



DS-UA 112

Introduction to Data Science

Lecture 10

Visualization III - transforming data

Reminders

- ▶ Homework 3
 - ▶ Due Friday October 18
- ▶ Project 1
 - ▶ Due Sunday October 20
- ▶ Cluster
 - ▶ Maintenance on Monday October 14
 - ▶ Please download any materials for assignments



Agenda

- ▶ Lecture 9
 - ▶ File size
 - ▶ File formats

```
factor_dict = {  
    'holiday': {  
        0: 'no',  
        1: 'yes'  
    },  
    'weekday': {  
        0: 'Sun',  
        1: 'Mon',  
        2: 'Tue',  
        3: 'Wed',  
        4: 'Thu',  
        5: 'Fri',  
        6: 'Sat'  
    },  
    'workingday': {  
        0: 'no',  
        1: 'yes'  
    },  
    'weathersit': {  
        1: 'Clear',  
        2: 'Mist',  
        3: 'Light',  
        4: 'Heavy'  
    }  
}
```

Agenda

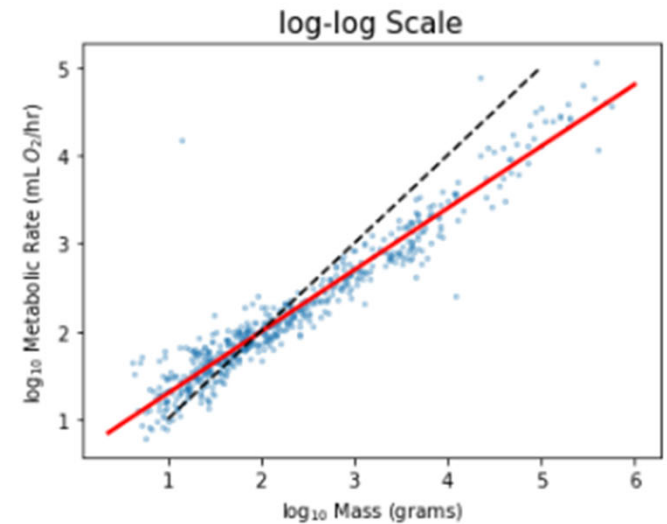
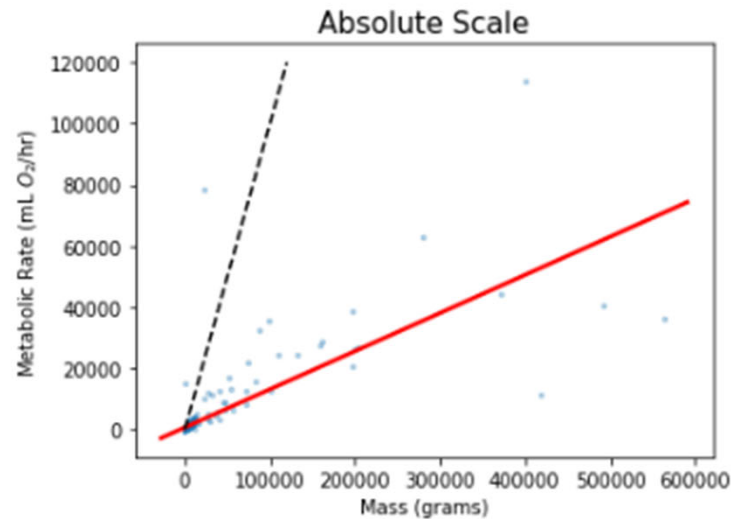
► Review

► Police Reports

Column	Description	Type
Incident Number	Number of incident created by Computer Aided Dispatch (CAD) program	Plain Text
Call Date/Time	Date and time of the incident/stop	Date & Time
Location	General location of the incident/stop	Plain Text
Incident Type	This is the occurred incident type created in the CAD program. A code signifies a traffic stop (T), suspicious vehicle stop (1196), pedestrian stop (1194) and bicycle stop (1194B).	Plain Text
Dispositions	Ordered in the following sequence: 1st Character = Race, as follows: A (Asian) B (Black) H (Hispanic) O (Other) W (White) 2nd Character = Gender, as follows: F (Female) M (Male) 3rd Character = Age Range, as follows: 1 (Less than 18) 2 (18-29) 3 (30-39), 4 (Greater than 40) 4th Character = Reason, as follows: I (Investigation) T (Traffic) R (Reasonable Suspicion) K (Probation/Parole) W (Wanted) 5th Character = Enforcement, as follows: A (Arrest) C (Citation) O (Other) W (Warning) 6th Character = Car Search, as follows: S (Search) N (No Search) Additional dispositions may also appear. They are: P - Primary case report M - MDT narrative only AR - Arrest report only (no case report submitted) IN - Incident report FC - Field Card CO - Collision investigation report MH - Emergency Psychiatric Evaluation TOW - Impounded vehicle 0 or 00000 – Officer made a stop of more than five persons	Plain Text
Location - Latitude	General latitude of the call. This data is only uploaded after January 2017	Number
Location - Longitude	General longitude of the call. This data is only uploaded after January 2017.	Number

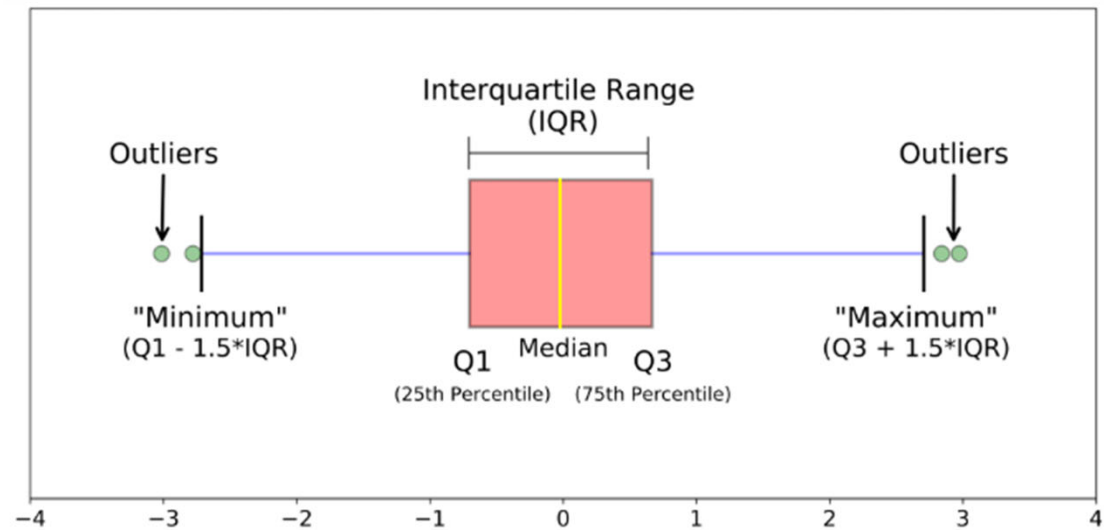
Agenda

- ▶ Lesson
 - ▶ Find effective visualizations for different types of data.
 - ▶ Transform data for visualization



