

# STA130 Week 2 Class Supplementary Materials

Nathalie Moon

## Supplementary materials and R Code

### Class demo

#### Demo: Loading pokemon data

```
library(pokemon) # R package pokemon, which contains a tibble called pokemon
pokemon |>
  select(pokemon, height, weight, base_experience,
         type_1, attack, defense, speed, generation_id) |>
  glimpse()
```

Rows: 949

Columns: 9

```
$ pokemon      <chr> "bulbasaur", "ivysaur", "venusaur", "charmander", "cha~
$ height       <dbl> 0.7, 1.0, 2.0, 0.6, 1.1, 1.7, 0.5, 1.0, 1.6, 0.3, 0.7,~
$ weight       <dbl> 6.9, 13.0, 100.0, 8.5, 19.0, 90.5, 9.0, 22.5, 85.5, 2.~
$ base_experience <dbl> 64, 142, 236, 62, 142, 240, 63, 142, 239, 39, 72, 178,~
$ type_1       <chr> "grass", "grass", "grass", "fire", "fire", "fire", "wa~
$ attack       <dbl> 49, 62, 82, 52, 64, 84, 48, 63, 83, 30, 20, 45, 35, 25~
$ defense      <dbl> 49, 63, 83, 43, 58, 78, 65, 80, 100, 35, 55, 50, 30, 5~
$ speed        <dbl> 45, 60, 80, 65, 80, 100, 43, 58, 78, 45, 30, 70, 50, 3~
$ generation_id <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
```

Does the `pokemon` tibble contain all pokemons?

Where does the data come from? Is it reliable?

## Demo: Creating histograms

```
library(pokemon)

### Histogram: Choose a numerical variable to visualize here ###
# Most basic histogram (just with default values)

# Let's change the number of bins

# Another way of changing the number of bins: specifying binwidth

# Let's change the axis labels using `+ labs(x = "New label")`

# How can we change the colours of the bars (fill and outline)
```

## Demo: Creating Boxplots

```
### Boxplot ###
# Most basic boxplot (just with default values)

# Let's look at the association between our numerical variable and a categorical
# variable using boxplots
```

## Demo: Creating Barplots

```
### Barplot ###
# Most basic barplot (just with default values)

# Do we want to make the bars horizontal instead?

# Let's change the axis titles to be more descriptive

# Can we change the colours (with fill and colour)?
# Lots of colour choices in R: https://r-graph-gallery.com/42-colors-names.html
```

## Demo: Creating Scatterplots

```
### Scatterplot ###  
# Most basic scatterplot (just with default values)  
  
# Let's improve the axis titles  
  
# Are there lots of overlapping points? We can change the transparency using alpha  
  
# Can we colour the points by the value of a categorical variable?  
  
# Can we split this into separate scatterplots for each value of the categorical  
# variable?
```

## Code from class slides

### Histograms

```
# Loading pokemon
library(pokemon) # R package pokemon, which contains a tibble called pokemon
glimpse(pokemon)
```

```
Rows: 949
Columns: 22
$ id          <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, ~
$ pokemon     <chr> "bulbasaur", "ivysaur", "venusaur", "charmander", "cha~
$ species_id  <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, ~
$ height      <dbl> 0.7, 1.0, 2.0, 0.6, 1.1, 1.7, 0.5, 1.0, 1.6, 0.3, 0.7, ~
$ weight      <dbl> 6.9, 13.0, 100.0, 8.5, 19.0, 90.5, 9.0, 22.5, 85.5, 2.~
$ base_experience <dbl> 64, 142, 236, 62, 142, 240, 63, 142, 239, 39, 72, 178, ~
$ type_1      <chr> "grass", "grass", "grass", "fire", "fire", "fire", "wa~
$ type_2      <chr> "poison", "poison", "poison", NA, NA, "flying", NA, NA~
$ hp          <dbl> 45, 60, 80, 39, 58, 78, 44, 59, 79, 45, 50, 60, 40, 45~
$ attack      <dbl> 49, 62, 82, 52, 64, 84, 48, 63, 83, 30, 20, 45, 35, 25~
$ defense     <dbl> 49, 63, 83, 43, 58, 78, 65, 80, 100, 35, 55, 50, 30, 5~
$ special_attack <dbl> 65, 80, 100, 60, 80, 109, 50, 65, 85, 20, 25, 90, 20, ~
$ special_defense <dbl> 65, 80, 100, 50, 65, 85, 64, 80, 105, 20, 25, 80, 20, ~
$ speed       <dbl> 45, 60, 80, 65, 80, 100, 43, 58, 78, 45, 30, 70, 50, 3~
$ color_1     <chr> "#78C850", "#78C850", "#78C850", "#F08030", "#F08030", ~
$ color_2     <chr> "#A040A0", "#A040A0", "#A040A0", NA, NA, "#A890F0", NA~
$ color_f     <chr> "#81A763", "#81A763", "#81A763", NA, NA, "#DE835E", NA~
$ egg_group_1 <chr> "monster", "monster", "monster", "monster", "monster", ~
$ egg_group_2 <chr> "plant", "plant", "plant", "dragon", "dragon", "dragon~
$ url_icon    <chr> "//archives.bulbagarden.net/media/upload/7/7b/001MS6.p~
$ generation_id <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
$ url_image   <chr> "https://raw.githubusercontent.com/HybridShivam/Pokemo~
```

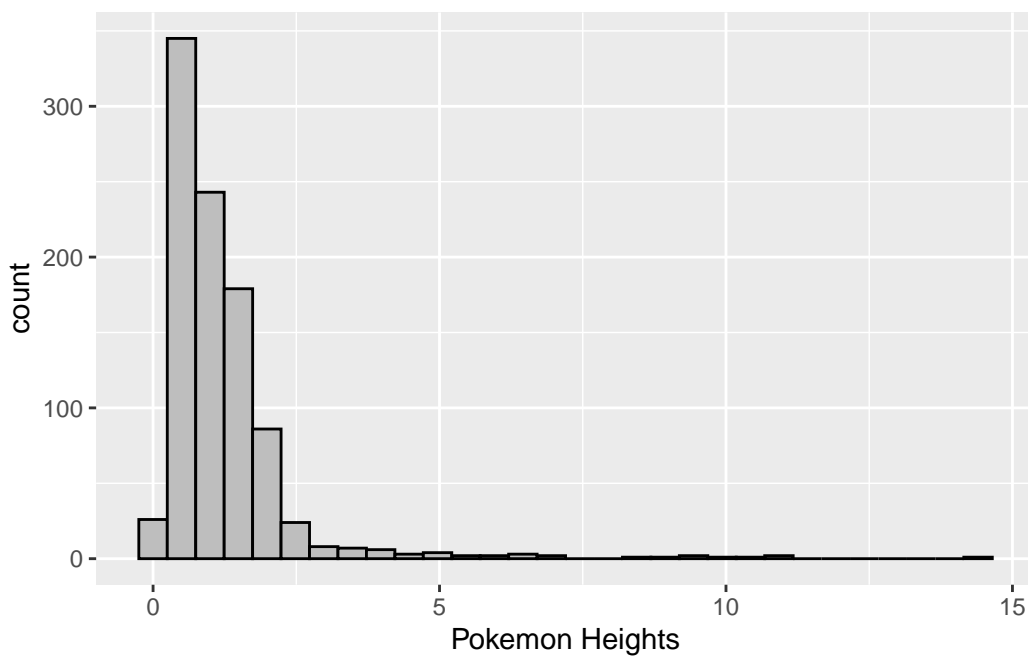
```
##### Code to create new variable for experience stage #####
### We'll learn about mutate() in week 3, and may touch on kmeans clustering later
### in the course (not required knowledge for this week!)
pokemon <- pokemon |>
  mutate(
    cluster = kmeans(base_experience, centers = 3)$cluster,
    avg_exp_by_cluster = ave(base_experience, cluster, FUN = mean),
    evolution_stage = case_when(
```

```

    avg_exp_by_cluster == min(avg_exp_by_cluster) ~ "1st stage",
    avg_exp_by_cluster == max(avg_exp_by_cluster) ~ "3rd/final stage",
    TRUE ~ "2nd stage"
  )
) |>
select(-cluster, -avg_exp_by_cluster) # Clean up temporary columns
####

# Creating a histogram in R
pokemon |> ggplot(aes(x = height)) +
  geom_histogram(color = "black",
                fill = "gray",
                bins = 30) +
  labs(x = "Pokemon Heights")

```



```

# Another example - adjusting number of bins
hist1 <- pokemon |> ggplot(aes(x = base_experience)) +
  geom_histogram(color = "black",
                fill = "gray",

```

```

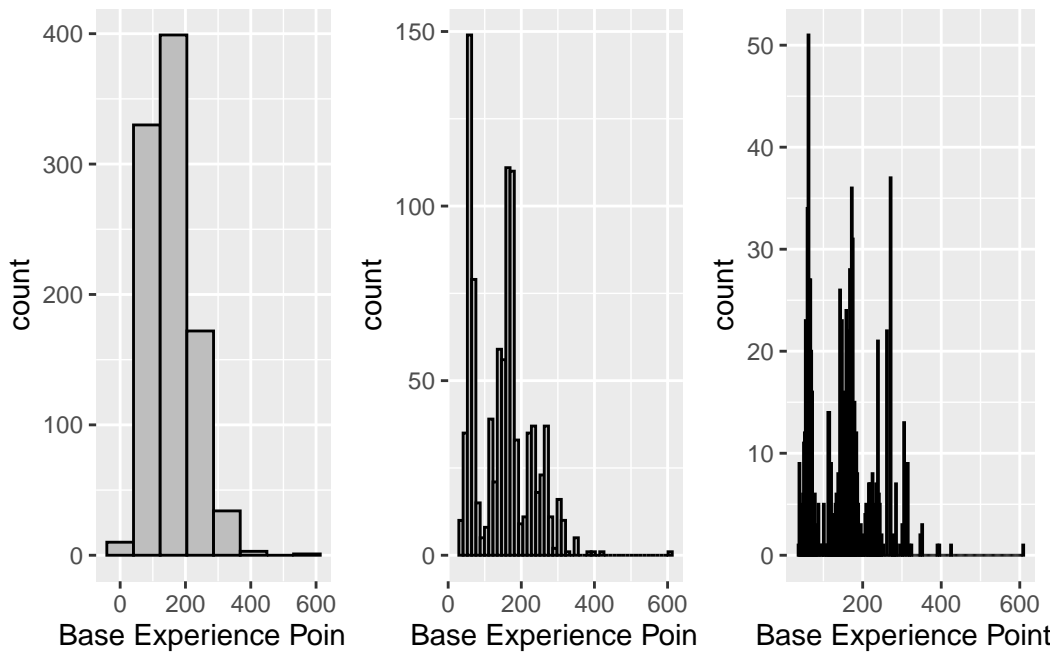
      bins = 8) +
  labs(x = "Base Experience Points")

hist2 <- pokemon |> ggplot(aes(x = base_experience)) +
  geom_histogram(color = "black",
    fill = "gray",
    bins = 50) +
  labs(x = "Base Experience Points")

hist3 <- pokemon |> ggplot(aes(x = base_experience)) +
  geom_histogram(color = "black",
    fill = "gray",
    bins = 250) +
  labs(x = "Base Experience Points")

cowplot::plot_grid(hist1, hist2, hist3, ncol=3)

```



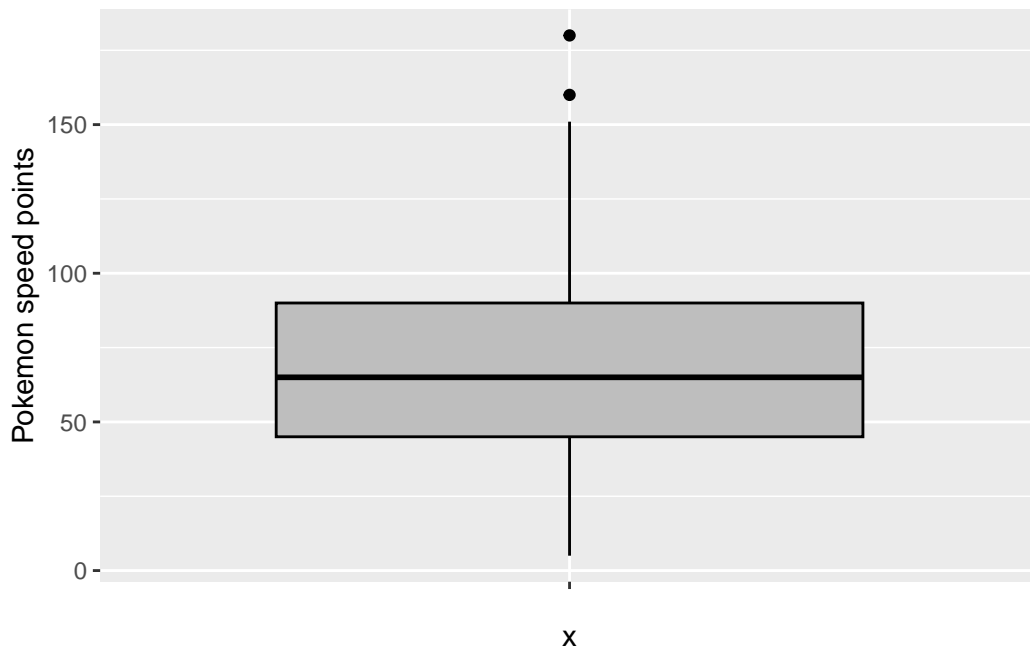
## Boxplots

```

# Boxplot
pokemon |> ggplot(aes(x = "", y = speed)) +

```

```
geom_boxplot(color="black", fill="gray") +
labs(y = "Pokemon speed points")
```



```
# Compare boxplot to histogram
pokemon_boxplot <- pokemon |>
  ggplot(aes(x = "", y = speed)) +
  geom_boxplot(color="black", fill="gray") +
  ylim(0, 185) +
  labs(y = "Pokemon speed points",
       x = "") +
  coord_flip()

pokemon_hist <- pokemon |>
  ggplot(aes(x = speed)) +
  geom_histogram(binwidth=10, color="black", fill="gray") +
  labs(x = "Pokemon speed points")

# The function below, from the cowplot package, lets you combine plots into a single R object
library(cowplot)
```

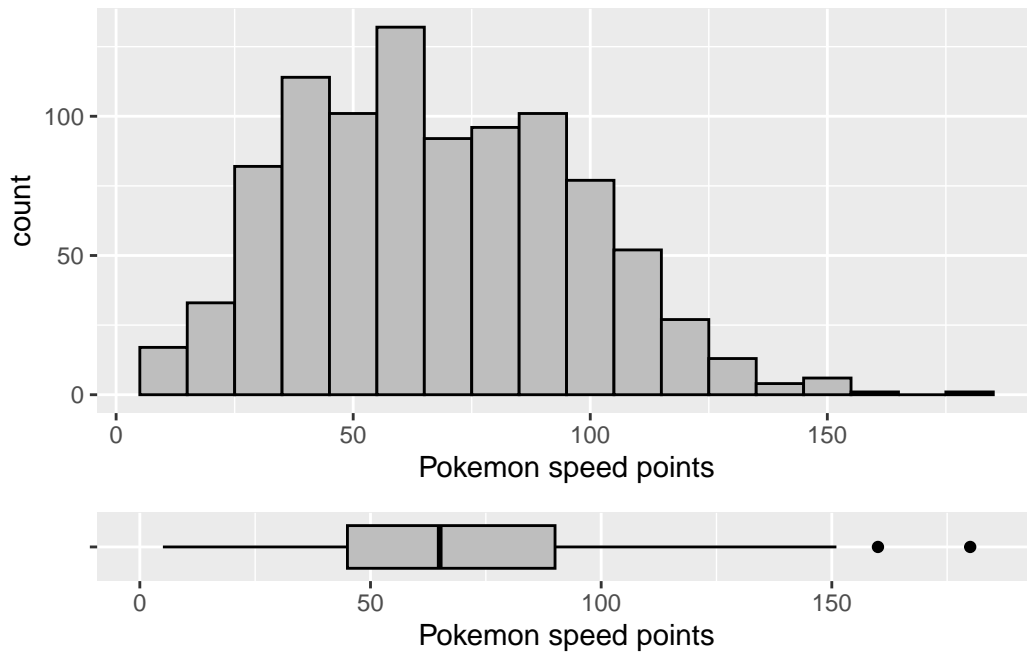
Warning: package 'cowplot' was built under R version 4.4.3

Attaching package: 'cowplot'

The following object is masked from 'package:lubridate':

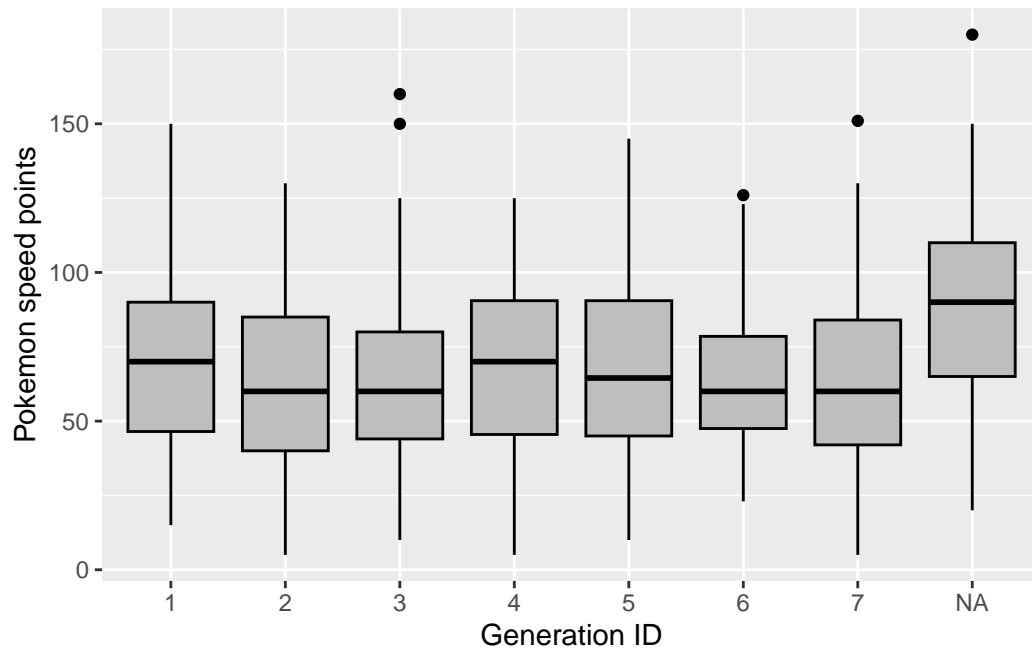
stamp

```
plot_grid(pokemon_hist, pokemon_boxplot, align = "v", ncol = 1, rel_heights = c(3, 1))
```

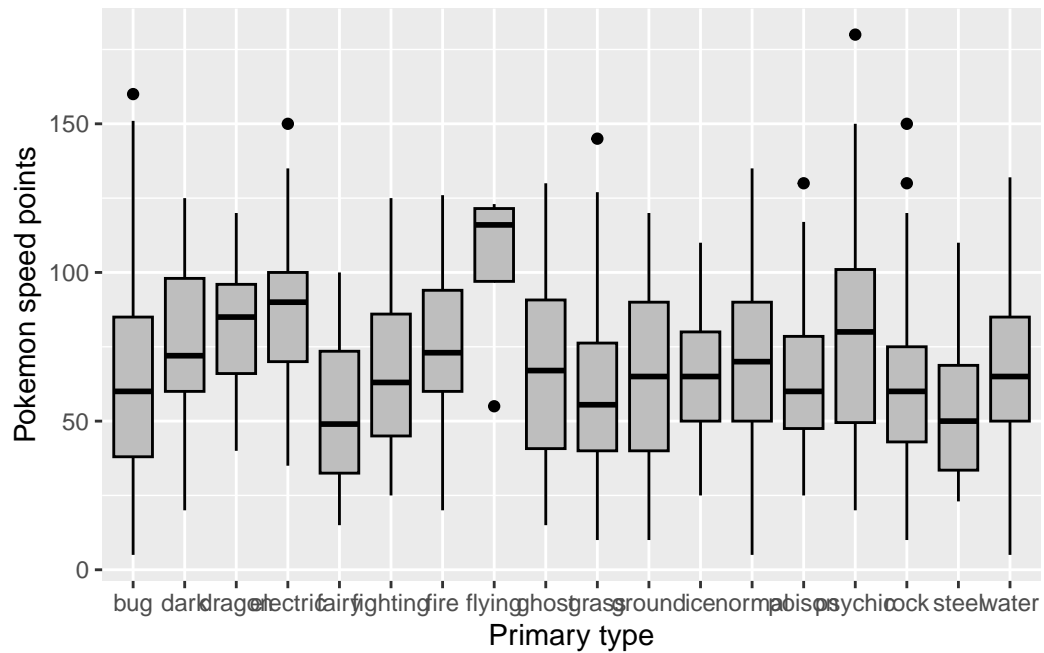


```
# Side-by-side boxplots
pokemon |> ggplot(aes(x = factor(generation_id), y=speed)) +
  geom_boxplot(color="black", fill="gray") +
  labs(y = "Pokemon speed points",
       x = "Generation ID")
```

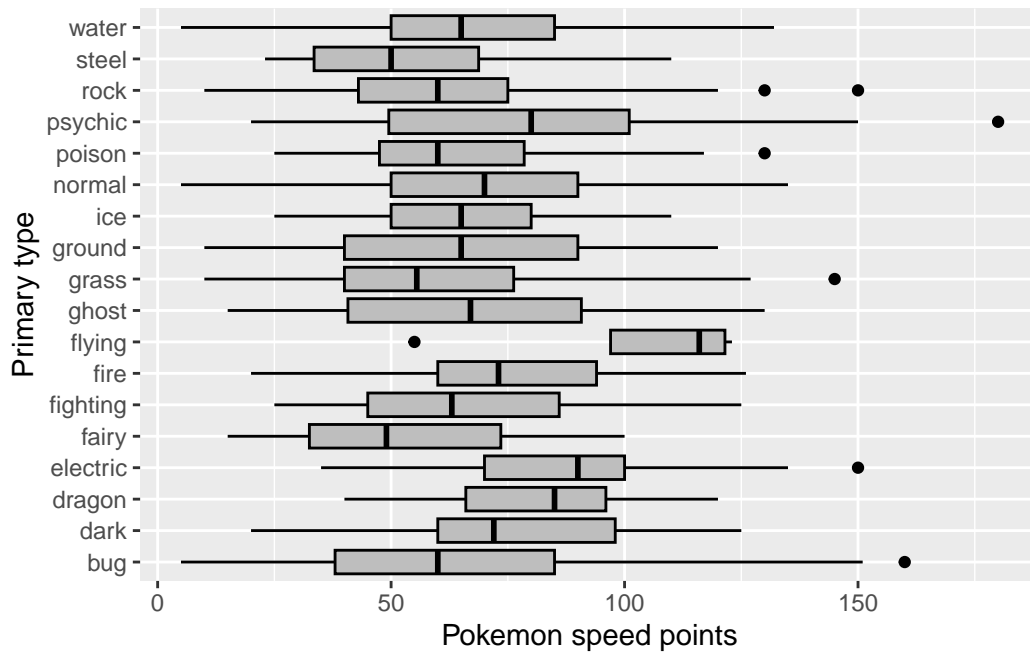




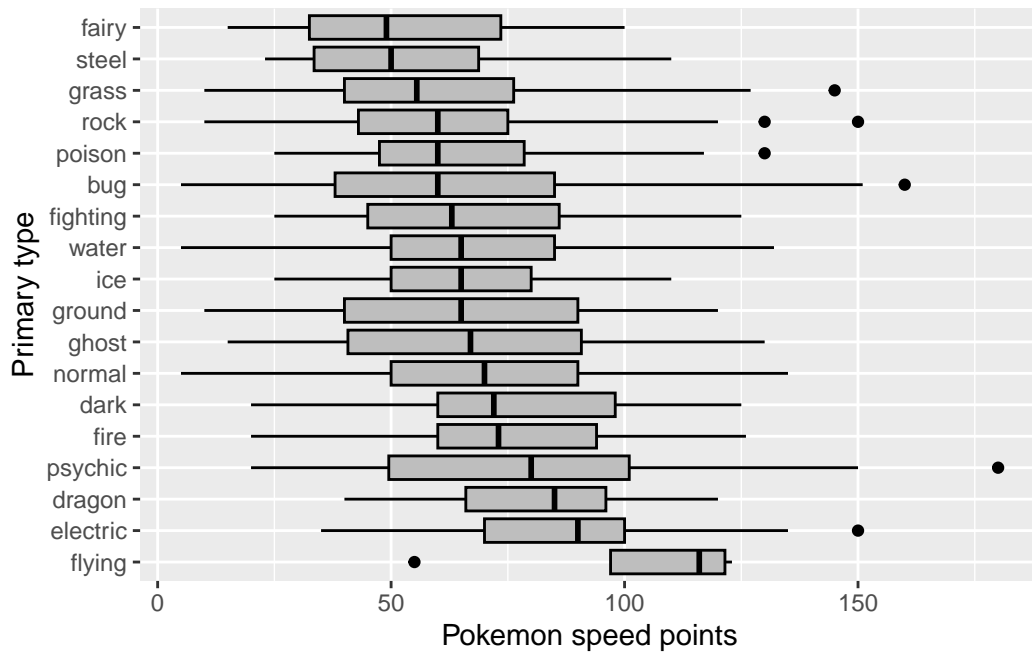
```
pokemon |> ggplot(aes(x = type_1, y=speed)) +  
  geom_boxplot(color="black", fill="gray") +  
  labs(y = "Pokemon speed points",  
       x = "Primary type")
```



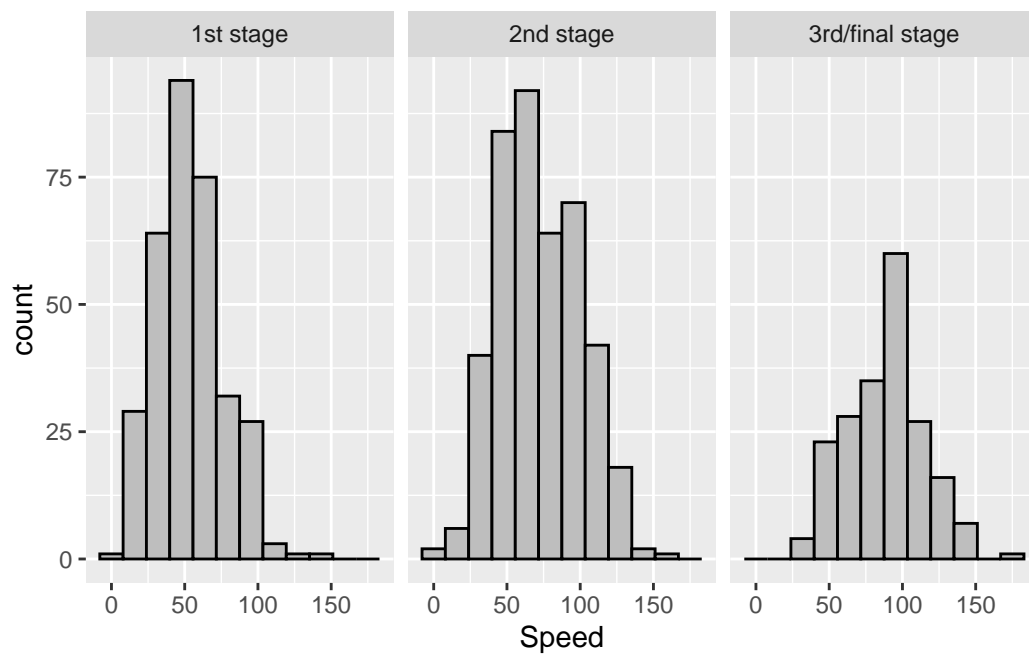
```
pokemon |> ggplot(aes(x = type_1, y=speed)) +
  geom_boxplot(color="black", fill="gray") +
  labs(y = "Pokemon speed points",
       x = "Primary type") +
  coord_flip()
```



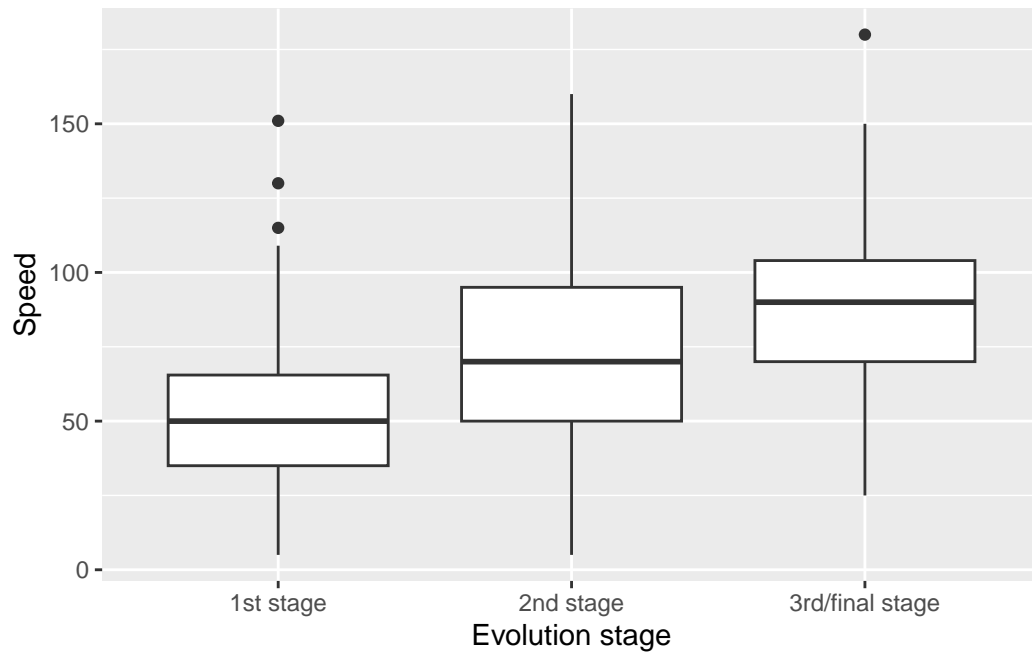
```
## Not responsible for fct_reorder() ##
pokemon |>
  ggplot(aes(x = fct_reorder(type_1, speed, median, .desc = TRUE), y = speed)) +
  geom_boxplot(color = "black", fill = "gray") +
  labs(y = "Pokemon speed points",
       x = "Primary type") +
  coord_flip()
```



```
# Comparing across group with histograms
pokemon |> ggplot(aes(x = speed)) +
  geom_histogram(color="black", fill="gray", bins=12) +
  facet_wrap(~evolution_stage) +
  xlab("Speed")
```

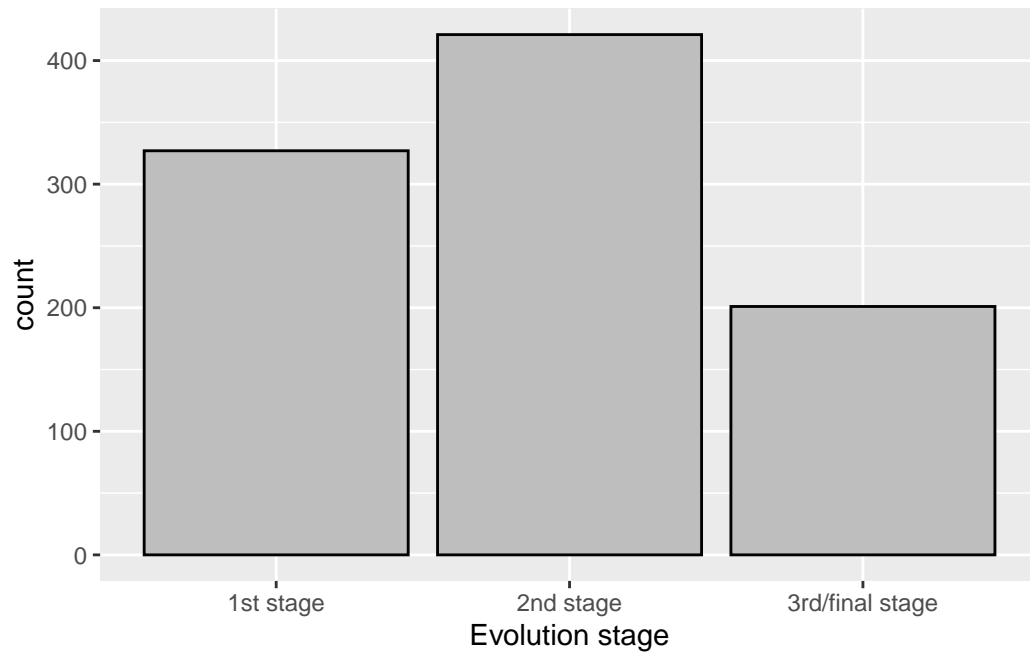


```
pokemon |> ggplot(aes(x = evolution_stage, y = speed)) +  
  geom_boxplot() +  
  xlab("Evolution stage") +  
  ylab("Speed")
```

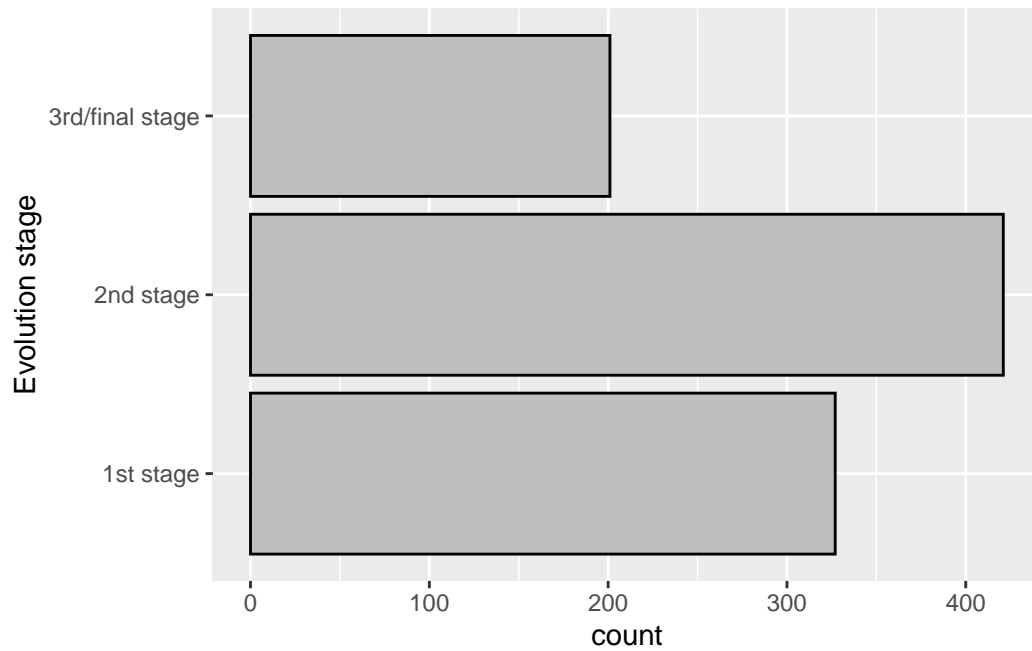


## Barplots

```
## Bar plots
pokemon |> ggplot(aes(x = evolution_stage)) +
  geom_bar(color="black", fill="gray") +
  xlab("Evolution stage")
```

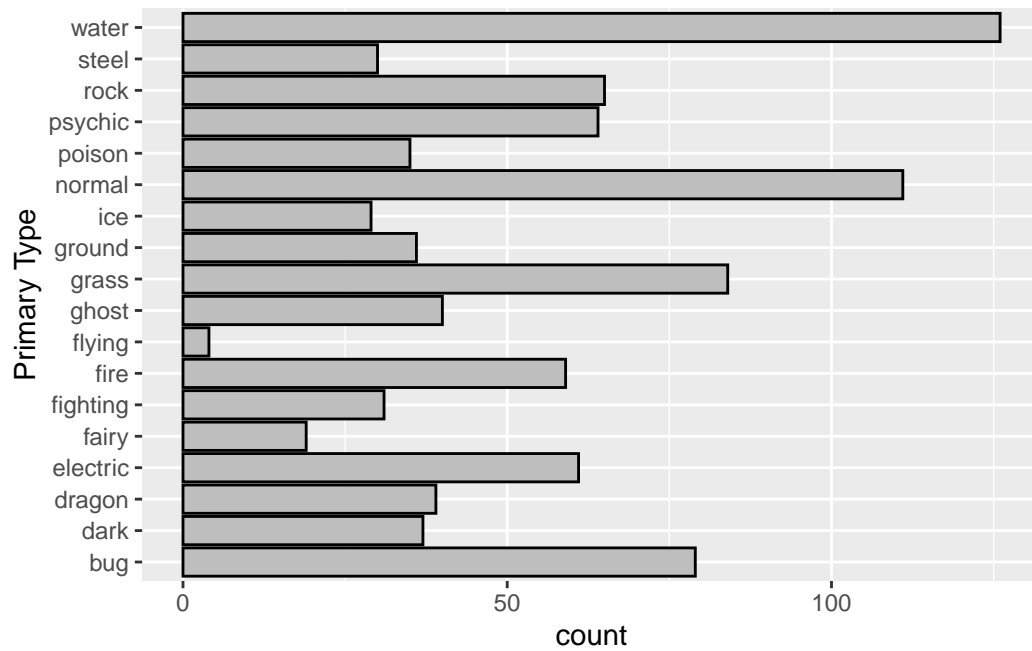


```
pokemon |> ggplot(aes(x = evolution_stage)) +  
  geom_bar(color="black", fill="gray") +  
  xlab("Evolution stage") +  
  coord_flip()
```

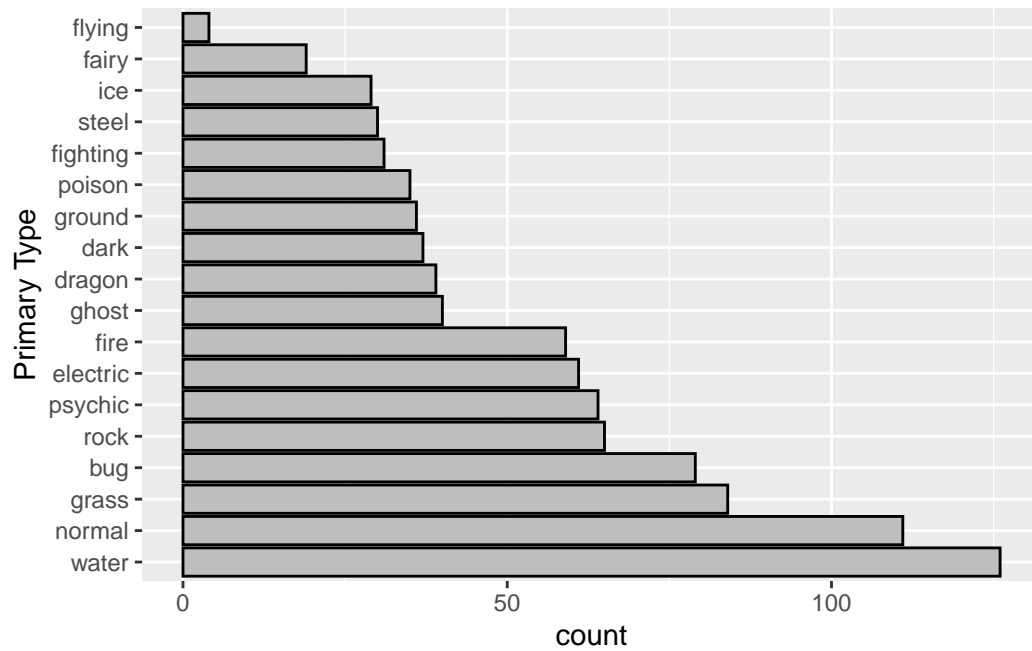


```
pokemon |> ggplot(aes(x = type_1)) +  
  geom_bar(color="black", fill="gray") +  
  xlab("Primary Type") +  
  coord_flip()
```

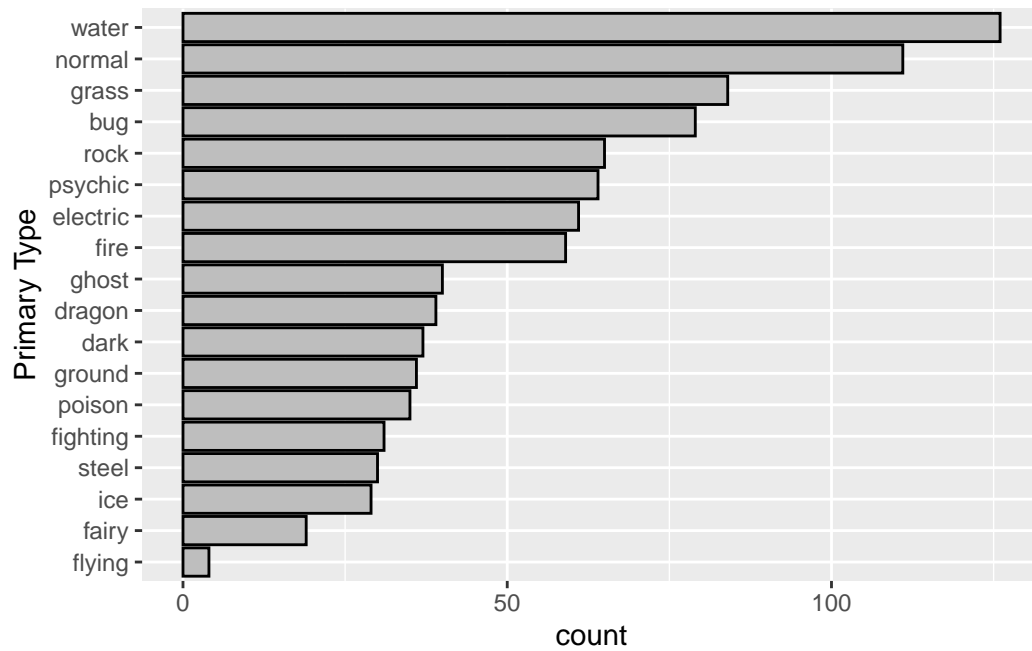




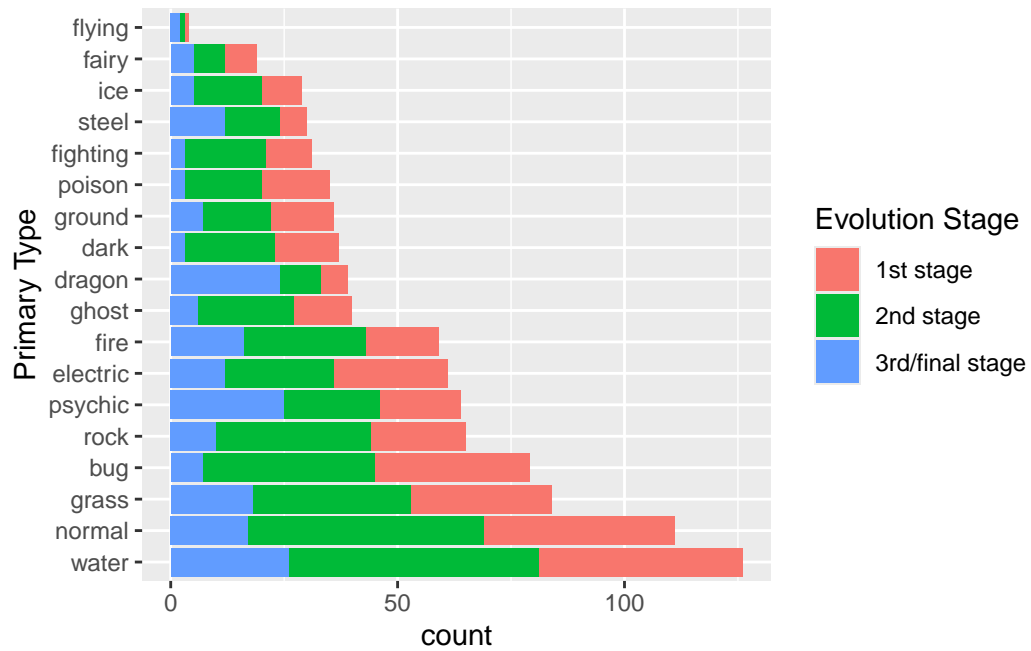
```
pokemon |>  
  ggplot(aes(x = fct_infreq(type_1))) +  
  geom_bar(color = "black", fill = "gray") +  
  xlab("Primary Type") +  
  coord_flip()
```



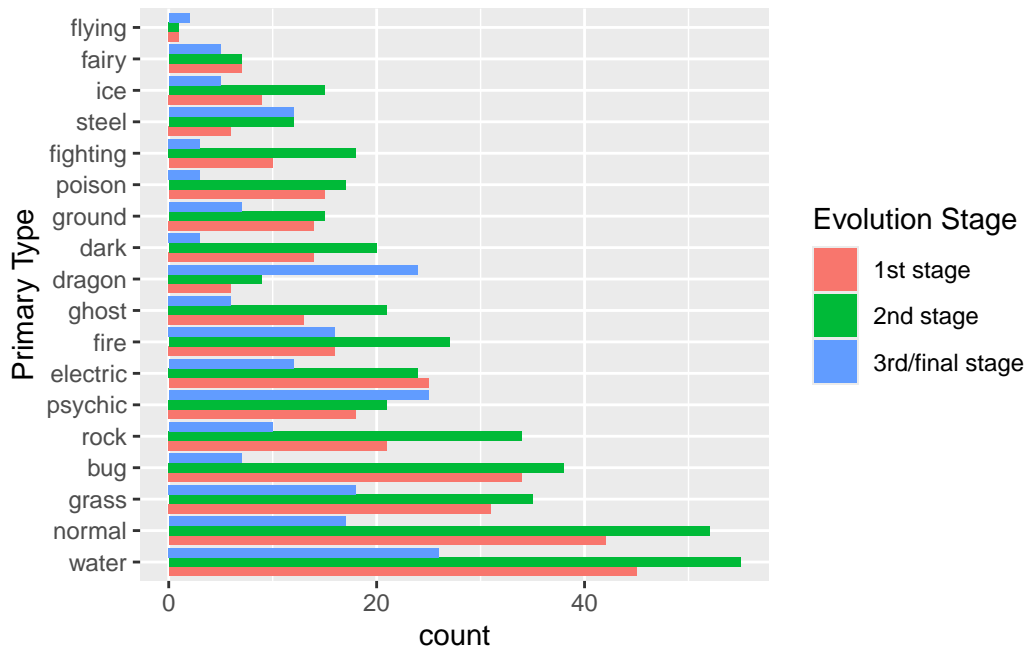
```
pokemon |>  
  ggplot(aes(x = fct_rev(fct_infreq(type_1)))) +  
  geom_bar(color = "black", fill = "gray") +  
  xlab("Primary Type") +  
  coord_flip()
```



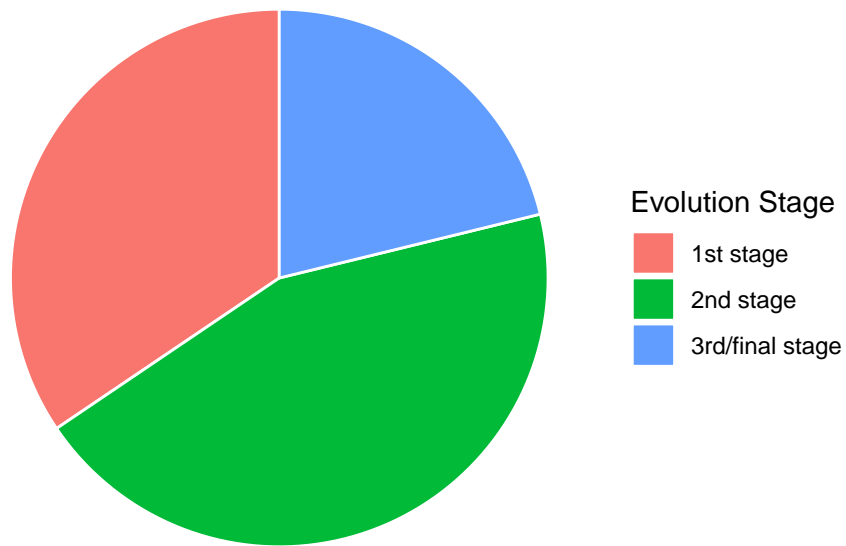
```
## Stacked barplots
pokemon |>
  ggplot(aes(x=fct_infreq(type_1), fill=evolution_stage)) +
  geom_bar(position = "stack") +
  labs(x = "Primary Type",
       fill = "Evolution Stage") +
  coord_flip()
```



