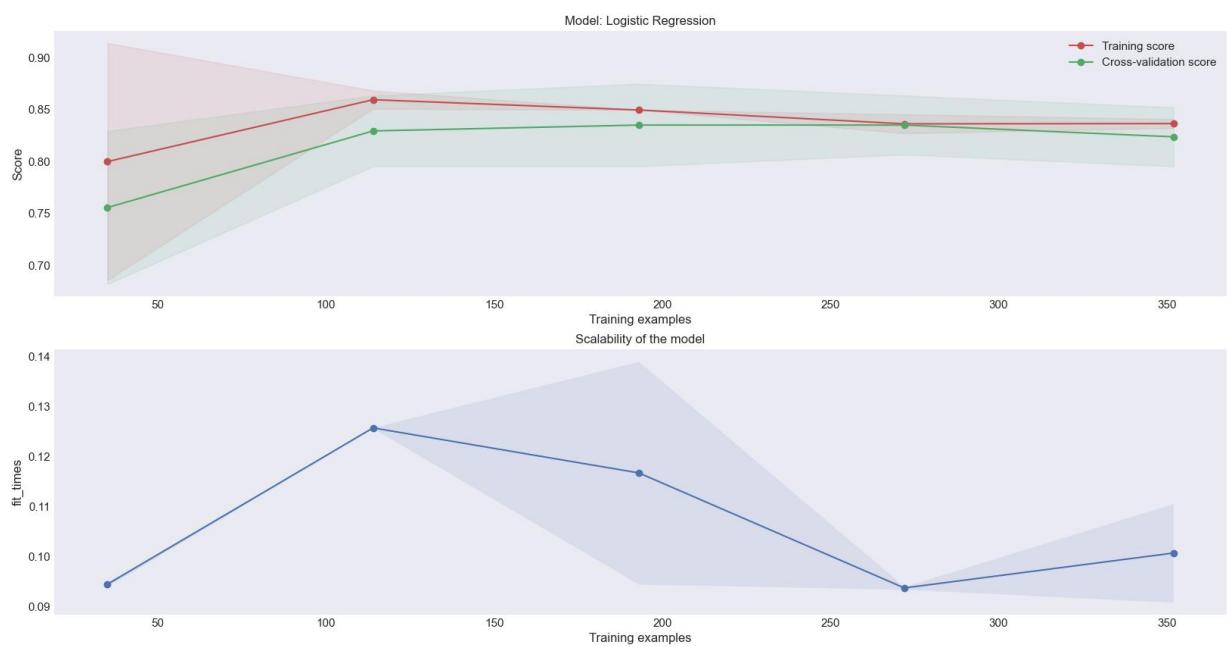
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: Logistic Regression

Logistic Regression is a useful model to run early in the workflow. Logistic regression measures the relationship between the categorical dependent variable (feature) and one or more independent variables (features) by estimating probabilities using a logistic function, which is the cumulative logistic distribution. Reference Wikipedia.

Таблица классификации

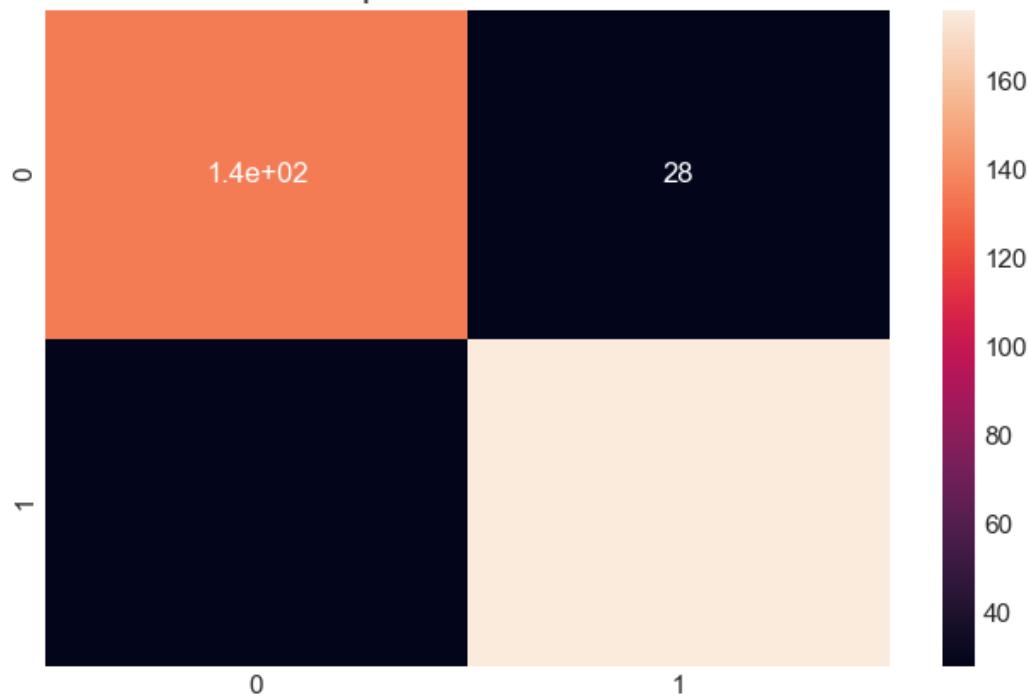
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.849	0.833	0.841	162.0	0.913
class 1	0.87	0.883	0.877	205.0	0.913
accuracy	0.861	0.861	0.861	0.861	0.913
macro avg	0.86	0.858	0.859	367.0	0.913
weighted avg	0.861	0.861	0.861	367.0	0.913

© Dr. Alexander Wagner. Все права охраняются законом

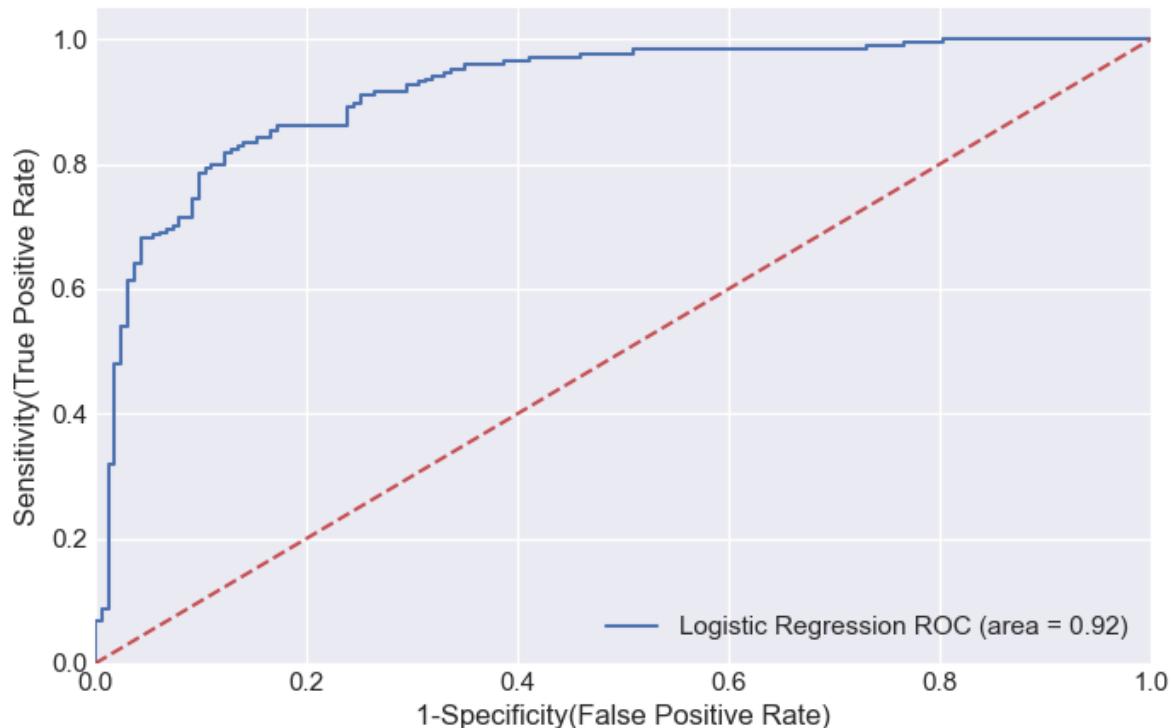
Confusion Matrix



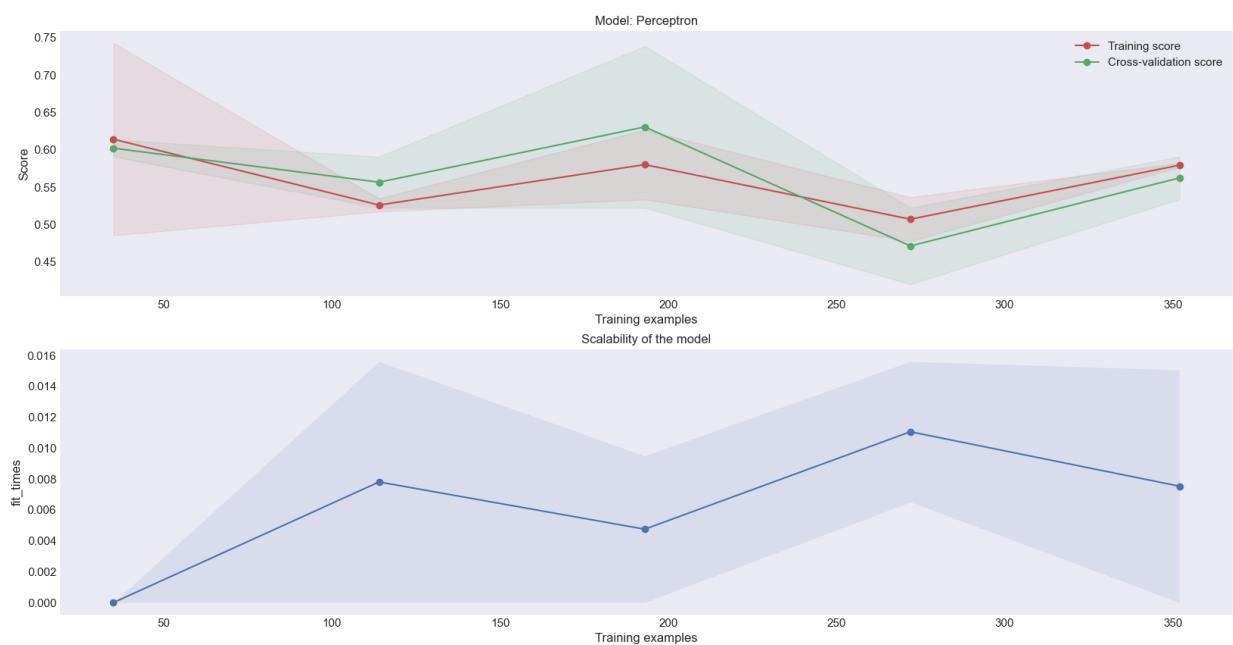
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: Perceptron

Thanks to <https://www.kaggle.com/startupsci/titanic-data-science-solutions> The Perceptron is an algorithm for supervised learning of binary classifiers (functions that can decide whether an input, represented by a vector of numbers, belongs to some specific class or not). It is a type of linear classifier, i.e. a classification algorithm that makes its predictions based on a linear predictor function combining a set of weights with the feature vector. The algorithm allows for online learning, in that it processes elements in the training set one at a time. Reference Wikipedia.

Таблица классификации

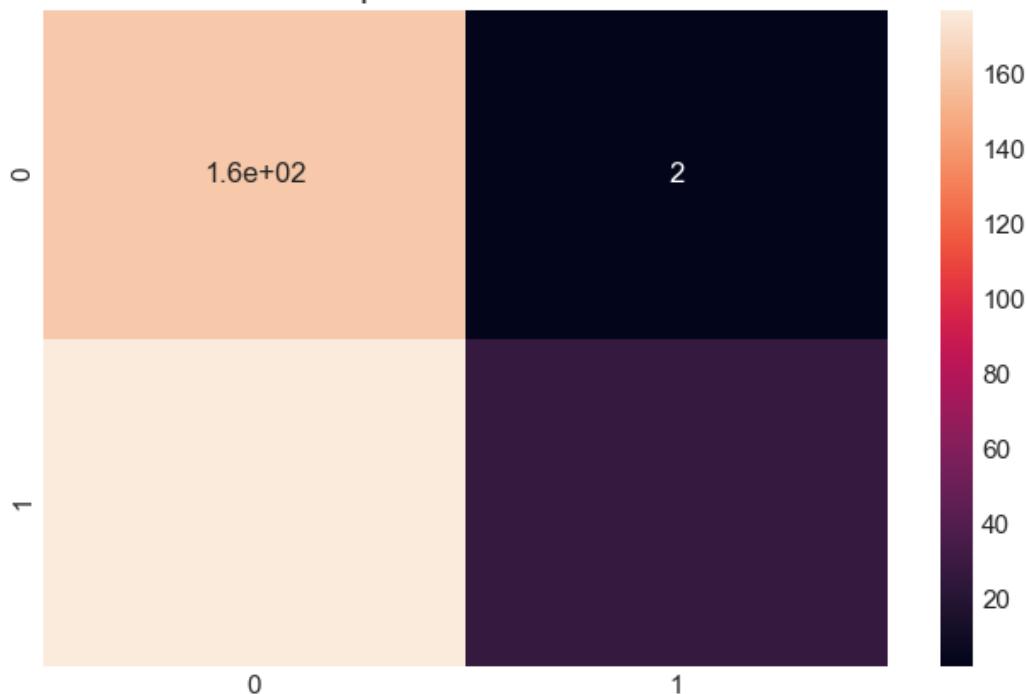
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.2	0.006	0.012	162.0	0.5
class 1	0.555	0.98	0.709	205.0	0.5
accuracy	0.55	0.55	0.55	0.55	0.5
macro avg	0.378	0.493	0.36	367.0	0.5
weighted avg	0.398	0.55	0.401	367.0	0.5

© Dr. Alexander Wagner. Все права охраняются законом

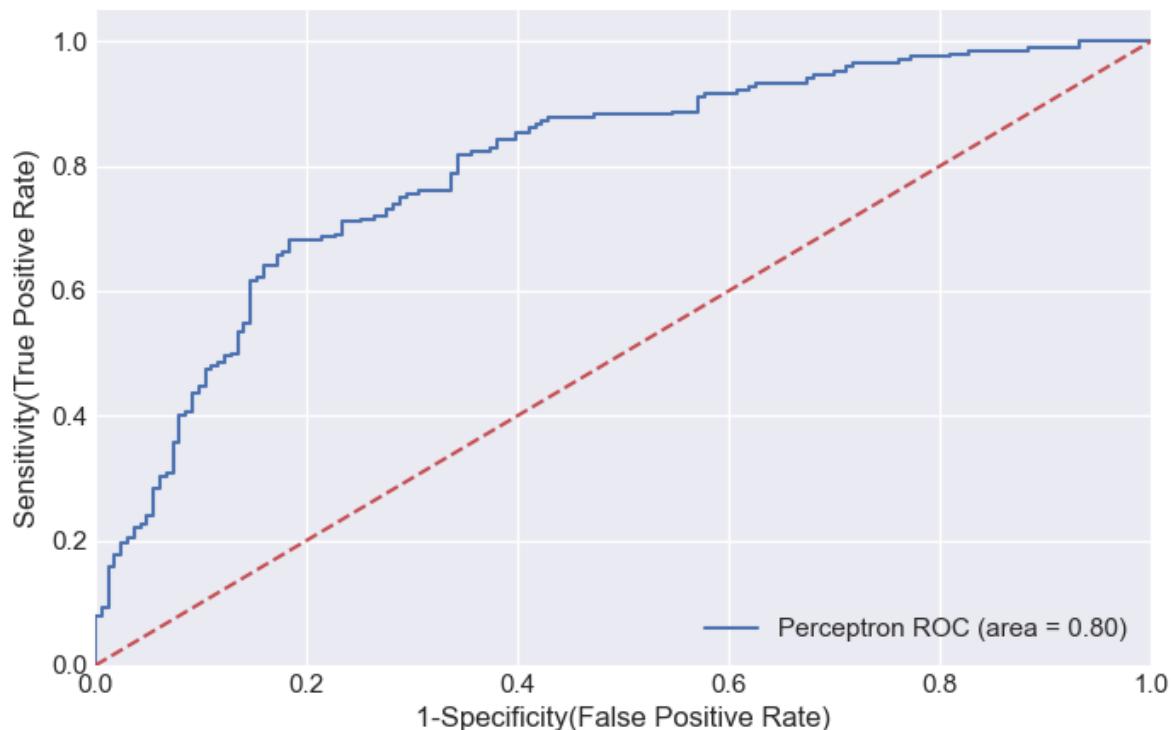
Confusion Matrix



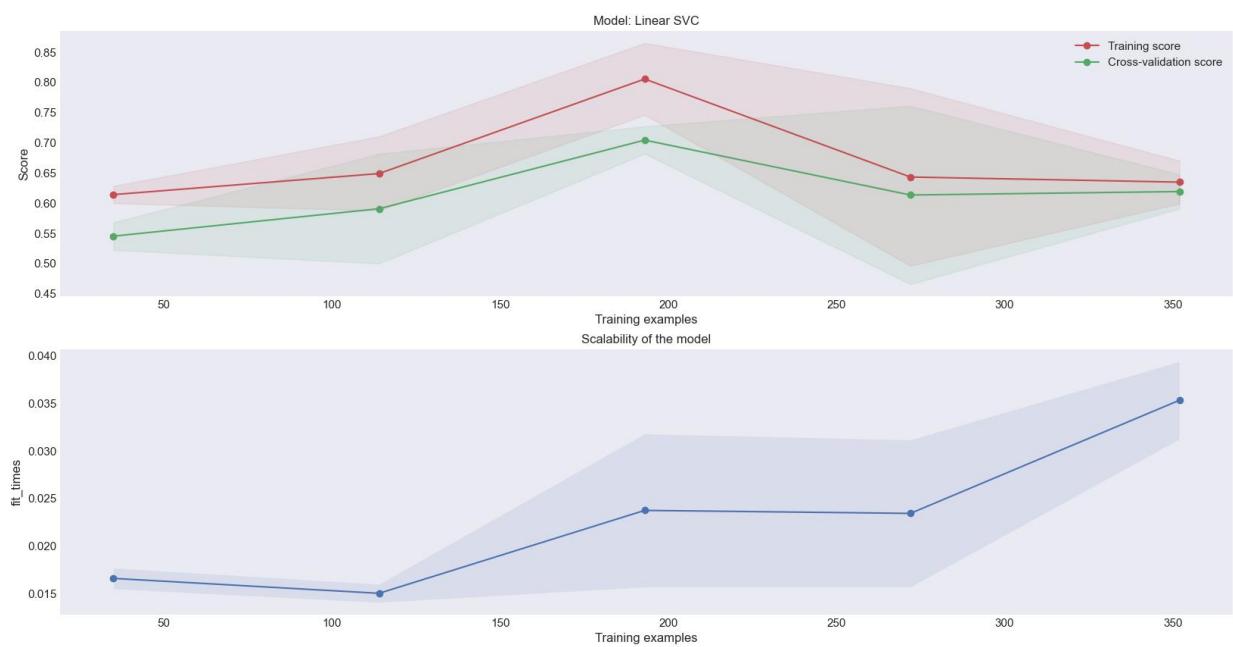
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: Linear SVC

Linear SVC is a similar to SVM method. Its also builds on kernel functions but is appropriate for unsupervised learning. Reference Wikipedia.

Таблица классификации

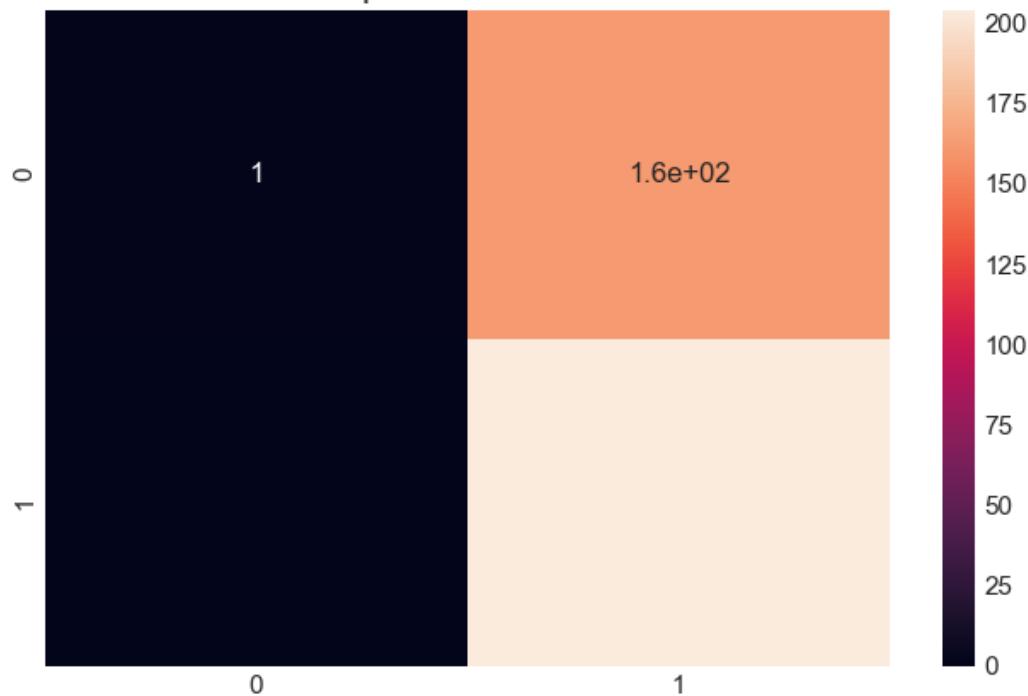
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.957	0.549	0.698	162.0	0.911
class 1	0.734	0.98	0.839	205.0	0.911
accuracy	0.79	0.79	0.79	0.79	0.911
macro avg	0.845	0.765	0.769	367.0	0.911
weighted avg	0.832	0.79	0.777	367.0	0.911

© Dr. Alexander Wagner. Все права охраняются законом

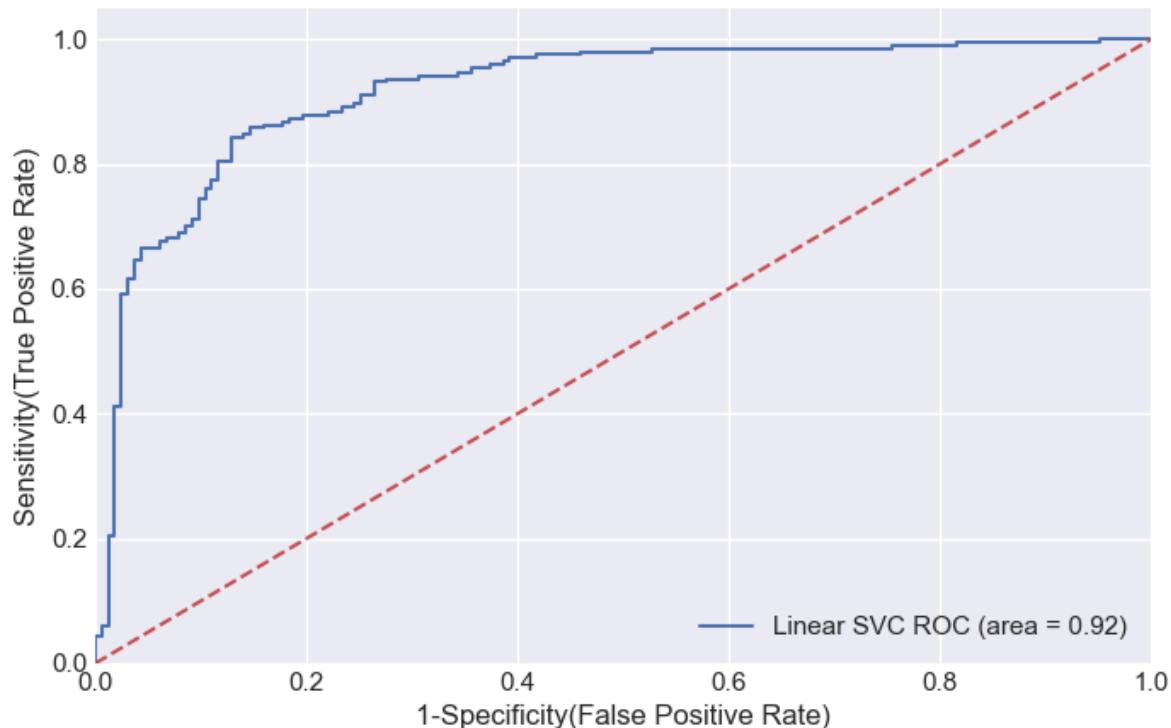
Confusion Matrix



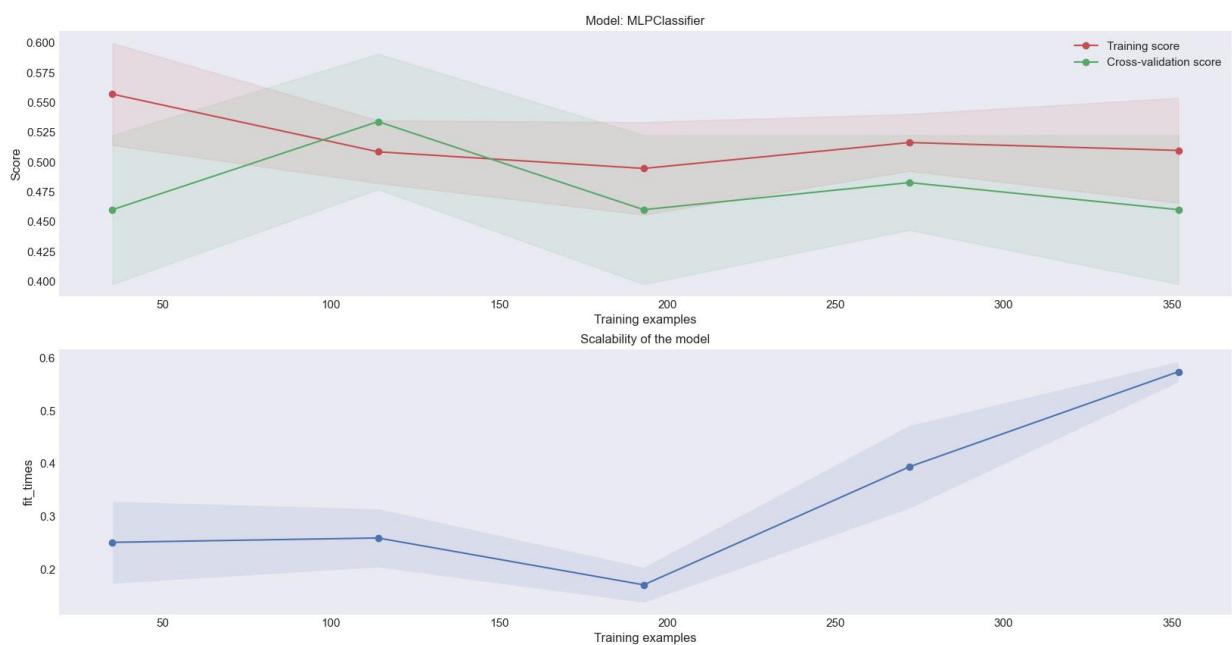
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: MLPClassifier

The MLPClassifier optimizes the squared-loss using LBFGS or stochastic gradient descent by the Multi-layer Perceptron regressor. Reference Sklearn documentation.

Thanks to: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html#sklearn.neural_network.MLPRegressor <https://stackoverflow.com/questions/44803596/scikit-learn-mlpregressor-performance-cap>

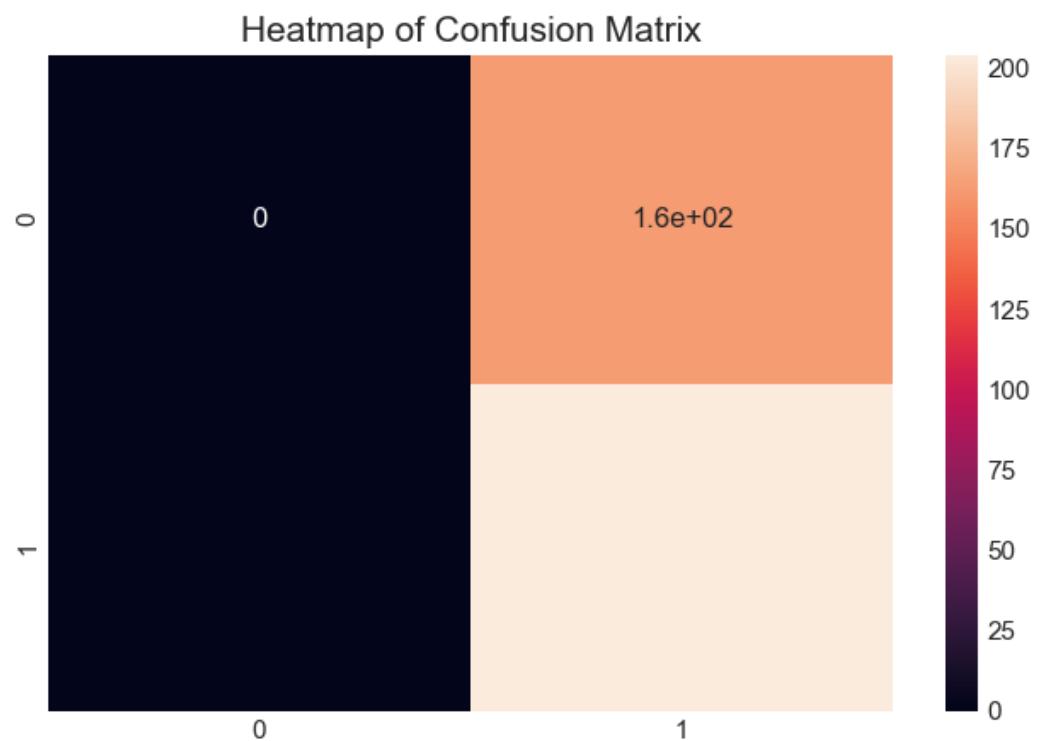
Linear SVC is a similar to SVM method. Its also builds on kernel functions but is appropriate for unsupervised learning. Reference Wikipedia.

