Numerical Analysis — Problem Set 7 — Standard Deviation and *r*-Value

Due Tuesday, Nov. 1 (beginning of class)

We are going to work with Aaron Judge's run-value-weighted Offensive Batting Average (rOBA) as found on Max's Problem Set 6. Max had the following table:

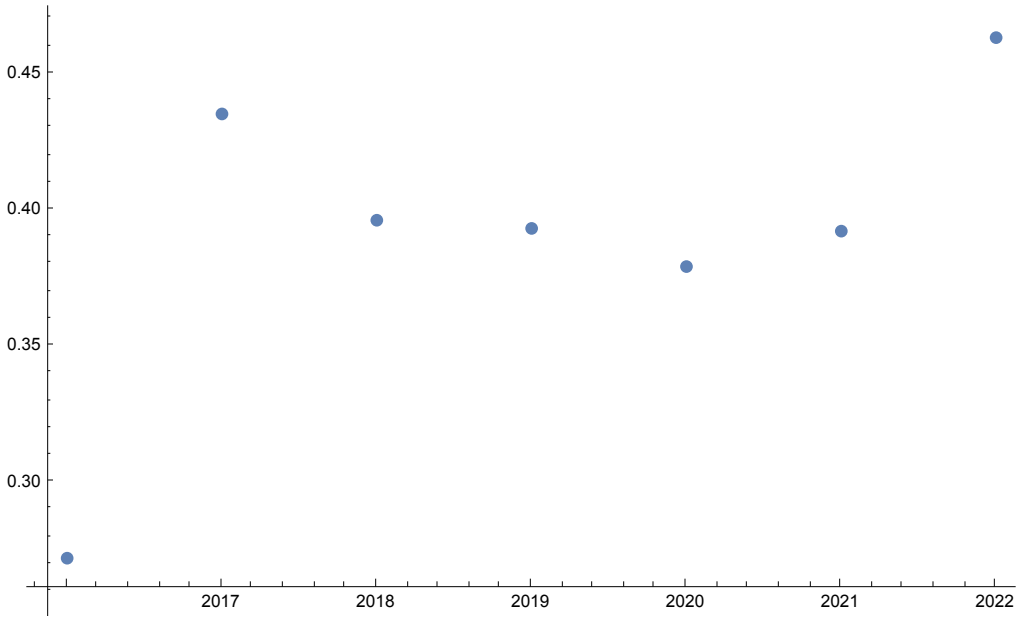
In[1]:=
```
data = {{"Year", "Average"}, {2016, 0.272}, {2017, 0.435},
  {2018, 0.396}, {2019, 0.393}, {2020, 0.379}, {2021, 0.392}, {2022, 0.463}};
Grid[data]
```

Out[1]=

Year	Average
2016	0.272
2017	0.435
2018	0.396
2019	0.393
2020	0.379
2021	0.392
2022	0.463

In[11]:=
```
ListPlot[Drop[data]]
```

Out[11]=



Let's ignore the years for a moment:

In[2]:=
```
battingAverages = Transpose[Drop[data, 1]][[2]]
```

Out[2]=
```
{0.272, 0.435, 0.396, 0.393, 0.379, 0.392, 0.463}
```

1. Mean and Total Deviation of the Batting Averages

(a) There are 7 numbers in the battingAverages list at the bottom of the previous page. The first thing to do is calculate their mean. “Mean” is a fancy word for average commonly used in the world of statistics. So I am asking you to take an average of seven things (which themselves are averages). So you just add the seven numbers up and divide by 7. Store the average in some register (for example R0). You will use the average a lot in the next part and you don’t want to have to type it in over and over again.

(b) Make a new list from the original list. For each number in the old list, you just subtract the mean you stored in R0 to get the corresponding number in the new list. Record that list as your answer to this part.

(c) This new list tells you how much each number in the original list deviates from the mean. You could add up all the numbers in this new list to find the total deviation. Do that. The answer may surprise you.

(d) Try it again with a different list to make sure that the surprise wasn’t an accident. We’ll use this list, {20, 40, 50, 65, 15, 8}, of six numbers. Find its mean. Store that in R1 so that you can use it repeatedly.

(e) Subtract the mean you stored in R1 from each number in the list. This is the list of deviations from the mean.

(f) Sum up the deviations. The surprise is not an accident.

2. Variance of the Batting Averages

From 1(c) and 1(f), you are starting to get the idea that summing up the deviations is useless. The total deviation is always going to be 0.

(a) From the list in 1(b) make another list by squaring each of the numbers.

