

# Correlation

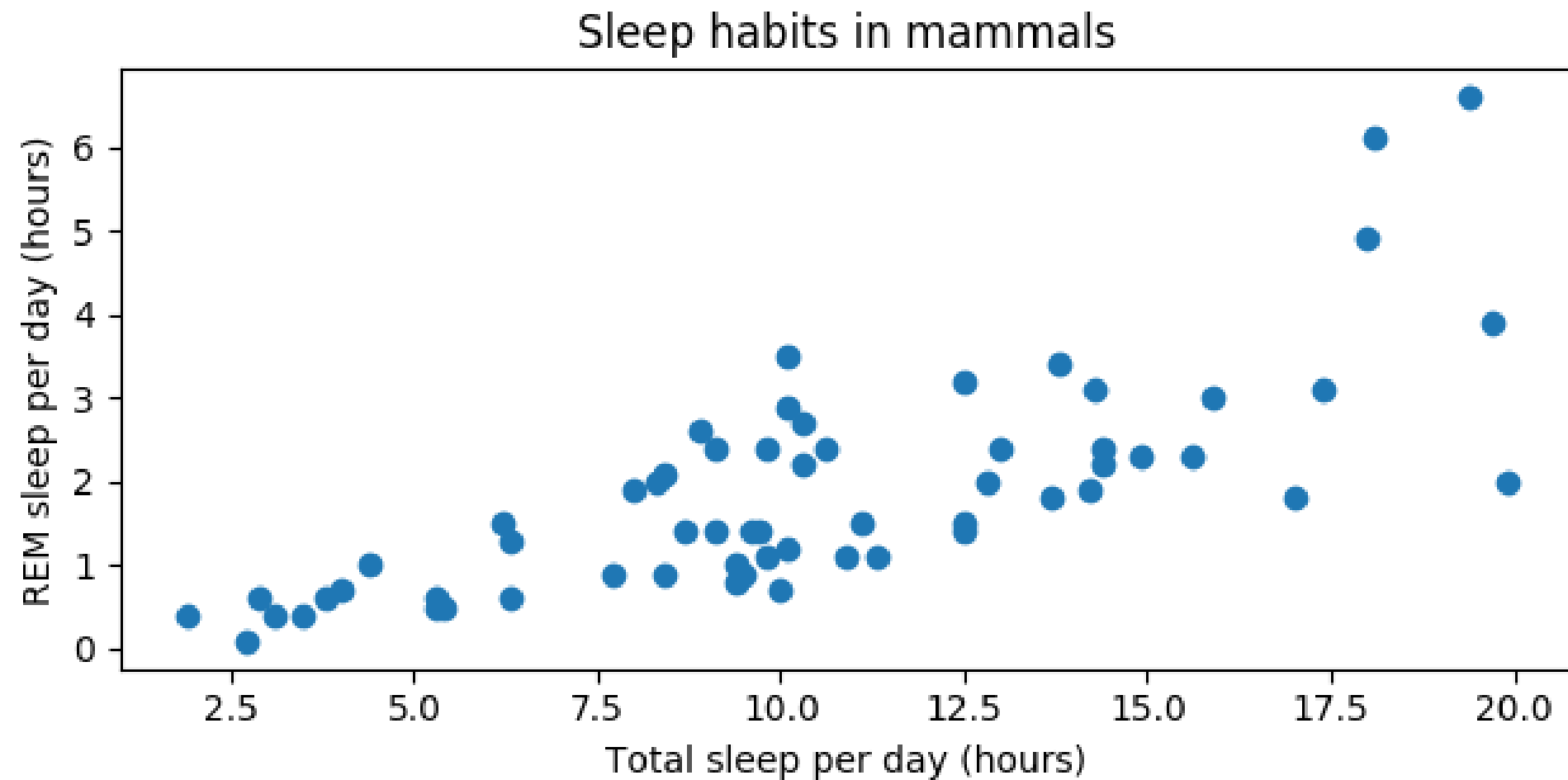
INTRODUCTION TO STATISTICS IN PYTHON



**Maggie Matsui**

Content Developer, DataCamp

# Relationships between two variables



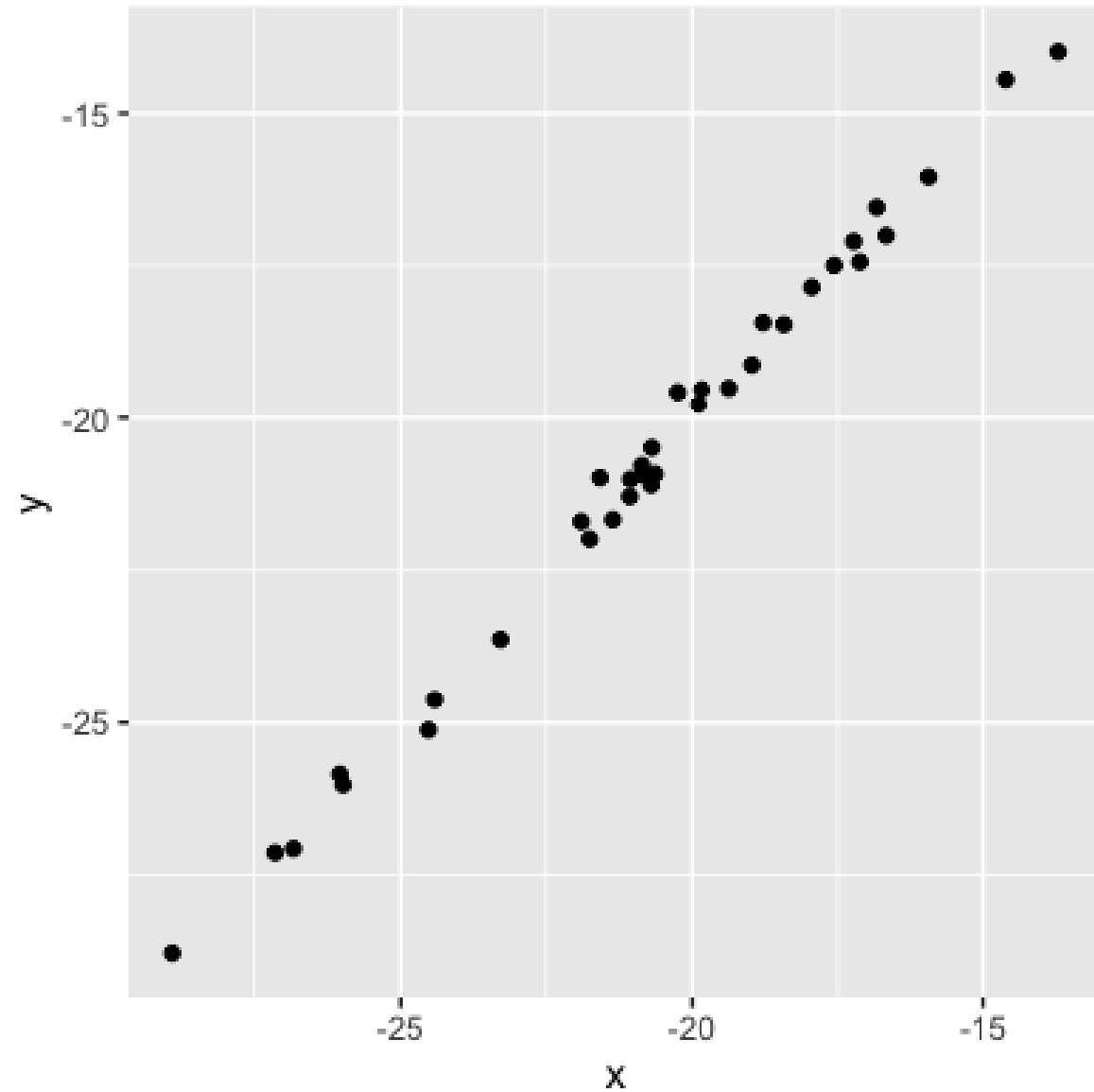
- $x$  = explanatory/independent variable
- $y$  = response/dependent variable

# Correlation coefficient

- Quantifies the linear relationship between two variables
- Number between -1 and 1
- Magnitude corresponds to strength of relationship
- Sign (+ or -) corresponds to direction of relationship

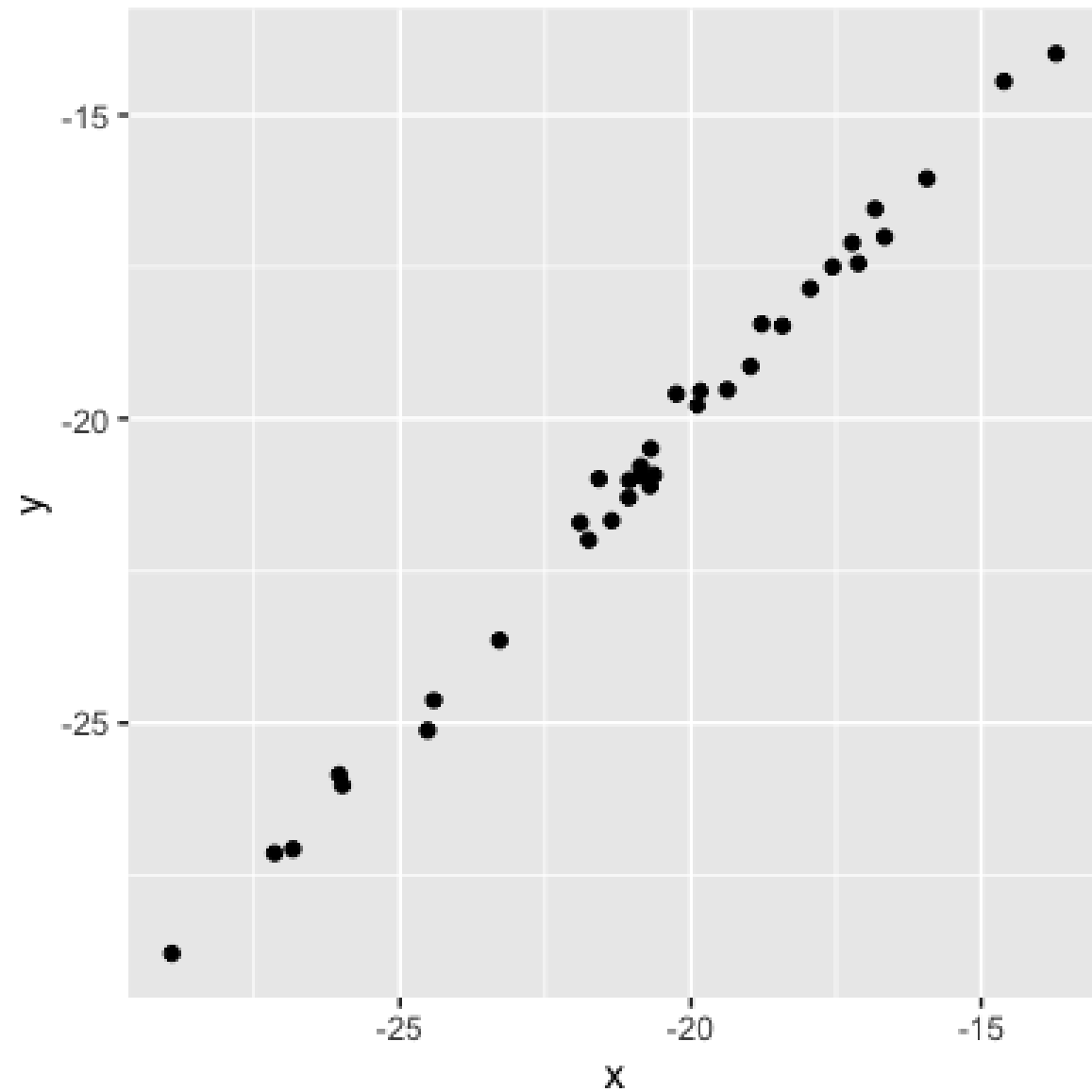
# Magnitude = strength of relationship

0.99 (very strong relationship)

