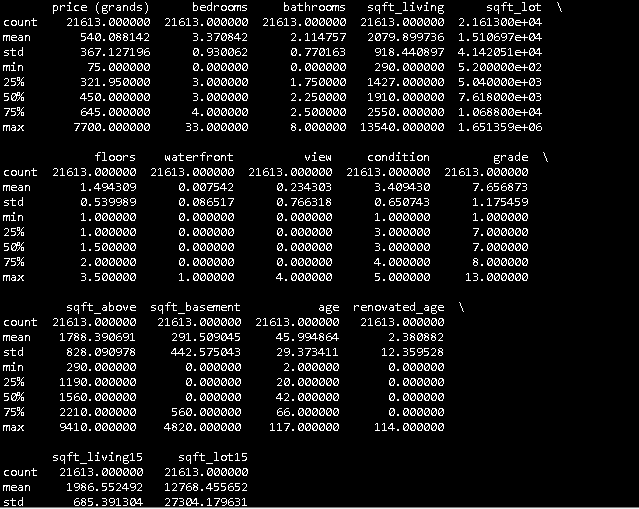
**Solutions and Screenshots**

**A1**

**1.1**

A data frame is a labeled two-dimensional data structure containing columns that can be of different types.



**1.2** In python file

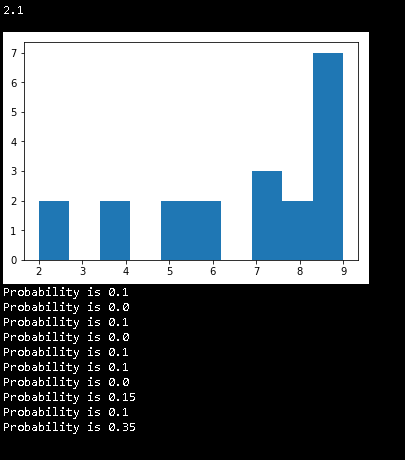
**1.3** Best model obtained is with using learning rate 1.0. This wouldn’t create a linear line as N dimensional is projected on to 2 d plane.

**1.4** In Sol1 python file

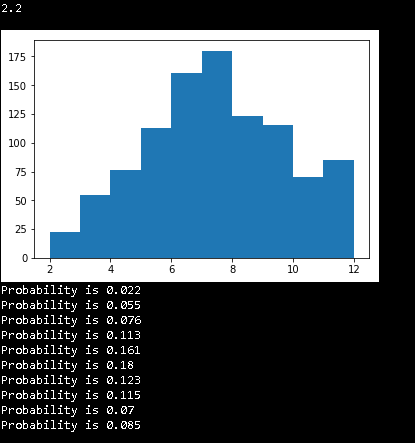
**1.5** In sol1 python file

**A2**

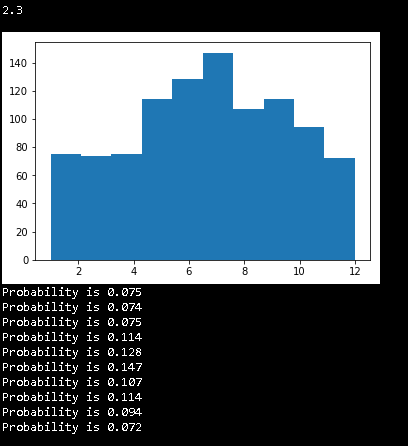
**2.1**



**2.2**



**2.3**



**A3**

Probability of sensor to be faulty P(faulty) = 0.01

Probability of sensor to be not faulty P (¬ faulty) ­­­­­­ = 0.99

Probability of getting a reading less than 1 when sensor is faulty = P(<1|faulty) = 1.0

Probability of getting a reading less than 1 when sensor is not faulty = P (<1|¬ faulty) ­­­­­­= .33

P(faulty|<1) = P(<1|faulty) \*P(faulty)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

P(<1|faulty) \*P(faulty) + P (<1|¬ faulty) ­­­­­­ \* P (¬ faulty) ­­­­­­

For 1st reading by the sensor = 1\*.01/ (1\*.01 + 1/3\*.99) = .029

For 2nd consecutive reading by the sensor = 1\*.01/ (1\*.01 + 1/3\*1/3\*.99) = .083

Similarly, for 10th reading = 1\*.01/ (1\*.01 + 1/3^10\*.99) = .998

So, model will be = 1\*.01/ (1\*.01 + 1/3^N\*.99)

**A4**

The mean (µ1) of the first class is 3 and second (µ2) is 1. The Bell curve will intersect at 2. The standard deviation is 1. The Standard Score (z) is calculated by

z = (x - µ1)/ σ

So, z will be -1. The probability of area under the curve onto the left is .1587. So, probability of remaining area is 1 - .1587 = 0.8413. That is why the theoretical limit will be around 84%.

Another way to prove that is 68/95/99.7 rule. The first 68% area under the bell curve lies at the first standard deviation from the mean which is 0 and 2. Rest we must find the area between -1 to 0. As bell curves are symmetrical, remaining area will be half of 32% which is 16%. So, theoretical limit will be 68% + 16% = 84%

References

[1] "Intro to Data Structures — pandas 0.20.3 documentation", Pandas.pydata.org, 2017. [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/dsintro.html. [Accessed: 03- Oct- 2017].

[2] "numpy.random.choice — NumPy v1.13 Manual", Docs.scipy.org, 2017. [Online]. Available: https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.random.choice.html. [Accessed: 03- Oct- 2017].

[3] F. Y-scaling?, "Fitting a Gaussian to a histogram with MatPlotLib and Numpy - wrong Y-scaling?", Stackoverflow.com, 2017. [Online]. Available: https://stackoverflow.com/questions/23447262/fitting-a-gaussian-to-a-histogram-with-matplotlib-and-numpy-wrong-y-scaling. [Accessed: 03- Oct- 2017].

[4] 2017. [Online]. Available: http://www.stat.ufl.edu/~athienit/Tables/Ztable.pdf. [Accessed: 03- Oct- 2017].