**Документация**

**Дипломная работа по теме:**

**«Анализ данных клеток рака**

**(поиск инсайтов, составление рекомендаций стейкхолдерам, построение предиктивной модели классификации типа клеток)»**

**Профессия «Аналитик данных», DAU-42**

**Григорьев Семен Павлович**

**г. Тольятти**

**2023**

**СОДЕРЖАНИЕ**

**Введение**……………………………………………………………………….3

**Блок 1. Описание исходного датасета и типов данных**…………………4

**Блок 2. Подготовка и преобразование данных**….………………………..5

2.1 Подготовка..…………….……………………………………...………5

2.2 Преобразование………………………………………………………..5

2.3 Вывод по датасету…………………………………………………..…5

**Блок 3. Очистка данных**……..……………………………………………...5

3.1 Аномалии в данных……………………….…………………………...6

**Блок 4. Анализ данных**……………………………………………………....7

**Блок 5. Алгоритмы классификации для проверки результатов**………9

5.1 Логистическая регрессия..…………………………………...…….….9

5.2 Линейный дискриминантный анализ…………………………………9

5.3 Случайный лес……………………………………………………......10

4.4 Градиентный бустинг………………………………………………...10

**Блок 6. Оптимизация моделей**………………………………………….....10

6.1 Удаление данных с высокой корреляцией……………………….....10

6.2 Метод главных компонент (PCA)……………………………….…..13

**Итоги проекта и заключение**……………………….……………………..17

**Введение**

**Цели проекта**

В рамках проведения исследования была поставлена задача разработки модели для классификации раковых клеток на основе имеющихся медицинских данных. Цель проекта заключалась в создании инструмента для более точного и быстрого выявления заболеваний, что может существенно повысить эффективность лечения и спасти жизни пациентов.

**Бизнес-Задачи**

1. Подбор эффективной модели для классификации раковых клеток на основе имеющихся медицинских данных с целью повышения эффективности диагностики и лечения заболеваний.
2. Увеличение точности определения типа раковых клеток, что может способствовать своевременному началу лечения и повысить шансы на выздоровление.

**Блок 1. Описание исходного датасета и типов данных   
(16 столбцов)**

Для исследования был взят датасет «[Breast Cancer Wisconsin Data set](https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data)» с данными раковых клеток пациентов.

| **№** | **Имя столбца** | **Описание** | **Тип данных** |
| --- | --- | --- | --- |
| *1* | *id* | Внутренний номер проверяемого пациента | int64 |
| *2* | *diagnosis / диагноз* | Тип раковых клеток пациента  M = злокачественные  (в дальнейшем 1)  B = доброкачественные  (в дальнейшем 0) | object |
| Для каждого клеточного ядра вычисляются десять вещественных признаков: | | | |
| *а* | *radius / радиус* | Среднее расстояний от центра до точек на периметре | float64 |
| *б* | *texture / строение* | Стандартное отклонение значений шкалы среднего | float64 |
| *в* | *perimeter / периметр* | Периметр клетки | float64 |
| *г* | *area / область* | Область где находится клетка | float64 |
| *д* | *smoothness / плавность* | Локальное изменение длины радиуса | float64 |
| *е* | *compactness / компактность* | Периметр^2 / область - 1.0 | float64 |
| *ж* | *concavity / вогнутость* | Выраженность вогнутых участков контура | float64 |
| *з* | *concave points / вогнутые точки* | Количество вогнутых частей контура | float64 |
| *и* | *symmetry / симметрия* | Симметрия n-го порядка | float64 |
| *к* | *fractal dimension / фрактальная размерность* | Приближенная длина границы | float64 |
| Каждый признак повторяется три раза в сочетании с каждой из следующих метрик: | | | |
| *3-12* | *mean / среднее* | Среднее арифметическое всех наблюдений пациента | float64 |
| *13-22* | *standard error (se)/ стандартная ошибка* | Стандартная ошибка среднего всех наблюдений пациента | float64 |
| *23-32* | *worst / худшее* | Худший результат из всех наблюдений пациента | float64 |
| *33* | *Unnamed: 32* | Пустые значения | float64 |

**Блок 2. Подготовка и преобразование данных**

В ходе исследования качества данных были сделаны следующие изменения:

**2.1 Подготовка**

* Столбец “**id**” был удален, так как он содержит бесполезную для нас информацию;
* Столбец “**Unnamed: 32**” содержит пустые значения, он также был удален из датасета;
* Значения столбца “**diagnosis**” B и M заменены на 0 и 1 соответственно, это упрощает использование категориальных данных в моделях классификации.

