# R Notebook

Code ▾

Hide

```
library(ggplot2)
library(dplyr)
```

```
package <U+393C><U+3E31>dplyr<U+393C><U+3E32> was built under R version 3.3.3
Attaching package: <U+393C><U+3E31>dplyr<U+393C><U+3E32>

The following objects are masked from <U+393C><U+3E31>package:stats<U+393C><U+3E32>:

    filter, lag

The following objects are masked from <U+393C><U+3E31>package:base<U+393C><U+3E32>:

    intersect, setdiff, setequal, union
```

Hide

```
iris %>% summarise(min=min(Sepal.Length))
```

| min<br><dbl> |
|---:|
| 4.3 |

1 row

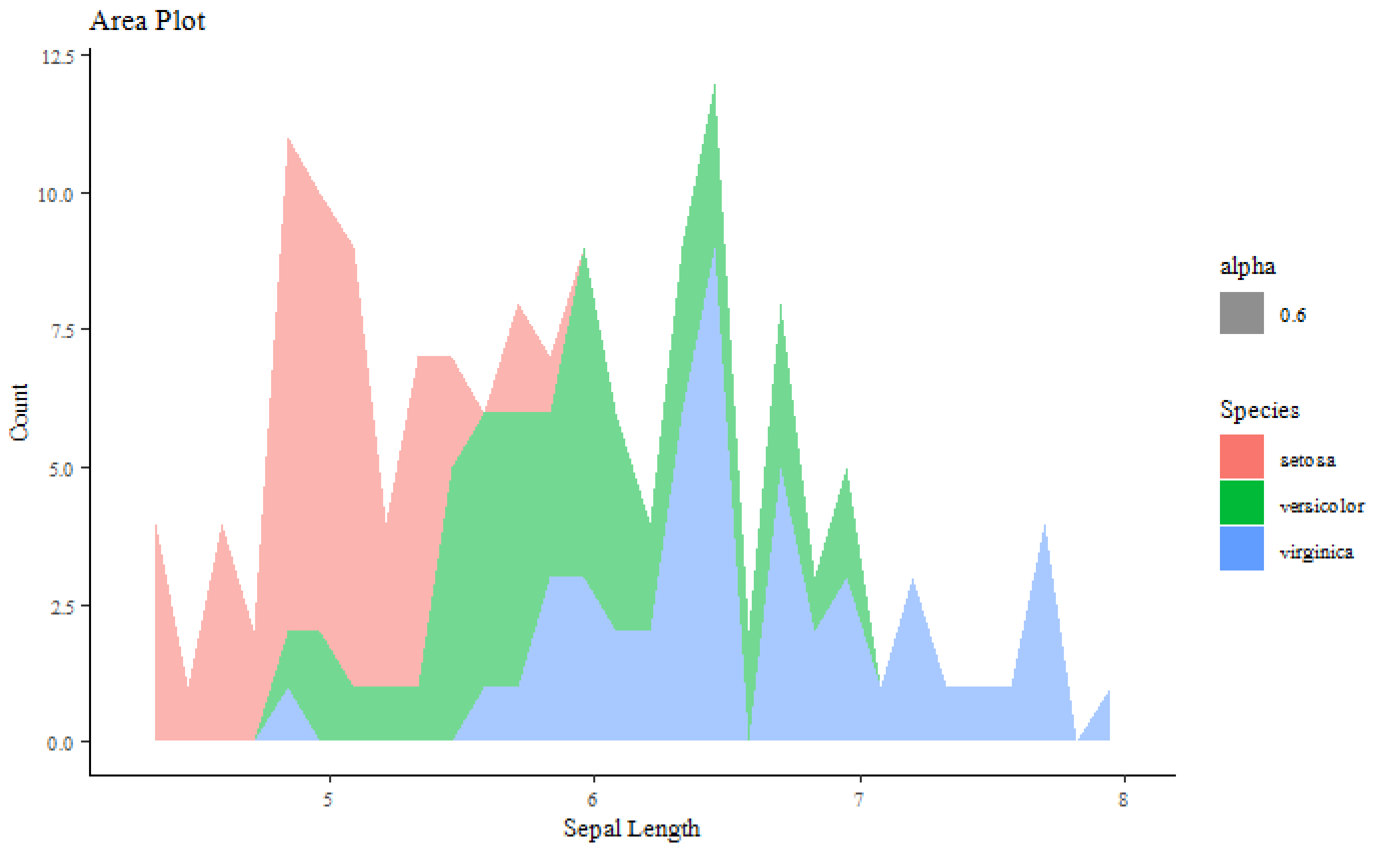
For 1 continuous variable:-

Area charts- are mainly applied to represent data that follows a time-series relationship. Difference between line graphs and area chart is that it’s easy to see that the area below plotted lines are filled with colors. The scale of a color fill indicates the magnitude of the corresponding value. When the magnitude of the trend needs to be communicated, rather than individual values, area charts work best.

For numbers such as 1, 2, 3, and so on, the magnitude is simply the number itself. If the number is negative, the magnitudebecomes the absolute value of the number.

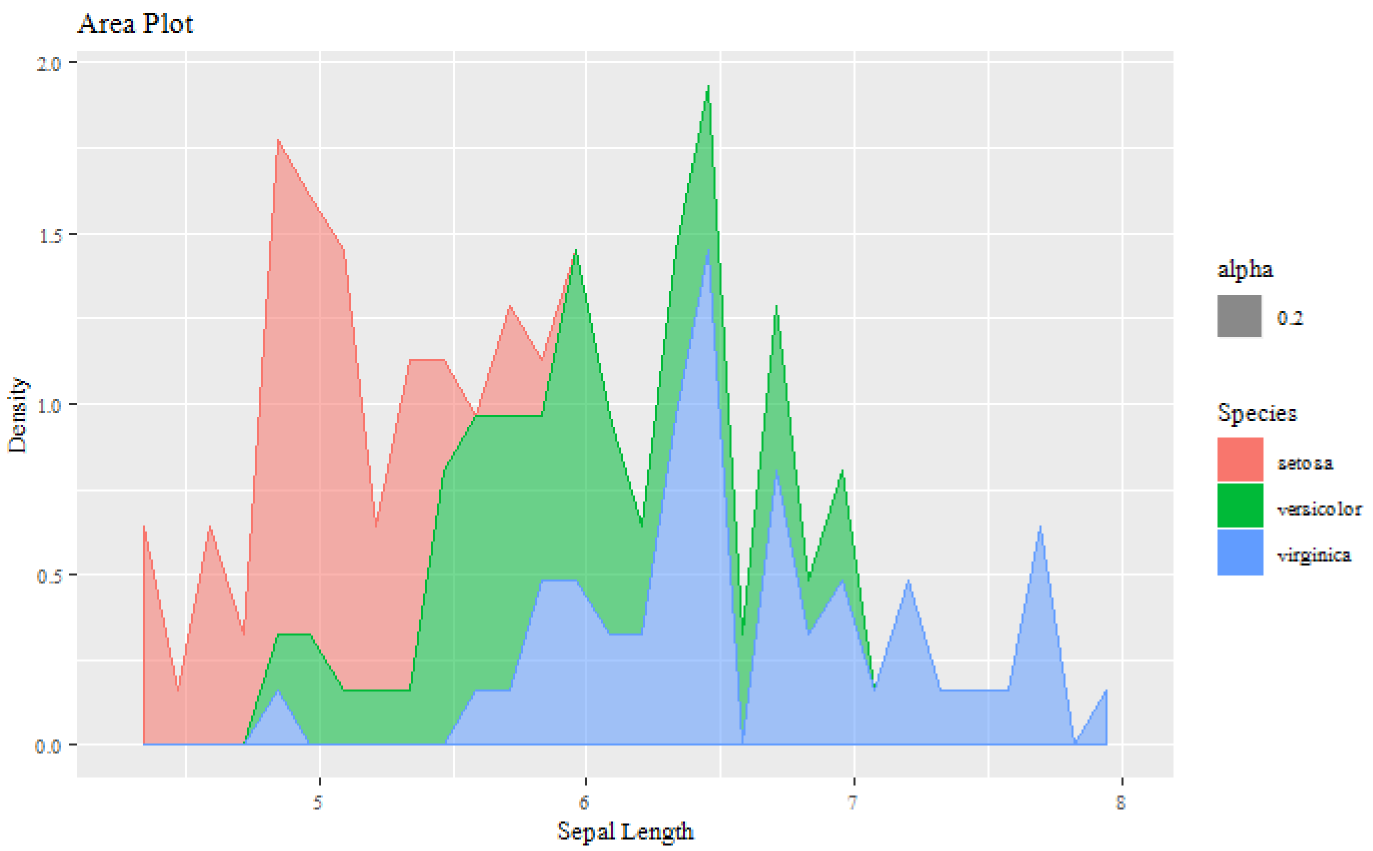
Hide

```
#Area Plot
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
labs(x="Sepal Length",y="Count",title="Area Plot")+
theme_classic()
```



Hide

```
ggplot(iris,aes(x=Sepal.Length,fill=Species,color=Species,alpha=0.2))+
geom_area(aes(y = ..density..), stat ="bin")+
ggtitle("Area Plot")+
xlab("Sepal Length")+
ylab("Density")
```



Hide

```
table(iris$Sepal.Length)
```

```


4.3 4.4 4.5 4.6 4.7 4.8 4.9   5 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9   6 6.1 6.2 6.3 6.4 6.5 6.6
6.7 6.8 6.9   7 7.1
  1   3   1   4   2   5   6  10   9   4   1   6   7   6   8   7   3   6   6   4   9   7   5   2
  8   3   4   1   1
7.2 7.3 7.4 7.6 7.7 7.9
  3   1   1   1   4   1
```

Hide

```
# Aesthetics could be : x, y, alpha, color, fill, linetype, size
```

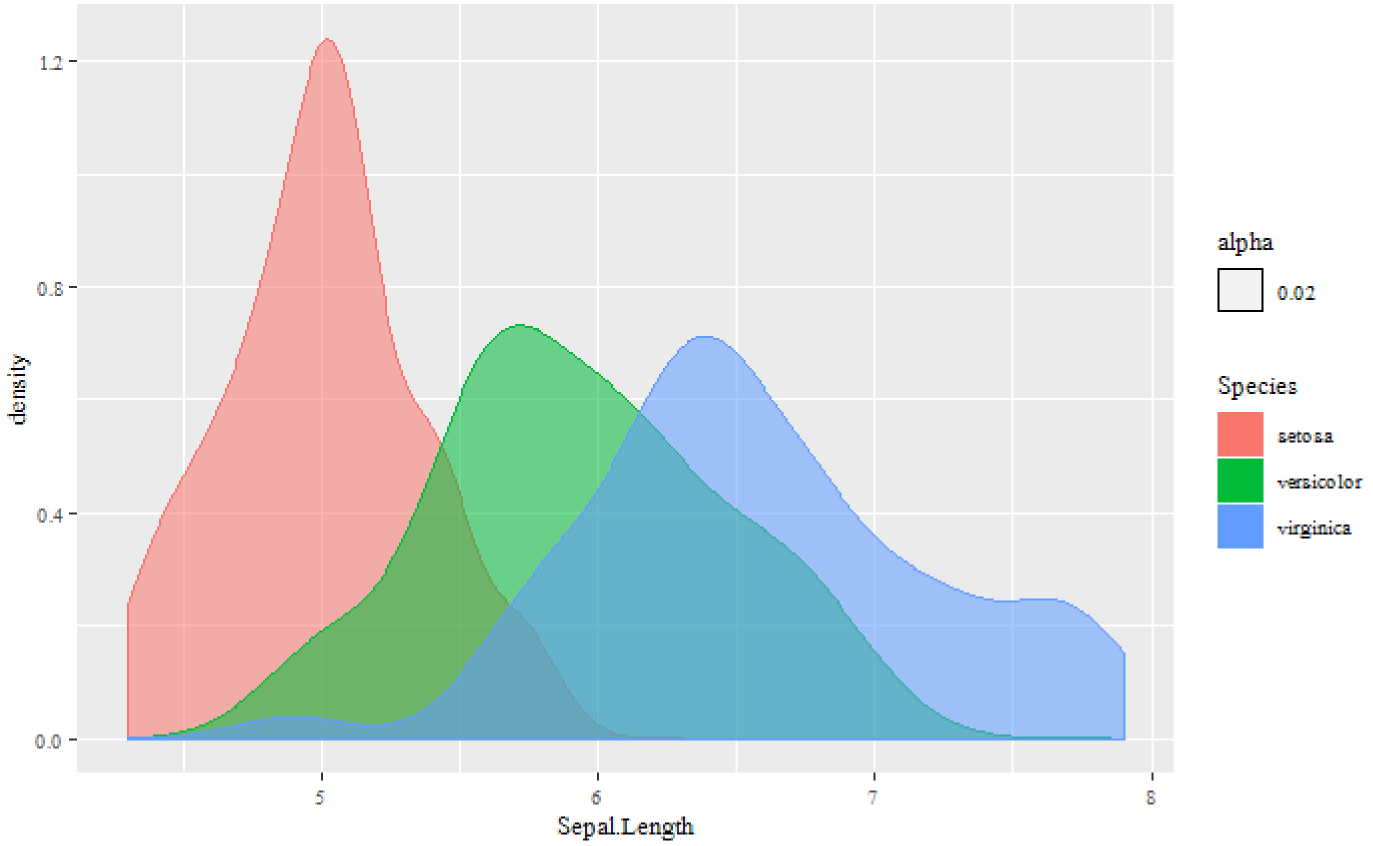
Use labels i.e. labs after geom

Typical Aesthetics:-

x- X axis position y- Y axis position colour- Colour of dots, outlines of other shapes fill- Fill colour size- Diameter of points, thickness of lines(0 to 10 may be) alpha- Transparency(value 0 to 1) linetype- Line dash pa!ern(value- 0 to 6, 0 for no line) labels- Text on a plot or axes(xlab,ylab,title) shape- Shape (value- 0 to 25)

