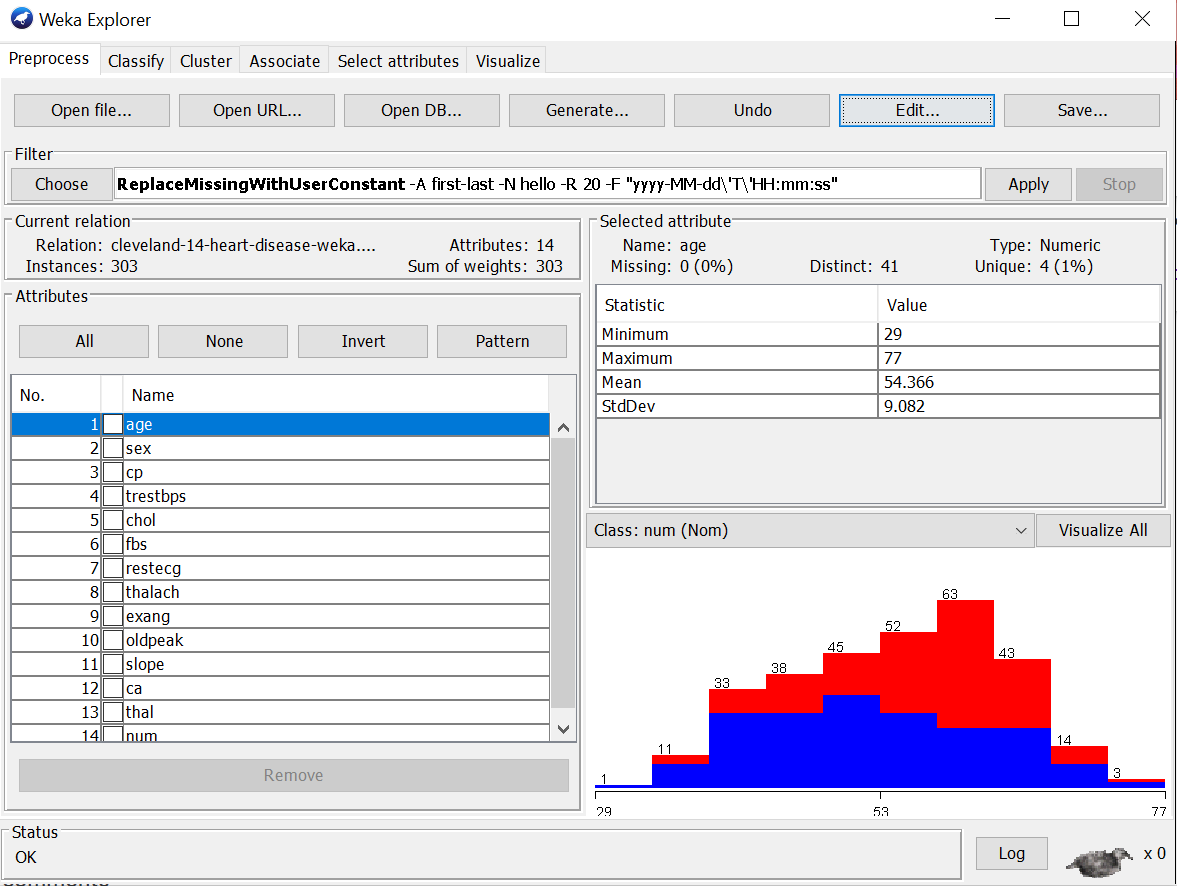
