ГУАП

КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

ассистент		В. В. Боженко
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

Анализ зависимостей между признаками в двумерном наборе данных

по курсу: ВВЕДЕНИЕ В АНАЛИЗ ДАННЫХ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4917		Е.А. Ясиновский
	подпись, дата	инициалы, фамилия

Санкт-Петербург 2022

Цель работы: изучения связи между признаками набора данных.

Вариант 2: Файл liver.csv, в котором предоставленны данные о анализах для диагностирования заболеваний печени у пациентов.

```
import pandas as pd
import numpy as np
import seaborn as sns
df = pd.read_csv("liver.csv")
df.duplicated().where(lambda x: x == True).dropna() ##  Явных дубликатов не найдено
df = df.dropna().reset_index()

# for col in df.columns:
#     print(df[col].unique())

df["Gender"] = df["Gender"].replace("Mal","Male")
df["Gender"] = df["Gender"].replace(["Male","Female"],[0,1])
df["Dataset123"] = df["Dataset123"].replace(["yes","no"],["1","2"])
df["Dataset123"] = df["Dataset123"].replace(["1","2"],["1","0"])
df["Aspartate_Aminotransferase"] = df["Aspartate_Aminotransferase"].replace("3a4","34")
df = df.rename(columns={"Dataset123":"IsSick"})
df = df.astype({
    'Gender':'int64',
    "IsSick":"int64",
    "Alkaline_Phosphotase":"int64",
    "Aspartate_Aminotransferase":"int64"})
df.head()
```

	index	Age	Gender	TotalBilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin	Albumin_and_Globulin_Ratio	IsSick
0	0	65	1	0.7	0.1	187	16	18	6.8	3.3	0.90	1
1	1	62	0	10.9	5.5	699	64	100	7.5	3.2	0.74	1
2	2	62	0	7.3	4.1	490	60	68	7.0	3.3	0.89	1
3	3	58	0	1.0	0.4	182	14	20	6.8	3.4	1.00	1
4	4	72	0	3.9	2.0	195	27	59	7.3	2.4	0.40	1

Рисунок 1 — Предварительная подготовка данных, чистка и дубликатов, устранение некорректных строк

Далее были построены графики рассеяния для интересующих меня параметров

```
df.plot(x='Direct_Bilirubin',y='TotalBilirubin',kind='scatter')
df.plot(x='Alamine_Aminotransferase',y='Aspartate_Aminotransferase',kind='scatter')
df.plot(x='Albumin',y='Total_Protiens',kind='scatter')
```

