INFO 2950: Intro to Data Science

Lecture 10
2023-09-25

https://twitter.com/overflow_meme/status/1168418079228997632/photo/1

Agenda

1. **Admin**
2. **Logistic Regression Review**
    a. Log odds intuition
    b. Interpretations
3. **Multivariable Regressions**
    a. Python
    b. Dummy variables
    c. Interpretations: linear
    d. Collinear variables

Homework Formatting

- Reminder: HW3 due tomorrow (9/26)
- The absolute latest day we can accept homework is 9/29 so that we can post HW3 solutions
    - You cannot use more than 3 slip days
- Make sure your problems are tagged correctly & PDFs do not cut off code/solutions

Academic integrity

- See homework headers (and Problem 0's) and syllabus for policies
- Working on these cases takes instructor and TA time away from helping you
- This is why we give you slip days!!

Prelim

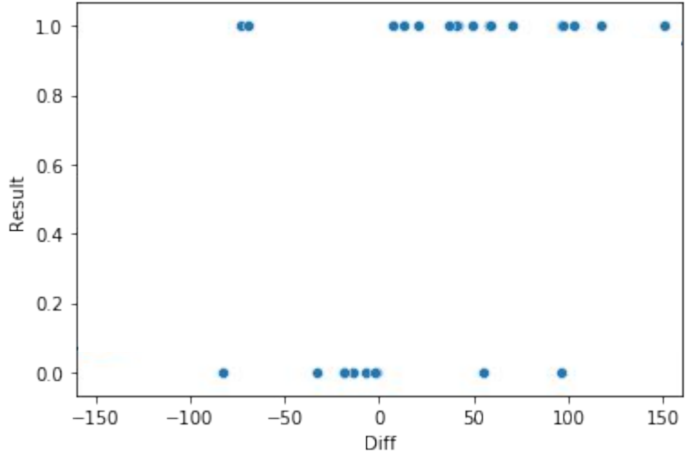
- In-class on Monday, Oct 2nd
- Friday discussion this week is a review session
- Last year’s midterm & review topics on Canvas
- Prelim locations will be posted on Canvas

Prelim locations

- Last name A-K in Ives 305 (this room)
- Last name L-Z in Sage Hall B01
- SDS accommodations: check for an email from the SDS Alternative Testing Program (ATP) with your room number; email me if you do not know where to go

Last time...

The two players involved in the controversy

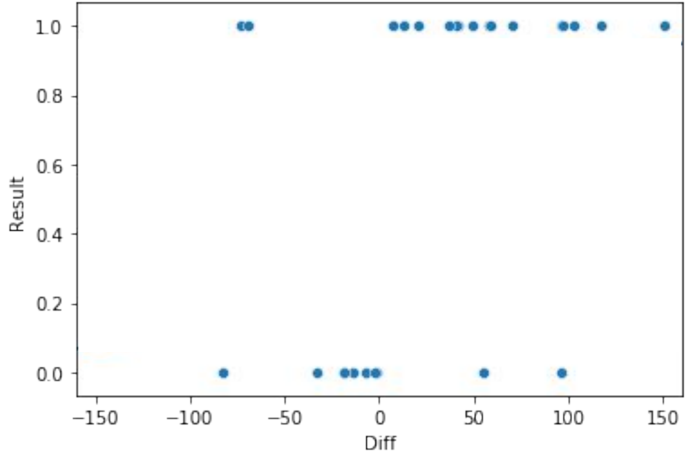


https://en.wikipedia.org/wiki/Carlsen%E2%80%93Niemann_controversy

Should you use a linear regression on binary output data?


Magnus Carlsen Hans Niemann

The two players involved in the controversy

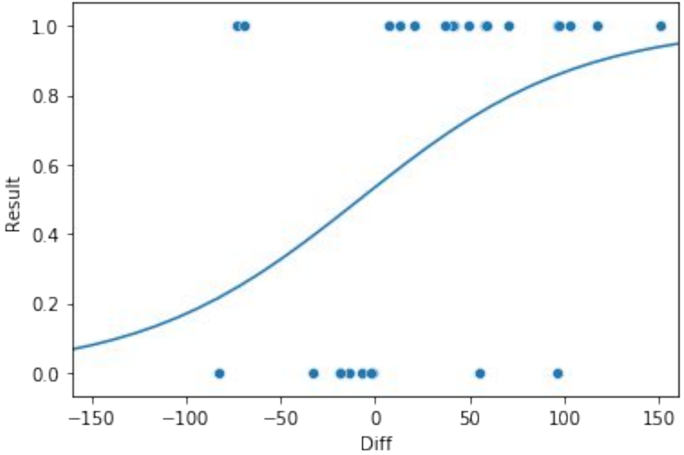


https://en.wikipedia.org/wiki/Carlsen%E2%80%93Niemann_controversy

No! Use logistic regression if your y's are all 0's and 1's


Magnus Carlsen Hans Niemann

The two players involved in the controversy



https://en.wikipedia.org/wiki/Carlsen%E2%80%93Niemann_controversy

Summarizing logistic regressions

Logistic $y \sim \sigma(\alpha + \beta x)$ (y must be binary)	For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}

p / (1-p)

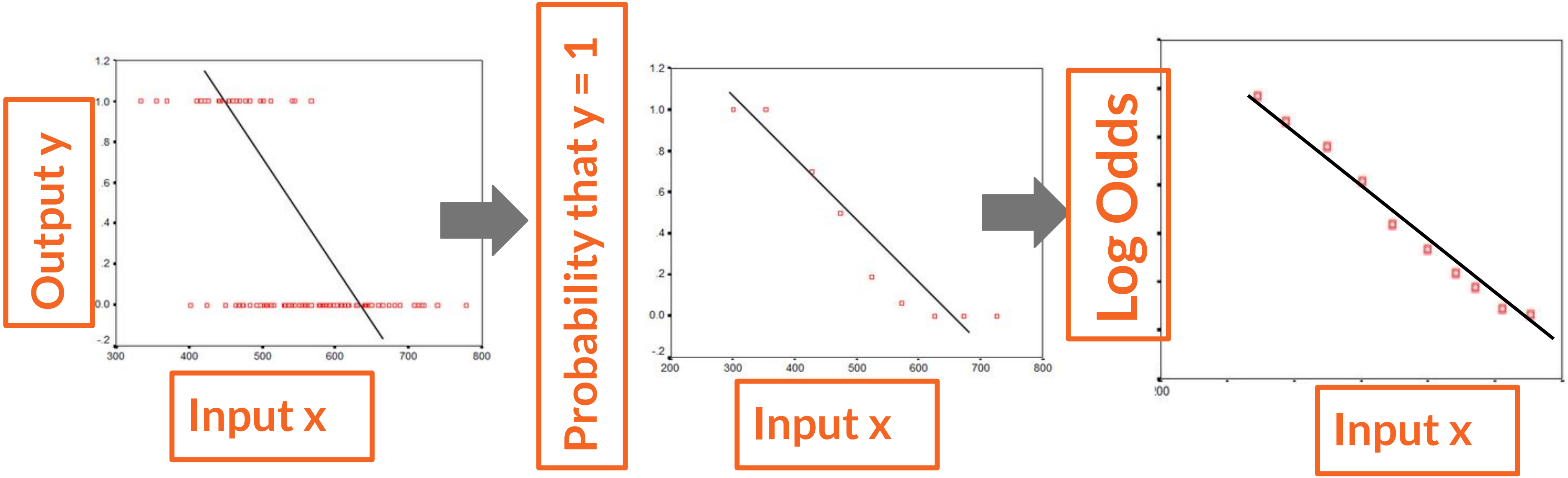
Prob of y = 1 / Prob of y = 0

Pr(Magnus win) / Pr(Magnus lose)

Ways to describe probabilities

Numbers between 0 and 1	p, (1-p)	
Frequencies	10 wins, 2 losses	p = 10 / (10 + 2)
Odds	10:2	hard to use in math
Odds ratios	10 / 2	= p / (1-p)
Log odds ratios	log(10/2) = -log(2/10)	logit function!

