Visualization for Data Science
Exploratory Data Analysis II

Cholera in 1854 London

Cholera is an infectious disease that affects the small intestine.
In 1854, a deadly cholera outbreak swept through the Soho district in London. In the first week alone, more than 150 people died. It's 1854 (no computers!) and the government has hired your team to investigate the cause of the outbreak.

Your Task (work in small groups, 15 minutes):

- **Form a Hypothesis:** What do you suspect might be causing the outbreak?
- **Collect Evidence:** What kinds of information would you want to gather from the community to test your suspicion? (Think about *people, places, and timing*.)
- **Show the Evidence**
  - How could you present this information so patterns become clear to others?
  - What would your specific visualization be, describe the mark, channels used,
  - If you have time, go a step further and visualize it

Clicker Question – Select ALL

EVIDENCE:

Which types of information would be useful to collect to investigate the cholera outbreak?

A. Home addresses of cholera victims

B. Dates when symptoms began

C. Number of people in each household

D. Locations of nearby water pumps

E. Reports of deaths from other neighborhoods

Clicker Question – Select ONE

VISUALIZATION

Which visualization would make it easiest to spot the source of the outbreak?

A. A line graph showing deaths over time
B. A pie chart of total deaths by gender
C. A map showing deaths by household location
D. A bar chart of deaths by day of the week
E. A scatter plot showing the mortality rate relative to size of household

Cholera in 1854 London

"The deaths which occurred during this fatal outbreak of cholera are indicated in the accompanying map, as far as I could ascertain them."

– Dr. John Snow

http://www.who.int/wer/2010/wer8513.pdf

Cholera in 1854 London

"The deaths which occurred during this fatal outbreak of cholera are indicated in the accompanying map, as far as I could ascertain them."

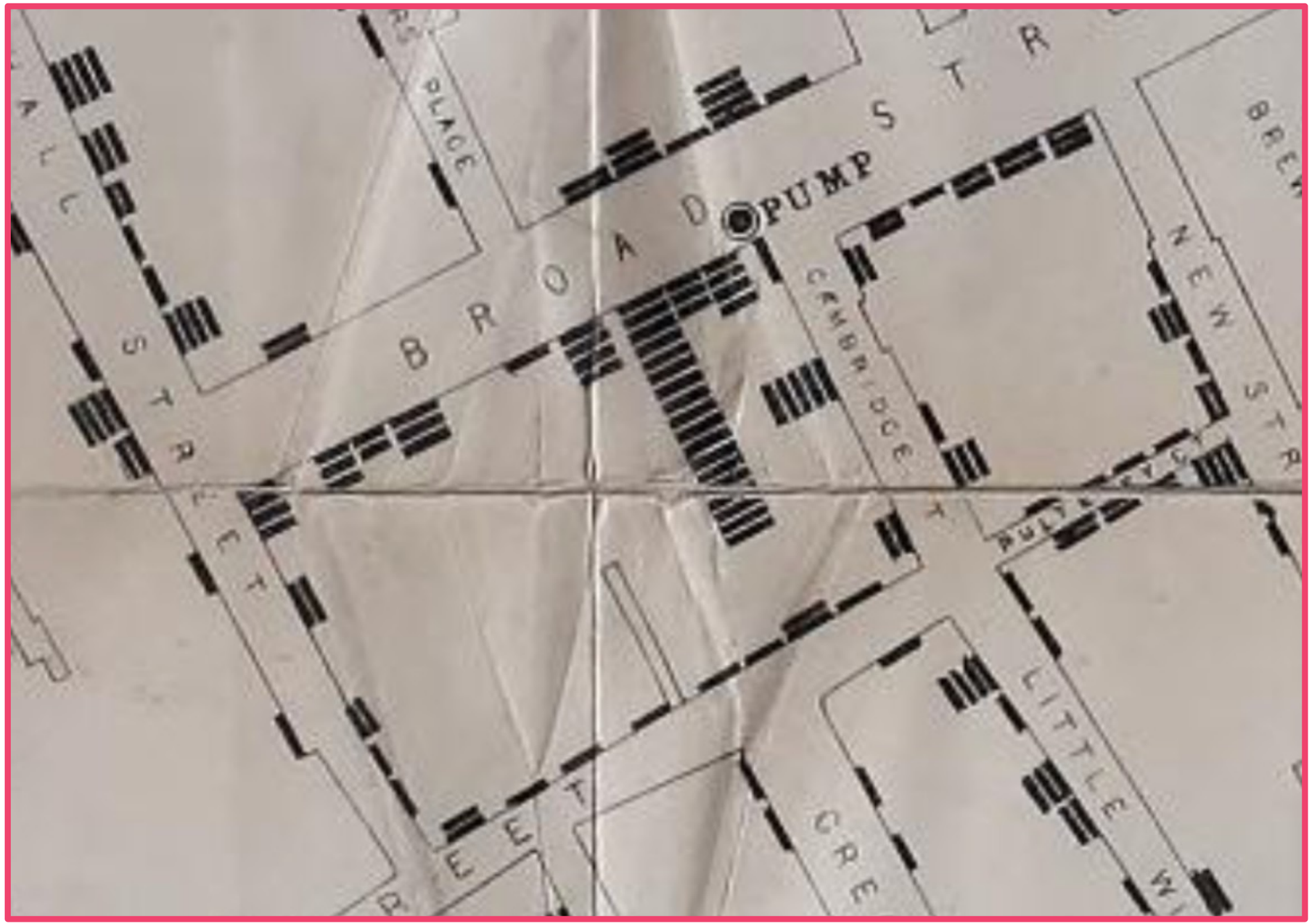
– Dr. John Snow

http://www.who.int/wer/2010/wer8513.pdf

Cholera in 1854 London

"The deaths which occurred during this fatal outbreak of cholera are indicated in the accompanying map, as far as I could ascertain them."

– Dr. John Snow

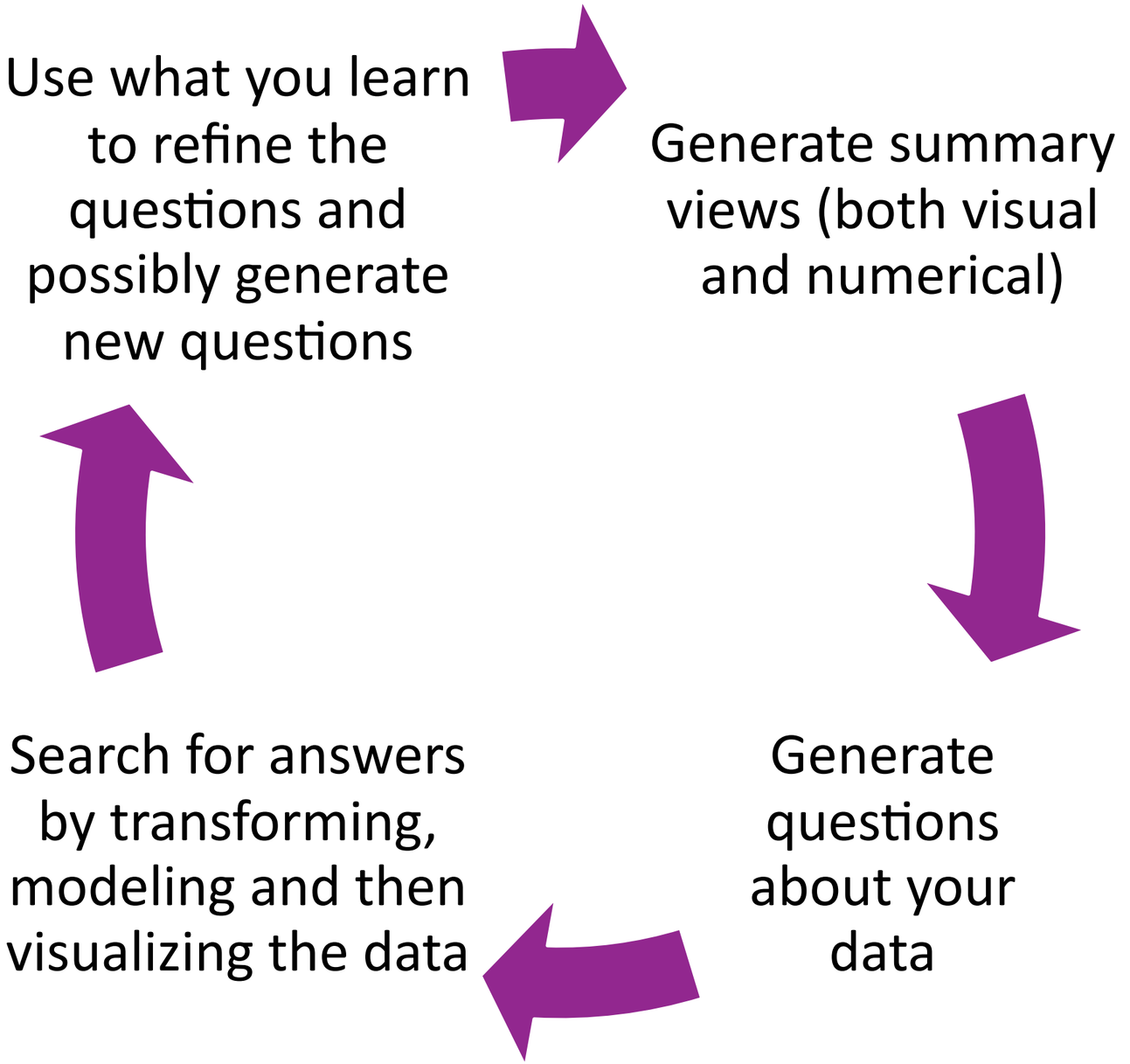


Exploratory Data Analysis (EDA)

Is a process that includes

- detection of mistakes
- checking of assumptions
- preliminary selection of appropriate models
- determining relationships among the explanatory variables, and
- assessing the direction and rough size of relationships between explanatory and outcome variables.

