*

Connecting the Dots
- or -
Using Plotly for Straight Line and Broken Line Plots

ISC1057

Janet Peterson and John Burkardt

Computational Thinking

Fall Semester 2016

Example 1: Sales Tax

A 7.5% sales tax is imposed in the city of Tallahassee, representing the sum of a 6.0% state sales tax and a 1.5% county sales tax.

One way to think about this is to say that the tax is computed by multiplying the list price of an item by 7.5%, that is, multiplying by the number 0.075, after which the sales price is determined by adding the tax to the list price.

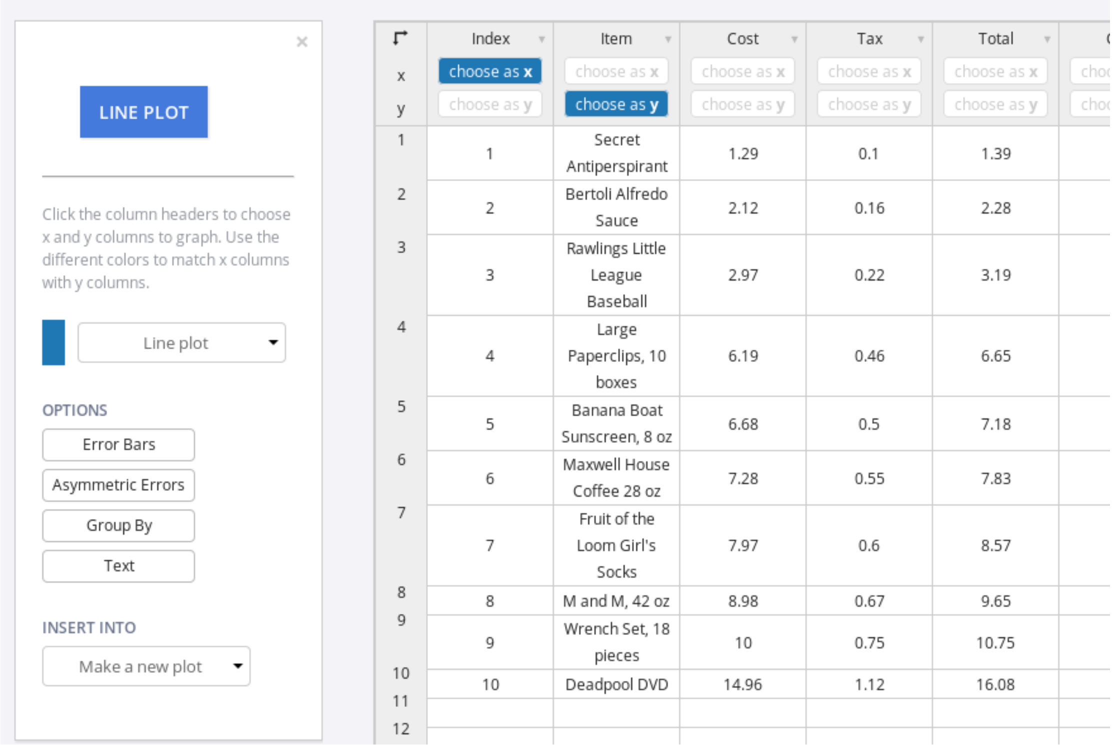
An equivalent calculation is simply to multiply the list price by 1.075 to get the sales price.

The resulting sales price will probably include a fraction of a penny. This is rounded down if less than half a cent, and rounded up otherwise.

After buying 10 items in a local store, we have the following data (also stored in the file *taxables.csv*):

“Index”,	”Item”,	”List ($)”,	”Tax ($)”,	”Sales ($)”
1,	”Secret Antiperspirant”,	1.29,	0.10,	1.39
2,	”Bertoli Alfredo Sauce”,	2.12,	0.16,	2.28
3,	”Rawlings Baseball”,	2.97,	0.22,	3.19
4,	”Large Paperclips, 10 boxes”,	6.19,	0.46,	6.65
5,	”Banana Boat Sunscreen, 8 oz”,	6.68,	0.50,	7.18
6,	”Maxwell House Coffee 28 oz”,	7.28,	0.55,	7.83
7,	”Girl’s Socks”,	7.97,	0.60,	8.57
8,	”M and M, 42 oz”,	8.98,	0.67,	9.65
9,	”Wrench Set, 18 pieces”,	10.00,	0.75,	10.75
10,	”Deadpool DVD”,	14.96,	1.12,	16.08

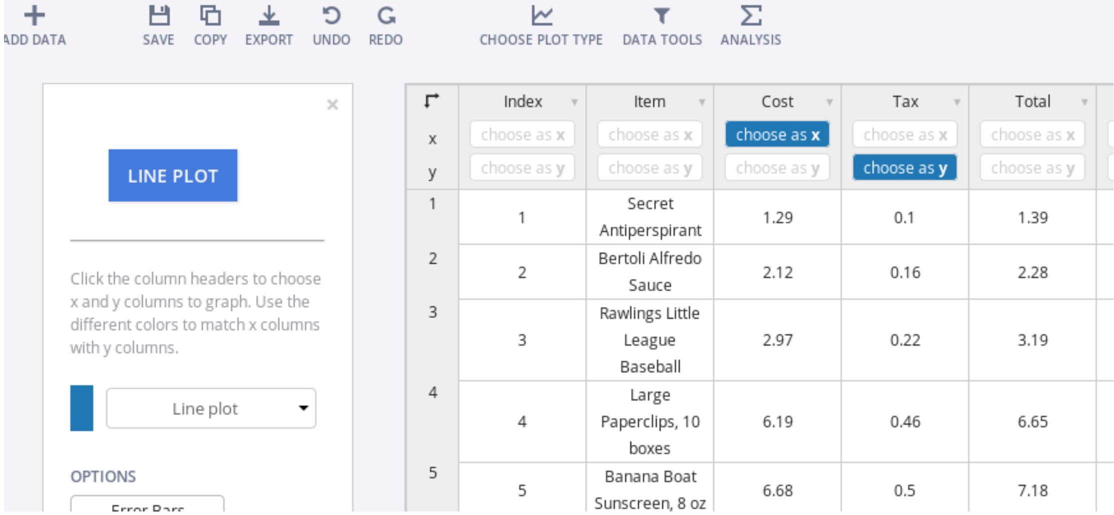
To examine this data, we can log into Plotly, then use the “import” menu to browse for the file *taxables.csv* and bring it into the Plotly grid, so that we see something like this:



Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

	Index	Item	Cost	Tax	Total	(
x	choose as x	choose as x	choose as x	choose as x	choose as x	cho
y	choose as y	choose as y	choose as y	choose as y	choose as y	cho
1	1	Secret Antiperspirant	1.29	0.1	1.39	
2	2	Bertoli Alfredo Sauce	2.12	0.16	2.28	
3	3	Rawlings Little League Baseball	2.97	0.22	3.19	
4	4	Large Paperclips, 10 boxes	6.19	0.46	6.65	
5	5	Banana Boat Sunscreen, 8 oz	6.68	0.5	7.18	
6	6	Maxwell House Coffee 28 oz	7.28	0.55	7.83	
7	7	Fruit of the Loom Girl's Socks	7.97	0.6	8.57	
8	8	M and M, 42 oz	8.98	0.67	9.65	
9	9	Wrench Set, 18 pieces	10	0.75	10.75	
10	10	Deadpool DVD	14.96	1.12	16.08	
11						
12						

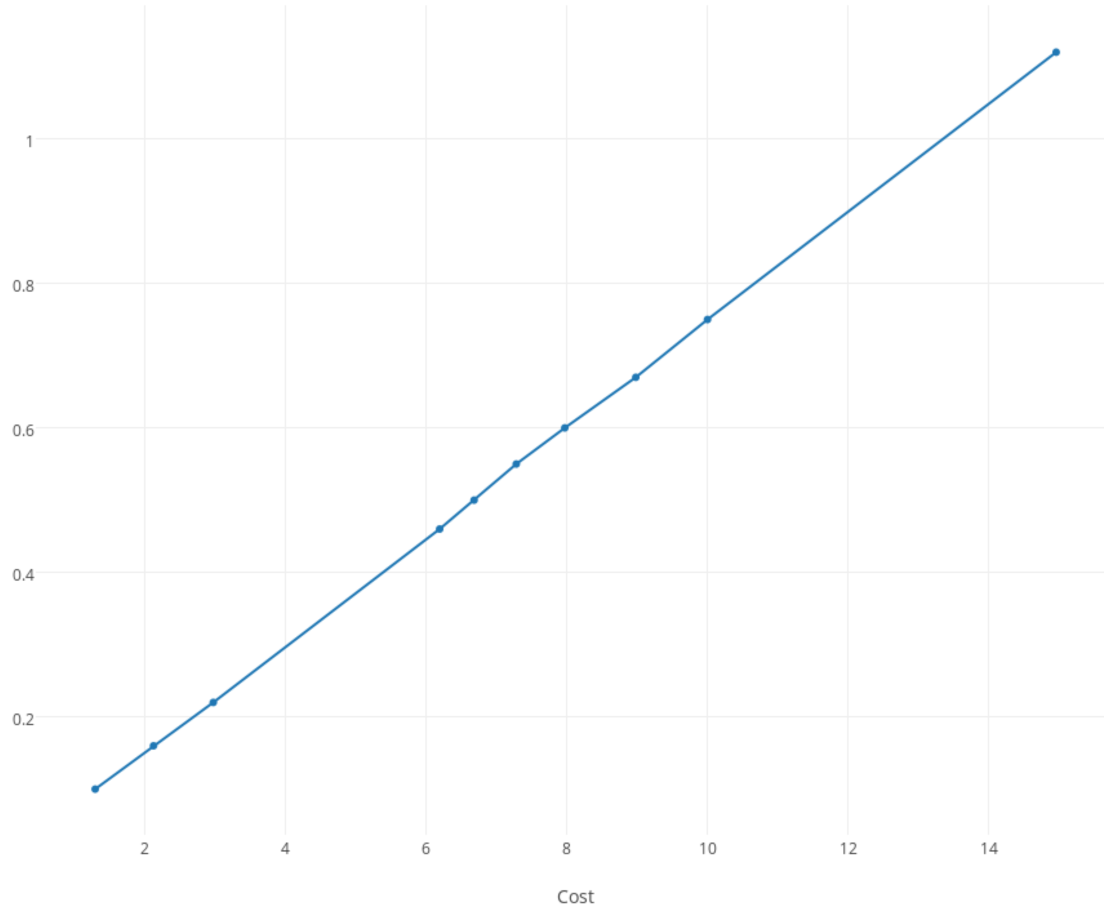
Plotly is ready to make a line plot, so we **choose as x** the cost, and **choose as y** the tax, and then click on the big blue LINE PLOT.



