Assignment 3 - Visualization

Abrar Faruque

October 16, 2025

Assignment 3 - Visualization

The dataset can be accessed here: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

The phenotype attributions are as follows:

1) ID number
2) Diagnosis (M = malignant, B = benign) 3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

Here is some code to read in the dataset

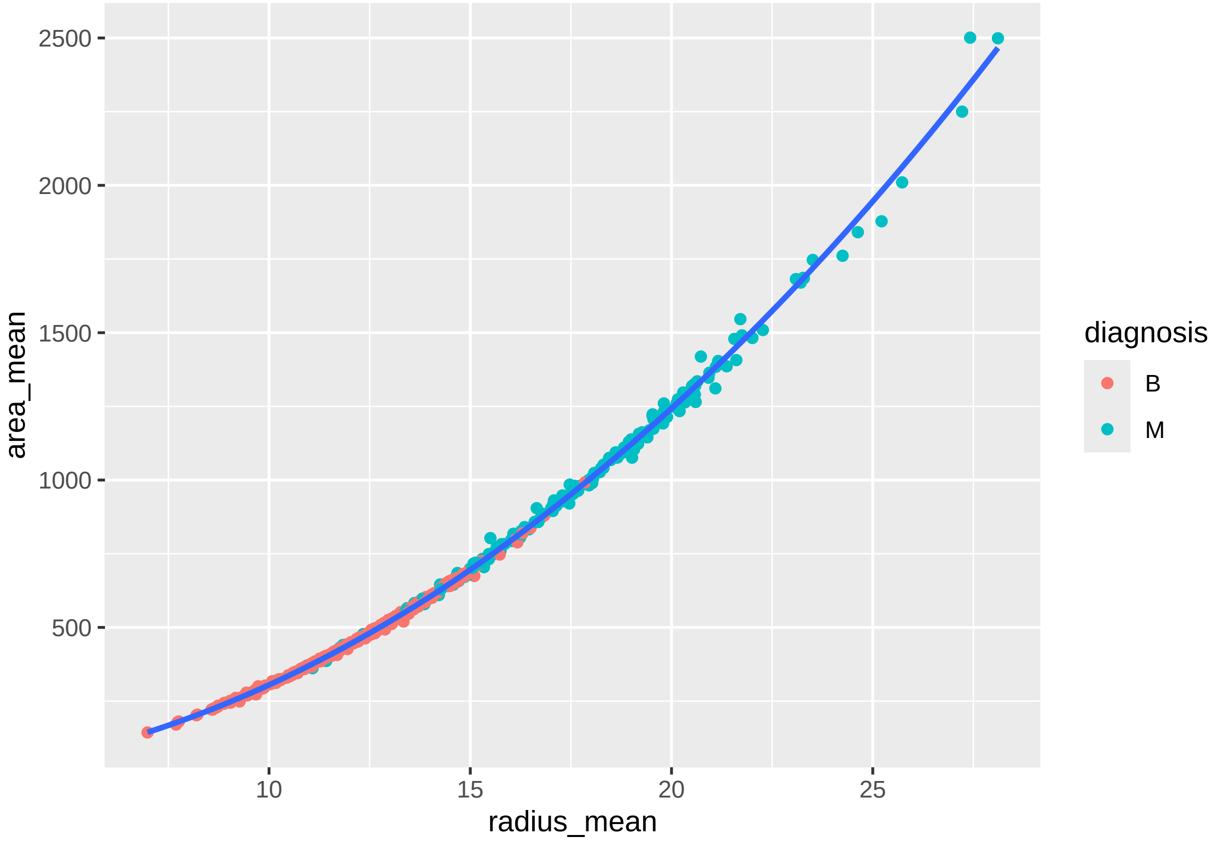
```
bCancer <- read.csv("breastCancer.csv", header=T)
```

Hint: draw out the plot that the question is asking you for **by hand** before plotting using ggplot2. This will help you create and add to your plots.