Exploratory Data Analysis Iterative Cycle



https://r4ds.had.co.nz/exploratory-data-analysis.html

Making the case for EDA – John Tukey (Father of EDA)

- Between 1900 – 1970s the status quo was
  - Formal theories of statistics
  - Rapid advancement in computers (e.g. Moore's law, applications, storage)
  - Data collection increases (based on measurements and growth of the scholar class)
  - Quantification vs. qualitative
- "Exposure" is critical
  - Numbers are great, formal processes are good
  - But the **flexibility** of an **informed human mind** is **irreplaceable**"
- We need strategies and techniques that
  - Facilitate the exploration of data by humans
  - Externalize representations of data
  - Support manipulation of data models

Categorizing Exploratory Data Analysis

Medium

- Numerical Statistical Summaries
- Visual Data Analysis

Attribute

- Univariate - one column at a time
- Multivariate – two or more variables at a time, looking for relationships.
- Attribute Type – categorical or quantitative

Role

- Outcome
- Explanatory

Administrivia – Office Hours (THIS WEEK ONLY)

TA Office Hours (online)

Monday 11:00am – 12pm

Wednesday 10:00 – 11am

Pop up in-person office hours Thursday 3 – 6 in LAB ROOM.

Instructor Office Hours

Monday and Wednesday 5 – 6pm (currently in this room)

~~Tuesdays 2 – 3pm ICCS 227~~

Participation is Clickers, not PL (unless otherwise stated)

Administrivia – Quiz Update

Quiz 4

- Wednesday and Thursday this week

Quiz 5

- Monday and Tuesday next week
- All tests will be hidden

Quiz 6

- Tuesday and Wednesday the following week
- All tests will be hidden

Project Group

- What is your group situation looking like

A. I have found a group
B. Wait we need groups?
C. I have found a group and we have multiple datasets
D. I don’t like people and so I refuse to find a group

Administrivia – Instructor Absences

Monday October 6th

- No in-person class. BUTTTTTTT
- Class 6A will be a programming file that looks into how to handle missing data.
- You can also use this time to work on your project

October 20th through October 27th

- Dr. K is in Gaborone for CompEd Conference
- October 20th Class8A. No in-person class. Interaction Programming File.
- October 22nd Class8B. Matt will lead the lecture and it will be half on Interaction and the other half will be on the Project.
- October 27th Class 9A. No in-person class. Recorded Zoom lecture

My apologies

Read through the rest of the slides. Make sure you understand the different visualizations and when they will be used.

Use the references (links in the lower-right corner) to aid your understanding.

We will NOT spend time in class describing each standard visualization so please do this work.

EDA with Hawks Dataset

Cooper’s hawk, photo by Mike Baird

Red-tailed hawk, photo by Don Sniegowski

Sharp-shinned hawk, photo by Tod Petit

Meet the Hawks Dataset

When presented with a new dataset, how do you start your exploration?
What questions should you ask?

1. How big is the dataset (i.e. rows and columns)
2. How many attributes are there?
3. What are the data types of each attribute?
4. What is the missing data situation?
5. Which attributes seem to be a best starting spot for exploration?

Let's start by systematically exploring our new dataset using pandas:

Hawks.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 908 entries, 0 to 907
Data columns (total 20 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Unnamed: 0    908 non-null    int64
 1   Month         908 non-null    int64
 2   Day           908 non-null    int64
 3   Year          908 non-null    int64
 4   CaptureTime   908 non-null    object
 5   ReleaseTime   907 non-null    object
 6   BandNumber    908 non-null    object
 7   Species       908 non-null    object
 8   Age           908 non-null    object
 9   Sex           332 non-null    object
 10  Wing          907 non-null    float64
 11  Weight        898 non-null    float64
 12  Culmen        901 non-null    float64
 13  Hallux        902 non-null    float64
 14  Tail          908 non-null    int64
 15  StandardTail  571 non-null    float64
 16  Tarsus        75 non-null     float64
 17  WingPitFat    77 non-null     float64
 18  KeelFat       567 non-null    float64
 19  Crop          565 non-null    float64
dtypes: float64(9), int64(5), object(6)
memory usage: 142.0+ KB
```

Hawks.sample(10)

```
1 # First few rows to understand the data structure
2 hawks.head()
3 # or you can use sample so it doesn't just give what is at the top
4 hawks.sample(10)
```

Unnamed: 0	Month	Day	Year	CaptureTime	ReleaseTime	BandNumber	Species	Age	Sex	Wing
677	10	25	2001	11:04		2003-99334	SS	A	F	200.0
167	10	24	1994	9:05		1423-16208	SS	I	F	209.0
491	11	10	1999	10:15		1177-04622	RT	I	NaN	351.0
512	9	18	2000	12:05		1207-72632	RT	I	NaN	394.0
206	11	21	1994	10:40		1387-79166	RT	I	NaN	390.0
876	10	21	2003	13:02		1177-04762	RT	I	NaN	370.0
374	11	6	1997	11:22	11:41	1343-78476	SS	I	F	202.0
560	10	9	2000	14:38		1177-04648	RT	I	NaN	377.0
759	10	13	2002	11:33		1177-04720	RT	A	NaN	406.0
670	10	19	2001	11:53		1142-19256	SS	A	M	167.0

creates a DataFrame of booleans (`True` where a value is missing, `False` otherwise).

```
missing_summary = hawks.isnull().sum()
```

```
missing_summary[missing_summary > 0]
```

This prints the number of missing values, **but only for columns that actually have missing values** (ignores columns with 0 missing values).

```
missing_summary.sum()
```

adds up all the missing counts across columns → the **total number of missing entries in the whole dataset**.

```
Missing values per column:
ReleaseTime       1
Sex             576
Wing              1
Weight           10
Culmen            7
Hallux            6
StandardTail    337
Tarsus          833
WingPitFat      831
KeelFat         341
Crop            343
dtype: int64

Total missing values: 3286
Percentage of complete rows: 3.9%
```

```
{(hawks.dropna().shape[0] / hawks.shape[0] * 100
```

removes any row that contains at least one missing value.

gives the number of remaining rows (rows without missing values).

Missing Data ☹

- Total number of missing values is 3286.
- Percentage of complete rows is <4%
- BIG PROBLEM
  - Today we will avoid – next week we will address
  - Today we will remove the worst offending columns (e.g. Tarsus, WingPitFat, etc.)

```
Total missing values: 24
Percentage of complete rows: 98.1%
```

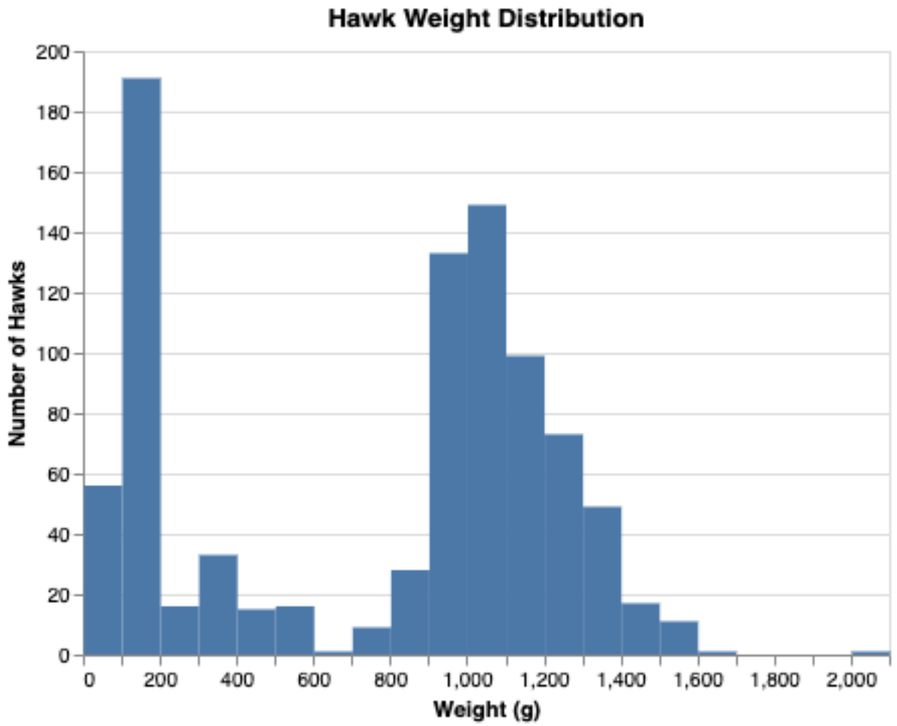
Data Attributes

Column	Description	Attribute Type
[redacted]		
species	hawk's species	Nominal
age	age group: A=Adult or I=Immature	Ordinal
[redacted]		
weight	Body weight (in gm)	Quantitative
[redacted]		

Understanding Distributions

How does a quantitative variable distribute across our observed data?”

Go and create this visualization, the specs are in the workbook



A. Still working
B. Stuck
C. Done

The Analytical Tension:
Should we investigate this STATISTICALLY or VISUALLY

🔢 **Statistical Approach**

⚡ Fast, precise numbers

 Good for exact values

Good for Hypothesis testing

📊 **Visual Approach**

👁️ Intuitive patterns

🔍 Good for exploration

Good for Pattern recognition

```
Color = scale(scheme= ‘red’)
axis = axis(ticks = False, tickCount = 10)
bin = alt.Bin(maxBins = 100)
```

Statistical Approach – describe()

```python
# 1. Overall distribution characteristics
print("Overall weight distribution:")
print(hawks_clean['weight'].describe())
```

```
Overall weight distribution:
count     898.000000
mean      772.080178
std       462.311760
min        56.000000
25%       185.000000
50%       970.000000
75%      1120.000000
max      2030.000000
Name: weight, dtype: float64
```

Statistical Approach – Counts by Nominal Attributes

```python
# 2. Investigate potential grouping variables
print("Available categorical variables:")
categorical_cols = hawks_clean.select_dtypes(include=['object']).columns

for col in categorical_cols:
    print(f"\n{col}: {hawks_clean[col].nunique()} unique values")
    print(hawks_clean[col].value_counts())
```

counts how many unique categories it has.

counts how many rows fall into each category.