Таблица классификации

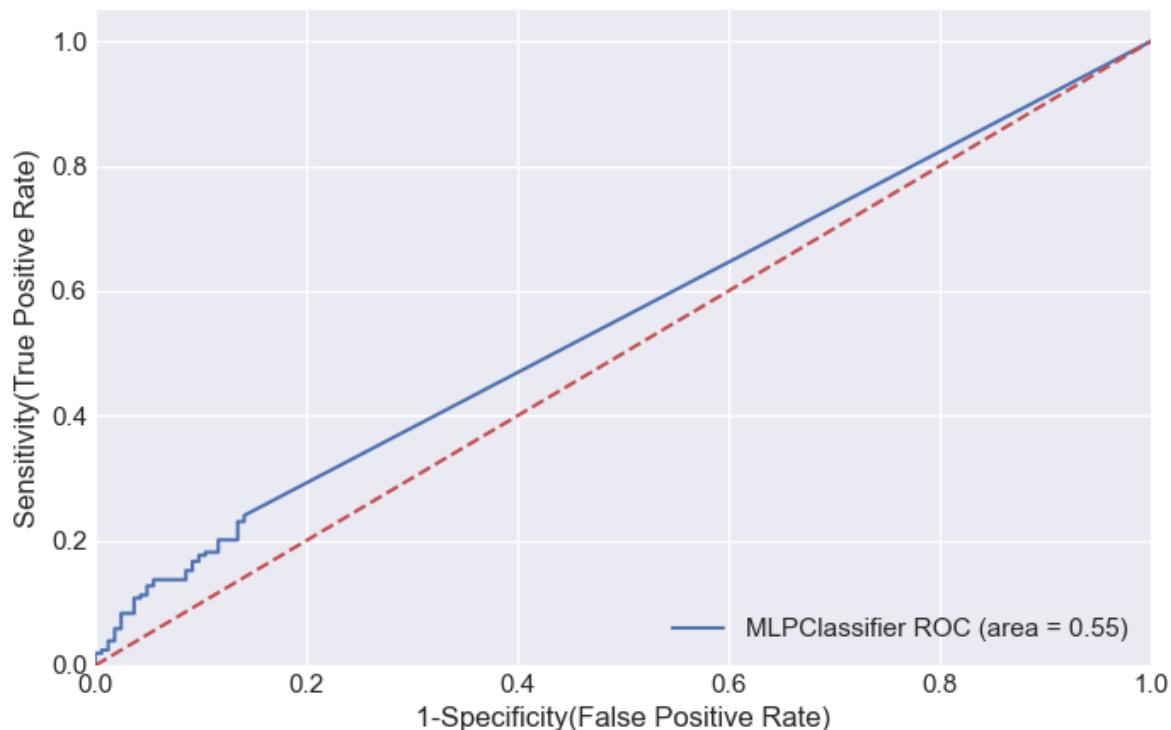
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.656	0.66	0.658	162.0	0.763
class 1	0.73	0.727	0.729	205.0	0.763
accuracy	0.698	0.698	0.698	0.698	0.763
macro avg	0.693	0.694	0.694	367.0	0.763
weighted avg	0.698	0.698	0.698	367.0	0.763

© Dr. Alexander Wagner. Все права охраняются законом

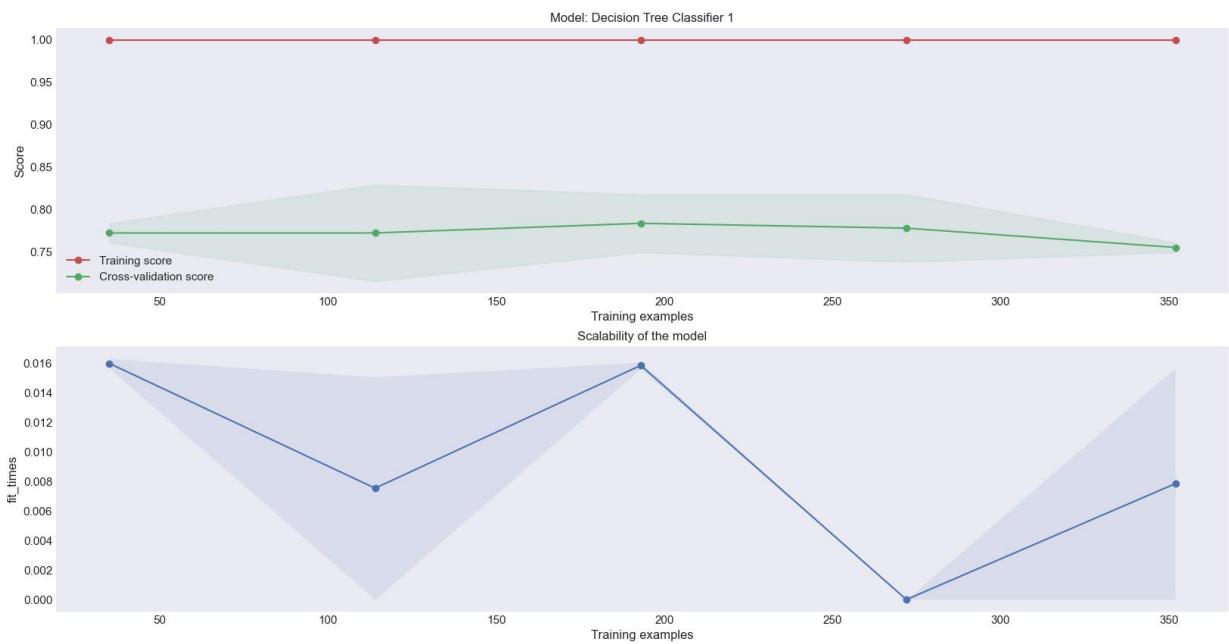
Confusion Matrix



ROC Curve



Score plot



Модель: Decision Tree Classifier 1

Таблица классификации

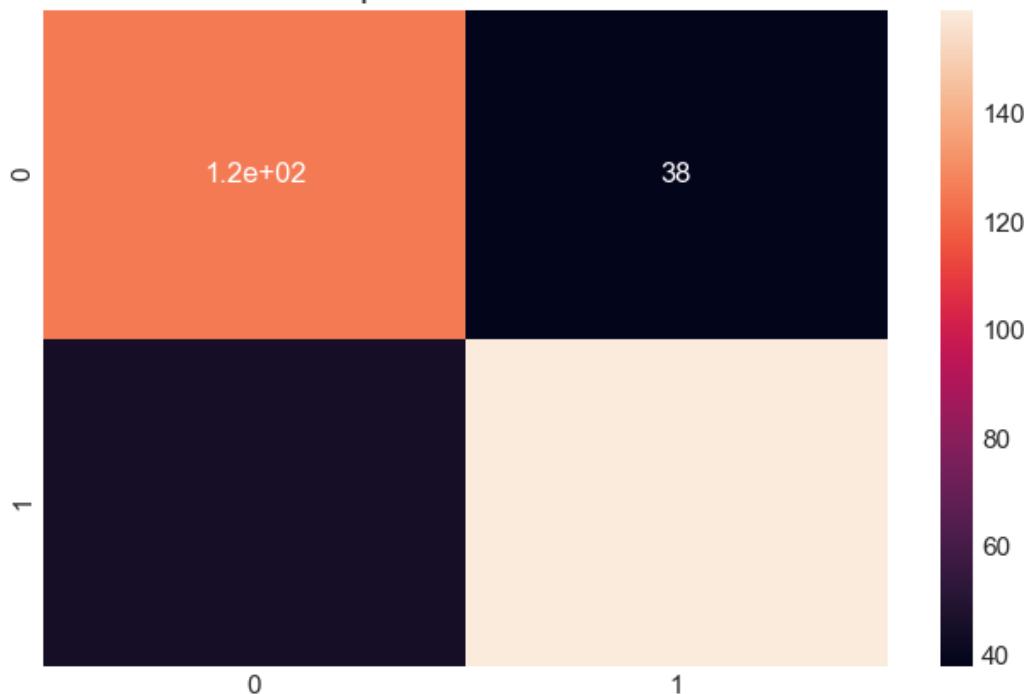
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.808	0.778	0.792	162.0	0.816
class 1	0.829	0.854	0.841	205.0	0.816
accuracy	0.82	0.82	0.82	0.82	0.816
macro avg	0.819	0.816	0.817	367.0	0.816
weighted avg	0.82	0.82	0.82	367.0	0.816

© Dr. Alexander Wagner. Все права охраняются законом

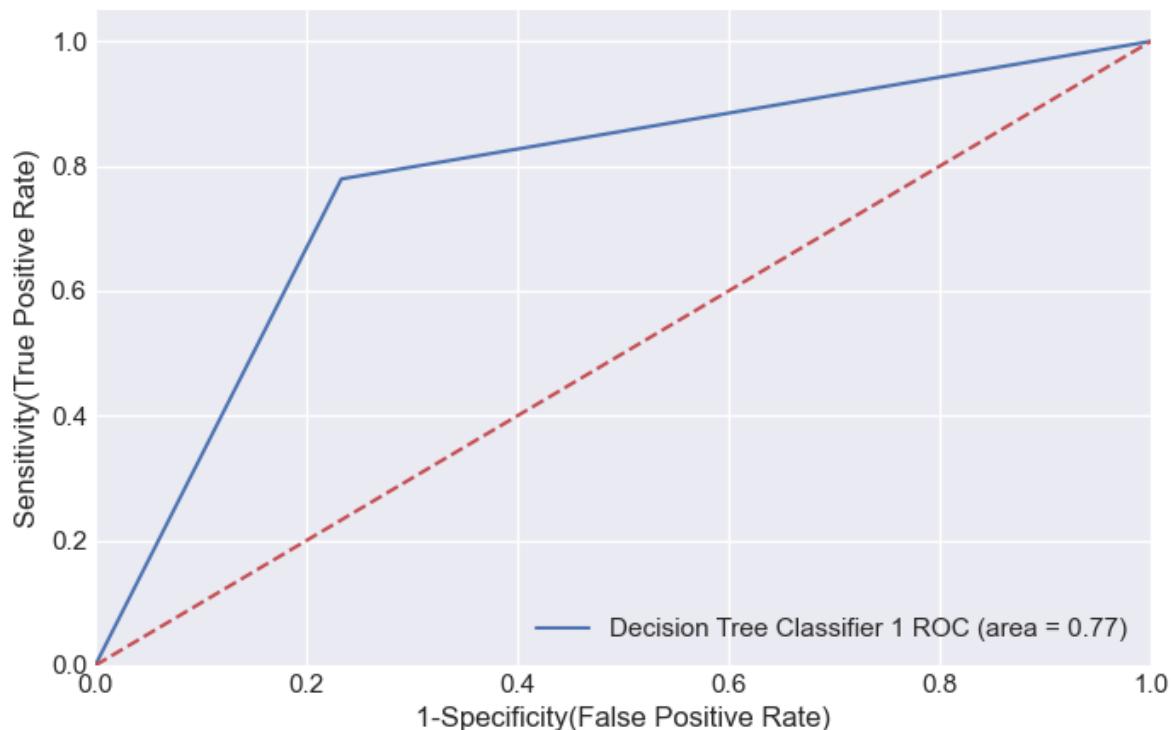
Confusion Matrix



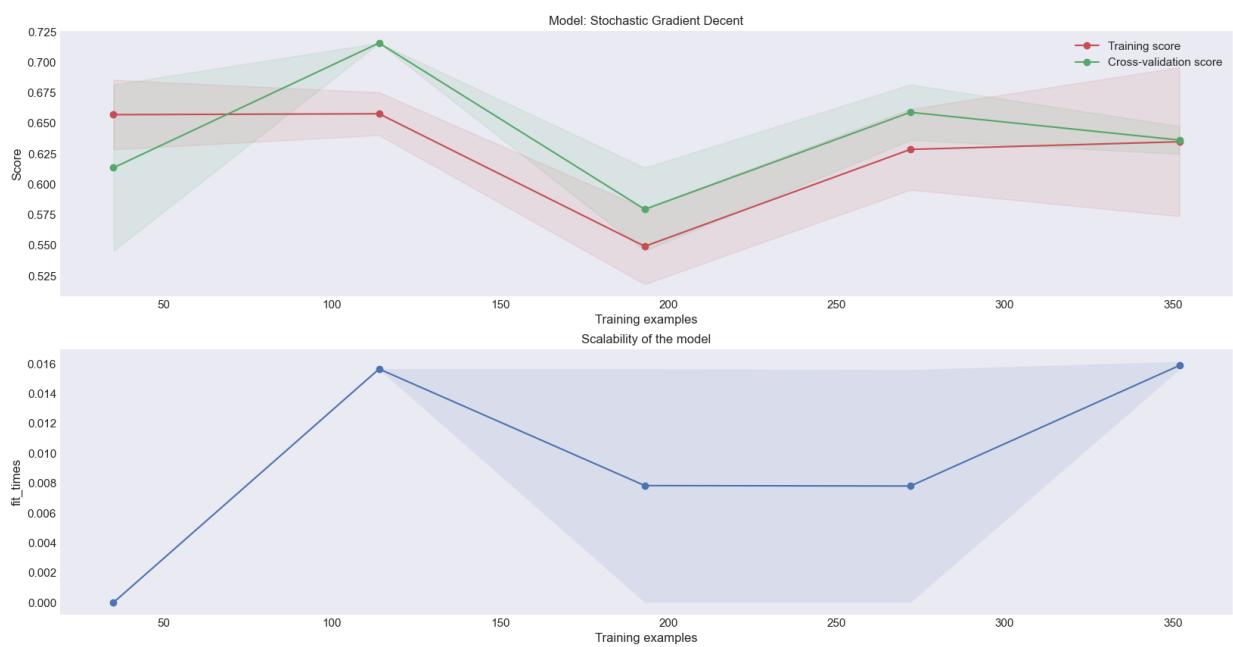
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: Stochastic Gradient Decent

Stochastic gradient descent (often abbreviated SGD) is an iterative method for optimizing an objective function with suitable smoothness properties (e.g. differentiable or subdifferentiable). It can be regarded as a stochastic approximation of gradient descent optimization, since it replaces the actual gradient (calculated from the entire data set) by an estimate thereof (calculated from a randomly selected subset of the data). Especially in big data applications this reduces the computational burden, achieving faster iterations in trade for a slightly lower convergence rate. Reference Wikipedia.

Таблица классификации

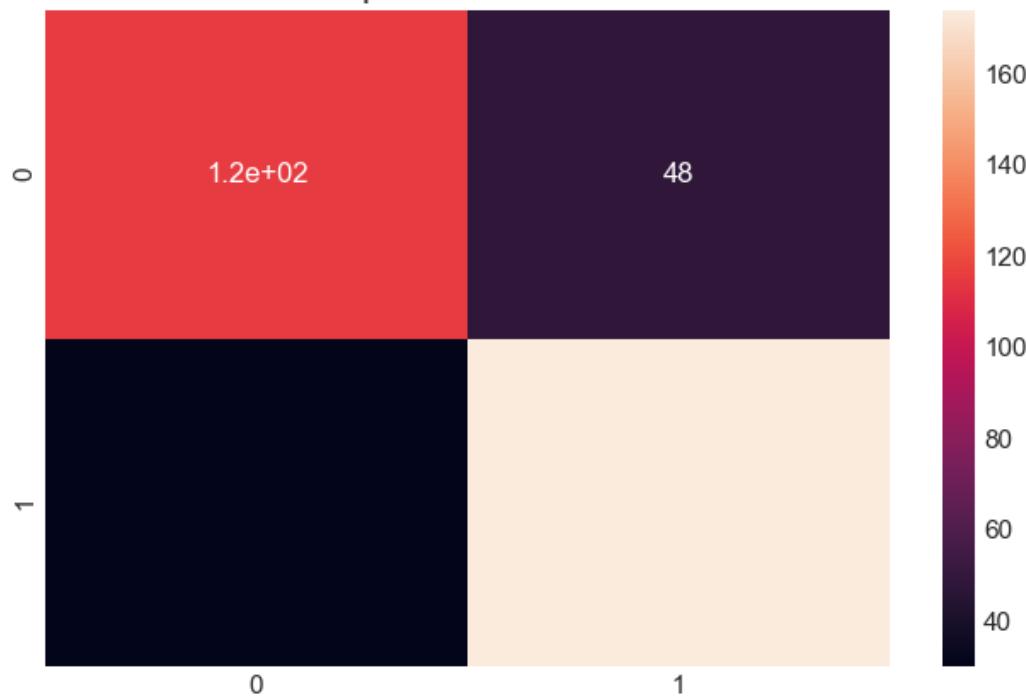
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.65	0.815	0.723	162.0	0.824
class 1	0.817	0.654	0.726	205.0	0.824
accuracy	0.725	0.725	0.725	0.725	0.824
macro avg	0.734	0.734	0.725	367.0	0.824
weighted avg	0.743	0.725	0.725	367.0	0.824

© Dr. Alexander Wagner. Все права охраняются законом

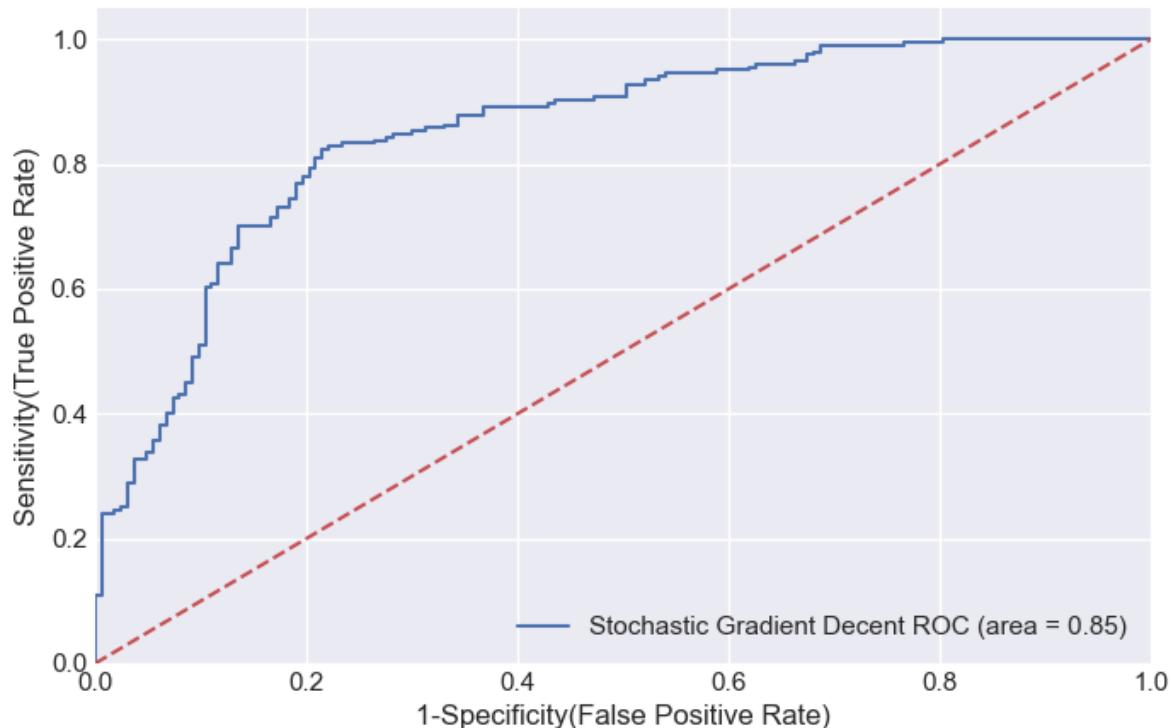
Confusion Matrix



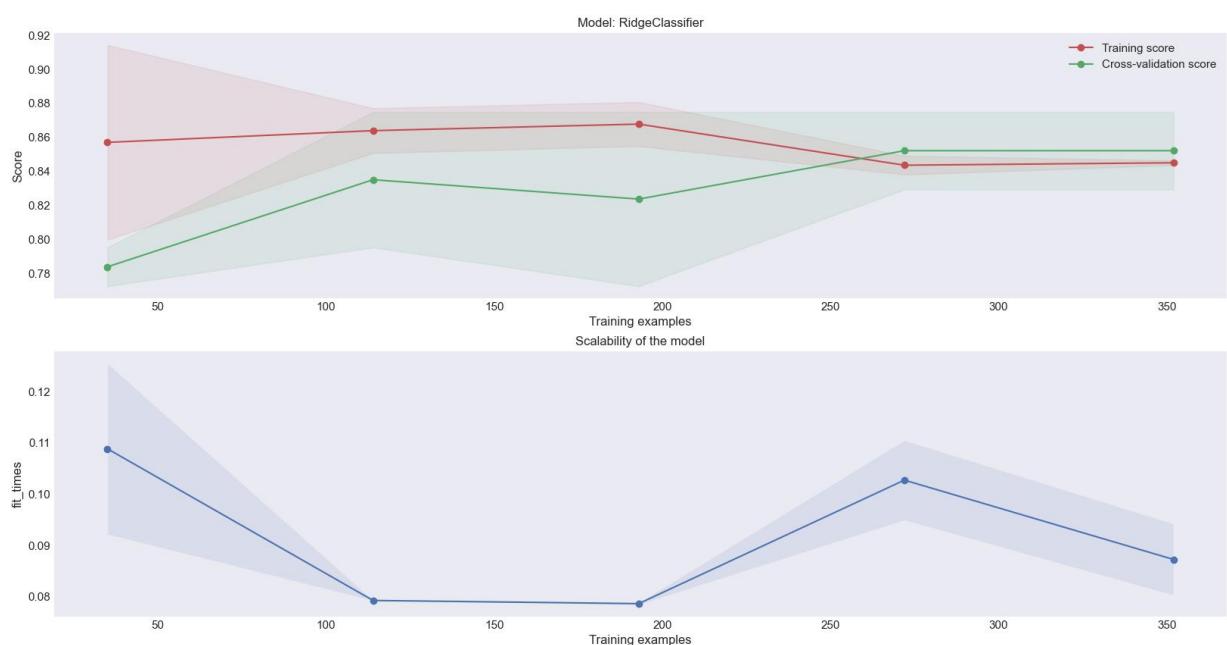
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: RidgeClassifier

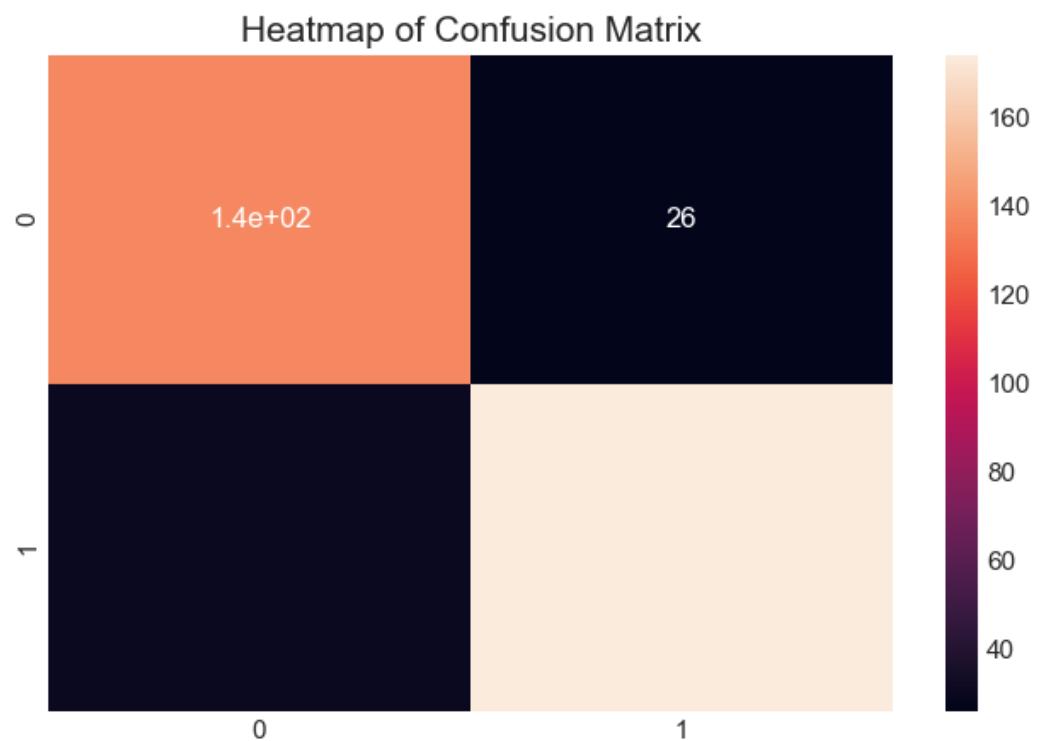
Tikhonov Regularization, colloquially known as Ridge Classifier, is the most commonly used regression algorithm to approximate an answer for an equation with no unique solution. This type of problem is very common in machine learning tasks, where the "best" solution must be chosen using limited data. If a unique solution exists, algorithm will return the optimal value. However, if multiple solutions exist, it may choose any of them. Reference Brilliant.org.

Таблица классификации

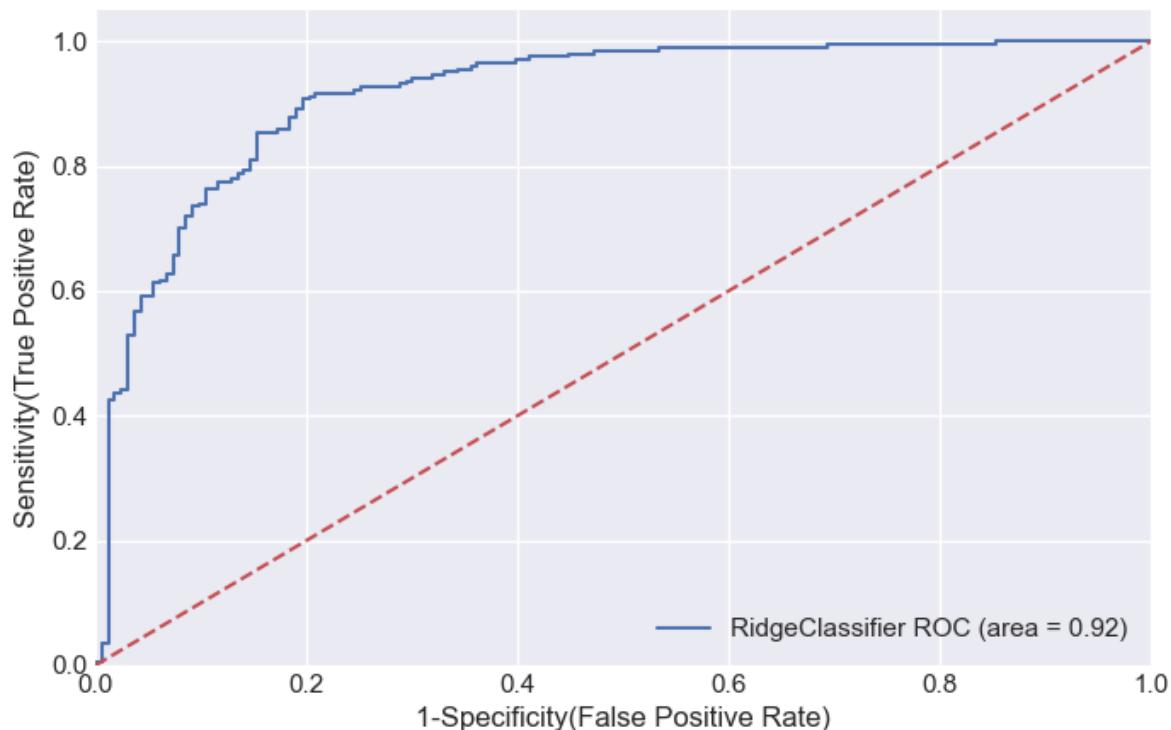
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.848	0.827	0.837	162.0	0.911
class 1	0.866	0.883	0.874	205.0	0.911
accuracy	0.858	0.858	0.858	0.858	0.911
macro avg	0.857	0.855	0.856	367.0	0.911
weighted avg	0.858	0.858	0.858	367.0	0.911

© Dr. Alexander Wagner. Все права охраняются законом

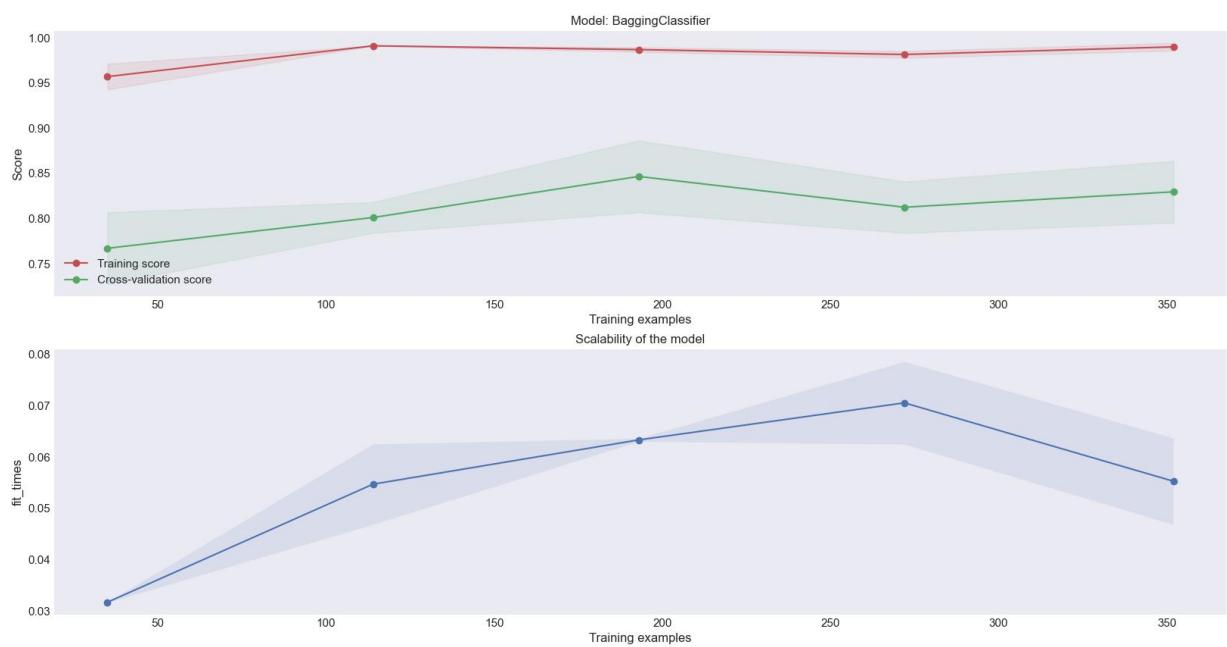
Confusion Matrix



ROC Curve



Score plot



Модель: BaggingClassifier

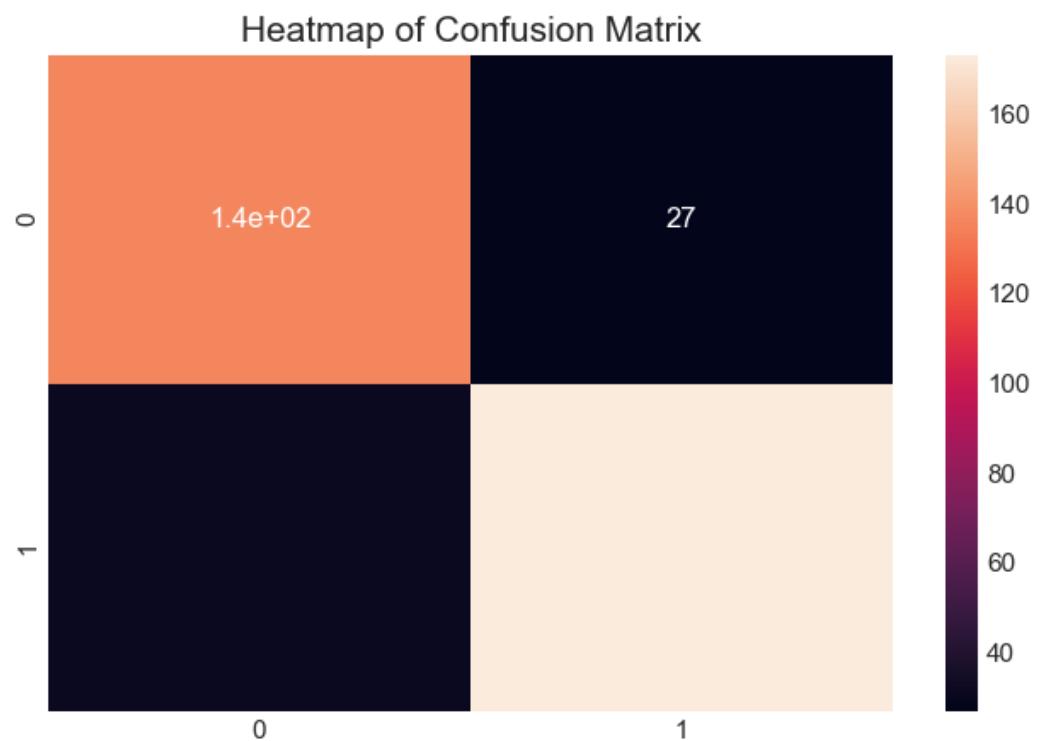
Bootstrap aggregating, also called Bagging, is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also reduces variance and helps to avoid overfitting. Although it is usually applied to decision tree methods, it can be used with any type of method. Bagging is a special case of the model averaging approach. Bagging leads to "improvements for unstable procedures", which include, for example, artificial neural networks, classification and regression trees, and subset selection in linear regression. On the other hand, it can mildly degrade the performance of stable methods such as K-nearest neighbors. Reference Wikipedia.

