Descriptive Statistics: Additional Examples with Answers

Download the "Financial Dataset" from the Course Blackboard Site

Course Content> Datasets > Financial.rda

Clear your R workspace using the following command

```
rm(list = ls())
```

Load the Financial data into R, using `load()` command e.g.

```
load("F:/Kent Teaching/Datasets/Financial.rda")
```

where F:/Kent Teaching/Datasets/ is a directory at which the downloaded file is located at.

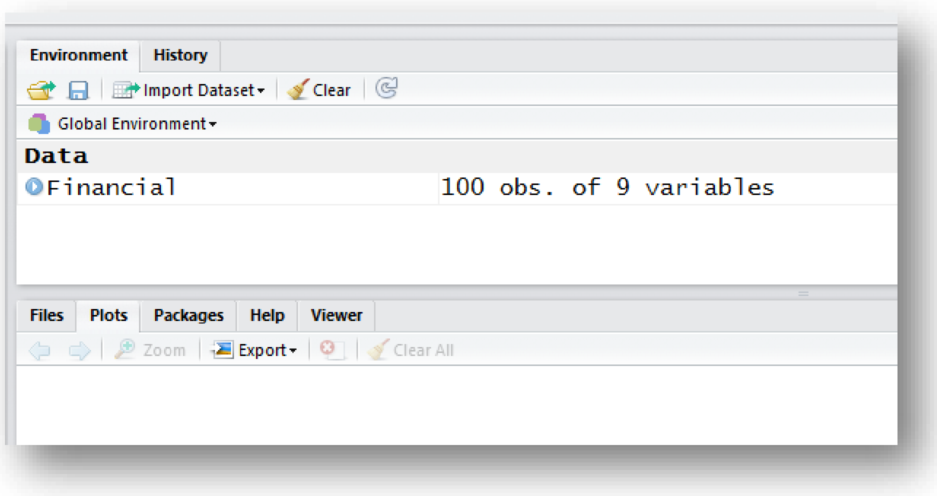
Note that that "/" is used instead of "\" alternatively you could use "//" i.e.

```
load("F:\\Kent Teaching\\Datasets\\Financial.rda")
```

As discussed in the class, if you are using a Mac, download the file. Then right click on the file and click on "Get info" then in front of 'where' you get the path e.g.

```
load("/Users/Razavi/Desktop/Financial.rda")
```

Now you should see a Data frame named Financial in your Global Environment Section



This dataset contains the financial information about 100 companies including their type (type), revenue (rev), assets (assets), return of investment (roe), Earnings Per Share (eps), yield (yield), Dividend Per Share (dps) and finally their Property, Plant and Equipment (ppe).

Answer the following questions:

1. **What are the variable names in the "Financial" dataframe?**
   Double click on the data frame to see

100 obse

	company	type	rev	assets	roe	eps	yield	dps	ppe
1	AFLAC	6	7251	29454	17.1	2.08	0.9	0.22	11.5
2	Albertson's	4	14690	5219	21.4	2.08	1.6	0.63	19.0
3	Allstate	6	20106	80918	20.1	3.56	1.0	0.36	10.6
4	Amerada Hess	7	8340	7935	0.2	0.08	1.1	0.60	698.3
5	American General	6	3362	80620	7.1	2.19	3.0	1.40	21.2
6	American Stores	4	19139	8536	12.2	1.01	1.4	0.34	23.5
7	Amoco	7	36287	32489	16.7	2.76	3.1	1.40	16.1
8	Arco Chemical	2	3995	4116	6.2	1.14	6.1	2.80	40.4
9	Ashland	7	14319	7777	9.5	3.80	2.3	1.10	12.4
10	Atlantic Richfield	7	19272	25322	21.8	5.41	3.8	2.83	3.8
11	Bausch & Lomb	5	1916	2773	6.0	0.89	2.6	1.04	2.6

   Or print the name of the columns of the data frame using

```
> colnames(Financial)
[1] "company" "type"    "rev"     "assets"  "roe"     "eps"     "yield"
[8] "dps"     "ppe"
```

2. **What is the mean, median and standard deviation of revenue across all 100 companies?**

```
> mean(Financial$rev)
[1] 11043.37
> median(Financial$rev)
[1] 6101
> sd(Financial$rev)
[1] 17479.12
```