Agenda

- Questions
 - Computing Outliers



Memory Leak

```
import json

# Note that this could cause our computer to run
# out of memory if the file
# is large. We have to verify that the file is small enough to
# read in beforehand.

with open('data/stops.json') as f:
    stops_dict = json.load(f)
```

Nested Dictionaries

```
# Load the data from JSON and assign column titles  
stops = pd.DataFrame(  
    stops_dict['data'],  
    columns=[c['name'] for c in stops_dict['meta']['view']['columns']])
```


Meta-Data

```
columns_to_drop = ['sid', 'id', 'position', 'created_at', 'created_meta',  
                  'updated_at', 'updated_meta', 'meta']  
  
# This function takes in a DF and returns a DF so we can use it for .pipe  
def drop_unneeded_cols(stops):  
    return stops.drop(columns=columns_to_drop)
```

Sentinels for Missing Values

```
# True if row contains at least one null value  
null_rows = stops.isnull().any(axis=1)  
  
stops[null_rows]
```

Typos

```
def clean_dispositions(stops):  
    cleaned = (stops['Dispositions']  
               .str.strip()  
               .str.rstrip(';')  
               .str.replace(';', ','))  
    return stops.assign(Dispositions=cleaned)
```

Pipes

```
tumble_after(  
  broke(  
    fell_down(  
      fetch(went_up(jack_jill, "hill"), "water"),  
      jack),  
    "crown"),  
  "jill"  
)
```

Pipes

```
on_hill = went_up(jack_jill, 'hill')
with_water = fetch(on_hill, 'water')
fallen = fell_down(with_water, 'jack')
broken = broke(fallen, 'jack')
after = tmple_after(broken, 'jill')
```

Pipes

```
jack_jill = pd.DataFrame()  
(jack_jill.pipe(went_up, 'hill')  
  .pipe(fetch, 'water')  
  .pipe(fell_down, 'jack')  
  .pipe(broke, 'crown')  
  .pipe(tumble_after, 'jill')  
)
```

Pipes

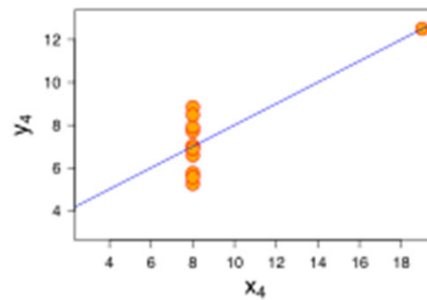
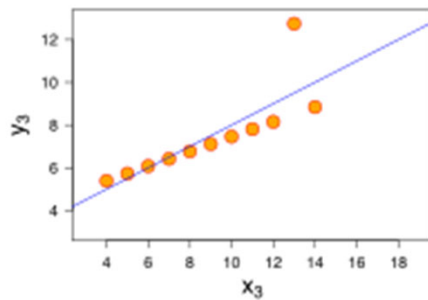
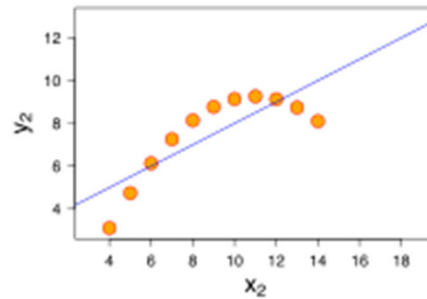
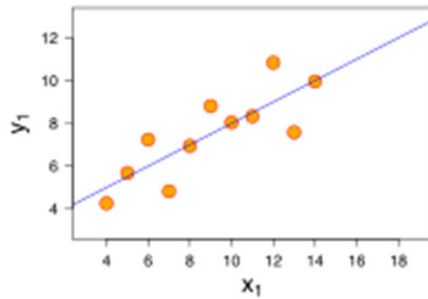
```
stops_final = (stops  
               .pipe(drop_unneeded_cols)  
               .pipe(clean_dispositions))
```

Pipes

```
stops_final = (stops  
               .pipe(drop_unneeded_cols)  
               .pipe(clean_dispositions))
```

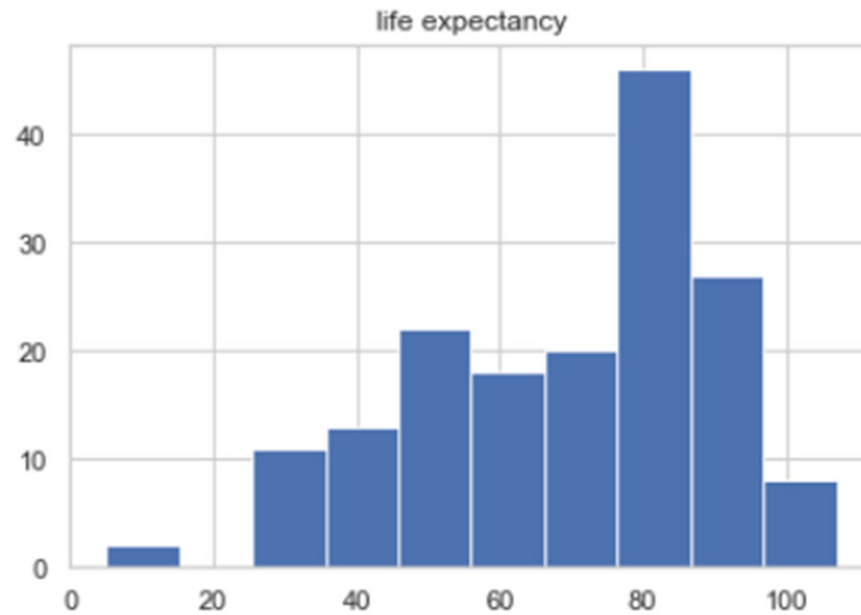

Why Visualization?