1. The aesthetic of the plot can be defined in both the main `ggplot()` function or the plot specific geometry function. What advantage does this redundancy allow for? Give an example of this use case. [0.5]

```
#1.
library(ggplot2)
# ggplot redundancy allows for variables to be called either globally or locally for instance:
ggplot(bCancer, aes(x=radius_mean, y = area_mean))+
  geom_point(aes(color = diagnosis))+             #local aesthetic override
  geom_smooth(se=FALSE)                           #inherits x and y but not color
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula = ’y ~ x’
```



```
#geom_smooth inherits x and y but not the 'color' variable that was defined inside
#(locally) in geom_point, this allows the smoothing line not to be colored
#This keeps code clean, organized and flexible
```

2. What is the difference between a bar plot and a histogram? Which is more appropriate when I am plotting the heights of all students in this course? [0.5]

```
#The difference between Bar plot and histogram is the following:
#   Bar plot -> used to display categorical values often used to compare two or more #   static variabl
#   Histogram ->used to display distribution of a single continuous variable
#'If you are trying to plot the distribution of heights (continuous variable) of all
#'students in the course in a categorical fashion, you would use a histogram to show
#'distribution continuously in terms of hight versus density
```

3. Use the `bCancer` dataset for this entire question.

a) When comparing the mean radius between diagnosis groups, what plot type would be most appropriate? Justify your response by relating back to the type of data we are portraying. Then, create the plot. Make sure to format the plot properly such as labeling the titles and axis. [1]

b) When visualizing if there is a relationship between the mean smoothness and mean compactness of the tumors, what plot type would be most appropriate? Justify your response by relating back to the type of data we are portraying. Then, create the plot. Make sure to format the plot properly such as labeling the titles and axis. [1]

c) Plot both the `radius_mean` and the `texture_mean` of tumours in two panels (Hint: you will need to facet), both versus diagnosis in a violin plot. Then, colour by diagnosis. Hint # 2: you will need to wrangle the data first before plotting. What is the format of the data you will need? [1.5]

d) Finally, we want to see the spread of our data, thus we want to plot all of our data in a histogram. Plot all of the mean measurements (radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, and symmetry). Plot the histograph of all of these measurements in a 9-panel figure. Colour again by diagnosis. Hint: you may need to use the `scales` argument to free the x-axis (`free_x`). [1.5]