3. **What is the highest revenue amongst all companies?**

```
> max(Financial$rev)
[1] 137242
```

   Alternatively, since we know revenue is the 3rd column, we could write it as

```
> max(Financial[,3])
[1] 137242
```

4. **Which company has the highest revenue?**

   You can use the command `which.max()` which returns the index of the highest value e.g. returns 4 if the 4th element is the largest element. Once you know which row contains the highest value you can then print that row:

```
> which.max(Financial$rev)
[1] 31
```

   And Then:

```
> Financial[31,]
   company type    rev assets  roe  eps yield  dps  ppe
31   Exxon    7 137242  96064 19.4 3.37   2.8 1.63 17.1
```

Or alternatively you could do it in one go:

```
> Financial[which.max(Financial$rev),]
   company type    rev assets  roe  eps yield  dps  ppe
31   Exxon    7 137242  96064 19.4 3.37   2.8 1.63 17.1
```

5. Which company has the lowest Return on Investment?

```
> Financial[which.min(Financial$roe),]
        company type  rev assets roe  eps yield dps   ppe
4 Amerada Hess    7 8340   7935 0.2 0.08   1.1 0.6 698.3
```

6. What are the top 5 companies with highest assets?

`order()` can be used to get the index of the sorted values.

```
order(Financial$assets)
  [1]  72  38  34  65  14  59  77  96  80  30  39 100  97  60  47  35  48  90
 [19]  63  52  58  37  99  15  61  40  94  27  84  11  57  98  49  36  44  33
 [37]  23  64  76   8  45  54  68  81  62   2  20  88  51  25  86  78  73  89
 [55]  91  75   9   4  95   6  12  18  74  50  70  93  56  85  32  92  71  66
 [73]  19  13  69  41  21  43  42  67  22  26  10  53  28   1  83   7  16  82
 [91]  79  24  55  46   5   3  31  17  29  87
```

Unless otherwise stated, order always sort values ascendingly i.e. the lowest value is always on top. This means that the company on row 72 has the lowest assets and the one at row 87 has the highest. So the top 5 companies with highest assets are in rows 87,29, 17,31, and 3.

```
> Financial[c(87,29, 17,31,3),]
     company type    rev assets  roe  eps yield  dps  ppe
87 Travelers    6  37609 386555 14.9 2.54   0.9 0.40 17.0
29 Equitable    6   9666 151438 12.3 2.86   0.5 0.20 13.4
17     CIGNA    6  14935 108199 13.7 4.88   2.0 1.10 11.4
31     Exxon    7 137242  96064 19.4 3.37   2.8 1.63 17.1
3   Allstate    6  20106  80918 20.1 3.56   1.0 0.36 10.6
```

7. What is the standard deviation of "roe" values Try to calculate the standard deviation without using the sd() command and by formula, then use the sd() command and compare the results.

The formula for calculating standard deviation of x, is

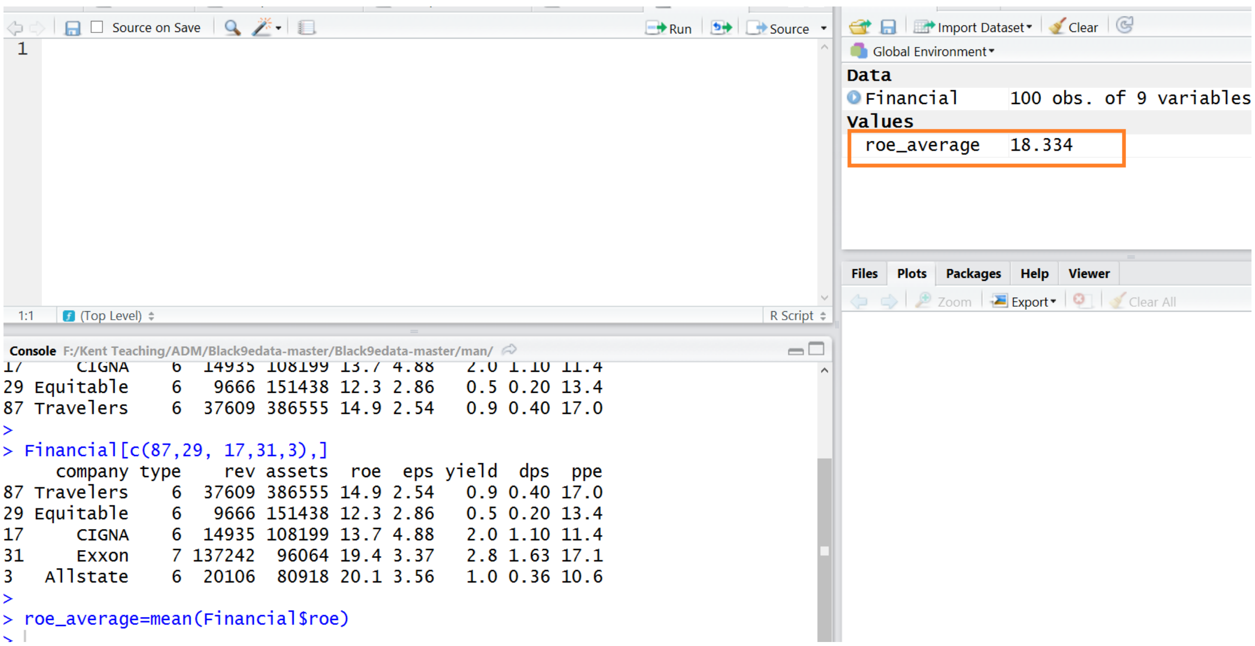
$$\sigma = \sqrt{\frac{\sum(x-\overline{x})^2}{n-1}}$$