Property	Value	Accuracy
Mean of x	9	exact
Sample variance of x	11	exact
Mean of y	7.50	to 2 decimal places
Sample variance of y	4.125	± 0.003



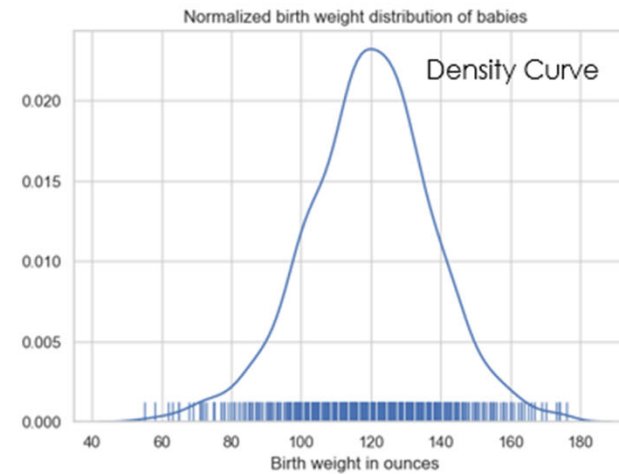
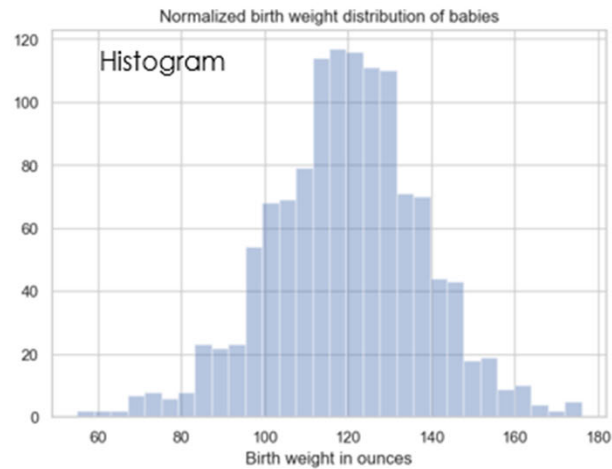
Finding the Right Visualization