**2.2 Преобразование**

Никаких преобразований не потребовалось

**2.3 Вывод по датасету**

Во всем датасете хорошие и чистые данные, дальнейший анализ показал, что данные не требуют нормализации.

**Блок 3. Очистка данных**

Так как первичной очистки данных не потребовалось, то структура данных со всеми колонками приведена в таблице:

| **№** | **Новое имя столбца** | **Преобразование данных** | **% NaN** |
| --- | --- | --- | --- |
| *1* | *diagnosis* | int64 | 0.00 |
| *2* | *radius\_mean* | float64 | 0.00 |
| *3* | *texture\_mean* | float64 | 0.00 |
| *4* | *perimeter\_mean* | float64 | 0.00 |
| *5* | *area\_mean* | float64 | 0.00 |
| *6* | *smoothness\_mean* | float64 | 0.00 |
| *7* | *compactness\_mean* | float64 | 0.00 |
| *8* | *concavity\_mean* | float64 | 0.00 |
| *9* | *concave points\_mean* | float64 | 0.00 |
| *10* | *symmetry\_mean* | float64 | 0.00 |
| *11* | *fractal\_dimension\_mean* | float64 | 0.00 |
| *12* | *radius\_se* | float64 | 0.00 |
| *13* | *texture\_se* | float64 | 0.00 |
| *14* | *perimeter\_se* | float64 | 0.00 |
| *15* | *area\_se* | float64 | 0.00 |
| *16* | *smoothness\_se* | float64 | 0.00 |
| *17* | *compactness\_se* | float64 | 0.00 |
| *18* | *concavity\_se* | float64 | 0.00 |
| *19* | *concave points\_se* | float64 | 0.00 |
| *20* | *symmetry\_se* | float64 | 0.00 |
| *21* | *fractal\_dimension\_se* | float64 | 0.00 |
| *22* | *radius\_worst* | float64 | 0.00 |
| *23* | *texture\_worst* | float64 | 0.00 |
| *24* | *perimeter\_worst* | float64 | 0.00 |
| *25* | *area\_worst* | float64 | 0.00 |
| *26* | *smoothness\_worst* | float64 | 0.00 |
| *27* | *compactness\_worst* | float64 | 0.00 |
| *28* | *concavity\_worst* | float64 | 0.00 |
| *29* | *concave points\_worst* | float64 | 0.00 |
| *30* | *symmetry\_worst* | float64 | 0.00 |
| *31* | *fractal\_dimension\_worst* | float64 | 0.00 |

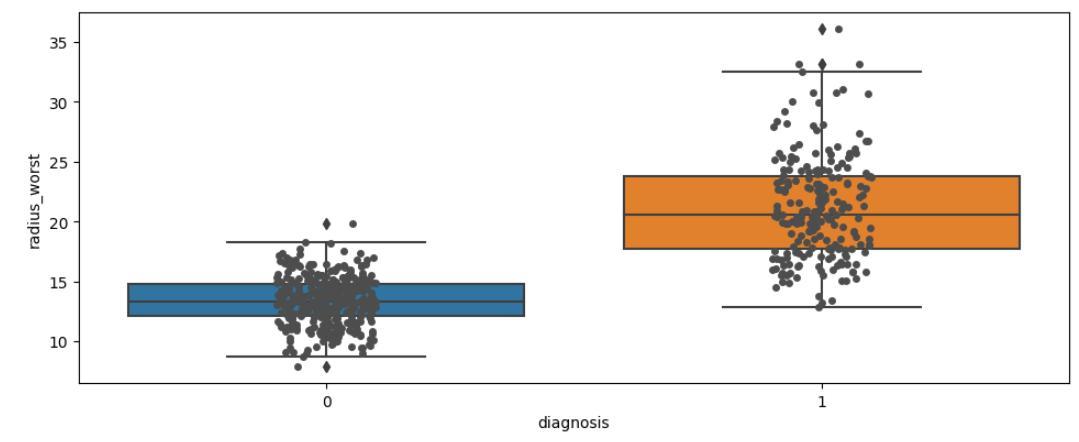
**3.1 Аномалии в данных**

В ходе подробного анализа данных было принято решение оставить все данные на своих местах. Как таковых аномалий обнаружено не было.

