ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”

ВЫСШАЯ ШКОЛА ЦИФРОВОЙ КУЛЬТУРЫ

Курсовой проект на тему:

“Прогнозирование дохода жителей США на основе переписи населения с помощью методов машинного обучения”

Выполнила:

Антоненко Елизавета Павловна

Группа:

S4101

Санкт-Петербург, 2022

# Содержание

Постановка задачи и описание датасета 3

Основные определения 4

Предобработка данных 8

Построение и обучение моделей 10

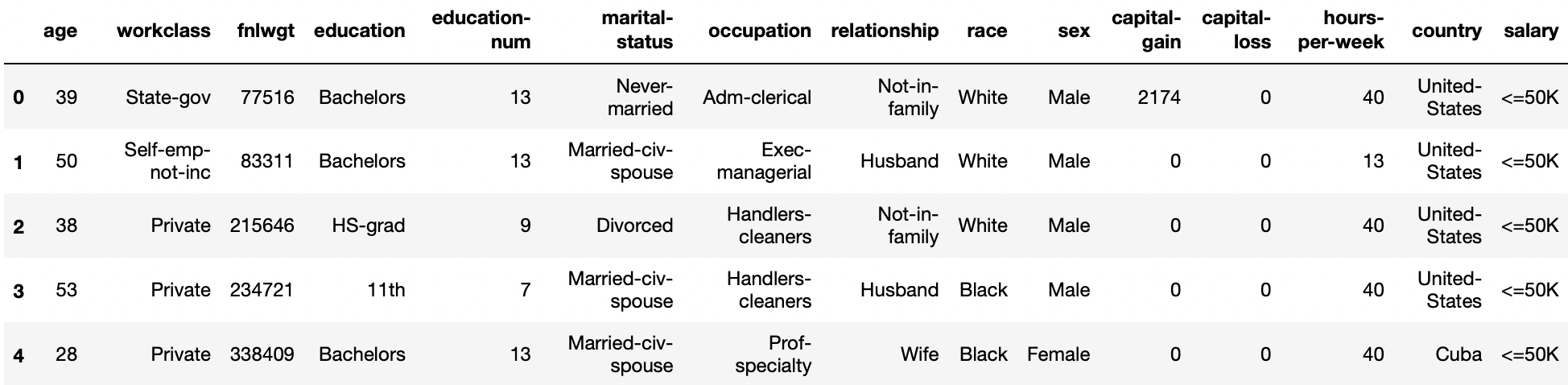
Обзор литературы 13

Постановка задачи и описание датасета

Любые методы обработки данных используются для структурирования и анализа существующей информации. Методов машинного обучения много, однако в этой работе рассмотрены самые основные, частоиспользуемые методы, которые эффективно работают для решения задач классификации данных.

В задаче мы стремимся предсказать, будет ли доход человека больше 50 000 долларов в год, основываясь на нескольких атрибутах из данных переписи.

Для работы взяты открытый набор данных с сайта Kaggle[1]

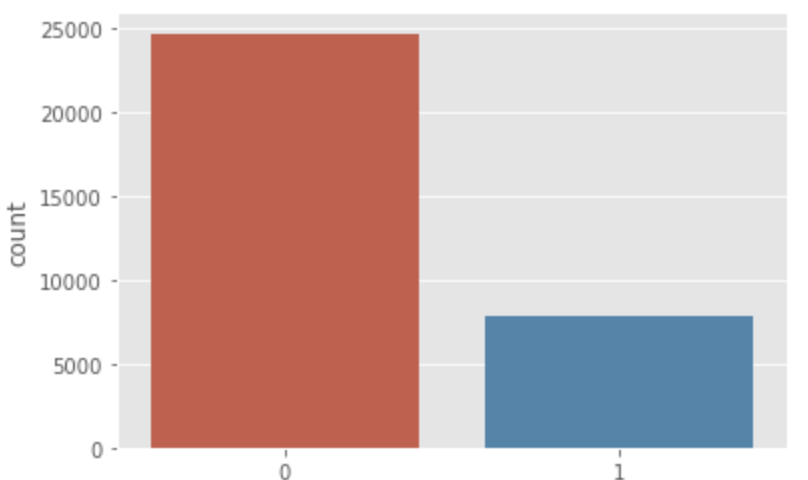


Датасет Census Income содержит 32 561 записи. Каждая запись содержит следующую информацию о человеке:

1. age – возраст (показатель числового типа);
2. workclass – рабочий класс (признак номинального типа): Private, Selfempnotinc, Selfempinc, Federalgov, Localgov, Stategov;
3. fnlwgt – конечный «вес» (влияние) человека (показатель числового типа). Люди со схожими демографическими характеристиками должны иметь одинаковый вес;
4. education – высший уровень образования, достигнутый человеком (признак номинального типа);
5. education-num – высший уровень образования, достигнутый человеком в числовой форме (признак числового типа): Bachelors, Some­college, 11th, HS­grad, Prof­school, Assoc­acdm, Assoc­voc, 9th, 7th­8th, 12th, Masters, 1st­4th, 10th, Doctorate, 5th­6th, Preschool;
6. marital-status – семейное положение человека (признак номинального типа): Married­civ­spouse, Divorced, Never­married, Separated, Widowed, Married­spouse­absent, Married­AF­spouse;
7. occupation – общий вид занятости (признак номинального типа): Tech­support, Craft­repair, Other­service, Sales, Exec­managerial, Prof­specialty, Handlers­cleaners, Machine­op­inspct, Adm­clerical, Farming­fishing, Transport­moving, Priv­house­serv, Protective­serv, Armed­Forces.;
8. relationship – отношения. Представляет, чем этот человек является по отношению к другим (признак номинального типа): Wife, Own­child, Husband, Not­in­family, Other­relative, Unmarried;
9. race – раса или этническая группа (признак номинального типа);
10. sex – пол (номинальный показатель);
11. capital-gain – доходы,прирост капитала (признак числового типа);
12. capital-loss – расходы, убыток капитала (признак числового типа);
13. hours-per-week – работает часов в неделю (признак числового типа);
14. country – страна рождения (признак номинального типа): United­States, Cambodia, England, Puerto­Rico, Canada, Germany, Outlying­US(Guam­USVI­etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican­Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El­Salvador, Trinadad&Tobago, Peru, Hong, Holand­Netherlands.
15. Salary - годовой доход человека (признак номинального типа): “>50K”, “<= 50K”

Целевым (классифицирующим) атрибутом является Salary. Тем самым приходим к выводу о том, что в работе рассматривается задача бинарной классификации.

| Значение параметра | Количество | Процент от общего числа (%) |
| --- | --- | --- |
| <= 50K | 24720 | 75,9 |
| > 50K | 7841 | 24,1 |



Основные определения

# Типы данных

Данные, которые могут быть использованы для анализа, бывают четырех типов [2]:

1. Численные данные (Н-р, стоимость товара: 100 рублей).
2. Интервальные данные (Н-р, доля рынка компании: 5 %).
3. Ранговые данные (Н-р, оценка покупателя: напиток Coca-Cola нра-  
   вится больше, чем Pepsi-Cola).
4. Номинальные данные (Н-р, профессия: ученый, военный,  
   врач).

Любой набор данных может быть адекватно представлен комбинацией перечисленных типов. Для успешного анализа необходимо привести данные к одному типу. Лучше численному.

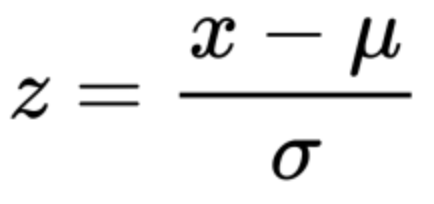
# Методы анализа и обработки данных

Задача классификации — формализованная задача, в которой имеется множество объектов, разделённых некоторым образом на классы. Задано конечное множество (выборка) объектов, для которых известно, к каким классам они относятся. Классовая принадлежность остальных объектов неизвестна. Задача классификации: разработать алгоритм, способный классифицировать произвольный объект из исходного множества. Под классификацией объекта следует понимать указание номера класса, к которому относится данный объект.

В машинном обучении задача классификации решается с использованием обучения с учителем, поскольку классы определяются заранее и для примеров обучающего множества метки классов заданы. Аналитические модели, решающие задачу классификации, называются классификаторами.

В работе рассматриваются 5 разных методов:

1. k-nearest neighbors algorithm  
   knn-классификатор относит объекты к классу, которому принадлежит большинство из k его ближайших соседей в многомерном пространстве признаков. Выбор параметра k противоречив. С одной стороны, увеличение его значения повышает достоверность классификации, но при этом границы между классами становятся менее четкими.  
   Так как разные атрибуты могут иметь разный диапазон представленных значений в выборке, то значения дистанции могут сильно зависеть от атрибутов с бо́льшими диапазонами. Поэтому данные обычно подлежат нормализации - метод, при котором данные масштабируются путем преобразования статистического распределения данных в формат: среднее = 0, ст.отклонение = 1.

  
  
Недостатки: Нет теоретических оснований выбора определенного числа соседей — только перебор. Как правило, плохо работает, когда признаков много  
Параметры: k - число ближайших соседей.

# Decision Tree Classifier Дерево решений представляет собой иерархическую древовидную структуру, состоящую из правила вида «Если …, то ...». За счет обучающего множества правила генерируются автоматически в процессе обучения. Недостатки: Без регулировки глубины дерева часты переобучения. Потому что в процессе построения дерева решений могут создаваться слишком сложные конструкции, которые недостаточно полно представляют данные. Параметры:

max\_depth – максимальная глубина дерева

max\_features — максимальное число признаков, по которым ищется лучшее разбиение в дереве  
min\_samples\_leaf – минимальное число объектов в листе.

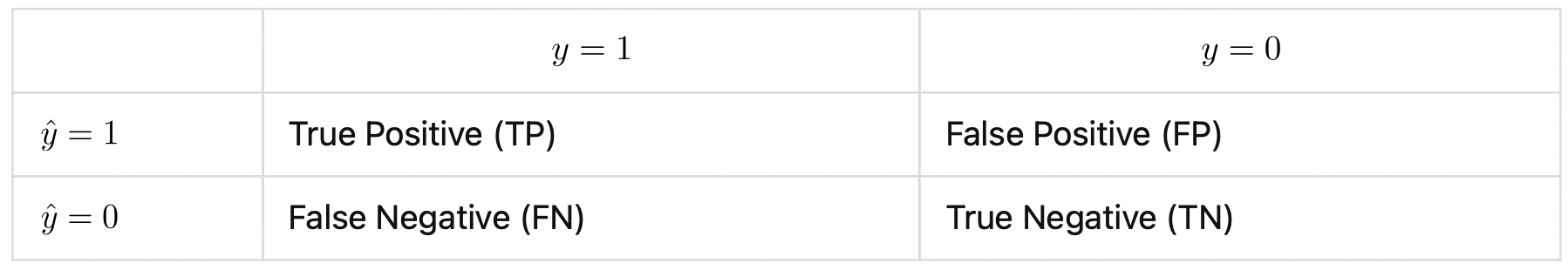
1. Random Forest Classifier

это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству.  
  