Таблица классификации

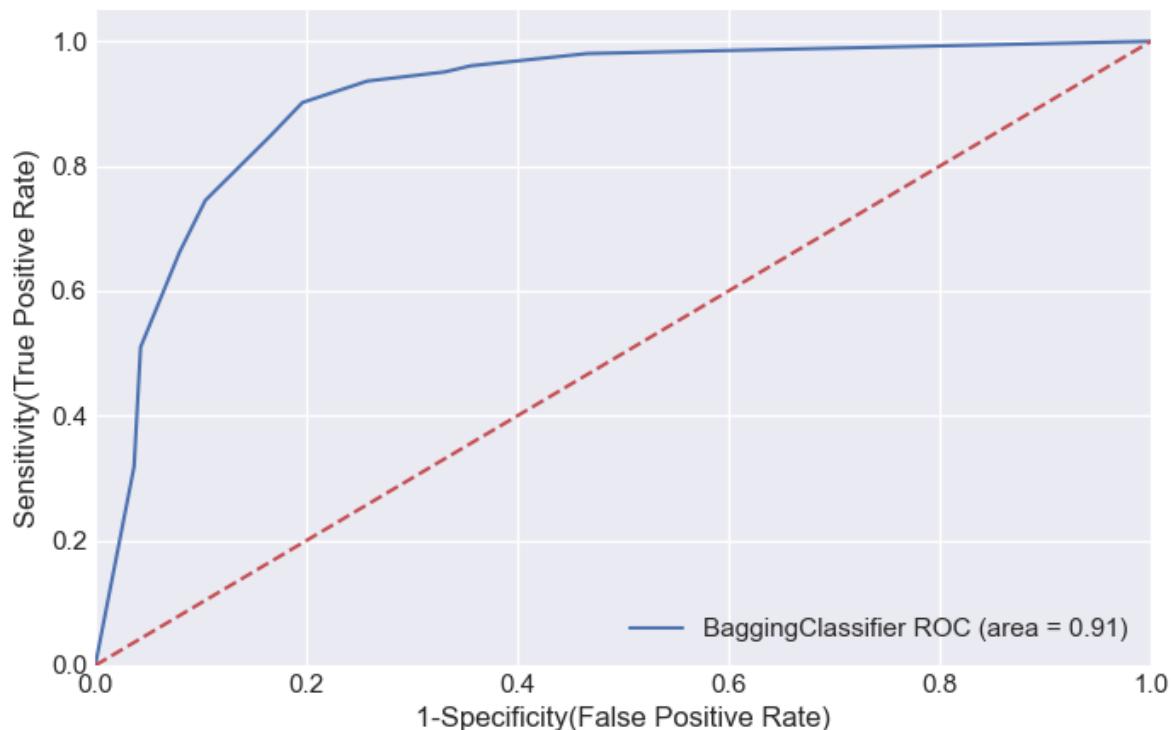
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.843	0.827	0.835	162.0	0.91
class 1	0.865	0.878	0.872	205.0	0.91
accuracy	0.856	0.856	0.856	0.856	0.91
macro avg	0.854	0.853	0.853	367.0	0.91
weighted avg	0.855	0.856	0.855	367.0	0.91

© Dr. Alexander Wagner. Все права охраняются законом

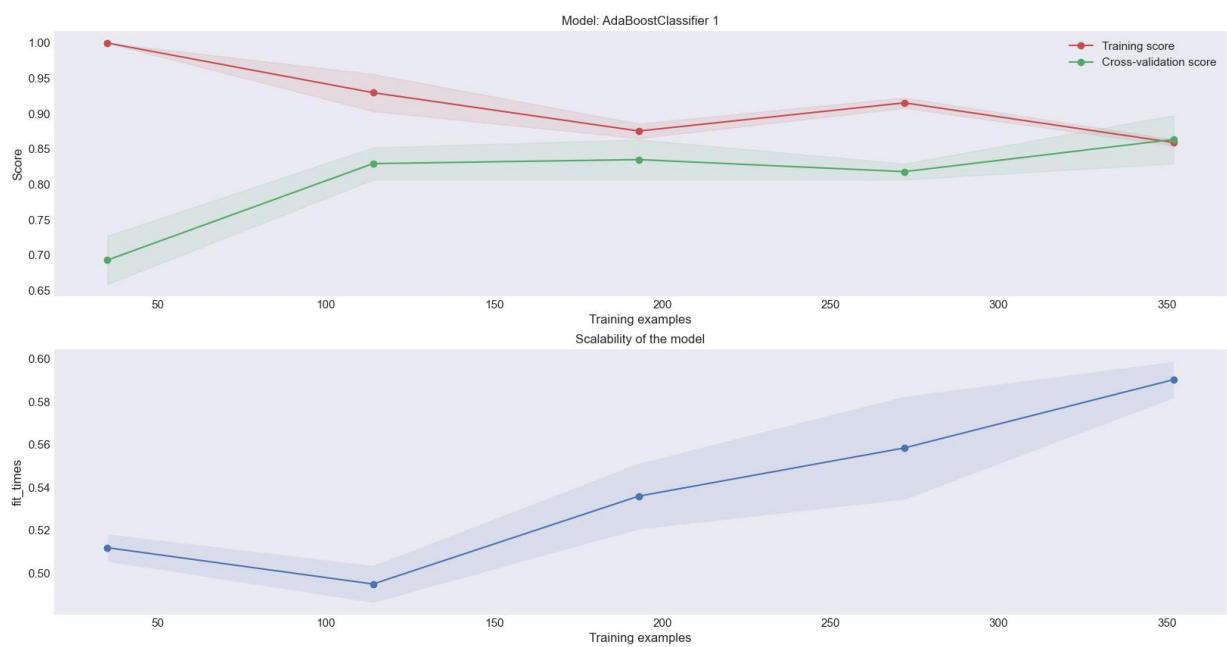
Confusion Matrix



ROC Curve



Score plot



Модель: AdaBoostClassifier 1

The core principle of AdaBoost ("Adaptive Boosting") is to fit a sequence of weak learners (i.e., models that are only slightly better than random guessing, such as small decision trees) on repeatedly modified versions of the data. The predictions from all of them are then combined through a weighted majority vote (or sum) to produce the final prediction. The data modifications at each so-called boosting iteration consist of applying N weights to each of the training samples. Initially, those weights are all set to $1/N$, so that the first step simply trains a weak learner on the original data. For each successive iteration, the sample weights are individually modified and the learning algorithm is reapplied to the reweighted data. At a given step, those training examples that were incorrectly predicted by the boosted model induced at the previous step have their weights increased, whereas the weights are decreased for those that were predicted correctly. As iterations proceed, examples that are difficult to predict receive ever-increasing influence. Each subsequent weak learner is thereby forced to concentrate on the examples that are missed by the previous ones in the sequence. Reference sklearn documentation.

Таблица классификации

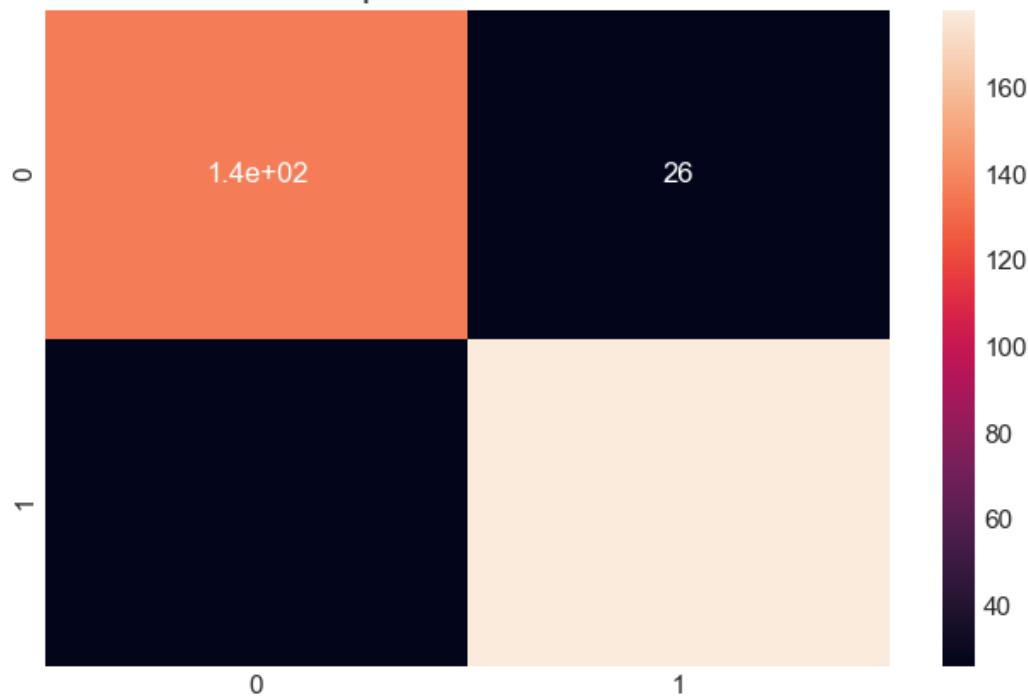
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.857	0.815	0.835	162.0	0.926
class 1	0.859	0.893	0.876	205.0	0.926
accuracy	0.858	0.858	0.858	0.858	0.926
macro avg	0.858	0.854	0.856	367.0	0.926
weighted avg	0.858	0.858	0.858	367.0	0.926

© Dr. Alexander Wagner. Все права охраняются законом

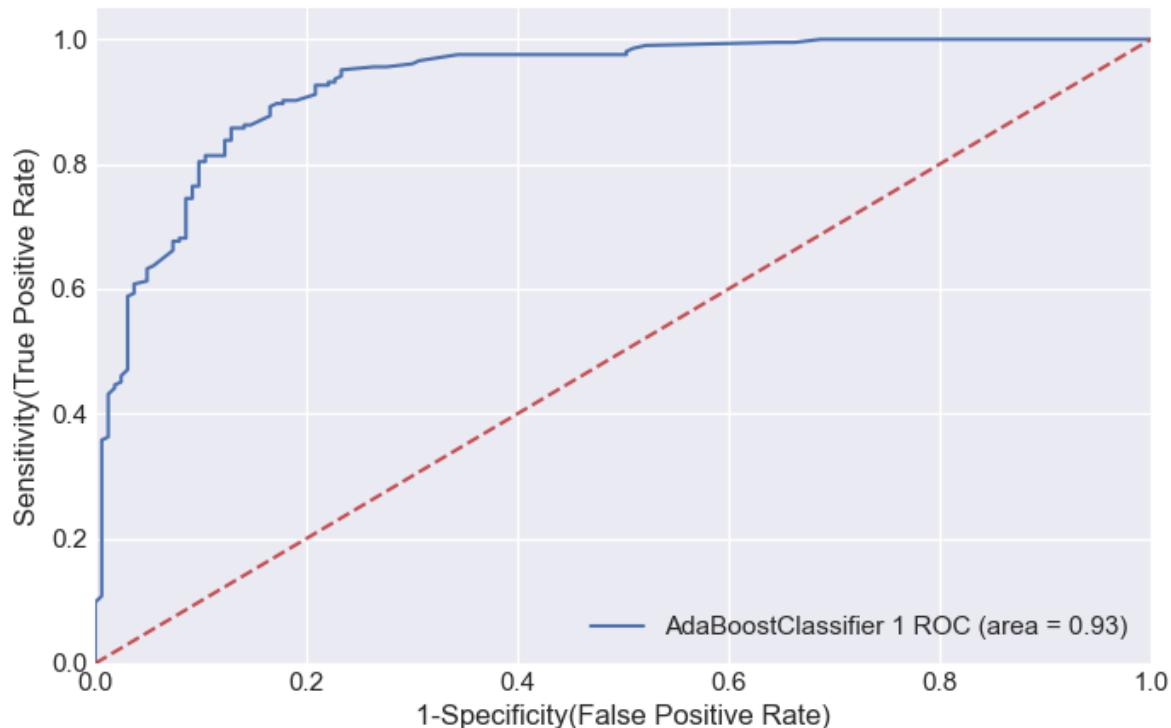
Confusion Matrix



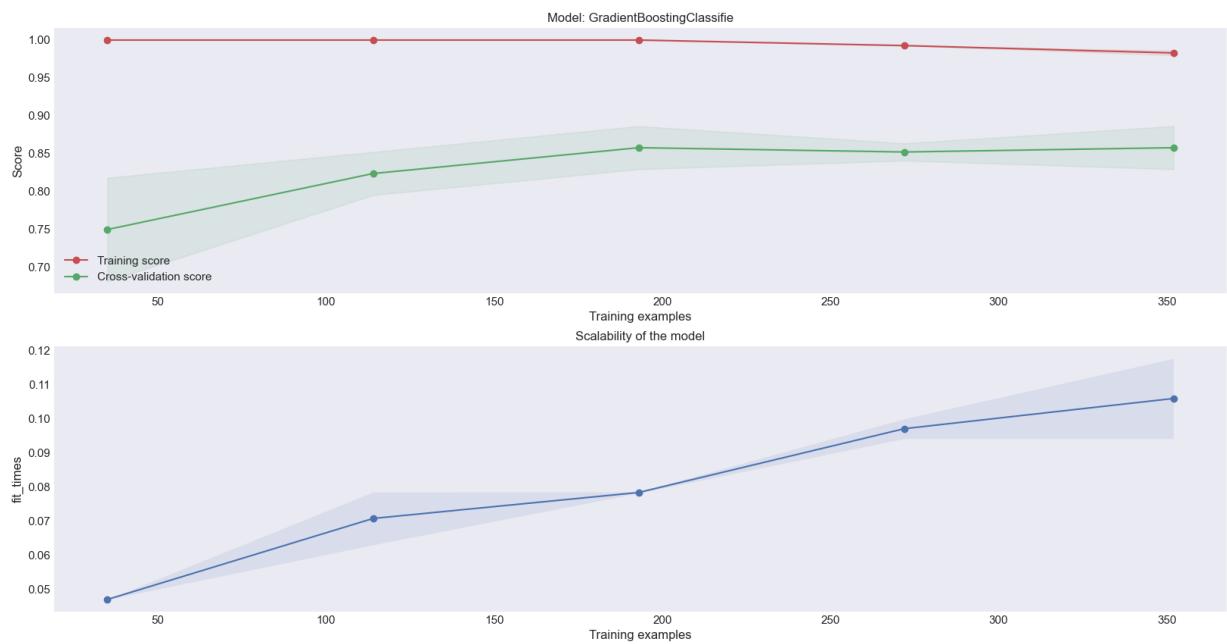
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: GradientBoostingClassifier

Thanks to <https://www.kaggle.com/kabure/titanic-eda-model-pipeline-keras-nn>

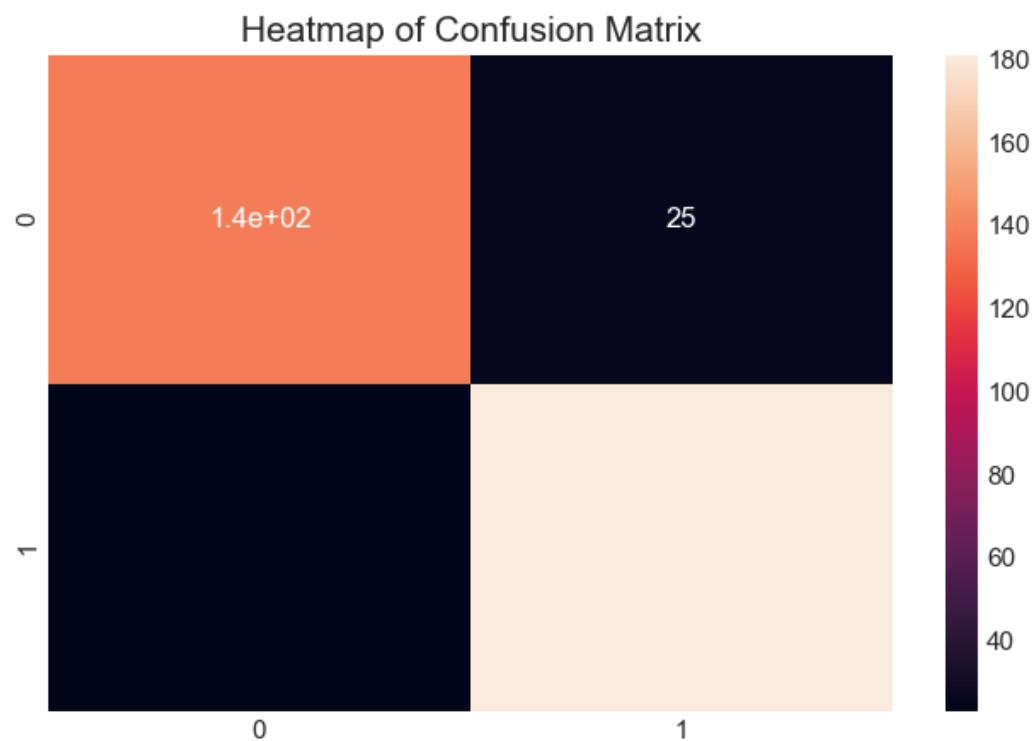
Gradient Boosting builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions. In each stage `n_classes_` regression trees are fit on the negative gradient of the binomial or multinomial deviance loss function. Binary classification is a special case where only a single regression tree is induced. The features are always randomly permuted at each split. Therefore, the best found split may vary, even with the same training data and `max_features=n_features`, if the improvement of the criterion is identical for several splits enumerated during the search of the best split. To obtain a deterministic behaviour during fitting, `random_state` has to be fixed. Reference sklearn documentation.

Таблица классификации

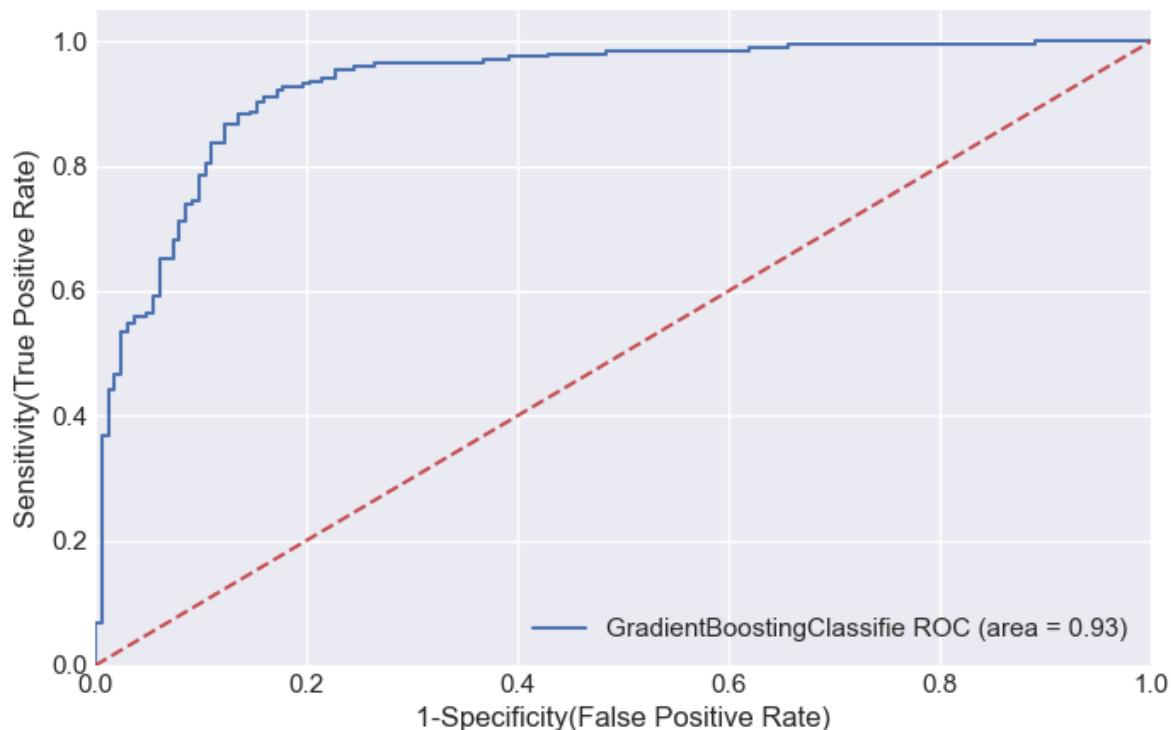
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.882	0.833	0.857	162.0	0.927
class 1	0.874	0.912	0.893	205.0	0.927
accuracy	0.877	0.877	0.877	0.877	0.927
macro avg	0.878	0.873	0.875	367.0	0.927
weighted avg	0.878	0.877	0.877	367.0	0.927

© Dr. Alexander Wagner. Все права охраняются законом

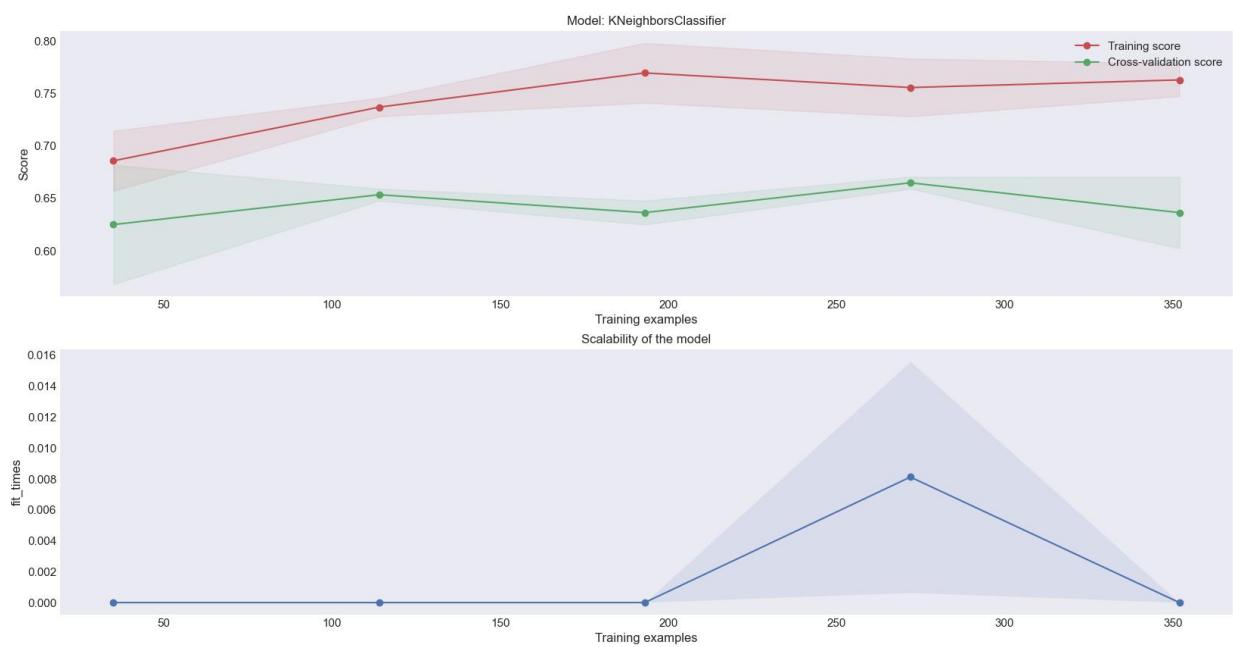
Confusion Matrix



ROC Curve



Score plot



Модель: KNeighborsClassifier

Thanks to <https://www.kaggle.com/startupsci/titanic-data-science-solutions>

In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. A sample is classified by a majority vote of its neighbors, with the sample being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). Reference Wikipedia.