- One Quantitative Variable



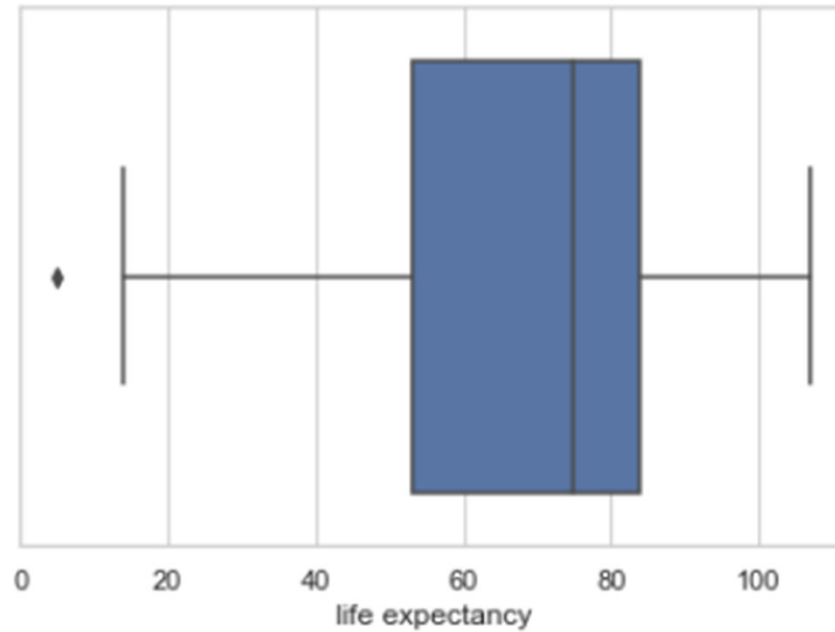
Finding the Right Visualization

► One Quantitative Variable



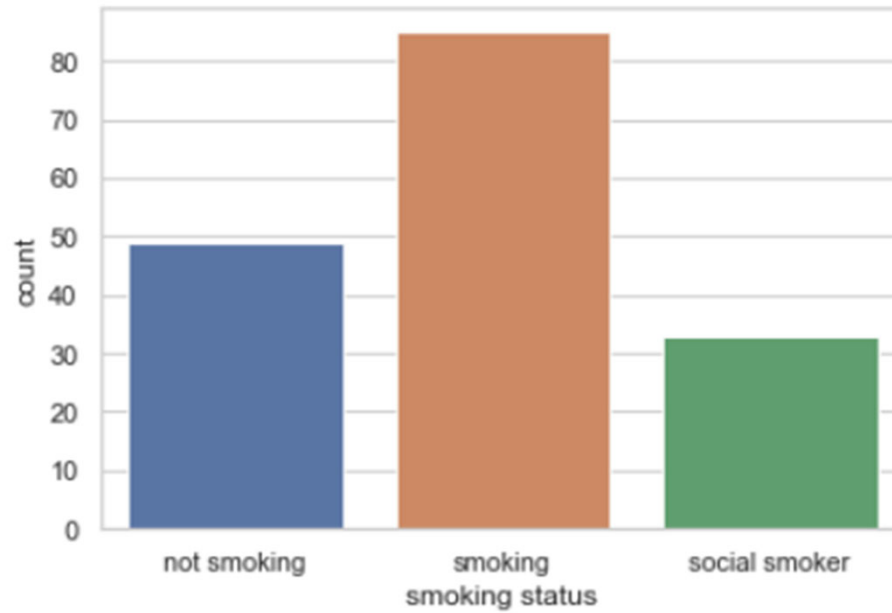
Finding the Right Visualization

► One Quantitative Variable



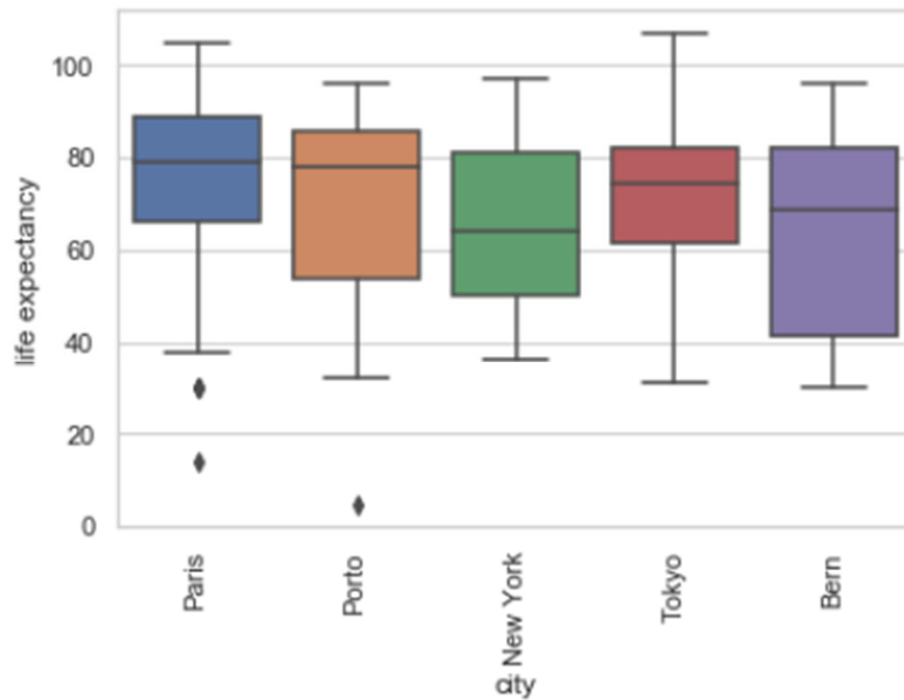
Finding the Right Visualization

- One Qualitative Variable



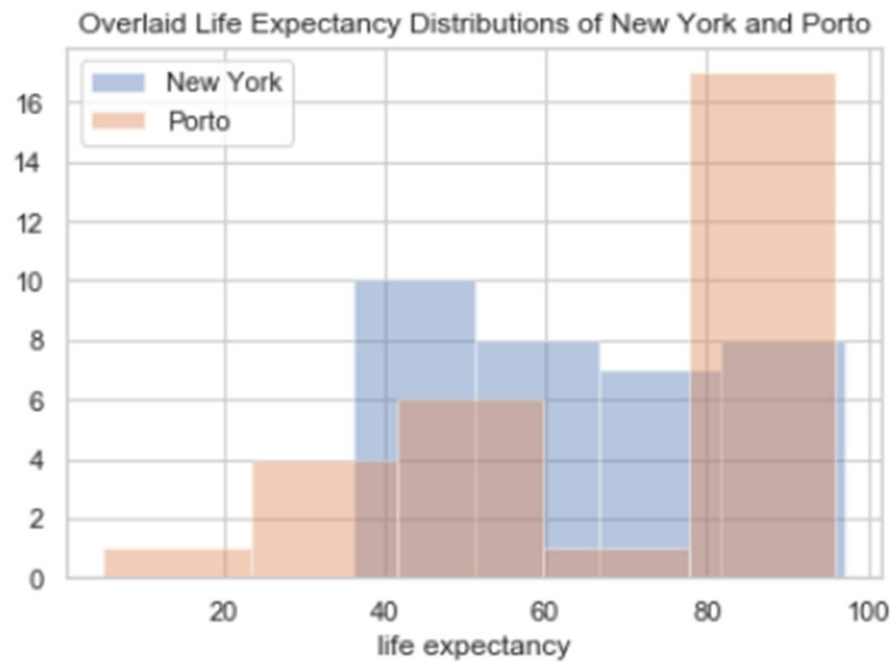
Finding the Right Visualization

