

# Python And R Statistical

March 5, 2019

## 1 Python code

### 1.1 define and import libraries

```
In [4]: import math
import matplotlib.pyplot as plt
```

### 1.2 define graph

```
In [5]: #define graph
dataVector = [0, 7, 6, 3, 7, 8, 7, 11, 7, 8, 6, 2, 6, 9, 2, 11, 10, \
              11, 10, 11, 7, 5, 6, 7, 10, 9, 3, 7, 7, 6, 6, 7, 12, \
              3, 8, 9, 8, 13, 6, 8, 2, 11, 7, 8, 5, 11, 5, 6, 8, 8, \
              9, 4, 6, 9, 16, 7, 6, 7, 6, 7, 7, 7, 4, 6, 9, 7, 11, \
              8, 6, 9, 10, 9, 12, 5, 4, 8, 9, 3, 9, 8, 5, 8, 12, 7, \
              8, 11, 4, 4, 10, 5, 7, 5, 10, 10, 4, 9, 6, 14, 6, 12, \
              5, 11, 9, 5, 6, 7, 4, 5, 8, 9, 7, 4, 10, 6, 5, 7, 8, \
              5, 8, 14, 8, 12, 10, 6, 4, 6, 9, 7, 7, 13, 0]

#y is used for graphs and a 0 needs to be added to the first value so
#the graphs prints correctly
y = [0]
for val in dataVector:
    y.append(val)

#x is the x value for each y value so a for loop is enough
length = len(y)
x = [i for i in range(length)]
```

### 1.3 Create statistical data

```
In [6]: #calculate mean value from the "dataVector" and not
#"y" due to an addition of 0 in first value
lengthData=length-1

#calculate mean value
summ=0
for val in dataVector:
    summ=summ+val
```

```

mean=summ/lengthData

#calculate standard deviation value
standard=0
for val in dataVector:
    standard+=(val-mean)*(val-mean)
standard=math.sqrt(standard/lengthData)

#print data
print("stadard deviation s is:", standard)
print("mean values m is:", mean)
xmean=[0, length]
ymean=[mean, mean]

```

```

stadard deviation s is: 2.8142999779186573
mean values m is: 7.404580152671755

```

## 1.4 Print statistical graph

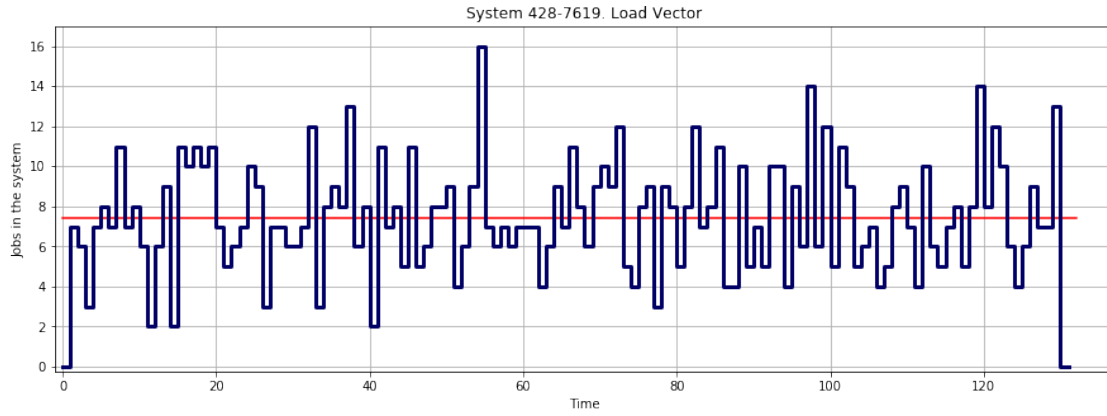
```

In [8]: #if graph shows up small, re-execute this cell
#set graph attributes and print graph
plt.plot(xmean, ymean, color='red')
plt.plot(x, y, drawstyle="steps-pre", linewidth=3, color="#000066")
plt.ylabel('Jobs in the system')
plt.xlabel('Time')
plt.title('System 428-7619. Load Vector')
plt.grid(True)
plt.xlim(0,(length+5))
plt.ylim(0.0,17.0)
plt.axis([-1, (length+5), -0.25, 17.0])

#attempt to stretch the graph so it is easier to read
fig_size = plt.rcParams["figure.figsize"]
fig_size[0] = 15
fig_size[1] = 5
plt.rcParams["figure.figsize"] = fig_size

#shows the graph
plt.show()
#print caption with color
print("\x1b[31m\x1b[31m\"Red line\"\x1b[0m", "is the mean value of the graphs data")
print("stadard deviation s is:", standard)
print("mean values m is:", mean)

```



"Red line" is the mean value of the graphs data  
stadard deviation s is: 2.8142999779186573  
mean values m is: 7.404580152671755

## 2 R code

### 2.1 defining graph values

```
In [1]: y <-c(0, 7, 6, 3, 7, 8, 7, 11, 7, 8, 6, 2, 6, 9, 2, 11,
            10, 11, 10, 11, 7, 5, 6, 7, 10, 9, 3, 7, 7, 6, 6, 7, 12,
            3, 8, 9, 8, 13, 6, 8, 2, 11, 7, 8, 5, 11, 5, 6, 8, 8,
            9, 4, 6, 9, 16, 7, 6, 7, 6, 7, 7, 7, 4, 6, 9, 7, 11,
            8, 6, 9, 10, 9, 12, 5, 4, 8, 9, 3, 9, 8, 5, 8, 12, 7,
            8, 11, 4, 4, 10, 5, 7, 5, 10, 10, 4, 9, 6, 14, 6, 12,
            5, 11, 9, 5, 6, 7, 4, 5, 8, 9, 7, 4, 10, 6, 5, 7, 8,
            5, 8, 14, 8, 12, 10, 6, 4, 6, 9, 7, 7, 13, 0)
x <- c(0:130)
```

### 2.2 Create statistical data

```
In [2]: leng <- length(y)
sum <- 0
for (value in y) {
  sum <- sum + value
}
mean <- sum / leng

mdev <- sd(y)

print("Mean value is: ")
print(mean)
print("Standard deviation is: ")
print(mdev)
```

```
[1] "Mean value is: "
[1] 7.40458
[1] "Standard deviation is: "
[1] 2.825103
```

### 2.3 Print statistical graph

```
In [4]: plot(x, y, "s", main = "System 428-7619. Load Vector",
            col = "#318CE7", lwd = 3, xlab = "Time",
            ylab = "Jobs in the System")
xm <- c(0, 130)
ym <- c(mean, mean)
plot(x, y, "s", main = "System 428-7619. Load Vector",
      col = "#318CE7", lwd = 3, xlab = "Time",
      ylab = "Jobs in the System")
lines(xm, ym, col="red")
abline(v = 0:130, lty = 2, col = "grey")
abline(h = 0:3, lty = 2, col = "grey")
```

```
options(repr.plot.width=14, repr.plot.height=9)
print("Mean value is: ")
print(mean)
print("Standard deviation is: ")
print(mdev)
```

```
[1] "Mean value is: "  
[1] 7.40458  
[1] "Standard deviation is: "  
[1] 2.825103
```

