**МИНОБРНАУКИ РОССИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНЫЙ УНИВЕРСИТЕТ»**

**(РГГУ)**

ОТДЕЛЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ В ГУМАНИТАРНОЙ СФЕРЕ

Кафедра математики, логики

и интеллектуальных систем

в гуманитарной сфере

#### Юдина Анастасия Михайловна

#### АНАЛИЗ ДАННЫХ МИРОВОГО ОТЧЁТА О СЧАСТЬЕ С ПРИМЕНЕНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Курсовая работа студентки 2-го курса

очной формы обучения

Направление 45.03.04 «Интеллектуальные системы в гуманитарной сфере»

|  |  |
| --- | --- |
|  | Научный руководитель:  Кандидат физико-  математических наук, доцент \_\_\_\_\_\_\_\_\_\_\_\_ (Л.О.Шашкин)  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 2022 г. |

Москва 2022

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc104362229)

[ОБЗОР ДАННЫХ 4](#_Toc104362230)

[2015 ГОД 5](#_Toc104362231)

[2016 ГОД 7](#_Toc104362232)

[2017 ГОД 9](#_Toc104362233)

[2018 ГОД 11](#_Toc104362234)

[2019 ГОД 13](#_Toc104362235)

[2020 ГОД 15](#_Toc104362236)

[2021 ГОД 17](#_Toc104362237)

[2022 ГОД 19](#_Toc104362238)

[ЛИНЕЙНАЯ РЕГРЕССИЯ 22](#_Toc104362239)

[СЛУЧАЙНЫЙ ЛЕС 24](#_Toc104362240)

[ГРАДИЕНТНЫЙ БУСТИНГ 26](#_Toc104362241)

[XGBOOST 30](#_Toc104362242)

[CATBOOST 31](#_Toc104362243)

[ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ИСПОЛЬЗУЕМЫХ МЕТРИК 34](#_Toc104362244)

[MEAN SQUARED ERROR 35](#_Toc104362245)

[ROOT MEAN SQUARED ERROR 36](#_Toc104362246)

[MEAN ABSOLUTE ERROR 37](#_Toc104362247)

[R² 38](#_Toc104362248)

[ПРЕДОБРАБОТКА ДАННЫХ 39](#_Toc104362249)

[ВИЗУАЛИЗАЦИЯ ДАННЫХ 43](#_Toc104362250)

[ПОДГОТОВКА К ОБУЧЕНИЮ МОДЕЛЕЙ 52](#_Toc104362251)

[ОБУЧЕНИЕ МОДЕЛЕЙ 53](#_Toc104362252)

[ЗАКЛЮЧЕНИЕ 56](#_Toc104362253)

[СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ 57](#_Toc104362254)

[ПРИЛОЖЕНИЕ 1 58](#_Toc104362255)

# ВВЕДЕНИЕ

Машинное обучение – широко используемая область, которая помогает человеку анализировать данные, выводить из них закономерности и на их основании делать предсказания. Историю машинного обучения можно начать с 1950-х годов, когда начали разрабатывать первые программы для игры в шашки. Позже список задач, которые можно решить, применяя методы машинного обучения, значительно расширился. Сегодня машинное обучение используется для диагностики заболеваний, предсказания платёжеспособности пациента для выдачи кредита, предсказания рыночной стоимости квартиры, машинного перевода, определения места для бурения скважин нефти, распознавания почерка и ещё в большом количестве сфер, на первый взгляд не связанных с компьютерными технологиями. В данной работе рассматривается задача предсказания уровня счастья.

В классическом машинном обучении используются два основных подхода:

* Обучение с учителем
* Обучение без учителя

Обучение с учителем применяется для заранее размеченных данных и решает два основных вида задач

* Классификация – т.е. предсказание категориального признака
* Регрессия – т.е. предсказание количественного признака

В этой работе рассматривается задача регрессии, поскольку в данных признак «уровень счастья» представляет собой вещественное число в диапазоне от 0 до 10.

Итак, **цель** данной работы – подготовить прототип модели машинного обучения для эффективного предсказания уровня счастья.

Соответственно, **задачи**:

1. Загрузить и изучить данные
2. Предобработать данные
3. Провести исследовательский анализ данных
4. Построить и обучить модель

# ОБЗОР ДАННЫХ

Данные, используемые в работе, взяты с сайта Kaggle и находятся в открытом доступе. Они содержат материалы Всемирного доклада о счастье, ежегодно публикуемого подразделением ООН. Первый доклад был выпущен в 2012 году, а с 2016 года он выходит ежегодно 20 марта, приуроченный к Международному дню счастья ООН. Поскольку тема счастья является довольно значимой для людей, были разработаны разные шкалы для измерения его уровня. Например, шкала субъективного счастья С. Любомирски [Lyubomirsky, Lepper, 1999], шкала удовлетворённости жизнью, разработанная Эдом Динером и другие. В том числе существует шкала, называемая лестницей Кэнтрила. Именно она используется в докладе о мировом счастье. Подход заключается в том, что испытуемого просят вообразить лестницу со ступенями, пронумерованными от 0 до 10, где 0 находится у подножья, а 10 – на вершине, в которой верхняя ступень показывает наилучшую возможную для респондента жизнь, а нижняя – наихудшую. Задача испытуемого отметить, на какой ступени лестницы он находится в данный период жизни.

Уровень счастья – один из столбцов, встречающийся в каждом датасете. Помимо него, в данных есть другие показатели, отличающиеся в разных датафреймах.

В работе рассмотрен период с 2015 по 2022 год. Далее будет представлена информация по каждому году – в каком состоянии были данные на момент начала работы с ними.

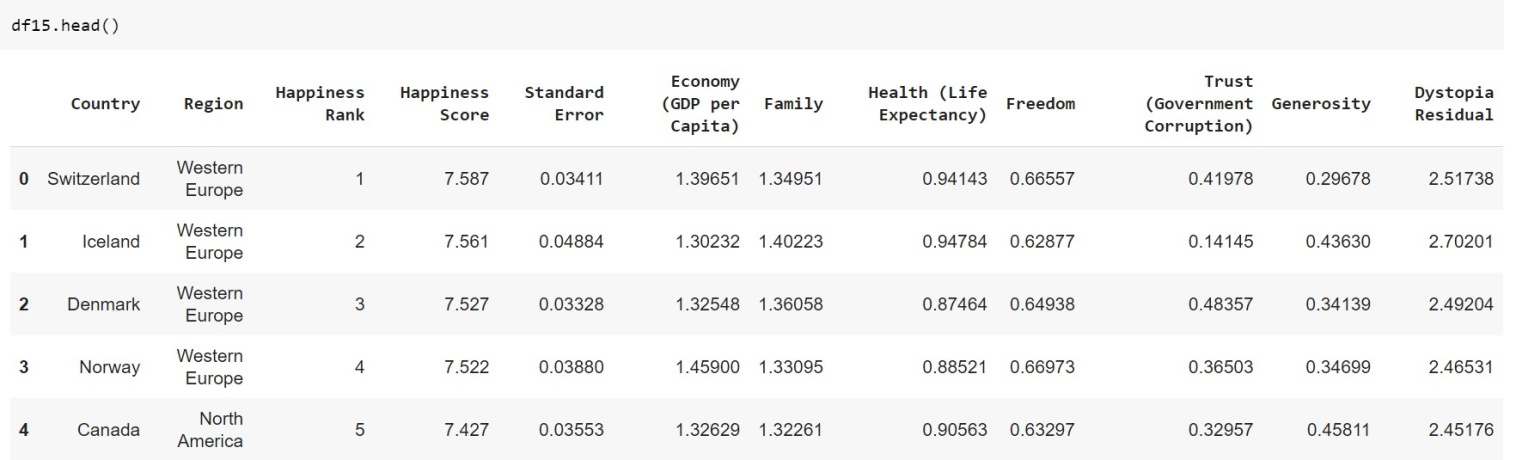
# 2015 ГОД

Таблица за 2015 год содержит 158 строк и 12 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Country – название страны, тип данных – объект
* Region – регион, в котором находится страна, тип данных – объект
* Happiness Rank – ранг страны по уровню счастья, целое число от 1 до 158, тип данных – целое число
* Happiness Score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Standard Error – стандартная ошибка оценки счастья, вещественное число от 0 до 1, тип данных - число с плавающей точкой
* Economy (GDP per Capita) – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Family – оценка, показывающая уровень семейной поддержки, тип данных – число с плавающей точкой
* Health (Life Expectancy) – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Trust (Government Corruption) – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Dystopia Residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Отметим, что в данных за 2015 год не было обнаружено пропусков.

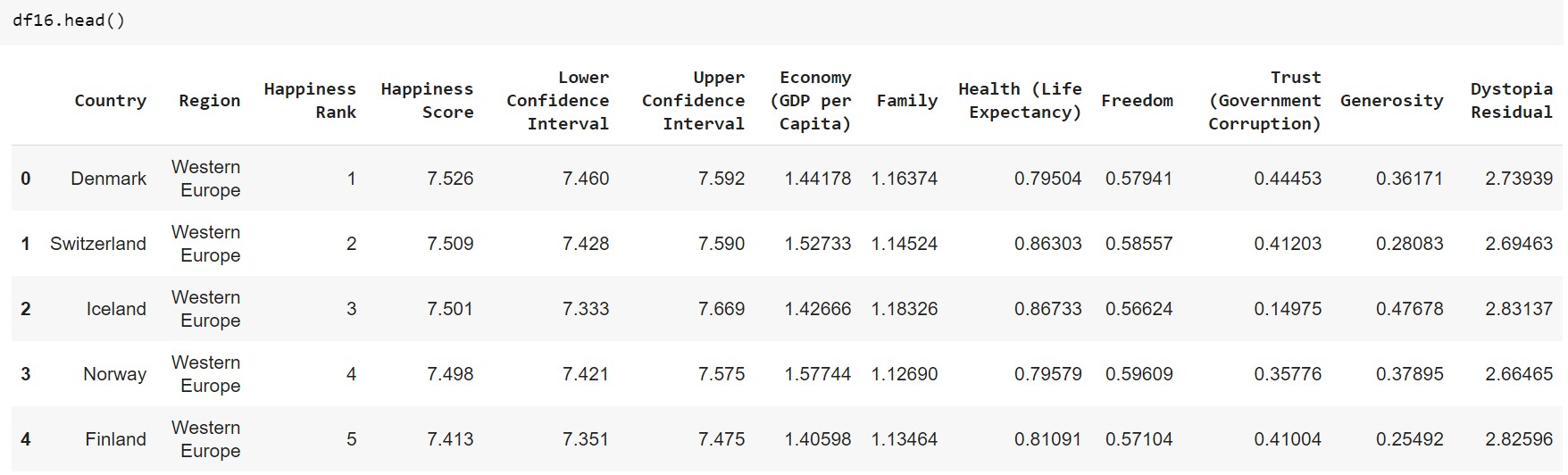
# 2016 ГОД

Таблица за 2016 год содержит 157 строк и 13 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Country – название страны, тип данных – объект
* Region – регион, в котором находится страна, тип данных – объект
* Happiness Rank – ранг страны по уровню счастья, целое число от 1 до 158, тип данных – целое число
* Happiness Score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Lower Confidence Interval – нижний доверительный интервал, тип данных - число с плавающей точкой
* Upper Confidence Interval – верхний доверительный интервал, тип данных - число с плавающей точкой
* Economy (GDP per Capita) – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Family – оценка, показывающая уровень семейной поддержки, тип данных – число с плавающей точкой
* Health (Life Expectancy) – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Trust (Government Corruption) – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Dystopia Residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Отметим, что в данных на 2016 год нет пропусков.

# 2017 ГОД

Таблица за 2017 год содержит 155 строк и 12 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Country – название страны, тип данных – объект
* Happiness.Rank – ранг страны по уровню счастья, целое число от 1 до 158, тип данных – целое число
* Happiness.Score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Whisker.high – верхний доверительный интервал, тип данных - число с плавающей точкой
* Whisker.low – нижний доверительный интервал, тип данных - число с плавающей точкой
* Economy..GDP.per.Capita. – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Family – оценка, показывающая уровень семейной поддержки, тип данных – число с плавающей точкой
* Health..Life.Expectancy. – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Trust..Government.Corruption. – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Dystopia.Residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных за 2017 год не встречается пропусков.

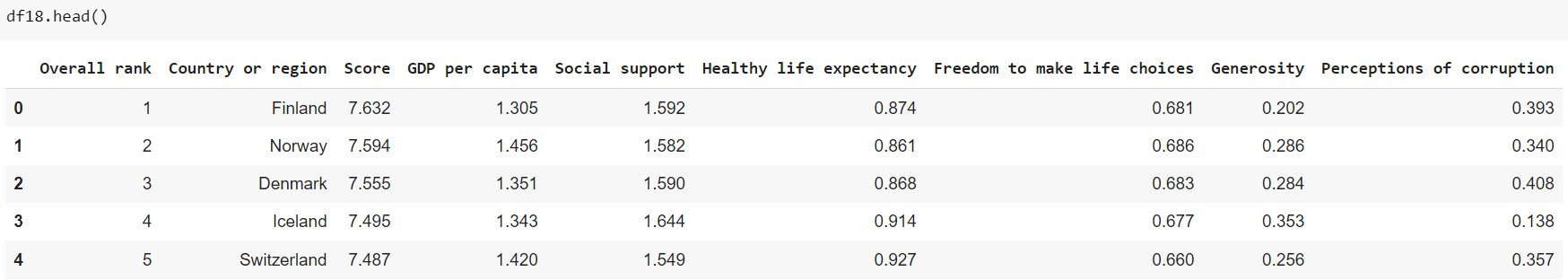
# 2018 ГОД

Таблица за 2018 год содержит 156 строк и 9 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Overall rank – ранг страны по уровню счастья, целое число от 1 до 156, тип данных – целое число
* Country or region – название страны, тип данных – объект
* Score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* GDP per capita – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Social support – оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Healthy life expectancy – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom to make life choices – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Perceptions of corruption – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных на 2018 год есть пропуски.

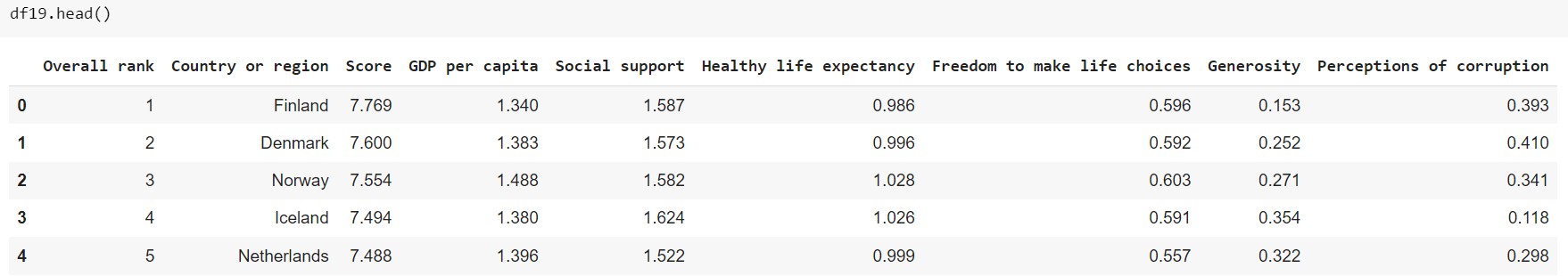
# 2019 ГОД

Таблица за 2019 год содержит 156 строк и 9 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Overall rank – ранг страны по уровню счастья, целое число от 1 до 156, тип данных – целое число
* Country or region – название страны, тип данных – объект
* Score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* GDP per capita – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Social support – оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Healthy life expectancy – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom to make life choices – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Perceptions of corruption – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных на 2019 год нет пропусков.

# 2020 ГОД

Таблица за 2020 год содержит 153 строки и 20 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Country name – название страны, тип данных – объект
* Regional indicator – название региона, тип данных - объект
* Ladder score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Standard error of ladder score – стандартная ошибка для уровня счастья, тип данных – число с плавающей точкой
* Upperwhisker – верхний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* Lowerwhisker – нижний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* Logged GDP per capita – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Social support – оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Healthy life expectancy – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom to make life choices – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Perceptions of corruption – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Ladder score in Dystopia – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой
* Explained by: Log GDP per capita - оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Explained by: Social support - оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Explained by: Healthy life expectancy - оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Explained by: Freedom to make life choices - оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Explained by: Generosity - оценка уровня щедрости, тип данных – число с плавающей точкой
* Explained by: Perceptions of corruption - оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Dystopia + residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных на 2020 год нет пропусков.

