Министерство науки и высшего образования
Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования «Рыбинский государственный
авиационный технический университет имени П.А. Соловьева»

ИНСТИТУТ НЕПРЕРЫВНОГО ОБРАЗОВАНИЯ

Кафедра общественных наук

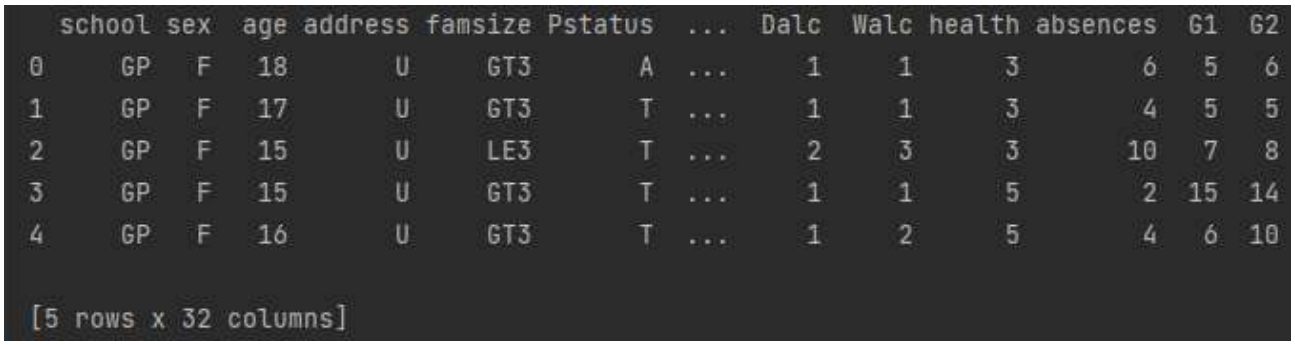
ОТЧЁТ
по дисциплине:
«Методы и алгоритмы анализа данных»
на тему:
«Основы работы с библиотекой *pandas*»

Выполнил: студент группы ИВМ-24 Морозов А. А.
Руководитель: ассистент Вязниковцев Д. А.

Рыбинск 2024

Цель работы: изучить основы работы с библиотекой *pandas*. Провести анализ данных с помощью библиотеки. Построить график с помощью библиотеки *matplotlib*.

1. Основная работа с библиотекой *pandas* – это работа с так называемыми *pandas dataframe*. С помощью функции *head*() выводятся в консоль первые 5 записей, а с помощью функции *tail*() выводятся в консоль последние 5 записей *dataframe*. Также есть возможность применить срезы с помощью различных методов, например, *iloc*(). На рисунке 1 изображён вывод в консоль после исполнения строки *print*(*data.iloc*[:, :-1].*head*())



	school	sex	age	address	famsize	Pstatus	...	Dalc	Walc	health	absences	G1	G2
0	GP	F	18	U	GT3	A	...	1	1	3	6	5	6
1	GP	F	17	U	GT3	T	...	1	1	3	4	5	5
2	GP	F	15	U	LE3	T	...	2	3	3	10	7	8
3	GP	F	15	U	GT3	T	...	1	1	5	2	15	14
4	GP	F	16	U	GT3	T	...	1	2	5	4	6	10

[5 rows x 32 columns]

Рисунок 1 – Вывод в консоль

Pandas позволяет производить так называемую булевую индексацию (*boolean indexing*), например строка *print*(*data*[(*data*['*guardian*'] == '*mother*') & ((*data*['*Mjob*'] == '*teacher*') | (*data*['*Mjob*'] == '*at_home*'))].*head*()) выведет в консоль только те записи из *dataframe*, которые будут удовлетворят условию того, что опекуном является мать и она работает учителем или домохозяйкой (рисунок 2).

```
   school sex  age address famsize Pstatus  ...  Walc  health absences  G1  G2  G3
0      GP   F   18       U     GT3       A  ...     1       3        6   5   6   6
2      GP   F   15       U     LE3       T  ...     3       3       10   7   8  10
10     GP   F   15       U     GT3       T  ...     2       2        0  10   8   9
13     GP   M   15       U     GT3       T  ...     2       3        2  10  10  11
20     GP   M   15       U     GT3       T  ...     1       1        0  13  14  15

[5 rows x 33 columns]
```

Рисунок 2 – Вывод в консоль с условием

2. Анализ данных по заданиям

2.1 Какая причина выбора школы была самой частой? В качестве ответа приведите соответствующее значение признака.

