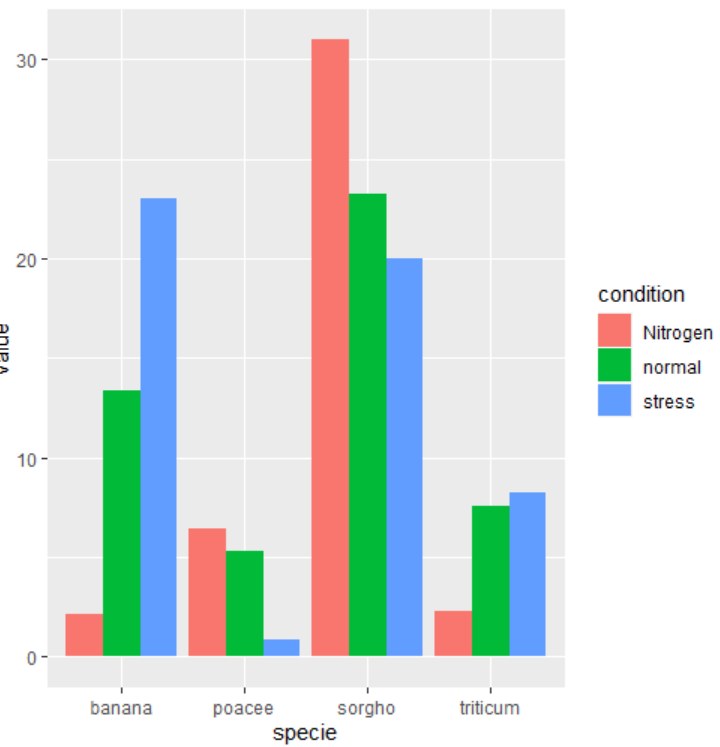
**Comparing Categories**

**Stacked Bar Chart**



# library

library(ggplot2)

# create a dataset

specie <- c(rep("sorgho" , 3) , rep("poacee" , 3) , rep("banana" , 3) , rep("triticum" , 3) )

condition <- rep(c("normal" , "stress" , "Nitrogen") , 4)

value <- abs(rnorm(12 , 0 , 15))

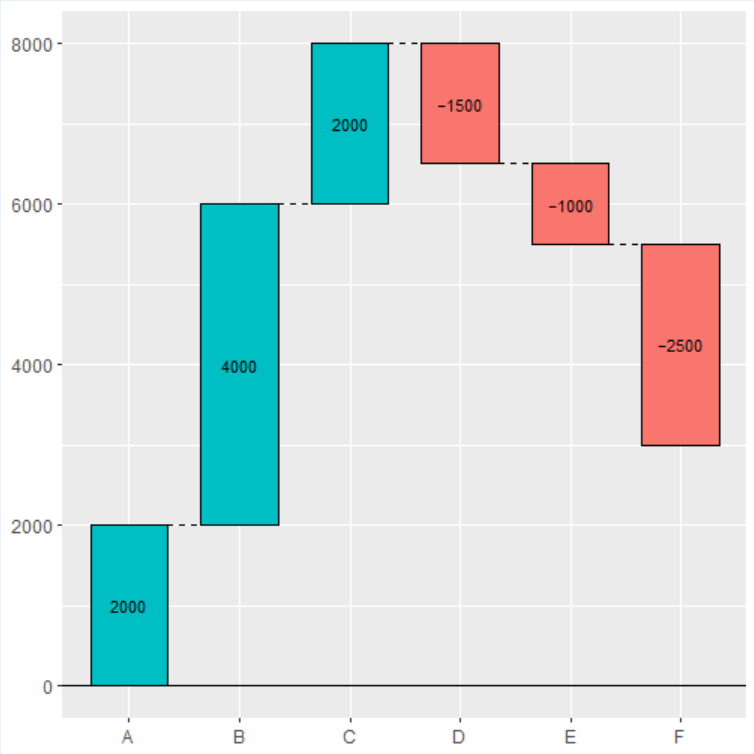
data <- data.frame(specie,condition,value)

# Grouped

ggplot(data, aes(fill=condition, y=value, x=specie)) +

geom\_bar(position="dodge", stat="identity")

**Waterfall Chart**



# install.packages("ggalluvial")

library(ggalluvial)

group <- LETTERS[1:6]

value <- c(2000, 4000, 2000,

-1500, -1000, -2500)

df <- data.frame(x = group, y = value)

