# Data Visualization: Assignment 2

Student: Baris Surmelioglu

## Three preliminary visualizations

**For task 2 I select the most interesting variables or samples for visualization, so I focus on the top-n performing distance sets based on mean ROC AUC scores across all datasets. Here's how we can do it:**

- Calculate the mean ROC AUC score for each distance set across all datasets.
- Select the top-n distance sets with the highest mean ROC AUC scores.
- Use these top-n distance sets for visualization to focus on the most promising predictors.

So I did as:

```
> mean_ROC_AUC_per_distance_set <- combined_df %>%
+     group_by(distances) %>%
+     summarize(mean_ROC_AUC = mean(auc))
> top_n_distance_sets <- mean_ROC_AUC_per_distance_set %>%
+     top_n(20, wt = mean_ROC_AUC)
> filtered_combined_df <- combined_df %>%
+     filter(distances %in% top_n_distance_sets$distances)
```

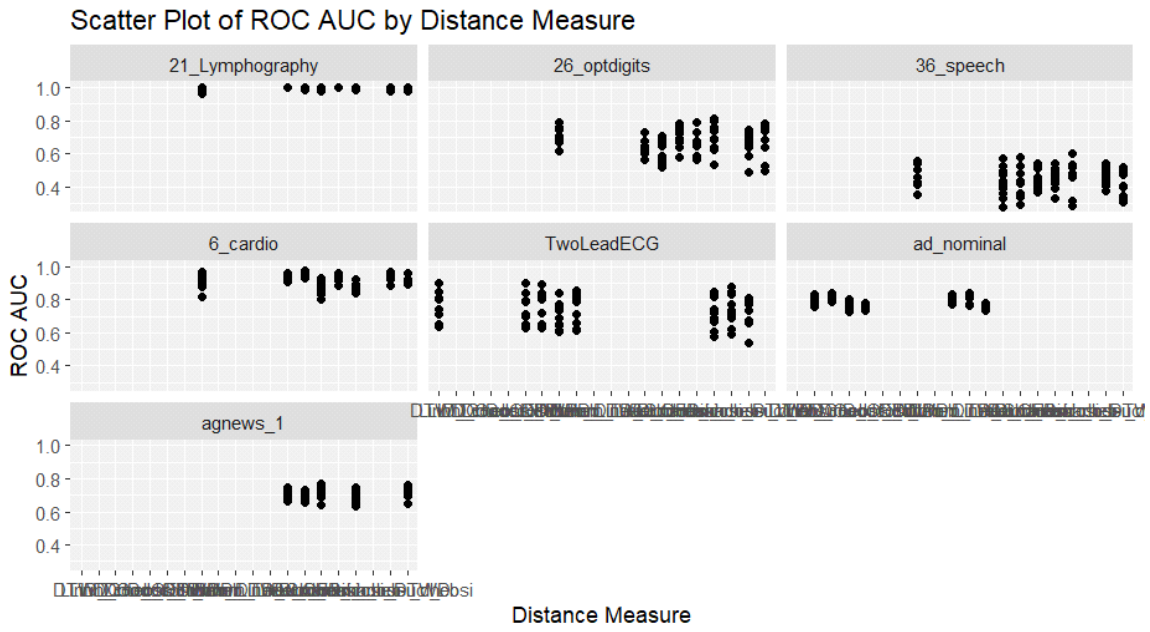
selecting the top 20 distance sets based on their mean ROC AUC scores, and filtered the combined dataframe to include only the data related to these top distance sets.

### Sketch 1

**Scatter Plot Matrix:**

- Create a scatter plot matrix where each point represents the ROC AUC score for different distance measures.
- Use different colors or symbols to distinguish between different dataset types.
- This visualization will allow you to observe the relationships between ROC AUC scores for different distance measures across all datasets.