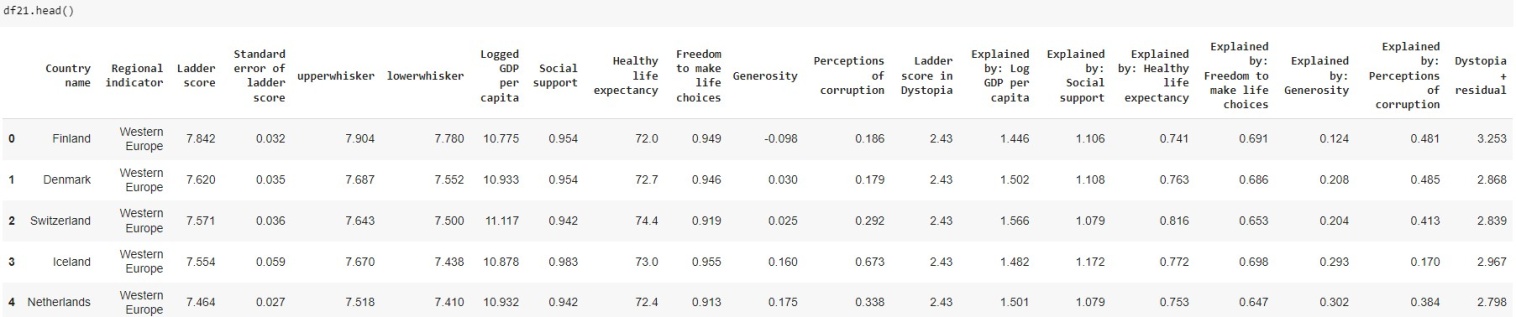
# 2021 ГОД

Таблица за 2021 год содержит 149 строк и 20 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* Country name – название страны, тип данных – объект
* Regional indicator – название региона, тип данных - объект
* Ladder score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Standard error of ladder score – стандартная ошибка для уровня счастья, тип данных – число с плавающей точкой
* upperwhisker – верхний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* lowerwhisker – нижний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* Logged GDP per capita – оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Social support – оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Healthy life expectancy – оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Freedom to make life choices – оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Generosity – оценка уровня щедрости, тип данных – число с плавающей точкой
* Perceptions of corruption – оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Ladder score in Dystopia – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой
* Explained by: Log GDP per capita - оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Explained by: Social support - оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Explained by: Healthy life expectancy - оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Explained by: Freedom to make life choices - оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Explained by: Generosity - оценка уровня щедрости, тип данных – число с плавающей точкой
* Explained by: Perceptions of corruption - оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Dystopia + residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных на 2021 год нет пропусков.

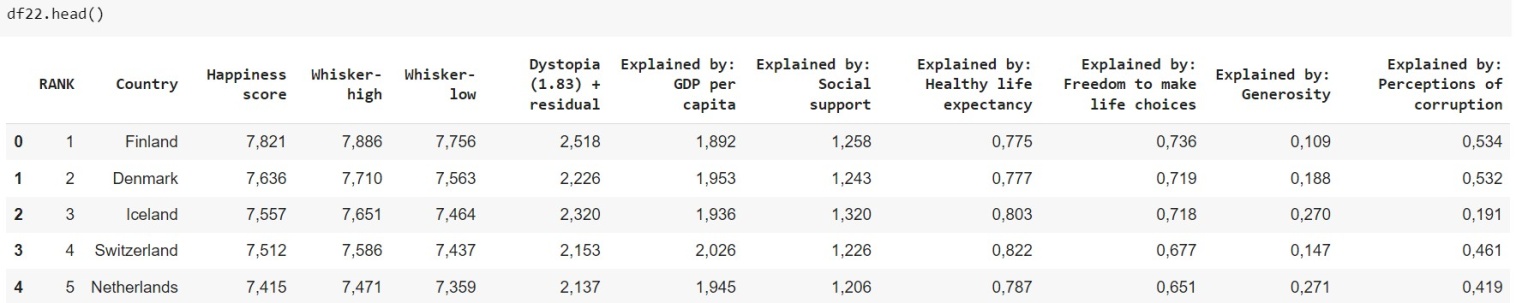
# 2022 ГОД

Таблица за 2022 год содержит 147 строк и 12 столбцов. В каждой строке показатели для одной страны. Список стран можно посмотреть в приложении 1.

Список столбцов и типов данных значений в них:

* RANK - ранг страны по уровню счастья, целое число от 1 до 156, тип данных – целое число
* Country – название страны, тип данных – объект
* Happiness score – средний уровень счастья в стране, вещественное число от 0 до 10, тип данных – число с плавающей точкой
* Whisker-high – верхний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* Whisker-low – нижний доверительный интервал для уровня счастья, тип данных – число с плавающей точкой
* Dystopia (1.83) + residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой
* Explained by: GDP per capita - оценка, показывающая ВВП на душу населения, тип данных – число с плавающей точкой
* Explained by: Social support - оценка, показывающая уровень социальной поддержки, тип данных – число с плавающей точкой
* Explained by: Healthy life expectancy - оценка, показывающая среднюю продолжительность жизни, тип данных – число с плавающей точкой
* Explained by: Freedom to make life choices - оценка, показывающая уровень свободы, тип данных – число с плавающей точкой
* Explained by: Generosity - оценка уровня щедрости, тип данных – число с плавающей точкой
* Explained by: Perceptions of corruption - оценка, показывающая уровень коррупции, тип данных – число с плавающей точкой
* Dystopia + residual – оценка уровня счастья страны относительно воображаемой страны-антиутопии, с самыми несчастными людьми, тип данных – число с плавающей точкой

На рисунке ниже представлены первые 5 строк таблицы.



Стоит отметить, что в данных на 2022 год есть пропуски.

**ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ИСПОЛЬЗУЕМЫХ АЛГОРИТМОВ**

Алгоритмы машинного обучения определяют параметры модели на основе данных. Алгоритмов существует много, и классифицируются они в зависимости от типа решаемой задачи. Часть алгоритмов подходит для задач классификации, такие как логистическая регрессия, метод k ближайших соседей и другие, часть алгоритмов решает задачи регрессии, такие как, например, линейная регрессия, а существуют и те, которые подходят для решения задач обоих видов, например, дерево решений или случайный лес.

В работе используются следующие алгоритмы машинного обучения:

* Линейная регрессия
* Случайный лес
* Базовый градиентный бустинг
* Xgboost
* Catboost

Далее каждый из них будет рассмотрен подробно.

# ЛИНЕЙНАЯ РЕГРЕССИЯ

Несмотря на то, что существуют более сложные модели, нейронные сети, линейные модели до сих пор используются, поскольку являются легко интерпретируемыми и не переобучаются.

В общем случае линейна функция выглядит следующим образом:

y = w1x1+…+wDxD+w0, где

y – целевая переменная,

(x1,…,xD) – вектор, соответствующий объекту выборки (**вектор признаков, вектор весов**), а w1,…,wD,w0 – параметры модели.

Линейная регрессия подгоняет линейную модель с коэффицентами w=(w1,…,wD) к минимизации остаточной суммы квадратов между наблюдаемым целевым признаком в наборе данных и предсказанным целевым признаком по линейной аппроксимации, иными словами находит вектор (w0,w1,…,wD)∈RD+1.

Стоит отметить особенность линейных моделей – каждый объект должен быть представлен вектором из чисел x1,…,xD, что важно в рассматриваемой в работе задаче. Модель называют линейной, если она является линейной по этим вещественным признакам.

В случае единственного признака (D = 1) линейная функция будет задавать предсказание значения в точке.



В случае более высоких размерностей вместо прямой будет гиперплоскость с аналогичным смыслом.

Метод [LinearRegression](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html" \l "sklearn.linear_model.LinearRegression" \t "_blank) библиотеки sklearn имеет метод **fit** который принимает матрицу **X** и целевой признак y и будет хранить коэффиценты w линейной модели в перменной **coef\_.**

# СЛУЧАЙНЫЙ ЛЕС

Случайный лес относится к ансамблевым методам, которые объединяют прогнозы нескольких базовых оценок, построенных с заданным алгоритмом обучения, чтобы улучшить метрики качества по сравнению с одной оценкой. Кроме того, он относится к, так называемым, методам усреднения, которые строят несколько оценщиков независимо, а затем усредняют их прогнозы.

В случае случайного леса базовым алгоритмом является решающее дерево. Алгоритм принятия решения решающим деревом интуитивно понятен человеку. Он предсказывает значения целевой переменной, применяя последовательность правил (вопросов, предикатов).

 Пусть задано бинарное дерево, в котором:

* каждой внутренней вершине v приписан предикат  Bv: X→{0,1};
* каждой листовой вершине v приписан прогноз cv∈Y, где Y — область значений целевой переменной (в случае классификации листу может быть также приписан вектор вероятностей классов).

В ходе предсказания осуществляется проход по этому дереву к некоторому листу. Для каждого объекта выборки x движение начинается из корня. В очередной внутренней вершине v проход продолжится вправо, если Bv(x)=1, и влево, если Bv(x)=0. Проход продолжается до момента, пока не будет достигнут некоторый лист, и ответом алгоритма на объекте x считается прогноз cv, приписанный этому листу.

Стоит отметить, что предикат Bv может иметь произвольную структуру, но, как правило, на практике используют сравнение с порогом t∈R по какому-то j-му признаку:

Bv(x,j,t)=[xj≤t]

При проходе через узел дерева с данным предикатом объекты будут отправлены в правое поддерево, если значение j-го признака у них меньше либо равно t, и в левое — если больше.

Случайный лес из решающих деревьев строится следующим образом:

Для построения i-го дерева из обучающей выборки X выбирается с возвращением случайная подвыборка Xi того же размера. Затем случайно выбираются n < N признаков, где N – полное число признаков. На n выбранных признаках подвыборки Xi строится i-ое дерево.

Чтобы получить предсказания на тестовой выборке ответы отдельных деревьев усредняются.

В модуле sklearn.ensemble для случайного леса ([RandomForestRegressor](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html#sklearn.ensemble.RandomForestRegressor)) основными параметрами являются следующие:

* n\_estimators — число деревьев
* criterion — критерий для разбиения выборки в вершине
* max\_features — число признаков, по которым ищется разбиение
* min\_samples\_leaf — минимальное число объектов в листе
* max\_depth — максимальная глубина дерева

# ГРАДИЕНТНЫЙ БУСТИНГ

Ещё один способ собирать из базовых алгоритмов композицию – градиентный бустинг. В бустинге каждый следующий алгоритм старается уменьшить ошибку текущего ансамбля.

Градиентный бустинг — это частный случай бустинга. Если мы обучим единственное решающее дерево, то качество модели, скорее всего, будет низкое, но нам известно про это дерево, на каких объектах оно давало правильные ответы, а на каких ошибалась. Поэтому следующую модель можно обучить таким образом, чтобы на каждом объекте сумма ответов моделей приближалась к истинному значению. Например, если на объекте xi первое решающее дерево предсказывало значение на 5 больше истинного b1(xi)=yi+5, то новая модель на объекте xi выдавала бы значение -5. И сумма получилась бы следующая:

b1(xi)+b2(xi)=(yi+10)+(−10)=yi

Другими словами, если вторая модель научится предсказывать разницу между реальным значением и ответом первой, то это позволит уменьшить ошибку композиции.

В реальности вторая модель тоже не сможет обучиться идеально, поэтому обучим третью, которая будет «компенсировать» неточности первых двух. Будем продолжать так, пока не построим композицию из нужного количества алгоритмов.

Далее будет разобран пример для задачи регрессии и квадратичной функции потерь:

→min

Для решения строится композицию из K базовых алгоритмов семейства B:

a(x)=aK(x)=b1(x)+b2(x)+⋯+bK(x)

В качестве базовых выбирается, семейство B решающих деревьев фиксированной глубины.

Далее обучается алгоритм b1(x)∈B, который наилучшим образом приближает целевую переменную:

Поскольку построенный алгоритм b1(x) работает не идеально, вычисляется разница предсказания и истинных значений:

После b1(x) корректируется с помощью b2(x). В лучшем случае так, чтобы b2(x) идеально предсказывал величины , потому что

a2(xi)=b1(xi)+b2(xi)=b1(xi)+=b1(xi)+(yi−b1(xi))=yi

Итак, второе решающее дерево будет обучаться предсказывать разности

Ожидается, что композиция из двух таких моделей a2(x)=b1(x)+b2(x) станет более качественно предсказывать целевую переменную y.

Далее рассуждения повторяются до построения всей композиции. На k-ом шаге вычисляется разность между правильным ответом и текущим предсказанием композиции из k−1 алгоритмов:

Затем k-й алгоритм учится предсказывать эту разность:

а композиция в целом обновляется по формуле

ak(x)=ak−1(x)+bk(x)

Обучение K базовых алгоритмов завершает построение композиции.

Для того, чтобы обобщить алгоритм построения бустинга на произвольную дифференцируемую функцию потерь будет посчитана производная функции потерь по предсказанию z=ak(xi) модели для i-го объекта:

Разность, на которую обучается k-й алгоритм, выражается через производную:

Таким образом, для каждого объекта xi очередной алгоритм в бустинге обучается предсказывать антиградиент функции потерь по предсказанию модели  в точке ak(xi) предсказания текущей части композиции на объекте xi.

Далее, обучение на разность  заменяется обучением на антиградиент функции потерь (−), где

При построении очередного базового алгоритма bk+1 решается задача регрессии с целевым признаком, равным антиградиенту функции потерь исходной задачи на предсказании ak=b1+…+bk

Важно выбрать оценочную функцию S, которая будет показывать, насколько текущая структура дерева хорошо приближает антиградиент. Её нужно будет использовать для построения критерия ветвления:

|R|⋅S(R)−|Rright|⋅S(Rright)−|Rleft|⋅S(Rleft)→max,

где S(R) – значение функции S в вершине R, S(Rleft),S(Rright) – значения в левом и правом сыновьях R после добавления предиката, ∣⋅∣∣⋅∣ – количество элементов, пришедших в вершину.

В итоге обучение базового алгоритма проходит в два шага:

1. по функции потерь вычисляется целевая переменная для обучения следующего базового алгоритма:
2. строится регрессионное дерево на обучающей выборке (xi,−), минимизирующее выбранную оценочную функцию.

# XGBOOST

XGBoost – разновидность градиентного бустинга. Впервые фреймворк был представлен в 2016 году. Алгоритм был разработан таким образом, чтобы оптимально использовать аппаратные ресурсы. Это достигается путём создания внутренних буферов в каждом потоке для хранения статистики градиента.  Одной из особенностей XGBoost является то, что деревья решений строятся по принципу параллелизации. Для улучшения времени работы алгоритма инициализация проходит при считывании данных, а не во вложенном цикле, затем выполняется сортировка, использующая параллельные потоки. Кроме того, алгоритм сам штрафует сложные модели, чтобы избежать переобучения, легко справляется с разряженными данными, заполняя 0 в процессе обучения в зависимости от значения функции потерь, умеет самостоятельно обрабатывать пропуски, использует метод взвешенных квантилей для оптимизации работы со взвешенным датасетом и имеет встроенную кросс-валидацию, чтобы лучше работать с датасетами разных размеров.

# CATBOOST

Библиотека CatBoost написана разработчиками из Яндекса. Этот алгоритм использует небрежные деревья решений, в которых для создания левых и правых потомков на каждом уровне дерева используются одинаковые функции. Одним из основных преимуществ CatBoost является то, что он умеет работать как с количественными, так и с категориальными признаками, которые для многих моделей необходимо предварительно обрабатывать. Кроме того, алгоритм самостоятельно умеет заполнять пропущенные значения. Для решения многих задач в CatBoost можно не подбирать гиперпараметры, поскольку заданные по умолчанию дают высокие результаты.

