# lab2_finale

June 4, 2025

[1]:

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

[2]:

```
train = pd.read_csv('mai-ml-lab-2/train.csv')
test  = pd.read_csv('mai-ml-lab-2/test.csv')

display(train.info())
display(test.info())

print('Train duplication: ', train.duplicated().sum())
print('Test duplication: ', test.duplicated().sum())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15000 entries, 0 to 14999
Data columns (total 24 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   id                   15000 non-null  int64
 1   age                  15000 non-null  float64
 2   height(cm)           15000 non-null  float64
 3   weight(kg)           15000 non-null  float64
 4   waist(cm)            15000 non-null  float64
 5   eyesight(left)       15000 non-null  float64
 6   eyesight(right)      15000 non-null  float64
 7   hearing(left)        15000 non-null  float64
 8   hearing(right)       15000 non-null  float64
 9   systolic             15000 non-null  float64
 10  relaxation           15000 non-null  float64
 11  fasting blood sugar  15000 non-null  float64
 12  Cholesterol          15000 non-null  float64
 13  triglyceride         15000 non-null  float64
 14  HDL                  15000 non-null  float64
 15  LDL                  15000 non-null  float64
 16  hemoglobin           15000 non-null  float64
 17  Urine protein        15000 non-null  float64
 18  serum creatinine     15000 non-null  float64
 19  AST                  15000 non-null  float64
 20  ALT                  15000 non-null  float64
 21  Gtp                  15000 non-null  float64
 22  dental caries        15000 non-null  float64
 23  smoking              15000 non-null  float64
dtypes: float64(23), int64(1)
memory usage: 2.7 MB

None

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 23 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   id                   10000 non-null  int64
 1   age                  10000 non-null  float64
 2   height(cm)           10000 non-null  float64
 3   weight(kg)           10000 non-null  float64
 4   waist(cm)            10000 non-null  float64
 5   eyesight(left)       10000 non-null  float64
 6   eyesight(right)      10000 non-null  float64
 7   hearing(left)        10000 non-null  float64
 8   hearing(right)       10000 non-null  float64
 9   systolic             10000 non-null  float64
 10  relaxation           10000 non-null  float64
 11  fasting blood sugar  10000 non-null  float64
 12  Cholesterol          10000 non-null  float64
 13  triglyceride         10000 non-null  float64
 14  HDL                  10000 non-null  float64
 15  LDL                  10000 non-null  float64
 16  hemoglobin           10000 non-null  float64
 17  Urine protein        10000 non-null  float64
 18  serum creatinine     10000 non-null  float64
 19  AST                  10000 non-null  float64
 20  ALT                  10000 non-null  float64
 21  Gtp                  10000 non-null  float64
 22  dental caries        10000 non-null  float64
dtypes: float64(22), int64(1)
memory usage: 1.8 MB

None

Train duplication:  0
Test duplication:  0
```

[3]:

```python
print(f"Train shape: {train.shape}")
print(f"Test  shape: {test.shape}")
print("Columns in train:", train.columns.tolist())
```

```
Train shape: (15000, 24)
Test  shape: (10000, 23)
Columns in train: ['id', 'age', 'height(cm)', 'weight(kg)', 'waist(cm)',
'eyesight(left)', 'eyesight(right)', 'hearing(left)', 'hearing(right)',
'systolic', 'relaxation', 'fasting blood sugar', 'Cholesterol', 'triglyceride',
'HDL', 'LDL', 'hemoglobin', 'Urine protein', 'serum creatinine', 'AST', 'ALT',
'Gtp', 'dental caries', 'smoking']
```

```
[4]: features = list(set(train.columns) - {'id', 'smoking'})

     df_all = pd.concat([train[features], test[features]], axis=0)
```

```
[5]: print('Duplicated rows between train and test datasets: ', df_all.duplicated().
       ↪sum())
```

```
Duplicated rows between train and test datasets:  0
```

```
[6]: display(train.head(5))
     train.info()
     display(train.describe())
```

```
   id   age  height(cm)  weight(kg)  waist(cm)  eyesight(left)  \
0   0  55.0       150.0        50.0       73.0             0.8
1   1  50.0       160.0        60.0       79.6             1.0
2   2  55.0       145.0        50.0       73.0             1.2
3   3  40.0       175.0        80.0       83.2             0.9
4   4  30.0       170.0        70.0       79.5             1.2

   eyesight(right)  hearing(left)  hearing(right)  systolic  …   HDL    LDL  \
0              1.0            1.0             1.0     122.0  …  62.0  113.0
1              0.8            1.0             1.0     124.0  …  74.0  112.0
2              1.2            1.0             1.0     110.0  …  66.0  123.0
3              1.0            1.0             1.0      99.0  …  38.0  114.0
4              1.2            1.0             1.0     119.0  …  49.0  148.0

   hemoglobin  Urine protein  serum creatinine   AST   ALT   Gtp  \
0        12.0            1.0               0.6  22.0  21.0  12.0
1        14.1            1.0               0.9  19.0  14.0  17.0
2        13.3            1.0               0.6  15.0  12.0  13.0
3        16.1            1.0               1.0  22.0  13.0  20.0
4        15.5            1.0               1.2  18.0  19.0  25.0

   dental caries  smoking
0            0.0      0.0
1            0.0      0.0
2            0.0      0.0
3            0.0      1.0
4            0.0      1.0
```

```
[5 rows x 24 columns]
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15000 entries, 0 to 14999
Data columns (total 24 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   id                   15000 non-null  int64
 1   age                  15000 non-null  float64
 2   height(cm)           15000 non-null  float64
 3   weight(kg)           15000 non-null  float64
 4   waist(cm)            15000 non-null  float64
 5   eyesight(left)       15000 non-null  float64
 6   eyesight(right)      15000 non-null  float64
 7   hearing(left)        15000 non-null  float64
 8   hearing(right)       15000 non-null  float64
 9   systolic             15000 non-null  float64
 10  relaxation           15000 non-null  float64
 11  fasting blood sugar  15000 non-null  float64
 12  Cholesterol          15000 non-null  float64
 13  triglyceride         15000 non-null  float64
 14  HDL                  15000 non-null  float64
 15  LDL                  15000 non-null  float64
 16  hemoglobin           15000 non-null  float64
 17  Urine protein        15000 non-null  float64
 18  serum creatinine     15000 non-null  float64
 19  AST                  15000 non-null  float64
 20  ALT                  15000 non-null  float64
 21  Gtp                  15000 non-null  float64
 22  dental caries        15000 non-null  float64
 23  smoking              15000 non-null  float64
dtypes: float64(23), int64(1)
memory usage: 2.7 MB
```

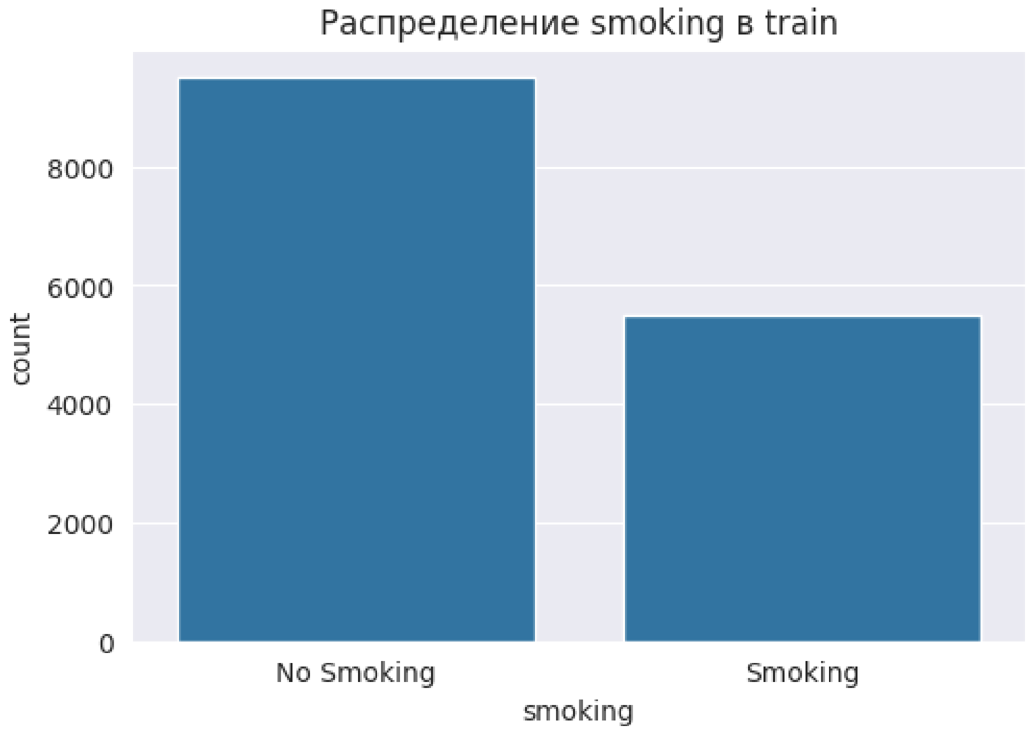
```
                 id           age    height(cm)    weight(kg)     waist(cm)  \
count  15000.000000  15000.000000  15000.000000  15000.000000  15000.000000
mean    7499.500000     42.606000    164.672667     64.282000     80.323180
std     4330.271354     10.415341      8.558684     10.973871      7.932701
min        0.000000     20.000000    140.000000     35.000000      0.900000
25%     3749.750000     40.000000    160.000000     55.000000     75.000000
50%     7499.500000     40.000000    165.000000     65.000000     80.000000
75%    11249.250000     50.000000    170.000000     70.000000     86.000000
max    14999.000000     80.000000    190.000000    120.000000    121.000000

       eyesight(left)  eyesight(right)  hearing(left)  hearing(right)  \
count    15000.000000     15000.000000   15000.000000    15000.000000
mean         1.032727         1.025927       1.005533        1.004933
std          0.312845         0.313488       0.074183        0.070067
min          0.100000         0.100000       1.000000        1.000000
```

```
25%          0.800000         0.800000       1.000000        1.000000
50%          1.000000         1.000000       1.000000        1.000000
75%          1.200000         1.200000       1.000000        1.000000
max          9.900000         9.900000       2.000000        2.000000

           systolic  …           HDL           LDL    hemoglobin  \
count  15000.000000  …  15000.000000  15000.000000  15000.000000
mean     119.149467  …     57.535667    113.158533     14.510340
std       11.167046  …     11.380739     18.442012      1.392921
min       82.000000  …     28.000000     41.000000      7.200000
25%      110.000000  …     49.000000    100.000000     13.500000
50%      119.000000  …     56.000000    112.000000     14.700000
75%      128.000000  …     65.000000    126.000000     15.600000
max      190.000000  …    105.000000    193.000000     28.900000

       Urine protein  serum creatinine           AST           ALT  \
count   15000.000000      15000.000000  15000.000000  15000.000000
mean        1.016533          0.866580     22.272667     20.874933
std         0.157464          0.168492      6.324422     10.314009
min         1.000000          0.100000      9.000000      4.000000
25%         1.000000          0.800000     18.000000     14.000000
50%         1.000000          0.900000     21.000000     18.000000
75%         1.000000          1.000000     25.000000     24.000000
max         5.000000          1.600000    324.000000    190.000000

                Gtp  dental caries       smoking
count  15000.000000   15000.000000  15000.000000
mean      26.290200       0.139467      0.366600
std       20.431882       0.346444      0.481892
min        6.000000       0.000000      0.000000
25%       15.000000       0.000000      0.000000
50%       21.000000       0.000000      0.000000
75%       31.000000       0.000000      1.000000
max      566.000000       1.000000      1.000000

[8 rows x 24 columns]
```

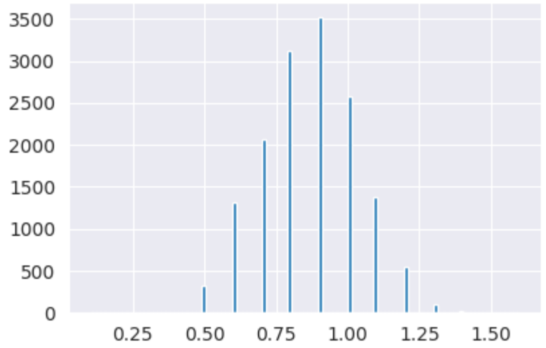
[7]:
```python
missing = train.isnull().sum().sort_values(ascending=False)
print("       train:")
print(missing[missing > 0])
```

```
       train:
Series([], dtype: int64)
```

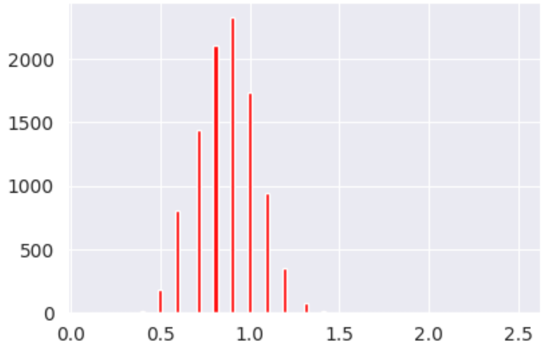
[8]:
```python
plt.figure(figsize=(6,4))
sns.countplot(x='smoking', data=train)
plt.title('        smoking   train')
plt.xticks(ticks=[0, 1], labels=['No Smoking', 'Smoking'])
plt.ylabel('count')
plt.show()
```



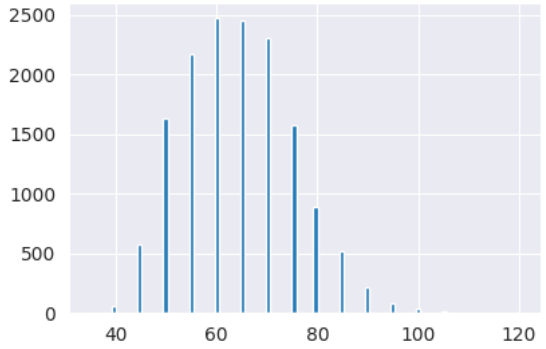
[9]:
```
def features_histogram(train, test):
    for feature in features:
        plt.figure(figsize=(10, 3))
        plt.subplot(1, 2, 1)
        train[feature].hist(bins=100)
        plt.title(f"Histogram of {feature} in train")
        plt.subplot(1, 2, 2)
        test[feature].hist(bins=100, color='r')
        plt.title(f"Histogram of {feature} in test")
        plt.show()
features_histogram(train, test)
```



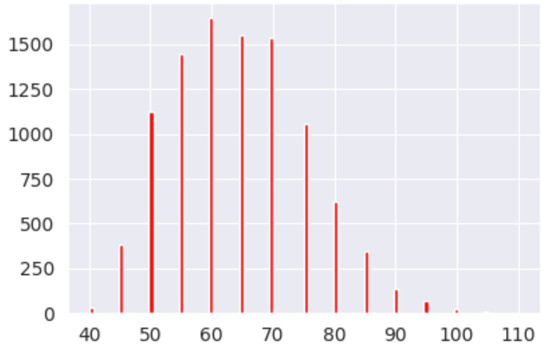
Histogram of serum creatinine in train



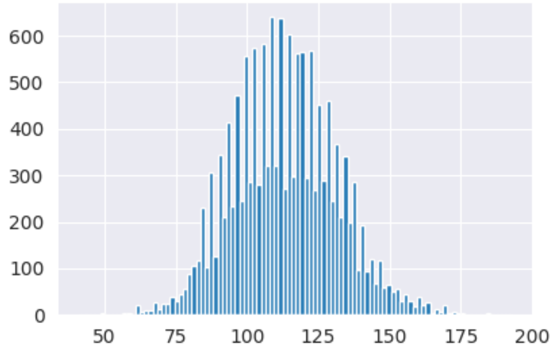
Histogram of serum creatinine in test



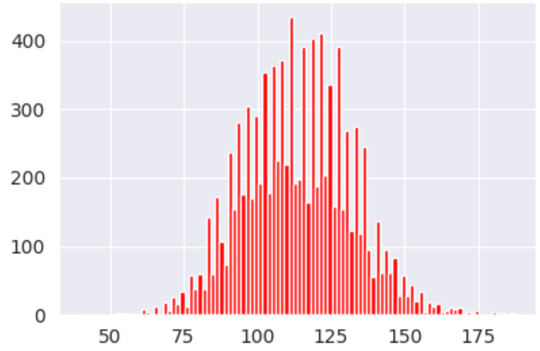
Histogram of weight(kg) in train



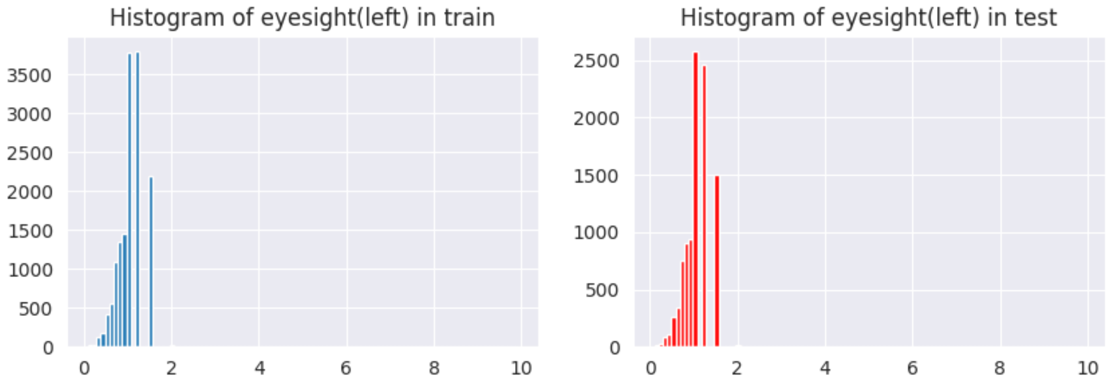
Histogram of weight(kg) in test

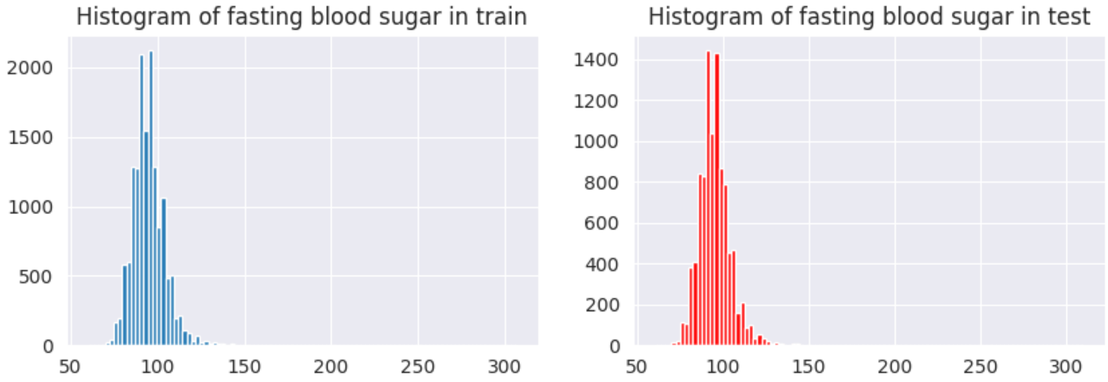


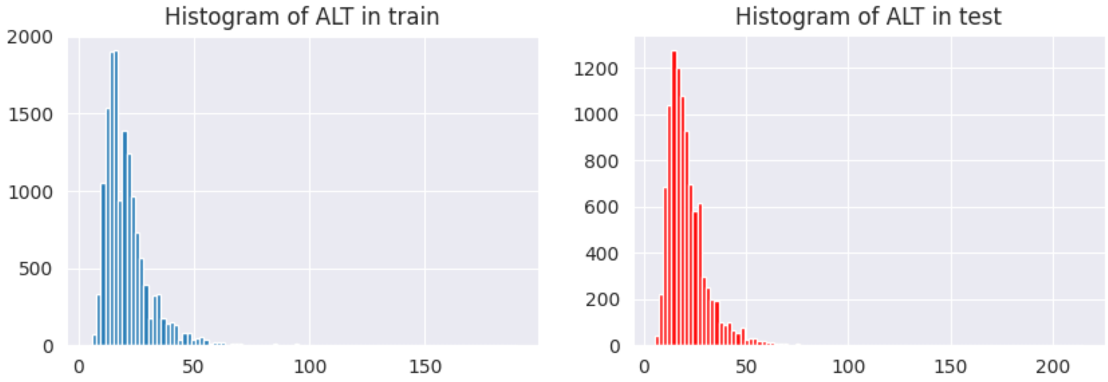
Histogram of LDL in train

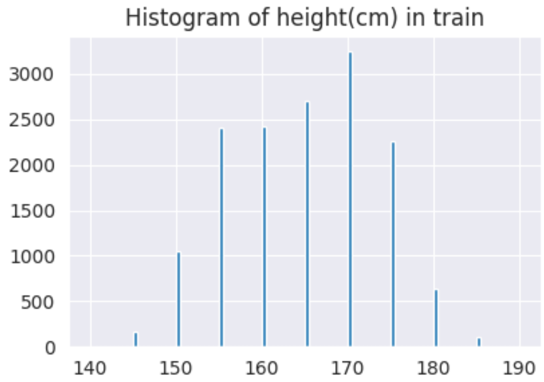


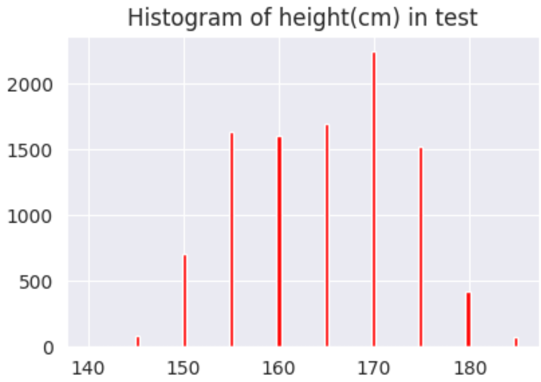
Histogram of LDL in test

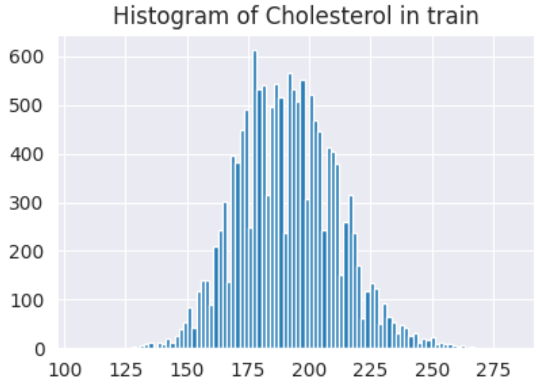

Histogram of eyesight(left) in train
0
500
1000
1500
2000
2500
3000
3500
0
2
4
6
8
10
Histogram of eyesight(left) in test
0
500
1000
1500
2000
2500
0
2
4
6
8
10