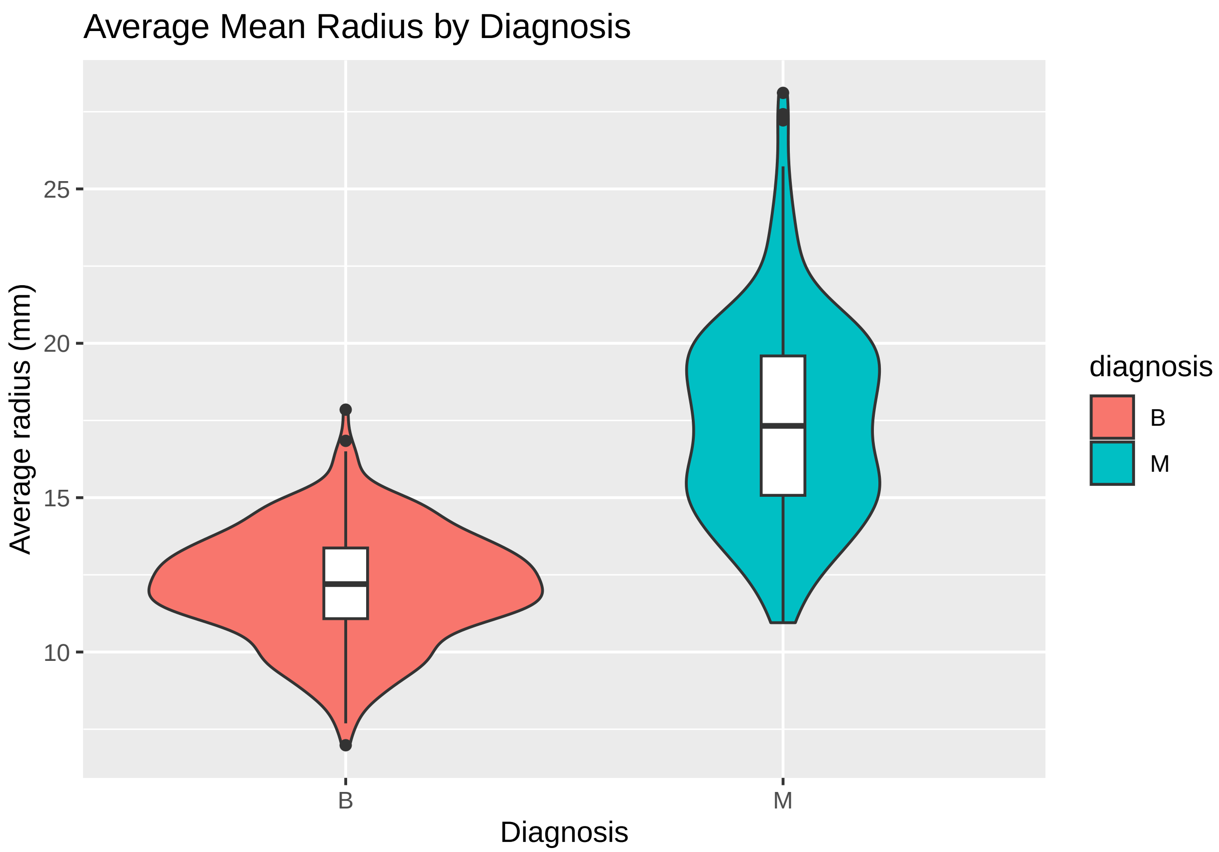
```r
#'a) I would use a box plot (with additional violin plot to show density
#'distribution, they both compliment the shortcomings of each other), the data
#'is COMPARING the mean radius BETWEEN two CATAGORICAL
#'variables:'M' and 'B' while also showing the distribution and the standard
#'deviation between the two variables
library(ggplot2) # plotting
library(dplyr) # manipulation
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyverse)
```