Общие параметры в CatBoost:

* loss\_function или objective – показатель, используемый для обучения. Есть регрессионные показатели, такие как среднеквадратичная ошибка для регрессии и logloss для классификации.
* eval\_metric – метрика, используемая для обнаружения переобучения.
* Iterations – максимальное количество построенных деревьев, по умолчанию 1000. Альтернативные названия num\_boost\_round, n\_estimators и num\_trees.
* learning\_rate или eta – скорость обучения, которая определяет насколько быстро или медленно модель будет учиться. Значение по умолчанию обычно равно 0.03.
* random\_seed или random\_state– случайное зерно, используемое для обучения.
* l2\_leaf\_reg или reg\_lambda – коэффициент при члене регуляризации L2 функции потерь. Значение по умолчанию – 3.0.
* bootstrap\_type – определяет метод сэмплинга весов объектов, например это может быть Байес, Бернулли, многомерная случайная величина или Пуассон.
* depth = глубина дерева.
* grow\_policy – определяет, как будет применяться жадный алгоритм поиска. Может стоять в значении SymmetricTree, Depthwise или Lossguide. По умолчанию SymmetricTree. В SymmetricTree дерево строится уровень за уровнем, пока не достигнет необходимой глубины. На каждом шаге листья с предыдущего дерева разделяются с тем же условием. При выборе параметра Depthwise дерево строится шаг за шагом, пока не достигнет необходимой глубины. Листья разделяются с использованием условия, которое приводит к лучшему уменьшению потерь. В Lossguide дерево строится по листьям до тех пор, пока не будет достигнуто заданное количество листьев. На каждом шаге разделяется нетерминальный лист с лучшим уменьшением потерь.
* min\_data\_in\_leaf или min\_child\_samples – это минимальное количество обучающих сэмплов в листе. Этот параметр используется только с политиками роста Lossguide и Depthwise.
* max\_leaves или num\_leaves – этот параметр используется только с политикой Lossguide и определяет количество листьев в дереве.
* ignored\_features — указывает на признаки, которые нужно игнорировать в процессе обучения.
* nan\_mode – метод работы с пропущенными значениями. Параметры Forbidden, Min и Max. При использовании Forbidden наличие пропущенных значений вызовет ошибку. При использовании параметра Min пропущенные значения будут приняты за максимальные значения для данного признака. В Max пропущенные значения будут приняты как минимальные значения для данного признака.
* leaf\_estimation\_backtracking– тип бэктрекинга, использующийся при градиентном спуске. По умолчанию используется AnyImprovement. AnyImprovement уменьшает шаг спуска до того, как значение функции потерь будет меньшим, чем оно было на последней итерации. Armijo уменьшает шаг спуска до тех пор, пока не будет выполнено [условие Вольфе](https://en.wikipedia.org/wiki/Wolfe_conditions#Armijo_rule_and_curvature).
* boosting\_type — схема бустинга. Она может быть простой для классической схемы градиентного бустинга или упорядоченной, что обеспечит лучшее качество на небольших наборах данных.
* score\_function – тип [оценки](https://catboost.ai/docs/concepts/algorithm-score-functions.html), используемой для выбора следующего разбиения при построении дерева. Cosine используется по умолчанию. Другие доступные варианты L2, NewtonL2 и NewtonCosine.
* early\_stopping\_rounds— если стоит True, устанавливает тип детектора переобучения в Iter и останавливает обучение, когда достигается оптимальное значение.
* classes\_count – количество классов для задач мультиклассификации.
* task\_type – используете вы CPU или GPU. По умолчанию стоит CPU.
* devices — идентификаторы устройств GPU, которые будут использоваться для обучения.
* cat\_features — массив с категориальными столбцами.
* text\_features —используется для объявления текстовых столбцов в задачах классификации.

# ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ИСПОЛЬЗУЕМЫХ МЕТРИК

В задачах машинного обучения для оценки качества моделей используются различные метрики. Они оценивают работу модели и выражаются в числовой форме. Например, для некоторых задач классификации метриками являются точность и полнота.

* Точность (precision) показывает, какая доля объектов, которым модель присваивает 1 класс, действительно к нему относятся.
* Полнота (recall) показывает, какую долю объектов, относящихся к 1 классу, модель выделила.

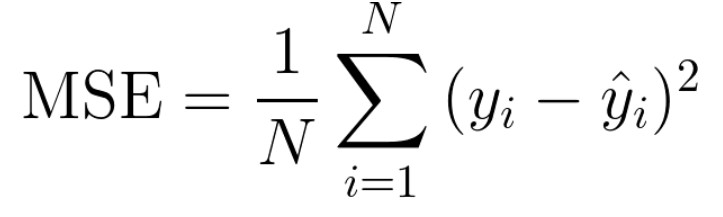
Но поставленная в работе задача относится к задаче регрессии, поэтому используются следующие метрики:

* MSE
* RMSE
* MAE
* R2

Далее каждая из них будет рассмотрена подробно.

# MEAN SQUARED ERROR

Средняя квадратичная ошибка – наиболее распространённая метрика для оценки качества в задачах регрессии. Вычисляется она по следующей формуле:



где N – количество объектов, yi – фактически ожидаемый результат, а ŷᵢ - предсказание модели.

Для каждого предсказания вычисляется разница между прогнозом и целью, затем, с целью убрать разницу между переоценкой и недооценкой, возводится в квадрат, а после полученные значения усредняются.

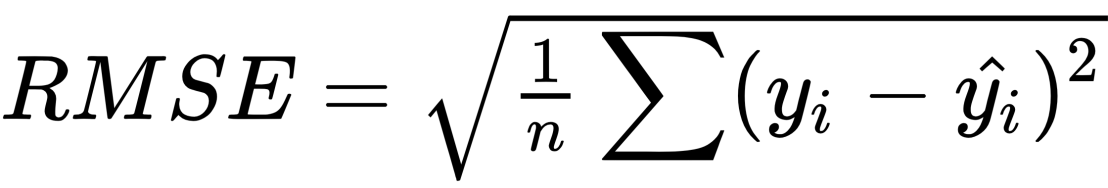
Чем выше получившееся значение, тем хуже качество модели.

Преимуществом данной метрики можно назвать то, что она неустойчива к выбросам. Одно очень высокое значение сильно увеличит MSE и заставит обратить на себя внимание.

Однако эту же особенность можно отнести и к недостаткам метрики. В результате возведения в квадрат одного плохого прогноза, можно предположить о низком качестве работы модели и недооценить её, что может быть особенно заметно при зашумлённых данных.

# ROOT MEAN SQUARED ERROR

Следующая метрика – корень из средней квадратической ошибки. Она была введена для того, чтобы масштаб ошибок был сопоставим с масштабом цели.

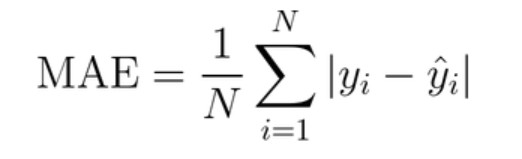


Поскольку для MSE и RMSE выполняется соотношение

MSE(a) > MSE(b) ⬄RMSE(a) > RMSE(b), то для RMSE аналогично: чем больше значение метрики, тем меньше качество модели

# MEAN ABSOLUTE ERROR

Средняя абсолютная ошибка рассчитывается как среднее абсолютных значений разности между значениями целевого признака и прогнозами



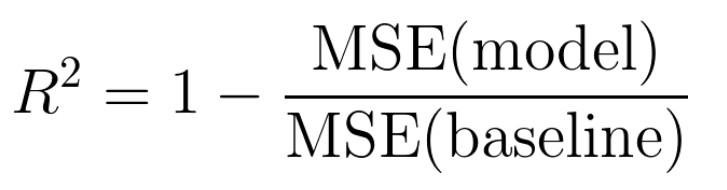
Преимущество данной метрики заключается в том, что она, в отличие от MSE, не так сильно штрафует единичные плохие прогнозы.

Стоит отметить, что когда предсказание является совершенным, градиент не определён и формально MAE не дифференцируема. Однако, предсказания почти никогда в точности не совпадают с истинными значениями, но даже если это будет так, можно добавить ветвление, которое в этом случае выдавало бы 0.

Чем больше значение метрики, тем хуже качество модели.

# R²

Коэффициент детерминации вычисляет долю средней квадратичной ошибки модели от средней квадратичной ошибки среднего, а затем вычитает получившуюся величину из 1.



Его преимущество в том, что эта величина всегда находится в промежутке от **-∞** до 1. В том случае, если значение R2 отрицательно, модель предсказывает хуже среднего.

Чем выше значение метрики, тем лучше качество модели.

# ПРЕДОБРАБОТКА ДАННЫХ

В этом разделе будет рассказано о том, с какими трудностями пришлось столкнуться в ходе предобработки данных и какие были приняты решения для их преодоления.

Первое, на что можно обратить внимание при просмотре данных – разное количество признаков – столбцов в датасетах. В результате анализа было отобрано 6 колонок, несущих в себе важную информацию и встречающихся во всех датасетах. Кроме того, столбцы для удобства были переименованы. Ниже представлен их список

* Рейтинг (score)
* Страна (country)
* ВВП (GDP)
* Социальная поддержка (support)
* Здоровье (health)
* Свобода (freedom)
* Щедрость (generosity)
* Уровень коррупции (corruption)

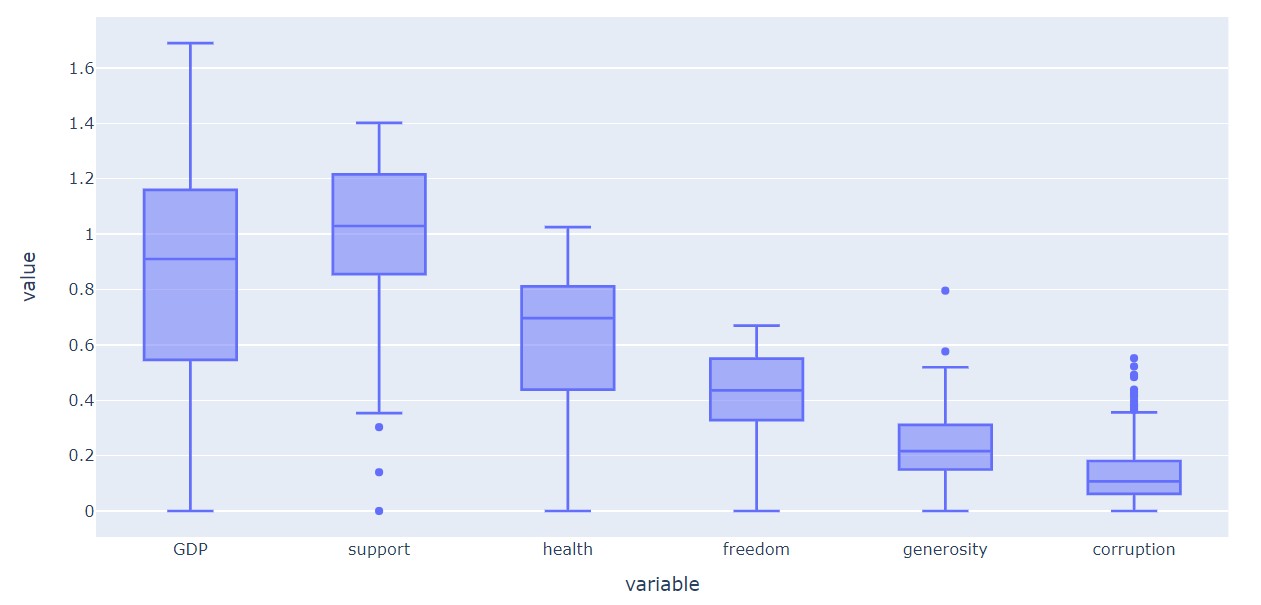
Прочие столбцы были удалены из таблиц при помощи составления списка нужных таблиц и реализации функции для удаления лишних.



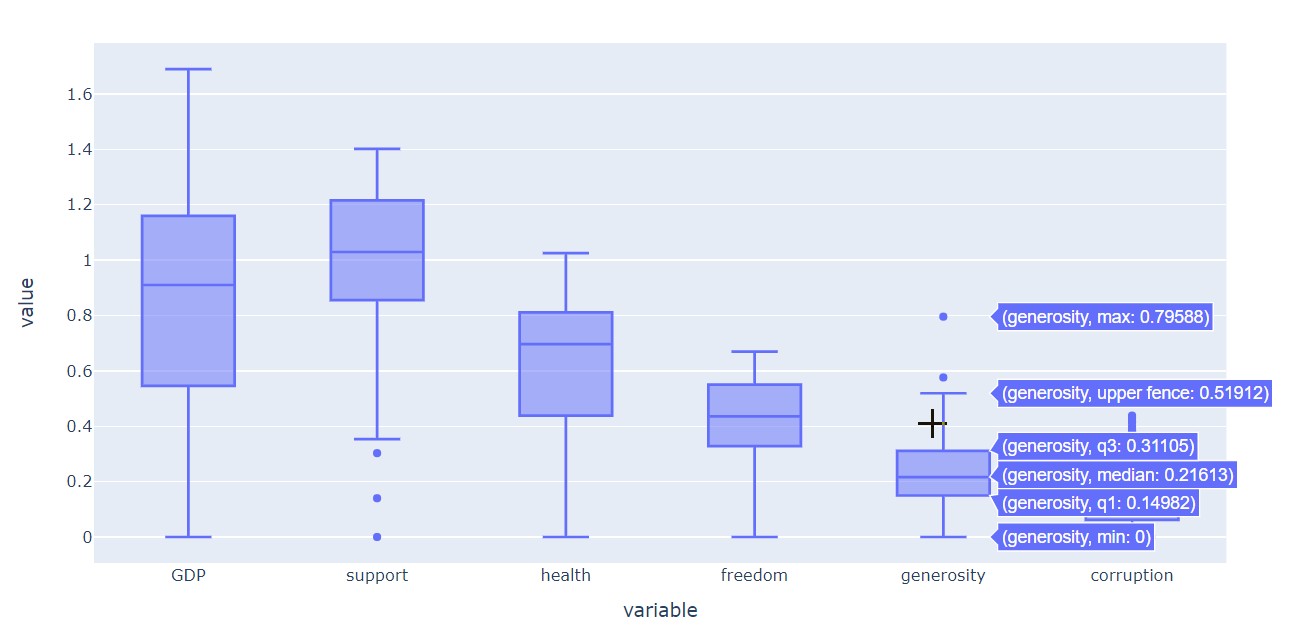
Пропусков в датафреймах оказалось меньше 1 процента, поэтому они были удалены.

Дубликатов в таблицах обнаружено не было.

Для проверки на аномальные значения с помощью средств библиотеки plotly были построены ящики с усами по всем значениям признаков.



Преимущество использования библиотеки plotly заключается в интерактивности графиков. При желании можно навести курсор на заинтересовавший график и подробнее посмотреть на полученные значения

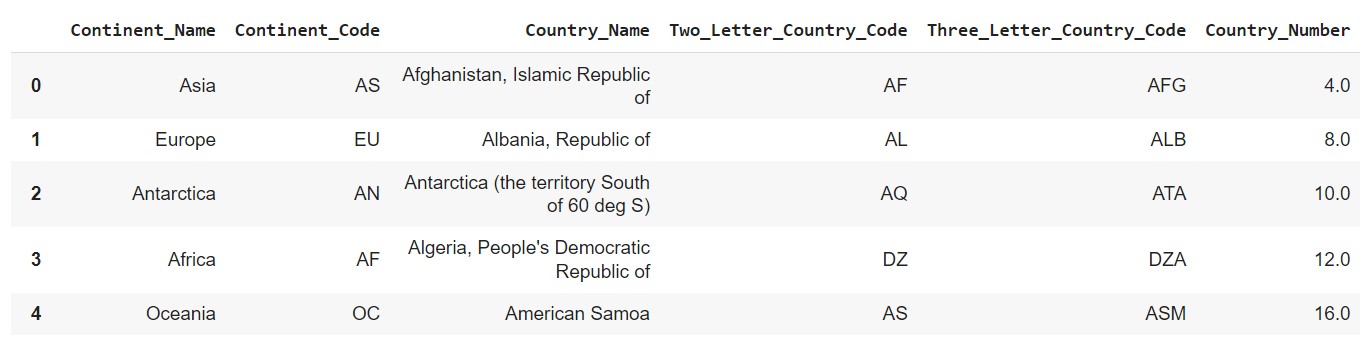


В результате анализа на выбросы были обнаружены выбивающиеся значения, но позже было принято решение не избавляться от них, поскольку они не искажают общую картину и выглядят правдоподобно, пусть и встречаются единожды.

Далее, нужно было подготовить данные к визуализации.

Для более наглядного представления данных было решено добавить в них информацию о континенте, к которому относится страна. В открытом доступе на Гитхабе был найден датасет, содержащий информацию о стране и континенте, к которому она относится.

Первые 5 строк датасета представлены ниже.



Как видно, названия стран отличаются от тех, которые встречаются у нас в данных – здесь названия более подробные, поэтому для использования этого датасета с целью добавить к нашим данным континент придётся использовать предикат *in.* Реализация функции добавления континента.



При помощи этой функции в наши данные был добавлен столбец ‘continent’. Для стран, континент которым не удалось добавить таким образом, он был добавлен вручную и взят из общедоступной информации.

После этого была замечена ещё одна проблема в данных. Тип данных у всех float столбцов в датафрейме за 2022 год оказался object, поскольку в качестве разделителя была использована не точка, а запятая. При помощи лямба-функций данные были исправлены и приведены к нужным типам.