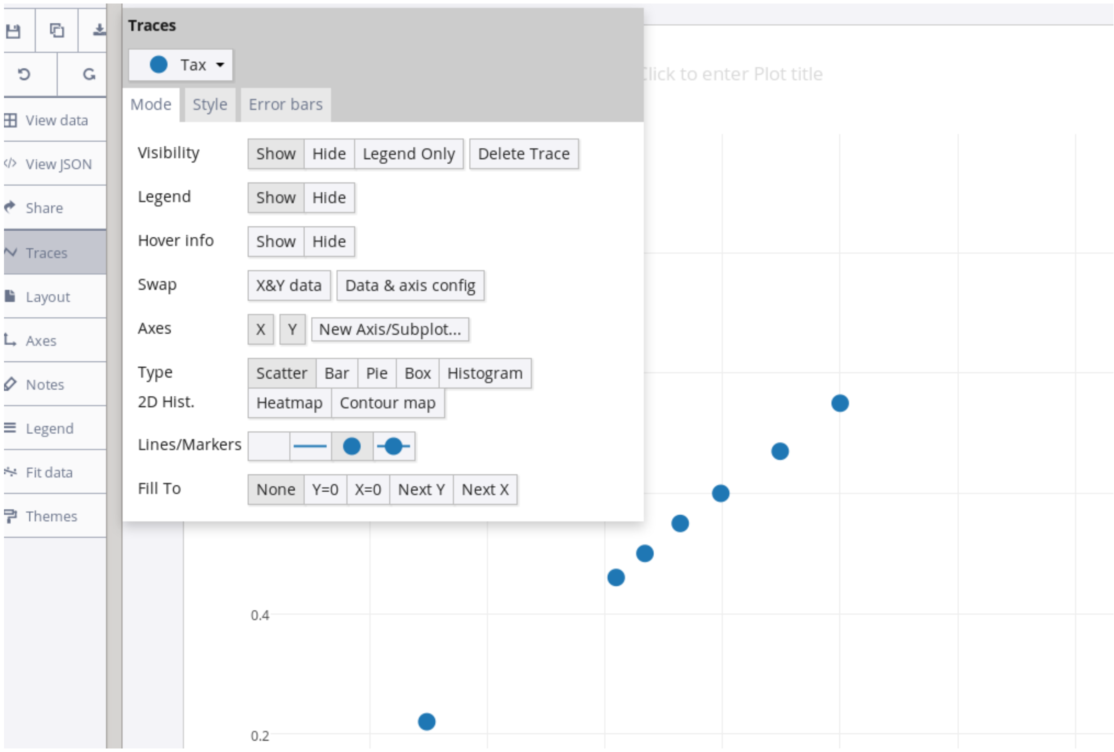
Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

	Index	Item	Cost	Tax	Total
x	choose as x	choose as x	choose as x	choose as x	choose as x
y	choose as y	choose as y	choose as y	choose as y	choose as y
1	1	Secret Antiperspirant	1.29	0.1	1.39
2	2	Bertoli Alfredo Sauce	2.12	0.16	2.28
3	3	Rawlings Little League Baseball	2.97	0.22	3.19
4	4	Large Paperclips, 10 boxes	6.19	0.46	6.65
5	5	Banana Boat Sunscreen, 8 oz	6.68	0.5	7.18

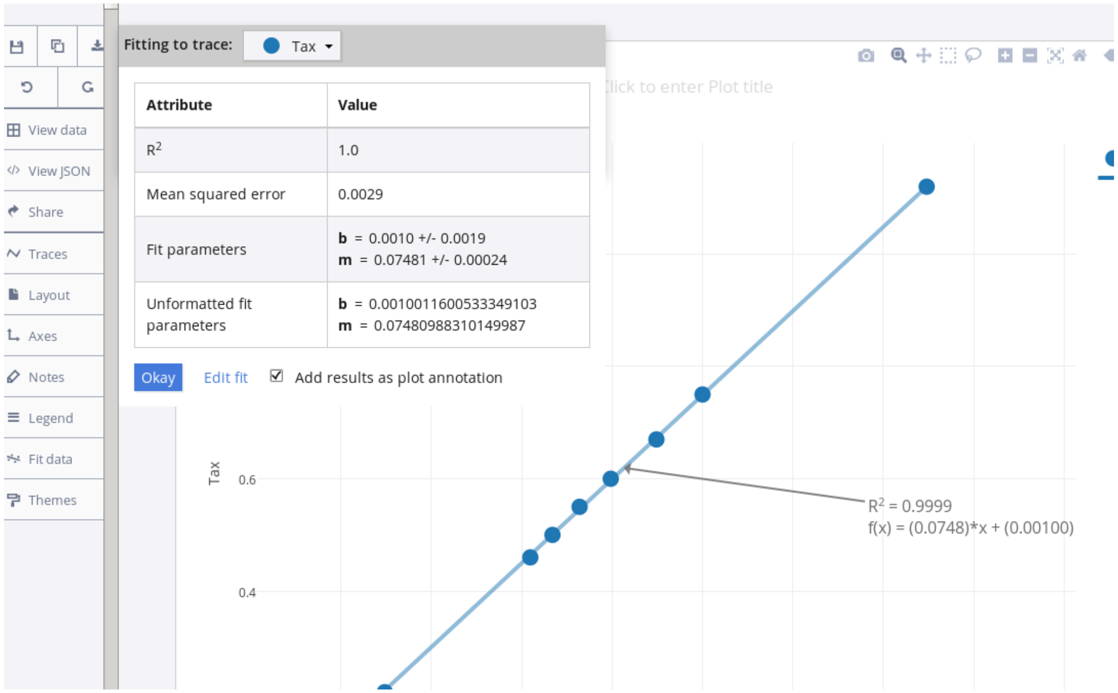
Plotly's data dots are too small, but I used the **Traces** menu item, the **Style** tab, and the **Marker** menu to magnify them.



I'd like to compare the data to a formula, so I made the dots bigger with the **Traces** menu and **Style** tab, then used the **Mode** tab and **Lines/Markers** option to switch from dots-and-lines to dots.



Now the **Fit Data** menu allows me to compute a straight line estimate of the data, which turns out pretty close to my formula. (Why is it not exact?)



Fitting to trace: ● Tax

Attribute	Value
R^2	1.0
Mean squared error	0.0029
Fit parameters	**b** = 0.0010 +/- 0.0019 **m** = 0.07481 +/- 0.00024
Unformatted fit parameters	**b** = 0.0010011600533349103 **m** = 0.07480988310149987

