Problem 1

• $u^T * u$

```
[[26]]
```

• $u * u^T$

```
[[16  4 12]
 [ 4  1  3]
 [12  3  9]]
```

• $v * u$

```
[71]
```

• $u + 5$

```
[[9]
 [6]
 [8]]
```

• A^T

```
[[1 3]
 [2 4]
 [5 6]]
```

• $B * u$

```
[[56]
 [19]
 [42]]
```

• B^{-1}

```
[[ 1.00000000e+00 -5.50000000e+00  1.25000000e+00]
 [ 1.49453099e-17 -5.00000000e-01  2.50000000e-01]
 [-6.66666667e-01  4.33333333e+00 -1.00000000e+00]]
```

• $B + C$

```
[[15  7 14]
 [ 3 -1  7]
 [ 3  6 10]]
```

• $B - C$

[[-1 -5 4]

[1 5 -1]

[5 10 2]]

• $A * B$

[[31 45 45]

[53 59 75]]

• $B * C$

[[48 21 75]

[15   0 30]

[34 -12 76]]

• $B * A$

Cannot calculate, the shapes are not compatible, as given by Python error:

“ValueError: shapes (3,3) and (2,3) not aligned: 3 (dim 1) != 2 (dim 0)”

Problem 2

(a) & (b)

	preg	plas	pres	skin	test	mass	pedi	age
min	0	0	0	0	0	0	0.078	21
max	17	199	122	99	846	67.1	2.42	81
mean	3.84505208	120.89453125	69.10546875	20.53645833	79.79947917	31.99257812	0.4718763	33.24088542
std	3.36738361	31.95179591	19.34320163	15.94182863	115.16894926	7.87902573	0.33111282	11.75257265

(c)

(1) the set with class label 0

the mean for class 0 attributes:

preg 3.298000

plas 109.980000

pres 68.184000

skin 19.664000

test 68.792000

mass 30.304200
pedi 0.429734
age 31.190000
class 0.000000

the std for class 0 attributes:
preg 3.014166
plas 26.115045
pres 18.045003
skin 14.875050
test 98.766375
mass 7.682161
pedi 0.298786
age 11.655981
class 0.000000

(2) the set with class label 1
the mean for class 1 attributes:
preg 4.865672
plas 141.257463
pres 70.824627
skin 22.164179
test 100.335821
mass 35.142537
pedi 0.550500
age 37.067164
class 1.000000

the std for class 1 attributes:
preg 3.734253

plas 31.879978

pres 21.451678

skin 17.646696

test 138.430135

mass 7.249404

pedi 0.371659

age 10.947771

class 0.000000

The attribute that I think can help distinguish these two sets best is the “test” attribute. Compare the mean and standard deviation of these two subsets, the test attribute is the one that varies the most between the sets for both mean and standard deviation (mean: 68.792 in class 0 and 100.335821 in class 1; standard deviation: 98.766375 in class 0 and 138.430135 in class 1). Based on this significant difference, I think “test” should work better.

(d) the correlation between each of the attribute with the class label:

preg 0.22189815

plas 0.4665814

pres 0.06506836

skin 0.07475223

test 0.13054795

mass 0.29269466

pedi 0.17384407

age 0.23835598

Thus, the “plas” has the highest positive correlation (with value 0.4665814) to the target class attribute. I think it’s the best for predicting because high correlation means these two attributes are highly related.

(e) mutual correlations between attributes:

preg vs. plas: 0.12945867149927245

preg vs. pres: 0.14128197740713996

preg vs. skin: -0.08167177444900725

preg vs. test: -0.07353461435162816
preg vs. mass: 0.01768309072783063
preg vs. pedi: -0.03352267296261308
preg vs. age: 0.5443412284023389

plas vs. pres: 0.15258958656866448
plas vs. skin: 0.05732789073817711
plas vs. test: 0.3313571099202094
plas vs. mass: 0.22107106945898297
plas vs. pedi: 0.13733729982837073
plas vs. age: 0.26351431982433354

pres vs. skin: 0.207370538403071
pres vs. test: 0.08893337837319312
pres vs. mass: 0.28180528884991063
pres vs. pedi: 0.04126494793009856 4
pres vs. age: 0.23952794642136352

skin vs. test: 0.436782570120014
skin vs. mass: 0.39257320415903857
skin vs. pedi: 0.18392757295416337
skin vs. age: -0.1139702623677417

test vs. mass: 0.1978590564931011
test vs. pedi: 0.18507092916809914
test vs. age: -0.04216295473537688

