



Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления» (ИУ)

КАФЕДРА «Системы обработки информации и управления» (ИУ5)

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

**«Построение модели для классификации
пользователей онлайн-курсов на основе анализа
данных»**

Студент группы ИУ5-32М

_____ А. С. Ищенко

Руководитель

_____ Ю. Е. Гапанюк

2025 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ
Заведующий кафедрой ИУ5

_____ В.И. Терехов

« ____ » _____ 2025 г.

ЗАДАНИЕ
на выполнение научно-исследовательской работы

по теме: Использование метаграфового подхода для анализа архитектурных паттернов

Студент группы ИУ5-22М _____ Ищенко Анастасия Сергеевна
(Фамилия имя отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)
_____ исследовательская

Источник тематики (кафедра, предприятие, НИР) _____ учебная тематика

График выполнения НИР: 25% к 5 нед., 50% к 9 нед., 75% к 13 нед., 100% к 16 нед.

Техническое задание: Произвести анализ датасета о прохождении онлайн-курсов и построить прогнозную модель

Оформление научно-исследовательской работы:
Расчетно-пояснительная записка, 25 листов формата А4.

Приложения: нет

Дата выдачи задания «15» февраля 2025 г.

Научный руководитель

_____ Ю.Е. Гапанюк
(подпись, дата) (инициалы и фамилия)

Студент группы ИУ5-32М
(код группы)

_____ А.С. Ищенко
(подпись, дата) (инициалы и фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Содержание

Введение	4
1 Исследование датасета	5
2 Визуализация данных	8
3 Подготовка данных для машинного обучения	15
4 Библиотека XGBoost	18
5 Список гиперпараметров XGBoost	19
6 Визуализация деревьев XGBoost.....	21
7 Вывод	23
Список использованных источников.....	24

Введение

В настоящее время большой популярностью пользуются различные онлайн-курсы. Однако, распространена ситуация, когда пользователь бросает прохождение курса и не получает сертификат. В данной работе будет использован датасет, основанный на реальных данных одной из платформ онлайн курсов. Будет произведена очистка данных и их визуализация, а также анализ. С использованием библиотеки XGBoost будет произведена попытка создания модели, которая будет предсказывать, завершит ли пользователь более 50% курса или нет.

1 Исследование датасета

Импорт библиотек

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

Будет использована библиотека XGBoost для прогнозирования студентов, которые прошли более 50% курса

Импорт данных из csv файла

```
df_course= pd.read_csv('/Users/ /Downloads/3.csv')
df_course.head()
```

	Launch Date	Course title	Teachers	Course subject	Participants	50% course content accessed (audited)	Certified	% Audited	% Certified	% Certified of > 50% course content accessed	% Played video	% Posted in forum	% Grade higher than 0	Total course hours	Median hours for certification	Median age	% Male	% Female	% Bachelor's degree or higher
0	11/17/2018	Java Developer. Professional	Стрекалов Павел	OOP, Backend	36105	5431	3003	15.04	8.32	54.98	83.20	8.17	28.97	418.94	64.45	26.0	88.28	11.72	60.68
1	12/25/2019	Разработчик Android (deprecated)	Стрекалов Павел	Mobile	62709	8949	5783	14.27	9.22	64.05	89.14	14.38	39.50	884.04	78.53	28.0	83.50	16.50	63.04
2	6/25/2019	Python Developer. Professional	Чибриков Виталий	OOP, Backend	16663	2855	2082	17.13	12.49	72.85	87.49	14.42	34.89	227.55	61.28	27.0	70.32	29.68	58.76
3	3/27/2020	Разработчик Ruby	Чибриков Виталий	OOP, Backend	129400	12888	1439	9.96	1.11	11.11	0.00	0.00	1.11	220.90	0.00	28.0	80.02	19.98	58.78
4	10/18/2019	C++ Developer. Professional	Петрелевич Сергей	OOP, Backend	52521	10729	5058	20.44	9.64	47.12	77.45	15.98	32.52	804.41	76.10	32.0	56.78	43.22	88.33

Просмотр типов данных. Данные представлены типами object, int, float

```
df_course.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 19 columns):
#   Column                                                                 Non-Null Count  Dtype
---  -
0   Launch Date                                                            205 non-null   object
1   Course title                                                            205 non-null   object
2   Teachers                                                                205 non-null   object
3   Course subject                                                          205 non-null   object
4   Participants                                                            205 non-null   int64
5   50% course content accessed (audited)                                205 non-null   int64
6   Certified                                                                205 non-null   int64
7   % Audited                                                                205 non-null   float64
8   % Certified                                                            205 non-null   float64
9   % Certified of > 50% course content accessed                        205 non-null   float64
10  % Played video                                                            205 non-null   float64
11  % Posted in forum                                                        205 non-null   float64
12  % Grade higher than 0                                                    205 non-null   float64
13  Total course hours                                                       205 non-null   float64
14  Median hours for certification                                           205 non-null   float64
15  Median age                                                                205 non-null   float64
16  % Male                                                                    205 non-null   float64
17  % Female                                                                  205 non-null   float64
18  % Bachelor's degree or higher                                           205 non-null   float64
dtypes: float64(12), int64(3), object(4)
memory usage: 30.6+ KB
```

Список категорий курсов и количество курсов в каждой категории

```
df_course['Course subject'].value_counts()
```

```

OOP, Backend      50
Management        43
Test              20
Administration    19
Security          16
ML                15
Mobile            11
DB                11
DevOps            7
Frontend          6
Math              5
Design            2
```

```
Name: Course subject, dtype: int64
```

Список преподавателей и количество курсов, на которых он преподает

```
df_course['Teachers'].value_counts()
```