Для получения данных соответствующим поставленному условию проанализируем признак *reason* и подсчитаем с помощью метода *value_counts*() количество значений для всех записей (рисунок 3).

```
def reasons(data):
    lib = {"home": "близко к дому", "reputation": "репутация школы", "course": "предпочтение некоторым предметам",
           "other": "другое"}
    print("Самая частая причина выбора была", lib[data['reason'].value_counts().index[0]])
```

Рисунок 3 – Код для условия 2.1

После выполнения кода с рисунка 3 ответом на вопрос будет «Самая частая причина выбора была предпочтение некоторым предметам»

2.2 Найдите количество студентов, у родителей которых нет никакого образования.

Для получения данных соответствующим поставленному условию проанализируем признаки *Fedu* и *Medu* и подсчитаем с помощью метода *shape*() количество записей (рисунок 4).

```
def MFedu(data):
    print("Нету образования у отца", data[(data['Fedu'] == 0)].shape[0])
    print("Нету образования у матери", data[(data['Medu'] == 0)].shape[0])
    print("Нету образования у обоих родителей", data[((data['Medu'] == 0) & (data['Fedu'] == 0))].shape[0])
```

Рисунок 4 – Код для условия 2.2

После выполнения кода с рисунка 4 ответом на вопрос будет «Нету образования у отца 2. Нету образования у матери 3. Нету образования у обоих родителей 0».

2.3 Найдите минимальный возраст учащегося школы Mousinho da Silveira.

Для получения данных соответствующим поставленному условию проанализируем признак *school* на совпадение со значением *MS* и из полученных данных выберем запись с минимальным значением признака *age* (рисунок 5).

```
def age(data):
    print("Минимальный возраст ученика в школе Mousinho da Silveira составляет",
          data[(data['school'] == "MS")]['age'].min(), "лет")
```

Рисунок 5 – Код для условия 2.3

После выполнения кода с рисунка 5 ответом на вопрос будет «Минимальный возраст ученика в школе *Mousinho da Silveira* составляет 17 лет».

2.4 Найдите количество студентов, имеющих нечетное число пропусков

Для получения данных соответствующим поставленному условию проанализируем признак *absences* на нечётное значение у записи и подсчитаем количество нечётных (рисунок 6).

```
def cntofabsences(data):
    print(f"Количество учеников с нечётным числом пропусков составляет \
{len([i for i in data['absences'] if i % 2 == 1])}")
```

Рисунок 6 – Код для условия 2.4

После выполнения кода с рисунка 6 ответом на вопрос будет «Количество учеников с нечётным числом пропусков составляет 41».

2.5 Найдите разность между средними итоговыми оценками студентов, состоящих и не состоящих в романтических отношениях. В качестве ответа приведите число, округленное до двух значащих цифр после запятой.

Для получения данных соответствующим поставленному условию проанализируем признак *romantic* на «*yes*» и «*no*» и сформируем таким образом 2 группы. У этих двух групп у признака *G3* найдём среднее значение и разность между ними по модулю (рисунок 7).

```
def raznosti(data):
    d = data[["G3", "romantic"]]
    NoRomantic = d.query("romantic == 'no'")['G3'].mean()
    YesRomantic = d.query("romantic == 'yes'")['G3'].mean()
    print(
        f"Разность между средними итоговыми оценками студентов, состоящих и не состоящих в романтических "
        f"отношениях составляет {round(abs(NoRomantic - YesRomantic), 2)} баллов")
```

Рисунок 7 – Код для условия 2.5

После выполнения кода с рисунка 7 ответом на вопрос будет «Разность между средними итоговыми оценками студентов, состоящих и не состоящих в романтических отношениях составляет 1.26 баллов».

2.6 Сколько занятий пропустило большинство студентов с самым частым значением наличия внеклассных активностей?

Для получения данных соответствующим поставленному условию проанализируем признак *activities* на «*yes*» и «*no*» и определим самое частое значение. Далее определим самое частое значения признака *absences* с помощью метода *value_counts*() (рисунок 8).

```python
def six(data):
    activities = data['activities'].value_counts().index[0]
    d = data[['absences', 'activities']]
    absences = d.query(f"activities == '{activities}'")['absences'].value_counts()
    print(
        f"Чаще всего студенты с внеклассными занятиями имели {absences.index[0]} "
        f"пропусков и количество таких студентов составляет {absences[0]}")
```

Рисунок 8 – Код для условия 2.6

После выполнения кода с рисунка 8 ответом на вопрос будет «Чаще всего студенты с внеклассными занятиями имели 0 пропусков и количество таких студентов составляет 51».