```
Available categorical variables:

species: 3 unique values
species
RT    577
SS    261
CH     70
Name: count, dtype: int64

age: 2 unique values
age
I    684
A    224
Name: count, dtype: int64
```

Statistical Approach – describe() by categories

In [11]:
```python
# 3. Group-wise statistical analysis
# Hypothesis 1: Does species explain the distribution?
species_stats = hawks_clean.groupby('species')['weight'].describe()
print("Weight statistics by species:")
print(species_stats)
```

```
Weight statistics by species:
         count         mean         std    min    25%     50%      75%     max
species
CH        70.0   420.485714  162.031643   56.0  335.0   377.5   505.00  1119.0
RT       572.0  1094.430070  189.210250  101.0  980.0  1070.0  1210.00  2030.0
SS       256.0   147.968750   80.652675   85.0  100.0   155.0   177.75  1094.0
```

In [12]:
```python
# 3. Group-wise statistical analysis
# Hypothesis 2: Does age explain the distribution?
age_stats = hawks_clean.groupby('age')['weight'].describe()
print("Weight statistics by age:")
print(age_stats)
```

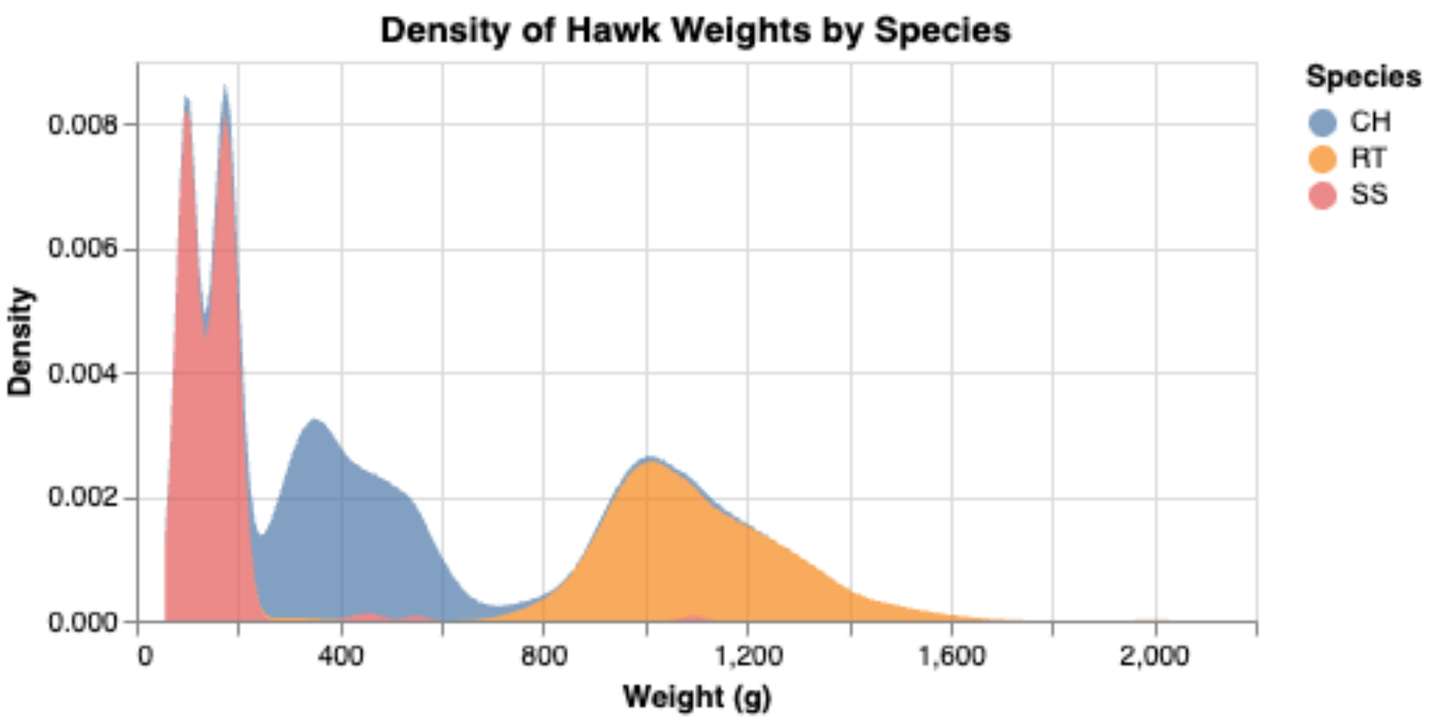
```
Weight statistics by age:
     count         mean         std   min    25%    50%     75%     max
age
A    221.0  747.366516  493.883991  56.0  185.0  960.0  1140.0  1670.0
I    677.0  780.147710  451.617738  85.0  188.0  971.0  1120.0  2030.0
```

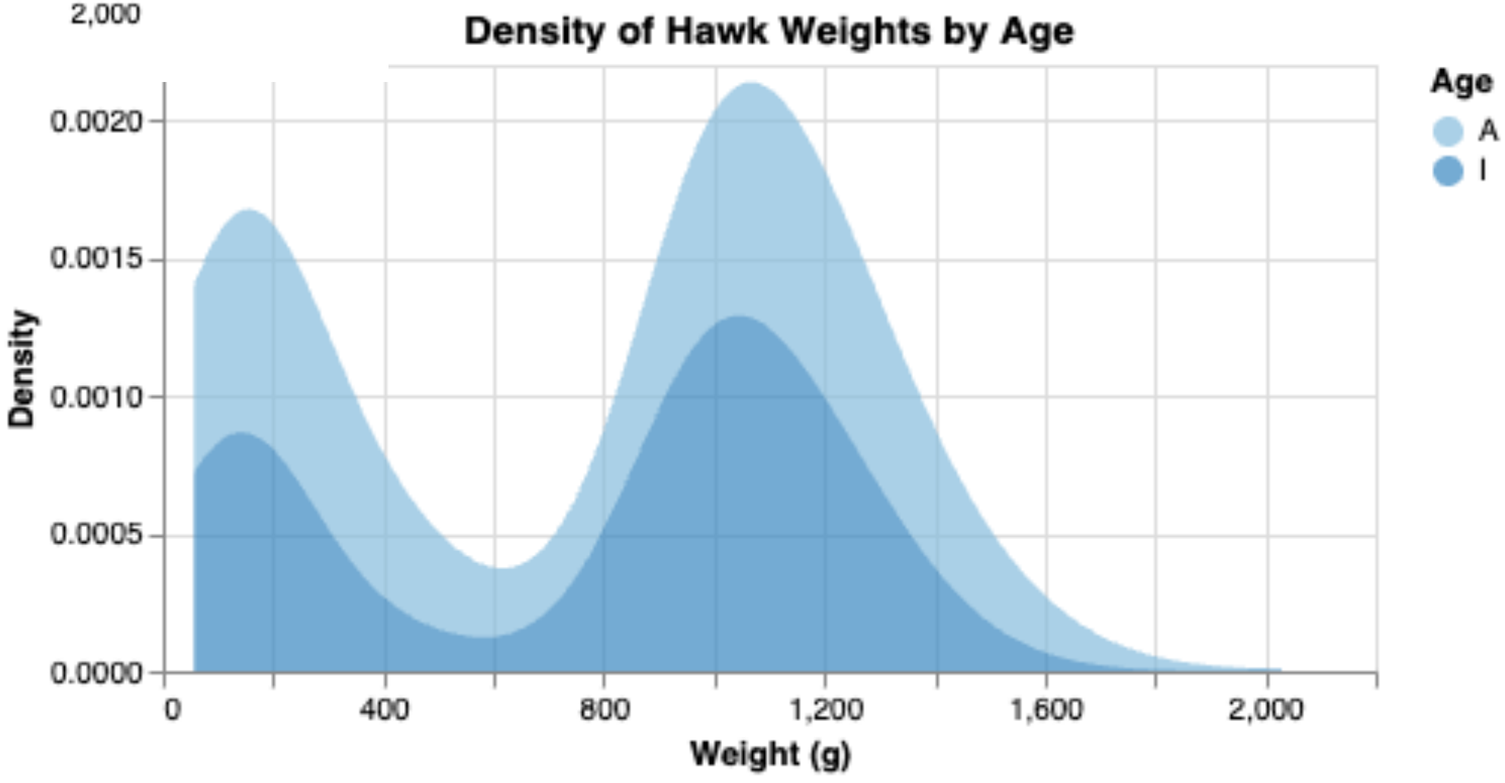
Observation – What do you notice

Visual Approach – Create the layered density charts for weight by species and then by age following the instructions laid out in the notebook

A. Still working on species
B. Done with species & working on age
C. Stuck
D. Done with both

```
density_by_species = alt.Chart(hawks_clean).transform_density(
    'weight',
    as_=['weight', 'density'],
    groupby=['species']  # Key: group by categorical variable
).mark_area(
    opacity=0.7,
).encode(
    x=alt.X('weight:Q', title='Weight (g)'),
    y=alt.Y('density:Q', title='Density'),
    color = alt.Color('species:N', title='Species')
).properties(
    width=400,
    height=200
).properties(
    title="Density of Hawk Weights by Species"
)
density_by_species
```


Density of Hawk Weights by Species
Species
CH
RT
SS
Density
0.008
0.006
0.004
0.002
0.000
0
400
800
1,200
1,600
2,000
Weight (g)


Density of Hawk Weights by Age
Age
A
I
Density
0.0020
0.0015
0.0010
0.0005
0.0000
0
400
800
1,200
1,600
2,000
Weight (g)

The real power comes from using **both approaches together** for deeper understanding:

```python
# Start with visual exploration → Species clearly explains multimodal distribution
# Now use statistics to quantify the visual findings

species_summary = hawks_clean.groupby('species')['weight'].agg([
    'count', 'mean', 'std', 'min', 'max'
]).round(1)

print("Quantified species differences:")
print(species_summary)
```

```
Quantified species differences:
         count    mean    std    min     max
species
CH          70   420.5  162.0   56.0  1119.0
RT         572  1094.4  189.2  101.0  2030.0
SS         256   148.0   80.7   85.0  1094.0
```

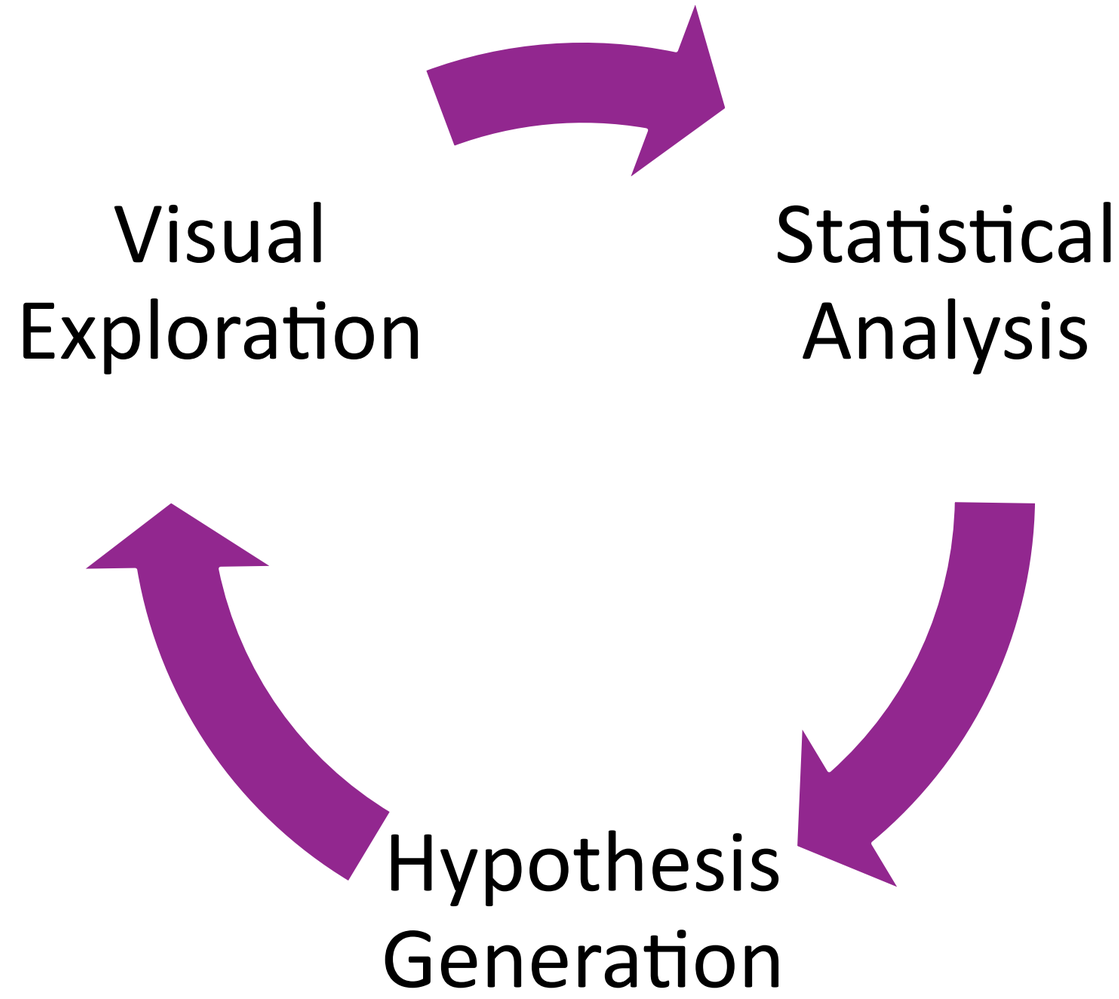
```python
# The visual analysis suggested we should investigate age differences too
# Let's get statistical confirmation:

detailed_stats = hawks_clean.groupby(['species', 'age'])['weight'].agg([
    'count', 'mean', 'std'
]).round(1)