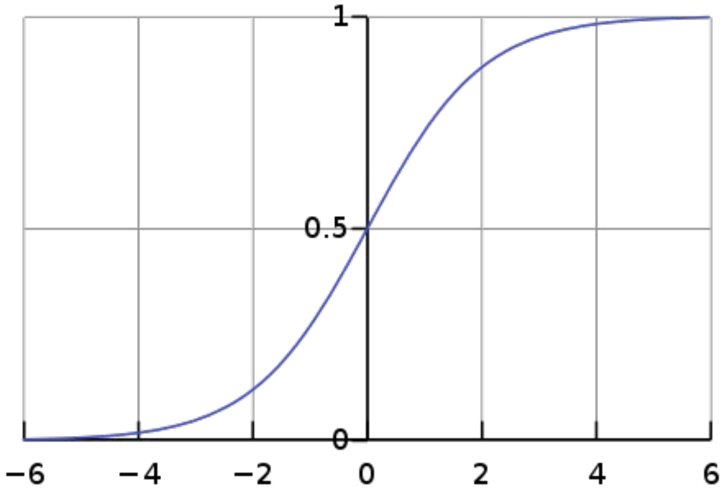
Last time on Logistic Regressions…



https://www.theanalysisfactor.com/what-is-logit-function/

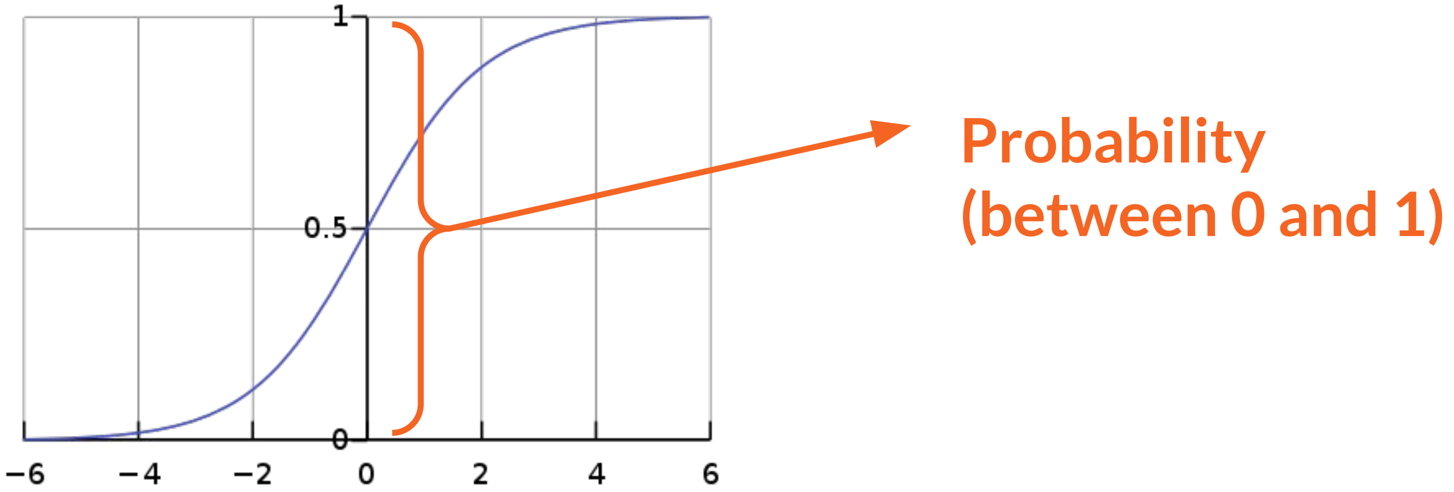
Intuition: log odds ratio

Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



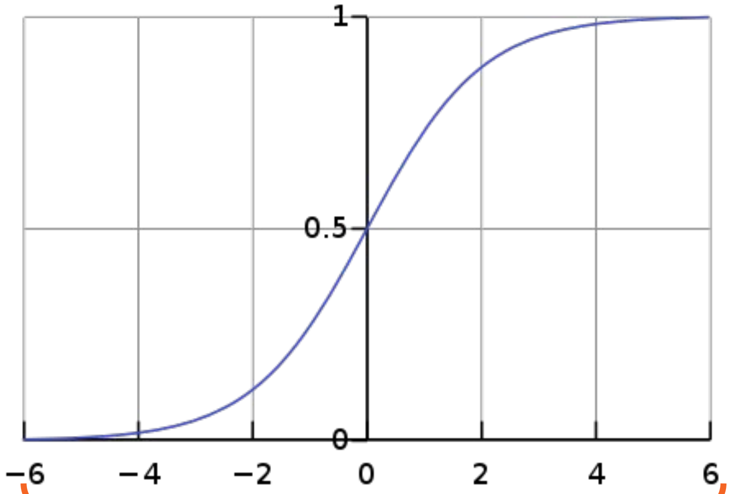
Intuition: log odds ratio

Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Intuition: log odds ratio

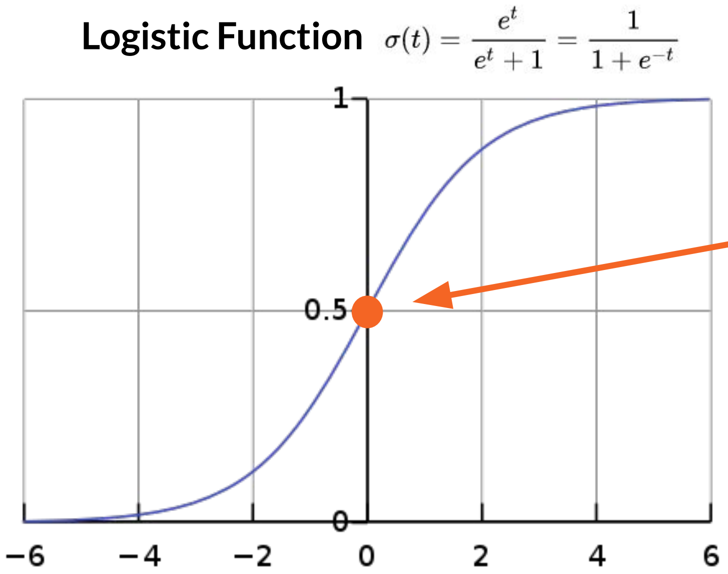
Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Log **Odds Ratio**

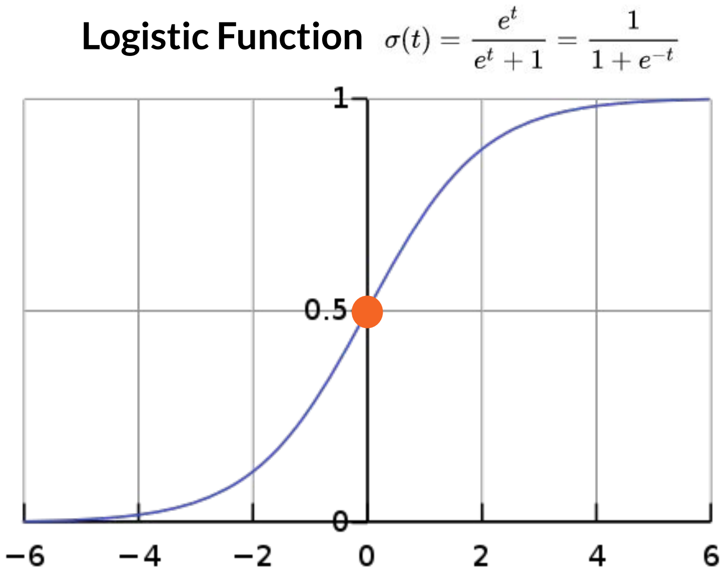
log(p(x) / (1-p(x))

Intuition: log odds ratio



Log odds x-axis	Probability y-axis	Odds
0.0	0.5	1:1

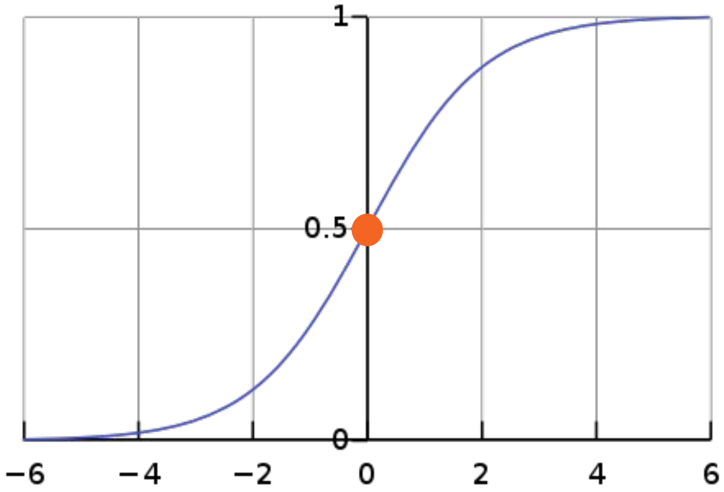
Intuition: log odds ratio



Log odds = log(Odds)	Probability	Odds $=e^{(\text{Log odds})}$
0.0	0.5	$e^0=1$ 1:1

Intuition: log odds ratio

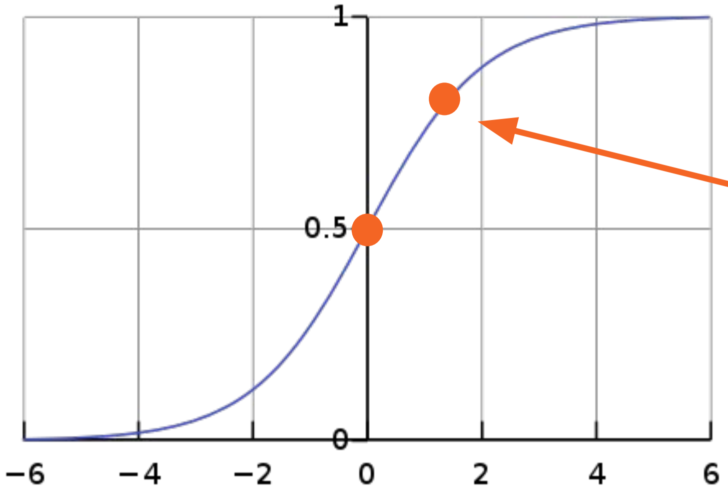
Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Log odds	Probability	Odds
0.0	1 / (1+1) = 0.5	1:1

Intuition: log odds ratio

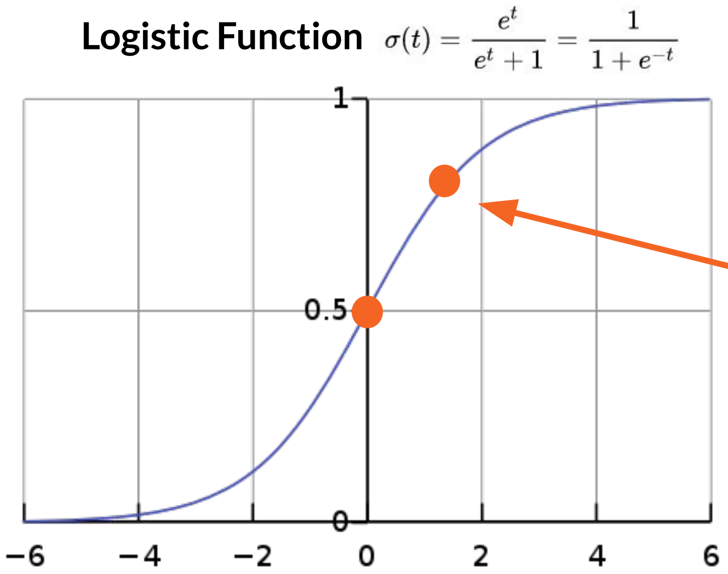
Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Log odds x-axis	Probability y-axis	Odds
0.0	0.5	1:1
1.38	0.8	4:1

σ(1.38)=0.8

Intuition: log odds ratio

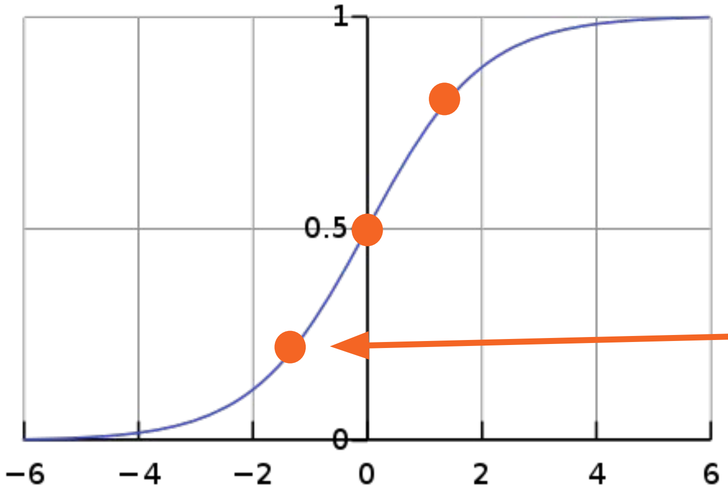


Log odds	Probability	Odds $e^{1.38} = 4$
0.0	0.5	1:1
1.38	0.8	4:1

0.8 = 4/(4+1)

Intuition: log odds ratio

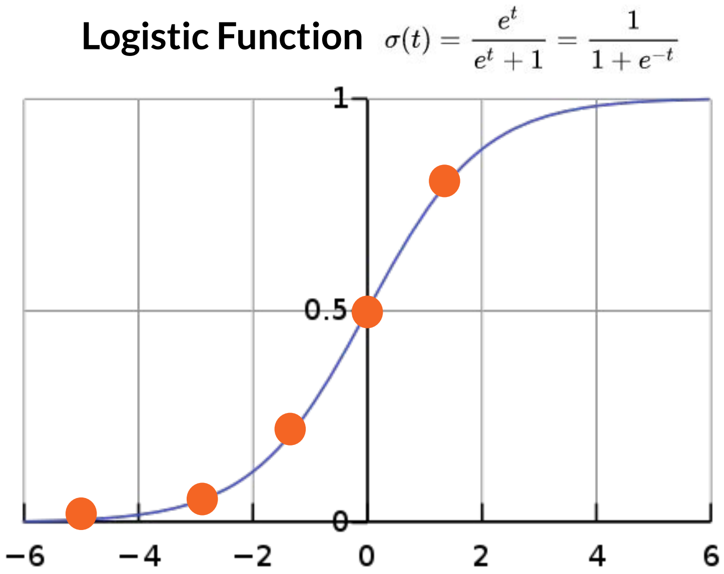
Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Log odds	Probability	Odds $e^{-1.38} = 0.25$
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4

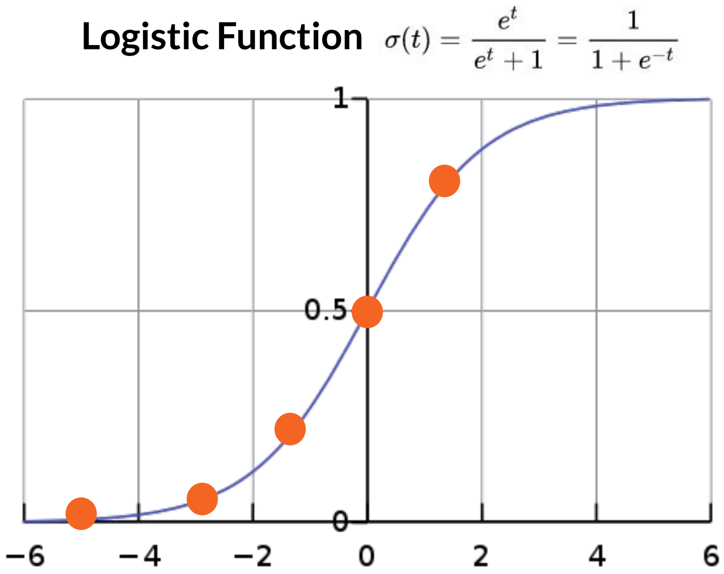
$\sigma(-1.38)=0.2$ $0.2 = 1/(1+4)$

Intuition: log odds ratio



Log odds	Probability	Odds
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4
-2.94	0.05	1:?
-4.59	0.01	1:?

