Travion White-Austin, Tallapaneni Venkateshwara Chowdary, Aastha Prasad
INFO 526
12/13/24
Write Up

Introduction:

In an increasingly data-driven world, visualizations have become essential tools for making sense of complex information. They not only help interpret patterns and trends but also enable effective communication of insights to diverse audiences. However, creating impactful visualizations requires more than simply plotting data, it involves choosing the right methods, ensuring accessibility, and understanding how different visualizations align with specific datasets.

Our project dives into key questions about the role and design of data visualizations:

- Why are visualizations critical for representing data effectively?
- What types of visualizations are available, and how can they be used?
- How can visualizations be made accessible to a wider audience?
- How do similar visualizations differ in their application depending on the data?

- What are the consequences of using an inappropriate visualization for a dataset?

By addressing these questions, this project aims to provide practical guidance for creating clear, accurate, and accessible visualizations, empowering users to present their data in meaningful and impactful ways.

Approach:

Our project is divided into three main sections: Different Visualizations, Accessibility, and Comparison Between Visualizations. In the Different Visualizations section, we explained six types of visualizations: bar graphs, scatter plots, line graphs, box plots, heatmaps, and forced connection plots, going over their benefits, typical use case. The Accessibility section focused on addressing a common accessibility issue in data visualization and proposing a practical solution to ensure inclusivity for all users. Finally, the Comparison Between Visualizations section compared three pairs of visualizations with similar functionalities or visual characteristics, emphasizing how differences in design or application can influence their effectiveness depending on the dataset. Together, these sections provide a comprehensive framework for understanding, designing, and evaluating data visualizations.

Code:

Bar graph:

```
df =
pd.read_csv('../input/palmer-archipelago-antarctica-penguin-data/penguins_size.
csv') df.head() df['species'].value_counts().iplot(kind='bar')
```

Scatter Plot:

```
 selected_columns2 <- data_200m[, c("Athlete",
"Wind","Time")]


UofA2$School = 'UofA2'
ASU2$School = 'ASU2'
NAU2$School = 'NAU2'
UW2$School = 'UW2'
Cal2$School = 'Cal2'
CU2$School = 'CU2'
WSU2$School = 'WSU2'
SU2$School = 'SU2'
USC2$School = 'USC2'


Merged_200m_data <- bind_rows(UofA2,ASU2,NAU2,UW2,Cal2,CU2,WSU2,SU2,USC2)


# correlation coefficient correlation <- cor(Merged_200m_data$Wind,
Merged_200m_data$Time, use = "complete.obs")
```

```
# Load necessary library
library(ggplot2)

# Create a scatter plot with regression line ggplot(data_200m, aes(x = Wind, y
= Time)) + geom_point(size = 3, alpha = 0.7, color = "blue") + # Scatter plot
with points geom_smooth(method = "lm", se = FALSE, color = "red") + #
Linear regression line scale_y_reverse() + # Reverse the y-axis so higher
times are at the top

 labs(
  title = "Scatter Plot: Wind Speed vs 200m
  Time", x = "Wind Speed (m/s)", y = "200m
  Time (seconds)"
 ) + theme_minimal() + # Clean theme theme( plot.title =
 element_text(hjust = 0.5, face = "bold", size = 16),
 axis.title = element_text(size = 14)
 )
```

Line Graph:

```
Load necessary
libraries library
library(dplyr)

Load the data from CSV file
```

```
data <- read.csv("C:/Users/aasth/OneDrive/data
viz/project-2-infocrew/data/world-education-data.csv")
```

```
pakistan_data <- data %>% filter(country_code == "PAK")
```

Inspect the data (optional)

```
head(pakistan_data) Create the plot for

Pakistan with specific colors for each line

ggplot(pakistan_data, aes(x = year)) +

geom_line(aes(y = gov_exp_pct_gdp,

color = 'Govt Expenditure (% of GDP)'),

size = 1) + geom_point(aes(y =

gov_exp_pct_gdp, color = 'Govt

Expenditure (% of GDP)'), size = 2) +

geom_line(aes(y = lit_rate_adult_pct,

color = 'Literacy Rate for Adults (%)'), size

= 1) + geom_point(aes(y =

lit_rate_adult_pct, color = 'Literacy Rate

for Adults (%)'), size = 2) +

geom_line(aes(y = pri_comp_rate_pct,

color = 'Primary Completion Rate (%)'),

size = 1) + geom_point(aes(y =

pri_comp_rate_pct, color = 'Primary

Completion Rate (%)'), size = 2) +

labs(title = "Trends of Educational
Indicators in Pakistan (1999-2004)", x =

"Year", y = "Percentage", color =

"Indicators") +

scale_color_manual(values = c('Govt

Expenditure (% of GDP)' = 'blue',

                              'Literacy Rate for Adults (%)' = 'green',

                              'Primary Completion Rate (%)' = 'red')) +

  theme_minimal() +

  theme(legend.position = "top")
```

