

# Dataframes and Series

EXPLORATORY DATA ANALYSIS IN PYTHON



**Allen Downey**  
Professor, Olin College

# Using data to answer questions

What is the average birth weight of babies in the United States?

- Find appropriate data, or collect it
- Read data in your development environment
- Clean and validate

# National Survey of Family Growth (NSFG)

NSFG data, from the National Center for Health Statistics

"nationally representative of women 15-44 years of age in the ... United States

"information on family life, marriage and divorce, pregnancy, infertility, use of contraception, and general and reproductive health."

The screenshot shows the official website for the National Survey of Family Growth (NSFG), hosted by the Centers for Disease Control and Prevention (CDC). The header includes the CDC logo and the tagline 'CDC 24/7: Saving Lives, Protecting People™'. A search bar for NCHS is also present. The main navigation bar identifies the 'National Center for Health Statistics'. Below this, the 'National Survey of Family Growth' is prominently displayed with its logo. A left-hand menu provides quick access to various resources: 'About NSFG', 'What's New', 'Questionnaires, Datasets, and Related Documentation', 'Key Statistics from NSFG', 'Publications and Information Products', 'Bibliography', 'Research Conferences', and 'NSFG Survey Participants'. The central content area offers a brief overview of the survey's purpose, which is to gather data on family life, marriage, divorce, pregnancy, and reproductive health. It also includes social media icons for Facebook, Twitter, and a general share button. On the right, a 'What's New' section lists recent updates, including 'Data Releases' (specifically '2013-2015 N') and 'Publications' (specifically 'Adoption-rel:').

# Reading data

```
import pandas as pd  
nsfg = pd.read_hdf('nsfg.hdf5', 'nsfg')  
type(nsfg)
```

```
pandas.core.frame.DataFrame
```

# Reading data

```
nsfg.head()
```

```
   caseid  outcome  birthwgt_lb1  birthwgt_oz1  prglngth  nbrnaliv  agecon  \
0   60418         1           5.0           4.0         40         1.0    2000
1   60418         1           4.0          12.0         36         1.0    2291
2   60418         1           5.0           4.0         36         1.0    3241
3   60419         6           NaN           NaN         33         NaN    3650
4   60420         1           8.0          13.0         41         1.0    2191

   agepreg  hpage1b  wgt2013_2015
0   2075.0     22.0   3554.964843
1   2358.0     25.0   3554.964843
2   3308.0     52.0   3554.964843
3        NaN     NaN   2484.535358
4   2266.0     24.0   2903.782914
```

# Columns and rows

```
nsfg.shape
```

```
(9358, 10)
```

```
nsfg.columns
```

```
Index(['caseid', 'outcome', 'birthwgt_lb1', 'birthwgt_oz1', 'prglngth',  
      'nbrnaliv', 'agecon', 'agepreg', 'hpagelb', 'wgt2013_2015'],  
      dtype='object')
```

# Columns and rows

## BIRTHWGT\_LB1 ( 46-47 )

Variable Type : raw

**BD-3** : How much did (BABY'S NAME/this 1st baby) weigh at birth? (POUNDS)

value	label	Total
.	INAPPLICABLE	2873
0-5	UNDER 6 POUNDS	936
6	6 POUNDS	1666
7	7 POUNDS	2146
8	8 POUNDS	1168
9-95	9 POUNDS OR MORE	474
98	Refused	1
99	Don't know	94
	Total	9358

# Each column is a Series

```
pounds = nsfg['birthwgt_lb1']  
type(pounds)
```

```
pandas.core.series.Series
```



# Each column is a series

```
pounds.head()
```

```
0    5.0  
1    4.0  
2    5.0  
3    NaN  
4    8.0  
Name: birthwgt_lb1, dtype: float64
```

# Let's start exploring!

EXPLORATORY DATA ANALYSIS IN PYTHON

# Clean and Validate

EXPLORATORY DATA ANALYSIS IN PYTHON



**Allen Downey**

Professor, Olin College

# Selecting columns

```
pounds = nsfg['birthwgt_lb1']
```

```
ounces = nsfg['birthwgt_oz1']
```

```
pounds.value_counts().sort_index()
```

```
0.0      6
1.0     34
2.0     47
3.0     67
4.0    196
5.0    586
6.0   1666
7.0   2146
8.0   1168
9.0    363
10.0    82
11.0    17
12.0     7
13.0     2
14.0     2
17.0     1
98.0     1
99.0    94
Name: birthwgt_lb1, dtype: int64
```

## BIRTHWGT\_LB1 ( 46-47 )

Variable Type : raw

**BD-3** : How much did (BABY'S NAME/this 1st baby) weigh at birth? (POUNDS)

value label		Total
.	INAPPLICABLE	2873
0-5	UNDER 6 POUNDS	936
6	6 POUNDS	1666
7	7 POUNDS	2146
8	8 POUNDS	1168
9-95	9 POUNDS OR MORE	474
98	Refused	1
99	Don't know	94
Total		9358

# Describe

```
pounds.describe()
```

```
count      6485.000000  
mean         8.055204  
std         11.178893  
min          0.000000  
25%          6.000000  
50%          7.000000  
75%          8.000000  
max         99.000000  
Name: birthwgt_lb1, dtype: float64
```

# Replace

```
pounds = pounds.replace([98, 99], np.nan)  
pounds.mean()
```

```
6.703286384976526
```

```
ounces.replace([98, 99], np.nan, inplace=True)
```