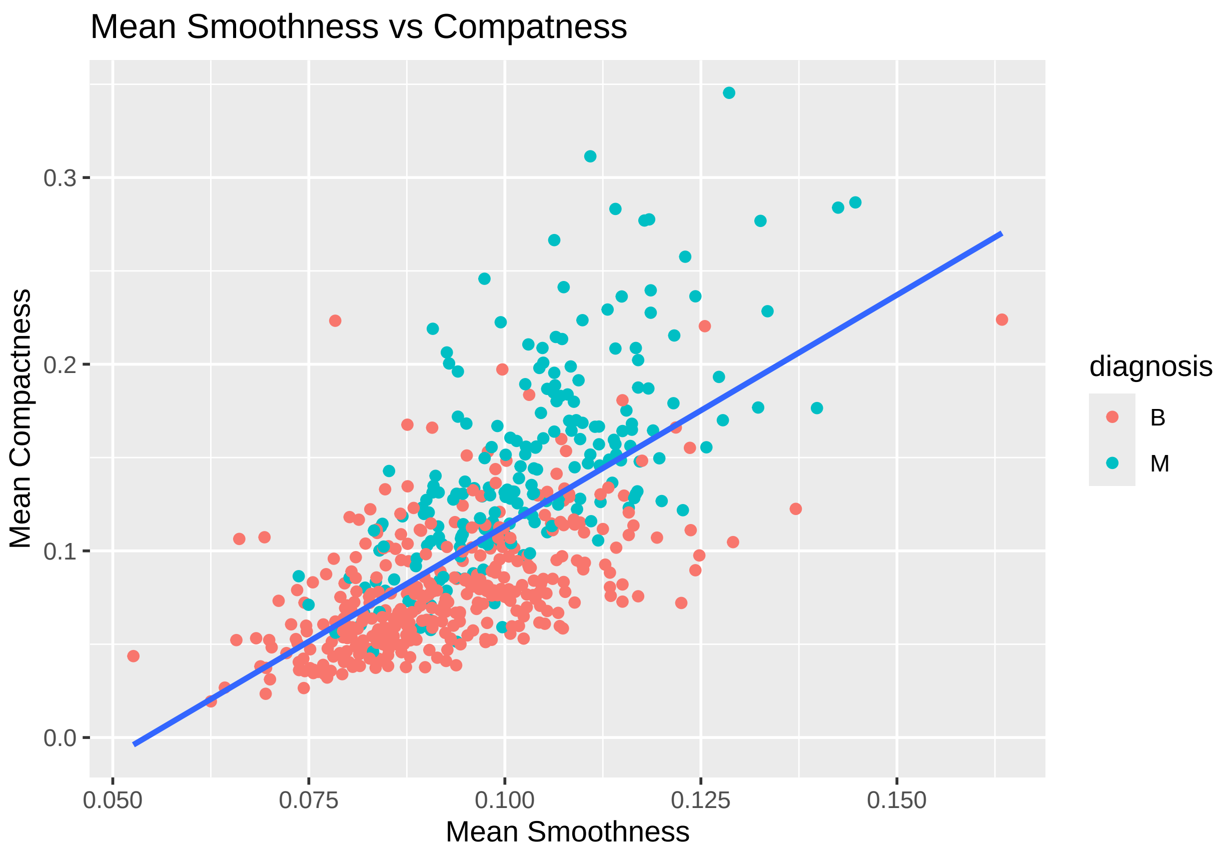
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.2
## v lubridate 1.9.4     v tibble    3.3.0
## v purrr     1.1.0     v tidyr     1.3.1
## v readr     2.1.5

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