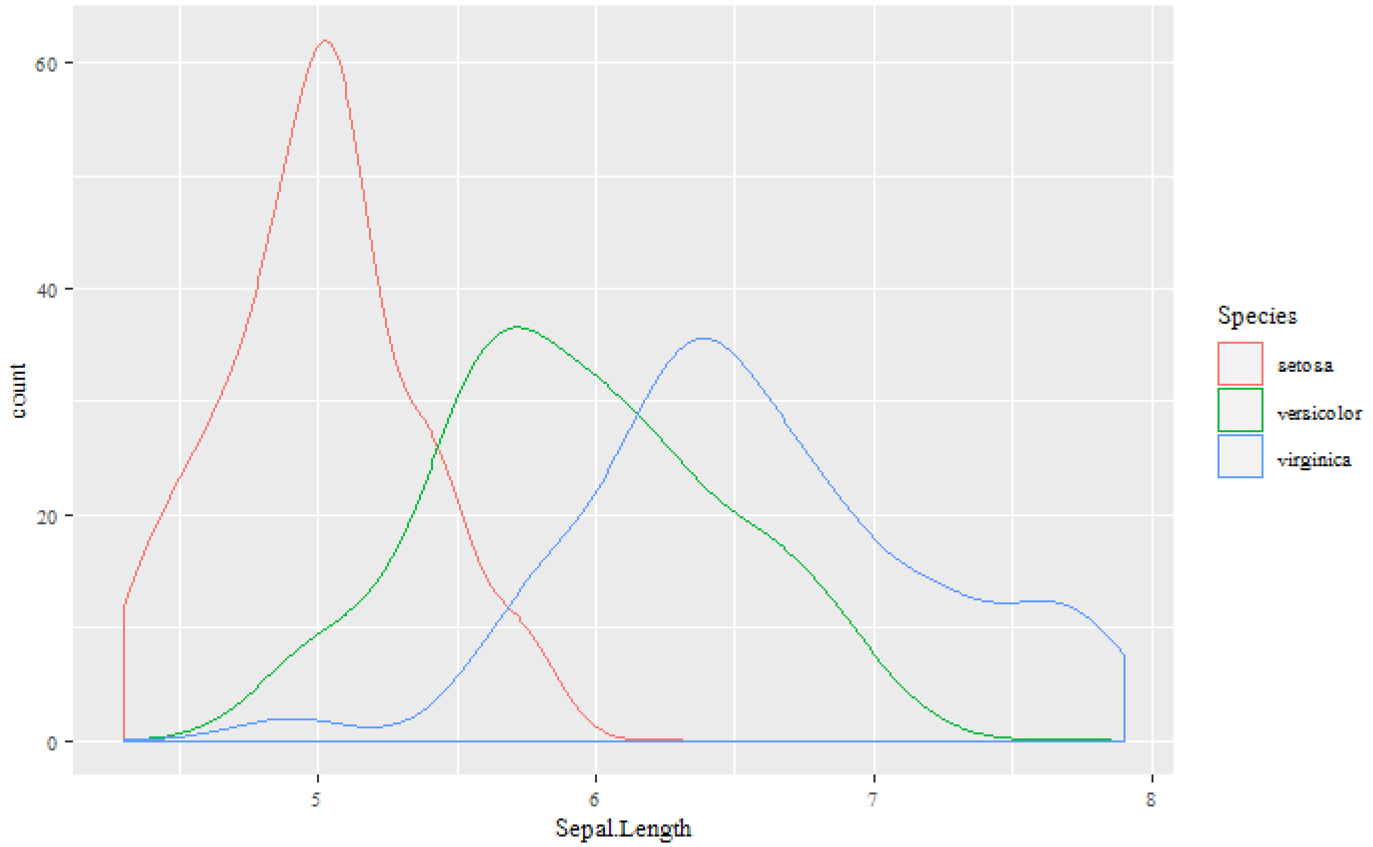
Hide

```
#Density Plot
ggplot(iris,aes(x=Sepal.Length,alpha=0.02,fill=Species,color=Species))+
geom_density(kernel = "gaussian")
```



Hide

```
ggplot(iris,aes(x=Sepal.Length,color=Species))+
geom_density(aes(y = ..count..))
```

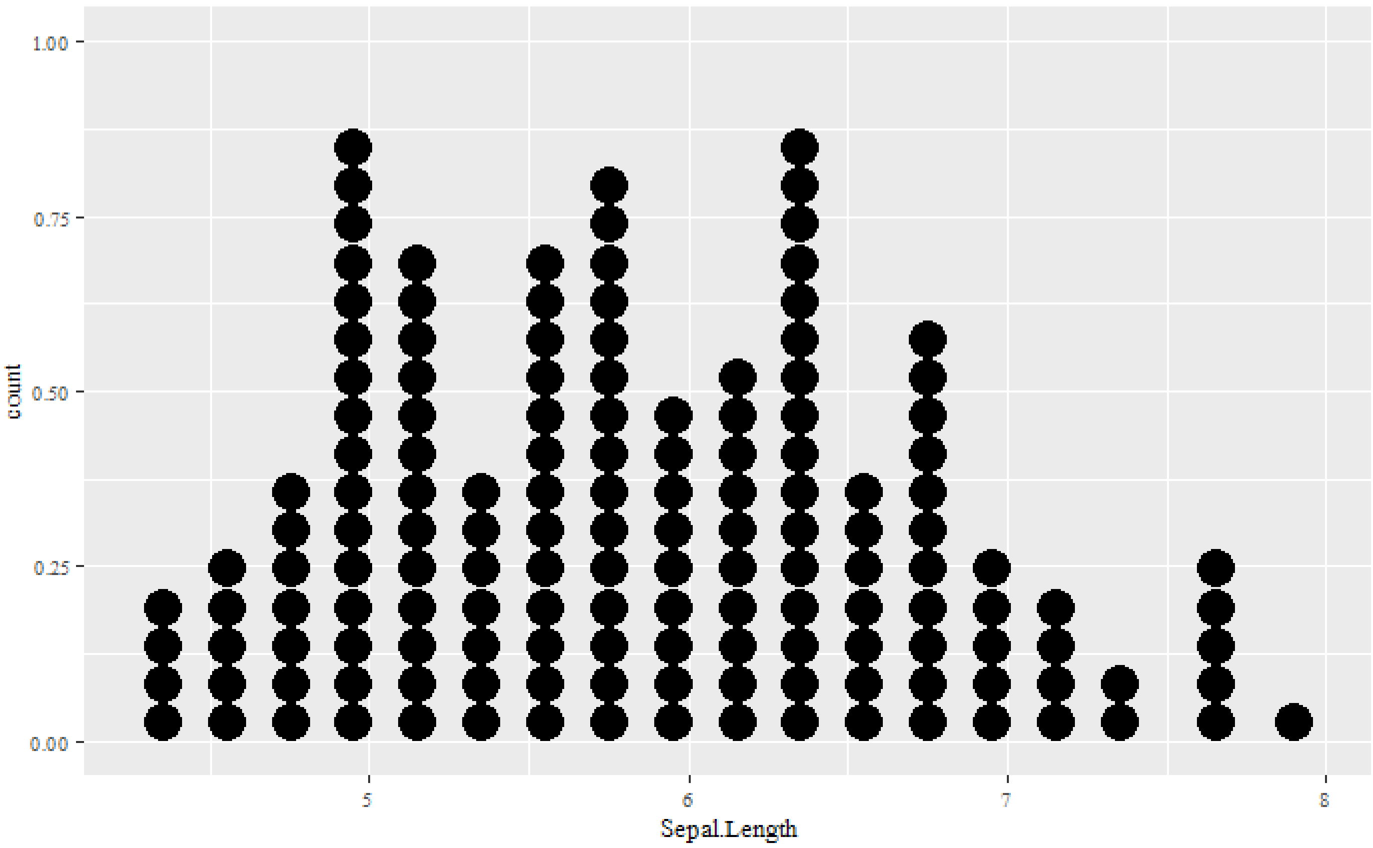


Hide

```
#Aesthetics could be - x, y, alpha, color, fill, linetype, size, weight(it helps in computation
 of quantiles)
```

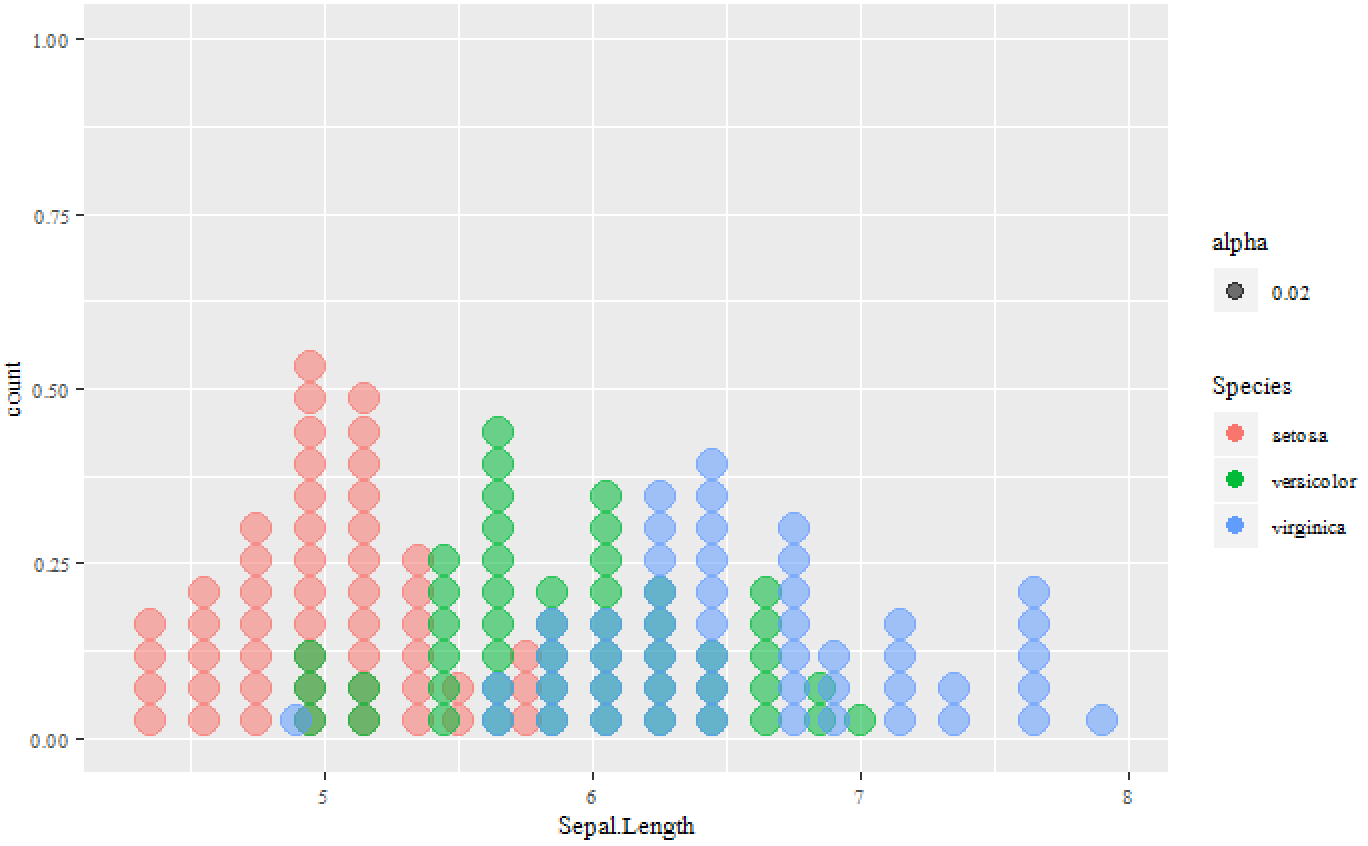
Hide

```
#Dot plot
ggplot(iris,aes(x=Sepal.Length))+
geom_dotplot()
```