The formula says take the mean of x from each element of x and then square it (i.e. power 2), then sum them up, then divide the result by the number of elements in x mines 1 (that is n-1) and finally calculate the square root of the result. So let's do this step by step:

Step 1: calculate the mean/average of Financial$roe. We call this roe_average (you can use any other variable name, as long as they are meaningful to you)

```
roe_average=mean(Financial$roe)
```

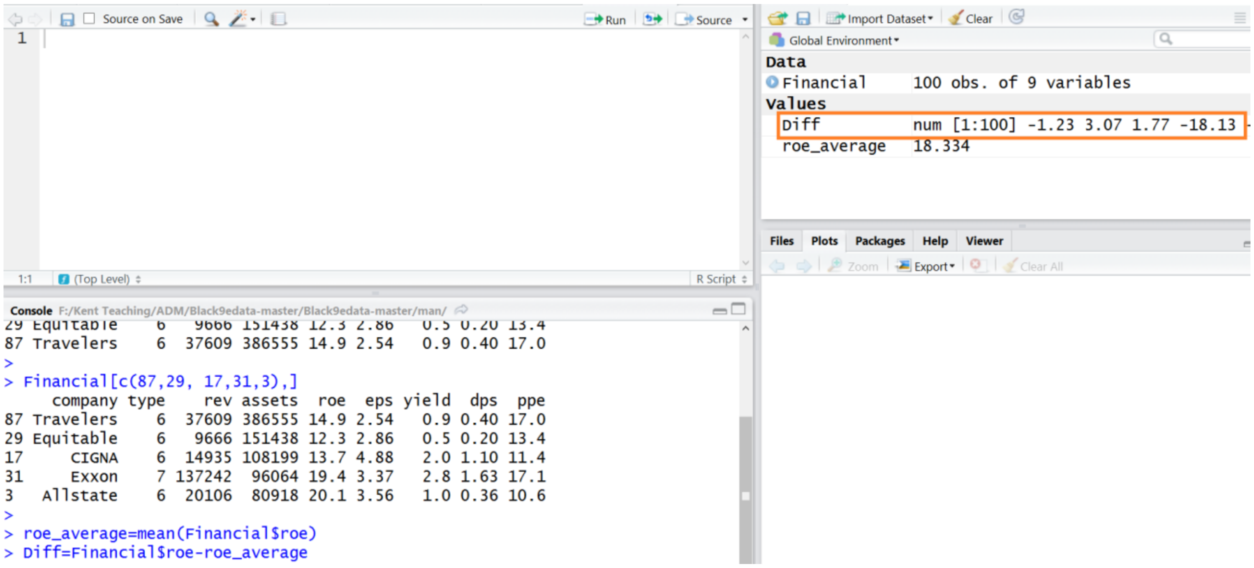
Note that as soon as you execute this line the new variable (value) called roe_average is created in your Global Environment Section



