МГТУ им. Н.Э. Баумана

Факультет "Информатика и системы управления"

Кафедра "Системы обработки информации и управления"

ДИСЦИПЛИНА:

"Технологии машинного обучения"

Отчет по лабораторной работе №1

"Разведочный анализ данных. Исследование и визуализация данных"

Выполнил:

Студент группы ИУ5-61Б

Сукиасян В.М.

Преподаватель:

Гапанюк Ю.Е.

Москва 2020

Задание:

•Выбрать набор данных (датасет).

Для лабораторных работ не рекомендуется выбирать датасеты большого размера.

•Создать ноутбук, который содержит следующие разделы:

1.Текстовое описание выбранного Вами набора данных.

2.Основные характеристики датасета.

3.Визуальное исследование датасета.

4.Информация о корреляции признаков.

•Сформировать отчет и разместить его в своем репозитории на github.

Выполнение ЛР:

1) Текстовое описание набора данных

В качестве набора данных мы будем использовать набор данных по сердечным заболеваниям.

Файл содержит следующие колонки:

- age — возраст
- sex - пол. Целевой признак(0-женщина, 1-мужчина).
- cp - тип грудной боли (1 — 4).
- trestbps - артериальное давление в покое (в мм рт. ст. при поступлении в больницу).
- chol - сыворотка холесторальная в мг / дл.
- fbs - (уровень сахара в крови натощак> 120 мг / дл) (1 = верно; 0 = неверно).
- restecg - результаты электрокардиографии в покое.
- thalach - максимальная частота сердечных сокращений.
- exang - стенокардия, вызванная физической нагрузкой (1 = да; 0 = нет).
- oldpeak - Депрессия ST, вызванная физическими упражнениями относительно отдыха.
- slope - наклон пика упражнений сегмента ST.
- ca - количество крупных сосудов (0-3), окрашенных по цвету.
- thal - 3 = нормально; 6 = исправленный дефект; 7 = обратимый дефект.
- target - 1 or 0

1) Текстовое описание набора данных

В качестве набора данных мы будем использовать набор данных по сердечным заболеваниям.

Файл содержит следующие колонки:

age - возраст

sex - пол. Целевой признак(0-женщина, 1-мужчина).

cp - тип грудной боли (1 - 4).

trestbps - артериальное давление в покое (в мм рт. ст. при поступлении в больницу).

chol - сыворотка холестеральная в мг / дл.

fbs - (уровень сахара в крови натощак> 120 мг / дл) (1 = верно; 0 = неверно).

restecg - результаты электрокардиографии в покое.

thalach - максимальная частота сердечных сокращений.

exang - стенокардия, вызванная физической нагрузкой (1 = да; 0 = нет).

oldpeak - Депрессия ST, вызванная физическими упражнениями относительно отдыха.

slope - наклон пика упражнений сегмента ST.

ca - количество крупных сосудов (0-3), окрашенных по цвету.

thal - 3 = нормально; 6 = исправленный дефект; 7 = обратимый дефект.

target - 1 or 0

2) Основные характеристики датасета

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [2]:
```
# Будем анализировать данные только на обучающей выборке
data = pd.read_csv('heart.csv', sep=",")
```

In [3]:
```
# Первые 5 строк датасета
data.head()
```

Out[3]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

In [4]:
```
data.shape
```

Out[4]:
```
(303, 14)
```

In [5]:
```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 303
```

In [6]:
```
# Список колонок
data.columns
```

Out[6]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

In [7]:
```
# Список колонок с типами данных
data.dtypes
```

Out[7]:
```
age           int64
sex           int64
cp            int64
trestbps      int64
chol          int64
fbs           int64
restecg       int64
thalach       int64
exang         int64
oldpeak     float64
slope         int64
ca            int64
thal          int64
target        int64
dtype: object
```

In [8]:
```
# Проверим наличие пустых значений
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
age - 0
sex - 0
cp - 0
trestbps - 0
chol - 0
fbs - 0
restecg - 0
thalach - 0
exang - 0
oldpeak - 0
slope - 0
ca - 0
thal - 0
target - 0
```

In [9]:
```
# Основные статистические характеристки набора данных
data.describe()
```