(b) Sum up the numbers you got in 2(a). Divide it by 7. This is called the variance.

The truth is, if you are a serious statistician, you don’t divide by 7. You divide by 6. That is because the 7 deviations are not independent. They add up to 0, so of course they are not fully independent! One of the deviations is somehow redundant. The upshot of this redundancy is that you don’t have 7 estimates of the variance. You really only have 6.

(c) Sum up the numbers you got in 2(a) again. Divide it by 6. *This is what statisticians actually use for the variance.*

3. Standard Deviation of the Batting Averages

Take the square root of the number you found in 2(c). This is called the standard deviation. What I had you calculate in Problems 1-3 is:

$$s_y = \sqrt{\frac{\sum(y_i-\bar{y})^2}{n-1}}$$

Take a look at p. 101 of the *HP-25 Applications Programs* book now. Specifically, take a look at the formula for s_y. You just calculated s_y for Max's Aaron Judge data! There are actually subtle differences between this formula and what I had you do above. However, with some rearrangement, I will show you on Tuesday that the procedure you used to get s_y and the one on p. 101 are the same.

4. Standard Deviation of the Years!?!

In[3]:=
```
years = Transpose[ Drop[data, 1]][[1]]
```

Out[3]=
```
{2016, 2017, 2018, 2019, 2020, 2021, 2022}
```

(a) Take the mean of the years. Go ahead and put that in R1.

(b) Make a new list from the years list. For each number in the years list, you just subtract the mean you stored in R1 to get the corresponding number in the new list. Record that list as your answer to this part.

Isn't it kind of weird that we are doing the same thing to the years as we did to the data!?! Years are not uncertain. They do not really have statistical variance. And yet we are going to calculate their variance and their standard deviation....

(c) Square the numbers you got in (b), add them up, and divide by 6. This is the variance of the years.

(d) Take the square root of what you got in (c). This is the standard deviation of the years.

Take a look at p. 101 again. Specifically, look at the formula for s_x. You just calculated s_x for Max's Aaron Judge data! If you scrutinize carefully, you will see that I had you calculate in this Problem is:

$$s_x = \sqrt{\frac{\sum(x_i-\bar{x})^2}{n-1}}$$

Just like for s_y this are subtle differences between this formula for s_x and what is in the *HP-25 Applications Programs* book, but when I have a blackboard I will show you that they are the same.

5. Covariance and *r*-value

The HP-25 Applications book also defines the covariance and the *r* value.

If there is a strong positive correlation between the *y*-values and the *x*-values, then the *r* value will be close to 1.

If there is a strong negative correlation between the *y*-values and the *x*-values, then the *r* value will be close to -1.

Type in the program on p. 101. Apply the program to Max's data. What are s_x, s_y, and r?

6. Covariance and *r*-Value Again

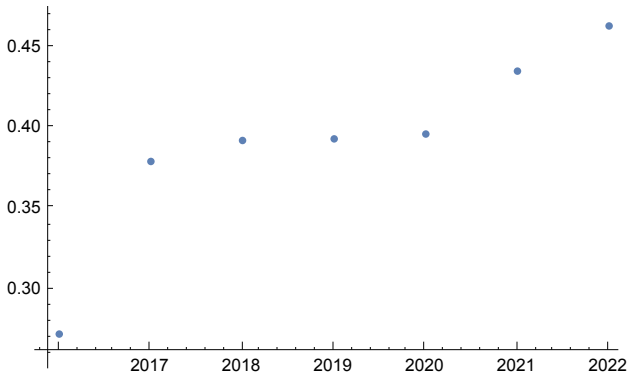
In[4]:=
```
rearrangedAscendingData = {{"Year", "Average"}, {2016, 0.272}, {2017, 0.379},
   {2018, 0.392}, {2019, 0.393}, {2020, 0.396}, {2021, 0.435}, {2022, 0.463}};
Grid[rearrangedAscendingData]
```

Out[5]=

Year	Average
2016	0.272
2017	0.379
2018	0.392
2019	0.393
2020	0.396
2021	0.435
2022	0.463

In[10]:=
```
ListPlot[Drop[rearrangedAscendingData]]
```

Out[10]=



You can see that all I did was rearrange the data from the different years so that Aaron Judge's batting average is constantly improving. Apply the program to this data. What are s_x, s_y, and r?

NB: If you didn't get the same values for s_x and s_y something is wrong. Can you see that s_x and s_y should have been the same?