Step 2: Let's calculate the difference between each of the roe values and their mean as we calculated in step 1. We call this "Diff".

```
Diff=Financial$roe-roe_average
```

Note that once you run this command the new variable, Diff, appears under the Global Environment Section:



Step 3: Let’s calculate the square of the Diff values and call it “Diff_square”

```
Diff_square=Diff^2
```

Or we could write

```
Diff_square=Diff*Diff
```

Similarly, new variable Diff_square will be created

Step 4: Lets calculate the sum of the squared differences. We call it “sum_diff_square”.

```
sum_diff_square=sum(Diff_square);
```

Step 5: Finally, lets divide the product (i.e. sum_diff_square) by the number of elements (that is 100) mines 1 and calculate the square root of the results to calculate the standard deviation.

```
sqrt(sum_diff_square /99)
```

Or

```
(sum_diff_square /99)^0.5
```

```
> sqrt(sum_diff_square /99)
[1] 23.57264
> sd(Financial$roe)
[1] 23.57264
```

We can see the results are the same.

Tip: Instead of creating intermediate variables such as roe_average, Diff, Diff_square, sum_diff_square etc. we could write everything in one line:

```
> sqrt(sum((Financial$roe-mean(Financial$roe))^2)/99)
[1] 23.57264
```

8. Use the summary() command to get a summary of all variables in the Financial dataframe. By comparing the mean and median, can you say which variables are highly skewed? And in which direction (positive/right skewed versus negative/left skewed). Then use skewness() command to confirm that.

```
summary(Financial)
   company               type            rev             assets      
 Length:100         Min.   :1.00   Min.   :   129   Min.   :   194  
 Class :character   1st Qu.:2.00   1st Qu.:  2259   1st Qu.:  2393  
 Mode  :character   Median :4.50   Median :  6101   Median :  5876  
                    Mean   :4.23   Mean   : 11043   Mean   : 18855  
                    3rd Qu.:6.00   3rd Qu.: 12818   3rd Qu.: 16106  
                    Max.   :7.00   Max.   :137242   Max.   :386555  
      roe                eps             yield            dps        
 Min.   :  0.200   Min.   :0.080   Min.   :0.100   Min.   :0.0200  
 1st Qu.:  9.475   1st Qu.:1.295   1st Qu.:1.375   1st Qu.:0.4650  
 Median : 13.950   Median :1.990   Median :2.200   Median :0.9000  
 Mean   : 18.334   Mean   :2.247   Mean   :2.543   Mean   :0.9633  
 3rd Qu.: 20.150   3rd Qu.:3.027   3rd Qu.:3.100   3rd Qu.:1.2550  
 Max.   :228.000   Max.   :7.700   Max.   :7.800   Max.   :3.2400  
      ppe        
 Min.   :  2.60  
 1st Qu.: 13.93  
 Median : 17.00  
 Mean   : 30.27  
 3rd Qu.: 24.68  
 Max.   :698.30  
```

Looking at the results above, we can see the mean is much higher than the median for rev, assets, roe, and ppe as such we expect these variables to be highly skewed towards right (i.e., positive skewness). Let us use the skewness() command from the 'modeest' library or 'moments' library to check the skewness of each variable

```
> library('modeest')
> skewness(Financial$rev)
[1] 4.384549
attr(,"method")
[1] "moment"
> skewness(Financial$assets)
[1] 6.094588
attr(,"method")
[1] "moment"
> skewness(Financial$roe)
[1] 7.115177
attr(,"method")
[1] "moment"
> skewness(Financial$eps)
[1] 1.054758
```

```
attr(,"method")
[1] "moment"
> skewness(Financial$yield)
[1] 1.304713
attr(,"method")
[1] "moment"
> skewness(Financial$dps)
[1] 1.054382
attr(,"method")
[1] "moment"
> skewness(Financial$ppe)
[1] 7.772048
attr(,"method")
[1] "moment"
```

The above results confirms our predictions as those identified variables have high skewness coefficients.