Out[9]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	3
mean	54.366337	0.683168	0.966997	131.623762	246.264026	0.148515	0.528053	149.646865	0.326733	1.039604	1.399340	0.729373	
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198	0.525860	22.905161	0.469794	1.161075	0.616226	1.022606	
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	
25%	47.500000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.500000	0.000000	0.000000	1.000000	0.000000	
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000	153.000000	0.000000	0.800000	1.000000	0.000000	
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000	1.000000	166.000000	1.000000	1.600000	2.000000	1.000000	
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	

In [10]:
```
# Определим уникальные значения для целевого признака
data['sex'].unique()
```

Out[10]:
```
array([1, 0])
```

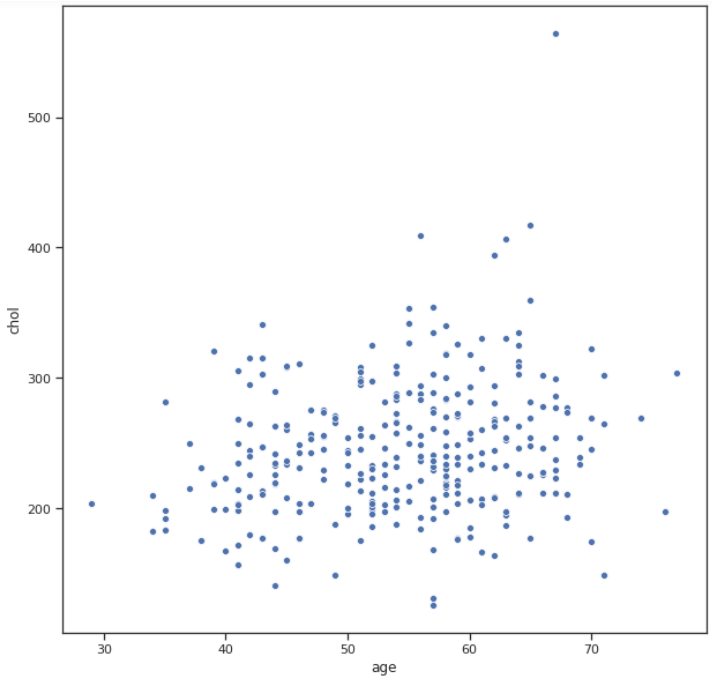
3) Визуальное исследование датасета

Для визуального исследования могут быть использованы различные виды диаграмм, мы построим только некоторые варианты диаграмм, которые используются достаточно часто.

Диаграмма рассеяния Позволяет построить распределение двух колонок данных и визуально обнаружить наличие зависимости. Не предполагается, что значения упорядочены (например, по времени).

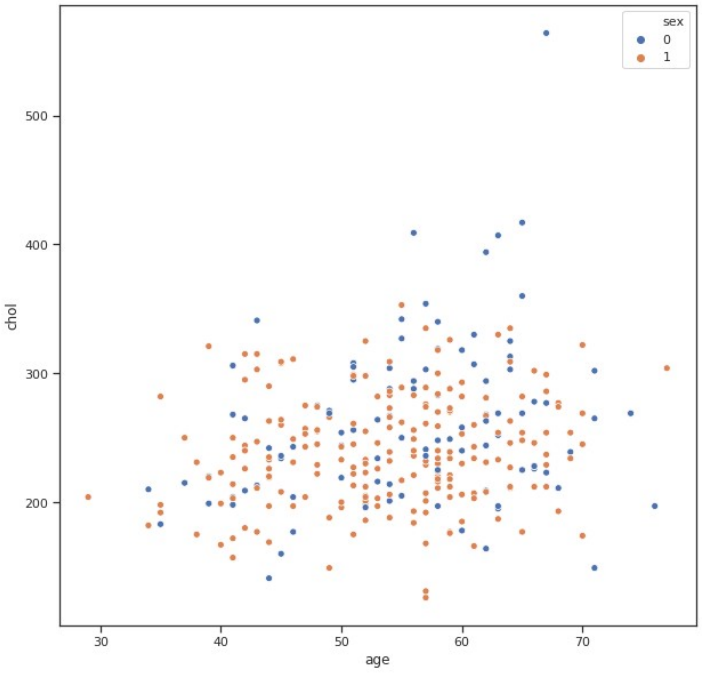
In [11]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='age', y='chol', data=data)
```

Out[11]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8ded503a90>
```