**Блок 4. Анализ данных**

Анализ данных показал, что целевая переменная diagnosis (диагноз) не имеет баланса классов, точнее говоря класс доброкачественных клеток составляет 357 значений, класс злокачественных составляет 212 значений в датасете.

На формирование класса злокачественных клеток очень сильно влияют вещественные признаки с метрикой worst (худшее), а если быть точнее, то самые классифицирующими колонками оказались: radius\_worst perimeter\_worst, area\_worst. Визуализация ниже показывает, что все же есть некоторая группа значений, которая имеет общие значения, так для radius\_worst это значения от 15 до 20.

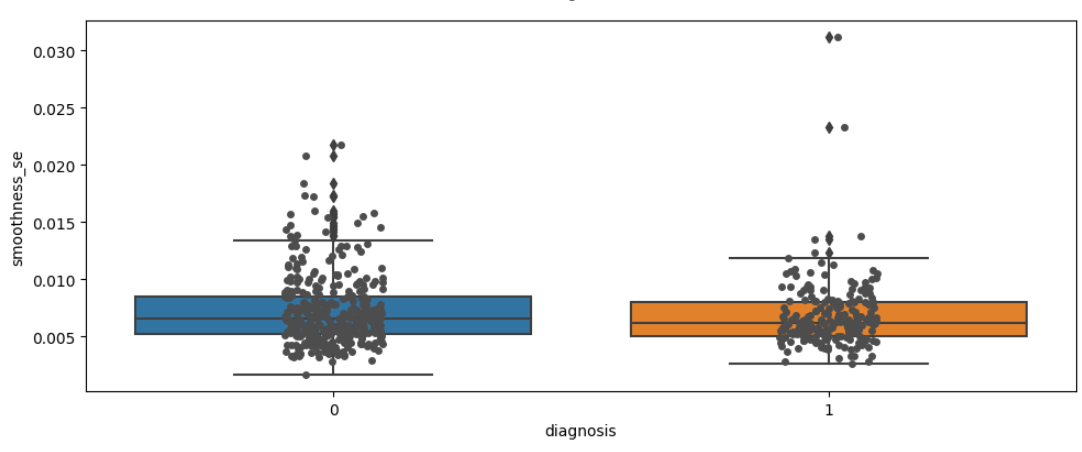
****

*Рисунок 1.1.*

Коробчатая диаграмма колонки radius\_worst с разделением на классы diagnosis,

где 0 – доброкачественные клетки, 1 – злокачественные клетки

Также в ходе анализа данных было обнаружено, что классы большинства вещественных признаков с метрикой se (standard error) имеют похожую структуру. Так, например, в колонке smoothness\_se мы можем увидеть похожее распределение по классам, без учета того, что данных в классе доброкачественных клеток больше.