# Implementation 1




# Implementation code 1

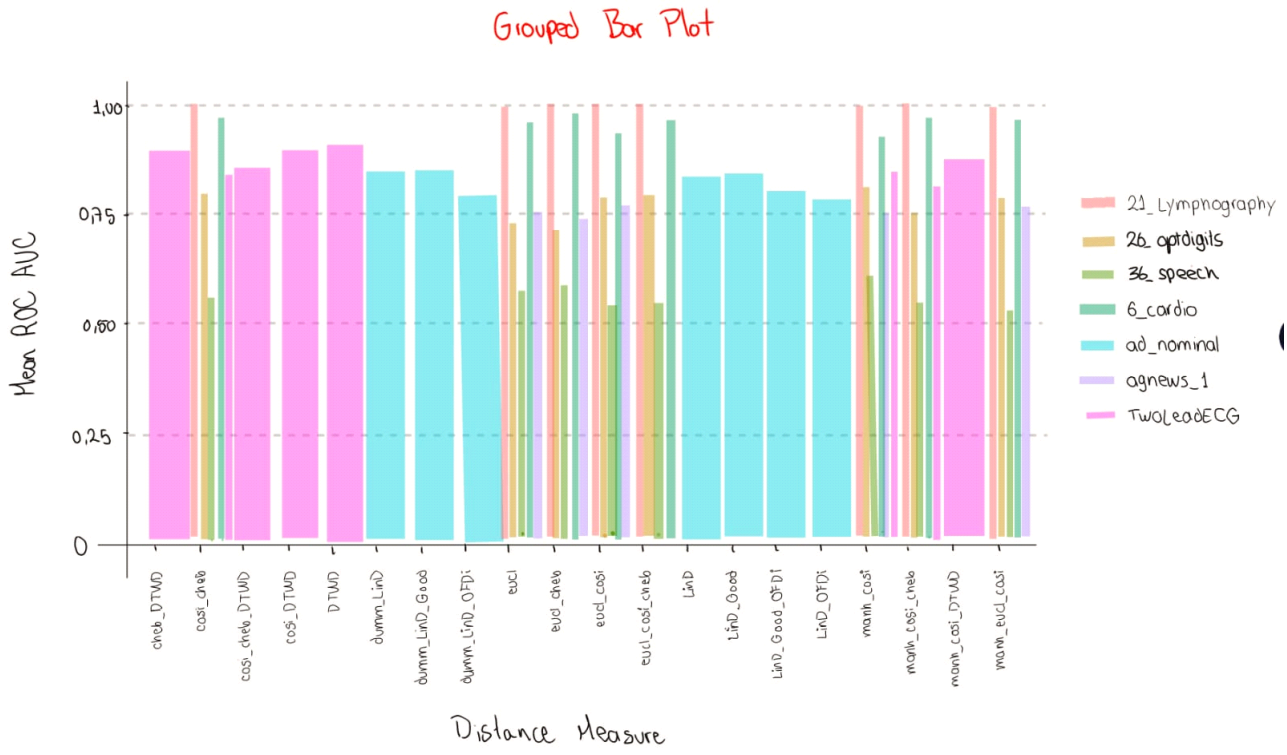
```
library(ggplot2)

library(dplyr)


setwd("C:/Users/MSI/Desktop/TEST/selected_distances")


binary <- read.csv("binary.csv")
```

```
graph <- read.csv("graph.csv")

nlp <- read.csv("nlp.csv")

numerical <- read.csv("numerical.csv")

timeseries <- read.csv("timeseries.csv")


binary$dataset_type <- "binary"

graph$dataset_type <- "graph"

nlp$dataset_type <- "nlp"

numerical$dataset_type <- "numerical"

timeseries$dataset_type <- "timeseries"


combined_df <- bind_rows(binary, graph, nlp, numerical, timeseries)


mean_ROC_AUC_per_distance_set <- combined_df %>% group_by(distances) %>%
summarize(mean_ROC_AUC = mean(auc))


top_n_distance_sets <- mean_ROC_AUC_per_distance_set %>% top_n(20, wt = mean_ROC_AUC)


filtered_combined_df <- combined_df %>% filter(distances %in% top_n_distance_sets$distances)


scatter_plots <- ggplot(filtered_combined_df, aes(x = distances, y = auc)) +
 geom_point() +
 facet_wrap(~ dataset_name) +
 labs(x = "Distance Measure", y = "ROC AUC") +
 ggtitle("Scatter Plot of ROC AUC by Distance Measure")


print(scatter_plots)
```

