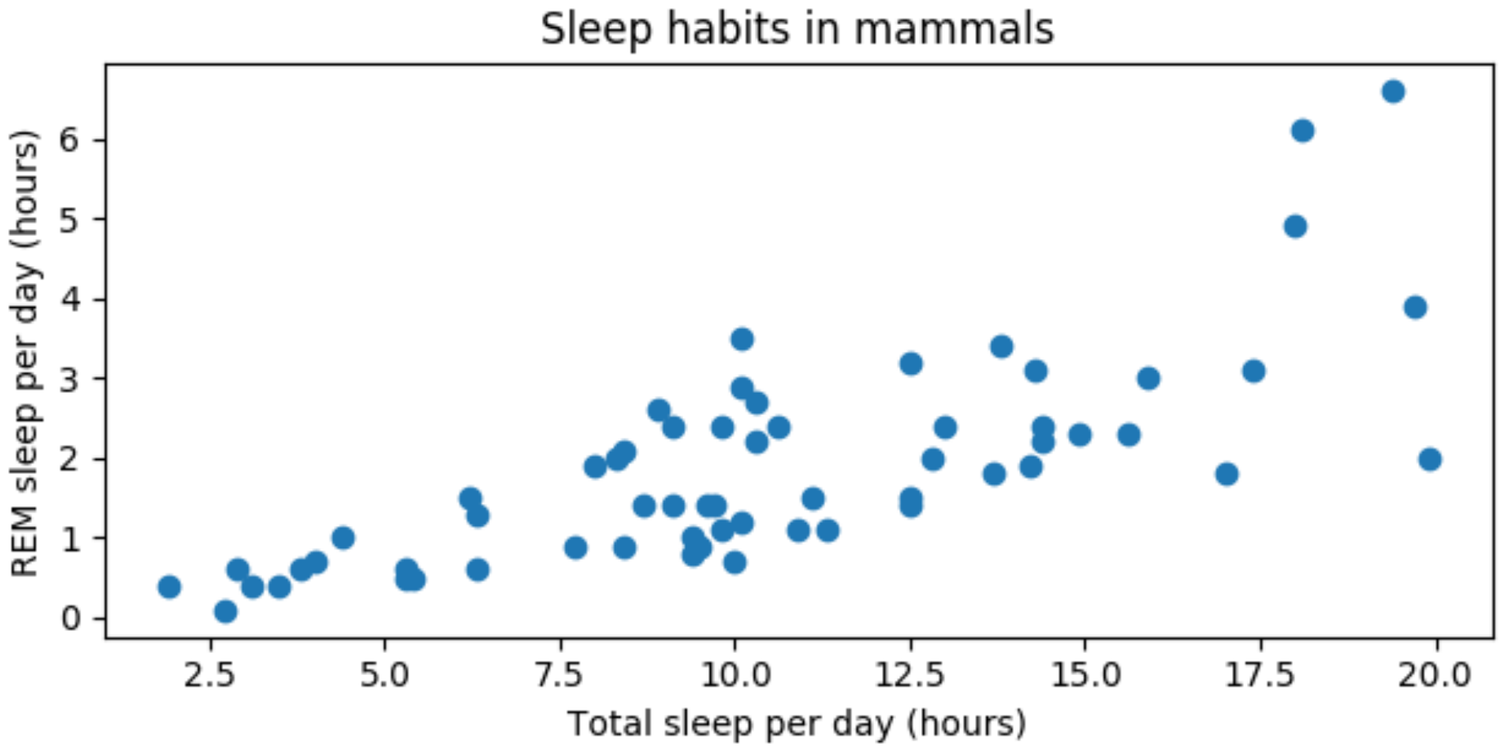
Correlation

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

datacamp

Relationships between two variables



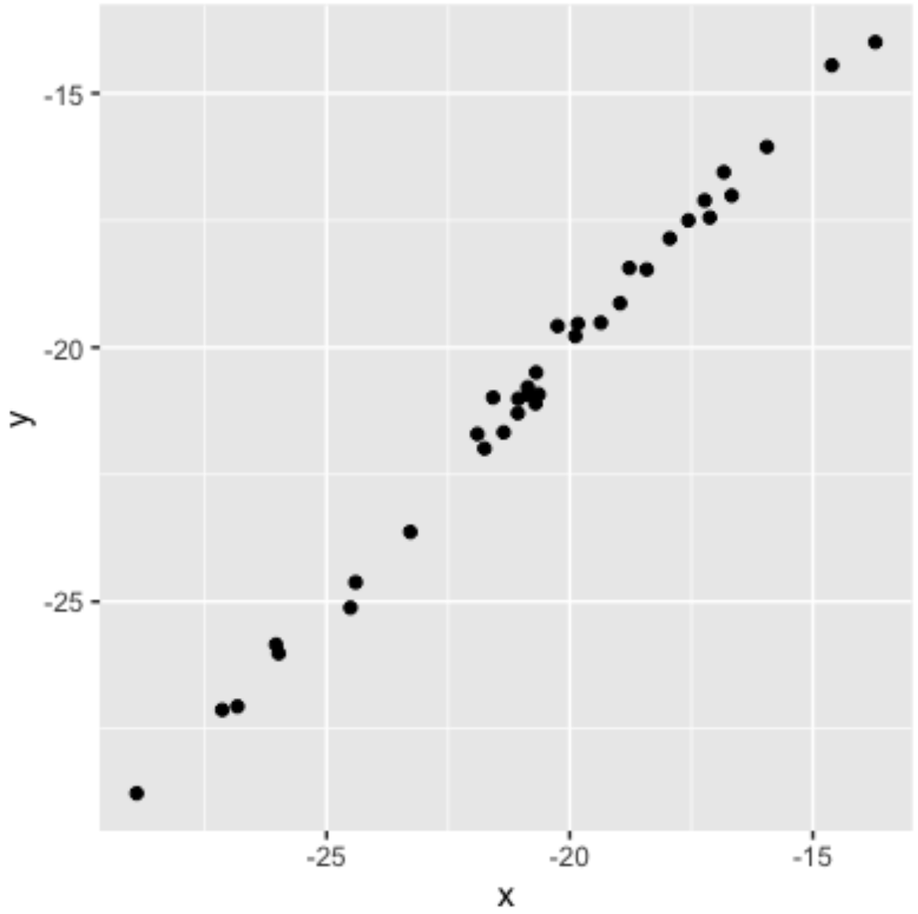
- x = explanatory/independent variable
- y = response/dependent variable