print("Detailed breakdown by species AND age:")
print(detailed_stats)
```

```
Detailed breakdown by species AND age:
             count    mean    std
species age
CH      A       32   450.3  197.6
        I       38   395.3  121.8
RT      A      121  1161.4  194.6
        I      451  1076.5  183.9
SS      A       68   150.4   40.3
        I      188   147.1   91.0
```

The Optimal EDA Workflow¶



Decision Framework: When to Use What?

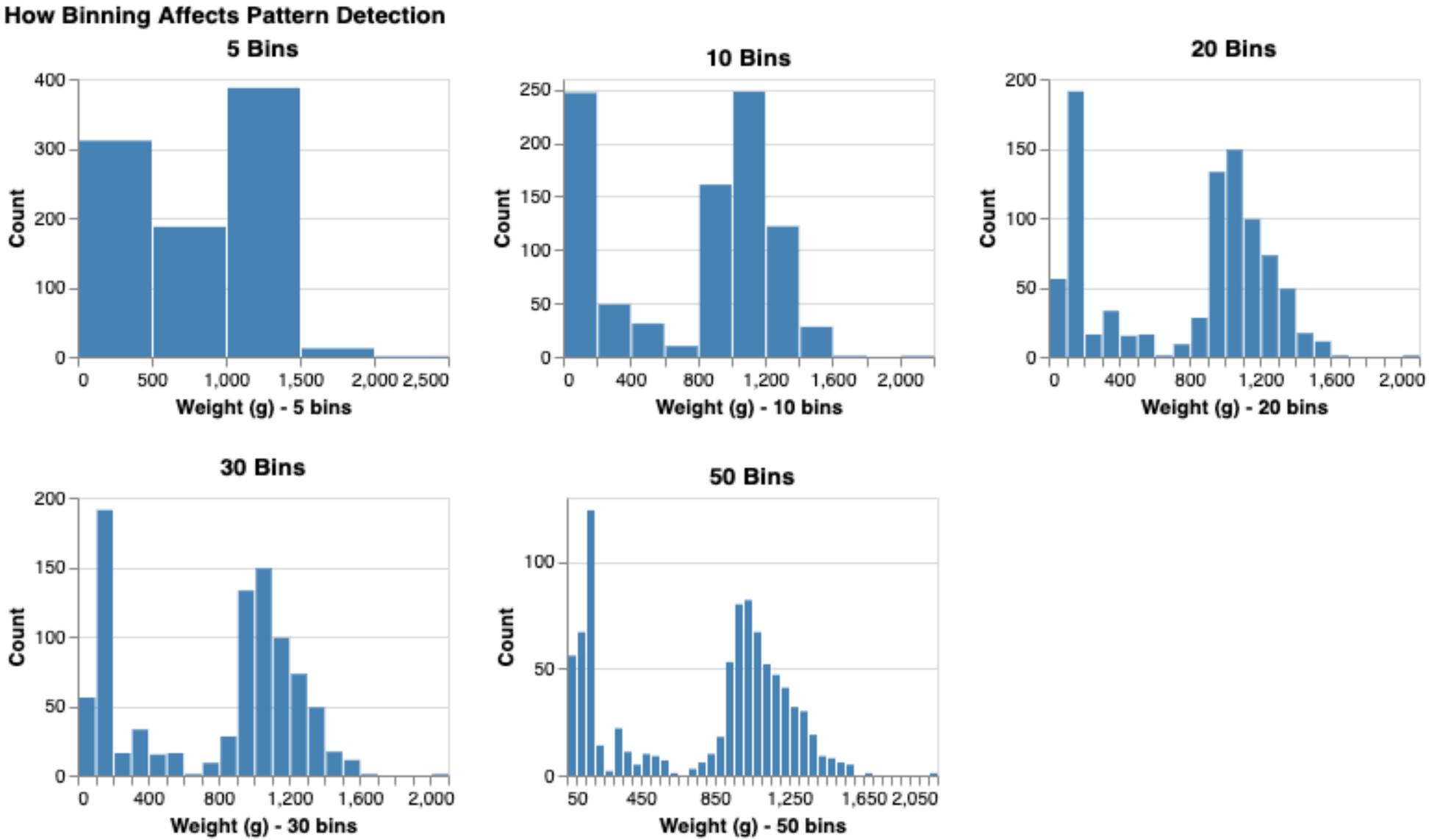
Situation	Use Pandas When:	Use Altair When:
Initial exploration	Need quick data overview	Want to spot unexpected patterns
Hypothesis testing	Testing specific numerical hypotheses	Exploring relationships visually
Precision required	Need exact values for reports	Communication and presentation
Time constraints	Need fast answers	Have time for thorough exploration
Audience	Technical/statistical audience	General audience, stakeholders

🎯 Best Practice: Use Both!

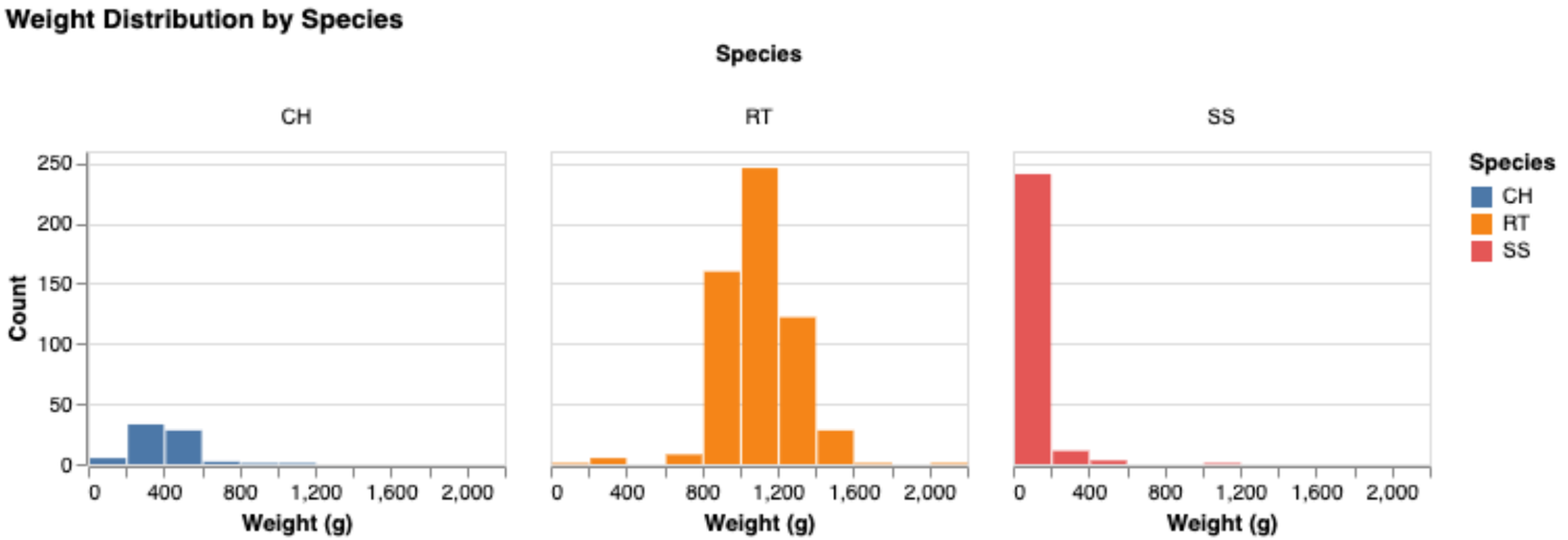
In real-world EDA:

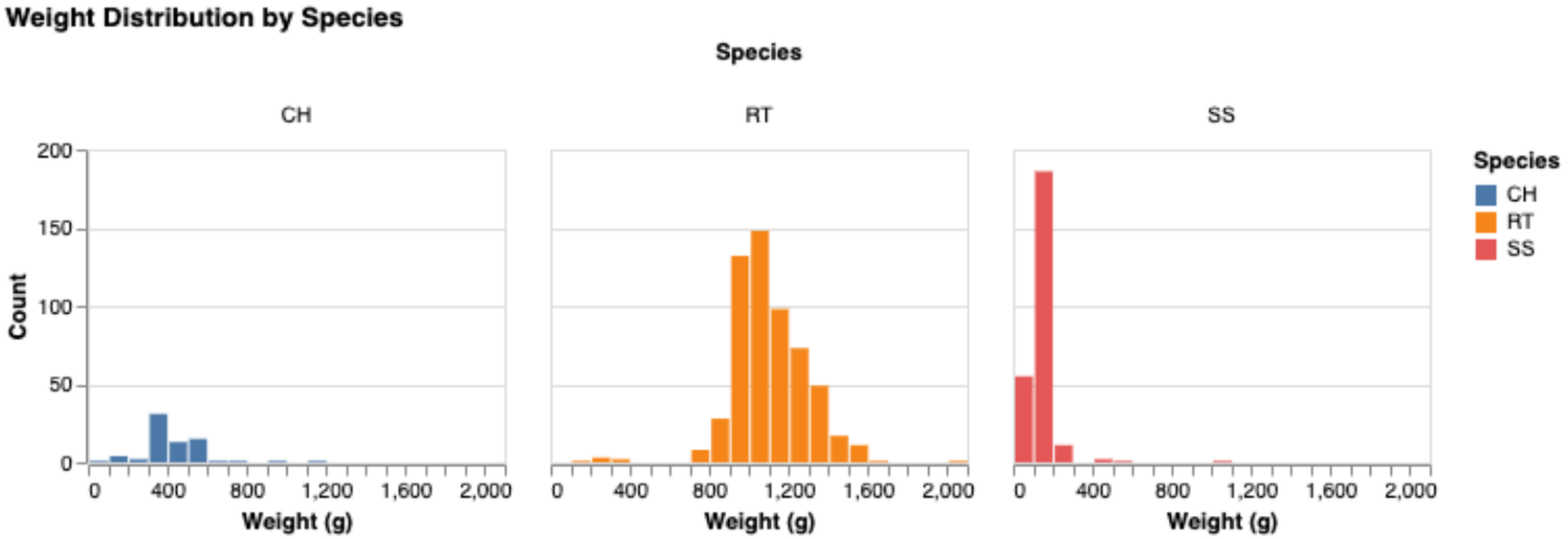
- **Start visual** → Spot interesting patterns quickly
- **Confirm statistical** → Get precise quantification
- **Communicate visual** → Present findings effectively

Bin Sizes


How Binning Affects Pattern Detection
5 Bins
Count
Weight (g) - 5 bins
10 Bins
Count
Weight (g) - 10 bins
20 Bins
Count
Weight (g) - 20 bins
30 Bins
Count
Weight (g) - 30 bins
50 Bins
Count
Weight (g) - 50 bins

Bin Sizes


Weight Distribution by Species
Species
CH
RT
SS
Count
250
200
150
100
50
0
0
400
800
1,200
1,600
2,000
Weight (g)
Species
CH
RT
SS


Weight Distribution by Species
Species
CH
RT
SS
Count
200
150
100
50
0
0
400
800
1,200
1,600
2,000
Weight (g)
Species
CH
RT
SS

How to choose bin sizes – Existing Methods

- Square Root Rule: $\sqrt{(number\ of\ data\ items)}$
  - Simple, most commont, not sensitive to distribution shape
- Sturges' Rule: $\lceil \log_2(n)+1 \rceil$
