# ML - Data Plotting and Visualization


CK Raju

PhD IIT Kharagpur


October 8, 2023

# Quotes

- Data, like a constellation in the night, finds its brilliance in the art of visualization, revealing the hidden stories of the universe it holds within.

# Quotes

- Data, like a constellation in the night, finds its brilliance in the art of visualization, revealing the hidden stories of the universe it holds within.
- Dimensionality reduction, where data’s symphony finds its graceful, concise melody

# Data Plots

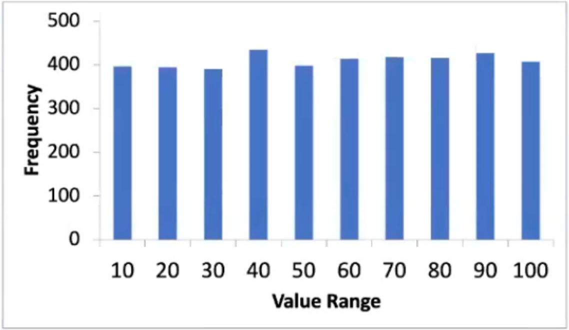
Diagrams to Gain Insights

# Why Plot Data

| | | | | | |
|---|---|---|---|---|---|
| 22.65 | 42.12 | 67.24 | 59.13 | | 81.49 |
| 23.03 | 53.42 | 40.54 | 89.97 | | 21.85 |
| 12.07 | 93.43 | 51.93 | 49.30 | | 43.76 |
| 47.68 | 51.91 | 13.12 | 73.88 | | 60.29 |
| 86.20 | 41.28 | 66.24 | 62.15 | | 46.87 |
| 20.02 | 92.09 | 26.50 | 83.53 | | 70.99 |
| 48.38 | 46.21 | 10.85 | 29.61 | | 62.15 |
| 55.23 | 84.90 | 15.37 | 35.00 | | 83.23 |
| 65.30 | 26.56 | 5.78 | 72.59 | | 12.47 |
| 75.71 | 93.15 | 3.67 | 49.80 | ... | 43.05 |
| 69.73 | 53.77 | 82.80 | 43.59 | | 32.35 |
| 77.95 | 14.94 | 63.71 | 9.30 | | 1.31 |
| 58.90 | 42.53 | 62.74 | 99.91 | | 53.17 |
| 6.45 | 46.29 | 67.34 | 32.65 | | 23.94 |
| 32.39 | 57.39 | 10.61 | 54.07 | | 53.28 |
| 74.35 | 60.10 | 2.25 | 77.55 | | 12.05 |
| 82.87 | 17.02 | 80.73 | 29.60 | | 9.96 |
| 47.05 | 97.01 | 19.84 | 76.59 | | 45.90 |
| 34.26 | 86.80 | 19.11 | 4.80 | | 1.24 |
| 12.54 | 30.40 | 67.94 | 55.53 | | 58.25 |
| | | | 73.13 | | 0.23 |

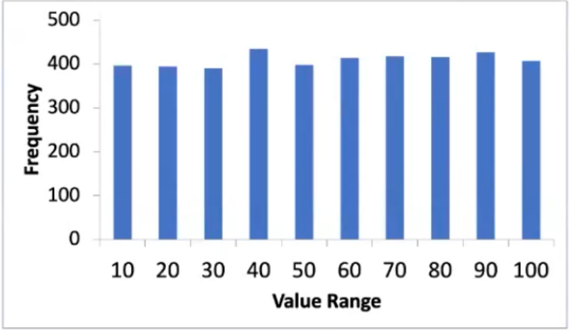
# Why Plot Data

| | | | | | |
|---|---|---|---|---|---|
| 22.65 | 42.12 | 67.24 | 59.13 | | 81.49 |
| 23.03 | 53.42 | 40.54 | 89.97 | | 21.85 |
| 12.07 | 93.43 | 51.93 | 49.30 | | 43.76 |
| 47.68 | 51.91 | 13.12 | 73.88 | | 60.29 |
| 86.20 | 41.28 | 66.24 | 62.15 | | 46.87 |
| 20.02 | 92.09 | 26.50 | 83.53 | | 70.99 |
| 48.38 | 46.21 | 10.85 | 29.61 | | 62.15 |
| 55.23 | 84.90 | 15.37 | 35.00 | | 83.23 |
| 65.30 | 26.56 | 5.78 | 72.59 | | 12.47 |
| 75.71 | 93.15 | 3.67 | 49.80 | ... | 43.05 |
| 69.73 | 53.77 | 82.80 | 43.59 | | 32.35 |
| 77.95 | 14.94 | 63.71 | 9.30 | | 1.31 |
| 58.90 | 42.53 | 62.74 | 99.91 | | 53.17 |
| 6.45 | 46.29 | 67.34 | 32.65 | | 23.94 |
| 32.39 | 57.39 | 10.61 | 54.07 | | 53.28 |
| 74.35 | 60.10 | 2.25 | 77.55 | | 12.05 |
| 82.87 | 17.02 | 80.73 | 29.60 | | 9.96 |
| 47.05 | 97.01 | 19.84 | 76.59 | | 45.90 |
| 34.26 | 86.80 | 19.11 | 4.80 | | 1.24 |
| 12.54 | 30.40 | 67.94 | 55.53 | | 58.25 |
| | | | 73.13 | | 0.23 |