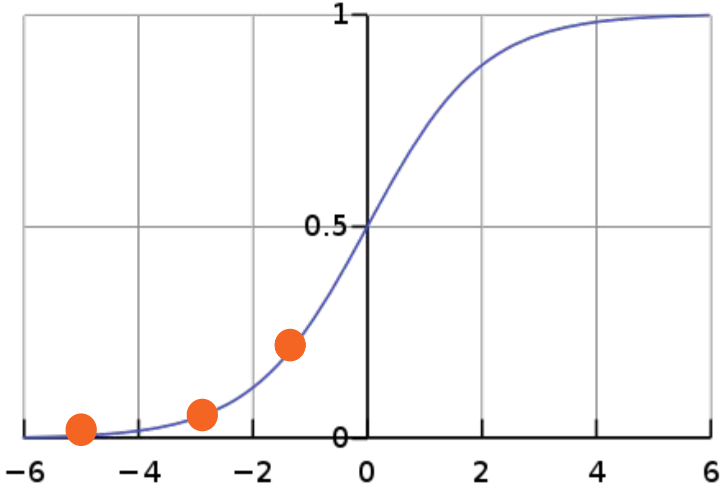
Intuition: log odds ratio



Log odds	Probability	Odds
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4
-2.94	0.05	1:19
-4.59	0.01	1:99

Intuition: log odds ratio

Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$

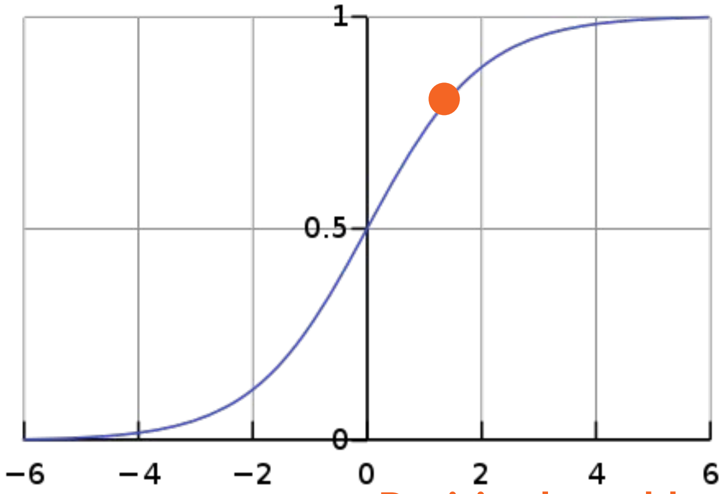


Negative log odds: Magnus more likely to lose than to win

Log odds	Probability	Odds
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4
-2.94	0.05	1:19
-4.59	0.01	1:99

Intuition: log odds ratio

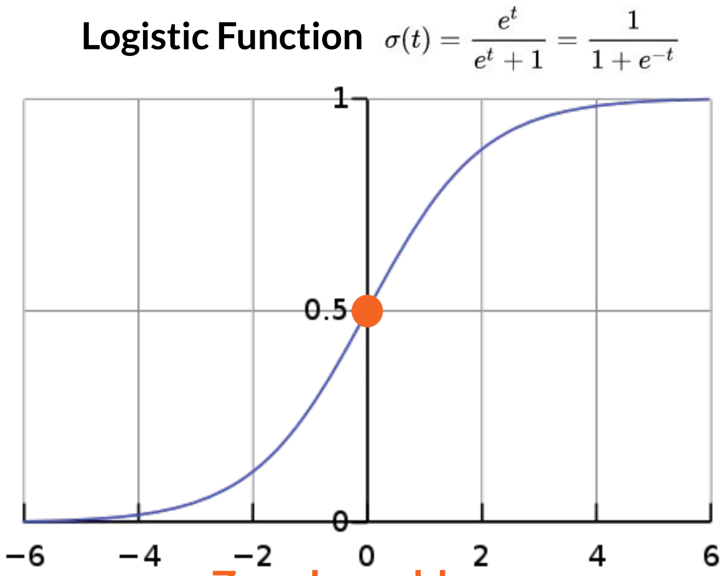
Logistic Function $\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$



Positive log odds: Magnus more likely to win than to lose

Log odds	Probability	Odds
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4
-2.94	0.05	1:19
-4.59	0.01	1:99

Intuition: log odds ratio



Zero log odds:
Magnus equally likely
to win or lose

Log odds	Probability	Odds
0.0	0.5	1:1
1.38	0.8	4:1
-1.38	0.2	1:4
-2.94	0.05	1:19
-4.59	0.01	1:99

Summarizing logistic regressions

Logistic	
$y \sim \sigma(\alpha + \beta x)$	For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}
(y must be binary)	1 unit change in x is associated with a $100*(e^{\beta} - 1)\%$ change in y

From last time...

For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}

- x = **whether you're a smoker**,
  y = whether you develop heart disease,
  α = -1.93,
  β = 0.38

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

From last time...

For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}

- x = whether you're a smoker, y = whether you develop heart disease, α = -1.93, β = 0.38
- According to our model, smokers have $e^{0.38}$ =1.46 times the odds of non-smokers of having heart disease. Smokers have 46% more odds of having heart disease than non-smokers.

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

When interpreting regressions on the prelim...

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

What about predicting?

Logistic	The probability that x=0 yields output y=1 is $e^{\alpha}/(e^{\alpha}+1)$
$y \sim \sigma(\alpha + \beta x)$	For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}
(y must be binary)	1 unit change in x is associated with a $100^{*}(e^{\beta} - 1)\%$ change in y

Predicting logistic regression

There is a $e^{\alpha}/(e^{\alpha}+1)$ probability that x=0 will have output y=1

$e^{-1.93}/(1+e^{-1.93}) = 0.13$

- x = kg of tobacco smoked,
  y = whether you develop heart disease,
  α = -1.93,
  β = 0.38
- **Your prediction at x=0:**

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

Summarizing logistic regression

There is a $e^{\alpha}/(e^{\alpha}+1)$ probability that x=0 will have output y=1

- x = kg of tobacco smoked,
  y = whether you develop heart disease,
  α = -1.93,
  β = 0.38
- Our model estimates that the probability that someone who has smoked 0 kg of tobacco will develop heart disease is $e^{-1.93}/(1+e^{-1.93}) = 0.13$.

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

What if x is also binary?

There is a $e^{\alpha}/(e^{\alpha}+1)$ probability that x=0 will have output y=1

$e^{-1.93}/(1+e^{-1.93}) = 0.13$

- x = **whether you're a smoker**,
  y = whether you develop heart disease,
  $\alpha = -1.93$,
  $\beta = 0.38$
- **Your prediction at x=0:**

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

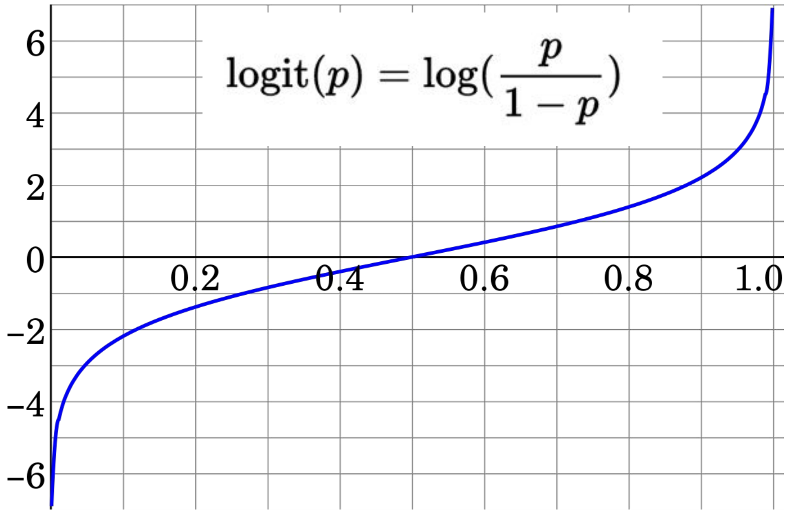
What if x is also binary?

There is a $e^{\alpha}/(e^{\alpha}+1)$ probability that x=0 will have output y=1

- x = whether you're a smoker,
  y = whether you develop heart disease,
  α = -1.93,
  β = 0.38
- Our model estimates that the probability that a non-smoker will develop heart disease is $e^{-1.93}/(1+e^{-1.93}) = 0.13$.

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

Next lecture: deriving the interpretations on the midterm handout posted on Canvas



The probability that x=0 yields output y=1 is $e^{\alpha}/(e^{\alpha}+1)$

For a 1 unit change in x, we expect the odds of y to be multiplied by e^{β}

1 unit change in x is associated with a $100*(e^{\beta} - 1)$% change in y

https://en.wikipedia.org/wiki/Logit

Oddities / outliers for logistic reg

- x = **kg of tobacco smoked**,
  y = whether you develop heart disease,
  $\alpha = -1.93$,
  $\beta = 0.38$
- **Oddities:**

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

Oddities / outliers for logistic reg

- x = kg of tobacco smoked,

  y = whether you develop heart disease,

  α = -1.93,

  β = 0.38
- **Oddities:**

  Our model doesn't make sense for negative inputs of x.

  Our model only estimates probabilities of developing heart disease; maybe you'd prefer predicting other y's (like the # times you have to go to the cardiologist)

  Our model only takes into account tobacco smoking (and no other factors), but lots of other things affect heart disease!

https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

Logistic Regression on single variable

- $y \sim \sigma(\alpha + \beta x)$
- LogisticRegression.fit(x,y)
- One unit change in x corresponds with e^{β} times the odds of y

Linear Regression on single variable

- $y = \alpha + \beta x$
- LinearRegression.fit(x,y)
- One unit change in x corresponds with a β unit change in y

Regression on multiple variables?

- What if we have multiple inputs that we want to use to predict y?

https://twitter.com/nosnibor_mot/status/1436012809708580868/photo/1

Explaining the direction (sign of corr) and strength (closeness of corr to 1 or -1) of the symmetric relationship between x and y

https://twitter.com/nosnibor_mot/status/1436012809708580868/photo/1

Explaining the effect of x on y (direction = sign of β, strength = magnitude of β) →

https://twitter.com/nosnibor_mot/status/1436012809708580868/photo/1

Explaining the effects of *multiple x's* on y

https://twitter.com/nosnibor_mot/status/1436012809708580868/photo/1

The joke here is to just give up on data science, but we'll teach you more methods!

https://twitter.com/nosnibor_mot/status/1436012809708580868/photo/1

1 min break & attendance!