Рисунок 2 — Вызов отрисовки графиков

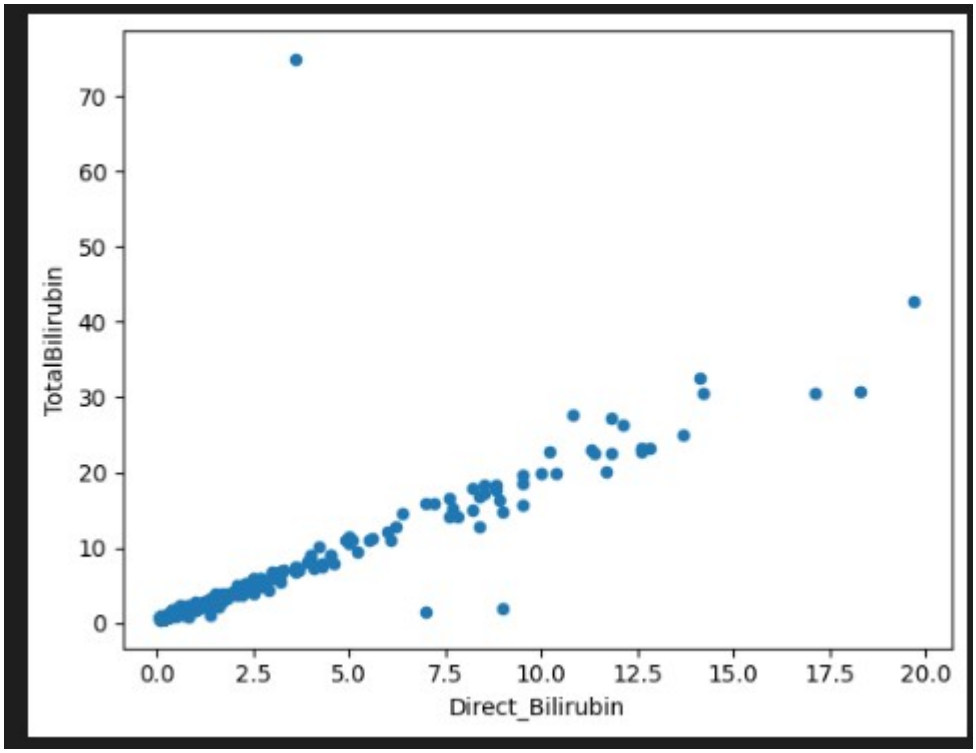


Рисунок 3 — Зависимость общего билирубина от прямого

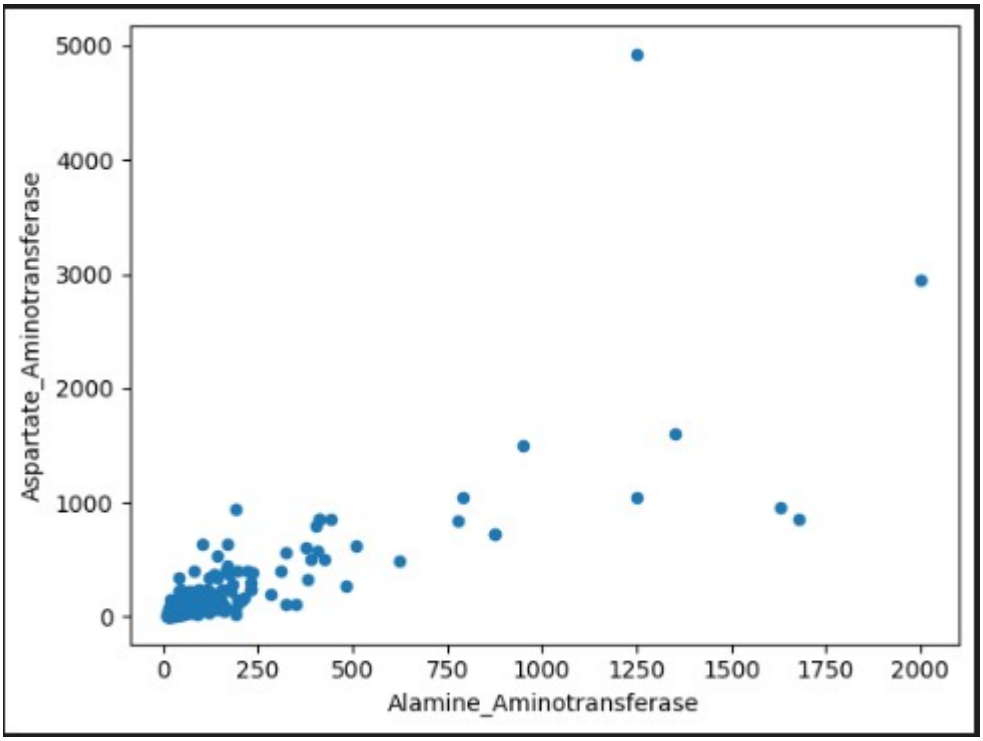


Рисунок 4 — Зависимость аспартатаминотрансфераза от аламиноаминотрансферазы

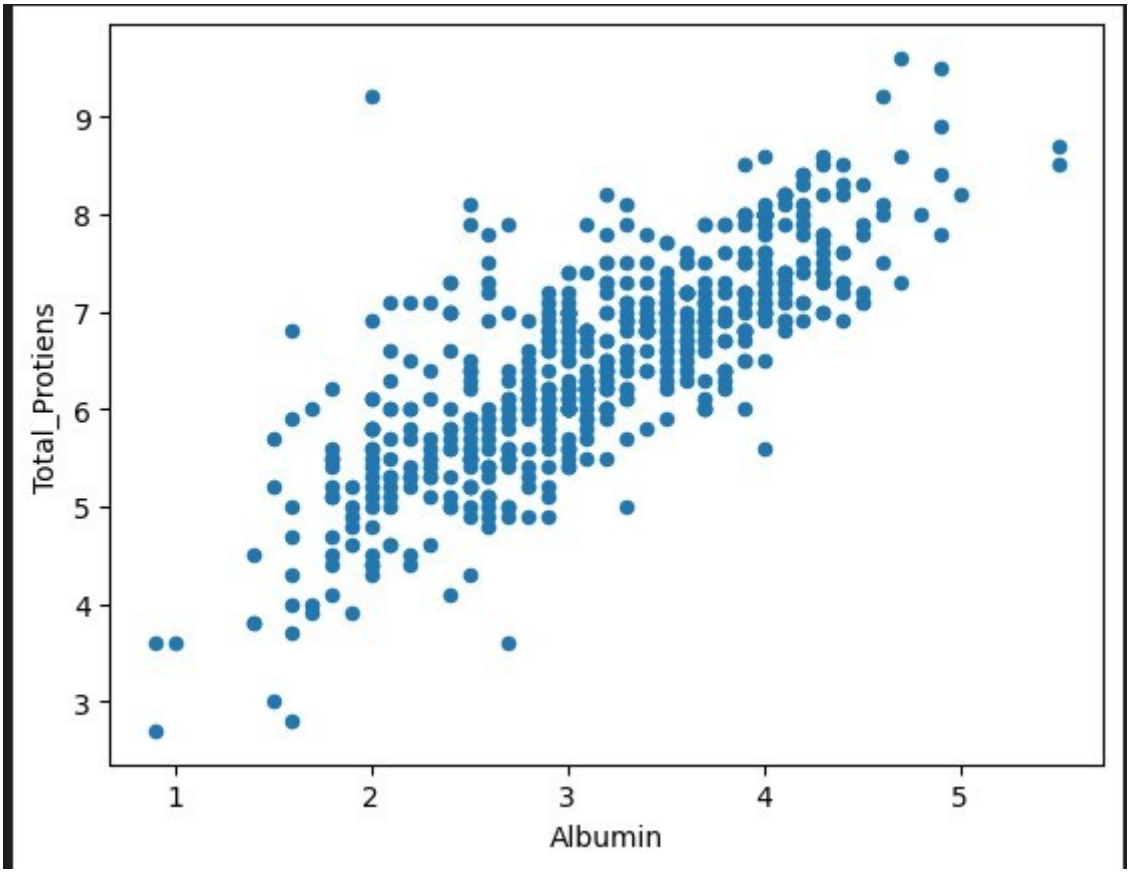


Рисунок 5 — Зависимость общего количества белков от альбумина

Исходя из графиков выше можно увидеть явную прямую зависимость общего количества билирубина от прямого, а также некую размытую прямую зависимость уровня белков от уровня альбумина. В связи с тем, почти все точки графика зависимости аспартатаминотрансфераза от аламиноаминотрансферазы лежат в одной области сложно точно говорить о их прямой зависимости, однако это можно предположить, учитывая то, что крайние правые точки этого графика лежат выше чем его левые точки.

Далее мною была построена матрица рассеяния для всех показателей датафрейма

```
pd.plotting.scatter_matrix(df,figsize=(20,20))
```