  - best used for normally distributed data, works best for smaller datasets, one limitation is that it can oversimplify for large datasets
- Doane's Formula: improvement to Sturges', best used for skewed distributions, it adjusts for skewness
- Rice Rule: $\lceil 2 \cdot n^{1/3} \rceil$
  - Similar to square-root rule but gives more bins
- Scott's Rule: based on the standard deviation, it minimizes bias
- Freedman-Diaconis Rule: based on the IQR, it is more robust to outliers and skewed distributions.

So what should I do?

By Sample Size

Sample Size	Recommended Method	Typical Bins
$n < 30$	5-7 bins fixed	5-7
$30 \leqslant n < 100$	Square Root	6-10
$100 \leqslant n < 1000$	Sturges' or Square Root	10-30
$n \geqslant 1000$	Freedman-Diaconis	20-50
$n \geqslant 10{,}000$	Freedman-Diaconis or Scott's	50-200

By Data Characteristics

Data Type	Best Method	Why
Normal distribution	Sturges' or Scott's	Designed for normal data
Skewed distribution	Doane's or Freedman-Diaconis	Handles asymmetry
With outliers	Freedman-Diaconis	Uses robust IQR
Unknown distribution	Freedman-Diaconis	Most generally applicable
Multimodal	Start with more bins (30-50)	Need to see multiple peaks

By Purpose

Purpose	Recommendation	Bins
Quick exploration	Square Root	Medium (10-30)
Presentation	Manual tuning for clarity	Fewer (5-15)
Detailed analysis	Freedman-Diaconis	More (20-50)
Finding patterns	Try multiple, compare	Iterate
Publication	Freedman-Diaconis + adjust	Well-justified

By default, Altairs uses maxbins=10 as default.
It isn’t data specific.
So make sure that you do some investigation before proceeding.

Read through the rest of the slides. Make sure you understand the different visualizations and when they will be used.

Use the references (links in the lower-right corner) to aid your understanding.

We will NOT spend time in class describing each standard visualization so please do this work.

Univariate Numerical Summaries

- Categorical Variable