In [12]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='age', y='chol', data=data, hue='sex')
```

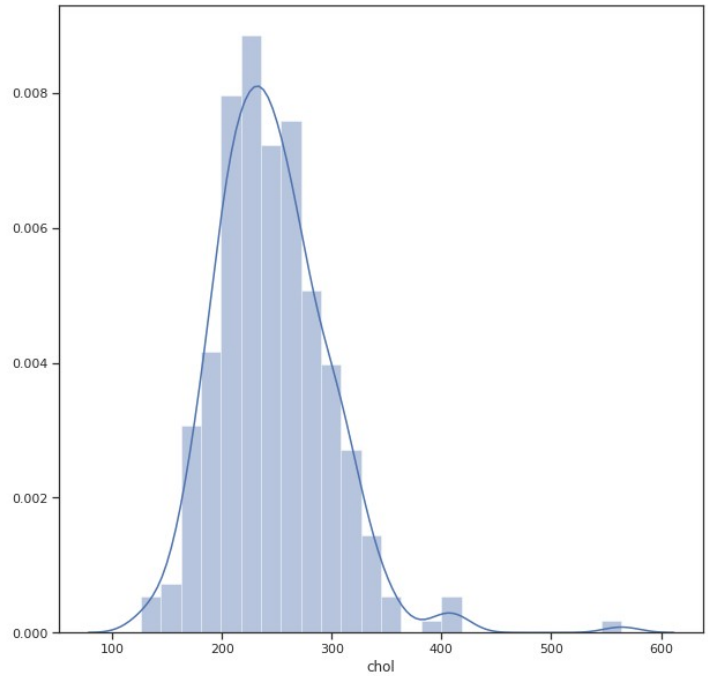
Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8decc60550>



In [13]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['chol'])
```

Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8decc21b90>

Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8decc21b90>

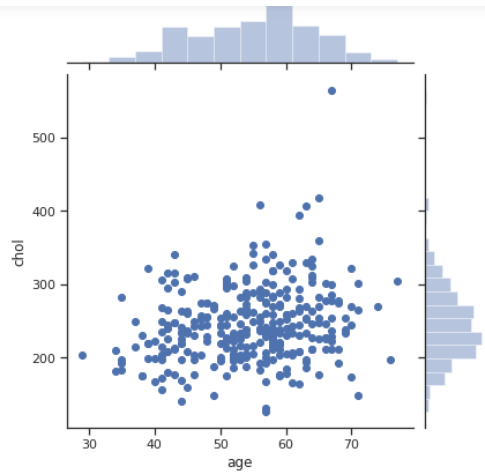


Jointplot

Комбинация гистограмм и диаграмм рассеивания.

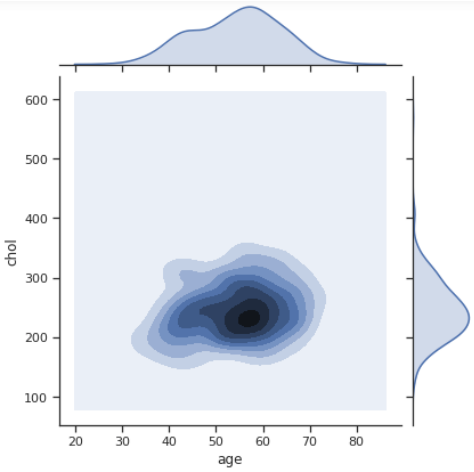
In [14]:
```
sns.jointplot(x='age', y='chol', data=data)
```

Out[14]: <seaborn.axisgrid.JointGrid at 0x7f8decb77390>



In [15]:
```
sns.jointplot(x='age', y='chol', data=data, kind="kde")
```

Out[15]: <seaborn.axisgrid.JointGrid at 0x7f8dec7eb810>



"Парные диаграммы"

Комбинация гистограмм и диаграмм рассеивания для всего набора данных.

Выводится матрица графиков. На пересечении строки и столбца, которые соответствуют двум показателям, строится диаграмма рассеивания. В главной диагонали матрицы строятся гистограммы распределения соответствующих показателей.