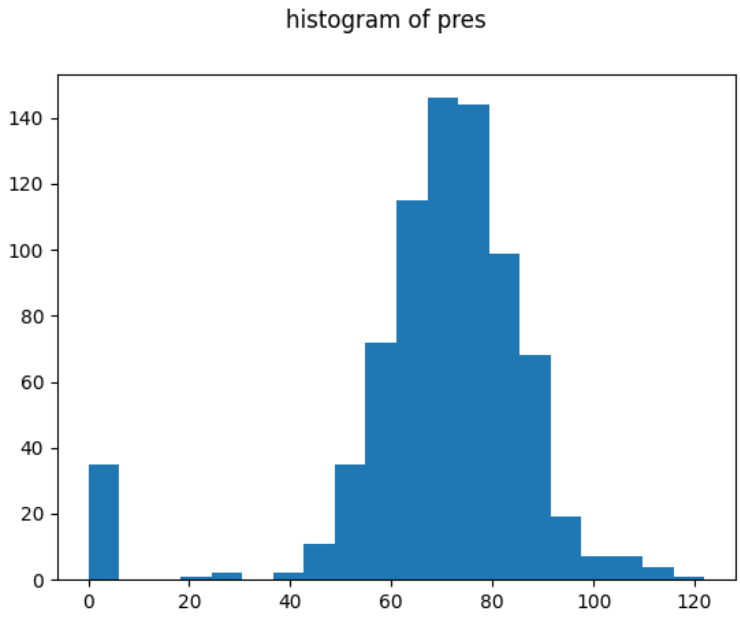
mass vs. pedi: 0.1406469525451052
mass vs. age: 0.036241870092294126

pedi vs. age: 0.03356131243480553

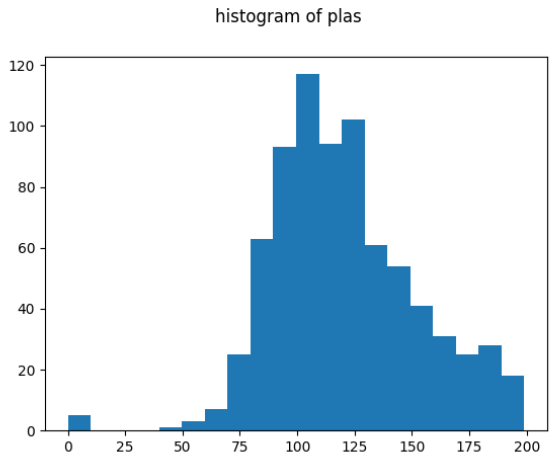
Based on the correlation data, the largest mutual correlation is 0.5443412284023389 between “preg” and “age.”

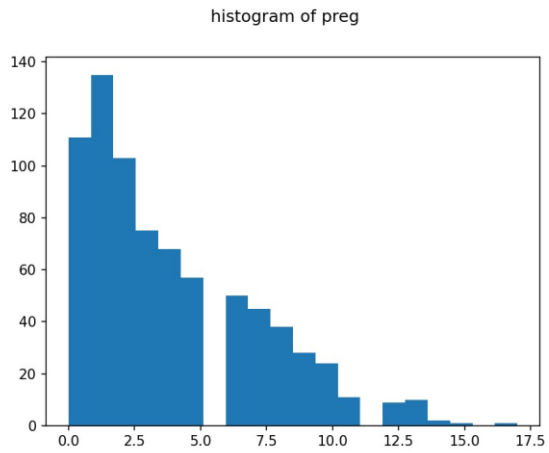
(f)