  - Range of values
  - Frequency of each value (proportion)
- Quantitative Variable
  - Distribution: center, spread, modality, shape and outliers
  - Central Tendency: mean, mode, median
  - Spread: variance, standard deviations, interquartile range

Tukey advocates for focusing on max, min, median and quartiles

Univariate Visual Idioms

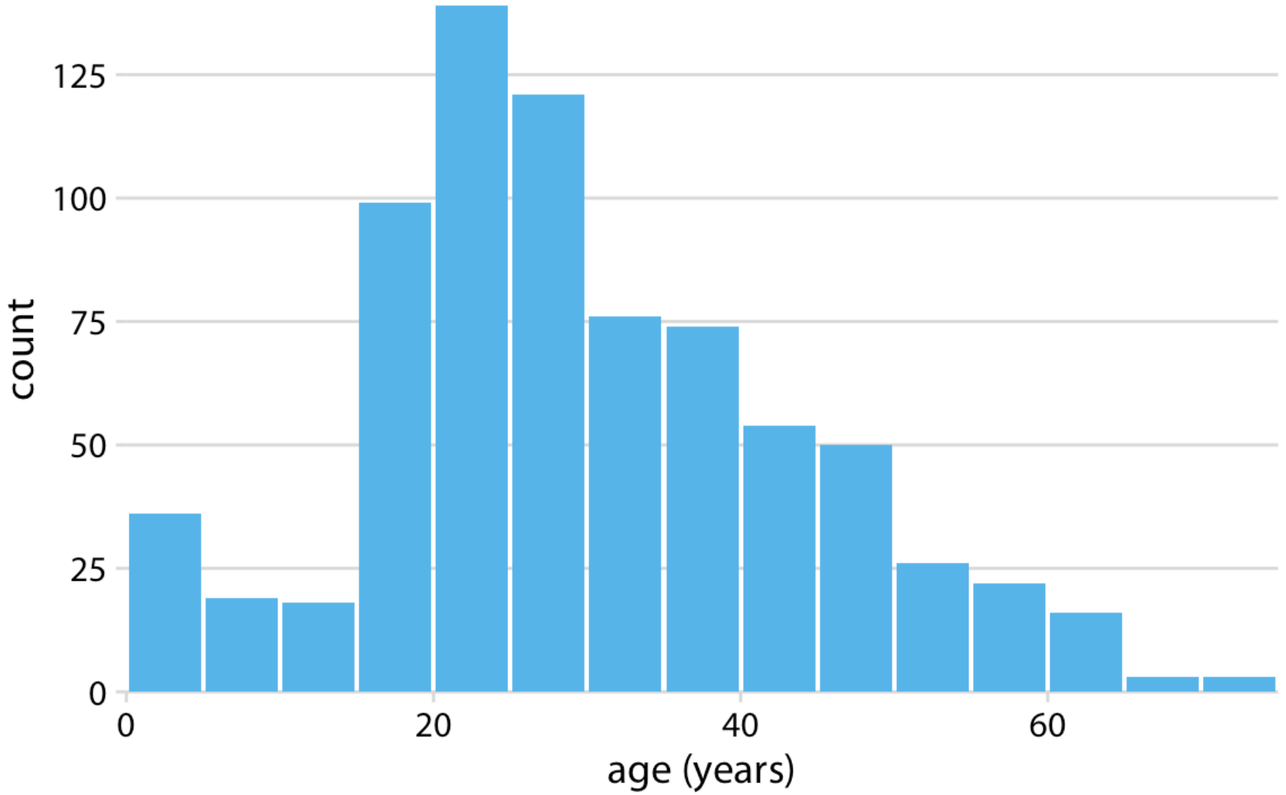
- Distribution of Categorical Variable
  - Bar Charts
- Distribution of Quantitative Variable
  - Histogram
  - Density Plots

Univariate Visual Idiom: Histograms

Histograms are used to visualize the shape, center, range and variation for a continuous variable.

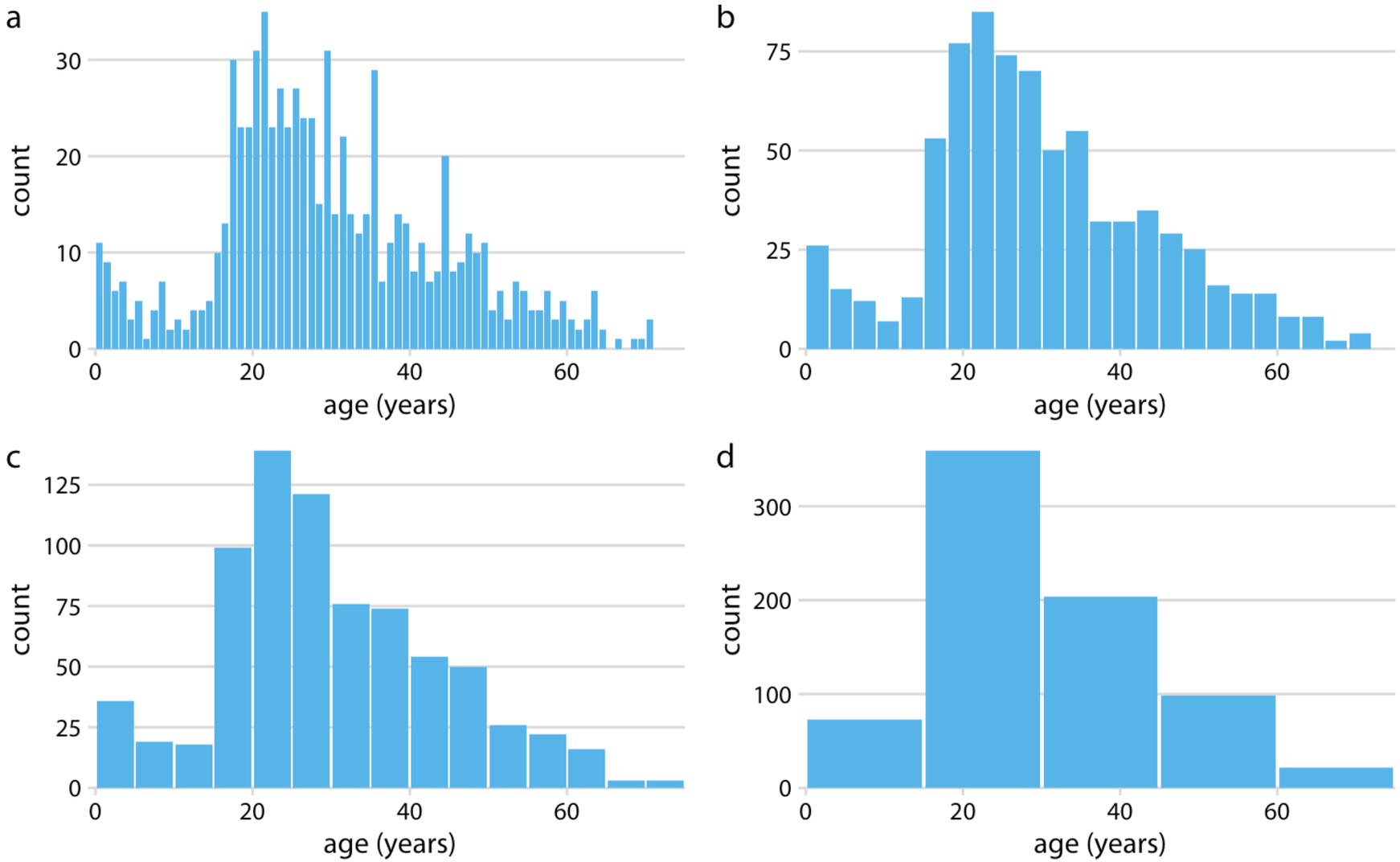
The size of the bin is extremely important.

Design Tip: Vary bin size during EDA



https://clauswilke.com/dataviz/histograms-density-plots.html

Univariate Visual Idiom: Histograms

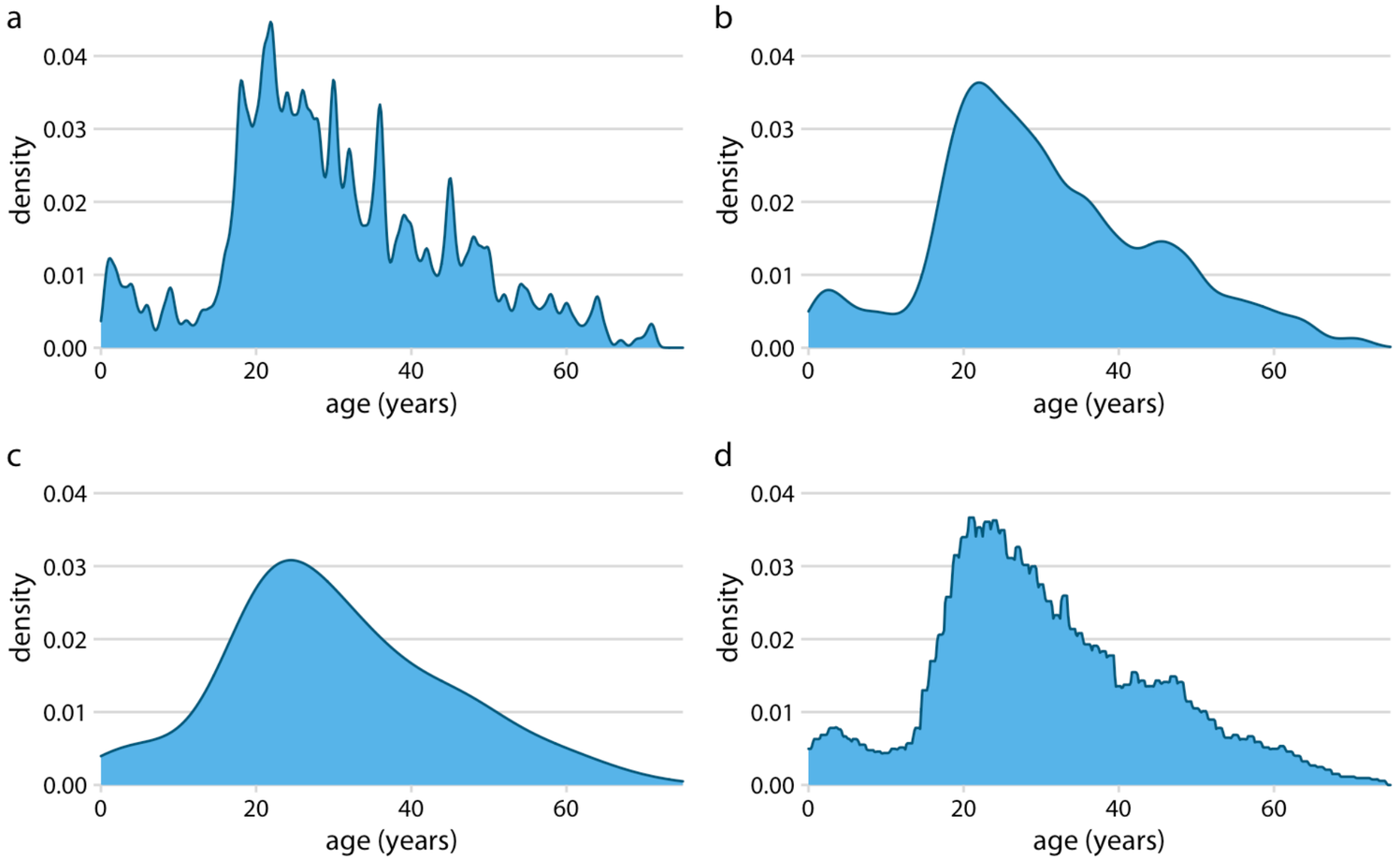


Histograms depend on the chosen bin width. Here, the same age distribution of Titanic passengers is shown with four different bin widths: (a) one year; (b) three years; (c) five years; (d) fifteen years.

https://clauswilke.com/dataviz/histograms-density-plots.html

Univariate Visual Idioms: Density Plot

A density plot is a representation of the distribution of a numeric variable. It uses the kernel density estimate to show the probability density function of a variable. It is basically a smoothed out version of the histogram.



Kernel density estimates depend on the chosen kernel and bandwidth. Here, the same age distribution of Titanic passengers is shown for four different combinations of these parameters: (a) Gaussian kernel, bandwidth = 0.5; (b) Gaussian kernel, bandwidth = 2; (c) Gaussian kernel, bandwidth = 5; (d) Rectangular kernel, bandwidth = 2.

https://clauswilke.com/dataviz/histograms-density-plots.html

Multivariate Numerical Summaries

Categorical Variable

- cross-tabulation