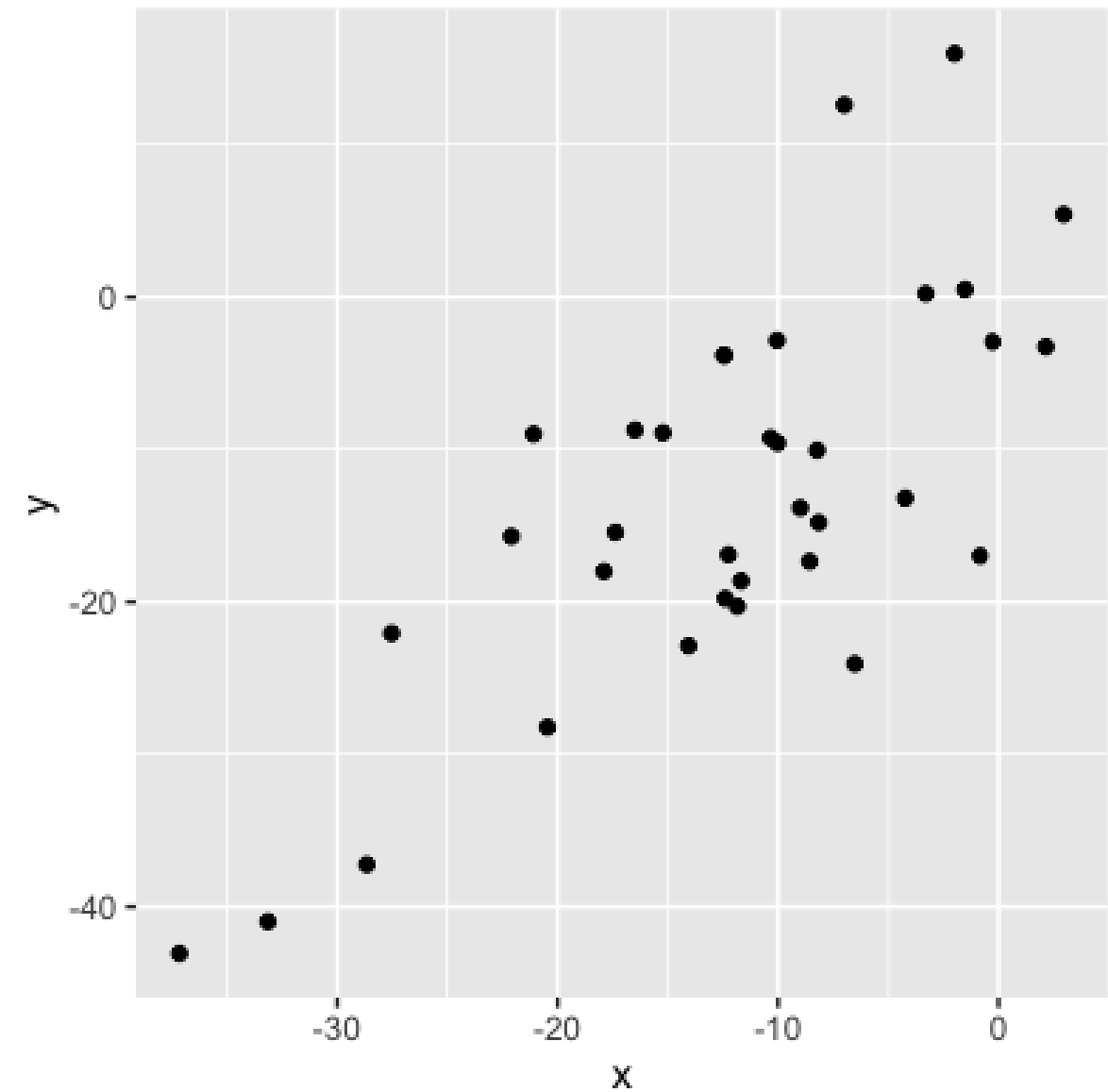


# Magnitude = strength of relationship

0.99 (very strong relationship)

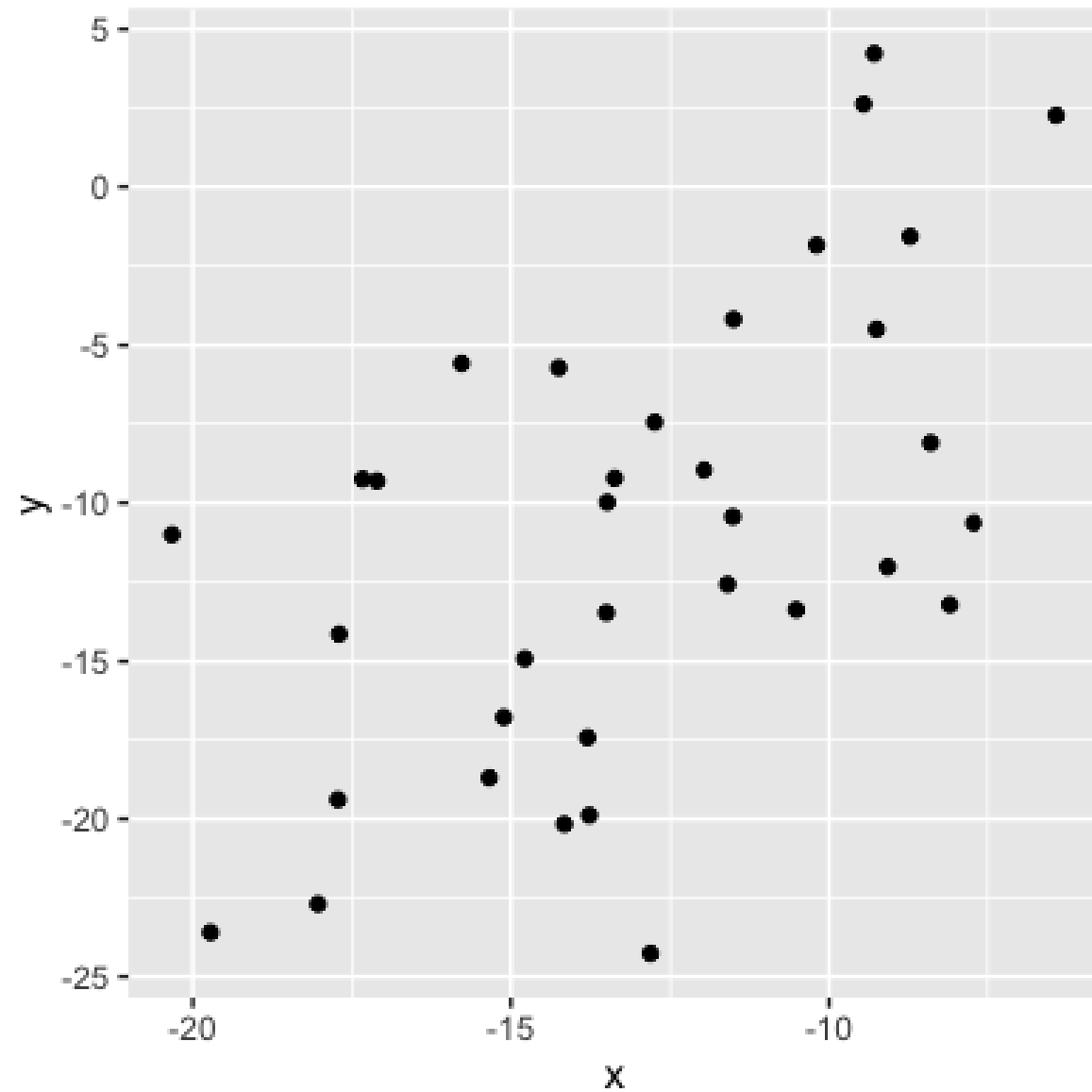


0.75 (strong relationship)



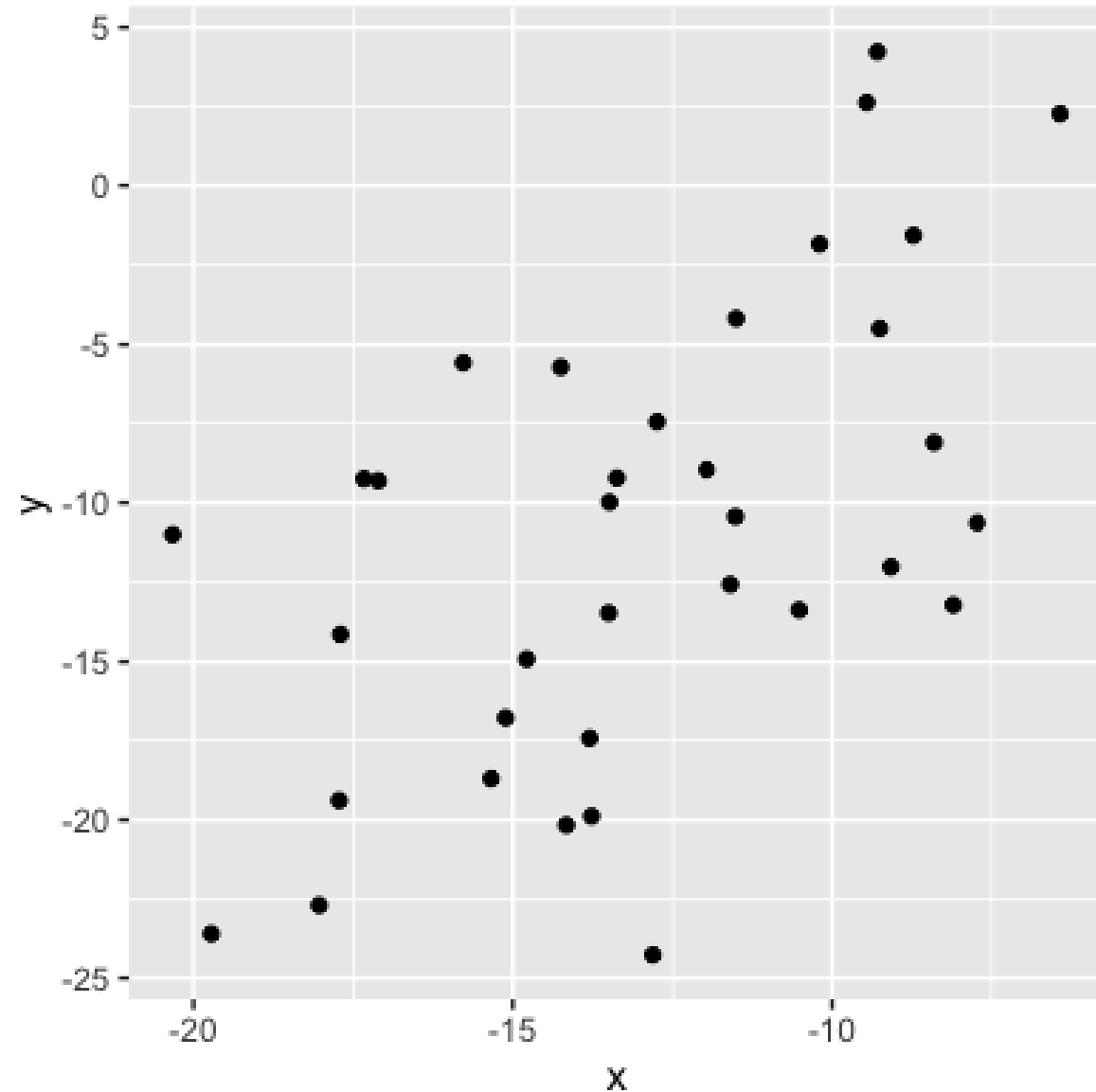
# Magnitude = strength of relationship

0.56 (moderate relationship)