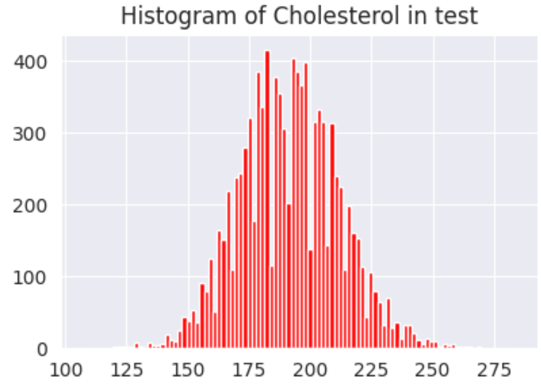

Histogram of fasting blood sugar in train
0
500
1000
1500
2000
50
100
150
200
250
300
Histogram of fasting blood sugar in test
0
200
400
600
800
1000
1200
1400
50
100
150
200
250
300

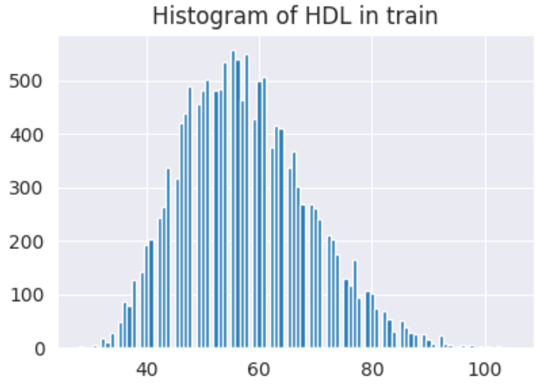

Histogram of ALT in train
0
500
1000
1500
2000
0
50
100
150
Histogram of ALT in test
0
200
400
600
800
1000
1200
0
50
100
150
200

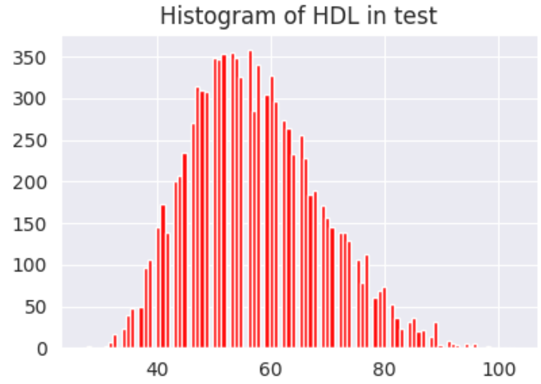

Histogram of height(cm) in train

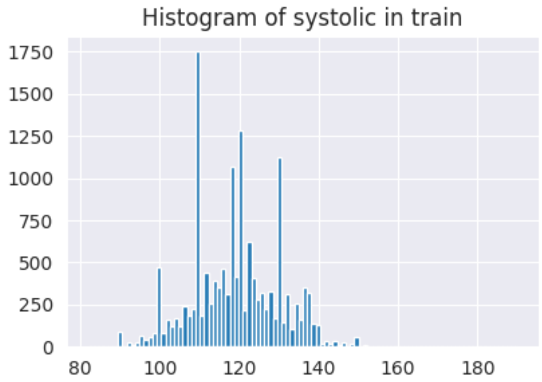

Histogram of height(cm) in test

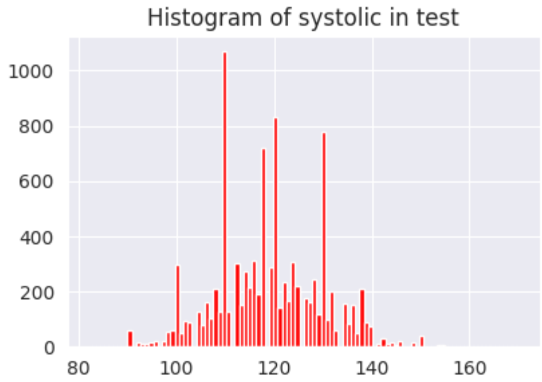

Histogram of Cholesterol in train

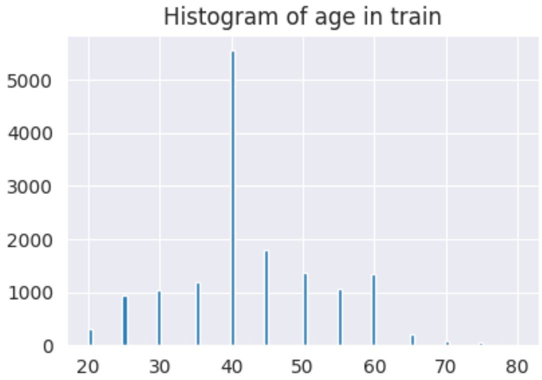

Histogram of Cholesterol in test

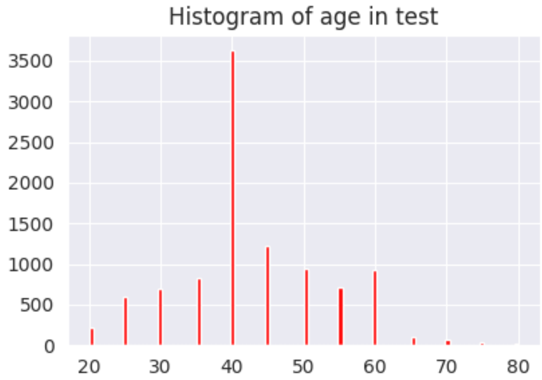

Histogram of HDL in train

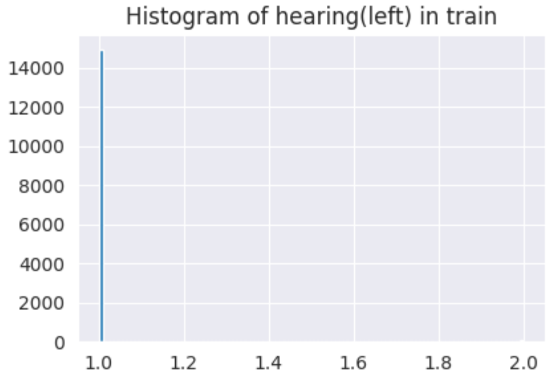

Histogram of HDL in test

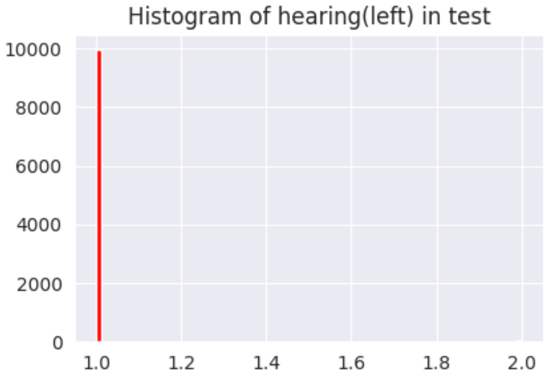

Histogram of systolic in train
1750
1500
1250
1000
750
500
250
0
80
100
120
140
160
180

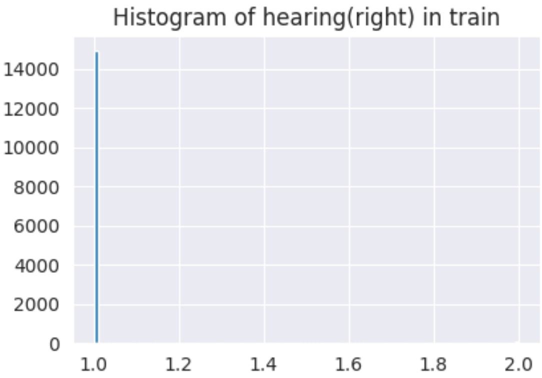

Histogram of systolic in test
1000
800
600
400
200
0
80
100
120
140
160

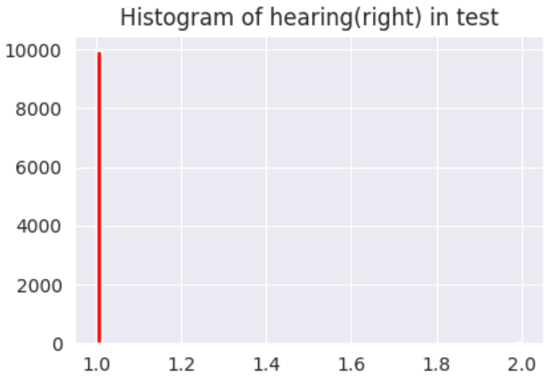

Histogram of age in train
5000
4000
3000
2000
1000
0
20
30
40
50
60
70
80

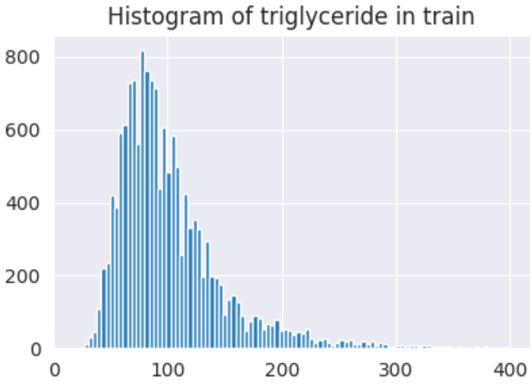

Histogram of age in test
3500
3000
2500
2000
1500
1000
500
0
20
30
40
50
60
70
80

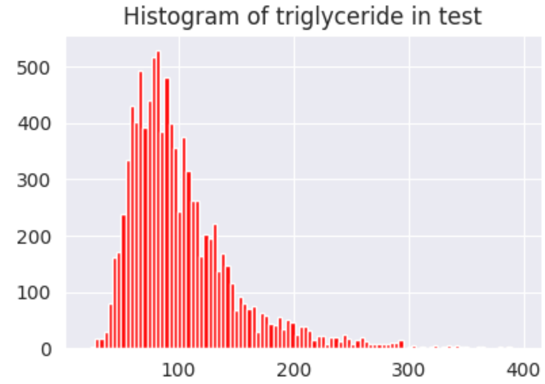

Histogram of hearing(left) in train
14000
12000
10000
8000
6000
4000
2000
0
1.0
1.2
1.4
1.6
1.8
2.0

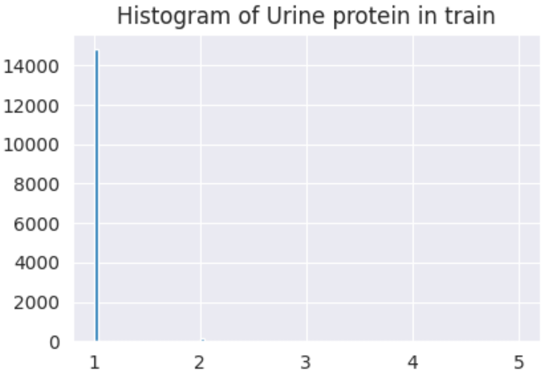

Histogram of hearing(left) in test
10000
8000
6000
4000
2000
0
1.0
1.2
1.4
1.6
1.8
2.0

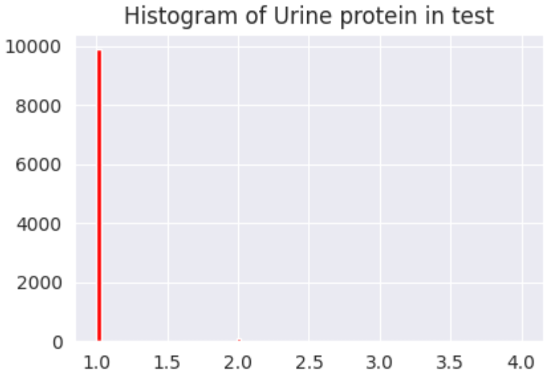

Histogram of hearing(right) in train
14000
12000
10000
8000
6000
4000
2000
0
1.0
1.2
1.4
1.6
1.8
2.0

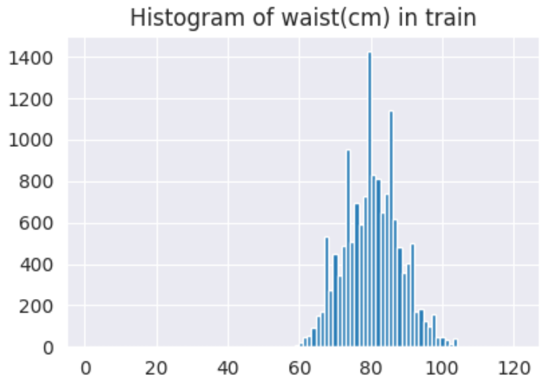

Histogram of hearing(right) in test
10000
8000
6000
4000
2000
0
1.0
1.2
1.4
1.6
1.8
2.0

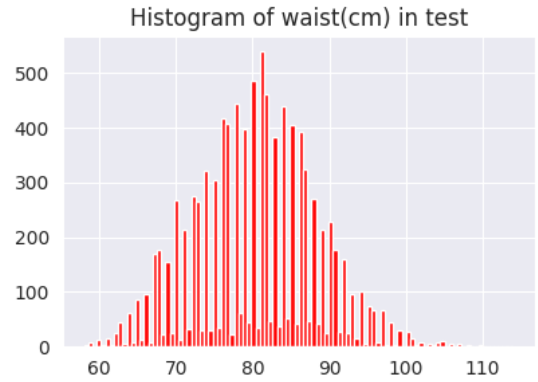

Histogram of triglyceride in train
800
600
400
200
0
0
100
200
300
400

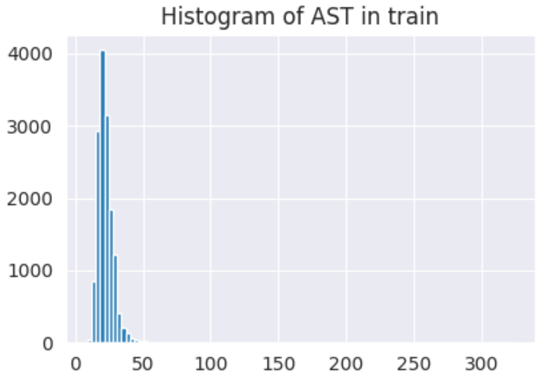

Histogram of triglyceride in test
500
400
300
200
100
0
100
200
300
400

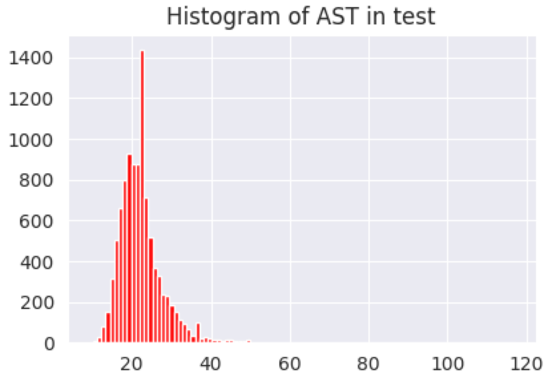

Histogram of Urine protein in train
14000
12000
10000
8000
6000
4000
2000
0
1
2
3
4
5

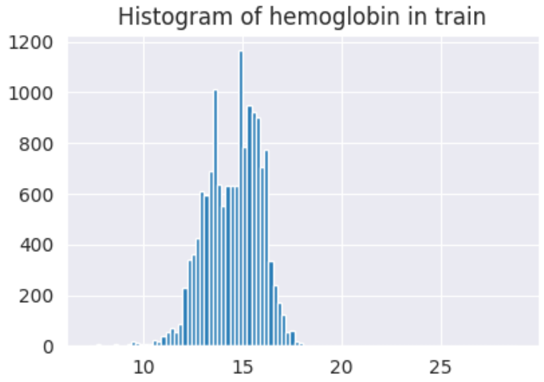

Histogram of Urine protein in test
10000
8000
6000
4000
2000
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0

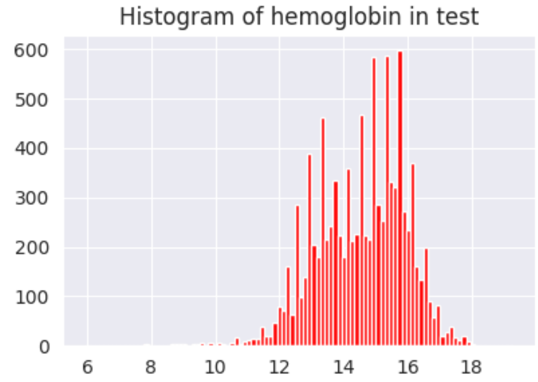

Histogram of waist(cm) in train
1400
1200
1000
800
600
400
200
0
0
20
40
60
80
100
120

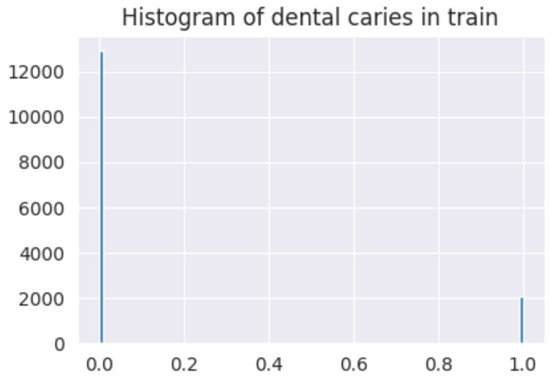

Histogram of waist(cm) in test
500
400
300
200
100
0
60
70
80
90
100
110

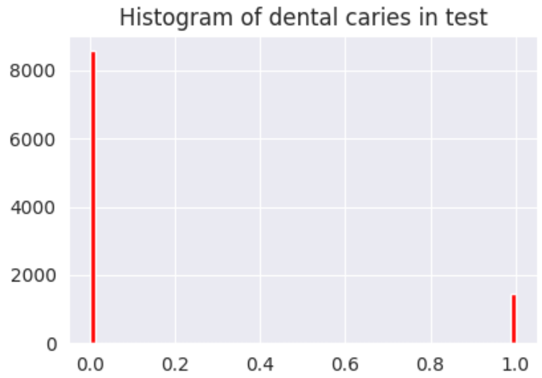

Histogram of AST in train
4000
3000
2000
1000
0
0
50
100
150
200
250
300

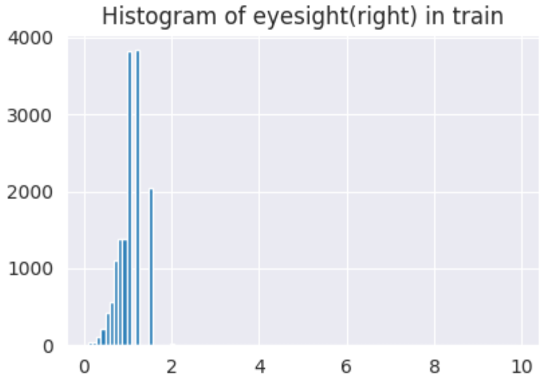

Histogram of AST in test
1400
1200
1000
800
600
400
200
0
20
40
60
80
100
120

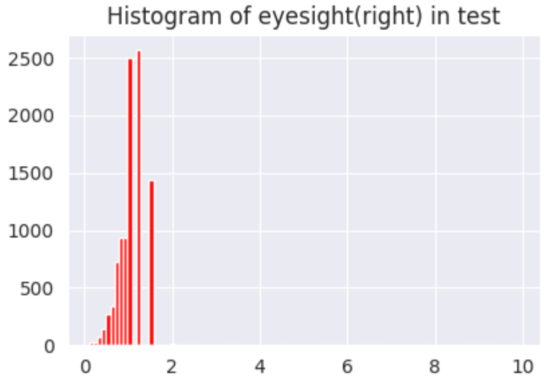

Histogram of hemoglobin in train
1200
1000
800
600
400
200
0
10
15
20
25

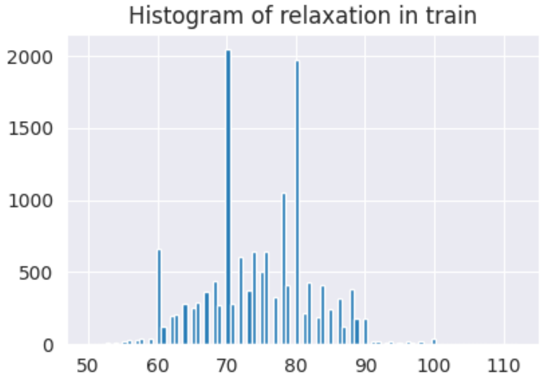

Histogram of hemoglobin in test
600
500
400
300
200
100
0
6
8
10
12
14
16
18

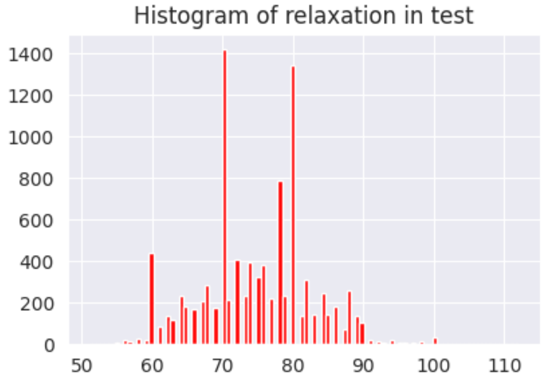

Histogram of dental caries in train
12000
10000
8000
6000
4000
2000
0
0.0
0.2
0.4
0.6
0.8
1.0