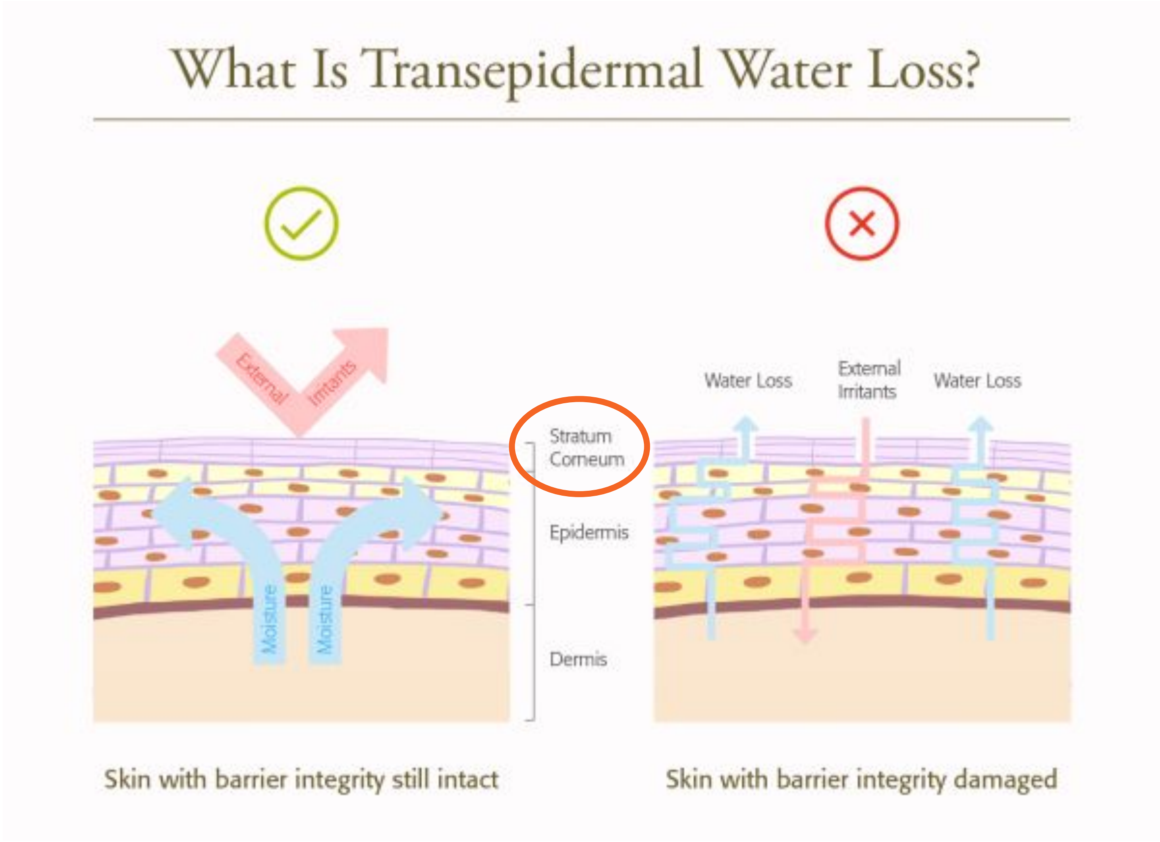
tinyurl.com/mcbv8v2j

Regression on multiple variables?

- What if we have multiple inputs that we want to use to predict y?

Regression on multiple variables?

- What if we have multiple inputs that we want to use to predict y?
- Example:
  - y = stratum corneum hydration
  - what are some inputs that could explain this output variable?



https://eminenceorganics.com/ca/blog/2021/09/20/what-transepidermal-water-loss-how-stop-it

https://www.femina.in/beauty/10-step-korean-skin-care-routine-203955.html

Lots of things can affect the dewiness of your skin!

- **y = stratum corneum hydration**
  - **x_1** = amount of moisturizer used (ml)
  - **x_2** = do you use exfoliant (y/n)
  - **x_3** = # times/week sheet mask used
  - …and many more potential x's!
- **How do we put this all in one model?**

Formalizing multivariable regression

i	x	y
1	78	18
2	83	14
…	…	…

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

Formalizing multivariable regression

i	x	y
1	78	18
2	83	14
…	…	…

i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

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

$$\Downarrow$$

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

Formalizing multivariable regression

i	x	y
1	78	18
2	83	14
…	…	…

i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

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

$$y = 5 + 10x$$

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

$$y = 3 + 5x_1 + 6x_2 - 8x_3$$

Formalizing multivariable regression

i	x_1	y
1	78	18
2	83	14
…	…	…

i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

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

$$y = 5 + 10\boxed{x_1}$$

$$\Downarrow$$

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

$$y = 3 + 5x_1 + 6x_2 - 8x_3$$

Formalizing multivariable regression

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

$$\boxed{y_i} = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

output y for ith data point

Formalizing multivariable regression

covariate ≈ feature ≈ variable ≈ input ≈ independent variables

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

$$y_i = \alpha + \beta_1 \boxed{x_{1,i}} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

input for 1st "covariate" x_1
for i^{th} data point

Formalizing multivariable regression

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

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 \boxed{x_{2,i}} + \beta_3 \boxed{x_{3,i}} + \ldots + \varepsilon_i$$

inputs for n^{th} "covariates" x_n
for i^{th} data point

Formalizing multivariable regression

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

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

$x_{2,1}$

i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

$x_{3,2}$

Formalizing multivariable regression

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

$$y_i = \alpha + \beta_1 \boxed{x_{1,i}} + \beta_2 \boxed{x_{2,i}} + \beta_3 \boxed{x_{3,i}} + \ldots + \varepsilon_i$$

i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

Do x_1, x_2, and x_3 all need to be the same data type as each other?

Formalizing multivariable regression

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

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

These are potentially…

	int	bool	float	
i	x_1	x_2	x_3	y
1	78	0	30.5	18
2	83	1	28.0	14
…	…	…	…	…

No, they just need to each be a data type that can be used with regression (i.e., not strings/objects).

But, the rows within column x_1 (i.e., $x_{1,i}$ for all i's) need to all be the same data type (dataframe definition!)

Formalizing multivariable regression

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

$$\Downarrow$$

$$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \boxed{\varepsilon_i}$$

error for i^{th}
data point

Formalizing multivariable regression

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

$$\Downarrow$$

$$y_i = \underbrace{\alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i}}_{\text{Deterministic Model: } \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3} + \ldots + \varepsilon_i$$

Formalizing multivariable regression

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

$$y_i = \boxed{\alpha} + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

intercept (same for all data points i)

Formalizing multivariable regression

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

$$y_i = \alpha + \boxed{\beta_1} x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

Slope for 1st "covariate" x_1
(same for all data points *i*)

Formalizing multivariable regression

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

$$y_i = \alpha + \beta_1 x_{1,i} + \boxed{\beta_2} x_{2,i} + \beta_3 x_{3,i} + \ldots + \varepsilon_i$$

Slope for 2nd "covariate" x_2
(same for all data points *i*)

Formalizing multivariable regression

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

$$y_i = \alpha + \boxed{\beta_1} x_{1,i} + \boxed{\beta_2} x_{2,i} + \boxed{\beta_3} x_{3,i} + \ldots + \varepsilon_i$$

Are these β's always going to be the same value?

Formalizing multivariable regression

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

$$y_i = \alpha + \boxed{\beta_1} x_{1,i} + \boxed{\beta_2} x_{2,i} + \boxed{\beta_3} x_{3,i} + \ldots + \varepsilon_i$$

Nope! β_1 will be the same across all x's plugged into the regression, as will β_2 and β_3, but there's no reason that β_1 would need to be the same as β_2 or β_3

https://www.femina.in/beauty/10-step-korean-skin-care-routine-203955.html

skincare_df (data from Indonesia)

Product	USD	Category	Brand	OverallRating
Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8
Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2
Thermal Spring Water	13.13	Face Mist	Avene	4.4
White Secret Night Cream	6.47	Night Cream	Wardah	3.6
Mineral Water Spray	10.56	Face Mist	Evian	3.8
...	...	...	...	...
Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1
Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3
Facial Lotion	0.99	Toner	Ovale	2.9
Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0
Rose Water Toner	12.76	Toner	Mamonde	4.2

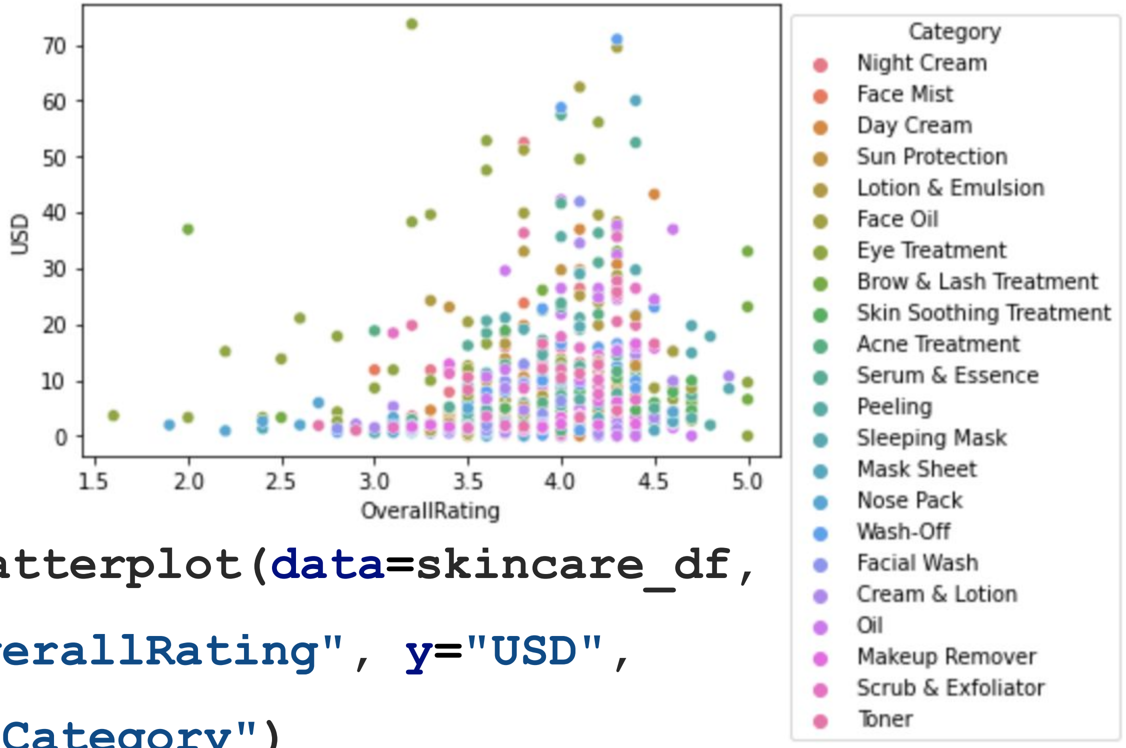
https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

skincare_df (data from Indonesia)

Disclaimer: data found on the internet and not validated; take results with grain of salt!