# install.packages("waterfalls")

library(waterfalls)

waterfall(df)

# Equivalent to:

waterfall(values = value, labels = group)

**TIME**

**Gantt Chart**



#create data frame

data <- data.frame(name = c('Bob', 'Greg', 'Mike', 'Andy'),

start = c(4, 7, 12, 16),

end = c(12, 11, 8, 22),

shift\_type = c('early', 'mid\_day', 'mid\_day', 'late')

)

#create gantt chart that visualizes start and end time for each worker

ggplot(data, aes(x=start, xend=end, y=name, yend=name, color=shift\_type)) +

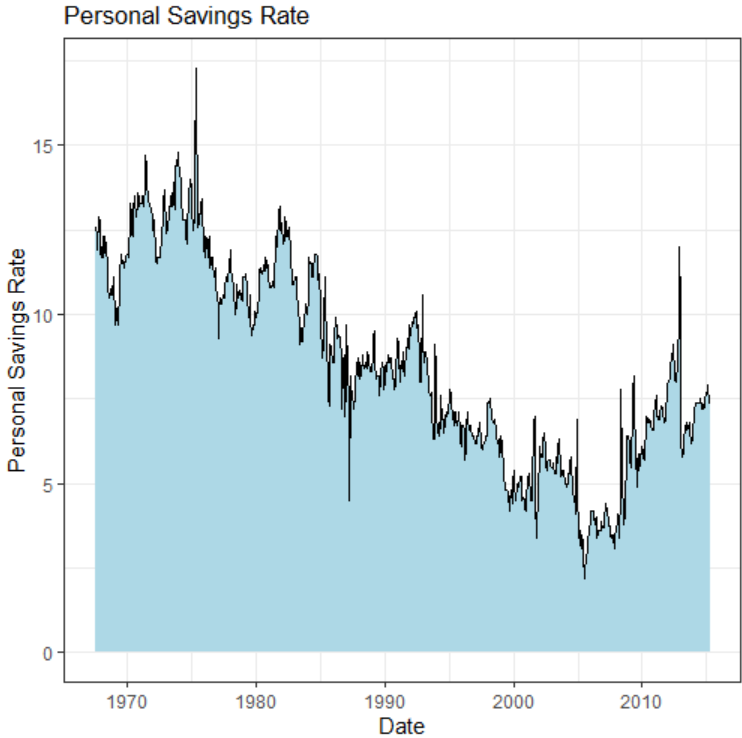
theme\_bw()+ #use ggplot theme with black gridlines and white background

geom\_segment(size=8) +

#increase line width of segments in the chart

labs(title='Worker Schedule', x='Time', y='Worker Name')

**Area Chart**



ggplot(economics, aes(x = date, y = psavert)) +

geom\_area(fill="lightblue", color="black") +

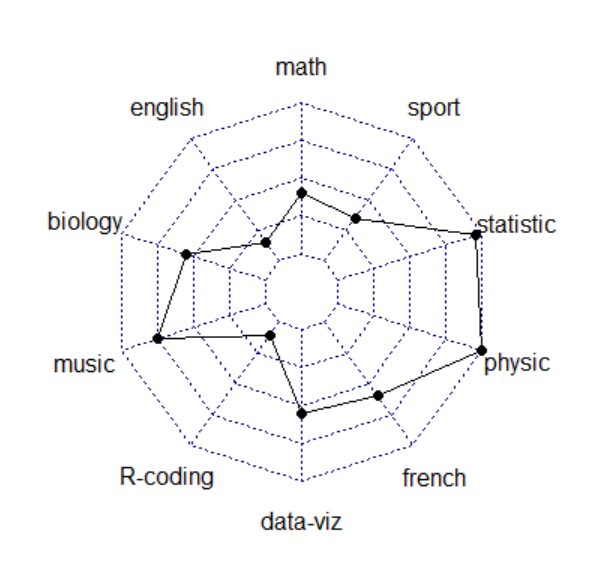
labs(title = "Personal Savings Rate",

x = "Date",

y = "Personal Savings Rate")

**Relationships**

**Radar Chart**



# Library

library(fmsb)