Correlation coefficient

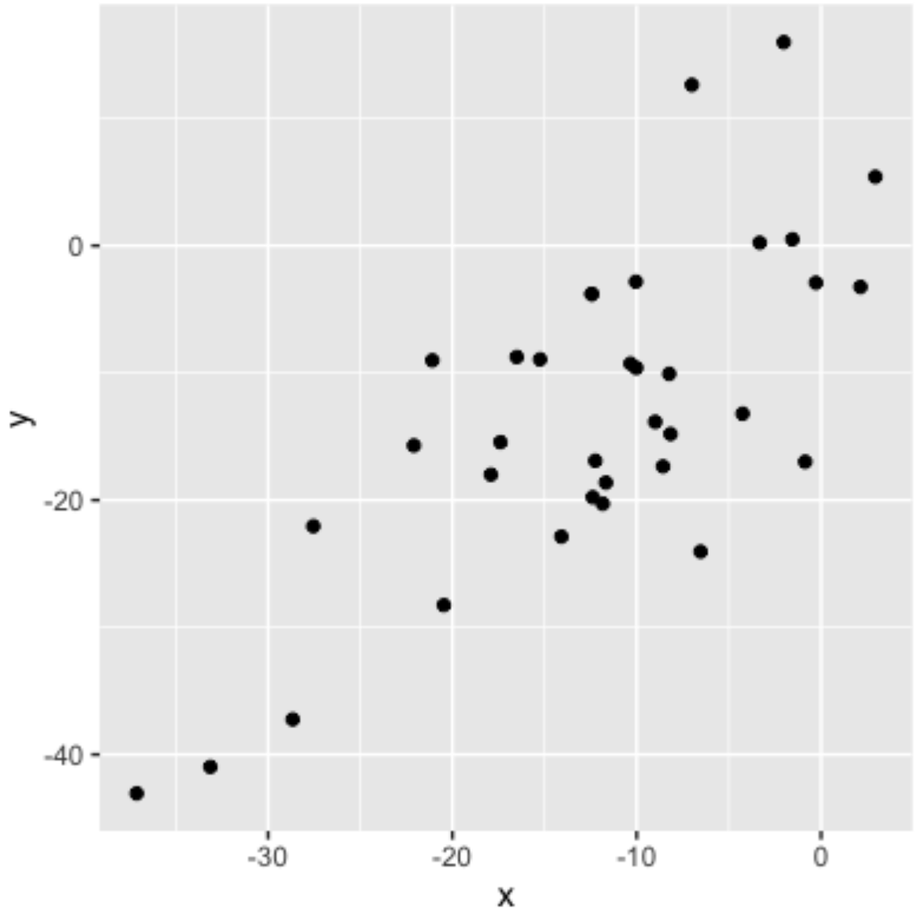
- Quantifies the linear relationship between two variables
- Number between -1 and 1
- Magnitude corresponds to strength of relationship
- Sign (+ or -) corresponds to direction of relationship

Magnitude = strength of relationship

0.99 (very strong relationship)

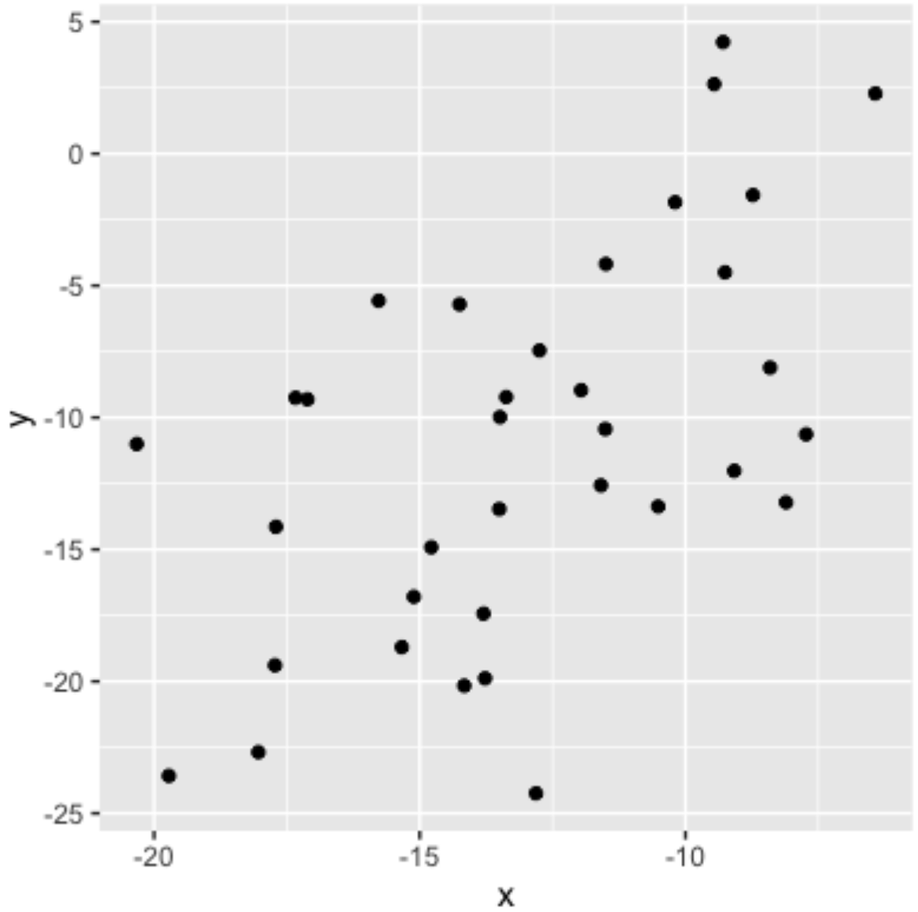


0.75 (strong relationship)

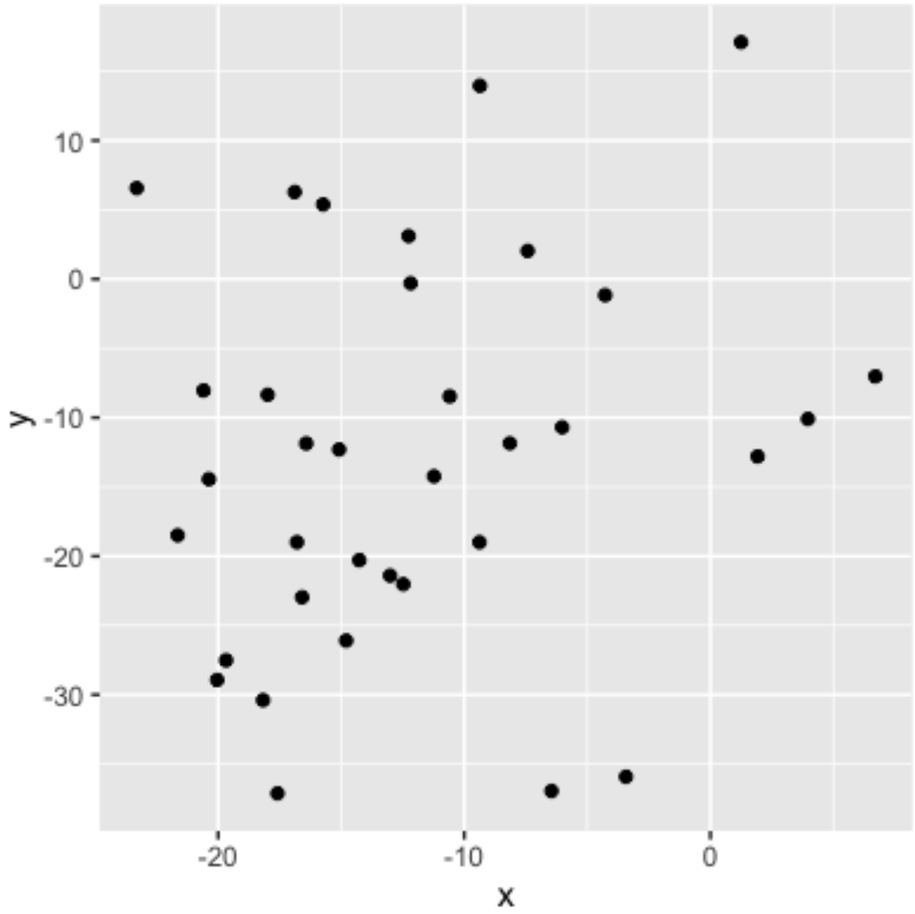


Magnitude = strength of relationship

0.56 (moderate relationship)