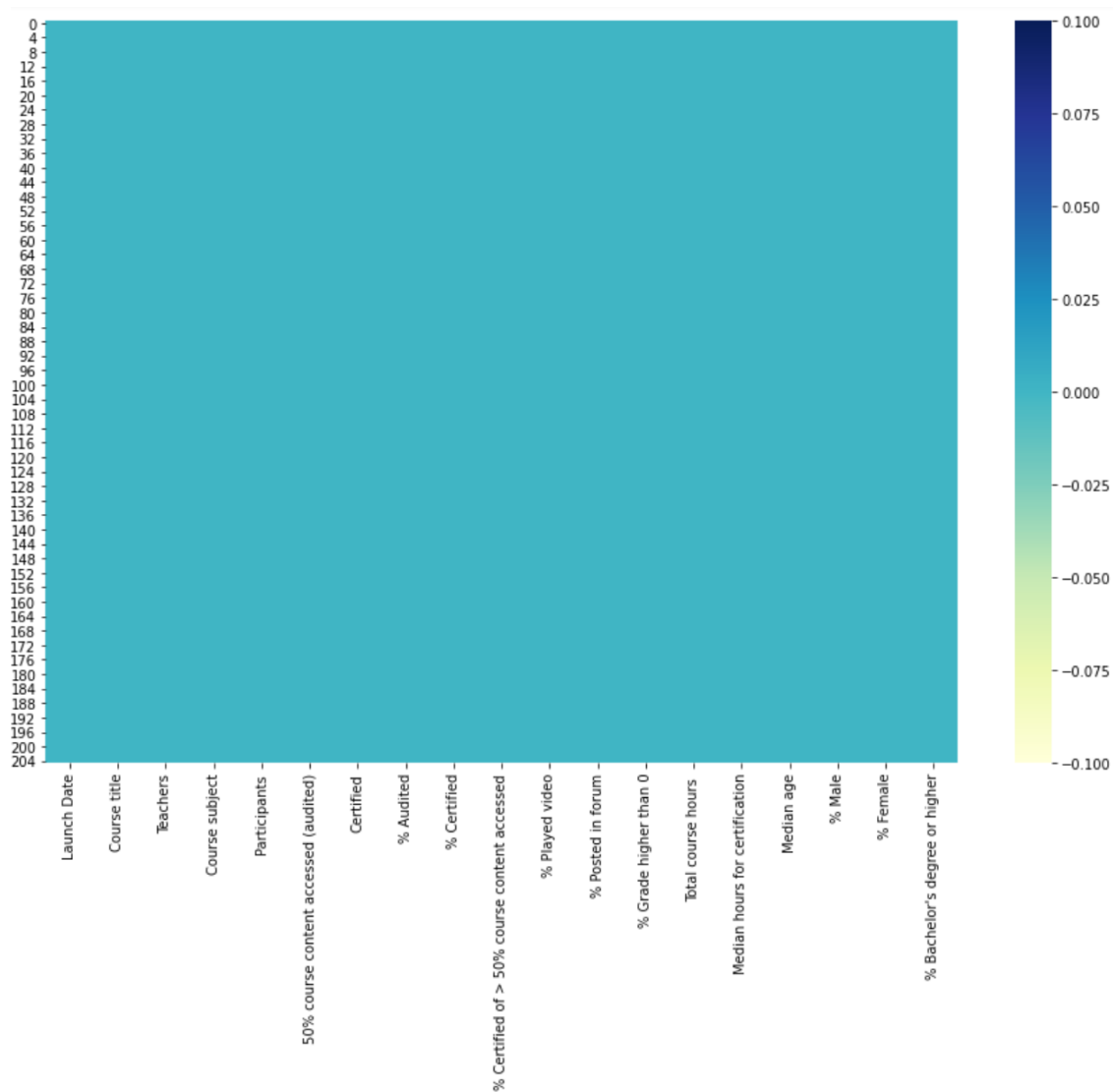
```

Волосатов Евгений      4
Темирханова Эльвира    3
Петрелевич Сергей      3
```

Дроздецкий Владимир	3
Цыкунов Алексей	3
..	
Гуторов Владимир	1
Левчук Мартин	1
Курочкин Игорь	1
Пулявин Артем	1
Швец Олег	1

Просмотр значений NULL в данных, установленных с помощью тепловой карты:

```
plt.figure(figsize=(15,10))
sns.heatmap(df_course.isnull(),cmap="YlGnBu")
```



Мы убедились, что пустых ячеек нет.

2 Визуализация данных

Представим названия курсов в виде облака слов, где более часто используемые слова имеют больший размер, чем другие.

```
from wordcloud import WordCloud, STOPWORDS
```

```
wordcloud = WordCloud(
    stopwords=STOPWORDS,
    background_color='white',
    width=1200,
    height=1000
).generate(" ".join(df_course['Course
title'])))
```

```
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```



Отообразим категории курсов в виде облака слов.

```
wordcloud = WordCloud(  
    stopwords=STOPWORDS,  
    background_color='white',  
    width=1200,  
    height=1000
```



```

).generate(" ".join(df_course['Course
subject']))

plt.imshow(wordcloud)
plt.axis('off')
plt.show()

```



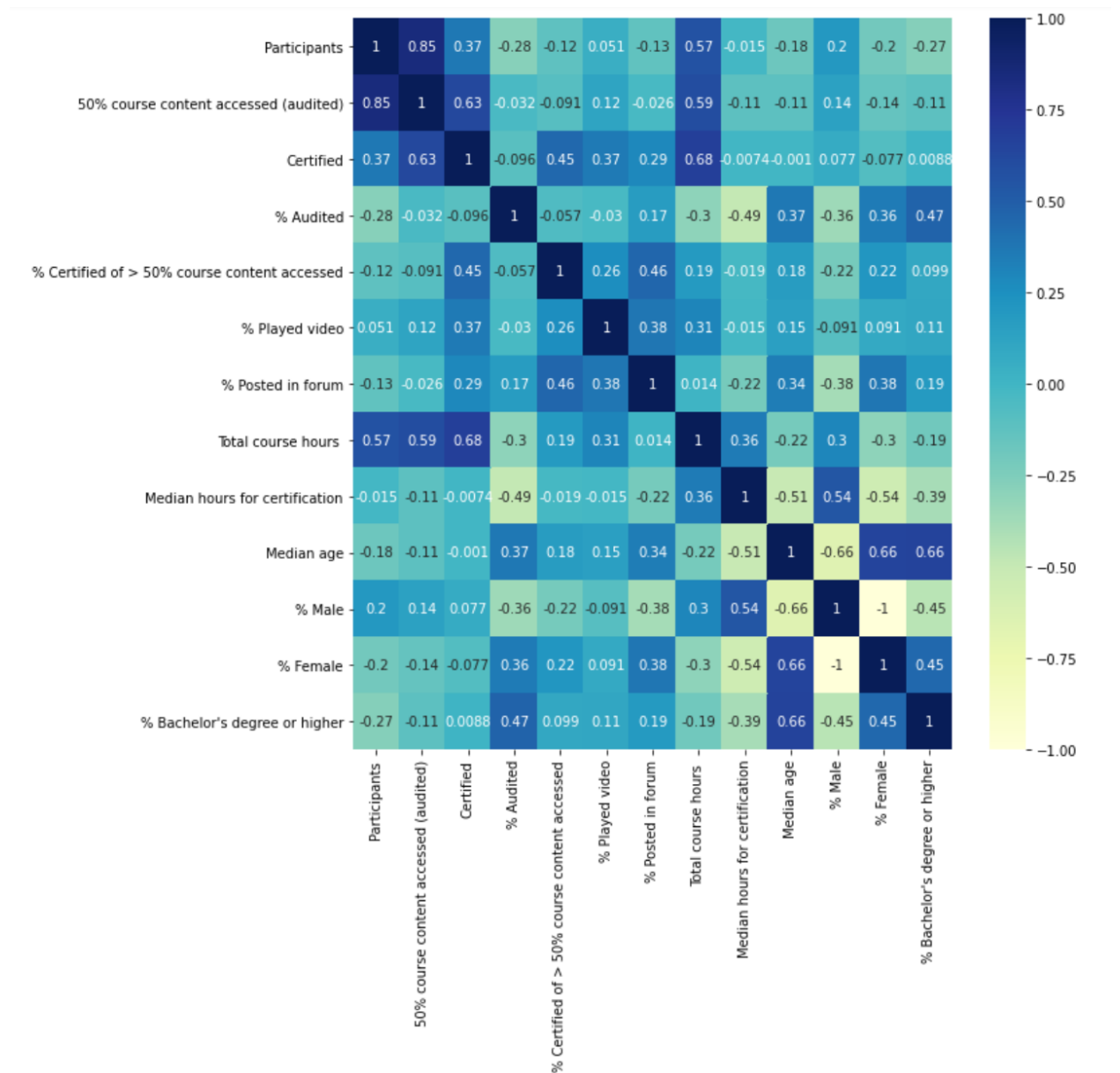
Судя по облакам слов, в названии курса чаще всего встречается слово «разработчик», а самой популярной категорией является бэкенд разработка и объектно-ориентированное программирование.

Построим матрицу корреляций между различными признаками.