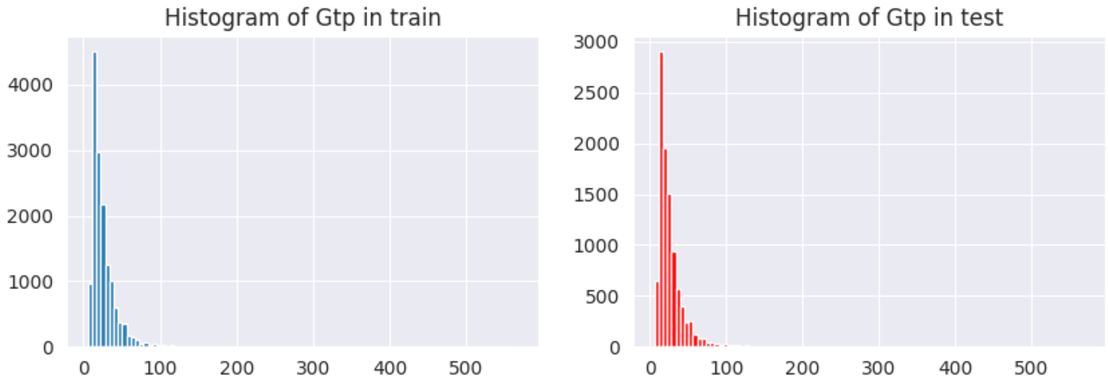

Histogram of dental caries in test
8000
6000
4000
2000
0
0.0
0.2
0.4
0.6
0.8
1.0


Histogram of eyesight(right) in train
4000
3000
2000
1000
0
0
2
4
6
8
10

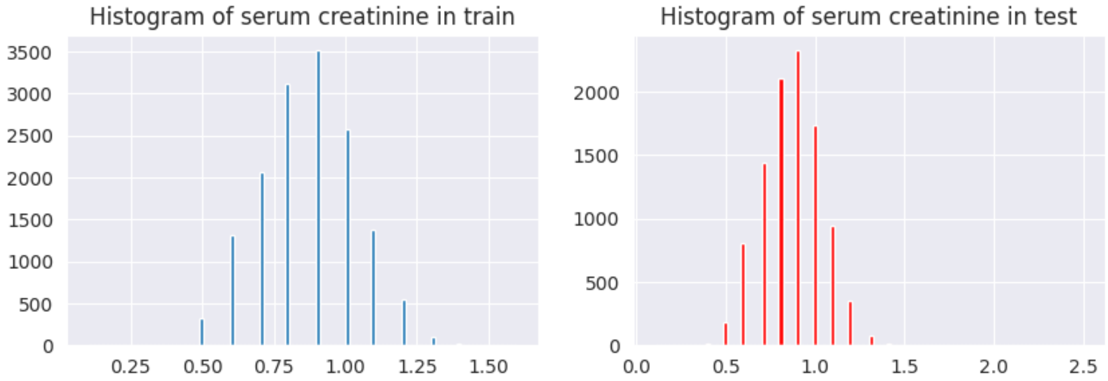

Histogram of eyesight(right) in test
2500
2000
1500
1000
500
0
0
2
4
6
8
10

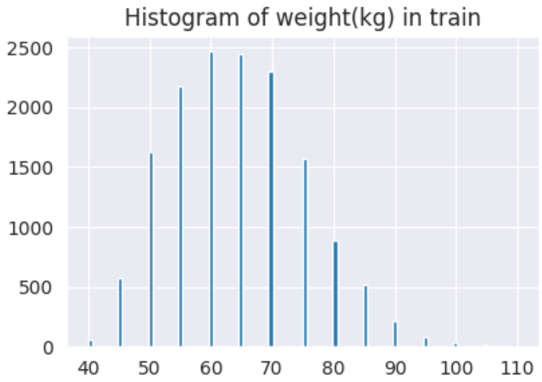

Histogram of relaxation in train
2000
1500
1000
500
0
50
60
70
80
90
100
110


Histogram of relaxation in test
1400
1200
1000
800
600
400
200
0
50
60
70
80
90
100
110

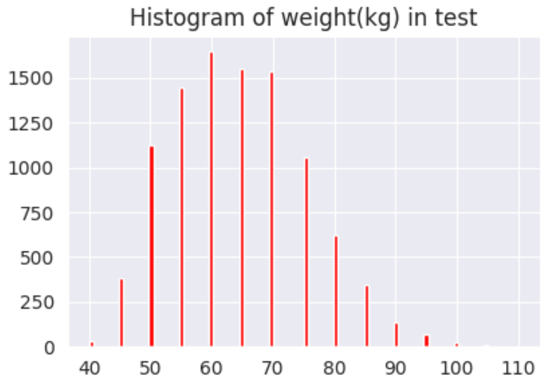


```
[10]: for feature in features:
          feature_min = test[feature].min()
          feature_max = test[feature].max()
          # adjust train data
          train = train[(train[feature] >= feature_min) & (train[feature] <=␣
       ↪feature_max)]

      train = train.reset_index(drop=True)
      features_histogram(train, test)
```