параметры:  
n\_estimators - число деревьев  
max\_depth - Максимальная глубина деревьев  
max\_features - число признаков для выбора расщепления  
min\_samples\_split - Минимальное число объектов, при котором выполняется расщепление  
  
Недостатки: Построение Случайных лесов намного сложнее и отнимает больше времени, чем деревья решений.

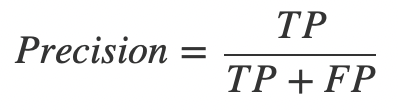
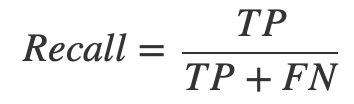
1. Gradient Boosting Classifier  
   Это продвинутый алгоритм машинного обучения для решения задач классификации и регрессии. Он строит предсказание в виде ансамбля слабых предсказывающих моделей, которыми в основном являются деревья решений. Общая идея алгоритма – последовательное применение предиктора (предсказателя) таким образом, что каждая последующая модель сводит ошибку предыдущей к минимуму.  
     
   Параметры:  
   n\_estimators – число итераций в бустинге. Чем больше, тем лучше качество, однако слишком большой увеличение данного параметра может привести к ухудшению производительности и переобучению.  
   min\_samples\_split – минимальное число объектов, при котором происходит расщепление. С данным параметром мы можем избежать переобучение.  
   min\_samples\_leaf – минимальное число объектов в листе (узле). Меньшие значения стоит выбирать для менее сбалансированных выборок.   
   max\_depth – максимальная глубина дерева. Используется для того, чтобы исключить возможность переобучения.
2. XGBClassifier  
   XGBoost (eXtreme Gradient Boosting) является оптимизированной версией алгоритма Gradient Boosting.

# Метрики качества

Рассмотрим матрицу ошибок:



#### Accuracy - доля правильных ответов алгоритма. Одна из самых распространенных метрик. У данной метрики есть важный недостаток - она присваивает каждому классу одинаковый вес, что плохо, в случае дисбаланса классов. На практике это приводит к тому, что accuracy, например, = 80%, но при этом в рамках какого-то конкретного класса классификатор работает плохо, не определяя правильно даже треть значений класса.

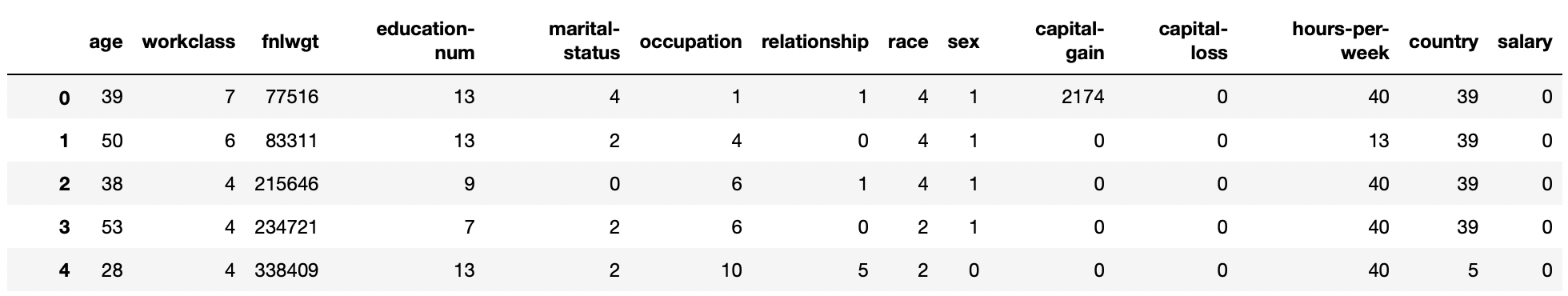
1. Точность(Precision) — то доля наблюдений, действительно принадлежащих данному классу относительно всех наблюдений которые система отнесла к этому классу.   
     
   Precision отображает способность отличать этот класс от других классов.
2. Полнота(Recall) — это доля найденных моделью наблюдений, принадлежащих классу относительно всех наблюдений этого класса в тестовой выборке.  
     
   Recall демонстрирует способность алгоритма обнаруживать данный класс вообще.

## F-мера - гармоническое среднее между точностью и полнотой. Она стремится к нулю, если точность или полнота стремится к нулю.