- One Qualitative Variable and One Quantitative Variable



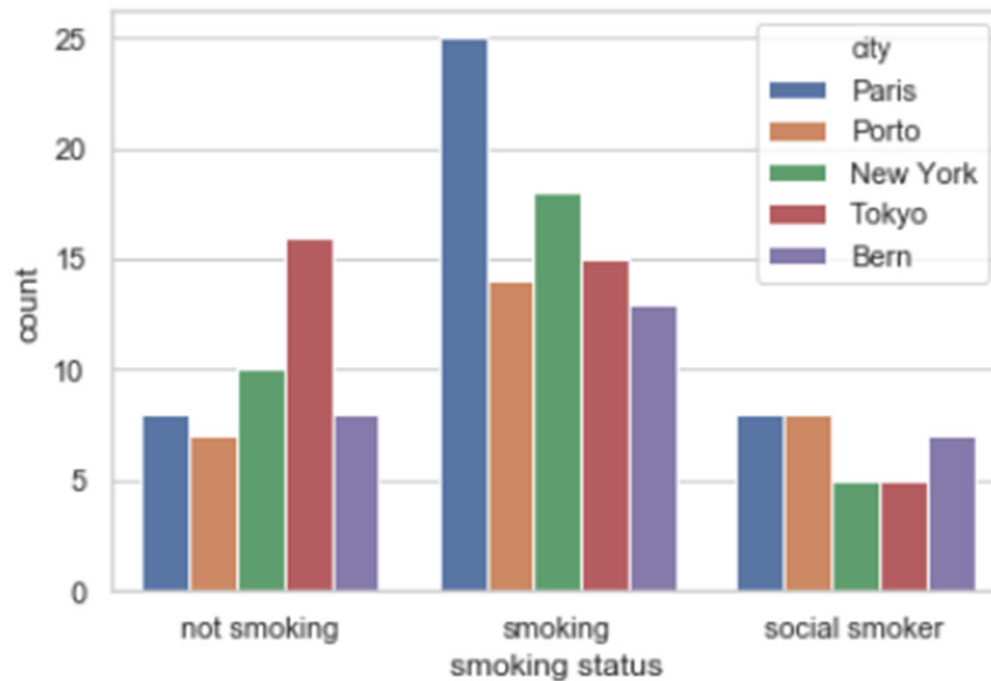
Finding the Right Visualization

- One Qualitative Variable and One Quantitative Variable



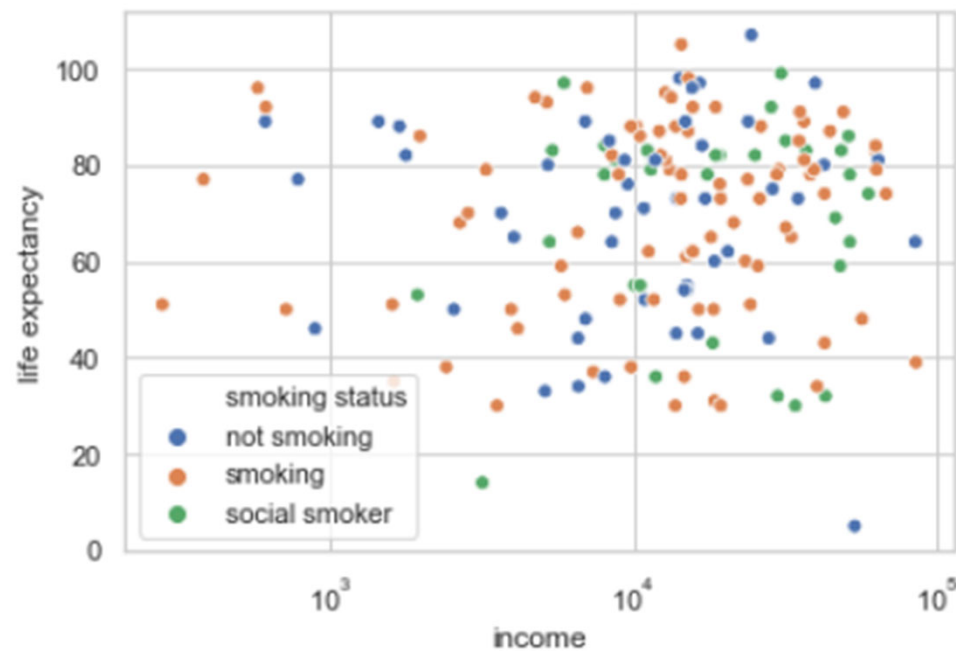
Finding the Right Visualization

► Multiple Qualitative Variables



Finding the Right Visualization

- ▶ Multiple Quantitative Variables
- ▶ Check out `sns.pairplot` !