Product	USD	Category	Brand	OverallRating
Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8
Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2
Thermal Spring Water	13.13	Face Mist	Avene	4.4
White Secret Night Cream	6.47	Night Cream	Wardah	3.6
Mineral Water Spray	10.56	Face Mist	Evian	3.8
...	...	...	...	...
Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1
Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3
Facial Lotion	0.99	Toner	Ovale	2.9
Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0
Rose Water Toner	12.76	Toner	Mamonde	4.2

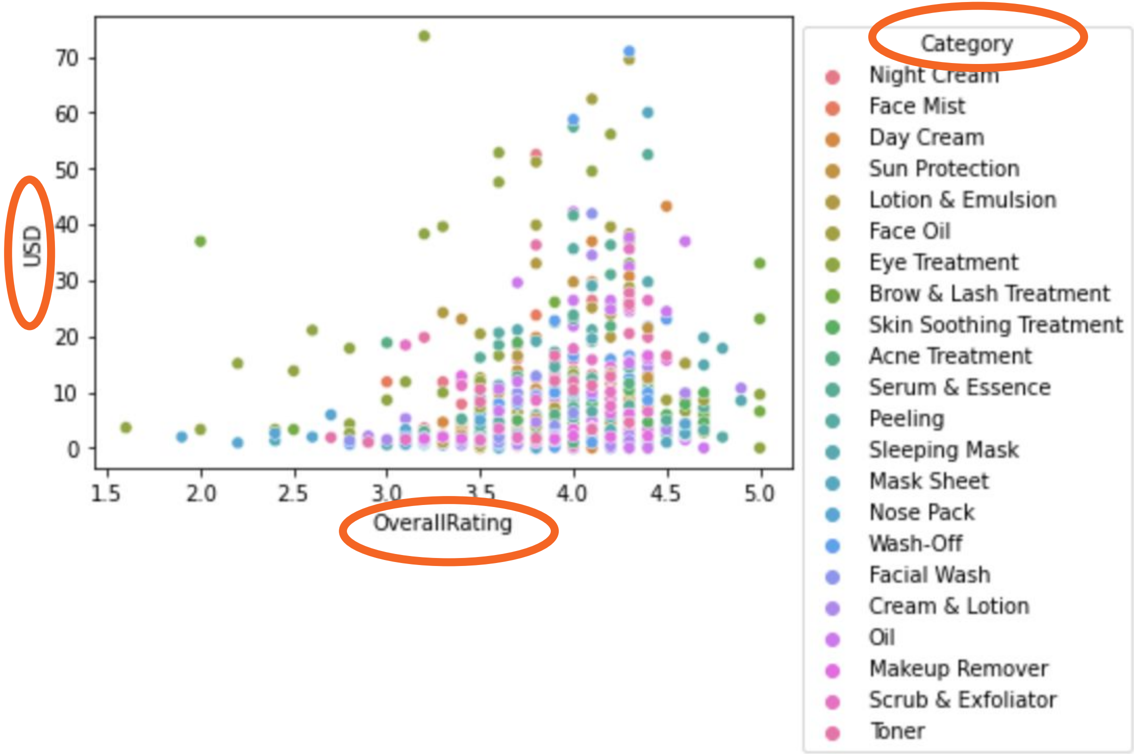
https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review




```
ax = sns.scatterplot(data=skincare_df,
     x="OverallRating", y="USD",
     hue="Category")
```

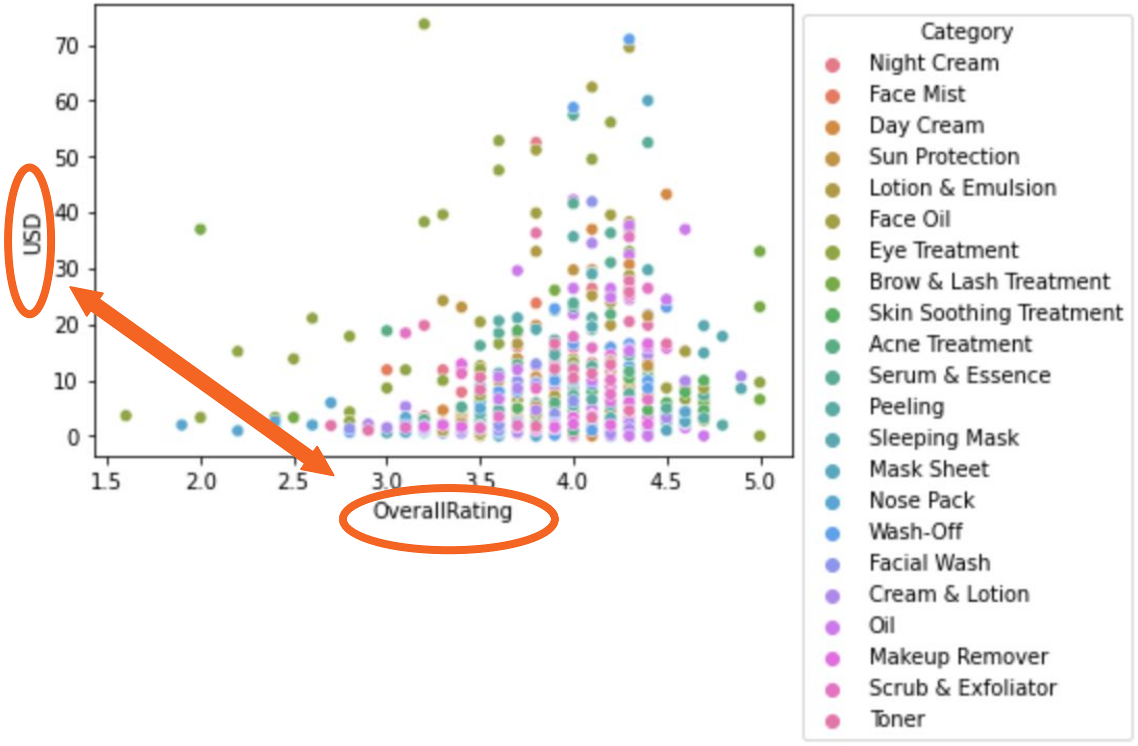
https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

What are some hypotheses you can make about these data?

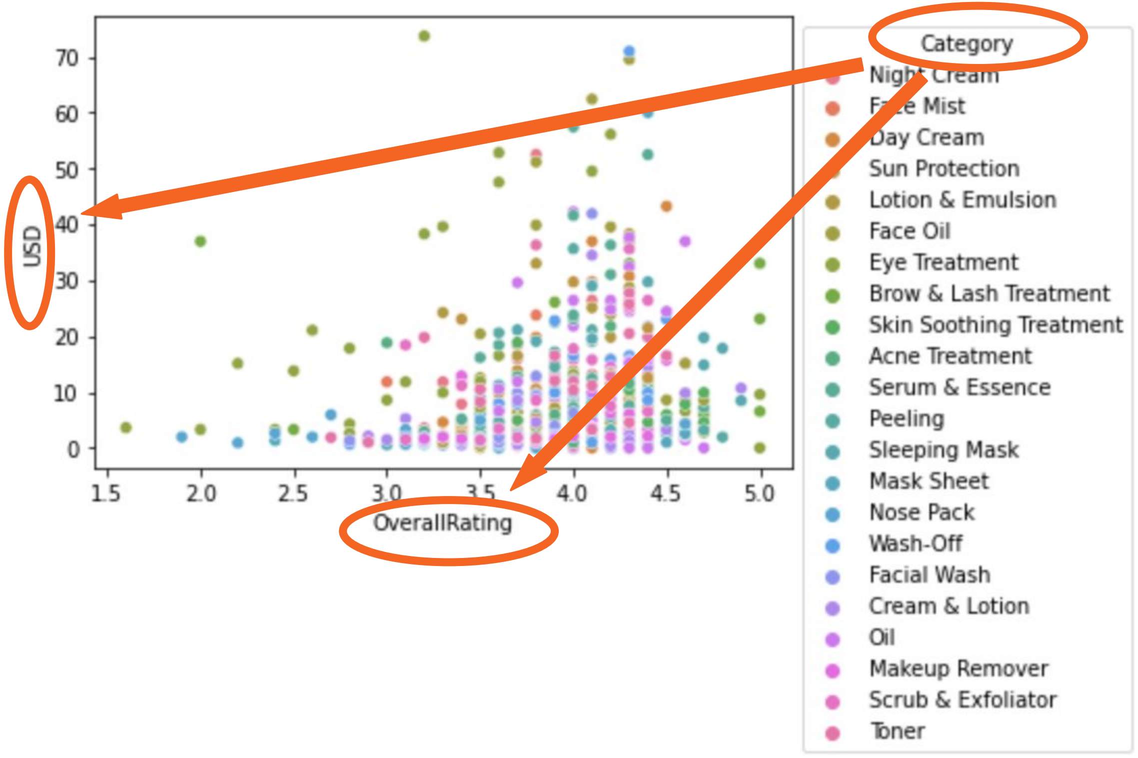


https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

- Better skincare products are more expensive?
- More expensive products are higher rated?



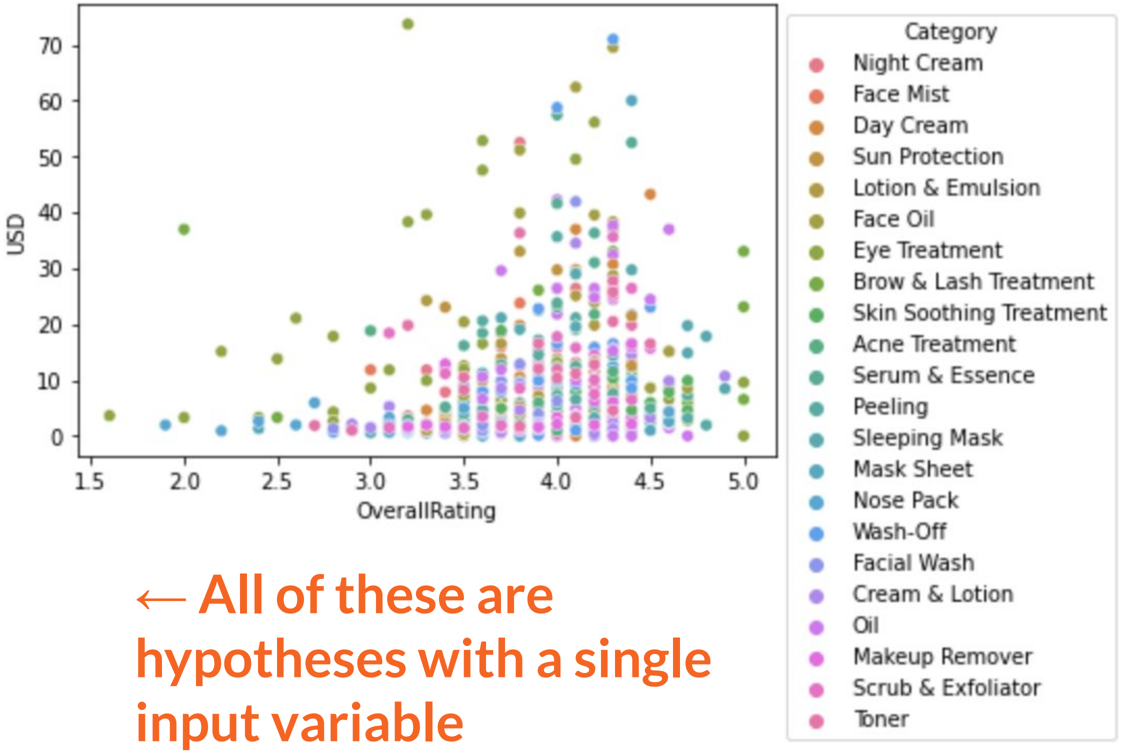
https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review



- Certain categories of skincare are more expensive?
- Certain categories of skincare are rated better?

https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

- Better skincare products are more expensive?
- More expensive products are higher rated?
- Certain categories of skincare are more expensive?
- Certain categories of skincare are rated better?