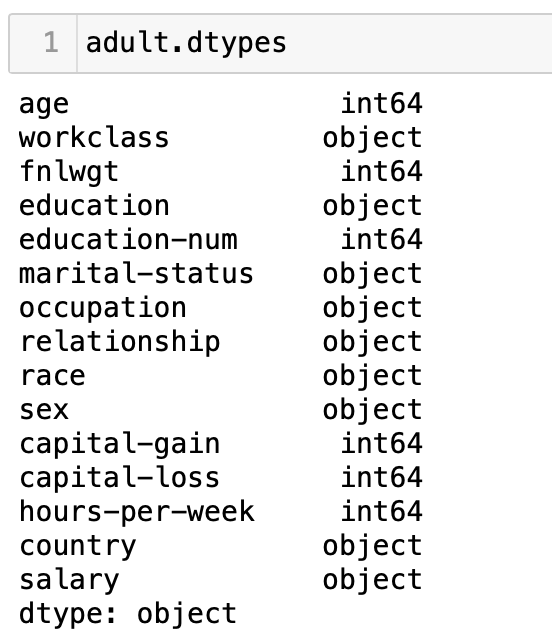
## F1-мера - частный случай F-меры (beta = 1) F-мера является хорошим кандидатом на формальную метрику оценки качества классификатора. Она сводит к одному числу две других основополагающих метрики: точность и полноту. Поэтому будем полагаться именно на значение этой метрики.

Предобработка данных

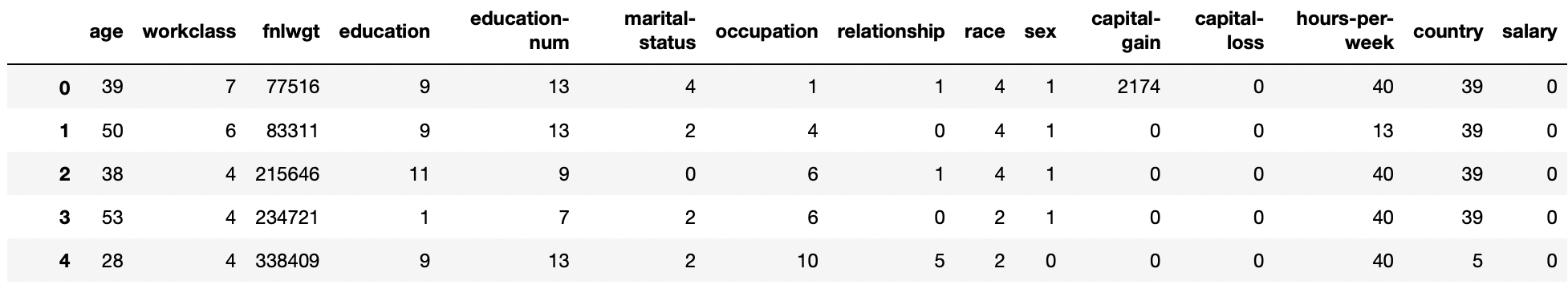
Так как параметры education и education-num несут в себе одну и ту же информацию, удалим один из них из датасета.



С помощью функции dtypes проанализируем типы данных датасета:

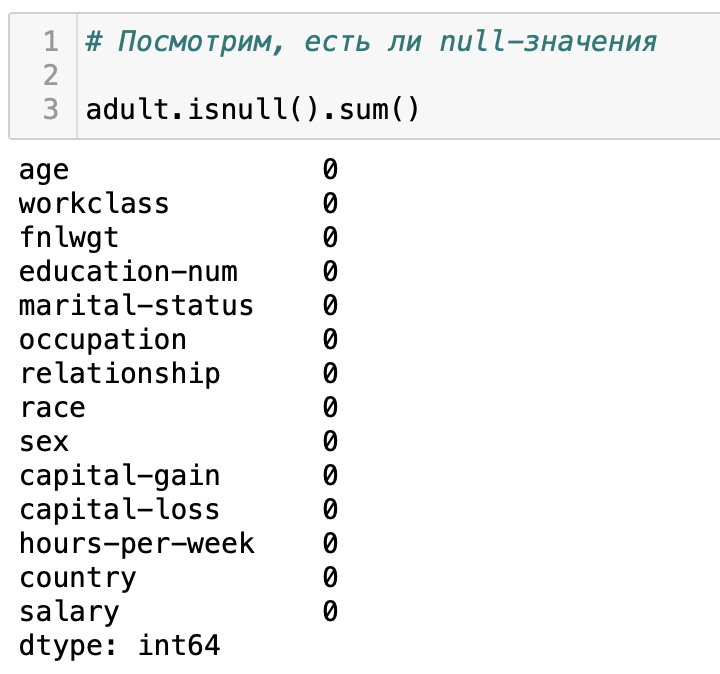


Видно, что есть числовые и номинальные. С помощью функции .fit\_transform из библиотеки sklearn приведем номинальные значения параметров к числовому типу.



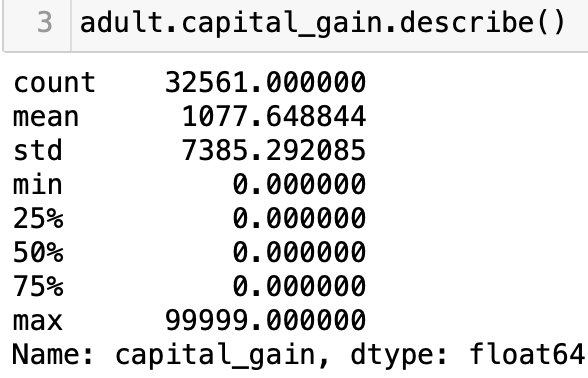
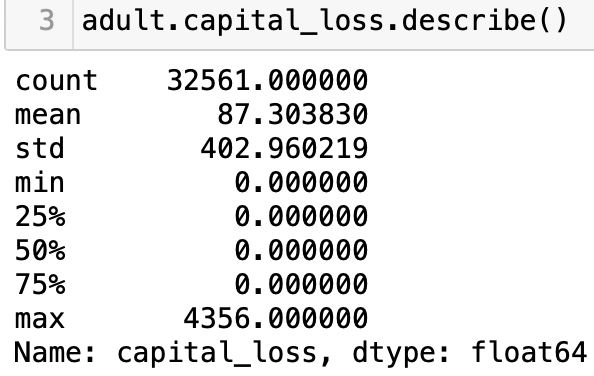
С помощью функции isnull() было обнаружено, что в данных нет пропущенных