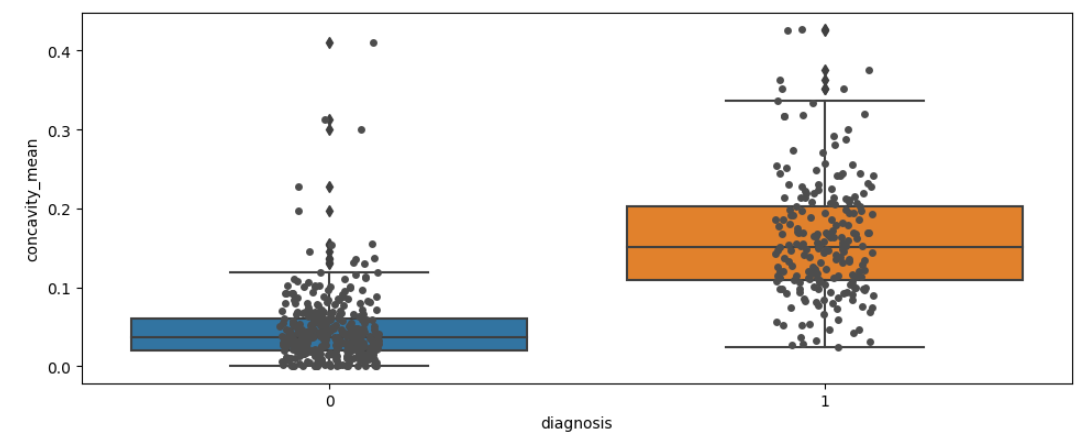


*Рисунок 1.2.*

Коробчатая диаграмма колонки smoothness\_se с разделением на классы diagnosis,

где 0 – доброкачественные клетки, 1 – злокачественные клетки

Говоря о вещественных признаках с метрикой mean, можно также, как и с метрикой worst, заметить явное отделение классов друг от друга, однако большое количество колонок имеют тенденцию выбрасывать значения доброкачественных клеток до уровня выброшенных значений злокачественных клеток, как на примере колонки concavity\_mean.



*Рисунок 1.3.*

Коробчатая диаграмма колонки concavity\_mean с разделением на классы diagnosis,

где 0 – доброкачественные клетки, 1 – злокачественные клетки

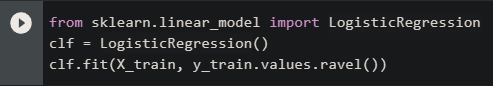
Это все говорит нам о том, что каждая метрика играет свою непосредственную роль, и мы не можем заставить алгоритм классификации работать лишь на одной метрике. Поэтому необходимо подстроить данные так, чтобы алгоритмы могли работать с высокой точностью.

**Блок 5. Алгоритмы классификации для проверки результатов**

Прежде всего хотелось бы обозначить алгоритмы классификации, которые были использованы в ходе поиска наилучшего результата. Так как ограничивать себя всего одной моделью было не серьезно, при подгонке нормализованных данных были рассмотрены четыре модели классификации. Каждая модель использовала размер тренировочной выборки равный 80% и тестовая 20% от всего датасета соответственно. Далее вкратце будет описан каждый из использованных алгоритмов.

**5.1 Логистическая регрессия**

Первым рассматриваемым алгоритмом стала логистическая регрессия. Логистическая регрессия: это алгоритм бинарной классификации, который используется для предсказания вероятности отнесения объекта к одному из двух классов. Логистическая регрессия хорошо работает в случаях, когда нужно определить, к какому классу относится объект, основываясь на наборе предикторов.

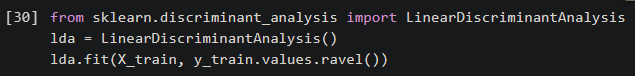


*Рисунок 5.1.*

Пример кода модели логистической регрессии.

**5.2 Линейный дискриминантный анализ**

Второй моделью стал линейный дискриминантный анализ. Линейный дискриминантный анализ: это алгоритм классификации, который основывается на нахождении различий между классами по статистическим параметрам. Он позволяет отделить объекты одного класса от объектов другого класса на основе их характеристик.

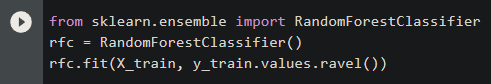
**

*Рисунок 5.2.*

Пример кода модели линейного дискриминантного анализа

**5.3 Случайный лес**

Третья модель случайный лес. Случайный лес: это алгоритм машинного обучения, который создает множество деревьев решений и комбинирует их для получения более точных прогнозов. Случайный лес хорошо работает в случаях, когда нужно предсказать категорию объекта или выявить важные признаки.