Таблица классификации

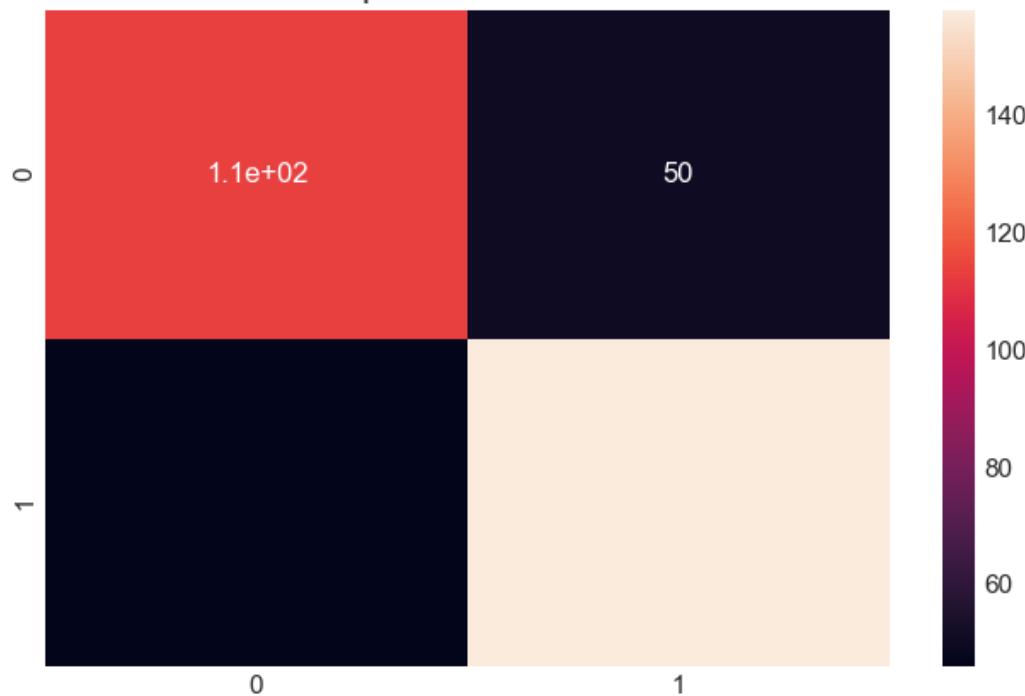
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.649	0.685	0.667	162.0	0.737
class 1	0.74	0.707	0.723	205.0	0.737
accuracy	0.698	0.698	0.698	0.698	0.737
macro avg	0.694	0.696	0.695	367.0	0.737
weighted avg	0.7	0.698	0.698	367.0	0.737

© Dr. Alexander Wagner. Все права охраняются законом

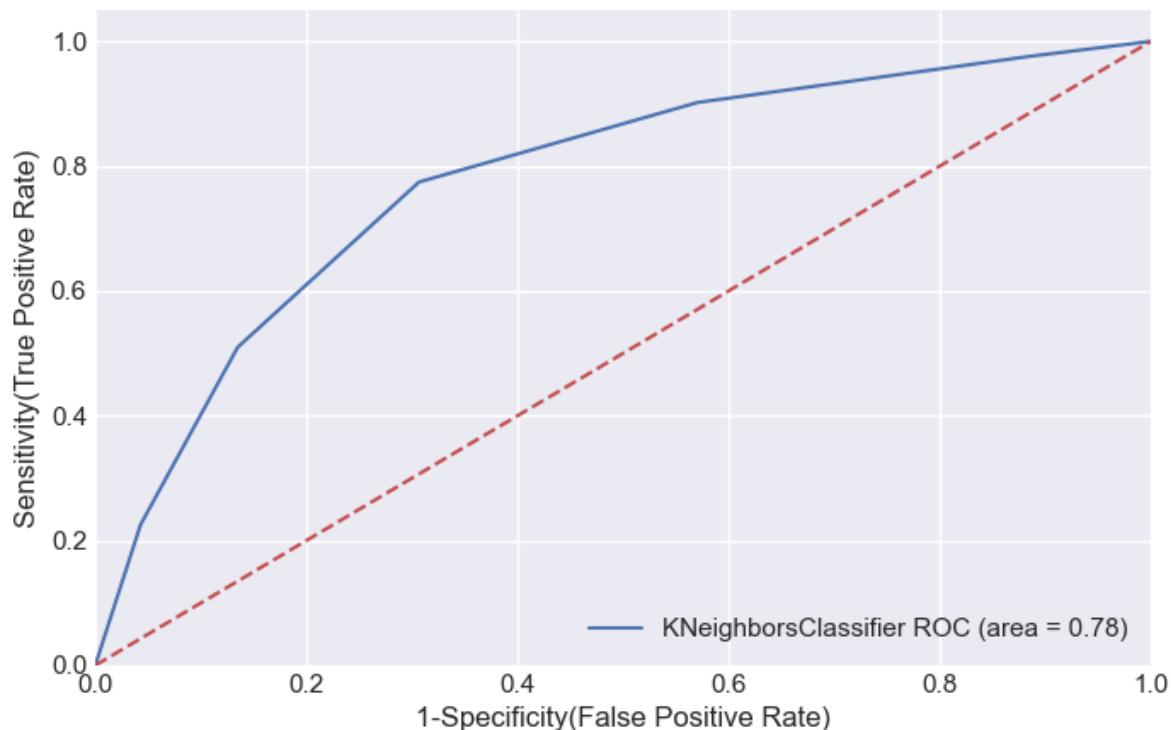
Confusion Matrix



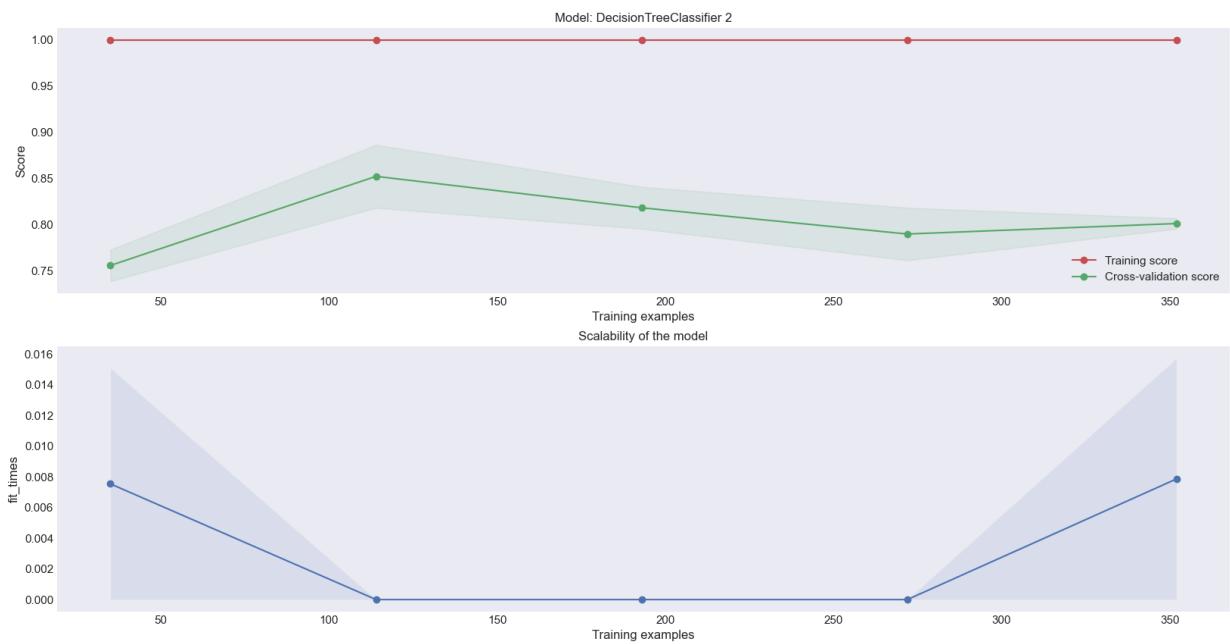
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: DecisionTreeClassifier 2

This model uses a Decision Tree as a predictive model which maps features (tree branches) to conclusions about the target value (tree leaves). Tree models where the target variable can take a finite set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. Reference Wikipedia.

Таблица классификации

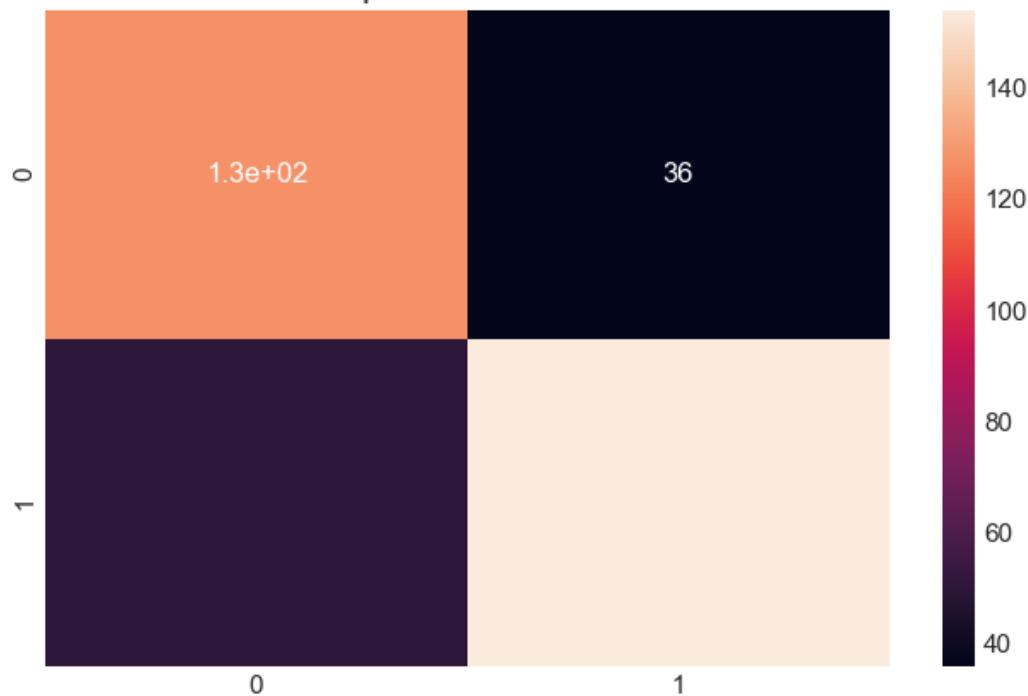
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.781	0.815	0.798	162.0	0.817
class 1	0.848	0.82	0.834	205.0	0.817
accuracy	0.817	0.817	0.817	0.817	0.817
macro avg	0.815	0.817	0.816	367.0	0.817
weighted avg	0.819	0.817	0.818	367.0	0.817

© Dr. Alexander Wagner. Все права охраняются законом

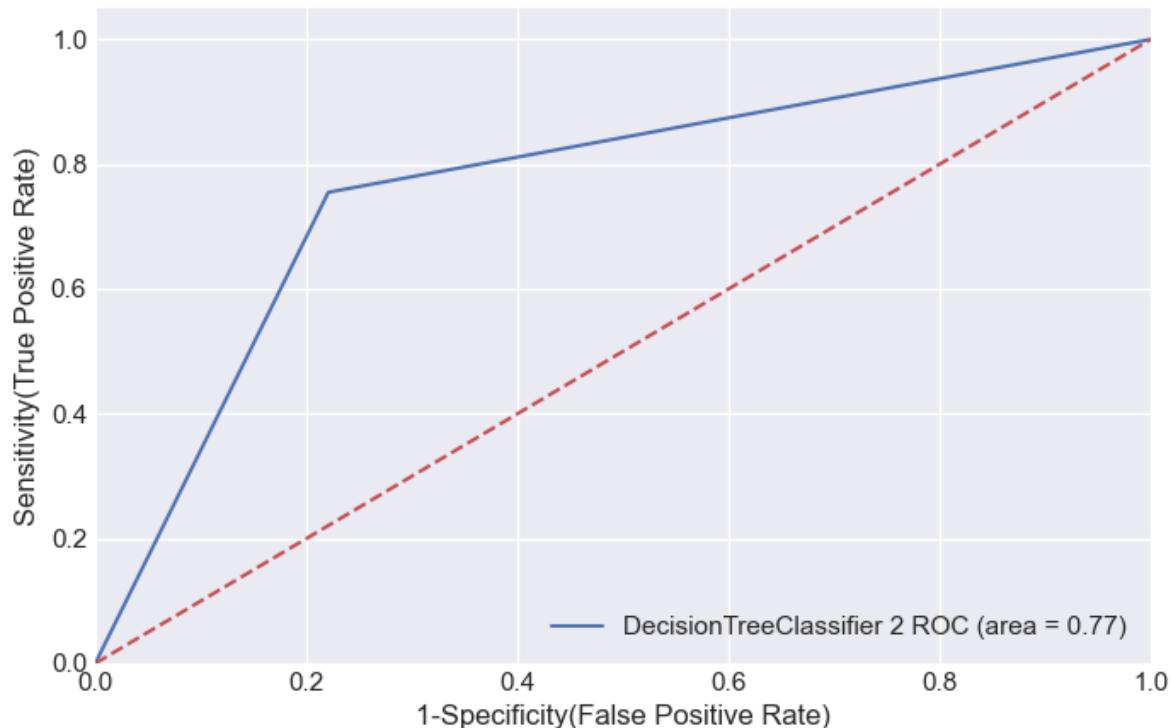
Confusion Matrix



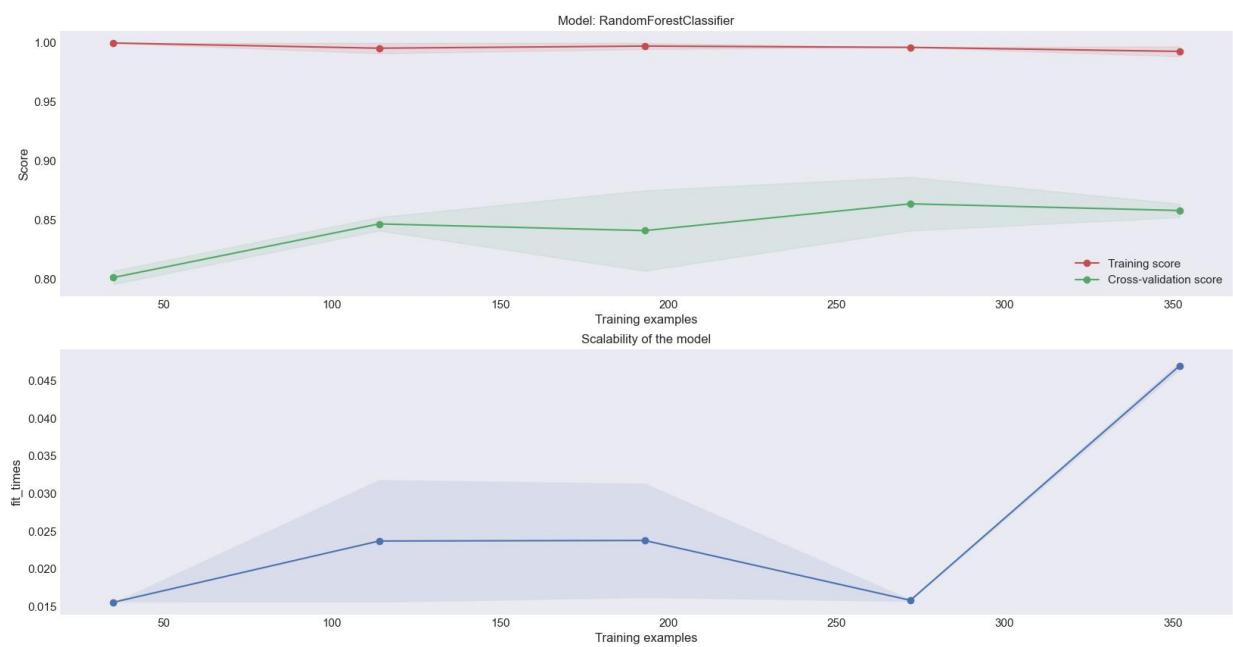
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: RandomForestClassifier

Random Forest is one of the most popular models. Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees (n_estimators= [100, 300]) at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Reference Wikipedia.

Таблица классификации

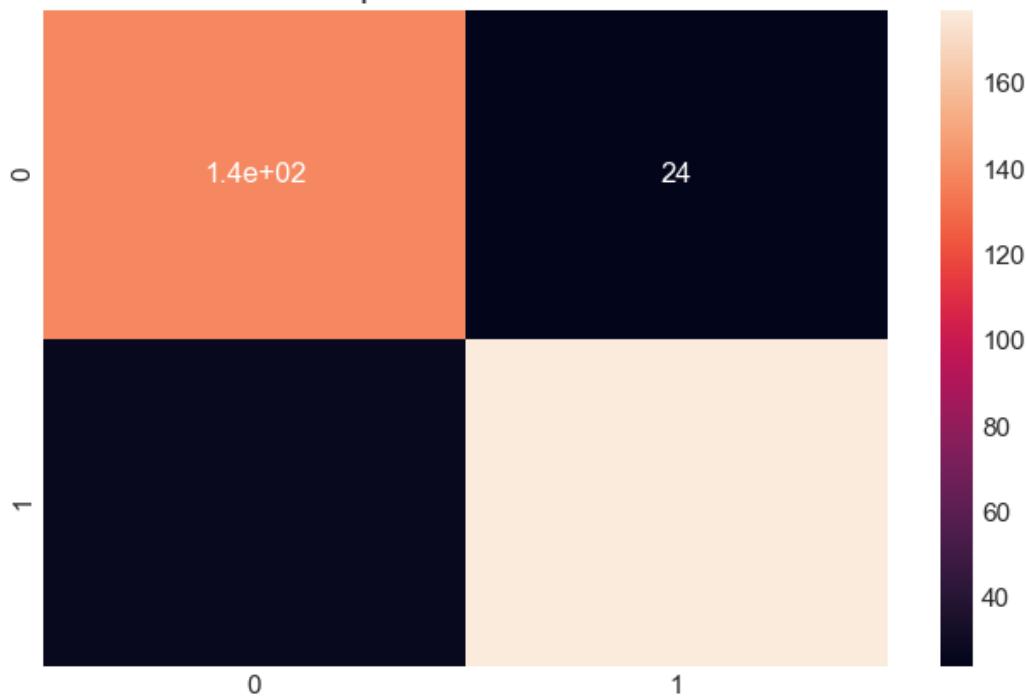
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.866	0.84	0.853	162.0	0.935
class 1	0.876	0.898	0.887	205.0	0.935
accuracy	0.872	0.872	0.872	0.872	0.935
macro avg	0.871	0.869	0.87	367.0	0.935
weighted avg	0.872	0.872	0.872	367.0	0.935

© Dr. Alexander Wagner. Все права охраняются законом

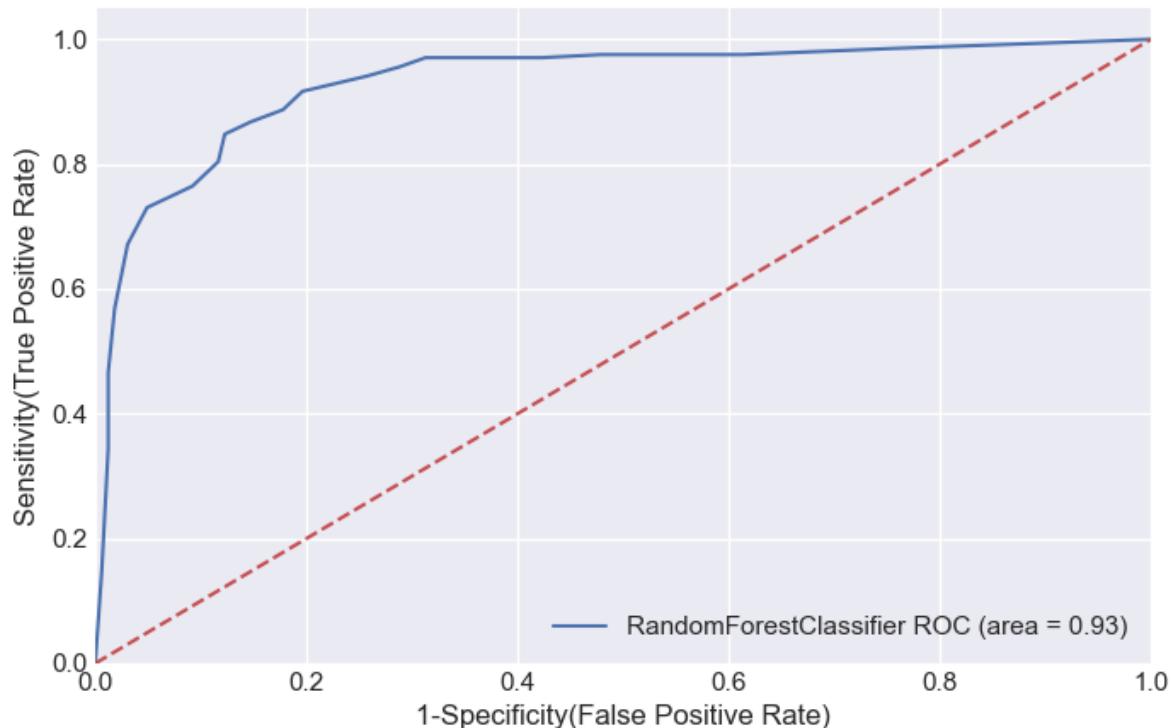
Confusion Matrix



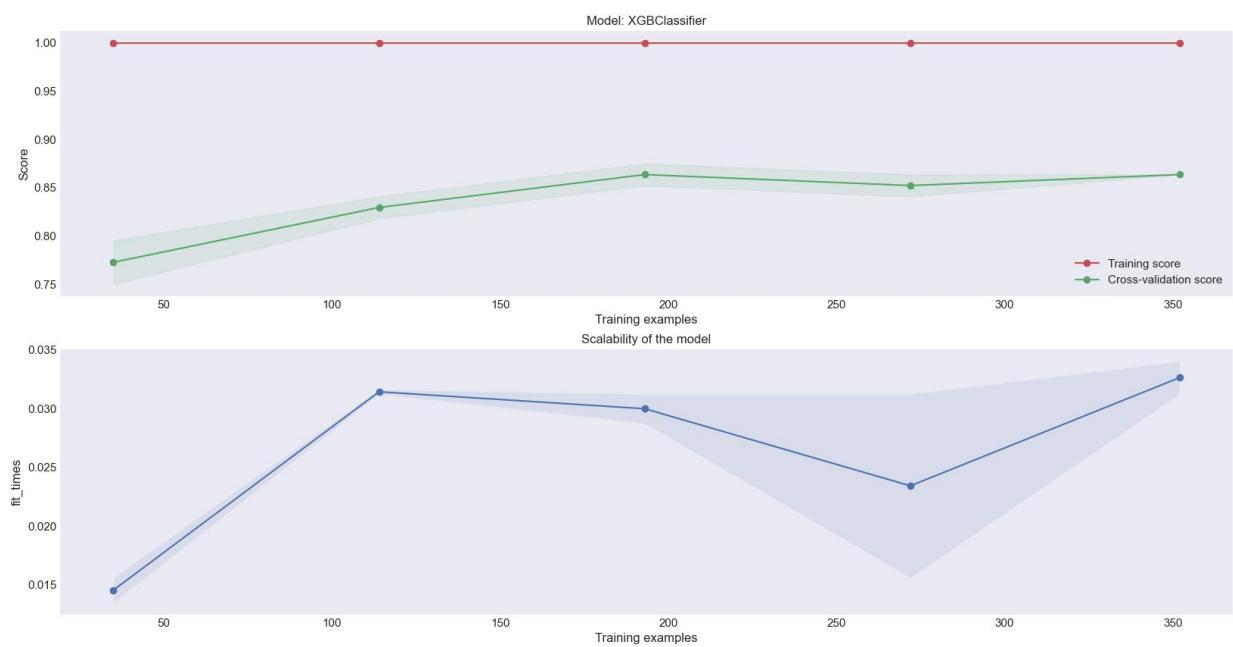
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: XGBClassifier

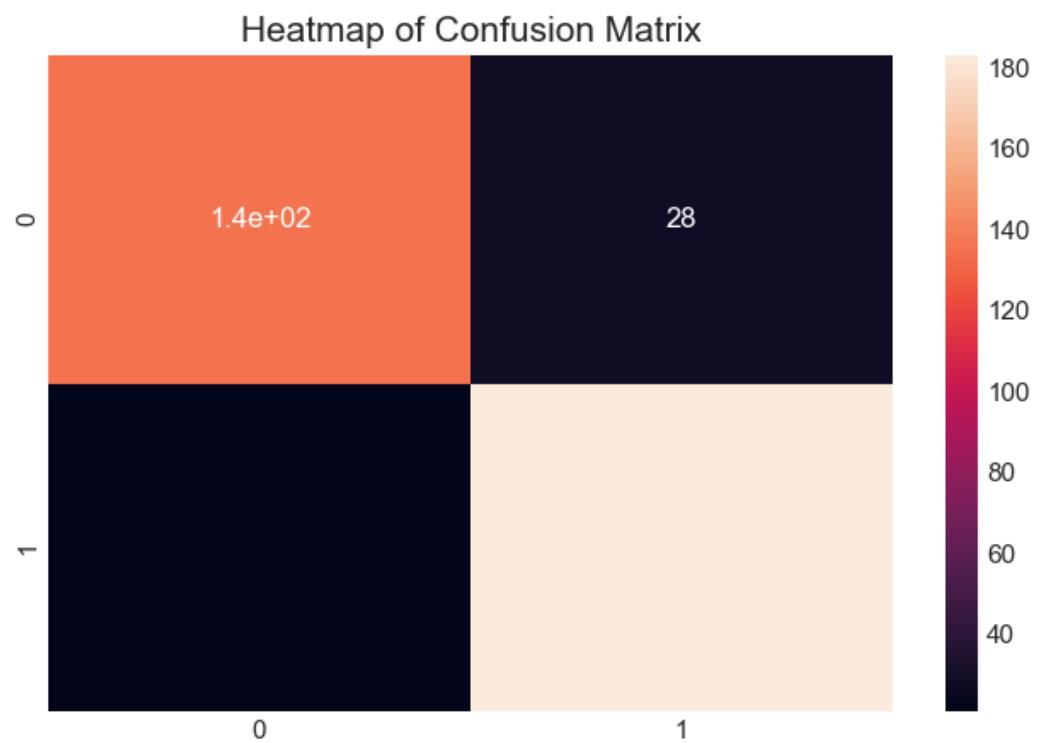
XGBoost is an ensemble tree method that applies the principle of boosting weak learners (CARTs generally) using the gradient descent architecture. XGBoost improves upon the base Gradient Boosting Machines (GBM) framework through systems optimization and algorithmic enhancements. Reference Towards Data Science.

Таблица классификации

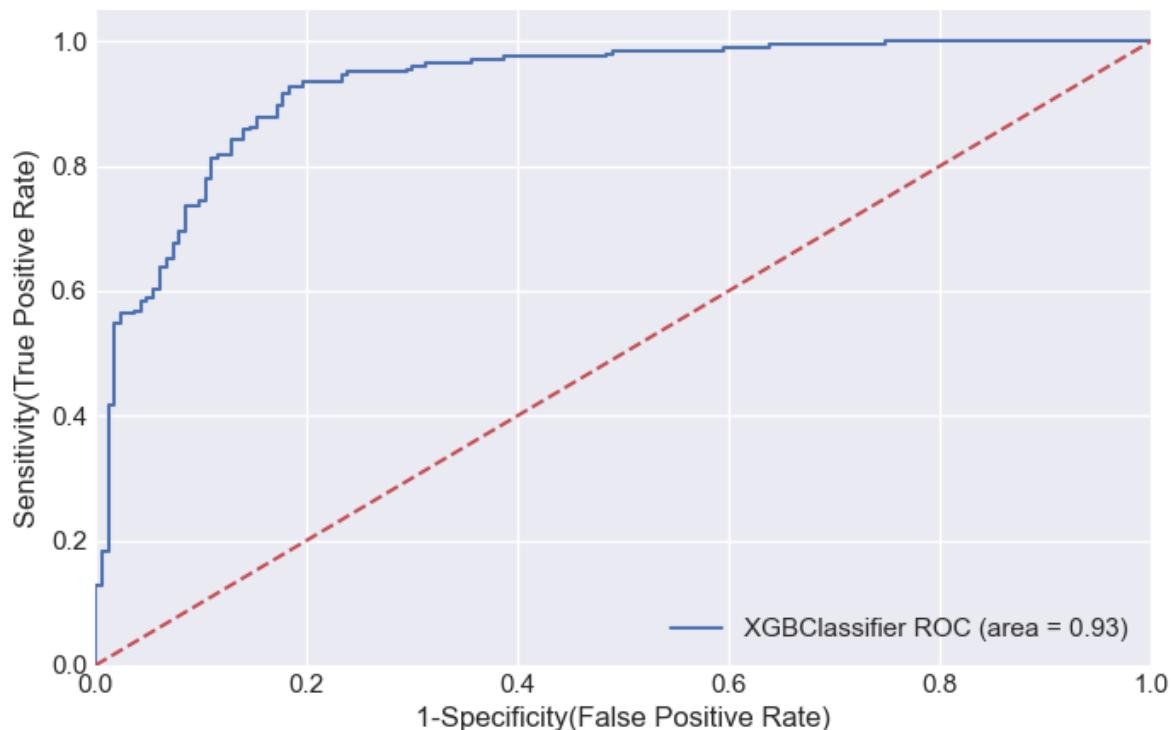
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.868	0.815	0.841	162.0	0.918
class 1	0.86	0.902	0.881	205.0	0.918
accuracy	0.864	0.864	0.864	0.864	0.918
macro avg	0.864	0.859	0.861	367.0	0.918
weighted avg	0.864	0.864	0.863	367.0	0.918

© Dr. Alexander Wagner. Все права охраняются законом

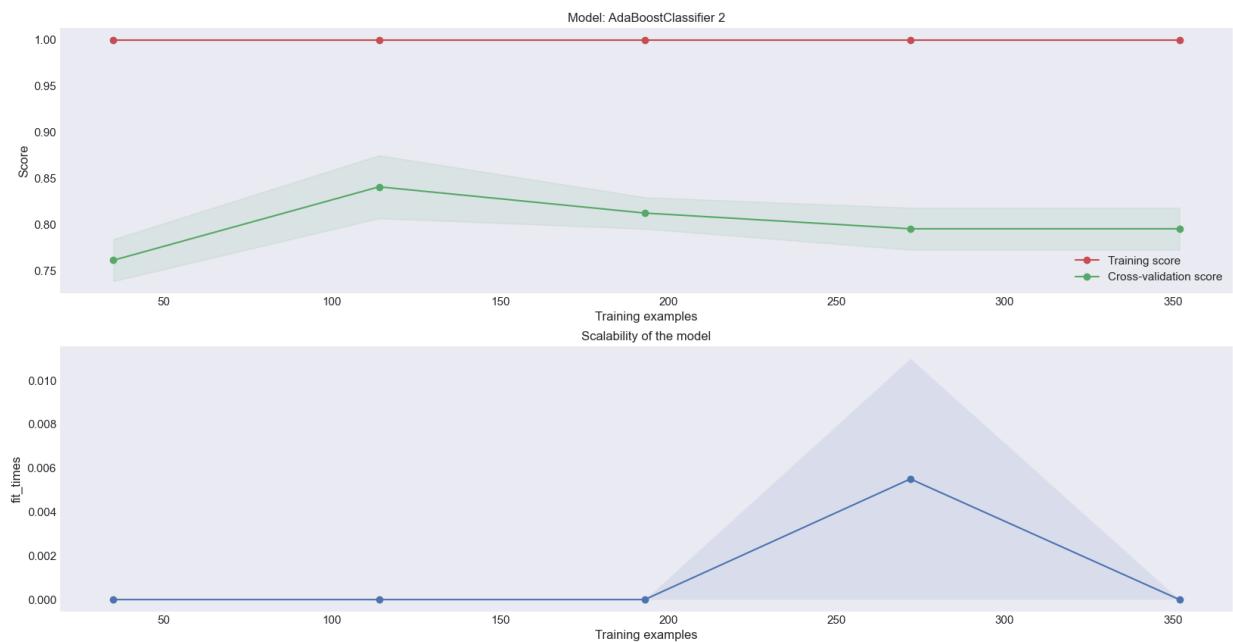
Confusion Matrix



ROC Curve



Score plot



Модель: AdaBoostClassifier 2

The core principle of AdaBoost ("Adaptive Boosting") is to fit a sequence of weak learners (i.e., models that are only slightly better than random guessing, such as small decision trees) on repeatedly modified versions of the data. The predictions from all of them are then combined through a weighted majority vote (or sum) to produce the final prediction. The data modifications at each so-called boosting iteration consist of applying N weights to each of the training samples. Initially, those weights are all set to $1/N$, so that the first step simply trains a weak learner on the original data. For each successive iteration, the sample weights are individually modified and the learning algorithm is reapplied to the reweighted data. At a given step, those training examples that were incorrectly predicted by the boosted model induced at the previous step have their weights increased, whereas the weights are decreased for those that were predicted correctly. As iterations proceed, examples that are difficult to predict receive ever-increasing influence. Each subsequent weak learner is thereby forced to concentrate on the examples that are missed by the previous ones in the sequence. Reference sklearn documentation.