Hide

```
ggplot(iris,aes(x=Sepal.Length,alpha=0.02,fill=Species,color=Species))+
geom_dotplot()
```



Hide

```
#Aesthetics could be: x, y, alpha, color, fill
```

Hide

```
as.factor(iris$Sepal.Length)
```

```
  [1] 5.1 4.9 4.7 4.6 5   5.4 4.6 5   4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4 5.1 5.7 5.1 5.4 5.1 4.
6 5.1 4.8 5   5
 [28] 5.2 5.2 4.7 4.8 5.4 5.2 5.5 4.9 5   5.5 4.9 4.4 5.1 5   4.5 4.4 5   5.1 4.8 5.1 4.6 5.3 5  
  7   6.4 6.9 5.5
 [55] 6.5 5.7 6.3 4.9 6.6 5.2 5   5.9 6   6.1 5.6 6.7 5.6 5.8 6.2 5.6 5.9 6.1 6.3 6.1 6.4 6.6 6.
8 6.7 6   5.7 5.5
 [82] 5.5 5.8 6   5.4 6   6.7 6.3 5.6 5.5 5.5 6.1 5.8 5   5.6 5.7 5.7 6.2 5.1 5.7 6.3 5.8 7.1 6.
3 6.5 7.6 4.9 7.3
[109] 6.7 7.2 6.5 6.4 6.8 5.7 5.8 6.4 6.5 7.7 7.7 6   6.9 5.6 7.7 6.3 6.7 7.2 6.2 6.1 6.4 7.2 7.
4 7.9 6.4 6.3 6.1
[136] 7.7 6.3 6.4 6   6.9 6.7 6.9 5.8 6.8 6.7 6.7 6.3 6.5 6.2 5.9
35 Levels: 4.3 4.4 4.5 4.6 4.7 4.8 4.9 5 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 6 6.1 6.2 6.3 6.4
6.5 6.6 6.7 ... 7.9
```

Hide

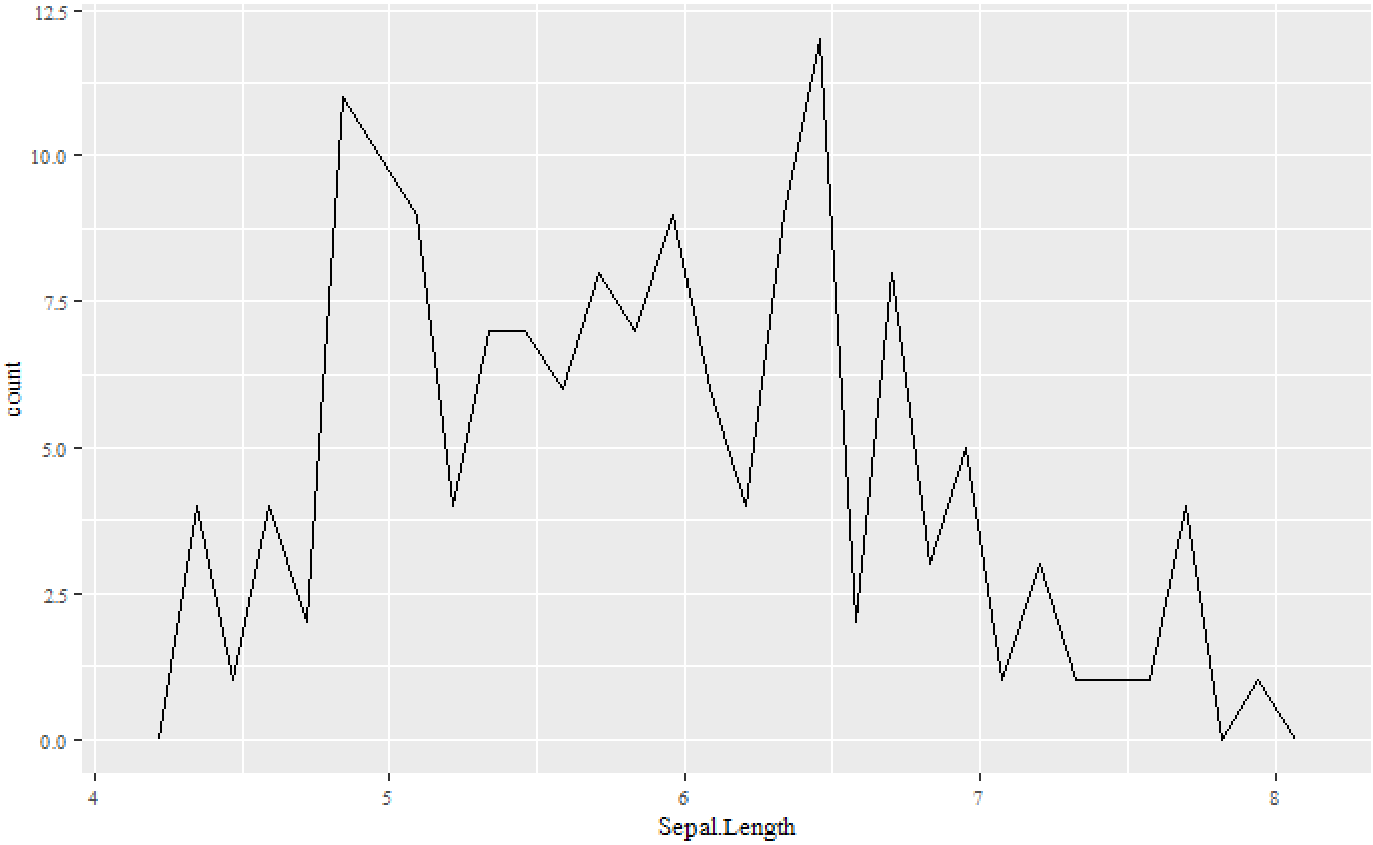
```
table(iris$Sepal.Length,iris$Species)
```

```
      setosa versicolor virginica
  4.3      1          0         0
  4.4      3          0         0
  4.5      1          0         0
  4.6      4          0         0
  4.7      2          0         0
  4.8      5          0         0
  4.9      4          1         1
  5        8          2         0
  5.1      8          1         0
  5.2      3          1         0
  5.3      1          0         0
  5.4      5          1         0
  5.5      2          5         0
  5.6      0          5         1
  5.7      2          5         1
  5.8      1          3         3
  5.9      0          2         1
  6        0          4         2
  6.1      0          4         2
  6.2      0          2         2
  6.3      0          3         6
  6.4      0          2         5
  6.5      0          1         4
  6.6      0          2         0
  6.7      0          3         5
  6.8      0          1         2
  6.9      0          1         3
  7        0          1         0
  7.1      0          0         1
  7.2      0          0         3
  7.3      0          0         1
  7.4      0          0         1
  7.6      0          0         1
  7.7      0          0         4
  7.9      0          0         1
```

geom_dotplot “count” on y-axis is meaningless. It does not tell the actual count of x-variable. It only tells at paticular point of x variable, how many times that x value occurs.

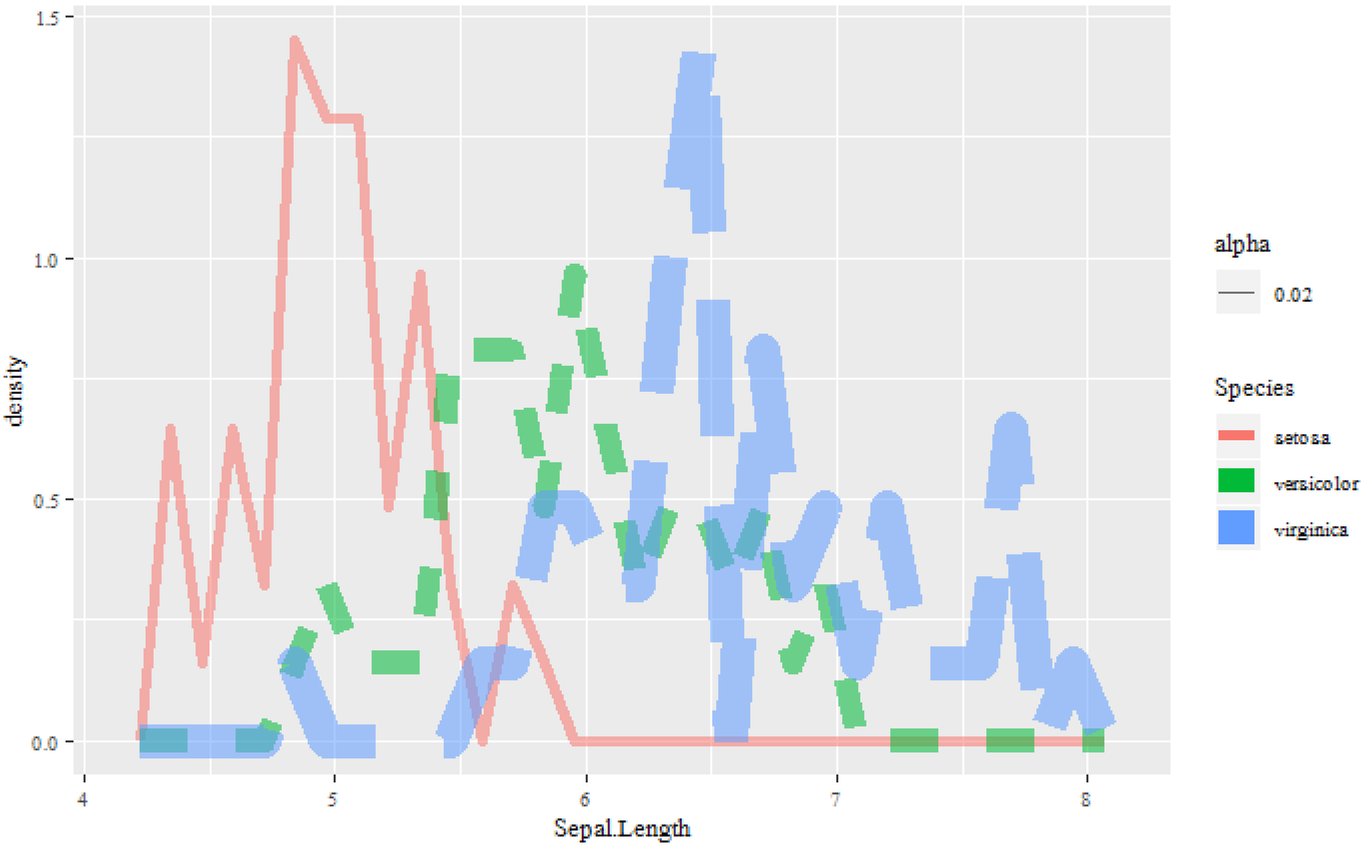
Hide

```
#Frequency Polygon
ggplot(iris,aes(x=Sepal.Length))+
geom_freqpoly()
```



Hide

```
ggplot(iris,aes(x=Sepal.Length,alpha=0.02,color=Species,size=Species,linetype=Species))+
geom_freqpoly(aes(y = ..density..))
```