7. Covariance and *r*-Value One More Time

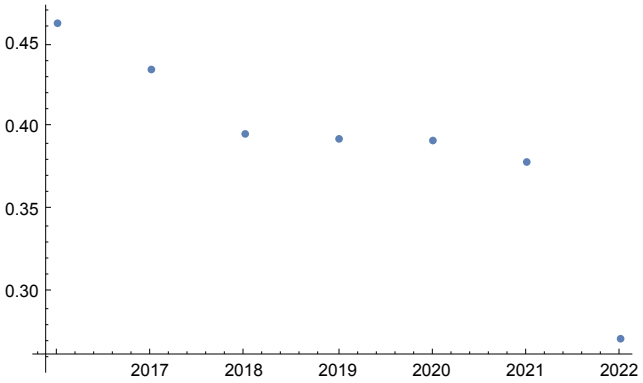
In[6]:=
```
rearrangedDescendingData = {{"Year", "Average"}, {2016, 0.463}, {2017, 0.435},
   {2018, 0.396}, {2019, 0.393}, {2020, 0.392}, {2021, 0.379}, {2022, 0.272}};
Grid[rearrangedDescendingData]
```

Out[7]=

Year	Average
2016	0.463
2017	0.435
2018	0.396
2019	0.393
2020	0.392
2021	0.379
2022	0.272

In[9]:=
```
ListPlot[Drop[rearrangedDescendingData]]
```

Out[9]=



You can see that all I did was rearrange the data from the different years so that Aaron Judge's batting average is now constantly deteriorating. Apply the program to this data.

Now what are s_x, s_y, and r?

Conclusion

You will see *r*-value and *r*-squared-value quoted a lot when data is fitted. An *r*-value near 1 is interpreted as "the data on the *y*-axis is positively correlated with the data on the *x*-axis." An *r*-value near -1 is interpreted as "the data on the *y*-axis is negatively correlated with the data on the *x*-axis." Often the *r*-value is squared. Then an r^2-value near 1 means that the data on the *y*-axis is correlated with the data on the *x*-axis, but because you have squared *r*, it is no longer possible to tell whether it is positively or negatively correlated. In all cases an *r*-value or an *r*-squared value near 0 means that the values on the *x*-axis do little to help explain the values on the *y*-axis. To put it another way, if *r* is near 0, a simple mean is just as good an approximation to the *y*-values as a linear fit.

Take a look at the attached pages to develop some additional intuition on *r*-squared.

R-squared intuition

AP.STATS: DAT-1 (EU), DAT-1.G (LO), DAT-1.G.4 (EK)

When we first learned about the correlation coefficient, r, we focused on what it meant rather than how to calculate it, since the computations are lengthy and computers usually take care of them for us.

We'll do the same with r^2 and concentrate on how to interpret what it means.

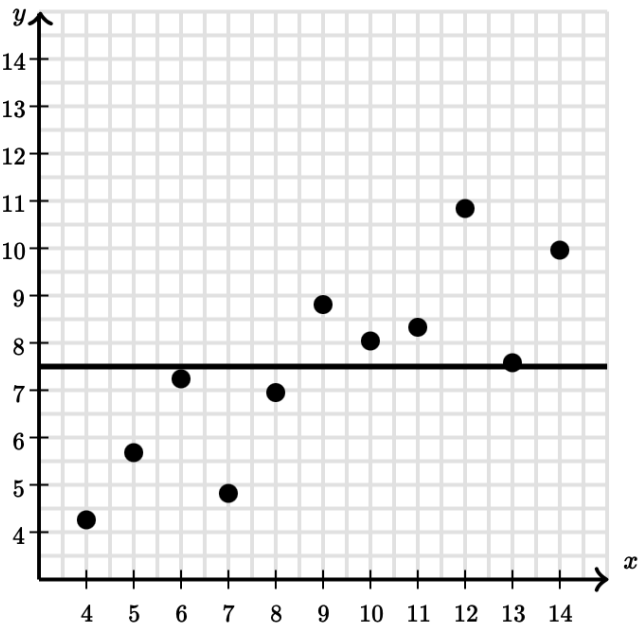
In a way, r^2 measures how much prediction error is eliminated when we use least-squares regression.