Таблица классификации

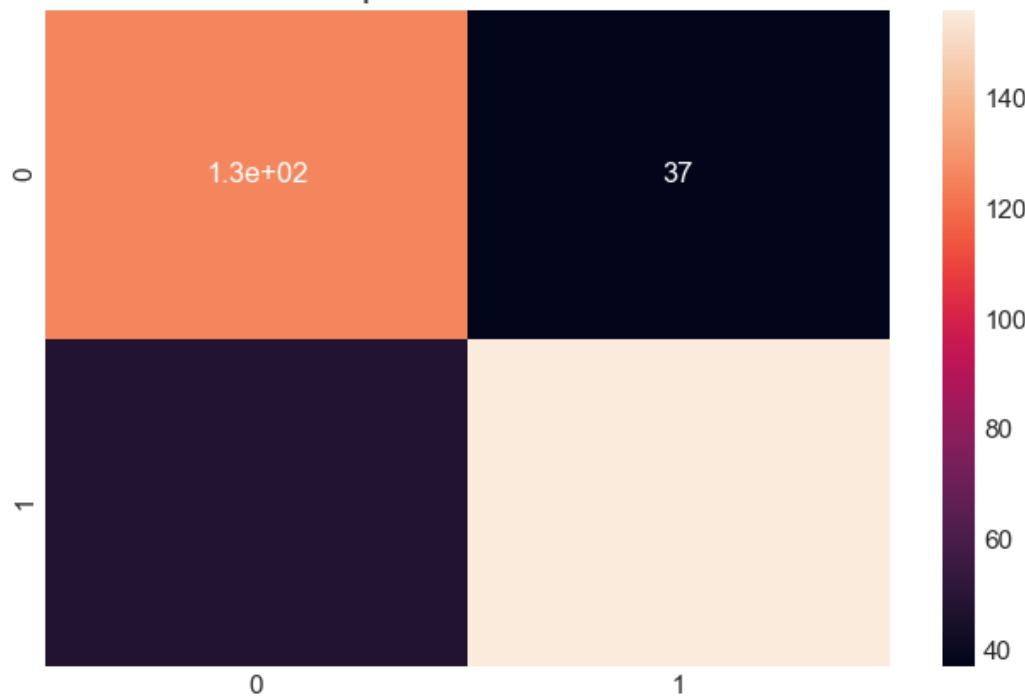
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.773	0.821	0.796	162.0	0.815
class 1	0.851	0.81	0.83	205.0	0.815
accuracy	0.815	0.815	0.815	0.815	0.815
macro avg	0.812	0.815	0.813	367.0	0.815
weighted avg	0.817	0.815	0.815	367.0	0.815

© Dr. Alexander Wagner. Все права охраняются законом

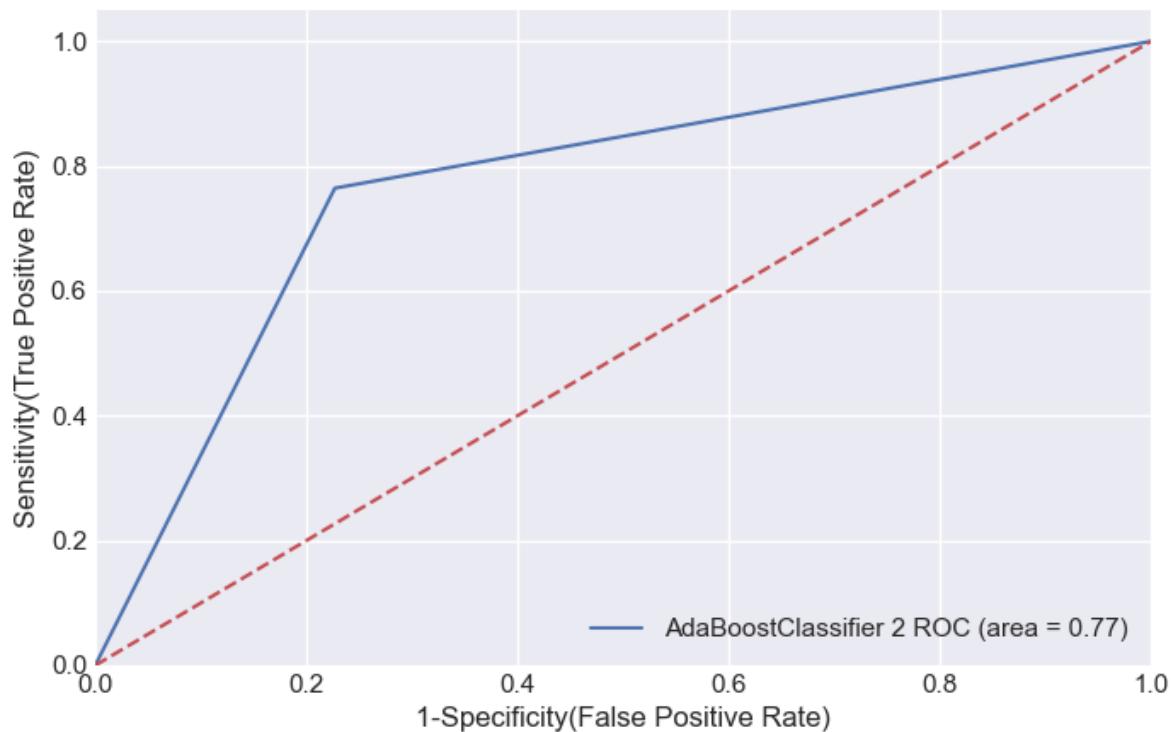
Confusion Matrix



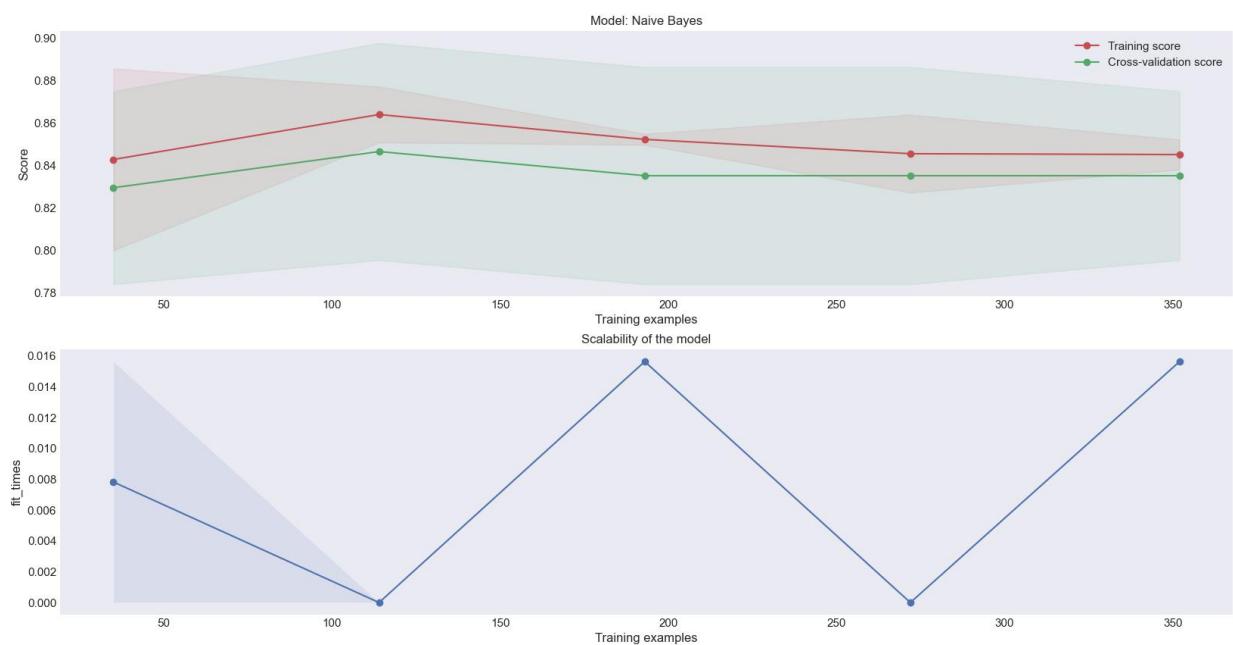
Heatmap of Confusion Matrix



ROC Curve



Score plot



Модель: Naive Bayes

Thanks to <https://www.kaggle.com/startupsci/titanic-data-science-solutions>

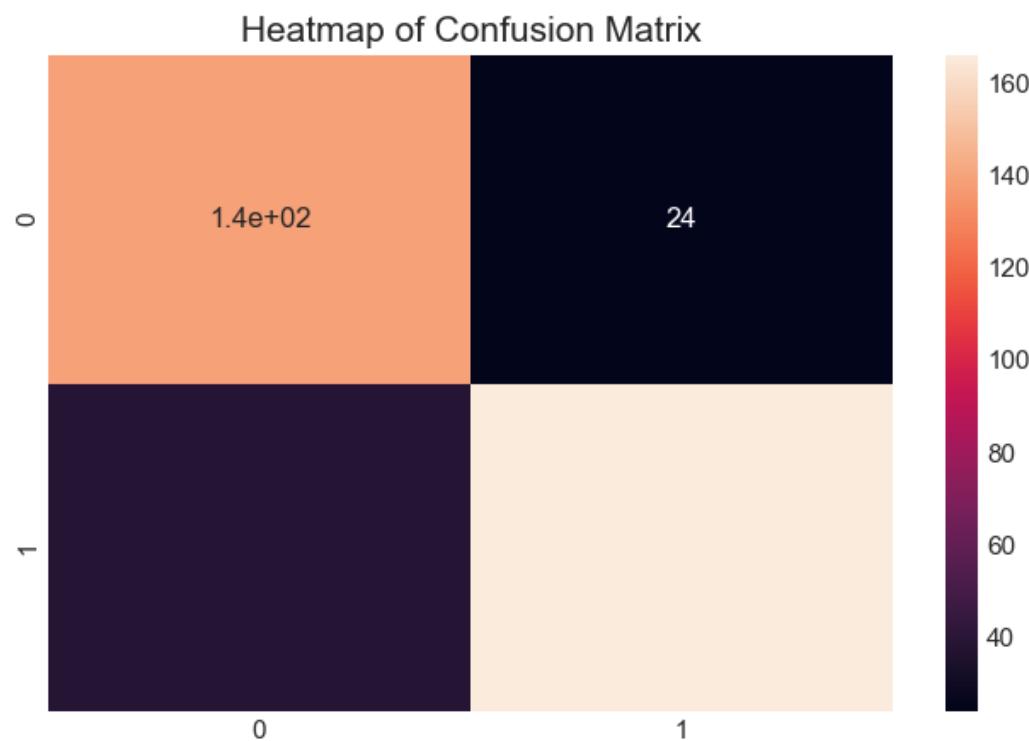
In machine learning, Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features) in a learning problem. Reference Wikipedia.

Таблица классификации

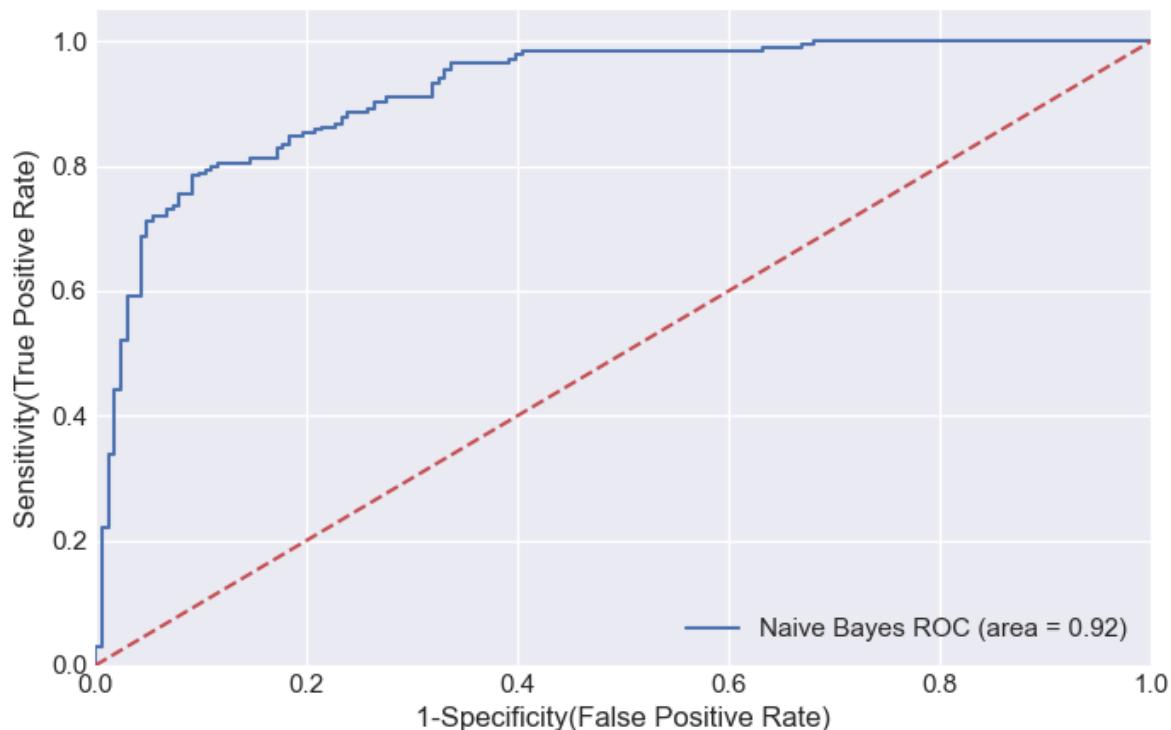
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.852	0.852	0.852	162.0	0.918
class 1	0.883	0.883	0.883	205.0	0.918
accuracy	0.869	0.869	0.869	0.869	0.918
macro avg	0.867	0.867	0.867	367.0	0.918
weighted avg	0.869	0.869	0.869	367.0	0.918

© Dr. Alexander Wagner. Все права охраняются законом

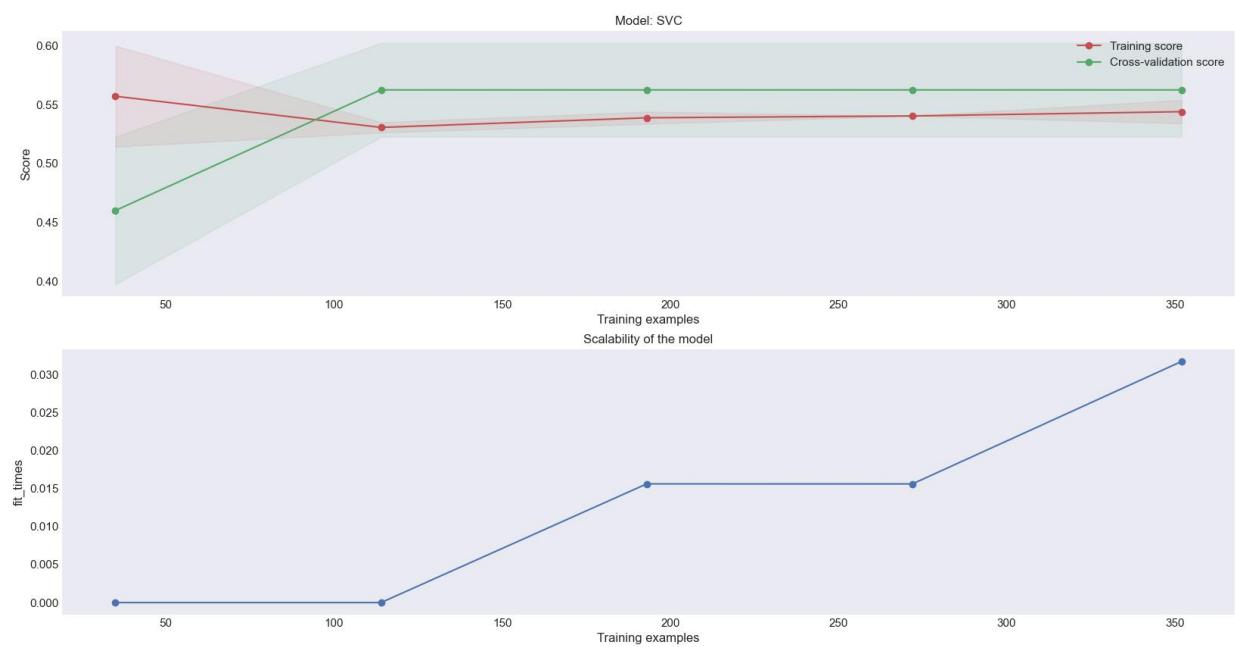
Confusion Matrix



ROC Curve



Score plot



Модель: SVC

Support Vector Machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training samples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new test samples to one category or the other, making it a non-probabilistic binary linear classifier. Reference Wikipedia.

Таблица классификации

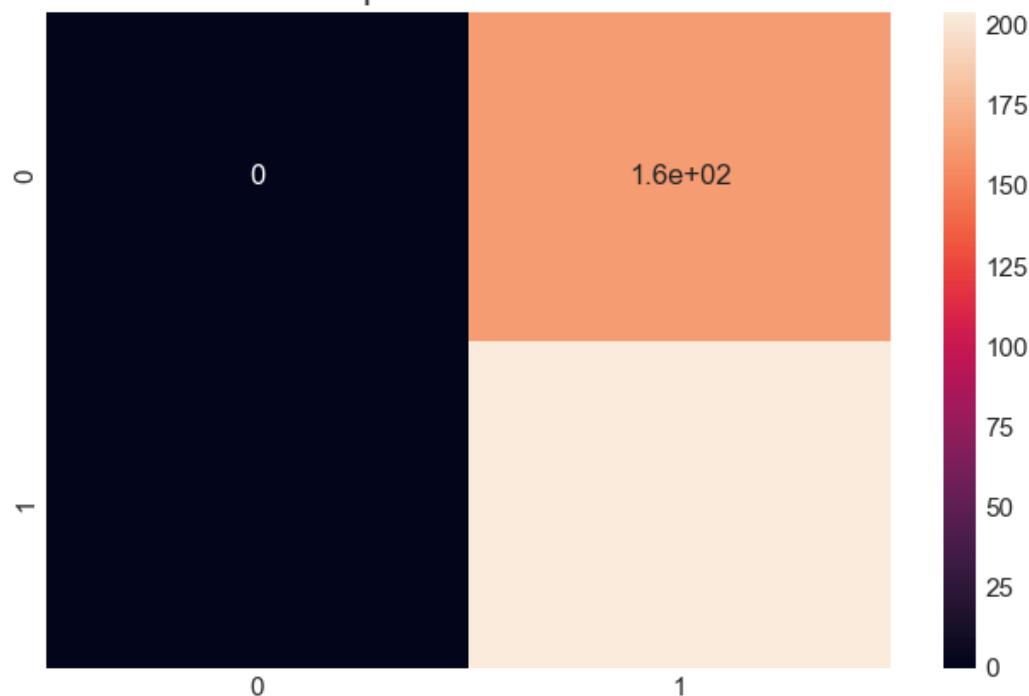
Classes+Metrics	precision	recall	f1-score	support	AUC
class 0	0.0	0.0	0.0	162.0	0.782
class 1	0.559	1.0	0.717	205.0	0.782
accuracy	0.559	0.559	0.559	0.559	0.782
macro avg	0.279	0.5	0.358	367.0	0.782
weighted avg	0.312	0.559	0.4	367.0	0.782

© Dr. Alexander Wagner. Все права охраняются законом

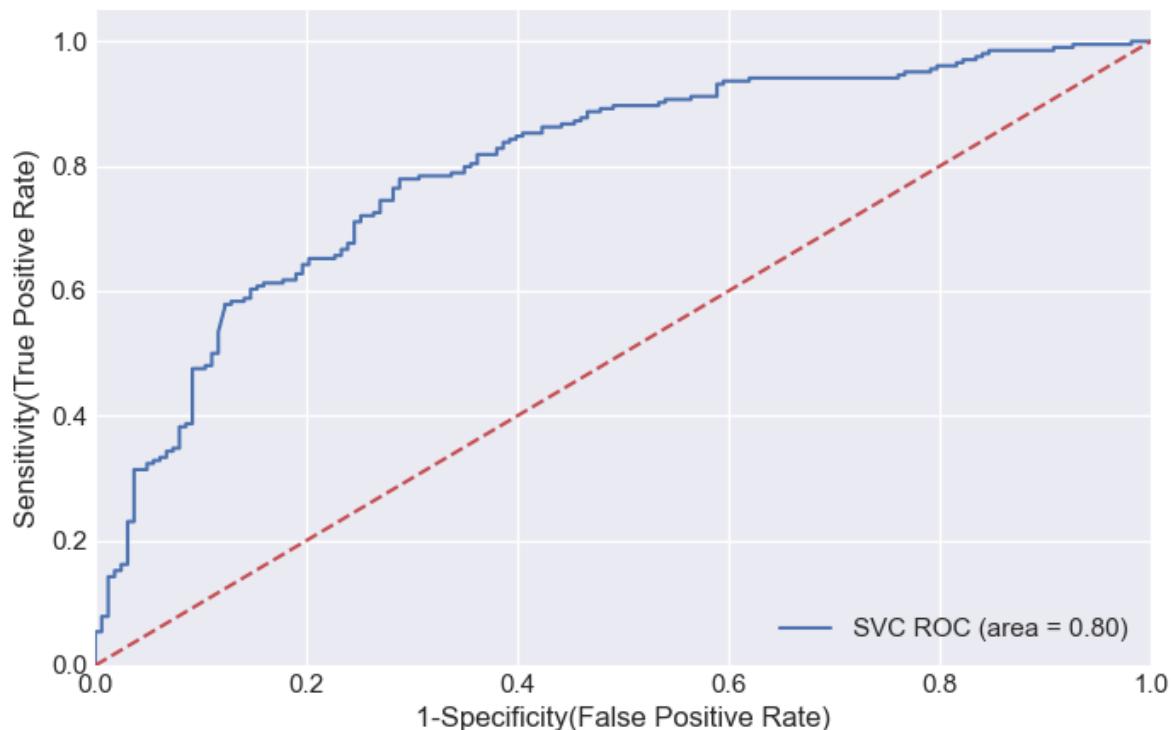
Confusion Matrix



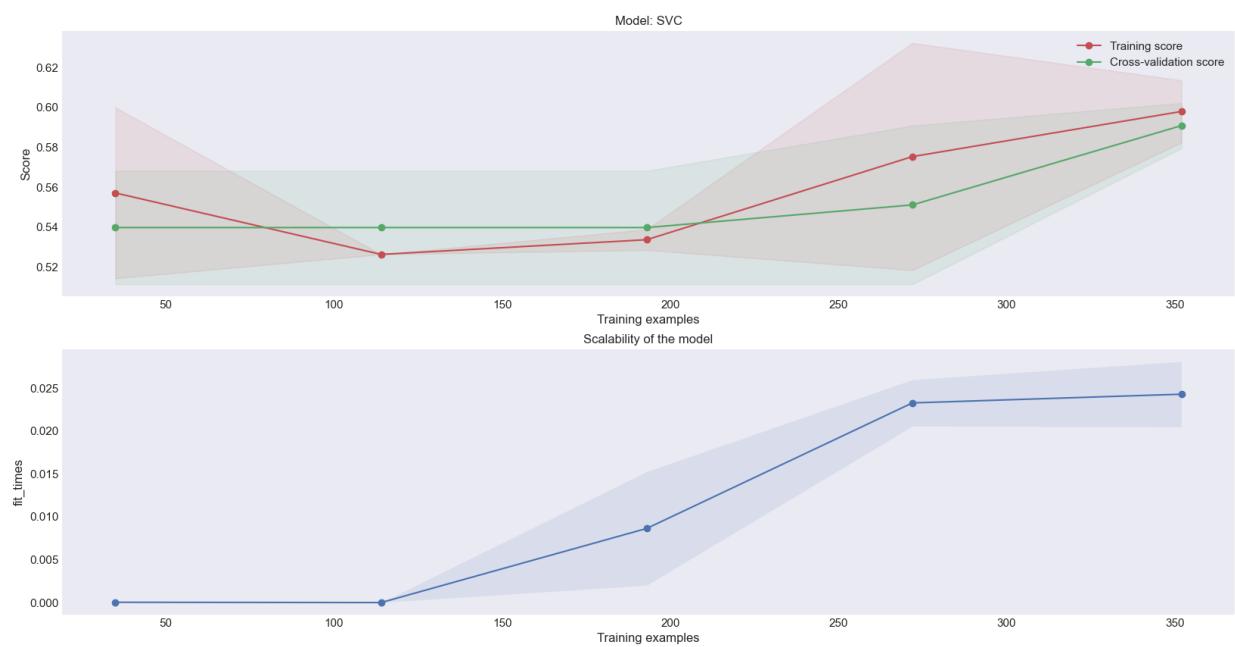
Heatmap of Confusion Matrix



ROC Curve



Score plot





Результаты моделирования

Моделирование осуществляется при помощи методов машинного обучения. Для исследования проблемы нами выбраны 18 моделей в том числе:

- Linear Regression
- Logistic Regression
- Perceptron
- Linear SVC
- MLPClassifier
- Decision Tree Classifier 1
- Stochastic Gradient Decent
- RidgeClassifier
- BaggingClassifier
- AdaBoostClassifier 1
- GradientBoostingClassifie
- KNeighborsClassifier
- DecisionTreeClassifier 2
- RandomForestClassifier
- XGBClassifier
- AdaBoostClassifier 2
- Naive Bayes
- SVC

В результате работы каждой модели выдаются следующая информация в форме таблиц и графиков:

- Таблица классификация
- Матрица Confusion Matrix
- ROC график
- Score график

Таблица классификация даёт оценку модели по критериям:

- *precision*
- *recall*
- *f1-score*

ROC график дает оценку модели по критерию AUC, т.е. значению площади под кривой. Что означает – чем ближе этот показатель к значению равному 1, тем выше соответствие прогнозных значений модели реальным входным данным.

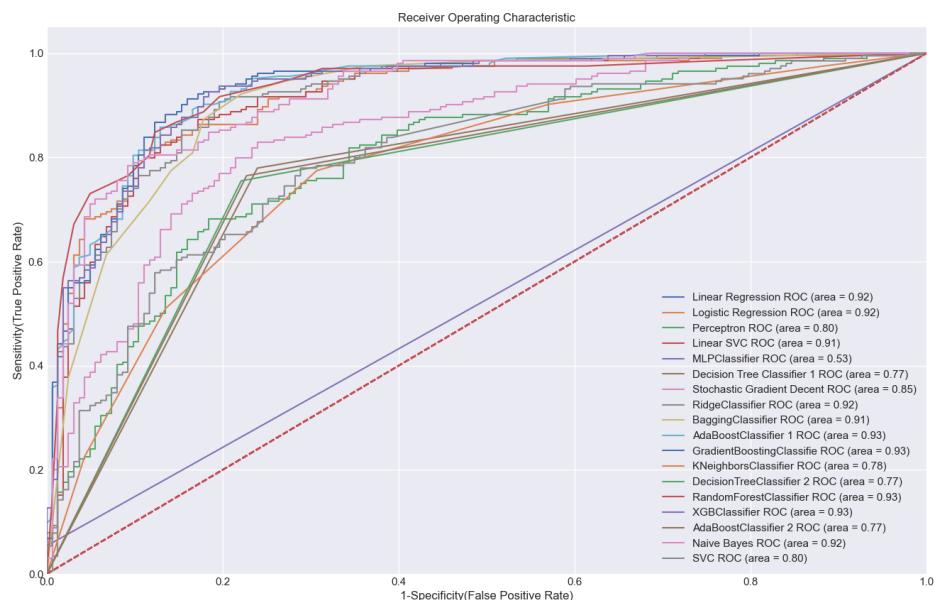
Исследовательский анализ включает в себя изучение данных и поиск связей между переменными, которые ранее были неизвестны. Вот что вам нужно знать



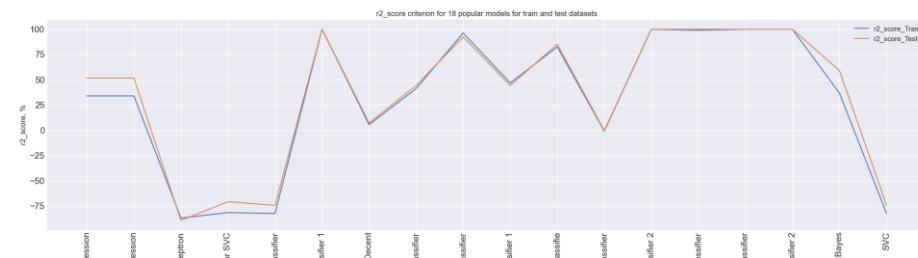
Оценка моделей и рекомендации

В результате анализа (ЕДА) и моделирования получена оценка исходных данных и характеристики 18 моделей машинного обучения. Результаты представлены в виде таблиц и графиков, объединенный график ROC приведен Рис. 19, обобщенные таблицы 19-21 показывают сравнительные характеристики моделей.

