INFO 2950: Intro to Data Science

Lecture 6
2023-09-11

https://twitter.com/christinelove/status/994442910165807104

Agenda

1. **Time series**
    a. Time series data
    b. Line plots

2. **Regressions**
    a. Motivation
    b. Notation
    c. Interpretation

Time Series

- Data where one axis denotes time

Time →

Time Series

- Data where one axis (**one column**) denotes time

Time	Y
2010	0
2011	3
…	…

Time Series

- Data where one axis (one column) denotes time
- Each row = one "time step"

	Time	Y
one time step	2010	0
another time step	2011	3
	…	…

Time Series

- Most meaningful when data is aggregated so that each "time step":
  1. Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically

Time Series

- Most meaningful when data is aggregated so that each "time step":
  1. Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  2. Has corresponding data per time step

Time Series

- Most meaningful when data is aggregated so that each "time step":
  1. Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  2. Has corresponding data per time step
  3. Is unique

Time Series

- Most meaningful when data is aggregated so that each "time step":
    1. Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
    2. Has corresponding data per time step
    3. Is unique
    4. Deals with missing values

Time Series

- Most meaningful when data is aggregated so that each "time step":
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has corresponding data per time step
  - **Is unique**
  - Deals with missing values
- **"Daily Temperature Data": each time step represents one day.**

Time Series

- Most meaningful when data is aggregated so that each "time step":
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has corresponding data per time step
  - **Is unique**
  - Deals with missing values
- **"Daily Temperature Data": each time step represents one day.** **How many temperatures do we expect for each day?**

Time Series

- Most meaningful when data is aggregated so that each “time step”:
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has corresponding data per time step
  - **Is unique**
  - Deals with missing values
- **“Daily Temperature Data”: each time step represents one day. How many temperatures do we expect for each day? 1 (i.e., unique per time step)!**

A dataframe called *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

How do we get array data types?

`type(a)` or `a.dtype`?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

How do we get array data types?

`a.dtype` is what we want for each array a!

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

daily_temps_df has date type data

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

```
daily_temps_df.dtypes
```

✓ 0.3s

```
day                         datetime64[ns]
time_of_measurement                 object
temperature                          int64
dtype: object
```

slate
@PleaseBeGneiss

excel: is that a date?

me: 57.39 is very much not a date

excel: strong date vibes to me

me: h-how

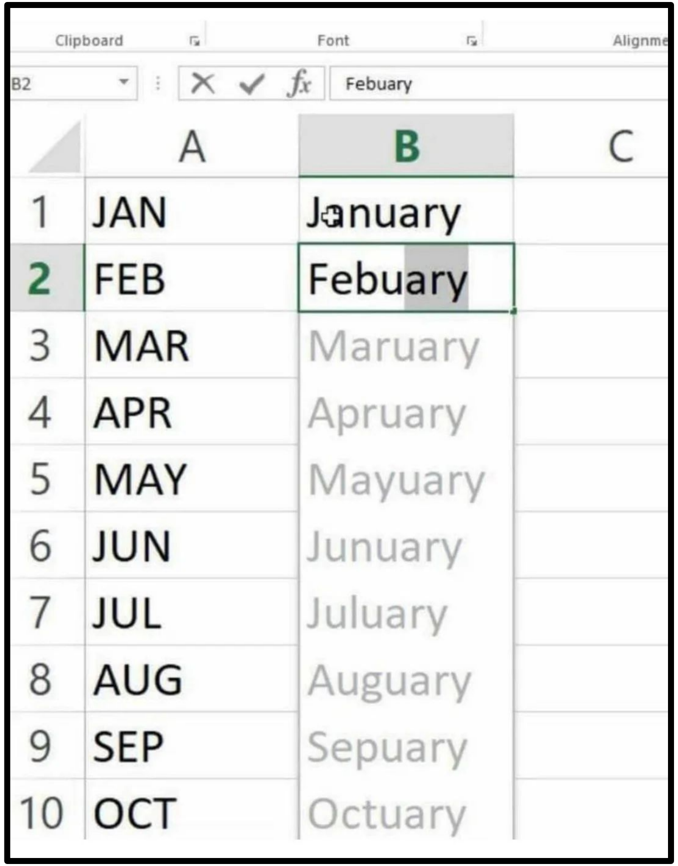
excel: fixed it

me: 57/39/2020?

excel: you’re welcome

11:23 AM · Nov 17, 2020

https://twitter.com/PleaseBeGneiss/status/1328735477923336192 https://twitter.com/ExcelHumor/status/1558965529506889728



https://twitter.com/PleaseBeGneiss/status/1328735477923336192 https://twitter.com/ExcelHumor/status/1558965529506889728

Date issues in the wild?

Date issues in the wild?

- Non-standard formatting around the world
  - month/day vs. day/month (Sep 10th vs. Oct 9th)
  - Even if the same ordering, -YYYY vs. .YY vs. ,Year (09-10-2023 vs. 09.10.23 vs. September 10th, 2023)
  - Leading zeroes (09-10-2023 vs. 9-10-2023)

Date issues in the wild?

- Non-standard formatting around the world
  - **month/day vs. day/month** (Sep 10th vs. Oct 9th)
  - Even if the same ordering, **-YYYY vs. .YY vs. ,Year** (09-10-2023 vs. 09.10.23 vs. September 10th, 2023)
  - **Leading zeroes** (09-10-2023 vs. 9-10-2023)

- **Sorting is annoying** (e.g. sorting "Monday" thru "Friday" is alphabetical, not in MTWThF order)
  - MM-DD-YY: "12-31-2023" < "9-10-2023"
  - DD-MM-YY: "01-12-2023" < "31-9-2023"

Now throw in *time* too...

- Hours:minutes:seconds? **HH:MM:SS**?
- Are all clock measurements of a **millisecond** identical?
- What about tracking **time zones**?
- What about when *exactly* **daylight savings time** happens? What about **leap years**?
- What if you generate data while flying over the **international date line**?
- ...

https://gist.github.com/timvisee/fcda9bbdff88d45cc9061606b4b923ca

Is the day before Saturday always Friday?

https://gist.github.com/timvisee/fcda9bbdff88d45cc9061606b4b923ca

No!

Samoa jumps forward in time

Published 9-May-2011. Changed 1-Sep-2011

Samoa will switch time zones by redrawing the international dateline.

The change will occur at midnight on December 29, 2011, taking the Pacific island nation straight into December 31, 2011. Neighboring American Samoa will remain on the eastern side of the dateline, resulting in a time difference of a whole day between the two territories, which are a mere 30 miles apart.

Any guesses why Samoa did this?

https://www.timeanddate.com/news/time/samoa-dateline.html

Samoa jumps forward in time

Published 9-May-2011. Changed 1-Sep-2011

Samoa will switch time zones by redrawing the international dateline.

The change will occur at midnight on December 29, 2011, taking the Pacific island nation straight into December 31, 2011. Neighboring American Samoa will remain on the eastern side of the dateline, resulting in a time difference of a whole day between the two territories, which are a mere 30 miles apart.

Prime Minister Tuilaepa Sailele Malielegaoi maintains that this constricts Samoa's economy: "In doing business with New Zealand and Australia, we're losing out on two working days a week," he told the government newspaper *Sivali*. "While it's Friday here, it's Saturday in New Zealand, and when we're at church on Sunday, they're already conducting business in Sydney and Brisbane."

https://www.timeanddate.com/news/time/samoa-dateline.html

Moral of the story

- Do not try to be clever when dealing with dates & times **yourself**

- Professionals have dealt with this for *years*; you should **rely on packages** they've written for this

https://me.me/i/everytime-i-call-in-datetime-for-python-2b999decf29b45328203e491270a0228

Data types for dates

- If you have any time-related data, **USE DATETIME**!
  - **Lessens confusion** (Sep 10th vs. Oct 9th)
  - **Sorts correctly**, plots in correct order
  - Allows you to **extract important parts** of your time data (e.g., month, day, day of week, etc.) & **format** as desired

Data types for dates

- Convert from string to datetime in pandas:

```
daily_temps_df["day"]=pd.to_datetime(daily_temps_df["day"])
```

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

How do we make *daily_temps_df* meaningful?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

- aggregated so that each “time step”:
    - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
    - Has corresponding data per time step
    - Is unique
    - Deals with missing values

How do we make *daily_temps_df* meaningful?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

Datetime default of YYYY-MM-DD
(even if accidentally string)
sorts chronologically!

- aggregated so that each “time step”:
  - Is regularly spaced (e.g. daily, monthly, quarterly data) **chronologically**
  - Has corresponding data per time step
  - Is unique
  - Deals with missing values

How do we make *daily_temps_df* meaningful?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

- aggregated so that each “time step”:
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has **corresponding data per time step**
  - Is unique
  - Deals with missing values

How do we make *daily_temps_df* meaningful?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

- aggregated so that each "time step":
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has corresponding data per time step
  - **Is unique**
  - Deals with missing values

Aggregate *daily_temps_df* : `1` daily avg temp

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

What SQL code yields this new table of averaged daily temperatures?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

SELECT ______, ____________ FROM daily_temps_df ________ __ ______

What SQL code yields this new table of averaged daily temperatures?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

```
SELECT day, AVG(temperature) FROM daily_temps_df GROUP BY day
```

Is this a meaningful time series now?

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

SELECT day, AVG(temperature) FROM daily_temps_df GROUP BY day

Is this a meaningful time series now?

- aggregated so that each “time step”:
    - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
    - Has corresponding data per time step
    - Is unique
    - Deals with missing values

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

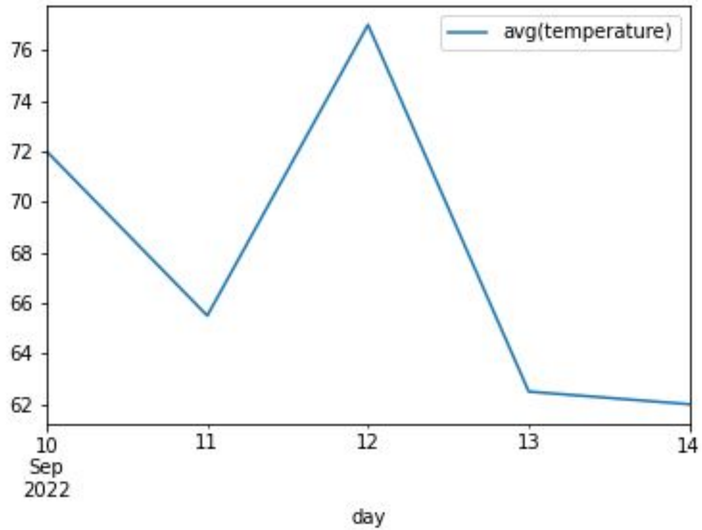
SELECT day, AVG(temperature) FROM daily_temps_df GROUP BY day

Let’s plot the time series!

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

```
avg_temps.plot("day", "avg(temperature)")
```

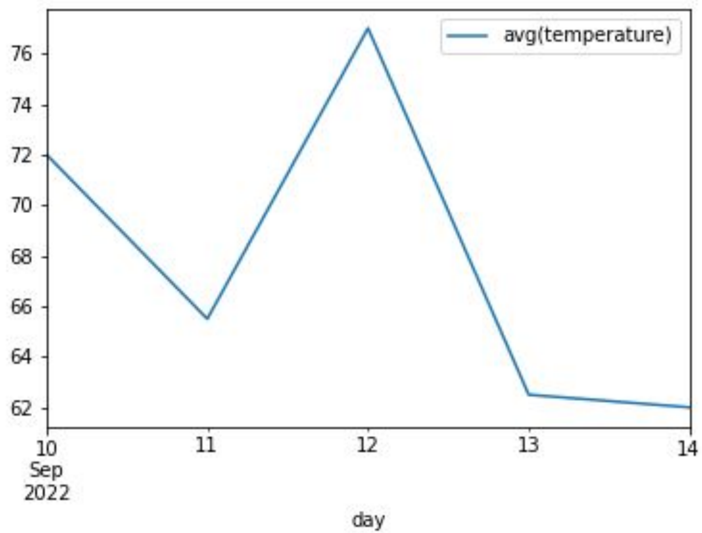
Let's plot the time series!