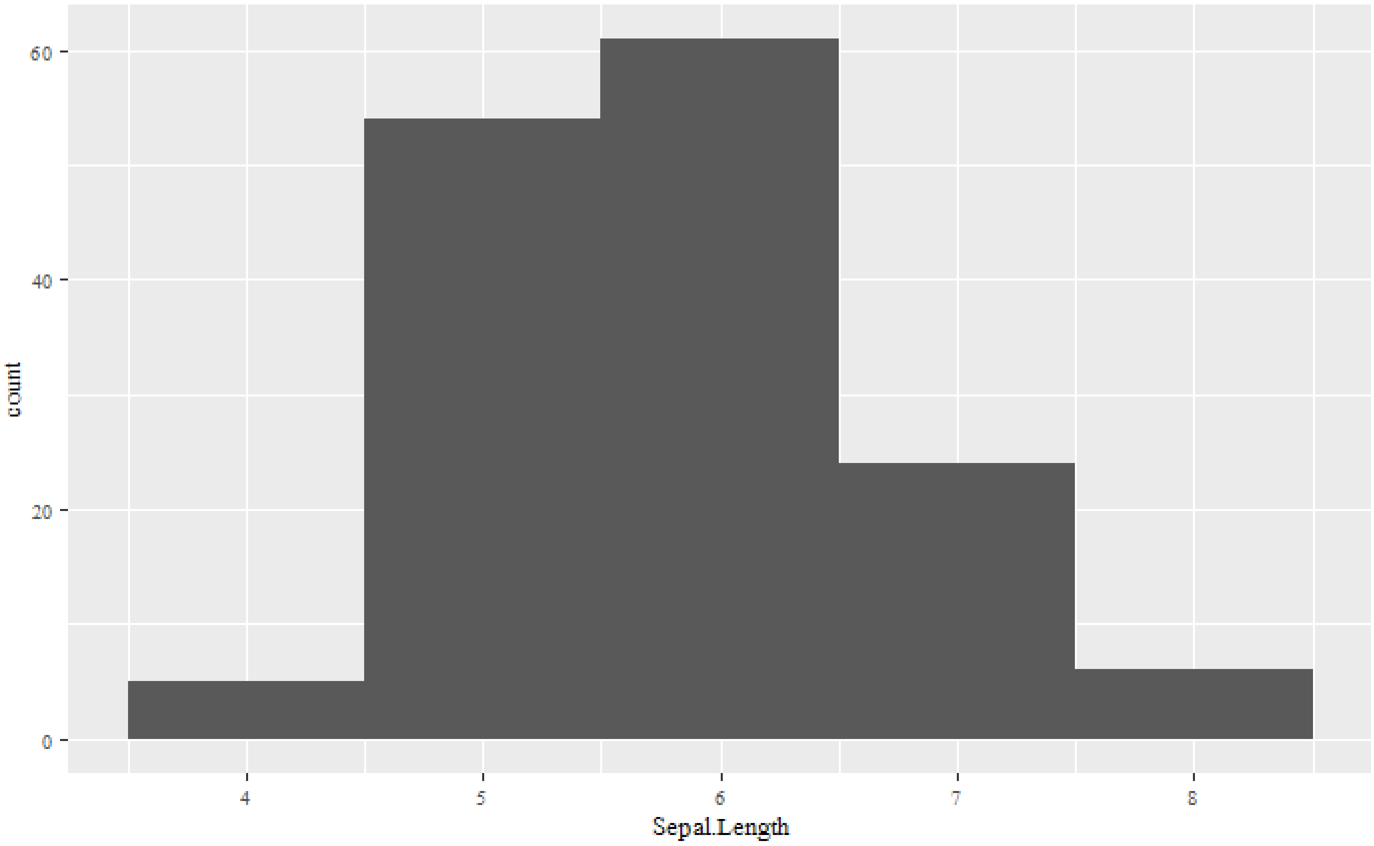


Hide

```
#Aesthetics could be: x, y,alpha,color,linetype,size
```

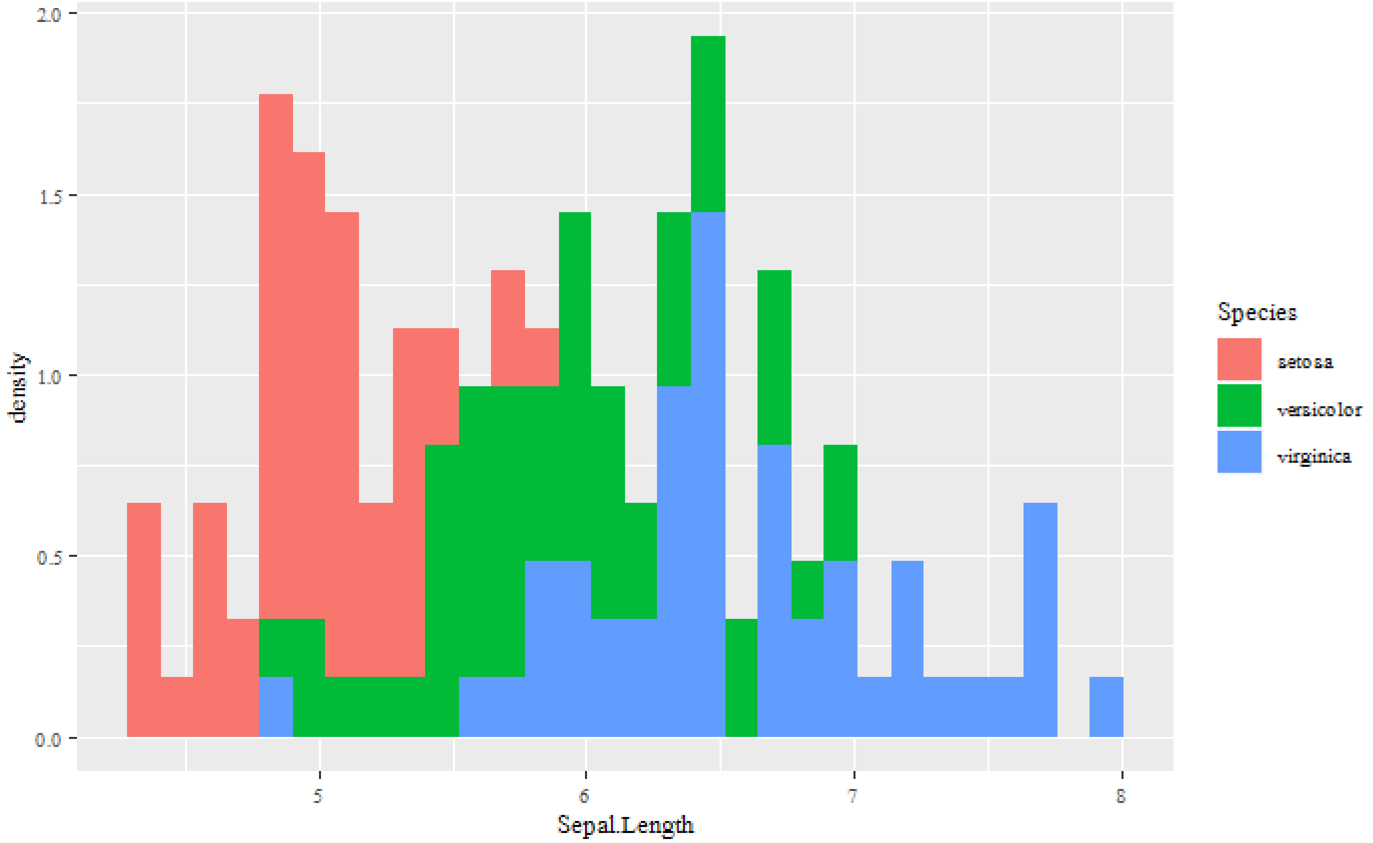
Hide

```
#Histogram
ggplot(iris,aes(x=Sepal.Length))+
geom_histogram(binwidth = 1)
```



Hide

```
ggplot(iris,aes(x=Sepal.Length,fill=Species))+
geom_histogram(aes(y = ..density..))
```



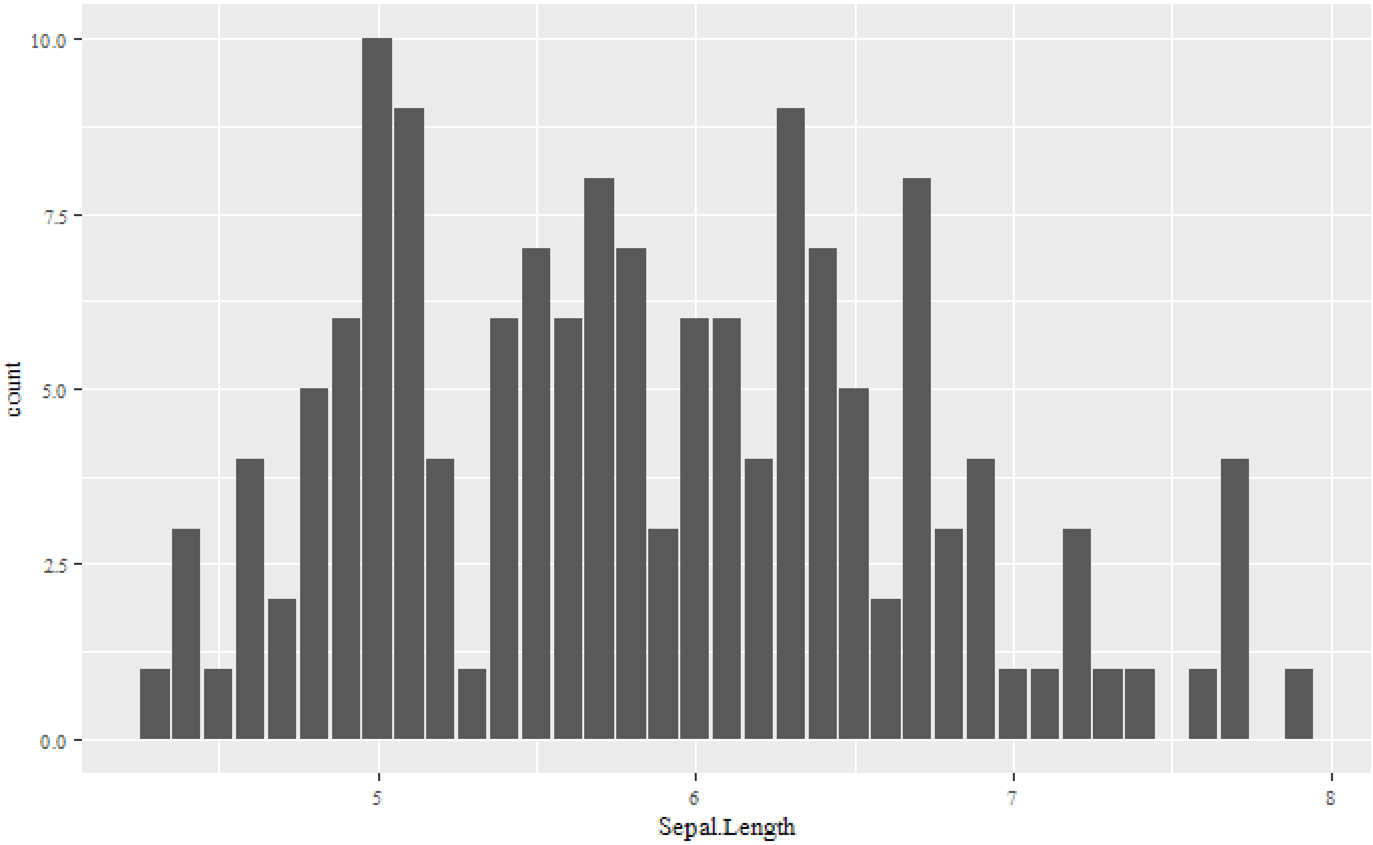
Hide

```
#Aesthetics could be- x, y, alpha, color, fill, linetype, size, weight
```