ggplot(bCancer, aes(x=diagnosis,y=radius_mean))+
  geom_violin(aes(fill=diagnosis))+
  geom_boxplot(width=0.1)+
  labs(title="Average Mean Radius by Diagnosis",
       x="Diagnosis", y="Average radius (mm)")
```

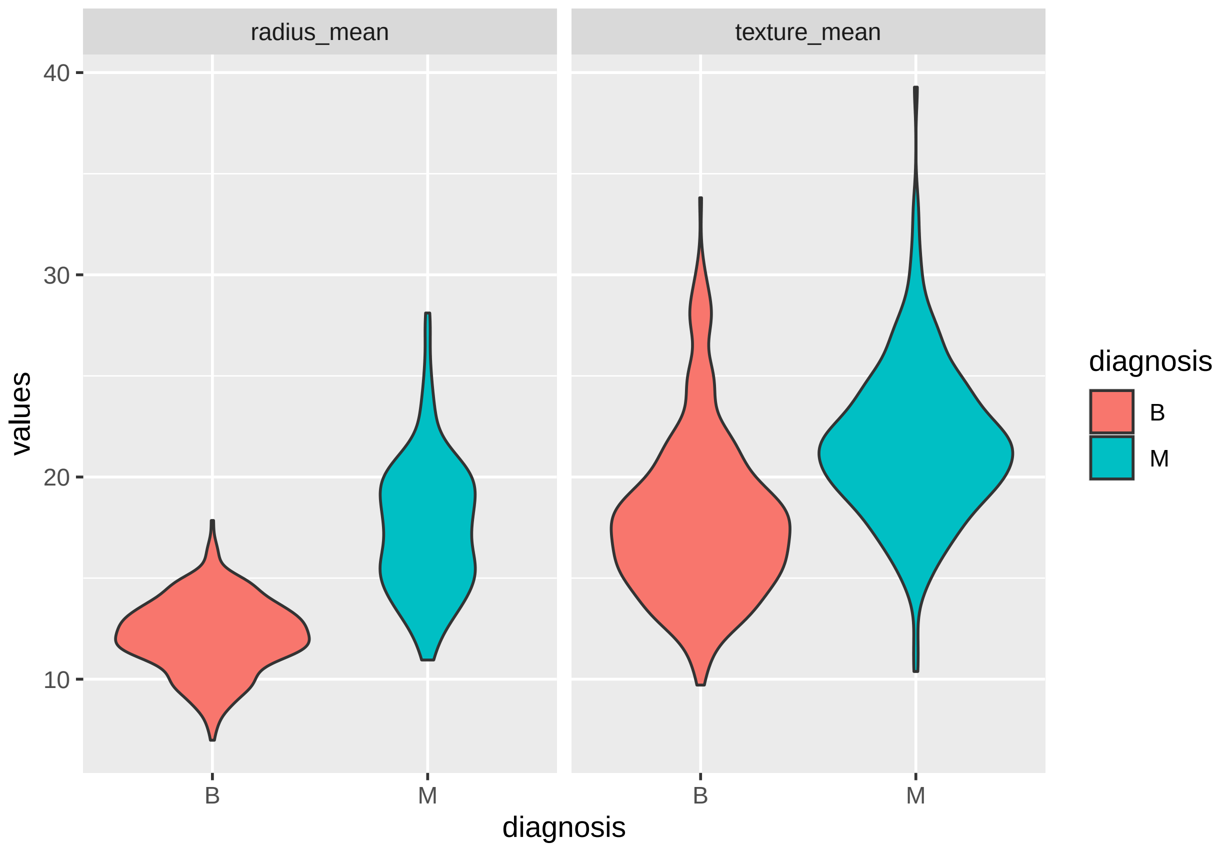