	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

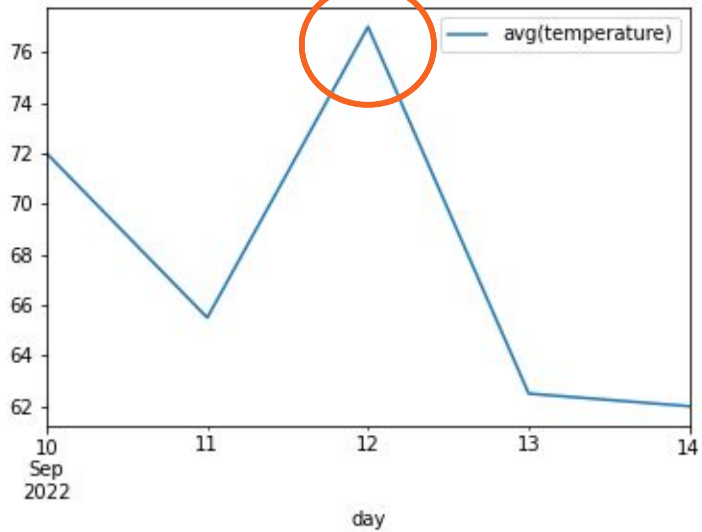
```
avg_temps.plot("day", "avg(temperature)")
```

Are there any outliers here?



	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

Are there any outliers here? Maybe...



	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

Domain knowledge?

Let's look back at our original data...

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

Anything weird?

Let's look back at our original data...

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	avg(temperature)
0	2022-09-10	72.0
1	2022-09-11	65.5
2	2022-09-12	77.0
3	2022-09-13	62.5
4	2022-09-14	62.0

Anything weird?

Let’s look only at temperature highs.

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

Pandas code for this df?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[____________________________ ___ "high"]
```

Pandas code for this df?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[daily_temps_df['time_of_measurement'] == "high"]
```

Need to grab the correct column
(by re-invoking the df)

Pandas code for this df?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[daily_temps_df['time_of_measurement'] == "high"]
```

Don't forget to use quotes for the column name

Pandas code for this df?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[daily_temps_df['time_of_measurement'] == "high"]
```

Double equals in Pandas,
Single equals in SQL

Pandas code for this df?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[daily_temps_df['time_of_measurement'] == "high"]
```

The output of the code in [ ] yields:
[True, False, True, False, True, True, False, True, False]

Pandas code for this df?

table name: *daily_temps_df*

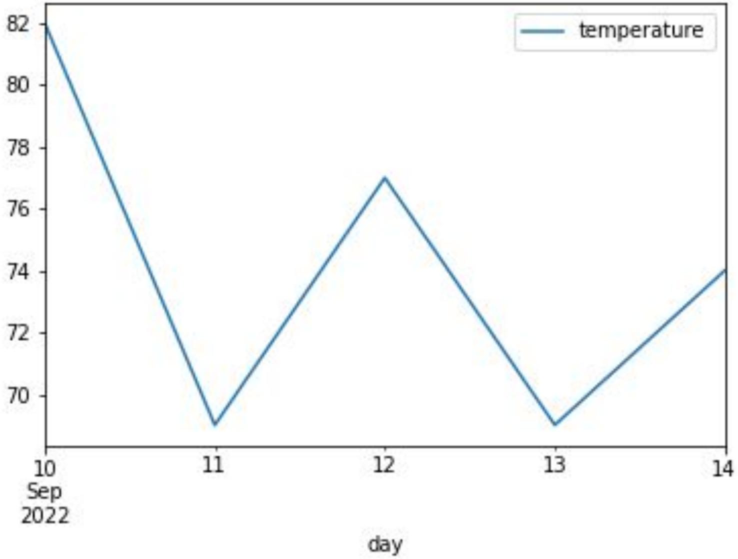
	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
daily_temps_df[daily_temps_df['time_of_measurement'] == "high"]
```

daily_tempts_df[[True, False, True, False, True, True, False, True, False]] restricts to only the rows corresponding to True

What do the highs look like?



	day	time_of_measurement	temperature
0	2022-09-10	high	82
2	2022-09-11	high	69
4	2022-09-12	high	77
5	2022-09-13	high	69
7	2022-09-14	high	74

```
high_temps.plot("day", "temperature")
```

Now let's look at the lows only.

table name: ***daily_temps_df***

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

Now let's look at the lows only.

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50

```
low_temps = daily_temps_df[daily_temps_df['time_of_measurement'] == "low"]
```

Now let's look at the lows only.

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

Notice: only 4 rows!

	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50

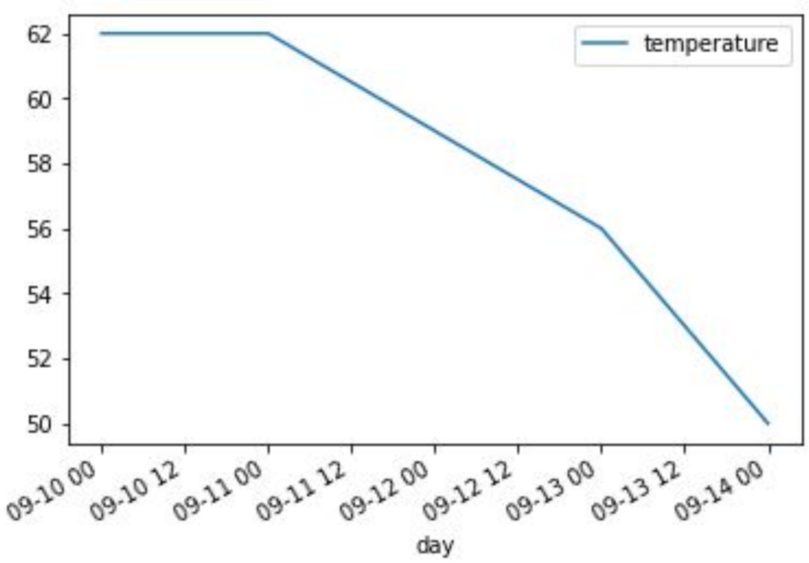
```
low_temps = daily_temps_df[daily_temps_df['time_of_measurement'] == "low"]
```

What happens when we plot the lows?

	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50

What happens when we plot the lows?

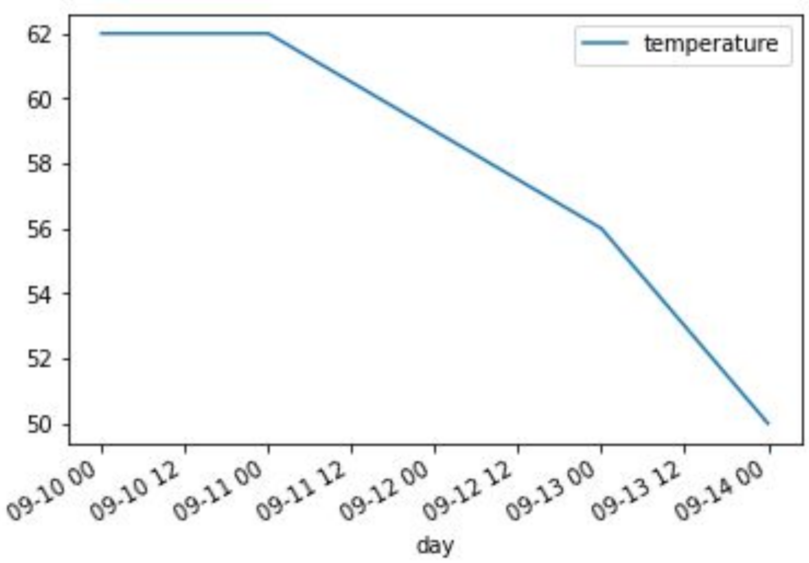
Is this expected behavior?



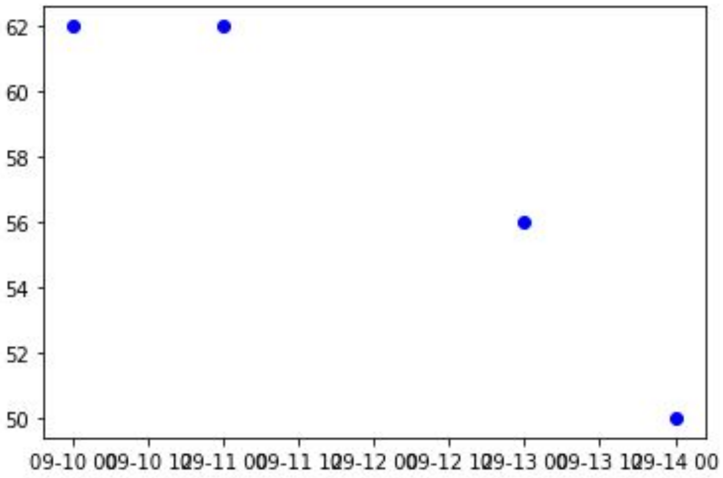
	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50

```
low_temps.plot("day", "temperature")
```

Missing data can be easy to miss!

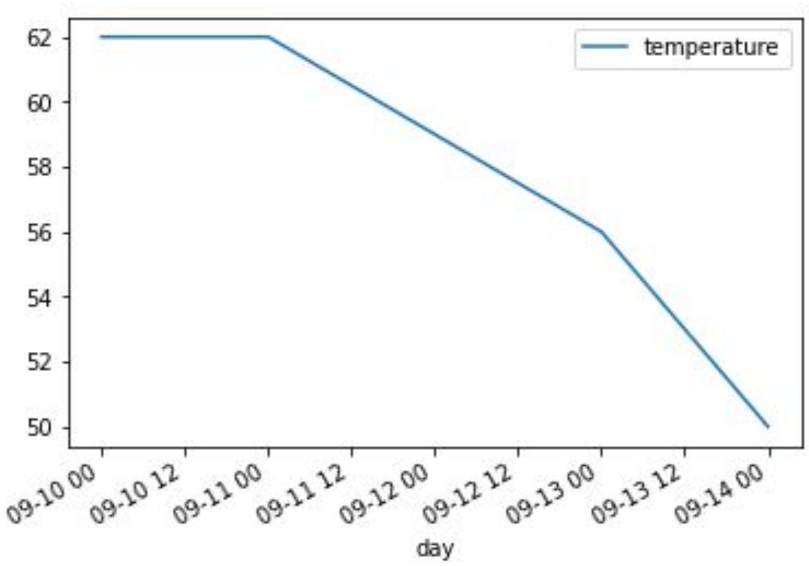



```
low_temps.plot("day", "temperature")
```

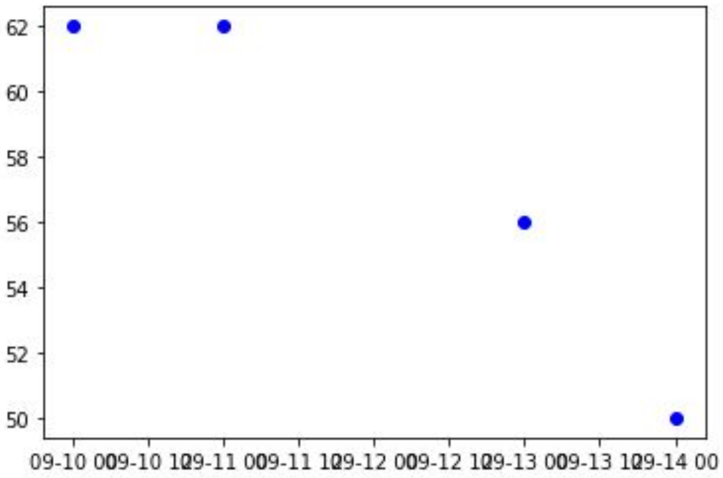



```
plot(low_temps["day"],
low_temps["temperature"], "bo")
```

Missing data can be easy to miss!




```
low_temps.plot("day", "temperature")
```




```
plot(low_temps["day"],
low_temps["temperature"], "bo")
```

blue; circle markers

How do we make *daily_temps_df* meaningful?

