info_98_project_02_data_exploration_through_visualization

March 12, 2019

1 INFO 98: Data Science Skills, Spring 2019

1.1 Lecture 05: Data Visualization

1.2 Table of Contents

Setup ____

In [66]:
```
# Comment out !pip install statements if you have those packages installed.
# Note: Add any additional import statements you think you need
#!pip install numpy
#!pip install pandas
#!pip install matplotlib
#!pip install seaborn


!pip3 install seaborn==0.9.0

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

```
Requirement already satisfied: seaborn==0.9.0 in /srv/app/venv/lib/python3.6/site-packages
Requirement already satisfied: pandas>=0.15.2 in /srv/app/venv/lib/python3.6/site-packages (fr
```

```
Requirement already satisfied: numpy>=1.9.3 in /srv/app/venv/lib/python3.6/site-packages (from
Requirement already satisfied: matplotlib>=1.4.3 in /srv/app/venv/lib/python3.6/site-packages
Requirement already satisfied: scipy>=0.14.0 in /srv/app/venv/lib/python3.6/site-packages (fro
Requirement already satisfied: python-dateutil>=2.5.0 in /srv/app/venv/lib/python3.6/site-packa
Requirement already satisfied: pytz>=2011k in /srv/app/venv/lib/python3.6/site-packages (from p
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /srv/app/venv/lib/py
Requirement already satisfied: cycler>=0.10 in /srv/app/venv/lib/python3.6/site-packages (from
Requirement already satisfied: kiwisolver>=1.0.1 in /srv/app/venv/lib/python3.6/site-packages
Requirement already satisfied: six>=1.5 in /srv/app/venv/lib/python3.6/site-packages (from pyth
Requirement already satisfied: setuptools in /srv/app/venv/lib/python3.6/site-packages (from ki
```

Dataset 1: Heart Disease Data Set ___

1.2.1 Background:

Link to Dataset: https://www.kaggle.com/ronitf/heart-disease-uci Link to Background Information: https://archive.ics.uci.edu/ml/datasets/Heart+Disease

This database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0).

The names and social security numbers of the patients were recently removed from the database, replaced with dummy values.

Data Preprocessing and Manipulation

In [67]:
```
#import data of hear_disease
heart_disease=pd.read_csv("heart.csv")
heart_disease
```

Out[67]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	\
0	63	1	3	145	233	1	0	150	0	2.3	
1	37	1	2	130	250	0	1	187	0	3.5	
2	41	0	1	130	204	0	0	172	0	1.4	
3	56	1	1	120	236	0	1	178	0	0.8	
4	57	0	0	120	354	0	1	163	1	0.6	