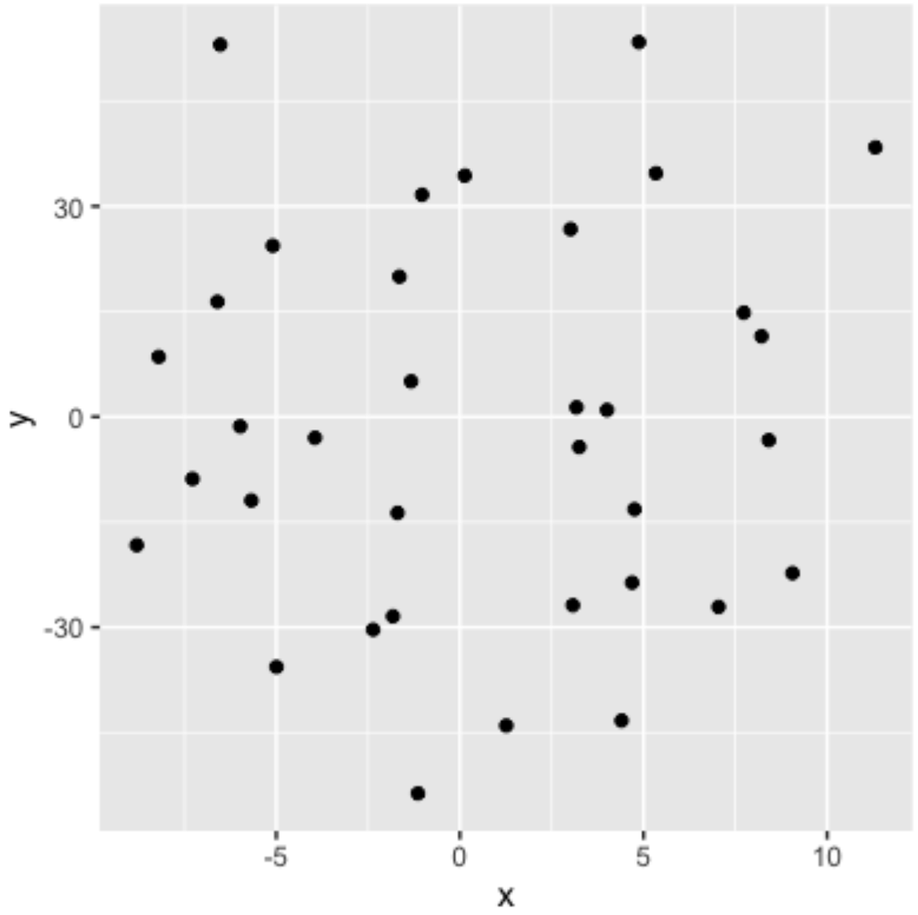


0.21 (weak relationship)



Magnitude = strength of relationship

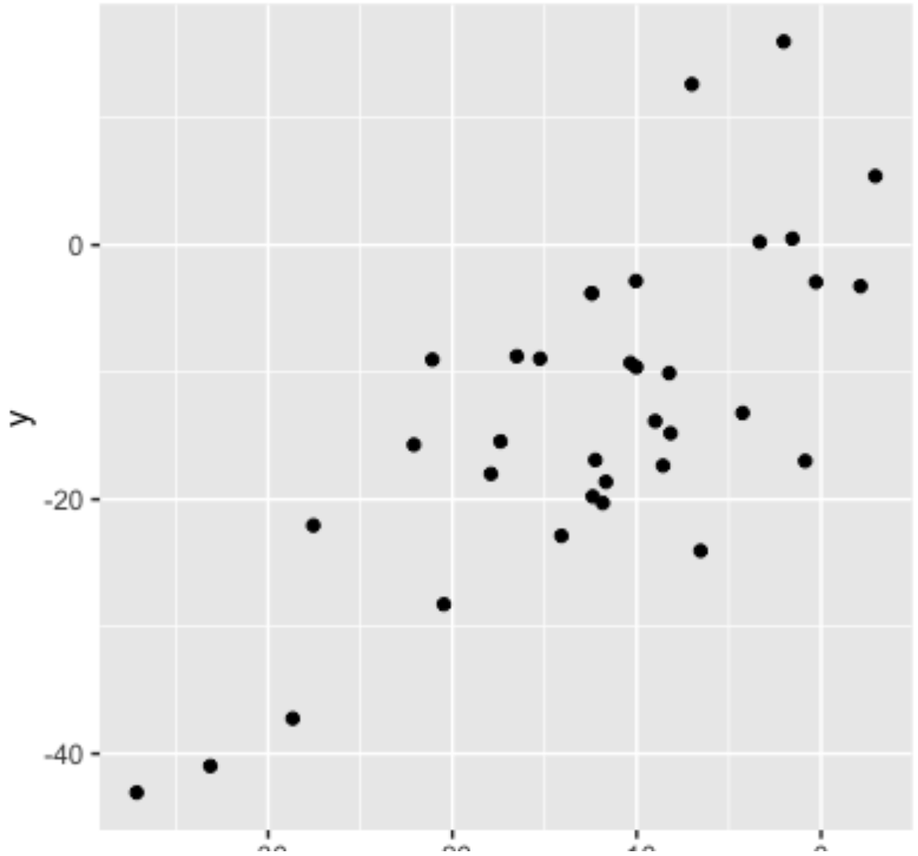
0.04 (no relationship)