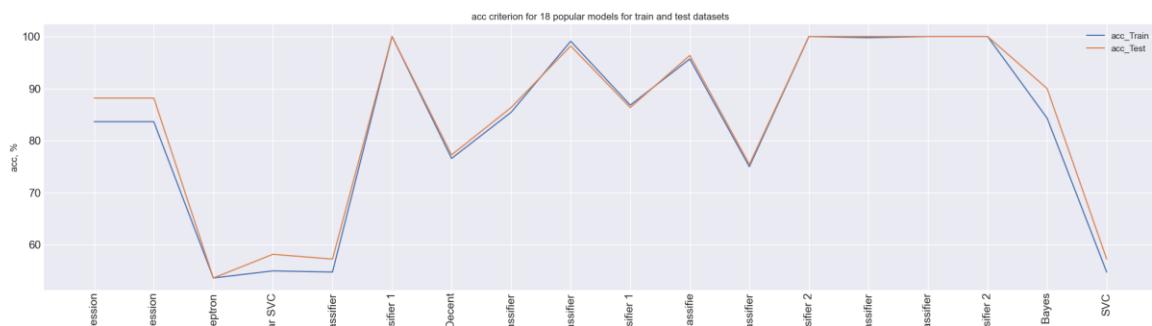
Общий Roc-график для всех моделей



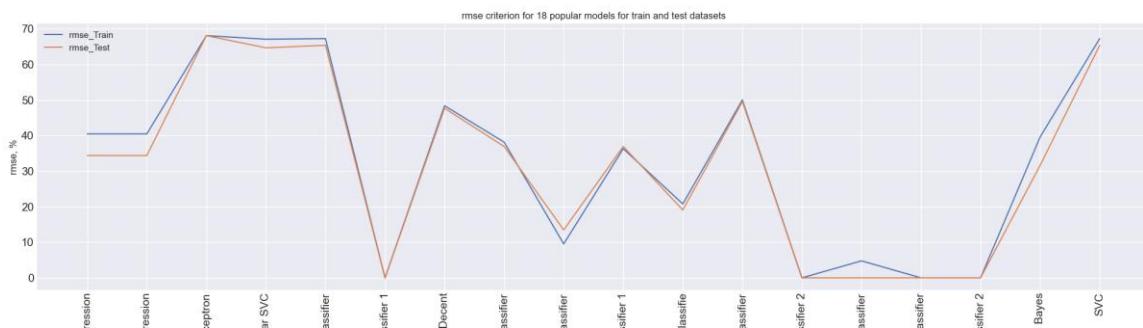
Линейный график №1 для всех моделей



Линейный график №2 для всех моделей



Линейный график №3 для всех моделей



Линейный график №4 для всех моделей

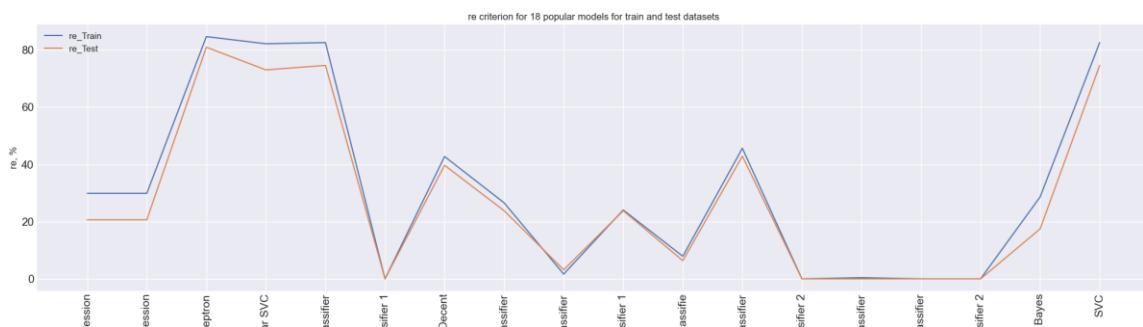


Table №19. Характеристика всех моделей после первого этапа

Model	r2_score	r2_score	acc_Tra	acc_Tes	acc_Diff	rmse_Tr	rmse_Te	re_Trai	re_Test
Decision Tree	100.0	100.0	100.0	100.0	0.0	0.0	0.0	0.0	0.0
DecisionTreeClass	100.0	100.0	100.0	100.0	0.0	0.0	0.0	0.0	0.0
XGBClassifier	100.0	100.0	100.0	100.0	0.0	0.0	0.0	0.0	0.0
AdaBoostClassifie	100.0	100.0	100.0	100.0	0.0	0.0	0.0	0.0	0.0
RandomForestClas	99.08	100.0	99.77	100.0	-	4.77	0.0	0.42	0.0
BaggingClassifier	95.42	88.64	98.86	97.27	1.59000	10.66	16.51	2.09	4.55
GradientBoosting	81.68	88.64	95.45	97.27	-	21.32	16.51	8.37	4.55
AdaBoostClassifie	62.45	69.7	90.68	92.73	-	30.53	26.97	17.15	12.12

© Dr. Alexander Wagner. Все права охраняются законом

Table №20. Характеристика всех моделей после второго этапа

Model	acc_train
RandomForestClas	100.0
Decision Tree	100.0
DecisionTreeClass	100.0
BaggingClassifier	99.09
GradientBoosting	95.82
XGBClassifier	91.27
AdaBoostClassifie	91.09
AdaBoostClassifie	91.09
RidgeClassifier	85.09
Logistic	84.73
Linear Regression	84.18
KNeighborsClassif	78.18
Linear SVC	55.45



<i>Model</i>	<i>acc_train</i>
SVC	55.45
Perceptron	54.36
Stochastic	49.27
MLPClassifier	44.55
© Dr. Alexander Wagner. Все права охраняются законом	

Table №21. Характеристика всех моделей после третьего этапа

<i>Model</i>	<i>r2_score_train</i>	<i>acc_train</i>	<i>rmse_train</i>	<i>re_train</i>
RandomForestClas	100.0	100.0	0.0	0.0
Decision Tree	100.0	100.0	0.0	0.0
DecisionTreeClass	100.0	100.0	0.0	0.0
BaggingClassifier	96.32	99.09	9.53	1.64
GradientBoosting	83.07	95.82	20.45	7.54
XGBClassifier	64.67	91.27	29.54	15.74
AdaBoostClassifie	63.93	91.09	29.85	16.07
AdaBoostClassifie	63.93	91.09	29.85	16.07
RidgeClassifier	39.65	85.09	38.61	26.89
Logistic	38.17	84.73	39.08	27.54
Linear Regression	35.97	84.18	39.77	28.52
KNeighborsClassif	11.68	78.18	46.71	39.34
Linear SVC	-80.33	55.45	66.74	80.33
SVC	-80.33	55.45	66.74	80.33
Perceptron	-84.74	54.36	67.55	82.3
Stochastic	-105.35	49.27	71.22	91.48
MLPClassifier	-124.49	44.55	74.47	100.0
© Dr. Alexander Wagner. Все права охраняются законом				

- Es ist Variable val1: **1234567**
- Es ist Variable val2: **7654321№**



Обсуждение и выводы

В данной работе предложена модель ансамбля мягкого голосования для раннего прогнозирования и диагностики сегрегации случаев МАСЭ на основе инфаркта миокарда с подъемом сегмента ST (ИМпST) и инфаркта миокарда без подъема сегмента ST (ИМнST) у пациентов с острым коронарным синдромом в течение 2-летнего клинического наблюдения после выписки из стационара. Следовательно, эффективность классификатора ансамбля мягкого голосования для прогнозирования возникновения МАСЭ в течение двухлетнего наблюдения у пациентов с острым коронарным синдромом была достоверно выше, чем у других моделей машинного обучения (RF, ET, GBM), а его основные прогностические факторы отличались. Наконец, этот ансамблевый классификатор на основе машинного обучения может привести к разработке прогностической модели оценки риска у пациентов с сердечно-сосудистыми заболеваниями в будущем.



Заключение

Системы и модели поддержки принятия решений на основе машинного обучения для раннего прогнозирования и диагностики находят широкое применение в здравоохранении. Эти системы помогают пациентам и медицинскому персоналу улучшить процесс принятия решений и раннее прогнозирование возникновения МАСЕ у пациентов с острым инфарктом миокарда. По сравнению с другими известными алгоритмами и системами прогнозирования, мы обнаружили, что алгоритмы машинного обучения лучше работают в прогнозировании и диагностике МАСЕ. Лучшими алгоритмами машинного обучения стали случайный лес, дополнительное дерево и машина для градиентного бустинга. Другие модели прогнозирования, основанные на машинном обучении, также были протестированы, но они показали худшие результаты, а их точность была меньше, чем у этих моделей, поэтому эти три модели были доработаны для нашего исследования и применены эти модели. В отличие от других моделей прогнозирования риска и ранней диагностики, модели, основанные на машинном обучении, работали с большим набором факторов риска, а также учитывали факторы риска, используемые в предыдущих моделях прогнозирования риска.

В данной статье мы применили алгоритмы машинного обучения для раннего прогнозирования и диагностики МАСЕ у пациентов с острым коронарным синдромом и использовали для экспериментов медицинский датасет за 2 года. Производительность этих моделей была сравнена с нашей моделью ансамбля мягкого голосования на основе машинного обучения. По результатам эксперимента мы обнаружили, что производительность нашего классификатора ансамбля мягкого голосования превзошла производительность других моделей машинного обучения. Кроме того, прогностические факторы для классификатора ансамбля мягкого голосования отличались от регрессионных моделей. Прогностические факторы в нашей модели включали прогностические факторы в предыдущих моделях машинного обучения, а также недавно добавленные прогностические факторы (например, артериальное давление, ИМТ и т. д.). Согласно результатам эксперимента, результаты прогнозирования нашего классификатора ансамбля мягкого голосования были достоверно выше, чем у других моделей машинного обучения в группах ИМпСТ и ИМпСТ у пациентов с острым коронарным синдромом в AUC, точность, запоминаемость, F-оценка и точность (Таблицы Таблицы 5–10).

Матрица несоответствий показала, что классификатор ансамбля мягкого голосования превзошел результаты и удовлетворительно предсказывает все классы, кроме инфаркта миокарда. Причина этой неправильной классификации заключалась в том, что она содержала зашумленные данные, а также содержала выбросы, поэтому предложенная нами модель, как и другие модели машинного обучения, не могли точно предсказать это сердечное событие с высокой



Литература

- [1] W. Guan et al., “Clinical characteristics of coronavirus disease 2019 in China,” *New England Journal of Medicine*, 2020, doi: [10.1056/NEJMoa2002032](https://doi.org/10.1056/NEJMoa2002032).
- [2] WHO Team, “Report of the WHO-China joint mission on coronavirus disease 2019 (COVID-19).” Available: [https://www.who.int/publications-detail/report-of-the-who-china-joint-mission-on-coronavirus-disease-2019-\(covid-19\)](https://www.who.int/publications-detail/report-of-the-who-china-joint-mission-on-coronavirus-disease-2019-(covid-19))
- [3] Center for Systems Science and Engineering (CSSE), “COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University.” 2020. Available: <https://github.com/CSSEGISandData/COVID-19>
- [4] J. Textor, B. van der Zander, M. S. Gilthorpe, M. Liśkiewicz, and G. T. Ellison, “Robust causal inference using directed acyclic graphs: The R package ‘dagitty’,” *International Journal of Epidemiology*, vol. 45, no. 6, pp. 1887–1894, Jan. 2017, doi: [10.1093/ije/dyw341](https://doi.org/10.1093/ije/dyw341).
- [5] S. Schipf, S. Knüppel, J. Hardt, and A. Stang, “Directed Acyclic Graphs (DAGs) – Die Anwendung kausaler Graphen in der Epidemiologie,” *Gesundheitswesen*, vol. 73, no. 12, pp. 888–892, Dec. 2011, doi: [10.1055/s-0031-1291192](https://doi.org/10.1055/s-0031-1291192).
- [6] S. Greenland, J. M. Robins, and J. Pearl, “Confounding and Collapsibility in Causal Inference,” *Statistical Science*, vol. 14, no. 1, pp. 29–46, 1999, Available: <http://www.jstor.org/stable/2676645>
- [7] M. Chinazzi et al., “The effect of travel restrictions on the spread of the 2019 novel coronavirus (COVID-19) outbreak,” *Science*, vol. 368, no. 6489, pp. 395–400, 2020, doi: [10.1126/science.aba9757](https://doi.org/10.1126/science.aba9757).
- [8] M. U. G. Kraemer et al., “The effect of human mobility and control measures on the COVID-19 epidemic in China.” *Science (New York, N.Y.)*, vol. 368, no. 6490, pp. 493–497, May 2020, doi: [10.1126/science.abb4218](https://doi.org/10.1126/science.abb4218).
- [9] K. Linka, M. Peirlinck, F. Sahli Costabal, and E. Kuhl, “Outbreak dynamics of COVID-19 in Europe and the effect of travel restrictions.” *Computer methods in biomechanics and biomedical engineering*, pp. 1–8, May 2020, doi: [10.1080/10255842.2020.1759560](https://doi.org/10.1080/10255842.2020.1759560).
- [10] C. Santana, F. Botta, H. Barbosa, F. Privitera, R. Menezes, and R. Di Clemente, “Analysis of human mobility in the UK during the COVID-19 pandemic,” 2020.
- [11] S. Engle, J. Stromme, and A. Zhou, “Staying at home: Mobility effects of COVID-19,” Available at SSRN, 2020, Available: <http://dx.doi.org/10.2139/ssrn.3565703>
- [12] M. Mazzoli, D. Mateo, A. Hernando, S. Meloni, and J. J. Ramasco, “Effects of mobility and multi-seeding on the propagation of the COVID-19 in Spain,” *medRxiv*, p. 2020.05.09.20096339, Jan. 2020, doi: [10.1101/2020.05.09.20096339](https://doi.org/10.1101/2020.05.09.20096339).
- [13] G. A. Wellenius et al., “Impacts of State-Level Policies on Social Distancing in the United States Using Aggregated Mobility Data during the COVID-19 Pandemic,” *arXiv preprint arXiv:2004.10172*, 2020.
- [14] F. C. Coelho et al., “Assessing the potential impact of COVID-19 in Brazil: Mobility, Morbidity and the burden on the Health Care System,” *medRxiv*, p. 2020.03.19.20039131, Jan. 2020, doi: [10.1101/2020.03.19.20039131](https://doi.org/10.1101/2020.03.19.20039131).



- [15] A. Lasry *et al.*, “Timing of community mitigation and changes in reported COVID-19 and community mobility - four U.S. Metropolitan areas, February 26-April 1, 2020,” *MMWR. Morbidity and mortality weekly report*, vol. 69, no. 15, p. 451—457, 2020, doi: [10.15585/mmwr.mm6915e2](https://doi.org/10.15585/mmwr.mm6915e2).
- [16] C. Xiong *et al.*, “Data-Driven Modeling Reveals the Impact of Stay-at-Home Orders on Human Mobility during the COVID-19 Pandemic in the U.S.,” *arXiv e-prints*, p. arXiv:2005.00667, May 2020, Available: <https://arxiv.org/abs/2005.00667>
- [17] R. Goel and R. Sharma, “Mobility Based SIR Model For Pandemics–With Case Study Of COVID-19,” *arXiv preprint arXiv:2004.13015*, 2020.
- [18] L. Zhang *et al.*, “AN INTERACTIVE COVID-19 MOBILITY IMPACT AND SOCIAL DISTANCING ANALYSIS PLATFORM,” *medRxiv*, p. 2020.04.29.20085472, Jan. 2020, doi: [10.1101/2020.04.29.20085472](https://doi.org/10.1101/2020.04.29.20085472).
- [19] M. S. Warren and S. W. Skillman, “Mobility changes in response to COVID-19,” *arXiv preprint arXiv:2003.14228*, 2020.
- [20] A. Scala *et al.*, “Between Geography and Demography: Key Interdependencies and Exit Mechanisms for Covid-19,” Available at SSRN 3572141, 2020.
- [21] J. H. Fowler, S. J. Hill, N. Obradovich, and R. Levin, “The Effect of Stay-at-Home Orders on COVID-19 Cases and Fatalities in the United States,” *medRxiv*, 2020, doi: [10.1101/2020.04.13.20063628](https://doi.org/10.1101/2020.04.13.20063628).
- [22] S. Lai *et al.*, “Effect of non-pharmaceutical interventions to contain COVID-19 in China,” *Nature*, May 2020, doi: [10.1038/s41586-020-2293-x](https://doi.org/10.1038/s41586-020-2293-x).
- [23] M.-C. Chang, R. Kahn, Y.-A. Li, C.-S. Lee, C. O. Buckee, and H.-H. Chang, “Modeling the impact of human mobility and travel restrictions on the potential spread of SARS-CoV-2 in Taiwan,” *medRxiv*, p. 2020.04.07.20053439, Jan. 2020, doi: [10.1101/2020.04.07.20053439](https://doi.org/10.1101/2020.04.07.20053439).
- [24] J. Liu *et al.*, “Impact of meteorological factors on the COVID-19 transmission: A multi-city study in China.” *The Science of the total environment*, vol. 726, p. 138513, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138513](https://doi.org/10.1016/j.scitotenv.2020.138513).
- [25] M. Jahangiri, M. Jahangiri, and M. Najafgholipour, “The sensitivity and specificity analyses of ambient temperature and population size on the transmission rate of the novel coronavirus (COVID-19) in different provinces of Iran.” *The Science of the total environment*, vol. 728, p. 138872, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138872](https://doi.org/10.1016/j.scitotenv.2020.138872).
- [26] J. Xie and Y. Zhu, “Association between ambient temperature and COVID-19 infection in 122 cities from China.” *The Science of the total environment*, vol. 724, p. 138201, Jul. 2020, doi: [10.1016/j.scitotenv.2020.138201](https://doi.org/10.1016/j.scitotenv.2020.138201).
- [27] Y. Ma *et al.*, “Effects of temperature variation and humidity on the death of COVID-19 in Wuhan, China.” *The Science of the total environment*, vol. 724, p. 138226, Jul. 2020, doi: [10.1016/j.scitotenv.2020.138226](https://doi.org/10.1016/j.scitotenv.2020.138226).
- [28] R. Tosepu *et al.*, “Correlation between weather and Covid-19 pandemic in Jakarta, Indonesia.” *The Science of the total environment*, vol. 725, p. 138436, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138436](https://doi.org/10.1016/j.scitotenv.2020.138436).



- [29] Á. Briz-Redón and Á. Serrano-Aroca, “A spatio-temporal analysis for exploring the effect of temperature on COVID-19 early evolution in Spain.” *The Science of the total environment*, vol. 728, p. 138811, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138811](https://doi.org/10.1016/j.scitotenv.2020.138811).
- [30] N. Iqbal, Z. Fareed, F. Shahzad, X. He, U. Shahzad, and M. Lina, “The nexus between COVID-19, temperature and exchange rate in Wuhan city: New findings from partial and multiple wavelet coherence.” *The Science of the total environment*, vol. 729, p. 138916, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138916](https://doi.org/10.1016/j.scitotenv.2020.138916).
- [31] M. F. Bashir et al., “Correlation between climate indicators and COVID-19 pandemic in New York, USA.” *The Science of the total environment*, vol. 728, p. 138835, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138835](https://doi.org/10.1016/j.scitotenv.2020.138835).
- [32] C. Del Rio and A. Camacho-Ortiz, “Will environmental changes in temperature affect the course of COVID-19?” *The Brazilian journal of infectious diseases : an official publication of the Brazilian Society of Infectious Diseases*, May 2020, doi: [10.1016/j.bjid.2020.04.007](https://doi.org/10.1016/j.bjid.2020.04.007).
- [33] Y. Yao et al., “No association of COVID-19 transmission with temperature or UV radiation in Chinese cities.” *The European respiratory journal*, vol. 55, no. 5, May 2020, doi: [10.1183/13993003.00517-2020](https://doi.org/10.1183/13993003.00517-2020).
- [34] M. Ujiie, S. Tsuzuki, and N. Ohmagari, “Effect of temperature on the infectivity of COVID-19.” *International Journal of Infectious Diseases*, vol. 95, pp. 301–303, Apr. 2020, doi: [10.1016/j.ijid.2020.04.068](https://doi.org/10.1016/j.ijid.2020.04.068).
- [35] P. Mecenas, R. Bastos, A. Vallinoto, and D. Normando, “Effects of temperature and humidity on the spread of COVID-19: A systematic review.” *medRxiv*, p. 2020.04.14.20064923, Jan. 2020, doi: [10.1101/2020.04.14.20064923](https://doi.org/10.1101/2020.04.14.20064923).
- [36] A. Vantarakis, I. Chatziprodromidou, and T. Apostolou, “COVID-19 and Environmental factors. A PRISMA-compliant systematic review,” *medRxiv*, p. 2020.05.10.20069732, Jan. 2020, doi: [10.1101/2020.05.10.20069732](https://doi.org/10.1101/2020.05.10.20069732).
- [37] M. F. F. Sobral, G. B. Duarte, A. I. G. da Penha Sobral, M. L. M. Marinho, and A. de Souza Melo, “Association between climate variables and global transmission of SARS-CoV-2.” *Science of the Total Environment*, vol. 729, p. 138997, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138997](https://doi.org/10.1016/j.scitotenv.2020.138997).
- [38] H. Eslami and M. Jalili, “The role of environmental factors to transmission of SARS-CoV-2 (COVID-19).” *AMB Express*, vol. 10, no. 1, p. 92, May 2020, doi: [10.1186/s13568-020-01028-0](https://doi.org/10.1186/s13568-020-01028-0).
- [39] Y. Wu et al., “Effects of temperature and humidity on the daily new cases and new deaths of COVID-19 in 166 countries,” *Science of The Total Environment*, vol. 729, p. 139051, 2020, doi: <https://doi.org/10.1016/j.scitotenv.2020.139051>.
- [40] S. Gupta, G. S. Raghuwanshi, and A. Chanda, “Effect of weather on COVID-19 spread in the US: A prediction model for India in 2020.” *The Science of the total environment*, vol. 728, p. 138860, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138860](https://doi.org/10.1016/j.scitotenv.2020.138860).
- [41] M. Şahin, “Impact of weather on COVID-19 pandemic in Turkey.” *The Science of the total environment*, vol. 728, p. 138810, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138810](https://doi.org/10.1016/j.scitotenv.2020.138810).
- [42] P. Jüni et al., “Impact of climate and public health interventions on the COVID-19 pandemic: A prospective cohort study.” *Canadian Medical Association Journal*, May 2020, doi: [10.1503/cmaj.200920](https://doi.org/10.1503/cmaj.200920).



- [43] Y. Jiang, X.-J. Wu, and Y.-J. Guan, "Effect of ambient air pollutants and meteorological variables on COVID-19 incidence." *Infection control and hospital epidemiology*, pp. 1–11, May 2020, doi: [10.1017/ice.2020.222](https://doi.org/10.1017/ice.2020.222).
- [44] B. Pirouz, S. Shaffiee Haghshenas, B. Pirouz, S. Shaffiee Haghshenas, and P. Piro, "Development of an Assessment Method for Investigating the Impact of Climate and Urban Parameters in Confirmed Cases of COVID-19: A New Challenge in Sustainable Development." *International journal of environmental research and public health*, vol. 17, no. 8, Apr. 2020, doi: [10.3390/ijerph17082801](https://doi.org/10.3390/ijerph17082801).
- [45] A. C. Auler, F. A. M. Cássaro, V. O. da Silva, and L. F. Pires, "Evidence that high temperatures and intermediate relative humidity might favor the spread of COVID-19 in tropical climate: A case study for the most affected Brazilian cities." *The Science of the total environment*, vol. 729, p. 139090, Apr. 2020, doi: [10.1016/j.scitotenv.2020.139090](https://doi.org/10.1016/j.scitotenv.2020.139090).
- [46] M. P. Ward, S. Xiao, and Z. Zhang, "The Role of Climate During the COVID-19 epidemic in New South Wales, Australia." *Transboundary and emerging diseases*, May 2020, doi: [10.1111/tbed.13631](https://doi.org/10.1111/tbed.13631).
- [47] H. Qi *et al.*, "COVID-19 transmission in Mainland China is associated with temperature and humidity: A time-series analysis." *The Science of the total environment*, vol. 728, p. 138778, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138778](https://doi.org/10.1016/j.scitotenv.2020.138778).
- [48] D. N. Prata, W. Rodrigues, and P. H. Bermejo, "Temperature significantly changes COVID-19 transmission in (sub)tropical cities of Brazil." *The Science of the total environment*, vol. 729, p. 138862, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138862](https://doi.org/10.1016/j.scitotenv.2020.138862).
- [49] P. Shi *et al.*, "Impact of temperature on the dynamics of the COVID-19 outbreak in China." *The Science of the total environment*, vol. 728, p. 138890, Apr. 2020, doi: [10.1016/j.scitotenv.2020.138890](https://doi.org/10.1016/j.scitotenv.2020.138890).
- [50] J. Demongeot, Y. Flet-Berliac, and H. Seligmann, "Temperature Decreases Spread Parameters of the New Covid-19 Case Dynamics." *Biology*, vol. 9, no. 5, May 2020, doi: [10.3390/biology9050094](https://doi.org/10.3390/biology9050094).
- [51] M. Effenberger, A. Kronbichler, J. I. Shin, G. Mayer, H. Tilg, and P. Perco, "Association of the COVID-19 pandemic with Internet Search Volumes: A Google Trends(TM) Analysis." *International journal of infectious diseases : IJID : official publication of the International Society for Infectious Diseases*, vol. 95, pp. 192–197, Apr. 2020, doi: [10.1016/j.ijid.2020.04.033](https://doi.org/10.1016/j.ijid.2020.04.033).
- [52] Y.-H. Lin, C.-H. Liu, and Y.-C. Chiu, "Google searches for the keywords of "wash hands" predict the speed of national spread of COVID-19 outbreak among 21 countries." *Brain, behavior, and immunity*, Apr. 2020, doi: [10.1016/j.bbi.2020.04.020](https://doi.org/10.1016/j.bbi.2020.04.020).
- [53] A. Walker, C. Hopkins, and P. Surda, "Use of Google Trends to investigate loss-of-smell-related searches during the COVID-19 outbreak," *International Forum of Allergy & Rhinology*, vol. 10, no. 7, pp. 839–847, 2020, doi: [10.1002/alr.22580](https://doi.org/10.1002/alr.22580).
- [54] T. S. Higgins, A. W. Wu, D. Sharma, E. A. Illing, K. Rubel, and J. Y. Ting, "Correlations of Online Search Engine Trends With Coronavirus Disease (COVID-19) Incidence: Infodemiology Study." *JMIR public health and surveillance*, vol. 6, no. 2, p. e19702, May 2020, doi: [10.2196/19702](https://doi.org/10.2196/19702).



- [55] S. M. Ayyoubzadeh, S. M. Ayyoubzadeh, H. Zahedi, M. Ahmadi, and S. R Niakan Kalhor, “Predicting COVID-19 Incidence Through Analysis of Google Trends Data in Iran: Data Mining and Deep Learning Pilot Study.” *JMIR Public Health and Surveillance*, vol. 6, no. 2, p. e18828, Apr. 2020, doi: [10.2196/18828](https://doi.org/10.2196/18828).
- [56] X. Yuan, J. Xu, S. Hussain, H. Wang, N. Gao, and L. Zhang, “Trends and Prediction in Daily New Cases and Deaths of COVID-19 in the United States: An Internet Search-Interest Based Model.” *Exploratory research and hypothesis in medicine*, vol. 5, no. 2, pp. 1–6, Apr. 2020, doi: [10.14218/ERHM.2020.00023](https://doi.org/10.14218/ERHM.2020.00023).
- [57] A. Mavragani, “Tracking COVID-19 in Europe: Infodemiology Approach.” *JMIR public health and surveillance*, vol. 6, no. 2, p. e18941, Apr. 2020, doi: [10.2196/18941](https://doi.org/10.2196/18941).
- [58] D. Hu *et al.*, “More effective strategies are required to strengthen public awareness of COVID-19: Evidence from Google Trends.” *Journal of global health*, vol. 10, no. 1, p. 011003, Jun. 2020, doi: [10.7189/jogh.10.011003](https://doi.org/10.7189/jogh.10.011003).
- [59] Y. Ortiz-Martínez, J. E. Garcia-Robled, D. L. Vásquez-Castañeda, D. K. Bonilla-Aldana, and A. J. Rodriguez-Morales, “Can Google® trends predict COVID-19 incidence and help preparedness? The situation in Colombia.” *Travel medicine and infectious disease*, p. 101703, Apr. 2020, doi: [10.1016/j.tmaid.2020.101703](https://doi.org/10.1016/j.tmaid.2020.101703).
- [60] C. Li, L. J. Chen, X. Chen, M. Zhang, C. P. Pang, and H. Chen, “Retrospective analysis of the possibility of predicting the COVID-19 outbreak from Internet searches and social media data, China, 2020.” *Euro surveillance : bulletin European sur les maladies transmissibles = European communicable disease bulletin*, vol. 25, no. 10, Mar. 2020, doi: [10.2807/1560-7917.ES.2020.25.10.2000199](https://doi.org/10.2807/1560-7917.ES.2020.25.10.2000199).
- [61] A. I. Bento, T. Nguyen, C. Wing, F. Lozano-Rojas, Y.-Y. Ahn, and K. Simon, “Evidence from internet search data shows information-seeking responses to news of local COVID-19 cases.” *Proceedings of the National Academy of Sciences of the United States of America*, May 2020, doi: [10.1073/pnas.2005335117](https://doi.org/10.1073/pnas.2005335117).
- [62] S. Springer, L. M. Menzel, and M. Zieger, “Google Trends provides a tool to monitor population concerns and information needs during COVID-19 pandemic.” *Brain, behavior, and immunity*, Apr. 2020, doi: [10.1016/j.bbi.2020.04.073](https://doi.org/10.1016/j.bbi.2020.04.073).
- [63] W. K. Zhou, A. L. Wang, F. Xia, Y. N. Xiao, and S. Y. Tang, “Effects of media reporting on mitigating spread of COVID-19 in the early phase of the outbreak.” *Mathematical biosciences and engineering : MBE*, vol. 17, no. 3, pp. 2693–2707, Mar. 2020, doi: [10.3934/mbe.2020147](https://doi.org/10.3934/mbe.2020147).
- [64] M. Bannister-Tyrrell, A. Meyer, C. Faverjon, and A. Cameron, “Preliminary evidence that higher temperatures are associated with lower incidence of COVID-19, for cases reported globally up to 29th February 2020,” *medRxiv*, p. 2020.03.18.20036731, Jan. 2020, doi: [10.1101/2020.03.18.20036731](https://doi.org/10.1101/2020.03.18.20036731).
- [65] A. Strzelecki, “The second worldwide wave of interest in coronavirus since the COVID-19 outbreaks in South Korea, Italy and Iran: A Google Trends study,” *Brain, behavior, and immunity*, pp. S0889-1591(20)30551-1, Apr. 2020, doi: [10.1016/j.bbi.2020.04.042](https://doi.org/10.1016/j.bbi.2020.04.042).
- [66] M. Schröder, “AfD-Unterstützer sind nicht abgehängt, sondern ausländerfeindlich,” Deutsches Institut für Wirtschaftsforschung (DIW), Berlin, {SOEPpapers} on {Multidisciplinary} {Panel} {Data} {Research} 975, 2018. Available: <http://hdl.handle.net/10419/181028>