The most normally distributed histogram of attributes is pres:

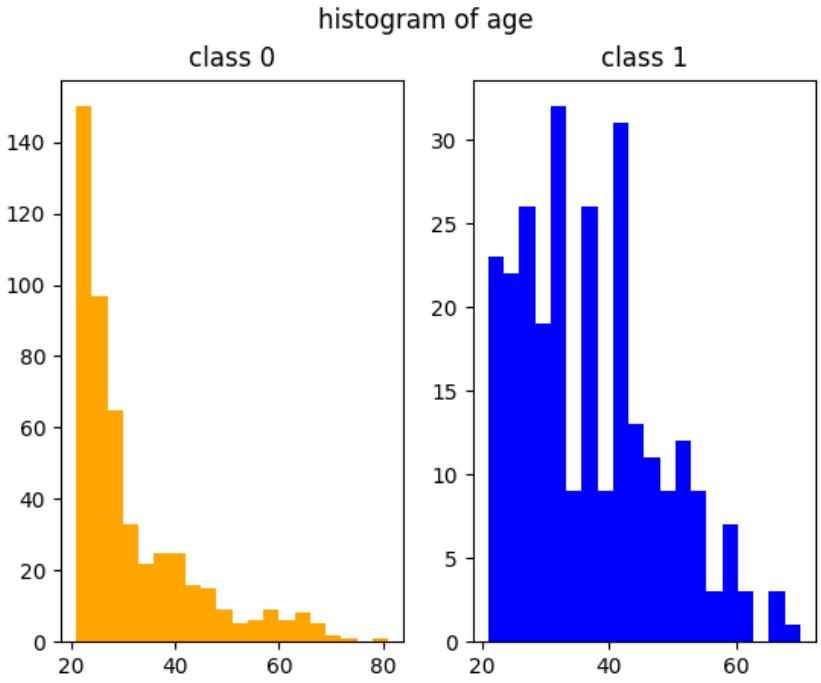


Another 2 histograms I included here:

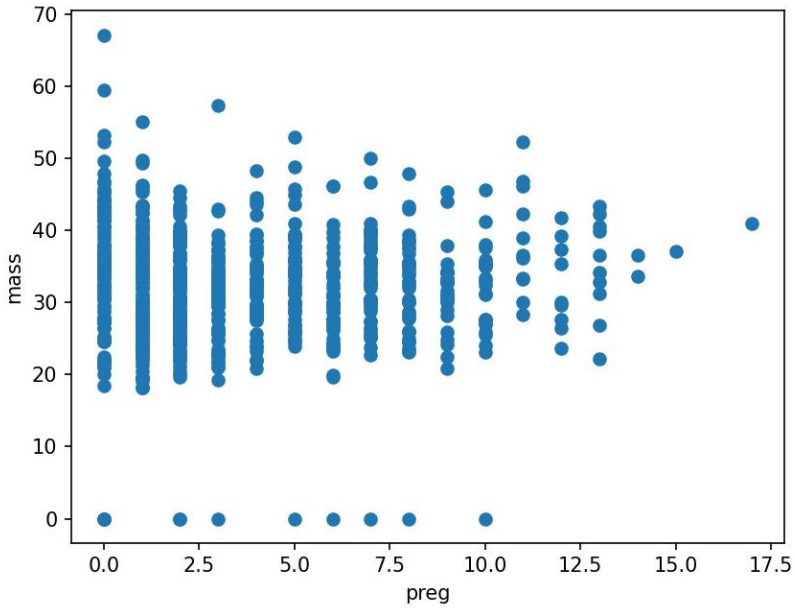




(g) I think the attribute, “age,” discriminates the two sets the best because as shown in the histogram below, the distribution of age in class 0 and 1 are very different, especially compared with the distributions of other attributes.