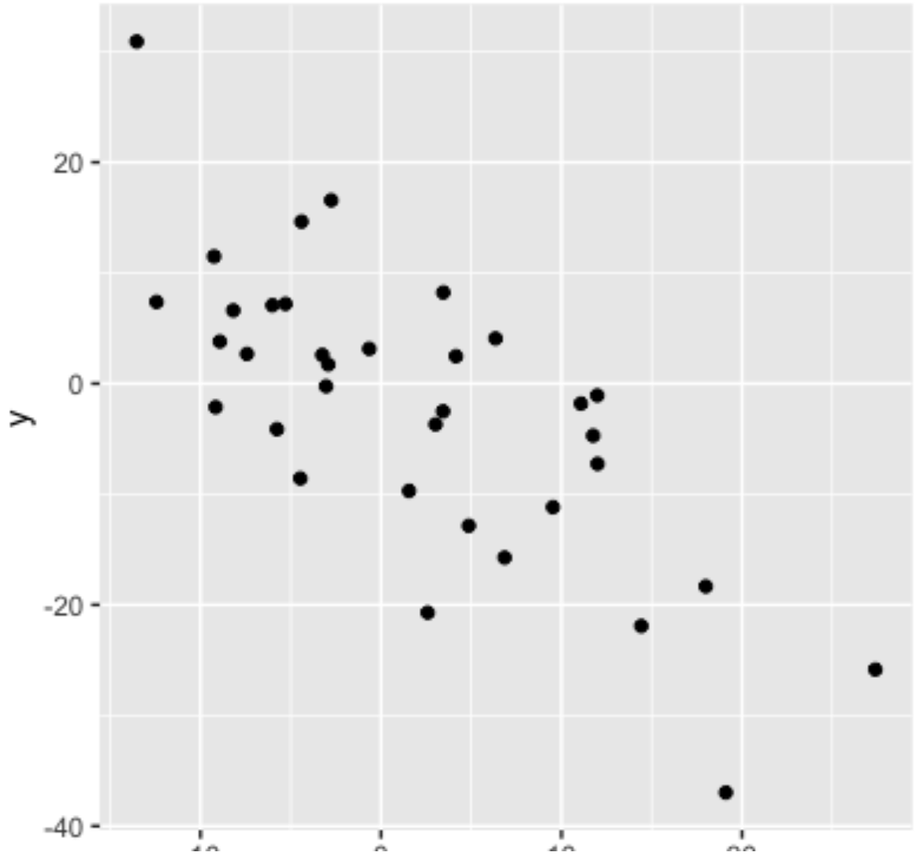
- Knowing the value of `x` doesn't tell us anything about `y`