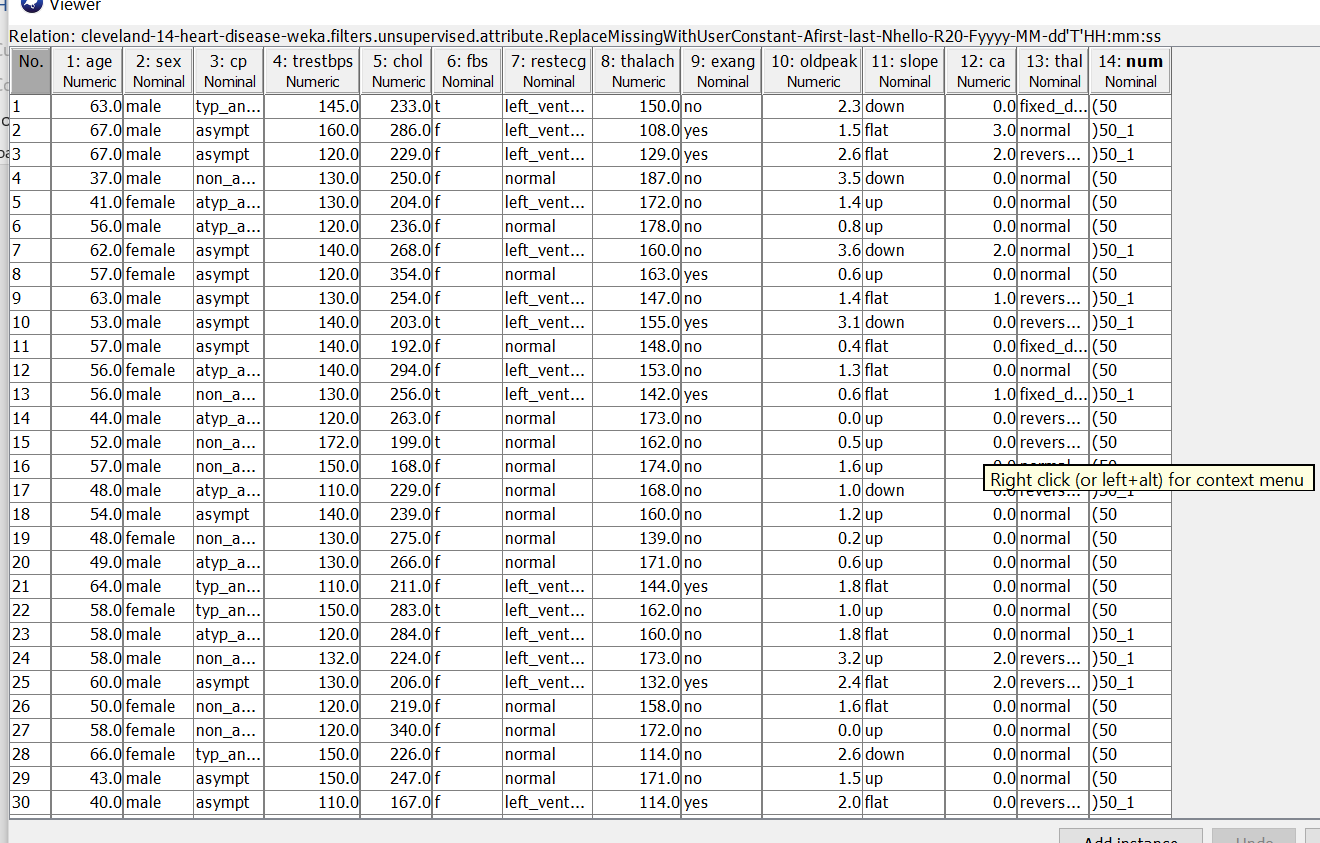
DMWA Lab

