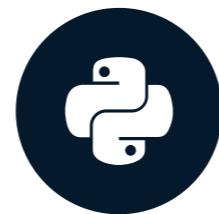


Probability density functions

STATISTICAL THINKING IN PYTHON (PART 1)



Justin Bois

Lecturer at the California Institute of
Technology

Continuous variables

- Quantities that can take any value, not just discrete values

Michelson's speed of light experiment



Image: public domain, Smithsonian

Data: Michelson, 1880

Michelson's speed of light experiment



measured speed of light (1000 km/s)

```
299.85 299.74 299.90 300.07 299.93  
299.85 299.95 299.98 299.98 299.88  
300.00 299.98 299.93 299.65 299.76  
299.81 300.00 300.00 299.96 299.96  
299.96 299.94 299.96 299.94 299.88  
299.80 299.85 299.88 299.90 299.84  
299.83 299.79 299.81 299.88 299.88  
299.83 299.80 299.79 299.76 299.80  
299.88 299.88 299.88 299.86 299.72  
299.72 299.62 299.86 299.97 299.95  
299.88 299.91 299.85 299.87 299.84  
299.84 299.85 299.84 299.84 299.84  
299.89 299.81 299.81 299.82 299.80  
299.77 299.76 299.74 299.75 299.76  
299.91 299.92 299.89 299.86 299.88  
299.72 299.84 299.85 299.85 299.78  
299.89 299.84 299.78 299.81 299.76  
299.81 299.79 299.81 299.82 299.85  
299.87 299.87 299.81 299.74 299.81  
299.94 299.95 299.80 299.81 299.87
```

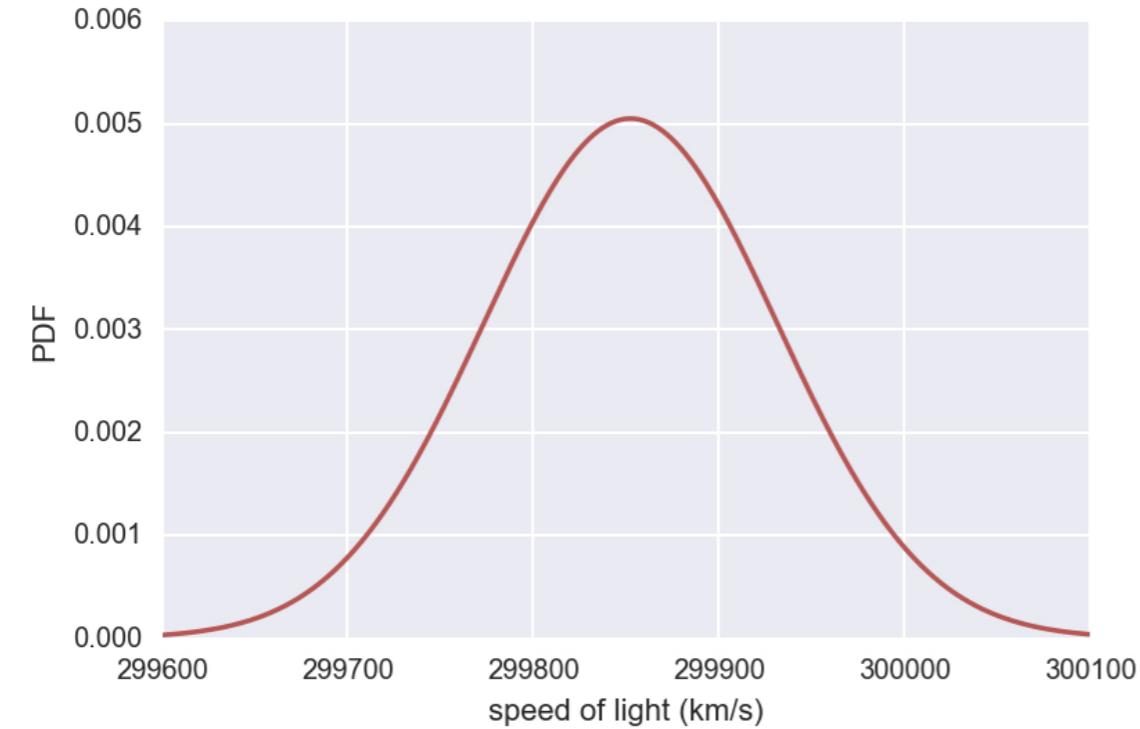
Image: public domain, Smithsonian

Data: Michelson, 1880

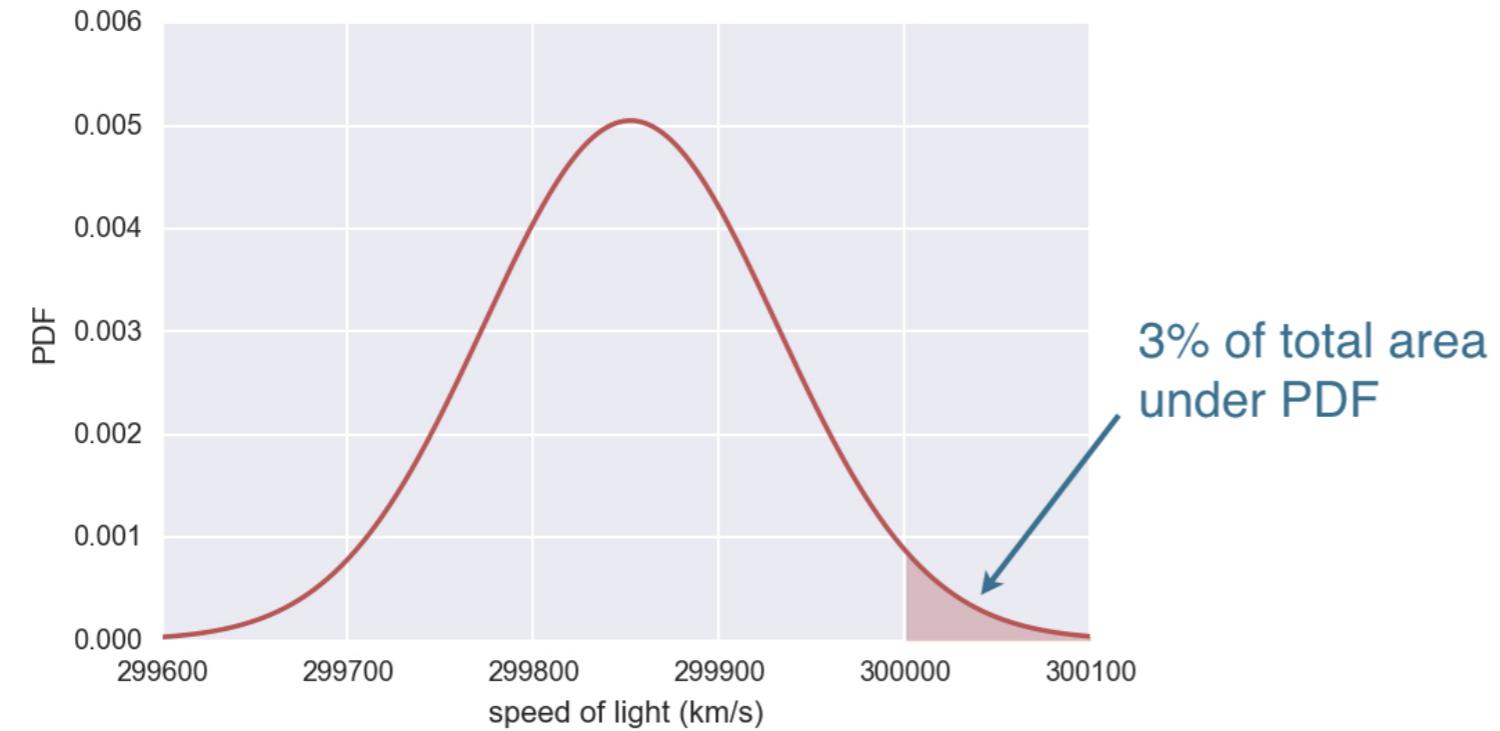
Probability density function (PDF)