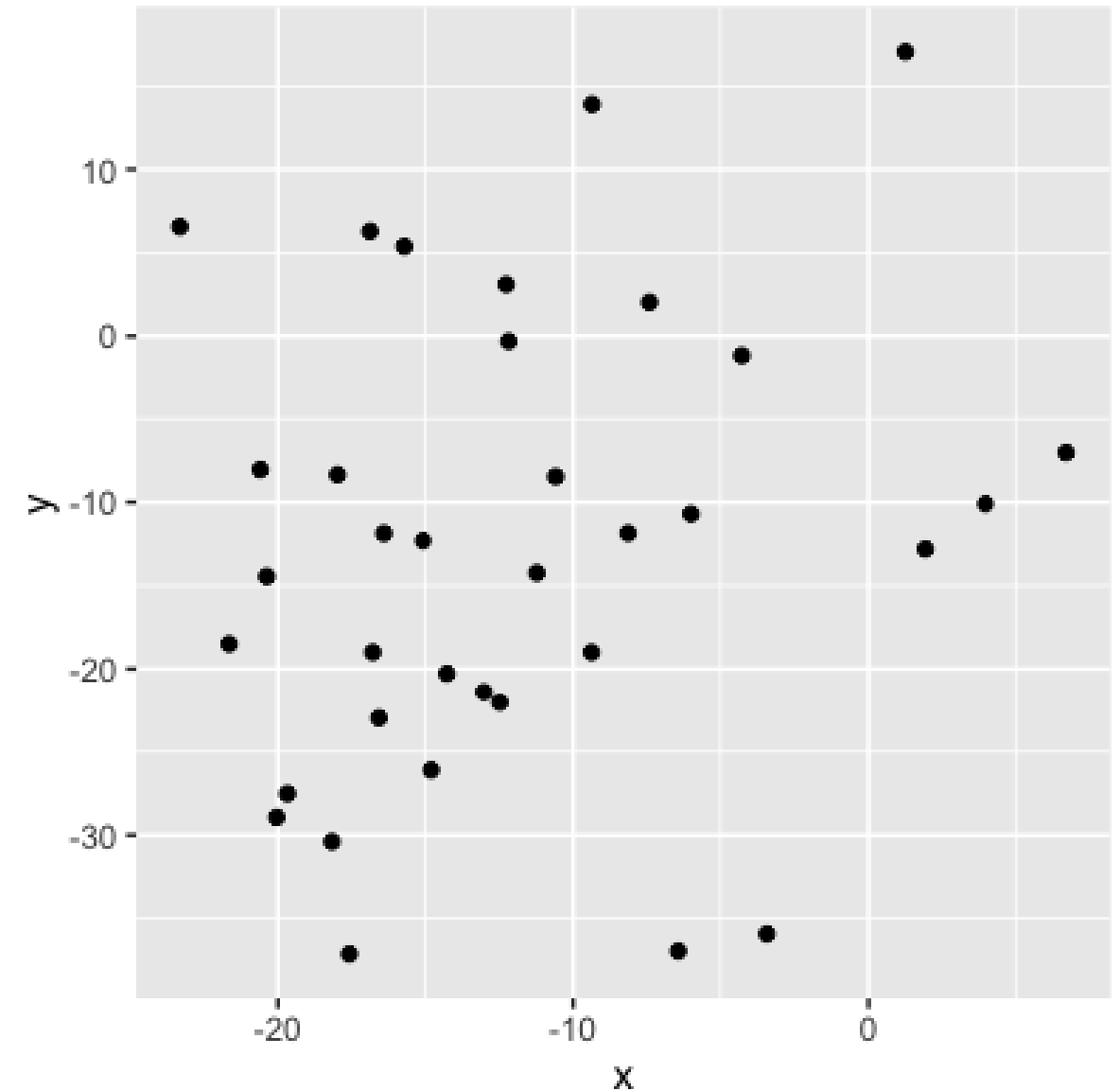


# Magnitude = strength of relationship

0.56 (moderate relationship)



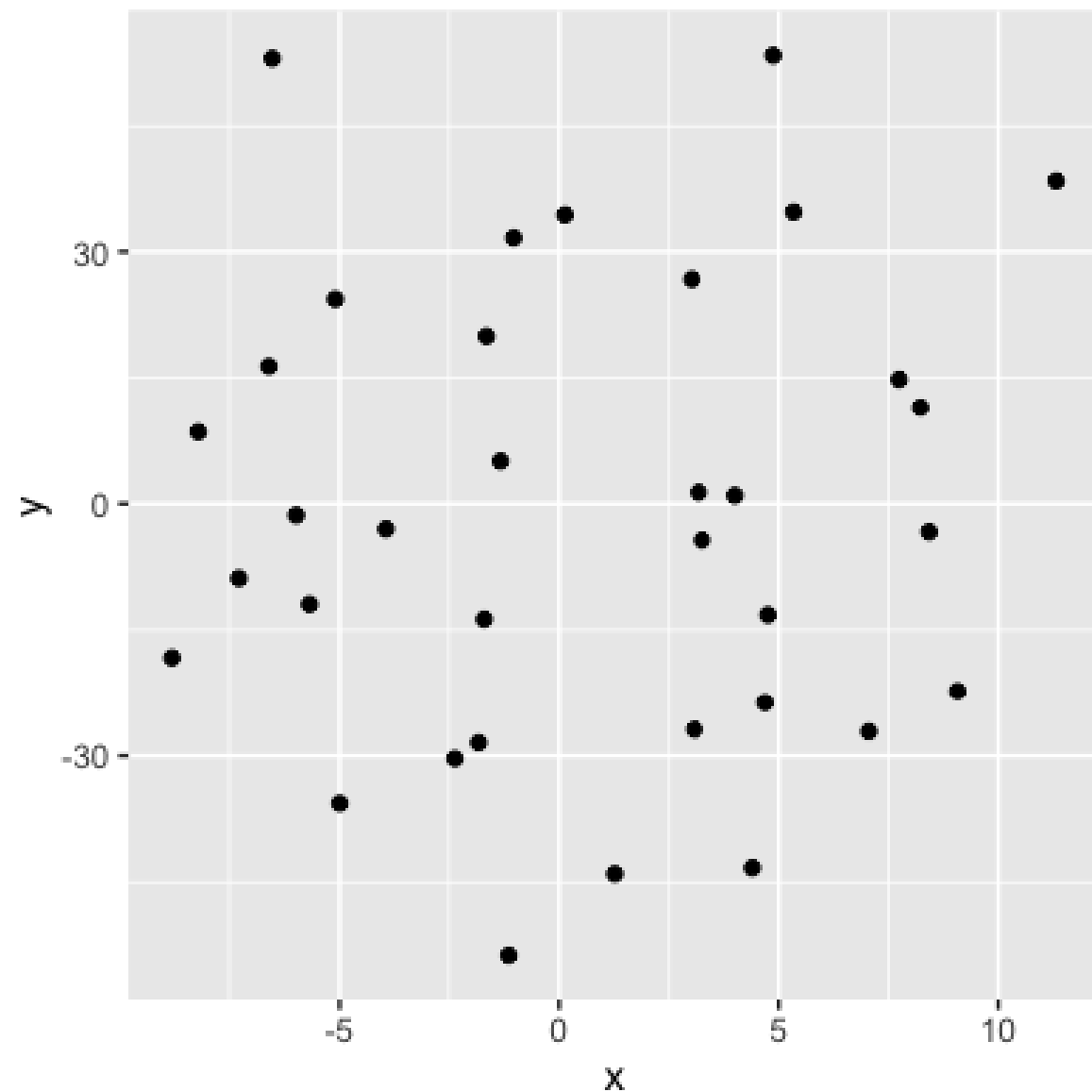
0.21 (weak relationship)



# Magnitude = strength of relationship

0.04 (no relationship)

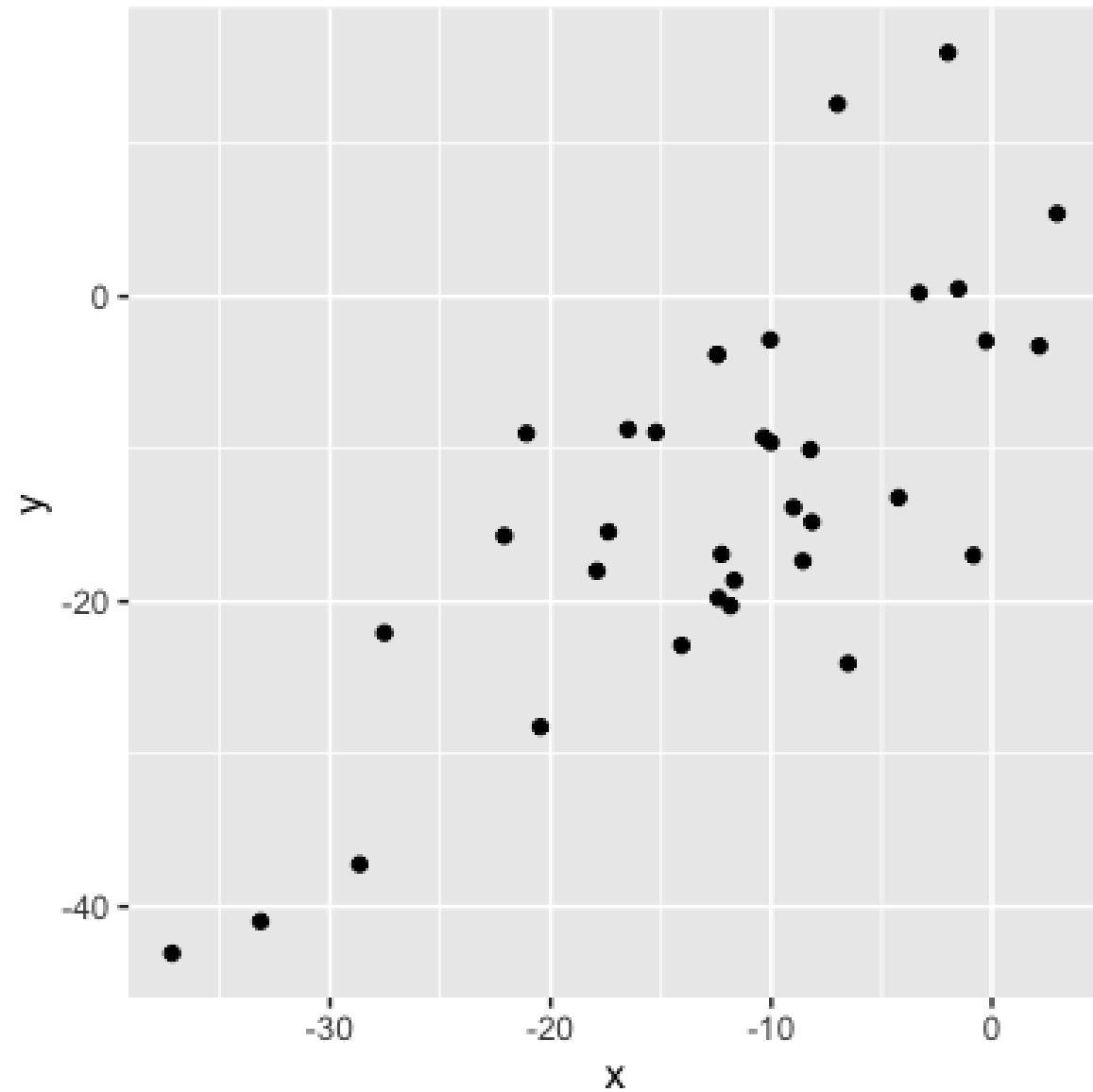
- Knowing the value of `x` doesn't tell us anything about `y`