9. Calculate the skewness of "assets" without using the skewness() command. Then use the command to compare the results.

The formula for calculating the skewness of X, is

$$\gamma_1 = \mathbf{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right]$$

In other words we need to calculate the average of the X, that is Mu and the standard deviation of X that is sigma (in the denominator) , E[] also represents the expected value which is the same as the mean. Let's do this step by step.

```
#Step 1: Calculate Mu (the average of X)

Mean_assets=mean(Financial$assets)

#Step 2: Calculate sigma (the standard deviation of X)

Sd_assets=sd(Financial$assets)

#Step 3: Calculate the difference between the values of X and its mean.

Diff= Financial$assets- Mean_assets

#Step 4: Divide by standard deviation

Diff_normalized=Diff/Sd_assets

#Step 5: To power 3 and calculate the mean

mean(Diff_normalized^3)
```

```
> #Step 1: Calculate Mu (the average of X)
> Mean_assets=mean(Financial$assets)
> #Step 2: Calculate sigma (the standard deviation of X)
> Sd_assets=sd(Financial$assets)
> #Step 3: Calculate the difference between the values of X and its mean.
> Diff= Financial$assets- Mean_assets
> #Step 4: Divide by standard deviation
> Diff_normalized=Diff/Sd_assets
> #Step 5: To power 3 and calculate the mean
> mean(Diff_normalized^3)
[1] 6.094588
```

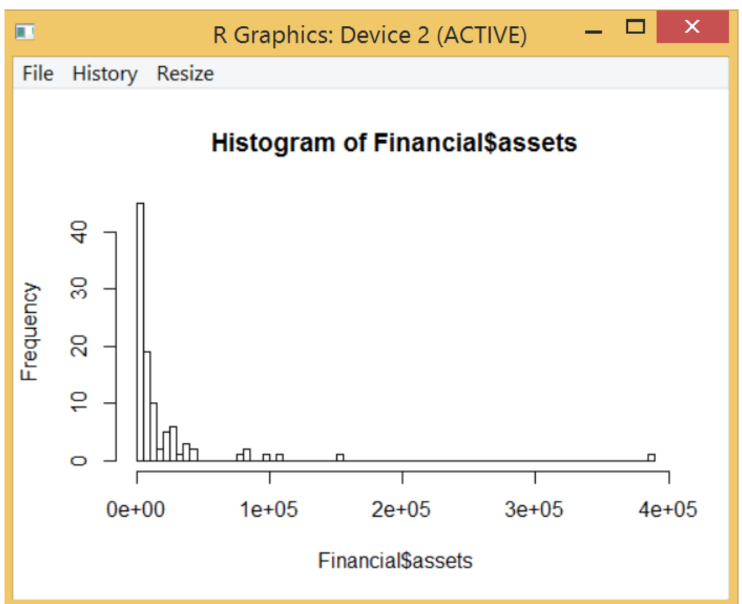
Now let's use the formula:

```
> library('modeest')
> skewness(Financial$assets)
[1] 6.094588
attr(,"method")
[1] "moment
```

Same answer!

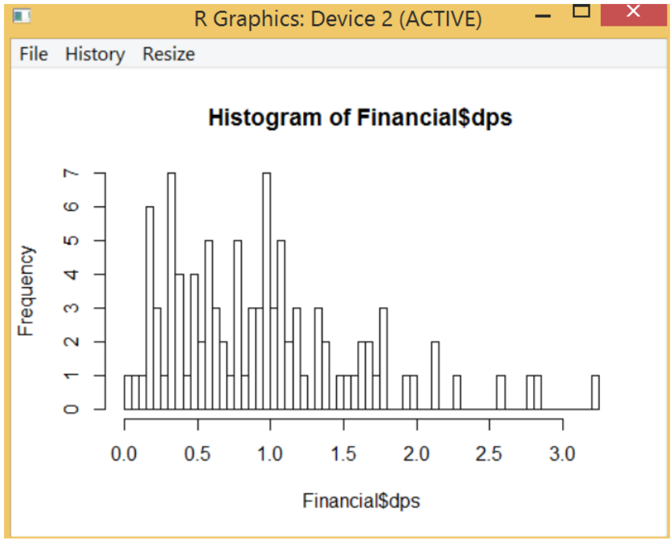
10. Examine the distribution of the "assets" and compare it with "dps".

```
hist(Financial$assets,n=100)
```



We can see the distribution is highly skewed towards right. We expected this since the skewness coefficient was also high.