- Continuous analog to the PMF
- Mathematical description of the relative likelihood of observing a value of a continuous variable

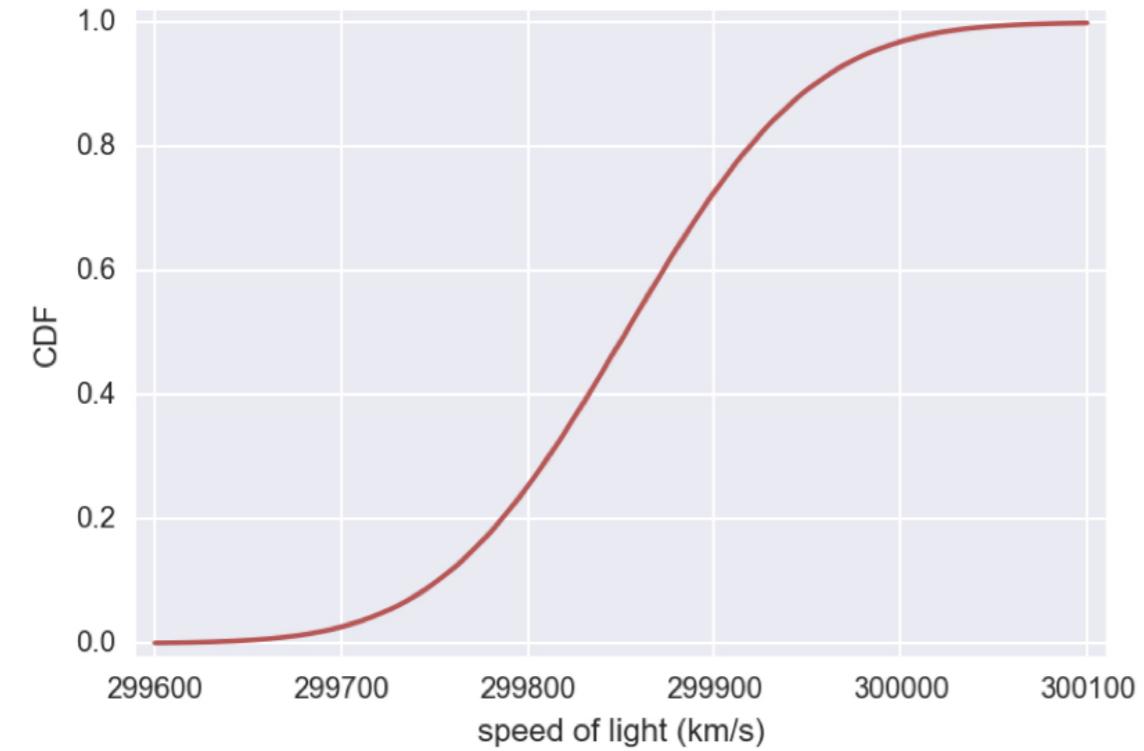
Normal PDF



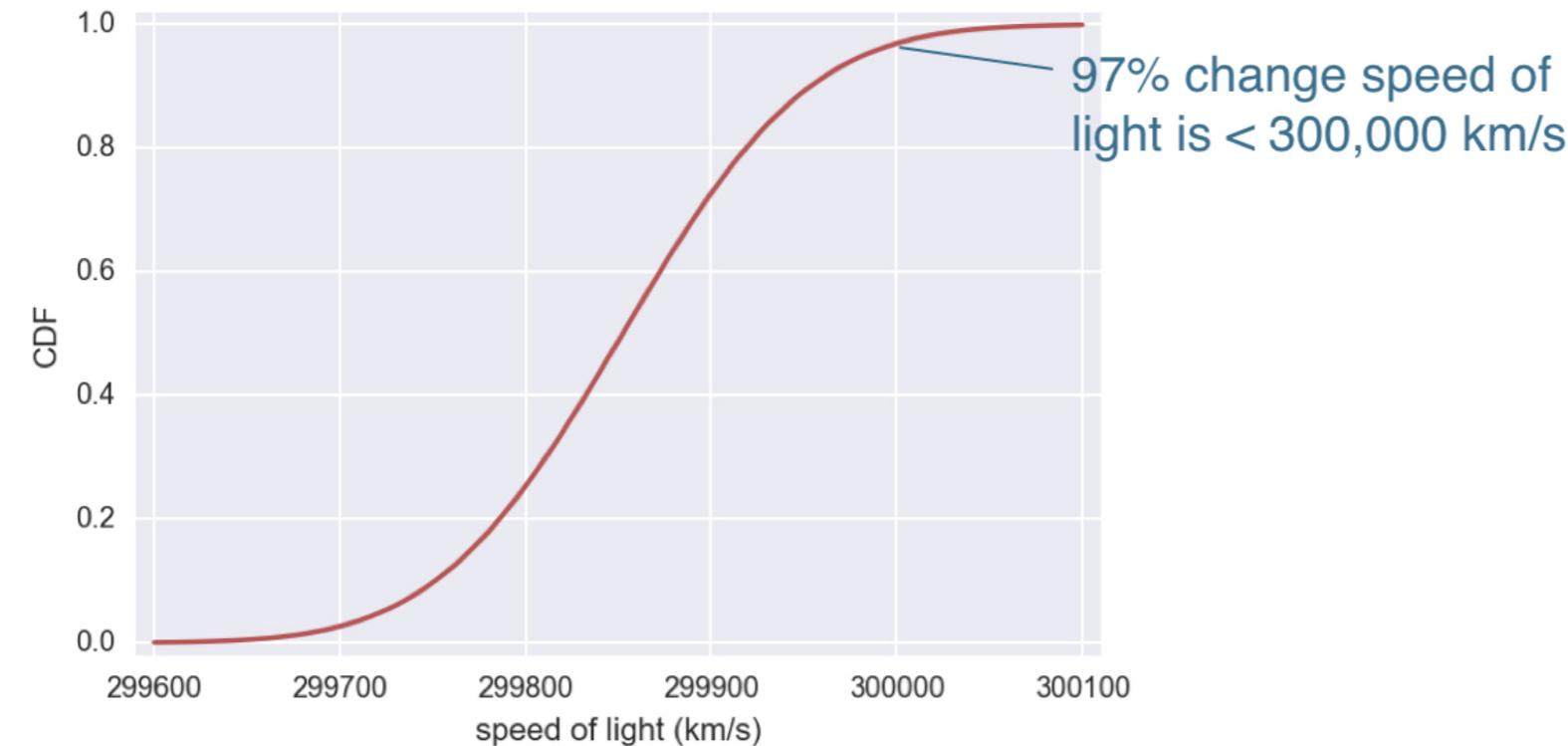
Normal PDF



Normal CDF



Normal CDF

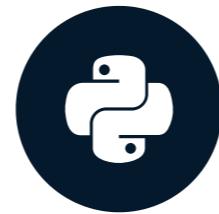


Let's practice!