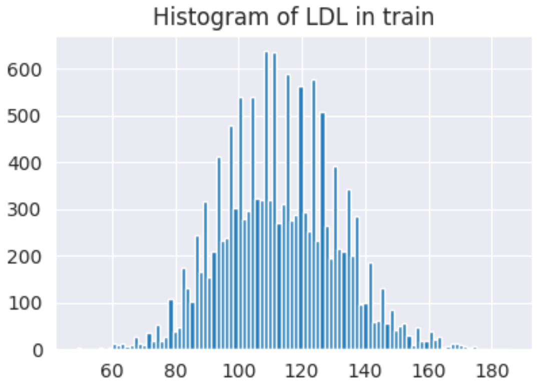




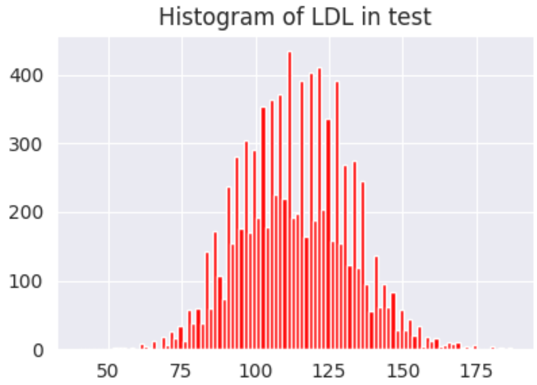
Histogram of weight(kg) in train



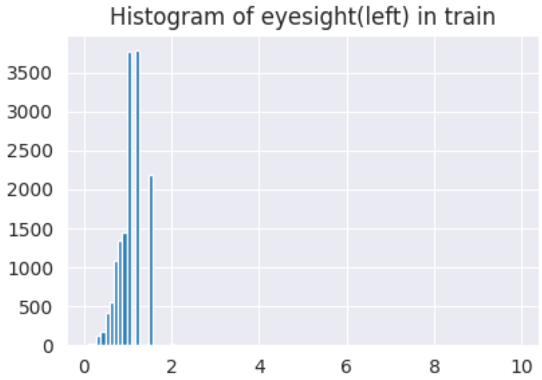
Histogram of weight(kg) in test



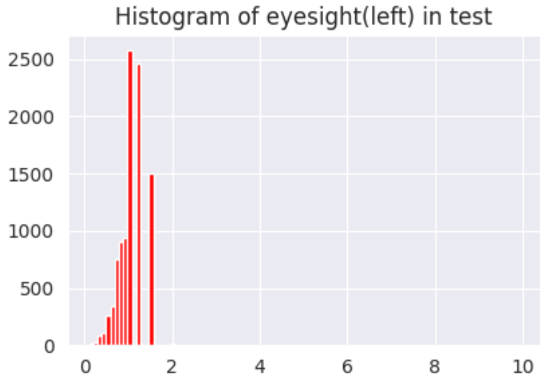
Histogram of LDL in train



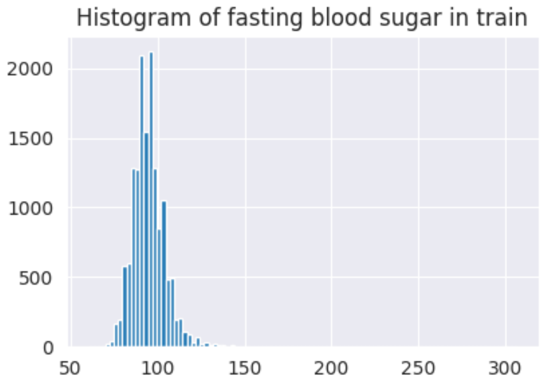
Histogram of LDL in test

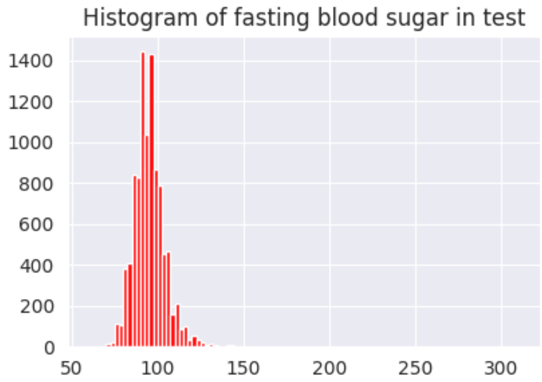


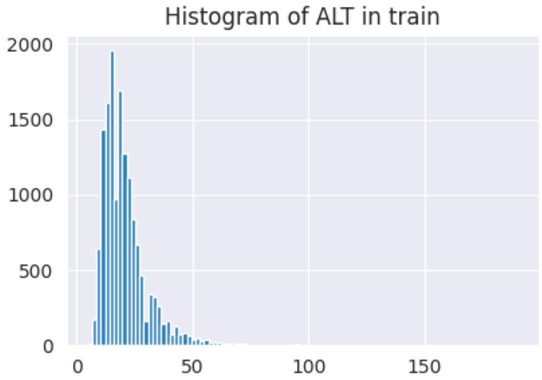
Histogram of eyesight(left) in train

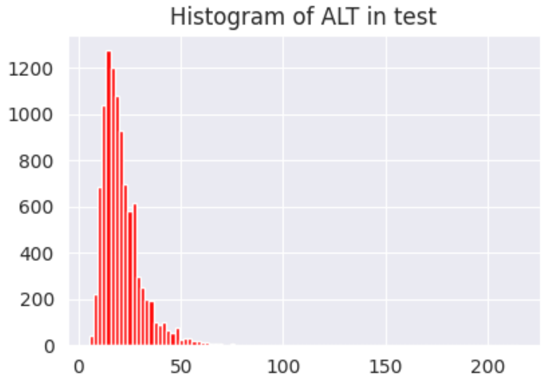


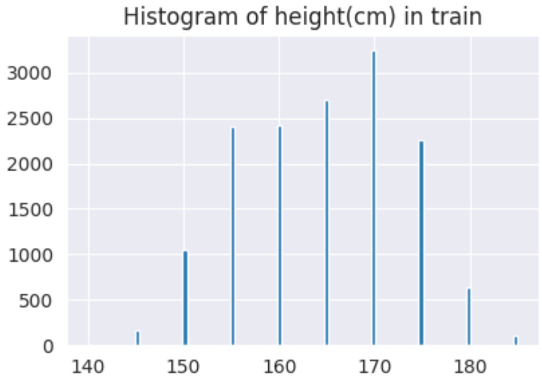
Histogram of eyesight(left) in test

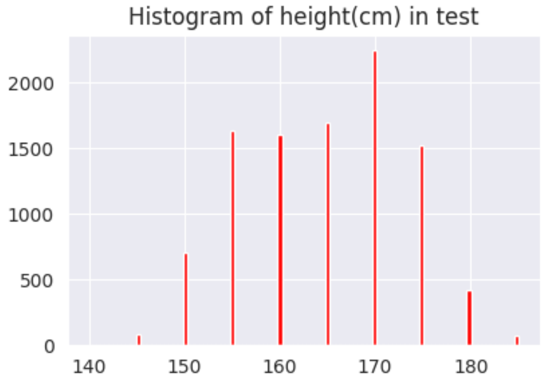

Histogram of fasting blood sugar in train

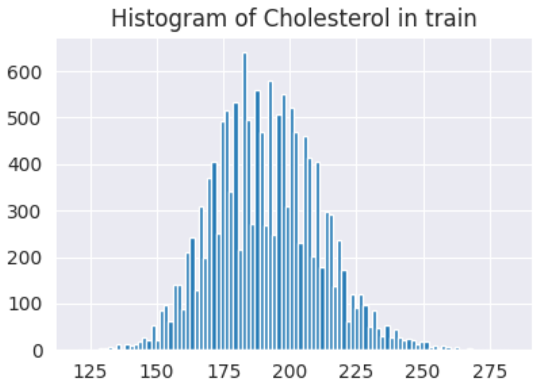

Histogram of fasting blood sugar in test

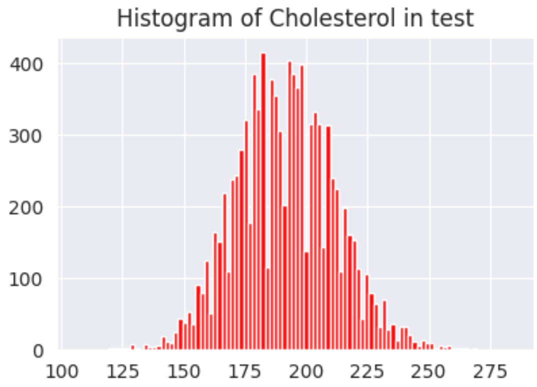

Histogram of ALT in train

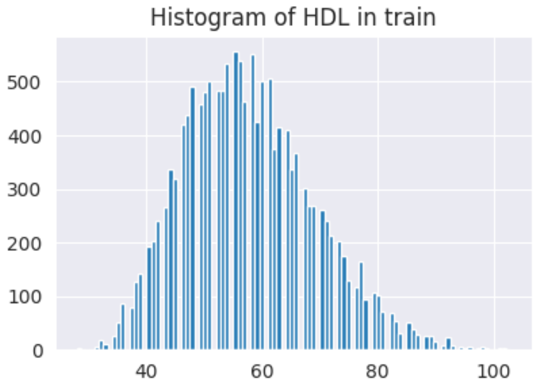

Histogram of ALT in test

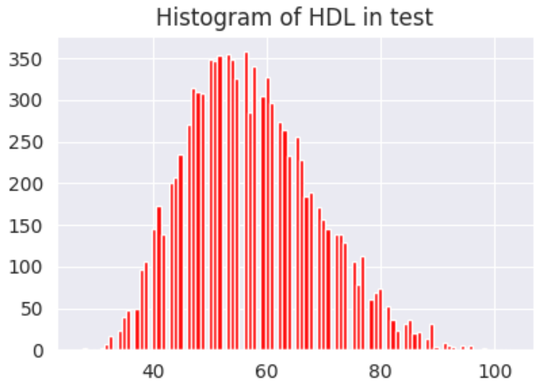

Histogram of height(cm) in train

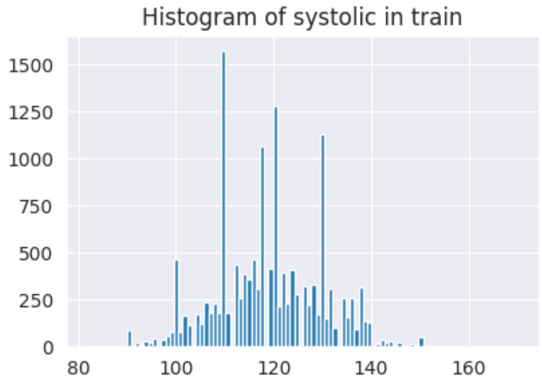

Histogram of height(cm) in test

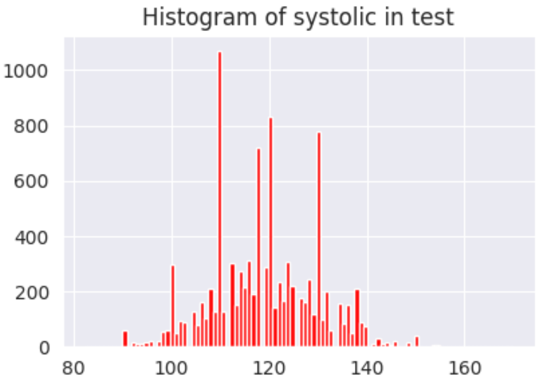

Histogram of Cholesterol in train

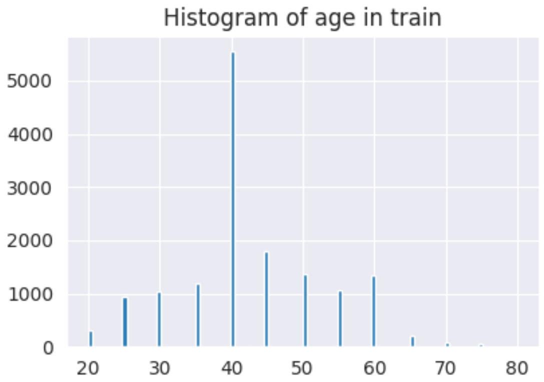

Histogram of Cholesterol in test

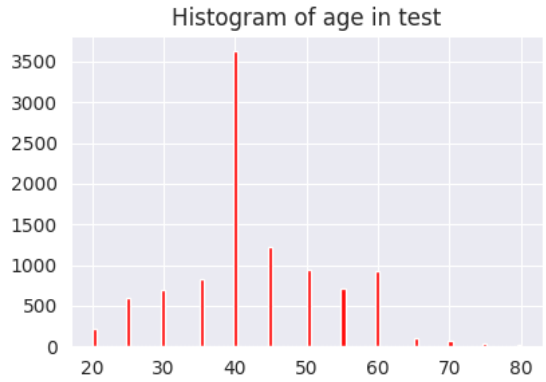

Histogram of HDL in train

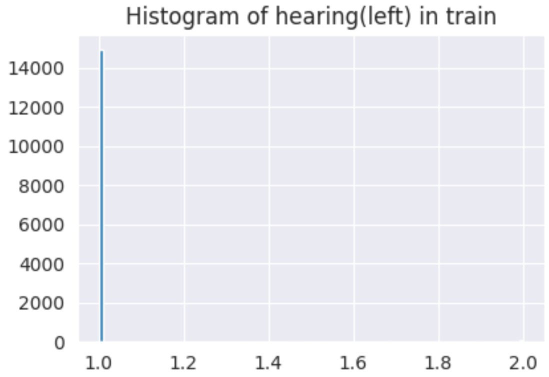

Histogram of HDL in test

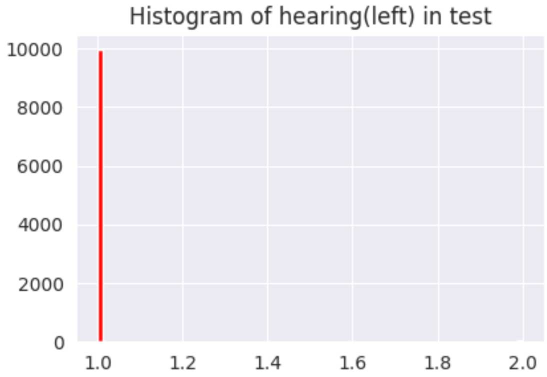

Histogram of systolic in train

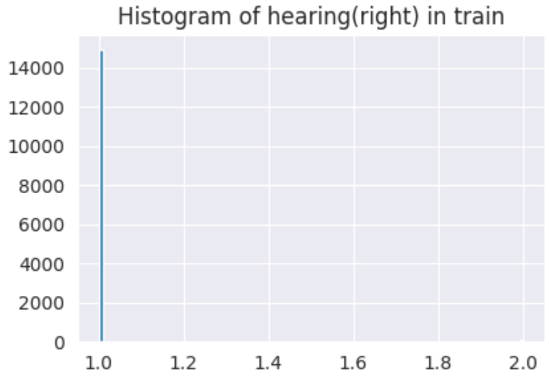

Histogram of systolic in test

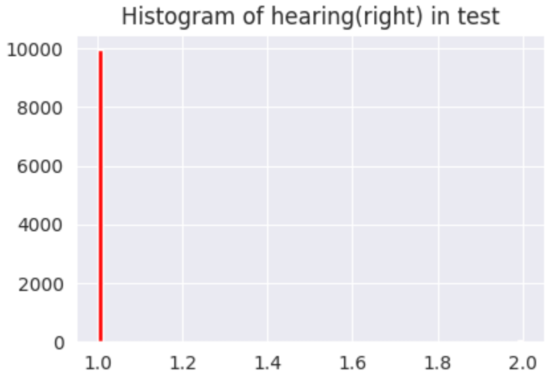

Histogram of age in train

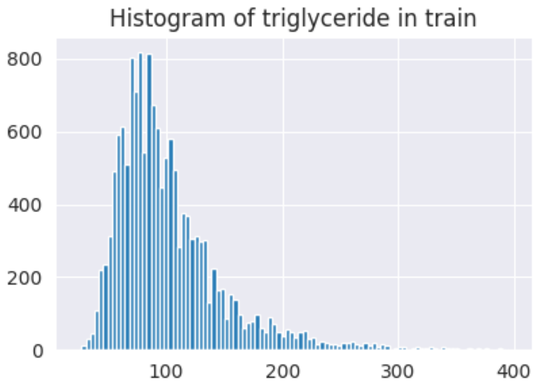

Histogram of age in test

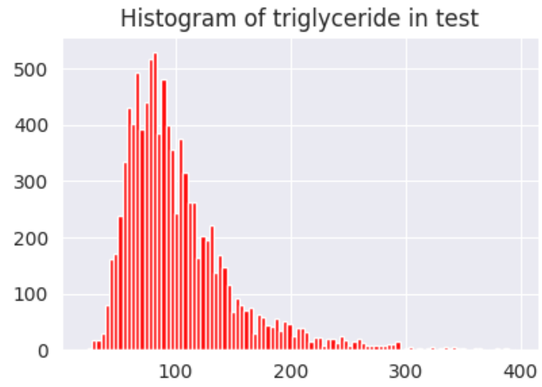

Histogram of hearing(left) in train

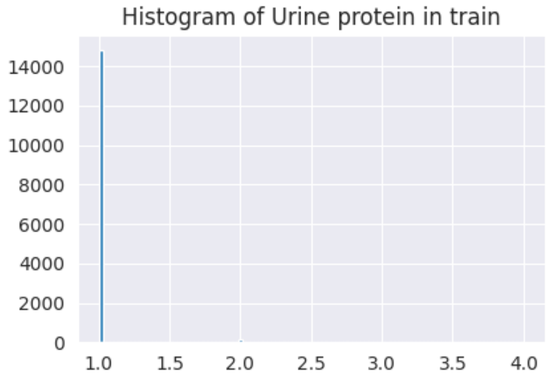

Histogram of hearing(left) in test

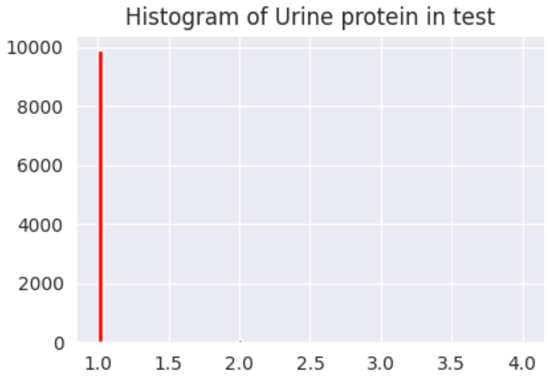

Histogram of hearing(right) in train

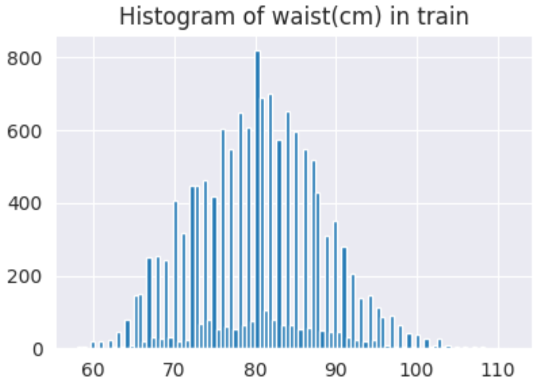

Histogram of hearing(right) in test

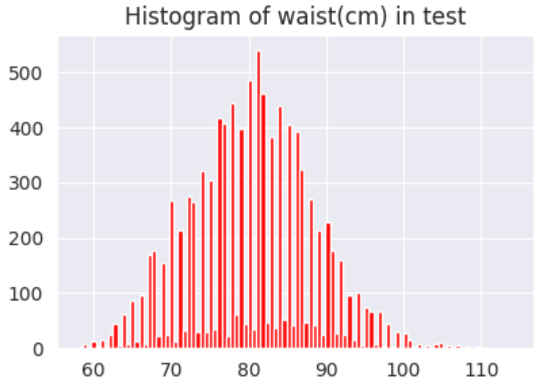

Histogram of triglyceride in train
800
600
400
200
0
100
200
300
400

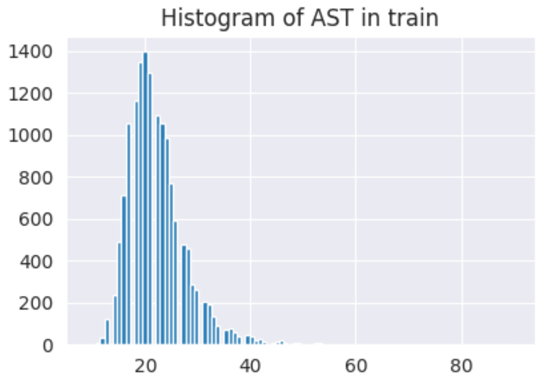

Histogram of triglyceride in test
500
400
300
200
100
0
100
200
300
400

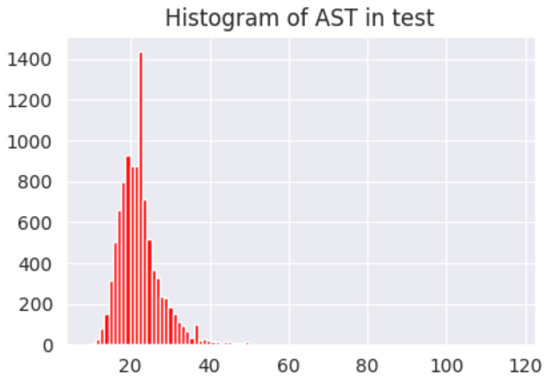

Histogram of Urine protein in train
14000
12000
10000
8000
6000
4000
2000
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0

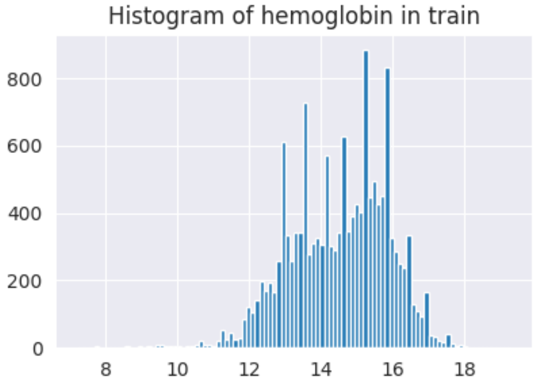

Histogram of Urine protein in test
10000
8000
6000
4000
2000
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0

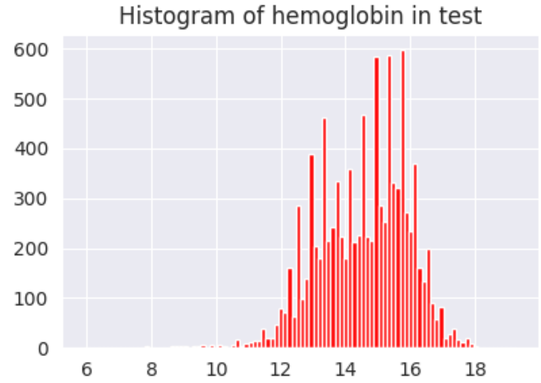

Histogram of waist(cm) in train
800
600
400
200
0
60
70
80
90
100
110

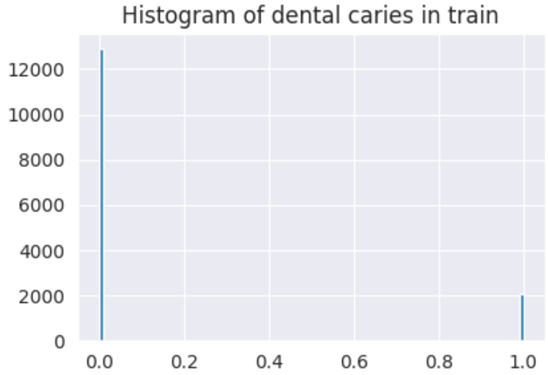

Histogram of waist(cm) in test
500
400
300
200
100
0
60
70
80
90
100
110

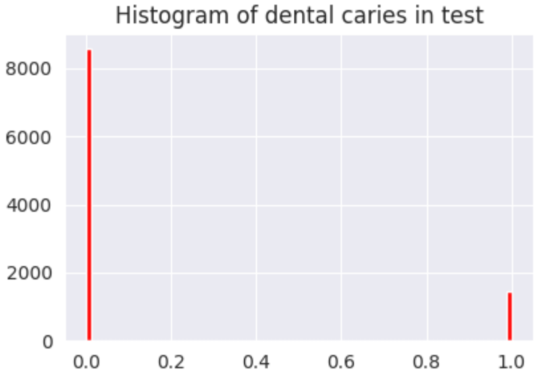

Histogram of AST in train

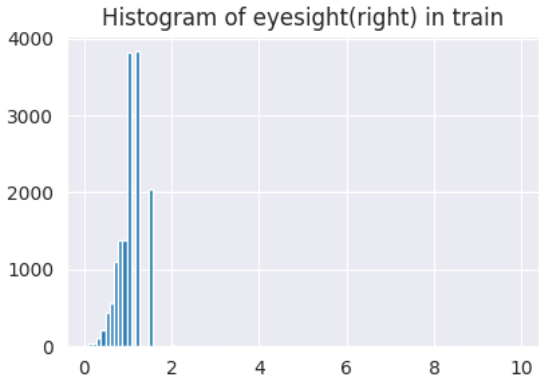

Histogram of AST in test

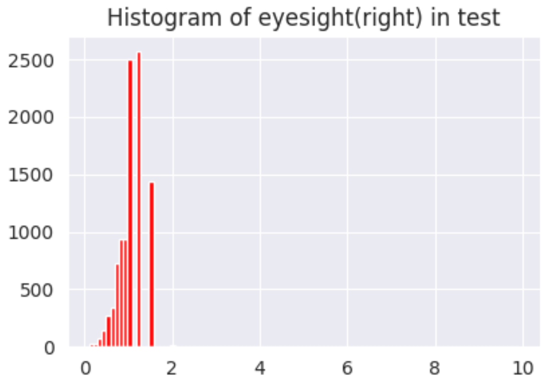

Histogram of hemoglobin in train

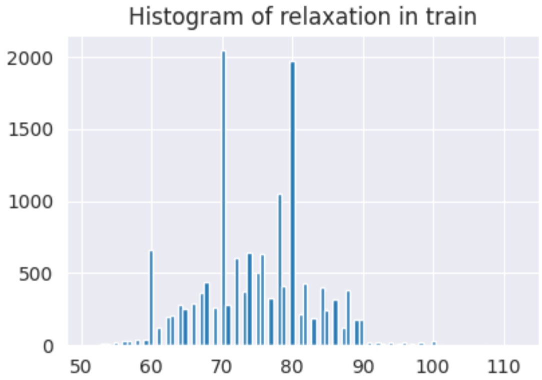

Histogram of hemoglobin in test

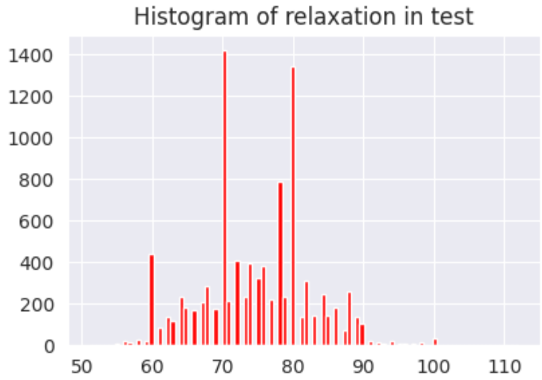

Histogram of dental caries in train

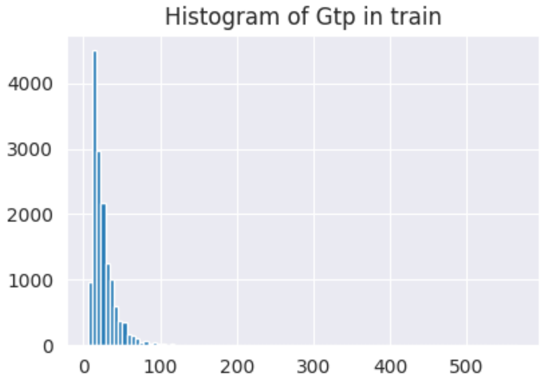

Histogram of dental caries in test

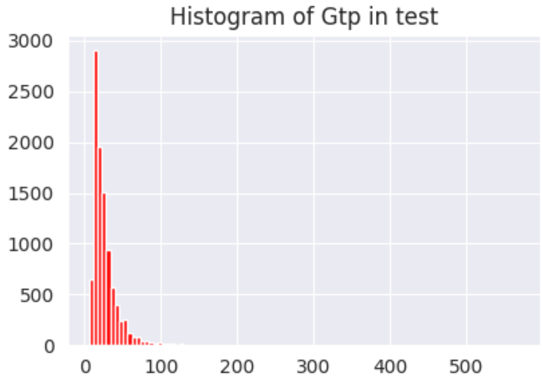

Histogram of eyesight(right) in train
4000
3000
2000
1000
0
0
2
4
6
8
10


Histogram of eyesight(right) in test
2500
2000
1500
1000
500
0
0
2
4
6
8
10


Histogram of relaxation in train
2000
1500
1000
500
0
50
60
70
80
90
100
110


Histogram of relaxation in test
1400
1200
1000
800
600
400
200
0
50
60
70
80
90
100
110

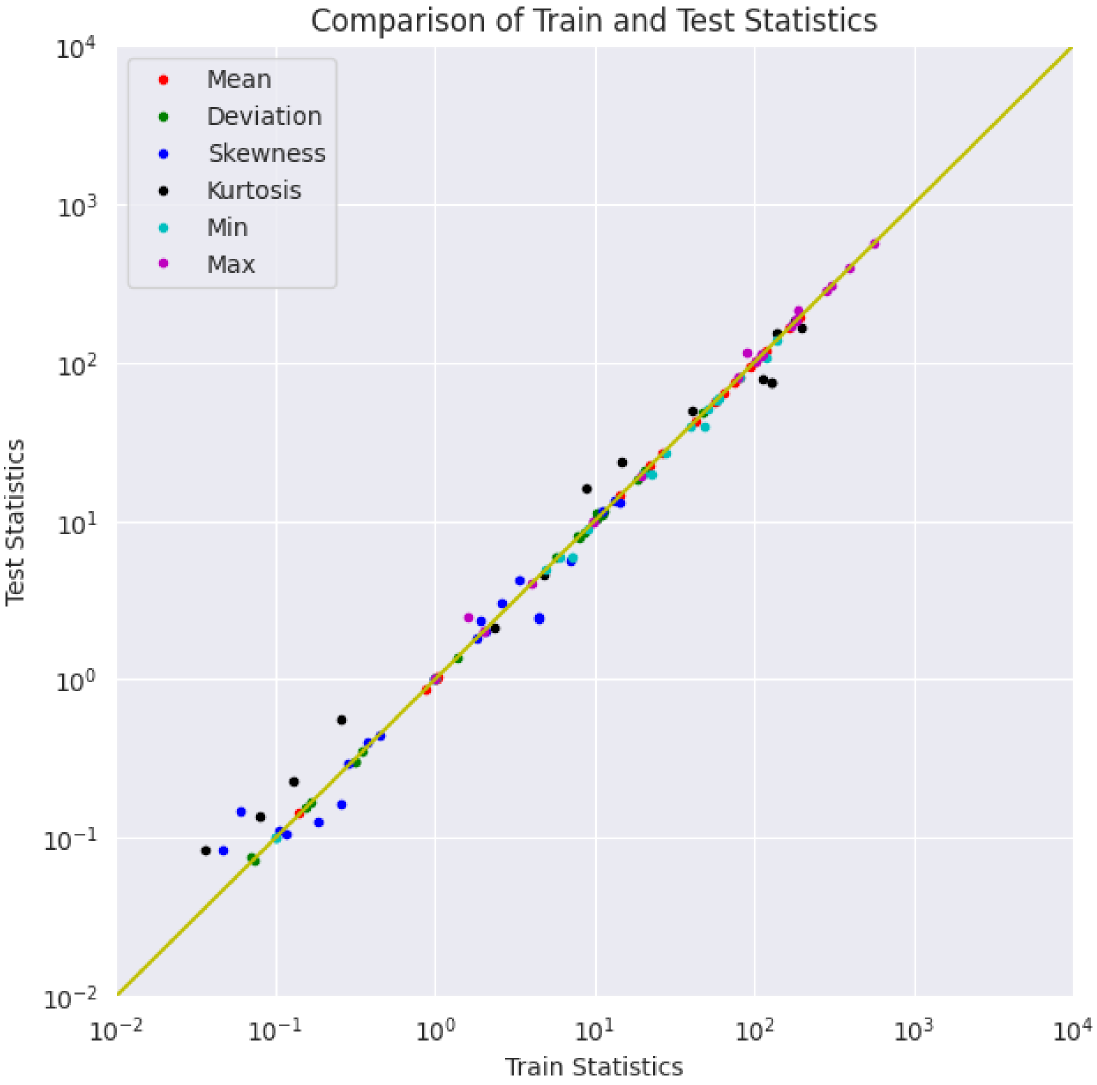

Histogram of Gtp in train
4000
3000
2000
1000
0
0
100
200
300
400
500


Histogram of Gtp in test
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500

### 0.0.1 Checking statistic of train and test

[11]:

```
plt.figure(figsize=(7, 7))

plt.plot(train[features].mean(), test[features].mean(), 'r.', label='Mean')
plt.plot(train[features].std(), test[features].std(), 'g.', label='Deviation')
plt.plot(train[features].skew(), test[features].skew(), 'b.', label='Skewness')
plt.plot(train[features].kurtosis(), test[features].kurtosis(), 'k.',␣
 ↪label='Kurtosis')
plt.plot(train[features].min(), test[features].min(), 'c.', label='Min')
plt.plot(train[features].max(), test[features].max(), 'm.', label='Max')

plt.plot([1e-2, 1e4], [1e-2, 1e4], 'y')
plt.title('Comparison of Train and Test Statistics')
plt.xlabel('Train Statistics')
plt.ylabel('Test Statistics')

plt.xlim(1e-2, 1e4)
plt.ylim(1e-2, 1e4)
plt.xscale('log')
plt.yscale('log')

plt.legend()

plt.show()
```