Sign = direction

0.75: as `x` increases, `y` increases

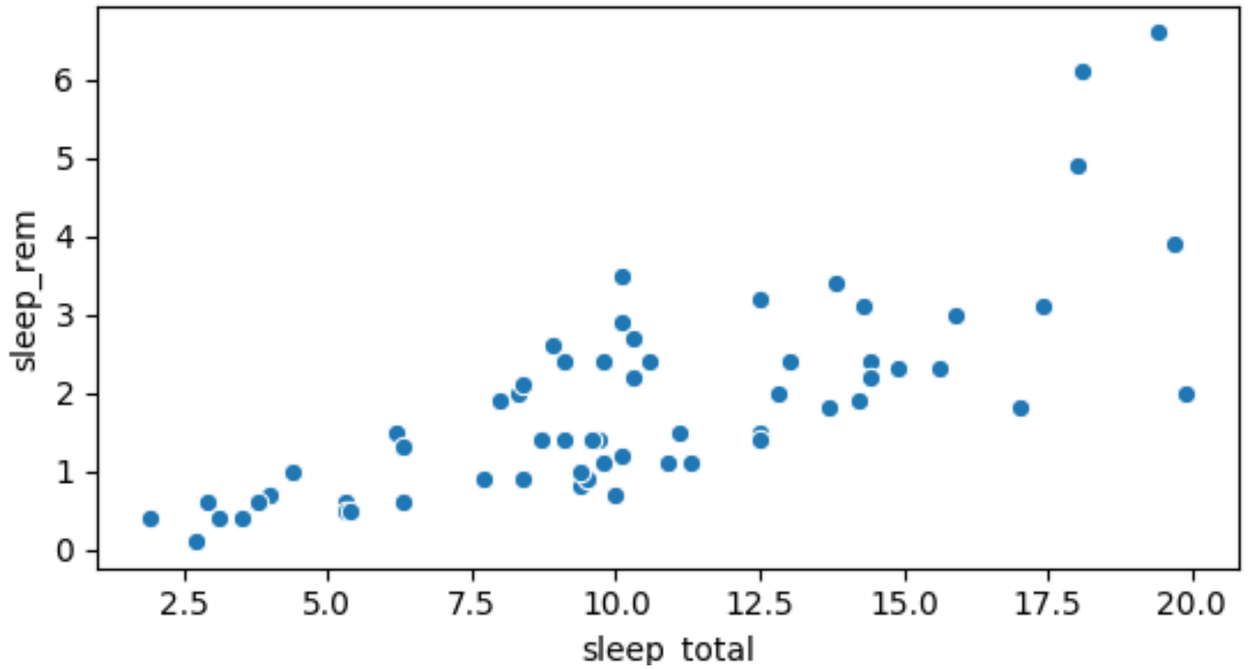


-0.75: as `x` increases, `y` decreases



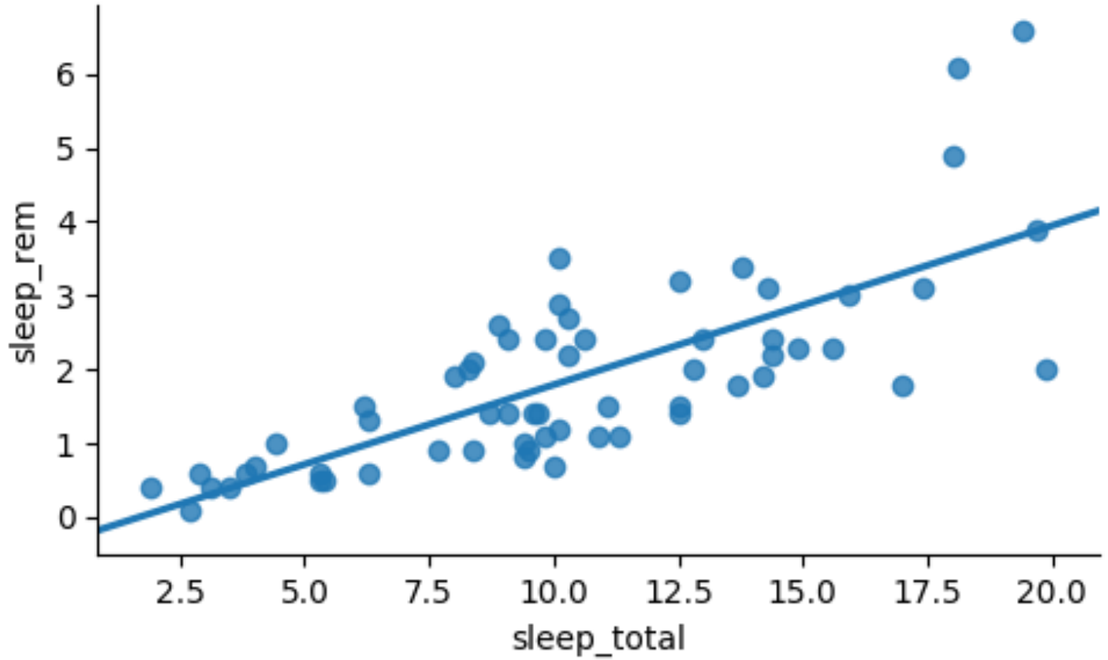
Visualizing relationships

```
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

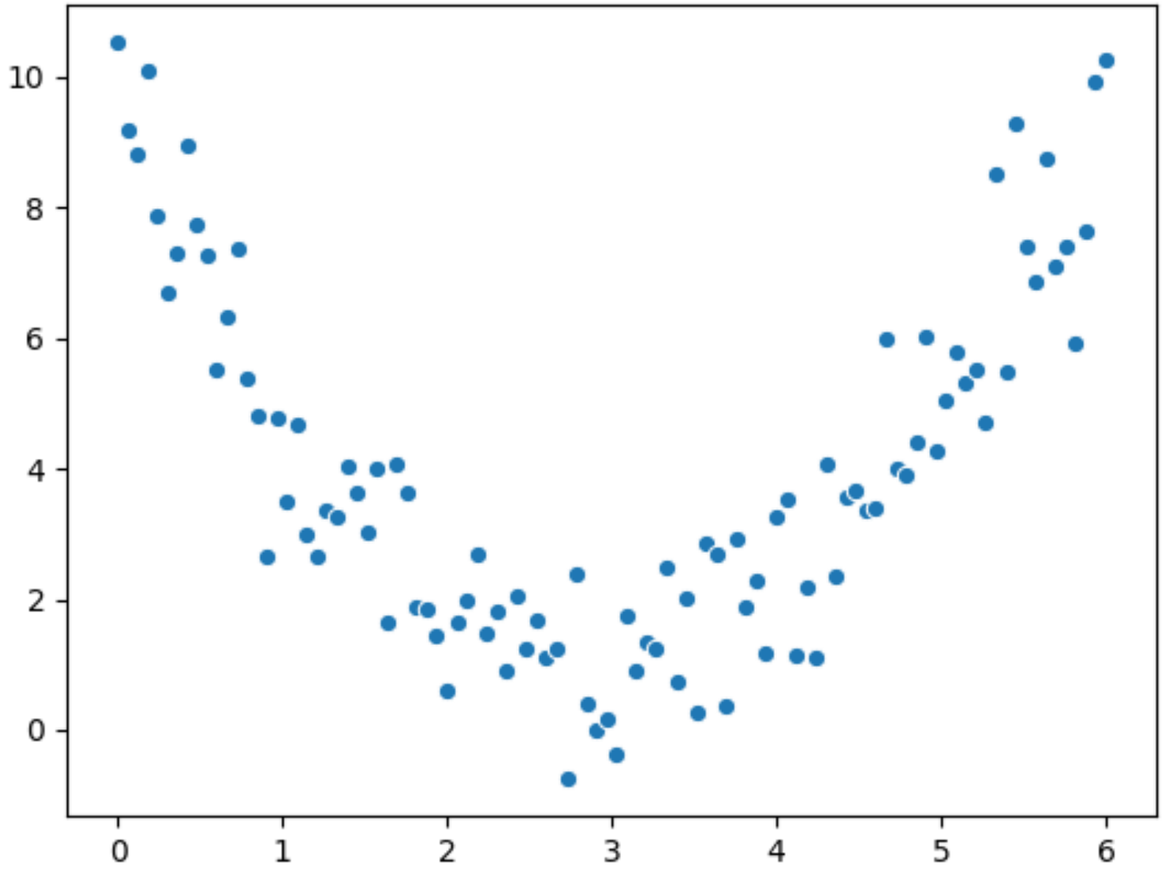
Correlation caveats

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

datacamp

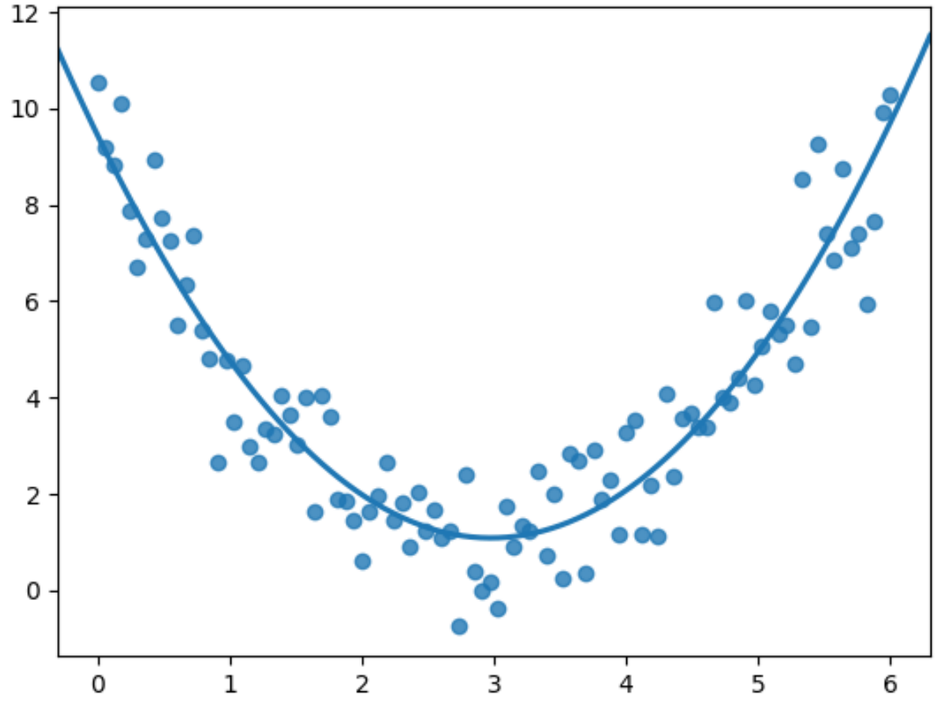
Non-linear relationships



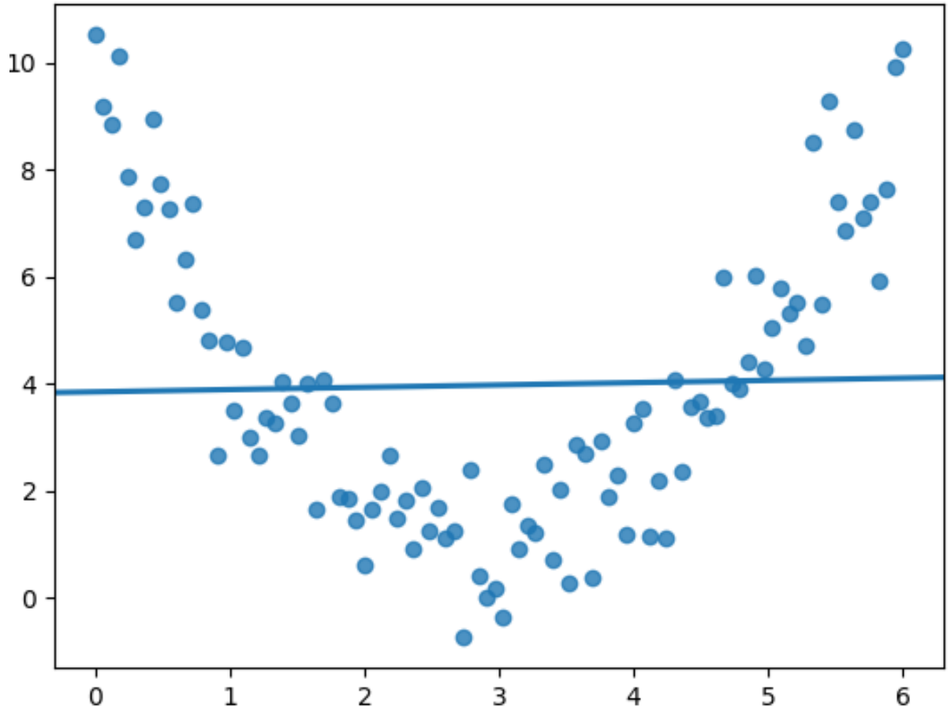
$$r = 0.18$$

Non-linear relationships

What we see:



What the correlation coefficient sees:



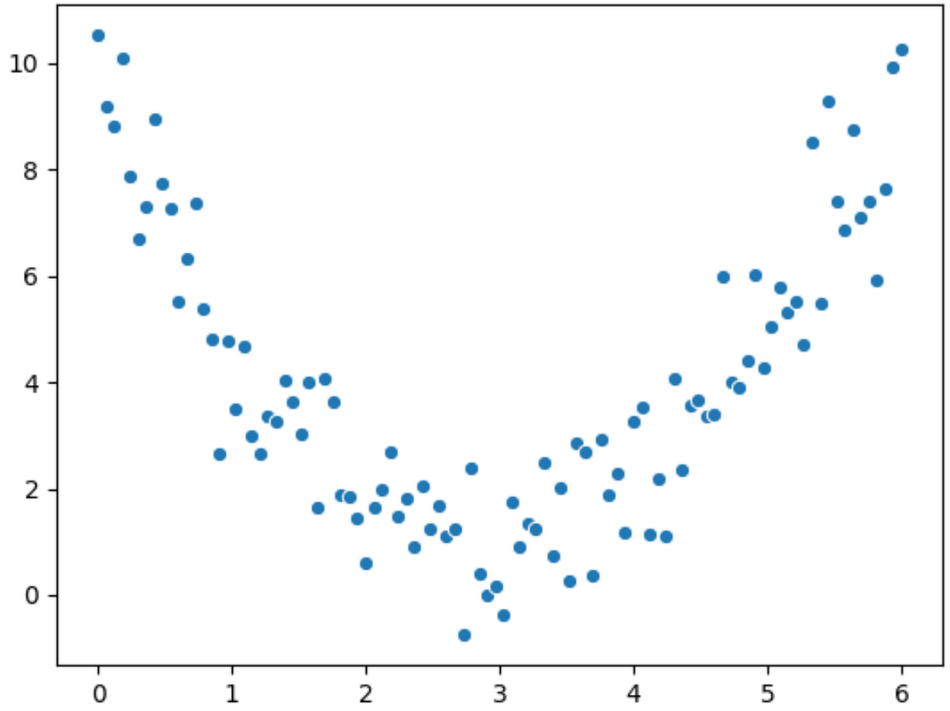
Correlation only accounts for linear relationships

Correlation shouldn't be used blindly

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