# Why Plot Data

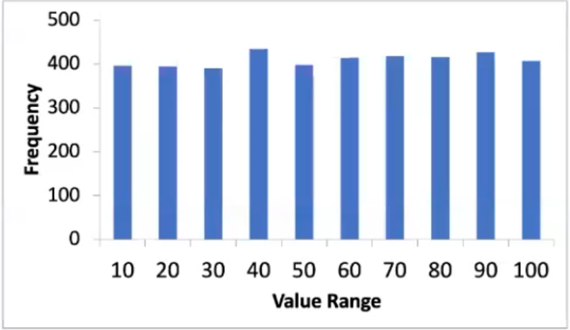
| | | | | | |
|---|---|---|---|---|---|
| 22.65 | 42.12 | 67.24 | 59.13 | | 81.49 |
| 23.03 | 53.42 | 40.54 | 89.97 | | 21.85 |
| 12.07 | 93.43 | 51.93 | 49.30 | | 43.76 |
| 47.68 | 51.91 | 13.12 | 73.88 | | 60.29 |
| 86.20 | 41.28 | 66.24 | 62.15 | | 46.87 |
| 20.02 | 92.09 | 26.50 | 83.53 | | 70.99 |
| 48.38 | 46.21 | 10.85 | 29.61 | | 62.15 |
| 55.23 | 84.90 | 15.37 | 35.00 | | 83.23 |
| 65.30 | 26.56 | 5.78 | 72.59 | | 12.47 |
| 75.71 | 93.15 | 3.67 | 49.80 | ... | 43.05 |
| 69.73 | 53.77 | 82.80 | 43.59 | | 32.35 |
| 77.95 | 14.94 | 63.71 | 9.30 | | 1.31 |
| 58.90 | 42.53 | 62.74 | 99.91 | | 53.17 |
| 6.45 | 46.29 | 67.34 | 32.65 | | 23.94 |
| 32.39 | 57.39 | 10.61 | 54.07 | | 53.28 |
| 74.35 | 60.10 | 2.25 | 77.55 | | 12.05 |
| 82.87 | 17.02 | 80.73 | 29.60 | | 9.96 |
| 47.05 | 97.01 | 19.84 | 76.59 | | 45.90 |
| 34.26 | 86.80 | 19.11 | 4.80 | | 1.24 |
| 12.54 | 30.40 | 67.94 | 55.53 | | 58.25 |
| | | | 73.13 | | 0.23 |




- Visualization of data provides specific insights into the nature of the data.

# Why Plot Data

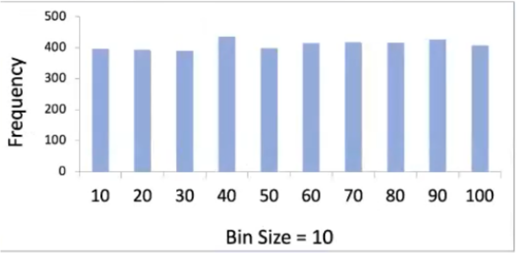
| | | | | | |
|---|---|---|---|---|---|
| 22.65 | 42.12 | 67.24 | 59.13 | | 81.49 |
| 23.03 | 53.42 | 40.54 | 89.97 | | 21.85 |
| 12.07 | 93.43 | 51.93 | 49.30 | | 43.76 |
| 47.68 | 51.91 | 13.12 | 73.88 | | 60.29 |
| 86.20 | 41.28 | 66.24 | 62.15 | | 46.87 |
| 20.02 | 92.09 | 26.50 | 83.53 | | 70.99 |
| 48.38 | 46.21 | 10.85 | 29.61 | | 62.15 |
| 55.23 | 84.90 | 15.37 | 35.00 | | 83.23 |
| 65.30 | 26.56 | 5.78 | 72.59 | | 12.47 |
| 75.71 | 93.15 | 3.67 | 49.80 | ... | 43.05 |
| 69.73 | 53.77 | 82.80 | 43.59 | | 32.35 |
| 77.95 | 14.94 | 63.71 | 9.30 | | 1.31 |
| 58.90 | 42.53 | 62.74 | 99.91 | | 53.17 |
| 6.45 | 46.29 | 67.34 | 32.65 | | 23.94 |
| 32.39 | 57.39 | 10.61 | 54.07 | | 53.28 |
| 74.35 | 60.10 | 2.25 | 77.55 | | 12.05 |
| 82.87 | 17.02 | 80.73 | 29.60 | | 9.96 |
| 47.05 | 97.01 | 19.84 | 76.59 | | 45.90 |
| 34.26 | 86.80 | 19.11 | 4.80 | | 1.24 |
| 12.54 | 30.40 | 67.94 | 55.53 | | 58.25 |
| | | | 73.13 | | 0.23 |




- Visualization of data provides specific insights into the nature of the data.
- Depending on the plot, we gain different insights