STATISTICAL THINKING IN PYTHON (PART 1)

Introduction to the Normal distribution

STATISTICAL THINKING IN PYTHON (PART 1)



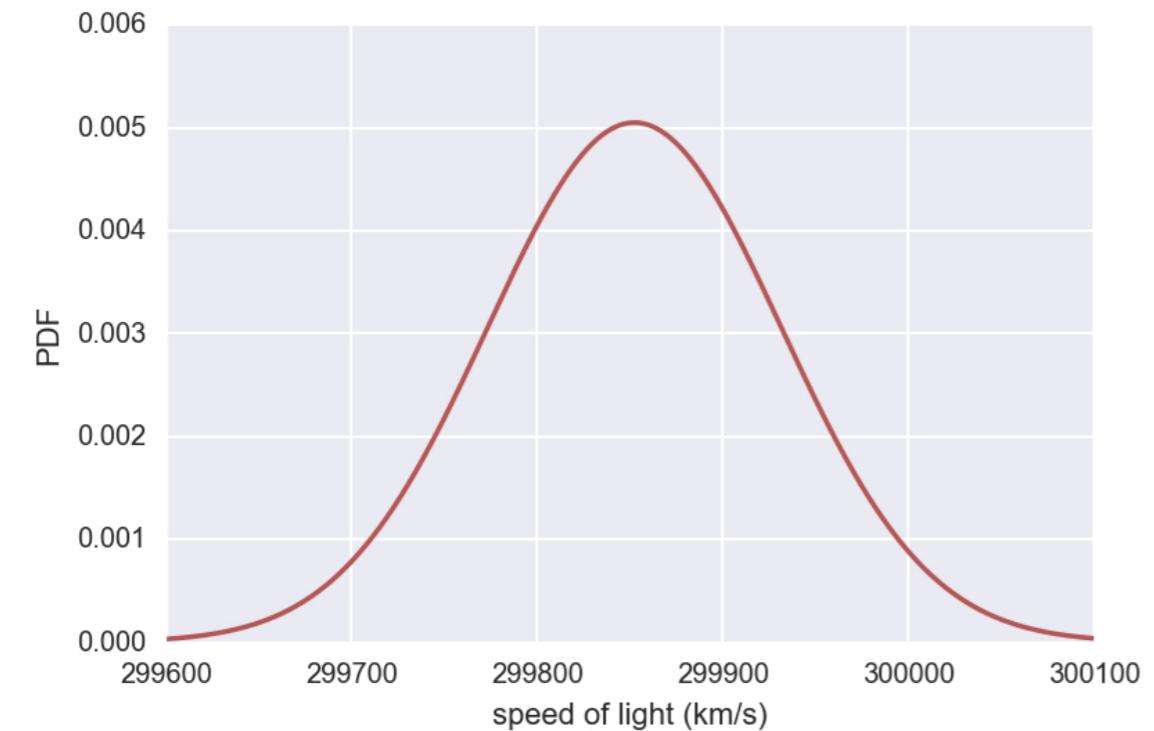
Justin Bois

Lecturer at the California Institute of
Technology

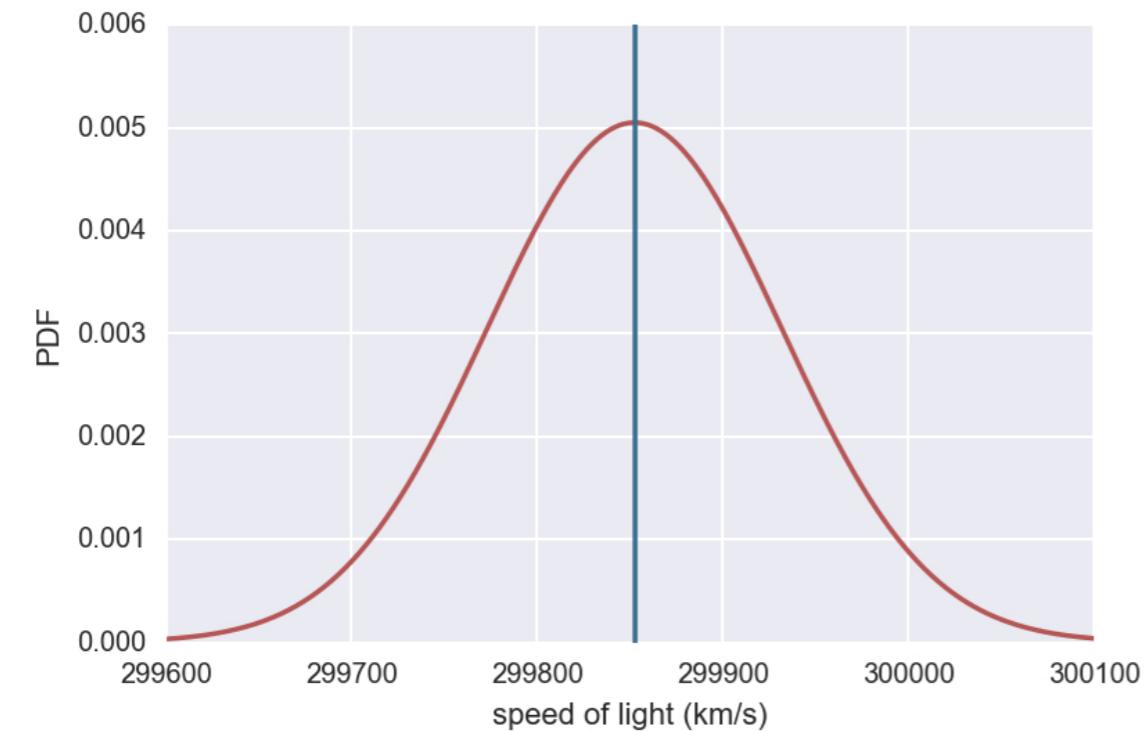
Normal distribution

- Describes a continuous variable whose PDF has a single symmetric peak.