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

- aggregated so that each "time step":
  - Is regularly spaced (e.g. daily, monthly, quarterly data) chronologically
  - Has corresponding data per time step
  - Is unique
  - **Deals with missing values**

Back to the original data.
What if we include missingness using NaNs?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

Back to the original data.
What if we include missingness using NaNs?

table name: *daily_temps_df*

	day	time_of_measurement	temperature
0	2022-09-10	high	82
1	2022-09-10	low	62
2	2022-09-11	high	69
3	2022-09-11	low	62
4	2022-09-12	high	77
5	2022-09-13	high	69
6	2022-09-13	low	56
7	2022-09-14	high	74
8	2022-09-14	low	50

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

Could use something like pd.concat to manually add in missing rows of data

conCATenated

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

How to include missingness? (SQL’s Version)

```
SELECT a.day, a.time_of_measurement, b.temperature
FROM
(SELECT * FROM (SELECT distinct day FROM
daily_temps_df) CROSS JOIN (SELECT distinct
time_of_measurement FROM daily_temps_df)) as a
LEFT JOIN daily_temps_df as b
ON a.day=b.day AND
a.time_of_measurement=b.time_of_measurement
ORDER BY a.day
```

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

What are pros of cons of including NaN?

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

What are pros of cons of including NaN?

- **Pros:**
  - easier to tell which values are missing
  - consistent number of rows no matter how you slice data

- **Cons:** now you need to deal with NaN

	day	time_of_measurement	temperature
0	2022-09-10	high	82.0
1	2022-09-10	low	62.0
2	2022-09-11	high	69.0
3	2022-09-11	low	62.0
4	2022-09-12	high	77.0
5	2022-09-12	low	NaN
6	2022-09-13	high	69.0
7	2022-09-13	low	56.0
8	2022-09-14	high	74.0
9	2022-09-14	low	50.0

What happens if we plot time series with NaN?

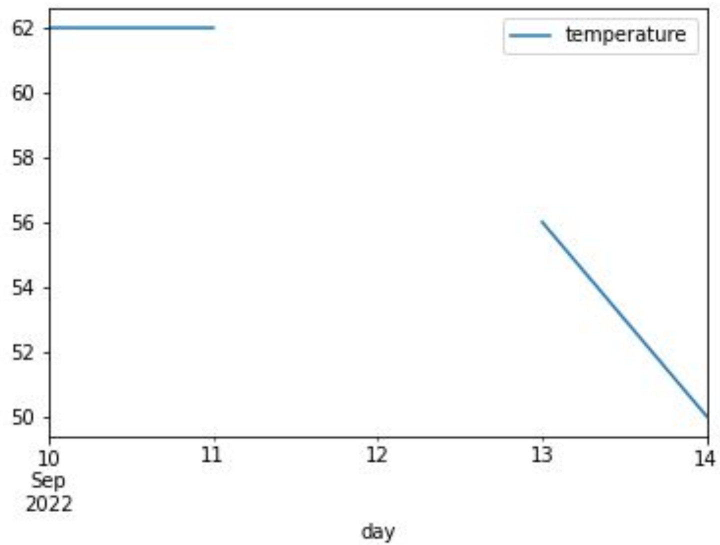
(Lows only)

	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50

What happens if we plot time series with NaN?

(Lows only)

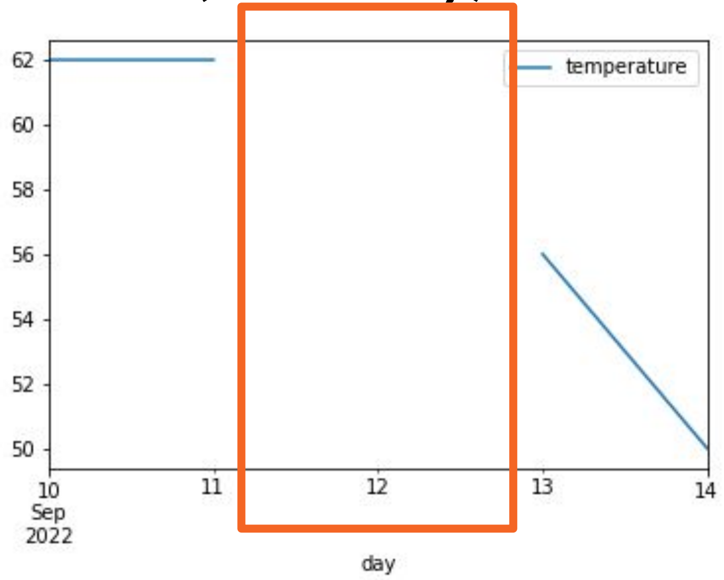
	day	time_of_measurement	temperature
1	2022-09-10	low	62
3	2022-09-11	low	62
6	2022-09-13	low	56
8	2022-09-14	low	50



What happens if we plot time series with NaN?

If you ever see a missing chunk in your plots, **CHECK FOR NaNS**

(Lows only)



Takeaways on Time Series:

1. Make sure your dates are in **datetime** type

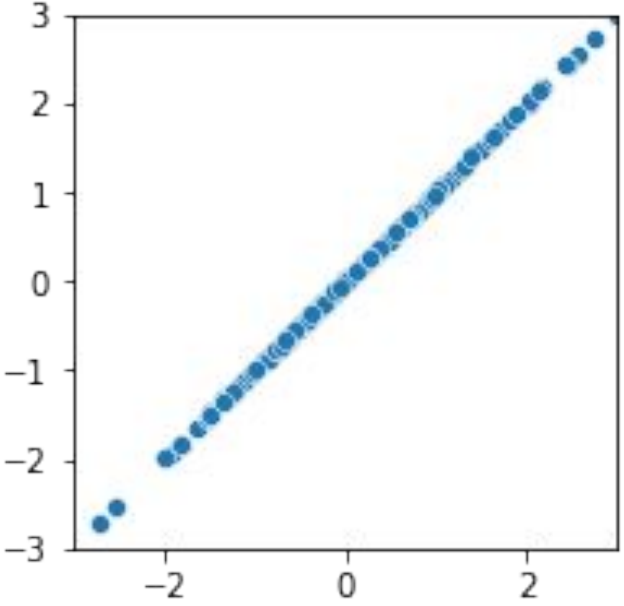
2. Make sure your time series data are **meaningful**: check for inconsistencies

3. Have a plan for how to deal with **missing data**: **explain why they're not in the data**, substitute with another value, impute a value, etc. *(more later in course)*

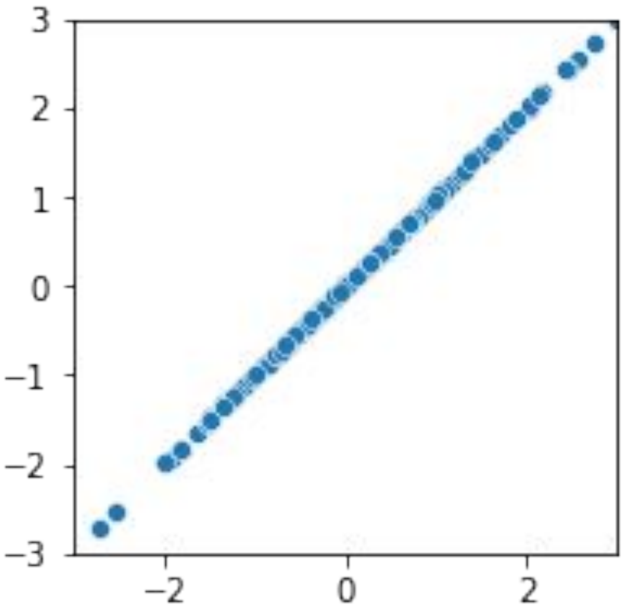
1 min break

https://ragvenderrawat.medium.com/miss-forest-imputaion-the-best-way-to-handle-missing-data-feature-engineering-techniques-2e6922e5cecb

Refresher: what is the covariance here?



Refresher: what is the covariance here?



1. (The axes match here — both X and Y go from -3 to 3, unlike Lec 5 slide 68)

Last time: how to measure the extent that X, Y move together?

Covariance (measures direction of X, Y relationship)	**Correlation** (also measures strength of X, Y relationship)
$\Sigma_i (X_i - \bar{X})(Y_i - \bar{Y}) / N$	$Cov(X,Y)/(\sigma_x \sigma_y)$

Last time: how to measure the extent that X, Y move together?

Covariance (measures direction of X, Y relationship)	**Correlation** (also measures strength of X, Y relationship)
$\Sigma_i (X_i - \bar{X})(Y_i - \bar{Y}) / N$	$Cov(X,Y)/(\sigma_x \sigma_y)$
Doesn't normalize X, Y	Normalizes to deal with different X, Y scales
Between -∞ and ∞	Between -1 and 1
Has interpretable units (X*Y)	Is unitless

When cov or corr aren't enough

- Recall that Cov and Corr are **symmetric**
  - Cov(X,Y) = Cov(Y,X)
  - Corr(X,Y) = Corr(Y,X)

When cov or corr aren't enough

- Recall that Cov and Corr are symmetric
  - Cov(X,Y) = Cov(Y,X)
  - Corr(X,Y) = Corr(Y,X)
- **But what if you want to measure how one variable (X) *affects* another variable (Y)?**

When cov or corr aren't enough

- Cov and Corr each summarizes the relationship between X and Y into a **single number**

When cov or corr aren't enough

- Cov and Corr each summarizes the relationship between X and Y into a **single number**
- **But what if you want *more information* about a X→Y relationship than just a single summary statistic?**

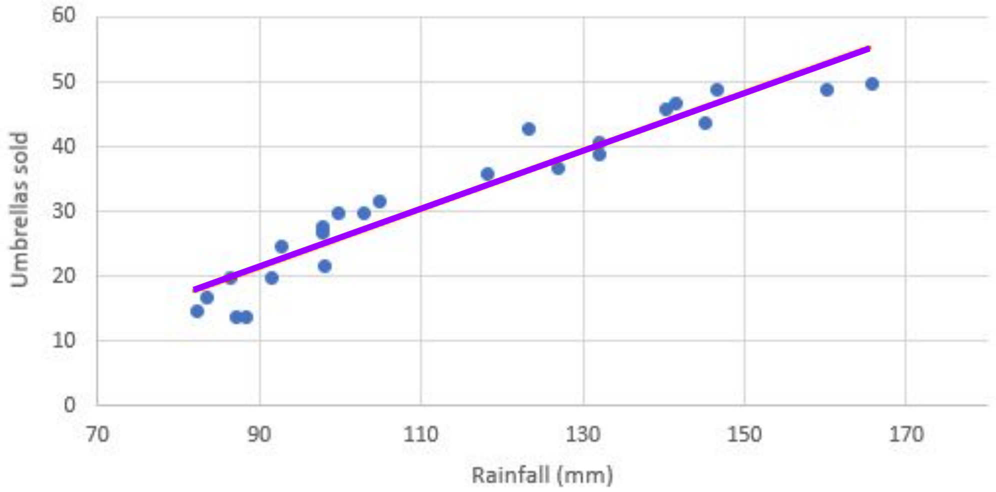
Regression

- **Why do we use regressions?**
- **If I give you a regression line, what can we do with it?**

Regression

- **Why do we use regressions?**
- **If I give you a regression line, what can we do with it?**
- This week: regression on one variable
- Next week: regression on multiple variables