```

df_course=df_course.drop(['% Certified','Course title','% Grade
higher than 0'],axis=1)
df_course
figure= plt.figure(figsize=(10,10))
sns.heatmap(df_course.corr(), annot=True,cmap="YlGnBu")

```



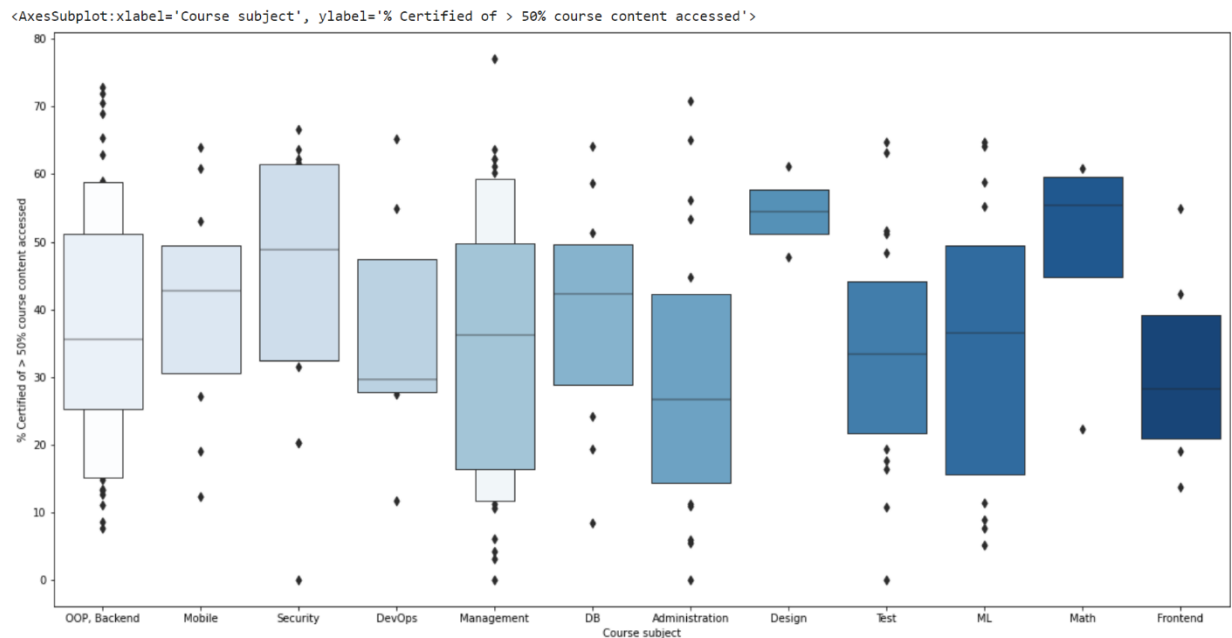
Как видно из матрицы, сильной корреляции между признаками нет, будем продолжать исследование дальше.

Построим диаграммы размаха («ящик с усами») для признаков Course subject, Certified

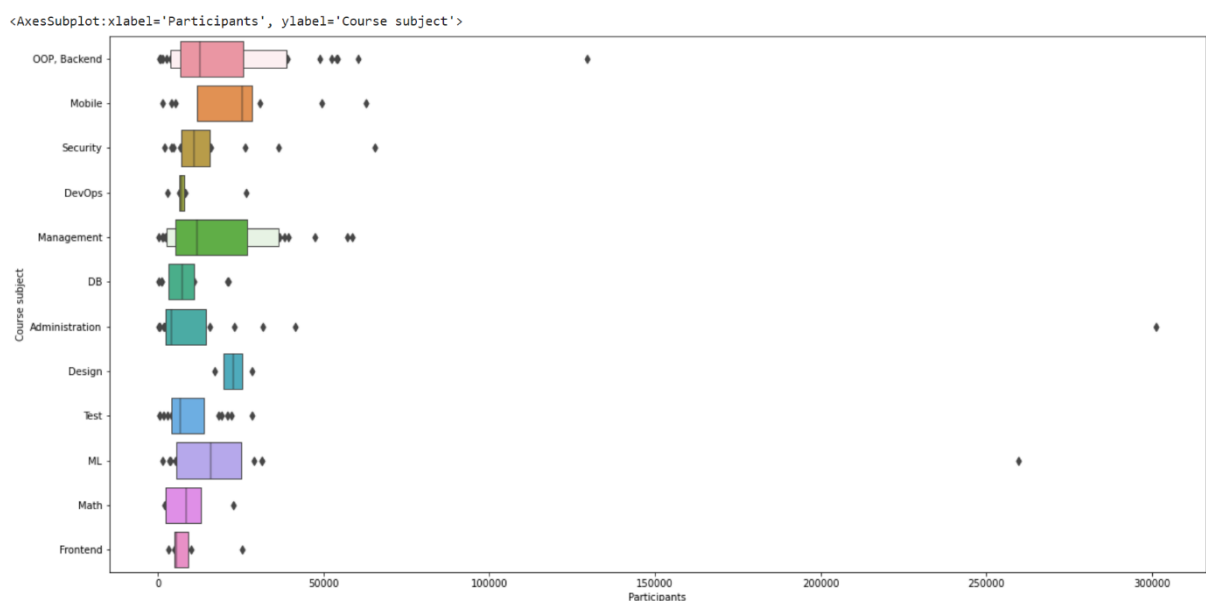
of > 50% course content accessed и Participants, Course subject