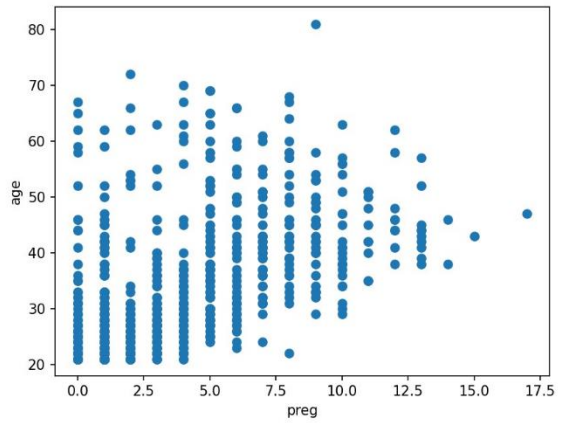


(h) If two attributes are more independent and random, in the scatter plot, I expect points display less separately along a vertical/horizontal line. Also, according to the correlation value we calculated before, this can be verified. For example, the weakest correlation is between preg and mass (as shown below), the points are near each other on vertical lines.



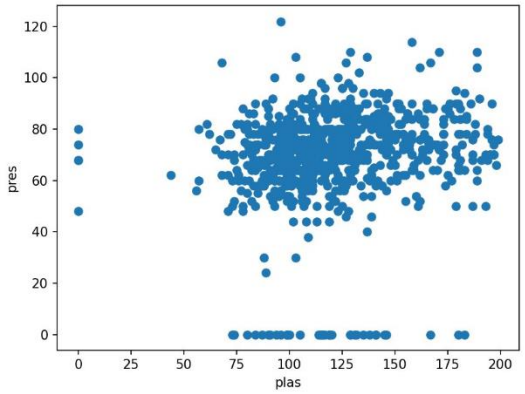
2 interesting scatter plots I see:

(1) preg vs. age



preg vs. age has the strongest correlation among mutual correlations of all attributes (around 0.544, largest correlation value calculated previously). In the plot, the points are more separated and there is a general direction of going up indicating positive correlation. As the preg value increases, there are almost no points on the right-bottom corner, forming a clean triangle. Thus, we can infer that the correlation is strong.

(2) plas vs. pres



In this scatter plot, we can infer that the correlation between plas and pres is positive, but the points clustering does not have a clear pattern, which means that the correlation is weak. There are some outliers near the axes, and they are probably measuring errors.

Problem 3

(a) one-hot encoding design:

for each of the 8 values, {brown, blue, white, red, yellow, orange, green, black}, use an array of size 8 with the value at the corresponding index be 1 and 0's at other indices. Here are the mappings that illustrate this idea:

first, encode for "brown." In the original categorical array, brown is at index 0. Thus, in correspondence, we put "1" at index 0 in the one-hot encoding for brown and leave other indices as 0: (same logic follows for other categories)

brown ------ [1 0 0 0 0 0 0 0]

blue ------- [0 1 0 0 0 0 0 0]

white ------- [0 0 1 0 0 0 0 0]

red ------- [0 0 0 1 0 0 0 0]

yellow ------- [0 0 0 0 1 0 0 0]

orange ------- [0 0 0 0 0 1 0 0]

green ------- [0 0 0 0 0 0 1 0]

black ------- [0 0 0 0 0 0 0 1]

Use the above mappings, the vector can be represented by one-hot as:

[[0 0 0 1 0 0 0 0]

[0 0 0 0 0 0 0 1]

[0 0 0 0 1 0 0 0]

[0 0 0 1 0 0 0 0]

[0 0 0 0 0 0 1 0]

[0 1 0 0 0 0 0 0]

[0 1 0 0 0 0 0 0]]

(b) the new normalized values for the attribute “pres:” (the first five entries)

0.14964075

-0.16054575

-0.26394125

-0.16054575

-1.50468724

(c) the first five entries of the discretized data of attribute “pres:”

5, 3, 3, 3, 0

Problem 4

(a) train set size: 514

test set size: 254

(b) train set size: 514

test set size: 254

Yes, the data instances in (a) and (b) are the same.

(c) train set size: 576

test set size: 192