For 1 Discrete variable

Hide

```
#Bar Plot
ggplot(iris,aes(x=Sepal.Length))+
geom_bar()
```



Hide

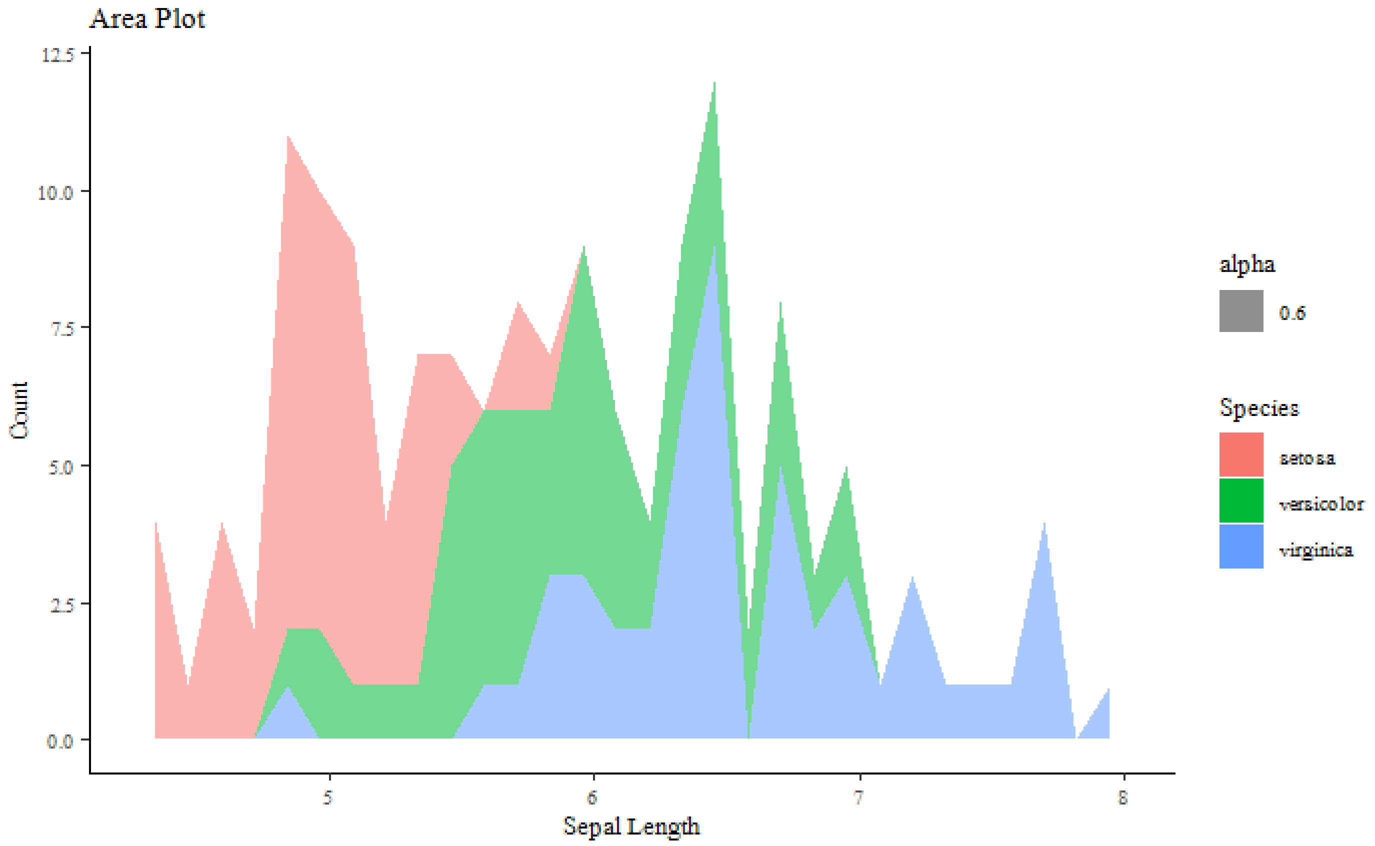
```
#Aesthetics could be - x, alpha, color, fill, linetype, size, weight
```

We can use Aesthetics inside geometries to make changes.

Labels:-Use labels i.e. labs after geom

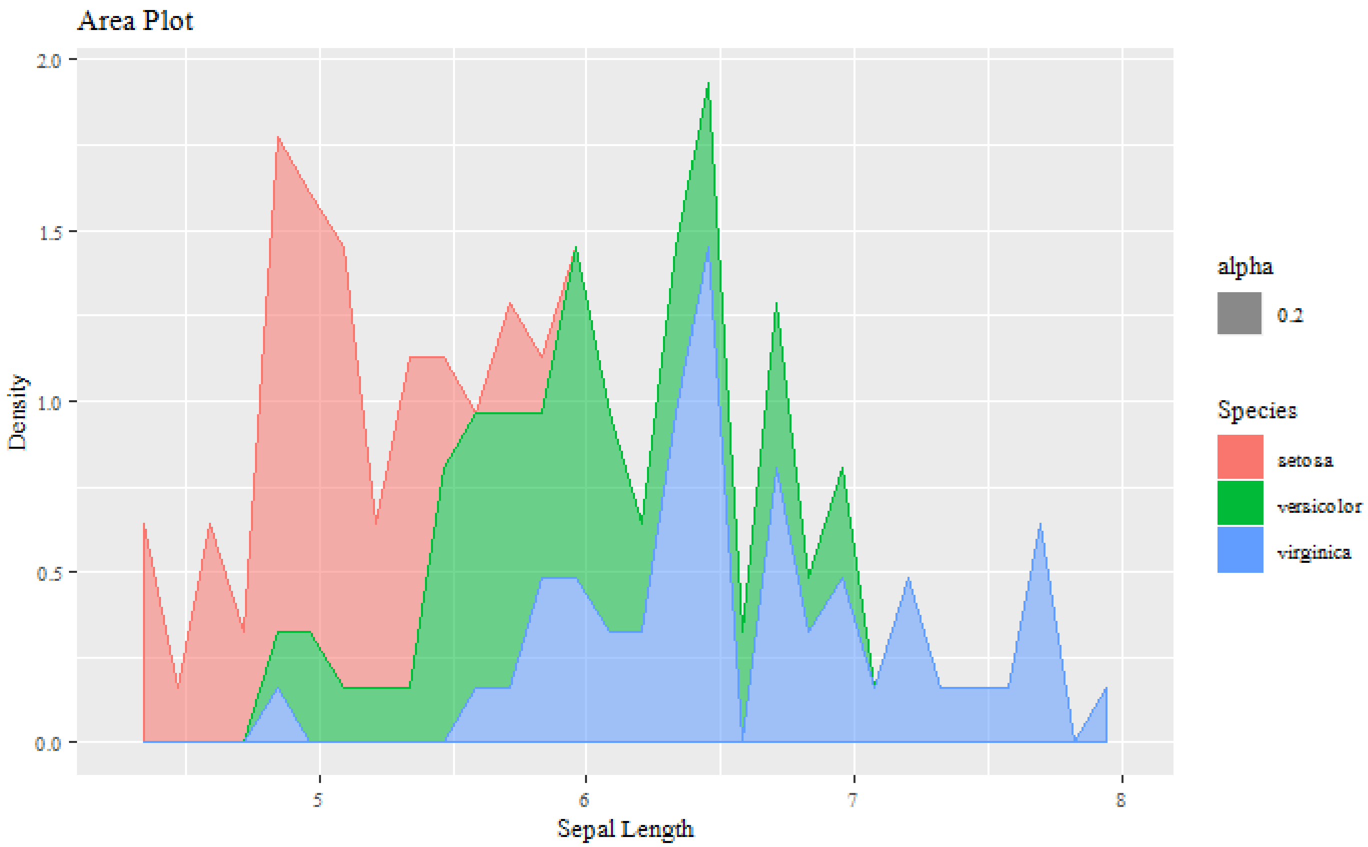
Hide

```
#Area Plot
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
labs(x="Sepal Length",y="Count",title="Area Plot")+
theme_classic()
```



Hide

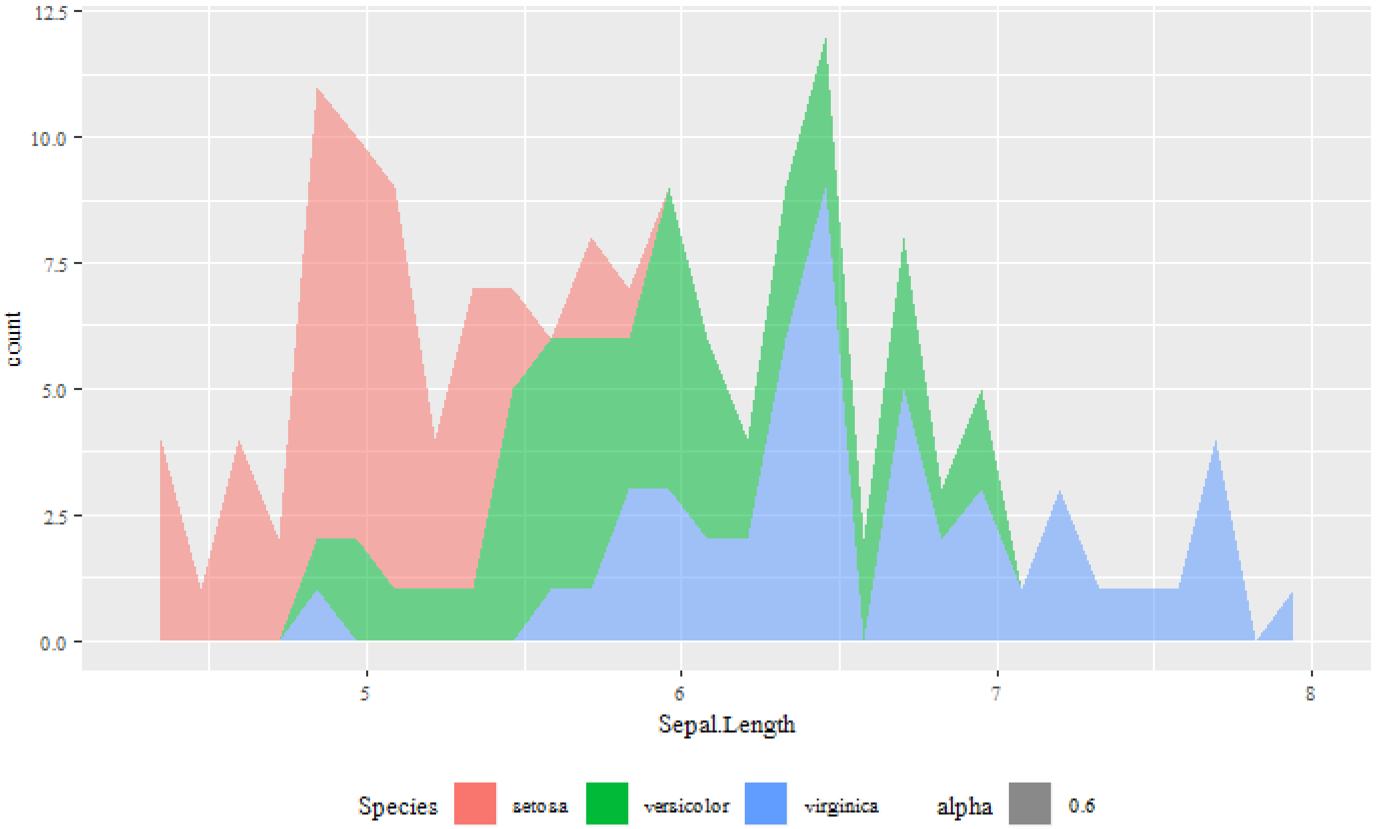
```
ggplot(iris,aes(x=Sepal.Length,fill=Species,color=Species,alpha=0.2))+
geom_area(aes(y = ..density..), stat ="bin")+
ggtitle("Area Plot")+
xlab("Sepal Length")+
ylab("Density")
```