# Create data: note in High school for Jonathan:

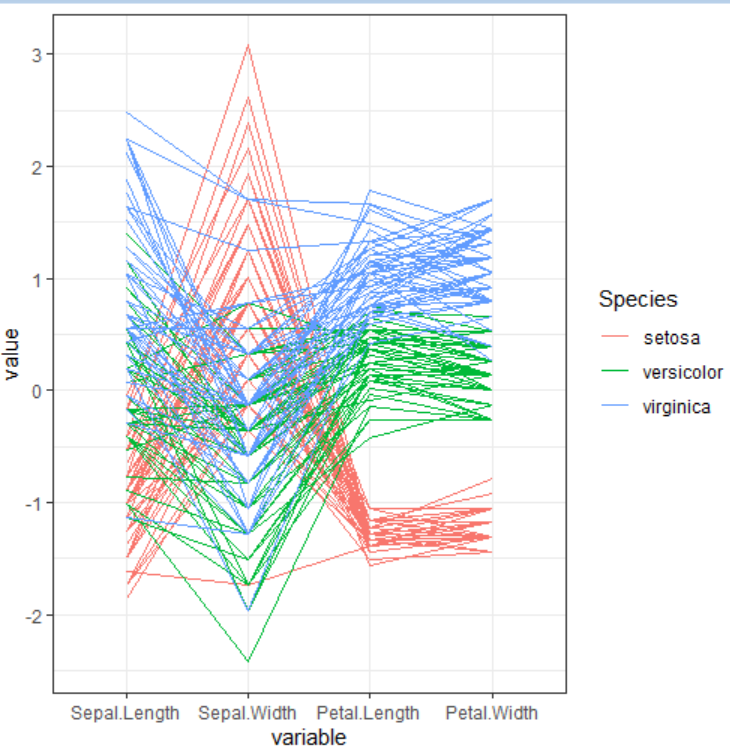
data <- as.data.frame(matrix( sample( 2:20 , 10 , replace=T) , ncol=10))

colnames(data) <- c("math" , "english" , "biology" , "music" , "R-coding", "data-viz" , "french" , "physic", "statistic", "sport" )

data <- rbind(rep(20,10) , rep(0,10) , data)

radarchart(data)

**Parallel Coordinates Chart**



# Libraries

library(GGally)

# Data set is provided by R natively

data <- iris

# Plot

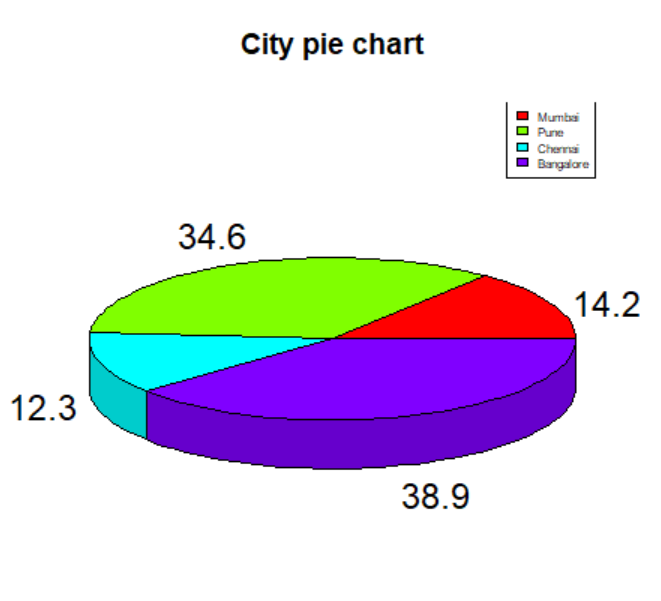
ggparcoord(data,

columns = 1:4, groupColumn = 5

)

**Part to Whole**

**Pie Chart**



# Get the library.

library(plotrix)

# Create data for the graph.

geeks <- c(23, 56, 20, 63)

labels <- c("Mumbai", "Pune", "Chennai", "Bangalore")

piepercent<- round(100 \* geeks / sum(geeks), 1)