Okay Edit fit ☑ Add results as plot annotation

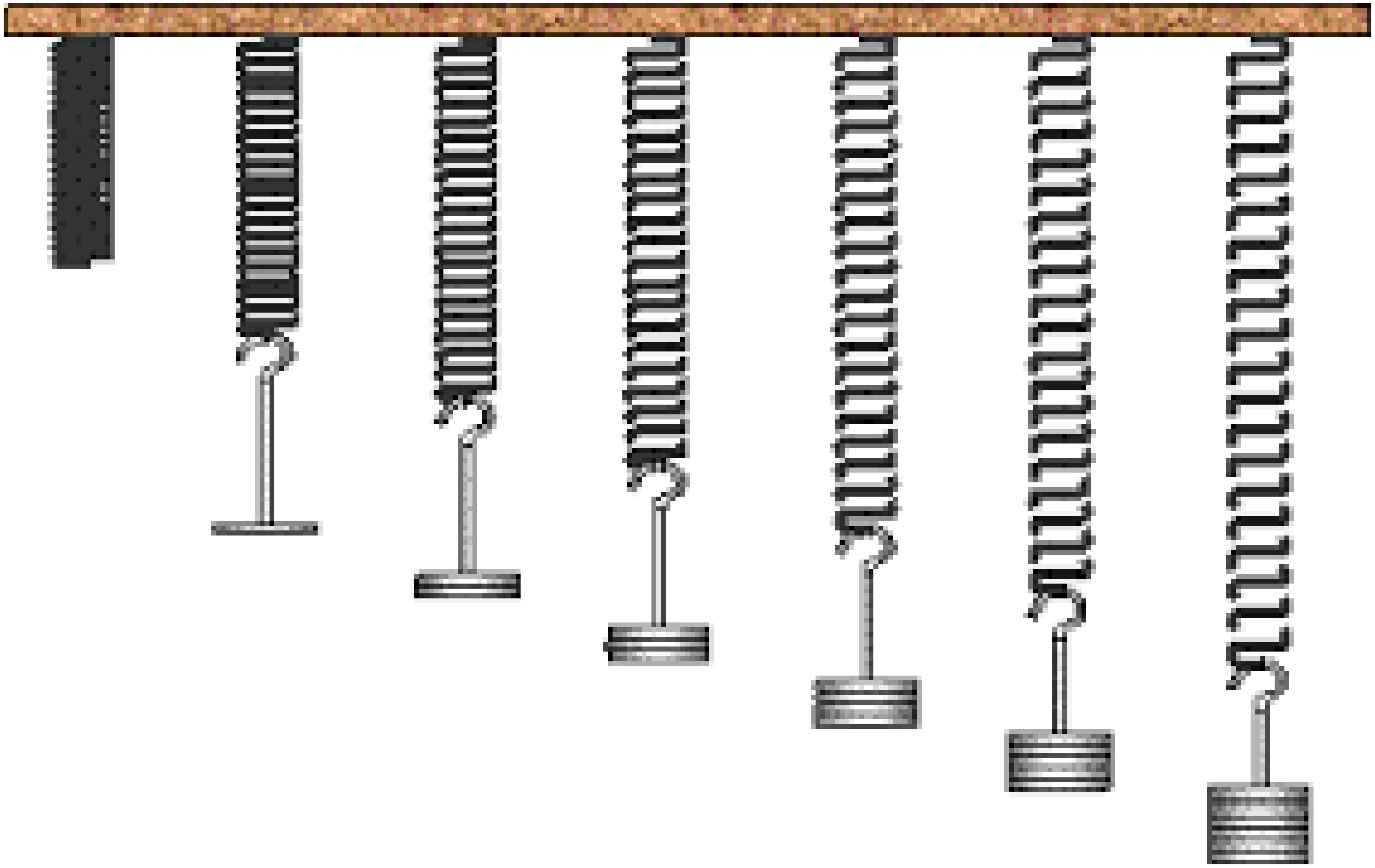
Example 2: Stretching a Spring

A rubber band is useful because it has a property known as elasticity.

We know a few things about a rubber band:

- it has a natural length of one or two inches;
- as we pull on it harder and harder, it stretches to six or seven inches;
- if we let go, it snaps back;
- if we pull too hard, it breaks.

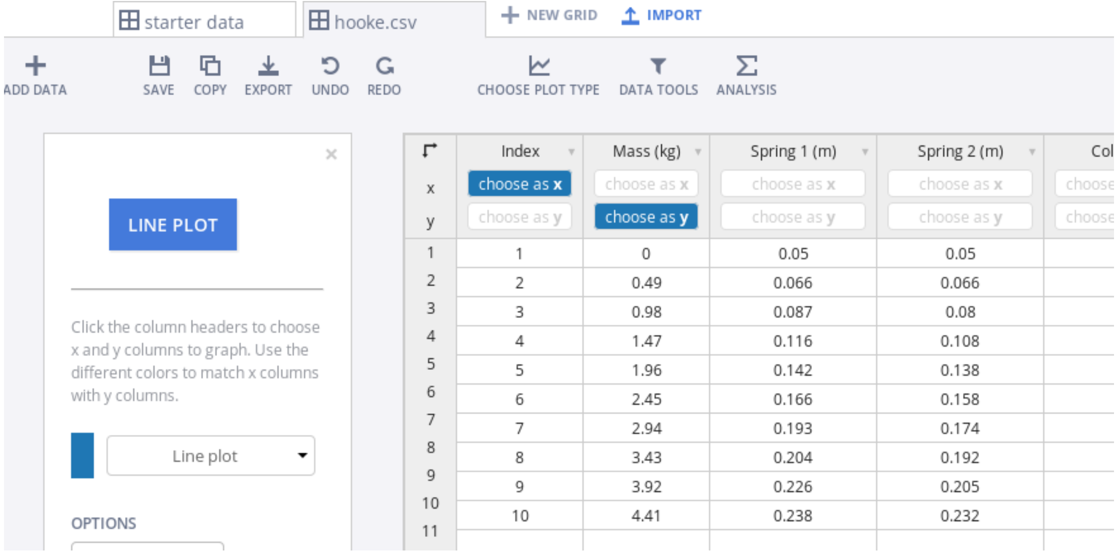
Robert Hooke searched for a pattern to describe the stretching of a spring.



Here is sample data stored in the file *hooke.csv*, in which two springs were stretched by adding one more mass at a time to the end of the spring and measuring the position of the end of the spring.

"Index",	"Mass (kg)",	"Spring 1 (m)",	"Spring 2 (m)"
1,	0.00,	0.050,	0.050
2,	0.49,	0.066,	0.066
3,	0.98,	0.087,	0.080
—	*more data*	—	—
8,	3.43,	0.204,	0.192
9,	3.92,	0.226,	0.205
10,	4.41,	0.238,	0.232

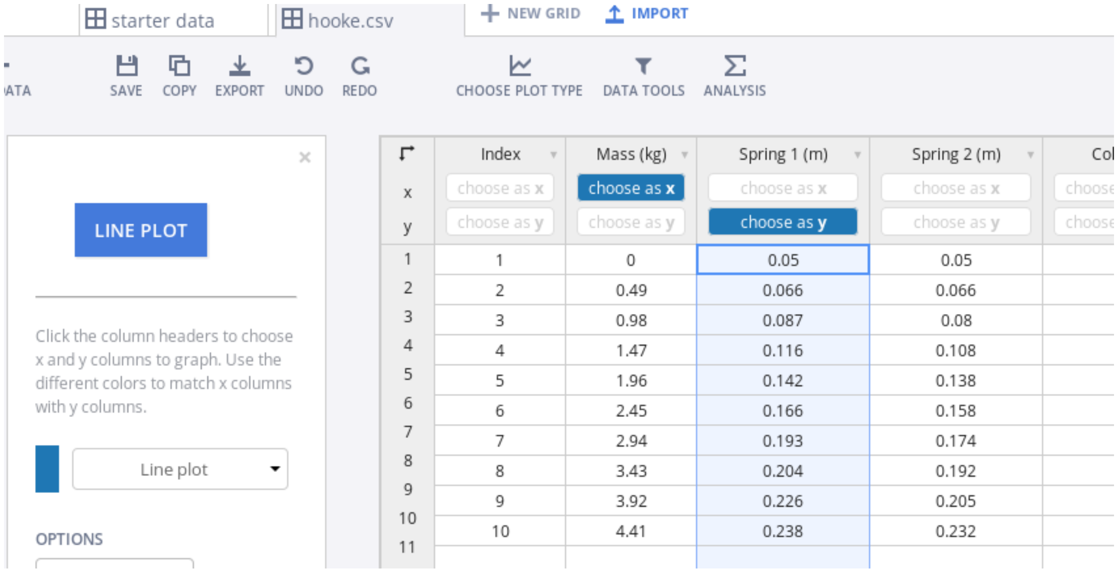
To examine this data, we can log into Plotly, then use the “import” menu to browse for the file *hooke.csv* and bring it into the Plotly grid, so that we see something like this:



Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

	Index	Mass (kg)	Spring 1 (m)	Spring 2 (m)	Col
x	choose as x	choose as x	choose as x	choose as x	choose
y	choose as y	choose as y	choose as y	choose as y	choose
1	1	0	0.05	0.05	
2	2	0.49	0.066	0.066	
3	3	0.98	0.087	0.08	
4	4	1.47	0.116	0.108	
5	5	1.96	0.142	0.138	
6	6	2.45	0.166	0.158	
7	7	2.94	0.193	0.174	
8	8	3.43	0.204	0.192	
9	9	3.92	0.226	0.205	
10	10	4.41	0.238	0.232	
11					

We need to change “Choose as x” and “Choose as y” so that we plot Mass versus Spring 1 Length,