Legends:- Use legends after geom

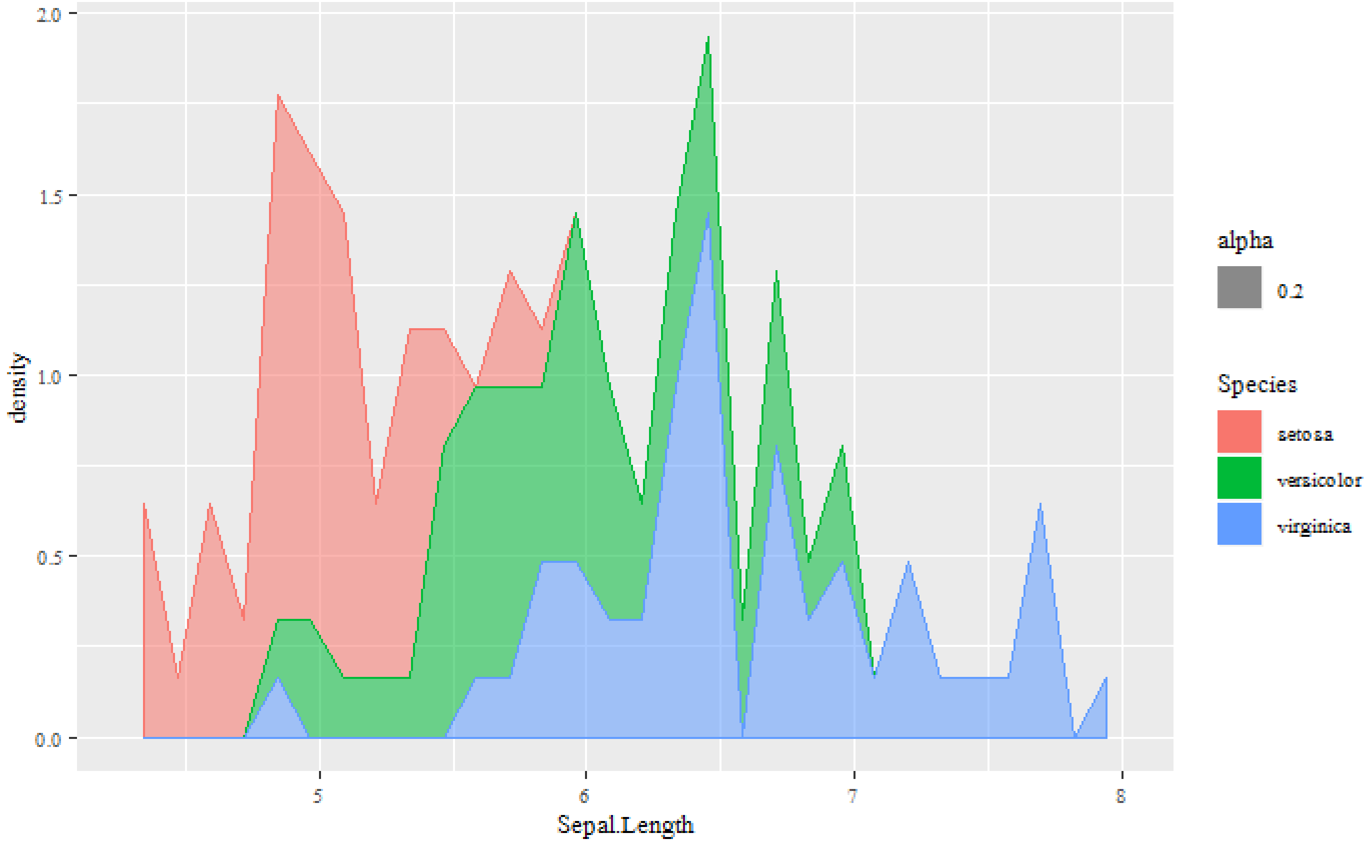
Hide

```
#Area Plot
#Place legend at "bottom", "top", "left", or "right"
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
theme(legend.position="bottom")
```



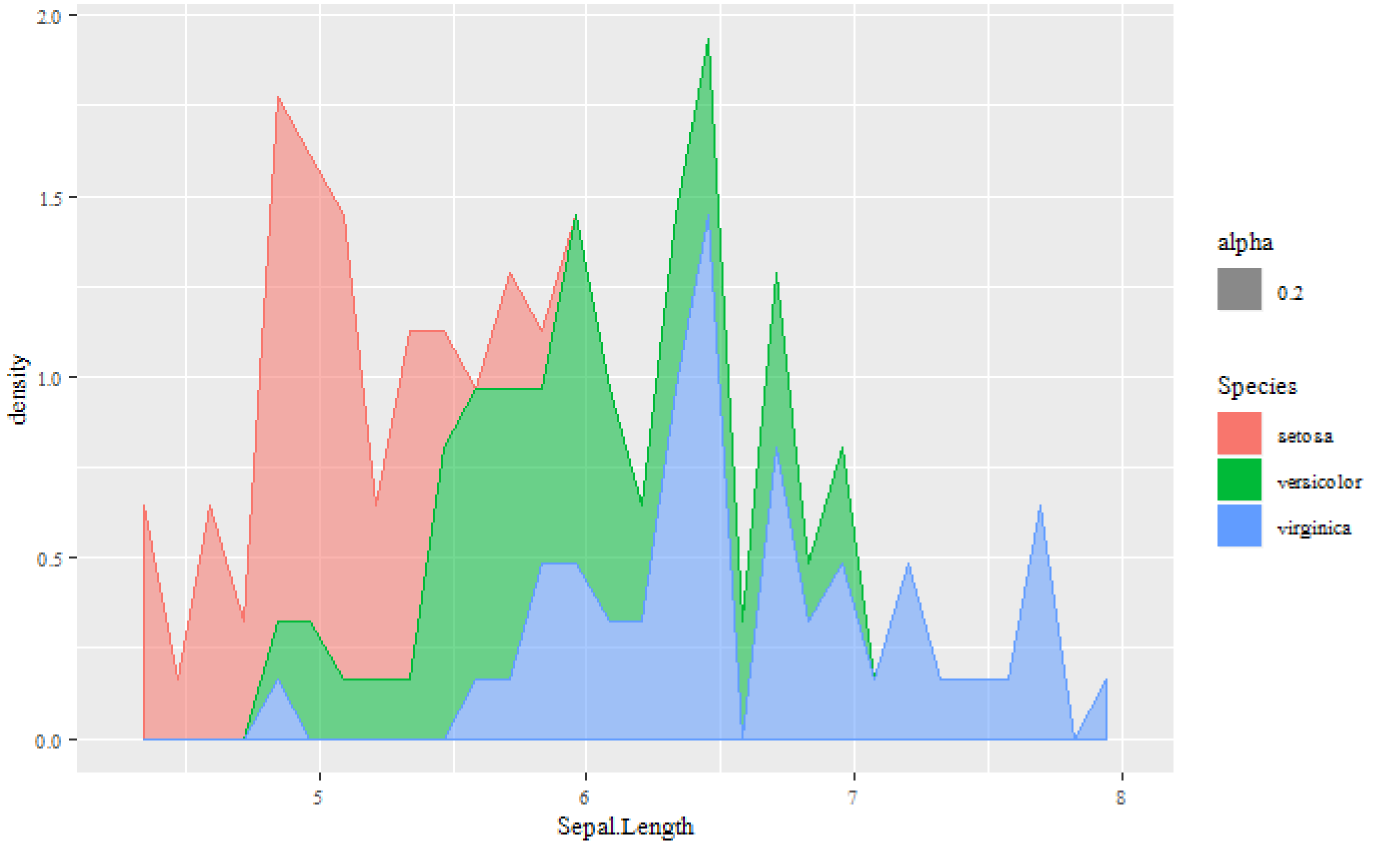
Hide

```
#Set legend type for each aesthetic: colorbar, legend, or none (no legend)
ggplot(iris,aes(x=Sepal.Length,fill=Species,color=Species,alpha=0.2))+
geom_area(aes(y = ..density..), stat ="bin")+
guides(color="none") #same output for color="legend"
```



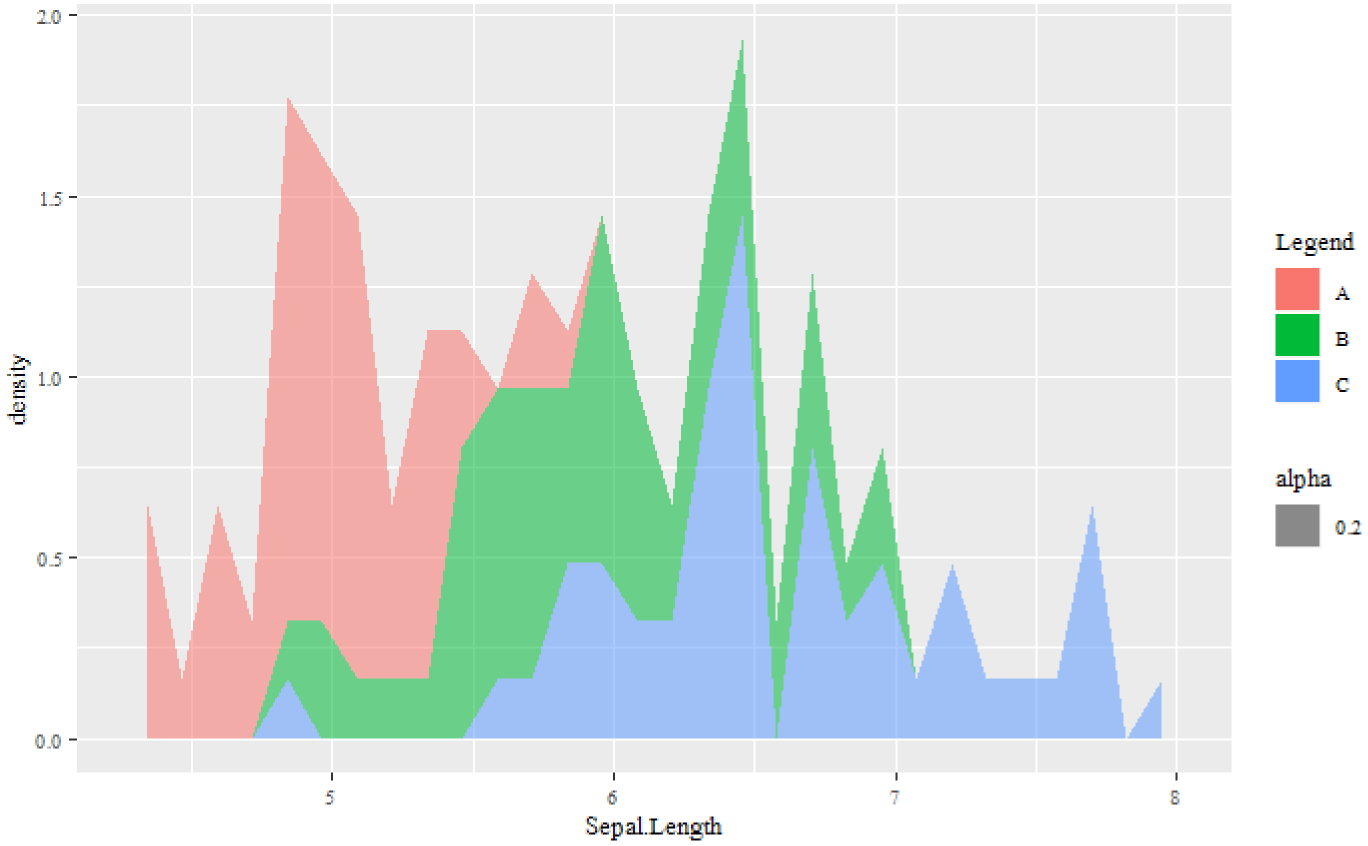
Hide

```
ggplot(iris,aes(x=Sepal.Length,fill=Species,color=Species,alpha=0.2))+
geom_area(aes(y = ..density..), stat ="bin")+
guides(color="colorbar") #colorbar guide needs continuous scales.
```



Hide

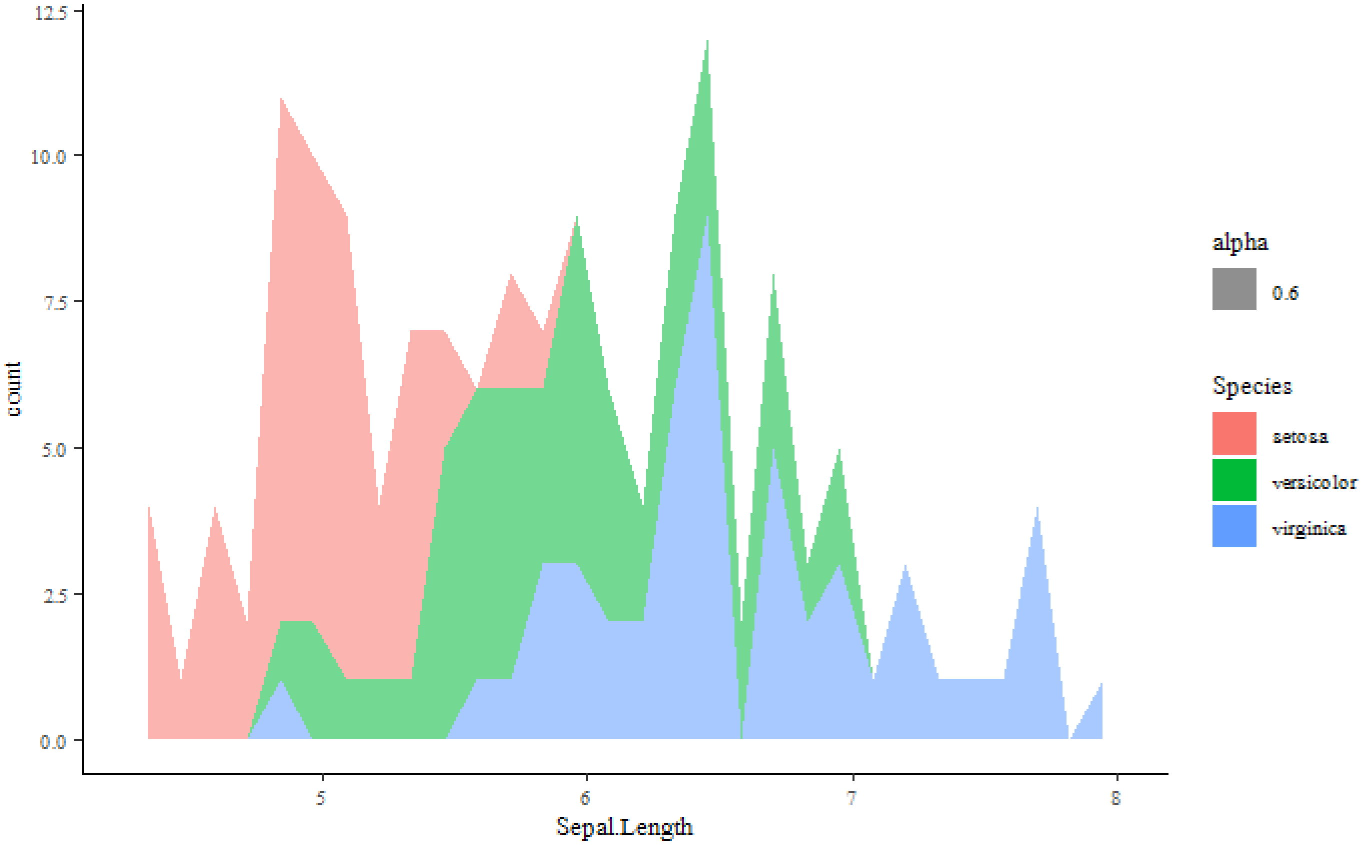
```
#Set legend title and labels with a scale function.
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.2))+
geom_area(aes(y = ..density..), stat ="bin")+
scale_fill_discrete(name = "Legend",labels = c("A", "B", "C"))
```