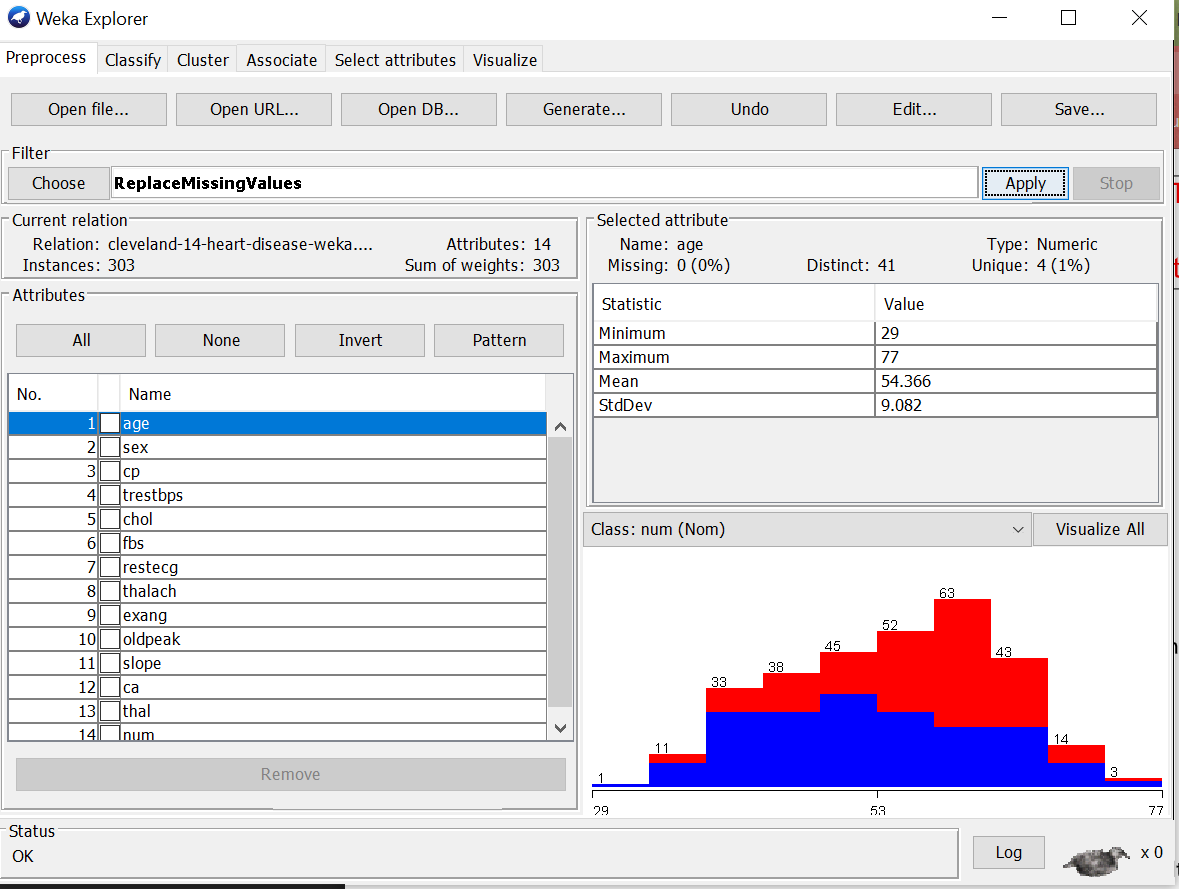
WEEK-4

Qs-1

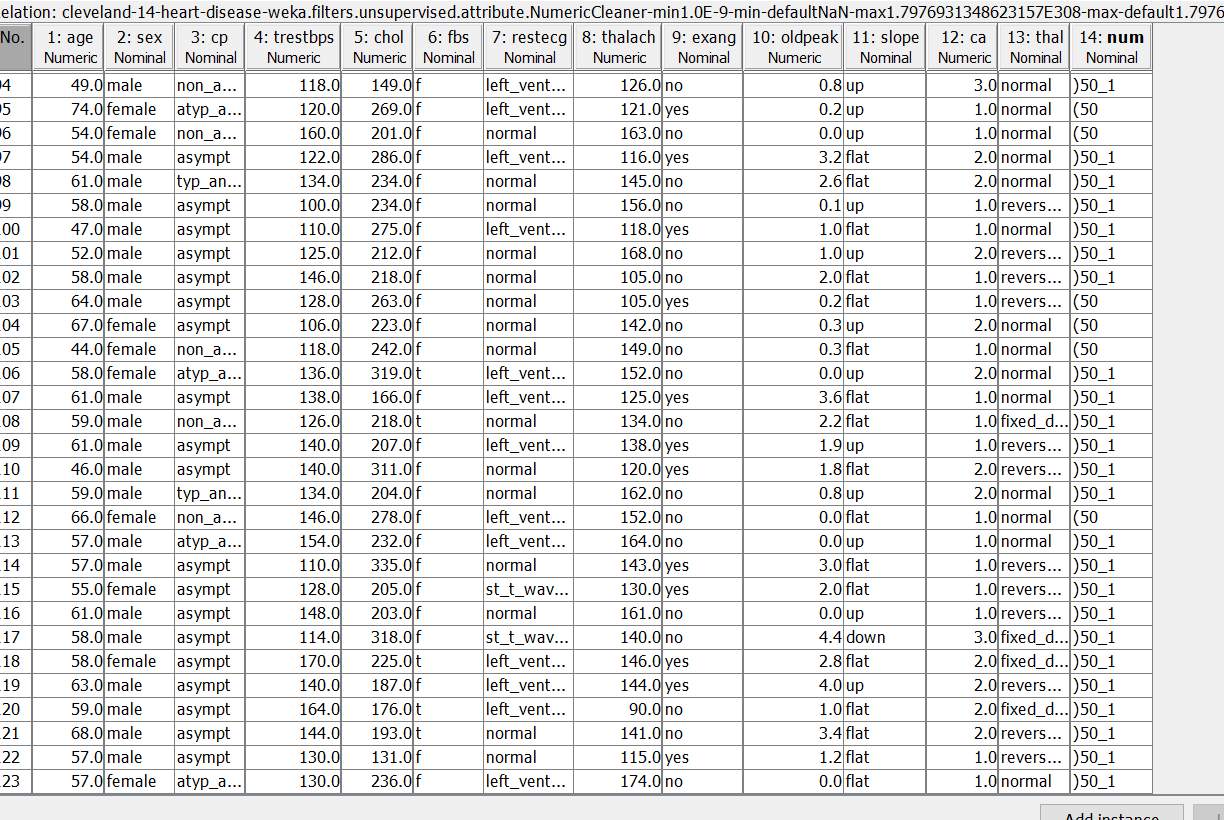
a) ReplaceMissingWithUserConstant

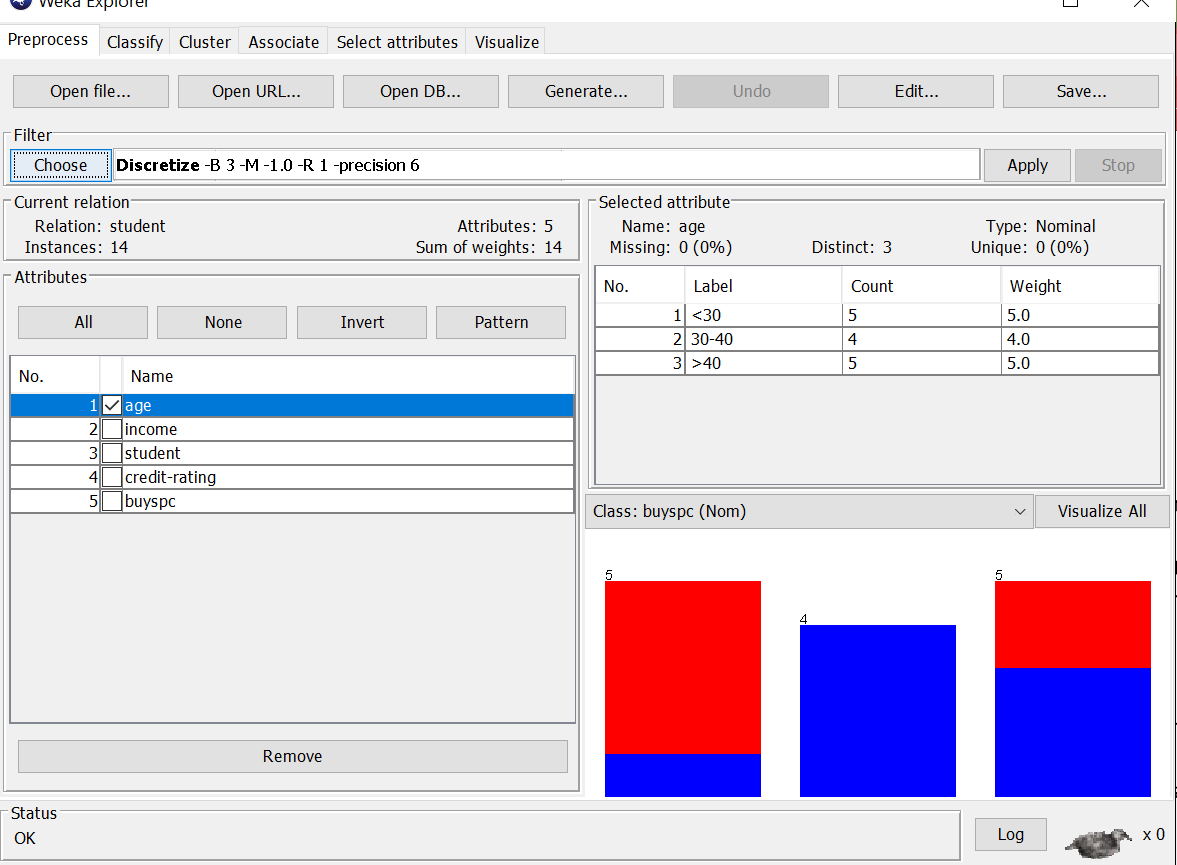


b) ReplaceMissingValues

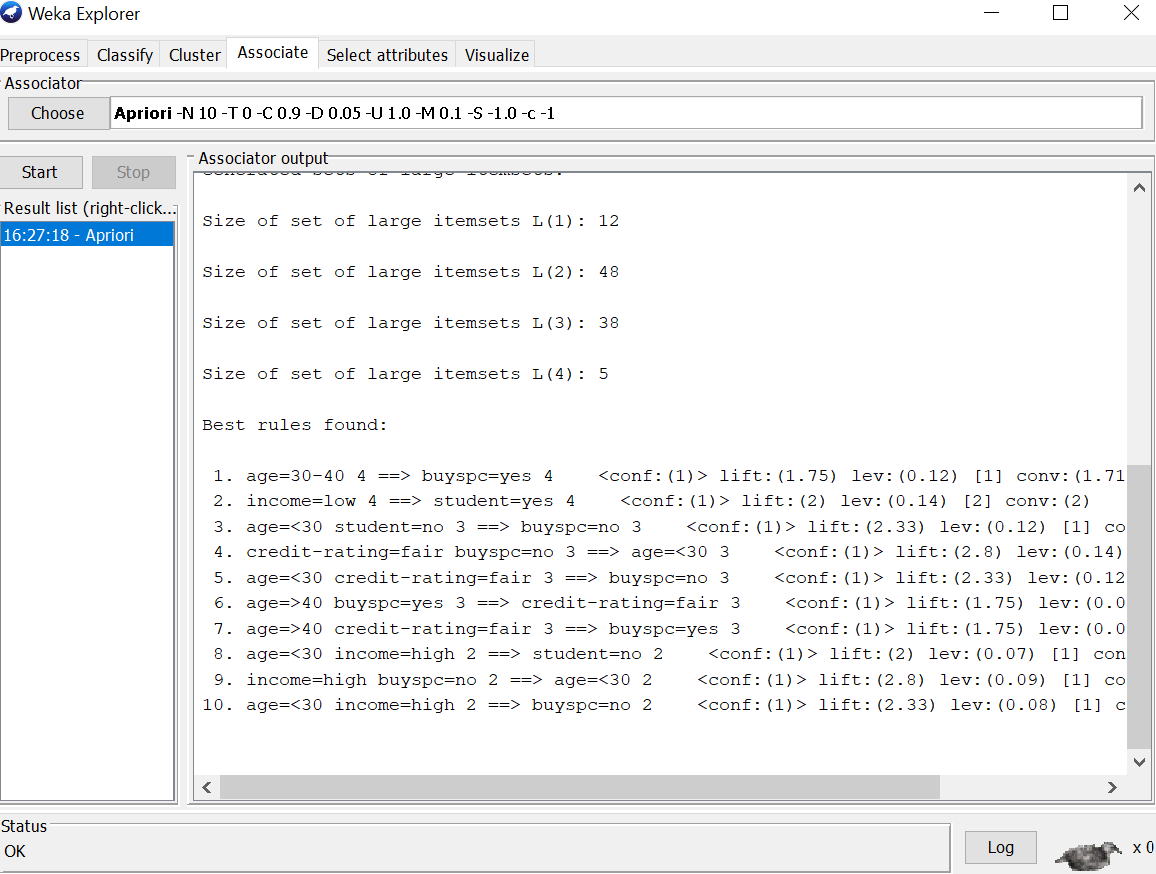


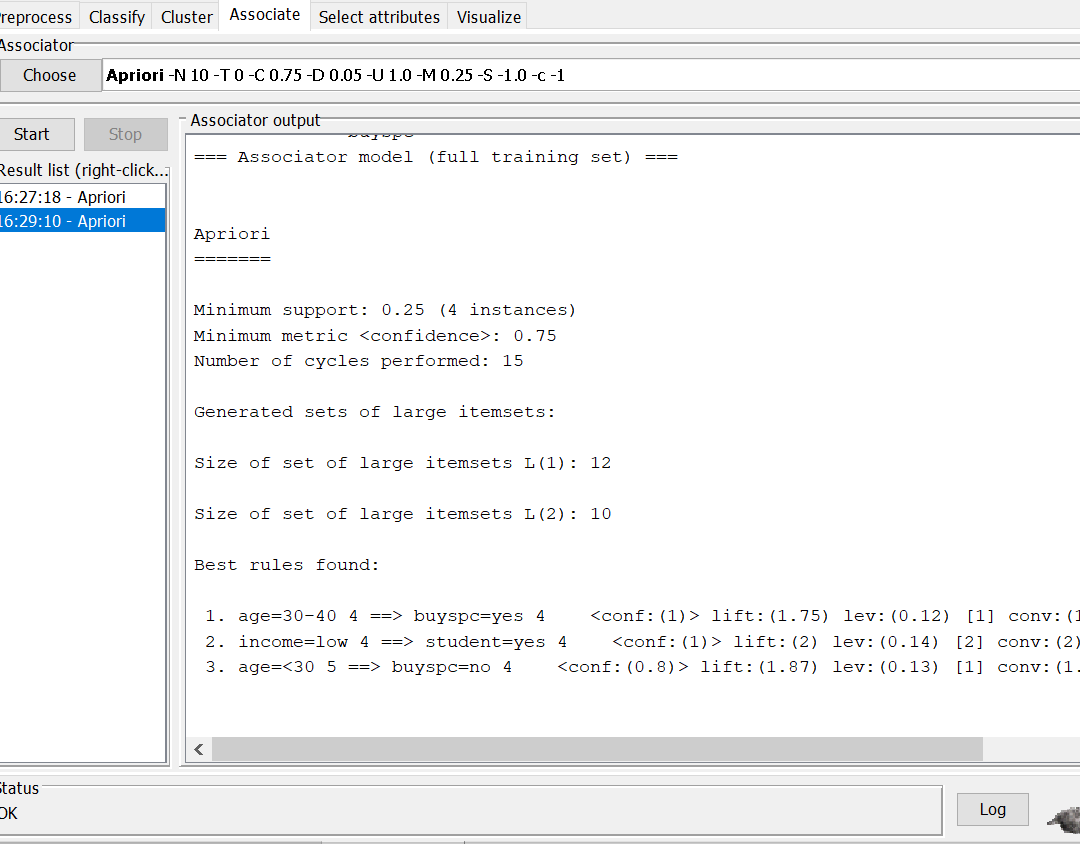
c) RemoveMissingValues



Qs-2 

Qs-3





Qs-4

a)

n\_num = [1, 2, 3, 4, 5]

n = len(n\_num)

get\_sum = sum(n\_num)