- [67] E. von Elm, G. Schreiber, and C. C. Haupt, “Methodische Anleitung für Scoping Reviews (JBI-Methodologie),” *Zeitschrift für Evidenz, Fortbildung und Qualität im Gesundheitswesen*, vol. 143, pp. 1–7, Jun. 2019, doi: [10.1016/j.zefq.2019.05.004](https://doi.org/10.1016/j.zefq.2019.05.004).
- [68] M. Wang *et al.*, “Temperature significant change COVID-19 transmission in 429 cities,” *medRxiv*, 2020, doi: [10.1101/2020.02.22.20025791](https://doi.org/10.1101/2020.02.22.20025791).
- [69] N. Dragano, C. J. Rupprecht, O. Dortmann, M. Scheider, and M. Wahrendorf, “Higher risk of COVID-19 hospitalization for unemployed: An analysis of 1,298,416 health insured individuals in Germany,” *medRxiv*, 2020, doi: [10.1101/2020.06.17.20133918](https://doi.org/10.1101/2020.06.17.20133918).
- [70] S. Dohle, T. Wingen, and M. Schreiber, “Acceptance and adoption of protective measures during the COVID-19 pandemic: The role of trust in politics and trust in science.” *OSF Preprints*, May 2020. doi: [10.31219/osf.io/w52nv](https://doi.org/10.31219/osf.io/w52nv).
- [71] S. de Lusignan *et al.*, “Risk factors for SARS-CoV-2 among patients in the Oxford Royal College of General Practitioners Research and Surveillance Centre primary care network: A cross-sectional study,” *The Lancet Infectious Diseases*, doi: [10.1016/S1473-3099\(20\)30371-6](https://doi.org/10.1016/S1473-3099(20)30371-6).
- [72] B. J. Cowling *et al.*, “Impact assessment of non-pharmaceutical interventions against coronavirus disease 2019 and influenza in Hong Kong: An observational study,” *The Lancet Public Health*, vol. 5, no. 5, pp. e279–e288, May 2020, doi: [10.1016/S2468-2667\(20\)30090-6](https://doi.org/10.1016/S2468-2667(20)30090-6).
- [73] S. Openshaw, “Ecological Fallacies and the Analysis of Areal Census Data,” *Environment and Planning A: Economy and Space*, vol. 16, no. 1, pp. 17–31, 1984, doi: [10.1068/a160017](https://doi.org/10.1068/a160017).
- [74] J. Pearl and E. Bareinboim, “External validity: From do-calculus to transportability across populations,” *Statistical Science*, vol. 29, no. 4, pp. 579–595, Nov. 2014, doi: [10.1214/14-sts486](https://doi.org/10.1214/14-sts486).
- [75] Google LLC, “Google Trends, search term "corona".” Accessed: Jun. 25, 2020. [Online]. Available: <https://www.google.com/trends>
- [76] Robert Koch-Institut (RKI), “Fallzahlen in Deutschland (COVID-19).” Available: https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Fallzahlen.html
- [77] Google LLC, “Google COVID-19 community mobility reports.” Accessed: Jun. 25, 2020. [Online]. Available: <https://www.google.com/covid19/mobility/>
- [78] Deutscher Wetterdienst (DWD) Climate Data Center (CDC), “Recent daily station observations (temperature, pressure, precipitation, sunshine duration, etc.) For Germany, quality control not completed yet, version recent.” Accessed: Jun. 25, 2020. [Online]. Available: https://opendata.dwd.de/climate_environment/CDC/observations_germany/climate/daily/kl/recent/
- [79] Bundesinstitut für Bau-, Stadt- und Raumforschung (BBSR), “INKAR – Indikatoren und Karten zur Raum- und Stadtentwicklung.” Accessed: Jun. 25, 2020. [Online]. Available: <https://www.inkar.de/>
- [80] T. Mitze, R. Kosfeld, J. Rode, and K. Wälde, “Face masks considerably reduce COVID-19 cases in Germany: A synthetic control method approach,” Institute of Labor Economics (IZA), IZA Discussion Papers 13319, 2020. Available: <https://EconPapers.repec.org/RePEc:iza:izadps:dp13319>



- [81] O. Gencoglu and M. Gruber, “Causal modeling of twitter activity during COVID-19,” *medRxiv*, 2020, doi: [10.1101/2020.05.16.20103903](https://doi.org/10.1101/2020.05.16.20103903).
- [82] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, prediction, and search*. MIT press, 2000.
- [83] J. Pearl, *Causality*. Cambridge: Cambridge University Press, 2009. Available: <https://www.cambridge.org/core/books/causality/B0046844FAE10CBF274D4ACBDAEB5F5B>
- [84] S. Greenland, J. Pearl, and J. M. Robins, “Causal Diagrams for Epidemiologic Research,” *Epidemiology*, vol. 10, no. 1, pp. 37–48, 1999, Available: https://journals.lww.com/epidem/Fulltext/1999/01000/Causal_Diagrams_for_Epidemiologic_Research.8.aspx
- [85] L. Henckel, E. Perković, and M. H. Maathuis, “Graphical Criteria for Efficient Total Effect Estimation via Adjustment in Causal Linear Models,” *arXiv e-prints*, p. arXiv:1907.02435, Jul. 2019, Available: <https://arxiv.org/abs/1907.02435>
- [86] R Core Team, *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing, 2019. Available: <https://www.R-project.org/>
- [87] M. Kalisch, M. Mächler, D. Colombo, M. H. Maathuis, and P. Bühlmann, “Causal Inference Using Graphical Models with the R Package *pcalg*,” *Journal of Statistical Software*, vol. 47, no. 11, pp. 1–26, 2012, doi: [10.18637/jss.v047.i11](https://doi.org/10.18637/jss.v047.i11).
- [88] J. M. Hilbe and W. H. Greene, “4 - Count Response Regression Models,” in *Essential Statistical Methods for Medical Statistics*, C. R. Rao, J. P. Miller, and D. C. Rao, Eds., Boston: North-Holland, 2011, pp. 104–145. Available: <http://www.sciencedirect.com/science/article/pii/B9780444537379500074>
- [89] E. Perković, J. Textor, M. Kalisch, and M. H. Maathuis, “A Complete Generalized Adjustment Criterion,” *arXiv e-prints*, p. arXiv:1507.01524, Jul. 2015, Available: <https://arxiv.org/abs/1507.01524>
- [90] W. N. Venables and B. D. Ripley, *Modern applied statistics with s*, Fourth. New York: Springer, 2002. Available: <http://www.stats.ox.ac.uk/pub/MASS4>
- [91] W. O. Kermack and A. G. McKendrick, “Contributions to the mathematical theory of epidemics–i. 1927,” *Bulletin of mathematical biology*, vol. 53, no. 1–2, p. 33—55, 1991, doi: [10.1007/bf02464423](https://doi.org/10.1007/bf02464423).
- [92] M. an der Heiden and U. Buchholz, “Modellierung von Beispielszenarien der SARS-CoV-2-Epidemie 2020 in Deutschland,” 2020, doi: [10.25646/6571.2](https://doi.org/10.25646/6571.2).
- [93] H. R. Kunsch, “The jackknife and the bootstrap for general stationary observations,” *Ann. Statist.*, vol. 17, no. 3, pp. 1217–1241, Sep. 1989, doi: [10.1214/aos/1176347265](https://doi.org/10.1214/aos/1176347265).
- [94] C. A. Field and A. H. Welsh, “Bootstrapping clustered data,” *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 69, no. 3, pp. 369–390, 2007, Available: <http://www.jstor.org/stable/4623274>



Приложение

```
def getConfigFile(config):
    with open(config, encoding='utf-8') as json_file:
        return json.load(json_file)

def addTextParagraphToDocumentInStyle(text, document, style):
    p = document.add_paragraph(text)
    p.style = document.styles[style]

def AddBookmark(t, mark):
    # Find a specific text or phrase in the document
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
    HorizontalAlignment.Center
    text = paragraph.AppendText(t)
    text = document.FindString(t, False, True)
    # Get the found text as a single text range
    textRange = text.GetAsOneRange()
    # Get the paragraph where the text range is located
    paragraph = textRange.OwnerParagraph

    # Get the index position of the text in the paragraph
    index = paragraph.ChildObjects.IndexOf(textRange)

    # Add a bookmark start mark to the paragraph
    start = paragraph.AppendBookmarkStart(mark)
    # Insert the bookmark start mark at the index position of the
    text range
    paragraph.ChildObjects.Insert(index, start)
    # Add a bookmark end mark to the paragraph
    end = paragraph.AppendBookmarkEnd(mark)
    # Insert the bookmark end mark after the text range
    paragraph.ChildObjects.Insert(index + 2, end)

def update_toc(docx_file):
    word = win32com.client.DispatchEx("Word.Application")
    word.Visible = 1
    word.DisplayAlerts = 0

    doc = word.Documents.Open(docx_file)
    #wd_section = doc.Sections(1)
    toc_count = doc.TablesOfContents.Count
    print(toc_count)
    stringG='INHALTSVERZEICHNIS'
    stringK='Содержание'
    if toc_count == 0:
        for i, p in enumerate(doc.Paragraphs):
            if stringK in p.Range.Text:
                try:
                    p.Range.InsertParagraphAfter()
                    parag_range = doc.Paragraphs(i+2).Range
                    parag_range.Font.Name = 'Arial'
                    parag_range.Font.Size = 14
                    parag_range.Font.Bold = constants.wdToggle
                    parag_range.Font.Size = 12
                    doc.TablesOfContents.Add(Range=parag_range,
                                           UseHeadingStyles=True,
                                           LowerHeadingLevel=3)
                except Exception as e:
                    print("Ja : ", e, "Nein")
                    break
    elif toc_count == 1:
```



```
toc = doc.TablesOfContents(1)
toc.Update()
print('TOC should have been updated.')
else:
    print('TOC has not been updated for sure...')

doc.Close()
word.Quit()

def set_column_width(table, column, width_mm):
    table.allow_autofit = False
    for row in table.rows:
        row.cells[column].width = Mm(width_mm)

def set_repeat_table_header(row):
    tr = row._tr
    trPr = tr.get_or_add_trPr()
    tblHeader = OxmlElement('w:tblHeader')
    tblHeader.set(qn('w:val'), "true")
    trPr.append(tblHeader)
    return row

def change_table_cell(cell, background_color=None, font_color=None,
font_size=None, bold=None, italic=None):
    if background_color:
        shading_elm = parse_xml(r'<w:shd {}'
w:fill="{}"/>'.format(nsdecls('w'), background_color))
        cell._tc.get_or_add_tcPr().append(shading_elm)

    if font_color:
        for p in cell.paragraphs:
            for r in p.runs:
                r.font.color.rgb =
docx.shared.RGBColor.from_string(font_color)

    if font_size:
        for p in cell.paragraphs:
            for r in p.runs:
                r.font.size = docx.shared.Pt(font_size)

    if bold is not None:
        for p in cell.paragraphs:
            for r in p.runs:
                r.bold = bold

    if italic is not None:
        for p in cell.paragraphs:
            for r in p.runs:
                r.italic = italic

def set_cell_border(cell, **kwargs):
    tc = cell._tc
    tcPr = tc.get_or_add_tcPr()

    # check for tag existnace, if none found, then create one
    tcBorders = tcPr.first_child_found_in("w:tcBorders")
    if tcBorders is None:
        tcBorders = OxmlElement('w:tcBorders')
        tcPr.append(tcBorders)

    for edge in ('left', 'top', 'right', 'bottom', 'insideH',
'insideV'):
        edge_data = kwargs.get(edge)
        if edge_data:
```



```
tag = 'w:{}'.format(edge)

        # check for tag existnace, if none found, then create
one
        element = tcBorders.find(qn(tag))
        if element is None:
            element = OxmlElement(tag)
            tcBorders.append(element)

        # looks like order of attributes is important
        for key in ["sz", "val", "color", "space", "shadow"]:
            if key in edge_data:
                element.set(qn('w:{}').format(key)),
str(edge_data[key]))

def delete_paragraph(paragraph):
    p = paragraph._element
    p.getparent().remove(p)
    p._p = p._element = None

def replace_copy(file, txt):
    wordapp =
win32com.client.gencache.EnsureDispatch("Word.Application")
    wordapp.Visible = True
    newdoc = wordapp.Documents.Open(file)
    #print(newdoc.Paragraphs.Count)
    finder = wordapp.Selection.Find
    finder.Text = txt #"Hier"
    finder.Execute()
    #wordapp.Selection.MoveLeft()
    #wordapp.Selection.MoveDown()
    wordapp.Selection.MoveStart
    wordapp.Selection.Paste()
    newdoc.SaveAs("C:\\IPYNBgesamt\\ASNI-FEN\\ASNI-
Report\\ASNI_ReportResult.docx")
    newdoc .ActiveWindow.Close()
    wordapp.Application.Quit(-1)

def Text_copy(file):
    wordapp =
win32com.client.gencache.EnsureDispatch("Word.Application")
    wordapp.Visible = True
    worddoc = wordapp.Documents.Open(file)
    worddoc.Select()
    wordapp.Selection.Copy()
    worddoc.ActiveWindow.Close()
    wordapp.Application.Quit(-1)

# In[3]:
```



```
doc = DocxTemplate("rTemplateTest.docx")
reportWordPath = 'ASNI_ReportTest01.docx'

Inmodels = pd.DataFrame({'Model': [
    'Linear Regression',
    'Logistic Regression',
    'Perceptron',
    'Linear SVC',
    'MLPClassifier',
    'Decision Tree Classifier 1',
    'Stochastic Gradient Descent',
    'RidgeClassifier',
```



```
'BaggingClassifier',
'AdaBoostClassifier 1',
'GradientBoostingClassifier',
'KNeighborsClassifier',
'DecisionTreeClassifier 2',
'RandomForestClassifier',
'XGBClassifier',
'AdaBoostClassifier 2',
'Naive Bayes',
'SVC' ]})  
  
Nk=len(Inmodels)
print(Nk)  
  
Asni = {
    'Projekt': 'Asfendiyarov Kazakh National Medical University',
    'Projekt2': 'Statistik und Data Science Projekt',
    'Projekt3': 'Практическое применение Автоматизированной системы научных исследований в медицине, здравоохранении и смежных областях',
    'Thema': 'Анализ факторов риска сердечно сосудистых заболеваний и прогноз исходов лечения при помощи методов Машинного Обучения',
    'Forscher': 'Dr. Alexander Wagner (Berlin)',
    'Site' : 'Berlin-Almaty',
    'Year' : str(year),
    "tasks" : [
        {
            "folder" : "0",
            "include" : ["Book", "Dog"],
            "topic" : "Содержание",
            "models" : " ",
            "Text" : " "
        },
        {
            "folder" : "0",
            "include" : ["Book", "Dog"],
            "topic" : "Предисловие",
            "models" : "xHier",
            "Text" : "File"
        },
        {
            "folder" : "1",
            "include" : ["Book", "Dog"],
            "topic" : "Введение",
            "models" : " ",
            "Text" : "Kap01"
        },
        {
            "folder" : "2",
            "include" : ["Author", "Ball"],
            "topic" : "Цель исследования",
            "models" : " ",
            "Text" : "Kap02"
        },
        {
            "folder" : "3",
            "include" : ["Author", "Ball"],
            "topic" : "Материалы и методы",
            "models" : " ",
            "Text" : "Kap03"
        },
        {
            "folder" : "4",
            "include" : ["Author", "Ball"],
```



```
"topic" : "Исходные данные и их организация",
"models" : " ",
"Text" : "Kap04"
},
{
"folder" : "5",
"include" : ["Author", "Ball"],
"topic" : "Предварительный анализ данных",
"models" : " ",
"Text" : "Kap05"
},
{
"folder" : "6",
"include" : ["MovablePoint", "Rectangle"],
"topic" : "Моделирование",
"models" : [
    'Linear Regression',
    'Logistic Regression',
    'Perceptron',
    'Linear SVC',
    'MLPClassifier',
    'Decision Tree Classifier 1',
    'Stochastic Gradient Decent',
    'RidgeClassifier',
    'BaggingClassifier',
    'AdaBoostClassifier 1',
    'GradientBoostingClassifier',
    'KNeighborsClassifier',
    'DecisionTreeClassifier 2',
    'RandomForestClassifier',
    'XGBClassifier',
    'AdaBoostClassifier 2',
    'Naive Bayes',
    'SVC' ],
"Text" : "Kap06"
},
{
"folder" : "7",
"include" : ["Exercise", "TextAreaExample"],
"topic" : "Результаты моделирования",
"models" : " ",
"Text" : "Kap07"
},
{
"folder" : "8",
"include" : ["Exercise", "TextAreaExample"],
"topic" : "Оценка моделей и рекомендации",
"models" : " ",
"Text" : "Kap08"
},
{
"folder" : "9",
"include" : ["Exercise", "TextAreaExample"],
"topic" : "Обсуждение и выводы",
"models" : " ",
"Text" : "Kap09"
},
{
"folder" : "10",
"include" : ["Exercise", "TextAreaExample"],
"topic" : "Заключение",
"models" : " ",
"Text" : "Kap010"
},
```



```
{  
    "folder" : "11",  
    "include" : ["Exercise", "TextAreaExample"],  
    "topic" : "Литература",  
    "models" : "xLITER",  
    "Text" : "File"  
},  
]  
}  
  
with open("Templates/ASNIR.json", "w", encoding="utf-8") as  
file_handle:  
    json.dump(Asni, file_handle, indent=4)  
  
# In[4]:  
  
#Spire  
from spire.doc.common import *  
from spire.doc import *  
  
CONFIG_JSON = "Templates/ASNIR.json"  
# Используемые стили  
HEADER_STYLE = "BoldHeader"  
HEADER_LINK_STYLE = "BoldHeaderHyperlink"  
CONTENT_STYLE = "Content"  
CODE_STYLE = "Code"  
  
document = Document()  
section = document.AddSection()  
paragraph = section.AddParagraph()  
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center  
text = paragraph.AppendText("{Projekt}")  
text.CharacterFormat.FontName = "Times New Roman"  
text.CharacterFormat.FontSize = 26  
text.CharacterFormat.Bold = True  
text.CharacterFormat.TextColor = Color.get_Blue()  
  
paragraph = section.AddParagraph()  
paragraph = section.AddParagraph()  
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center  
text = paragraph.AppendText("Тема исследования: {{Projekt3}}")  
text.CharacterFormat.FontName = "Times New Roman"  
text.CharacterFormat.FontSize = 16  
text.CharacterFormat.Bold = True  
  
paragraph = section.AddParagraph()  
paragraph = section.AddParagraph()  
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center  
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center  
text = paragraph.AppendText("Проект: {{Thema}}")  
text.CharacterFormat.FontName = "Times New Roman"  
text.CharacterFormat.FontSize = 16  
text.CharacterFormat.Bold = True  
  
paragraph = section.AddParagraph()  
paragraph = section.AddParagraph()  
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center  
text = paragraph.AppendText("Автор исследования: {{Forscher}}")  
text.CharacterFormat.FontName = "Times New Roman"  
text.CharacterFormat.FontSize = 16  
text.CharacterFormat.Bold = True  
for num in range(9):  
    i=num+1
```



```
paragraph = section.AddParagraph()

paragraph = section.AddParagraph()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("{logo}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 16
text.CharacterFormat.Bold = True

for num in range(13):
    i=num+1
    paragraph = section.AddParagraph()

    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
    text = paragraph.AppendText("{Site}")
    text.CharacterFormat.FontName = "Times New Roman"
    text.CharacterFormat.FontSize = 14
    text.CharacterFormat.Bold = True

    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
    text = paragraph.AppendText("{Year}")
    text.CharacterFormat.FontName = "Times New Roman"
    text.CharacterFormat.FontSize = 14
    text.CharacterFormat.Bold = True
    outputFile = "TemplateTOC1.docx"
    document.SaveToFile(outputFile, FileFormat.Docx)
    document.Close()
```

In[5]:

```
# Используемые стили
HEADER_STYLE = "BoldHeader"
HEADER_LINK_STYLE = "BoldHeaderHyperlink"
CONTENT_STYLE = "Content"
CODE_STYLE = "Code"

document = Document("TemplateTOC1.docx")
print(CONFIG_JSON)
config = getConfigFile(CONFIG_JSON)
tasks = config["tasks"]

i=1
k=-1
for task in tasks:
    i=i+1
    header = f"{task['topic']}"
    Text = f"{task['Text']}"
    print("i: ", i)
    print("header: ", header)
    print("Text: ", Text)
    section = document.AddSection()
    paragraph = section.AddParagraph()
    paragraph = section.AddParagraph()
    paragraph.AppendText(header)
    paragraph.ApplyStyle(BuiltinStyle.Heading2)

    if header == "Содержание":
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        print("header Normal: ", header)
```



```
if Text != " " and header != "Моделирование":  
    #if Text != " " and header != "Моделирование":  
        models = task["models"]  
        print("header Heading2: ", header)  
        print("models: ", models)  
        if models == " ":  
            paragraph = section.AddParagraph()  
            paragraph.AppendText("{{" + Text + "}}")  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Left  
        else:  
            paragraph = section.AddParagraph()  
            paragraph.AppendText(models)  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Left  
  
    paragraph = section.AddParagraph()  
  
    if header == "Моделирование":  
        models = task["models"]  
        for m in models:  
            k=k+1  
            Rname = f"Model: {m}"  
            paragraph = section.AddParagraph()  
            paragraph.AppendText(Rname)  
            paragraph.ApplyStyle(BuiltinStyle.Heading3)  
  
            paragraph = section.AddParagraph()  
            paragraph.AppendText("{ClassBlk" + str(k) + "}")  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Left  
  
            paragraph = section.AddParagraph()  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Center  
            text = paragraph.AppendText("Таблица классификации")  
            text.CharacterFormat.FontName = "Times New Roman"  
            text.CharacterFormat.FontSize = 12  
            text.CharacterFormat.Bold = True  
            paragraph = section.AddParagraph()  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Center  
            AddBookmark("t" + str(k), "Table" + str(k))  
  
            paragraph = section.AddParagraph()  
            paragraph = section.AddParagraph()  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Center  
            text = paragraph.AppendText("Confusion Matrix")  
            text.CharacterFormat.FontName = "Times New Roman"  
            text.CharacterFormat.FontSize = 12  
            text.CharacterFormat.Bold = True  
  
            paragraph = section.AddParagraph()  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Center  
            paragraph.AppendText("{Heatmap" + str(k) + "}")  
  
            paragraph = section.AddParagraph()  
            paragraph = section.AddParagraph()  
            paragraph.Format.HorizontalAlignment =  
HorizontalAlignment.Center  
  
            text = paragraph.AppendText("ROC Curve")  
            text.CharacterFormat.FontName = "Times New Roman"
```



```
text.CharacterFormat.FontSize = 12
text.CharacterFormat.Bold = True
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
    paragraph.AppendText("{{PltR" + str(k) + "}}")

    paragraph = section.AddParagraph()
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center

    text = paragraph.AppendText("Score plot")
    text.CharacterFormat.FontName = "Times New Roman"
    text.CharacterFormat.FontSize = 12
    text.CharacterFormat.Bold = True
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
    paragraph.AppendText("{{PltL" + str(k) + "}}")

outputFile = "TemplateTOC.docx"
document.SaveToFile(outputFile, FileFormat.Docx)
document.Close()
```

In[6]:

```
path = Path(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report")
update_toc(str(path) + "\TemplateTOC.docx")
```

In[7]:

```
from docx import Document, enum

doc = Document("TemplateTOC.docx")
lines = doc.paragraphs
for line in lines:
    #print(line)
    if "{{ClassBlk" in line.text or "{{Kap" in line.text:
        print(line.text)
        line.paragraph_format.first_line_indent = Inches(0.25)
        continue

doc.save("rReportTest.docx")
```

In[8]:

```
doc = DocxTemplate("rReportTest.docx")
reportWordPath = 'ASNI_ReportTest01.docx'
Nk=18

with open("Templates/ASNIR.json", "w", encoding="utf-8") as file_handle:
    json.dump(Asni, file_handle, indent=4)

with open('Templates/ASNIR.json', 'r', encoding='utf-8') as file_object:
    ASNIDict = json.load(file_object)
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
ASNI_dict['logo'] = InlineImage(doc, 'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\GRAPH0.png', Cm(12))

for num in range(11):
    i=num
    #print(i)
    with open("data/rKap0" + str(i) + ".txt", encoding='UTF-8') as f:
        rTxt = f.read()
    ASNI_dict['Kap0' + str(i)] = rTxt

for num in range(Nk):
    i=num+1
    with open("data/rMod" + str(i) + ".txt") as f:
        Txt = f.read()

    ASNI_dict['ClassBlk'+ str(num)] = Txt
    ASNI_dict['Heatmap' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\Heatmap' + str(num) +
'.png', Cm(18))
    ASNI_dict['PltR' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\PlotROC' + str(num) +
'.png', Cm(18))
    ASNI_dict['PltL' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\plot_learning_curve' +
str(num+1) + '.png', Cm(20))

#print(ASNI_dict)

doc.render(ASNI_dict)
doc.save(reportWordPath)

reportWordPath="ASNI_ReportTest01.docx"
doc.save("ASNI_ReportTest01.docx")

# In[9]:


word = win32com.client.DispatchEx("Word.Application")
word.Visible = 1
doc = word.Documents.Open(str(path) + "\ASNI_ReportTest01.docx")
i=1
for num in range(Nk):
    i=i+1
    df=pd.read_csv(f'data/CLSB_{num}.csv')
    rng = doc.Bookmarks("Table" + str(num)).Range

Table=rng.Tables.Add(rng, NumRows=df.shape[0]+1, NumColumns=df.shape[1])
    for col in range(df.shape[1]):
        Table.Cell(1,col+1).Range.Text=str(df.columns[col])
        for row in range(df.shape[0]):
            Table.Cell(row+1+1,col+1).Range.Text=str(df.iloc[row,col])

    doc.Close()
    word.Quit()
    print("Table in Ordnung!")

# In[10]:
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
document = Document()
word_path="C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASNI_Report.docx"
doc = Document("ASNI_ReportTest01.docx")

for table in doc.tables:
    table.alignment = WD_TABLE_ALIGNMENT.CENTER
    table.autofit = False
    table.allow_autofit = False
    n_rows=len(table.rows)
    n_cols=len(table.columns)

    table.cell(0, 0).text = 'Classes+Metrics'
    table.add_row()
    set_column_width(table, 0, 35)
    for c in range(1, 5):
        set_column_width(table, c, 20)

    g = table.cell(n_rows, 0)
    h = table.cell(n_rows, n_cols-1)
    g.merge(h)
    cell = table.cell(n_rows, n_cols-1)

    cell.paragraphs[0].paragraph_format.space_before = Inches(0)
    cell.paragraphs[0].alignment = WD_PARAGRAPH_ALIGNMENT.LEFT
    cell.paragraphs[0].add_run("© Dr. Alexander Wagner. Все права
охраняются законом")
    change_table_cell(table.rows[n_rows].cells[2],
background_color="lightgreen", font_color="0000ff", font_size=8,
bold=True, italic=True)
    table.style = 'Table Grid'

    for i in range(1, n_rows):
        for j in range(1, n_cols):
            element=table.cell(i, j).text
            partition = element.partition('.')
            if (partition[0].isdigit() and partition[1] == '.' and
partition[2].isdigit()):
                newelement = float(element)
                y=round(newelement,3)
                table.cell(i, j).text=str(y)
                table.cell(i,
j).paragraphs[0].paragraph_format.alignment =
WD_TABLE_ALIGNMENT.RIGHT

            for c in range(0, n_cols):
                change_table_cell(table.rows[0].cells[c],
background_color="lightgreen", font_color="0000ff", font_size=12,
bold=True, italic=True)

                for cell in table.columns[c].cells:
                    cell.paragraphs[0].paragraph_format.space_after =
Inches(0)
                    cell.paragraphs[0].paragraph_format.space_before =
Inches(0)
                    cell.vertical_alignment =
WD_CELL_VERTICAL_ALIGNMENT.CENTER

                    set_cell_border(
                        cell,
                        top={"sz": 0.5, "val": "double", "color": "#000000",
"space": "0"},

                        bottom={"sz": 0.5, "val": "double", "color":
"#000000", "space": "0"},
```



Автоматизированная Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
        left={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        right={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        insideH={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        end={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"}  
    )  
  
    for row in table.rows:  
        row.height = Cm(0.55)  
        row.height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
    table.rows[0].height = Cm(0.6)  
    table.rows[0].height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
    table.rows[n_rows-1].height = Cm(0.45)  
    table.rows[n_rows-1].height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
doc.save("ASNI_Report.docx")  
print("ASNI_Report.docx fertig!")  
  