Predicting without regression

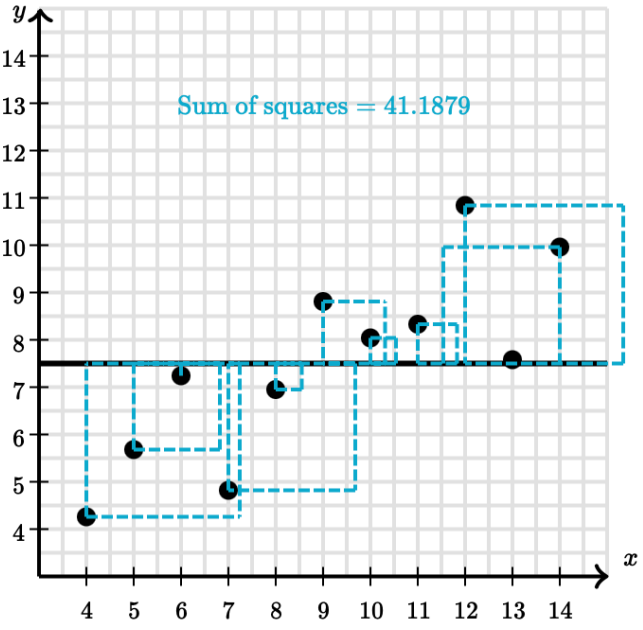
We use linear regression to predict y given some value of x. But suppose that we had to predict a y value without a corresponding x value.

Without using regression on the x variable, our most reasonable estimate would be to simply predict the average of the y values.

Here's an example, where the prediction line is simply the mean of the y data:



Notice that this line doesn't seem to fit the data very well. One way to measure the fit of the line is to calculate the sum of the squared residuals—this gives us an overall sense of how much prediction error a given model has.

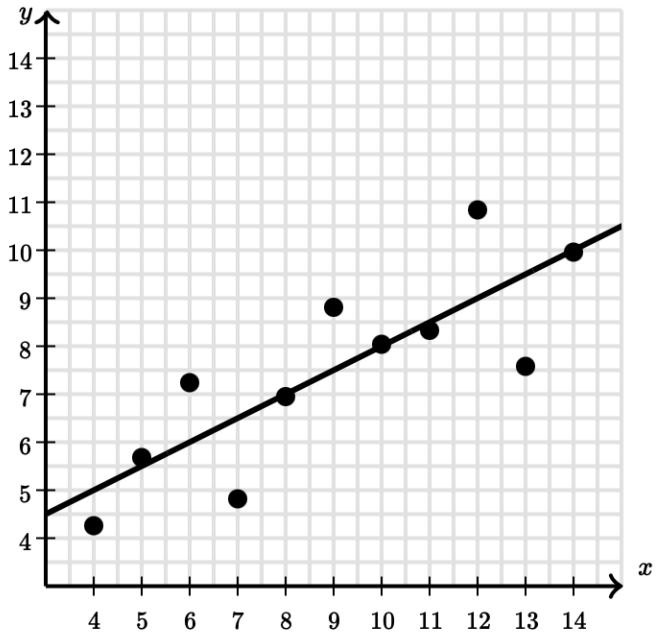


So without least-squares regression, our sum of squares is 41.1879

Would using least-squares regression reduce the amount of prediction error? If so, by how much? Let's see!

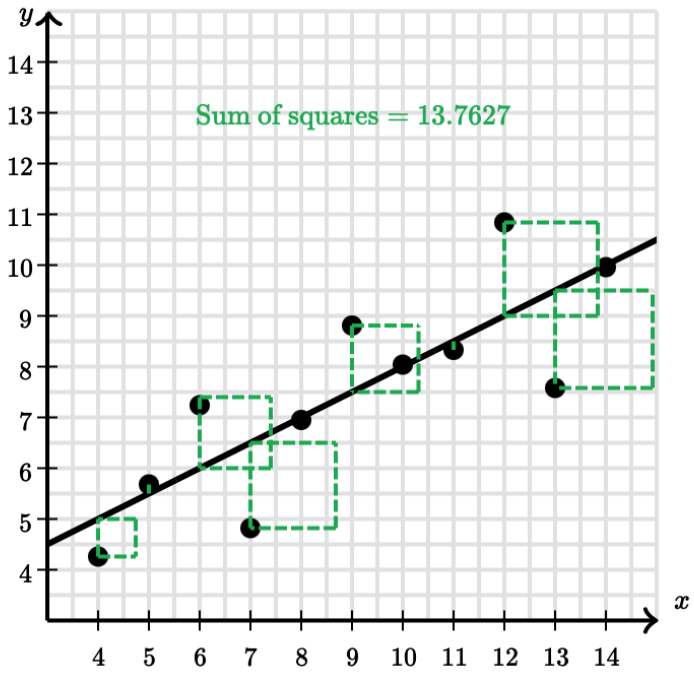
Predicting with regression

Here's the same data with the corresponding least-squares regression line and summary statistics:



Equation	r	r^2
$\hat{y} = 0.5x + 1.5$	0.816	0.6659

This line seems to fit the data pretty well, but to measure how much better it fits, we can look again at the sum of the squared residuals:



Using least-squares regression reduced the sum of the squared residuals from 41.1879 to 13.7627.