Box Plot:

```
import pandas as pd import

matplotlib.pyplot as plt

import seaborn as sns

import numpy as np from

scipy.stats import norm

from scipy import stats

import warnings

warnings.filterwarnings('ign

ore')

%matplotlib inline

Input data files are available in the "../input/" directory on kaggle.

from subprocess import check_output print(check_output(["ls",

"../input"]).decode("utf8")) var = 'Country_code' data_plt =
pd.concat([data_scale['mpg'], data_scale[var]], axis=1) f, ax =
plt.subplots(figsize=(8, 6)) fig = sns.boxplot(x=var, y="mpg",
data=data_plt) fig.axis(ymin=0, ymax=1)
plt.axhline(data_scale.mpg.mean(),color='r',linestyle='dashed',linewi
dth=2)
```

Heatmap:

```
# Load necessary libraries
library(ggplot2)
library(reshape2)
library(RColorBrewer)

# Load the data from CSV file
data <- read.csv("C:/Users/aasth/OneDrive/data
viz/project-2-infocrew/data/world-education-data.csv")

# Select relevant columns for correlation data_corr <- data[,
c("gov_exp_pct_gdp", "lit_rate_adult_pct", "pri_comp_rate_pct",

              "pupil_teacher_primary", "pupil_teacher_secondary",
              "school_enrol_primary_pct", "school_enrol_secondary_pct",
              "school_enrol_tertiary_pct")]

# Calculate the correlation matrix corr_matrix
<- cor(data_corr, use = "complete.obs")
```

```
# Reshape the correlation matrix for ggplot
corr_melt <- melt(corr_matrix)

# Create the correlation heatmap ggplot(corr_melt, aes(Var1, Var2, fill =
value)) + geom_tile() + scale_fill_gradientn(colors = brewer.pal(11,
"Spectral")) + # Custom color scheme labs(title = "Correlation Heatmap of
Educational Indicators", x = "", y = "", fill = "Correlation") + theme_minimal()
+ theme(axis.text.x = element_text(angle = 45, hjust = 1), axis.text.y =
element_text(angle = 0, hjust = 1)) + coord_fixed()
```

Forced Connected Plot:

```
Load libraries
library(networkD3)
library(webshot)
library(htmlwidgets)
library(htmltools)

Read data data <- read.csv("C:/Users/aasth/OneDrive/data
viz/project-2-infocrew/data/imdb_edgelist.csv", nrows = 500, header = TRUE,
stringsAsFactors = FALSE)

Check if data is read correctly if
(is.null(data) nrow(data) == 0) {
stop("Failed to read data.")
}
```

```
Unique nodes nodes <- unique(c(data$From, data$To)) nodes_df <-
data.frame(name = nodes, group = 1) # Add a 'group' column for compatibility

Map node names to IDs data$FromID <-
match(data$From, nodes) - 1 data$ToID
<- match(data$To, nodes) - 1

Check if IDs are mapped correctly if
(any(is.na(data$FromID)) any(is.na(data$ToID)))
{ stop("Failed to map node names to IDs.")

}

Create force network graph
graph <- forceNetwork(

  Links = data.frame(source = data$FromID, target = data$ToID, value = data$Strength),
  Nodes = nodes_df,
  Source = "source",
  Target = "target",
  Value = "value",
  NodeID = "name", Group = "group", fontSize = 16,
          # Increase font size fontFamily = "Arial",       # Use
  a bold-compatible font like Arial opacity = 0.9,  #
  Transparency for nodes

zoom = TRUE,           # Enable zoom
```

```
  linkDistance = 100,   # Distance between nodes
  opacityNoHover = 0.1 # Highlight connections on hover
)
```