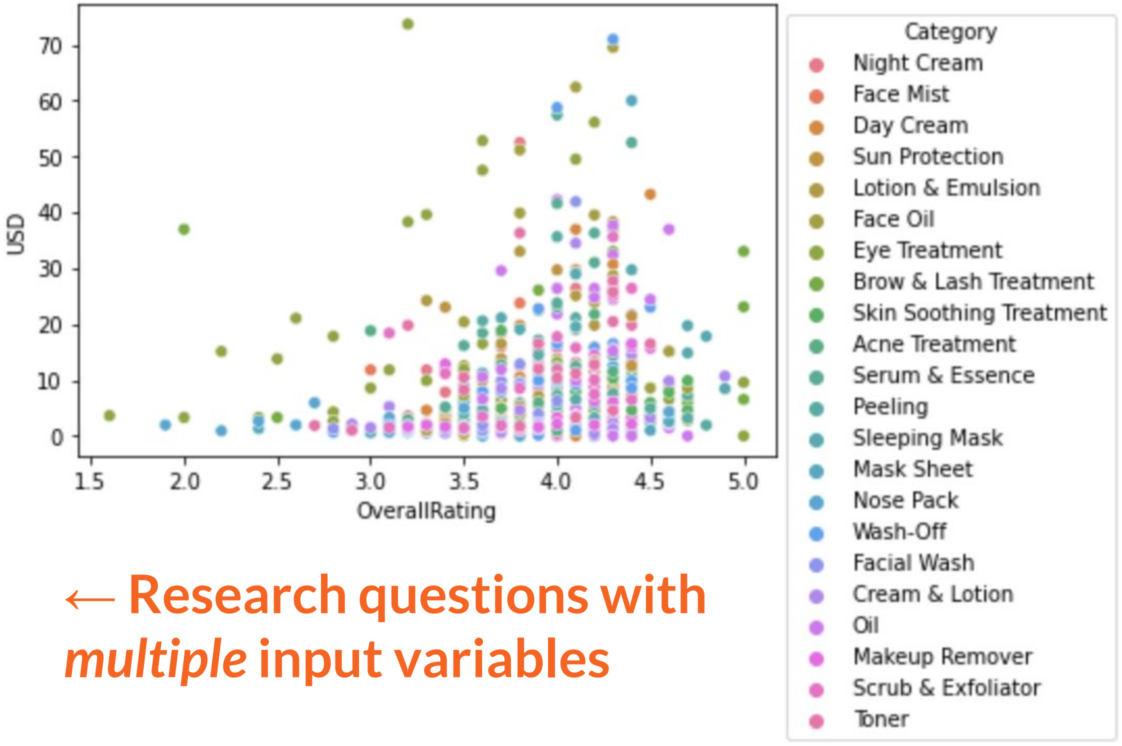


← All of these are hypotheses with a single input variable

https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

- Can we predict the price of skincare from its **ratings and category**?

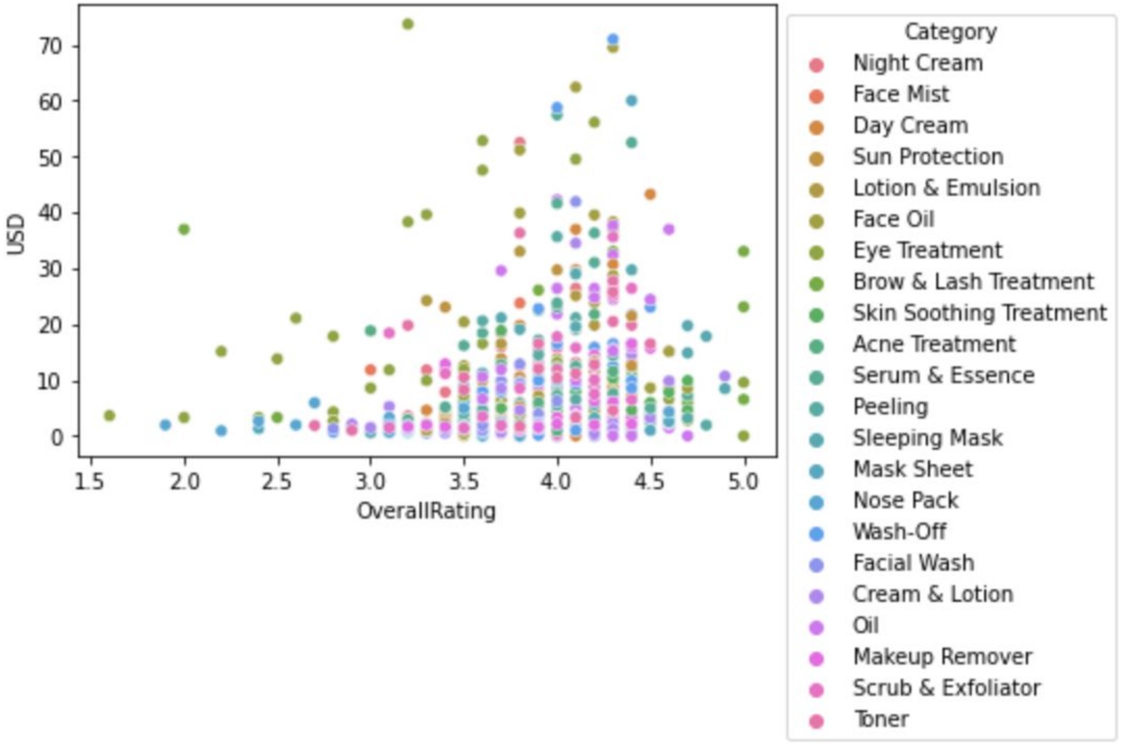
- Can we predict the category of skincare from **ratings and price**?



← Research questions with *multiple* input variables

Any guesses for which category is cheapest*?

*on average, according to our dataset from Indonesia



https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review


```
skincare_df.groupby('Category')['USD'].mean().sort_values()
```

```
Category
Nose Pack                   2.020000
Mask Sheet                  3.073590
Cream & Lotion              3.803514
Makeup Remover              4.145385
Facial Wash                 4.604595
Acne Treatment              4.729444
Skin Soothing Treatment     5.733056
Face Mist                   7.305556
Day Cream                   7.820882
Sun Protection              8.050882
Peeling                     8.885152
Lotion & Emulsion           9.252162
Toner                       9.574444
Night Cream                 9.803784
Scrub & Exfoliator          9.853056
Wash-Off                   10.715750
Sleeping Mask              12.296286
Brow & Lash Treatment      12.765238
Oil                        13.807838
Face Oil                   15.230000
Serum & Essence            16.082500
Eye Treatment              17.405714
```

https://www.amazon.com/Bior%C3%A9-Blackhead-Unclogging-Technology-Non-Comedogenic/dp/B0009EILKS

```
skincare_df.groupby('Category')['USD'].mean().sort_values()
```

```
Category
Nose Pack                   2.020000
Mask Sheet                  3.073590
Cream & Lotion              3.803514
Makeup Remover              4.145385
Facial Wash                 4.604595
Acne Treatment              4.729444
Skin Soothing Treatment     5.733056
Face Mist                   7.305556
Day Cream                   7.820882
Sun Protection              8.050882
Peeling                     8.885152
Lotion & Emulsion           9.252162
Toner                       9.574444
Night Cream                 9.803784
Scrub & Exfoliator          9.853056
Wash-Off                   10.715750
Sleeping Mask              12.296286
Brow & Lash Treatment      12.765238
Oil                        13.807838
Face Oil                   15.230000
Serum & Essence            16.082500
Eye Treatment              17.405714
```

pandas code != SQL code (can't be mixed in same line!)

```
skincare_df.groupby('Category')['USD'].mean().sort_values()
```

```
Category
Nose Pack                   2.020000
Mask Sheet                  3.073590
Cream & Lotion              3.803514
Makeup Remover              4.145385
Facial Wash                 4.604595
Acne Treatment              4.729444
Skin Soothing Treatment     5.733056
Face Mist                   7.305556
Day Cream                   7.820882
Sun Protection              8.050882
Peeling                     8.885152
Lotion & Emulsion           9.252162
Toner                       9.574444
Night Cream                 9.803784
Scrub & Exfoliator          9.853056
Wash-Off                   10.715750
Sleeping Mask              12.296286
Brow & Lash Treatment      12.765238
Oil                        13.807838
Face Oil                   15.230000
Serum & Essence            16.082500
Eye Treatment              17.405714
```

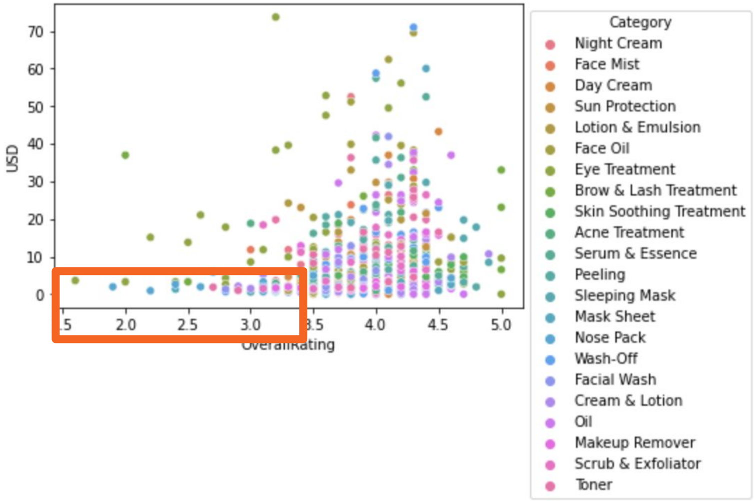
These seem pretty different!

Maybe both rating *and category* affect price...

```
skincare_df.groupby('Category')['USD'].mean().sort_values()
```

```
Category
Nose Pack                   2.020000
Mask Sheet                  3.073590
Cream & Lotion              3.803514
Makeup Remover              4.145385
Facial Wash                 4.604595
Acne Treatment              4.729444
Skin Soothing Treatment     5.733056
Face Mist
Day Cream
Sun Protection
Peeling
Lotion & Emulsion
Toner
Night Cream
Scrub & Exfoliator
```

Hypothesis: low ratings and/or being a nose pack are predictive of *low cost*



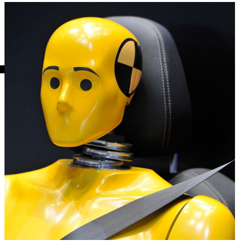
How do we get a binary variable for whether a product is a nose pack?