So using least-squares regression eliminated a considerable amount of prediction error. How much though?

R-squared measures how much prediction error we eliminated

Without using regression, our model had an overall sum of squares of 41.1879. Using least-squares regression reduced that down to 13.7627.

So the total reduction there is $41.1879 - 13.7627 = 27.4252$.

We can represent this reduction as a percentage of the original amount of prediction error:

$$\frac{41.1879 - 13.7627}{41.1879} = \frac{27.4252}{41.1879} \approx 66.59\%$$

If you look back up above, you'll see that $r^2 = 0.6659$.

R-squared tells us what percent of the prediction error in the y variable is eliminated when we use least-squares regression on the x variable.

As a result, r^2 is also called the **coefficient of determination.**

Many formal definitions say that r^2 tells us what percent of the variability in the y variable is accounted for by the regression on the x variable.

It seems pretty remarkable that simply squaring r gives us this measurement. Proving this relationship between r and r^2 is pretty complex, and is beyond the scope of an introductory statistics course.

COVARIANCE AND CORRELATION COEFFICIENT

For a set of given data points $\{(x_i, y_i), i = 1, 2, ..., n\}$, the covariance and the correlation coefficent are defined as:

$$\text{covariance } s_{xy} = \frac{1}{n-1}\left(\Sigma x_i y_i - \frac{1}{n}\Sigma x_i \Sigma y_i\right)$$

$$\text{or } s_{xy}' = \frac{1}{n}\left(\Sigma x_i y_i - \frac{1}{n}\Sigma x_i \Sigma y_i\right)$$

$$\text{correlation coefficient } r = \frac{s_{xy}}{s_x s_y}$$

where s_x and s_y are standard deviations

$$s_x = \sqrt{\frac{\Sigma x_i^2 - (\Sigma x_i)^2/n}{n-1}} \qquad s_y = \sqrt{\frac{\Sigma y_i^2 - (\Sigma y_i)^2/n}{n-1}}$$

Note:

$-1 \leqslant r \leqslant 1$

DISPLAY LINE	DISPLAY CODE	KEY ENTRY
00		
01	31	↑
02	15 02	g x^2
03	23 51 02	STO + 2
04	22	R↓
05	21	x⇄y
06	25	Σ+
07	13 00	GTO 00
08	24 05	RCL 5
09	24 04	RCL 4
10	24 07	RCL 7
11	61	x
12	24 03	RCL 3
13	71	÷
14	41	–
15	24 03	RCL 3
16	01	1
17	41	–
18	23 00	STO 0
19	71	÷
20	23 01	STO 1
21	74	R/S
22	24 00	RCL 0
23	61	x
24	24 03	RCL 3

DISPLAY LINE	DISPLAY CODE	KEY ENTRY
25	71	÷
26	74	R/S
27	14 22	f s
28	23 71 01	STO ÷ 1
29	24 02	RCL 2
30	24 04	RCL 4
31	15 02	g x^2
32	24 03	RCL 3
33	71	÷
34	41	–
35	24 00	RCL 0
36	71	÷
37	14 02	f $\sqrt{x}$
38	23 71 01	STO ÷ 1
39	24 01	RCL 1
40	13 00	GTO 00
41		
42		
43		
44		
45		
46		
47		
48		
49		

REGISTERS
R_0 n – 1
R_1 Used
R_2 Σy^2
R_3 n
R_4 Σy
R_5 Σxy
R_6 Σx^2
R_7 Σx

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS				OUTPUT DATA/UNITS
1	Key in program						
2	Initialize		f	PRGM	f	REG	
3	Perform this step for i = 1, 2,...,n	x_i	↑				
		y_i	R/S				i
4	Compute covariance s_{xy}		GTO	08	R/S		s_{xy}
5	Compute s_{xy}'		R/S				s_{xy}'
6	Compute correlation coefficient		R/S				r
7	For new case, go to step 2.						

Example:

x_i	26	30	44	50	62	68	74
y_i	92	85	78	81	54	51	40

Solution:

$s_{xy} = -354.14$

$s_{xy}' = -303.55$

$r = -0.96$