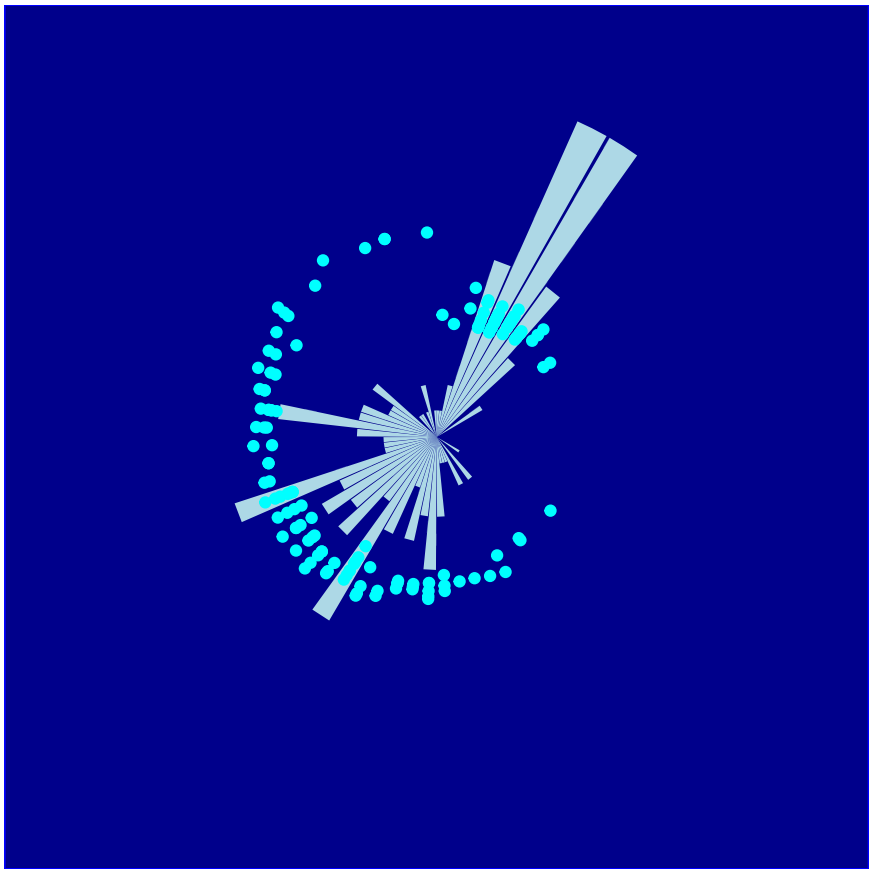
Battle of Blue

Gavin Martinez

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.2     v purrr   1.0.1
## v tibble  3.2.1     v dplyr   1.1.1
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.2     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
iris %>%
  ggplot(aes(x = Petal.Length)) +
  geom_bar(fill = "lightblue") +
  geom_point(aes(y = Sepal.Length), color = "cyan") +
  coord_polar() +
  theme_void() +
  theme(legend.position = "none") +
  theme(panel.background = element_rect(fill = "darkblue", colour = "blue"))
```



Title: “Battle of Blue”

Artist: Gavin Martinez

Medium: Digital Art (created using R)

Dimensions: Variable (Digital Display)

Description:

“Battle of Blue,” a piece of digital artwork by Gavin Martinez, showcases a collision of data visualization and artistic expression. Inspired by the iris dataset, this piece presents a captivating juxtaposition of polar coordinate bar plot and overlayed polar jitter plot.

In the bar plot, petal length variable takes center stage, showcased in light blue bins, forming an elegant and structured representation of this botanical characteristic. A contrasting overlay emerges in the form of cyan points, depicting the relationship between petal length and sepal length through the polar jitter plot.

The carefully selected dark blue background sets the stage for this artistic exploration, providing a stark yet captivating backdrop for the “Battle of Blue.” Through his masterful use of R, Martinez seamlessly merges the scientific realm with artistic expression, inviting viewers to engage in a visual dialogue where data and creativity converge.

About the Code By using geom_bar() I was able to create a comfortable view of the Petal.Length variable. Later, I included geom_point() to add more texture. It quickly became clear that those two plots are not usually overlaid on one another, one being a two-variable plot and the other being just a one-variable plot. However, the inclusion of the Sepal.Length variable allowed me to include both plots. What really brings it all together is the circular image being displayed. This is is due to the coord_polar() function that transitions the regular cartesian coordinates into the polar ones. Theme adjustments such as theme_void(), legend.position() and panel.background() allowed me to get rid of automatic titles and legends and include more color to elements such as the background.