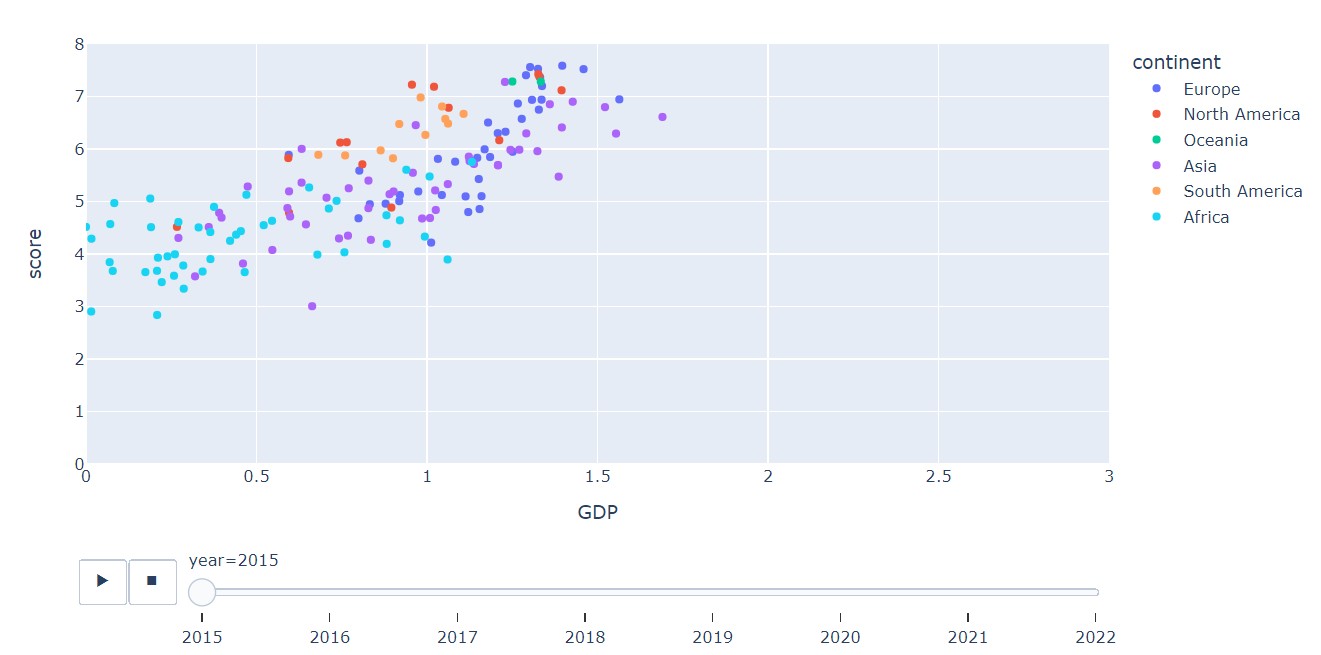
Позже был создан итоговый датафрейм с информацией по всем годам, в который был добавлен столбец с номером года. Общий датафрейм был так же предобработан и итоговое количества строк в нём составило 1229.

# ВИЗУАЛИЗАЦИЯ ДАННЫХ

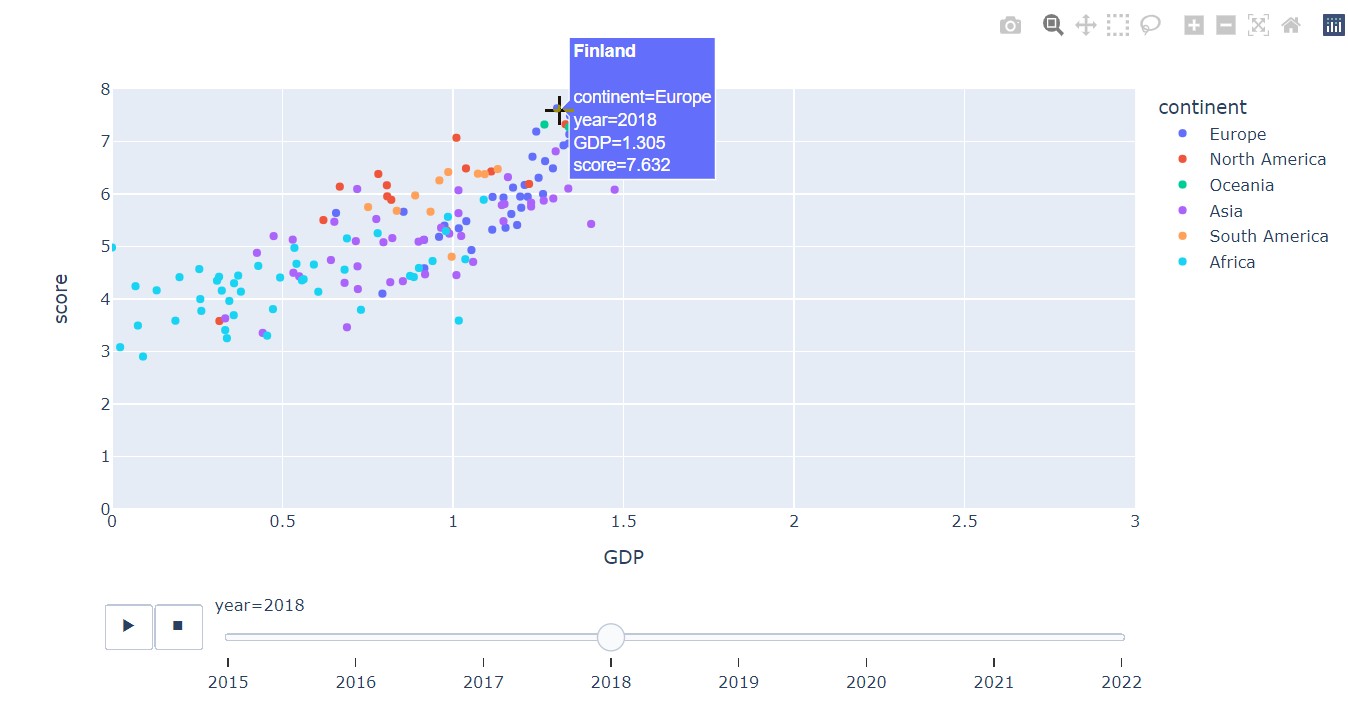
Для визуализации данных в основном были использованы средства библиотеки plotly.

Первый вид графиков, использованный в работе – диаграммы рассеяния, показывающие, как менялась корреляция разных величин с уровнем счастья с течением времени.

Ниже представлен график корреляции ВВП и уровня счастья

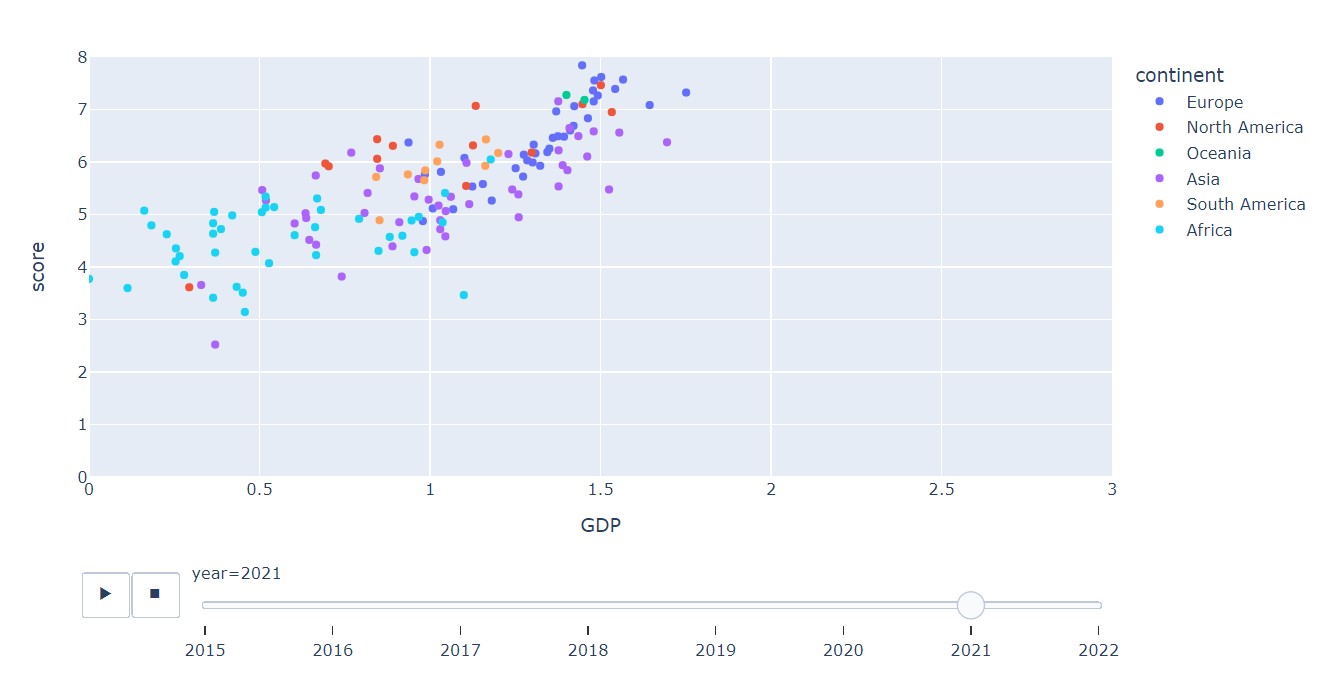


Этот график является интерактивным. Разными цветами обозначены разные континенты для упрощения визуализации. При необходимости, если нажать на легенде на любой цвет, точки этого цвета исчезнут. Если навести курсор на любую точку, высветится информация о том, какую страну данная точка обозначает, её оценку ВВП и уровня счастья.



Кроме того, нижний скролл меняет год и им можно управлять как вручную, при помощи мышки, так и запустив его при помощи нажатия на специальную клавишу.

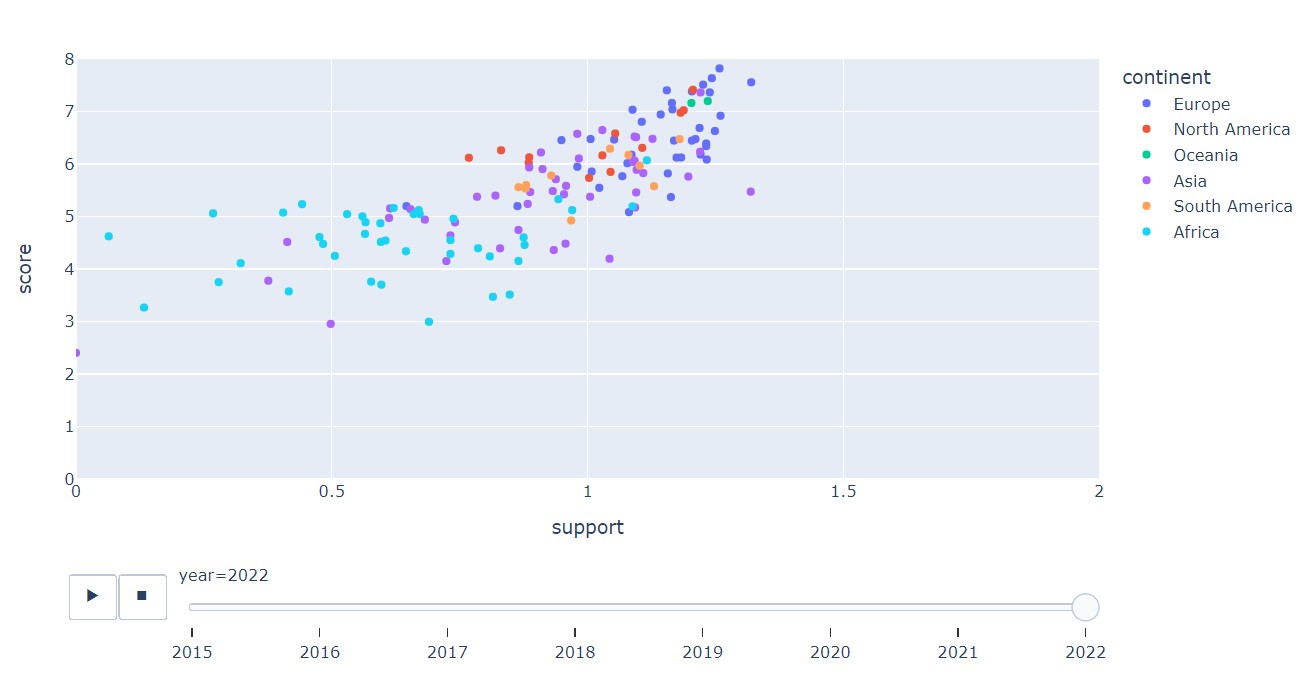
К примеру, данные за 2021 год с той же информацией выглядят следующим образом:



В целом на интерактивном графике видна положительная корреляция в каждый из годов. Можно также отметить, что в Европе наблюдается наибольший уровень как оценки счастья, так и ВВП, а в Африке в среднем оба эти показатели ниже.

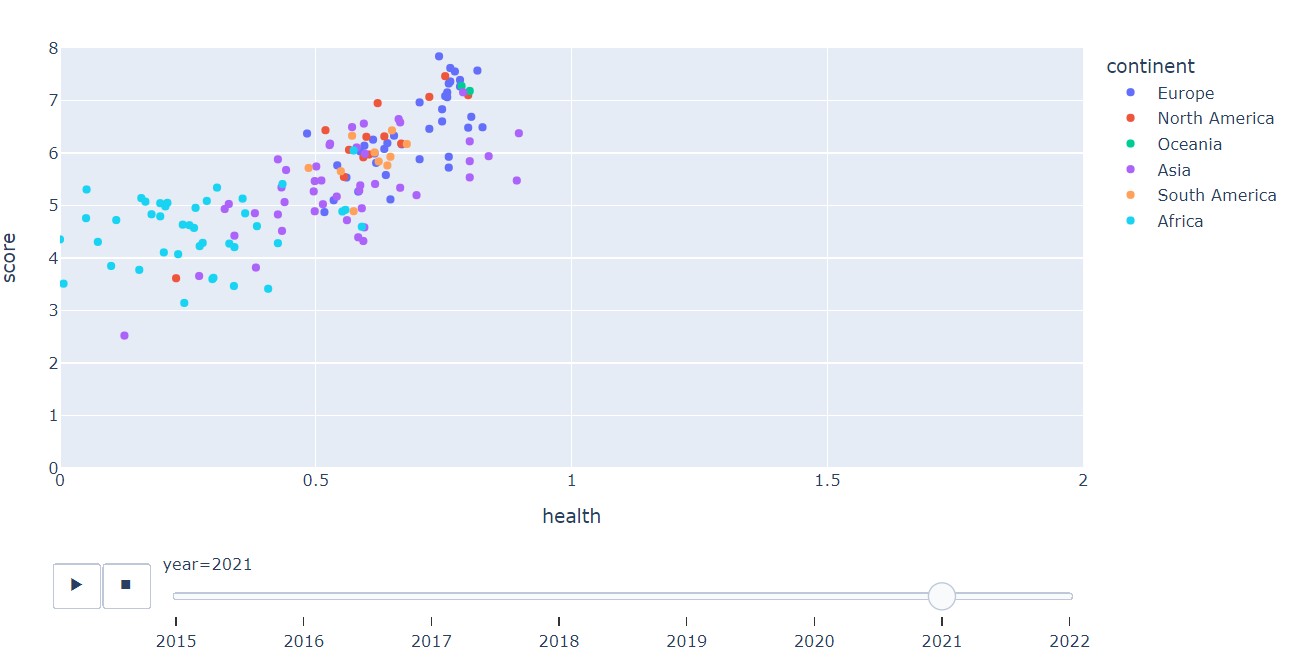
Аналогичные графики были построены для всех остальных признаков. Для более компактного изложения материала покажем для каждого признака график за один год.

Взаимосвязь между уровнем социальной поддержки и уровнем счастья за 2022 год представлена на графике ниже



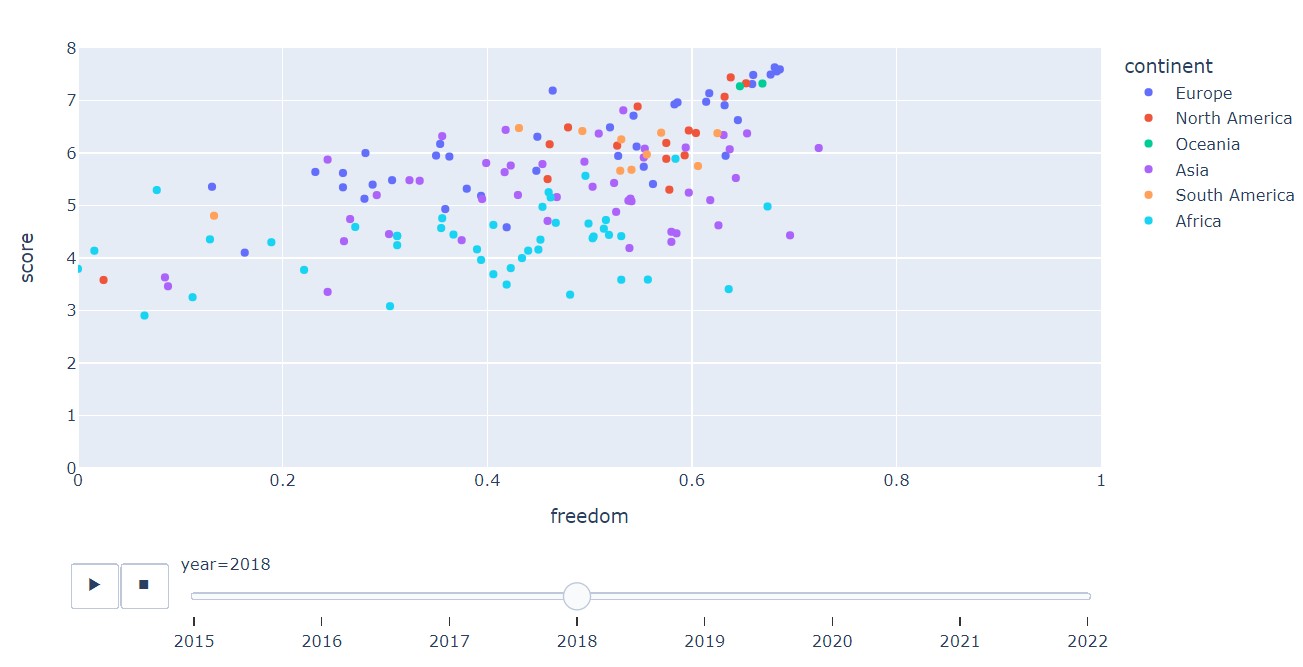
По графику можно сделать аналогичные выводы относительно континентов – страны Европы в среднем имеет выше и показатель социальной поддержки и показатель уровня счастья, в то время как у стран Африки оба эти показателя ниже, в страны остальных континентов в среднем располагаются где-то между ними.