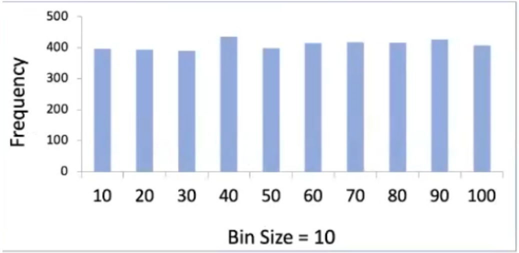
# Histogram

- Count of items in each bin



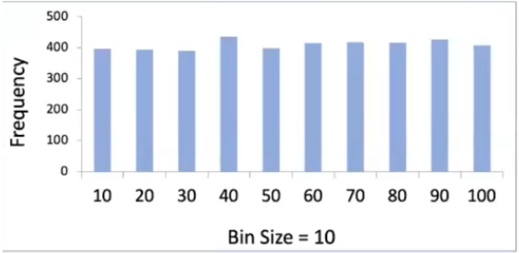
# Histogram

- Count of items in each bin
  - Not a bar chart of Data



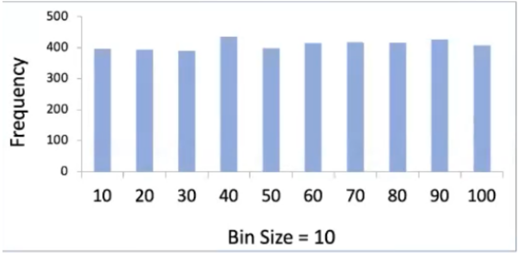
# Histogram

- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution



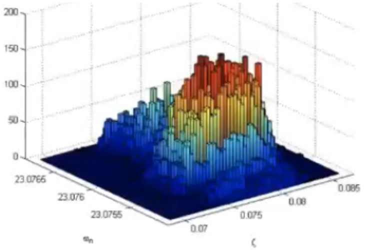
# Histogram

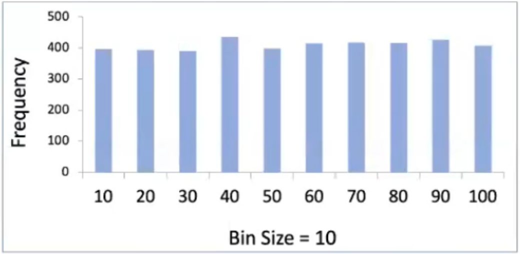
- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution
- Visualize one feature at a time



# Histogram

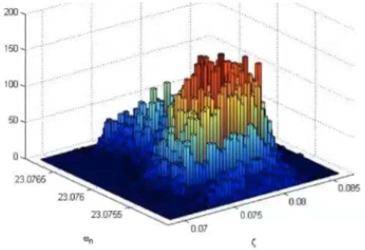
- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution
- Visualize one feature at a time
- Possible to extend to two

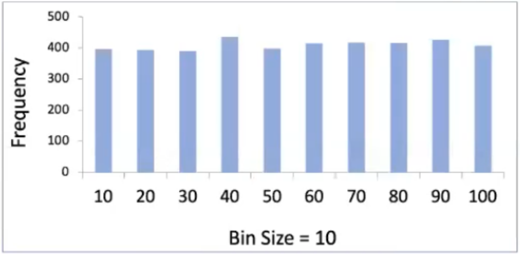


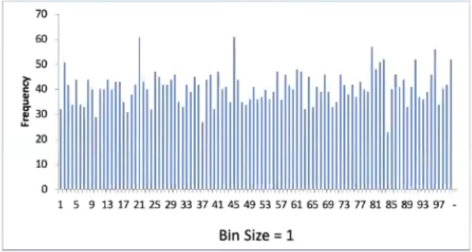



# Histogram

- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution
- Visualize one feature at a time
- Possible to extend to two
- Dependency on bins ($\sqrt{n}$;  $2\sqrt[3]{n}$)

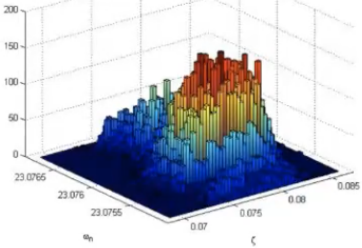


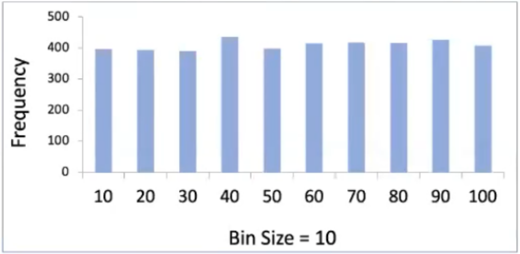



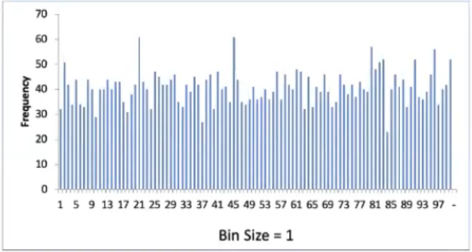



# Histogram

- Count of items in each bin
  - Not a bar chart of Data
  - Approximation of Distribution
- Visualize one feature at a time
- Possible to extend to two
- Dependency on bins ($\sqrt{n}$; $2\sqrt[3]{n}$)