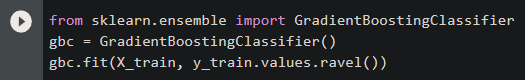


*Рисунок 5.3.*

Пример кода модели случайный лес

**5.4 Градиентный бустинг**

И последней, но не по значимости, а по сложности алгоритма решения следует модель градиентного бустинга. Градиентный бустинг: это алгоритм машинного обучения, который также использует множество деревьев решений, но создает их последовательно и настраивает каждое дерево на остатки предыдущих. Градиентный бустинг хорошо работает в случаях, когда нужно обработать большие объемы данных и получить высокое качество предсказаний.



*Рисунок 5.4.*

Пример кода модели градиентный бустинг

**Блок 6. Оптимизация моделей**

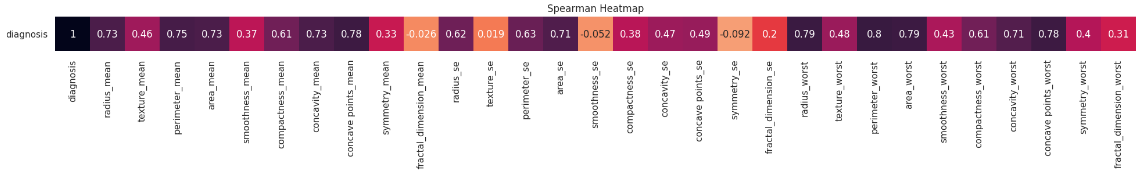
Целью блока является поиск эффективного метода оптимизации в задаче классификации.

**6.1 Удаление данных с высокой корреляцией**

Первым возможным решением по улучшению качества моделей классификации стало удаление коррелирующих по отношению с целевой переменной значений. Учитывая то, что имеющийся объем данных велик,   
а именно имеется 31 колонка, то от некоторых данных можно было избавиться. Такой вариант имеет место быть, так как имея большой объем переменных в модели не только усложнит задачу классификации, но и замедлит скорость выполнения самого алгоритма.

Для поиска коррелирующих значений использовался метод ранговой корреляции Спирмена. Метод Спирмена позволяет определить тесноту и направление корреляционной связи между двумя признаками.

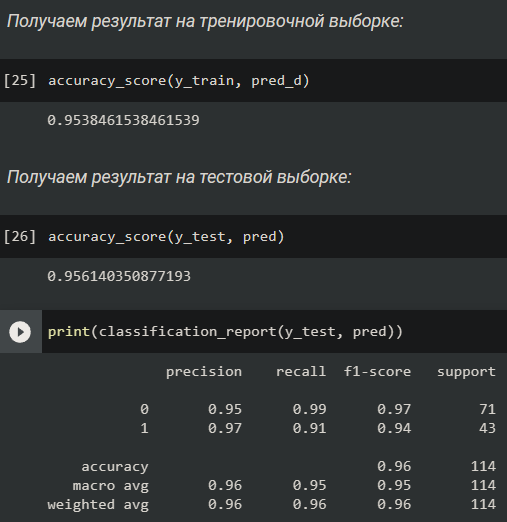
Основываясь на данных из тепловой карты ниже, где темные значения имеют высокую корреляцию и наоборот, было принято решение избавиться от следующих данных: perimeter\_mean, area\_mean, radius\_mean, radius\_worst, perimeter\_worst, area\_worst.



*Рисунок 6.1.*

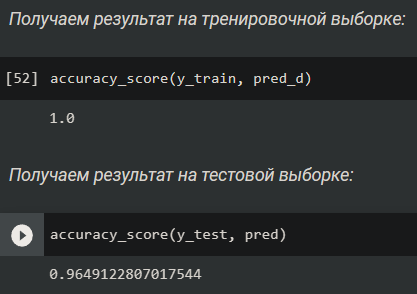
Тепловая карта корреляций Спирмена

Новый набор данных был протестирован на моделях классификации, в качестве первой, рассматриваемой выступала модель логистической регрессии. Результат классификации на новых данных составил 0.953 на тренировочной выборке и 0.956 на тестовой.



*Рисунок 6.2.*

Результат логистической регрессии на данных без коррелирующих значений

Результат куда лучше предоставила модель случайный лес. Результат на тренировочной выборке составил 1.0 и 0.964 на тестовой. Внушительный результат, как можно заметить модель хорошо обучилась на тренировочных данных и дает безупречный результат, но на тестовых данных результат не такой отличный.