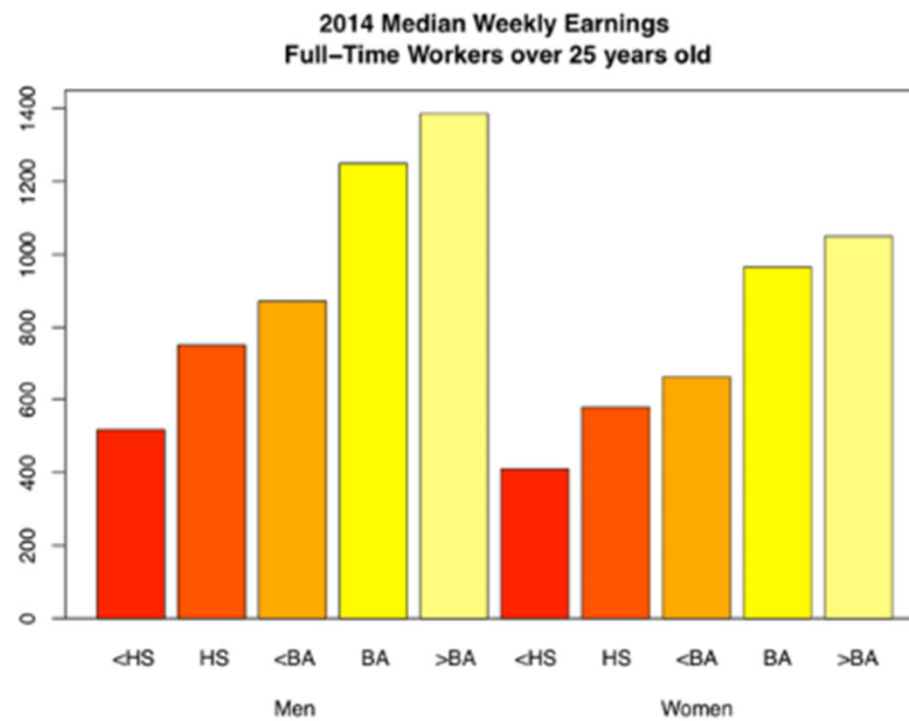


Finding the Right Visualization

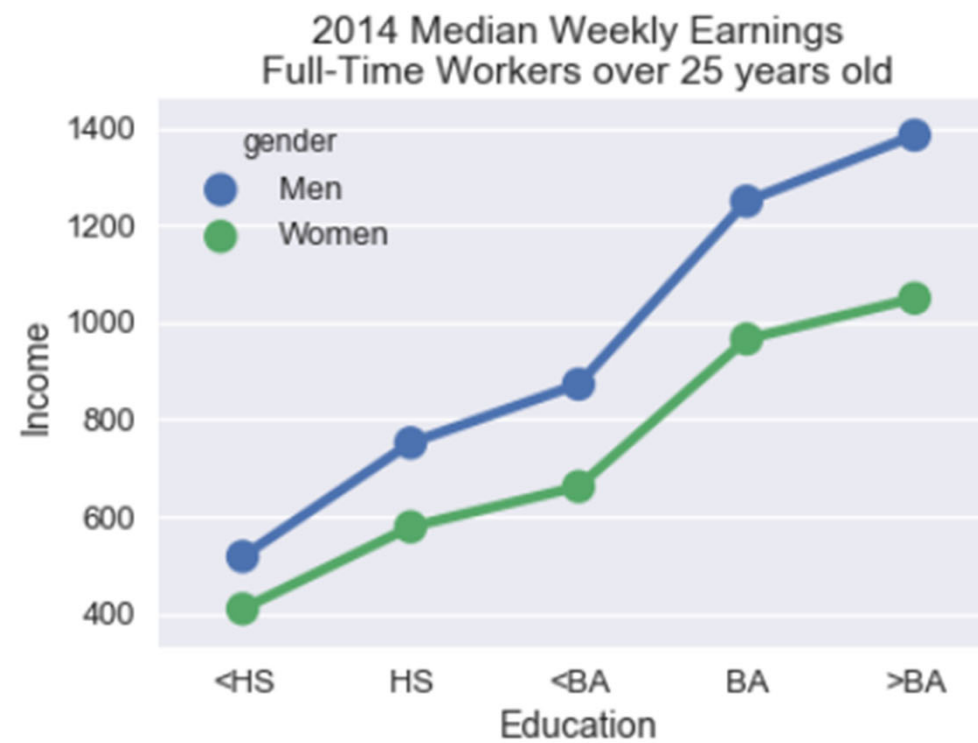
► Multiple Quantitative Variables



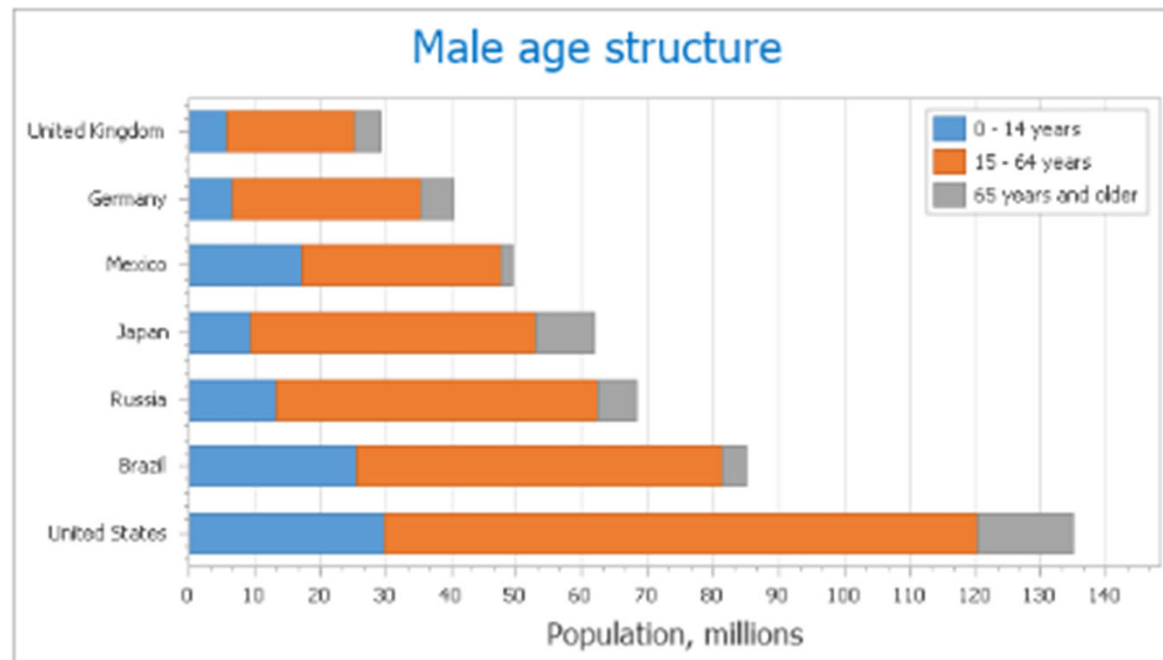
Conditioning



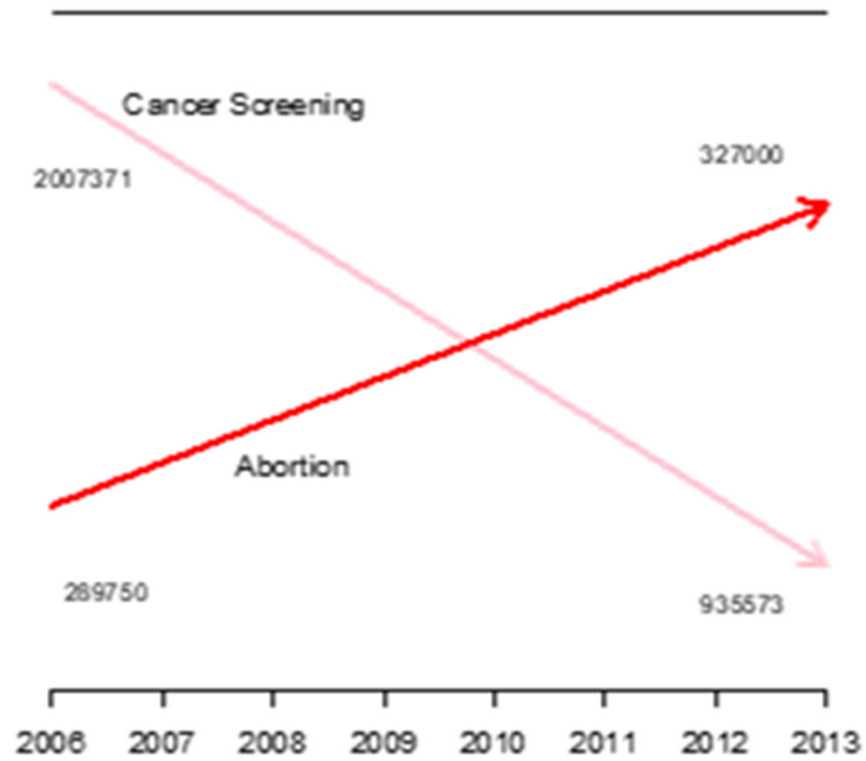
Conditioning



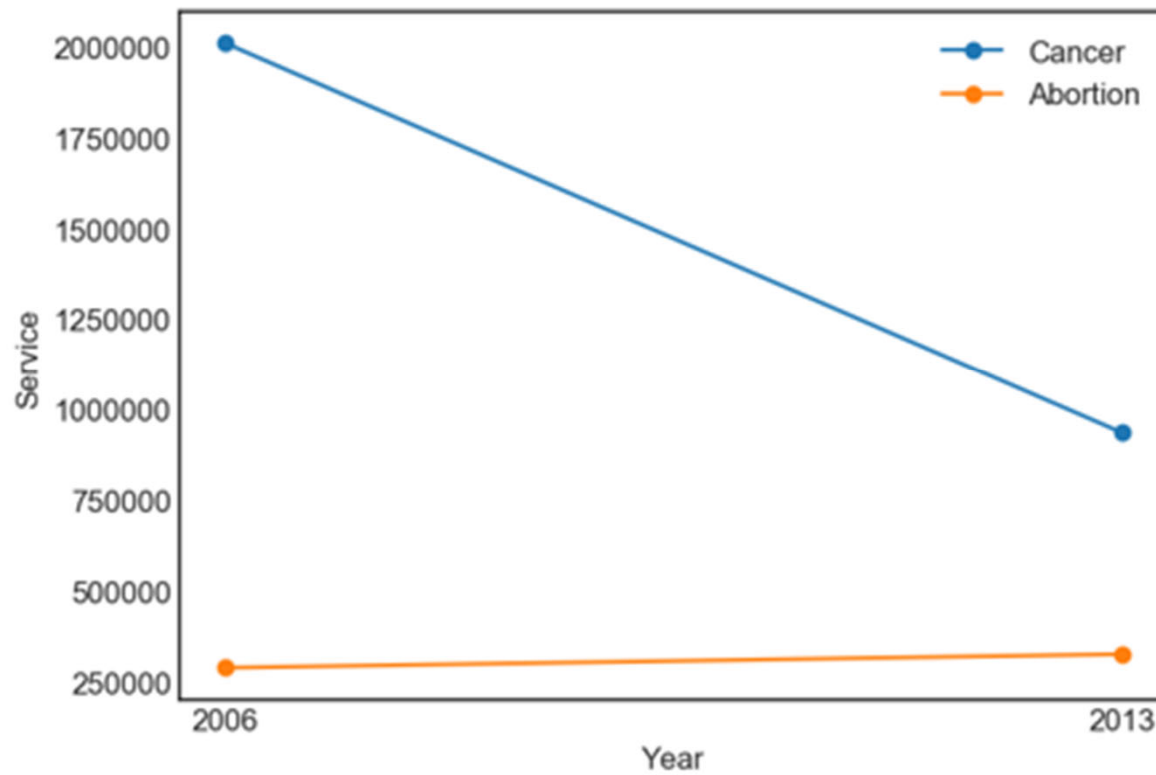
Jiggling Baseline



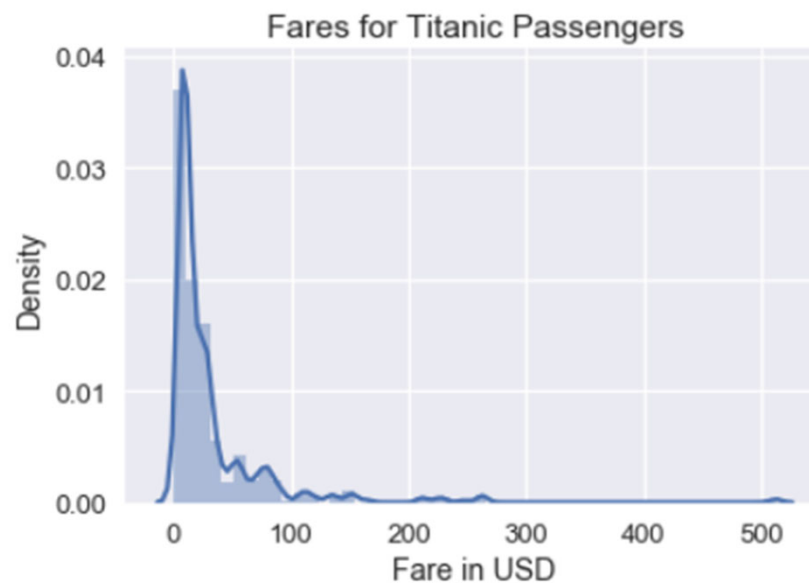
Scale



Scale

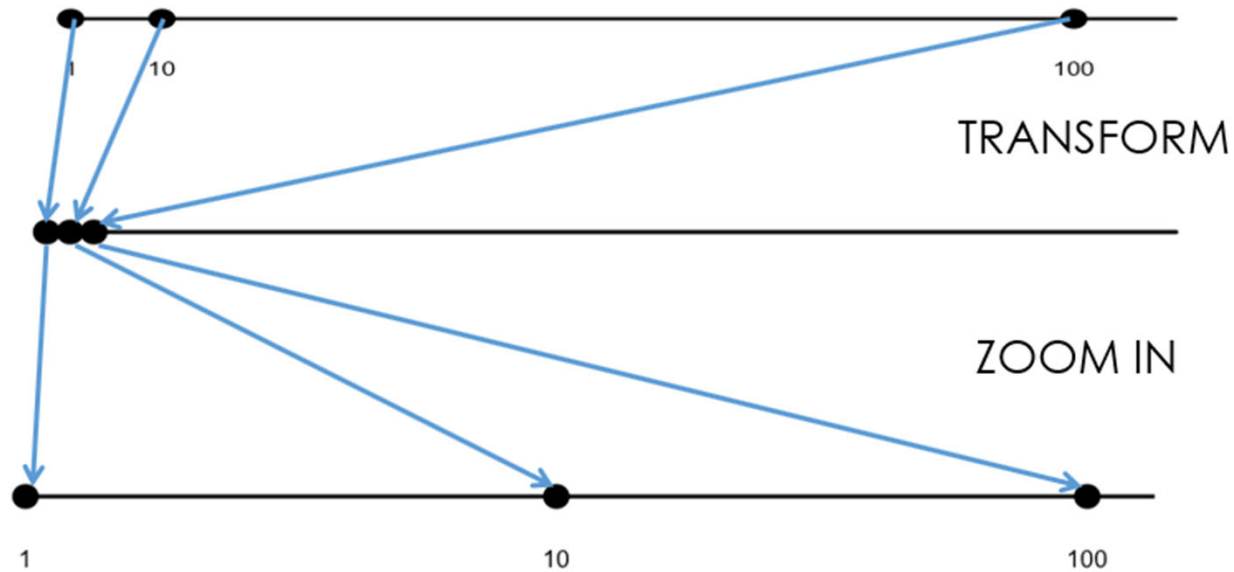


Transform with Logarithm