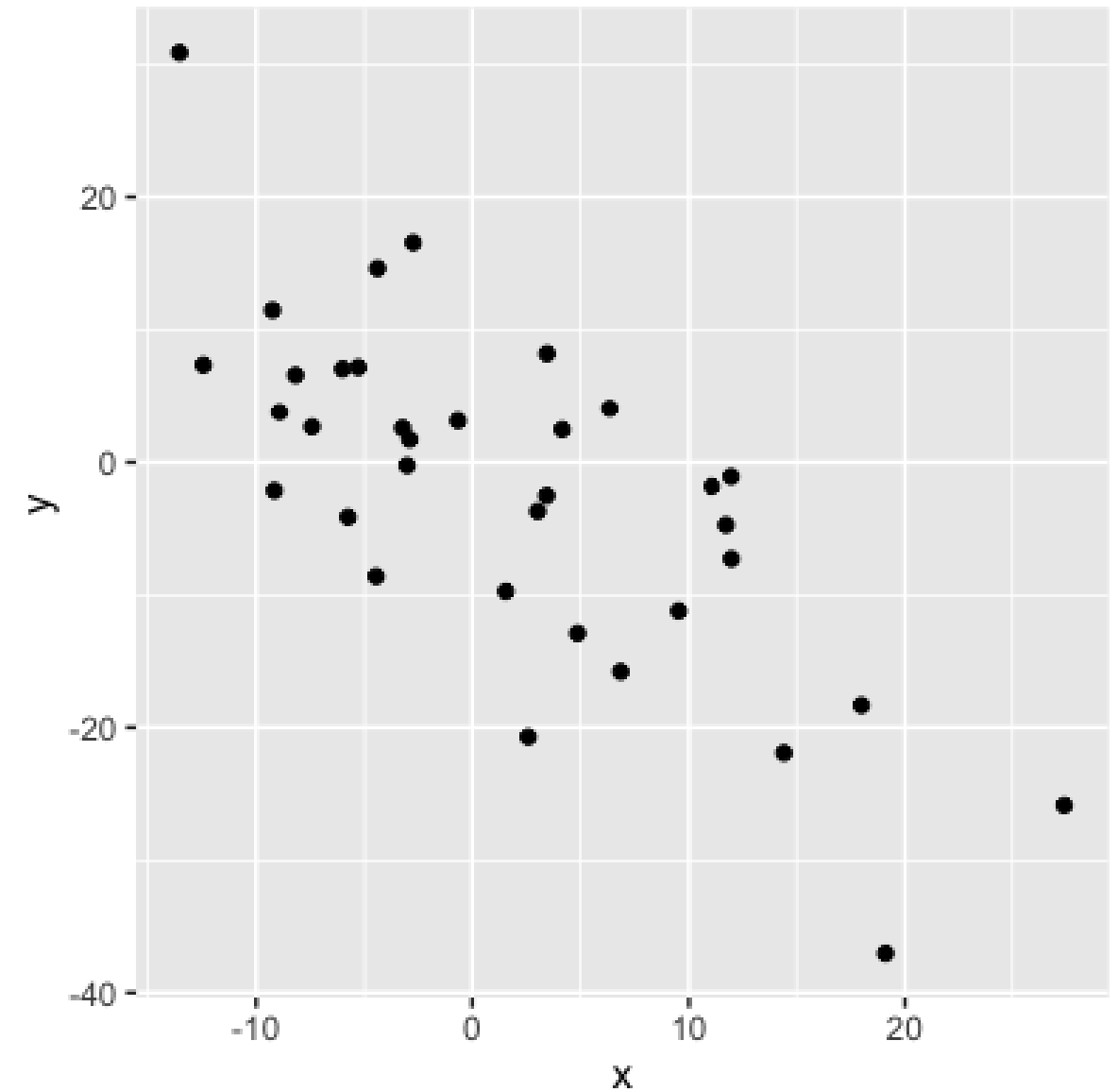


# Sign = direction

*0.75: as  $x$  increases,  $y$  increases*

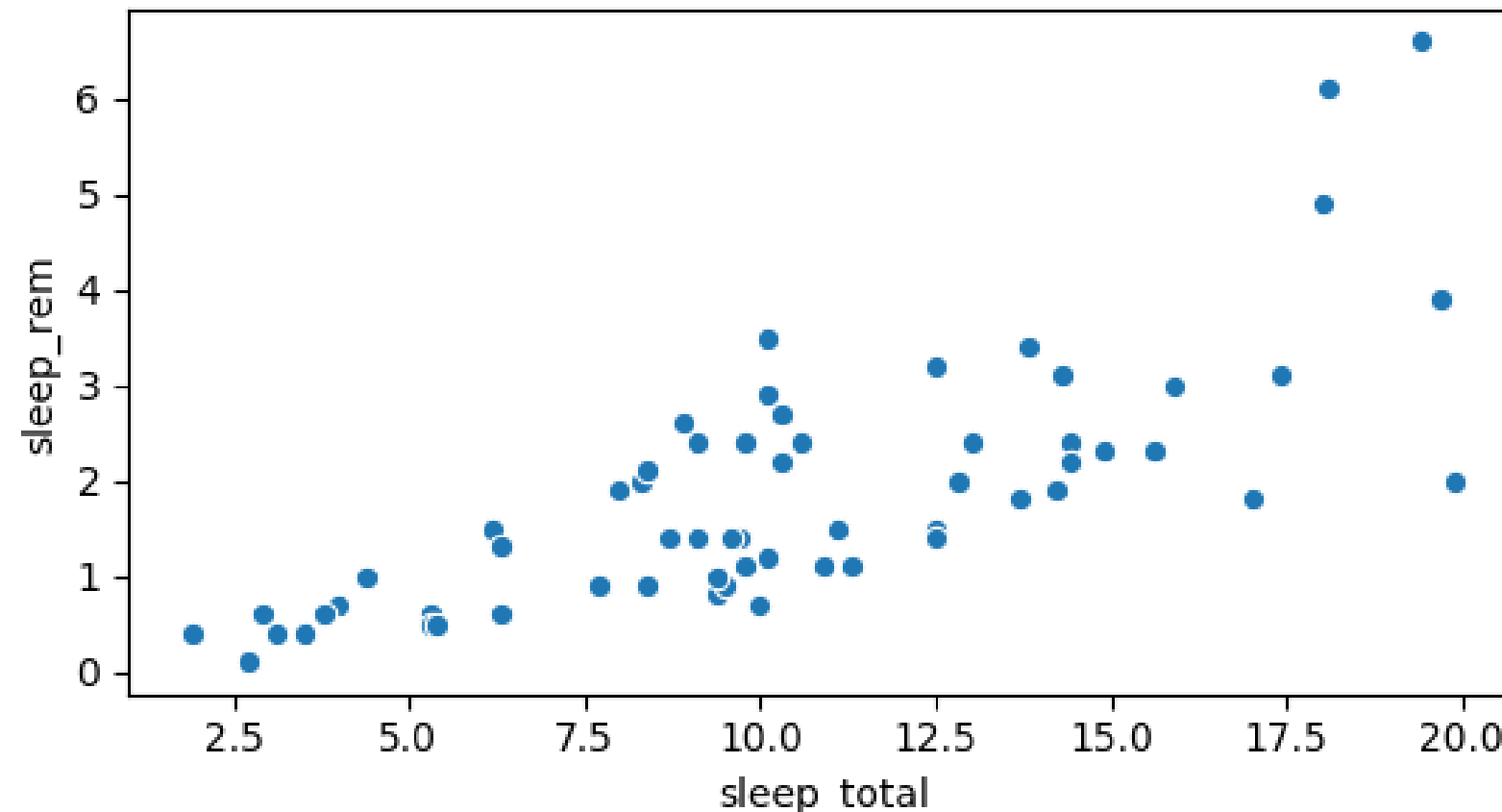


*-0.75: as  $x$  increases,  $y$  decreases*



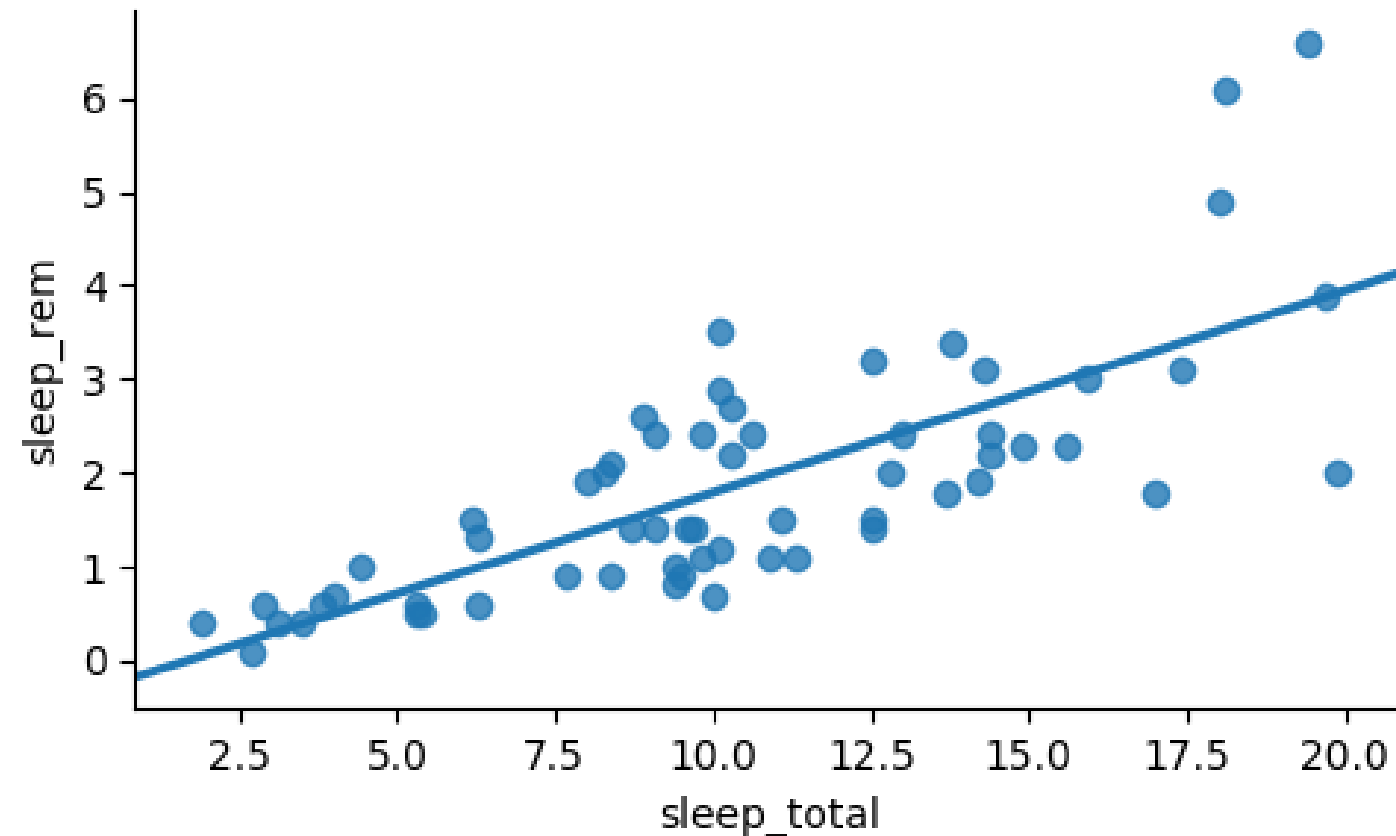
# Visualizing relationships

```
import seaborn as sns
sns.scatterplot(x="sleep_total", y="sleep_rem", data=msleep)
plt.show()
```



# Adding a trendline

```
import seaborn as sns
sns.lmplot(x="sleep_total", y="sleep_rem", data=msleep, ci=None)
plt.show()
```



# Computing correlation

```
msleep['sleep_total'].corr(msleep['sleep_rem'])
```

```
0.751755
```

```
msleep['sleep_rem'].corr(msleep['sleep_total'])
```

```
0.751755
```

# Many ways to calculate correlation

- Used in this course: Pearson product-moment correlation ( $r$ )
  - Most common
  - $\bar{x}$  = mean of  $x$
  - $\sigma_x$  = standard deviation of  $x$

$$r = \sum_{i=1}^n \frac{(x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \times \sigma_y}$$