```
figure= plt.figure(figsize=(20,10))
```

```
sns.boxenplot(x='Course subject',y='% Certified of > 50% course content accessed,data=df_course,palette="Blues")
```



```
figure= plt.figure(figsize=(20,10))
sns.boxenplot('Participants','Course subject',data=df_course)
```

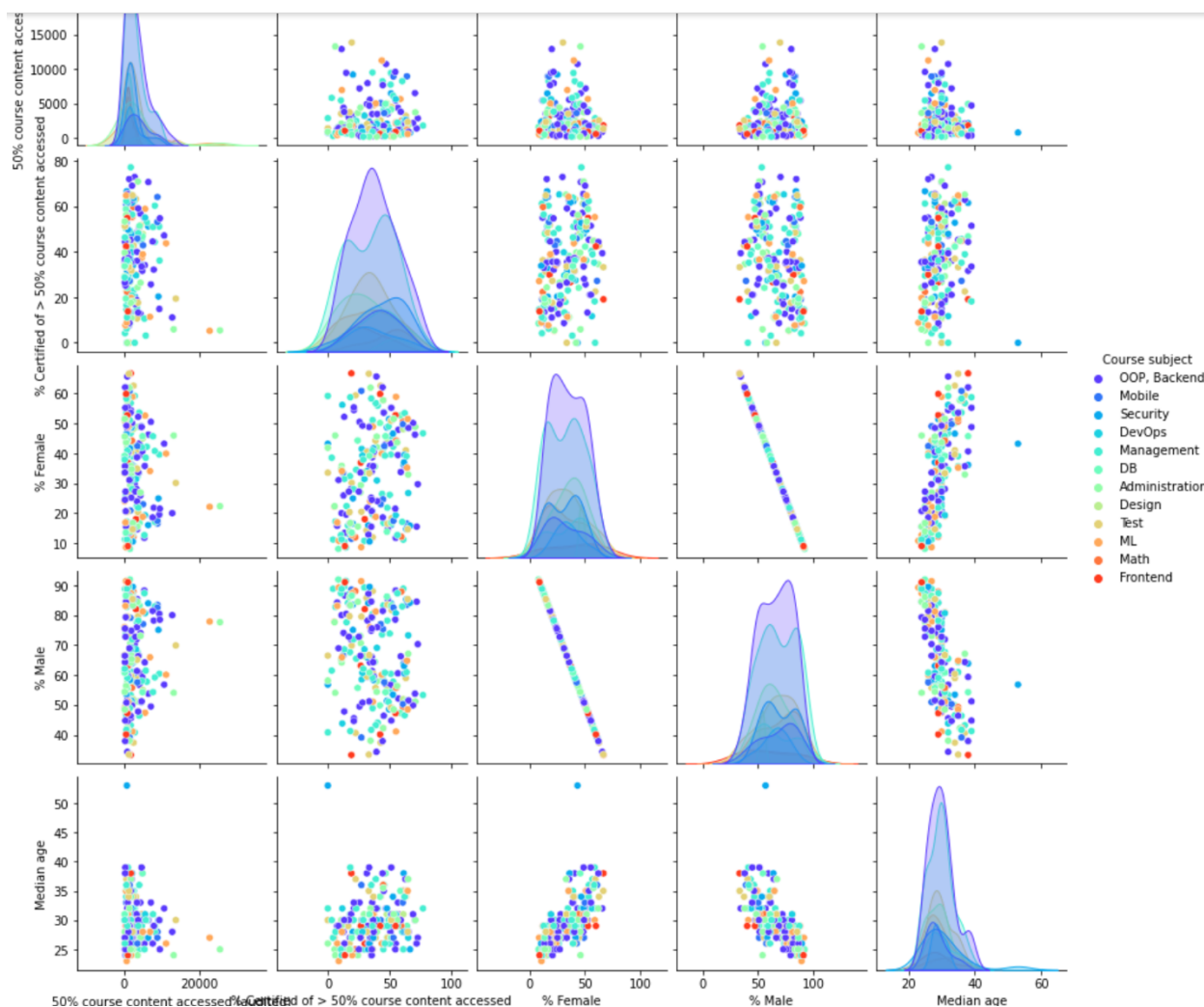


Данные графики в удобной форме показывают медиану (или, если нужно, среднее), нижний и верхний квартили, минимальное и максимальное значение выборки и выбросы. Расстояния между различными частями ящика позволяют определить степень разброса (дисперсии) и асимметрии данных и выявить выбросы.

График типа pairplot показывает отношения между всеми парами переменных.

```
df_pairplot_cols=df_course[['Course subject','50% course content
accessed (audited)','% Certified of > 50% course content
accessed','% Female','% Male','Median age']]
```

```
plt.figure(figsize=(20,20))
sns.pairplot(df_pairplot_cols,hue='Course
subject',palette='rainbow')
```

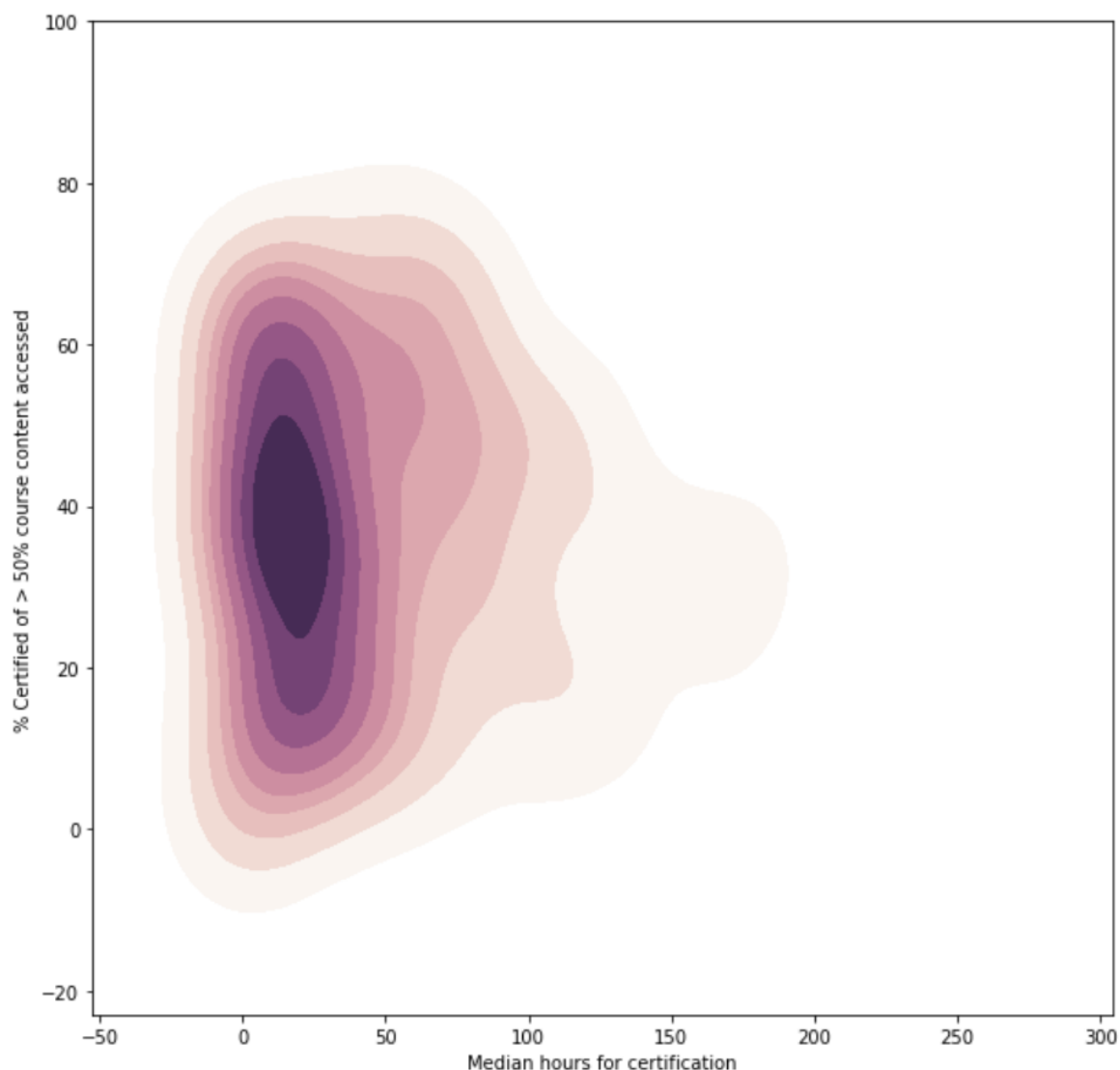


Построим kdeplot для Median hours for certification и % Certified of > 50% course content accessed.

График оценки плотности ядра (KDE) - это метод визуализации распределения наблюдений в наборе данных, аналогичный гистограмме. KDE представляет данные с помощью непрерывной кривой плотности вероятности в одном или нескольких измерениях.

По сравнению с гистограммой KDE может создавать график, который менее загроможден и более понятен, особенно при рисовании нескольких распределений. Но он может внести искажения, если основное распределение ограничено или негладко. Как и в случае гистограммы, качество представления также зависит от выбора хороших параметров сглаживания.

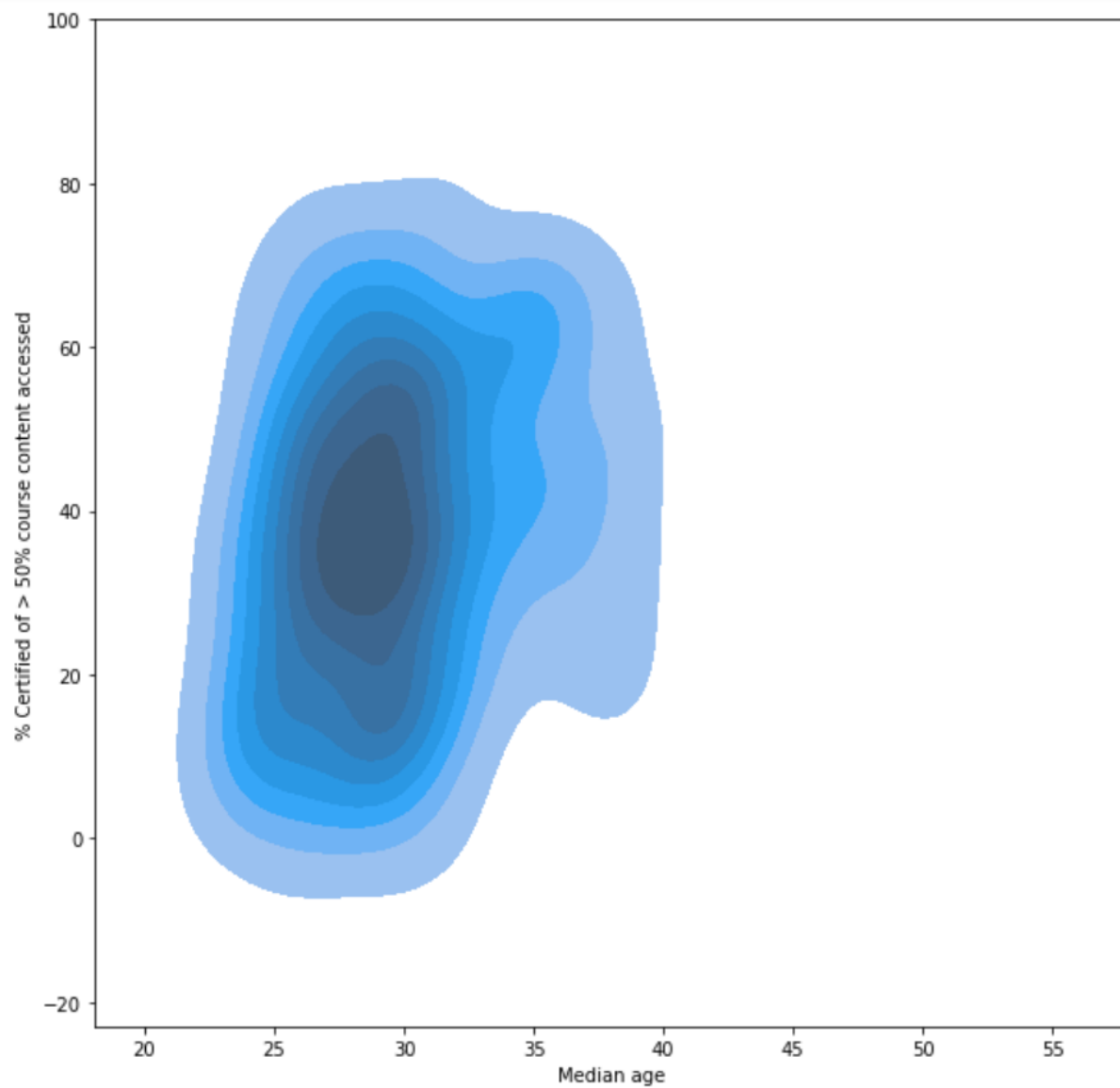
```
x= df_course['Median hours for certification']
y= df_course['% Certified of > 50% course content accessed']
cmap = sns.cubehelix_palette(light=1, as_cmap=True)
plt.figure(figsize=(10,10))
sns.kdeplot(x, y, cmap=cmap, shade=True);
```



Построим kdeplot для Median age и % Certified of > 50% course content accessed

По графику видно, что среднее время для получения сертификата составляет 25 часов?