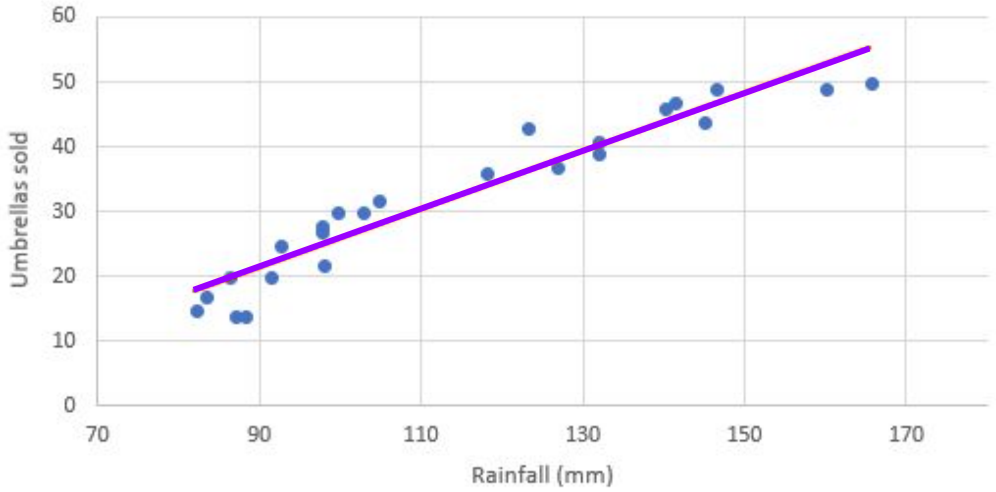
Regression: 3 motivating reasons



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

1. Ability to make predictions

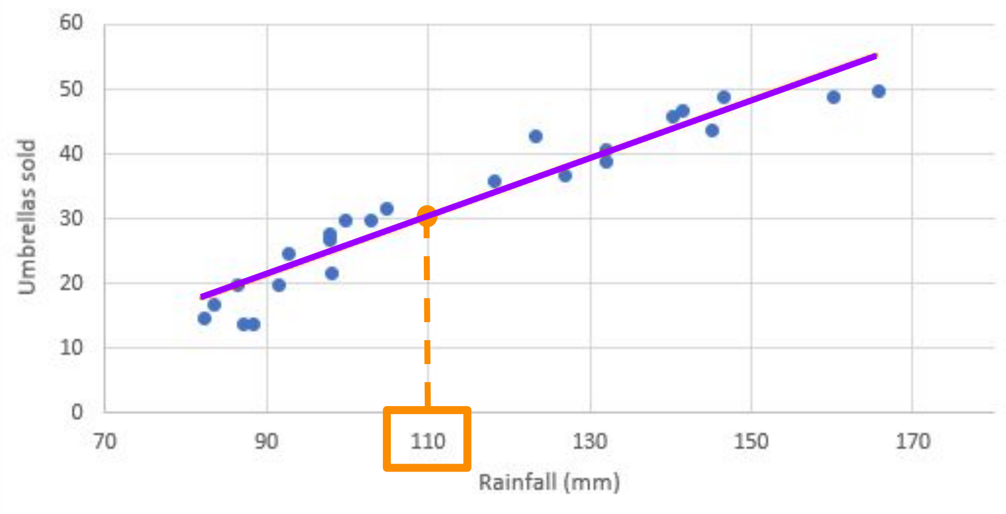


https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

1. Ability to make predictions

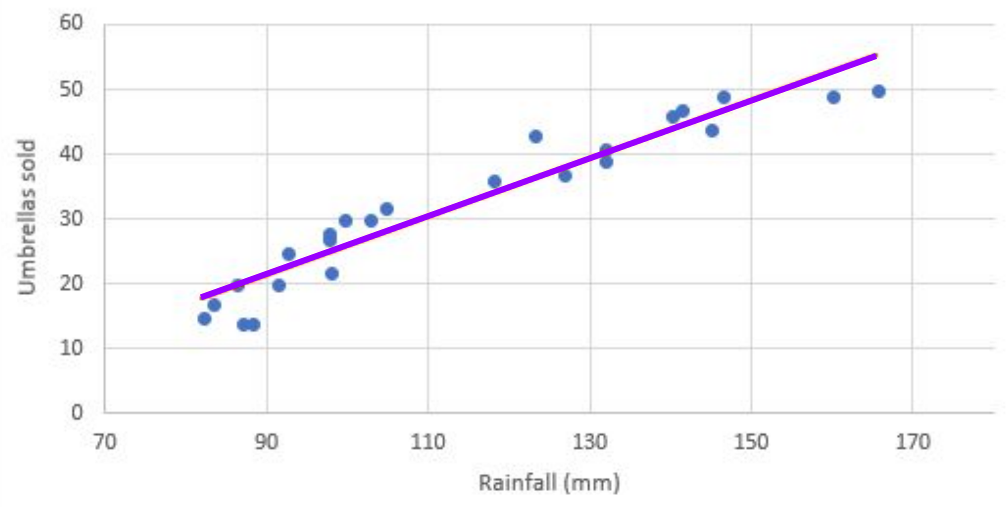
Today there was 110mm of rainfall, so I expect to sell 30 umbrellas



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

2. Summarize relationship between variables

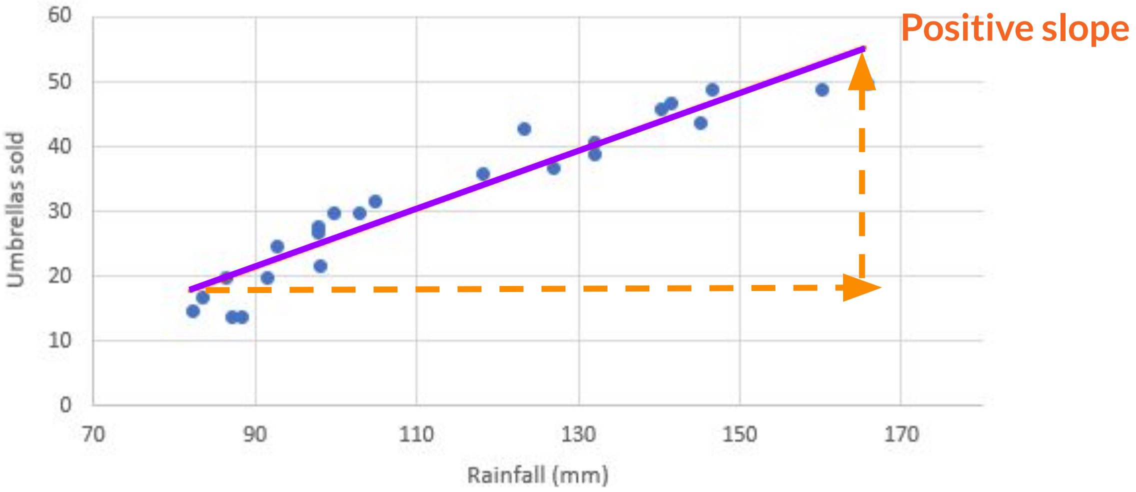


https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

2. Summarize relationship between variables

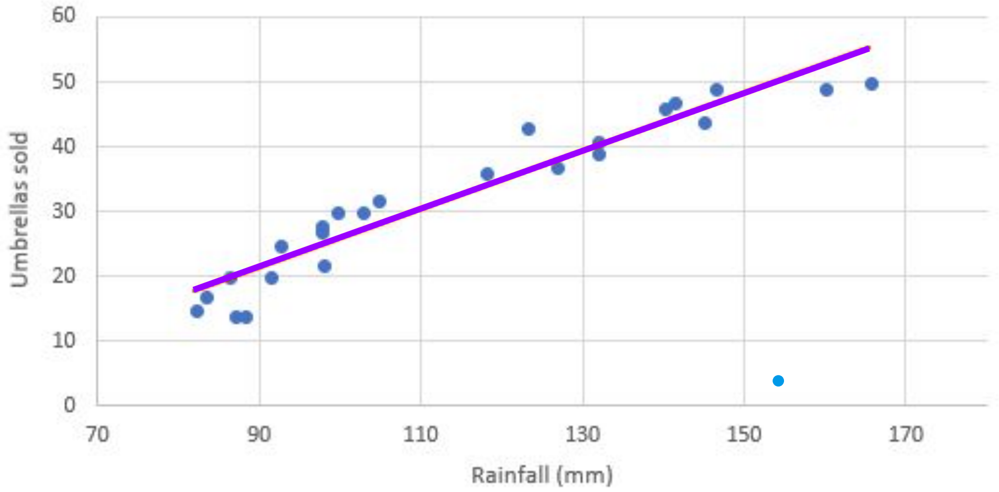
When there is more rain, sales of umbrellas increase. Each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

3. Inspect outliers and other oddities

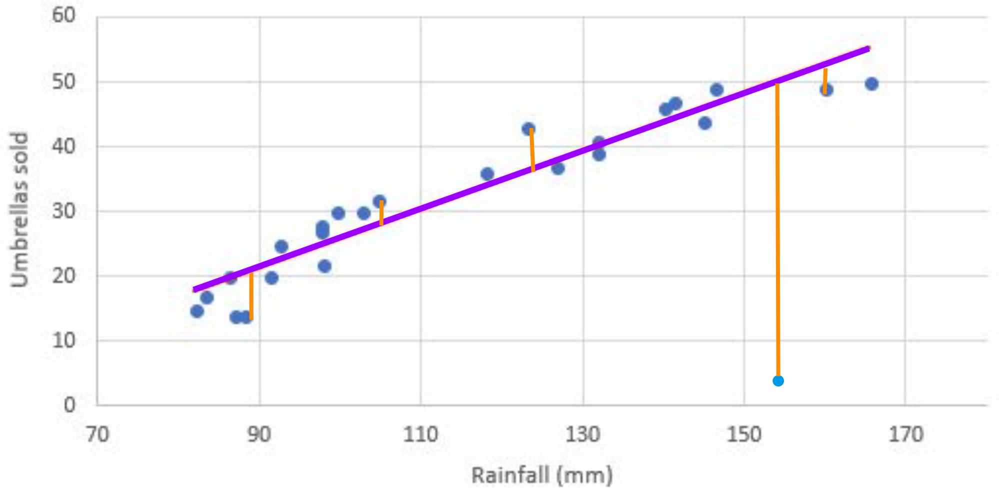


https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: 3 motivating reasons

3. Inspect outliers and other oddities

We only had one day with 155mm rain, but that day everyone was indoors for an all-day conference so we barely sold any umbrellas.



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression motivations

1. Make predictions
2. Summarize relationship between variables
3. Inspect outliers and other oddities

But how?

Regression motivations

Input

Output

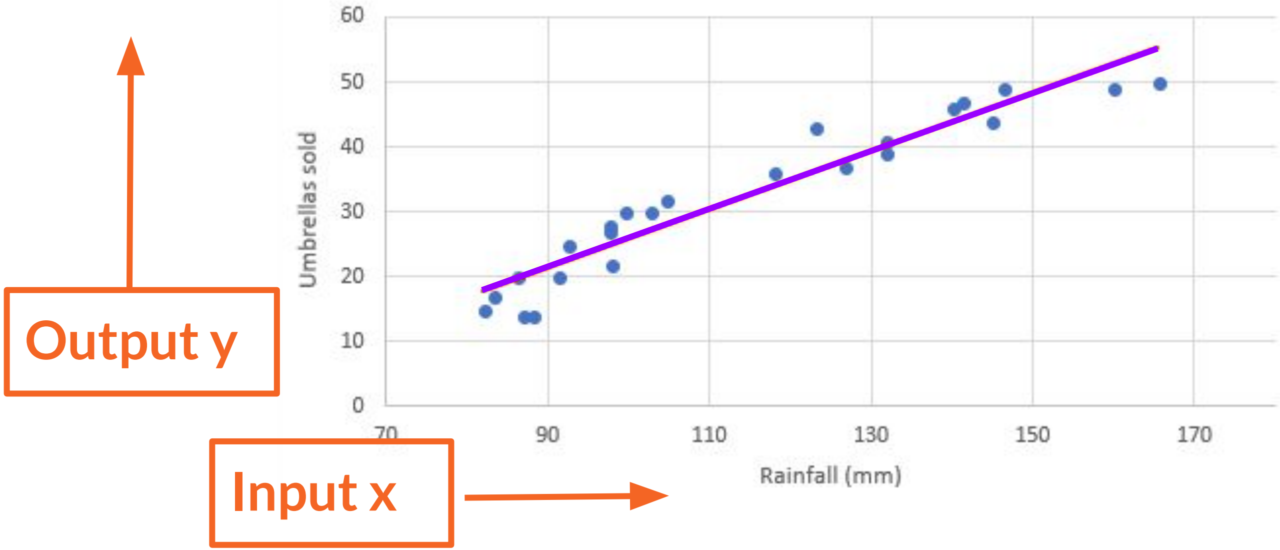
1. Make predictions
2. Summarize relationship between variables
3. Inspect outliers and other oddities

But how?