- Variations on this formula:
  - Kendall's tau
  - Spearman's rho

# Let's practice!

INTRODUCTION TO STATISTICS IN PYTHON

# Correlation caveats

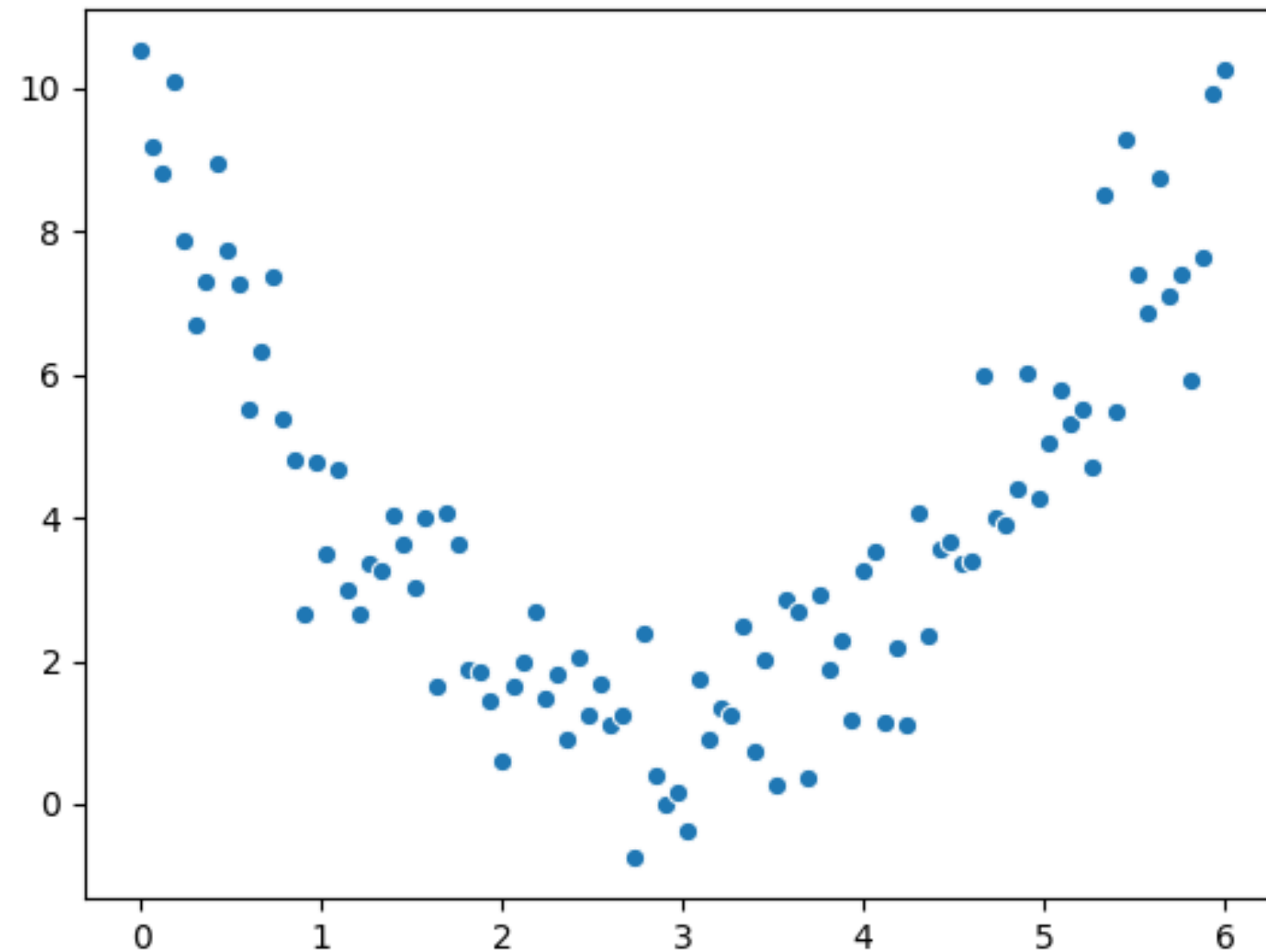
INTRODUCTION TO STATISTICS IN PYTHON



**Maggie Matsui**

Content Developer, DataCamp

# Non-linear relationships

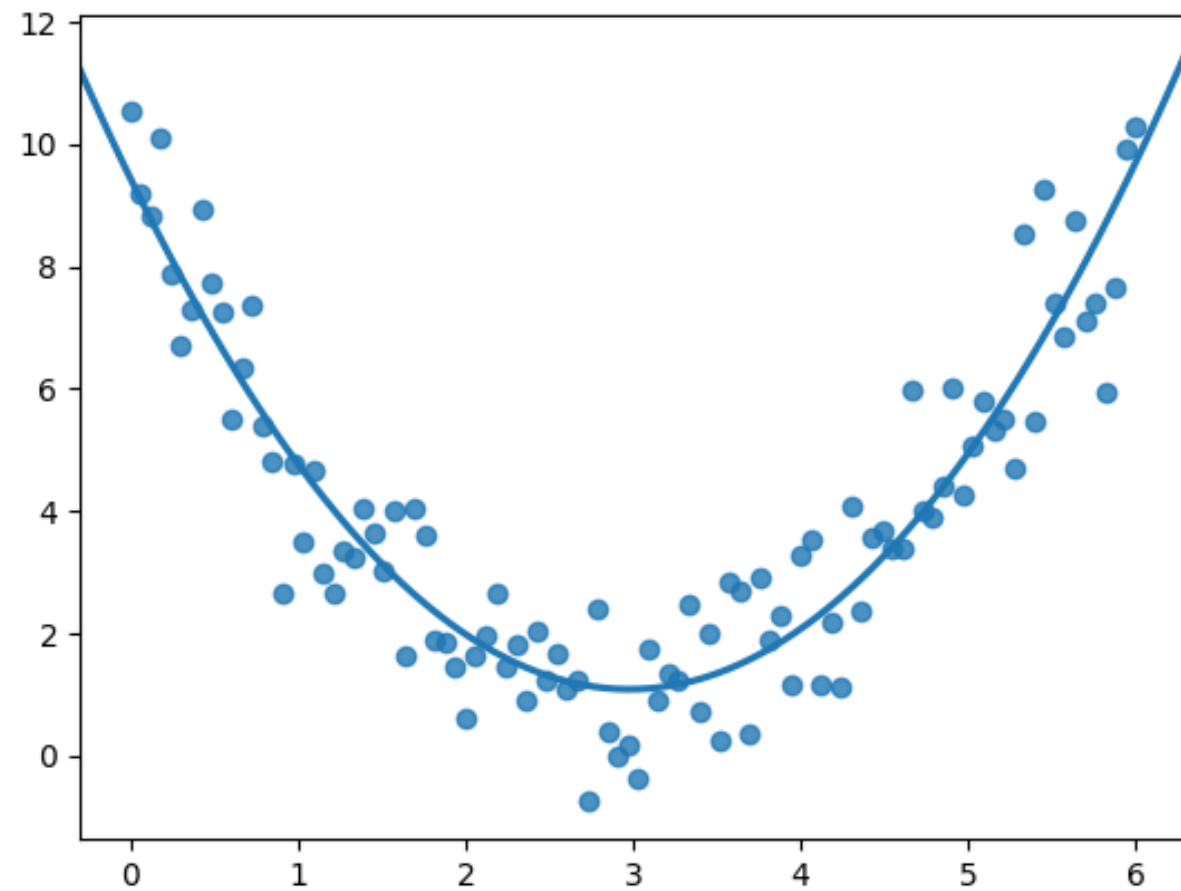


$$r = 0.18$$

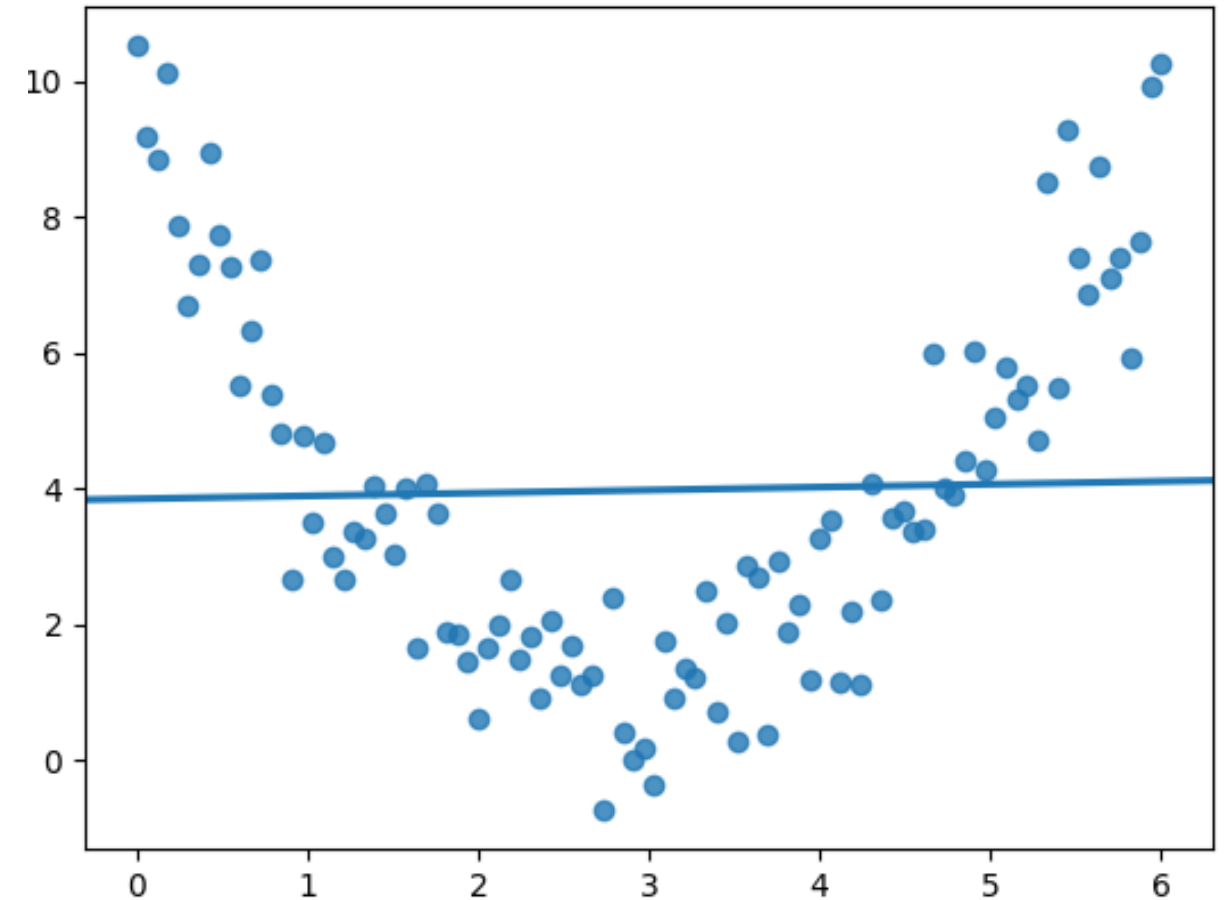


# Non-linear relationships

*What we see:*



*What the correlation coefficient sees:*



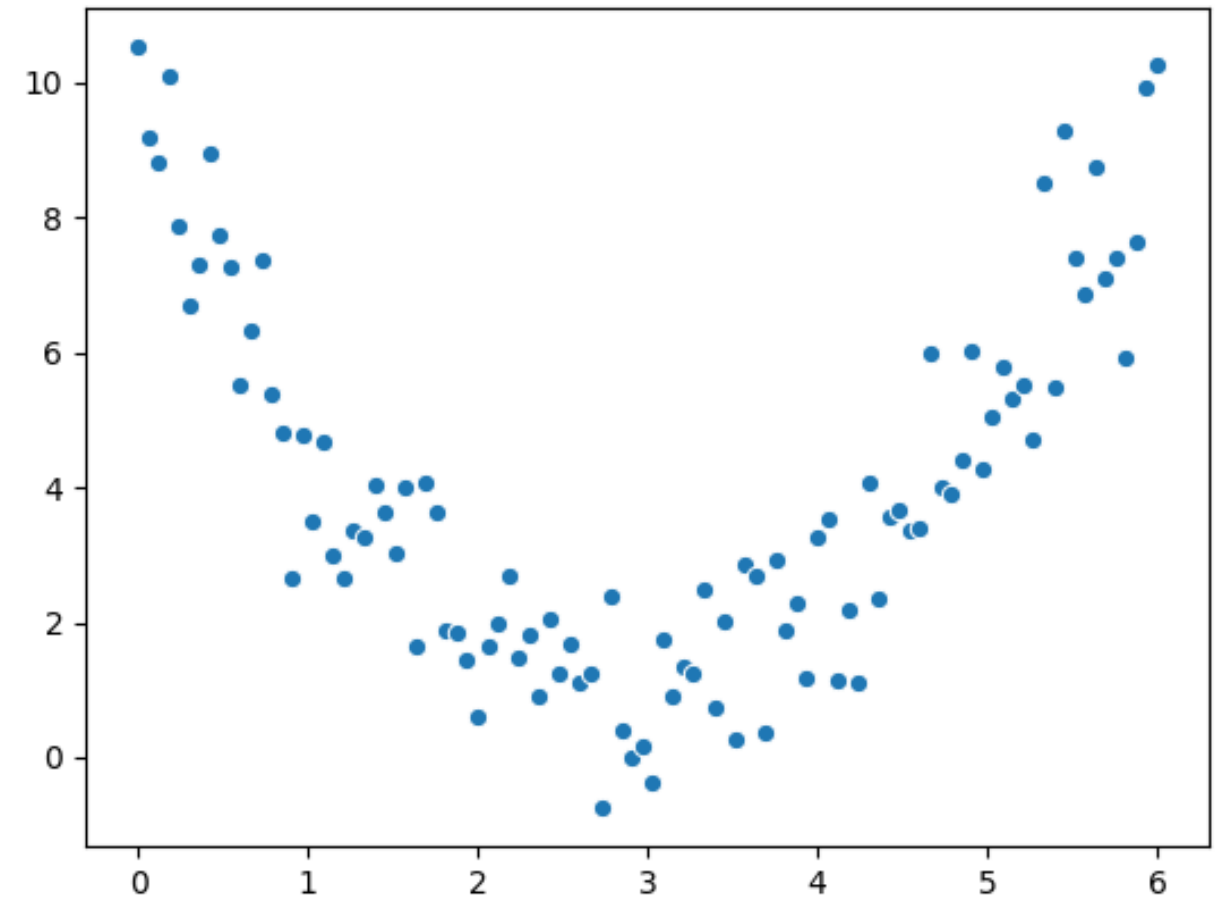
# Correlation only accounts for linear relationships