Далее представлен график корреляции показателя, отвечающего за здоровье с показателем уровня счастья за 2021 год.



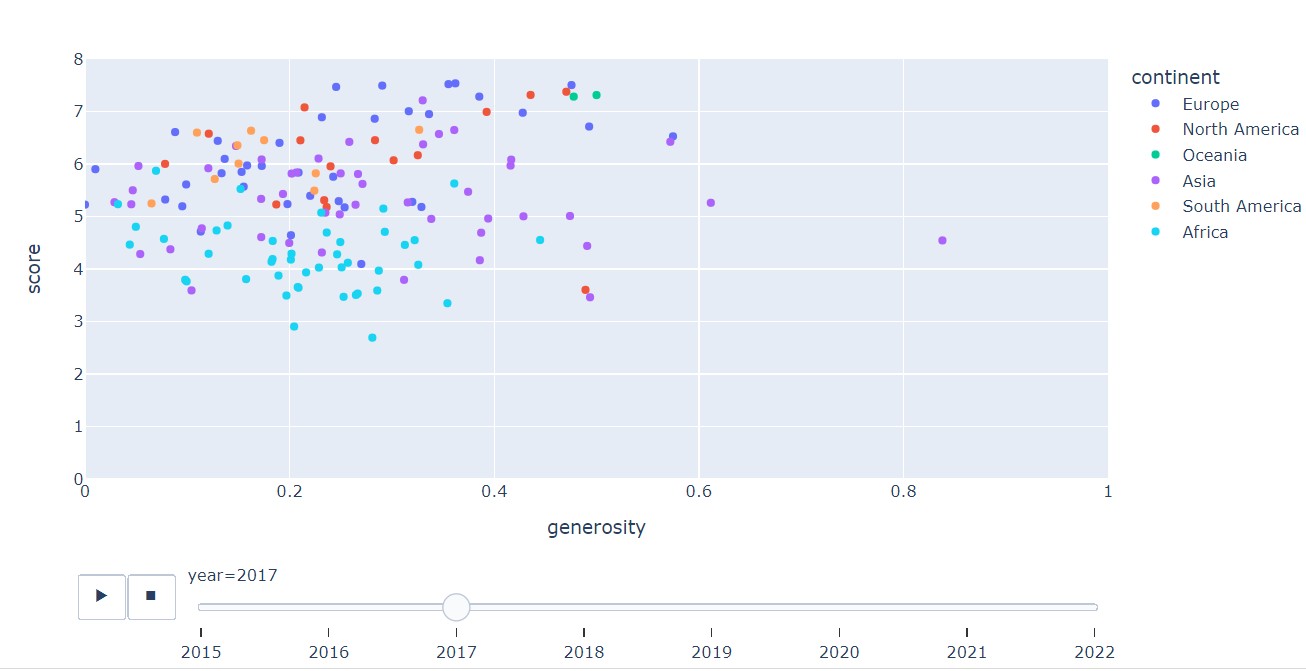
На данном графике видна несильная положительная корреляция, однако делать предположения относительно разных континентов уже не получится.

На графике совместно представлены два показателя – свобода и уровень счастья за 2018 год.

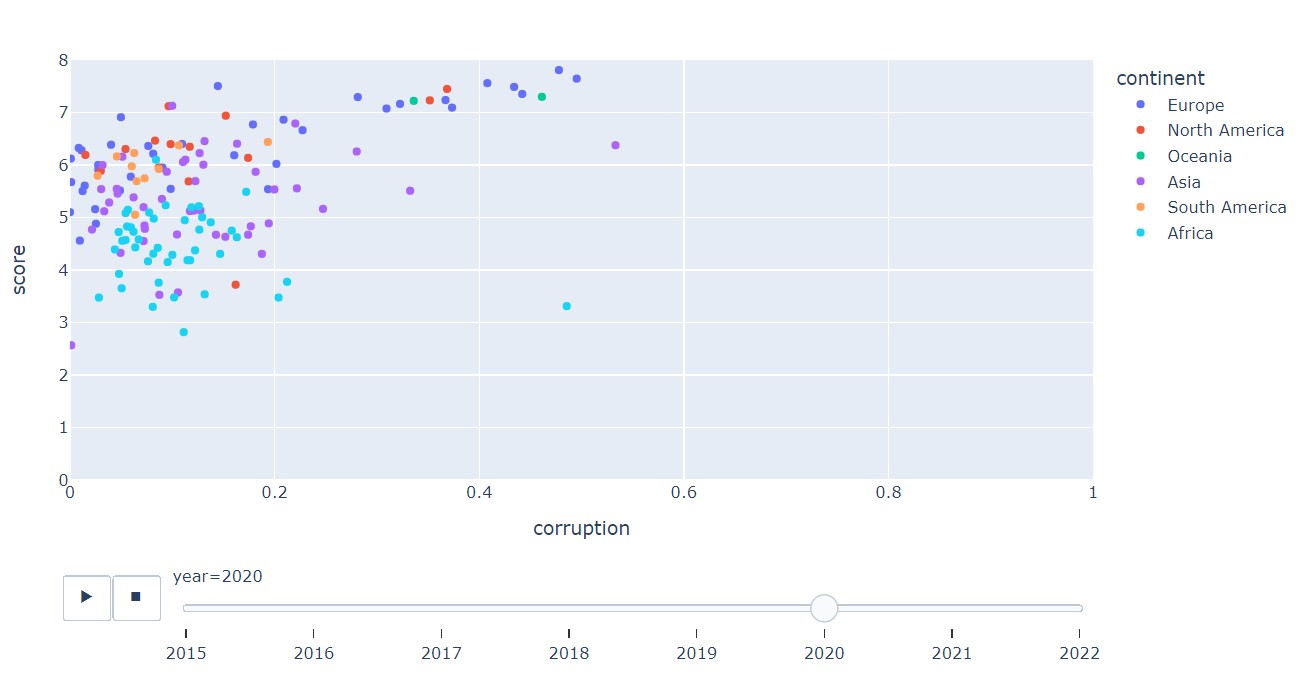


Как видно из графика корреляции практически не обнаруживается.

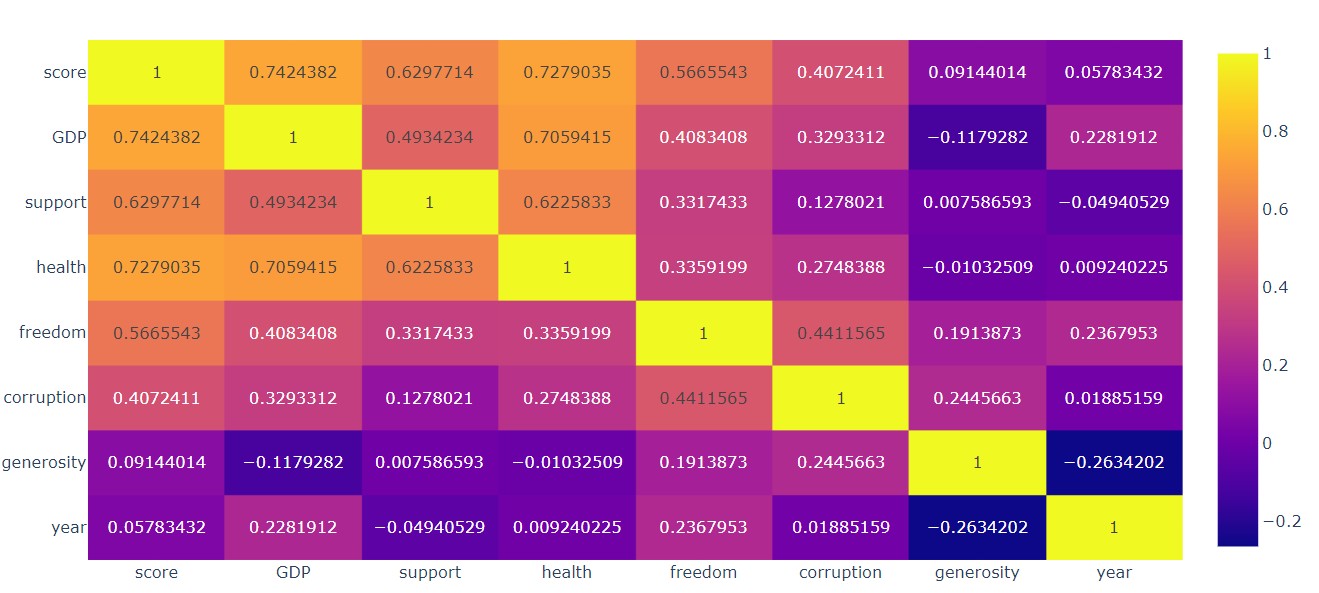
Для корреляции уровня счастья в зависимости от показателей щедрости аналогично корреляции практически нет – точки разбросаны хаотично. Для примера показан график за 2017 год.



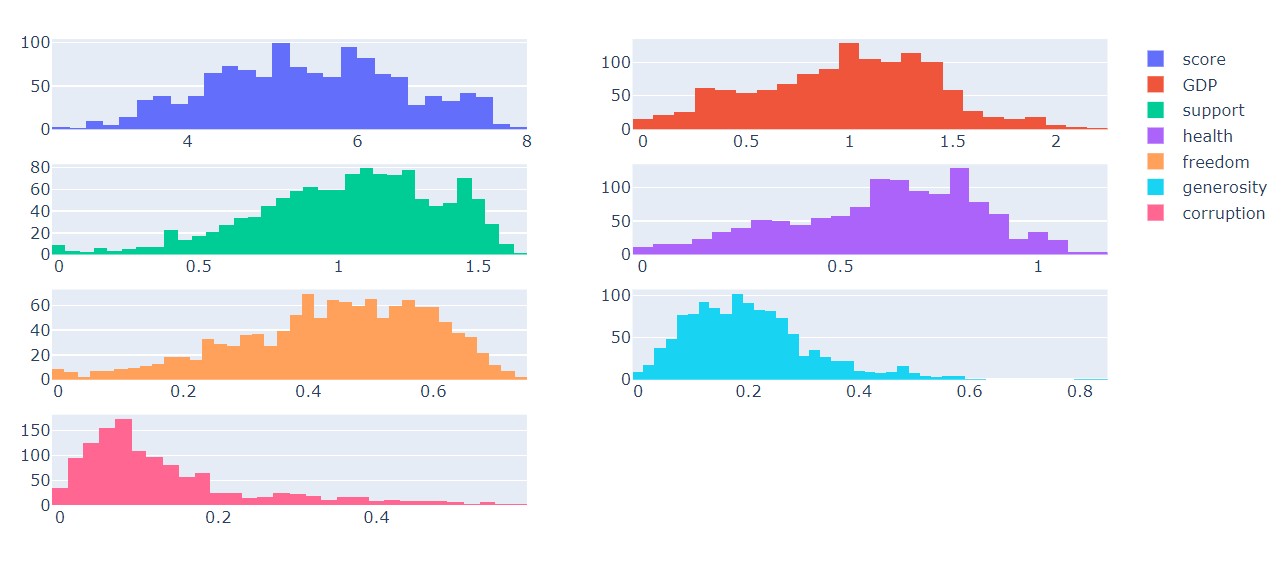
Корреляции уровня счастья от показателя уровня коррупции тоже не наблюдается. Для примера представлен график за 2020 год.



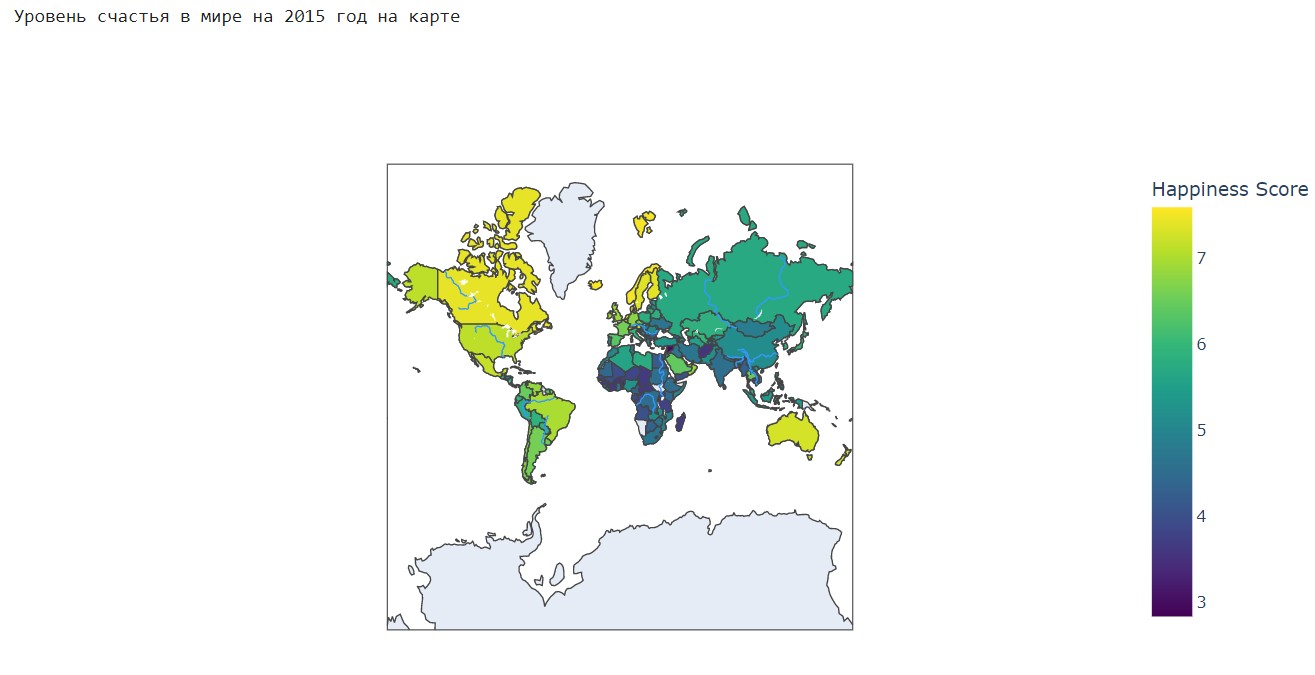
Ниже на построенной с помощью средств библиотеки plotly на корреляционной матрице показаны попарные корреляции всех числовых признаков.