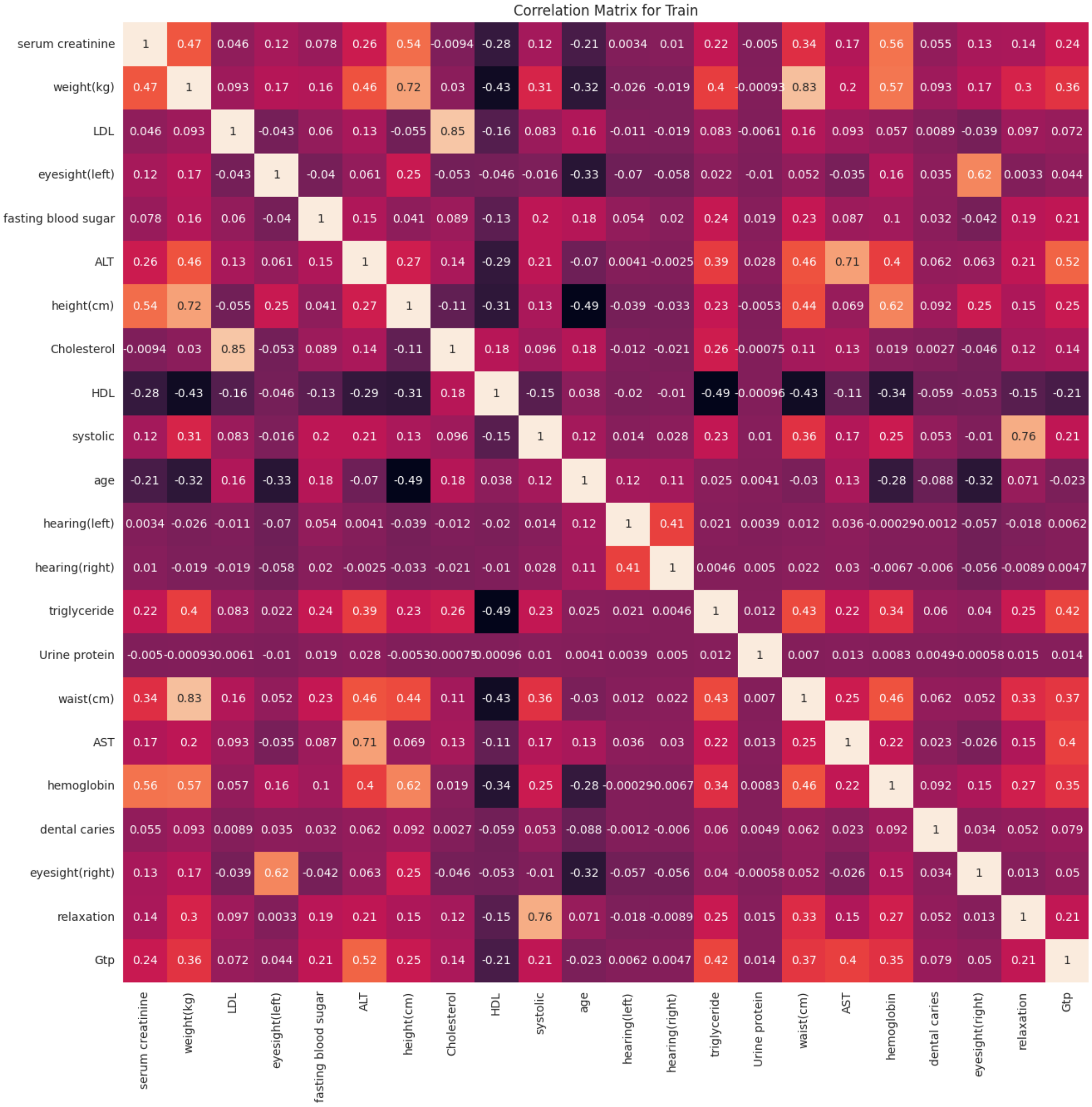


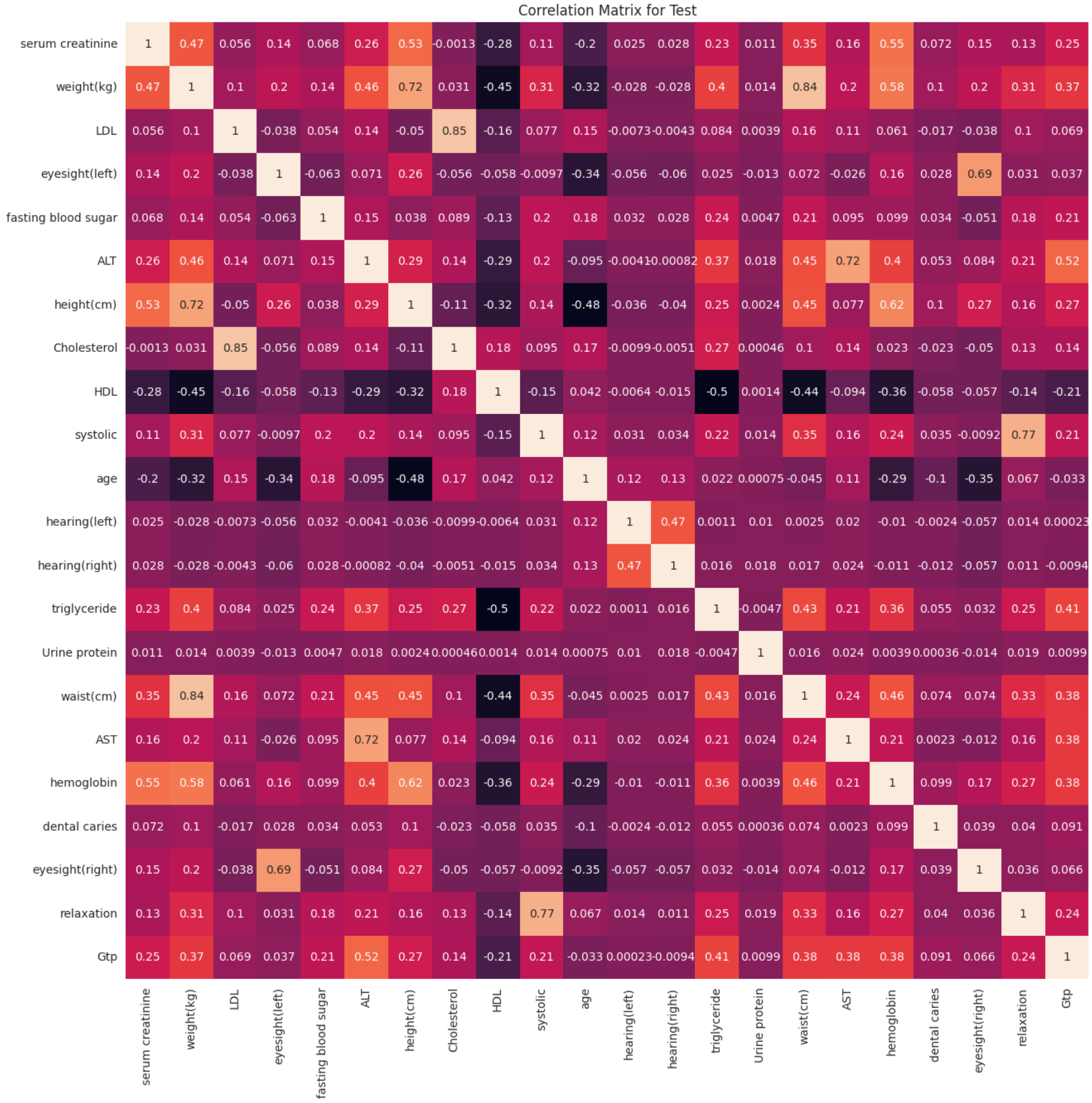
### 0.0.2

[12]:

```
train_corr = train[features].corr()
test_corr = test[features].corr()

plt.figure(figsize=(15, 15))

sns.heatmap(train_corr, annot=True, cbar=False)
plt.title('Correlation Matrix for Train')
plt.show()

plt.figure(figsize=(15, 15))
```

```
sns.heatmap(test_corr, annot=True, cbar=False)
plt.title('Correlation Matrix for Test')
plt.show()
```

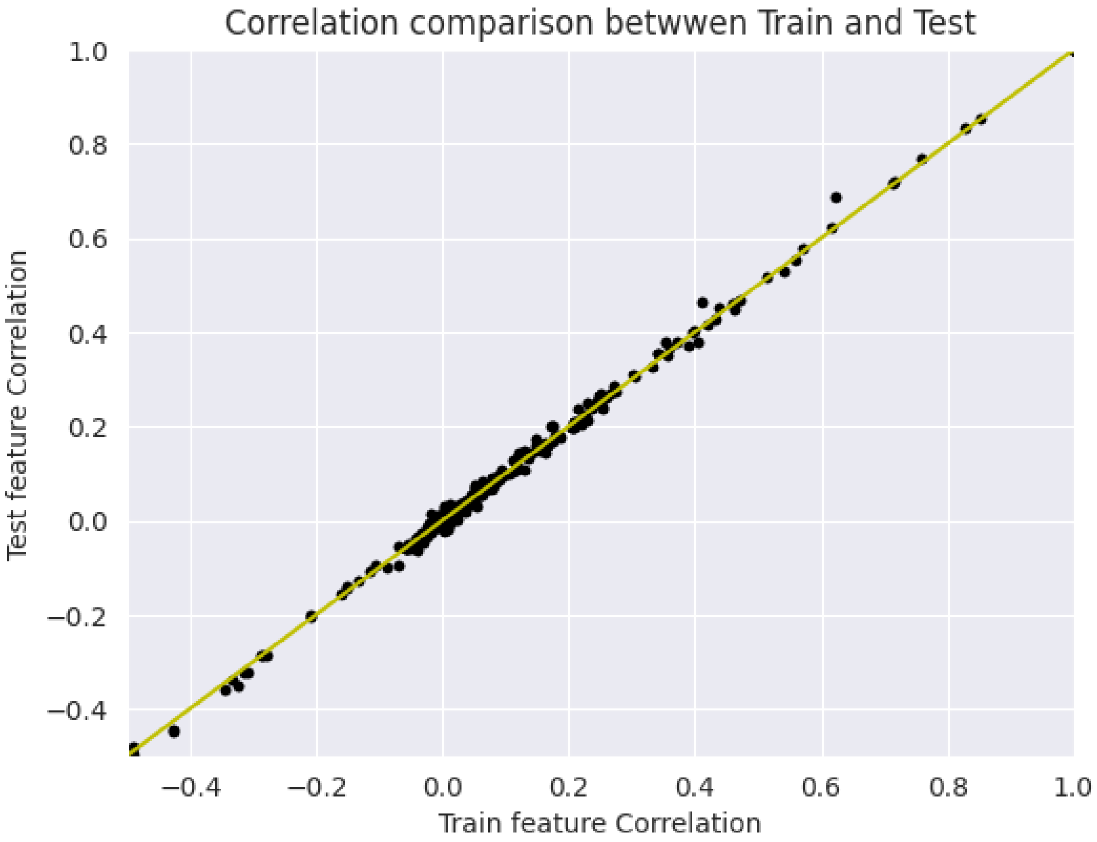




| | serum creatinine | weight(kg) | LDL | eyesight(left) | fasting blood sugar | ALT | height(cm) | Cholesterol | HDL | systolic | age | hearing(left) | hearing(right) | triglyceride | Urine protein | waist(cm) | AST | hemoglobin | dental caries | eyesight(right) | relaxation | Gtp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| serum creatinine | 1 | 0.47 | 0.056 | 0.14 | 0.068 | 0.26 | 0.53 | -0.0013 | -0.28 | 0.11 | -0.2 | 0.025 | 0.028 | 0.23 | 0.011 | 0.35 | 0.16 | 0.55 | 0.072 | 0.15 | 0.13 | 0.25 |
| weight(kg) | 0.47 | 1 | 0.1 | 0.2 | 0.14 | 0.46 | 0.72 | 0.031 | -0.45 | 0.31 | -0.32 | -0.028 | -0.028 | 0.4 | 0.014 | 0.84 | 0.2 | 0.58 | 0.1 | 0.2 | 0.31 | 0.37 |
| LDL | 0.056 | 0.1 | 1 | -0.038 | 0.054 | 0.14 | -0.05 | 0.85 | -0.16 | 0.077 | 0.15 | -0.0073 | -0.0043 | 0.084 | 0.0039 | 0.16 | 0.11 | 0.061 | -0.017 | -0.038 | 0.1 | 0.069 |
| eyesight(left) | 0.14 | 0.2 | -0.038 | 1 | -0.063 | 0.071 | 0.26 | -0.056 | -0.058 | -0.0097 | -0.34 | -0.056 | -0.06 | 0.025 | -0.013 | 0.072 | -0.026 | 0.16 | 0.028 | 0.69 | 0.031 | 0.037 |
| fasting blood sugar | 0.068 | 0.14 | 0.054 | -0.063 | 1 | 0.15 | 0.038 | 0.089 | -0.13 | 0.2 | 0.18 | 0.032 | 0.028 | 0.24 | 0.0047 | 0.21 | 0.095 | 0.099 | 0.034 | -0.051 | 0.18 | 0.21 |
| ALT | 0.26 | 0.46 | 0.14 | 0.071 | 0.15 | 1 | 0.29 | 0.14 | -0.29 | 0.2 | -0.095 | -0.0041 | -0.00082 | 0.37 | 0.018 | 0.45 | 0.72 | 0.4 | 0.053 | 0.084 | 0.21 | 0.52 |
| height(cm) | 0.53 | 0.72 | -0.05 | 0.26 | 0.038 | 0.29 | 1 | -0.11 | -0.32 | 0.14 | -0.48 | -0.036 | -0.04 | 0.25 | 0.0024 | 0.45 | 0.077 | 0.62 | 0.1 | 0.27 | 0.16 | 0.27 |
| Cholesterol | -0.0013 | 0.031 | 0.85 | -0.056 | 0.089 | 0.14 | -0.11 | 1 | 0.18 | 0.095 | 0.17 | -0.0099 | -0.0051 | 0.27 | 0.00046 | 0.1 | 0.14 | 0.023 | -0.023 | -0.05 | 0.13 | 0.14 |
| HDL | -0.28 | -0.45 | -0.16 | -0.058 | -0.13 | -0.29 | -0.32 | 0.18 | 1 | -0.15 | 0.042 | -0.0064 | -0.015 | -0.5 | 0.0014 | -0.44 | -0.094 | -0.36 | -0.058 | -0.057 | -0.14 | -0.21 |
| systolic | 0.11 | 0.31 | 0.077 | -0.0097 | 0.2 | 0.2 | 0.14 | 0.095 | -0.15 | 1 | 0.12 | 0.031 | 0.034 | 0.22 | 0.014 | 0.35 | 0.16 | 0.24 | 0.035 | -0.0092 | 0.77 | 0.21 |
| age | -0.2 | -0.32 | 0.15 | -0.34 | 0.18 | -0.095 | -0.48 | 0.17 | 0.042 | 0.12 | 1 | 0.12 | 0.13 | 0.022 | 0.00075 | -0.045 | 0.11 | -0.29 | -0.1 | -0.35 | 0.067 | -0.033 |
| hearing(left) | 0.025 | -0.028 | -0.0073 | -0.056 | 0.032 | -0.0041 | -0.036 | -0.0099 | -0.0064 | 0.031 | 0.12 | 1 | 0.47 | 0.0011 | 0.01 | 0.0025 | 0.02 | -0.01 | -0.0024 | -0.057 | 0.014 | 0.00023 |
| hearing(right) | 0.028 | -0.028 | -0.0043 | -0.06 | 0.028 | -0.00082 | -0.04 | -0.0051 | -0.015 | 0.034 | 0.13 | 0.47 | 1 | 0.016 | 0.018 | 0.017 | 0.024 | -0.011 | -0.012 | -0.057 | 0.011 | -0.0094 |
| triglyceride | 0.23 | 0.4 | 0.084 | 0.025 | 0.24 | 0.37 | 0.25 | 0.27 | -0.5 | 0.22 | 0.022 | 0.0011 | 0.016 | 1 | -0.0047 | 0.43 | 0.21 | 0.36 | 0.055 | 0.032 | 0.25 | 0.41 |
| Urine protein | 0.011 | 0.014 | 0.0039 | -0.013 | 0.0047 | 0.018 | 0.0024 | 0.00046 | 0.0014 | 0.014 | 0.00075 | 0.01 | 0.018 | -0.0047 | 1 | 0.016 | 0.024 | 0.0039 | 0.00036 | -0.014 | 0.019 | 0.0099 |
| waist(cm) | 0.35 | 0.84 | 0.16 | 0.072 | 0.21 | 0.45 | 0.45 | 0.1 | -0.44 | 0.35 | -0.045 | 0.0025 | 0.017 | 0.43 | 0.016 | 1 | 0.24 | 0.46 | 0.074 | 0.074 | 0.33 | 0.38 |
| AST | 0.16 | 0.2 | 0.11 | -0.026 | 0.095 | 0.72 | 0.077 | 0.14 | -0.094 | 0.16 | 0.11 | 0.02 | 0.024 | 0.21 | 0.024 | 0.24 | 1 | 0.21 | 0.0023 | -0.012 | 0.16 | 0.38 |
| hemoglobin | 0.55 | 0.58 | 0.061 | 0.16 | 0.099 | 0.4 | 0.62 | 0.023 | -0.36 | 0.24 | -0.29 | -0.01 | -0.011 | 0.36 | 0.0039 | 0.46 | 0.21 | 1 | 0.099 | 0.17 | 0.27 | 0.38 |
| dental caries | 0.072 | 0.1 | -0.017 | 0.028 | 0.034 | 0.053 | 0.1 | -0.023 | -0.058 | 0.035 | -0.1 | -0.0024 | -0.012 | 0.055 | 0.00036 | 0.074 | 0.0023 | 0.099 | 1 | 0.039 | 0.04 | 0.091 |
| eyesight(right) | 0.15 | 0.2 | -0.038 | 0.69 | -0.051 | 0.084 | 0.27 | -0.05 | -0.057 | -0.0092 | -0.35 | -0.057 | -0.057 | 0.032 | -0.014 | 0.074 | -0.012 | 0.17 | 0.039 | 1 | 0.036 | 0.066 |
| relaxation | 0.13 | 0.31 | 0.1 | 0.031 | 0.18 | 0.21 | 0.16 | 0.13 | -0.14 | 0.77 | 0.067 | 0.014 | 0.011 | 0.25 | 0.019 | 0.33 | 0.16 | 0.27 | 0.04 | 0.036 | 1 | 0.24 |
| Gtp | 0.25 | 0.37 | 0.069 | 0.037 | 0.21 | 0.52 | 0.27 | 0.14 | -0.21 | 0.21 | -0.033 | 0.00023 | -0.0094 | 0.41 | 0.0099 | 0.38 | 0.38 | 0.38 | 0.091 | 0.066 | 0.24 | 1 |

### 0.0.3 Linear Correlation between test and train features

```
[13]: plt.plot(np.array(train_corr).flatten(), np.array(test_corr).flatten(), 'k.')
      plt.plot([-0.5, 1], [-0.5, 1], 'y')
      plt.xlim(-0.5, 1)
      plt.ylim(-0.5, 1)

      plt.xlabel('Train feature Correlation')
      plt.ylabel('Test feature Correlation')

      plt.title('Correlation comparison betwwen Train and Test')
      plt.show()
```



[14]:
```
sorted(train[features].nunique())
```

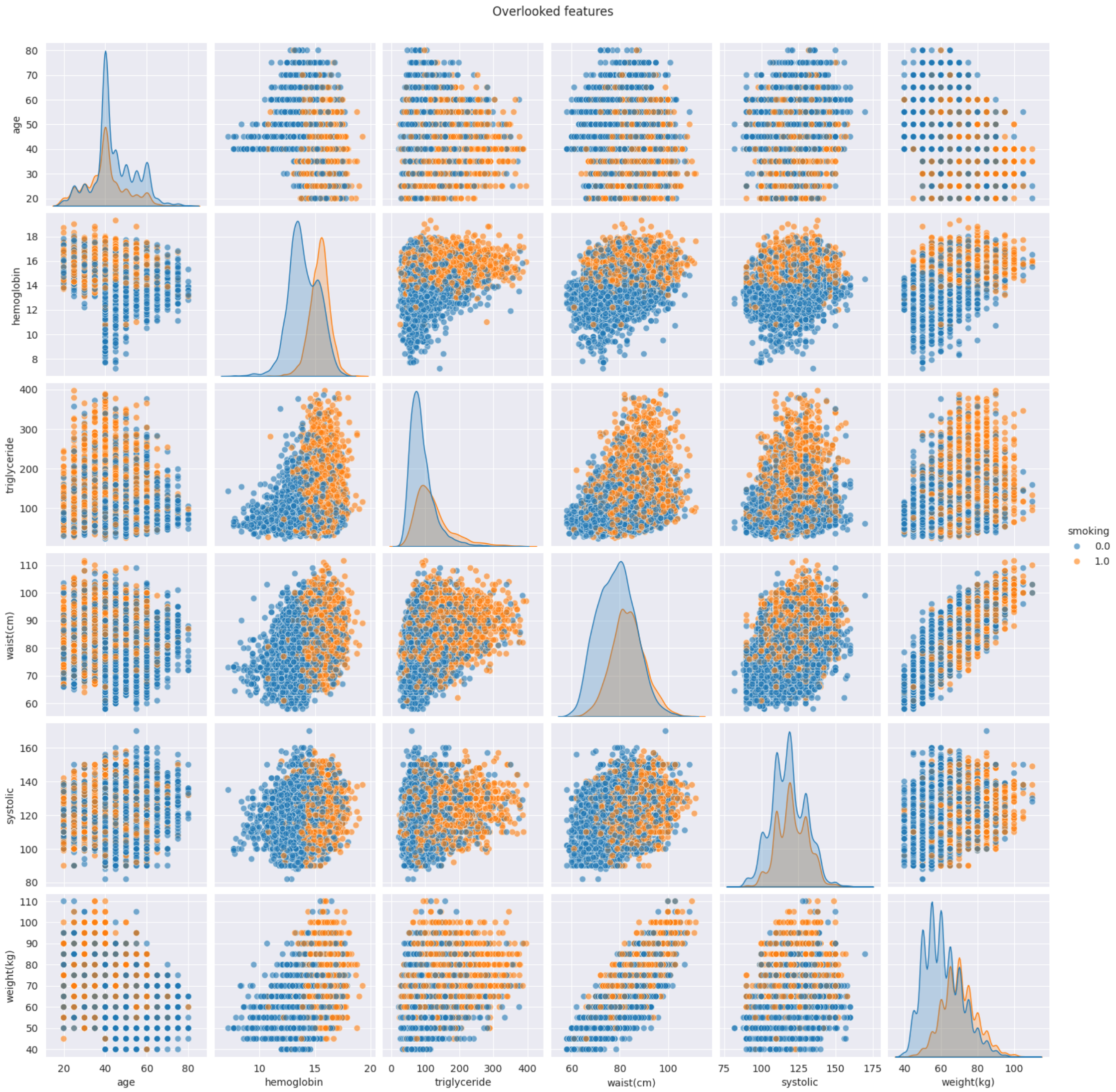
[14]:
```
[2,
 2,
 2,
 4,
 10,
 13,
 13,
 14,
 14,
 15,
 57,
 62,
 74,
 75,
 96,
 107,
 109,
 133,
 148,
 168,
 327,
 353]
```