Рисунок 6 — Построение матрицы рассеяния

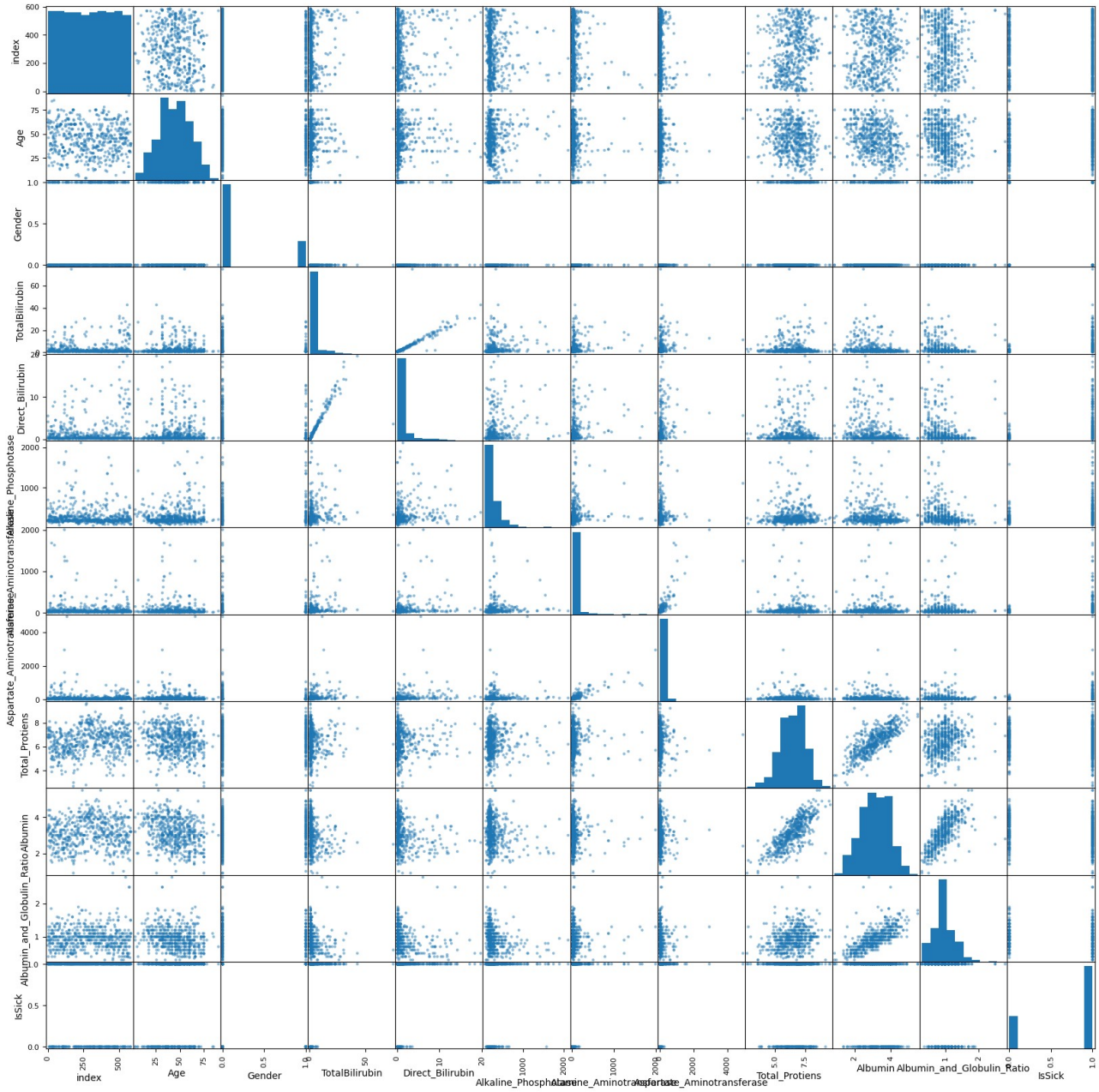


Рисунок 7 — Матрица рассеяния

Уже исходя их данных полученных в матрице рассеяния можно было сделать вывод зависимости ключевого показателя от некоторых других параметров, а именно: билирубина, щелочной фосфатазы, аламиноаминотранфаразы и ампартатаминотрансферазы, и почти не зависит от уровня белков, уровня альбумина и отношения уровня альбумина к уровню глобулина.

Проверить эти предположения мы можем посредством нахождения коэффициента Пирсона для заданных показателей.

```
BilirubCorr = df['TotalBilirubin'].corr(df["Direct_Bilirubin"])
print(BilirubCorr)
amf = df["Alamine_Aminotransferase"].corr(df["Aspartate_Aminotransferase"])
print(amf)
```

```
0.8745632082572773
0.7919116749914477
```