# Arithmetic with Series

```
birth_weight = pounds + ounces / 16.0  
birth_weight.describe()
```

```
count    6355.000000  
mean       7.120978  
std        1.422236  
min         0.000000  
25%        6.375000  
50%        7.187500  
75%        8.000000  
max       17.937500  
dtype: float64
```

# Let's practice!

EXPLORATORY DATA ANALYSIS IN PYTHON

# Filter and Visualize

EXPLORATORY DATA ANALYSIS IN PYTHON



**Allen Downey**

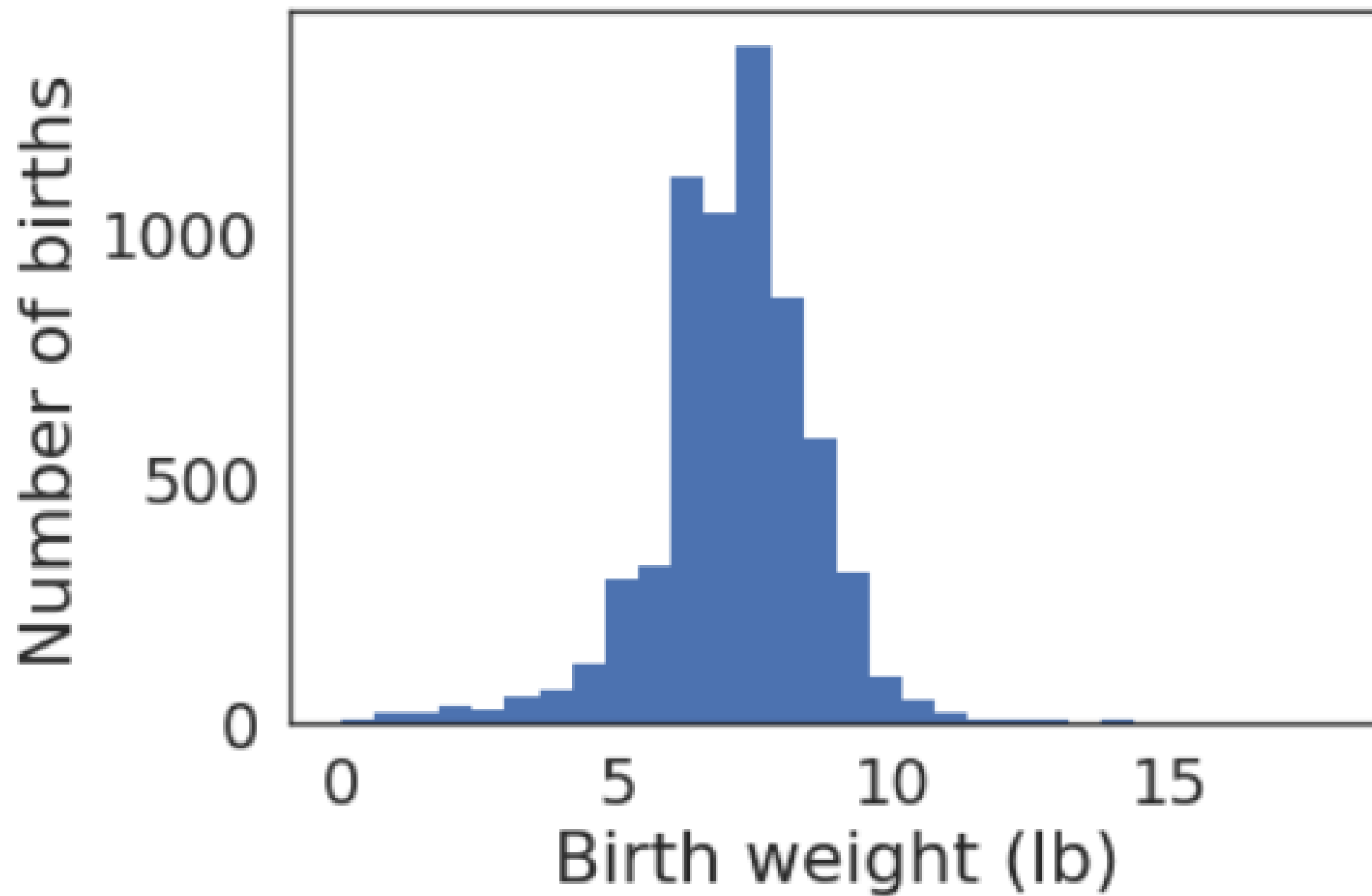
Professor, Olin College

# Histogram

```
import matplotlib.pyplot as plt
```

```
plt.hist(birth_weight.dropna(), bins=30)
```

```
plt.xlabel('Birth weight (lb)')  
plt.ylabel('Fraction of births')  
plt.show()
```



# Boolean Series

```
preterm = nsfg['prglnth'] < 37  
preterm.head()
```

```
0    False  
1     True  
2     True  
3     True  
4    False  
Name: prglnth, dtype: bool
```

# Boolean Series

```
preterm.sum()
```

```
3742
```

```
preterm.mean()
```

```
0.39987176747168196
```

# Filtering

```
preterm_weight = birth_weight[preterm]  
preterm_weight.mean()
```

```
5.577598314606742
```

```
full_term_weight = birth_weight[~preterm]  
full_term_weight.mean()
```

```
7.372323879231473
```



# Filtering

Other logical operators:

- `&` for AND (both must be true)
- `|` for OR (either or both can be true)

Example:

```
birth_weight[A & B]      # both true  
birth_weight[A | B]      # either or both true
```

# Resampling

- NSFG is not representative
- Some groups are "oversampled"
- We can correct using `resample_rows_weighted()`

# Finish it off!

EXPLORATORY DATA ANALYSIS IN PYTHON