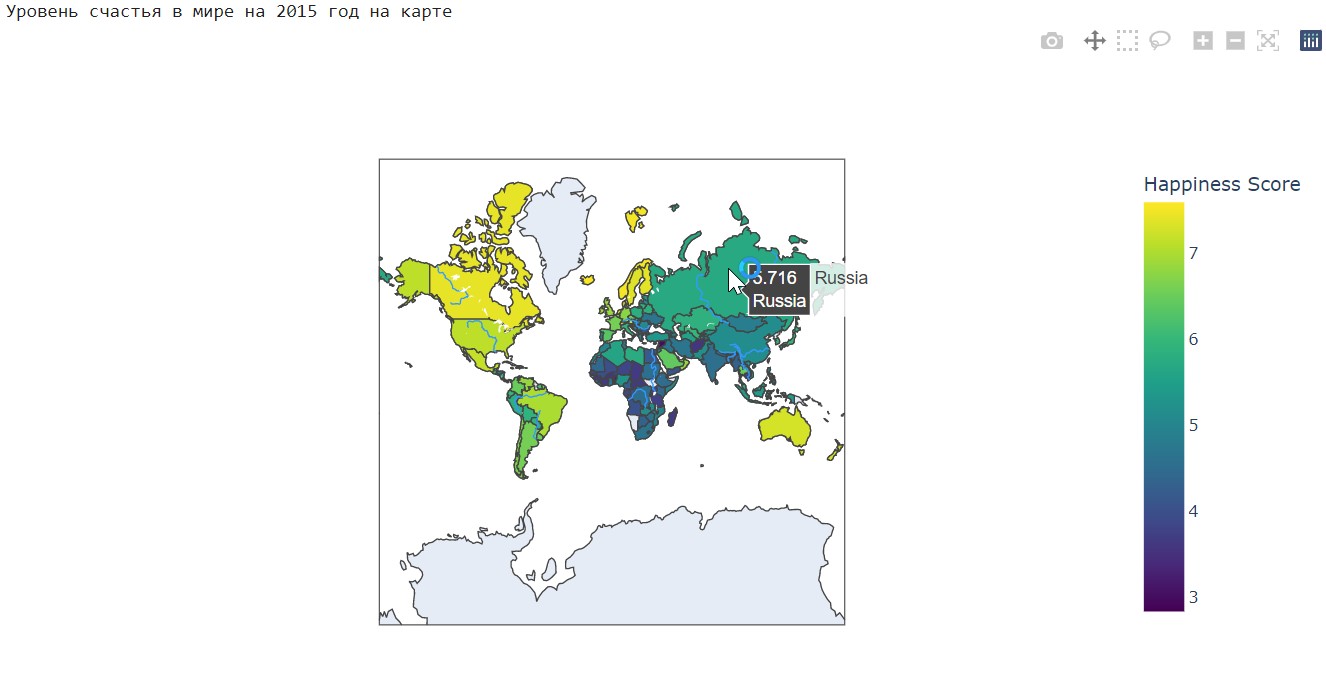
Таким образом, можно заметить, что между никакими двумя различными величинами нет настолько сильной корреляции, чтобы какой-либо признак пришлось удалять, как лишний для модели.

Следующий ряд гистограмм отображает распределение величин, поскольку для некоторых моделей машинного обучения важно, чтобы распределение данных было похоже на нормальное.

Как уровень счастья был распределён между странами, можно посмотреть на карте. Для этого была реализована функция построения карты и отображения на ней оценок.

Результатом вызова этой функции для словаря из датафреймов за каждый год, служит следующее изображение.

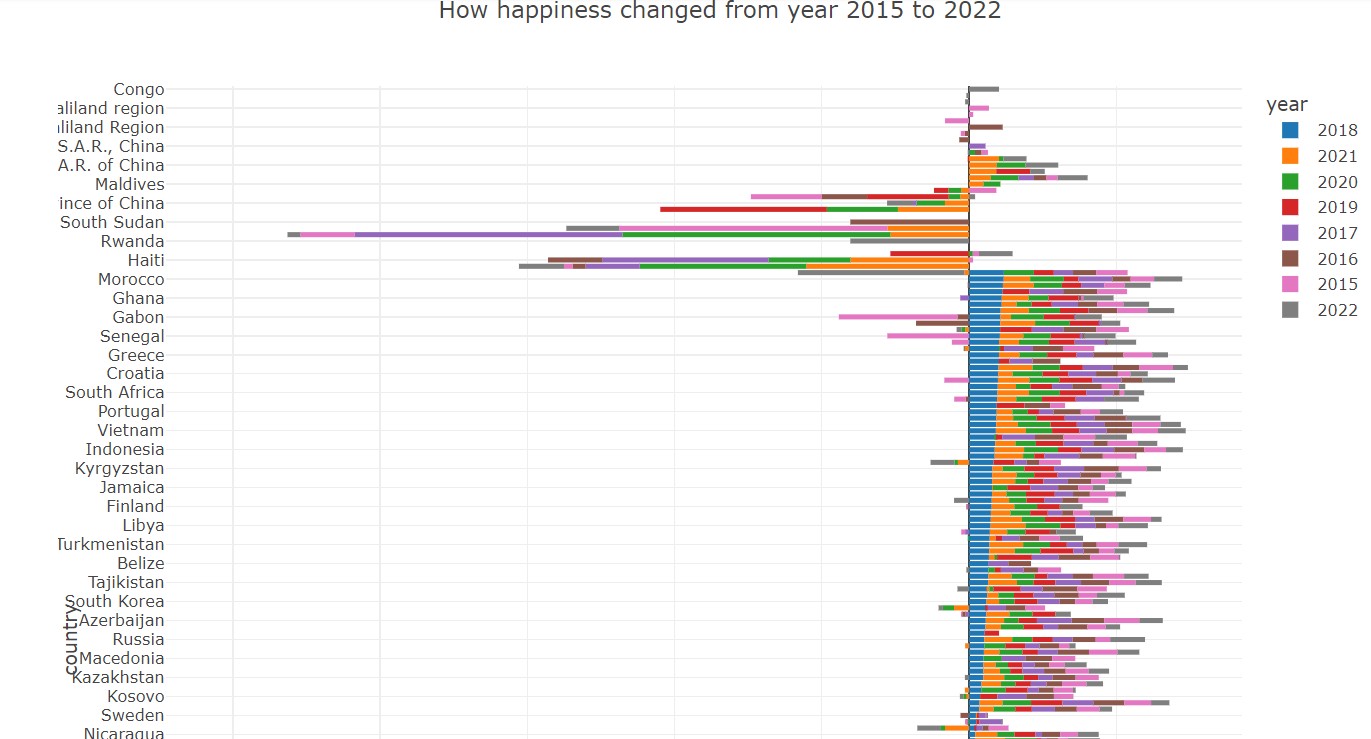
Аналогично графикам, построенным в библиотеке plotly на данной карте, наведя курсор на интересующую территорию, можно получить основные параметры для неё – название страны и уровень счастья.

С целью упрощения текста работы данные по остальным годам не приводятся, поскольку из года в год статистика менялась не так значительно, чтобы с заданной точностью это отразилось на карте.

Однако изменения в уровне счастья всё же встречаются. Покажем их на новой диаграмме. Реализация построения диаграммы выполняется через следующую функцию



И результатом служит следующая диаграмма, отражающая изменения для каждой страны её уровня счастья в разные года (в данном случае в тексте показано ограниченное число стран, однако в коде диаграмма отображает все)



# ПОДГОТОВКА К ОБУЧЕНИЮ МОДЕЛЕЙ

Первая проблема в данных – категориальные признаки (страна и континент). Многие модели умеют работать только с числовыми. Поэтому для представления категориальных признаков как числовых были выбраны два подхода – **LabelEncoder и** OneHotEncoder.

Далее кратко будет рассмотрено, в чём заключаются принципы.

**LabelEncoder каждому уникальному категориальному значению присваивает номер. Таким образом, все данные становятся числовыми. Однако у такого подхода есть проблема. Она заключается в том, что теперь над категориальными изначально признаками заданы математические отношения, хотя утверждать, что признаки одной категории «больше» признаков другой мы не можем. На самом деле иерархии в них нет.**

**Эту проблему решает подход** OneHotEncoding. В этой технике для каждого значения признака создаётся новый столбец. Числа заменяются на единицы и нули, в зависимости от того, какому столбцу какое значение подходит. Если объекту категория подходит, присваивается 1, если нет — 0. Стоит отметить, что при таком подходе имеет смысл удалять один из получившихся столбцов, поскольку его можно логически вывести из предыдущих и для модели это будет лишняя информация.

Данные в столбце country были закодированы при помощи техники **LabelEncoding, поскольку при использовании** OneHotEncoding таблица бы получилась слишком разряженной, так как количество уникальных стран велико. А для кодирования признака continent была применена техника OneHotEncoding с аргументом drop\_first=True. Таким образом, проблема с категориальными данными была решена.

Следующая проблема – масштаб данных. Поскольку все признаки, кроме года, - числа, не превосходящие 10, а год – числа превышающие 2000, было принято решение отмасштабировать признаки при помощи StandardScaler.

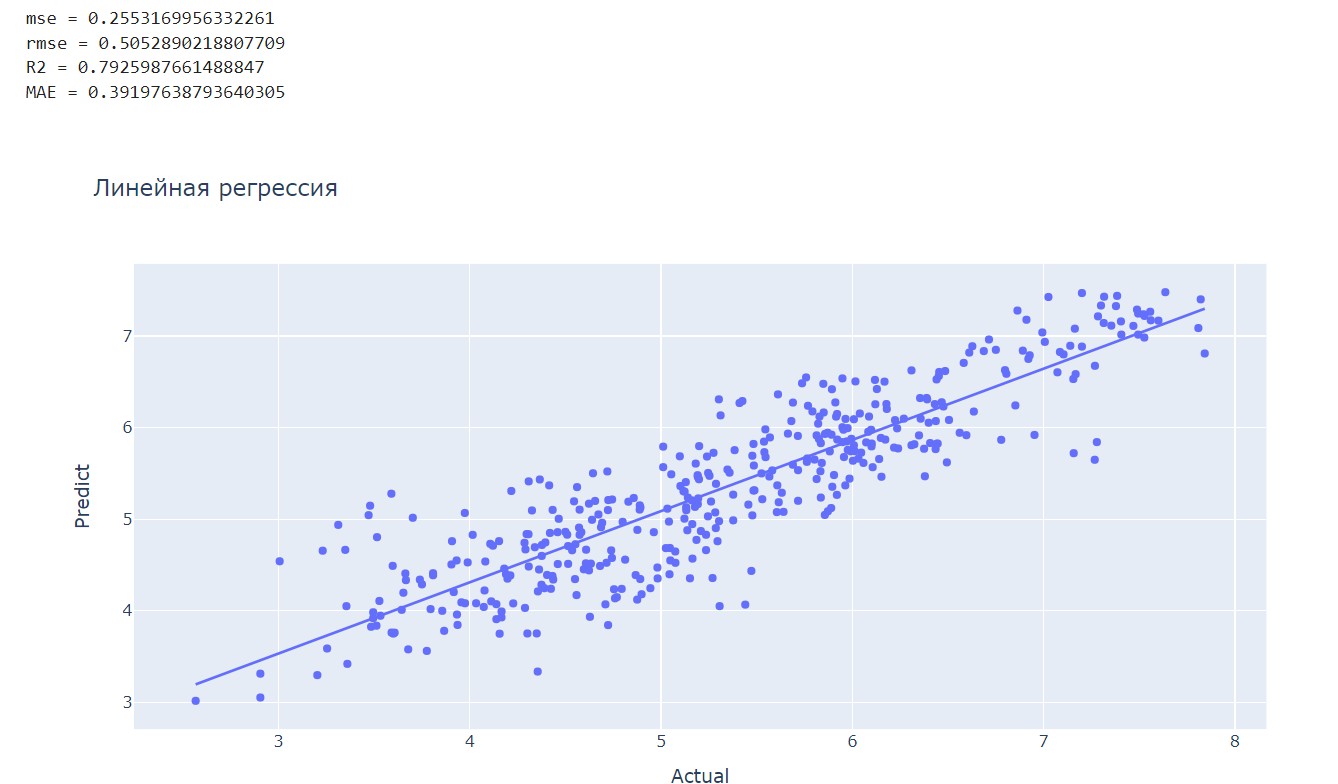
На обучающую и тестовую выборки данные были поделены в соотношении 2:1.

# ОБУЧЕНИЕ МОДЕЛЕЙ

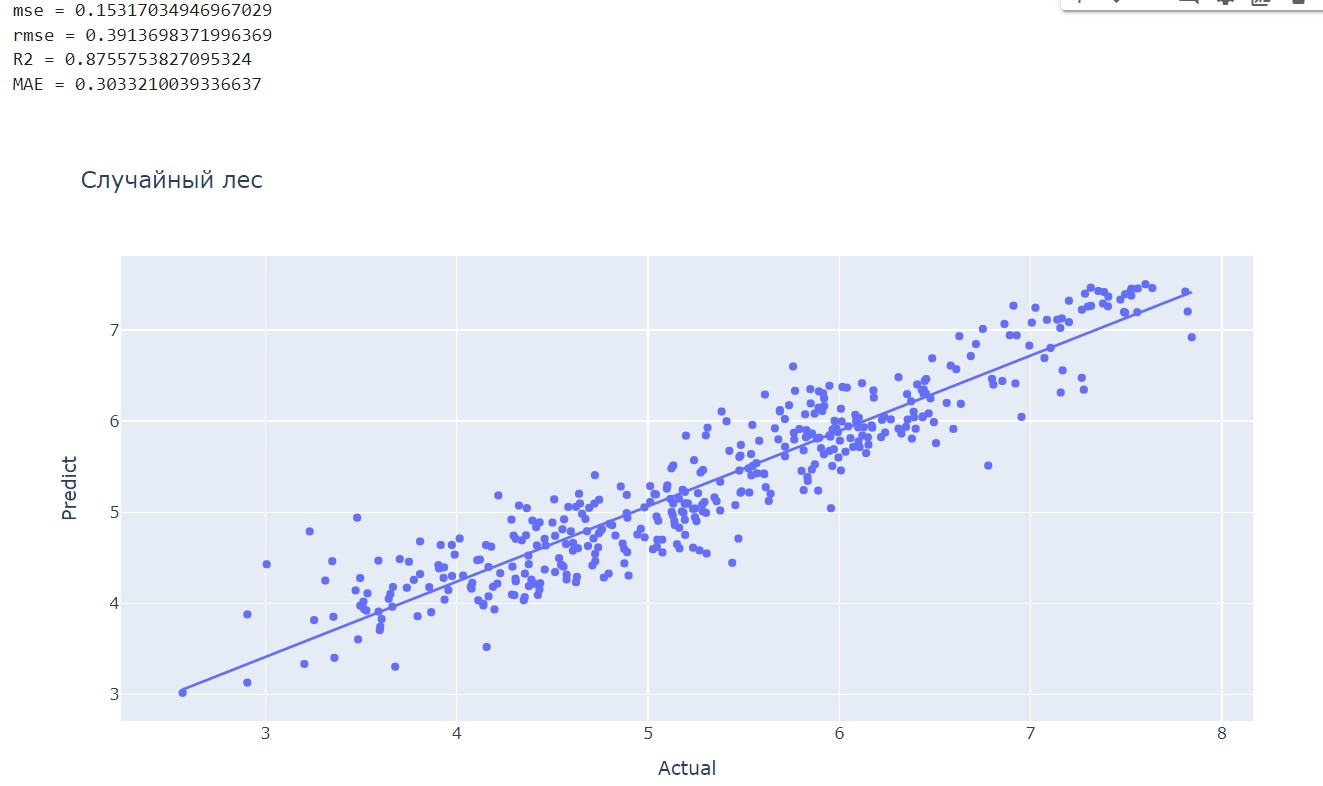
В работе рассмотрены 5 моделей. Далее будет показан результат, которого удалось достичь с их помощью.

Для каждой модели были построены графики, показывающие корреляцию между предсказаниями модели и истинными значениями.

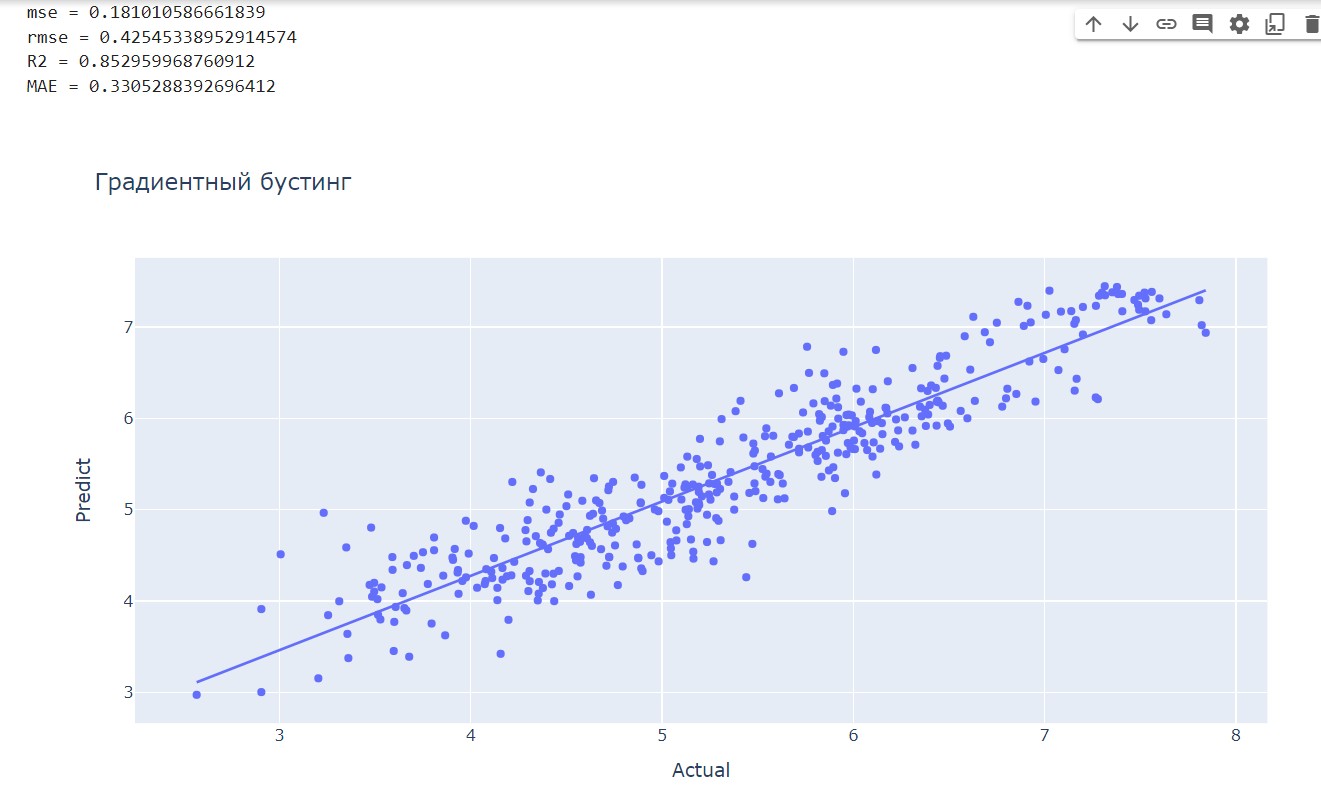
Далее представлен результат линейной регрессии.