5	57	1	0	140	192	0	1	148	0	0.4	
6	56	0	1	140	294	0	0	153	0	1.3	
7	44	1	1	120	263	0	1	173	0	0.0	
8	52	1	2	172	199	1	1	162	0	0.5	
9	57	1	2	150	168	0	1	174	0	1.6	
10	54	1	0	140	239	0	1	160	0	1.2	
11	48	0	2	130	275	0	1	139	0	0.2	
12	49	1	1	130	266	0	1	171	0	0.6	
13	64	1	3	110	211	0	0	144	1	1.8	
14	58	0	3	150	283	1	0	162	0	1.0	
15	50	0	2	120	219	0	1	158	0	1.6	
16	58	0	2	120	340	0	1	172	0	0.0	

17	66	0	3	150	226	0	1	114	0	2.6
18	43	1	0	150	247	0	1	171	0	1.5
19	69	0	3	140	239	0	1	151	0	1.8
20	59	1	0	135	234	0	1	161	0	0.5
21	44	1	2	130	233	0	1	179	1	0.4
22	42	1	0	140	226	0	1	178	0	0.0
23	61	1	2	150	243	1	1	137	1	1.0
24	40	1	3	140	199	0	1	178	1	1.4
25	71	0	1	160	302	0	1	162	0	0.4
26	59	1	2	150	212	1	1	157	0	1.6
27	51	1	2	110	175	0	1	123	0	0.6
28	65	0	2	140	417	1	0	157	0	0.8
29	53	1	2	130	197	1	0	152	0	1.2
..	...	...	..	...	...	...	...	...	...	...
273	58	1	0	100	234	0	1	156	0	0.1
274	47	1	0	110	275	0	0	118	1	1.0
275	52	1	0	125	212	0	1	168	0	1.0
276	58	1	0	146	218	0	1	105	0	2.0
277	57	1	1	124	261	0	1	141	0	0.3
278	58	0	1	136	319	1	0	152	0	0.0
279	61	1	0	138	166	0	0	125	1	3.6
280	42	1	0	136	315	0	1	125	1	1.8
281	52	1	0	128	204	1	1	156	1	1.0
282	59	1	2	126	218	1	1	134	0	2.2
283	40	1	0	152	223	0	1	181	0	0.0
284	61	1	0	140	207	0	0	138	1	1.9
285	46	1	0	140	311	0	1	120	1	1.8
286	59	1	3	134	204	0	1	162	0	0.8
287	57	1	1	154	232	0	0	164	0	0.0
288	57	1	0	110	335	0	1	143	1	3.0
289	55	0	0	128	205	0	2	130	1	2.0
290	61	1	0	148	203	0	1	161	0	0.0
291	58	1	0	114	318	0	2	140	0	4.4
292	58	0	0	170	225	1	0	146	1	2.8
293	67	1	2	152	212	0	0	150	0	0.8
294	44	1	0	120	169	0	1	144	1	2.8
295	63	1	0	140	187	0	0	144	1	4.0
296	63	0	0	124	197	0	1	136	1	0.0
297	59	1	0	164	176	1	0	90	0	1.0
298	57	0	0	140	241	0	1	123	1	0.2
299	45	1	3	110	264	0	1	132	0	1.2
300	68	1	0	144	193	1	1	141	0	3.4
301	57	1	0	130	131	0	1	115	1	1.2
302	57	0	1	130	236	0	0	174	0	0.0

	slope	ca	thal	target
0	0	0	1	1
1	0	0	2	1

2	2	0	2	1
3	2	0	2	1
4	2	0	2	1
5	1	0	1	1
6	1	0	2	1
7	2	0	3	1
8	2	0	3	1
9	2	0	2	1
10	2	0	2	1
11	2	0	2	1
12	2	0	2	1
13	1	0	2	1
14	2	0	2	1
15	1	0	2	1
16	2	0	2	1
17	0	0	2	1
18	2	0	2	1
19	2	2	2	1
20	1	0	3	1
21	2	0	2	1
22	2	0	2	1
23	1	0	2	1
24	2	0	3	1
25	2	2	2	1
26	2	0	2	1
27	2	0	2	1
28	2	1	2	1
29	0	0	2	1
..	...	..	...	...
273	2	1	3	0
274	1	1	2	0
275	2	2	3	0
276	1	1	3	0
277	2	0	3	0
278	2	2	2	0
279	1	1	2	0
280	1	0	1	0
281	1	0	0	0
282	1	1	1	0
283	2	0	3	0
284	2	1	3	0
285	1	2	3	0
286	2	2	2	0
287	2	1	2	0
288	1	1	3	0
289	1	1	3	0
290	2	1	3	0
291	0	3	1	0