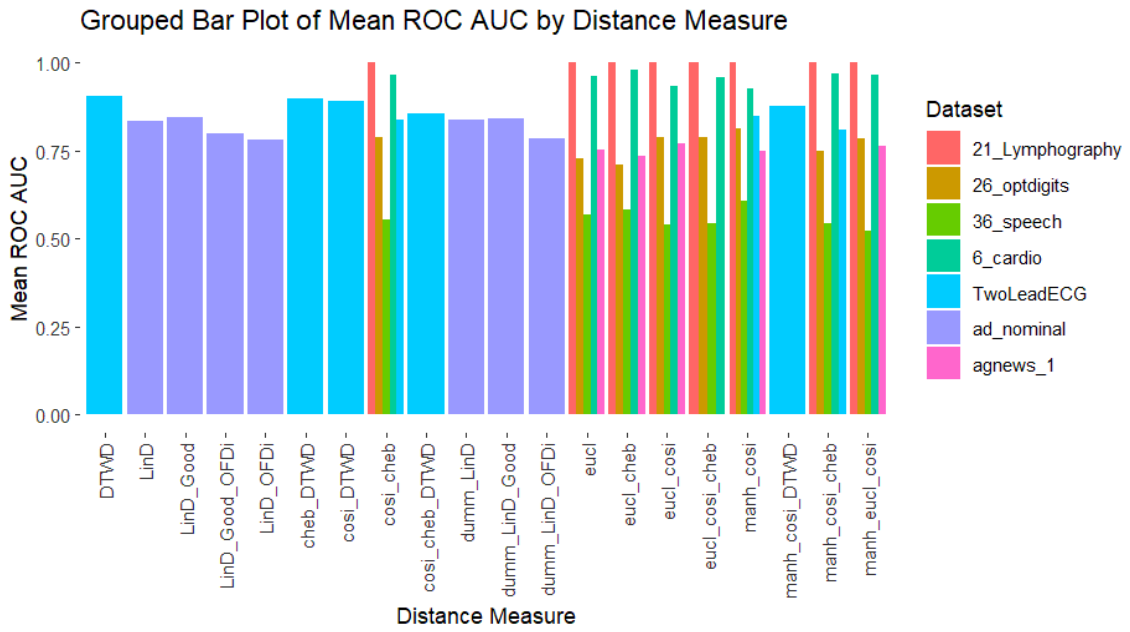
# Sketch 2

**Grouped Bar Plot:**

- Create a grouped bar plot where each bar represents the mean ROC AUC score for a specific distance measure, grouped by dataset type.
- Use different colors for each dataset type to distinguish between them.
- This visualization will allow you to compare the mean ROC AUC scores for different distance measures within each dataset type.



# Implementation 2




# Implementation code 2

```
library(ggplot2)

library(dplyr)


setwd("C:/Users/MSI/Desktop/TEST/selected_distances")
```