mean = get\_sum / n

print("Mean / Average is: " + str(mean))

n\_num.sort()

if n % 2 == 0:

    median1 = n\_num[n//2]

    median2 = n\_num[n//2 - 1]

    median = (median1 + median2)/2

else:

    median = n\_num[n//2]

print("Median is: " + str(median))

data = Counter(n\_num)

get\_mode = dict(data)

mode = [k for k, v in get\_mode.items() if v == max(list(data.values()))]

if len(mode) == n:

    get\_mode = "No mode found"

else:

    get\_mode = "Mode is / are: " + ', '.join(map(str, mode))

print(get\_mode)

b)

a = dataset.data

b = np.zeros(150)

for i in range (150):

    b[i]=a[i,1]

b=np.sort(b)  #sort the array

bin1=np.zeros((30,5))

bin2=np.zeros((30,5))

bin3=np.zeros((30,5))

for i in range (0,150,5):

    k=int(i/5)

    mean=(b[i] + b[i+1] + b[i+2] + b[i+3] + b[i+4])/5

    for j in range(5):

        bin1[k,j]=mean

print("Bin Mean: \n",bin1)

for i in range (0,150,5):

    k=int(i/5)

    for j in range (5):

        if (b[i+j]-b[i]) < (b[i+4]-b[i+j]):

            bin2[k,j]=b[i]

        else:

            bin2[k,j]=b[i+4]

print("Bin Boundaries: \n",bin2)

for i in range (0,150,5):

    k=int(i/5)

    for j in range (5):

        bin3[k,j]=b[i+2]

print("Bin Median: \n",bin3)