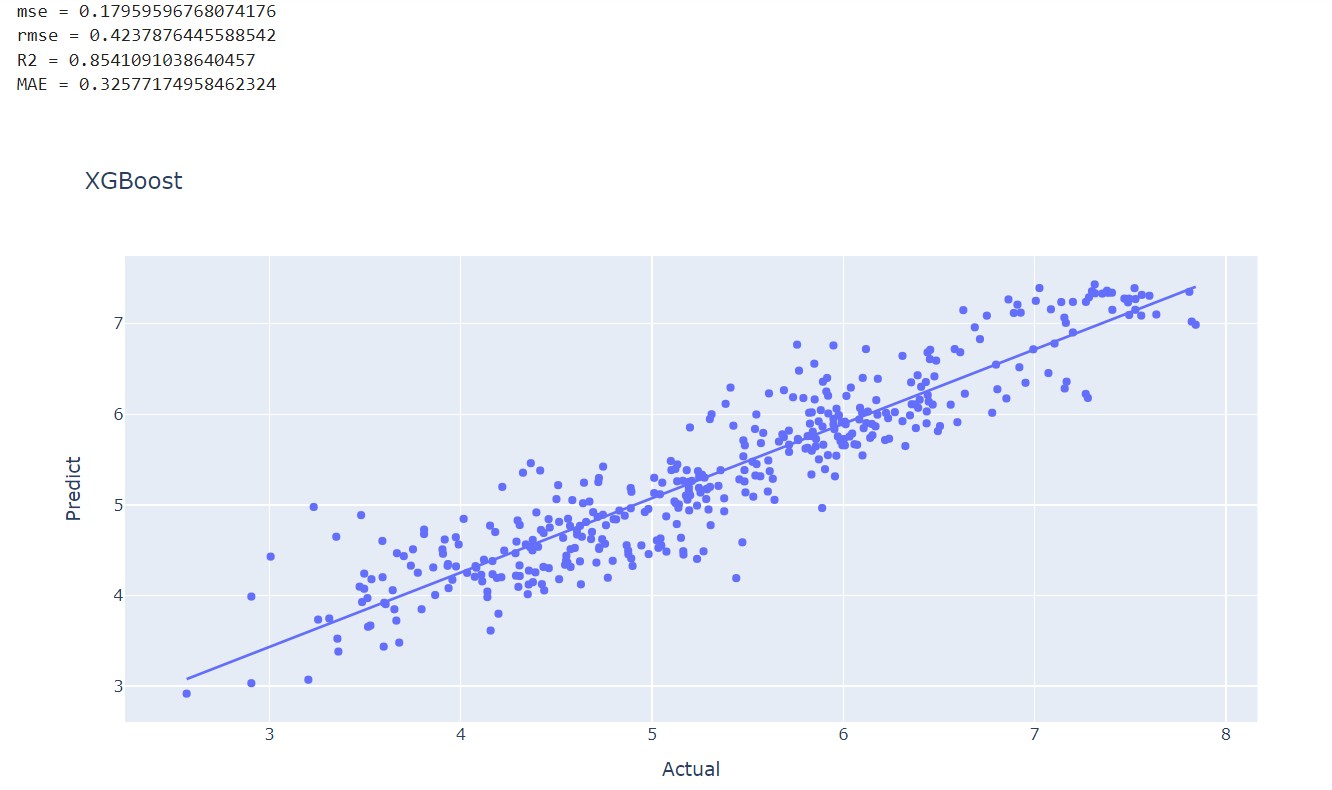
На следующем графике представлен результат работы случайного леса.



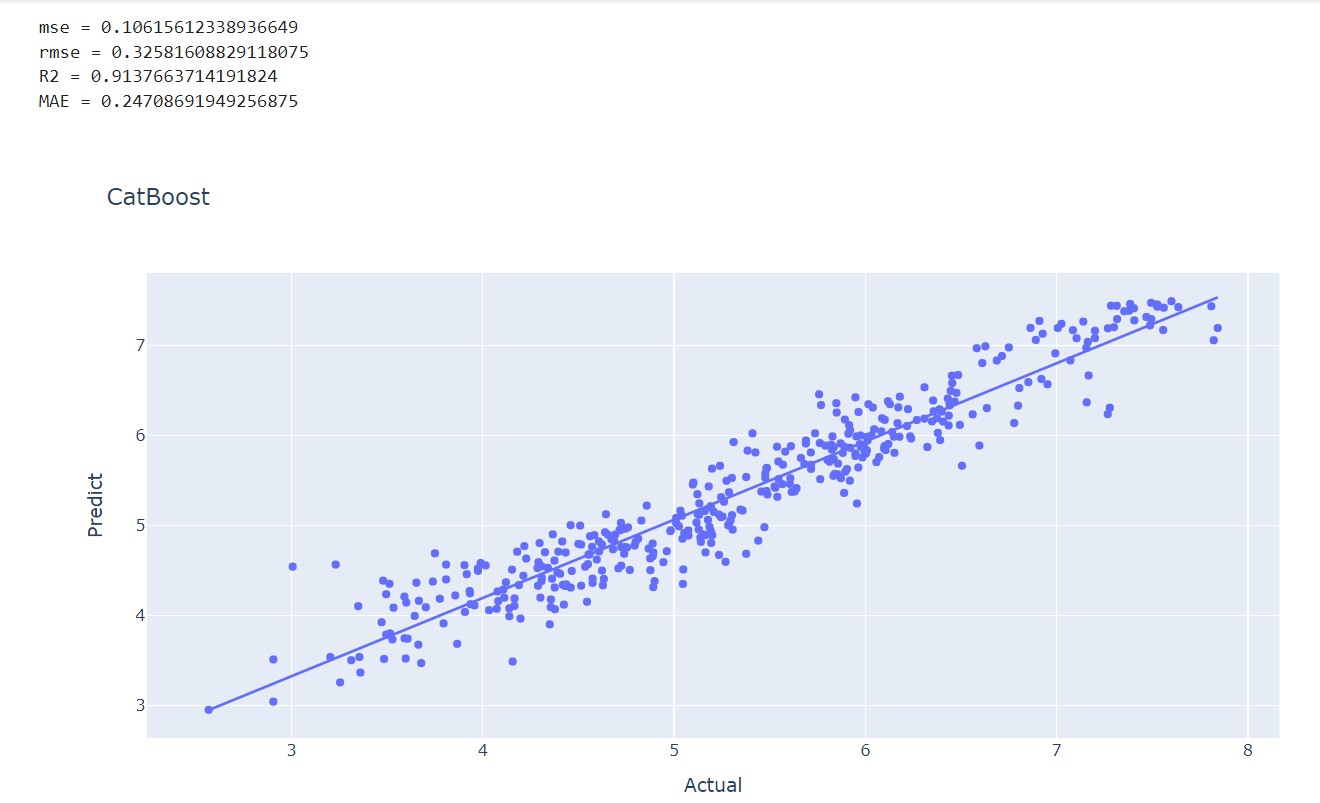
Следующий рисунок показывает результат работы градиентного бустинга.



Далее можно посмотреть, какие результаты показал XGBoost.



И рисунок ниже показывает результаты для CatBoost.



Как видно из проведённой работы лучший результат показывает Catboost. Его и возьмём в качестве прототипа для модели предсказания уровня счастья.

# ЗАКЛЮЧЕНИЕ

В ходе работы были проанализированы 8 датафреймов с информацией об уровне счастья в странах и прочими показателями с 2015 по 2022 год. Данные были предобработаны, проанализированы, подготовлены к визуализации, отображены на понятных графиках, подготовлены к обучению моделей. На данных было обучено 5 моделей – линейная регрессия, случайный лес, обычный градиентный бустинг, XGBoost и CatBoost.

В результате в качестве прототипа модели машинного обучения была выбрана модель Catboost, которая показала наилучший результат R2 = 0.91. В качестве дальнейшей работы над проектом можно перебрать гипер-параметры модели, возможно, это приведёт к повышению результата.

# СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

**Литература**

1.Динер, Эд; Эммонс, Роберт А .; Ларсен, Рэнди Дж .; Гриффин, Шарон (1985). «Шкала удовлетворенности жизнью». Журнал оценки личности

2.Елшанский С.П., Ануфриев А.Ф., Камалетдинова З.Ф., Сапарин О.Е., Семёнов Д.В. Некоторые психометрические показатели русскоязычного варианта Шкалы субъективного счастья С. Любомирски и Х. Леппер // Гуманитарные научные исследования. 2014. № 7

3.Helliwell, John; Layard, Richard; Sachs, Jeffrey - ["World Happiness Report"](http://www.earth.columbia.edu/sitefiles/file/Sachs%20Writing/2012/World%20Happiness%20Report.pdf) (PDF). Columbia University Earth Institute. Retrieved 2014-06-29.

4.Tianqi Chen, Carlos Guestrin - XGBoost: A Scalable Tree Boosting System

**Ресурсы интернет**

5.Учебник по машинному обучению от ШАД – Текст электронный - URL: <https://ml-handbook.ru/> (дата обращения: 13.05.2022)

6.CatBoost is a high-performance open source library for gradient boosting on decision trees - Текст электронный - URL: <https://catboost.ai/> (дата обращения: 13.05.2022)

# ПРИЛОЖЕНИЕ 1

**Список стран на 2015 год:**

'Switzerland', 'Iceland', 'Denmark', 'Norway', 'Canada', 'Finland', 'Netherlands', 'Sweden', 'New Zealand', 'Australia', 'Israel', 'Costa Rica', 'Austria', 'Mexico', 'United States', 'Brazil', 'Luxembourg', 'Ireland', 'Belgium', 'United Arab Emirates', 'United Kingdom', 'Oman', 'Venezuela', 'Singapore', 'Panama', 'Germany', 'Chile', 'Qatar', 'France', 'Argentina', 'Czech Republic', 'Uruguay', 'Colombia', 'Thailand', 'Saudi Arabia', 'Spain', 'Malta', 'Taiwan', 'Kuwait', 'Suriname', 'Trinidad and Tobago', 'El Salvador', 'Guatemala', 'Uzbekistan', 'Slovakia', 'Japan', 'South Korea', 'Ecuador', 'Bahrain', 'Italy', 'Bolivia', 'Moldova', 'Paraguay', 'Kazakhstan', 'Slovenia', 'Lithuania', 'Nicaragua', 'Peru', 'Belarus', 'Poland', 'Malaysia', 'Croatia', 'Libya', 'Russia', 'Jamaica', 'North Cyprus', 'Cyprus', 'Algeria', 'Kosovo', 'Turkmenistan', 'Mauritius', 'Hong Kong', 'Estonia', 'Indonesia', 'Vietnam', 'Turkey', 'Kyrgyzstan', 'Nigeria', 'Bhutan', 'Azerbaijan', 'Pakistan', 'Jordan', 'Montenegro', 'China', 'Zambia', 'Romania', 'Serbia', 'Portugal', 'Latvia', 'Philippines', 'Somaliland region', 'Morocco', 'Macedonia', 'Mozambique', 'Albania', 'Bosnia and Herzegovina', 'Lesotho', 'Dominican Republic', 'Laos', 'Mongolia', 'Swaziland', 'Greece', 'Lebanon', 'Hungary', 'Honduras', 'Tajikistan', 'Tunisia', 'Palestinian Territories', 'Bangladesh', 'Iran', 'Ukraine', 'Iraq', 'South Africa', 'Ghana', 'Zimbabwe', 'Liberia', 'India', 'Sudan', 'Haiti', 'Congo (Kinshasa)', 'Nepal', 'Ethiopia', 'Sierra Leone', 'Mauritania', 'Kenya', 'Djibouti', 'Armenia', 'Botswana', 'Myanmar', 'Georgia', 'Malawi', 'Sri Lanka', 'Cameroon', 'Bulgaria', 'Egypt', 'Yemen', 'Angola', 'Mali', 'Congo (Brazzaville)', 'Comoros', 'Uganda', 'Senegal', 'Gabon', 'Niger', 'Cambodia', 'Tanzania', 'Madagascar', 'Central African Republic', 'Chad', 'Guinea', 'Ivory Coast', 'Burkina Faso', 'Afghanistan', 'Rwanda', 'Benin', 'Syria', 'Burundi', 'Togo’

**Список стран на 2016 год:**

'Denmark', 'Switzerland', 'Iceland', 'Norway', 'Finland', 'Canada', 'Netherlands', 'New Zealand', 'Australia', 'Sweden', 'Israel', 'Austria', 'United States', 'Costa Rica', 'Puerto Rico', 'Germany', 'Brazil', 'Belgium', 'Ireland', 'Luxembourg', 'Mexico', 'Singapore', 'United Kingdom', 'Chile', 'Panama', 'Argentina', 'Czech Republic', 'United Arab Emirates', 'Uruguay', 'Malta', 'Colombia', 'France', 'Thailand', 'Saudi Arabia', 'Taiwan', 'Qatar', 'Spain', 'Algeria', 'Guatemala', 'Suriname', 'Kuwait', 'Bahrain', 'Trinidad and Tobago', 'Venezuela', 'Slovakia', 'El Salvador', 'Malaysia', 'Nicaragua', 'Uzbekistan', 'Italy', 'Ecuador', 'Belize', 'Japan', 'Kazakhstan', 'Moldova', 'Russia', 'Poland', 'South Korea', 'Bolivia', 'Lithuania', 'Belarus', 'North Cyprus', 'Slovenia', 'Peru', 'Turkmenistan', 'Mauritius', 'Libya', 'Latvia', 'Cyprus', 'Paraguay', 'Romania', 'Estonia', 'Jamaica', 'Croatia', 'Hong Kong', 'Somalia', 'Kosovo', 'Turkey', 'Indonesia', 'Jordan', 'Azerbaijan', 'Philippines', 'China', 'Bhutan', 'Kyrgyzstan', 'Serbia', 'Bosnia and Herzegovina', 'Montenegro', 'Dominican Republic', 'Morocco', 'Hungary', 'Pakistan', 'Lebanon', 'Portugal', 'Macedonia', 'Vietnam', 'Somaliland Region', 'Tunisia', 'Greece', 'Tajikistan', 'Mongolia', 'Laos', 'Nigeria', 'Honduras', 'Iran', 'Zambia', 'Nepal', 'Palestinian Territories', 'Albania', 'Bangladesh', 'Sierra Leone', 'Iraq', 'Namibia', 'Cameroon', 'Ethiopia', 'South Africa', 'Sri Lanka', 'India', 'Myanmar', 'Egypt', 'Armenia', 'Kenya', 'Ukraine', 'Ghana', 'Congo (Kinshasa)', 'Georgia', 'Congo (Brazzaville)', 'Senegal', 'Bulgaria', 'Mauritania', 'Zimbabwe', 'Malawi', 'Sudan', 'Gabon', 'Mali', 'Haiti', 'Botswana', 'Comoros', 'Ivory Coast', 'Cambodia', 'Angola', 'Niger', 'South Sudan', 'Chad', 'Burkina Faso', 'Uganda', 'Yemen', 'Madagascar', 'Tanzania', 'Liberia', 'Guinea', 'Rwanda', 'Benin', 'Afghanistan', 'Togo', 'Syria', 'Burundi'