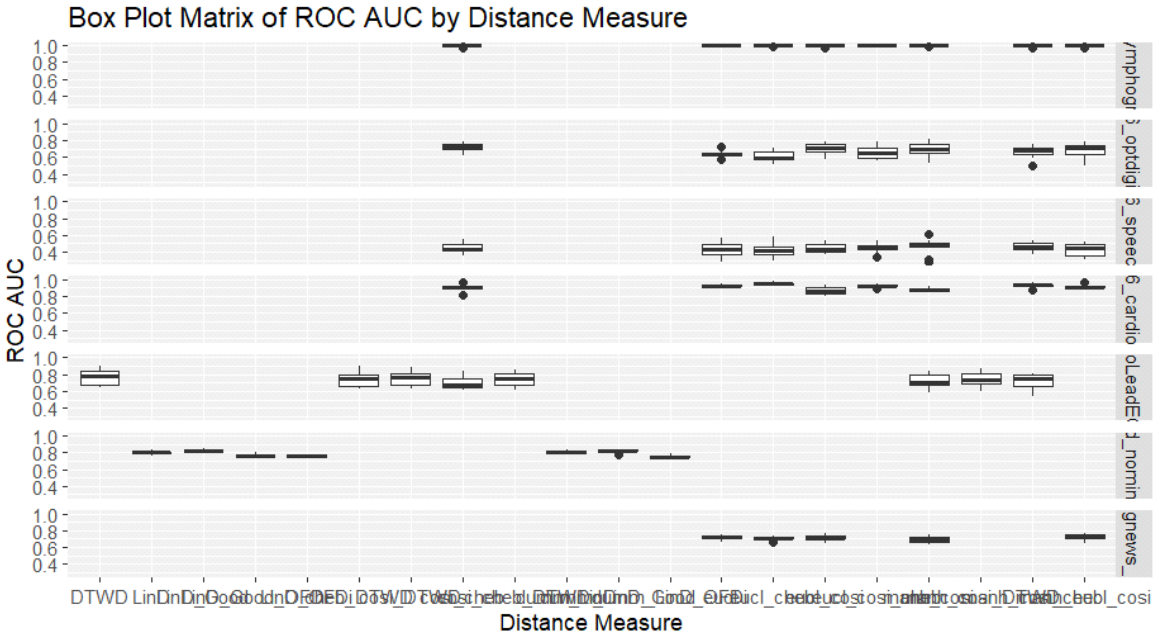
```
binary <- read.csv("binary.csv")
graph <- read.csv("graph.csv")
nlp <- read.csv("nlp.csv")
numerical <- read.csv("numerical.csv")
timeseries <- read.csv("timeseries.csv")

binary$dataset_type <- "binary"
graph$dataset_type <- "graph"
nlp$dataset_type <- "nlp"
numerical$dataset_type <- "numerical"
timeseries$dataset_type <- "timeseries"

grouped_bar_plot <- ggplot(filtered_combined_df, aes(x = distances, y = auc, fill = dataset_name)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Distance Measure", y = "Mean ROC AUC", fill = "Dataset") +
  ggtitle("Grouped Bar Plot of Mean ROC AUC by Distance Measure") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

print(grouped_bar_plot)
```

# Sketch 3

**Box Plot Matrix:**

- Create a matrix of box plots where each box plot represents the distribution of ROC AUC scores for a specific distance measure.
- Arrange the box plots in rows and columns, with each row/column representing a different dataset type.
- This visualization will allow you to compare the distribution of ROC AUC scores for different distance measures across all datasets.