Product	USD	Category	Brand	OverallRating
Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8
Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2
Thermal Spring Water	13.13	Face Mist	Avene	4.4
White Secret Night Cream	6.47	Night Cream	Wardah	3.6
Mineral Water Spray	10.56	Face Mist	Evian	3.8
...	...	...	...	...
Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1
Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3
Facial Lotion	0.99	Toner	Ovale	2.9
Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0
Rose Water Toner	12.76	Toner	Mamonde	4.2

```python
skincare_df["is_nosepack"] = np.where(skincare_df["Category"].isin(['Nose Pack']),
True, False)
```

	Product	USD	Category	Brand	OverallRating	is_nosepack
	Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8	False
	Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2	False
	Thermal Spring Water	13.13	Face Mist	Avene	4.4	False
	White Secret Night Cream	6.47	Night Cream	Wardah	3.6	False
	Mineral Water Spray	10.56	Face Mist	Evian	3.8	False
	...	...	...	...	...	...
	Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1	False
	Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3	False
	Facial Lotion	0.99	Toner	Ovale	2.9	False
	Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0	False
	Rose Water Toner	12.76	Toner	Mamonde	4.2	False

is_nosepack is a dummy variable generated from *Category*



Product	USD	Category	Brand	OverallRating	is_nosepack
Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8	False
Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2	False
Thermal Spring Water	13.13	Face Mist	Avene	4.4	False
White Secret Night Cream	6.47	Night Cream	Wardah	3.6	False
Mineral Water Spray	10.56	Face Mist	Evian	3.8	False
...	...	...	...	...	...
Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1	False
Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3	False
Facial Lotion	0.99	Toner	Ovale	2.9	False
Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0	False
Rose Water Toner	12.76	Toner	Mamonde	4.2	False

https://www.forbes.com/sites/brucelee/2017/04/28/crash-test-dummies-heres-what-the-obesity-epidemic-is-doing-to-them/

Multivar Linear Regression

$y \sim x_1 + x_2$

$y_i = \alpha + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \varepsilon_i$

y = USD
x_1 = average product rating
x_2 = whether the product is a nose pack

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

All of this code is the same as when we only had y~x

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

What's different is we now have 2 input variables stored in X

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_        array([-3.16964665])
m1.coef_       array([[ 3.12824371, -5.95242138]]])
```

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

```
array([[ 3.12824371, -5.95242138]]])
```

You get one coefficient for each input variable x

Multivar Lin Reg: Formulation

- y = price of product in \$
- x_1 = avg customer rating
  x_2 = whether product is a nose pack
- $y \sim x_1 + x_2$
- $y = \alpha + \beta_1 x_1 + \beta_2 x_2$
- $y = -3.2 + 3.1x_1 - 6.0x_2$

When interpreting regressions on the prelim…

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

Interpret by Summarizing

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$

Interpret by Summarizing

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- According to our model, for each additional star rating given to the product, all else equal, we expect the price of the product to increase by $3.10

Interpret by Summarizing

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- According to our model, for each additional star rating given to the product, all else equal, we expect the price of the product to increase by $3.10

Holding all other input variables (x_2) constant, e.g., pretend x_2 is being fixed at a single value

Interpret by Summarizing

- y = price of product in \$
- x_1 = avg customer rating

  x_2 = whether product is a nose pack (binary!)
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- ______________________________

  ____

  ____

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- According to our model, all else equal, the product being a nose pack corresponds to a $6 reduction in estimated product price relative to the product not being a nose pack

Must include the "all else equal"!

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- According to our model, all else equal, the product being a nose pack corresponds to a $6 reduction in estimated product price relative to the product not being a nose pack

Must include the "reference": $x_2 = 1$ (nose pack) means $6 less than what?
Answer: $x_2 = 0$

1 minute break

http://www.quickmeme.com/meme/3okk37

Multivar Linear Regression

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

What if we care about other categories too?

Do we have to manually make a new column for each binary variable?

Multivar Linear Regression

```
X = skincare_df[["OverallRating", "is_nosepack"]]
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

We can automate creation of dummy variables!

Which rows are face mists?

Product	USD	Category	Brand	OverallRating
Perfect 3D Gel	6.01	Night Cream	Hada Labo	3.8
Aqua Beauty Protecting Mist	1.78	Face Mist	PIXY	4.2
Thermal Spring Water	13.13	Face Mist	Avene	4.4
White Secret Night Cream	6.47	Night Cream	Wardah	3.6
Mineral Water Spray	10.56	Face Mist	Evian	3.8
...	...	...	...	...
Vitamin E Hydrating Toner	11.15	Toner	The Body Shop	4.1
Skin Perfecting 2% BHA Liquid Exfoliant	25.74	Toner	Paula's Choice	4.3
Facial Lotion	0.99	Toner	Ovale	2.9
Centella Water Alcohol-Free Toner	10.36	Toner	Cosrx	4.0
Rose Water Toner	12.76	Toner	Mamonde	4.2

https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review

Dummies with pandas

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

Dummies with pandas

pd.get_dummies takes in a categorical column *(usually one you want as an input x)*, and returns all unique values of the original Category as their own binary columns

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

	Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	...	...	...	...	...	...	...	...	...	...	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

Hmm...

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

In *skincare_df* there were 22 unique values of Category.

Why are there 21 columns in this output?

Hmm...

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

	Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	1	0	0	0	0	0	...
	...	...	...	...	...	...	...	...	...	...	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...
	0	0	0	0	0	0	0	0	0	0	...

776 rows x 21 columns

We're telling pandas to drop one of the columns!

Why would we do this?!

Multivar Lin Reg: Interpreting

- y = price of product in \$
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$
- According to our model, all else equal, the product being a nose pack corresponds to a \$6 reduction in estimated product price relative to the product not being a nose pack

Must include the "reference": $x_2 = 1$ (nose pack) means \$6 less than what? Answer: $x_2 = 0$

Multivar Lin Reg: Interpreting

- y = price of product in \$
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$

  How come we didn't include
  x_3 = whether the product is not a nose pack?

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$

x_3 wouldn't add any new information:

- $x_3=1$ means $x_2=0$
- $x_3=0$ means $x_2=1$

How come we didn't include x_3 = whether the product is not a nose pack?

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$

An invisible x_3 acts as our "reference level" when interpreting the x_2 coefficient

How come we didn't include
x_3 = whether the product is not a nose pack?

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating

  x_2 = whether product is a nose pack
- $y = -3.2 + 3.1x_1 - 6.0x_2$

An invisible x_3 acts as our "reference level" when interpreting the x_2 coefficient

According to our model, all else equal, the product being a nose pack corresponds to a $6 reduction in estimated product price relative to the product not being a nose pack

Hmm...

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

The dropped column is our "reference category"

Before, we had 2 categories (is or is not nosepack) and dropped one to give us only *is_nosepack*

Hmm...

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

Does it matter that we drop the 1st category?

You can choose any of the 22 categories to be the reference; "first" is just one convention

Hmm…

The first category (alphanumerically) that gets dropped is “Acne Treatment”

```
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
categories
```

✓ 0.3s

Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	1	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

776 rows × 21 columns

Multivar Linear Regression (sklearn)

```
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

All the same code as before, we just need to regenerate X to include all our dummy input variables

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)
```

Regenerate X to include all our dummy input variables

```
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)

y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

Start with same X as before

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)

y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

Make X only contain numeric var OverallRating

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)

y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

Contains 21 binary values (including nose pack, dropping the "reference level")

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)
```

Concatenate columns along 1 axis (side by side)

```
y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

Multivar Linear Regression (sklearn)

```
X = skincare_df[["OverallRating", "is_nosepack"]]
X = X.drop("is_nosepack",axis=1)
categories=pd.get_dummies(skincare_df["Category"],drop_first=True)
X = pd.concat([X, categories],axis=1)


y = skincare_df[["USD"]]
m1 = LinearRegression().fit(X,y)
yhat = m1.predict(X)
m1.intercept_
m1.coef_
```

m1.intercept_ → array([-10.6485749])

m1.coef_ →