**Список стран на 2017 год:**

'Norway', 'Denmark', 'Iceland', 'Switzerland', 'Finland', 'Netherlands', 'Canada', 'New Zealand', 'Sweden', 'Australia', 'Israel', 'Costa Rica', 'Austria', 'United States', 'Ireland', 'Germany', 'Belgium', 'Luxembourg', 'United Kingdom', 'Chile', 'United Arab Emirates', 'Brazil', 'Czech Republic', 'Argentina', 'Mexico', 'Singapore', 'Malta', 'Uruguay', 'Guatemala', 'Panama', 'France', 'Thailand', 'Taiwan Province of China', 'Spain', 'Qatar', 'Colombia', 'Saudi Arabia', 'Trinidad and Tobago', 'Kuwait', 'Slovakia', 'Bahrain', 'Malaysia', 'Nicaragua', 'Ecuador', 'El Salvador', 'Poland', 'Uzbekistan', 'Italy', 'Russia', 'Belize', 'Japan', 'Lithuania', 'Algeria', 'Latvia', 'South Korea', 'Moldova', 'Romania', 'Bolivia', 'Turkmenistan', 'Kazakhstan', 'North Cyprus', 'Slovenia', 'Peru', 'Mauritius', 'Cyprus', 'Estonia', 'Belarus', 'Libya', 'Turkey', 'Paraguay', 'Hong Kong S.A.R., China', 'Philippines', 'Serbia', 'Jordan', 'Hungary', 'Jamaica', 'Croatia', 'Kosovo', 'China', 'Pakistan', 'Indonesia', 'Venezuela', 'Montenegro', 'Morocco', 'Azerbaijan', 'Dominican Republic', 'Greece', 'Lebanon', 'Portugal', 'Bosnia and Herzegovina', 'Honduras', 'Macedonia', 'Somalia', 'Vietnam', 'Nigeria', 'Tajikistan', 'Bhutan', 'Kyrgyzstan', 'Nepal', 'Mongolia', 'South Africa', 'Tunisia', 'Palestinian Territories', 'Egypt', 'Bulgaria', 'Sierra Leone', 'Cameroon', 'Iran', 'Albania', 'Bangladesh', 'Namibia', 'Kenya', 'Mozambique', 'Myanmar', 'Senegal', 'Zambia', 'Iraq', 'Gabon', 'Ethiopia', 'Sri Lanka', 'Armenia', 'India', 'Mauritania', 'Congo (Brazzaville)', 'Georgia', 'Congo (Kinshasa)', 'Mali', 'Ivory Coast', 'Cambodia', 'Sudan', 'Ghana', 'Ukraine', 'Uganda', 'Burkina Faso', 'Niger', 'Malawi', 'Chad', 'Zimbabwe', 'Lesotho', 'Angola', 'Afghanistan', 'Botswana', 'Benin', 'Madagascar', 'Haiti', 'Yemen', 'South Sudan', 'Liberia', 'Guinea', 'Togo', 'Rwanda', 'Syria', 'Tanzania', 'Burundi', 'Central African Republic'

**Список стран на 2018 год:**

'Finland', 'Norway', 'Denmark', 'Iceland', 'Switzerland', 'Netherlands', 'Canada', 'New Zealand', 'Sweden', 'Australia', 'United Kingdom', 'Austria', 'Costa Rica', 'Ireland', 'Germany', 'Belgium', 'Luxembourg', 'United States', 'Israel', 'United Arab Emirates', 'Czech Republic', 'Malta', 'France', 'Mexico', 'Chile', 'Taiwan', 'Panama', 'Brazil', 'Argentina', 'Guatemala', 'Uruguay', 'Qatar', 'Saudi Arabia', 'Singapore', 'Malaysia', 'Spain', 'Colombia', 'Trinidad & Tobago', 'Slovakia', 'El Salvador', 'Nicaragua', 'Poland', 'Bahrain', 'Uzbekistan', 'Kuwait', 'Thailand', 'Italy', 'Ecuador', 'Belize', 'Lithuania', 'Slovenia', 'Romania', 'Latvia', 'Japan', 'Mauritius', 'Jamaica', 'South Korea', 'Northern Cyprus', 'Russia', 'Kazakhstan', 'Cyprus', 'Bolivia', 'Estonia', 'Paraguay', 'Peru', 'Kosovo', 'Moldova', 'Turkmenistan', 'Hungary', 'Libya', 'Philippines', 'Honduras', 'Belarus', 'Turkey', 'Pakistan', 'Hong Kong', 'Portugal', 'Serbia', 'Greece', 'Lebanon', 'Montenegro', 'Croatia', 'Dominican Republic', 'Algeria', 'Morocco', 'China', 'Azerbaijan', 'Tajikistan', 'Macedonia', 'Jordan', 'Nigeria', 'Kyrgyzstan', 'Bosnia and Herzegovina', 'Mongolia', 'Vietnam', 'Indonesia', 'Bhutan', 'Somalia', 'Cameroon', 'Bulgaria', 'Nepal', 'Venezuela', 'Gabon', 'Palestinian Territories', 'South Africa', 'Iran', 'Ivory Coast', 'Ghana', 'Senegal', 'Laos', 'Tunisia', 'Albania', 'Sierra Leone', 'Congo (Brazzaville)', 'Bangladesh', 'Sri Lanka', 'Iraq', 'Mali', 'Namibia', 'Cambodia', 'Burkina Faso', 'Egypt', 'Mozambique', 'Kenya', 'Zambia', 'Mauritania', 'Ethiopia', 'Georgia', 'Armenia', 'Myanmar', 'Chad', 'Congo (Kinshasa)', 'India', 'Niger', 'Uganda', 'Benin', 'Sudan', 'Ukraine', 'Togo', 'Guinea', 'Lesotho', 'Angola', 'Madagascar', 'Zimbabwe', 'Afghanistan', 'Botswana', 'Malawi', 'Haiti', 'Liberia', 'Syria', 'Rwanda', 'Yemen', 'Tanzania', 'South Sudan', 'Central African Republic', 'Burundi

**Список стран на 2019 год:**

'Finland', 'Denmark', 'Norway', 'Iceland', 'Netherlands', 'Switzerland', 'Sweden', 'New Zealand', 'Canada', 'Austria', 'Australia', 'Costa Rica', 'Israel', 'Luxembourg', 'United Kingdom', 'Ireland', 'Germany', 'Belgium', 'United States', 'Czech Republic', 'United Arab Emirates', 'Malta', 'Mexico', 'France', 'Taiwan', 'Chile', 'Guatemala', 'Saudi Arabia', 'Qatar', 'Spain', 'Panama', 'Brazil', 'Uruguay', 'Singapore', 'El Salvador', 'Italy', 'Bahrain', 'Slovakia', 'Trinidad & Tobago', 'Poland', 'Uzbekistan', 'Lithuania', 'Colombia', 'Slovenia', 'Nicaragua', 'Kosovo', 'Argentina', 'Romania', 'Cyprus', 'Ecuador', 'Kuwait', 'Thailand', 'Latvia', 'South Korea', 'Estonia', 'Jamaica', 'Mauritius', 'Japan', 'Honduras', 'Kazakhstan', 'Bolivia', 'Hungary', 'Paraguay', 'Northern Cyprus', 'Peru', 'Portugal', 'Pakistan', 'Russia', 'Philippines', 'Serbia', 'Moldova', 'Libya', 'Montenegro', 'Tajikistan', 'Croatia', 'Hong Kong', 'Dominican Republic', 'Bosnia and Herzegovina', 'Turkey', 'Malaysia', 'Belarus', 'Greece', 'Mongolia', 'North Macedonia', 'Nigeria', 'Kyrgyzstan', 'Turkmenistan', 'Algeria', 'Morocco', 'Azerbaijan', 'Lebanon', 'Indonesia', 'China', 'Vietnam', 'Bhutan', 'Cameroon', 'Bulgaria', 'Ghana', 'Ivory Coast', 'Nepal', 'Jordan', 'Benin', 'Congo (Brazzaville)', 'Gabon', 'Laos', 'South Africa', 'Albania', 'Venezuela', 'Cambodia', 'Palestinian Territories', 'Senegal', 'Somalia', 'Namibia', 'Niger', 'Burkina Faso', 'Armenia', 'Iran', 'Guinea', 'Georgia', 'Gambia', 'Kenya', 'Mauritania', 'Mozambique', 'Tunisia', 'Bangladesh', 'Iraq', 'Congo (Kinshasa)', 'Mali', 'Sierra Leone', 'Sri Lanka', 'Myanmar', 'Chad', 'Ukraine', 'Ethiopia', 'Swaziland', 'Uganda', 'Egypt', 'Zambia', 'Togo', 'India', 'Liberia', 'Comoros', 'Madagascar', 'Lesotho', 'Burundi', 'Zimbabwe', 'Haiti', 'Botswana', 'Syria', 'Malawi', 'Yemen', 'Rwanda', 'Tanzania', 'Afghanistan', 'Central African Republic', 'South Sudan'

**Список стран на 2020 год:**

'Finland', 'Denmark', 'Switzerland', 'Iceland', 'Norway', 'Netherlands', 'Sweden', 'New Zealand', 'Austria', 'Luxembourg', 'Canada', 'Australia', 'United Kingdom', 'Israel', 'Costa Rica', 'Ireland', 'Germany', 'United States', 'Czech Republic', 'Belgium', 'United Arab Emirates', 'Malta', 'France', 'Mexico', 'Taiwan Province of China', 'Uruguay', 'Saudi Arabia', 'Spain', 'Guatemala', 'Italy', 'Singapore', 'Brazil', 'Slovenia', 'El Salvador', 'Kosovo', 'Panama', 'Slovakia', 'Uzbekistan', 'Chile', 'Bahrain', 'Lithuania', 'Trinidad and Tobago', 'Poland', 'Colombia', 'Cyprus', 'Nicaragua', 'Romania', 'Kuwait', 'Mauritius', 'Kazakhstan', 'Estonia', 'Philippines', 'Hungary', 'Thailand', 'Argentina', 'Honduras', 'Latvia', 'Ecuador', 'Portugal', 'Jamaica', 'South Korea', 'Japan', 'Peru', 'Serbia', 'Bolivia', 'Pakistan', 'Paraguay', 'Dominican Republic', 'Bosnia and Herzegovina', 'Moldova', 'Tajikistan', 'Montenegro', 'Russia', 'Kyrgyzstan', 'Belarus', 'North Cyprus', 'Greece', 'Hong Kong S.A.R. of China', 'Croatia', 'Libya', 'Mongolia', 'Malaysia', 'Vietnam', 'Indonesia', 'Ivory Coast', 'Benin', 'Maldives', 'Congo (Brazzaville)', 'Azerbaijan', 'Macedonia', 'Ghana', 'Nepal', 'Turkey', 'China', 'Turkmenistan', 'Bulgaria', 'Morocco', 'Cameroon', 'Venezuela', 'Algeria', 'Senegal', 'Guinea', 'Niger', 'Laos', 'Albania', 'Cambodia', 'Bangladesh', 'Gabon', 'South Africa', 'Iraq', 'Lebanon', 'Burkina Faso', 'Gambia', 'Mali', 'Nigeria', 'Armenia', 'Georgia', 'Iran', 'Jordan', 'Mozambique', 'Kenya', 'Namibia', 'Ukraine', 'Liberia', 'Palestinian Territories', 'Uganda', 'Chad', 'Tunisia', 'Mauritania', 'Sri Lanka', 'Congo (Kinshasa)', 'Swaziland', 'Myanmar', 'Comoros', 'Togo', 'Ethiopia', 'Madagascar', 'Egypt', 'Sierra Leone', 'Burundi', 'Zambia', 'Haiti', 'Lesotho', 'India', 'Malawi', 'Yemen', 'Botswana', 'Tanzania', 'Central African Republic', 'Rwanda', 'Zimbabwe', 'South Sudan', 'Afghanistan'

**Список стран на 2021 год:**

'Finland', 'Denmark', 'Switzerland', 'Iceland', 'Netherlands', 'Norway', 'Sweden', 'Luxembourg', 'New Zealand', 'Austria', 'Australia', 'Israel', 'Germany', 'Canada', 'Ireland', 'Costa Rica', 'United Kingdom', 'Czech Republic', 'United States', 'Belgium', 'France', 'Bahrain', 'Malta', 'Taiwan Province of China', 'United Arab Emirates', 'Saudi Arabia', 'Spain', 'Italy', 'Slovenia', 'Guatemala', 'Uruguay', 'Singapore', 'Kosovo', 'Slovakia', 'Brazil', 'Mexico', 'Jamaica', 'Lithuania', 'Cyprus', 'Estonia', 'Panama', 'Uzbekistan', 'Chile', 'Poland', 'Kazakhstan', 'Romania', 'Kuwait', 'Serbia', 'El Salvador', 'Mauritius', 'Latvia', 'Colombia', 'Hungary', 'Thailand', 'Nicaragua', 'Japan', 'Argentina', 'Portugal', 'Honduras', 'Croatia', 'Philippines', 'South Korea', 'Peru', 'Bosnia and Herzegovina', 'Moldova', 'Ecuador', 'Kyrgyzstan', 'Greece', 'Bolivia', 'Mongolia', 'Paraguay', 'Montenegro', 'Dominican Republic', 'North Cyprus', 'Belarus', 'Russia', 'Hong Kong S.A.R. of China', 'Tajikistan', 'Vietnam', 'Libya', 'Malaysia', 'Indonesia', 'Congo (Brazzaville)', 'China', 'Ivory Coast', 'Armenia', 'Nepal', 'Bulgaria', 'Maldives', 'Azerbaijan', 'Cameroon', 'Senegal', 'Albania', 'North Macedonia', 'Ghana', 'Niger', 'Turkmenistan', 'Gambia', 'Benin', 'Laos', 'Bangladesh', 'Guinea', 'South Africa', 'Turkey', 'Pakistan', 'Morocco', 'Venezuela', 'Georgia', 'Algeria', 'Ukraine', 'Iraq', 'Gabon', 'Burkina Faso', 'Cambodia', 'Mozambique', 'Nigeria', 'Mali', 'Iran', 'Uganda', 'Liberia', 'Kenya', 'Tunisia', 'Lebanon', 'Namibia', 'Palestinian Territories', 'Myanmar', 'Jordan', 'Chad', 'Sri Lanka', 'Swaziland', 'Comoros', 'Egypt', 'Ethiopia', 'Mauritania', 'Madagascar', 'Togo', 'Zambia', 'Sierra Leone', 'India', 'Burundi', 'Yemen', 'Tanzania', 'Haiti', 'Malawi', 'Lesotho', 'Botswana', 'Rwanda', 'Zimbabwe', 'Afghanistan'

**Список стран на 2022 год:**

'Finland', 'Denmark', 'Iceland', 'Switzerland', 'Netherlands', 'Luxembourg\*', 'Sweden', 'Norway', 'Israel', 'New Zealand', 'Austria', 'Australia', 'Ireland', 'Germany', 'Canada', 'United States', 'United Kingdom', 'Czechia', 'Belgium', 'France', 'Bahrain', 'Slovenia', 'Costa Rica', 'United Arab Emirates', 'Saudi Arabia', 'Taiwan Province of China', 'Singapore', 'Romania', 'Spain', 'Uruguay', 'Italy', 'Kosovo', 'Malta', 'Lithuania', 'Slovakia', 'Estonia', 'Panama', 'Brazil', 'Guatemala\*', 'Kazakhstan', 'Cyprus', 'Latvia', 'Serbia', 'Chile', 'Nicaragua', 'Mexico', 'Croatia', 'Poland', 'El Salvador', 'Kuwait\*', 'Hungary', 'Mauritius', 'Uzbekistan', 'Japan', 'Honduras', 'Portugal', 'Argentina', 'Greece', 'South Korea', 'Philippines', 'Thailand', 'Moldova', 'Jamaica', 'Kyrgyzstan', 'Belarus\*', 'Colombia', 'Bosnia and Herzegovina', 'Mongolia', 'Dominican Republic', 'Malaysia', 'Bolivia', 'China', 'Paraguay', 'Peru', 'Montenegro', 'Ecuador', 'Vietnam', 'Turkmenistan\*', 'North Cyprus\*', 'Russia', 'Hong Kong S.A.R. of China', 'Armenia', 'Tajikistan', 'Nepal', 'Bulgaria', 'Libya\*', 'Indonesia', 'Ivory Coast', 'North Macedonia', 'Albania', 'South Africa', 'Azerbaijan\*', 'Gambia\*', 'Bangladesh', 'Laos', 'Algeria', 'Liberia\*', 'Ukraine', 'Congo', 'Morocco', 'Mozambique', 'Cameroon', 'Senegal', 'Niger\*', 'Georgia', 'Gabon', 'Iraq', 'Venezuela', 'Guinea', 'Iran', 'Ghana', 'Turkey', 'Burkina Faso', 'Cambodia', 'Benin', 'Comoros\*', 'Uganda', 'Nigeria', 'Kenya', 'Tunisia', 'Pakistan', 'Palestinian Territories\*', 'Mali', 'Namibia', 'Eswatini, Kingdom of\*', 'Myanmar', 'Sri Lanka', 'Madagascar\*', 'Egypt', 'Chad\*', 'Ethiopia', 'Yemen\*', 'Mauritania\*', 'Jordan', 'Togo', 'India', 'Zambia', 'Malawi', 'Tanzania', 'Sierra Leone', 'Lesotho\*', 'Botswana\*', 'Rwanda\*', 'Zimbabwe', 'Lebanon', 'Afghanistan', 'xx'