*Рисунок 6.3.*

Результат модели случайный лес на данных без коррелирующих значений

Результаты остальных моделей ожидают желать лучшего. Линейный дискриминантный анализ показал 0.962 на тренировочной и 0.938 на тестовой. Градиентный бустинг показывает 0.923 на тренировочной и 0.947 на тестовой.

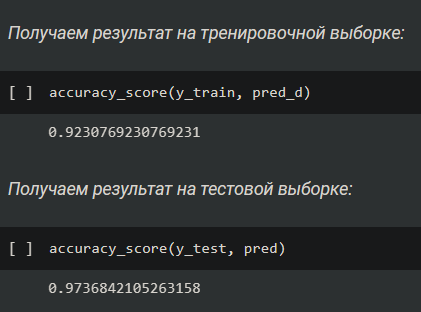
Если сравнивать с проверкой результата на полном объеме данных, которую я решил оставить за кадром, то результат удаления коррелирующих с целевой переменной данных очевидно выше. Однако, несмотря на это, были получены достаточно низкие значения.

Работая над задачей классификации, была поставлена цель добиться результата выше 0.97. Фильтрация коррелирующих значений нам не помогла, этот метод мы отбросим, так как по всей видимости мы лишаем алгоритм знаковых, влияющих на формирование целевой переменной данных по одному из классов. Идеального баланса мы не достигнем, но это не значит, что мы не сможем добиться лучшего результата от наших моделей.

**6.2 Метод главных компонент (PCA)**

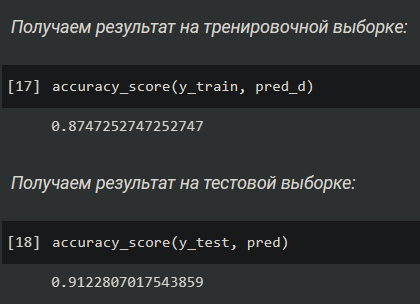
Метод главных компонент (PCA) ‑ это широко используемый метод снижения размерности данных. Он позволяет избавиться от избыточности информации в многомерных данных путем выделения наиболее важных компонентов в данных. Для получения хороших результатов классификации метод PCA требует нормализовать данные.

Итоговая проверка не заставляет себя ждать, разделив датасет на три компоненты мы получили прекрасный результат по модели логистической регрессии 0.923 на тренировочной и 0.973 на тестовой. Алгоритм логистической регрессии обновлялся не один десяток раз, тестовая выборка обновлялась и меняла свой состав, и каждый раз результат варьировался на уровне полученного значения, поэтому результат можно считать отличным.

*******Рисунок 6.4.*

Результат логистической регрессии на PCA данных

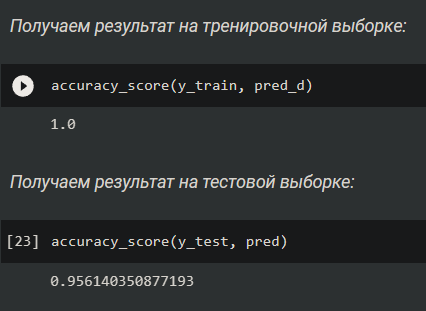
Линейный дискриминантный анализ показал следующие результаты: 0.874 на тренировочной выборке и 0.912 на тестовой выборке. Показатели оказались очень слабыми, модель не справилась и в этот раз.



*Рисунок 6.5.*

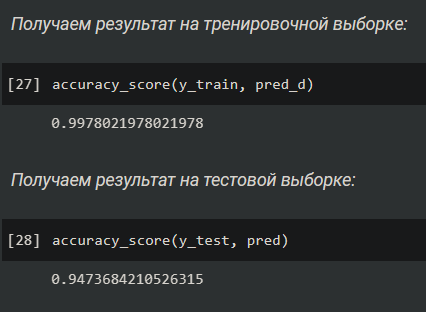
Результат Линейного дискриминантного анализа на PCA данных

Модель классификации случайный лес показала прекрасный результат на тренировочной выборке 1.0, но не смогла показать себя на тестовой 0.956. Несмотря на свой отличный результат на некоррелированных данных алгоритм классификации случайный лес не смог улучшить его на PCA компонентах.