# Implementation 3



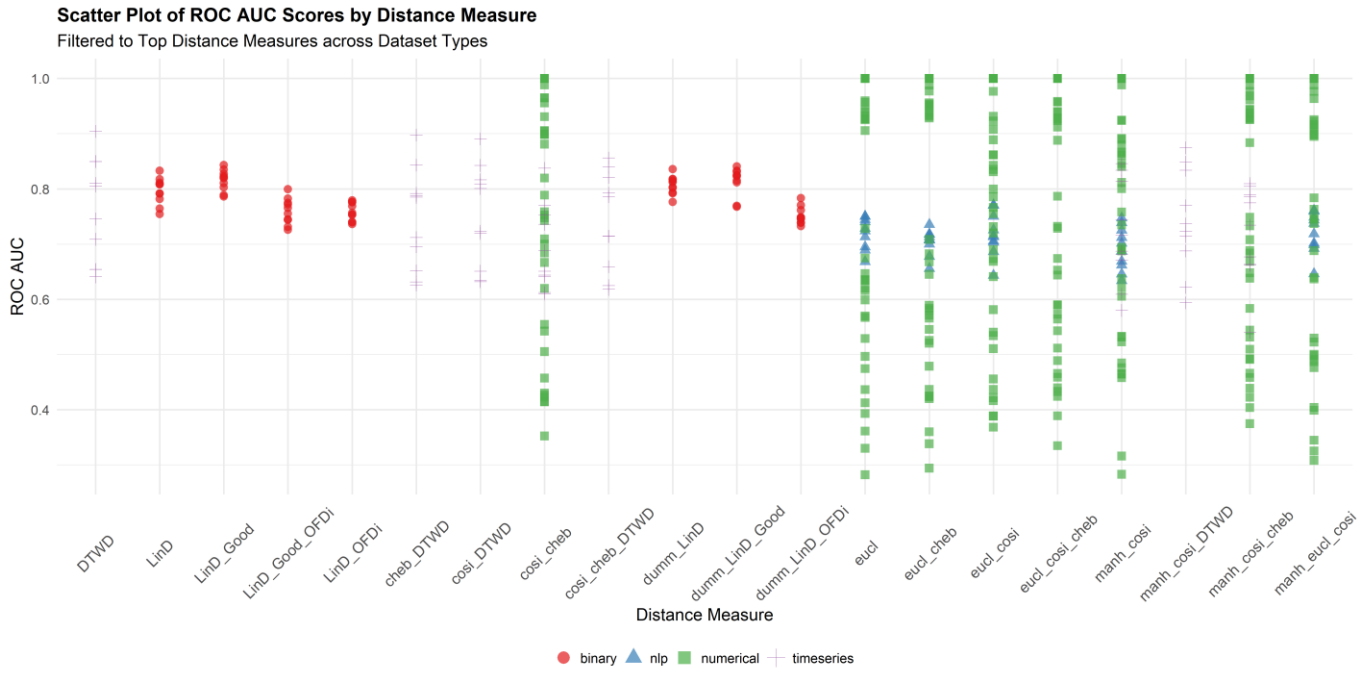

# Implementation code 3

```
library(ggplot2)

library(dplyr)


setwd("C:/Users/MSI/Desktop/TEST/selected_distances")
```

```
binary <- read.csv("binary.csv")
graph <- read.csv("graph.csv")
nlp <- read.csv("nlp.csv")
numerical <- read.csv("numerical.csv")
timeseries <- read.csv("timeseries.csv")

binary$dataset_type <- "binary"
graph$dataset_type <- "graph"
nlp$dataset_type <- "nlp"
numerical$dataset_type <- "numerical"
timeseries$dataset_type <- "timeseries"

box_plot_matrix <- ggplot(filtered_combined_df, aes(x = distances, y = auc)) +
  geom_boxplot() +
  facet_grid(dataset_name ~ .) +
  labs(x = "Distance Measure", y = "ROC AUC") +
  ggtitle("Box Plot Matrix of ROC AUC by Distance Measure")

print(box_plot_matrix)
```

# Selected final visualization

## Implementation




## Implementation code

```
library(ggplot2)

library(dplyr)


setwd("C:/Users/MSI/Desktop/TEST/selected_distances")


binary <- read.csv("binary.csv")

graph <- read.csv("graph.csv")
```

```
nlp <- read.csv("nlp.csv")
numerical <- read.csv("numerical.csv")
timeseries <- read.csv("timeseries.csv")

binary$dataset_type <- "binary"
graph$dataset_type <- "graph"
nlp$dataset_type <- "nlp"
numerical$dataset_type <- "numerical"
timeseries$dataset_type <- "timeseries"

final_scatter_plot <- ggplot(filtered_combined_df, aes(x = distances, y = auc, color = dataset_type)) +
 geom_point(aes(shape = dataset_type), size = 4, alpha = 0.7) +
 scale_color_brewer(palette = "Set1") +
 theme_minimal(base_size = 18) +
 theme(
  legend.position = "bottom",
  legend.title = element_blank(),
  axis.text.x = element_text(angle = 45, vjust = 0.5, size = 16),
  axis.title = element_text(size = 18),
  plot.title = element_text(size = 20, face = "bold"),
  plot.subtitle = element_text(size = 18),
  plot.caption = element_text(size = 16)
 ) +
 labs(
  title = "Scatter Plot of ROC AUC Scores by Distance Measure",
  subtitle = "Filtered to Top Distance Measures across Dataset Types",
  x = "Distance Measure",
  y = "ROC AUC",
  color = "Dataset Type",
```

```
    shape = "Dataset Type"
  ) +
  guides(color = guide_legend(override.aes = list(size = 6)))


print(final_scatter_plot)
```