```
292      1   2     1       0
293      1   0     3       0
294      0   0     1       0
295      2   2     3       0
296      1   0     2       0
297      1   2     1       0
298      1   0     3       0
299      1   0     3       0
300      1   2     3       0
301      1   1     3       0
302      1   1     2       0

[303 rows x 14 columns]
```

```
In [80]: #just an example to check
         heart_disease.loc[:,"cp"]
```

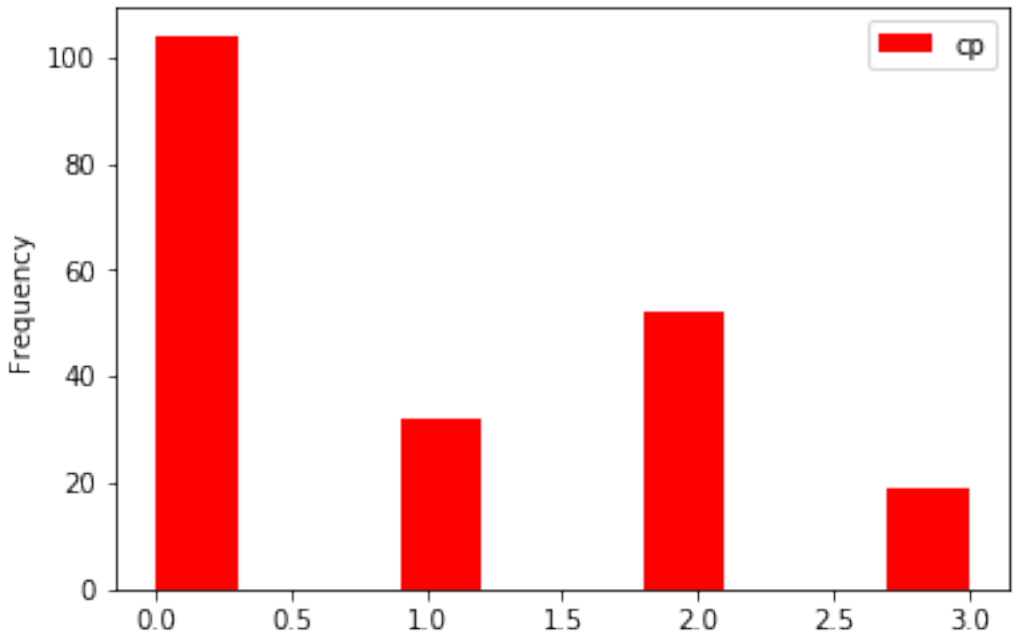
```
Out[80]: 0      3
         1      2
         2      1
         3      1
         4      0
         5      0
         6      1
         7      1
         8      2
         9      2
         10     0
         11     2
         12     1
         13     3
         14     3
         15     2
         16     2
         17     3
         18     0
         19     3
         20     0
         21     2
         22     0
         23     2
         24     3
         25     1
         26     2
         27     2
         28     2
         29     2
               ..
```

```
273    0
274    0
275    0
276    0
277    1
278    1
279    0
280    0
281    0
282    2
283    0
284    0
285    0
286    3
287    1
288    0
289    0
290    0
291    0
292    0
293    2
294    0
295    0
296    0
297    0
298    0
299    3
300    0
301    0
302    1
Name: cp, Length: 303, dtype: int64
```

Data Visualization one: Relationship between male and chest pain type

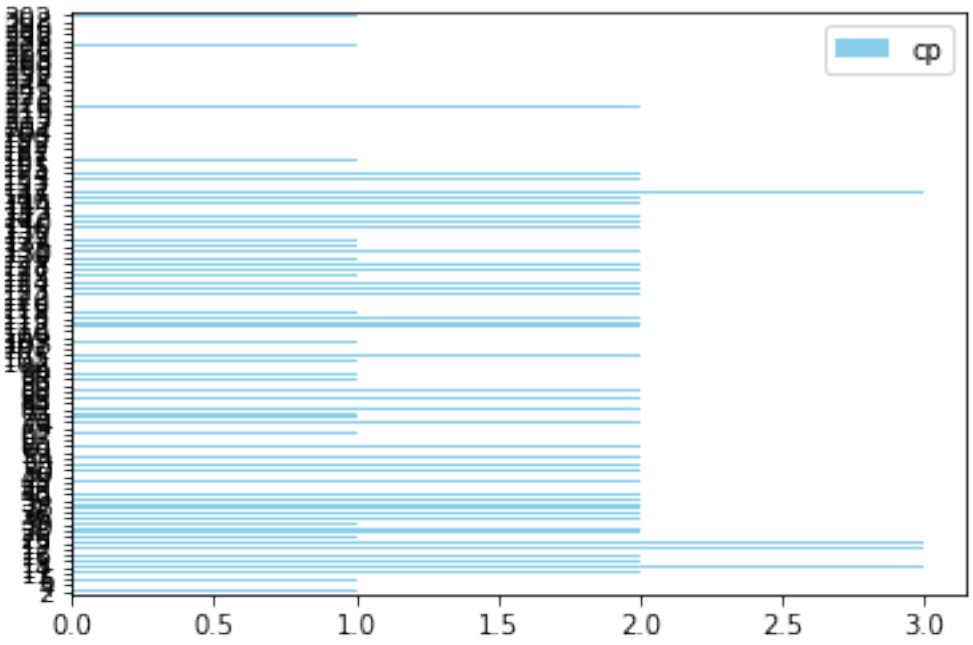
cp: chest pain type -- Value 1: typical angina -- Value 2: atypical angina -- Value 3: non-anginal pain -- Value 4: asymptomatic