2.7 Постройте гистограмму, отражающую распределение оценок за первый семестр *G*1, чтобы визуализировать частоту каждой оценки.

Для получения данных соответствующим поставленному условию проанализируем признак *G*1 и рассчитаем частоту вхождений оценок с помощью метода определим самое частое значение *value_counts*(). Далее изменим график гистограммы для удобного восприятия и выведем его (рисунок 9).

```python
def histogramma(data):
    import numpy as np
    d = data['G1'].value_counts()
    print(d)
    plt.figure(figsize=(10, 7))
    plt.title('Оценки за первый семестр')
    plt.bar(d.index, d.values, edgecolor='black', linewidth=2, width=1)
    x_ticks = np.linspace(1, 20, 20)
    plt.xticks(x_ticks)
    plt.xlabel('Оценка')
    plt.ylabel('Кол-во студентов')
    plt.show()
```

Рисунок 9 – Код для условия 2.7

На рисунке 10 изображена гистограмма, отражающая распределение оценок за первый семестр *G*1.

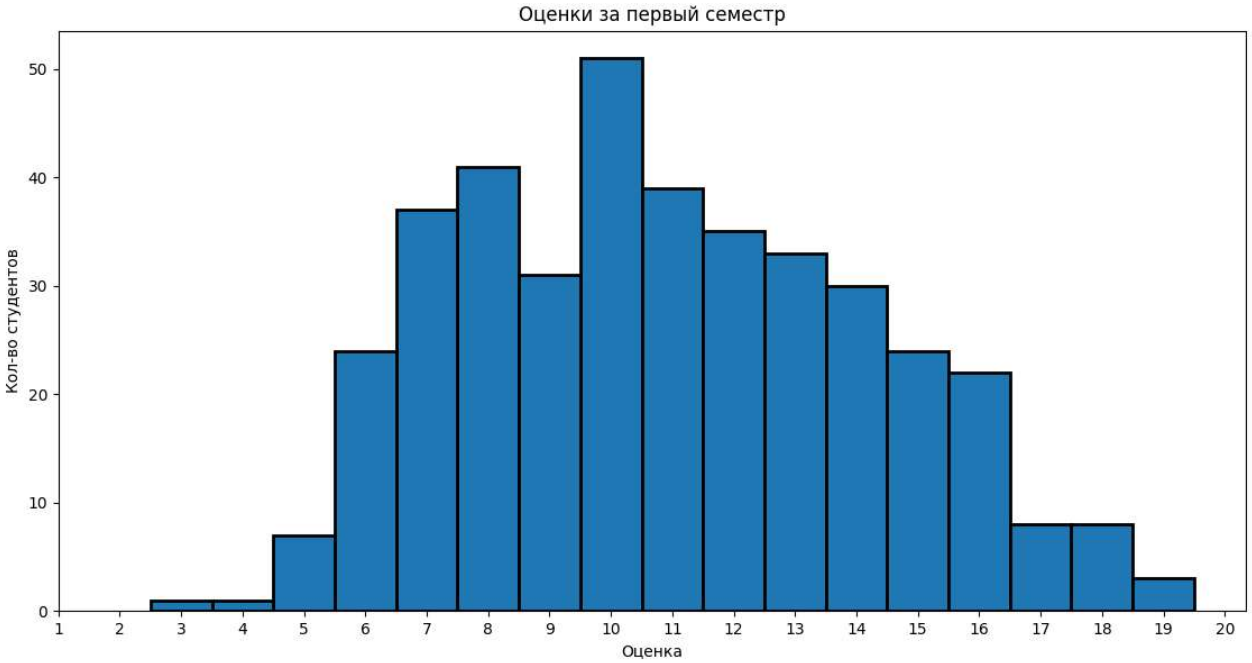


Рисунок 10 – Гистограмма с частотой оценок

Вывод: в результате выполнения лабораторной работы были изучены основы работы с библиотекой *pandas*. Проведён анализ данных с помощью библиотеки. Построен график с помощью библиотеки *matplotlib*.