```
#'b)I would use a scater plot to compare 2 continuous variables, I can then use
#'smooth to determine if there is a correlation between the two variables

ggplot(bCancer,aes(x=smoothness_mean,y=compactness_mean))+
  geom_point(aes(color=diagnosis))+
  geom_smooth(method="lm", se =FALSE)+
  labs(title="Mean Smoothness vs Compatness",
       x="Mean Smoothness",
       y="Mean Compactness",)
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



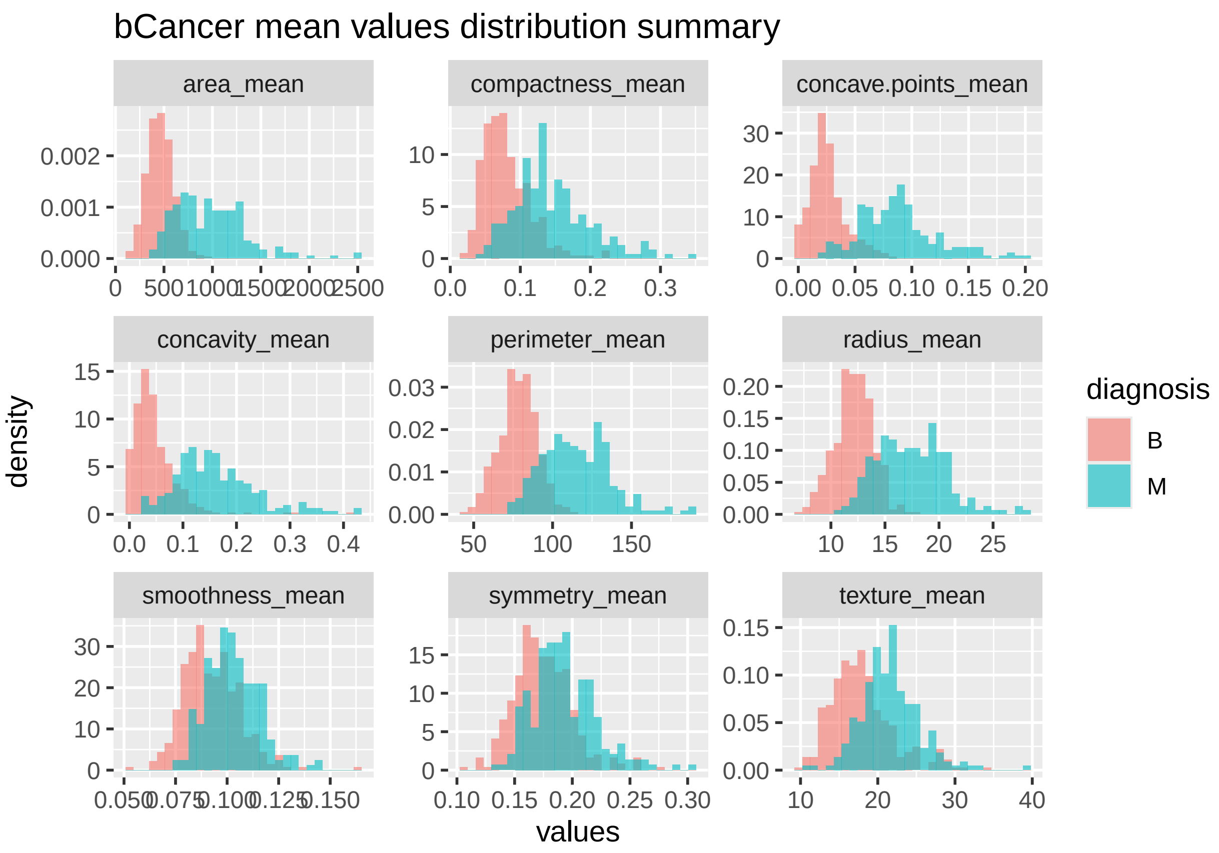
```
#based on the data, the correlation is significant (non-random), weak and positive between the continuo

#'C) The form of data needed for this plot is a dataframe, with variables
#''diagnosis, varibles(consisting of 'radius_mean' and 'texture_mean') and their
#'corresponding values
bCancerLong<- bCancer%>%
  select(diagnosis,radius_mean,texture_mean) %>%
  pivot_longer(cols = "radius_mean":"texture_mean",
               names_to = "variables",
               values_to = "values")

ggplot(bCancerLong, aes(x=diagnosis,y=values))+
  geom_violin(aes(fill=diagnosis))+
  facet_wrap(~variables)
```



```
#'d)
bCancerLongest<-bCancer %>%
  select(diagnosis:symmetry_mean) %>%
  pivot_longer(cols = "radius_mean":"symmetry_mean",
               names_to = "variables",
               values_to = "values")
ggplot(bCancerLongest, aes(x=values,y=..density..))+
  geom_histogram(aes(fill=diagnosis),alpha=0.6, position = 'identity')+
  labs(title='bCancer mean values distribution summary')+
  facet_wrap(~variables,scales='free')
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## `stat_bin()` using `bins = 30`. Pick better value `binwidth`.
```


bCancer mean values distribution summary
area_mean
compactness_mean
concave.points_mean
concavity_mean
perimeter_mean
radius_mean
smoothness_mean
symmetry_mean
texture_mean
density
values
diagnosis
B
M
0.000
0.001
0.002
0
500
1000
1500
2000
2500
0
5
10
0.0
0.1
0.2
0.3
0
10
20
30
0.00
0.05
0.10
0.15
0.20
0
5
10
15
0.0
0.1
0.2
0.3
0.4
0.00
0.01
0.02
0.03
50
100
150
0.00
0.05
0.10
0.15
0.20
10
15
20
25
0
10
20
30
0.050
0.075
0.100
0.125
0.150
0
5
10
15
0.10
0.15
0.20
0.25
0.30
0.00
0.05
0.10
0.15
10
20
30
40