Correlation shouldn't be used blindly

```
df['x'].corr(df['y'])
```

```
0.081094
```

*Always* visualize your data

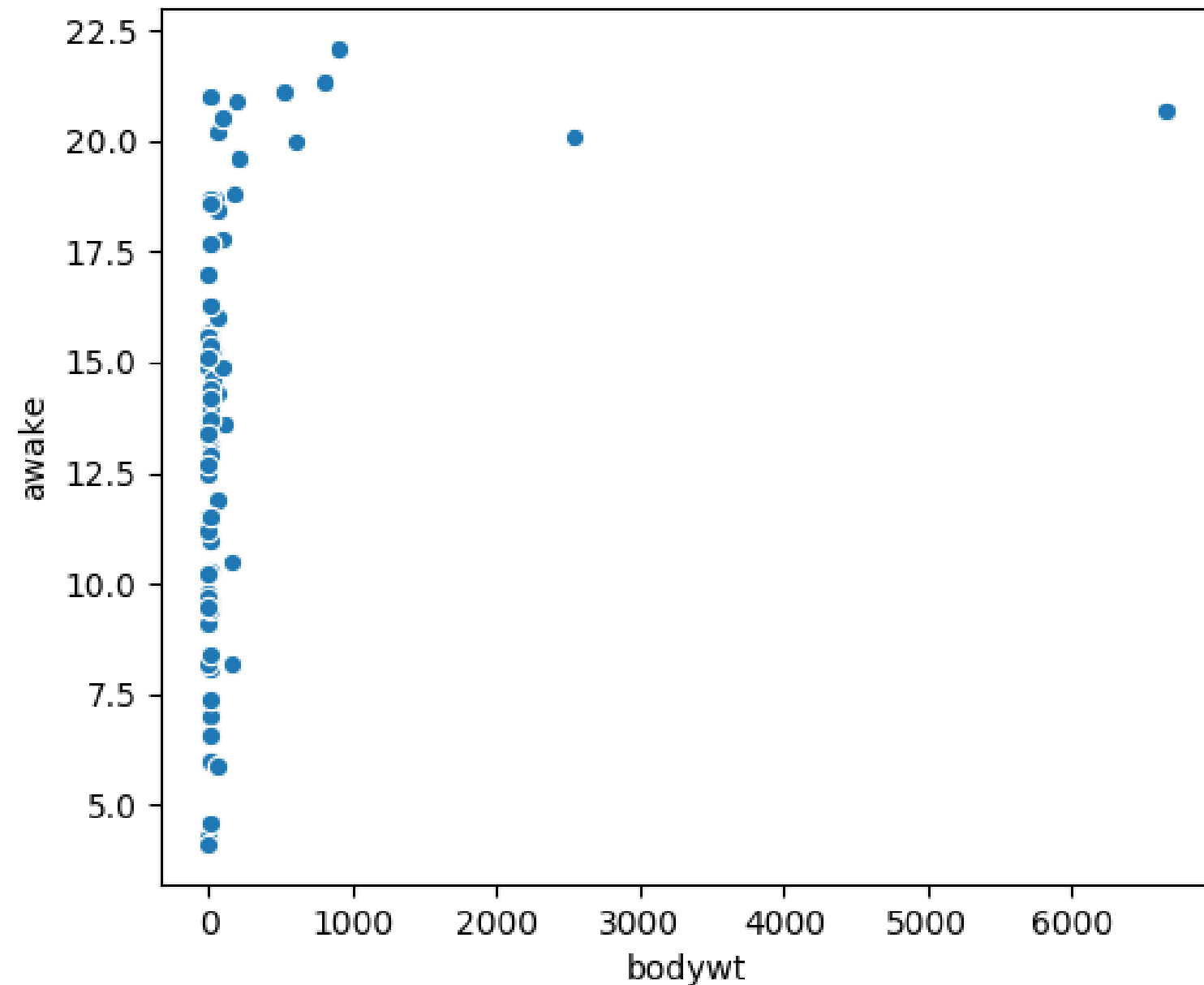


# Mammal sleep data

```
print(msleep)
```

|    | name                | genus      | vore  | order        | ... | sleep_cycle | awake | brainwt | bodywt  |
|----|---------------------|------------|-------|--------------|-----|-------------|-------|---------|---------|
| 1  | Cheetah             | Acinonyx   | carni | Carnivora    | ... | NaN         | 11.9  | NaN     | 50.000  |
| 2  | Owl monkey          | Aotus      | omni  | Primates     | ... | NaN         | 7.0   | 0.01550 | 0.480   |
| 3  | Mountain beaver     | Aplodontia | herbi | Rodentia     | ... | NaN         | 9.6   | NaN     | 1.350   |
| 4  | Greater short-ta... | Blarina    | omni  | Soricomorpha | ... | 0.133333    | 9.1   | 0.00029 | 0.019   |
| 5  | Cow                 | Bos        | herbi | Artiodactyla | ... | 0.666667    | 20.0  | 0.42300 | 600.000 |
| .. | ...                 | ...        | ...   | ...          | ... | ...         | ...   | ...     | ...     |
| 79 | Tree shrew          | Tupaia     | omni  | Scandentia   | ... | 0.233333    | 15.1  | 0.00250 | 0.104   |
| 80 | Bottle-nosed do...  | Tursiops   | carni | Cetacea      | ... | NaN         | 18.8  | NaN     | 173.330 |
| 81 | Genet               | Genetta    | carni | Carnivora    | ... | NaN         | 17.7  | 0.01750 | 2.000   |
| 82 | Arctic fox          | Vulpes     | carni | Carnivora    | ... | NaN         | 11.5  | 0.04450 | 3.380   |
| 83 | Red fox             | Vulpes     | carni | Carnivora    | ... | 0.350000    | 14.2  | 0.05040 | 4.230   |

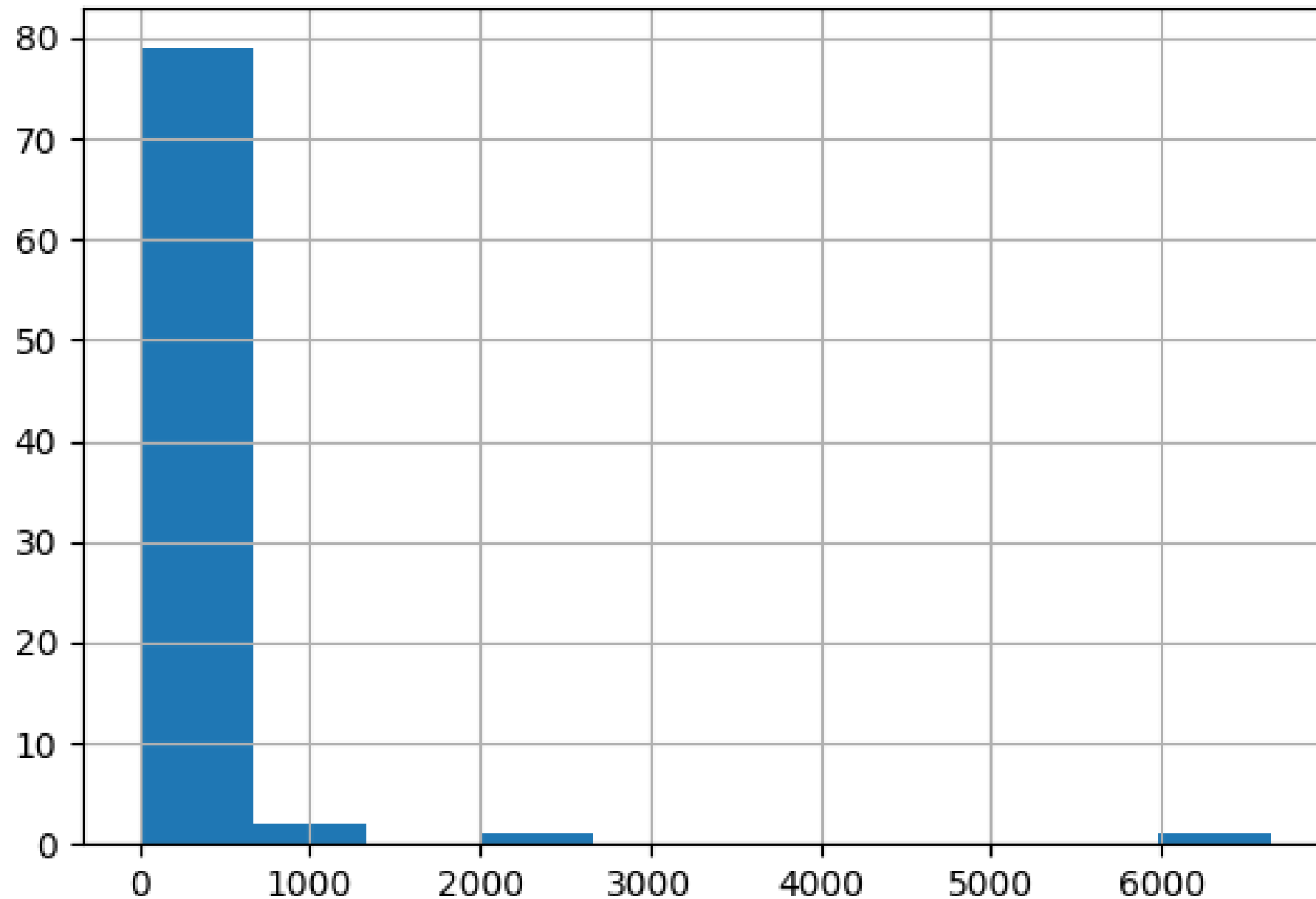
# Body weight vs. awake time



```
msleep['bodywt'].corr(msleep['awake'])
```

```
0.3119801
```