```
0.081094
```

Always visualize your data

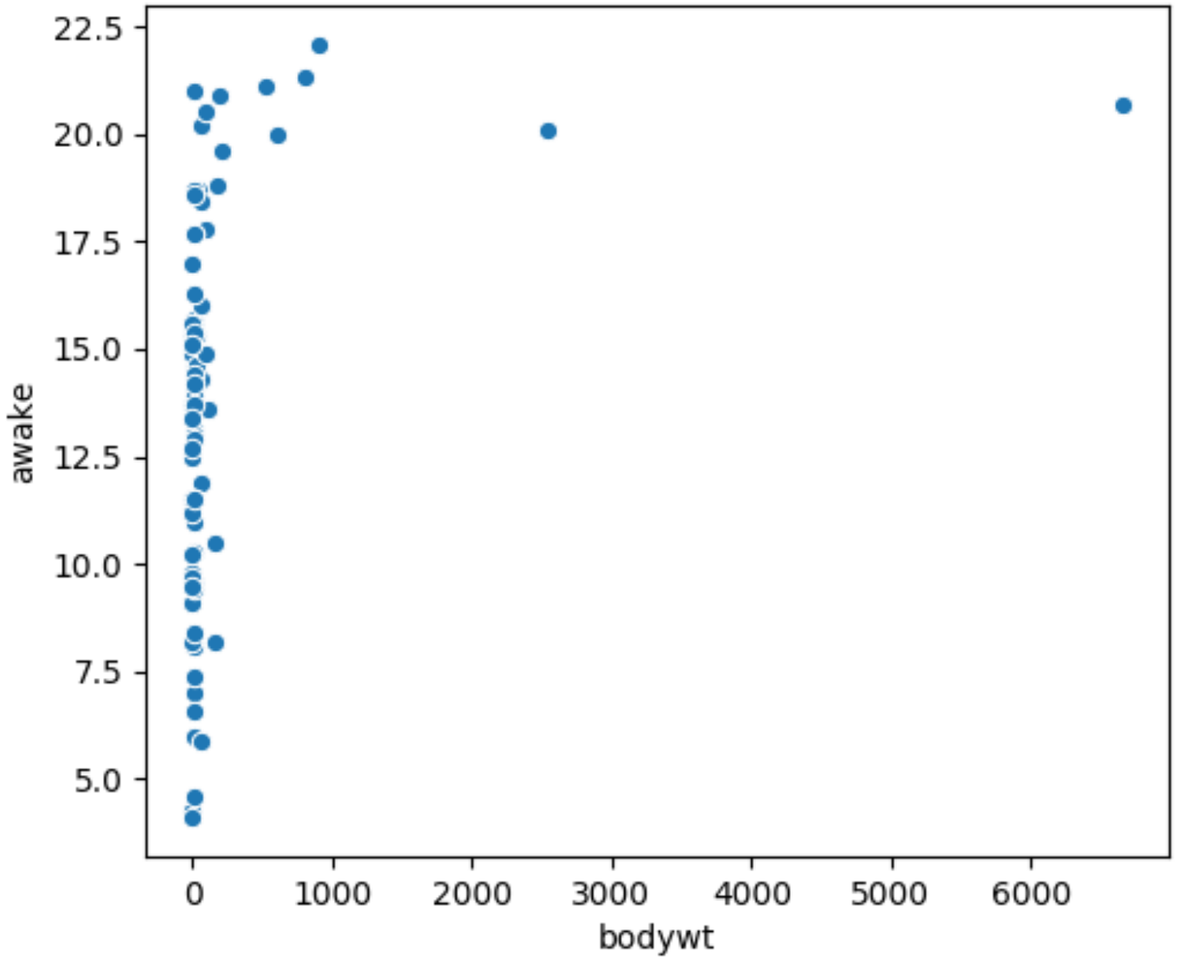


Mammal sleep data