Regressions: what are they?

- **A regression is given an input (x) and predicts an output (y)**
- **[input, output]** are also known as…
  - [independent, dependent]
  - [endogenous, exogenous]
  - [regressor, regressand]

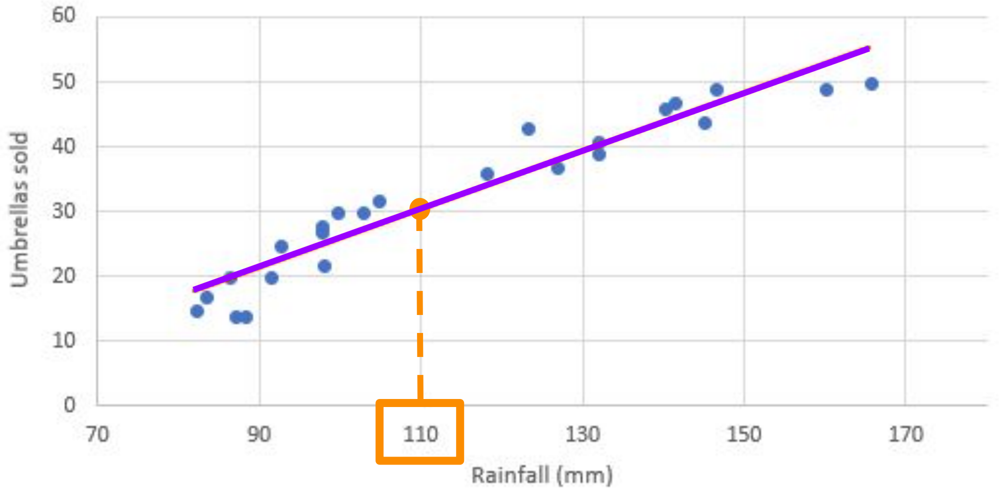
Regression: inputs & outputs



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: inputs & outputs

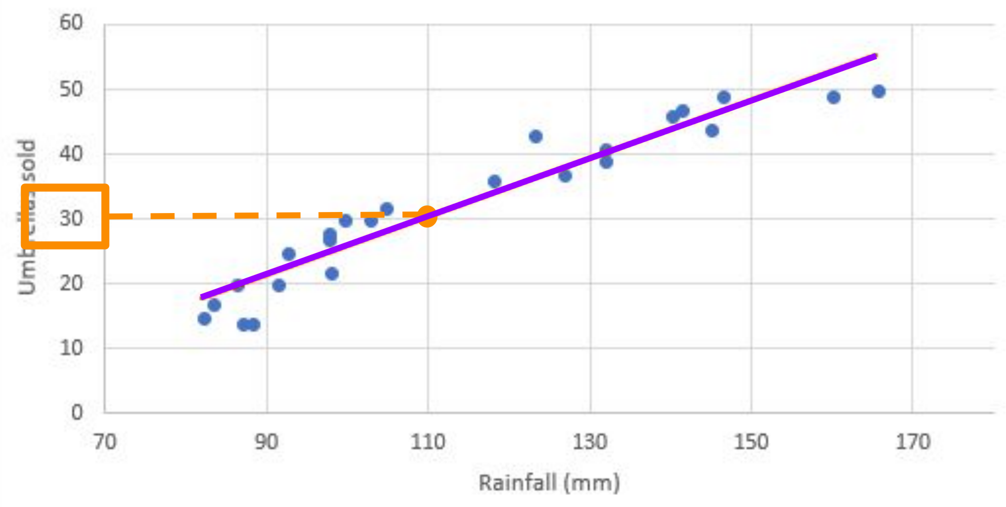
For the *input* 110 mm of rain we expect the *output* 30 umbrellas sold



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: inputs & outputs

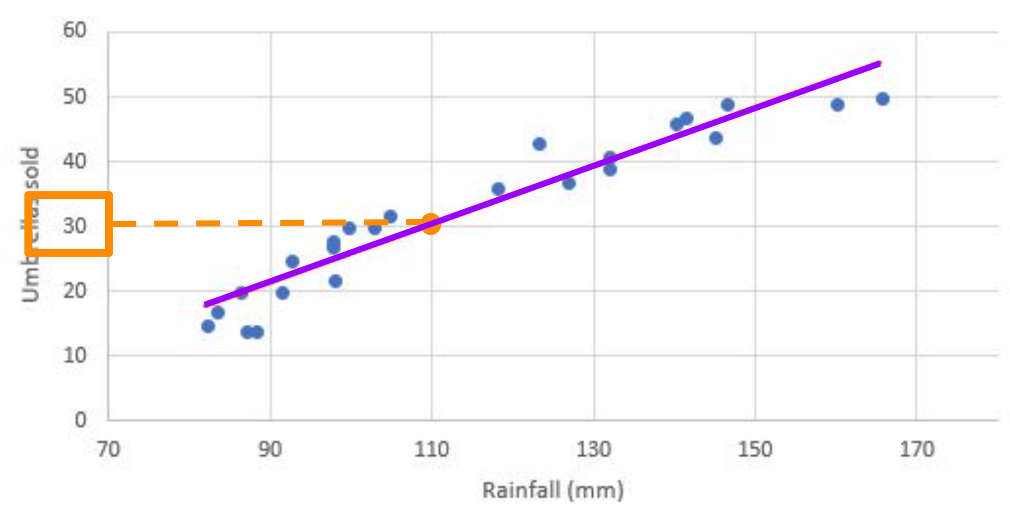
We *could* predict rainfall given umbrella sales, by swapping input/output.



https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression: inputs & outputs

We *could* predict rainfall given umbrella sales, by swapping input/output.



But, you should be upfront with what your main data question is. Are you studying the effect of umbrella sales on rainfall, or the effect of rainfall on umbrella sales?

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Components of a regression

English:	output = intercept + slope * input
Math:	$y = \alpha + \beta \cdot x$
Math shorthand:	$y \sim x$

Components of a regression

English:
- output = intercept + slope * input

Math:
- $y = \alpha + \beta \cdot x$

Math shorthand:
- $y \sim x$

what would *y* be if *x* = *0*.

(sometimes *x*=*0* is nonsensical)

Components of a regression

English:	•	output = intercept + slope * input
Math:	•	$y = \alpha + \beta \cdot x$
Math shorthand:	•	$y \sim x$

unit change in *x* leads to β unit change in *y*, regardless of where you start with *x*

Is y~x the same as x~y?

- English: output = intercept + slope * input
- Math: $y = \alpha + \beta \cdot x$
- Math shorthand: $y \sim x$

Is y~x the same as x~y?

English: • output = intercept + slope * input

Math: • $y = \alpha + \beta \cdot x$

Math shorthand: • $y \sim x$

One extra mm of rain corresponds to 0.45 more umbrellas being sold.

Does one extra umbrella being sold correspond to 0.45mm more rain? **No!**

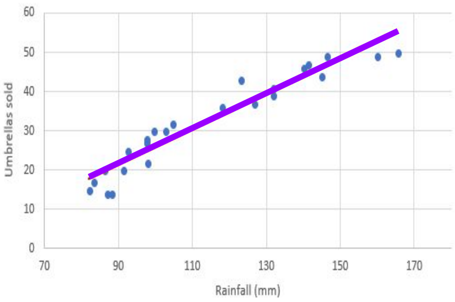
When cov or corr aren't enough

- Recall that Cov and Corr are **symmetric** and distill statistical summary into a **single number**
- Regression is **asymmetric** and distills statistical summary into a **line**

1 min break & attendance

tinyurl.com/2r4wmheh

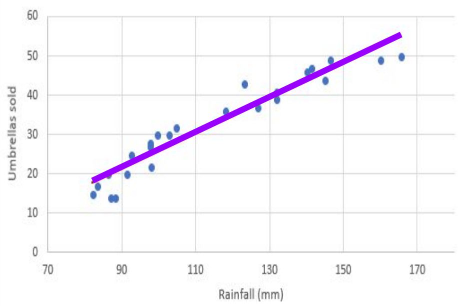
Regression notation



- We have a set of **points** that we want to draw a linear regression line through. That line will have form $\mathbf{y = \alpha + \beta x}$

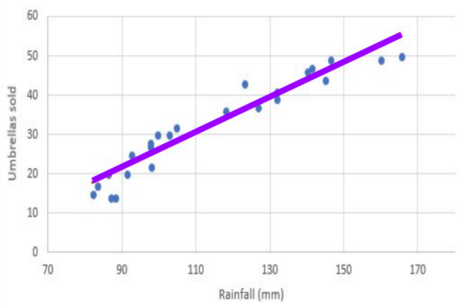
https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



- We have a set of **points** that we want to draw a linear regression line through. That line will have form **y = α + βx**
  - **(Today, let's assume I'm just giving you a regression line, so α and β are known)**

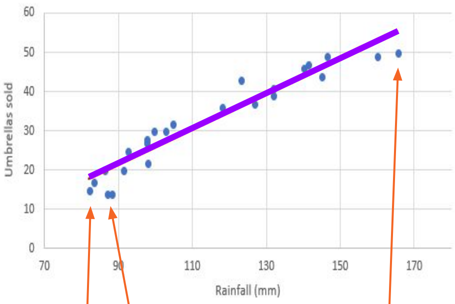
Regression notation



- We have a set of **points** that we want to draw a linear regression line through. That line will have form $y = \alpha + \beta x$
- But, that form doesn't describe the points themselves because they **aren't necessarily on the regression line**

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



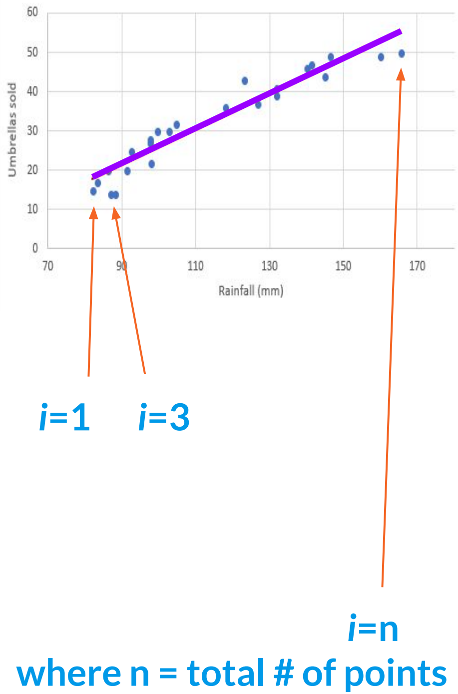
i=1 *i*=3

i=n
where n = total # of points

- We have a set of **points** that we want to draw a linear regression line through. That line will have form $\mathbf{y = \alpha + \beta x}$
- We need to talk about each individual point *i*