# Distribution of body weight



# Log transformation

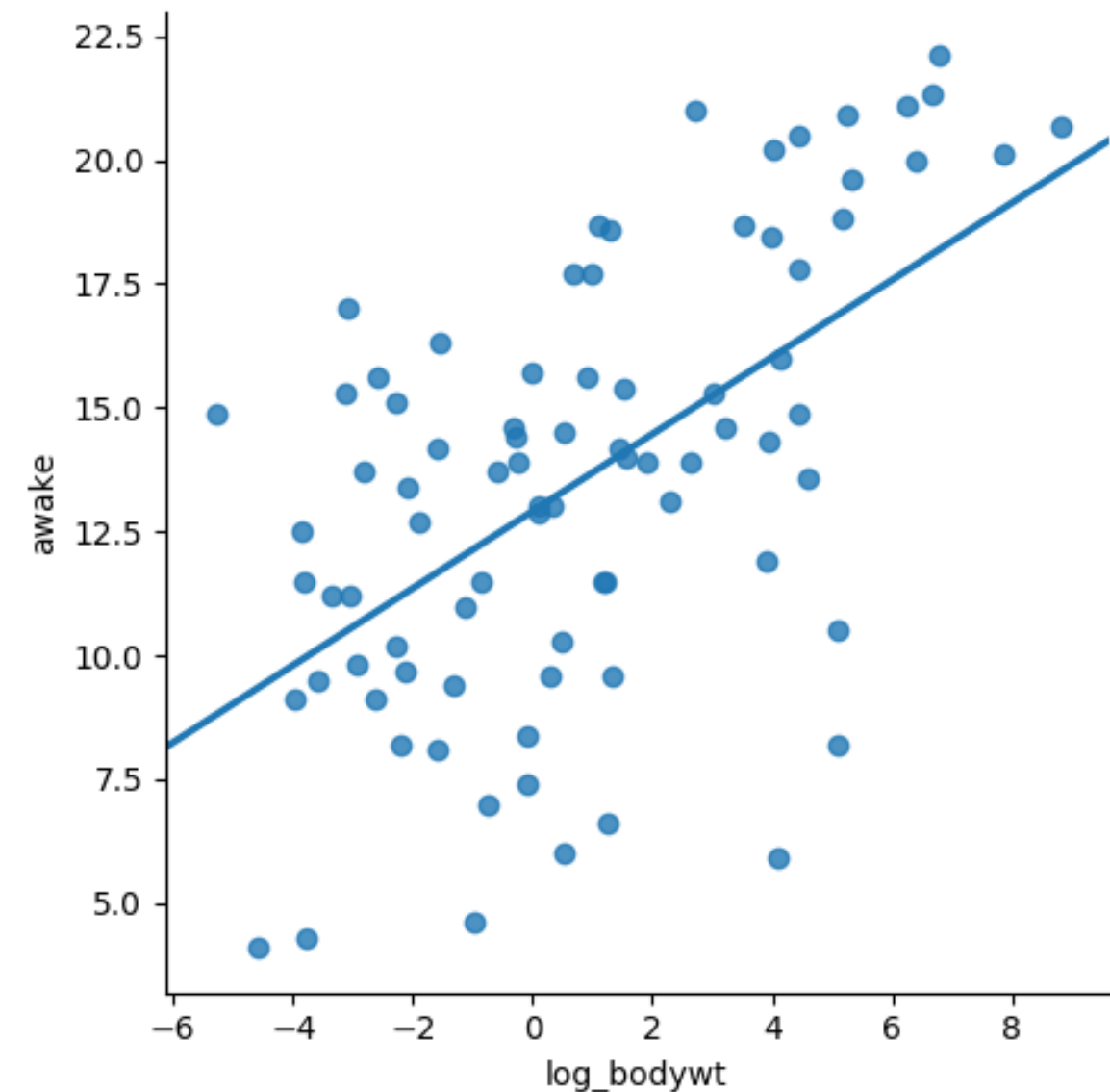
```
msleep['log_bodywt'] = np.log(msleep['bodywt'])

sns.lmplot(x='log_bodywt',
           y='awake',
           data=msleep,
           ci=None)

plt.show()
```

```
msleep['log_bodywt'].corr(msleep['awake'])
```

```
0.5687943
```



# Other transformations

- Log transformation (  $\log(x)$  )
- Square root transformation (  $\sqrt{x}$  )
- Reciprocal transformation (  $1 / x$  )
- Combinations of these, e.g.:
  - $\log(x)$  and  $\log(y)$
  - $\sqrt{x}$  and  $1 / y$

# Why use a transformation?

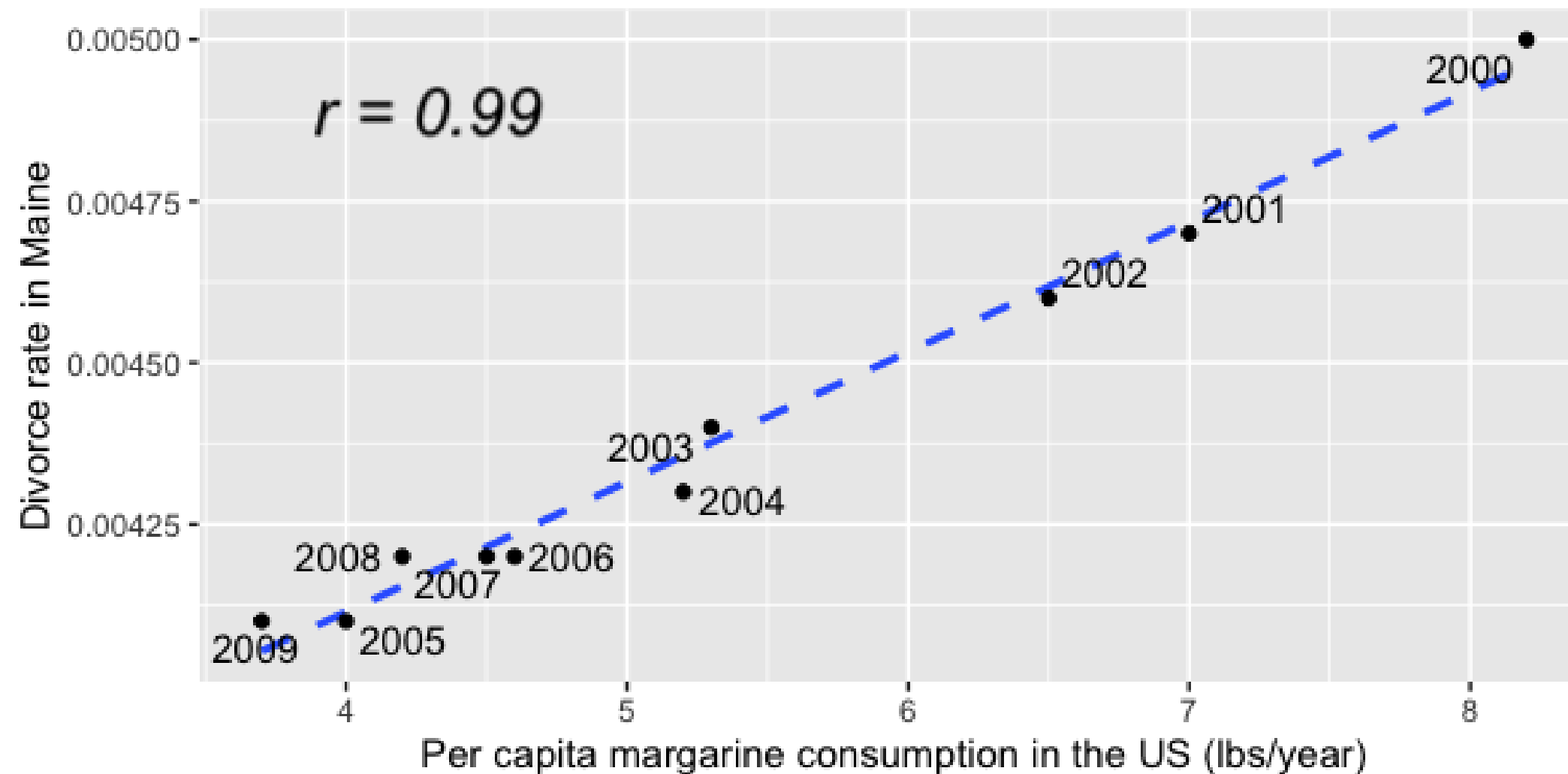
- Certain statistical methods rely on variables having a linear relationship
  - Correlation coefficient
  - Linear regression

## Introduction to Linear Modeling in Python

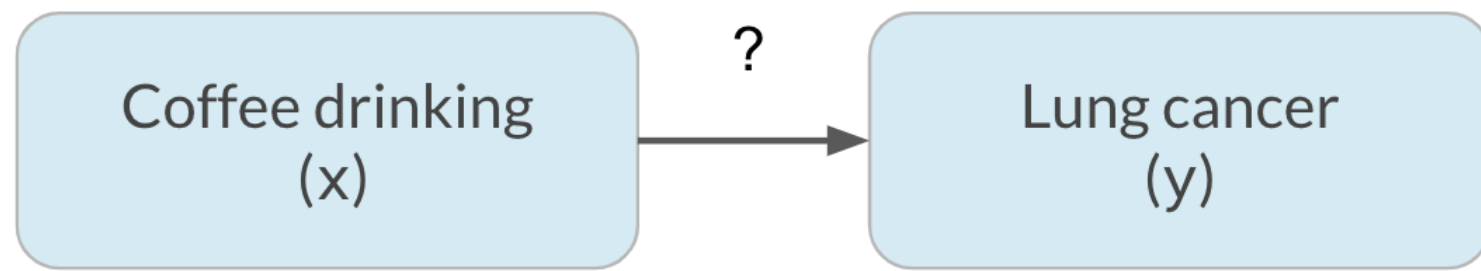


# Correlation does not imply causation

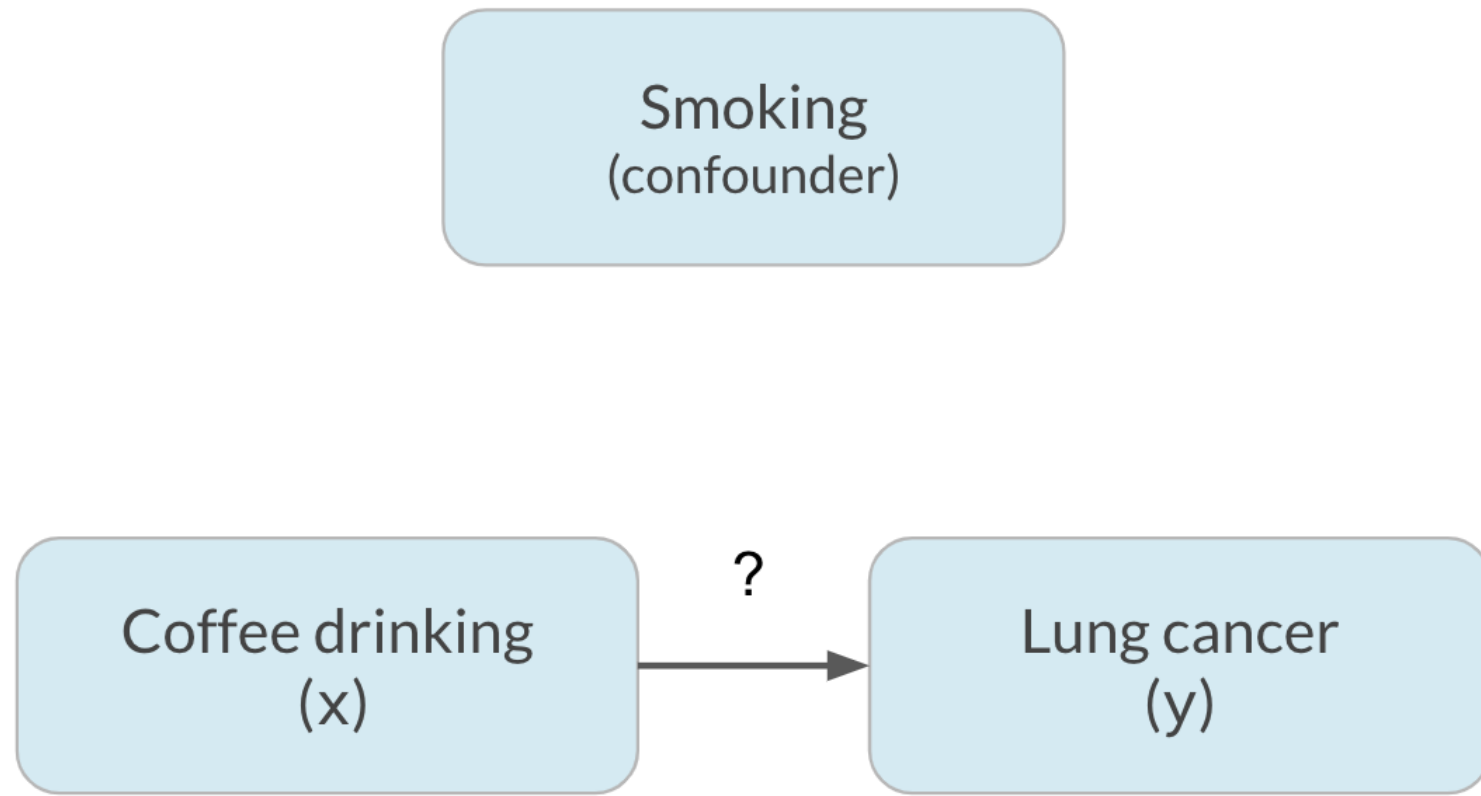
$x$  is correlated with  $y$  does not mean  $x$  causes  $y$