```
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
```

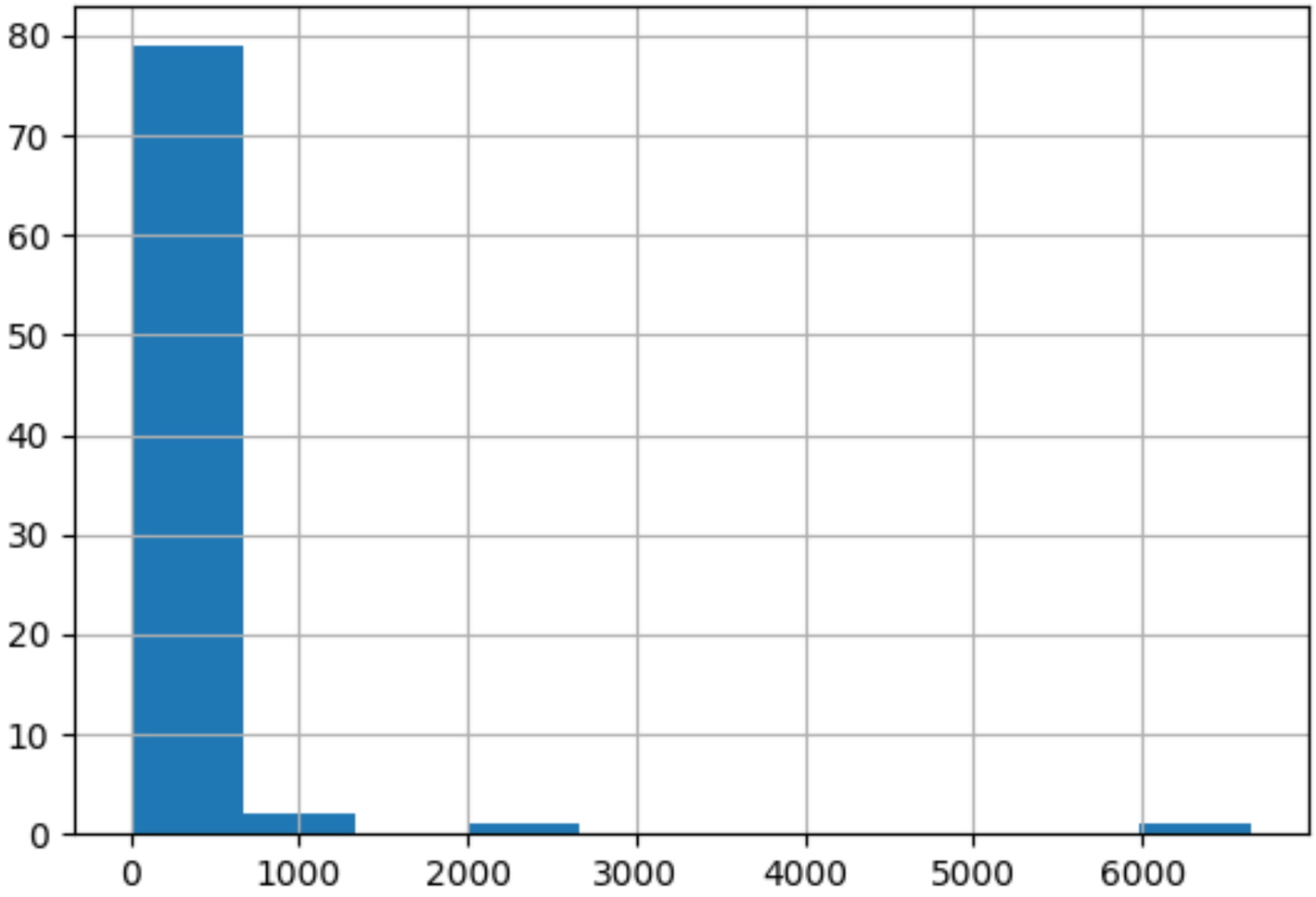
Body weight vs. awake time




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

```
0.3119801
```

Distribution of body weight



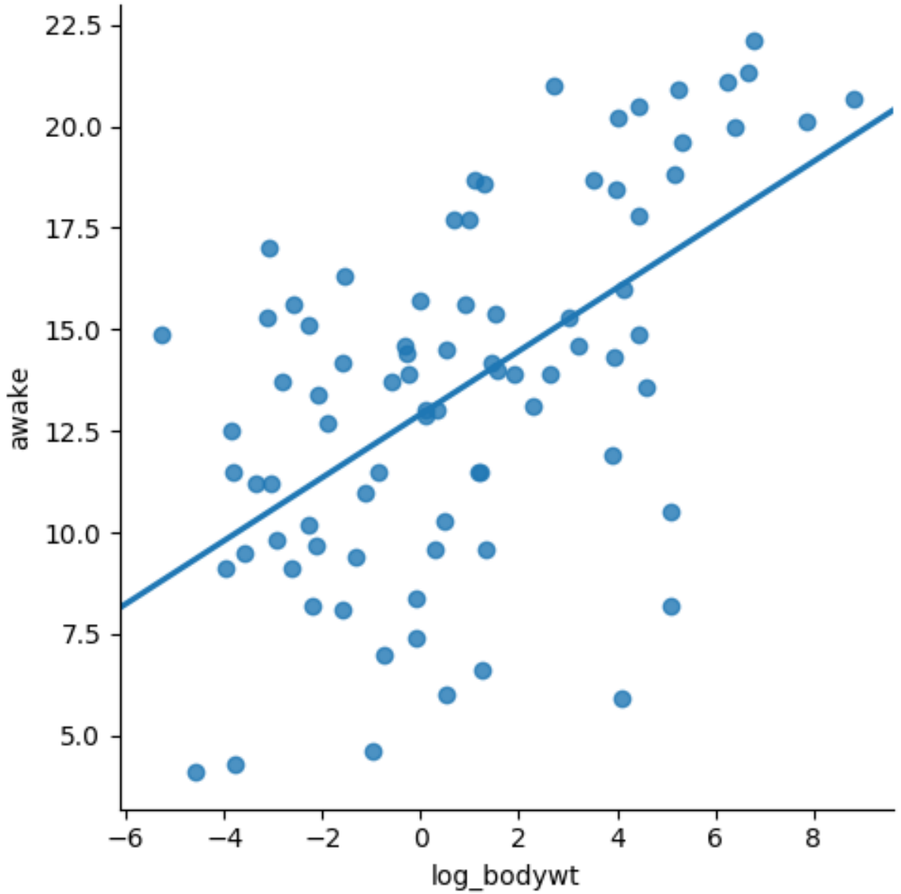
Log transformation