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



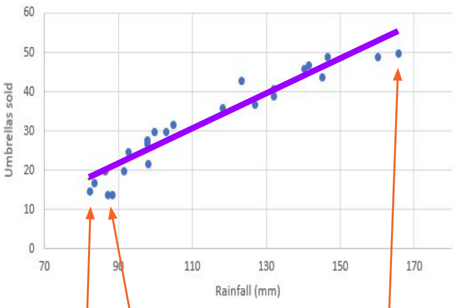
- We have a set of **points** that we want to draw a linear regression line through. That line will have form $y = \alpha + \beta x$
- We need to talk about each individual point *i*

Remember: any time you see math *i*'s, think of them as rows in a dataframe!

i	X	Y
1	78	18
2	83	14
…	…	…

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



i=1 *i*=3

i=n
where n = total # of points

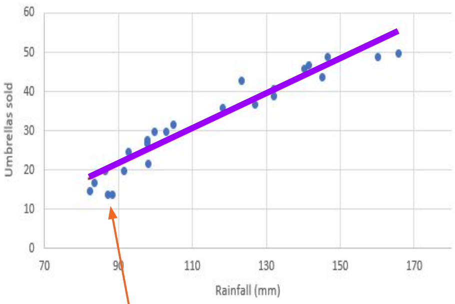
- Regression line: **$y = \alpha + \beta x$**
- Underlying relationship for each point:

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



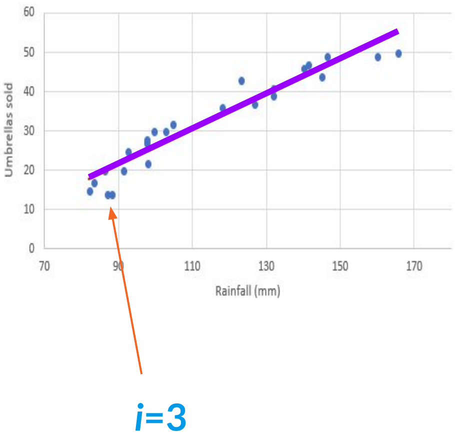
i=3

$$\boxed{y_i} = \alpha + \beta \boxed{x_i} + \varepsilon_i$$

output and input for the *i*th point

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



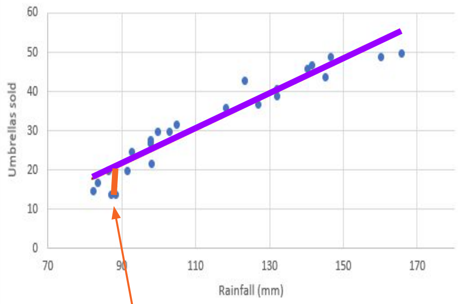
- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

$$y_i = \boxed{\alpha} + \boxed{\beta} x_i + \varepsilon_i$$

intercept and slope
(same for all *n* points)

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



i=3

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

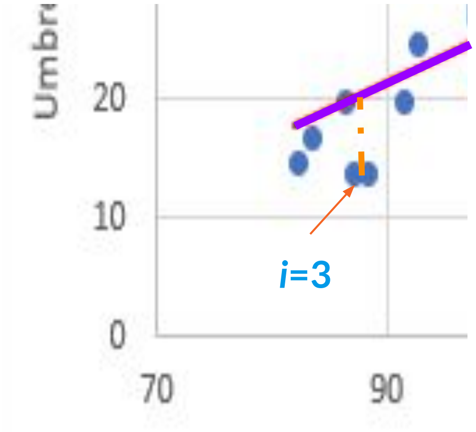
$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

error for the i^{th} point

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: **$y = \alpha + \beta x$**
- Underlying relationship for each point:



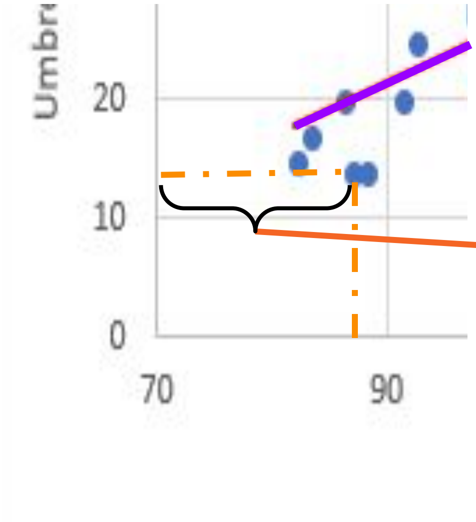
$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

error for the i^{th} point

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



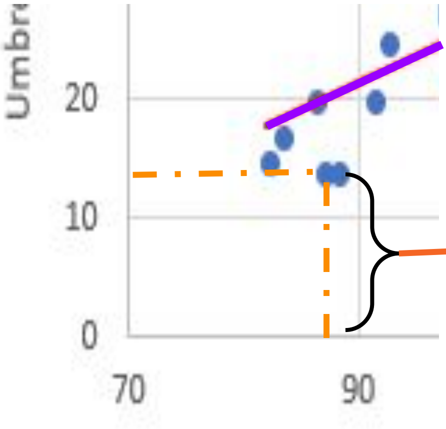
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Input *(x)* of
point *i*=3

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \varepsilon_i$$

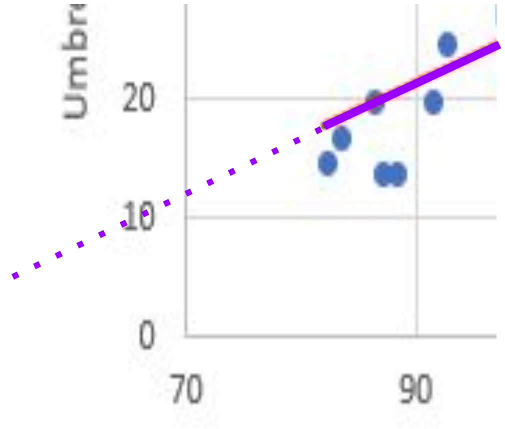
$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Output (*y*) of
point *i*=3

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



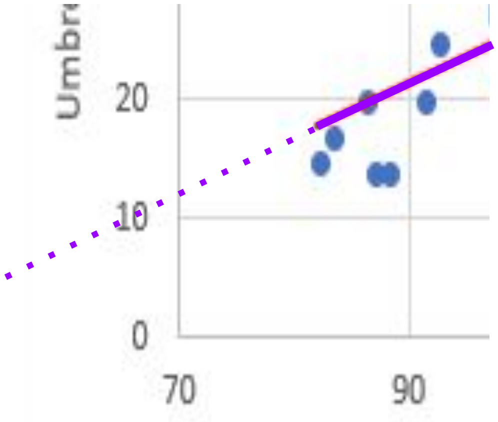
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Intercept (y value if x = 0).
Does -19 look right?

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



← goes off the slide
(notice the axes!)

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

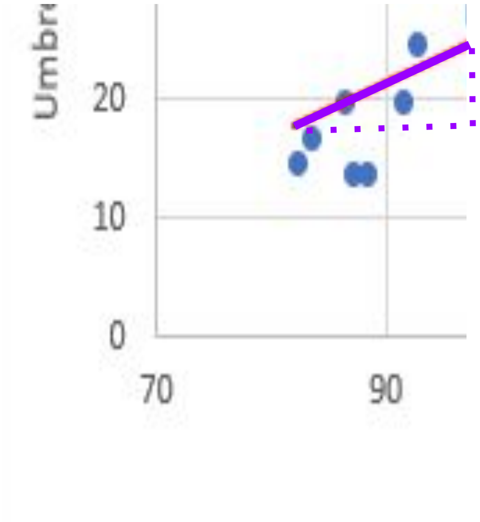
$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Intercept (y value if x = 0).

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \varepsilon_i$$

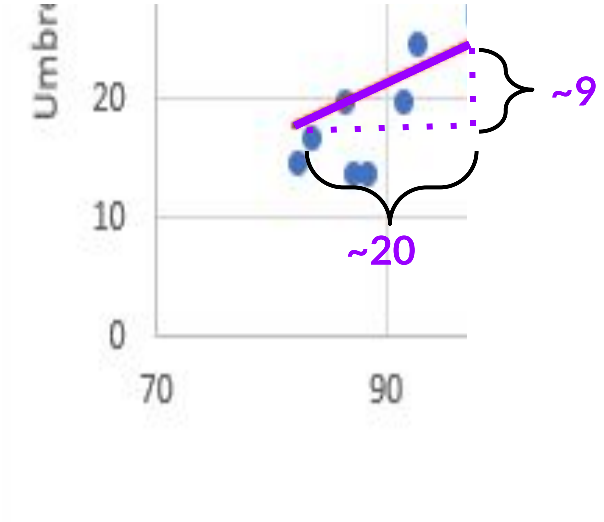
$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Slope (rise over run):
Does positive 0.45 seem right?

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



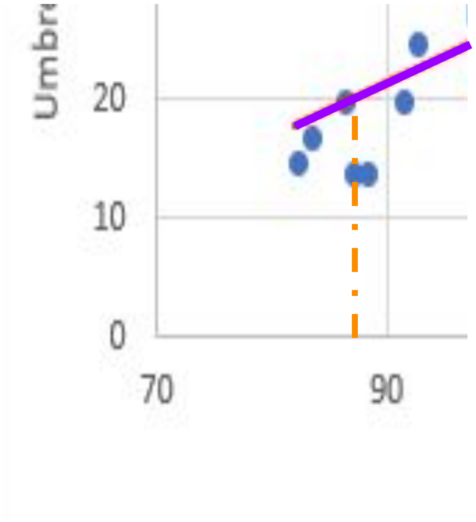
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$14 = -19 + 0.45 * 87 + \varepsilon_3$$

Slope (rise over run):
9/20 = 0.45

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \varepsilon_i$$

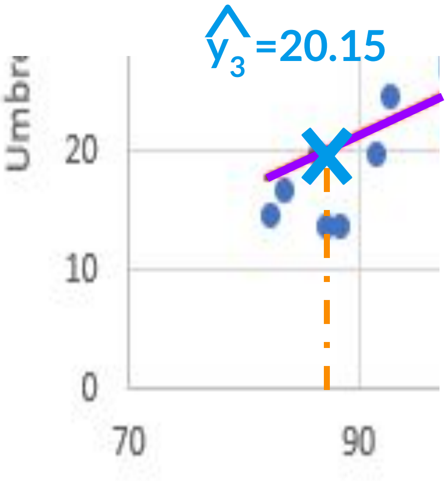
$$14 = \underbrace{-19 + 0.45 * 87}_{} + \varepsilon_3$$

What if we look along the purple regression line for expected output, given input 87?