значений:

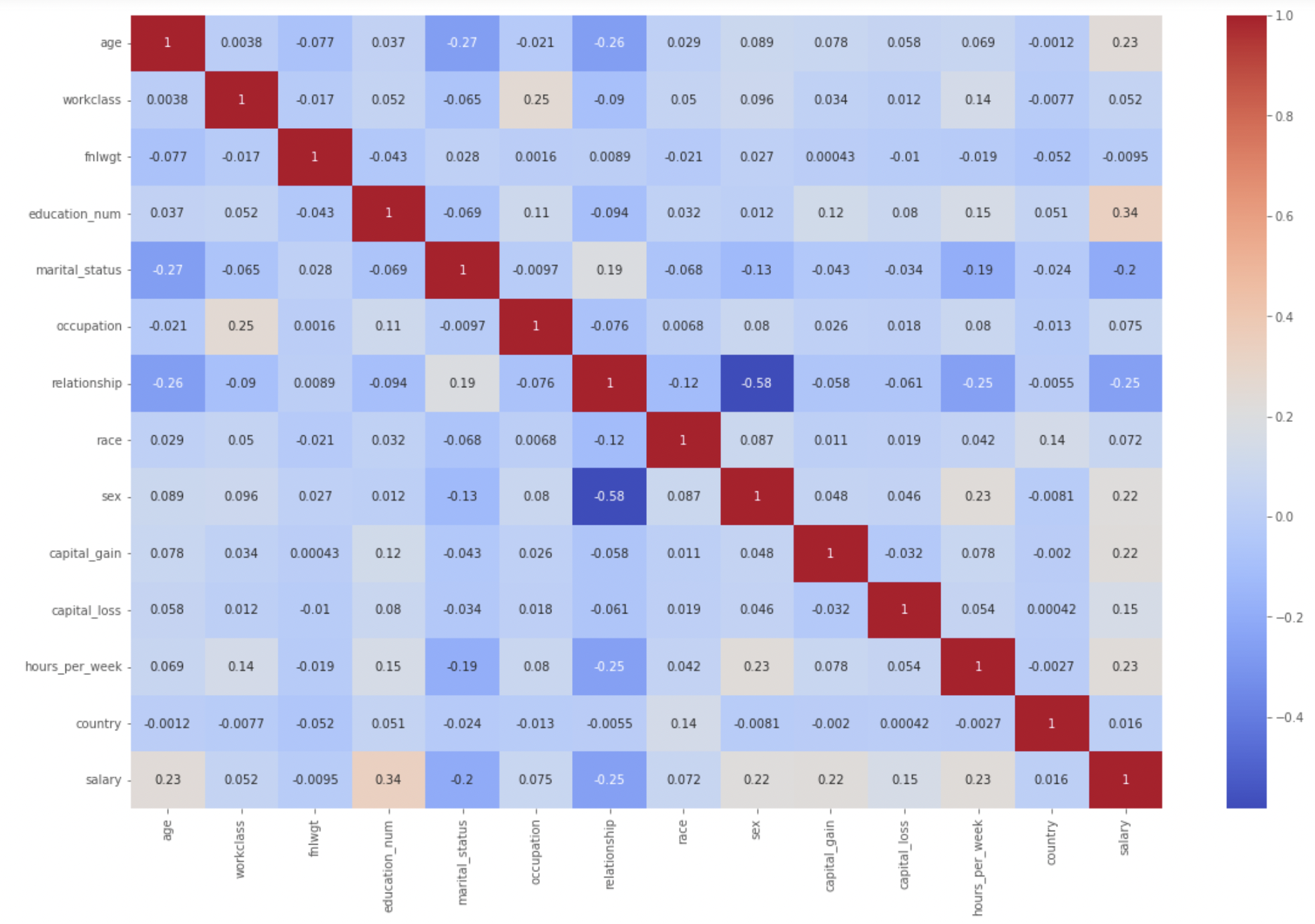


С помощью функции describe() мы можем увидеть различные статистики в

числовых данных. По таблице можно сделать вывод, что параметры capital\_gain и capital\_loss имеют выбросы. При анализе рассмотрим модели, обученные на данных с этими параметрами и без них.

С помощью функции corr() рассмотрим корреляционную матрицу:



По таблице можно сделать вывод, что ни одна пара признаков не имеет коэффициент корреляции выше 0.7, значит все признаки могут быть рассмотрены при анализе.