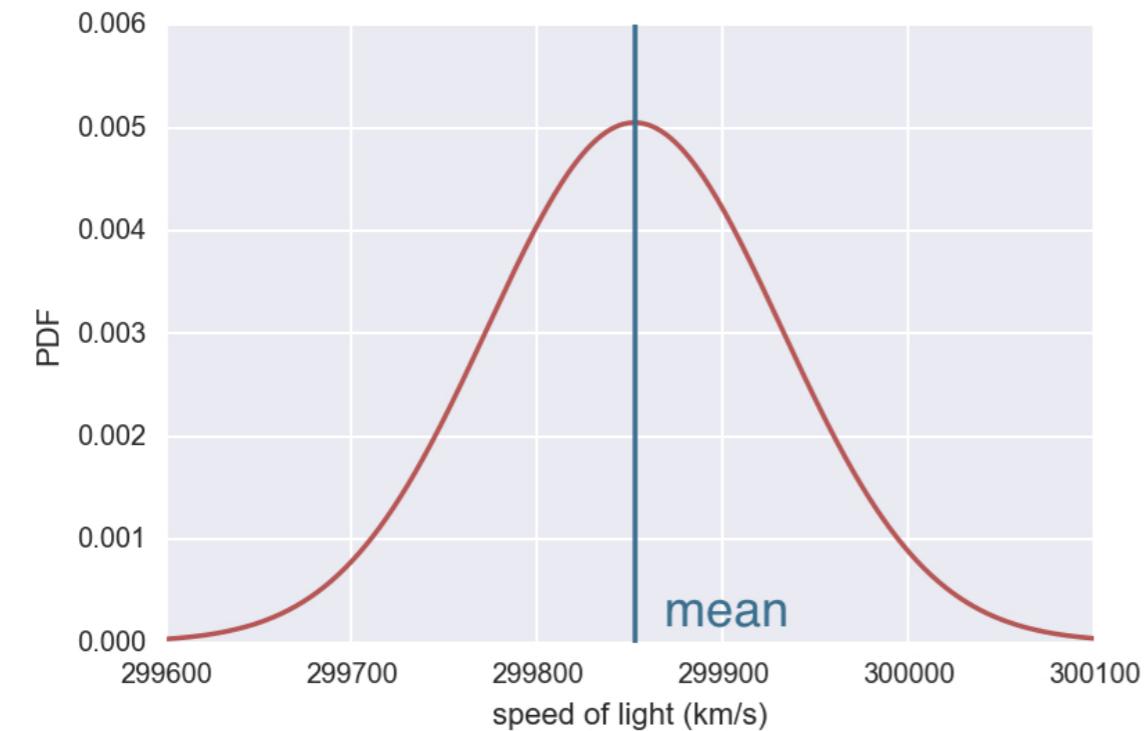
Normal distribution



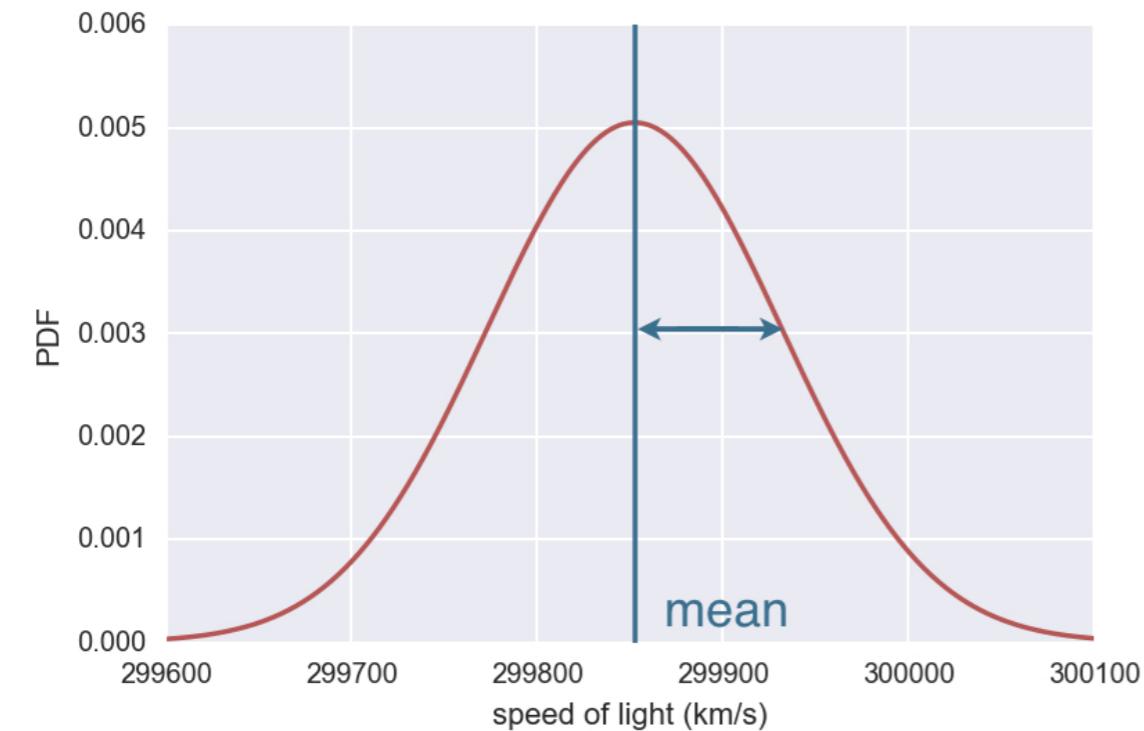
Normal distribution



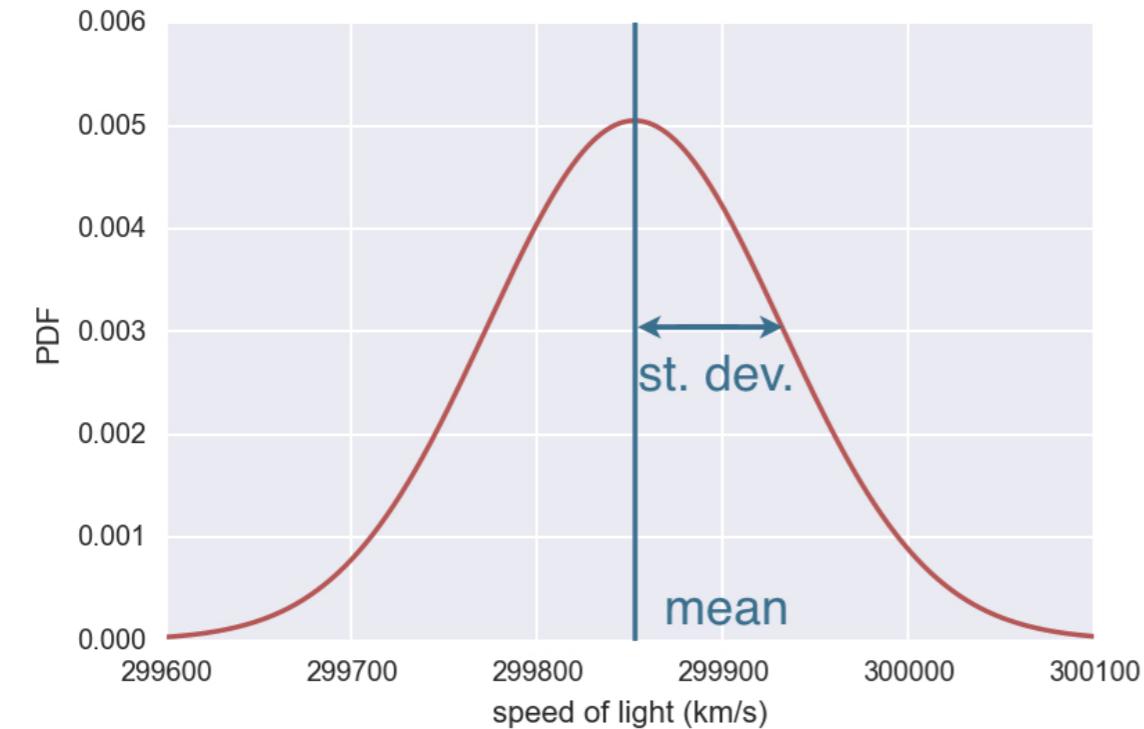
Normal distribution



Normal distribution



Normal distribution



Parameter

mean of a
Normal distribution

\neq

Calculated from data

mean computed
from data

Parameter

mean of a
Normal distribution

\neq

Calculated from data

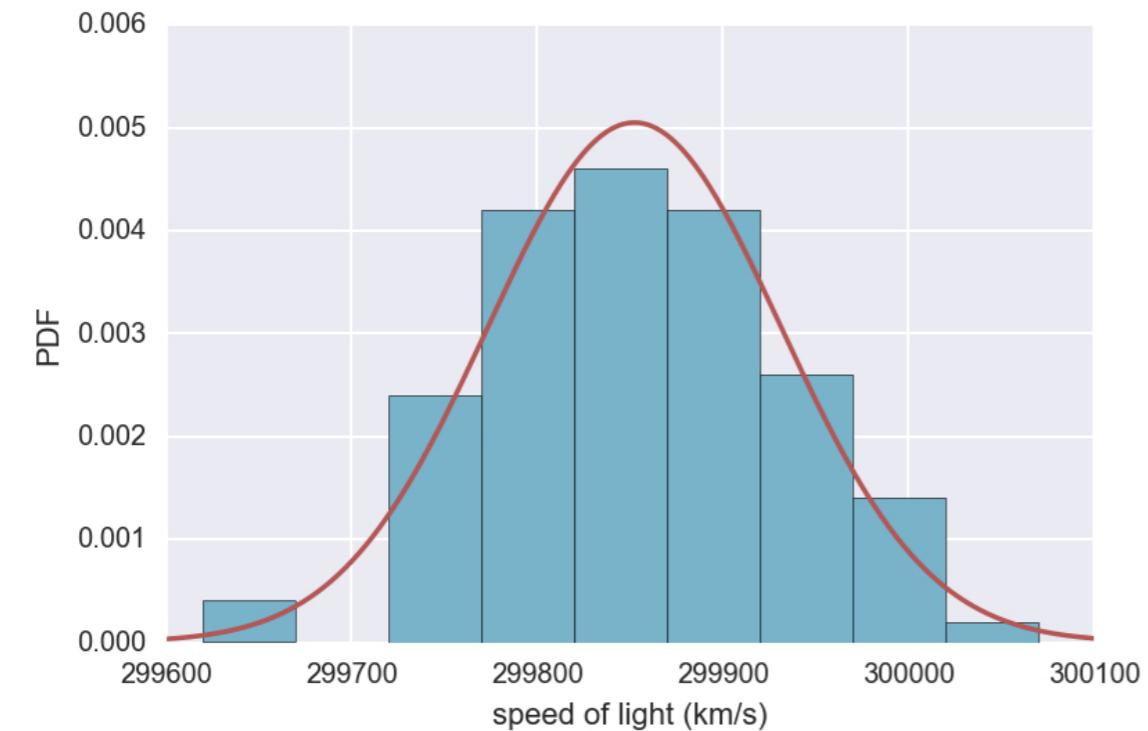
st. dev. of a
Normal distribution

\neq

mean computed
from data

standard deviation
computed from data

Comparing data to a Normal PDF



Data: Michelson, 1880

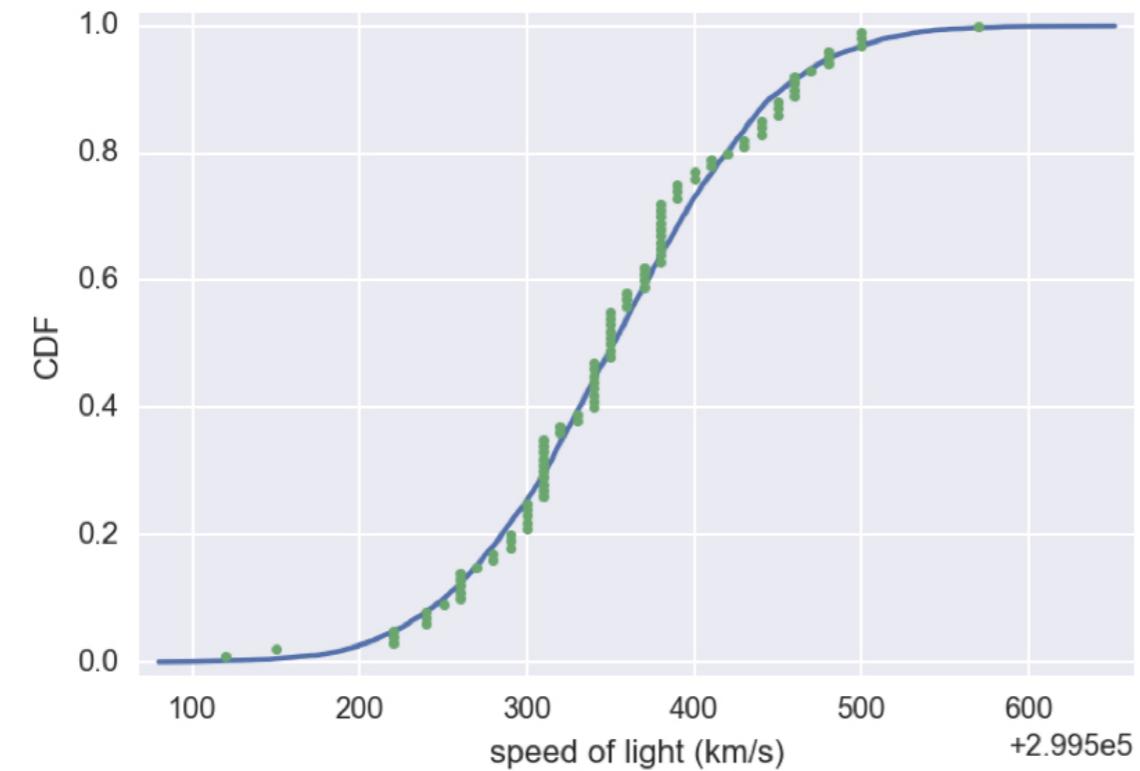
Checking Normality of Michelson data

```
import numpy as np  
  
mean = np.mean(michelson_speed_of_light)  
std = np.std(michelson_speed_of_light)  
samples = np.random.normal(mean, std, size=10000)  
x, y = ecdf(michelson_speed_of_light)  
x_theor, y_theor = ecdf(samples)
```

Checking Normality of Michelson data

```
import matplotlib.pyplot as plt  
import seaborn as sns  
  
sns.set()  
  
_ = plt.plot(x_theor, y_theor)  
_ = plt.plot(x, y, marker='.', linestyle='none')  
_ = plt.xlabel('speed of light (km/s)')  
_ = plt.ylabel('CDF')  
plt.show()
```

Checking Normality of Michelson data



Data: Michelson, 1880

Let's practice!

STATISTICAL THINKING IN PYTHON (PART 1)

The Normal distribution: Properties and warnings

STATISTICAL THINKING IN PYTHON (PART 1)