Save the force network graph as an HTML file html_file

```
<- "force_graph.html" saveNetwork(graph, file =
html_file, selfcontained = TRUE)
```

Convert the saved HTML to PNG using webshot webshot::webshot(html_file,

```
file = "force_graph.png", vwidth = 1200, vheight = 800)
```

Accessibility Section: (All Plots)

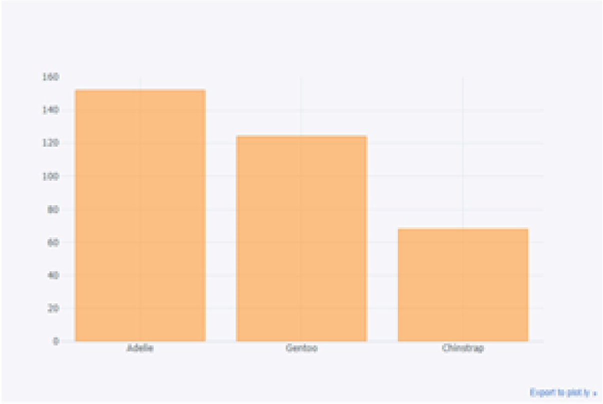
```
{r}
library(palmerpenguins)
library(dplyr)
library(ggplot2) {r}
head(penguins)
theme_set(theme_minima
l())
{r} ggplot(data = penguins, aes(x = flipper_length_mm, y =
body_mass_g)) + geom_point(aes(color = species), size = 2) +
scale_color_manual(values = c("#f27874","#00bc3e","#699afb"))
{r} ggplot(data = penguins, aes(x = flipper_length_mm, y =
body_mass_g)) + geom_point(aes(color = species, shape =
species), size = 2) + scale_color_manual(values =

c("#f27874","#00bc3e","#699afb"))

{r} ggplot(data = penguins, aes(x = flipper_length_mm, y =

body_mass_g)) + geom_point(aes(color = species), size = 2) +

scale_color_manual(values = c("#340042","#1f8079","#fce41e"))
```

```
{r}

 ggplot(data = penguins, aes(x = flipper_length_mm, y =

  body_mass_g)) + geom_point(aes(color = species, shape =

  species), size = 2) + scale_color_manual(values =

  c("#340042","#1f8079","#fce41e"))
```

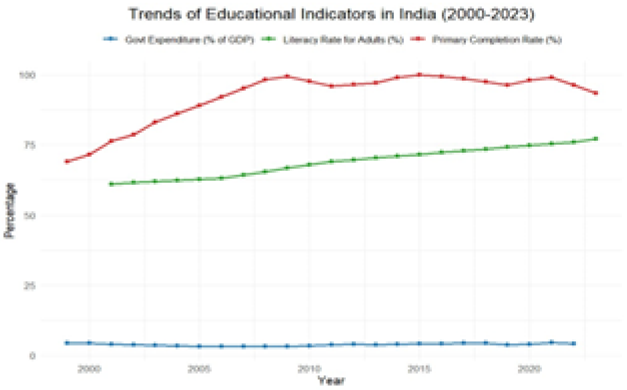
Visualizations:

Types of Visualizations:

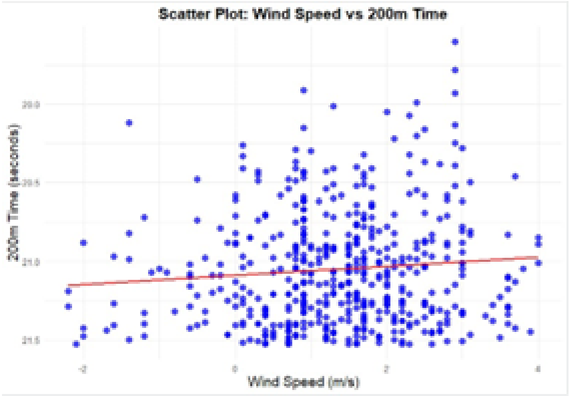
Bar graph:



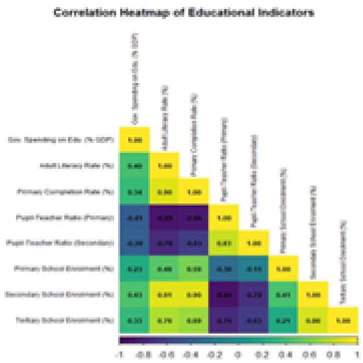
Line Chart:



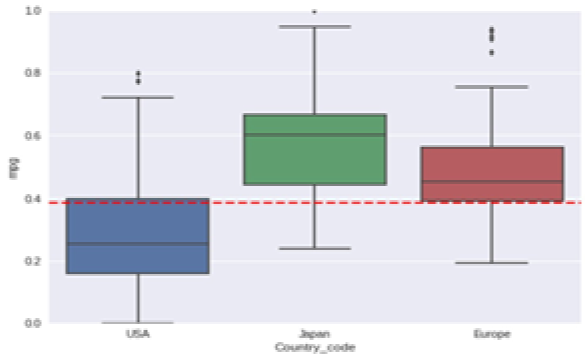
Scatter plot:



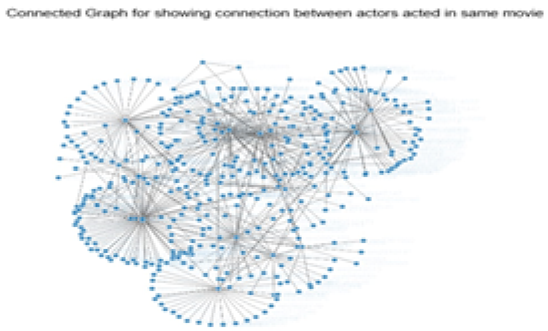
Heatmap:



Boxplot:



Force Connected graph:



Analysis:

Bar Plot:

- Best for comparing discrete categories or groups.
- Commonly used to show frequencies, counts, or averages across distinct groups.

Scatter Plot:

- Ideal for identifying relationships or correlations between two continuous variables.
- Frequently used in regression analysis or to detect outliers in datasets.

Line Plot:

- Used for showing trends or changes over time.
- Suitable for time-series data to highlight patterns or long-term movements.

Box Plot:

- Useful for visualizing the spread, central tendency, and outliers in continuous data.
- Common in statistical analysis to compare distributions across groups.

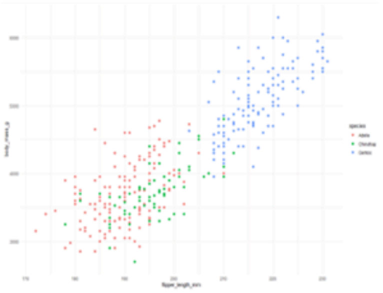
Heatmap:

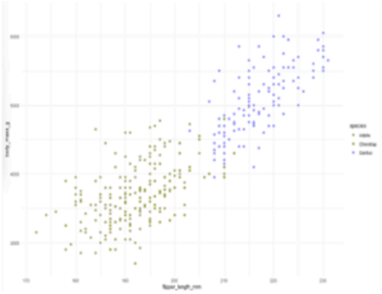
- Helps visualize correlations or relationships in a matrix format.
- Frequently used in feature selection, gene expression data, or educational performance indicators.

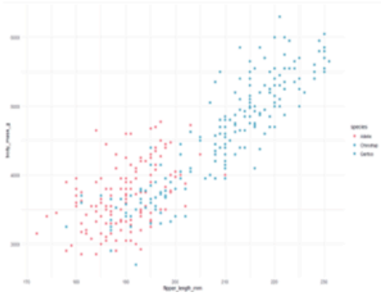
Network Plot:

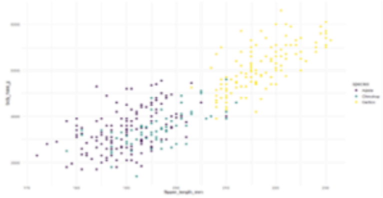
- Ideal for visualizing connections, relationships, or hierarchies in data.
- Commonly used in social network analysis or understanding interactions in large systems.