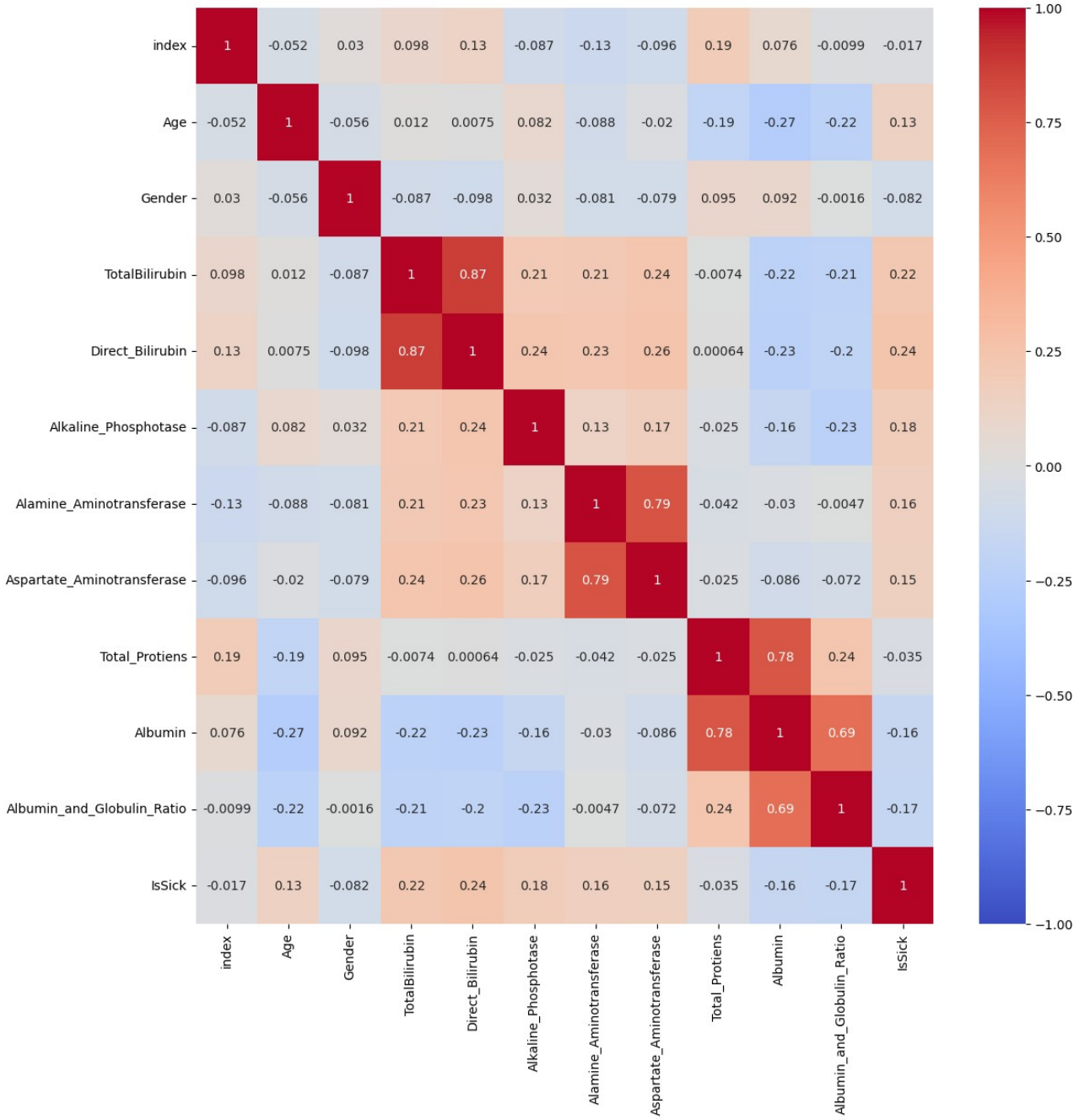
Рисунок 8 — Поиск коэффициента Пирсона

Рассчитанные коэффициенты подтверждают предположение о корреляции выбранных показателей, так как коэффициента Пирсона не положителен, и находится на достаточном удалении от нуля, чтобы не считать это погрешностью.

Также для наглядного визуального поиска корреляции и ковариции можно построить тепловую карту корреляции.

```
import matplotlib.pyplot as plt
fig, ax = plt.subplots(figsize=(13,13))
sns.heatmap(df.corr(),vmin=-1,vmax=1,annot=True,cmap="coolwarm",ax=ax)
```

Рисунок 9 — Построение тепловой карты корреляции



Рисуно 10 — Тепловая карта

При рассмотрении данной тепловой карты зависимости и корреляция\ковариация различных показателей становятся более наглядными, однако скорее всего стоит учесть, что если показатель корреляции близок к нулю(слишком мал), то стоит относится к такому показателю не как с слабой корреляции, а скорее как к погрешности.

Исходный файл Jupyter Notebook находится на сервере GitHub по адресу https://github.com/EgorYasinovskiy/Data-Analys/blob/master/JP2/main.ipynb

Вывод: в ходе выполнения данной лабораторной работы научился строить графики зависимости параметров в датафрейме а также матрицы рассеяния среди всех или указанных параметров в датафрейме. С помощью этого произвел анализ данных о пациентах и нашел зависимости уровня некоторых белков и гормонов от диагноза пациента. В ходе данной лабораторной работы выяснилось, что у пациентов с больной печенью обычно всегда выше

уровень следующих показателей: билирубин, щелочная фосфатаза, аламиноаминотранфараза и ампартатаминотрансфераза