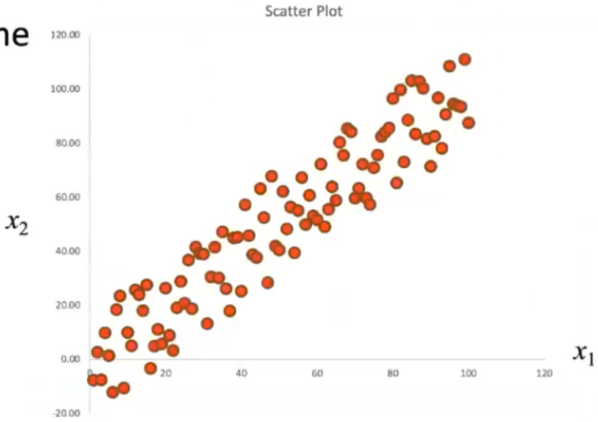


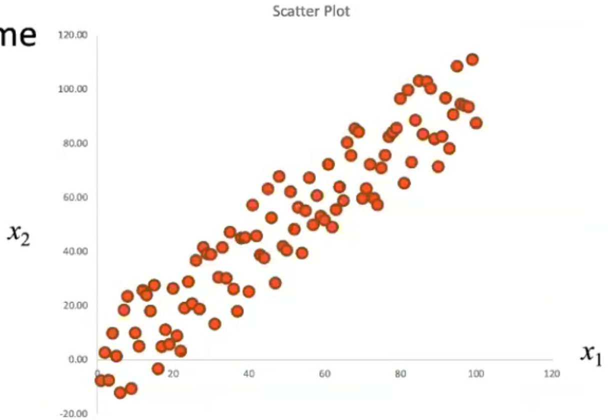
# Scatter Plot

- Plots two features at a time



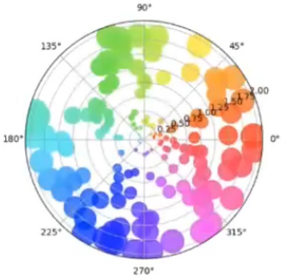
# Scatter Plot

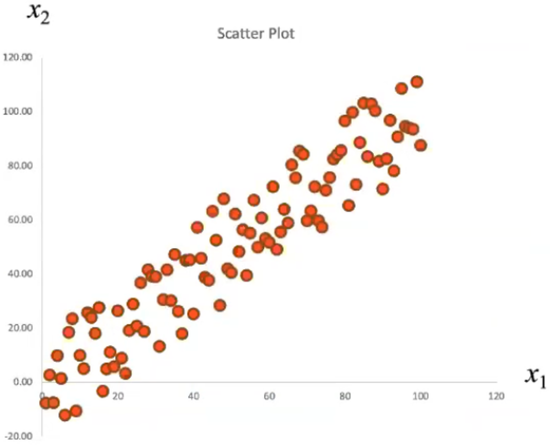
- Plots two features at a time
- Captures the correlation between the two



# Scatter Plot

- Plots two features at a time
- Captures the correlation between the two
- Other formats possible

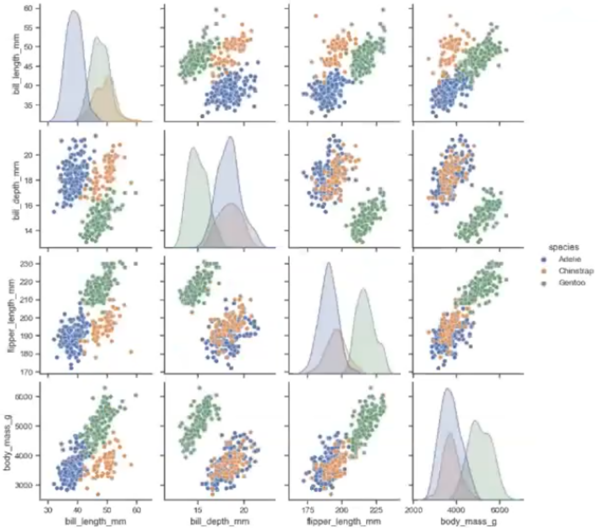



Polar Plot Courtesy: Scatterplot Documentation [matplotlib.org]

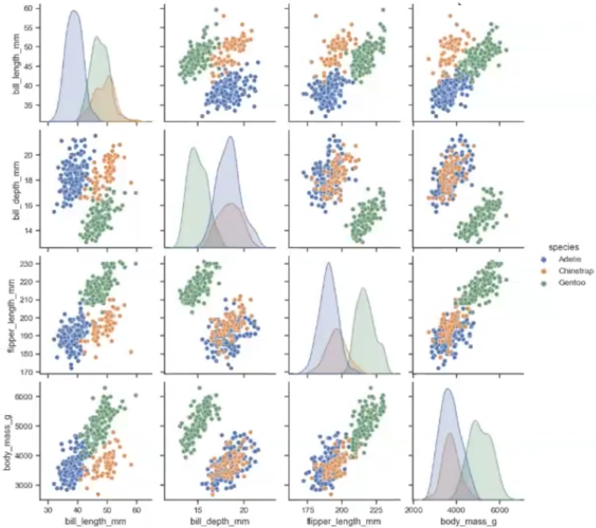
# Pair Plot

- Plot each pair of features as a matrix
  - Diagonal entries are histograms (densities)
  - Off-diagonal entries are scatter plots



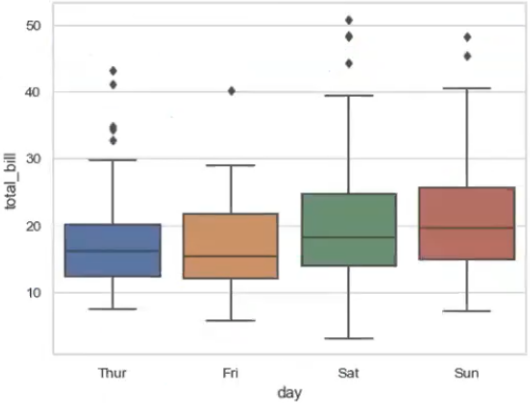
# Pair Plot

- Plot each pair of features as a matrix
  - Diagonal entries are histograms (densities)
  - Off-diagonal entries are scatter plots
- Can use other plots at each cell.