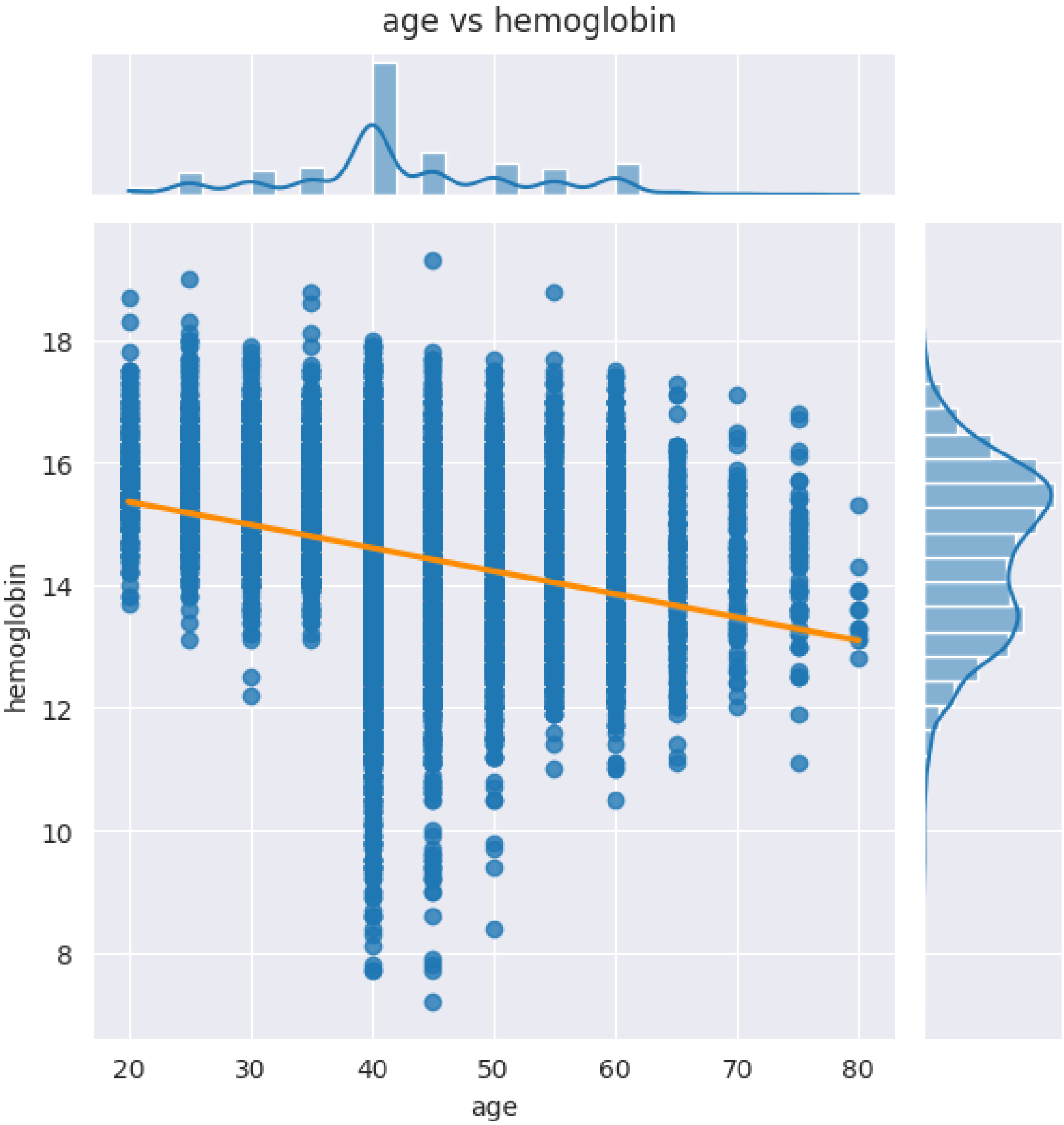
### 0.0.4

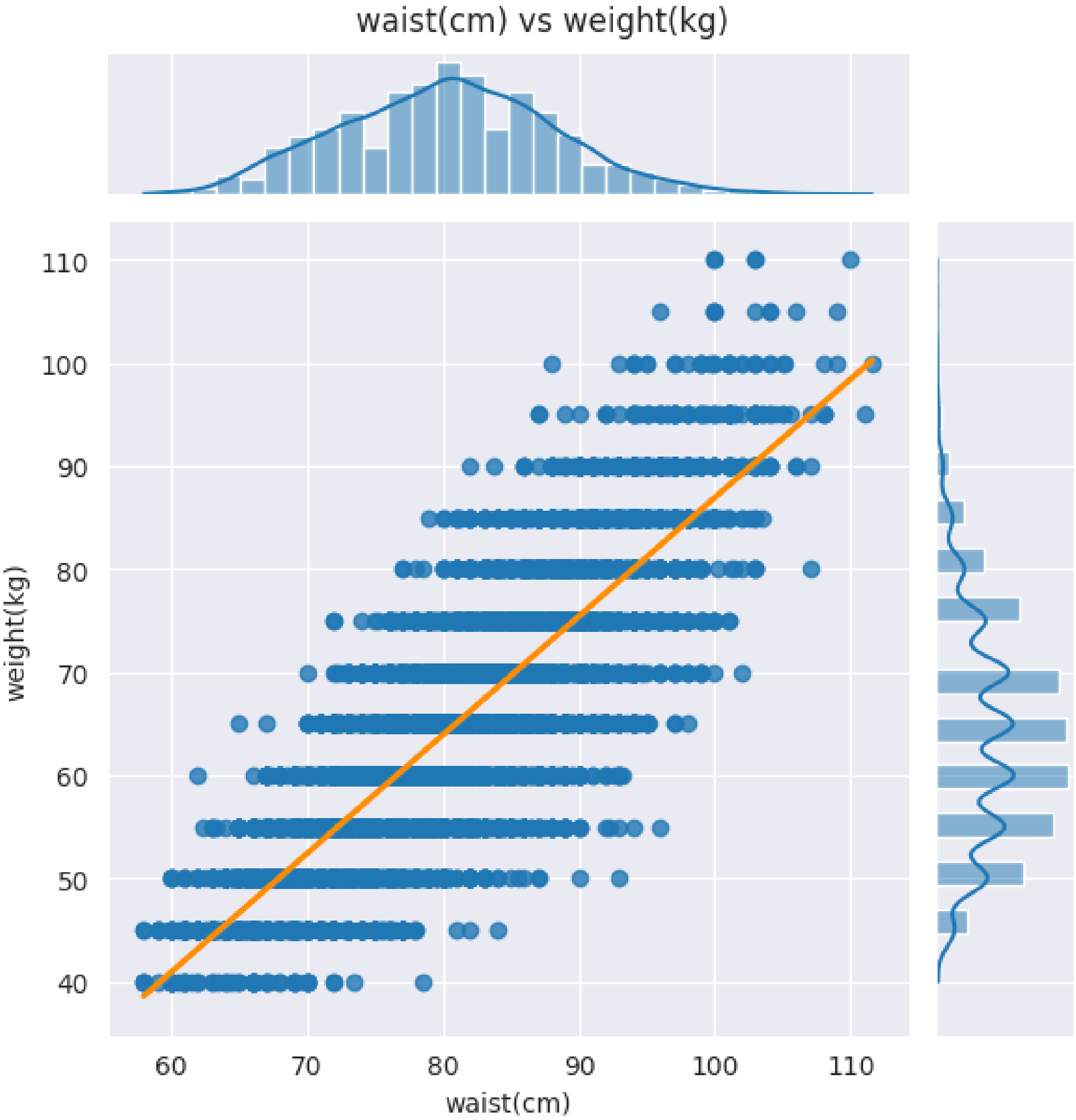
[15]:

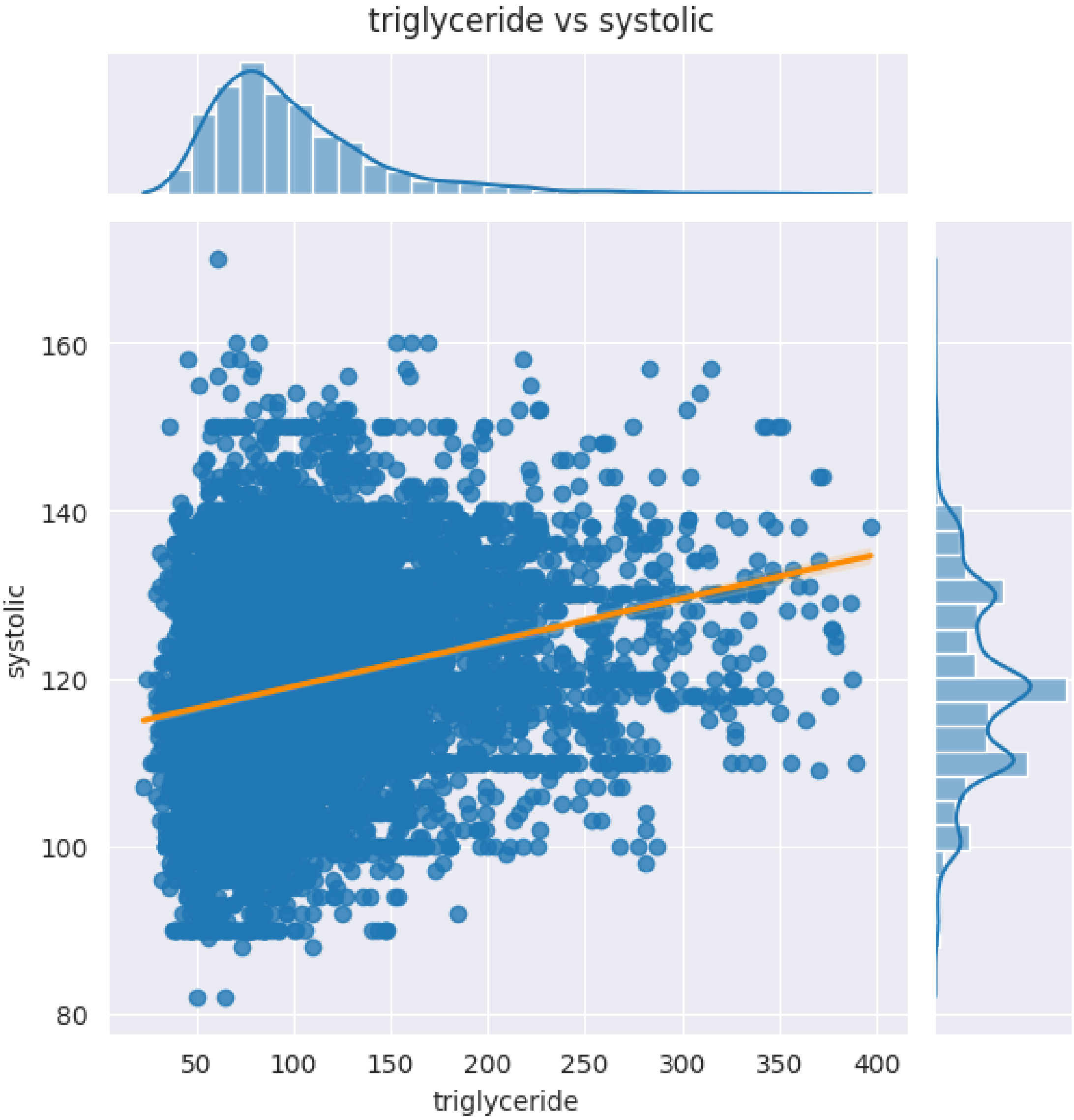
```python
overlooked_features = ['age', 'hemoglobin', 'triglyceride', 'waist(cm)', 'systolic', 'weight(kg)']

sns.pairplot(
    train[overlooked_features + ['smoking']],
    hue='smoking',
    vars=overlooked_features,
    diag_kind='kde',
    plot_kws={'alpha': 0.6, 's': 40},
    diag_kws={'fill': True}
)
plt.suptitle('Overlooked features', y=1.02)
plt.show()

pairs_to_plot = [
    ('age', 'hemoglobin'),
    ('waist(cm)', 'weight(kg)'),
    ('triglyceride', 'systolic')
]

for x_col, y_col in pairs_to_plot:
    g = sns.jointplot(
        data=train,
        x=x_col,
        y=y_col,
        kind='reg',
        height=6,
        marginal_kws={'bins': 30, 'fill': True},
        line_kws={'color': 'darkorange'}
    )
    g.fig.suptitle(f'{x_col} vs {y_col}', y=1.02)
    plt.show()

pd.plotting.scatter_matrix(
    train[overlooked_features],
    figsize=(10, 10),
    diagonal='kde',
    alpha=0.5,
    s=20
)
```

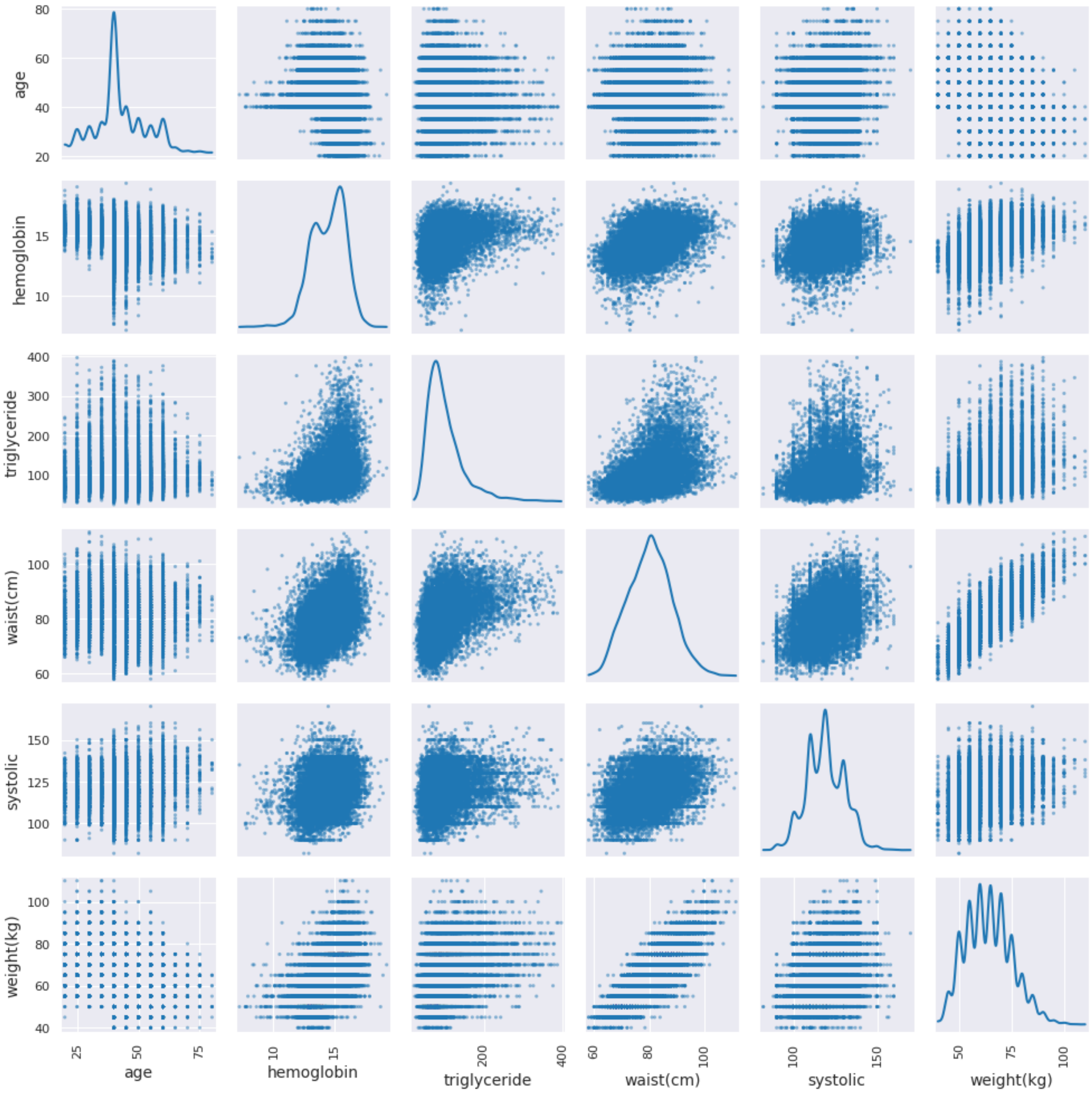
```
plt.tight_layout()
plt.show()
```




age vs hemoglobin
18
16
14
12
10
8
hemoglobin
20
30
40
50
60
70
80
age


waist(cm) vs weight(kg)
weight(kg)
110
100
90
80
70
60
50
40
60
70
80
90
100
110
waist(cm)


triglyceride vs systolic
systolic
160
140
120
100
80
50
100
150
200
250
300
350
400
triglyceride



### 0.0.5

EDA , , -
. , . , train test
, . ( , , , , )
train test, .

### 0.0.6

```
[16]: from normalization import impute_median, normalize_zscore
from feature_pruning import auto_prune_correlated_features
from cross_validation import train_test_split_stratified
from sklearn.preprocessing import OneHotEncoder
```

```
TARGET = 'smoking'
ID_COL = 'id'
overlooked_features = [column for column in train.columns if column not in␣
 ↪{ID_COL, TARGET}]

X = train[overlooked_features].copy()
X_test = test[overlooked_features].copy()
y = train[TARGET].values

#Imputation
X = impute_median(X)
X_test = impute_median(X_test)

# One-Hot Encoding
cateogorical_cols = [column for column in overlooked_features if X[column].
 ↪nunique() <= 10]
encorder = OneHotEncoder(sparse_output=False, drop='first')
encorder.fit(pd.concat([X[cateogorical_cols], X_test[cateogorical_cols]],␣
 ↪axis=0))

X_ohe      = pd.DataFrame(encorder.transform(X[cateogorical_cols]),
                          columns=encorder.
 ↪get_feature_names_out(cateogorical_cols),
                          index=X.index)
X_test_ohe = pd.DataFrame(encorder.transform(X_test[cateogorical_cols]),
                          columns=encorder.
 ↪get_feature_names_out(cateogorical_cols),
                          index=X_test.index)

X      = pd.concat([X.drop(columns=cateogorical_cols),      X_ohe], axis=1)
X_test = pd.concat([X_test.drop(columns=cateogorical_cols), X_test_ohe], axis=1)

# Drop too correlated data
df_prune = X.copy()
df_prune[TARGET] = y
corr = df_prune.drop(columns=[TARGET]).corr().abs()
upper = corr.where(np.triu(np.ones(corr.shape), k=1).astype(bool))
feature_pairs = [(c1, c2) for c1 in upper.columns for c2 in upper.index if␣
 ↪upper.loc[c2, c1] > 0.8]
all_feats = X.columns.tolist()

selected = auto_prune_correlated_features(
    df=df_prune,
    target_column=TARGET,
    feature_pairs=feature_pairs,
    all_features=all_feats,
    threshold=0.8,
    verbose=True
)

X      = X[selected]
X_test = X_test[selected]

# Normalization
numeric_cols = [c for c in X.columns if c not in encorder.
 ↪get_feature_names_out()]
X_norm, (mu, sigma)        = normalize_zscore(X[numeric_cols])
X[numeric_cols]            = X_norm
X_test[numeric_cols]       = (X_test[numeric_cols] - mu) / sigma

# Split
X_tr, X_val, y_tr, y_val = train_test_split_stratified(
    X, y, test_size=0.2, random_state=42
)

print(X_tr.shape, X_val.shape, X_test.shape)
```

```
waist(cm) vs weight(kg) → MSE_A=0.1471, MSE_B=0.1470 → drop f1
LDL vs Cholesterol → MSE_A=0.1470, MSE_B=0.1470 → drop f2
(11983, 30) (2996, 30) (10000, 30)
```

# 1 EDA

[17]:

```
from ensemble_manual import BaggingClassifierManual
from sklearn.ensemble import RandomForestClassifier, BaggingClassifier
from sklearn.metrics import roc_auc_score

bag_manual = BaggingClassifierManual(
    base_estimator=RandomForestClassifier(max_depth=3, random_state=42),
    n_estimators=50,
    max_samples=0.8,
    random_state=42
).fit(X_tr.values, y_tr)

bag_sklearn = BaggingClassifier(
    estimator=RandomForestClassifier(max_depth=3, random_state=42),
    n_estimators=50,
    max_samples=0.8,
    random_state=42,
    n_jobs=-1
).fit(X_tr, y_tr)
```

```python
p_manual = bag_manual.predict_proba(X_val.values)[:,1]
p_sklearn = bag_sklearn.predict_proba(X_val)[:,1]


print("Manual Bagging ROC-AUC:", roc_auc_score(y_val, p_manual))
print("Sklearn Bagging ROC-AUC:", roc_auc_score(y_val, p_sklearn))
```

```
Manual Bagging ROC-AUC: 0.8500621879804453
Sklearn Bagging ROC-AUC: 0.8506538375166265
```

[18]:
```python
from ensemble_manual import GradientBoostingClassifierManual
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.metrics import roc_auc_score
from sklearn.tree import DecisionTreeRegressor

base_tree = DecisionTreeRegressor(max_depth=7, random_state=42)
gb_manual = GradientBoostingClassifierManual(
    base_estimator=base_tree,
    n_estimators=100, learning_rate=0.1
).fit(X_tr.values, y_tr)
p_manual = gb_manual.predict_proba(X_val.values)[:,1]

gb_sklearn = GradientBoostingClassifier(
    n_estimators=100, learning_rate=0.1, max_depth=7, random_state=42
).fit(X_tr, y_tr)
p_skl = gb_sklearn.predict_proba(X_val)[:,1]

print("Manual GB AUC:", roc_auc_score(y_val, p_manual))
print("Sklearn GB AUC:", roc_auc_score(y_val, p_skl))
```

```
Manual GB AUC: 0.8748865165325977
Sklearn GB AUC: 0.8810832416828374
```

[19]:
```python
from sklearn.metrics import roc_auc_score
from sklearn.tree import DecisionTreeRegressor
from ensemble_manual import GradientBoostingClassifierManual

best_auc = -1
best_depth = None
best_model = None

for depth in range(1, 11):
    base_tree = DecisionTreeRegressor(max_depth=depth, random_state=42)
    gb_manual = GradientBoostingClassifierManual(
        base_estimator=base_tree,
        n_estimators=100,
        learning_rate=0.1
    ).fit(X_tr.values, y_tr)
    p_manual = gb_manual.predict_proba(X_val.values)[:, 1]
    auc = roc_auc_score(y_val, p_manual)

    print(f"Depth {depth}: AUC = {auc:.4f}")

    if auc > best_auc:
        best_auc = auc
        best_depth = depth
        best_model = gb_manual

print(f"\nBest manual GB depth: {best_depth} with AUC = {best_auc:.4f}")
```

```
Depth 1: AUC = 0.8384
Depth 2: AUC = 0.8511
Depth 3: AUC = 0.8603
Depth 4: AUC = 0.8650
Depth 5: AUC = 0.8695
Depth 6: AUC = 0.8732
Depth 7: AUC = 0.8749
Depth 8: AUC = 0.8742
Depth 9: AUC = 0.8729
Depth 10: AUC = 0.8713

Best manual GB depth: 7 with AUC = 0.8749
```