****

*Рисунок 6.6.*

Результат классификации случайный лес на PCA данных

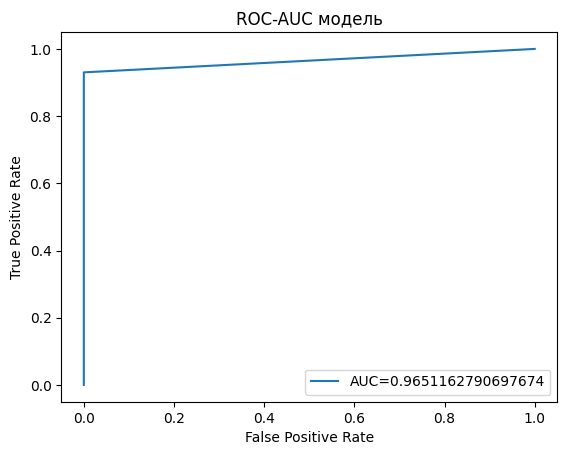
Градиентный бустинг предоставил ответы в 0.997 на тренировочной и 0.947 на тестовой выборке. Довольно слабый результат. Как и линейный дискриминантный анализ, градиентный бустинг довольно слаб в нашей задаче классификации.

*Рисунок 6.7*.

Результат градиентного бустинга на PCA данных

Модель логистической регрессии показал себя лучше других. Дабы проверить результат модели на ошибки первого и второго рода была выведена модель ROC-AUC. ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) является одним из наиболее распространенных инструментов для оценки качества моделей машинного обучения. Он используется для измерения способности модели различать между собой положительные и отрицательные классы.

В основе ROC-AUC лежит кривая ROC, которая показывает отношение между долей верных положительных результатов и долей ложных положительных результатов при изменении порога принятия решения моделью. ROC-AUC представляет собой площадь под кривой ROC, которая может принимать значения от 0 до 1. Чем выше значение ROC-AUC, тем лучше модель различает между собой положительные и отрицательные классы. ROC-AUC часто используется в задачах бинарной классификации, где необходимо разделить данные на два класса: положительный и отрицательный.

****

*Рисунок 6.8.*

ROC-AUC модель

Чем больше кривая охватывает верхний левый угол графика, тем лучше модель классифицирует данные по категориям.

Наша основная задача избежать ошибки первого рода (False Positive), чего мы и добились. Алгоритм логистической регрессии в совокупности с методом главных компонент показывает отличный результат.

**Итоги проекта и заключение:**

По ходу выполнения работы, были найдены зависимости, закономерности в данных, а также отображены важные визуализации, которые могут быть полезны для работников соответствующих медицинских учреждений. Такая информация может помочь врачам ставить диагнозы куда быстрее на первых этапах лечения.

Анализ данных клеток рака показал, что для точной классификации доброкачественных и злокачественных клеток необходимо было определить границу разделения данных на классы. Для ее определения был использован метод главных компонент (PCA), который позволил разобраться в структуре данных и обработать данные для дальнейшего построения предиктивной модели классификации. На основе обработанных PCA данных были построены 4 модели классификации, среди которых лучший результат (0.97) показала модель логистической регрессии, старалась не отставать и модель случайный лес, с результатом (0.96). В целом, данные оказались обширными, в них использовались наборы метрик для каждого наблюдения, что в какой-то степени усложняло процесс. Однако, обработка данных методом PCA существенно повысила эффективность классификации и помогла найти оптимальную границу для разделения на классы.

Был подобран подход сужения размерности данных для их последующей классификации методом логистической регрессии. Такая модель классифицирует клетки по их показателям на злокачественные и доброкачественные и имеет важное значение для медицины, помогая улучшать диагностику и снижать количество ошибочных диагнозов. Это поможет врачам находить заболевания с большей точностью и пациентам начинать лечение на ранних стадиях, увеличивая их шансы на выздоровление. Проект также принесет выгоды страховым компаниям и заказчику - медицинскому учреждению в виде повышения репутации, улучшения обслуживания и доверия пациентов.