- Univariate statistics by category

	Cake	Ice	Donut	Total
Female	4	3	6	**13**
Male	5	7	9	**21**
Total	**9**	**10**	**15**	**34**

Quantitative Variable

- Correlation & Covariance: a measure of how much (and in what direction) should we expect one variable to change when the other changes.
- Correlation & Covariance Matrix for >2 variables

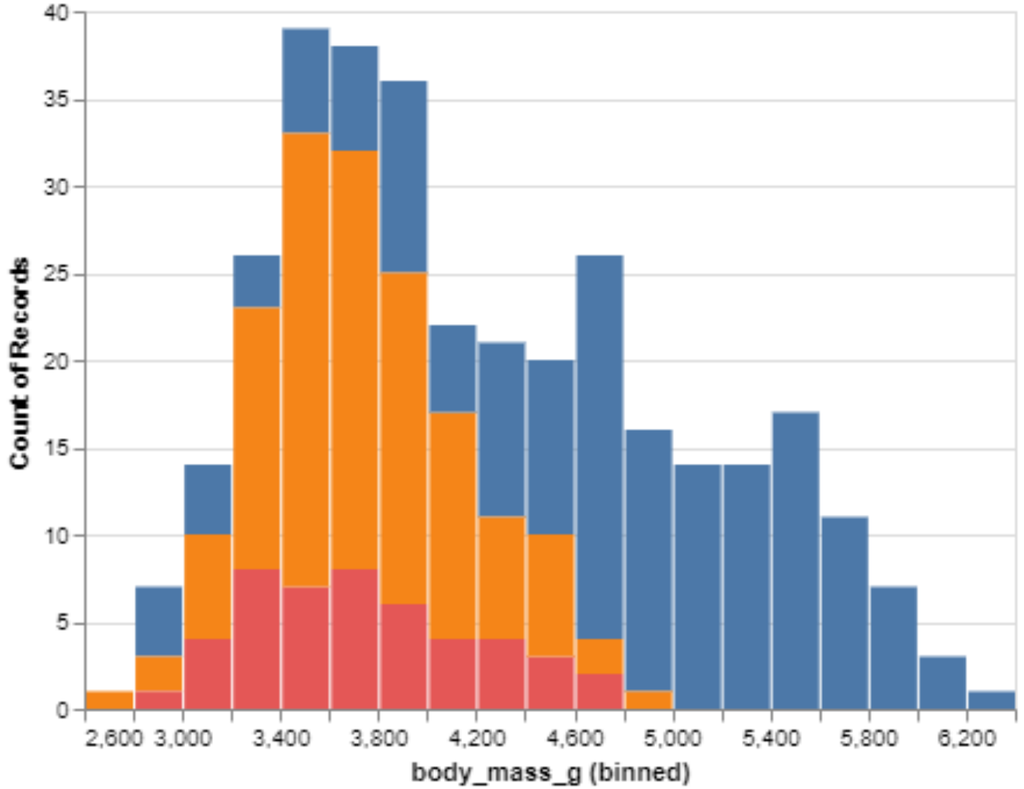
Experimental Design and Analysis. Howard J. Seltman

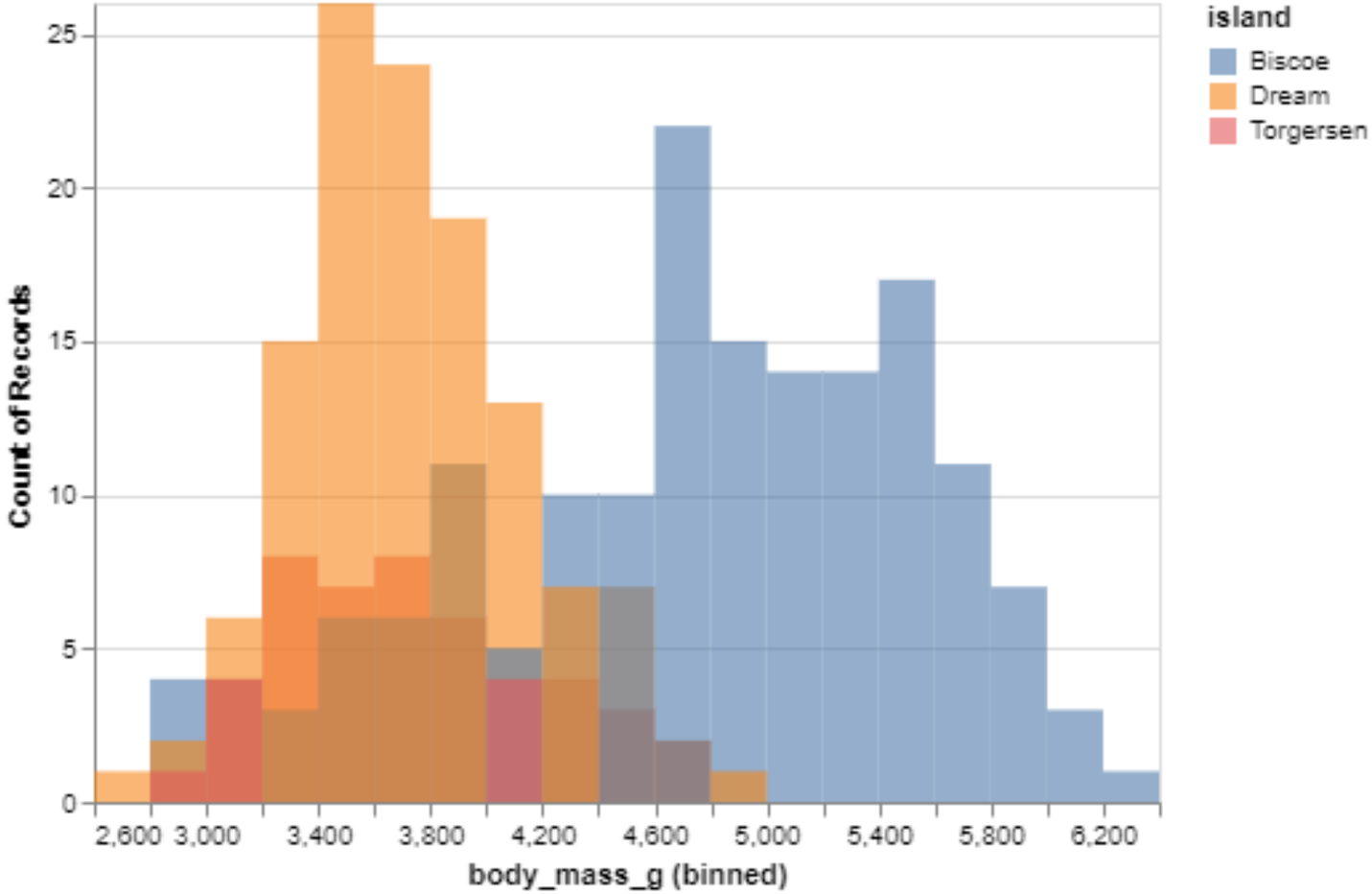
https://datatab.net/tutorial/cross-table

Multivariate Visual Idioms

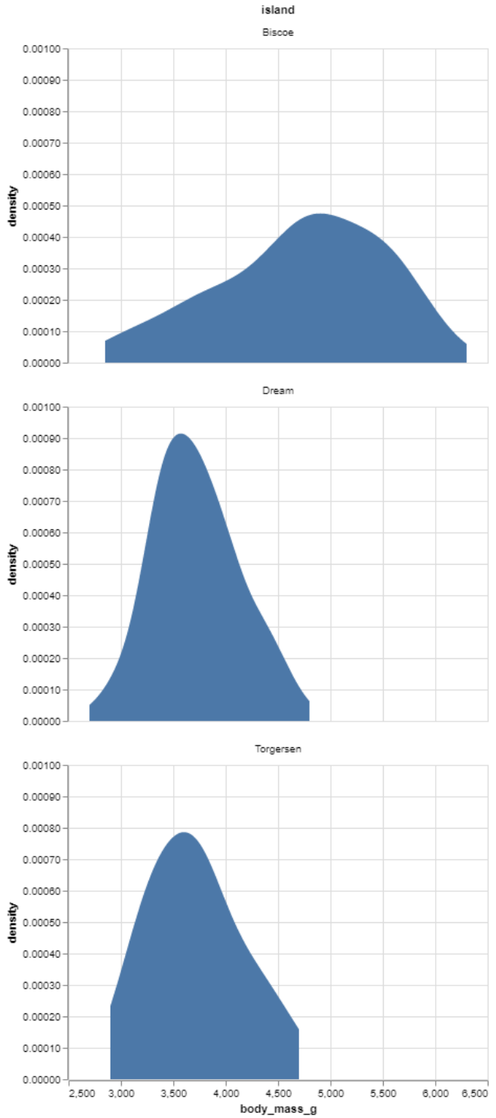
- Categorical Data
  - Stacked Bar Charts
- Quantitative Data
  - Overlapping Density Plots
  - Scatterplots
  - Box-plots & Violin Plots (see T8)
  - Univariate graphs by category – typically used when we have one explanatory variable (categorical) and one outcome (quantitative) variable

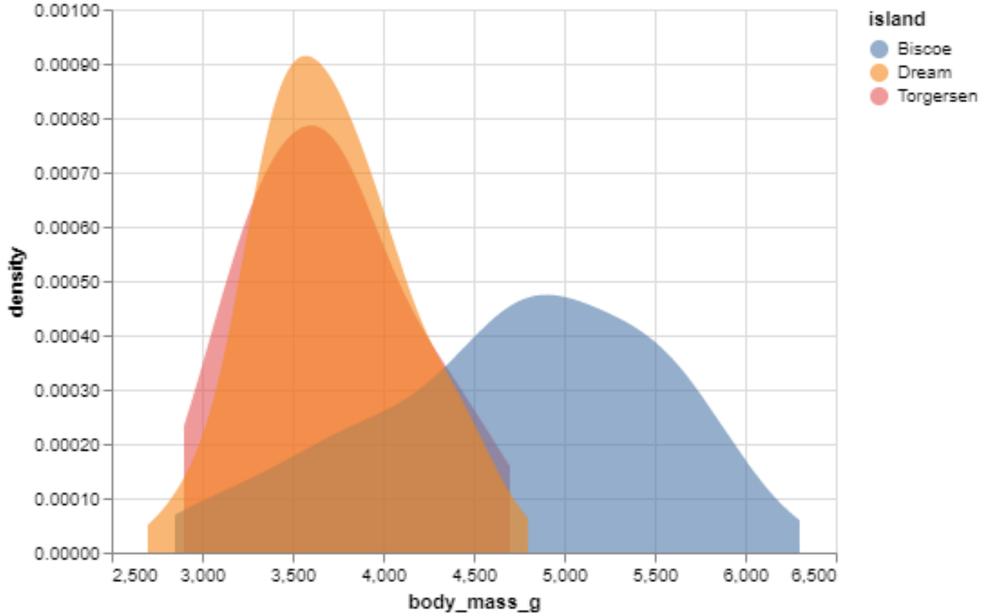
Bivariate Visual Idioms: Stacked Bar Chart

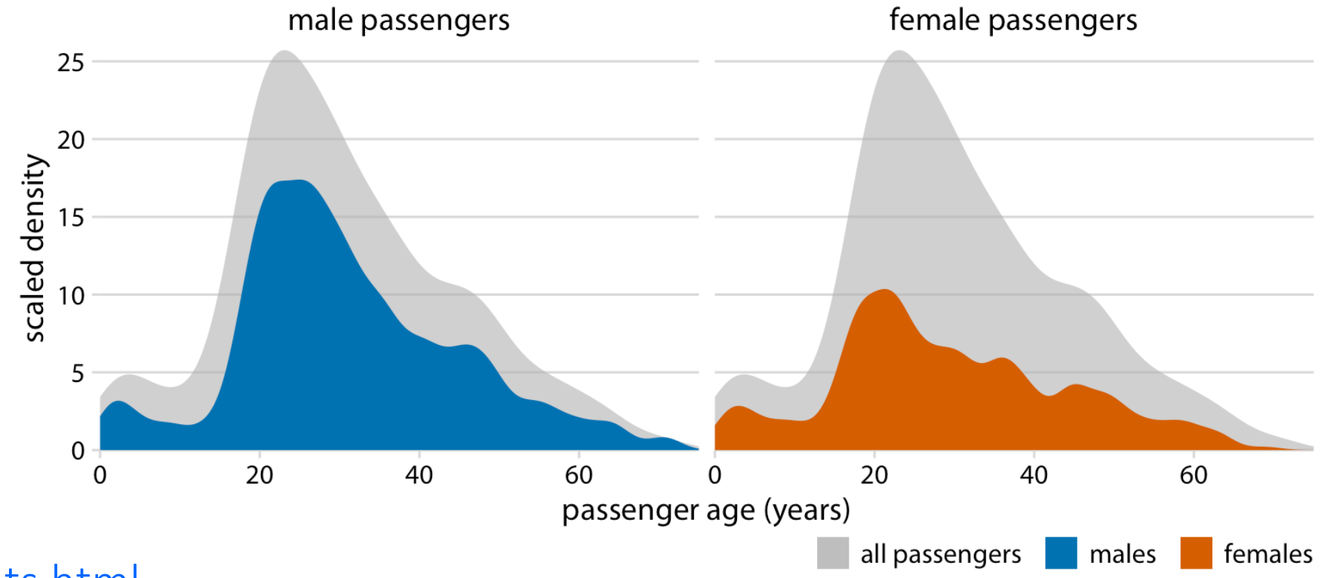




Multivariate Visual Idioms: Overlapping & Faceted Density Plots

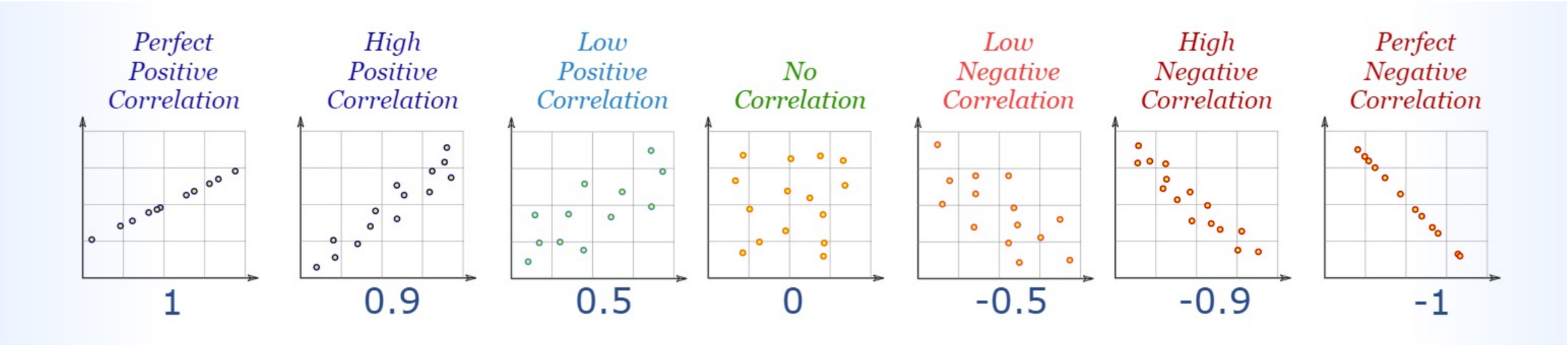






https://clauswilke.com/dataviz/histograms-density-plots.html

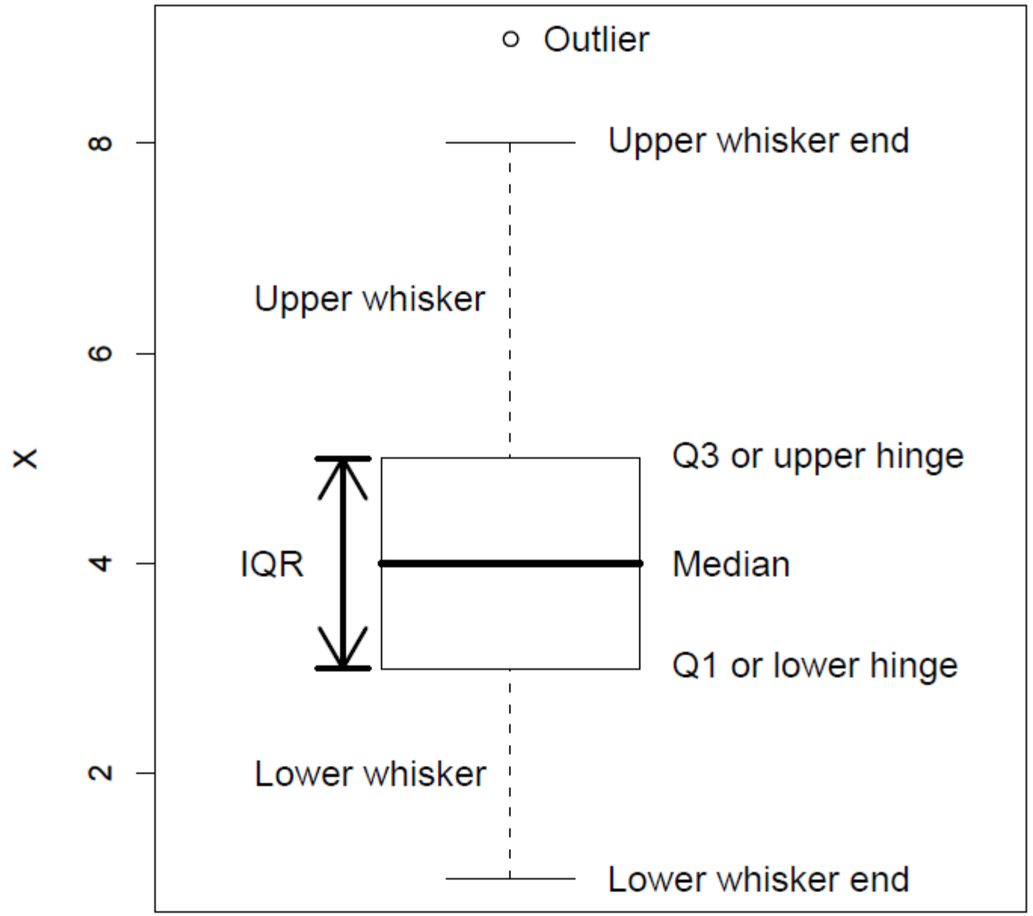
Bivariate Visual Idioms: Scatterplot



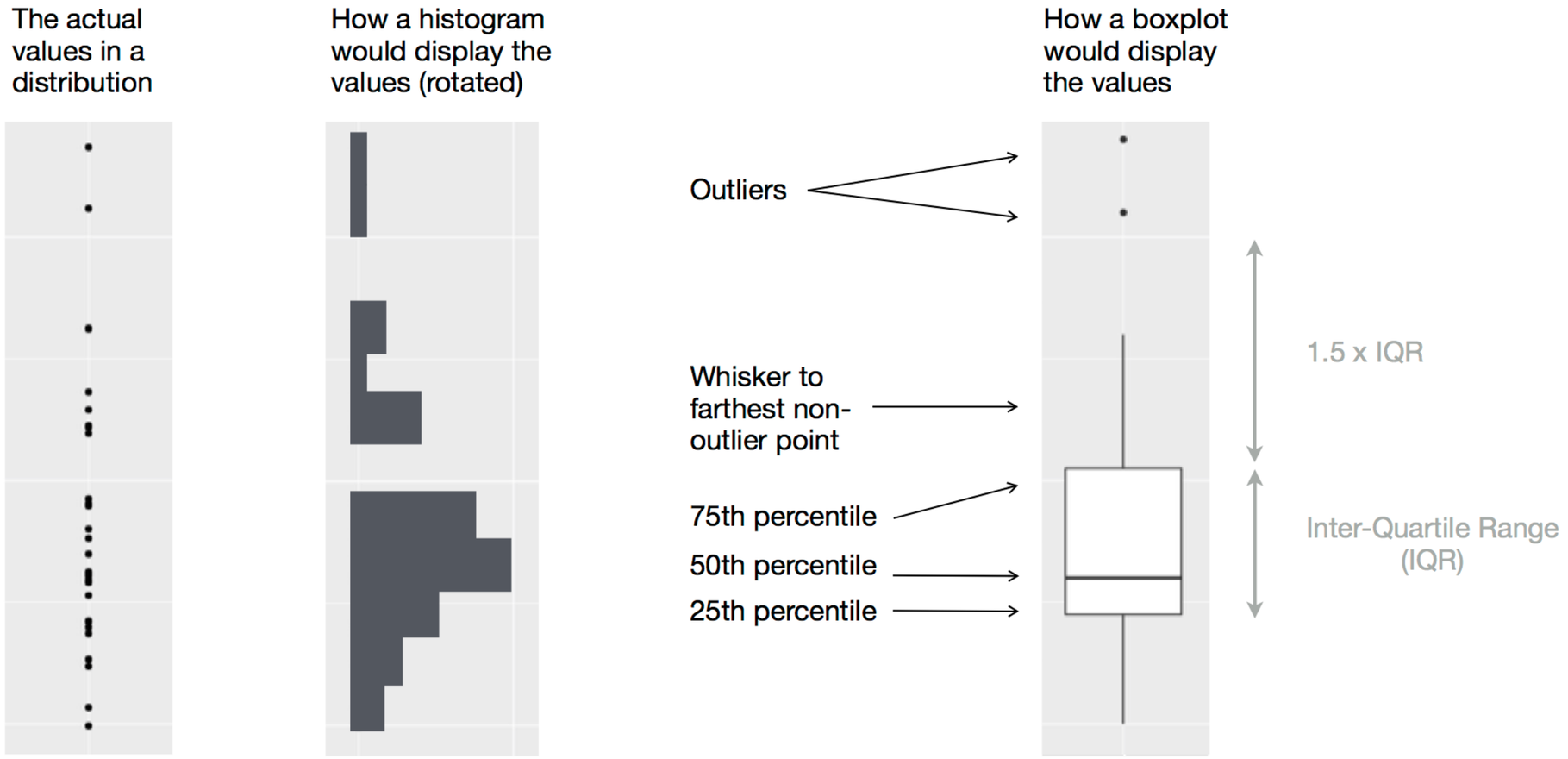
https://www.mathsisfun.com/data/scatter-xy-plots.html

Multivariate Visual Idioms: Boxplots

Very good at representing data related to the central tendency, symmetry and skew.



Multivariate Visual Idioms: Boxplots



https://r4ds.had.co.nz/exploratory-data-analysis.html

“In a nutshell: You should always perform appropriate EDA before further analysis of your data. Perform whatever steps are necessary to become more familiar with your data, check for obvious mistakes, learn about variable distributions, and learn about relationships between variables. EDA is not an exact science ***{ it is a very important art!}”***

- Howard J. Seltman