# Box Plot

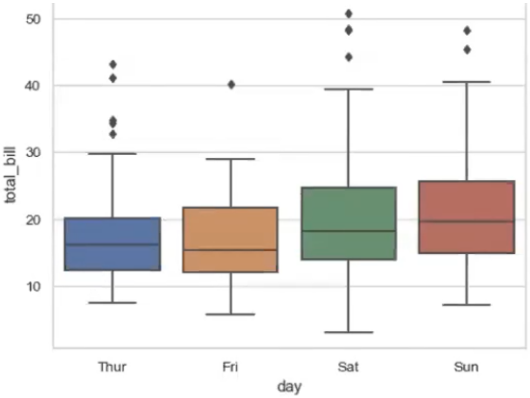
- Show median and quartiles of each feature




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Box Plot

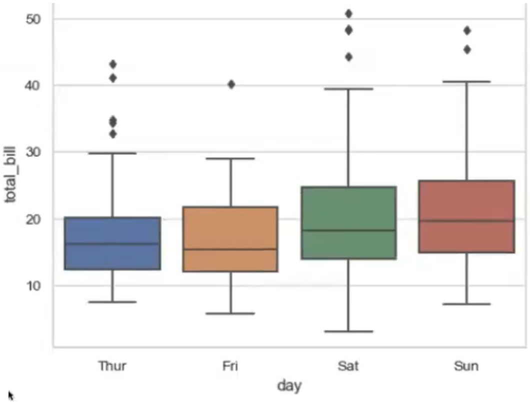
- Show median and quartiles of each feature
  - Outliers are removed




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Box Plot

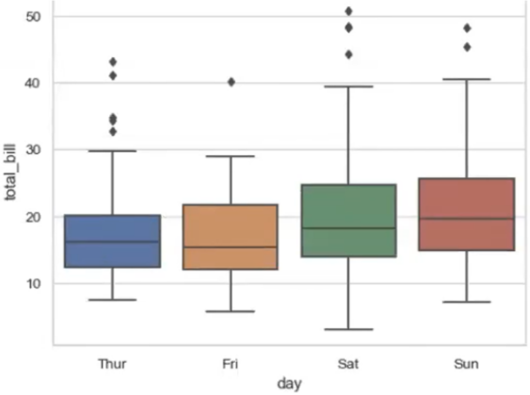
- Show median and quartiles of each feature
  - Outliers are removed
  - Box-and-whisker plot




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Box Plot

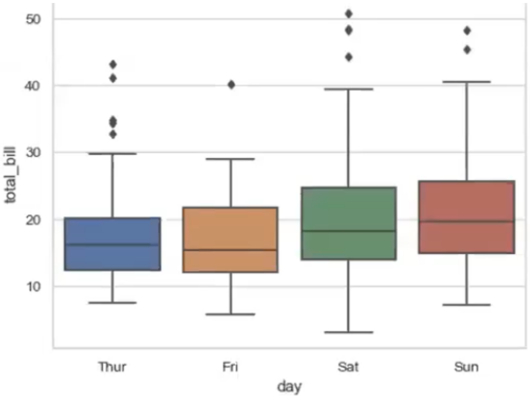
- Show median and quartiles of each feature
  - Outliers are removed
  - Box-and-whisker plot
- Whiskers can represent other percentiles/data




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Box Plot

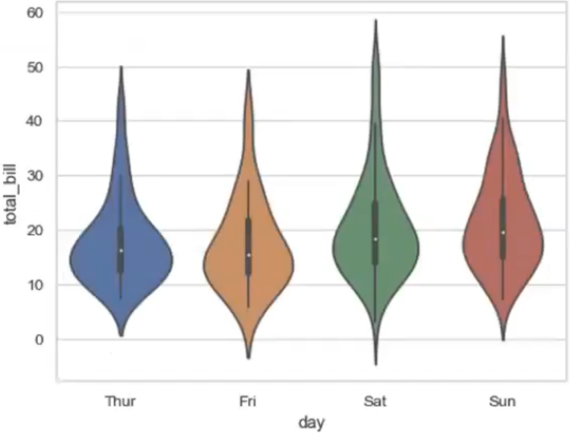
- Show median and quartiles of each feature
  - Outliers are removed
  - Box-and-whisker plot
- Whiskers can represent other percentiles/data
- Simpler than histograms of each feature




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Violin Plot

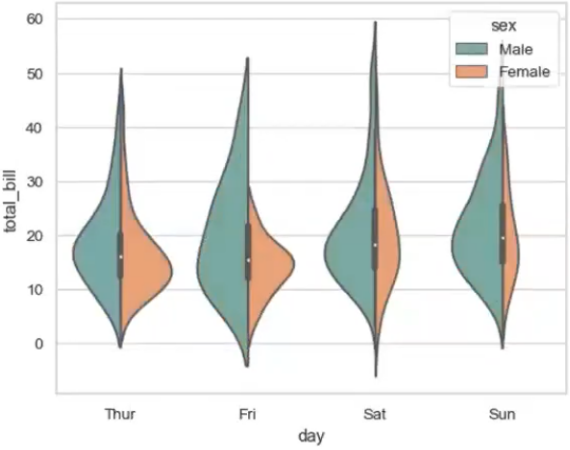
- Shows the density plot of each feature




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Violin Plot

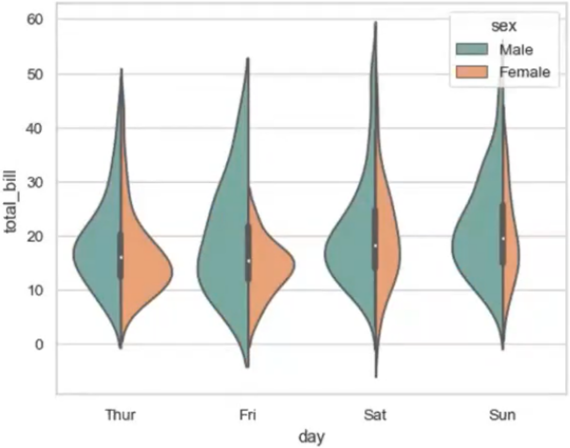
- Shows the density plot of each feature
  - Similar to Box Plot
- Either side can represent different densities