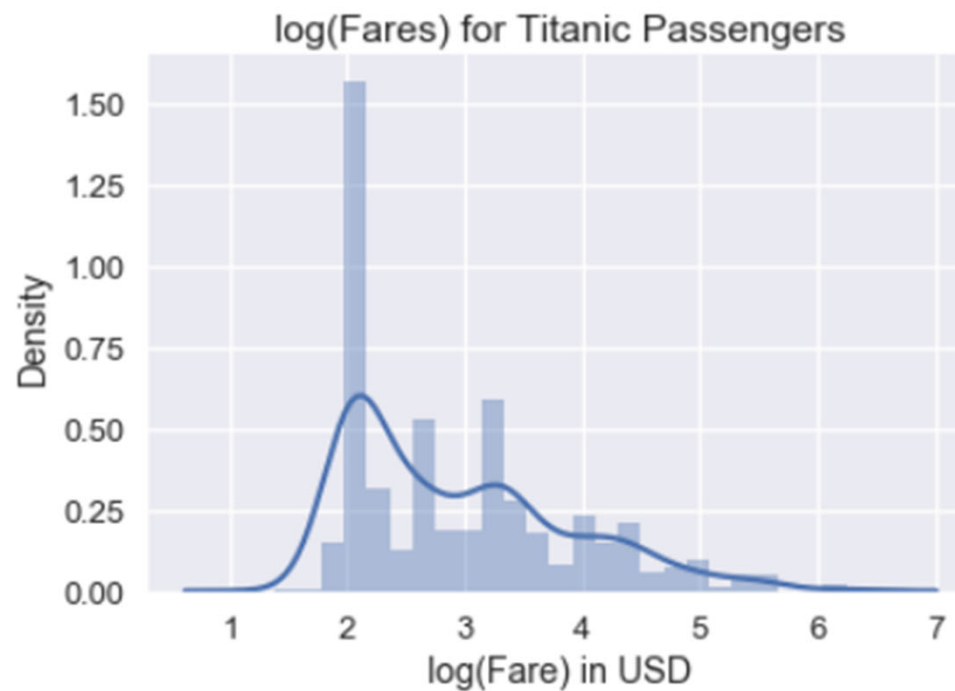
Transform with Logarithm

$$y = ax^k \rightarrow \log(y) = \log(a) + k \log(x)$$












Transform with Logarithm

value	log(value)
1	0.00
10	2.30
50	3.91
100	4.60
500	6.21
1000	6.90

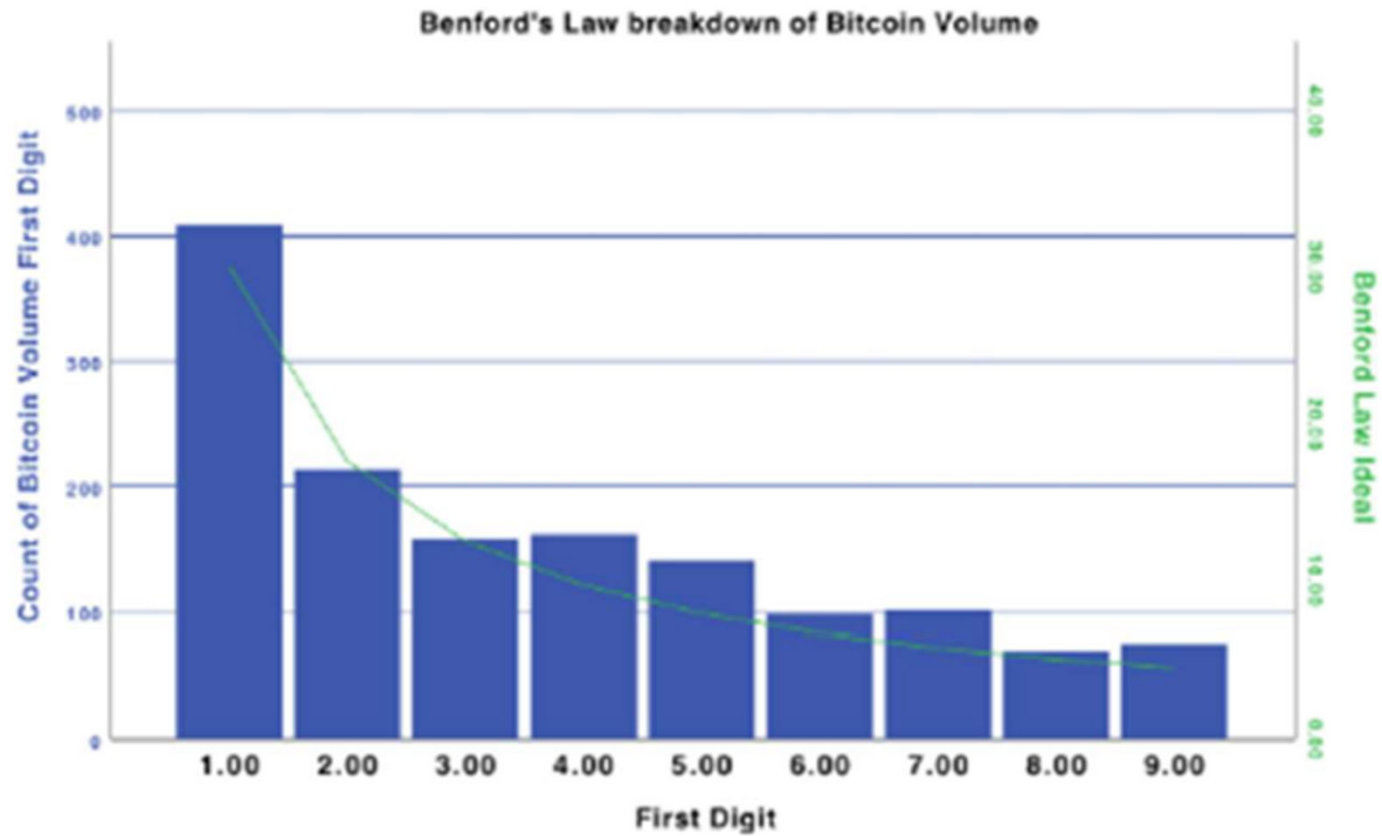


Benford's Law

$$P(d) = \log_{10}(d + 1) - \log_{10}(d)$$

d	$P(d)$	Relative size of $P(d)$
1	30.1%	
2	17.6%	
3	12.5%	
4	9.7%	
5	7.9%	
6	6.7%	
7	5.8%	
8	5.1%	