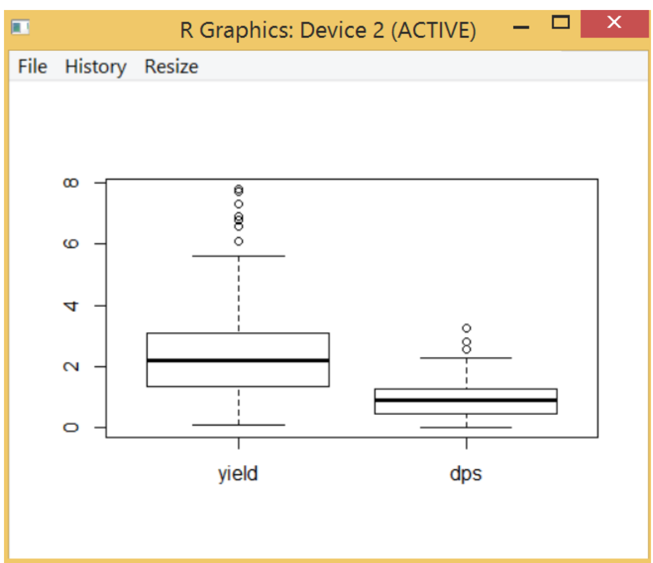
```
hist(Financial$dps,n=100)
```



The above distribution is also skewed towards right (hence a positive skewness coefficient) but to a much lesser extent. Skewness was 1.05 compared to 6.09 for assets.

11. Plot the Boxplot of "Yeild" and "dps" (i.e. columns 7 and 8 of the Financial dataframe). Just by looking at the boxplot, can you say which variable has a larger spread i.e. standard deviation? Confirm this by comparing the standard deviation values.

```
boxplot(Financial[,7:8])
```



Just by looking at the above, we can see that "Yeild" is much more spread compared to the "dps" values.

```
> sd(Financial$yield)
[1] 1.737597
> sd(Financial$dps)
[1] 0.6592436
```

We were right!

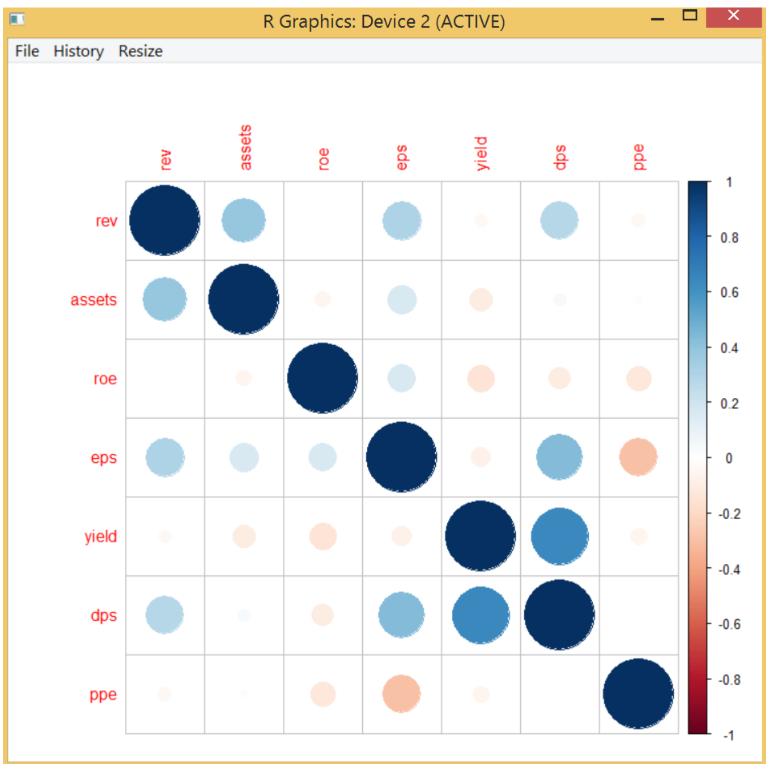
11. Calculate and plot the correlation between all numerical variables in the "Financial" dataframe.

The numerical variables are in columns 3-9. Lets create a new data frame containing only those variables.

```
Financial_numerical= Financial[,3:9]
```

Let's use the "corrplot" library to plot the correlations:

```
> corrplot(cor(Financial_numerical))
```



We can see that there is a very strong positive correlation between yield and dps as well as assets and revenue (which makes sense) and

Just by looking at the above, we can see that "Yeild" is much more spread compared to the "dps" values.