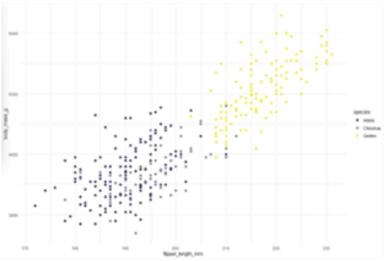
Accessibility in Visualizations: Color and Shape

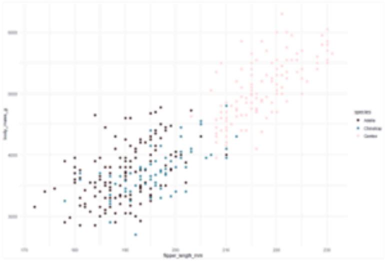


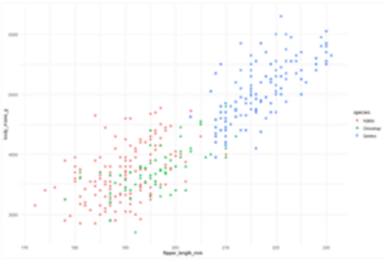












Accessibility in visualizations ensures that data is understandable and usable by everyone, including individuals with visual, cognitive, or motor impairments. To make visualizations accessible, careful use of color and shape is critical. Below are the key considerations and practices:

1. Accessibility Considerations for Color

Color is one of the most common ways to represent data, but it poses challenges for individuals with color blindness or visual impairments.

- **Avoid Color-Only Encoding**:
  - Do not rely solely on color to distinguish data points. Use additional attributes like shapes, textures, or labels.
  - Example: A scatter plot can use different **shapes** for data groups instead of colors alone.
- **Color-Blind Friendly Palettes**:
  - Use color schemes that accommodate common types of color blindness (e.g., red-green blindness).
  - Preferred colors: Blues, yellows, and purples are often distinguishable for most users.
- **Contrast and Clarity**:
  - Ensure high contrast between colors and backgrounds to improve readability.
  - Use tools to check contrast ratios (e.g., WCAG contrast checker).
- **Provide Legends and Labels**:

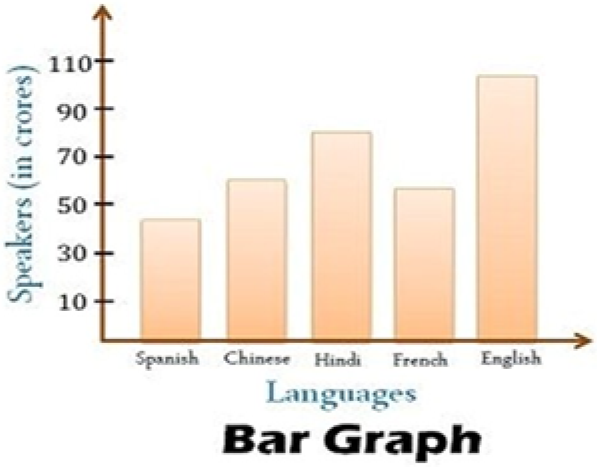
    - Include clear legends, annotations, or tooltips to explain colors. This supports users who cannot distinguish colors.

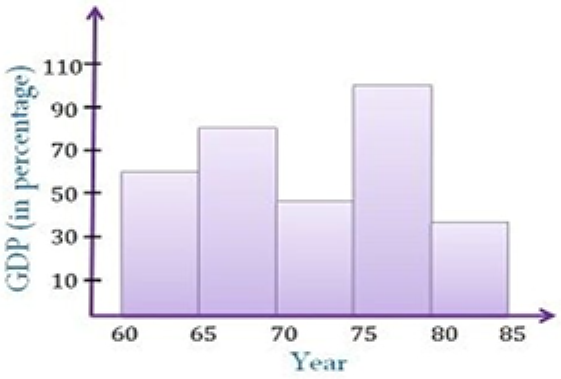
2. Accessibility Considerations for Shape

Shapes can provide an alternative or complement to color for differentiating data categories.

- **Use Distinct Shapes for Categorical Data**:
    - Example: Use circles, squares, and triangles to represent different categories in scatter plots.
    - This helps users with color vision deficiencies differentiate between groups.
- **Combine Shape and Color**:
    - Use color and shape redundantly for maximum clarity. For example, in a scatter plot:
        - **Group A**: Blue circles
        - **Group B:** Red triangles

Comparison Visualizations:



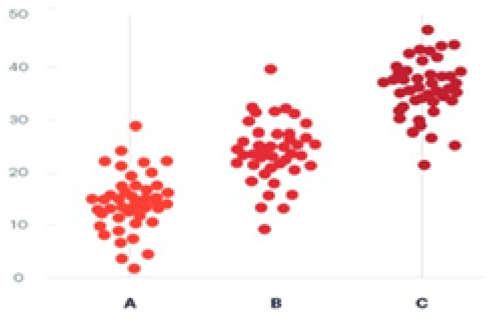


Histogram

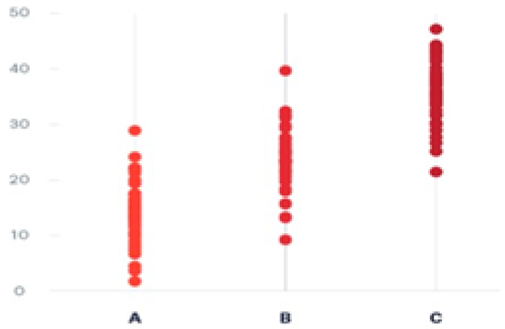
Bar Graph: Represents categorical data, with bars spaced apart to show distinct categories. Each bar's height reflects the frequency or value of the category.

Histogram: Represents continuous data by dividing it into intervals (bins). The bars are adjacent, showing the distribution of data within each bin.

Jitter Plot



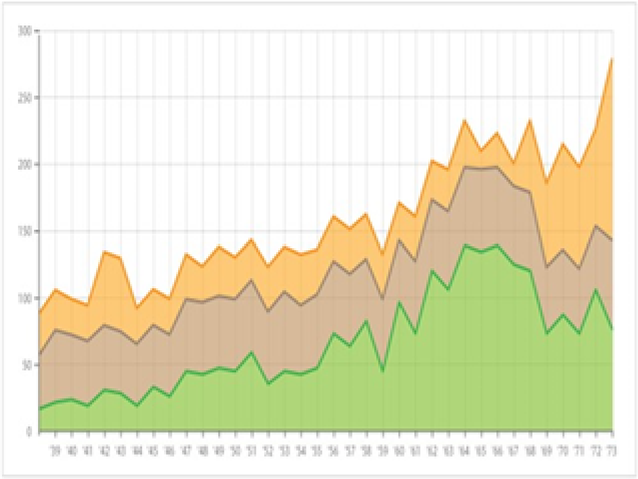
Strip Plot



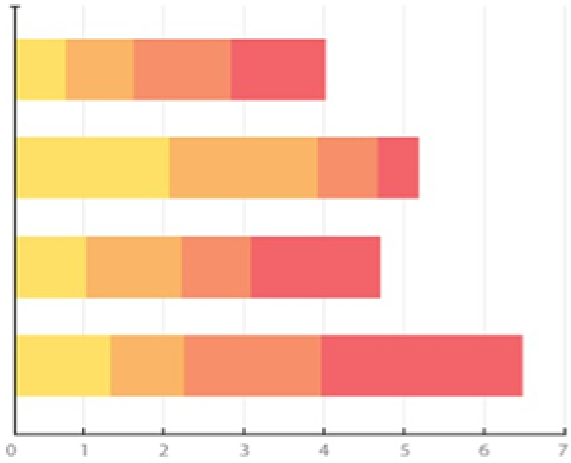
Jitter Plot: Adds random noise to the position of points in a scatter plot to avoid overlap and make individual data points distinguishable, especially when values are concentrated.

Strip Plot: Displays data points along a single axis without any random noise, leading to overlapping points when values are the same, making it harder to see individual data in dense

Stacked Area



Stacked Bar:



Stacked Area Plot: Represents quantities over time, stacking areas on top of each other to show cumulative trends. It emphasizes how each part contributes to the whole over a continuous range (e.g., time series data).

Stacked Bar Plot: Displays parts of a whole for categorical data, with bars divided into segments stacked vertically or horizontally. It highlights the contribution of each category to the total in a discrete way.

Summary:

In our project, we aimed to provide the audience with key considerations to keep in mind before starting their visualizations. We began by introducing a selection of versatile visualization types, showcasing their benefits and use cases to help the audience choose the most suitable options for their data. Next, we addressed the accessibility aspects of visualizations, emphasizing the importance of inclusivity and usability. Finally, we explored how the context both in terms of what the visualization is expected to convey and the type of data being visualized can significantly impact the choice of visualization. To illustrate this, we provided three examples where these contextual factors influenced the effectiveness of the selected visualizations.