Themes:- Use themes after geom

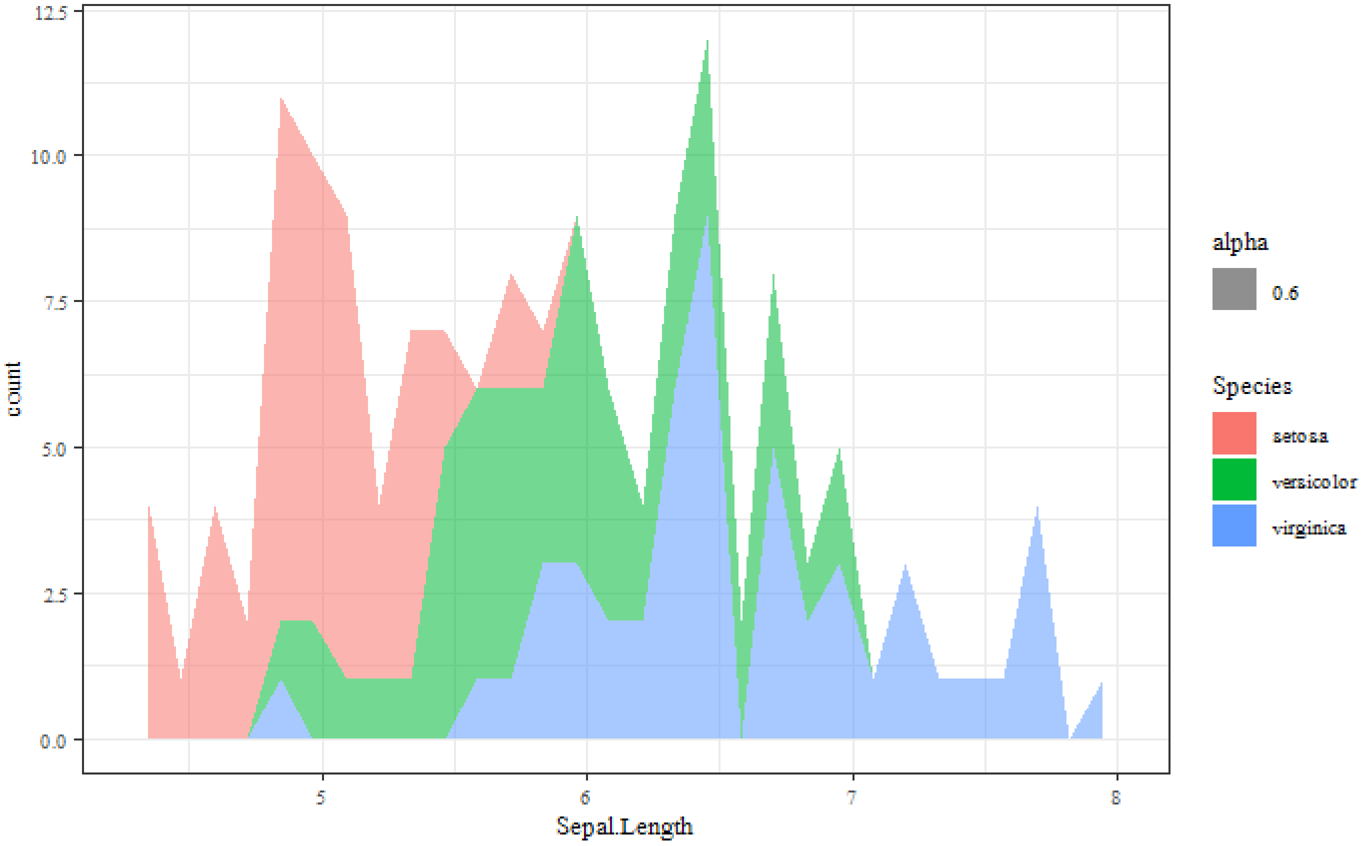
Hide

```
#White background with no gridlines
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
theme_classic()
```



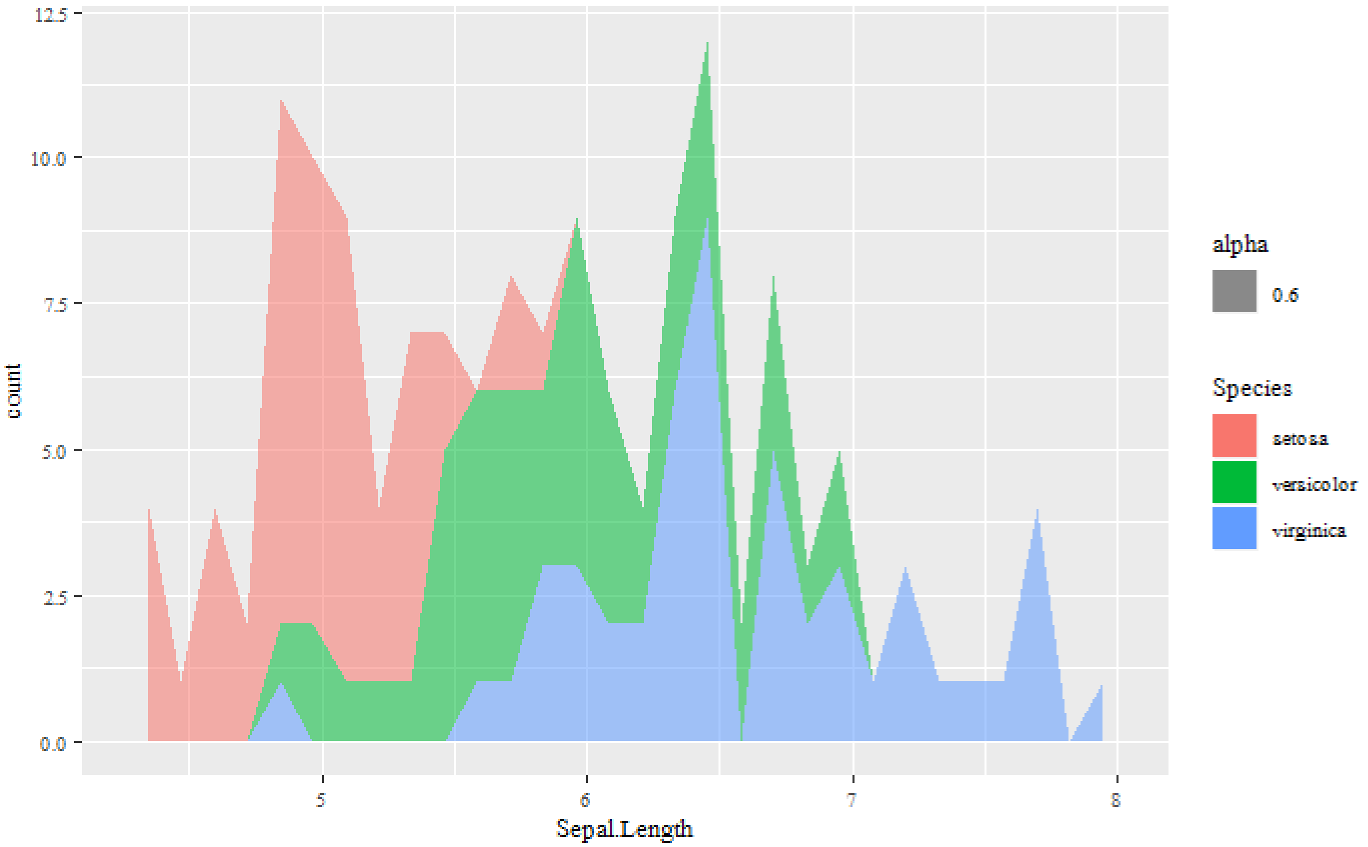
Hide

```
#White background with gridlines
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
theme_bw()
```



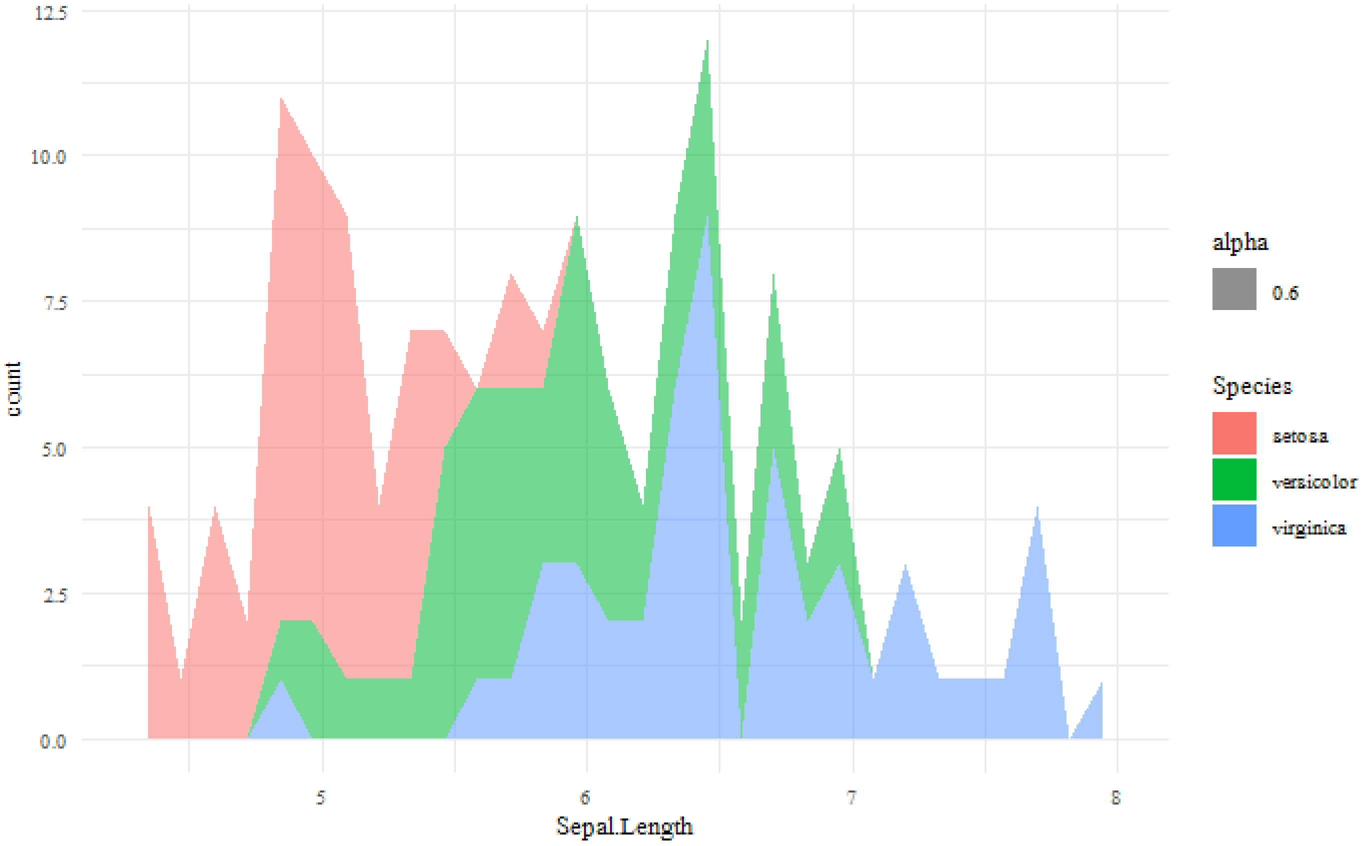
Hide

```
#Grey background(Default Theme)
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
theme_grey()
```