# Plot the chart.

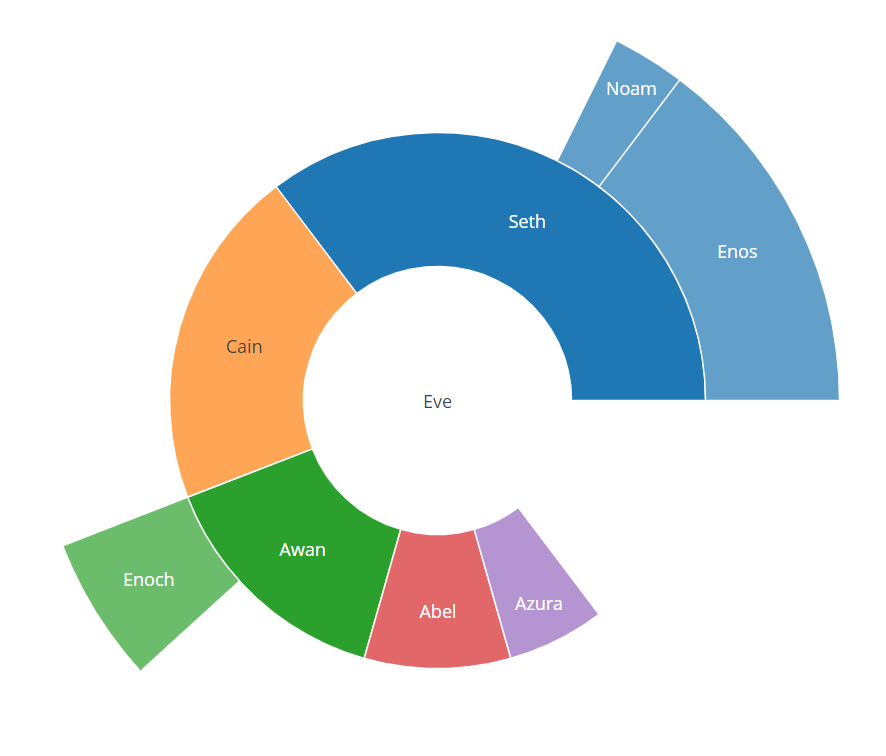
pie3D(geeks, labels = piepercent,

main = "City pie chart", col = rainbow(length(geeks)))

legend("topright", c("Mumbai", "Pune", "Chennai", "Bangalore"),

cex = 0.5, fill = rainbow(length(geeks)))

**Sunburst Diagram**



library(plotly)

fig <- plot\_ly(

labels = c("Eve", "Cain", "Seth", "Enos", "Noam", "Abel", "Awan", "Enoch", "Azura"),

parents = c("", "Eve", "Eve", "Seth", "Seth", "Eve", "Eve", "Awan", "Eve"),

values = c(10, 14, 12, 10, 2, 6, 6, 4, 4),

type = 'sunburst'

)

fig

**Qualitative**

**Word Cloud**



library(tidytext) # for transforming data

library(tidyverse) # for data wrangling

library(Cairo) # for ggwordcloud package to run

library(ggwordcloud) # to render wordclouds

library(DiagrammeR) # to make qual coding

library(rgexf) #To create graph files

glasgowData <- read.table("C:/Users/HP/Downloads/Glasgow.txt", header = FALSE, fill = TRUE, encoding = "UTF-8")

tidy\_glasgow <- gather(glasgowData, key, word) %>%

select(word)

#checks how many unique words there are in total

unique(tidy\_glasgow$word) %>%

length()

tokens <- tidy\_glasgow %>%

unnest\_tokens(word, word) %>%

count(word, sort = TRUE) %>%

ungroup()

top\_10 <- tokens %>%

head(10)

knitr::kable(top\_10, caption = "Top ten all words table")

# removing stop words with built in tidytext package

data("stop\_words")

tokens\_clean <- tokens %>%

anti\_join(stop\_words)

# removing numbers

nums <- tokens\_clean %>% filter(str\_detect(word, "^[0-9]")) %>% select(word) %>% unique()

tokens\_clean <- tokens\_clean %>%

anti\_join(nums, by = "word")

unique\_stopwords <- data.frame(word = c("glasgow's","city's", "scottish"))

tokens\_clean <- tokens\_clean %>%

anti\_join(unique\_stopwords, by = "word")

top\_10\_clean <- tokens\_clean %>%

head(10)

knitr::kable(top\_10\_clean, caption = "Top ten clean table")