[20]:

```
import numpy as np
from metrics import (
    accuracy_manual, precision_manual,
    recall_manual, f1_score_manual,
    roc_auc_score_manual, average_precision_score_manual
)
from sklearn.metrics import (
    accuracy_score, precision_score,
    recall_score, f1_score,
    roc_auc_score, average_precision_score
)

y_true  = np.array([0, 0, 1, 1, 0, 1, 1, 0])
y_pred  = np.array([0, 1, 1, 1, 0, 0, 1, 0])

y_score = y_pred.astype(float)

print("       Metric |  Manual  |  Sklearn")
print("-----------------------------------")
print(f"Accuracy     | {accuracy_manual(y_true, y_pred):7.4f} |␣
 ↪{accuracy_score(y_true, y_pred):7.4f}")
```

```
print(f"Precision    | {precision_manual(y_true, y_pred):7.4f} |␣
↪{precision_score(y_true, y_pred):7.4f}")
print(f"Recall       | {recall_manual(y_true, y_pred):7.4f} |␣
↪{recall_score(y_true, y_pred):7.4f}")
print(f"F1-Score     | {f1_score_manual(y_true, y_pred):7.4f} |␣
↪{f1_score(y_true, y_pred):7.4f}")
print(f"ROC-AUC      | {roc_auc_score_manual(y_true, y_score):7.4f} |␣
↪{roc_auc_score(y_true, y_score):7.4f}")
print(f"PR-AUC (AP)  | {average_precision_score_manual(y_true, y_score):7.4f} |␣
↪{average_precision_score(y_true, y_score):7.4f}")
```

```
       Metric |  Manual   |  Sklearn
-----------------------------------
Accuracy     |  0.7500 |  0.7500
Precision    |  0.7500 |  0.7500
Recall       |  0.7500 |  0.7500
F1-Score     |  0.7500 |  0.7500
ROC-AUC      |  0.7500 |  0.7500
PR-AUC (AP)  |  0.5193 |  0.6875
```

[21]:

```
from metrics import (
    accuracy_manual, precision_manual,
    recall_manual, f1_score_manual,
    roc_auc_score_manual, average_precision_score_manual
)
from ensemble_manual import GradientBoostingClassifierManual
from lightgbm import LGBMClassifier
from xgboost import XGBClassifier
from catboost import CatBoostClassifier
from sklearn.ensemble import HistGradientBoostingClassifier,␣
↪GradientBoostingClassifier

models = {
    'LightGBM':   LGBMClassifier(n_estimators=100, learning_rate=0.1,␣
↪random_state=42),
    'XGBoost':    XGBClassifier(n_estimators=100, learning_rate=0.1,␣
↪random_state=42, eval_metric='logloss'),
    'CatBoost':   CatBoostClassifier(iterations=100, learning_rate=0.1,␣
↪random_seed=42, verbose=0),
    'SklearnHGB': HistGradientBoostingClassifier(max_iter=100, learning_rate=0.
↪1, random_state=42),
    'SklearnGB':  GradientBoostingClassifier(n_estimators=100, learning_rate=0.
↪1, max_depth=3, random_state=42),
    'GB_Manual':  GradientBoostingClassifierManual(
                      base_estimator=DecisionTreeRegressor(max_depth=3,␣
↪random_state=42),
                      n_estimators=100, learning_rate=0.1
```

```
                   )
}

for name, mdl in models.items():
    if name == 'GB_Manual':
        mdl.fit(X_tr.values, y_tr)
    else:
        mdl.fit(X_tr, y_tr)

metrics = [
    ('Accuracy', accuracy_manual,      lambda y, y_s, y_p: y_p),
    ('Precision', precision_manual,    lambda y, y_s, y_p: y_p),
    ('Recall', recall_manual,          lambda y, y_s, y_p: y_p),
    ('F1', f1_score_manual,            lambda y, y_s, y_p: y_p),
    ('ROC-AUC', roc_auc_score_manual,  lambda y, y_s, y_p: y_s),
    ('PR-AUC', average_precision_score_manual, lambda y, y_s, y_p: y_s),
]

results = {name: {} for name in models}

for name, mdl in models.items():
    if name == 'GB_Manual':
        y_score = mdl.predict_proba(X_val.values)[:,1]
        y_pred  = mdl.predict(X_val.values)
    else:
        y_score = mdl.predict_proba(X_val)[:,1]
        y_pred  = mdl.predict(X_val)
    for mname, mfunc, selector in metrics:
        arr = selector(y_val, y_score, y_pred)
        if mname in ('ROC-AUC', 'PR-AUC'):
            val = mfunc(y_val, y_score)
        else:
            val = mfunc(y_val, y_pred)
        results[name][mname] = val

print(f"{'Model':<12}", "  ".join(f"{m[0]:>9}" for m in metrics))
print("-" * (12 + 11 * len(metrics)))
for name, res in results.items():
    print(f"{name:<12}", "  ".join(f"{res[m[0]]:9.4f}" for m in metrics))

best_by_metric = {}
for mname, _, _ in metrics:
    best_model = max(results, key=lambda nm: results[nm][mname])
    best_by_metric[mname] = (best_model, results[best_model][mname])

print("\nBest by metric:")
for mname, (mdl, val) in best_by_metric.items():
    print(f"{mname:<9} -> {mdl} ({val:.4f})")
```

```
[LightGBM] [Warning] Found whitespace in feature_names, replace with underlines
[LightGBM] [Info] Number of positive: 4393, number of negative: 7590
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of
testing was 0.000359 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 1099
[LightGBM] [Info] Number of data points in the train set: 11983, number of used
features: 29
[LightGBM] [Info] [binary:BoostFromScore]: pavg=0.366603 -> initscore=-0.546819
[LightGBM] [Info] Start training from score -0.546819
Model         Accuracy  Precision     Recall         F1    ROC-AUC     PR-AUC
-----------------------------------------------------------------------------
LightGBM        0.7981     0.7046     0.7732     0.7373     0.8804     0.7679
XGBoost         0.8027     0.7076     0.7869     0.7451     0.8804     0.7682
CatBoost        0.7951     0.6921     0.7942     0.7396     0.8797     0.7694
SklearnHGB      0.7977     0.7003     0.7832     0.7395     0.8785     0.7656
SklearnGB       0.7944     0.6944     0.7842     0.7365     0.8766     0.7660
GB_Manual       0.7727     0.6745     0.7341     0.7030     0.8603     0.7397

Best by metric:
Accuracy  -> XGBoost (0.8027)
Precision -> XGBoost (0.7076)
Recall    -> CatBoost (0.7942)
F1        -> XGBoost (0.7451)
ROC-AUC   -> LightGBM (0.8804)
PR-AUC    -> CatBoost (0.7694)
```