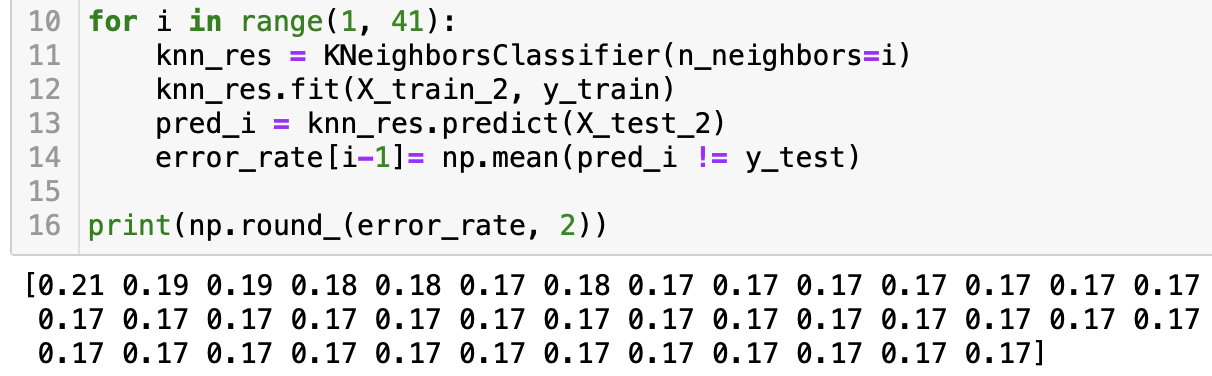
Построение и обучение моделей

Разделим данные на тренировочную и тестовые выборки с помощью функции train\_test\_split() из библиотеки sklearn. Процентное содержание тестовой выборки выберем равным 33%. В итоге получаем 21815 - мощность тренировочной выборки и 10746 - тестовой.

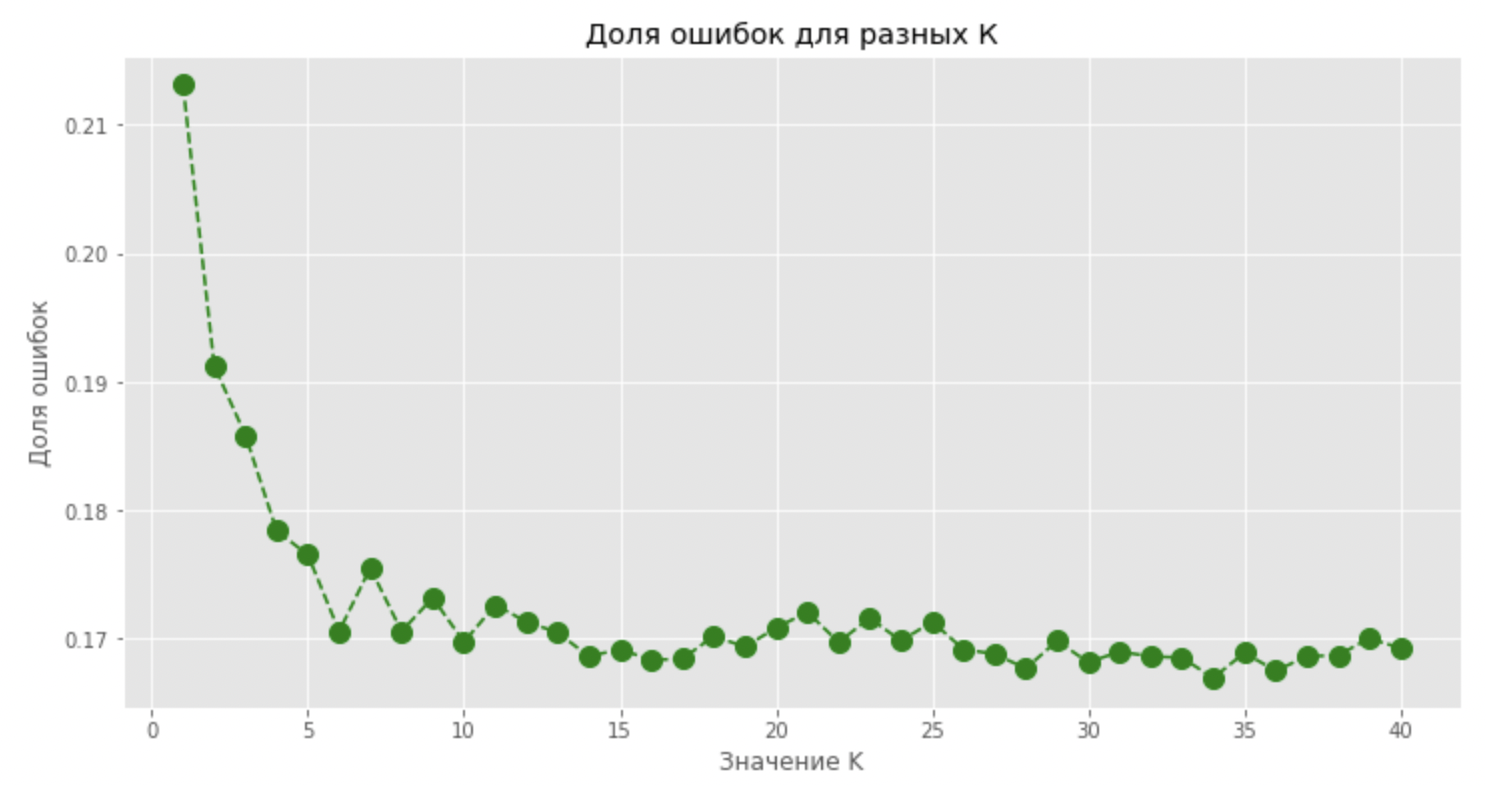
## k-nearest neighbors algorithm

Так как параметры датасета имеют разные диапазоны значений, проведем нормализацию, используя функцию StandardScaler().