```
x= df_course['Median age']
y= df_course['% Certified of > 50% course content accessed']
plt.figure(figsize=(10,10))
sns.kdeplot(x, y, shade=True);
```



По графику видно, что средний возраст составляет около 27 лет.

3 Подготовка данных для машинного обучения

Произведем удаление ненужных столбцов ('Certified','50% course content accessed (audited)','Teachers','Launch Date','% Played video')

```
df_course.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 16 columns):
#   Column                                                                 Non-Null Count  Dtype
---  -
0   Launch Date                                                            205 non-null   object
1   Teachers                                                                205 non-null   object
2   Course subject                                                         205 non-null   object
3   Participants                                                            205 non-null   int64
4   50% course content accessed (audited)                                205 non-null   int64
5   Certified                                                              205 non-null   int64
6   % Audited                                                              205 non-null   float64
7   % Certified of > 50% course content accessed                        205 non-null   float64
8   % Played video                                                         205 non-null   float64
9   % Posted in forum                                                      205 non-null   float64
10  Total course hours                                                     205 non-null   float64
11  Median hours for certification                                         205 non-null   float64
12  Median age                                                             205 non-null   float64
13  % Male                                                                  205 non-null   float64
14  % Female                                                                205 non-null   float64
15  % Bachelor's degree or higher                                         205 non-null   float64
dtypes: float64(10), int64(3), object(3)
memory usage: 25.8+ KB
```

```
df_XGB = df_course.drop(['Certified','50% course content accessed
(audited)','Teachers','Launch Date','% Played video'],axis=1)
df_XGB.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 11 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Course subject                             205 non-null    object
1   Participants                               205 non-null    int64
2   % Audited                                  205 non-null    float64
3   % Certified of > 50% course content accessed  205 non-null    float64
4   % Posted in forum                         205 non-null    float64
5   Total course hours                       205 non-null    float64
6   Median hours for certification            205 non-null    float64
7   Median age                               205 non-null    float64
8   % Male                                    205 non-null    float64
9   % Female                                  205 non-null    float64
10  % Bachelor's degree or higher            205 non-null    float64
dtypes: float64(9), int64(1), object(1)
memory usage: 17.7+ KB
```

Произведем преобразование категориальных переменных в серии нулей и единиц, что значительно упрощает их количественное определение и сравнение.

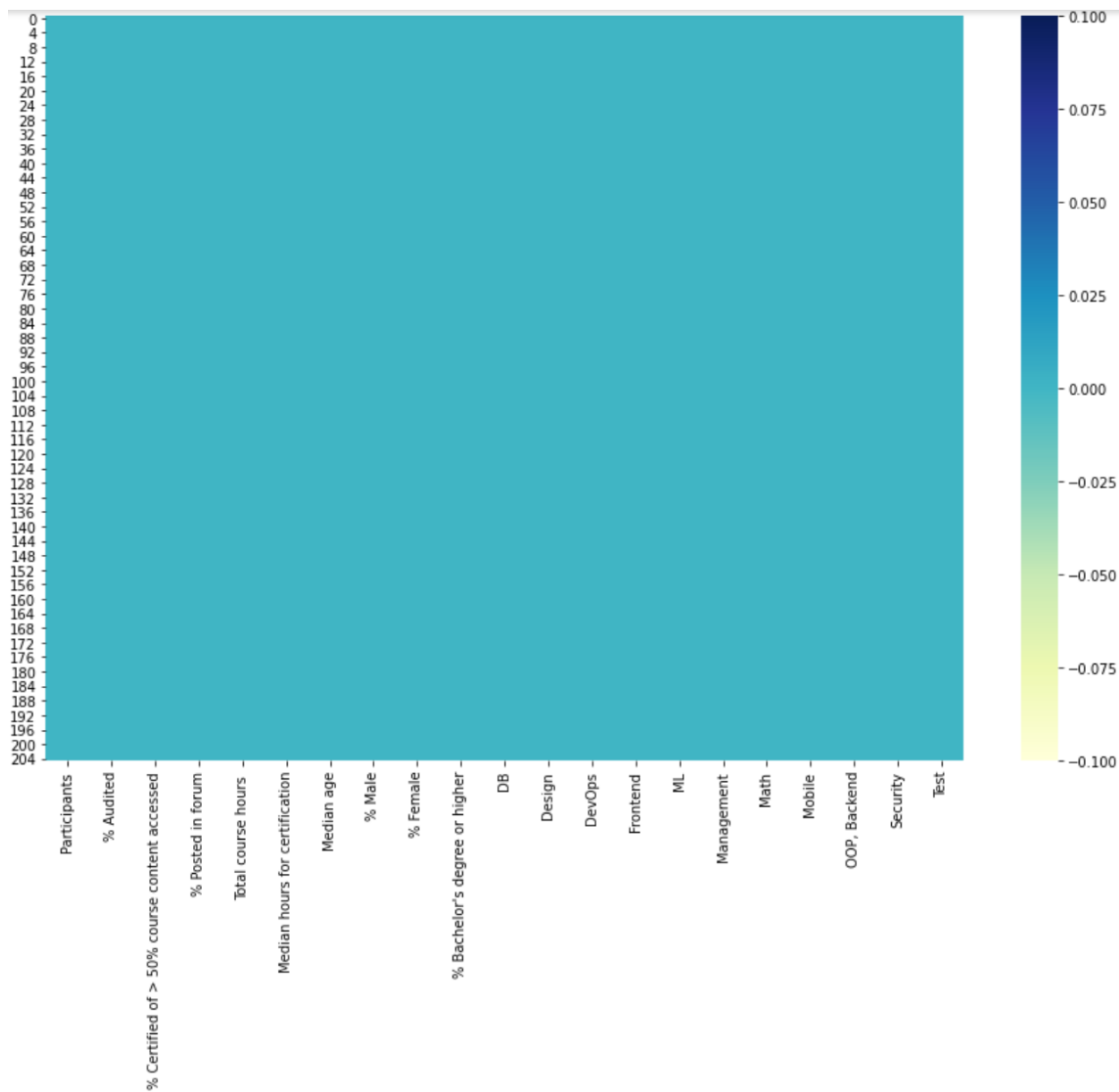
```
CourseSubject = pd.get_dummies(df_XGB['Course
subject'],drop_first=True)
df_XGB.drop(['Course subject'],axis=1,inplace=True)
df_XGB = pd.concat([df_XGB,CourseSubject],axis=1)
df_XGB
```

	Participants	% Audited	% Certified of > 50% course content accessed	% Posted in forum	Total course hours	Median hours for certification	Median age	% Male	% Female	% Bachelor's degree or higher	...	Design	DevOps	Frontend	ML	Management	Math	Mobile	OOP, Backend	Security	Test
0	36105	15.04	54.98	8.17	418.94	64.45	26.0	88.28	11.72	60.68	...	0	0	0	0	0	0	0	1	0	0
1	62709	14.27	64.05	14.38	884.04	78.53	28.0	83.50	16.50	63.04	...	0	0	0	0	0	0	1	0	0	0
2	16663	17.13	72.85	14.42	227.55	61.28	27.0	70.32	29.68	58.76	...	0	0	0	0	0	0	0	1	0	0
3	129400	9.96	11.11	0.00	220.90	0.00	28.0	80.02	19.98	58.78	...	0	0	0	0	0	0	0	1	0	0
4	52521	20.44	47.12	15.98	804.41	76.10	32.0	56.78	43.22	88.33	...	0	0	0	0	0	0	0	1	0	0
...
200	2860	32.17	36.20	6.78	47.23	77.55	29.0	59.10	40.90	76.33	...	0	0	0	0	1	0	0	0	0	0
201	948	25.95	26.42	8.44	4.94	20.87	27.0	66.45	33.55	73.56	...	0	0	0	0	0	0	0	1	0	0
202	1381	18.61	46.30	9.12	3.66	8.38	26.0	60.80	39.20	68.16	...	0	0	0	0	1	0	0	0	0	0
203	385	51.69	33.67	18.70	2.03	12.05	31.0	60.84	39.16	79.67	...	0	0	0	0	0	0	0	0	0	0
204	422	46.21	28.72	14.22	2.02	12.21	30.0	62.09	37.91	79.69	...	0	0	0	0	1	0	0	0	0	0

205 rows x 21 columns

Проверка на нулевые значения:

```
plt.figure(figsize=(15,10))
sns.heatmap(df_XGB.isnull(),cmap="YlGnBu")
```

Убедились, что пустых ячеек нет.

4 Библиотека XGBoost

XGBoost - это оптимизированная распределенная библиотека повышения градиента, разработанная для обеспечения высокой эффективности, гибкости и портативности. Он реализует алгоритмы машинного обучения в рамках платформы Gradient Boosting. XGBoost обеспечивает усиление параллельного дерева (также известное как GBDT, GBM), которое позволяет быстро и точно решить многие проблемы data science. Один и тот же код работает в основной распределенной среде (Hadoop, SGE, MPI) и может решать проблемы, выходящие за рамки множества примеров.

В основе **XGBoost** лежит алгоритм градиентного бустинга деревьев решений. Градиентный бустинг — это техника машинного обучения для задач классификации и регрессии, которая строит модель предсказания в форме ансамбля слабых предсказывающих моделей, обычно деревьев решений. Обучение ансамбля проводится последовательно в отличие, например от бэггинга. На каждой итерации вычисляются отклонения предсказаний уже обученного ансамбля на обучающей выборке. Следующая модель, которая будет добавлена в ансамбль будет предсказывать эти отклонения. Таким образом, добавив предсказания нового дерева к предсказаниям обученного ансамбля мы можем уменьшить среднее отклонение модели, которое является таргетом оптимизационной задачи.

Разделим данные на обучающую и тестовую выборки

```
from sklearn.model_selection import train_test_split
x= df_XGB
y=df_XGB['% Certified of > 50% course content accessed']
x_train, x_test, y_train, y_test = train_test_split(x,y,
test_size=0.4, random_state=109)
import xgboost as xgb
train= xgb.DMatrix(x_train,label=y_train)
test = xgb.DMatrix(x_test, label= y_test)
```

5 Список гиперпараметров XGBoost

`learning_rate`: уменьшение размера шага, используемое для предотвращения переобучения. Диапазон [0,1]

`max_depth`: определяет, насколько глубоко каждое дерево может расти во время любого раунда повышения.

`subsample`: процент использованных образцов на дерево. Низкое значение может привести к неполному оснащению.

`colsample_bytree`: процент функций, используемых в дереве. Высокое значение может привести к переобучению.

`n_estimators`: количество деревьев, которые вы хотите построить.

`objective`: определяет функцию потерь, которая будет использоваться, например, линейная для задач регрессии, логистическая для задач классификации с единственным решением, двоичная для задач классификации с вероятностью.

XGBoost также поддерживает параметры регуляризации, чтобы наказывать модели по мере их усложнения и сводить их к простым (экономным) моделям

`gamma`: контролирует, будет ли данный узел разделен на основе ожидаемого сокращения потерь после разделения. Чем выше значение, тем меньше расщеплений.

`alpha`: L1 регуляризация весов листьев. Большое значение ведет к большей регуляризации.

`lambda`: L2 регуляризация весов листьев и более плавная, чем регуляризация L1.

```
xg_reg = xgb.XGBRegressor(objective='reg:linear',
                           colsample_bytree = 0.5, learning_rate = 0.2,
                           max_depth = 7, alpha = 10, n_estimators = 75)
xg_reg.fit(x_train, y_train)
preds = xg_reg.predict(x_test)
```

Вычислим ошибку прогноза