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Violin Plot

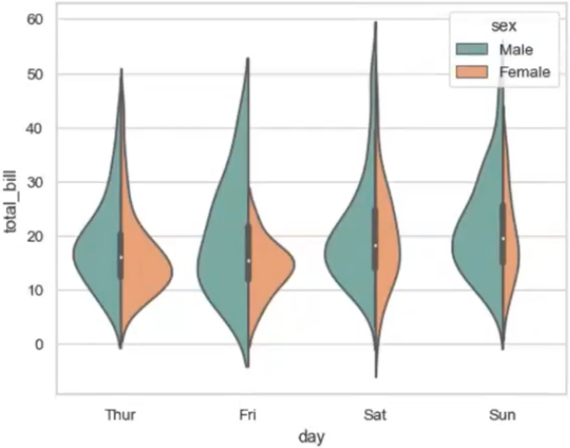
- Shows the density plot of each feature
  - Similar to Box Plot
- Either side can represent different densities




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Violin Plot

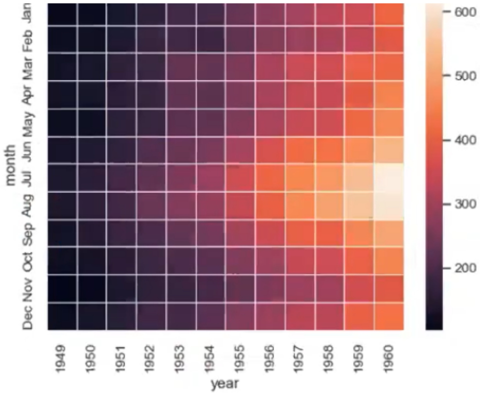
- Shows the density plot of each feature
  - Similar to Box Plot
- Either side can represent different densities
- Densities are smoothened estimates from data




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Heat Map

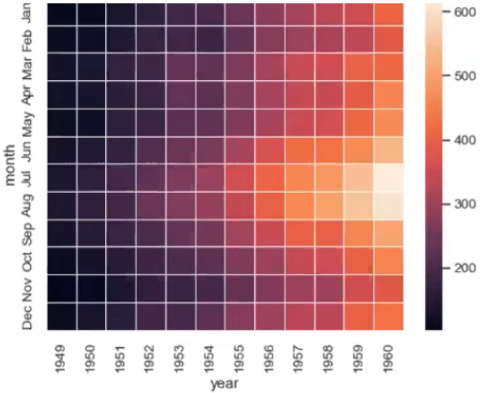
- A color-coded representation of 2D data
- Can be raw data, 2D histogram or any other function of 2 variables




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Heat Map

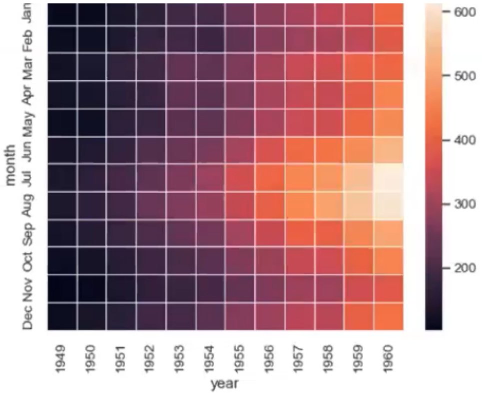
- A color-coded representation of 2D data
- Can be raw data, 2D histogram or any other function of 2 variables
- A color map accompanies the heat map




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Heat Map

- A color-coded representation of 2D data
- Can be raw data, 2D histogram or any other function of 2 variables
- A color map accompanies the heat map
- We will learn other metrics in future that may be visualized as a heat map




Plot Courtesy: Seaborn Documentation [seaborn.pydata.org]

# Dimensionality Reduction

Representing data using fewer features

# Dimensionality Reduction

- Feature Selection
  - Select a subset ($k$) of the original ($d$) features

# Dimensionality Reduction

- Feature Selection
  - Select a subset ($k$) of the original ($d$) features
  - Remove useless features
  - Select most useful features

# Dimensionality Reduction

- Feature Selection
  - Select a subset (***k***) of the original (***d***) features
  - Remove useless features
  - Select most useful features

- Feature Extraction
  - Create (***k***) new features as a linear combination of the (***d***) original features

# Dimensionality Reduction

- Feature Selection
  - Select a subset ($\boldsymbol{k}$) of the original ($\boldsymbol{d}$) features
  - Remove useless features
  - Select most useful features

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

- Feature Extraction
  - Create ($\boldsymbol{k}$) new features as a linear combination of the ($\boldsymbol{d}$) original features

# Dimensionality Reduction

- Feature Selection
  - Select a subset ($\boldsymbol{k}$) of the original ($\boldsymbol{d}$) features
  - Remove useless features
  - Select most useful features

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

- Feature Extraction
  - Create ($\boldsymbol{k}$) new features as a linear combination of the ($\boldsymbol{d}$) original features

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 1.2 & -0.4 & 2.1 & 0.3 \\ -2 & 1.1 & 0.1 & 5.8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

# Feature Selection: Forward Selection

***Algorithm:***

$z \leftarrow$ null vector; $x$ is the original feature vector

**for** each new feature to be added to $z$:

  **find** $x_i$ that gives highest accuracy when added to $z$

  **if** adding $x_i$ to $z$ improves accuracy:

    move $x_i$ to $z$

  **else**

    **break**

# Feature Selection: Backward Elimination

***Algorithm:***

$z \leftarrow x$

**for** each new feature to be removed from $z$:

    **find** $z_i$ that gives highest accuracy when removed from $z$

    **if** removing $z_i$ improves accuracy:

        remove $z_i$ from $z$

    **else**

        **break**

# Feature Selection: Other Methods

- Do forward and backward selection in alternating fashion

# Feature Selection: Other Methods

- Do forward and backward selection in alternating fashion
- Identify the best subset $z$ of $x$ that gives the highest accuracy for a given length of $z$