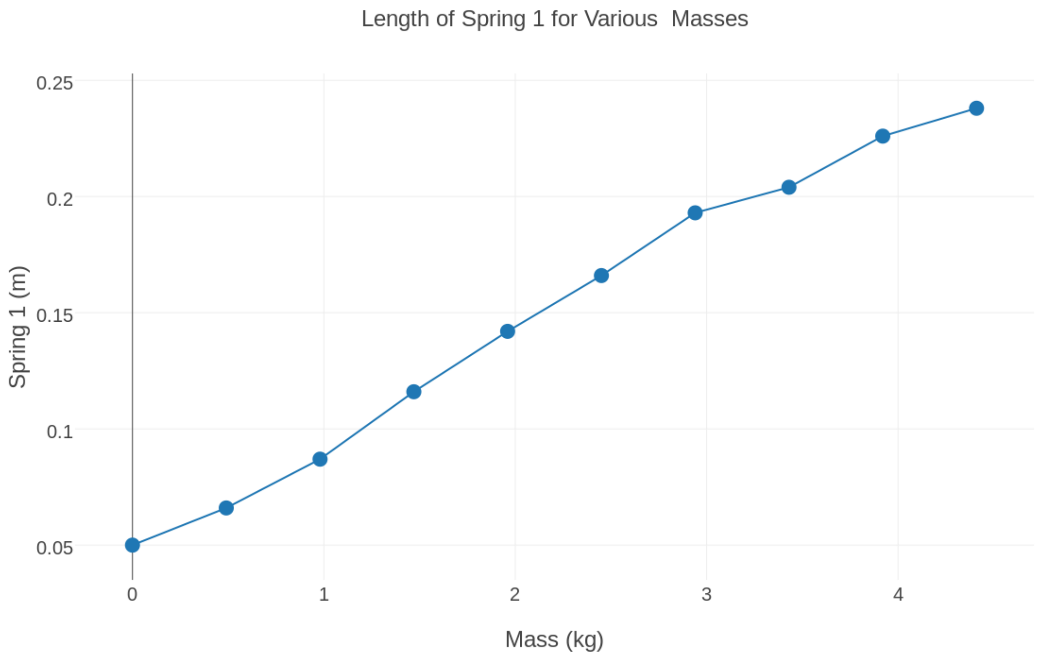


Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

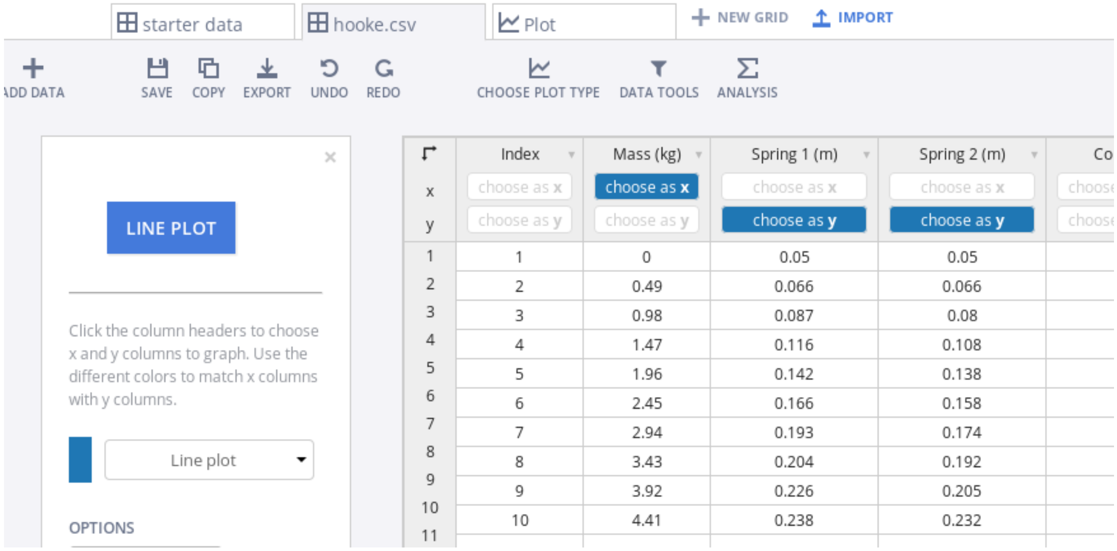
Line plot

	Index	Mass (kg)	Spring 1 (m)	Spring 2 (m)	Col
x	choose as x	choose as x	choose as x	choose as x	choose
y	choose as y	choose as y	choose as y	choose as y	choose
1	1	0	0.05	0.05	
2	2	0.49	0.066	0.066	
3	3	0.98	0.087	0.08	
4	4	1.47	0.116	0.108	
5	5	1.96	0.142	0.138	
6	6	2.45	0.166	0.158	
7	7	2.94	0.193	0.174	
8	8	3.43	0.204	0.192	
9	9	3.92	0.226	0.205	
10	10	4.41	0.238	0.232	
11					

Now we have a “broken line” plot of the data for spring 1.



We can go back to the data grid and add the Spring 2 data using "choose as y"

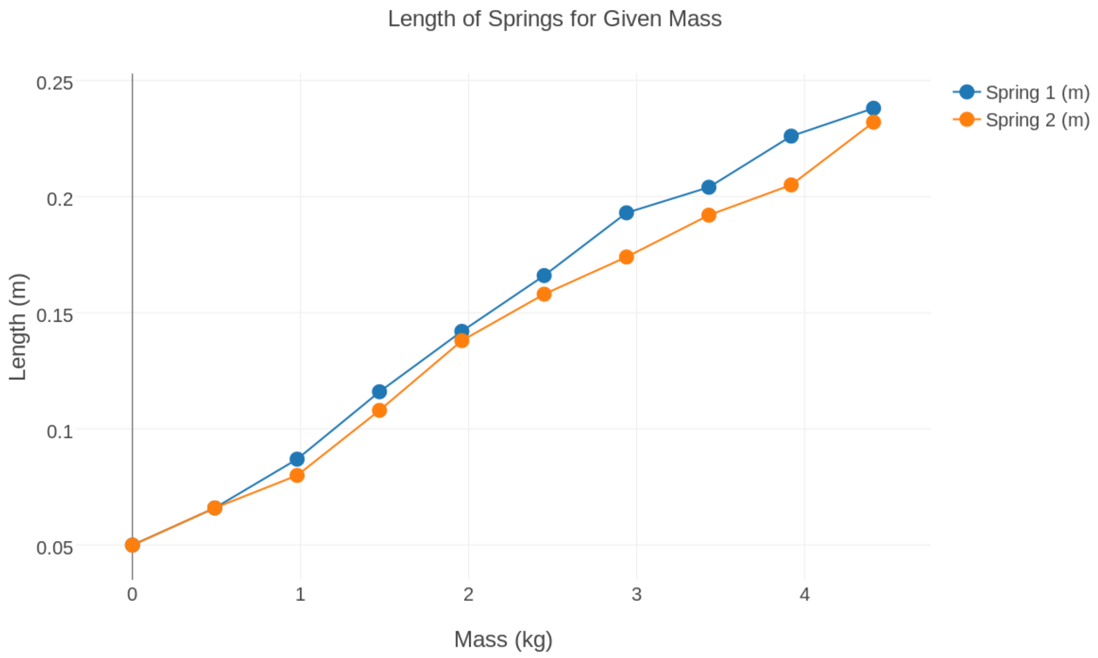


Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

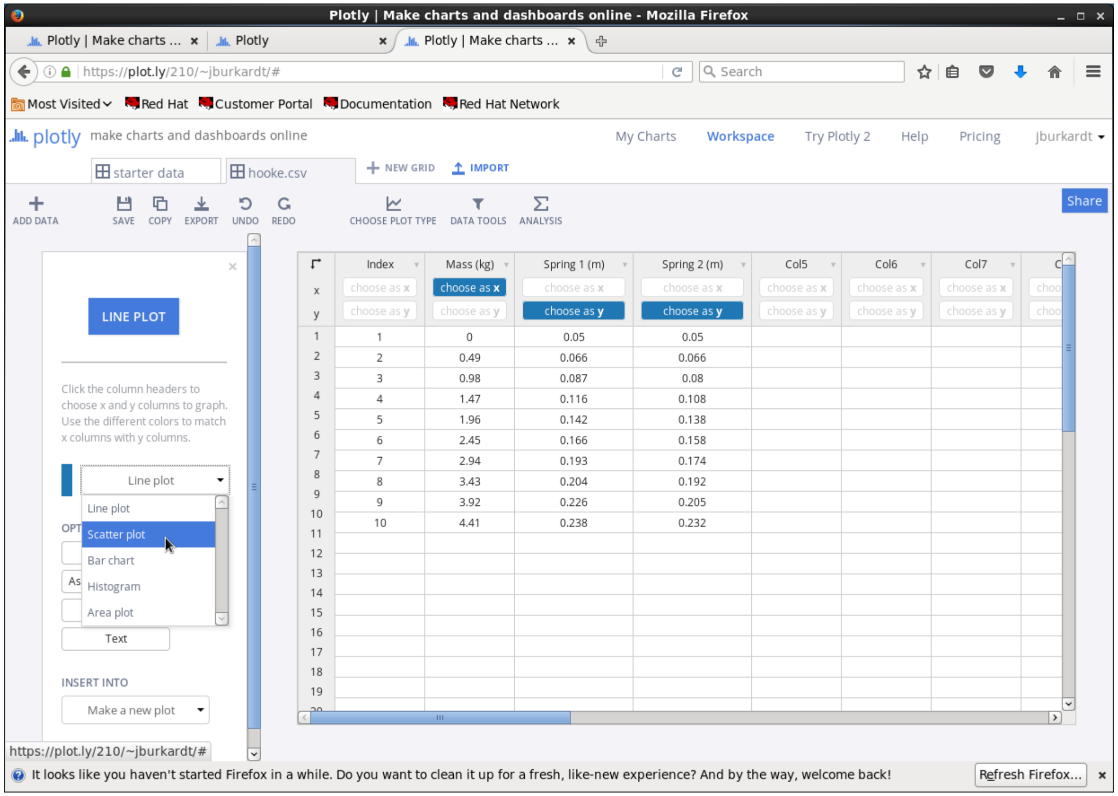
Line plot

	Index	Mass (kg)	Spring 1 (m)	Spring 2 (m)	Co
x	choose as x	choose as x	choose as x	choose as x	choose
y	choose as y	choose as y	choose as y	choose as y	choose
1	1	0	0.05	0.05	
2	2	0.49	0.066	0.066	
3	3	0.98	0.087	0.08	
4	4	1.47	0.116	0.108	
5	5	1.96	0.142	0.138	
6	6	2.45	0.166	0.158	
7	7	2.94	0.193	0.174	
8	8	3.43	0.204	0.192	
9	9	3.92	0.226	0.205	
10	10	4.41	0.238	0.232	
11					

The blue and orange lines are for springs 1 and 2.



Our data doesn't lie on straight lines. Let's redraw the points as scatterplots.

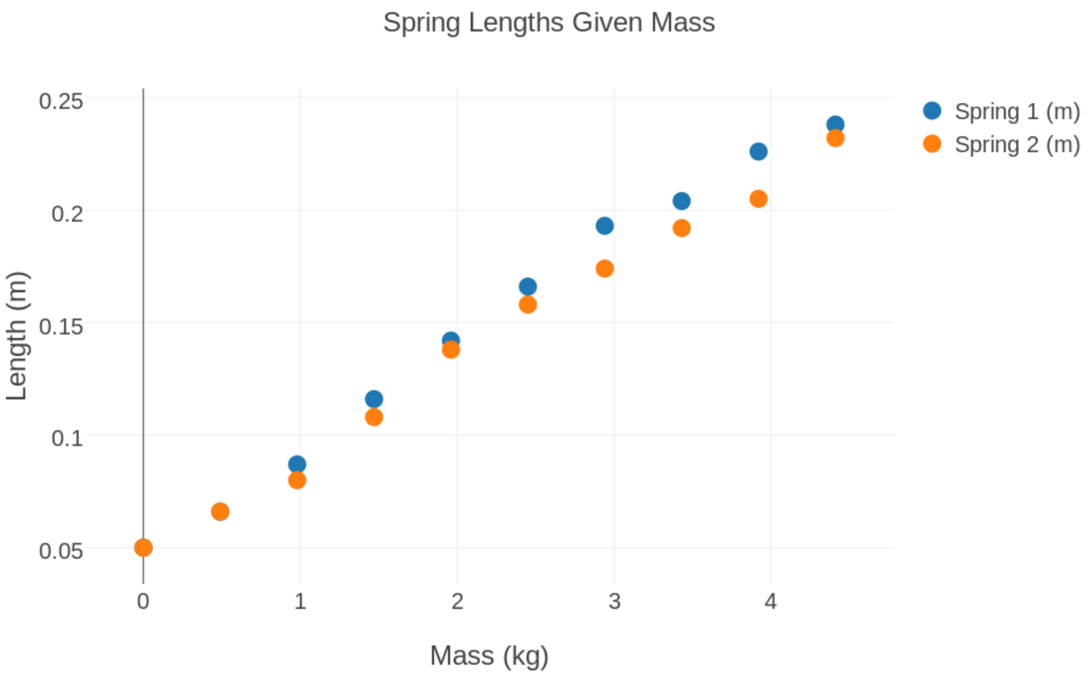


Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

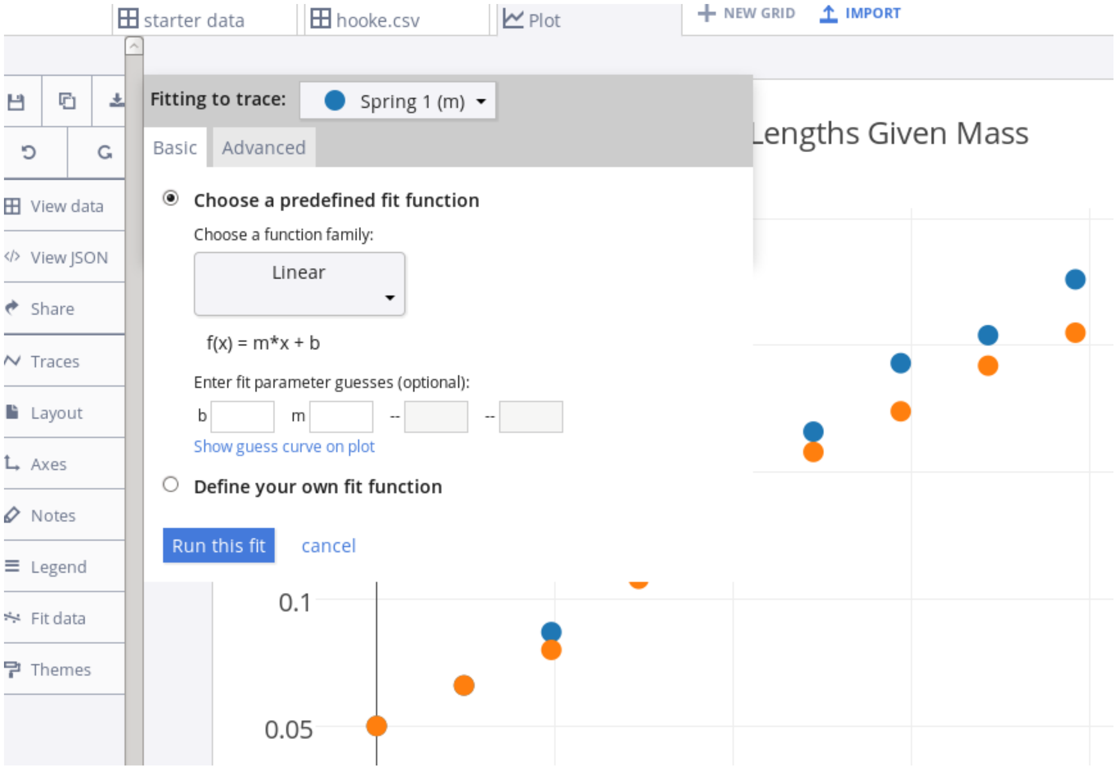
	Index	Mass (kg)	Spring 1 (m)	Spring 2 (m)	Col5	Col6	Col7
x		choose as x					
y			choose as y	choose as y			
1	1	0	0.05	0.05			
2	2	0.49	0.066	0.066			
3	3	0.98	0.087	0.08			
4	4	1.47	0.116	0.108			
5	5	1.96	0.142	0.138			
6	6	2.45	0.166	0.158			
7	7	2.94	0.193	0.174			
8	8	3.43	0.204	0.192			
9	9	3.92	0.226	0.205			
10	10	4.41	0.238	0.232			

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Here is just the data.



By using the Fit data menu, we can request a line like the (blue) data.



Fitting to trace: ● Spring 1 (m)

● Choose a predefined fit function

Choose a function family: Linear

f(x) = m*x + b

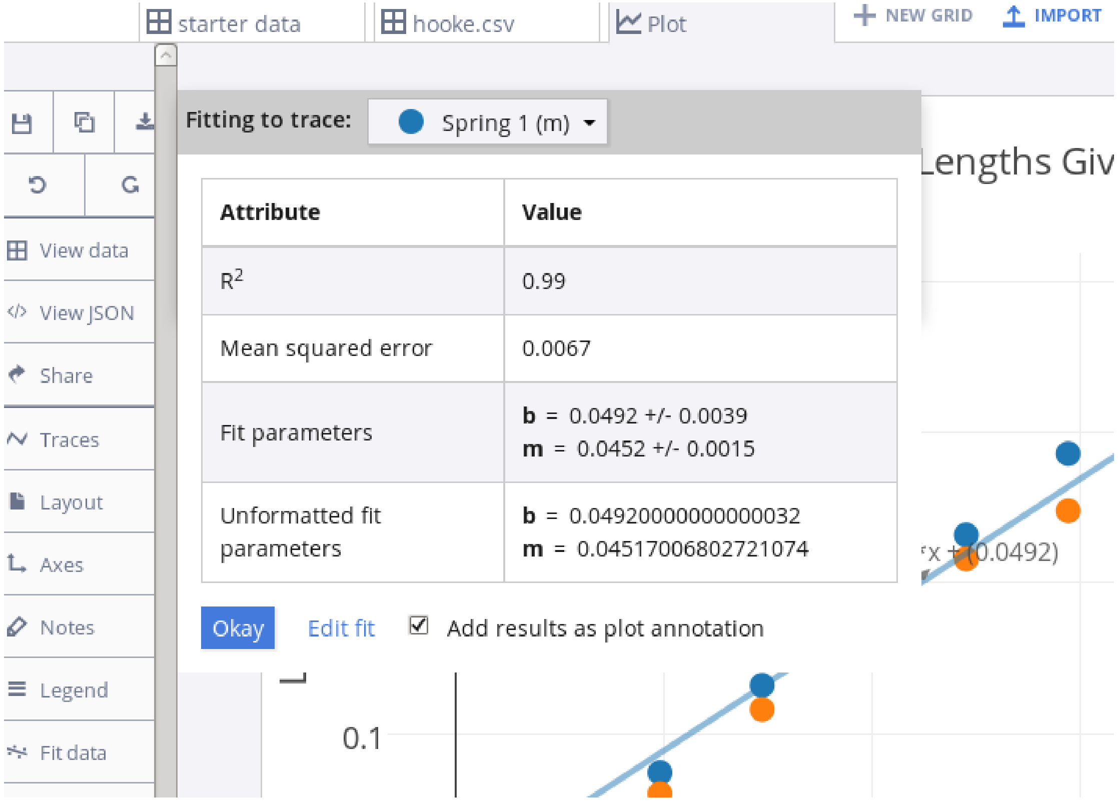
Enter fit parameter guesses (optional):

b □ m □ -- □ -- □

Show guess curve on plot

○ Define your own fit function

Plotly finds a formula $y = 0.0452x + 0.0492$ that best fits the blue data.

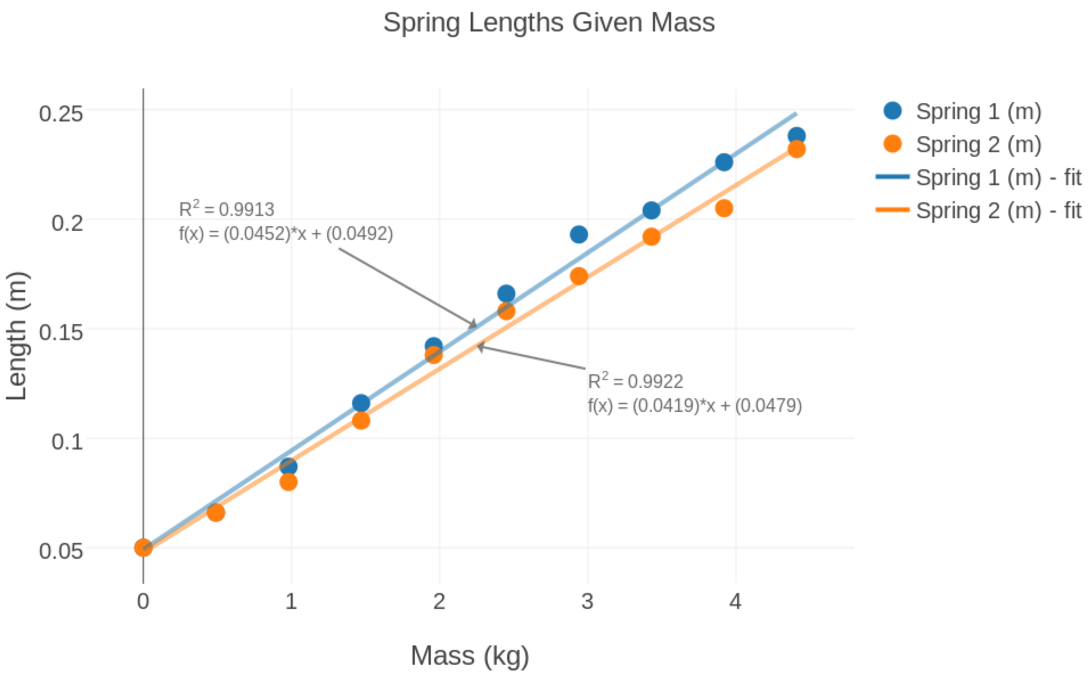


Fitting to trace: ● Spring 1 (m)

Attribute	Value
R^2	0.99
Mean squared error	0.0067
Fit parameters	**b** = 0.0492 +/- 0.0039 **m** = 0.0452 +/- 0.0015
Unformatted fit parameters	**b** = 0.04920000000000032 **m** = 0.04517006802721074

Okay Edit fit ☑ Add results as plot annotation

We can also get a fitting line for the orange data.



Example 3: Housing Data from Zillow

A realtor considers many factors when suggesting a listing price for a house:

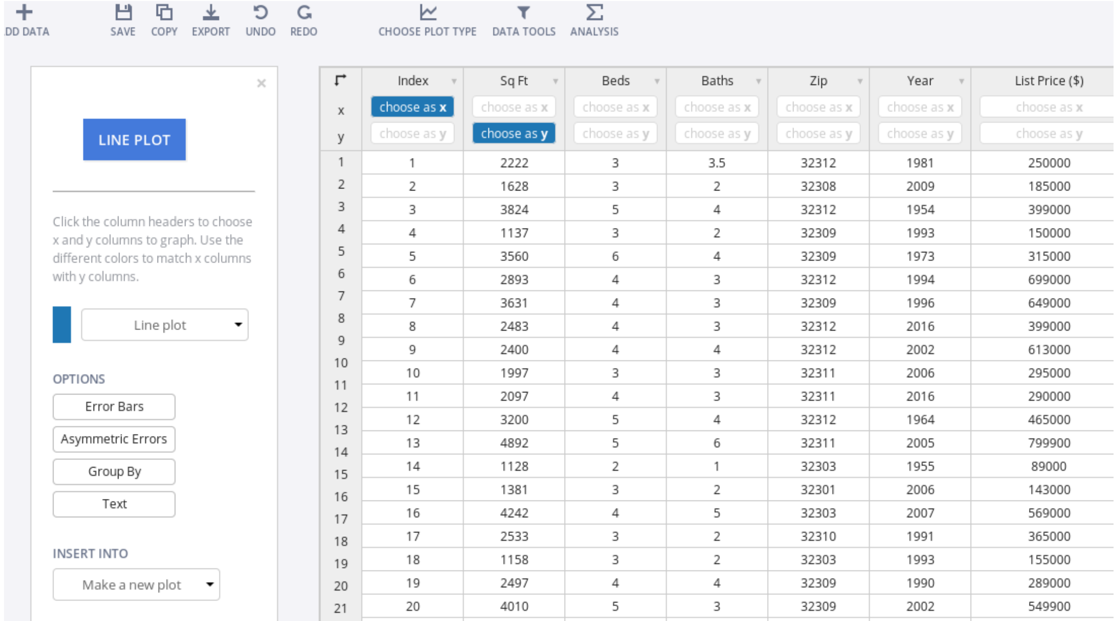
- the square footage;
- number of bedrooms and bathrooms;
- the zip code;
- the year the house was built.

Experience suggests that the square footage is one of the most important.

Here is sample data stored in the file *zillow.csv*

”Index”,	”(Sq Ft)”,	”Beds”,	”Baths”,	”Zip”,	”Year”,	”List Price ($)”
1,	2222,	3,	3.5,	32312,	1981,	250000
2,	1628,	3,	2,	32308,	2009,	185000
3,	3824,	5,	4,	32312,	1954,	399000
—	—	*more data*	—	—	—	—
18,	1158,	3,	2,	32303,	1993,	155000
19,	2497,	4,	4,	32309,	1990,	289000
20,	4010,	5,	3,	32309,	2002,	549900

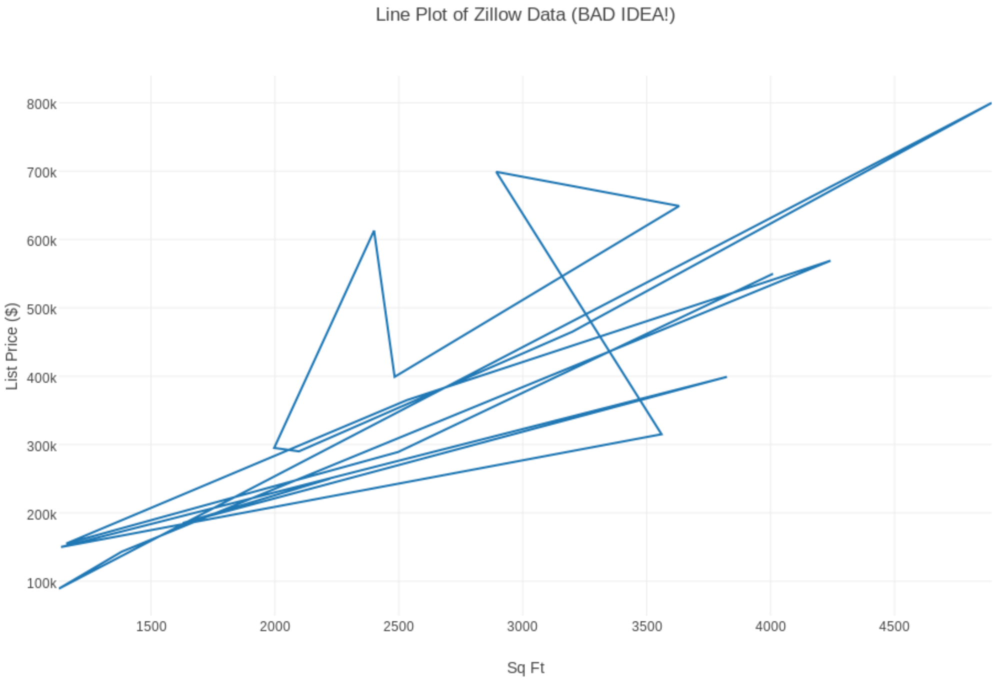
To examine this data, we can log into Plotly, then use the "import" menu to browse for the file *zillow.csv* and bring it into the Plotly grid, so that we see something like this:



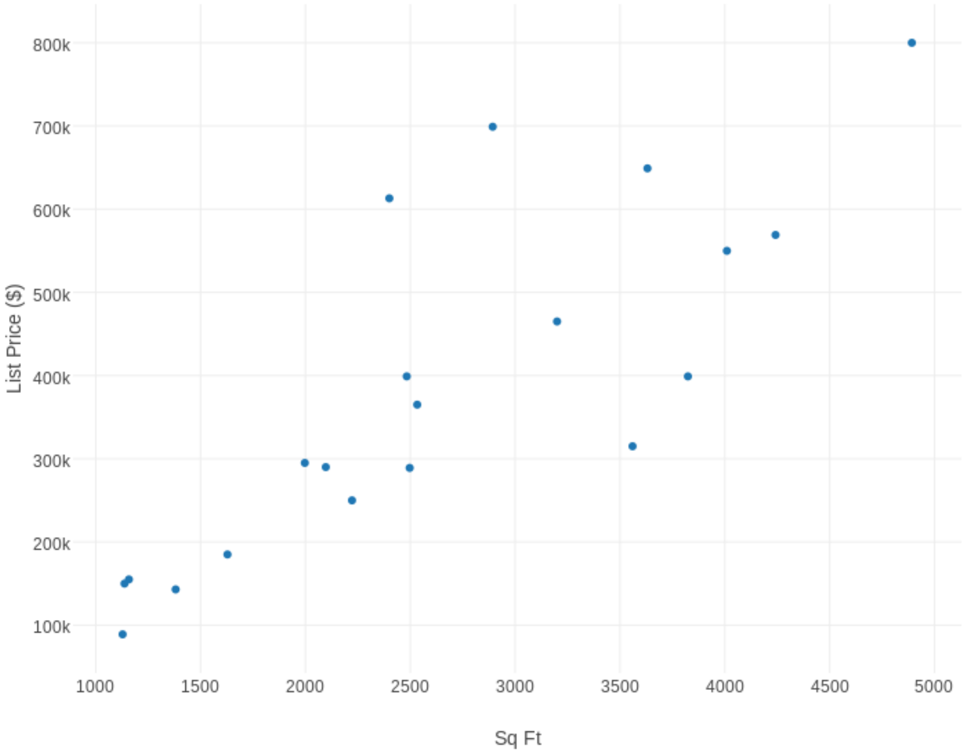
Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

	Index	Sq Ft	Beds	Baths	Zip	Year	List Price ($)
x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x
y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y
1	1	2222	3	3.5	32312	1981	250000
2	2	1628	3	2	32308	2009	185000
3	3	3824	5	4	32312	1954	399000
4	4	1137	3	2	32309	1993	150000
5	5	3560	6	4	32309	1973	315000
6	6	2893	4	3	32312	1994	699000
7	7	3631	4	3	32309	1996	649000
8	8	2483	4	3	32312	2016	399000
9	9	2400	4	4	32312	2002	613000
10	10	1997	3	3	32311	2006	295000
11	11	2097	4	3	32311	2016	290000
12	12	3200	5	4	32312	1964	465000
13	13	4892	5	6	32311	2005	799900
14	14	1128	2	1	32303	1955	89000
15	15	1381	3	2	32301	2006	143000
16	16	4242	4	5	32303	2007	569000
17	17	2533	3	2	32310	1991	365000
18	18	1158	3	2	32303	1993	155000
19	19	2497	4	4	32309	1990	289000
20	20	4010	5	3	32309	2002	549900
21							

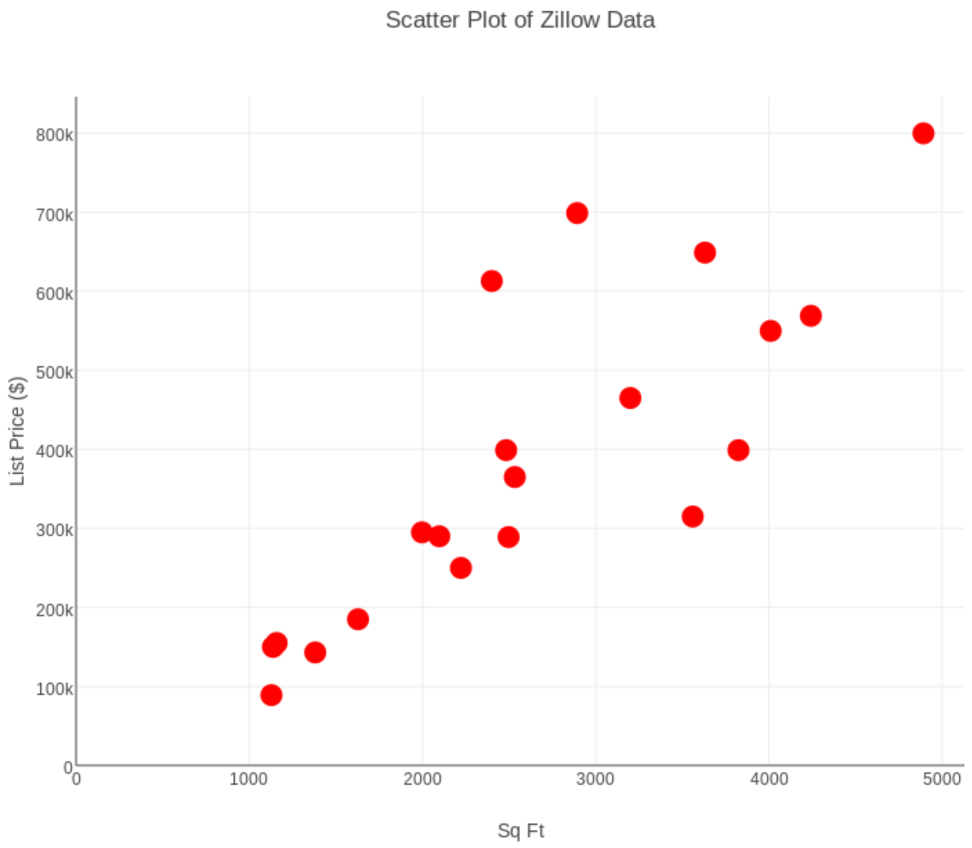
We can **choose as x** square footage, and **choose as y** the price and hit Line Plot, but here's what we get:



Our data is not suitable for line plots; we can make a scatter plot.



The **Traces** menu changes the size and color of dots. The **Axes** menu shifts the graph so we see the zero values for Footage and Price.



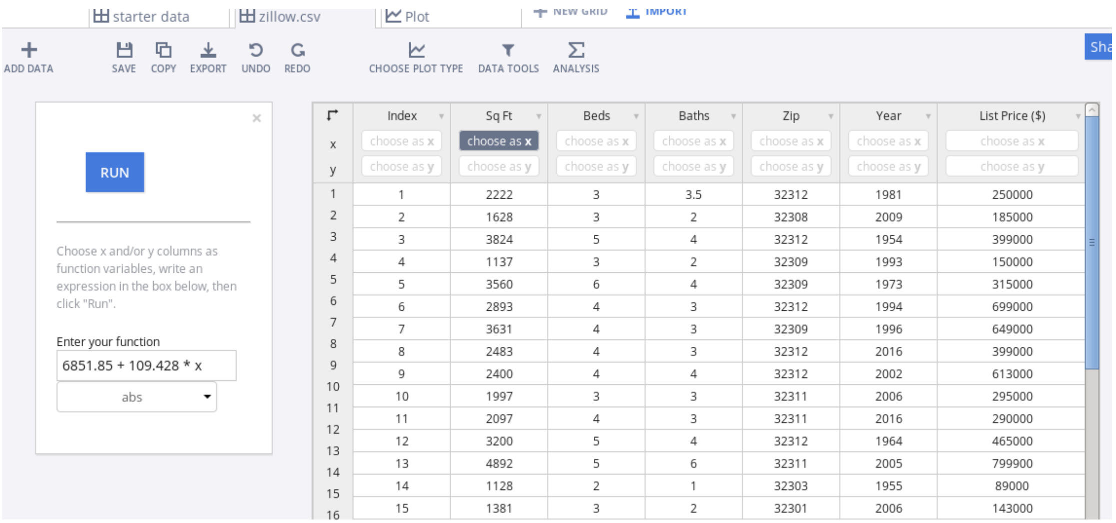
Now it seems like there is a general trend of the data, that the list price increases with the square footage.

If we think about it carefully, we can even find a formula that exact predicts the relationship between footage and price, if we just consider the first two pairs of data:

```
footage x = 2,222, price y = $250,000
footage x = 1,628, price y = $185,000
```

The formula is $y = 6851.85 + 109.428x$. We'd like to compare this formula to all 20 of our data points.

If we select the Data Tools item, and then formula f(x), we see a box to enter our formula:



Choose x and/or y columns as function variables, write an expression in the box below, then click "Run".

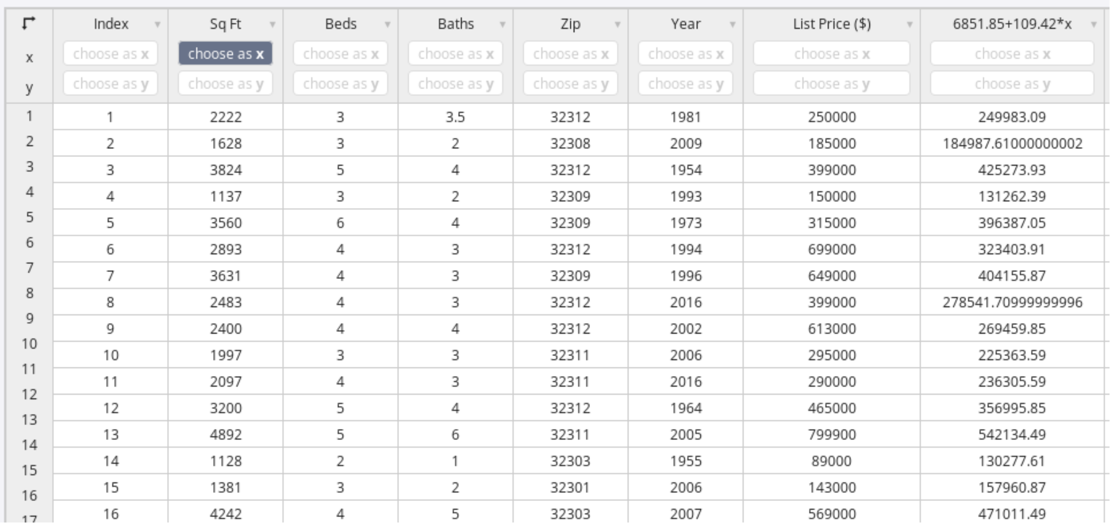
Enter your function

6851.85 + 109.428 * x

abs

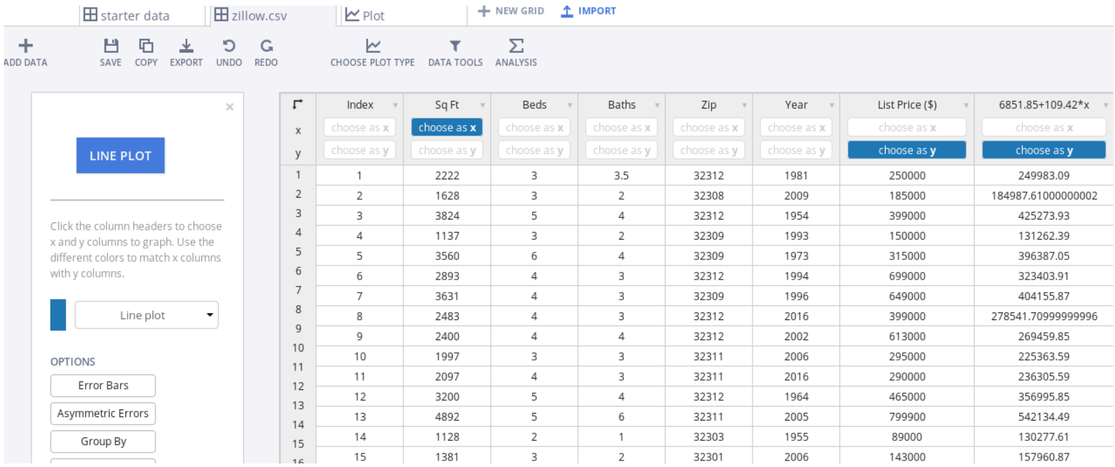
	Index	Sq Ft	Beds	Baths	Zip	Year	List Price ($)
x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x
y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y
1	1	2222	3	3.5	32312	1981	250000
2	2	1628	3	2	32308	2009	185000
3	3	3824	5	4	32312	1954	399000
4	4	1137	3	2	32309	1993	150000
5	5	3560	6	4	32309	1973	315000
6	6	2893	4	3	32312	1994	699000
7	7	3631	4	3	32309	1996	649000
8	8	2483	4	3	32312	2016	399000
9	9	2400	4	4	32312	2002	613000
10	10	1997	3	3	32311	2006	295000
11	11	2097	4	3	32311	2016	290000
12	12	3200	5	4	32312	1964	465000
13	13	4892	5	6	32311	2005	799900
14	14	1128	2	1	32303	1955	89000
15	15	1381	3	2	32301	2006	143000

Using square footage as x, Plotly creates a new column of formula values.



	Index	Sq Ft	Beds	Baths	Zip	Year	List Price ($)	6851.85+109.42*x
x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x
y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y
1	1	2222	3	3.5	32312	1981	250000	249983.09
2	2	1628	3	2	32308	2009	185000	184987.61000000002
3	3	3824	5	4	32312	1954	399000	425273.93
4	4	1137	3	2	32309	1993	150000	131262.39
5	5	3560	6	4	32309	1973	315000	396387.05
6	6	2893	4	3	32312	1994	699000	323403.91
7	7	3631	4	3	32309	1996	649000	404155.87
8	8	2483	4	3	32312	2016	399000	278541.70999999996
9	9	2400	4	4	32312	2002	613000	269459.85
10	10	1997	3	3	32311	2006	295000	225363.59
11	11	2097	4	3	32311	2016	290000	236305.59
12	12	3200	5	4	32312	1964	465000	356995.85
13	13	4892	5	6	32311	2005	799900	542134.49
14	14	1128	2	1	32303	1955	89000	130277.61
15	15	1381	3	2	32301	2006	143000	157960.87
16	16	4242	4	5	32303	2007	569000	471011.49

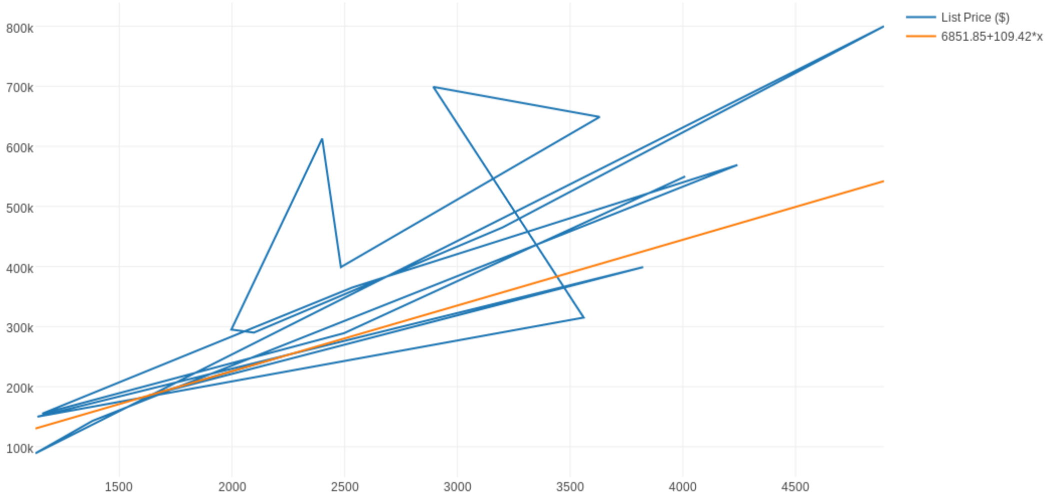
Now we can plan to plot square footage against our data **and** our formula values.



Click the column headers to choose x and y columns to graph. Use the different colors to match x columns with y columns.

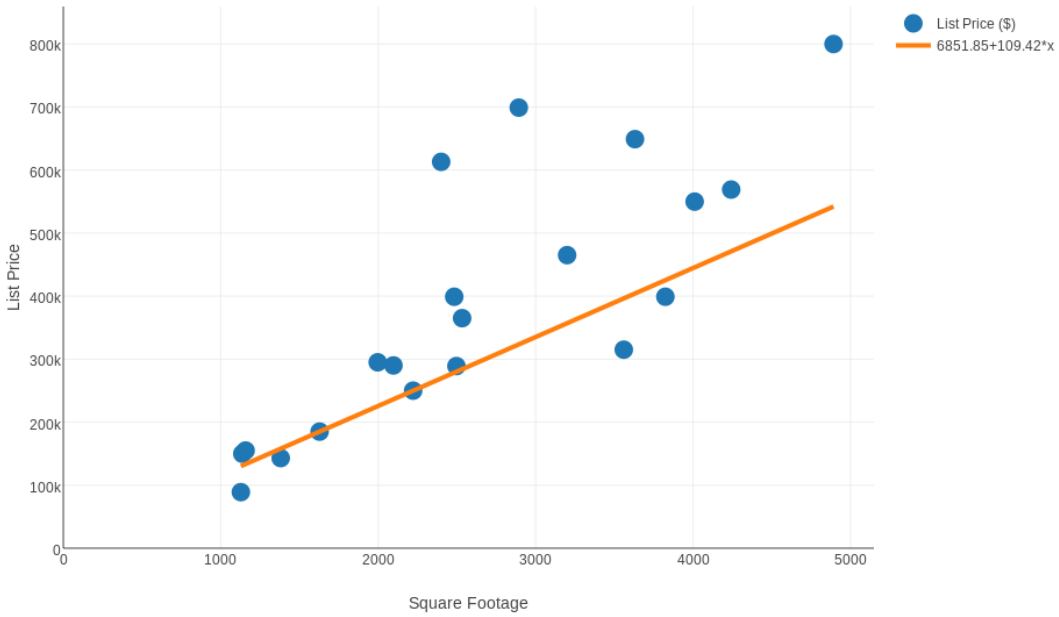
	Index	Sq Ft	Beds	Baths	Zip	Year	List Price ($)	6851.85+109.42*x
x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x	choose as x
y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y	choose as y
1	1	2222	3	3.5	32312	1981	250000	249983.09
2	2	1628	3	2	32308	2009	185000	184987.61000000002
3	3	3824	5	4	32312	1954	399000	425273.93
4	4	1137	3	2	32309	1993	150000	131262.39
5	5	3560	6	4	32309	1973	315000	396387.05
6	6	2893	4	3	32312	1994	699000	323403.91
7	7	3631	4	3	32309	1996	649000	404155.87
8	8	2483	4	3	32312	2016	399000	278541.70999999996
9	9	2400	4	4	32312	2002	613000	269459.85
10	10	1997	3	3	32311	2006	295000	225363.59
11	11	2097	4	3	32311	2016	290000	236305.59
12	12	3200	5	4	32312	1964	465000	356995.85
13	13	4892	5	6	32311	2005	799900	542134.49
14	14	1128	2	1	32303	1955	89000	130277.61
15	15	1381	3	2	32301	2006	143000	157960.87

Choosing **Line Plot**, Plotly will make a mess of our data (blue), but the line shows up just fine (in orange).



We can clean things up.

- Use **Traces/Mode/Lines/Markers** to change tha marker for the list price data from a blue line to blue dots;
- Use **Traces/Style** to change the size of dots from 6 to 16;
- Use **Axes** to shift the x-axis and y-axis to start at 0;



? Price = 6851.85 + 109.42 Square Footage ?