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



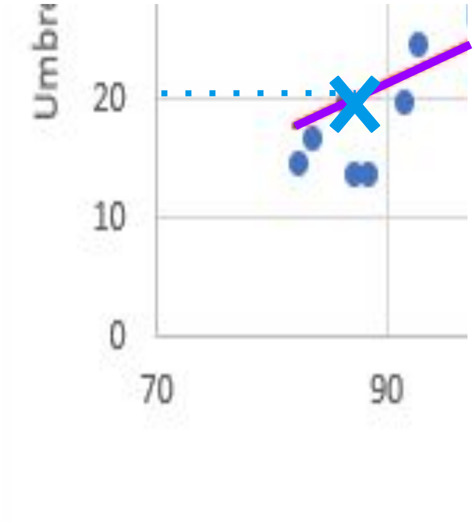
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$14 = \underbrace{-19 + 0.45 * 87}_{\text{Our prediction for how many umbrella sales there should be if rainfall is } x_3 \text{ millimeters}} + \varepsilon_3$$

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$14 = \underbrace{-19 + 0.45 * 87}_{\alpha + \beta x_3 = \hat{y}_3 = 20.15} + \varepsilon_3$$

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

I predict that I will marry JLo in 2022

https://www.si.com/extra-mustard/2014/10/03/ben-affleck-new-york-yankees-hat-gone-girl

Ben Affleck is wearing a HAT. Predictions also wear a HAT

$\hat{y}$

https://www.si.com/extra-mustard/2014/10/03/ben-affleck-new-york-yankees-hat-gone-girl

https://www.vogue.com/article/all-the-bennifer-theories-addressed

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

i	x	y
1	78	18
2	83	14
…	…	…

Regression notation

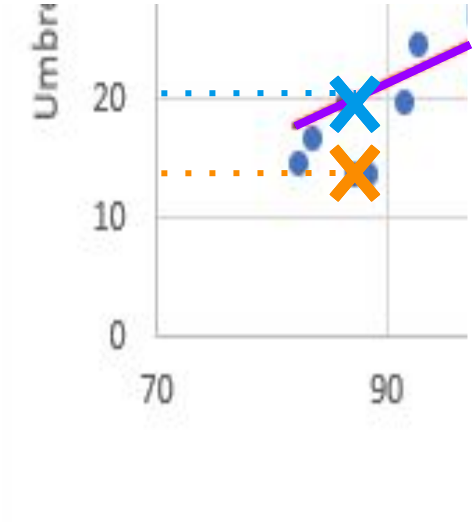
- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

$$y_i = \boxed{\alpha + \beta x_i} + \varepsilon_i$$

i	x	y	ŷ
1	78	18	-19+0.45*78
2	83	14	-19+0.45*83
…	…	…	…

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



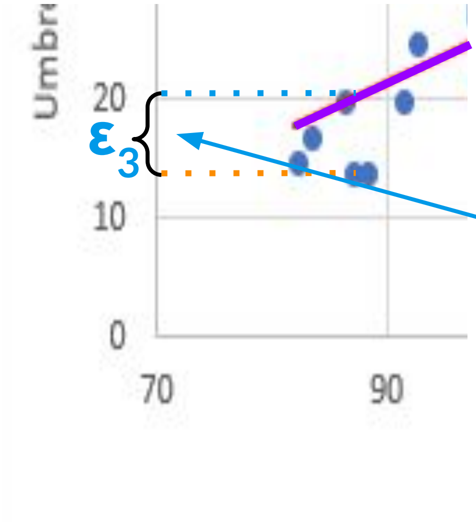
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$\underbrace{14}_{y_3 = 14} = \underbrace{-19 + 0.45 * 87}_{\hat{y}_3 = 20.15} + \varepsilon_3$$

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

$$14 = -19 + 0.45*87 + \underbrace{\varepsilon_3}_{\boxed{y_3 - \hat{y}_3 = -6.15}}$$

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

$$y_i = \boxed{\alpha + \beta x_i} + \varepsilon_i$$

i	x	y	ŷ
1	78	18	-19+0.45*78 = 16.10
2	83	14	-19+0.45*83 = 18.35
…	…	…	…

Regression notation

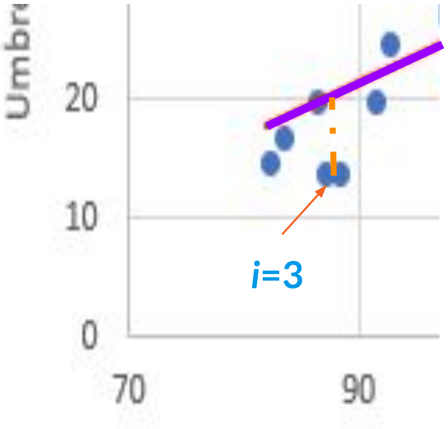
- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

i	x	y	ŷ	ε
1	78	18	-19+0.45*78 = 16.10	18 - 16.10
2	83	14	-19+0.45*83 = 18.35	14 - 18.35
…	…	…	…	…

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

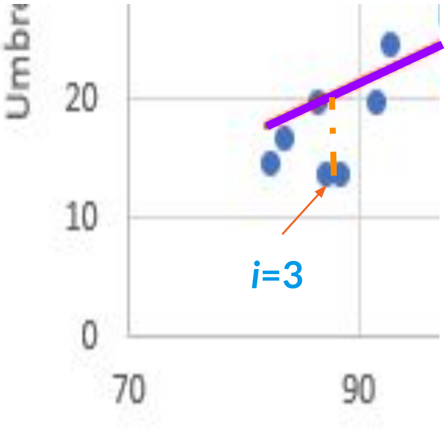
$$14 = -19 + 0.45 * 87 + \underbrace{\varepsilon_3}$$

Error: how wrong was our prediction?
Is -6.15 a lot or a little?

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



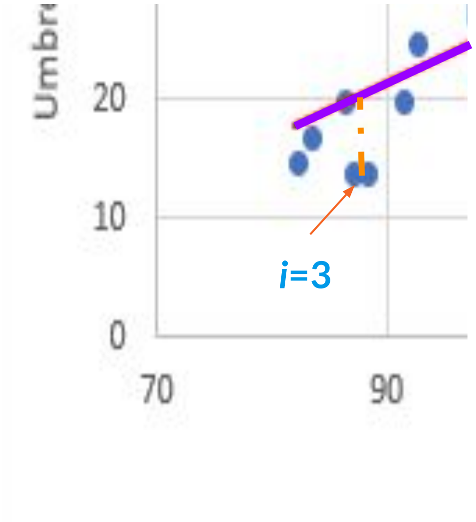
$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

$$14 = -19 + 0.45*87 + \underbrace{\varepsilon_3}_{}$$

Error: how wrong was our prediction? Is -6.15 a lot or a little? Error close to 0 is ideal

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation



- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:

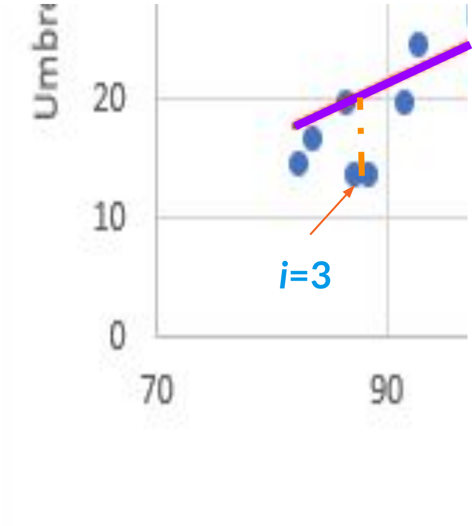
$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

KEY INSIGHT: we want to minimize abs(error) across all of our points *i*

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression notation

- Regression line: $y = \alpha + \beta x$
- Underlying relationship for each point:



$$y_i = \alpha + \beta x_i + \boxed{\varepsilon_i}$$

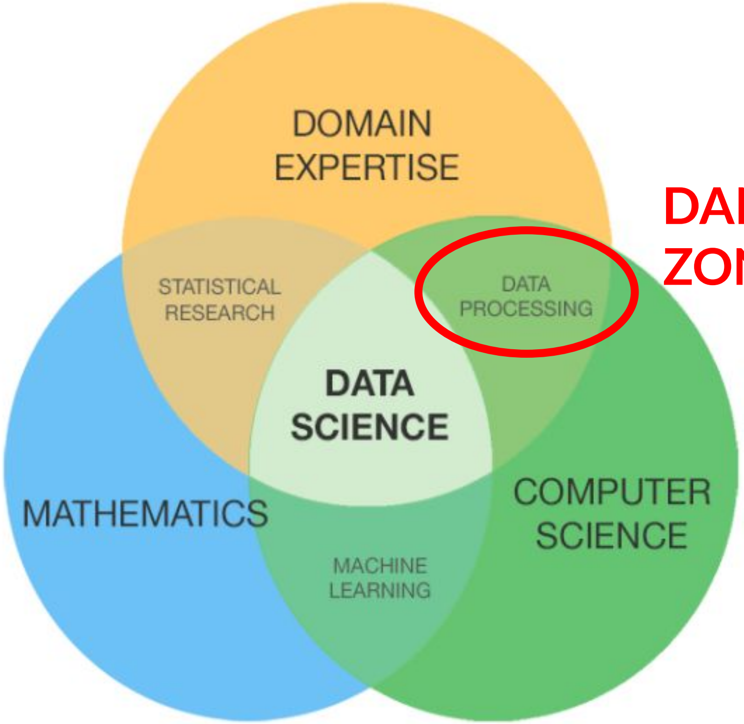
Next class: how we can use *minimizing error* to find the regression line (and get α and β)

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression motivations → interpretations

1. Summarize relationship between variables
2. Make predictions
3. Inspect outliers and other oddities

Why interpret?



DANGER
ZONE

Content source: http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram

Interpret regressions: summarize relationship

- Model: $y = \alpha + \beta x$

Interpret regressions: summarize relationship

- Model: $y = \alpha + \beta x$
- 1 unit increase in x corresponds to a β unit increase/decrease in y

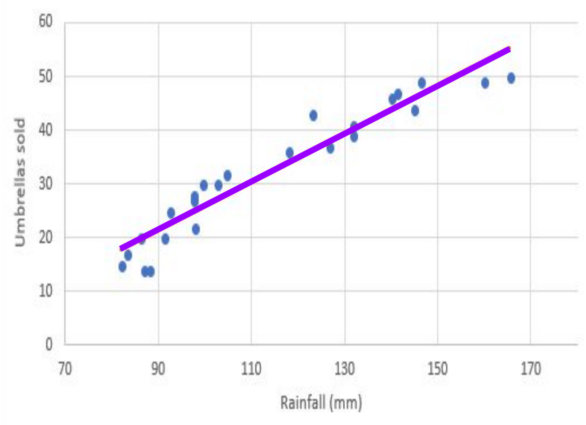
Interpret regressions: summarize relationship

- Model: $y = \alpha + \beta x$
- **1 unit increase in x corresponds to a β unit increase/decrease in y**
  - Why do we say "**corresponds to**"? Because we don't know if it's *causal*

Interpret regressions: summarize relationship

- Model: $y = \alpha + \beta x$
- **1 unit increase in x corresponds to a β unit increase/decrease in y**
    - Why do we say "**corresponds to**"? Because we don't know if it's *causal*
    - Why do we say "**increase/decrease**"? Because it depends on the sign of β

Regression motivations → interpretations



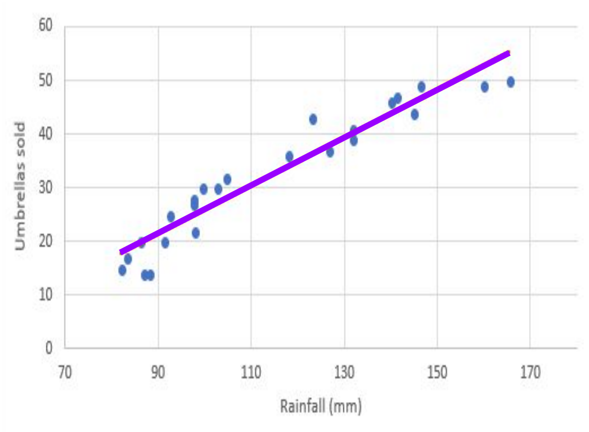
1. Summarize relationship between variables:

2. Make predictions:

3. Inspect oddities / outliers:

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression motivations → interpretations



1. **Summarize relationship between variables:**
   Our model shows a positive relationship between rain and sales of umbrellas; specifically, each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.

2. **Make predictions:**

3. **Inspect oddities / outliers:**

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Common task in data science job

- You are a data scientist at a dairy bar
- You need to write an email to your boss that interprets a regression that you recently ran
  - x: **Days (2023-08-12** to **2023-09-12)**
  - y: **Ice Cream Sales (**# units sold**)**
  - Regression model: **y = 100 - 3x**

Regression interpretations: summarize relationship

x = millimeters of rainfall

y = umbrellas sold

$y = -19 + 0.45x$

Summarize relationship between variables:

Our model shows a positive relationship between rain and sales of umbrellas; specifically, each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

$y = 100 - 3x$

Summarize relationship between variables:

Regression interpretations: summarize relationship

x = millimeters of rainfall

y = umbrellas sold

$y = -19 + 0.45x$

Summarize relationship between variables:

Our model shows a positive relationship between rain and sales of umbrellas; specifically, each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

$y = 100 - 3x$

Summarize relationship between variables:

Our model shows a negative relationship between days and sales of ice cream; specifically, each additional day that goes by (benchmarked where x=0 means Aug 12, 2023) corresponds to 3 fewer ice cream units we expect to be sold.