```
In [69]: #extract the necessary two columns that we need to use
         visual_one_data=heart_disease[["sex","cp"]]
         #visual_one_grouped_data=visual_one_data.groupby("sex")
         cleaned_data_for_male=visual_one_data.loc[visual_one_data['sex'] ==1]
         cleaned_data_for_male=cleaned_data_for_male[["cp"]]
         cleaned_data_for_male.plot.hist(color="red")

Out[69]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1138afa320>
```



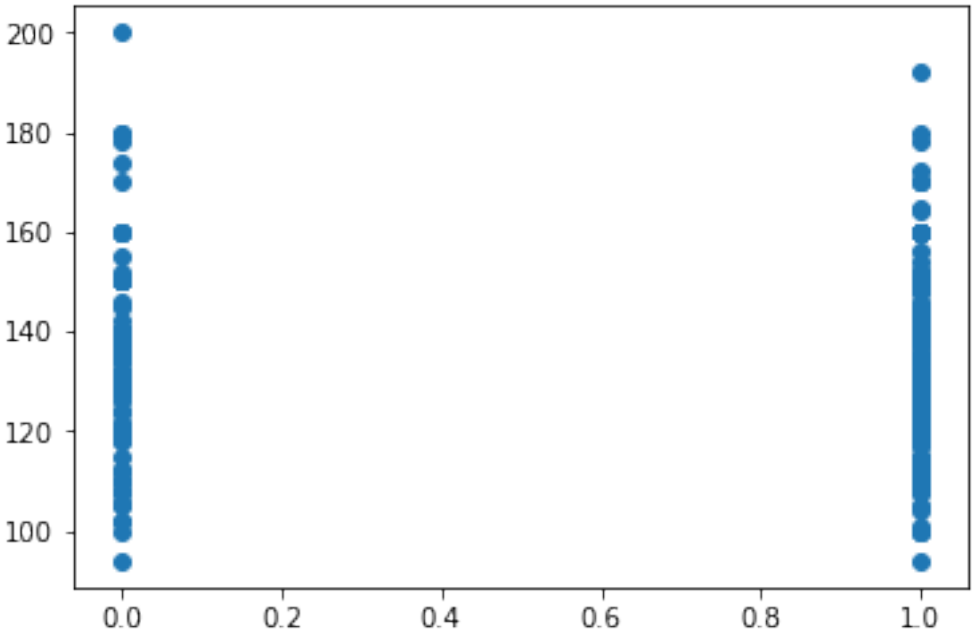
2 Data Visualization two:Relationship between female,male and chest pain type

```
In [70]: cleaned_data_for_female=visual_one_data.loc[visual_one_data['sex'] ==0]
         cleaned_data_for_female=cleaned_data_for_female[["cp"]]
         cleaned_data_for_female.plot.barh(color="skyblue")

Out[70]: <matplotlib.axes._subplots.AxesSubplot at 0x7f116984f748>
```



In [95]:
```
#extract the necessary two columns that we need to use
visual_one_data=heart_disease[["sex","trestbps"]]
#visual_one_grouped_data=visual_one_data.groupby("sex")
#cleaned_data_for_male=visual_one_data.loc[visual_one_data['sex'] ==1]
#cleaned_data_for_female=visual_one_data.loc[visual_one_data['sex'] ==0]
#final_data_for_male=cleaned_data_for_male.loc[:,"trestbps"]
#final_data_for_female=cleaned_data_for_female.loc[:,"trestbps"]

plt.scatter(heart_disease["sex"],heart_disease["trestbps"])
#plt.scatter(x="sex",y="trestbps")
#cleaned_data_for_male=cleaned_data_for_male[["restecg"]]
#cleaned_data_for_male.plt.scatter(color="red")
```

Out[95]:
```
<matplotlib.collections.PathCollection at 0x7f116940f940>
```



Dataset 2: Black Friday ___

In [ ]:

2.0.1 Background:

Link to Dataset: https://www.kaggle.com/mehdidag/black-friday/version/1

Data Preprocessing and Manipulation

In [ ]:

Data Visualization

In [ ]: