

Class Activity 4

Your name here

March 19 2024

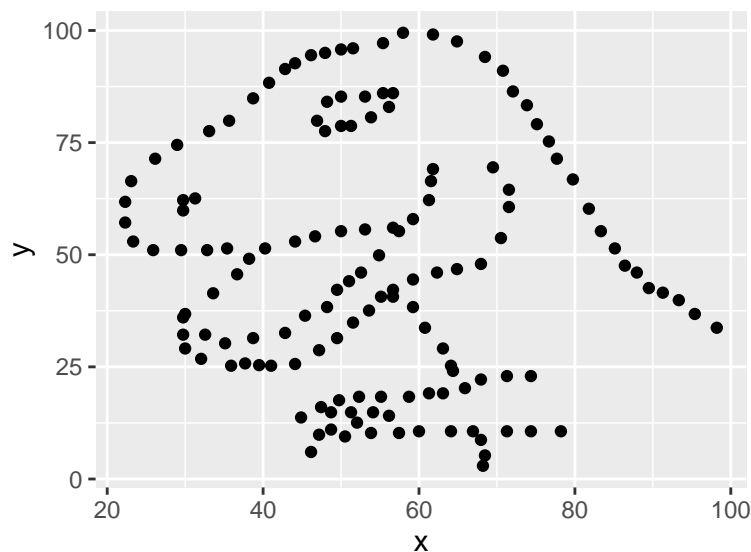
Your turn 1

This worksheet will guide you through creating various plots using the `ggplot2` package in R. We will be using the `datasaurus_dozen` dataset from the `datasauRus` package for demonstration purposes. The dataset contains 13 different datasets, and we'll use them to create a variety of plots.

Scatterplot

a. Run the following code.

```
ggplot(data = dino_data, mapping = aes(x = x, y = y)) +  
  geom_point()
```



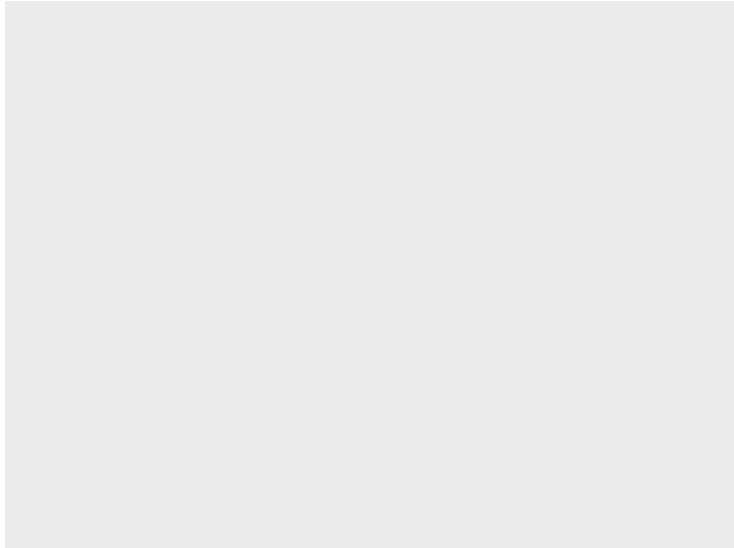
b. You *must* remember to put the aesthetic mappings in the `aes()` function! What happens if you forget?

Answer:

If you forget to put the aesthetic mappings inside the `aes()` function, `ggplot2` will not be able to map the variables to the aesthetics correctly, and you might encounter an error or unexpected behavior in your plot.

Add a layer and see what happens

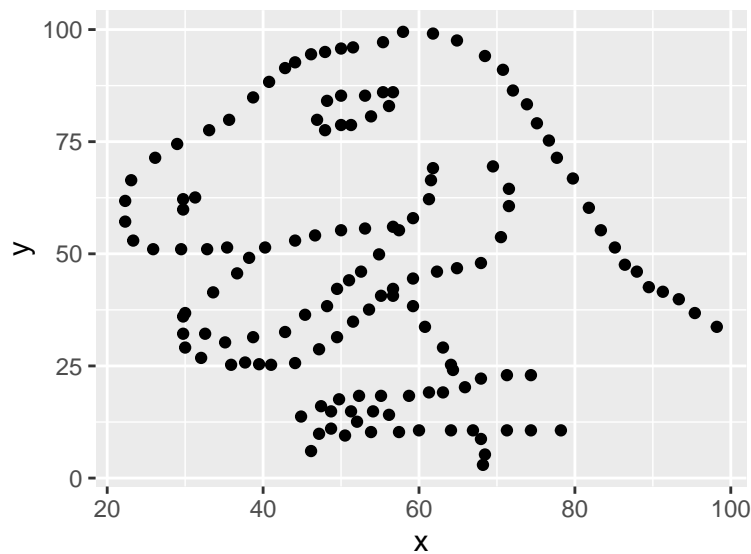
```
ggplot(data = dino_data , x = x , y = y)
```



- c. The aesthetic mappings can be specified in the geom layer if you prefer, instead of the main `ggplot()` call. Give it a try:

Answer:

```
# Rebuild the scatterplot with your aesthetic mapping in the geom layer
ggplot(data = dino_data) +
  geom_point(aes(x = x, y = y))
```



Bar Plot

In this problem, we'll explore creating a bar plot using the `datasaurus_dozen` dataset.

- a. Create a new data frame containing the count of observations in each dataset.

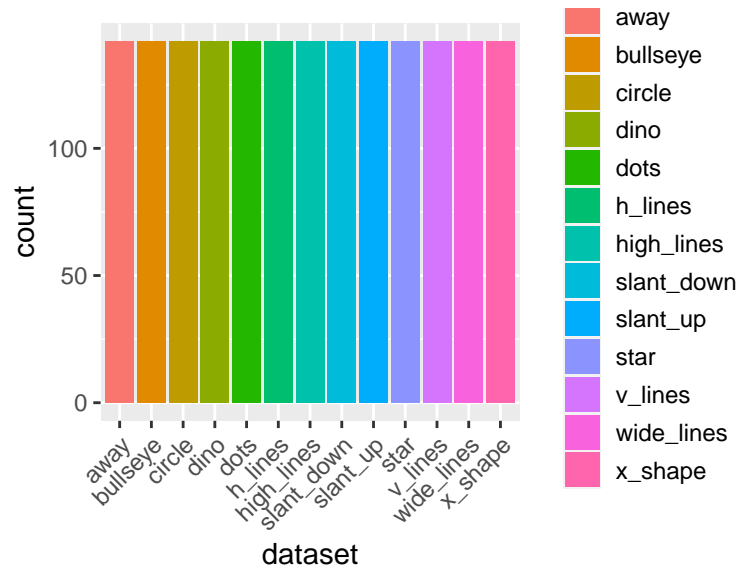
Answer:

```
dataset_counts <- datasaurus_dozen %>%
  group_by(dataset) %>%
  summarise(count = n())
```

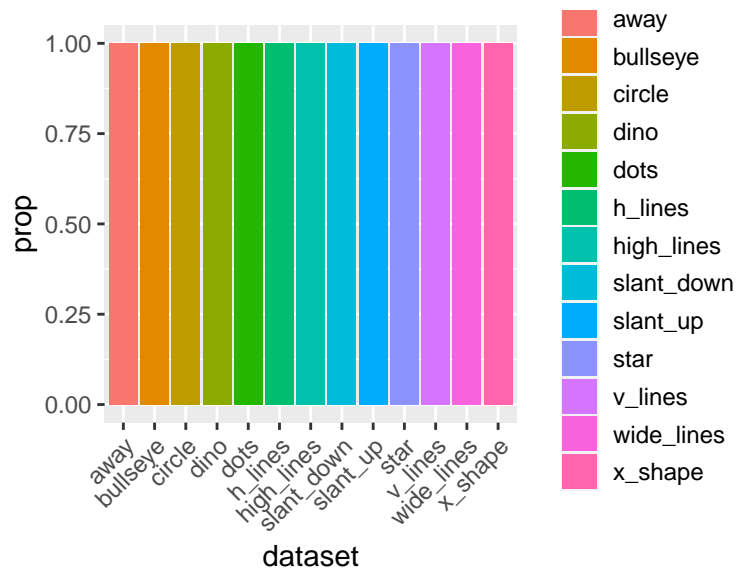
b. Create a bar plot showing the number of observations in each dataset.

Answer:

```
ggplot(data = dataset_counts, aes(x = dataset, y = count, fill = dataset)) +  
  geom_bar(stat = "identity") +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1),  
        legend.key.size = unit(0.5, "cm"))
```



```
ggplot(data = dataset_counts, aes(x = dataset, fill = dataset)) +  
  geom_bar(aes(y = after_stat(prop))) +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1),  
        legend.key.size = unit(0.5, "cm"))
```

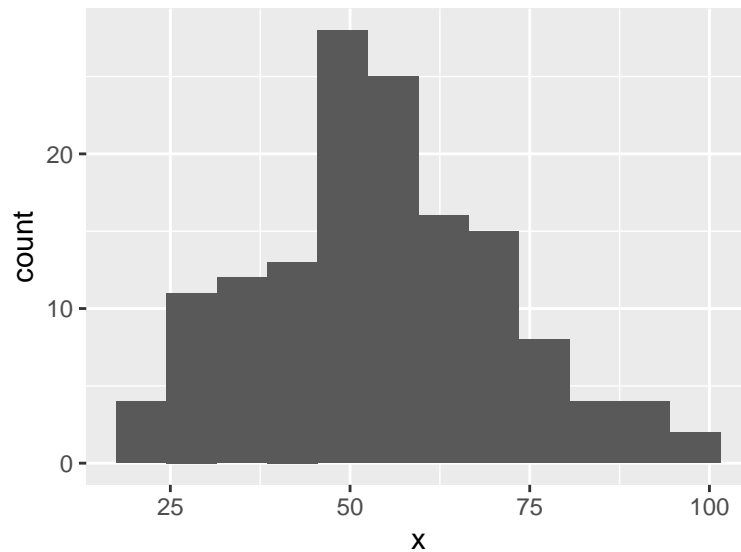


Histogram

a. Create a histogram of the x variable for the dino dataset.

Answer:

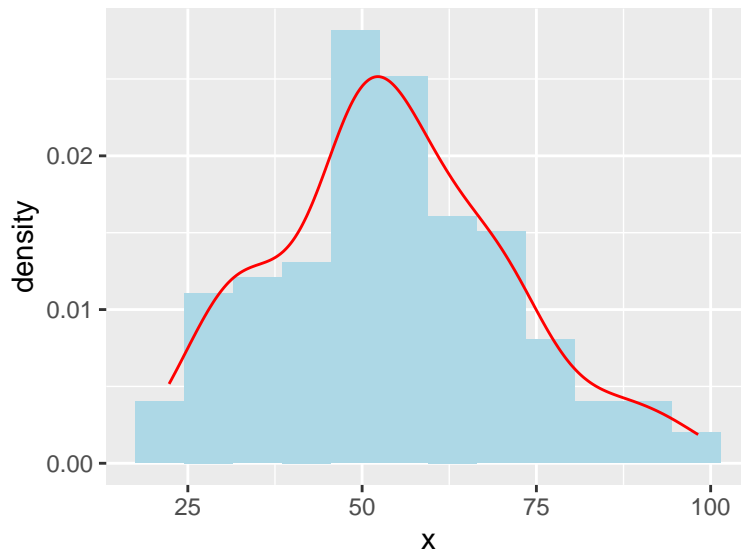
```
ggplot(data = dino_data, aes(x = x)) +  
  geom_histogram(binwidth = 7)
```



b. Overlay a density curve on the histogram.

Answer:

```
ggplot(data = dino_data, aes(x = x)) +  
  geom_histogram(aes(y = after_stat(density)), binwidth = 7, fill = "lightblue") +  
  geom_density(color = "red")
```

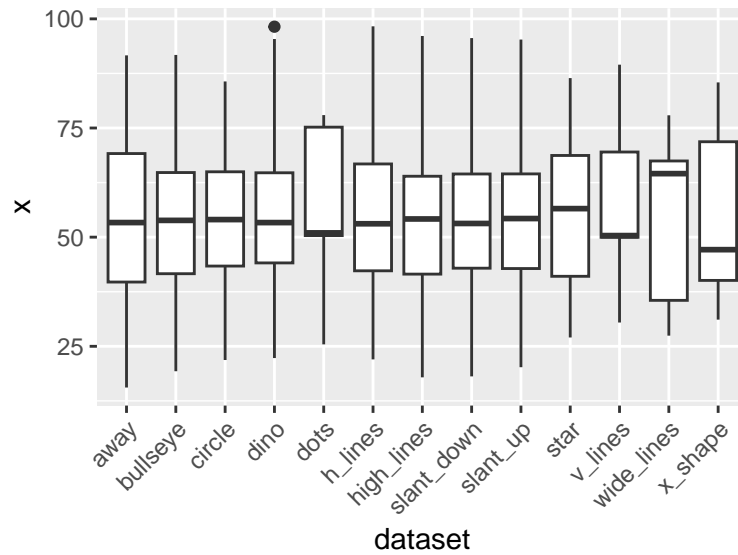


Boxplot

Answer:

a. Create a boxplot of the x variable for each dataset in datasaurus_dozen.

```
ggplot(data = datasaurus_dozen, aes(x = dataset, y = x)) +  
  geom_boxplot() +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

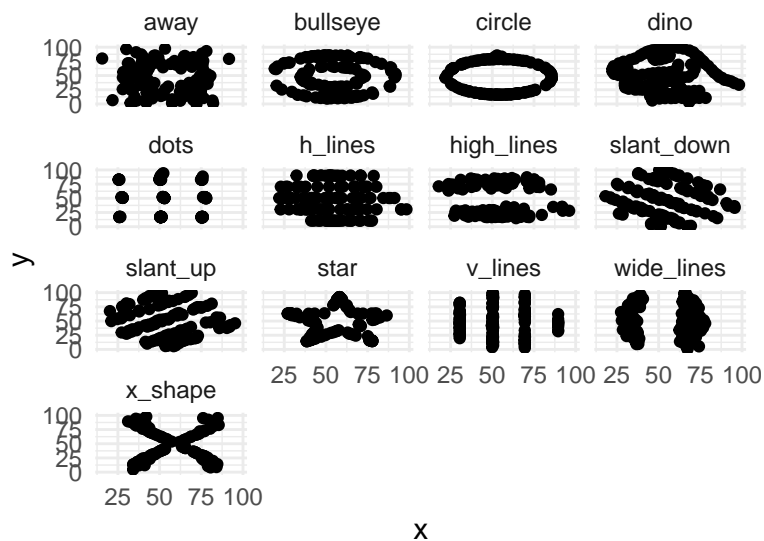


Faceting

Answer:

- a. Create a scatterplot of `x` vs. `y` for each dataset in `datasaurus_dozen` using `facet_wrap()`.

```
ggplot(data = datasaurus_dozen, aes(x = x, y = y)) +  
  geom_point() +  
  facet_wrap(~ dataset) +  
  theme_minimal()
```



Variable Transformation

- a. The scatterplot of the `dino` dataset without any transformations is given below.

Answer:

```
ggplot(data = dino_data, aes(x = x, y = y)) +  
  geom_point() +  
  theme_minimal() -> p1
```

- b. Now, apply the square root transformation to both the `x` and `y` axes using the `scale_x_sqrt()` and `scale_y_sqrt()` functions in the `dino` dataset.

Answer:

```
ggplot(data = dino_data, aes(x = x, y = y)) +  
  geom_point() +  
  scale_x_sqrt() +  
  scale_y_sqrt() +  
  theme_minimal() -> p2
```

- c. Finally, use `grid.arrange()` function from `gridExtra` package to plot the above two plots side-by-side. Which plot do you prefer and why?

Answer: The second plot is more revealing of a dinosaur than the first plot.

```
library(gridExtra)  
grid.arrange(p1, p2, nrow = 1)
```