Interpret regressions: predict

- Model: $y = \alpha + \beta x$

Interpret regressions: predict

- Model: **$y = \alpha + \beta x$**
- "For variable *x* = [*some number*], we expect *variable y* = [*the model's predicted number*]."

Interpret regressions: predict

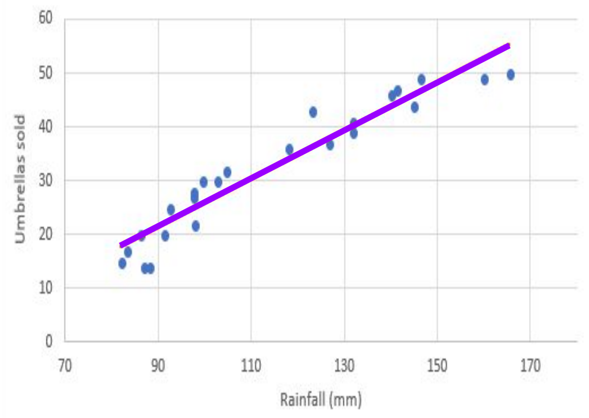
- **Model: $y = \alpha + \beta x$**
- **“For variable *x* = [*some number*], we expect *variable y* = [*the model’s predicted number*].”**
  - Why do we say “**we expect**”? This is a prediction, not a certainty!

Interpret regressions: predict

- Model: **$y = \alpha + \beta x$**
- "For variable *x* = [*some number*], we expect *variable y* = [*the model's predicted number*]."
  - Why do we say "we expect"?
    This is a prediction, not a certainty!

- **For x = 0, we expect y = α**

Regression motivations → interpretations



1. **Summarize relationship between variables:** Our model shows a positive correlation between rain and sales of umbrellas; specifically, each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.
2. **Make predictions:** This model indicates that at the annual Ithaca average of 110mm of rainfall, we should expect to sell 30 umbrellas.
3. **Inspect oddities / outliers:**

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression interpretations: make predictions

x = millimeters of rainfall

y = umbrellas sold

y = -19 + 0.45x

Make predictions:

This model indicates that at the annual Ithaca average of 110mm of rainfall, we should expect to sell 30 umbrellas; *however, if we have no rainfall, we expect to sell -19 umbrellas?!*

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

y = 100 - 3x

Make prediction(s):

Regression interpretations: make predictions

x = millimeters of rainfall

y = umbrellas sold

y = -19 + 0.45x

Make predictions:

This model indicates that at the annual Ithaca average of 110mm of rainfall, we should expect to sell 30 umbrellas; *however, if we have no rainfall, we expect to sell -19 umbrellas?!*

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

y = 100 - 3x

Make predictions:

This model indicates that for one day after Aug 12, 2023, we expect to sell 97 ice cream units.
It indicates that 30 days after Aug 12, 2023, we expect to sell 10 ice cream units.
It indicates that 100 days after Aug 12, 2023, we expect to sell -200 ice cream units?!

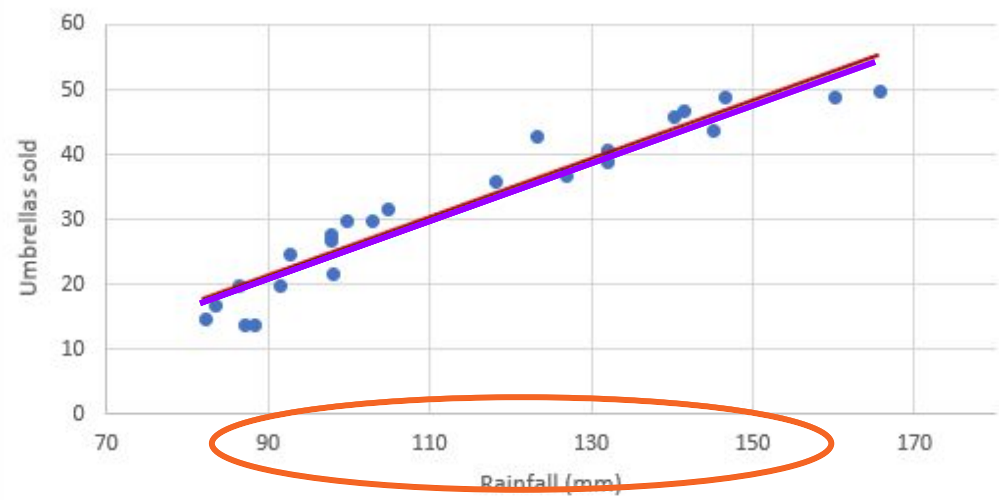
Interpret regressions: oddities

Interpret regressions: oddities

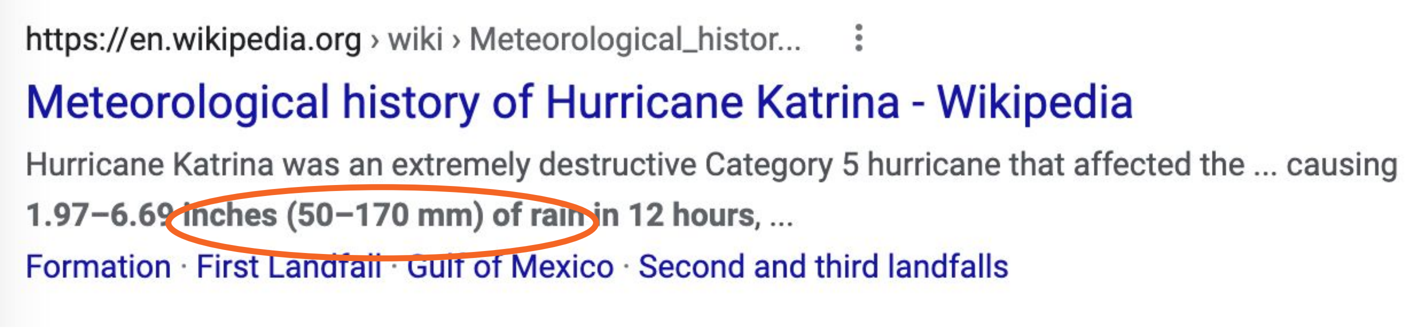
- **When are regression results non-sensical?**
  - We want to avoid extrapolating beyond the “scope of the model”

Interpret regressions: oddities

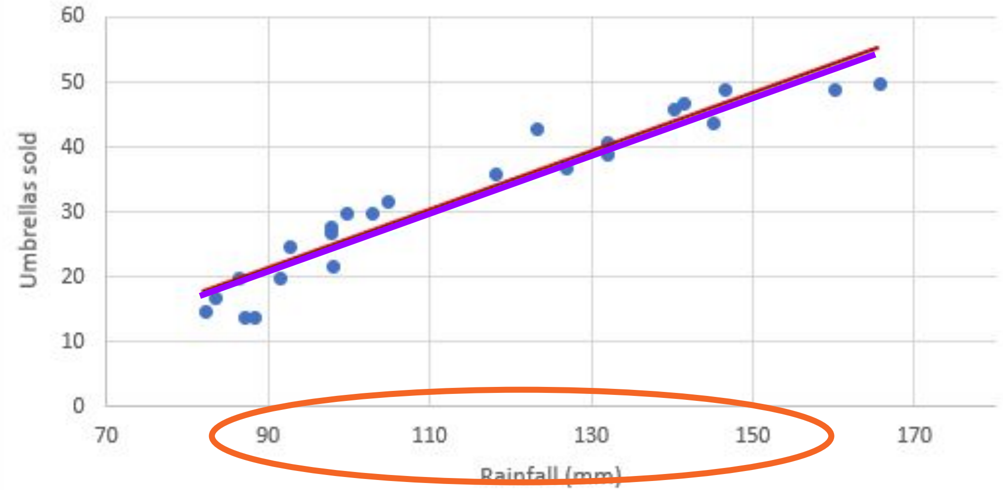
- **When are regression results non-sensical?**
  - We want to avoid extrapolating beyond the “scope of the model”
- **Are there outliers in the data?**
  - Are there explanations for them?
  - Should they be captured by a different model (e.g. quadratic instead of linear)?



60
50
40
30
20
10
0
Umbrellas sold
70
90
110
130
150
170
Rainfall (mm)



https://en.wikipedia.org › wiki › Meteorological_histor...
Meteorological history of Hurricane Katrina - Wikipedia
Hurricane Katrina was an extremely destructive Category 5 hurricane that affected the ... causing **1.97–6.69 inches (50–170 mm) of rain in 12 hours,** ...
Formation · First Landfall · Gulf of Mexico · Second and third landfalls



Use domain knowledge to check if your data make sense!

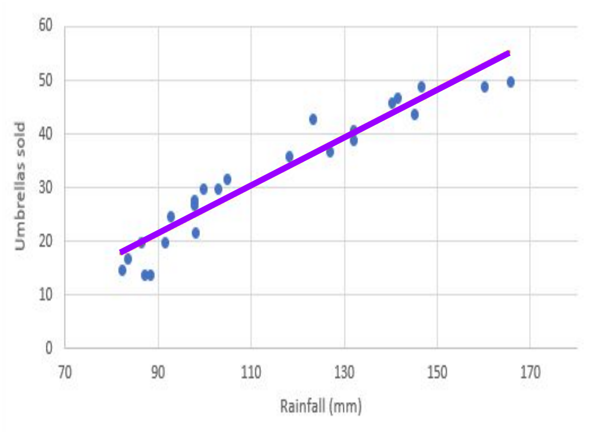
https://en.wikipedia.org › wiki › Meteorological_histor...

Meteorological history of Hurricane Katrina - Wikipedia

Hurricane Katrina was an extremely destructive Category 5 hurricane that affected the ... causing **1.97–6.69 inches (50–170 mm) of rain in 12 hours**, ...

Formation · First Landfall · Gulf of Mexico · Second and third landfalls

Regression motivations → interpretations



1. **Summarize relationship between variables:** Our model shows a positive correlation between rain and sales of umbrellas; specifically, each additional mm of rain corresponds to an extra 0.45 umbrellas we expect to be sold.
2. **Make predictions:** This model indicates that at the annual Ithaca average of 110mm of rainfall, we should expect to sell 30 umbrellas.
3. **Inspect oddities / outliers:** We expect this model to hold between for rainfall amounts between 80-170mm, but cannot extrapolate further.

https://www.ablebits.com/office-addins-blog/linear-regression-analysis-excel/

Regression interpretations: note oddities

x = millimeters of rainfall

y = umbrellas sold

$y = -19 + 0.45x$

Inspect oddities / outliers:

We expect this model to hold between for rainfall amounts between 80-120mm, but cannot extrapolate further.

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

$y = 100 - 3x$

Inspect oddities / outliers:

Regression interpretations: note oddities

x = millimeters of rainfall

y = umbrellas sold

$y = -19 + 0.45x$

Inspect oddities / outliers:

We expect this model to hold between for rainfall amounts between 80-120mm, but cannot extrapolate further.

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

$y = 100 - 3x$

Inspect oddities / outliers:

We expect this model to hold between 2023-08-12 * and 33.3 days after that day, but cannot extrapolate further since otherwise we predict negative sold units.

*Maybe it holds beforehand; need more domain knowledge! Slope is probably positive from early summer to Aug.

Regression interpretations: note oddities

x = millimeters of rainfall

y = umbrellas sold

$y = -19 + 0.45x$

Inspect oddities / outliers:

We expect this model to hold between for rainfall amounts between 80-120mm, but cannot extrapolate further.

x = days (2023-08-12 to 2023-09-12)

y = # ice cream units sold

$y = 100 - 3x$

Note: we can't inspect outliers here since I haven't given you any of the underlying x, y data for each individual data point i