# ### Programm-Block Приложение (в разработке)  
# In[11]:  
  
from spire.doc import *  
document = Document()  
file = os.path.join(cwd, "ASNI_Report.docx")  
dt=datetime.datetime.fromtimestamp(os.stat(file).st_mtime)  
  
txd = "Документ актуализирован: " + dt.strftime('%d.%m.%Y %H:%M:%S')  
print("Mody:", txd)  
  
# Load a Word document  
document.LoadFromFile(file)  
# Get the first section  
section = document.Sections[0]  
  
# Get header  
header = section.HeadersFooters.Header  
  
# Add a paragraph to the header and set its alignment style  
headerParagraph = header.AddParagraph()  
headerParagraph.Format.HorizontalAlignment = HorizontalAlignment.Left  
#headerParagraph.Format.VerticalAlignment = VerticalAlignment.Center  
section.header_distance = Cm(1.2)  
  
headerPicture = headerParagraph.AppendPicture("ASSETS\logo2.jpg")  
headerPicture.TextWrappingStyle = TextWrappingStyle.Square  
headerPicture.VerticalOrigin = VerticalOrigin.Line  
headerPicture.VerticalAlignment = ShapeVerticalAlignment.Center  
#headerPicture.HorizontalAlignment = ShapeHorizontalAlignment.Right  
headerPicture.HorizontalAlignment = ShapeHorizontalAlignment.Left  
headerPicture.VerticalOrigin = VerticalOrigin.TopMarginArea  
  
text = headerParagraph.AppendText("Автоматизированная Система Научных  
Исследований в медицине и здравоохранении «АСНИ-МЕД»")  
text.CharacterFormat.FontName = "Times New"  
text.CharacterFormat.FontSize = 9
```



```
text.CharacterFormat.Bold = True
text.CharacterFormat.TextColor = Color.get_Blue()

section = document.Sections[0]
# Get footer
footer = section.HeadersFooters.Footer

# Add a paragraph to the footer paragraph and set its alignment
style
footerParagraph = footer.AddParagraph()
footerParagraph.Format.HorizontalAlignment =
HorizontalAlignment.Left
# Add text to the footer paragraph and set its font style
text = footerParagraph.AppendText("© Dr. Alexander Wagner, Все права
охраняются законом. " + txd)
text.CharacterFormat.FontName = "Times New"
text.CharacterFormat.FontSize = 9
text.CharacterFormat.Bold = True
text.CharacterFormat.TextColor = Color.get_Blue()

footerParagraph = footer.AddParagraph()
footerParagraph.Format.HorizontalAlignment =
HorizontalAlignment.Right
text = footerParagraph.AppendText("Page ")
txt1=footerParagraph.AppendField("page number", FieldType.FieldPage)
txt2=footerParagraph.AppendText(" of ")
txt3=footerParagraph.AppendField("number of pages",
FieldType.FieldNumPages)
text.CharacterFormat.TextColor = Color.get_Blue()
txt1.CharacterFormat.TextColor = Color.get_Blue()
txt2.CharacterFormat.TextColor = Color.get_Blue()
txt3.CharacterFormat.TextColor = Color.get_Blue()

# Save the result file
document.SaveToFile("AddFootnoteForParagraph.docx",
FileFormat.Docx2016)
document.Close()
```

In[12]:

```
from docx import Document
#import time
doc = Document("AddFootnoteForParagraph.docx")
s=len(doc.sections)

for nt in range(s):
    section = doc.sections[nt]
    header = doc.sections[nt].header
    footer = doc.sections[nt].footer

    section.header_distance = Cm(1.0)
    section.footer_distance = Cm(1.0)

    header_para = header.paragraphs[0]
    header_para.paragraph_format.space_before = Pt(0)
    header_para.paragraph_format.space_after = Pt(7)

    footer_para = footer.paragraphs[0]
    footer_para.paragraph_format.space_before = Pt(0)
    footer_para.paragraph_format.space_after = Pt(0)

    footer_para = footer.paragraphs[1]
    footer_para.paragraph_format.space_before = Pt(0)
```



```
    footer_para.paragraph_format.space_after = Pt(0)

section = doc.sections[0]
section.different_first_page_header_footer = True

lines = doc.paragraphs
n=-1
for line in lines:
    n=n+1
    if line.text == "Evaluation Warning: The document was created
with Spire.Doc for Python.":
        delete_paragraph(line)
        continue

reportWordPath="ASNI_ReportPre.docx"
doc.save("ASNI_ReportPre.docx")
print("ASNI_ReportPre.docx fertig!")
```

In[13]:

```
Text_copy(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\data\Vorwort.docx")
time.sleep(2.4)
replace_copy("C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\ASNI_ReportPre.docx", "xHier")
time.sleep(4.4)

Text_copy(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\data\biblioTestKrollAu.docx")
time.sleep(4.4)
replace_copy("C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\ASNI_ReportResult.docx", "xLiter")
print("Programm Insert-Blöcke beendet!")
print("ASNI_ReportResult.docx fertig!")
```

In[14]:

```
path = Path(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report")
update_toc(str(path) + "\ASNI_ReportResult.docx")
```

In[15]:

```
from docx import Document
reportWordPath = os.path.join(cwd, "ASNI_ReportResult.docx")
reportOutPath = os.path.join(cwd, "ASNI_ReportV05R4.docx")
print("reportWordPath: ", reportWordPath)
print("reportOutPath: ", reportOutPath)
doc = Document(reportWordPath)
doc.save(reportOutPath)

print("Printed immediately2.4")
time.sleep(2.4)
print("Printed after 2.4 seconds.")
convert(reportOutPath, reportOutPath.replace(".docx", ".pdf"))
print(reportOutPath + " fertig!")

now = datetime.datetime.now()
timeend = now.replace(microsecond=0)
print("Programm Ende: ", timeend)
timedelta = (timeend-timestart)
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
print ("Programm Teil II dauert: " + str(timedelta) + " seconds")
----- End of File!
def getConfigFile(config):
    with open(config, encoding='utf-8') as json_file:
        return json.load(json_file)

def addTextParagraphToDocumentInStyle(text, document, style):
    p = document.add_paragraph(text)
    p.style = document.styles[style]

def AddBookmark(t, mark):
    # Find a specific text or phrase in the document
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
    HorizontalAlignment.Center
    text = paragraph.AppendText(t)
    text = document.FindString(t, False, True)
    # Get the found text as a single text range
    textRange = text.GetAsOneRange()
    # Get the paragraph where the text range is located
    paragraph = textRange.OwnerParagraph

    # Get the index position of the text in the paragraph
    index = paragraph.ChildObjects.IndexOf(textRange)

    # Add a bookmark start mark to the paragraph
    start = paragraph.AppendBookmarkStart(mark)
    # Insert the bookmark start mark at the index position of the
    text range
    paragraph.ChildObjects.Insert(index, start)
    # Add a bookmark end mark to the paragraph
    end = paragraph.AppendBookmarkEnd(mark)
    # Insert the bookmark end mark after the text range
    paragraph.ChildObjects.Insert(index + 2, end)

def update_toc(docx_file):
    word = win32com.client.DispatchEx("Word.Application")
    word.Visible = 1
    word.DisplayAlerts = 0

    doc = word.Documents.Open(docx_file)
    #wd_section = doc.Sections(1)
    toc_count = doc.TablesOfContents.Count
    print(toc_count)
    stringG='INHALTSVERZEICHNIS'
    stringK='Содержание'
    if toc_count == 0:
        for i, p in enumerate(doc.Paragraphs):
            if stringK in p.Range.Text:
                try:
                    p.Range.InsertParagraphAfter()
                    parag_range = doc.Paragraphs(i+2).Range
                    parag_range.Font.Name = 'Arial'
                    parag_range.Font.Size = 14
                    parag_range.Font.Bold = constants.wdToggle
                    parag_range.Font.Size = 12
                    doc.TablesOfContents.Add(Range=parag_range,
                                            UseHeadingStyles=True,
                                            LowerHeadingLevel=3)
                except Exception as e:
                    print("Ja:", e, "Nein")
                    break

    elif toc_count == 1:
        toc = doc.TablesOfContents(1)
```



```
        toc.Update()
        print('TOC should have been updated.')
    else:
        print('TOC has not been updated for sure...')

    doc.Close()
    word.Quit()

def set_column_width(table, column, width_mm):
    table.allow_autofit = False
    for row in table.rows:
        row.cells[column].width = Mm(width_mm)

def set_repeat_table_header(row):
    tr = row._tr
    trPr = tr.get_or_add_trPr()
    tblHeader = OxmlElement('w:tblHeader')
    tblHeader.set(qn('w:val'), "true")
    trPr.append(tblHeader)
    return row

def change_table_cell(cell, background_color=None, font_color=None,
                     font_size=None, bold=None, italic=None):
    if background_color:
        shading_elm = parse_xml(r'<w:shd {}')
        w:fill="{}"/>'.format(nsdecls('w'), background_color))
        cell._tc.get_or_add_tcPr().append(shading_elm)

    if font_color:
        for p in cell.paragraphs:
            for r in p.runs:
                r.font.color.rgb =
                    docx.shared.RGBColor.from_string(font_color)

    if font_size:
        for p in cell.paragraphs:
            for r in p.runs:
                r.font.size = docx.shared.Pt(font_size)

    if bold is not None:
        for p in cell.paragraphs:
            for r in p.runs:
                r.bold = bold

    if italic is not None:
        for p in cell.paragraphs:
            for r in p.runs:
                r.italic = italic

def set_cell_border(cell, **kwargs):
    tc = cell._tc
    tcPr = tc.get_or_add_tcPr()

    # check for tag existnace, if none found, then create one
    tcBorders = tcPr.first_child_found_in("w:tcBorders")
    if tcBorders is None:
        tcBorders = OxmlElement('w:tcBorders')
        tcPr.append(tcBorders)

    for edge in ('left', 'top', 'right', 'bottom', 'insideH',
                 'insideV'):
        edge_data = kwargs.get(edge)
        if edge_data:
            tag = 'w:{}{}'.format(edge)
```



```
# check for tag existnace, if none found, then create
one
element = tcBorders.find(qn(tag))
if element is None:
    element = OxmlElement(tag)
    tcBorders.append(element)

# looks like order of attributes is important
for key in ["sz", "val", "color", "space", "shadow"]:
    if key in edge_data:
        element.set(qn('w:{}').format(key)),
str(edge_data[key]))

def delete_paragraph(paragraph):
    p = paragraph._element
    p.getparent().remove(p)
    p._p = p._element = None

def replace_copy(file, txt):
    wordapp =
win32com.client.gencache.EnsureDispatch("Word.Application")
    wordapp.Visible = True
    newdoc = wordapp.Documents.Open(file)
    #print(newdoc.Paragraphs.Count)
    finder = wordapp.Selection.Find
    finder.Text = txt #"Hier"
    finder.Execute()
    #wordapp.Selection.MoveLeft()
    #wordapp.Selection.MoveDown()
    wordapp.Selection.MoveStart
    wordapp.Selection.Paste()
    newdoc.SaveAs("C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\ASNI_ReportResult.docx")
    newdoc .ActiveWindow.Close()
    wordapp.Application.Quit(-1)

def Text_copy(file):
    wordapp =
win32com.client.gencache.EnsureDispatch("Word.Application")
    wordapp.Visible = True
    worddoc = wordapp.Documents.Open(file)
    worddoc.Select()
    wordapp.Selection.Copy()
    worddoc.ActiveWindow.Close()
    wordapp.Application.Quit(-1)

# In[3]:


doc = DocxTemplate("rTemplateTest.docx")
reportWordPath = 'ASNI_ReportTest01.docx'

Inmodels = pd.DataFrame({'Model': [
    'Linear Regression',
    'Logistic Regression',
    'Perceptron',
    'Linear SVC',
    'MLPClassifier',
    'Decision Tree Classifier 1',
    'Stochastic Gradient Decent',
    'RidgeClassifier',
    'BaggingClassifier',
```



```
'AdaBoostClassifier 1',
'GradientBoostingClassifier',
'KNeighborsClassifier',
'DecisionTreeClassifier 2',
'RandomForestClassifier',
'XGBClassifier',
'AdaBoostClassifier 2',
'Naive Bayes',
'SVC' ]})
```



```
Nk=len(Inmodels)
print(Nk)
```



```
Asni = {
    'Projekt': 'Asfendiyarov Kazakh National Medical University',
    'Projekt2': 'Statistik und Data Science Projekt',
    'Projekt3': 'Практическое применение Автоматизированной системы научных исследований в медицине, здравоохранении и смежных областях',
    'Thema': 'Анализ факторов риска сердечно сосудистых заболеваний и прогноз исходов лечения при помощи методов Машинного Обучения',
    'Forscher': 'Dr. Alexander Wagner (Berlin)',
    'Site' : 'Berlin-Almaty',
    'Year' : str(year),
    "tasks" : [
        {
            "folder" : "0",
            "include" : ["Book", "Dog"],
            "topic" : "Содержание",
            "models" : " ",
            "Text" : " "
        },
        {
            "folder" : "0",
            "include" : ["Book", "Dog"],
            "topic" : "Предисловие",
            "models" : "xHier",
            "Text" : "File"
        },
        {
            "folder" : "1",
            "include" : ["Book", "Dog"],
            "topic" : "Введение",
            "models" : " ",
            "Text" : "Kap01"
        },
        {
            "folder" : "2",
            "include" : ["Author", "Ball"],
            "topic" : "Цель исследования",
            "models" : " ",
            "Text" : "Kap02"
        },
        {
            "folder" : "3",
            "include" : ["Author", "Ball"],
            "topic" : "Материалы и методы",
            "models" : " ",
            "Text" : "Kap03"
        },
        {
            "folder" : "4",
            "include" : ["Author", "Ball"],
            "topic" : "Исходные данные и их организация",
            "models" : " "
        }
    ]
}
```



```
        "models" : " ",
        "Text" : "Kap04"
    },
    {
        "folder" : "5",
        "include" : ["Author", "Ball"],
        "topic" : "Предварительный анализ данных",
        "models" : " ",
        "Text" : "Kap05"
    },
    {
        "folder" : "6",
        "include" : ["MovablePoint", "Rectangle"],
        "topic" : "Моделирование",
        "models" : [
            'Linear Regression',
            'Logistic Regression',
            'Perceptron',
            'Linear SVC',
            'MLPClassifier',
            'Decision Tree Classifier 1',
            'Stochastic Gradient Decent',
            'RidgeClassifier',
            'BaggingClassifier',
            'AdaBoostClassifier 1',
            'GradientBoostingClassifier',
            'KNeighborsClassifier',
            'DecisionTreeClassifier 2',
            'RandomForestClassifier',
            'XGBClassifier',
            'AdaBoostClassifier 2',
            'Naive Bayes',
            'SVC' ],
        "Text" : "Kap06"
    },
    {
        "folder" : "7",
        "include" : ["Exercise", "TextAreaExample"],
        "topic" : "Результаты моделирования",
        "models" : " ",
        "Text" : "Kap07"
    },
    {
        "folder" : "8",
        "include" : ["Exercise", "TextAreaExample"],
        "topic" : "Оценка моделей и рекомендации",
        "models" : " ",
        "Text" : "Kap08"
    },
    {
        "folder" : "9",
        "include" : ["Exercise", "TextAreaExample"],
        "topic" : "Обсуждение и выводы",
        "models" : " ",
        "Text" : "Kap09"
    },
    {
        "folder" : "10",
        "include" : ["Exercise", "TextAreaExample"],
        "topic" : "Заключение",
        "models" : " ",
        "Text" : "Kap010"
    },
    {
```



```
        "folder" : "11",
        "include" : ["Exercise", "TextAreaExample"],
        "topic" : "Литература",
        "models" : "xLiter",
        "Text" : "File"
    },
],
}

with open("Templates/ASNIR.json", "w", encoding="utf-8") as file_handle:
    json.dump(Asni, file_handle, indent=4)

# In[4]:
```



```
#Spire
from spire.doc.common import *
from spire.doc import *

CONFIG_JSON = "Templates/ASNIR.json"
# Используемые стили
HEADER_STYLE = "BoldHeader"
HEADER_LINK_STYLE = "BoldHeaderHyperlink"
CONTENT_STYLE = "Content"
CODE_STYLE = "Code"

document = Document()
section = document.AddSection()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("{Projekt}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 26
text.CharacterFormat.Bold = True
text.CharacterFormat.TextColor = Color.get_Blue()

paragraph = section.AddParagraph()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("Тема исследования: {{Projekt3}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 16
text.CharacterFormat.Bold = True

paragraph = section.AddParagraph()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("Проект: {{Thema}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 16
text.CharacterFormat.Bold = True

paragraph = section.AddParagraph()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("Автор исследования: {{Forscher}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 16
text.CharacterFormat.Bold = True
for num in range(9):
    i=num+1
    paragraph = section.AddParagraph()
```



```
paragraph = section.AddParagraph()
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("{\{logo\}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 16
text.CharacterFormat.Bold = True

for num in range(13):
    i=num+1
    paragraph = section.AddParagraph()

paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("{\{Site\}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 14
text.CharacterFormat.Bold = True

paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment = HorizontalAlignment.Center
text = paragraph.AppendText("{\{Year\}}")
text.CharacterFormat.FontName = "Times New Roman"
text.CharacterFormat.FontSize = 14
text.CharacterFormat.Bold = True
outputFile = "TemplateTOC1.docx"
document.SaveToFile(outputFile, FileFormat.Docx)
document.Close()

# In[5]:
```



```
# Используемые стили
HEADER_STYLE = "BoldHeader"
HEADER_LINK_STYLE = "BoldHeaderHyperlink"
CONTENT_STYLE = "Content"
CODE_STYLE = "Code"

document = Document("TemplateTOC1.docx")
print(CONFIG_JSON)
config = getConfigFile(CONFIG_JSON)
tasks = config["tasks"]

i=1
k=-1
for task in tasks:
    i=i+1
    header = f"{task['topic']}"
    Text = f"{task['Text']}"
    print("i: ", i)
    print("header: ", header)
    print("Text: ", Text)
    section = document.AddSection()
    paragraph = section.AddParagraph()
    paragraph = section.AddParagraph()
    paragraph.AppendText(header)
    paragraph.ApplyStyle(BuiltinStyle.Heading2)

    if header == "Содержание":
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        print("header Normal: ", header)
    if Text != " " and header != "Моделирование":
```



```
#if Text != " " and header != "Моделирование":
    models = task["models"]
    print("header Heading2: ", header)
    print("models: ", models)
    if models == " ":
        paragraph = section.AddParagraph()
        paragraph.AppendText("{ " + Text + " }")
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Left
    else:
        paragraph = section.AddParagraph()
        paragraph.AppendText(models)
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Left

paragraph = section.AddParagraph()

if header == "Моделирование":
    models = task["models"]
    for m in models:
        k=k+1
        Rname = f"Model: {m}"
        paragraph = section.AddParagraph()
        paragraph.AppendText(Rname)
        paragraph.ApplyStyle(BuiltinStyle.Heading3)

        paragraph = section.AddParagraph()
        paragraph.AppendText("{ {ClassBlk" + str(k) + " } }")
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Left

        paragraph = section.AddParagraph()
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        text = paragraph.AppendText("Таблица классификации")
        text.CharacterFormat.FontName = "Times New Roman"
        text.CharacterFormat.FontSize = 12
        text.CharacterFormat.Bold = True
        paragraph = section.AddParagraph()
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        AddBookmark("t" + str(k), "Table" + str(k))

        paragraph = section.AddParagraph()
        paragraph = section.AddParagraph()
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        text = paragraph.AppendText("Confusion Matrix")
        text.CharacterFormat.FontName = "Times New Roman"
        text.CharacterFormat.FontSize = 12
        text.CharacterFormat.Bold = True

        paragraph = section.AddParagraph()
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        paragraph.AppendText("{ {Heatmap" + str(k) + " } }")

        paragraph = section.AddParagraph()
        paragraph = section.AddParagraph()
        paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
        text = paragraph.AppendText("ROC Curve")
        text.CharacterFormat.FontName = "Times New Roman"
        text.CharacterFormat.FontSize = 12
```



```
text.CharacterFormat.Bold = True
paragraph = section.AddParagraph()
paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
    paragraph.AppendText("{{PltR" + str(k) + "}}")

    paragraph = section.AddParagraph()
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center

    text = paragraph.AppendText("Score plot")
    text.CharacterFormat.FontName = "Times New Roman"
    text.CharacterFormat.FontSize = 12
    text.CharacterFormat.Bold = True
    paragraph = section.AddParagraph()
    paragraph.Format.HorizontalAlignment =
HorizontalAlignment.Center
    paragraph.AppendText("{{PltL" + str(k) + "}}")

outputFile = "TemplateTOC.docx"
document.SaveToFile(outputFile, FileFormat.Docx)
document.Close()
```

In[6]:

```
path = Path(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report")
update_toc(str(path) + "\TemplateTOC.docx")
```

In[7]:

```
from docx import Document, enum

doc = Document("TemplateTOC.docx")
lines = doc.paragraphs
for line in lines:
    #print(line)
    if "{{ClassBlk" in line.text or "{{Kap" in line.text:
        print(line.text)
        line.paragraph_format.first_line_indent = Inches(0.25)
        continue

doc.save("rReportTest.docx")
```

In[8]:

```
doc = DocxTemplate("rReportTest.docx")
reportWordPath = 'ASNI_ReportTest01.docx'
Nk=18

with open("Templates/ASNIR.json", "w", encoding="utf-8") as file_handle:
    json.dump(Asni, file_handle, indent=4)

with open('Templates/ASNIR.json', 'r', encoding='utf-8') as file_object:
    ASNIDict = json.load(file_object)
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
ASNI_dict['logo'] = InlineImage(doc, 'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\GRAPH0.png', Cm(12))

for num in range(11):
    i=num
    #print(i)
    with open("data/rKap0" + str(i) + ".txt", encoding='UTF-8') as f:
        rTxt = f.read()
    ASNI_dict['Kap0' + str(i)] = rTxt

for num in range(Nk):
    i=num+1
    with open("data/rMod" + str(i) + ".txt") as f:
        Txt = f.read()

    ASNI_dict['ClassBlk'+ str(num)] = Txt
    ASNI_dict['Heatmap' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\Heatmap' + str(num) +
'.png', Cm(18))
    ASNI_dict['PltR' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\PlotROC' + str(num) +
'.png', Cm(18))
    ASNI_dict['PltL' + str(num)] = InlineImage(doc,
'C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASSETS\plot_learning_curve' +
str(num+1) + '.png', Cm(20))

#print(ASNI_dict)

doc.render(ASNI_dict)
doc.save(reportWordPath)

reportWordPath="ASNI_ReportTest01.docx"
doc.save("ASNI_ReportTest01.docx")

# In[9]:


word = win32com.client.DispatchEx("Word.Application")
word.Visible = 1
doc = word.Documents.Open(str(path) + "\ASNI_ReportTest01.docx")
i=1
for num in range(Nk):
    i=i+1
    df=pd.read_csv(f'data/CLSB_{num}.csv')
    rng = doc.Bookmarks("Table" + str(num)).Range

Table=rng.Tables.Add(rng, NumRows=df.shape[0]+1, NumColumns=df.shape[1])
    for col in range(df.shape[1]):
        Table.Cell(1,col+1).Range.Text=str(df.columns[col])
        for row in range(df.shape[0]):
            Table.Cell(row+1+1,col+1).Range.Text=str(df.iloc[row,col])

    doc.Close()
    word.Quit()
    print("Table in Ordnung!")

# In[10]:
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
document = Document()
word_path="C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\ASNI_Report.docx"
doc = Document("ASNI_ReportTest01.docx")

for table in doc.tables:
    table.alignment = WD_TABLE_ALIGNMENT.CENTER
    table.autofit = False
    table.allow_autofit = False
    n_rows=len(table.rows)
    n_cols=len(table.columns)

    table.cell(0, 0).text = 'Classes+Metrics'
    table.add_row()
    set_column_width(table, 0, 35)
    for c in range(1, 5):
        set_column_width(table, c, 20)

    g = table.cell(n_rows, 0)
    h = table.cell(n_rows, n_cols-1)
    g.merge(h)
    cell = table.cell(n_rows, n_cols-1)

    cell.paragraphs[0].paragraph_format.space_before = Inches(0)
    cell.paragraphs[0].alignment = WD_PARAGRAPH_ALIGNMENT.LEFT
    cell.paragraphs[0].add_run("© Dr. Alexander Wagner. Все права
охраняются законом")
    change_table_cell(table.rows[n_rows].cells[2],
background_color="lightgreen", font_color="0000ff", font_size=8,
bold=True, italic=True)
    table.style = 'Table Grid'

    for i in range(1, n_rows):
        for j in range(1, n_cols):
            element=table.cell(i, j).text
            partition = element.partition('.')
            if (partition[0].isdigit() and partition[1] == '.' and
partition[2].isdigit()):
                newelement = float(element)
                y=round(newelement,3)
                table.cell(i, j).text=str(y)
                table.cell(i,
j).paragraphs[0].paragraph_format.alignment =
WD_TABLE_ALIGNMENT.RIGHT

            for c in range(0, n_cols):
                change_table_cell(table.rows[0].cells[c],
background_color="lightgreen", font_color="0000ff", font_size=12,
bold=True, italic=True)

                for cell in table.columns[c].cells:
                    cell.paragraphs[0].paragraph_format.space_after =
Inches(0)
                    cell.paragraphs[0].paragraph_format.space_before =
Inches(0)
                    cell.vertical_alignment =
WD_CELL_VERTICAL_ALIGNMENT.CENTER

                    set_cell_border(
                        cell,
                        top={"sz": 0.5, "val": "double", "color": "#000000",
"space": "0"},

                        bottom={"sz": 0.5, "val": "double", "color":
"#000000", "space": "0"},
```



Автоматизированная Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
        left={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        right={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        insideH={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"},  
        end={"sz": 0.5, "val": "double", "color": "#000000", "space": "0"}  
    )  
  
    for row in table.rows:  
        row.height = Cm(0.55)  
        row.height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
    table.rows[0].height = Cm(0.6)  
    table.rows[0].height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
    table.rows[n_rows-1].height = Cm(0.45)  
    table.rows[n_rows-1].height_rule = WD_ROW_HEIGHT_RULE.EXACTLY  
  
doc.save("ASNI_Report.docx")  
print("ASNI_Report.docx fertig!")  
  
# ### Programm-Block Приложение (в разработке)  
# In[11]:  
  
from spire.doc import *  
document = Document()  
file = os.path.join(cwd, "ASNI_Report.docx")  
dt=datetime.datetime.fromtimestamp(os.stat(file).st_mtime)  
  
txd = "Документ актуализирован: " + dt.strftime('%d.%m.%Y %H:%M:%S')  
print("Mody:", txd)  
  
# Load a Word document  
document.LoadFromFile(file)  
# Get the first section  
section = document.Sections[0]  
  
# Get header  
header = section.HeadersFooters.Header  
  
# Add a paragraph to the header and set its alignment style  
headerParagraph = header.AddParagraph()  
headerParagraph.Format.HorizontalAlignment = HorizontalAlignment.Left  
#headerParagraph.Format.VerticalAlignment = VerticalAlignment.Center  
section.header_distance = Cm(1.2)  
  
headerPicture = headerParagraph.AppendPicture("ASSETS\logo2.jpg")  
headerPicture.TextWrappingStyle = TextWrappingStyle.Square  
headerPicture.VerticalOrigin = VerticalOrigin.Line  
headerPicture.VerticalAlignment = ShapeVerticalAlignment.Center  
#headerPicture.HorizontalAlignment = ShapeHorizontalAlignment.Right  
headerPicture.HorizontalAlignment = ShapeHorizontalAlignment.Left  
headerPicture.VerticalOrigin = VerticalOrigin.TopMarginArea  
  
text = headerParagraph.AppendText("Автоматизированная Система Научных  
Исследований в медицине и здравоохранении «АСНИ-МЕД»")  
text.CharacterFormat.FontName = "Times New"  
text.CharacterFormat.FontSize = 9
```



```
text.CharacterFormat.Bold = True
text.CharacterFormat.TextColor = Color.get_Blue()

section = document.Sections[0]
# Get footer
footer = section.HeadersFooters.Footer

# Add a paragraph to the footer paragraph and set its alignment
style
footerParagraph = footer.AddParagraph()
footerParagraph.Format.HorizontalAlignment =
HorizontalAlignment.Left
# Add text to the footer paragraph and set its font style
text = footerParagraph.AppendText("© Dr. Alexander Wagner, Все права
охраняются законом. " + txd)
text.CharacterFormat.FontName = "Times New"
text.CharacterFormat.FontSize = 9
text.CharacterFormat.Bold = True
text.CharacterFormat.TextColor = Color.get_Blue()

footerParagraph = footer.AddParagraph()
footerParagraph.Format.HorizontalAlignment =
HorizontalAlignment.Right
text = footerParagraph.AppendText("Page ")
txt1=footerParagraph.AppendField("page number", FieldType.FieldPage)
txt2=footerParagraph.AppendText(" of ")
txt3=footerParagraph.AppendField("number of pages",
FieldType.FieldNumPages)
text.CharacterFormat.TextColor = Color.get_Blue()
txt1.CharacterFormat.TextColor = Color.get_Blue()
txt2.CharacterFormat.TextColor = Color.get_Blue()
txt3.CharacterFormat.TextColor = Color.get_Blue()

# Save the result file
document.SaveToFile("AddFootnoteForParagraph.docx",
FileFormat.Docx2016)
document.Close()
```

In[12]:

```
from docx import Document
#import time
doc = Document("AddFootnoteForParagraph.docx")
s=len(doc.sections)

for nt in range(s):
    section = doc.sections[nt]
    header = doc.sections[nt].header
    footer = doc.sections[nt].footer

    section.header_distance = Cm(1.0)
    section.footer_distance = Cm(1.0)

    header_para = header.paragraphs[0]
    header_para.paragraph_format.space_before = Pt(0)
    header_para.paragraph_format.space_after = Pt(7)

    footer_para = footer.paragraphs[0]
    footer_para.paragraph_format.space_before = Pt(0)
    footer_para.paragraph_format.space_after = Pt(0)

    footer_para = footer.paragraphs[1]
    footer_para.paragraph_format.space_before = Pt(0)
```



```
footer_para.paragraph_format.space_after = Pt(0)

section = doc.sections[0]
section.different_first_page_header_footer = True

lines = doc.paragraphs
n=-1
for line in lines:
    n=n+1
    if line.text == "Evaluation Warning: The document was created
with Spire.Doc for Python.":
        delete_paragraph(line)
        continue

reportWordPath="ASNI_ReportPre.docx"
doc.save("ASNI_ReportPre.docx")
print("ASNI_ReportPre.docx fertig!")
```

In[13]:

```
Text_copy(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report\data\Vorwort.docx")
time.sleep(2.4)
replace_copy("C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\ASNI_ReportPre.docx", "xHier")
time.sleep(4.4)

Text_copy(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\data\biblioTestKrollAu.docx")
time.sleep(4.4)
replace_copy("C:\IPYNBgesamt\ASNI-FEN\ASNI-
Report\ASNI_ReportResult.docx", "xLiter")
print("Programm Insert-Blöcke beendet!")
print("ASNI_ReportResult.docx fertig!")
```

In[14]:

```
path = Path(r"C:\IPYNBgesamt\ASNI-FEN\ASNI-Report")
update_toc(str(path) + "\ASNI_ReportResult.docx")
```

In[15]:

```
from docx import Document
reportWordPath = os.path.join(cwd, "ASNI_ReportResult.docx")
reportOutPath = os.path.join(cwd, "ASNI_ReportV05R4.docx")
print("reportWordPath: ", reportWordPath)
print("reportOutPath: ", reportOutPath)
doc = Document(reportWordPath)
doc.save(reportOutPath)

print("Printed immediately2.4")
time.sleep(2.4)
print("Printed after 2.4 seconds.")
convert(reportOutPath, reportOutPath.replace(".docx", ".pdf"))
print(reportOutPath + " fertig!")

now = datetime.datetime.now()
timeend = now.replace(microsecond=0)
print("Programm Ende: ", timeend)
timedelta = (timeend-timestart)
```



Автоматизация Система Научных Исследований в медицине и здравоохранении «АСНИ-МЕД»

```
print ("Programm Teil II dauert: " + str(timedelta) + " seconds")
----- End of File! -----
```