```
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`

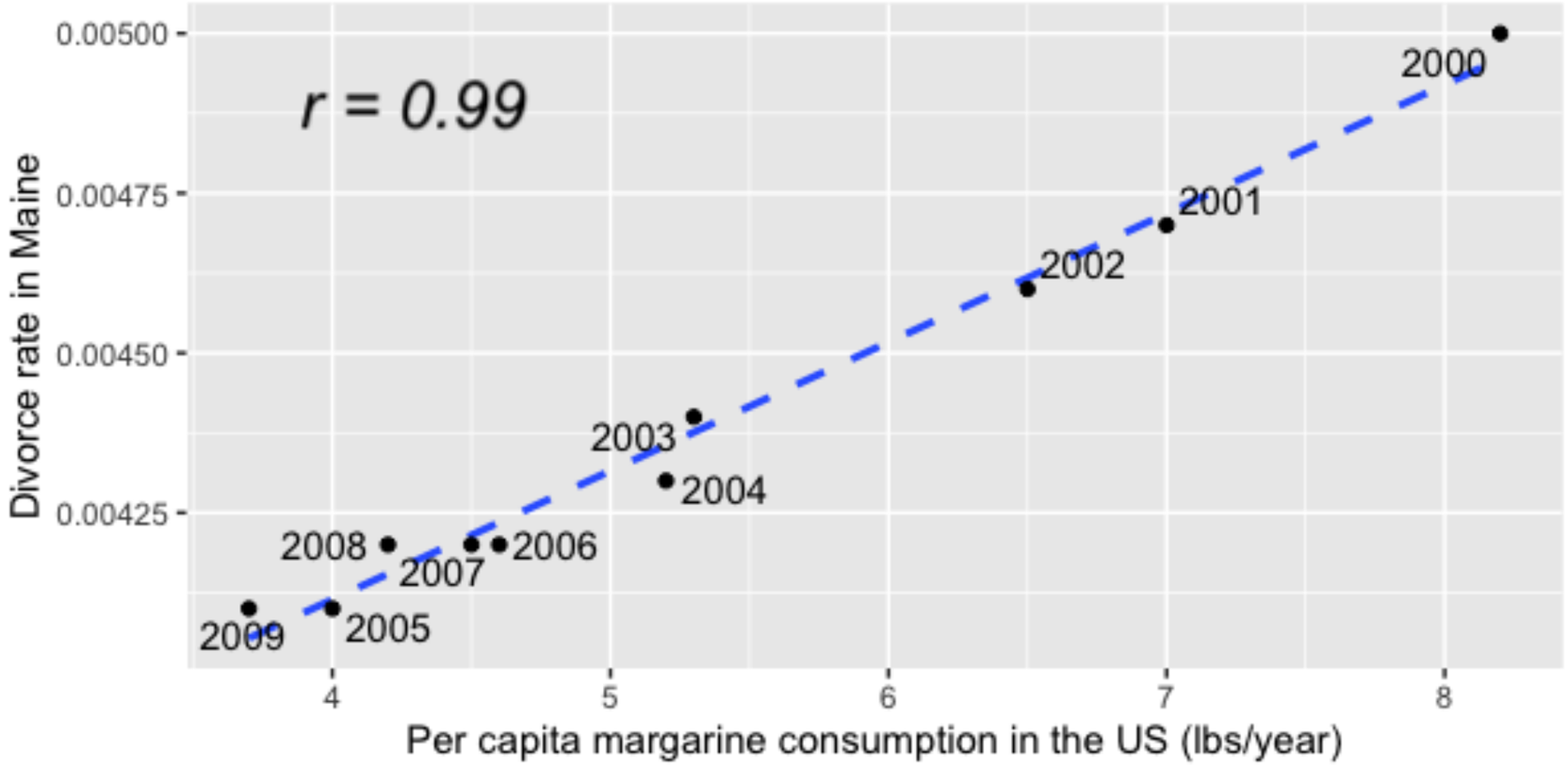
Why use a transformation?

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

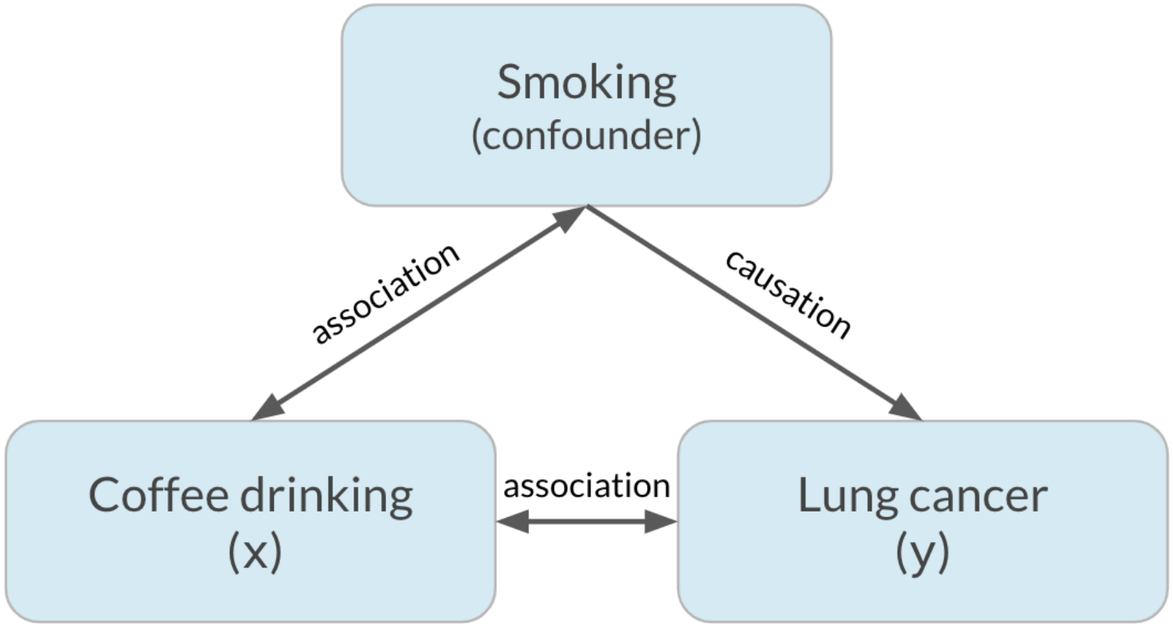
Introduction to Linear Modeling in Python

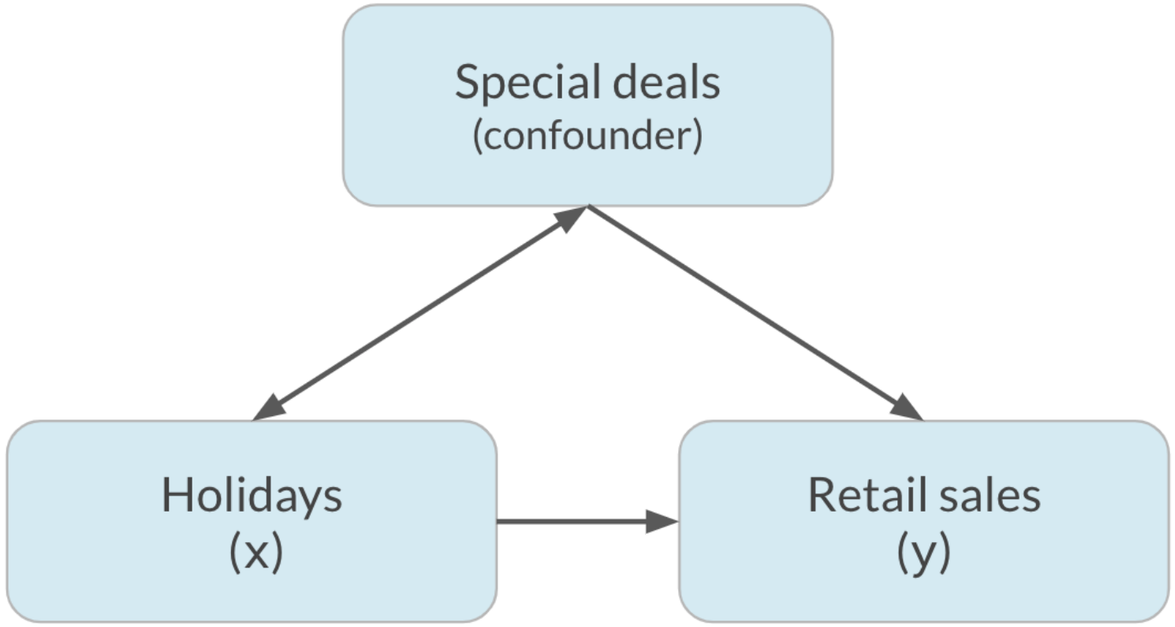
Correlation does not imply causation

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



Confounding





Design of experiments

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

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