9	4.6%	

Benford's Law



Outliers

Removing Outliers with Mean

average

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + \dots + x_n) = \frac{1}{n} \sum_{i=1}^n x_i$$

variance

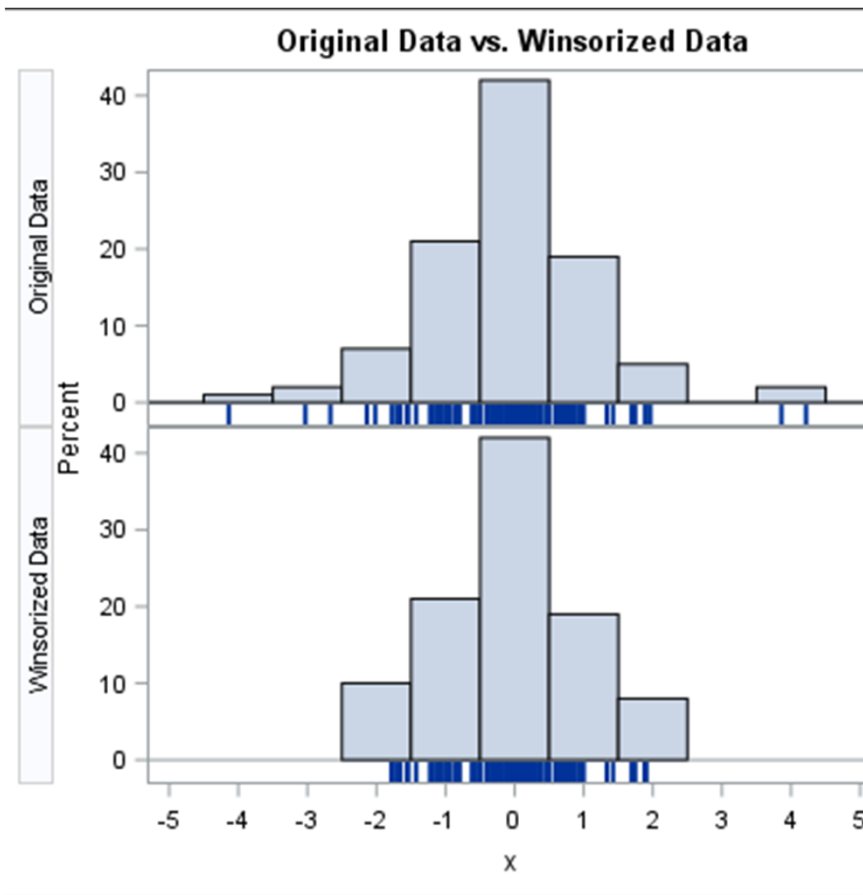
$$\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu)^2}{N}$$

outlier outside of range

$$(\bar{x} - a\sigma, \bar{x} + a\sigma)$$

where a a positive number

Removing Outliers with Percentiles



Take-Aways

► File