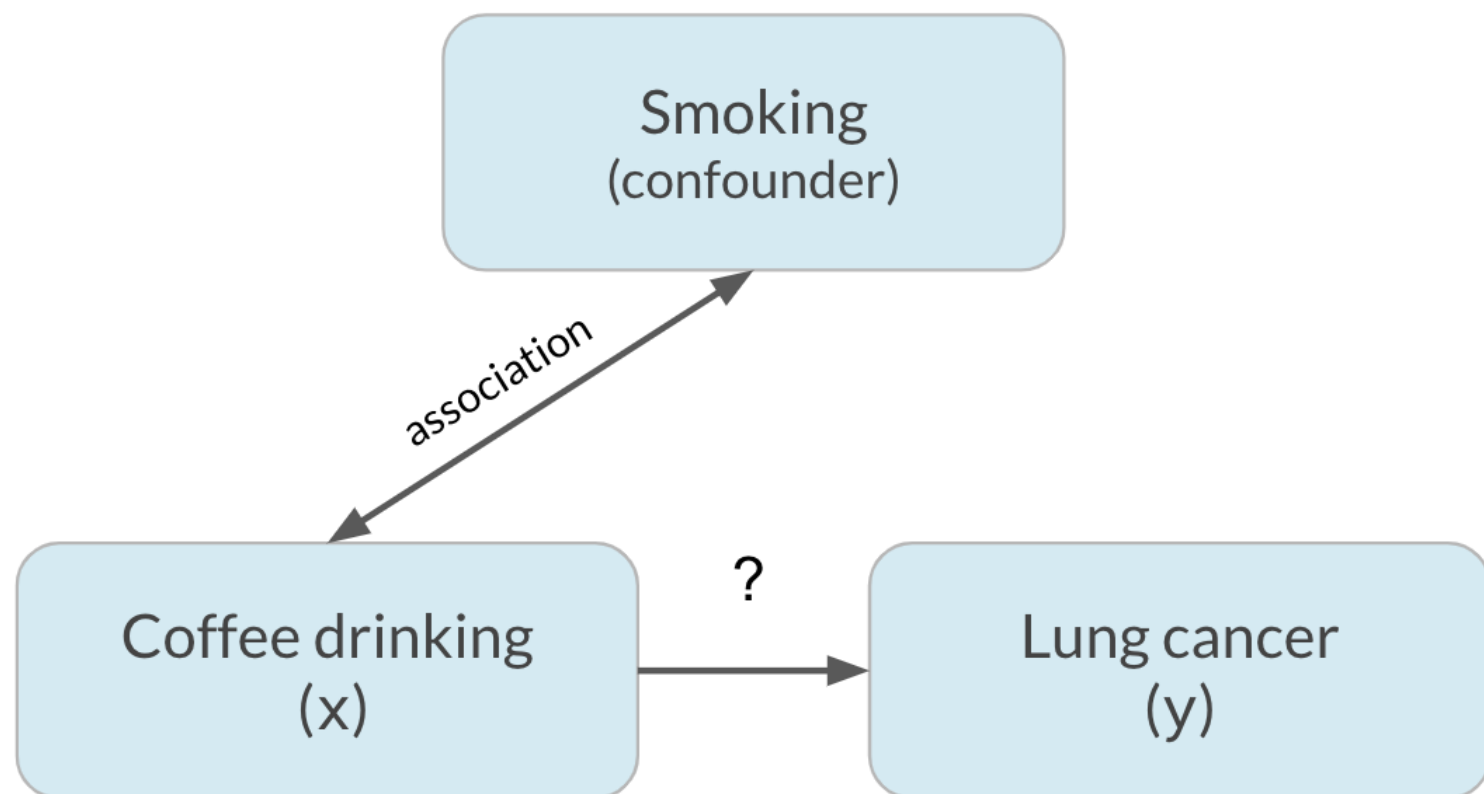
# Confounding



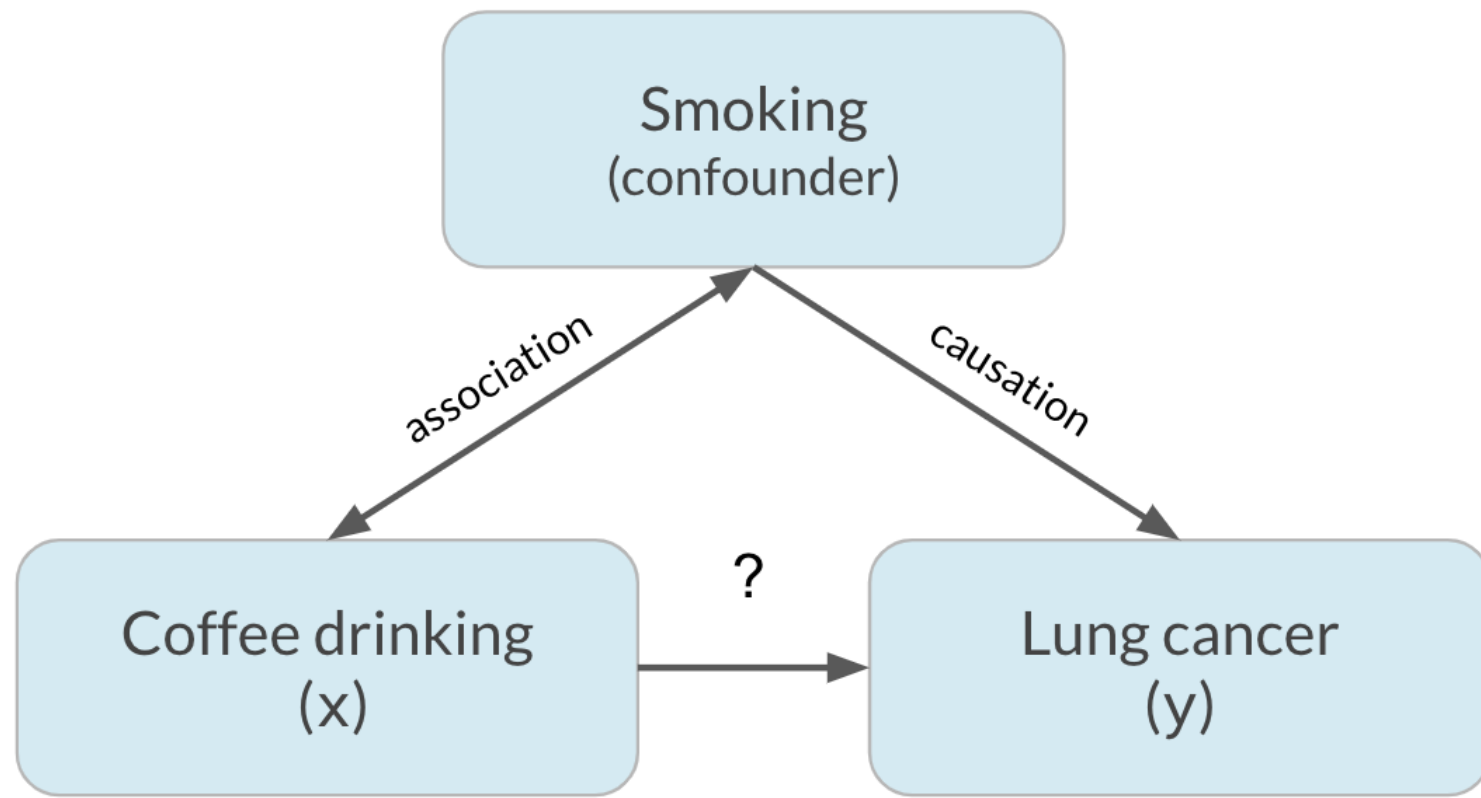
# Confounding



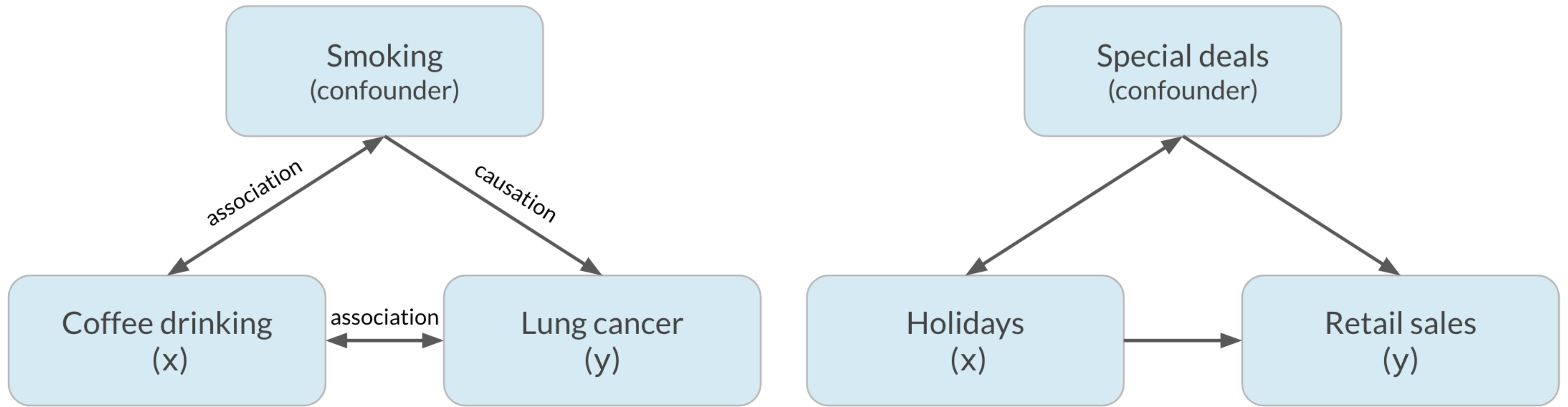
# Confounding



# Confounding



# Confounding

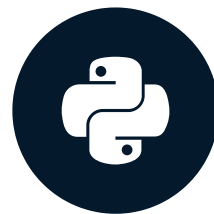


# Let's practice!

INTRODUCTION TO STATISTICS IN PYTHON

# Design of experiments

INTRODUCTION TO STATISTICS IN PYTHON



**Maggie Matsui**

Content Developer, DataCamp



# Vocabulary

Experiment aims to answer: *What is the effect of the treatment on the response?*

- Treatment: explanatory/independent variable
- Response: response/dependent variable

E.g.: *What is the effect of an advertisement on the number of products purchased?*

- Treatment: advertisement
- Response: number of products purchased

# Controlled experiments

- Participants are assigned by researchers to either treatment group or control group
  - Treatment group sees advertisement
  - Control group does not
- Groups should be comparable so that causation can be inferred
- If groups are not comparable, this could lead to confounding (bias)
  - Treatment group average age: 25
  - Control group average age: 50
  - Age is a potential confounder

# The gold standard of experiments will use...

- Randomized controlled trial
  - Participants are assigned to treatment/control *randomly*, not based on any other characteristics
  - Choosing randomly helps ensure that groups are comparable
- Placebo
  - Resembles treatment, but has no effect
  - Participants will not know which group they're in
  - In clinical trials, a sugar pill ensures that the effect of the drug is actually due to the drug itself and not the idea of receiving the drug

# The gold standard of experiments will use...

- Double-blind trial
  - Person administering the treatment/running the study doesn't know whether the treatment is real or a placebo
  - Prevents bias in the response and/or analysis of results

*Fewer opportunities for bias = more reliable conclusion about causation*

# Observational studies

- Participants are not assigned randomly to groups
  - Participants assign themselves, usually based on pre-existing characteristics
- Many research questions are not conducive to a controlled experiment
  - You can't force someone to smoke or have a disease
  - You can't make someone have certain past behavior
- Establish association, not causation
  - Effects can be confounded by factors that got certain people into the control or treatment group
  - There are ways to control for confounders to get more reliable conclusions about association

# Longitudinal vs. cross-sectional studies

## Longitudinal study

- Participants are followed over a period of time to examine effect of treatment on response
- Effect of age on height is not confounded by generation
- More expensive, results take longer

## Cross-sectional study

- Data on participants is collected from a single snapshot in time
- Effect of age on height is confounded by generation
- Cheaper, faster, more convenient

# Let's practice!

INTRODUCTION TO STATISTICS IN PYTHON

# Congratulations!

INTRODUCTION TO STATISTICS IN PYTHON



**Maggie Matsui**

Content Developer, DataCamp



# Overview

## Chapter 1

- What is statistics?
- Measures of center
- Measures of spread

## Chapter 3

- Normal distribution
- Central limit theorem
- Poisson distribution

## Chapter 2

- Measuring chance
- Probability distributions
- Binomial distribution

## Chapter 4

- Correlation
- Controlled experiments
- Observational studies

# Build on your skills

- [Introduction to Linear Modeling in Python](#)

# Congratulations!

INTRODUCTION TO STATISTICS IN PYTHON