```
from sklearn.metrics import mean_squared_error
rmse = np.sqrt(mean_squared_error(y_test, preds))
print("RMSE: %f" % (rmse))
```

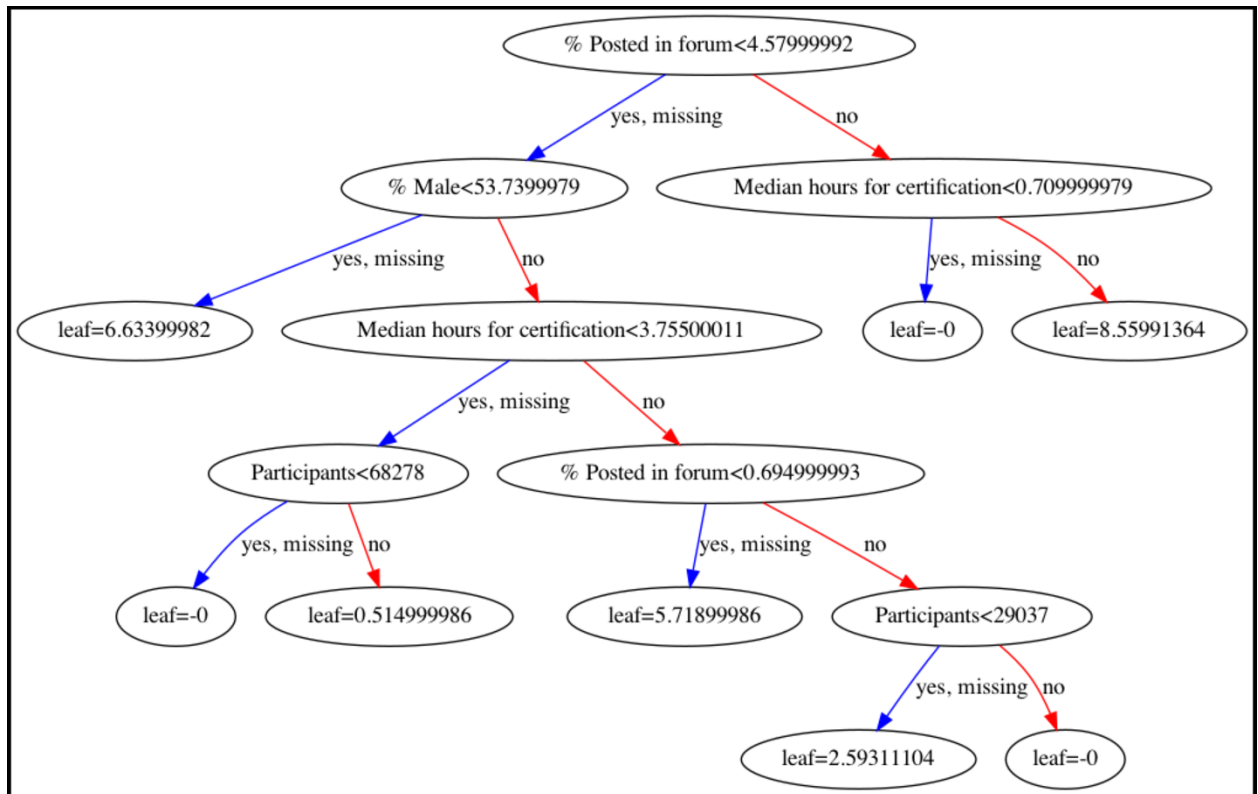
RMSE: 6.109611

Ошибка составила 6%

6 Визуализация деревьев XGBoost

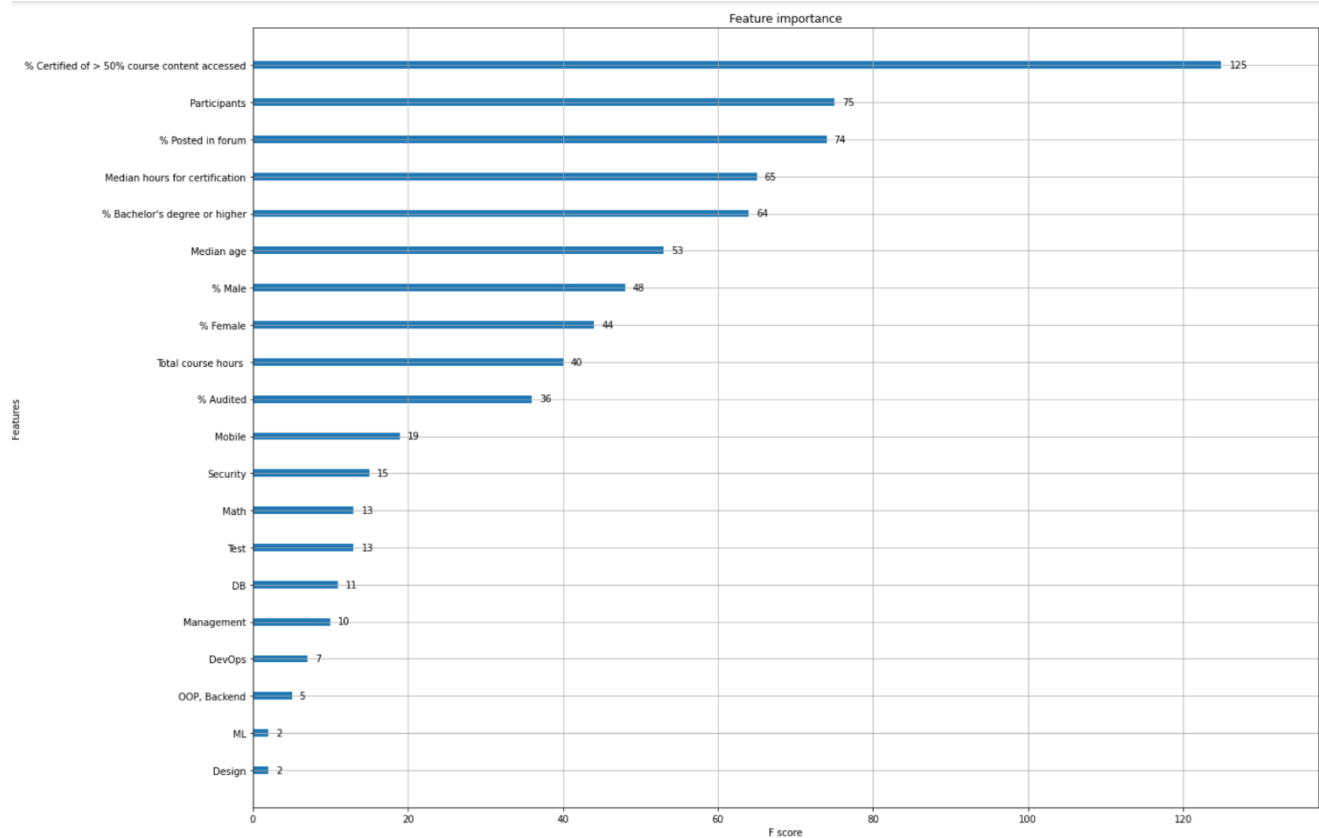
```
import matplotlib.pyplot as plt

xgb.plot_tree(xg_reg, num_trees=0)
plt.rcParams['figure.figsize'] = [20, 15]
plt.show()
```



Построим график важности признаков на основе подобранных деревьев.

```
xgb.plot_importance(xg_reg)
plt.rcParams['figure.figsize'] = [15, 15]
plt.show()
```



По графику видно, что наибольшей важностью обладает признак «% Certified of > 50% course content accessed»

7 Вывод

В данной работе была произведена очистка, визуализация, а также анализ данных от одной из платформ онлайн-курсов. С использованием библиотеки XGBoost была произведена попытка создания модели, которая предсказывает, завершит ли пользователь более 50% курса или нет. Точность прогноза модели составила около 94%.

Список использованных источников

1. Friedman J. Greedy Function Approximation: A Gradient Boosting Machine. — IMS 1999 Reitz Lecture.
2. Nonita Sharma, XGBoost. The Extreme Gradient Boosting for Mining Applications. - 2018 GRIN Verlag
3. xgboost documentation. Режим доступа: <https://xgboost.readthedocs.io/en/latest/> Дата обращения: 28.11.2020
4. seaborn documentation. Режим доступа: <https://seaborn.pydata.org/docs/> Дата обращения: 28.11.2020
5. pandas documentatio.n Режим доступа: <https://pandas.pydata.org/docs/> Дата обращения: 28.11.2020
6. Введение в pandas: анализ данных на Python. Режим доступа: <https://khashtamov.com/ru/pandas-introduction/> Дата обращения: 28.11.2020