Justin Bois

Lecturer at the California Institute of
Technology



Image: Deutsche Bundesbank



Image: Deutsche Bundesbank

The Gaussian distribution

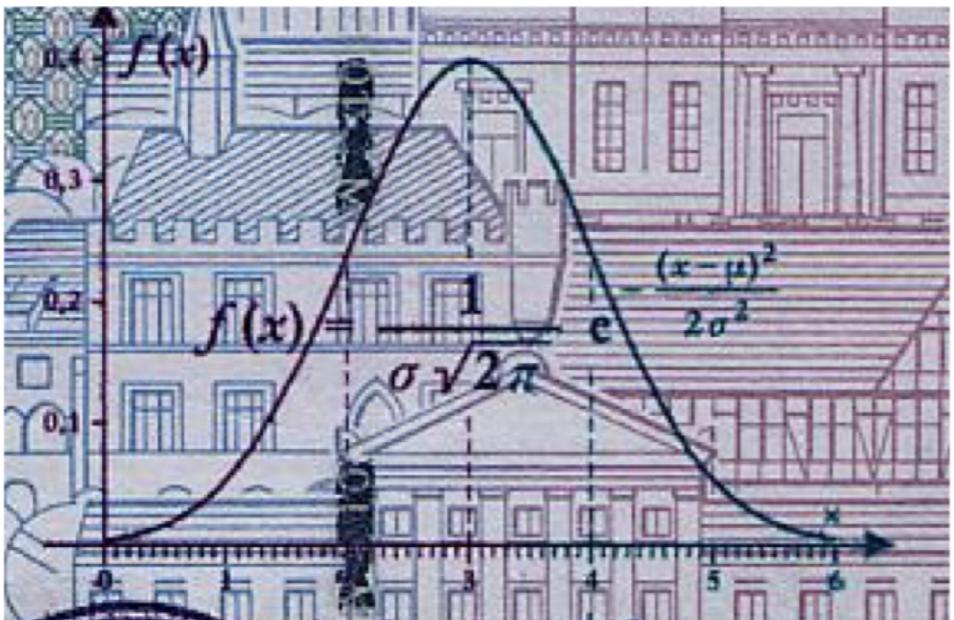
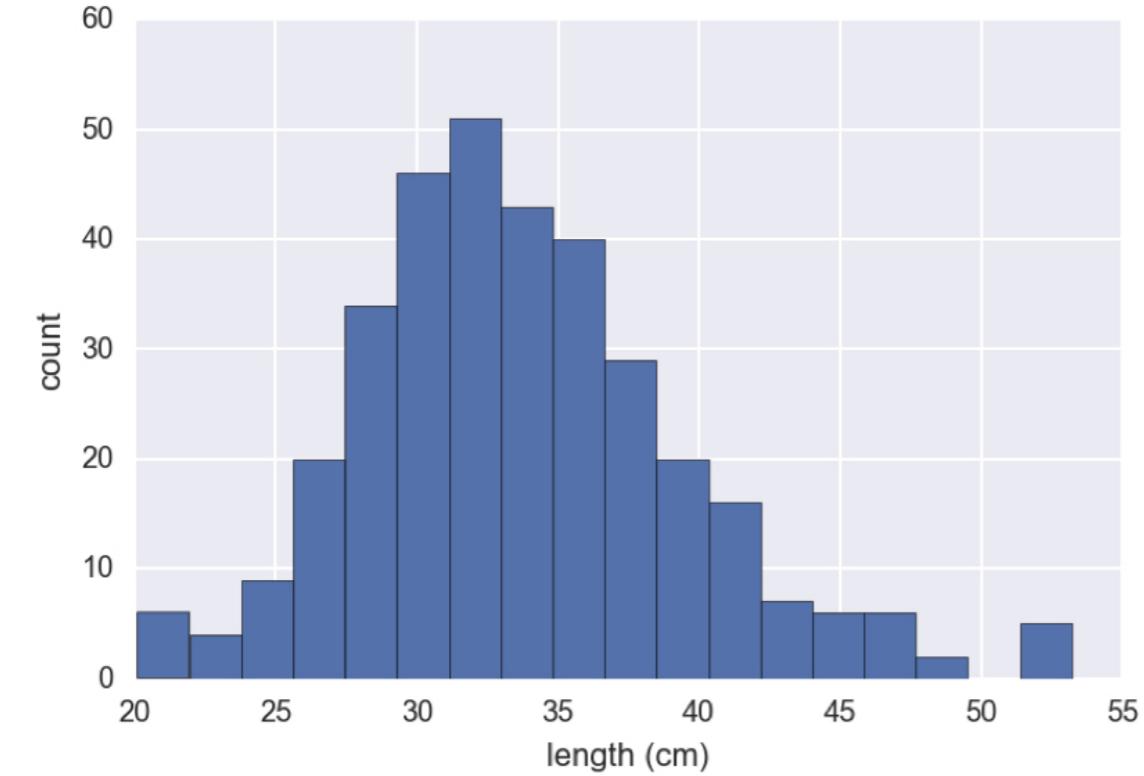


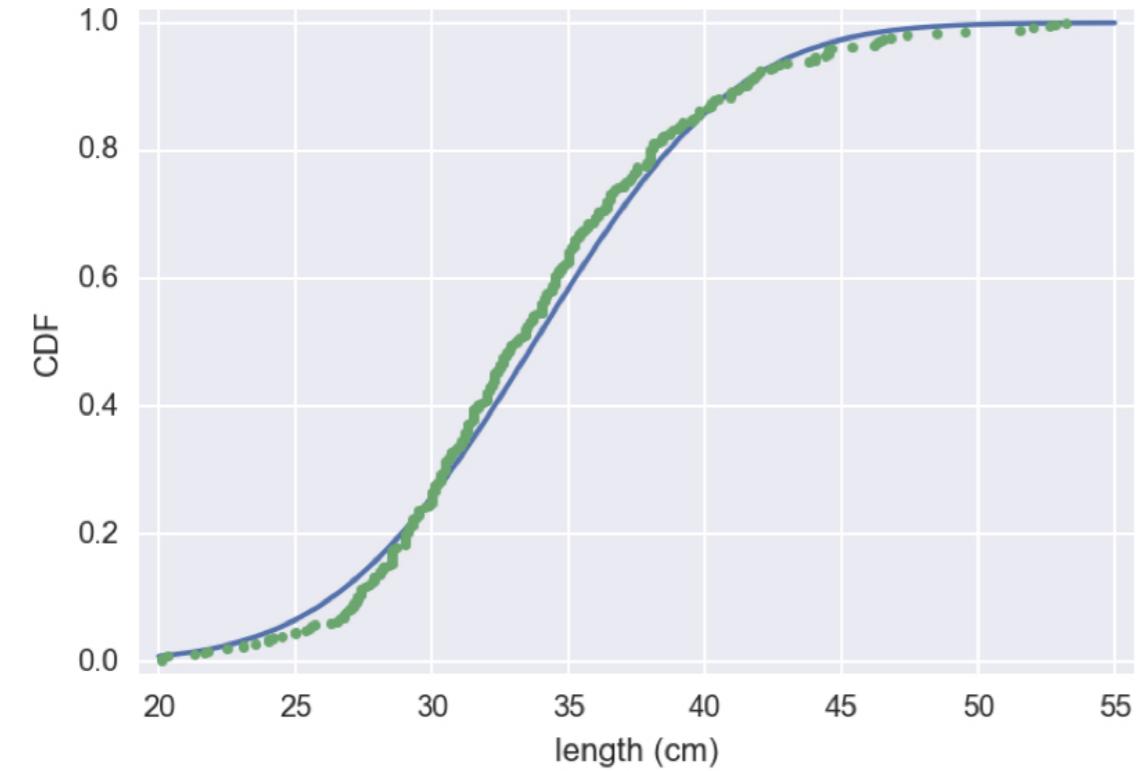
Image: Deutsche Bundesbank

Length of MA large mouth bass



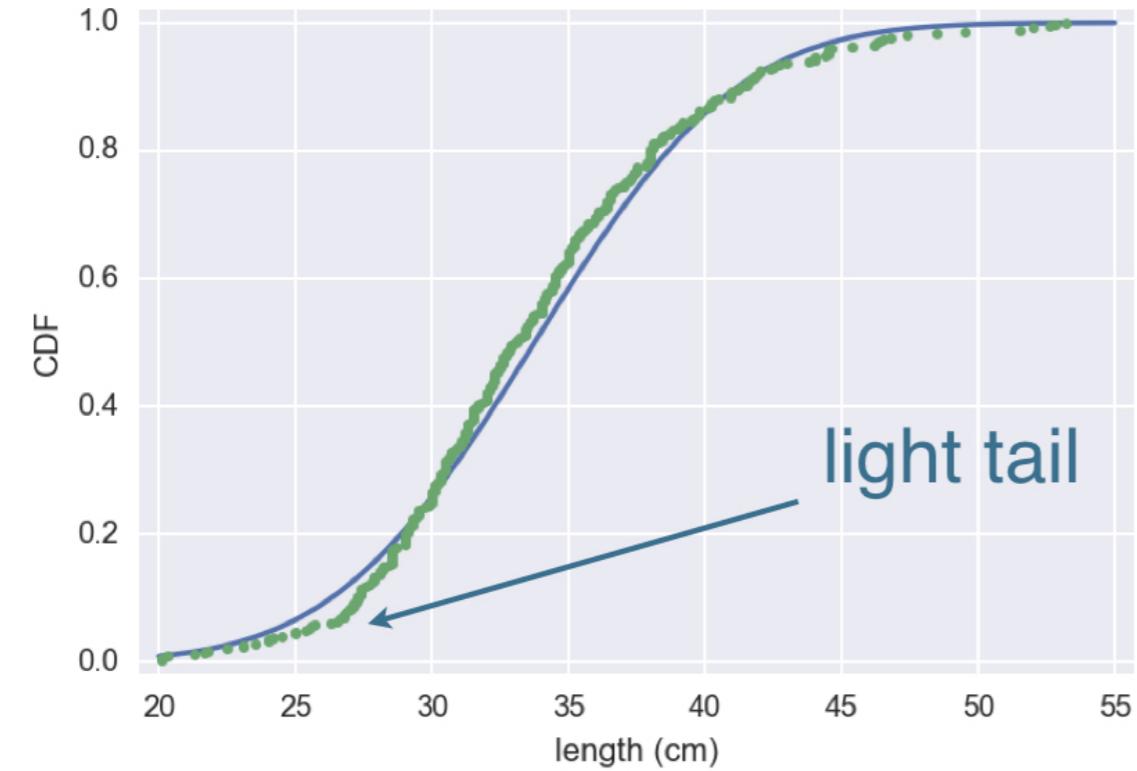
Source: Mass. Dept. of Environmental Protection