# Feature Selection: Other Methods

- Do forward and backward selection in alternating fashion
- Identify the best subset $z$ of $x$ that gives the highest accuracy for a given length of $z$
- Identify the smallest subset, $z$ of $x$, that gives a given accuracy or more

# Feature Selection: Other Methods

- Do forward and backward selection in alternating fashion
- Identify the best subset $z$ of $x$ that gives the highest accuracy for a given length of $z$
- Identify the smallest subset, $z$ of $x$, that gives a given accuracy or more
- We will learn specific algorithms that can do such selections

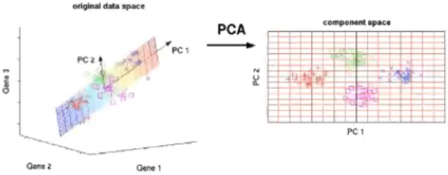
# Feature Selection: Other Methods

- Do forward and backward selection in alternating fashion
- Identify the best subset $z$ of $x$ that gives the highest accuracy for a given length of $z$
- Identify the smallest subset, $z$ of $x$, that gives a given accuracy or more
- We will learn specific algorithms that can do such selections
- The final subset may be written as a binary matrix

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

# Feature Extraction

- Goal: Identify a matrix $W$, such that $z = Wx$.

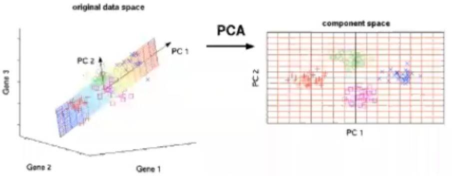



Courtesy: "PCA in Neuroscience", [math.waterloo.ca]

# Feature Extraction

- Goal: Identify a matrix $W$, such that $z = Wx$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

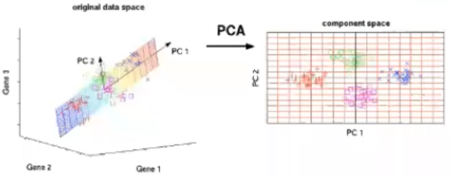



Courtesy: "PCA in Neuroscience", [math.waterloo.ca]

# Feature Extraction

- Goal: Identify a matrix $\boldsymbol{W}$, such that $\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$



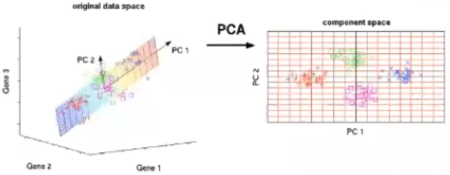

Courtesy: "PCA in Neuroscience", [math.waterloo.ca]

- We will learn specific algorithms that can learn $\boldsymbol{W}$ from data.

# Feature Extraction

- Goal: Identify a matrix $W$, such that $z = Wx$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$



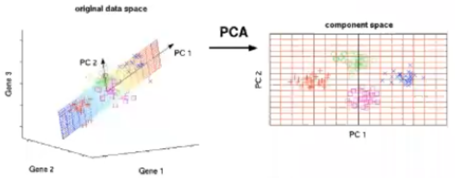

Courtesy: "PCA in Neuroscience", [math.waterloo.ca]

- We will learn specific algorithms that can learn $W$ from data.
- Reducing dimensionality helps in
  - Visualization (in a few dimensions)

# Feature Extraction

- Goal: Identify a matrix $\boldsymbol{W}$, such that $\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$



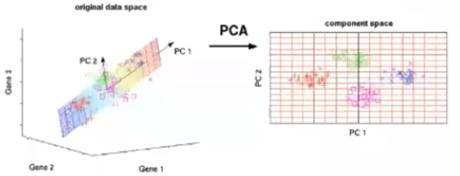

Courtesy: "PCA in Neuroscience", [math.waterloo.ca]

- We will learn specific algorithms that can learn $\boldsymbol{W}$ from data.
- Reducing dimensionality helps in
  - Visualization (in a few dimensions)
  - Also useful in learning efficient classifiers

# Feature Extraction

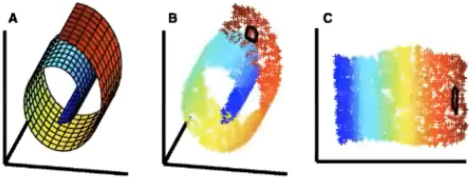
- Goal: Identify a matrix $W$, such that $z = Wx$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$




Courtesy: “PCA in Neuroscience”, [math.waterloo.ca]

- We will learn specific algorithms that can learn $W$ from data.
- Reducing dimensionality helps in
  - Visualization (in a few dimensions)
  - Also useful in learning efficient classifiers
- The above model is linear.
  Non-linear methods exists as well
  - $z = f_w(x)$