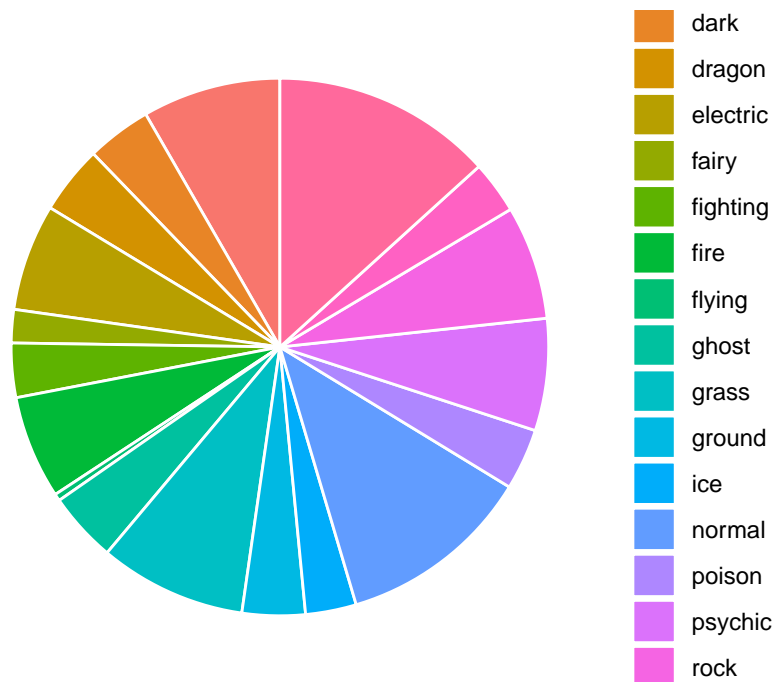
```
pokemon |>
  ggplot(aes(x=fct_infreq(type_1), fill=evolution_stage)) +
  geom_bar(position = "dodge") +
  labs(x = "Primary Type",
       fill = "Evolution Stage") +
  coord_flip()
```



```
## Other ways to visualize categorical data: Pie Charts
### Important: Note the important disadvantages of pie charts mentioned in class
### You're not responsible for creating pie charts in STA130
### For more advanced options: https://r-graph-gallery.com/piechart-ggplot2.html
pokemon |>
  group_by(evolution_stage) |>
  summarise(n=n()) |>
  ggplot(aes(x="", y=n, fill=evolution_stage)) +
  geom_bar(stat="identity", width=1, color="white") +
  coord_polar("y", start=0) +
  theme_void() +
  labs(fill="Evolution Stage")
```



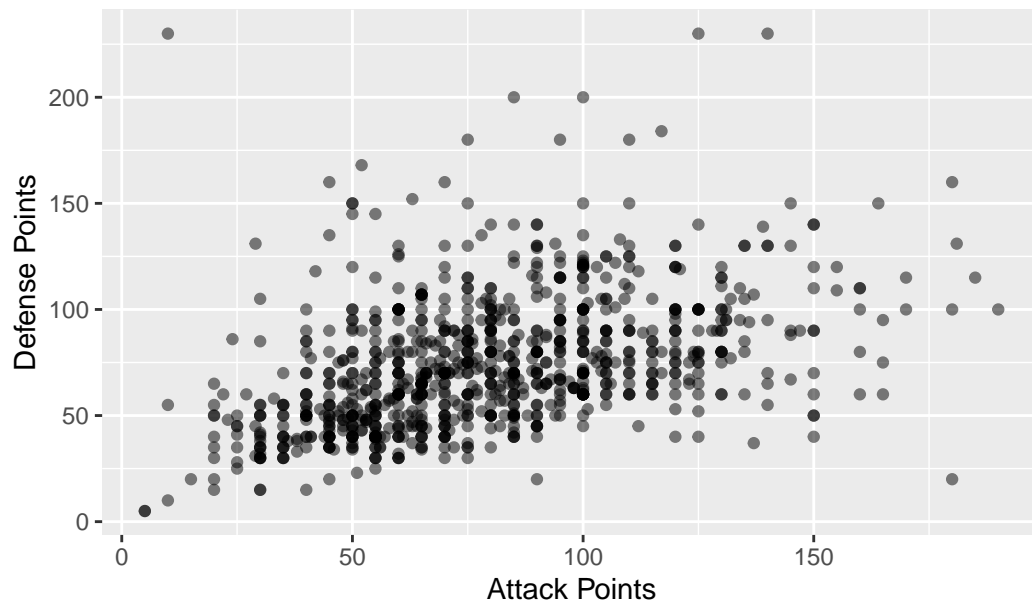
```
pokemon |>
  group_by(type_1) |>
  summarise(n=n()) |>
  ggplot(aes(x="", y=n, fill=type_1)) +
  geom_bar(stat="identity", width=1, color="white") +
  coord_polar("y", start=0) +
  theme_void() +
  labs(fill="Primary Type")
```



## Scatterplots

```
pokemon |>
  ggplot(aes(x = attack, y = defense)) +
  geom_point(alpha = 0.5) +
  labs(
    title = "Pokemon Attack vs Defense",
    x = "Attack Points",
    y = "Defense Points")
```

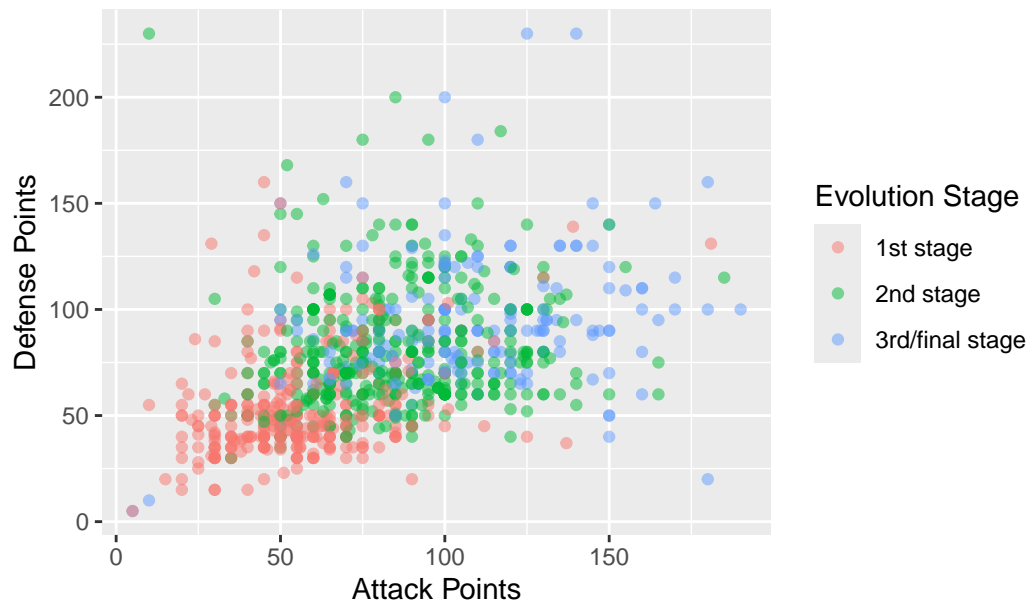
Pokemon Attack vs Defense



```
pokemon |>
  ggplot(aes(x = attack, y = defense, color=evolution_stage)) +
  geom_point(alpha = 0.5) +
  labs(
    title = "Pokemon Attack vs Defense by Evolution Stage",
    x = "Attack Points",
    y = "Defense Points",
    color = "Evolution Stage")
```



Pokemon Attack vs Defense by Evolution Stage



```
pokemon |>
  ggplot(aes(x = attack, y = defense, color=evolution_stage)) +
  geom_point(alpha = 0.5) +
  facet_wrap(~evolution_stage) +
  labs(
    title = "Pokemon Attack vs Defense by Evolution Stage",
    x = "Attack Points",
    y = "Defense Points",
    color = "Evolution Stage")
```

Pokemon Attack vs Defense by Evolution Stage