Length of MA large mouth bass



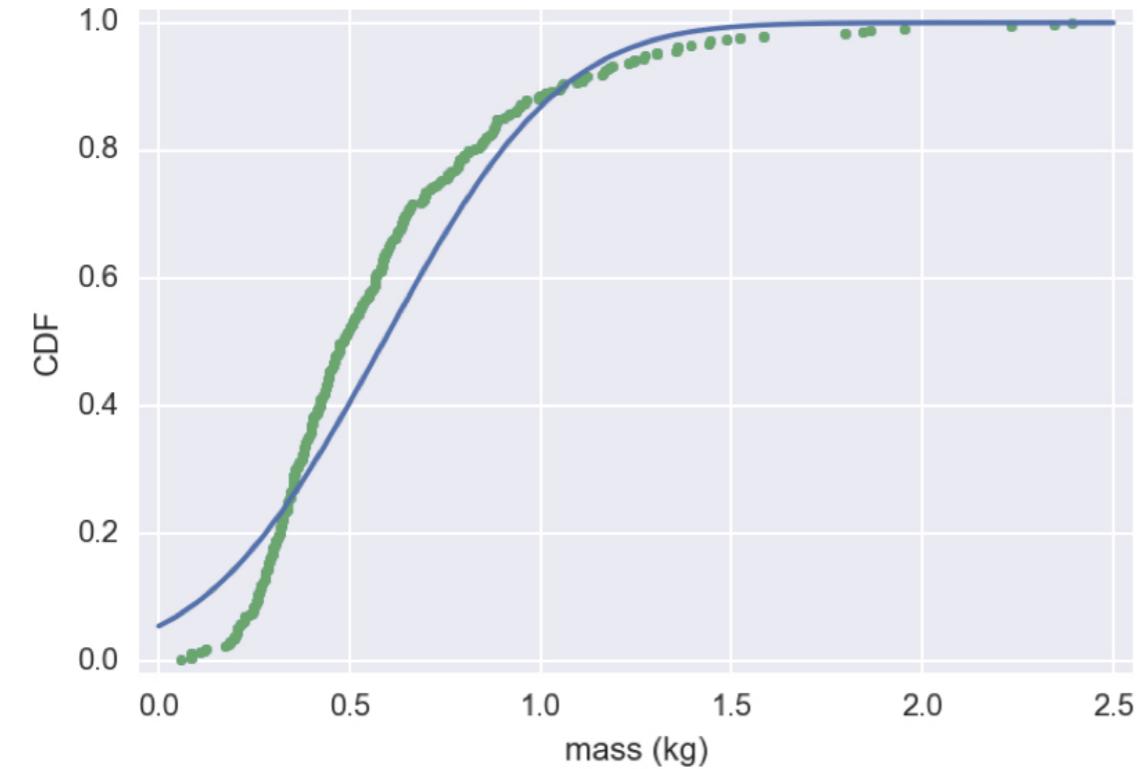
Source: Mass. Dept. of Environmental Protection

Length of MA large mouth bass



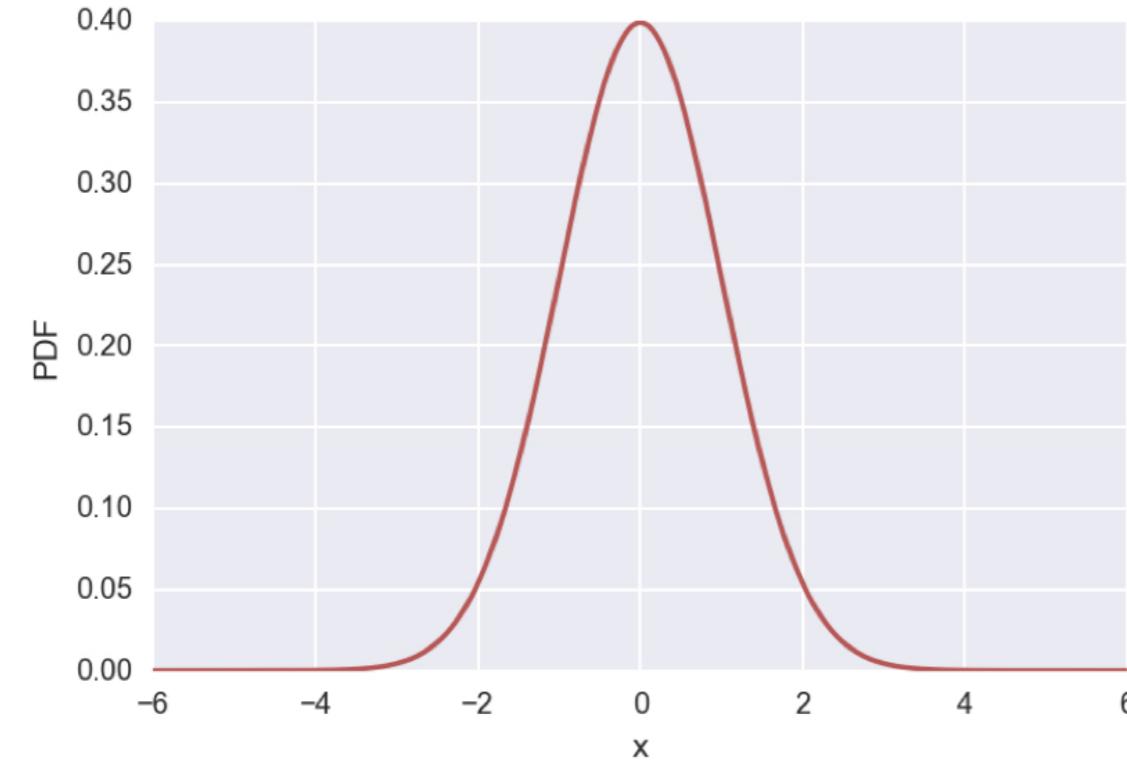
Source: Mass. Dept. of Environmental Protection