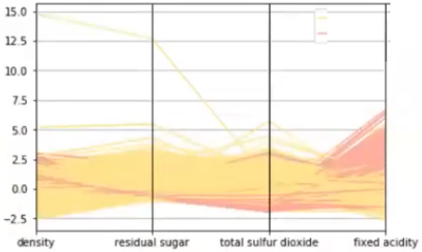

Courtesy: “DR by LLE”, by Roweis and Saul

# High Dimensional Data Visualisation

Direct Methods and Dimensionality Reduction

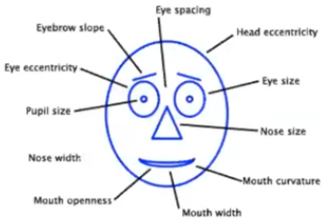
# Direct Visualisation

- Parallel Co-ordinates
  - Each vertical line is a dimension
  - A data item is connected by line segments
  - Large number of samples clutters the visualization



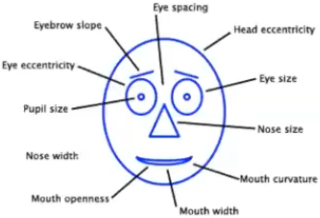
# Direct Visualisation

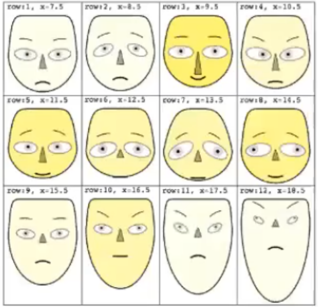
- Parallel Co-ordinates
  - Each vertical line is a dimension
  - A data item is connected by line segments
  - Large number of samples clutters the visualization
- Chernoff Faces [Herman Chernoff: 1973]



# Direct Visualisation

- Parallel Co-ordinates
  - Each vertical line is a dimension
  - A data item is connected by line segments
  - Large number of samples clutters the visualization
- Chernoff Faces [Herman Chernoff: 1973]



# Other Methods for Direct Visualisation

- A few approaches have been proposed to improve upon Chernoff Faces
- Empathic Visualization
    - Improve visual similarity of similar vectors

Image: The Empathic Visualization Algo: Chernoff Faces Revisited
http://www0.cs.ucl.ac.uk/staff/a.loizides/218.pdf

# Other Methods for Direct Visualisation

- A few approaches have been proposed to improve upon Chernoff Faces
- Empathic Visualization
  - Improve visual similarity of similar vectors
- Metaphoric 3D glyphs
  - Use metaphors other than faces (say Trees)

Image: The Empathic Visualization Algo: Chernoff Faces Revisited
http://www0.cs.ucl.ac.uk/staff/a.loizides/218.pdf

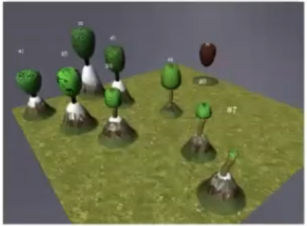


Image: "Beyond Chernoff faces: Multivariate visualization with metaphoric 3D glyphs", Marco Lardelli

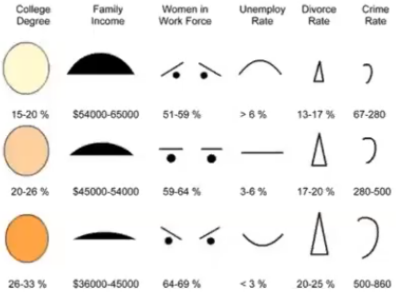
# Challenges in HD Visualisation

- Direct Methods does not preserve ordinal nature of features
    - e.g., In Chernoff faces, emotions are not ordinal in eyebrow slant


Courtesy: “The trouble with Chernoff”
[http://maphugger.com/post/44499755749/the-trouble-with-chernoff]

# Challenges in HD Visualisation

- Direct Methods does not preserve ordinal nature of features
  - e.g., In Chernoff faces, emotions are not ordinal in eyebrow slant
- Difficult to deal with large quantities of Data
  - Data Spread, Occlusion




Courtesy: "The trouble with Chernoff"
[http://maphugger.com/post/44499755749/the-trouble-with-chernoff]

# Challenges in HD Visualisation

- Direct Methods does not preserve ordinal nature of features
  - e.g., In Chernoff faces, emotions are not ordinal in eyebrow slant
- Difficult to deal with large quantities of Data
  - Data Spread, Occlusion
  - Often used in conjunction with clustering


Courtesy: "The trouble with Chernoff"
[http://maphugger.com/post/44499755749/the-trouble-with-chernoff]

# Challenges in HD Visualisation

- Direct Methods does not preserve ordinal nature of features
  - e.g., In Chernoff faces, emotions are not ordinal in eyebrow slant
- Difficult to deal with large quantities of Data
  - Data Spread, Occlusion
  - Often used in conjunction with clustering
- Modern approaches use DR for visualization


Courtesy: “The trouble with Chernoff”
[http://maphugger.com/post/44499755749/the-trouble-with-chernoff]

# Dimensionality Reduction for Visualisation

- Does not have to learn the mapping from $x$ to $z$ in a parametric form (say a matrix).

# Dimensionality Reduction for Visualisation

- Does not have to learn the mapping from $x$ to $z$ in a parametric form (say a matrix).
- The mapping need not be linear
- Need to map only training data (no test data)

# Dimensionality Reduction for Visualisation

- Does not have to learn the mapping from $x$ to $z$ in a parametric form (say a matrix).
- The mapping need not be linear
- Need to map only training data (no test data)
- Several Approaches
  - PCA, LDA, ISOMAP, LLE, SNE, t-SNE, UMAP, etc.

# Dimensionality Reduction for Visualisation

- Does not have to learn the mapping from $x$ to $z$ in a parametric form (say a matrix).
- The mapping need not be linear
- Need to map only training data (no test data)
- Several Approaches
  - PCA, LDA, ISOMAP, LLE, SNE, t-SNE, UMAP, etc.
- Need to maintain
  - Class Separation
  - Clusters in data