```
array([[ 4.09776977,  7.0227339 , -1.57351392,
3.54875437, 13.70038707,
         1.6768783 ,  9.25333753,  0.01389596,
4.39566224, -1.83878292,
        -3.55151102,  4.71470774, -1.92671685,
7.54480613,  3.61451172,
         4.2243783 , 10.35280134, -0.65826219,
6.28449951,  2.34453495,
         4.05959413,  5.4752226 ]])
```

Multivar Lin Reg: Interpreting

Now we have 22 input variables: 1 numeric and 21 dummy variables, of which only one of them can = 1 per row

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  ...
  x_{22} = whether product is Wash-Off

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  ...
  x_{22} = whether product is Wash-Off

What is the product if $x_2 = x_3 = \ldots = x_{22} = 0$?

Multivar Lin Reg: Interpreting

- y = price of product in \$
- x_1 = avg customer rating

  x_2 = whether product is Brow & Lash Treatment

  x_3 = whether product is Cream & Lotion

  ...

  x_{22} = whether product is Wash-Off

What is the product if $x_2 = x_3 = \ldots = x_{22} = 0$?

Acne Treatment (the reference variable!)

Multivar Lin Reg: Interpreting

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  ...
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + ... + 5.5x_{22}$

Interpret x_2 by Summarizing

- y = price of product in \$
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  …
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \dots + 5.5x_{22}$
- ____

  ____

Interpret x_2 by Summarizing

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  …
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \ldots + 5.5x_{22}$
- All else equal, our model finds that a Brow & Lash Treatment skincare product would be $7 more expensive than an Acne Treatment product

Predicting for multivariable regs?

What is ŷ for a 4-star Brow & Lash treatment?

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  …
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \ldots + 5.5x_{22}$
- ______________________________________

Predicting for multivariable regs?

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  ...
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \ldots + 5.5x_{22}$
- Our model predicts that the price of a 4-star Brow and Lash treatment would be -10.6+4.09*4+7.0 = $12.76

Plug in:
$x_1=4$
$x_2=1$
$x_{3,\ldots,22}=0$

Oddities for multivariable regs?

- y = price of product in \$
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  …
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \ldots + 5.5x_{22}$
- ____
  ____

Oddities for multivariable regs?

- y = price of product in $
- x_1 = avg customer rating
  x_2 = whether product is Brow & Lash Treatment
  x_3 = whether product is Cream & Lotion
  …
  x_{22} = whether product is Wash-Off
- $y = -10.6 + 4.09x_1 + 7.0x_2 - 1.6x_3 + \ldots + 5.5x_{22}$
- For 1-star Acne Treatment products, the model predicts the price is NEGATIVE $6.51. That doesn't make sense
- There's nothing stopping you from inputting negative or > 5 star values for x_1, which could be dangerous
- Should x_1 be numeric or should we turn those into dummies as well?

Multivariable Regression in Python

- In Python, running multivariable regression is basically the same as single variable regression, but with higher dimensions of X
- `model1 = LinearRegression().fit(X,y)`
- `model2 = LogisticRegression().fit(X,y)`

Capital X represents a *matrix* that contains *multiple* columns of your df

How does including more x's change our regressions?

- If most of the regression is the same, will the coefficients change?
- What could cause coefficients to change a lot (e.g. very different magnitudes, even changing signs)?

Four input x's

Regression: **sales ~ price + ad + loc + volume**

	Estimate
(Intercept)	125.931
price	-11.836
ad	131.283
loc	7.768
volume	11.870

https://www.edupristine.com/blog/detecting-multicollinearity

Three input x's

Regression: sales ~ price + ad + loc ~~+ volume~~

	Estimate
(Intercept)	662.733
price	-15.100
ad	20.500
loc	1.833

https://www.edupristine.com/blog/detecting-multicollinearity

Coefficients change when you add / remove inputs!

Coefficients are “jointly estimated” – more on this later

	Estimate
(Intercept)	125.931
price	-11.836
ad	131.283
loc	7.768
volume	11.870

	Estimate
(Intercept)	662.733
price	-15.100
ad	20.500
loc	1.833

https://www.edupristine.com/blog/detecting-multicollinearity

Should the effect of ads on sales be *that* different?

	Estimate
(Intercept)	125.931
price	-11.836
ad	131.283
loc	7.768
volume	11.870

	Estimate
(Intercept)	662.733
price	-15.100
ad	20.500
loc	1.833

https://www.edupristine.com/blog/detecting-multicollinearity

What happens if we include "collinear" inputs?

- **Collinearity =** correlation between inputs

What happens if we include "collinear" inputs?

- **Collinearity** = correlation between inputs
- **Are x_1 and x_2 correlated?**
  - $\mathbf{x_1}$ = binary: use oil cleanser daily
  - $\mathbf{x_2}$ = binary: does not use oil cleanser daily

What happens if we include "collinear" inputs?

- **Collinearity =** correlation between inputs
- **Are x_1 and x_2 correlated? Yes (corr = -1)**
  - **x_1 =** binary: use oil cleanser daily
  - **x_2 =** binary: does not use oil cleanser daily

Collinear cat variables

Collinear cat variables

Two distinct variables with different magnitudes

Collinear cat variables

... but they always provide the same information

Maybe we get funky* results because of collinear variables being added to the regression!

	Estimate
(Intercept)	125.931
price	-11.836
ad	131.283
loc	7.768
volume	11.870

	Estimate
(Intercept)	662.733
price	-15.100
ad	20.500
loc	1.833

*(big coefficient differences, including in some cases even changing signs)

https://www.edupristine.com/blog/detecting-multicollinearity

Multicollinearity

- Note: it might not always be obvious what covariates are collinear to each other
- To check for multicollinearity: get the **correlation matrix** of all the covariates
  - **What would be bad news?**

Multicollinearity in corr matrix

Bad news: *volume* highly correlated with *ad*

	sales	price	ad	loc	volume
sales	1.00	-0.70	0.12	0.01	0.39
price	-0.70	1.00	0.00	0.00	-0.18
ad	0.12	0.00	1.00	0.00	-0.74
loc	0.01	0.00	0.00	1.00	-0.04
volume	0.39	-0.18	-0.74	-0.04	1.00

https://www.edupristine.com/blog/detecting-multicollinearity

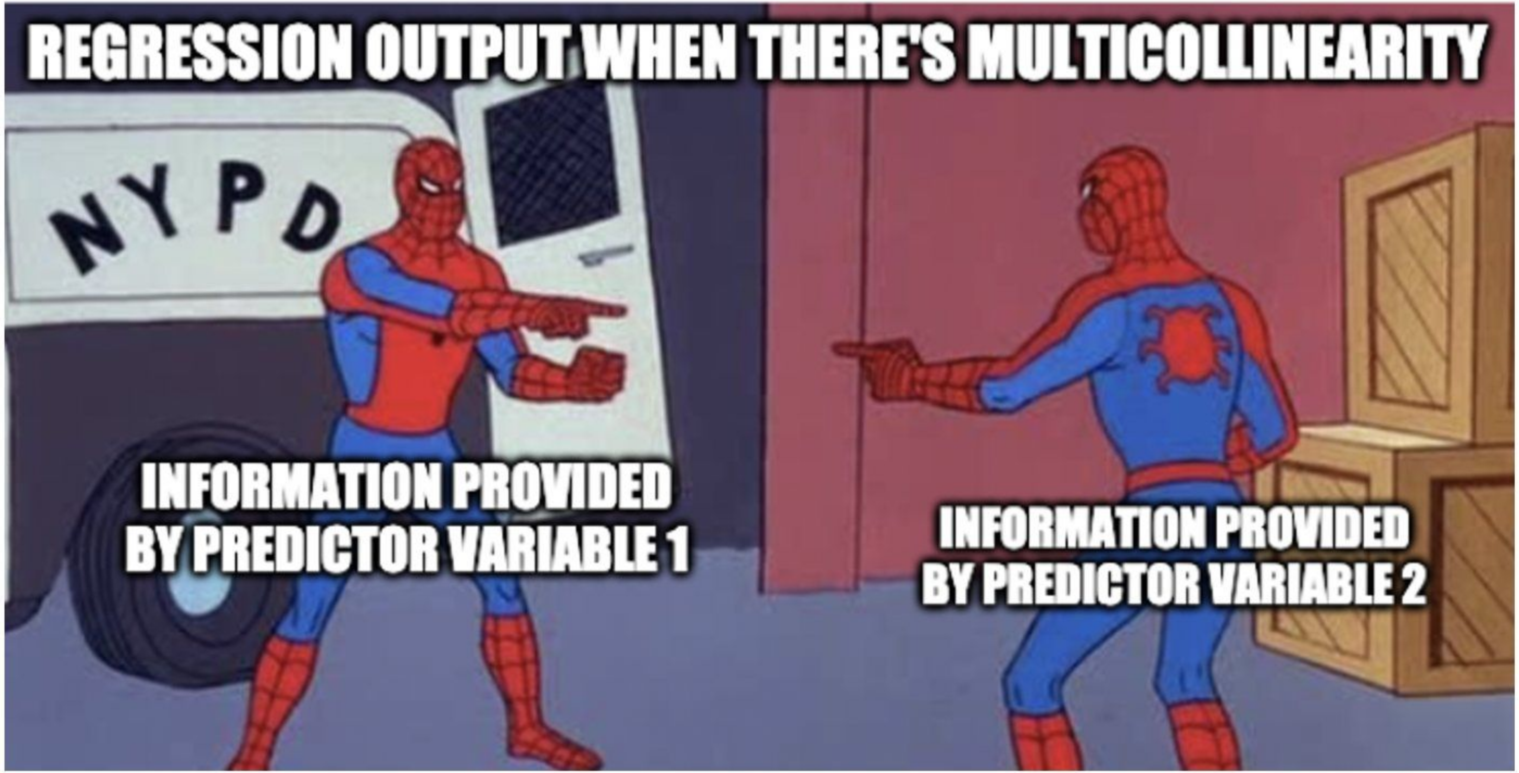
Multicollinearity in corr matrix

Bad news: *volume* highly correlated with *ad*

	sales	price	ad	loc	volume
sales	1.00	-0.70	0.12	0.01	0.39
price	-0.70	1.00	0.00	0.00	-0.18
ad	0.12	0.00	1.00	0.00	-0.74
loc	0.01	0.00	0.00	1.00	-0.04
volume	0.39	-0.18	-0.74	-0.04	1.00

Volume and ad are likely *collinear*; generally we should trust three-input regression more

https://www.edupristine.com/blog/detecting-multicollinearity



https://twitter.com/v_matzek/status/1376959863276806145/photo/3

What happens if we include "collinear" inputs?

- **Collinearity =** correlation between inputs
- **Are x_1 and x_2 correlated? Yes (corr = -1)**
  - **x_1 =** binary: use oil cleanser daily
  - **x_2 =** binary: does not use oil cleanser daily

Multicollinearity

- Why was it okay to include our skincare category dummies x_2 through x_{22} in one regression?

Multicollinearity

- Why was it okay to include our skincare category dummies x_2 through x_{22} in one regression?
- Because they don't include the reference variable (which would be perfectly collinear with the combination of other columns)

`categories.corr()` shows low correlations

	Brow & Lash Treatment	Cream & Lotion	Day Cream	Eye Treatment	Face Mist	Face Oil	Facial Wash	Lotion & Emulsion	Makeup Remover	Mask Sheet	...	Nose Pack	Oil	Peeling	Scrub & Exfoliator	Serum & Essence	Skin Soothing Treatment	Sleeping Mask	Sun Protection	Toner	Wash-Off
Brow & Lash Treatment	1.000000	-0.038127	-0.039715	-0.039715	-0.040233	-0.039192	-0.039715	-0.040233	-0.039715	-0.039715	...	-0.039715	-0.040233	-0.040233	-0.039192	-0.039715	-0.039715	-0.038663	-0.039192	-0.039192	-0.039715
Cream & Lotion	-0.038127	1.000000	-0.046773	-0.046773	-0.047383	-0.046157	-0.046773	-0.047383	-0.046773	-0.046773	...	-0.046773	-0.047383	-0.047383	-0.046157	-0.046773	-0.046773	-0.045533	-0.046157	-0.046157	-0.046773
Day Cream	-0.039715	-0.046773	1.000000	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Eye Treatment	-0.039715	-0.046773	-0.048721	1.000000	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Face Mist	-0.040233	-0.047383	-0.049356	-0.049356	1.000000	-0.048706	-0.049356	-0.050000	-0.049356	-0.049356	...	-0.049356	-0.050000	-0.050000	-0.048706	-0.049356	-0.049356	-0.048048	-0.048706	-0.048706	-0.049356
Face Oil	-0.039192	-0.046157	-0.048079	-0.048079	-0.048706	1.000000	-0.048079	-0.048706	-0.048079	-0.048079	...	-0.048079	-0.048706	-0.048706	-0.047445	-0.048079	-0.048079	-0.046805	-0.047445	-0.047445	-0.048079
Facial Wash	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	1.000000	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Lotion & Emulsion	-0.040233	-0.047383	-0.049356	-0.049356	-0.050000	-0.048706	-0.049356	1.000000	-0.049356	-0.049356	...	-0.049356	-0.050000	-0.050000	-0.048706	-0.049356	-0.049356	-0.048048	-0.048706	-0.048706	-0.049356
Makeup Remover	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	1.000000	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Mask Sheet	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	1.000000	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Night Cream	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Nose Pack	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	1.000000	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Oil	-0.040233	-0.047383	-0.049356	-0.049356	-0.050000	-0.048706	-0.049356	-0.050000	-0.049356	-0.049356	...	-0.049356	1.000000	-0.050000	-0.048706	-0.049356	-0.049356	-0.048048	-0.048706	-0.048706	-0.049356
Peeling	-0.040233	-0.047383	-0.049356	-0.049356	-0.050000	-0.048706	-0.049356	-0.050000	-0.049356	-0.049356	...	-0.049356	-0.050000	1.000000	-0.048706	-0.049356	-0.049356	-0.048048	-0.048706	-0.048706	-0.049356
Scrub & Exfoliator	-0.039192	-0.046157	-0.048079	-0.048079	-0.048706	-0.047445	-0.048079	-0.048706	-0.048079	-0.048079	...	-0.048079	-0.048706	-0.048706	1.000000	-0.048079	-0.048079	-0.046805	-0.047445	-0.047445	-0.048079
Serum & Essence	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	1.000000	-0.048721	-0.047430	-0.048079	-0.048079	-0.048721
Skin Soothing Treatment	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	1.000000	-0.047430	-0.048079	-0.048079	-0.048721
Sleeping Mask	-0.038663	-0.045533	-0.047430	-0.047430	-0.048048	-0.046805	-0.047430	-0.048048	-0.047430	-0.047430	...	-0.047430	-0.048048	-0.048048	-0.046805	-0.047430	-0.047430	1.000000	-0.046805	-0.046805	-0.047430
Sun Protection	-0.039192	-0.046157	-0.048079	-0.048079	-0.048706	-0.047445	-0.048079	-0.048706	-0.048079	-0.048079	...	-0.048079	-0.048706	-0.048706	-0.047445	-0.048079	-0.048079	-0.046805	1.000000	-0.047445	-0.048079
Toner	-0.039192	-0.046157	-0.048079	-0.048079	-0.048706	-0.047445	-0.048079	-0.048706	-0.048079	-0.048079	...	-0.048079	-0.048706	-0.048706	-0.047445	-0.048079	-0.048079	-0.046805	-0.047445	1.000000	-0.048079
Wash-Off	-0.039715	-0.046773	-0.048721	-0.048721	-0.049356	-0.048079	-0.048721	-0.049356	-0.048721	-0.048721	...	-0.048721	-0.049356	-0.049356	-0.048079	-0.048721	-0.048721	-0.047430	-0.048079	-0.048079	1.000000