In [16]:
```
sns.pairplot(data)
```

4) Информация о корреляции признаков

Проверка корреляции признаков позволяет решить две задачи:

Понять какие признаки (колонки датасета) наиболее сильно коррелируют с целевым признаком (в нашем примере это колонка "sex"). Именно эти признаки будут наиболее информативными для моделей машинного обучения. Признаки, которые слабо коррелируют с целевым признаком, можно попробовать исключить из построения модели, иногда это повышает качество модели. Нужно отметить, что некоторые алгоритмы машинного обучения автоматически определяют ценность того или иного признака для построения модели. Понять какие нецелевые признаки линейно зависимы между собой. Линейно зависимые признаки, как правило, очень плохо влияют на качество моделей. Поэтому если несколько признаков линейно зависимы, то для построения модели из них выбирают какой-то один признак.

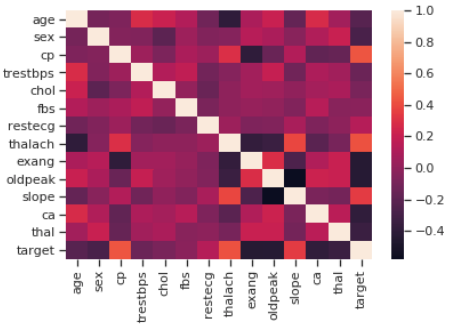
In [19]:
```
data.corr()
```

Out[19]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	ta
age	1.000000	-0.098447	-0.068653	0.279351	0.213678	0.121308	-0.116211	-0.398522	0.096801	0.210013	-0.168814	0.276326	0.068001	-0.225
sex	-0.098447	1.000000	-0.049353	-0.056769	-0.197912	0.045032	-0.058196	-0.044020	0.141664	0.096093	-0.030711	0.118261	0.210041	-0.280
cp	-0.068653	-0.049353	1.000000	0.047608	-0.076904	0.094444	0.044421	0.295762	-0.394280	-0.149230	0.119717	-0.181053	-0.161736	0.433
trestbps	0.279351	-0.056769	0.047608	1.000000	0.123174	0.177531	-0.114103	-0.046698	0.067616	0.193216	-0.121475	0.101389	0.062210	-0.144
chol	0.213678	-0.197912	-0.076904	0.123174	1.000000	0.013294	-0.151040	-0.009940	0.067023	0.053952	-0.004038	0.070511	0.098803	-0.085
fbs	0.121308	0.045032	0.094444	0.177531	0.013294	1.000000	-0.084189	-0.008567	0.025665	0.005747	-0.059894	0.137979	-0.032019	-0.028
restecg	-0.116211	-0.058196	0.044421	-0.114103	-0.151040	-0.084189	1.000000	0.044123	-0.070733	-0.058770	0.093045	-0.072042	-0.011981	0.137
thalach	-0.398522	-0.044020	0.295762	-0.046698	-0.009940	-0.008567	0.044123	1.000000	-0.378812	-0.344187	0.386784	-0.213177	-0.096439	0.421
exang	0.096801	0.141664	-0.394280	0.067616	0.067023	0.025665	-0.070733	-0.378812	1.000000	0.288223	-0.257748	0.115739	0.206754	-0.436
oldpeak	0.210013	0.096093	-0.149230	0.193216	0.053952	0.005747	-0.058770	-0.344187	0.288223	1.000000	-0.577537	0.222682	0.210244	-0.430
slope	-0.168814	-0.030711	0.119717	-0.121475	-0.004038	-0.059894	0.093045	0.386784	-0.257748	-0.577537	1.000000	-0.080155	-0.104764	0.345
ca	0.276326	0.118261	-0.181053	0.101389	0.070511	0.137979	-0.072042	-0.213177	0.115739	0.222682	-0.080155	1.000000	0.151832	-0.391
thal	0.068001	0.210041	-0.161736	0.062210	0.098803	-0.032019	-0.011981	-0.096439	0.206754	0.210244	-0.104764	0.151832	1.000000	-0.344
target	-0.225439	-0.280937	0.433798	-0.144931	-0.085239	-0.028046	0.137230	0.421741	-0.436757	-0.430696	0.345877	-0.391724	-0.344029	1.000

In [20]:
```
sns.heatmap(data.corr())
```

Out[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8dea6c5a50>
```



In [21]:
```
# Треугольный вариант матрицы
mask = np.zeros_like(data.corr(), dtype=np.bool)
# чтобы оставить нижнюю часть матрицы
# mask[np.triu_indices_from(mask)] = True
# чтобы оставить верхнюю часть матрицы
mask[np.tril_indices_from(mask)] = True
sns.heatmap(data.corr(), mask=mask, fmt='.3f')
```

Out[21]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8dea626a10>
```