### OPTUNA

[34]:
```
import numpy as np
import lightgbm as lgb
import optuna
from optuna.integration import LightGBMTunerCV
from sklearn.model_selection import StratifiedKFold
from lightgbm import LGBMClassifier
from sklearn.metrics import roc_auc_score

train_data = lgb.Dataset(X_tr, label=y_tr)

tuner = LightGBMTunerCV(
    params={
        'objective': 'binary',
        'metric': 'auc',
        'verbosity': -1,
        'seed': 42
    },
    train_set=train_data,
    folds=StratifiedKFold(n_splits=3, shuffle=True, random_state=42),
    num_boost_round=500,
    study=optuna.create_study(
        direction='maximize',
        pruner=optuna.pruners.SuccessiveHalvingPruner()
    )
)

tuner.run()

best_params_cv = tuner.best_params
print("TunerCV best params:", best_params_cv)

final_cv = LGBMClassifier(**best_params_cv)
final_cv.fit(
    np.vstack([X_tr, X_val]),
    np.concatenate([y_tr, y_val])
)
val_preds_cv1 = final_cv.predict_proba(X_val)[:, 1]
preds_cv = final_cv.predict_proba(X_test)[:, 1]
print("TunerCV ROC-AUC on val:", roc_auc_score(y_val, val_preds_cv1))
```

```
[I 2025-05-17 20:38:15,141] A new study created in memory with name: no-
name-8a97dee8-0552-4e3c-b0b0-01fe2b897035


  0%|          | 0/7 [00:00<?, ?it/s]

feature_fraction, val_score: -inf:   0%|          | 0/7 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.874942:   0%|          | 0/7 [00:17<?,
?it/s]

feature_fraction, val_score: 0.874942:  14%|#4        | 1/7 [00:17<01:45,
17.64s/it][I 2025-05-17 20:38:32,784] Trial 0 finished with value:
0.8749422751595265 and parameters: {'feature_fraction': 1.0}. Best is trial 0
with value: 0.8749422751595265.


feature_fraction, val_score: 0.874942:  14%|#4        | 1/7 [00:17<01:45,
17.64s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
```

```
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.875863:  14%|#4        | 1/7 [00:19<01:45,
17.64s/it]

feature_fraction, val_score: 0.875863:  29%|##8       | 2/7 [00:19<00:41,
8.22s/it][I 2025-05-17 20:38:34,410] Trial 1 finished with value:
0.8758628486608327 and parameters: {'feature_fraction': 0.6}. Best is trial 1
with value: 0.8758628486608327.


feature_fraction, val_score: 0.875863:  29%|##8       | 2/7 [00:19<00:41,
8.22s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.878770:  29%|##8       | 2/7 [00:21<00:41,
8.22s/it]

feature_fraction, val_score: 0.878770:  43%|####2     | 3/7 [00:21<00:21,
5.34s/it][I 2025-05-17 20:38:36,332] Trial 2 finished with value:
0.8787701914726952 and parameters: {'feature_fraction': 0.5}. Best is trial 2
with value: 0.8787701914726952.


feature_fraction, val_score: 0.878770:  43%|####2     | 3/7 [00:21<00:21,
5.34s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.878770:  43%|####2     | 3/7 [00:22<00:21,
5.34s/it]

feature_fraction, val_score: 0.878770:  57%|#####7    | 4/7 [00:22<00:11,
3.88s/it][I 2025-05-17 20:38:37,977] Trial 3 finished with value:
0.8784615114654778 and parameters: {'feature_fraction': 0.7}. Best is trial 2
with value: 0.8787701914726952.


feature_fraction, val_score: 0.878770:  57%|#####7    | 4/7 [00:22<00:11,
3.88s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
```

```
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.878922:  57%|#####7    | 4/7 [00:24<00:11,
3.88s/it]

feature_fraction, val_score: 0.878922:  71%|#######1  | 5/7 [00:24<00:06,
3.10s/it][I 2025-05-17 20:38:39,691] Trial 4 finished with value:
0.878922398447289 and parameters: {'feature_fraction': 0.4}. Best is trial 4
with value: 0.878922398447289.


feature_fraction, val_score: 0.878922:  71%|#######1  | 5/7 [00:24<00:06,
3.10s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.878922:  71%|#######1  | 5/7 [00:26<00:06,
3.10s/it]

feature_fraction, val_score: 0.878922:  86%|########5 | 6/7 [00:26<00:02,
2.66s/it][I 2025-05-17 20:38:41,498] Trial 5 finished with value:
0.8761271854010415 and parameters: {'feature_fraction': 0.8999999999999999}.
Best is trial 4 with value: 0.878922398447289.


feature_fraction, val_score: 0.878922:  86%|########5 | 6/7 [00:26<00:02,
2.66s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction, val_score: 0.878922:  86%|########5 | 6/7 [00:28<00:02,
2.66s/it]

feature_fraction, val_score: 0.878922: 100%|##########| 7/7 [00:28<00:00,
2.42s/it][I 2025-05-17 20:38:43,421] Trial 6 finished with value:
0.8757740381613591 and parameters: {'feature_fraction': 0.8}. Best is trial 4
with value: 0.878922398447289.
feature_fraction, val_score: 0.878922: 100%|##########| 7/7 [00:28<00:00,
4.04s/it]


  0%|          | 0/20 [00:00<?, ?it/s]
```

```
num_leaves, val_score: 0.878922:   0%|          | 0/20 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.878922:   0%|          | 0/20 [00:06<?, ?it/s]

num_leaves, val_score: 0.878922:   5%|5         | 1/20 [00:06<02:07,
6.74s/it][I 2025-05-17 20:38:50,160] Trial 7 finished with value:
0.8764987784303839 and parameters: {'num_leaves': 201}. Best is trial 7 with
value: 0.8764987784303839.


num_leaves, val_score: 0.878922:   5%|5         | 1/20 [00:06<02:07,
6.74s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.878922:   5%|5         | 1/20 [00:12<02:07,
6.74s/it]

num_leaves, val_score: 0.878922:  10%|#         | 2/20 [00:12<01:49,
6.06s/it][I 2025-05-17 20:38:55,752] Trial 8 finished with value:
0.8762219848818167 and parameters: {'num_leaves': 190}. Best is trial 7 with
value: 0.8764987784303839.


num_leaves, val_score: 0.878922:  10%|#         | 2/20 [00:12<01:49,
6.06s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.878922:  10%|#         | 2/20 [00:17<01:49,
6.06s/it]

num_leaves, val_score: 0.878922:  15%|#5        | 3/20 [00:17<01:37,
5.74s/it][I 2025-05-17 20:39:01,114] Trial 9 finished with value:
0.8769832058872608 and parameters: {'num_leaves': 134}. Best is trial 9 with
value: 0.8769832058872608.
```

```
num_leaves, val_score: 0.878922:  15%|#5        | 3/20 [00:17<01:37,
5.74s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.878922:  15%|#5        | 3/20 [00:22<01:37,
5.74s/it]

num_leaves, val_score: 0.878922:  20%|##        | 4/20 [00:22<01:24,
5.30s/it][I 2025-05-17 20:39:05,742] Trial 10 finished with value:
0.8776814012082172 and parameters: {'num_leaves': 36}. Best is trial 10 with
value: 0.8776814012082172.


num_leaves, val_score: 0.878922:  20%|##        | 4/20 [00:22<01:24,
5.30s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  20%|##        | 4/20 [00:23<01:24,
5.30s/it]

num_leaves, val_score: 0.882408:  25%|##5       | 5/20 [00:23<00:59,
3.99s/it][I 2025-05-17 20:39:07,412] Trial 11 finished with value:
0.8824083612697905 and parameters: {'num_leaves': 7}. Best is trial 11 with
value: 0.8824083612697905.


num_leaves, val_score: 0.882408:  25%|##5       | 5/20 [00:23<00:59,
3.99s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  25%|##5       | 5/20 [00:25<00:59,
3.99s/it]

num_leaves, val_score: 0.882408:  30%|###       | 6/20 [00:25<00:43,
3.14s/it][I 2025-05-17 20:39:08,890] Trial 12 finished with value:
0.8786403281352123 and parameters: {'num_leaves': 2}. Best is trial 11 with
value: 0.8824083612697905.
```

```
num_leaves, val_score: 0.882408:  30%|###       | 6/20 [00:25<00:43,
3.14s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  30%|###       | 6/20 [00:29<00:43,
3.14s/it]

num_leaves, val_score: 0.882408:  35%|###5      | 7/20 [00:29<00:42,
3.30s/it][I 2025-05-17 20:39:12,536] Trial 13 finished with value:
0.8767084543900862 and parameters: {'num_leaves': 78}. Best is trial 11 with
value: 0.8824083612697905.


num_leaves, val_score: 0.882408:  35%|###5      | 7/20 [00:29<00:42,
3.30s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  35%|###5      | 7/20 [00:36<00:42,
3.30s/it]

num_leaves, val_score: 0.882408:  40%|####      | 8/20 [00:36<00:56,
4.74s/it][I 2025-05-17 20:39:20,366] Trial 14 finished with value:
0.8773508704770823 and parameters: {'num_leaves': 255}. Best is trial 11 with
value: 0.8824083612697905.


num_leaves, val_score: 0.882408:  40%|####      | 8/20 [00:36<00:56,
4.74s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  40%|####      | 8/20 [00:40<00:56,
4.74s/it]

num_leaves, val_score: 0.882408:  45%|####5     | 9/20 [00:40<00:49,
4.50s/it][I 2025-05-17 20:39:24,328] Trial 15 finished with value:
0.8768188039183699 and parameters: {'num_leaves': 80}. Best is trial 11 with
value: 0.8824083612697905.
```

```
num_leaves, val_score: 0.882408:  45%|####5     | 9/20 [00:40<00:49,
4.50s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.882408:  45%|####5     | 9/20 [00:46<00:49,
4.50s/it]

num_leaves, val_score: 0.882408:  50%|#####     | 10/20 [00:46<00:48,
4.84s/it][I 2025-05-17 20:39:29,923] Trial 16 finished with value:
0.8763628979767241 and parameters: {'num_leaves': 115}. Best is trial 11 with
value: 0.8824083612697905.


num_leaves, val_score: 0.882408:  50%|#####     | 10/20 [00:46<00:48,
4.84s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  50%|#####     | 10/20 [00:48<00:48,
4.84s/it]

num_leaves, val_score: 0.884235:  55%|#####5    | 11/20 [00:48<00:35,
3.98s/it][I 2025-05-17 20:39:31,954] Trial 17 finished with value:
0.8842350296845054 and parameters: {'num_leaves': 5}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  55%|#####5    | 11/20 [00:48<00:35,
3.98s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  55%|#####5    | 11/20 [00:50<00:35,
3.98s/it]

num_leaves, val_score: 0.884235:  60%|######    | 12/20 [00:50<00:27,
3.44s/it][I 2025-05-17 20:39:34,155] Trial 18 finished with value:
0.8840966905007152 and parameters: {'num_leaves': 4}. Best is trial 17 with
value: 0.8842350296845054.
```

```
num_leaves, val_score: 0.884235:  60%|######    | 12/20 [00:50<00:27,
3.44s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  60%|######    | 12/20 [00:53<00:27,
3.44s/it]

num_leaves, val_score: 0.884235:  65%|######5   | 13/20 [00:53<00:23,
3.39s/it][I 2025-05-17 20:39:37,423] Trial 19 finished with value:
0.8786148512681126 and parameters: {'num_leaves': 51}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  65%|######5   | 13/20 [00:54<00:23,
3.39s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  65%|######5   | 13/20 [00:57<00:23,
3.39s/it]

num_leaves, val_score: 0.884235:  70%|#######   | 14/20 [00:57<00:21,
3.52s/it][I 2025-05-17 20:39:41,263] Trial 20 finished with value:
0.8773041605633031 and parameters: {'num_leaves': 40}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  70%|#######   | 14/20 [00:57<00:21,
3.52s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  70%|#######   | 14/20 [00:59<00:21,
3.52s/it]

num_leaves, val_score: 0.884235:  75%|#######5  | 15/20 [00:59<00:15,
3.05s/it][I 2025-05-17 20:39:43,231] Trial 21 finished with value:
0.8786403281352123 and parameters: {'num_leaves': 2}. Best is trial 17 with
value: 0.8842350296845054.
```

```
num_leaves, val_score: 0.884235:  75%|#######5  | 15/20 [00:59<00:15,
3.05s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  75%|#######5  | 15/20 [01:02<00:15,
3.05s/it]

num_leaves, val_score: 0.884235:  80%|########  | 16/20 [01:02<00:11,
2.82s/it][I 2025-05-17 20:39:45,517] Trial 22 finished with value:
0.8786403281352123 and parameters: {'num_leaves': 2}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  80%|########  | 16/20 [01:02<00:11,
2.82s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  80%|########  | 16/20 [01:05<00:11,
2.82s/it]

num_leaves, val_score: 0.884235:  85%|########5 | 17/20 [01:05<00:09,
3.05s/it][I 2025-05-17 20:39:49,084] Trial 23 finished with value:
0.8773041605633031 and parameters: {'num_leaves': 40}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  85%|########5 | 17/20 [01:05<00:09,
3.05s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  85%|########5 | 17/20 [01:08<00:09,
3.05s/it]

num_leaves, val_score: 0.884235:  90%|######### | 18/20 [01:08<00:05,
2.99s/it][I 2025-05-17 20:39:51,934] Trial 24 finished with value:
0.879339895860662 and parameters: {'num_leaves': 24}. Best is trial 17 with
value: 0.8842350296845054.
```

```
num_leaves, val_score: 0.884235:  90%|######### | 18/20 [01:08<00:05,
2.99s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  90%|######### | 18/20 [01:13<00:05,
2.99s/it]

num_leaves, val_score: 0.884235:  95%|#########5| 19/20 [01:13<00:03,
3.44s/it][I 2025-05-17 20:39:56,439] Trial 25 finished with value:
0.8779639165796267 and parameters: {'num_leaves': 77}. Best is trial 17 with
value: 0.8842350296845054.


num_leaves, val_score: 0.884235:  95%|#########5| 19/20 [01:13<00:03,
3.44s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


num_leaves, val_score: 0.884235:  95%|#########5| 19/20 [01:16<00:03,
3.44s/it]

num_leaves, val_score: 0.884235: 100%|##########| 20/20 [01:16<00:00,
3.49s/it][I 2025-05-17 20:40:00,040] Trial 26 finished with value:
0.8778498792940117 and parameters: {'num_leaves': 62}. Best is trial 17 with
value: 0.8842350296845054.
num_leaves, val_score: 0.884235: 100%|##########| 20/20 [01:16<00:00,  3.83s/it]


  0%|          | 0/10 [00:00<?, ?it/s]

bagging, val_score: 0.884235:   0%|          | 0/10 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.884503:   0%|          | 0/10 [00:02<?, ?it/s]

bagging, val_score: 0.884503:  10%|#         | 1/10 [00:02<00:26,
2.94s/it][I 2025-05-17 20:40:02,990] Trial 27 finished with value:
0.8845030527691584 and parameters: {'bagging_fraction': 0.6991179873200065,
'bagging_freq': 1}. Best is trial 27 with value: 0.8845030527691584.
```

```
bagging, val_score: 0.884503:  10%|#         | 1/10 [00:02<00:26,
2.94s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.884503:  10%|#         | 1/10 [00:05<00:26,
2.94s/it]

bagging, val_score: 0.884503:  20%|##        | 2/10 [00:05<00:22,
2.77s/it][I 2025-05-17 20:40:05,636] Trial 28 finished with value:
0.8844844905699145 and parameters: {'bagging_fraction': 0.6898969216948201,
'bagging_freq': 1}. Best is trial 27 with value: 0.8845030527691584.


bagging, val_score: 0.884503:  20%|##        | 2/10 [00:05<00:22,
2.77s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.884530:  20%|##        | 2/10 [00:07<00:22,
2.77s/it]

bagging, val_score: 0.884530:  30%|###       | 3/10 [00:07<00:18,
2.58s/it][I 2025-05-17 20:40:07,983] Trial 29 finished with value:
0.8845300668570649 and parameters: {'bagging_fraction': 0.6947900450527186,
'bagging_freq': 1}. Best is trial 29 with value: 0.8845300668570649.


bagging, val_score: 0.884530:  30%|###       | 3/10 [00:07<00:18,
2.58s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  30%|###       | 3/10 [00:10<00:18,
2.58s/it]

bagging, val_score: 0.885122:  40%|####      | 4/10 [00:10<00:14,
2.46s/it][I 2025-05-17 20:40:10,267] Trial 30 finished with value:
0.8851218139372605 and parameters: {'bagging_fraction': 0.6879109209649473,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.
```

```
bagging, val_score: 0.885122:  40%|####      | 4/10 [00:10<00:14,
2.46s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  40%|####      | 4/10 [00:12<00:14,
2.46s/it]

bagging, val_score: 0.885122:  50%|#####     | 5/10 [00:12<00:12,
2.55s/it][I 2025-05-17 20:40:12,980] Trial 31 finished with value:
0.8849202080796433 and parameters: {'bagging_fraction': 0.6928208089515608,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.


bagging, val_score: 0.885122:  50%|#####     | 5/10 [00:12<00:12,
2.55s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  50%|#####     | 5/10 [00:15<00:12,
2.55s/it]

bagging, val_score: 0.885122:  60%|######    | 6/10 [00:15<00:09,
2.44s/it][I 2025-05-17 20:40:15,189] Trial 32 finished with value:
0.8845877304519684 and parameters: {'bagging_fraction': 0.6862709480508,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.


bagging, val_score: 0.885122:  60%|######    | 6/10 [00:15<00:09,
2.44s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  60%|######    | 6/10 [00:17<00:09,
2.44s/it]

bagging, val_score: 0.885122:  70%|#######   | 7/10 [00:17<00:07,
2.43s/it][I 2025-05-17 20:40:17,599] Trial 33 finished with value:
0.8844496624195782 and parameters: {'bagging_fraction': 0.6713314907814028,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.
```

```
bagging, val_score: 0.885122:  70%|#######   | 7/10 [00:17<00:07,
2.43s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  70%|#######   | 7/10 [00:19<00:07,
2.43s/it]

bagging, val_score: 0.885122:  80%|########  | 8/10 [00:19<00:04,
2.42s/it][I 2025-05-17 20:40:19,992] Trial 34 finished with value:
0.8846420140531369 and parameters: {'bagging_fraction': 0.6999031237583604,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.


bagging, val_score: 0.885122:  80%|########  | 8/10 [00:19<00:04,
2.42s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  80%|########  | 8/10 [00:22<00:04,
2.42s/it]

bagging, val_score: 0.885122:  90%|######### | 9/10 [00:22<00:02,
2.36s/it][I 2025-05-17 20:40:22,244] Trial 35 finished with value:
0.8838750751563618 and parameters: {'bagging_fraction': 0.8249215395359706,
'bagging_freq': 1}. Best is trial 30 with value: 0.8851218139372605.


bagging, val_score: 0.885122:  90%|######### | 9/10 [00:22<00:02,
2.36s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


bagging, val_score: 0.885122:  90%|######### | 9/10 [00:25<00:02,
2.36s/it]

bagging, val_score: 0.885122: 100%|##########| 10/10 [00:25<00:00,
2.58s/it][I 2025-05-17 20:40:25,312] Trial 36 finished with value:
0.8831159425204245 and parameters: {'bagging_fraction': 0.5449483624916678,
'bagging_freq': 4}. Best is trial 30 with value: 0.8851218139372605.
```

```
bagging, val_score: 0.885122: 100%|##########| 10/10 [00:25<00:00,  2.53s/it]


  0%|          | 0/3 [00:00<?, ?it/s]

feature_fraction_stage2, val_score: 0.885122:   0%|          | 0/3 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction_stage2, val_score: 0.885122:   0%|          | 0/3 [00:02<?,
?it/s]

feature_fraction_stage2, val_score: 0.885122:  33%|###3      | 1/3 [00:02<00:05,
2.91s/it][I 2025-05-17 20:40:28,232] Trial 37 finished with value:
0.8849850427506366 and parameters: {'feature_fraction': 0.44800000000000006}.
Best is trial 37 with value: 0.8849850427506366.


feature_fraction_stage2, val_score: 0.885122:  33%|###3      | 1/3 [00:02<00:05,
2.91s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction_stage2, val_score: 0.885122:  33%|###3      | 1/3 [00:05<00:05,
2.91s/it]

feature_fraction_stage2, val_score: 0.885122:  67%|######6   | 2/3 [00:05<00:02,
2.59s/it][I 2025-05-17 20:40:30,601] Trial 38 finished with value:
0.8846952706699129 and parameters: {'feature_fraction': 0.48000000000000004}.
Best is trial 37 with value: 0.8849850427506366.


feature_fraction_stage2, val_score: 0.885122:  67%|######6   | 2/3 [00:05<00:02,
2.59s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


feature_fraction_stage2, val_score: 0.885122:  67%|######6   | 2/3 [00:07<00:02,
2.59s/it]

feature_fraction_stage2, val_score: 0.885122: 100%|##########| 3/3 [00:07<00:00,
2.46s/it][I 2025-05-17 20:40:32,908] Trial 39 finished with value:
0.8851218139372605 and parameters: {'feature_fraction': 0.41600000000000004}.
Best is trial 39 with value: 0.8851218139372605.
feature_fraction_stage2, val_score: 0.885122: 100%|##########| 3/3 [00:07<00:00,
2.53s/it]


  0%|          | 0/20 [00:00<?, ?it/s]

regularization_factors, val_score: 0.885122:   0%|          | 0/20 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.885122:   0%|          | 0/20 [00:02<?,
?it/s]

regularization_factors, val_score: 0.885122:   5%|5         | 1/20 [00:02<00:56,
2.96s/it][I 2025-05-17 20:40:35,876] Trial 40 finished with value:
0.8845594646692428 and parameters: {'lambda_l1': 0.006715080016653908,
'lambda_l2': 7.033668784265104e-06}. Best is trial 40 with value:
0.8845594646692428.


regularization_factors, val_score: 0.885122:   5%|5         | 1/20 [00:02<00:56,
2.96s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886033:   5%|5         | 1/20 [00:05<00:56,
2.96s/it]

regularization_factors, val_score: 0.886033:  10%|#         | 2/20 [00:05<00:46,
2.58s/it][I 2025-05-17 20:40:38,185] Trial 41 finished with value:
0.8860328124277815 and parameters: {'lambda_l1': 5.962957689728475e-08,
'lambda_l2': 7.364494123592756}. Best is trial 41 with value:
0.8860328124277815.


regularization_factors, val_score: 0.886033:  10%|#         | 2/20 [00:05<00:46,
2.58s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
regularization_factors, val_score: 0.886707:  10%|#         | 2/20 [00:07<00:46,
2.58s/it]

regularization_factors, val_score: 0.886707:  15%|#5        | 3/20 [00:07<00:41,
2.42s/it][I 2025-05-17 20:40:40,420] Trial 42 finished with value:
0.8867068250859201 and parameters: {'lambda_l1': 1.4619627113024688e-08,
'lambda_l2': 7.830054025447685}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  15%|#5        | 3/20 [00:07<00:41,
2.42s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  15%|#5        | 3/20 [00:09<00:41,
2.42s/it]

regularization_factors, val_score: 0.886707:  20%|##        | 4/20 [00:09<00:38,
2.39s/it][I 2025-05-17 20:40:42,770] Trial 43 finished with value:
0.8860668823493608 and parameters: {'lambda_l1': 1.7036489088261837e-08,
'lambda_l2': 8.885164541968038}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  20%|##        | 4/20 [00:09<00:38,
2.39s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  20%|##        | 4/20 [00:13<00:38,
2.39s/it]

regularization_factors, val_score: 0.886707:  25%|##5       | 5/20 [00:13<00:40,
2.70s/it][I 2025-05-17 20:40:46,012] Trial 44 finished with value:
0.886283103169164 and parameters: {'lambda_l1': 1.0078748015971085e-08,
'lambda_l2': 8.327798351391525}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  25%|##5       | 5/20 [00:13<00:40,
2.70s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
```

```
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  25%|##5       | 5/20 [00:15<00:40,
2.70s/it]

regularization_factors, val_score: 0.886707:  30%|###       | 6/20 [00:15<00:36,
2.58s/it][I 2025-05-17 20:40:48,363] Trial 45 finished with value:
0.8860363629508065 and parameters: {'lambda_l1': 1.1208101167456596e-08,
'lambda_l2': 5.271405600277059}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  30%|###       | 6/20 [00:15<00:36,
2.58s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  30%|###       | 6/20 [00:18<00:36,
2.58s/it]

regularization_factors, val_score: 0.886707:  35%|###5      | 7/20 [00:18<00:33,
2.57s/it][I 2025-05-17 20:40:50,919] Trial 46 finished with value:
0.8859480836967585 and parameters: {'lambda_l1': 1.2715263505638996e-08,
'lambda_l2': 7.2237717887996595}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  35%|###5      | 7/20 [00:18<00:33,
2.57s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  35%|###5      | 7/20 [00:20<00:33,
2.57s/it]

regularization_factors, val_score: 0.886707:  40%|####      | 8/20 [00:20<00:30,
2.52s/it][I 2025-05-17 20:40:53,314] Trial 47 finished with value:
0.8858095496574535 and parameters: {'lambda_l1': 1.141977350078899e-08,
'lambda_l2': 6.947767156047918}. Best is trial 42 with value:
0.8867068250859201.
```

```
regularization_factors, val_score: 0.886707:  40%|####      | 8/20 [00:20<00:30,
2.52s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
num_leaves, val_score: 0.884235:  50%|#####     | 10/20 [05:14<05:14, 31.40s/it]


regularization_factors, val_score: 0.886707:  40%|####      | 8/20 [00:23<00:30,
2.52s/it]

regularization_factors, val_score: 0.886707:  45%|####5     | 9/20 [00:23<00:30,
2.80s/it][I 2025-05-17 20:40:56,744] Trial 48 finished with value:
0.8856819625419883 and parameters: {'lambda_l1': 1.4022102814516587e-08,
'lambda_l2': 9.067745633148425}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  45%|####5     | 9/20 [00:23<00:30,
2.80s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  45%|####5     | 9/20 [00:26<00:30,
2.80s/it]

regularization_factors, val_score: 0.886707:  50%|#####     | 10/20
[00:26<00:26,  2.67s/it][I 2025-05-17 20:40:59,125] Trial 49 finished with
value: 0.8857398635018691 and parameters: {'lambda_l1': 1.1367317720112154e-08,
'lambda_l2': 6.942880635441359}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  50%|#####     | 10/20
[00:26<00:26,
2.67s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  50%|#####     | 10/20
[00:28<00:26,  2.67s/it]

regularization_factors, val_score: 0.886707:  55%|#####5    | 11/20
```

```
[00:28<00:23,  2.59s/it][I 2025-05-17 20:41:01,531] Trial 50 finished with
value: 0.885872909766097 and parameters: {'lambda_l1': 1.2766782408964144e-07,
'lambda_l2': 7.001281645161147}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  55%|#####5    | 11/20
[00:28<00:23,
2.59s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  55%|#####5    | 11/20
[00:31<00:23,  2.59s/it]

regularization_factors, val_score: 0.886707:  60%|######    | 12/20
[00:31<00:20,  2.57s/it][I 2025-05-17 20:41:04,061] Trial 51 finished with
value: 0.8860076453514637 and parameters: {'lambda_l1': 1.2972037957493148e-08,
'lambda_l2': 8.651445415945725}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  60%|######    | 12/20
[00:31<00:20,
2.57s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  60%|######    | 12/20
[00:34<00:20,  2.57s/it]

regularization_factors, val_score: 0.886707:  65%|######5   | 13/20
[00:34<00:19,  2.84s/it][I 2025-05-17 20:41:07,516] Trial 52 finished with
value: 0.8856652370341163 and parameters: {'lambda_l1': 6.695290538482802e-08,
'lambda_l2': 4.420357418925196}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  65%|######5   | 13/20
[00:34<00:19,
2.84s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
regularization_factors, val_score: 0.886707:  65%|######5   | 13/20
[00:37<00:19,  2.84s/it]

regularization_factors, val_score: 0.886707:  70%|#######   | 14/20
[00:37<00:16,  2.74s/it][I 2025-05-17 20:41:10,022] Trial 53 finished with
value: 0.8850525173084157 and parameters: {'lambda_l1': 7.275344039890602e-07,
'lambda_l2': 0.27470530459445136}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  70%|#######   | 14/20
[00:37<00:16,
2.74s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  70%|#######   | 14/20
[00:39<00:16,  2.74s/it]

regularization_factors, val_score: 0.886707:  75%|#######5  | 15/20
[00:39<00:13,  2.61s/it][I 2025-05-17 20:41:12,318] Trial 54 finished with
value: 0.8851680992478504 and parameters: {'lambda_l1': 4.872838664016751e-06,
'lambda_l2': 0.03989342804460906}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  75%|#######5  | 15/20
[00:39<00:13,
2.61s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  75%|#######5  | 15/20
[00:41<00:13,  2.61s/it]

regularization_factors, val_score: 0.886707:  80%|######## | 16/20
[00:41<00:10,  2.53s/it][I 2025-05-17 20:41:14,660] Trial 55 finished with
value: 0.885385092056699 and parameters: {'lambda_l1': 2.1670936752144563e-08,
'lambda_l2': 0.18154257885372585}. Best is trial 42 with value:
0.8867068250859201.
```

```
regularization_factors, val_score: 0.886707:  80%|########  | 16/20
[00:41<00:10,
2.53s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  80%|########  | 16/20
[00:44<00:10,  2.53s/it]

regularization_factors, val_score: 0.886707:  85%|########5 | 17/20
[00:44<00:08,  2.69s/it][I 2025-05-17 20:41:17,736] Trial 56 finished with
value: 0.8845010416169853 and parameters: {'lambda_l1': 6.89902659229063e-07,
'lambda_l2': 0.28950684709744046}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  85%|########5 | 17/20
[00:44<00:08,
2.69s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  85%|########5 | 17/20
[00:47<00:08,  2.69s/it]

regularization_factors, val_score: 0.886707:  90%|######### | 18/20
[00:47<00:05,  2.60s/it][I 2025-05-17 20:41:20,113] Trial 57 finished with
value: 0.8841830736607167 and parameters: {'lambda_l1': 8.932510071239893,
'lambda_l2': 0.4725059790304198}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  90%|######### | 18/20
[00:47<00:05,
2.60s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  90%|######### | 18/20
[00:49<00:05,  2.60s/it]

regularization_factors, val_score: 0.886707:  95%|#########5| 19/20
```

```
[00:49<00:02,  2.53s/it][I 2025-05-17 20:41:22,492] Trial 58 finished with
value: 0.8847930141767093 and parameters: {'lambda_l1': 1.9404593240889336e-07,
'lambda_l2': 0.0009288159394981387}. Best is trial 42 with value:
0.8867068250859201.


regularization_factors, val_score: 0.886707:  95%|#########5| 19/20
[00:49<00:02,
2.53s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(


regularization_factors, val_score: 0.886707:  95%|#########5| 19/20
[00:52<00:02,  2.53s/it]

regularization_factors, val_score: 0.886707: 100%|##########| 20/20
[00:52<00:00,  2.50s/it][I 2025-05-17 20:41:24,921] Trial 59 finished with
value: 0.8851218139372605 and parameters: {'lambda_l1': 1.0065620632728834e-08,
'lambda_l2': 1.1749010343263647e-08}. Best is trial 42 with value:
0.8867068250859201.
regularization_factors, val_score: 0.886707: 100%|##########| 20/20
[00:52<00:00,  2.60s/it]
min_child_samples, val_score: 0.886707:   0%|          | 0/5 [00:00<?,
?it/s]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
min_child_samples, val_score: 0.886707:  20%|##        | 1/5 [00:02<00:10,
2.51s/it][I 2025-05-17 20:41:27,432] Trial 60 finished with value:
0.8858991029053596 and parameters: {'min_child_samples': 50}. Best is trial 60
with value: 0.8858991029053596.
min_child_samples, val_score: 0.886707:  20%|##        | 1/5 [00:02<00:10,
2.51s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
min_child_samples, val_score: 0.886707:  40%|####      | 2/5 [00:05<00:08,
2.67s/it][I 2025-05-17 20:41:30,220] Trial 61 finished with value:
0.8853823713896474 and parameters: {'min_child_samples': 100}. Best is trial 60
with value: 0.8858991029053596.
min_child_samples, val_score: 0.886707:  40%|####      | 2/5 [00:05<00:08,
2.67s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
min_child_samples, val_score: 0.886707:  60%|######    | 3/5 [00:07<00:04,
2.27s/it][I 2025-05-17 20:41:32,009] Trial 62 finished with value:
0.886152051593907 and parameters: {'min_child_samples': 5}. Best is trial 62
with value: 0.886152051593907.
min_child_samples, val_score: 0.886707:  60%|######    | 3/5 [00:07<00:04,
2.27s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
min_child_samples, val_score: 0.886707:  80%|########  | 4/5 [00:08<00:02,
2.08s/it][I 2025-05-17 20:41:33,796] Trial 63 finished with value:
0.8862019982469884 and parameters: {'min_child_samples': 10}. Best is trial 63
with value: 0.8862019982469884.
min_child_samples, val_score: 0.886707:  80%|########  | 4/5 [00:08<00:02,
2.08s/it]/home/sergey/PycharmProjects/ML_UNI/.venv/lib/python3.13/site-
packages/sklearn/model_selection/_split.py:877: UserWarning: The groups
parameter is ignored by StratifiedKFold
  warnings.warn(
min_child_samples, val_score: 0.886707: 100%|##########| 5/5 [00:10<00:00,
1.94s/it][I 2025-05-17 20:41:35,496] Trial 64 finished with value:
0.8859250948004022 and parameters: {'min_child_samples': 25}. Best is trial 63
with value: 0.8862019982469884.
min_child_samples, val_score: 0.886707: 100%|##########| 5/5 [00:10<00:00,
2.11s/it]

TunerCV best params: {'objective': 'binary', 'metric': 'auc', 'verbosity': -1,
'seed': 42, 'feature_pre_filter': False, 'lambda_l1': 1.4619627113024688e-08,
'lambda_l2': 7.830054025447685, 'num_leaves': 5, 'feature_fraction': 0.4,
'bagging_fraction': 0.6879109209649473, 'bagging_freq': 1, 'min_child_samples':
20}
TunerCV ROC-AUC on val: 0.8854517553709109
```

[ ]:

```
from sklearn.model_selection import cross_val_score

def objective(trial):
    params = {
        'n_estimators':      trial.suggest_int('n_estimators', 50, 500),
        'max_depth':         trial.suggest_int('max_depth', 3, 12),
        'num_leaves':        trial.suggest_int('num_leaves', 20, 200),
        'learning_rate':     trial.suggest_loguniform('learning_rate', 1e-3, 1.
 ↪0),
        'min_child_samples': trial.suggest_int('min_child_samples', 5, 100),
        'subsample':         trial.suggest_uniform('subsample', 0.5, 1.0),
        'colsample_bytree':  trial.suggest_uniform('colsample_bytree', 0.5, 1.
 ↪0),
        'reg_alpha':         trial.suggest_loguniform('reg_alpha', 1e-8, 10.0),
        'reg_lambda':        trial.suggest_loguniform('reg_lambda', 1e-8, 10.
 ↪0),
        'random_state':      42,
        'objective':          'binary',
        'verbosity':          -1
    }
    model = LGBMClassifier(**params)
    cv = StratifiedKFold(n_splits=3, shuffle=True, random_state=42)
    scores = cross_val_score(model, X_tr, y_tr, cv=cv, scoring='roc_auc',␣
  ↪n_jobs=-1)
    return scores.mean()

study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=50, timeout=600)

best_params_cv2 = study.best_params
print("Classic Optuna best params:", best_params_cv2)

final2 = LGBMClassifier(**best_params_cv2)
final2.fit(
    np.vstack([X_tr, X_val]),
    np.concatenate([y_tr, y_val])
)
val_preds_cv2 = final2.predict_proba(X_val)[:, 1]
preds_cv2 = final2.predict_proba(X_test)[:, 1]

print("TunerCV ROC-AUC on val:", roc_auc_score(y_val, val_preds_cv1))
print("Classic Optuna ROC-AUC on val:", roc_auc_score(y_val, val_preds_cv2))
```

[35]:

```
import pandas as pd

sample = pd.read_csv('mai-ml-lab-2/sample_submission.csv')

sample['smoking'] = preds_cv

sample.to_csv('submission.csv', index=False)

print("submission.csv    :", sample.head())
```

```
submission.csv    :        id   smoking
0  15000  0.044976
1  15001  0.820014
2  15002  0.206059
3  15003  0.720090
4  15004  0.015359
```