Будем строить 40 моделей с разными значениями k = [1,40] и считать долю правильных распознаваний и долю ошибок.

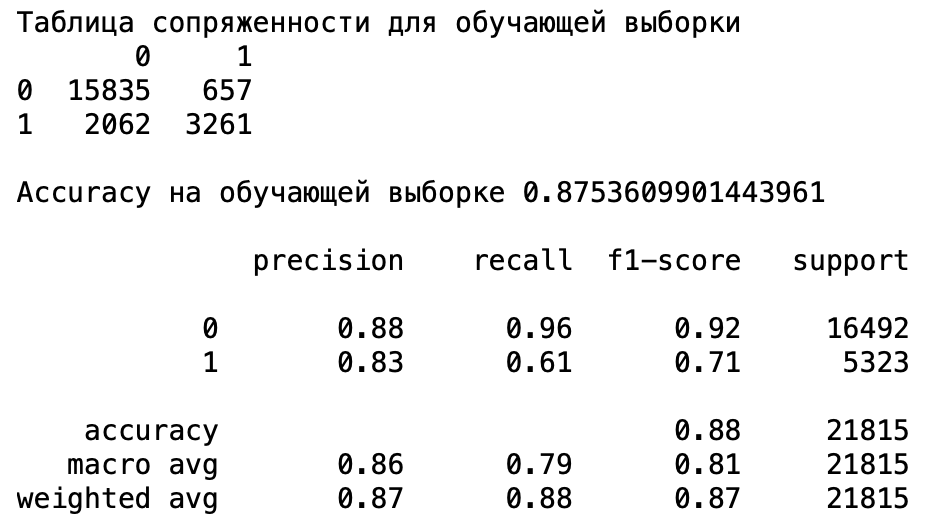
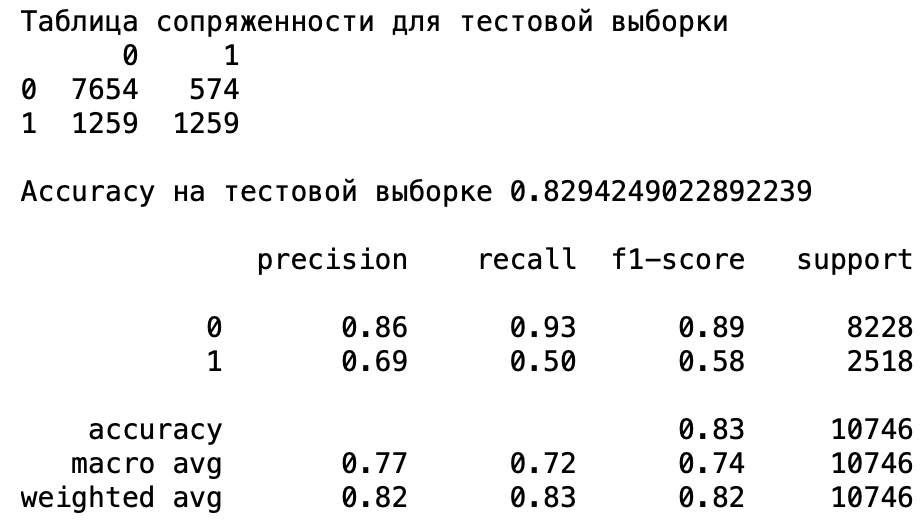


Визуализируем для наглядности:

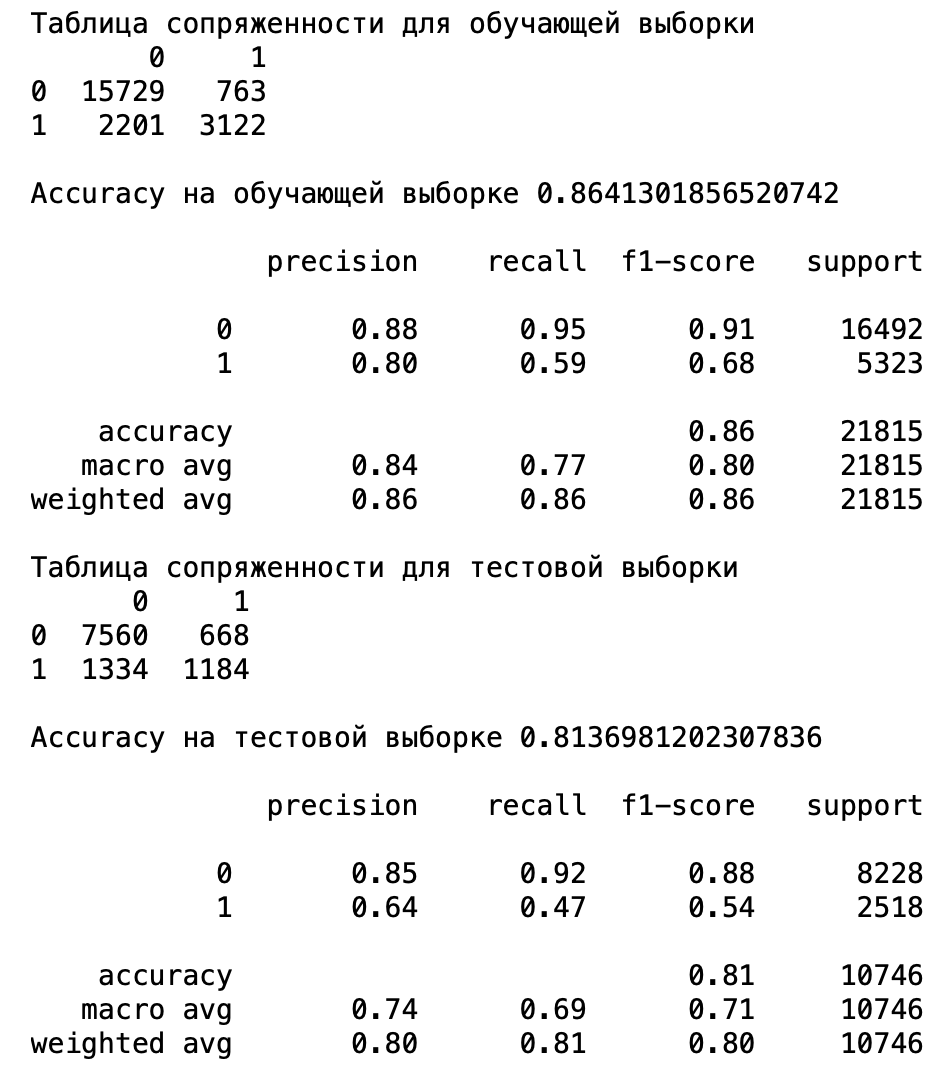


Был сделан вывод, что начиная с k = 6 доля ошибки изменяется незначительно, значит можно остановиться на этом значении.

Оценим качество модели 1 с параметром k = 6:

Значения метрик F1 для модели 2 (без столбцов capital\_gain и capital\_loss) получились меньше на 0,2 и для тестовой, и для тренировочной выборок. Из чего можно сделать вывод, что значения capital\_gain и capital\_loss все же вносили свой вклад в обучение модели.



## Остальные методы

Для последующих методов   
1. Происходит подбор параметров с помощью GridSearchCV(). После этого выбирается модель с лучшими параметрами и делается предсказание для тестовых данных.

2. Так же рассматриваются 2 датасета.

## Общие результаты и выводы

| Метод | параметры | лучшие параметры | accuracy | | F1-мера | |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | train | test | train | test |
| KNN | k = [1,40] | k >= 6 | 0.88 | 0.83 | 0.71 | 0.58 |
| DTC | 'max\_depth': [4, 6, 7],  'min\_samples\_split': [8, 10, 12],  'min\_samples\_leaf': [4, 8, 10] | 'max\_depth': 7, 'min\_samples\_leaf': 10, 'min\_samples\_split': 8 | 0.86 | 0.85 | 0.67 | 0.63 |
| RFC | 'n\_estimators': [30, 40, 50, 100],  'max\_depth': [4, 5, 6], 'max\_features': [1, 3, 10],  'min\_samples\_split': [3, 5, 10, 15] | 'max\_depth': 6, 'max\_features': 10, 'min\_samples\_split': 3, 'n\_estimators': 50 | 0.86 | 0.85 | 0.66 | 0.61 |
| GBC | 'n\_estimators': [400, 500, 450], 'min\_samples\_leaf': [4, 5, 6], 'max\_depth': [4, 5, 6],  'min\_samples\_split': [3, 5, 10, 15] | 'max\_depth': 4, 'min\_samples\_leaf': 5, 'min\_samples\_split': 3, 'n\_estimators': 400 | 0.9 | 0.86 | 0.79 | 0.69 |
| XGBC | 'n\_estimators': [100, 500, 800],  'max\_depth': [3, 5, 6] | 'max\_depth': 6, 'n\_estimators': 100 | 0.89 | 0.87 | 0.76 | 0.69 |

Обзор литературы

[1] [Kaggle. “Adult Census Income.”](https://www.kaggle.com/datasets/uciml/adult-census-income)

[2] [СОВРЕМЕННЫЕ МЕТОДЫ АНАЛИЗА ДАННЫХ](https://riep.ru/upload/iblock/031/031173bb40e099800b248497db44cb88.pdf)

[3] [Задача классификации](https://wiki.loginom.ru/articles/classification-problem.html)

[4] [Метод k-ближайших соседей](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_k-%D0%B1%D0%BB%D0%B8%D0%B6%D0%B0%D0%B9%D1%88%D0%B8%D1%85_%D1%81%D0%BE%D1%81%D0%B5%D0%B4%D0%B5%D0%B9)

[5] [Оценка классификатора](http://bazhenov.me/blog/2012/07/21/classification-performance-evaluation.html)

[6] [Метрики в задачах машинного обучения](https://habr.com/ru/company/ods/blog/328372/)

[7] [Деревья решений](https://habr.com/ru/company/productstar/blog/523044/)

[8] [Открытый курс машинного обучения. Тема 3. Классификация, деревья решений и метод ближайших соседей](https://habr.com/ru/company/ods/blog/322534/)

[9] [Случайный лес](https://dyakonov.org/2016/11/14/%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D1%8B%D0%B9-%D0%BB%D0%B5%D1%81-random-forest/)

[10] [Алгоритм градиентного спуска](https://proglib.io/p/reshaem-zadachi-mashinnogo-obucheniya-s-pomoshchyu-algoritma-gradientnogo-bustinga-2021-11-25)