wordcloudplot <- head(tokens\_clean, 50) %>%

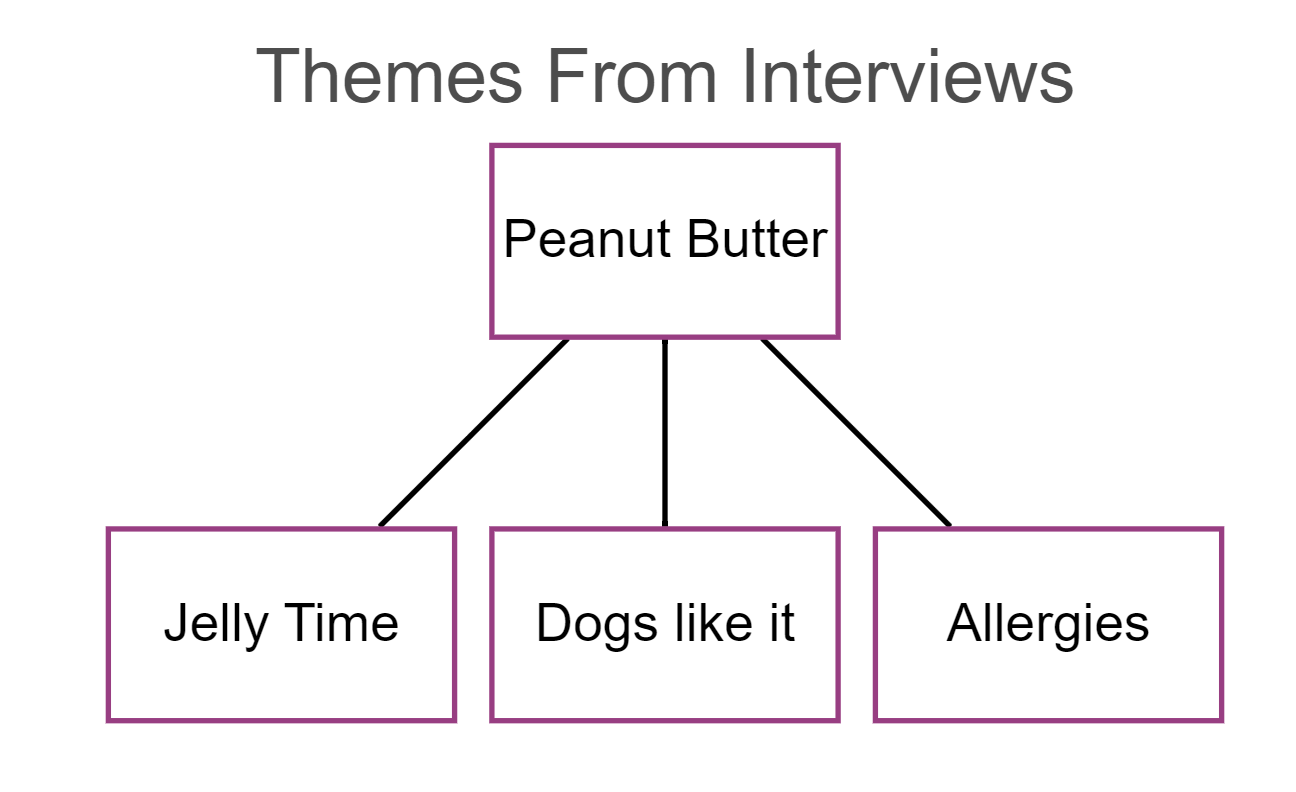
ggplot(aes(label = word, color = word, size = n)) +

geom\_text\_wordcloud\_area() +

scale\_size\_area(max\_size = 20) +

theme\_minimal() + ggtitle("Glasgow Word Cloud")

**Word Trees**

****

x <- scan("C:/Users/HP/Downloads/peanut.txt", what="", sep="\n", quiet = TRUE)

from=c(x[1], x[1], x[1])

to=c(x[2],x[3],x[4])

nodesn=c(x[1],x[2],x[3],x[4])

nodes <- create\_node\_df(n=length(nodesn), label=nodesn, width=0.9, shape = "rectangle", color = "#983E82", fillcolor = "white")

edges <- create\_edge\_df(from = factor(from, levels=nodesn), to = factor(to, levels=nodesn), arrowhead = "arrow", color = "black")

graph <- create\_graph(nodes\_df = nodes, edges\_df = edges, directed = FALSE)

graph %>%

render\_graph(title = "Themes From Interviews", layout = "tree")