Mass of MA large mouth bass

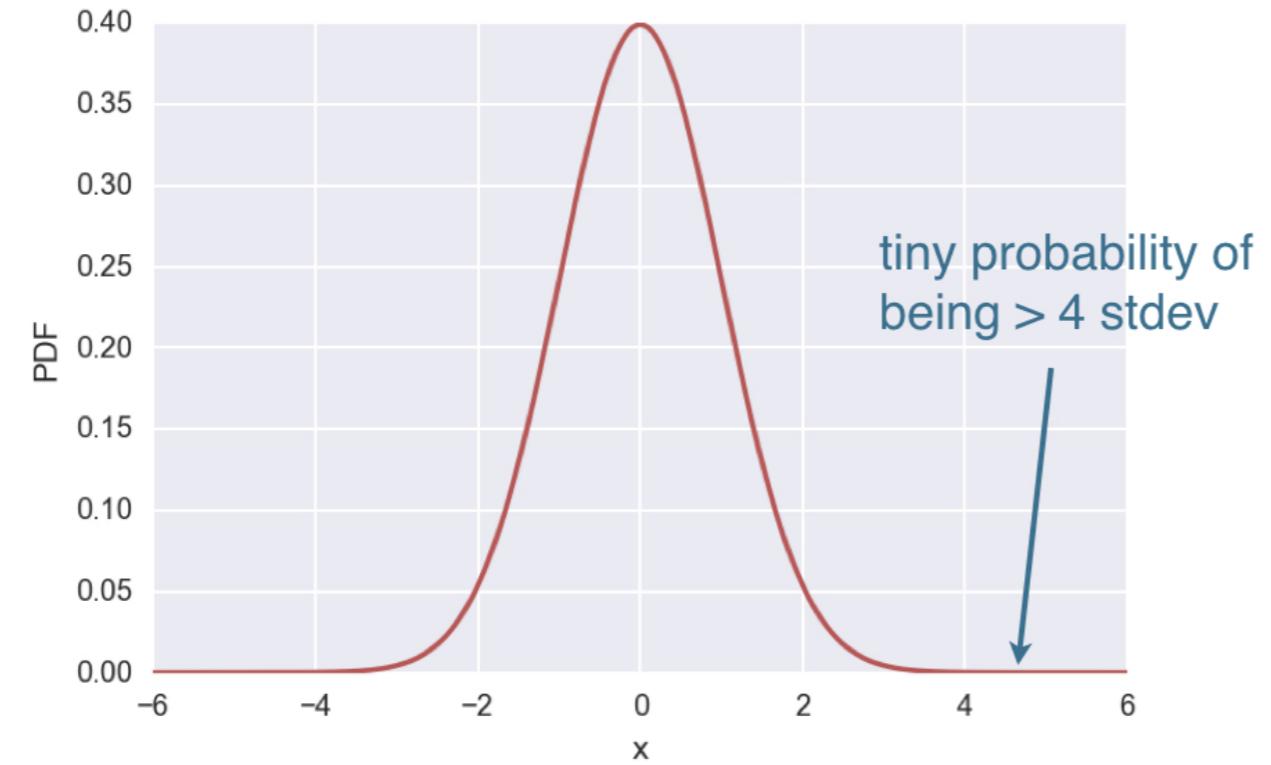


Source: Mass. Dept. of Environmental Protection

Light tails of the Normal distribution



Light tails of the Normal distribution

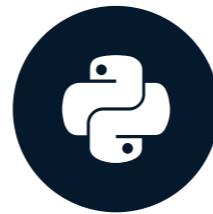


Let's practice!

STATISTICAL THINKING IN PYTHON (PART 1)

The Exponential distribution

STATISTICAL THINKING IN PYTHON (PART 1)



Justin Bois

Lecturer at the California Institute of
Technology

The Exponential distribution

- The waiting time between arrivals of a Poisson process is Exponentially distributed

The Exponential PDF



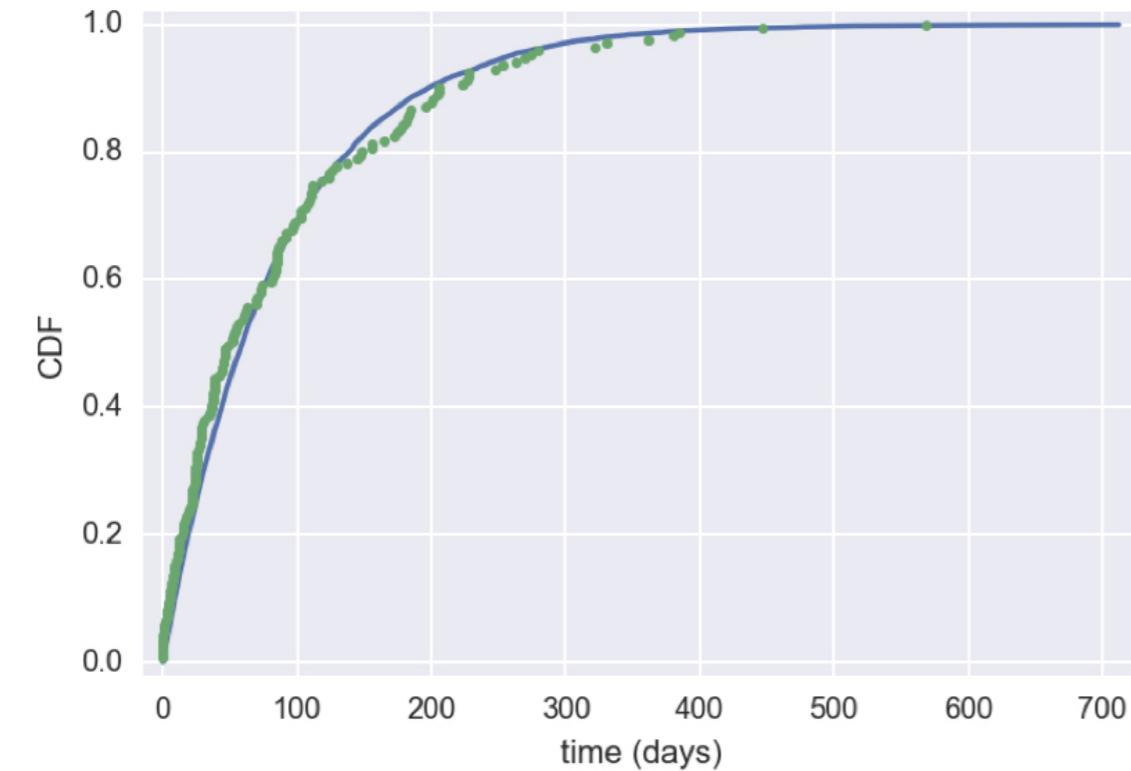
Possible Poisson process

- Nuclear incidents:
 - Timing of one is independent of all others

Exponential inter-incident times

```
mean = np.mean(inter_times)
samples = np.random.exponential(mean, size=10000)
x, y = ecdf(inter_times)
x_theor, y_theor = ecdf(samples)
_ = plt.plot(x_theor, y_theor)
_ = plt.plot(x, y, marker='.', linestyle='none')
_ = plt.xlabel('time (days)')
_ = plt.ylabel('CDF')
plt.show()
```

Exponential inter-incident times



Data Source: Wheatley, Sovacool, Sornette, Nuclear Events Database

Let's practice!

STATISTICAL THINKING IN PYTHON (PART 1)

Final thoughts

STATISTICAL THINKING IN PYTHON (PART 1)



Justin Bois

Lecturer at the California Institute of
Technology

You now can...

- Construct (beautiful) instructive plots
- Compute informative summary statistics
- Use hacker statistics
- Think probabilistically

In the sequel, you will...

- Estimate parameter values
- Perform linear regressions
- Compute confidence intervals
- Perform hypothesis tests

Let's practice!

STATISTICAL THINKING IN PYTHON (PART 1)