Hide

```
#Minimal theme
ggplot(iris,aes(x=Sepal.Length,fill=Species,alpha=0.6))+
geom_area(stat="bin")+
theme_minimal()
```



Hide

```
#ggthemes - Package with additional ggplot2 themes
```

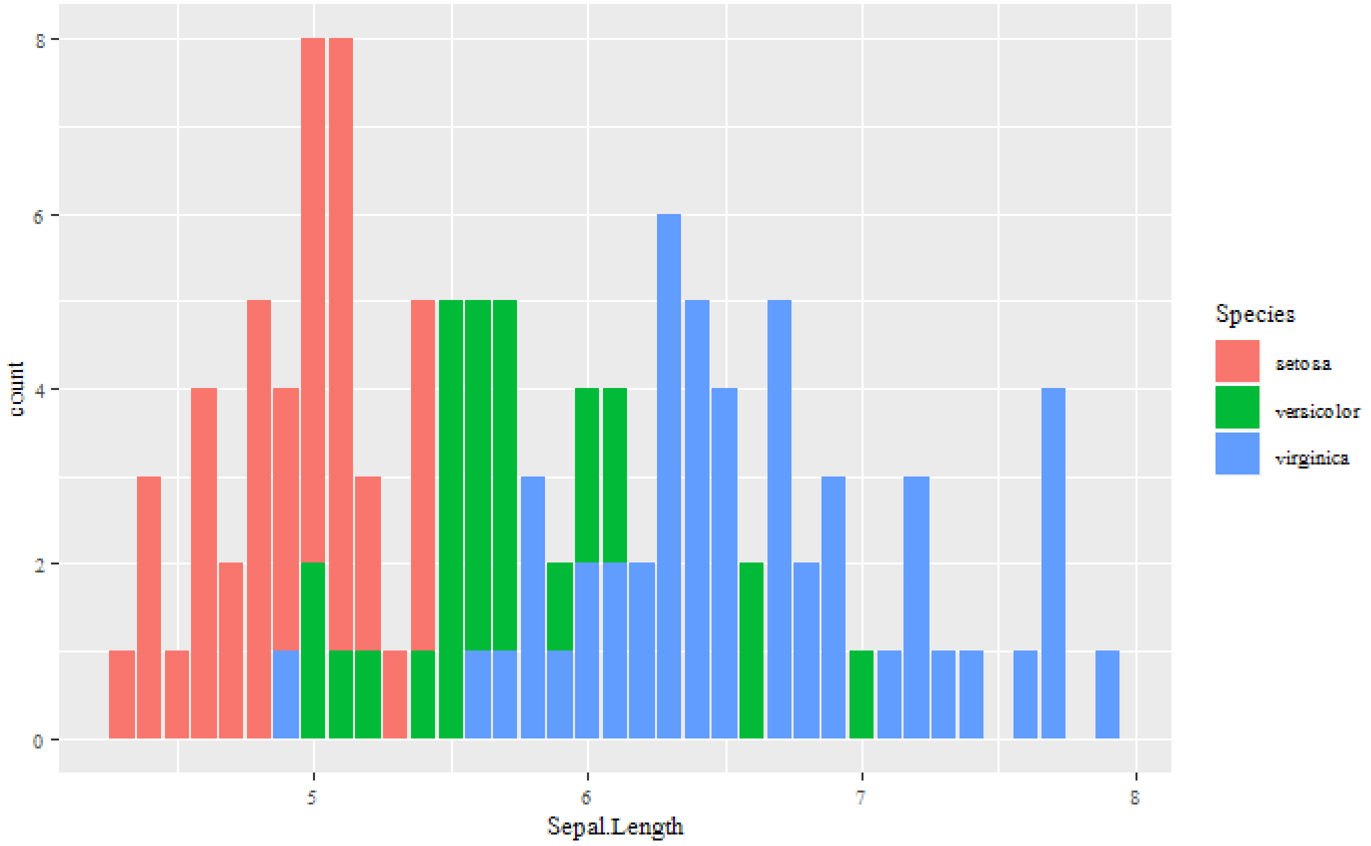
Position adjustments:- Give inside geom. Position adjustments determine how to arrange geoms that would otherwise occupy the same space.

Modifying Aesthetics

Positions ??? identity ??? dodge ??? stack ??? fill ??? jitter ??? jitterdodge

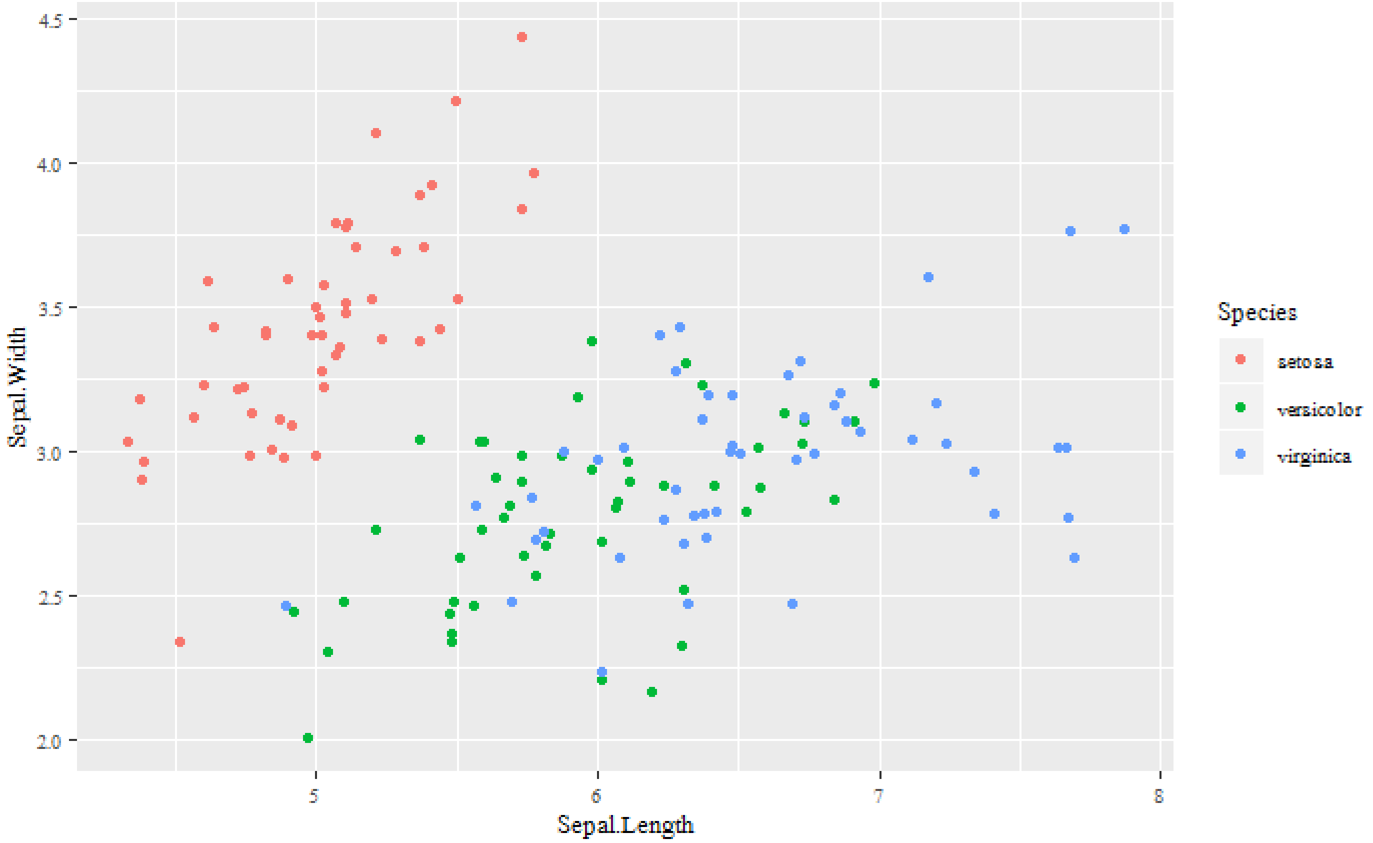
Hide

```
#position identity (default)
ggplot(iris, aes(x =Sepal.Length,fill=Species)) +
geom_bar(position="identity")
```



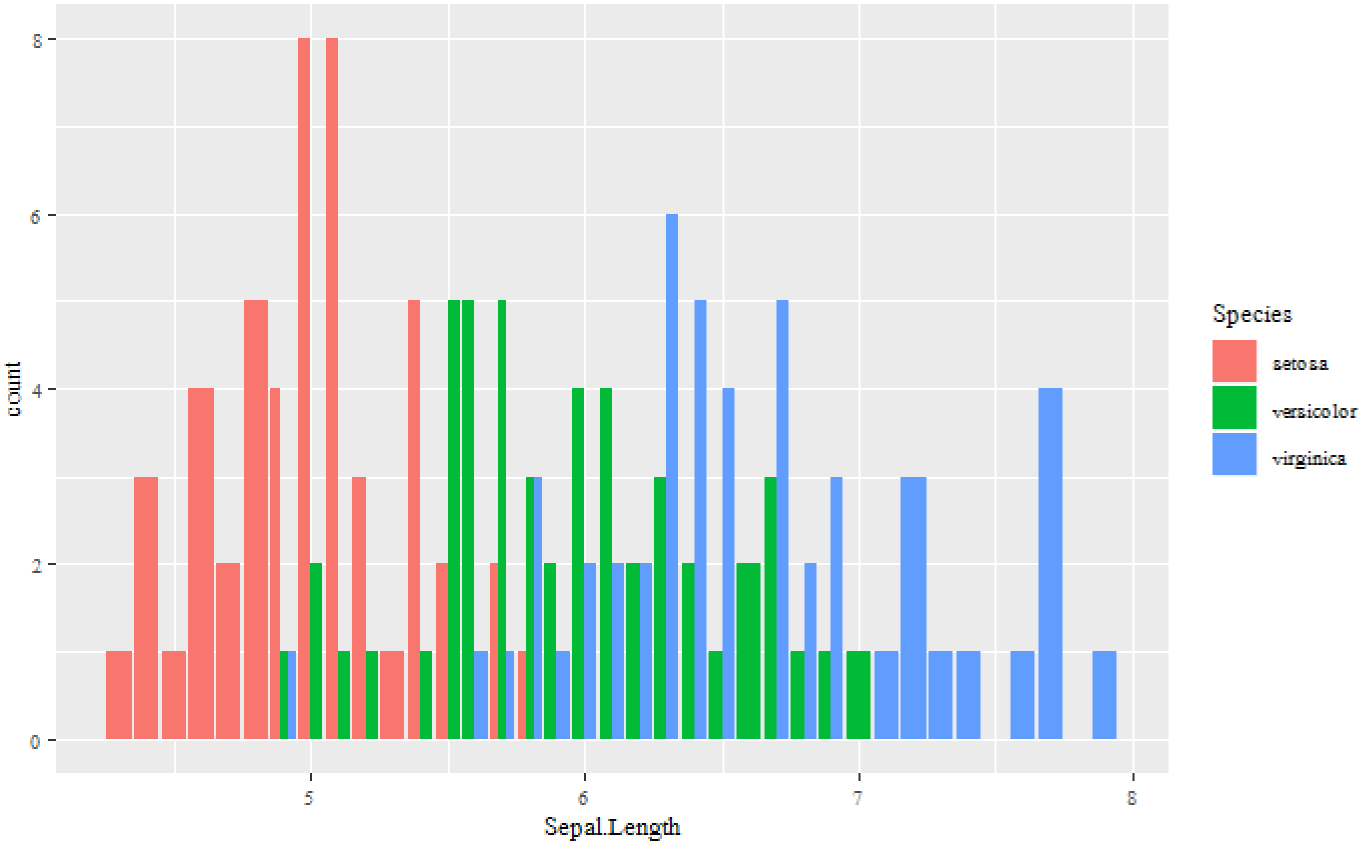
Hide

```
#Add random noise to X and Y position of each element to avoid overplotting
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
geom_point(position = "jitter")
```



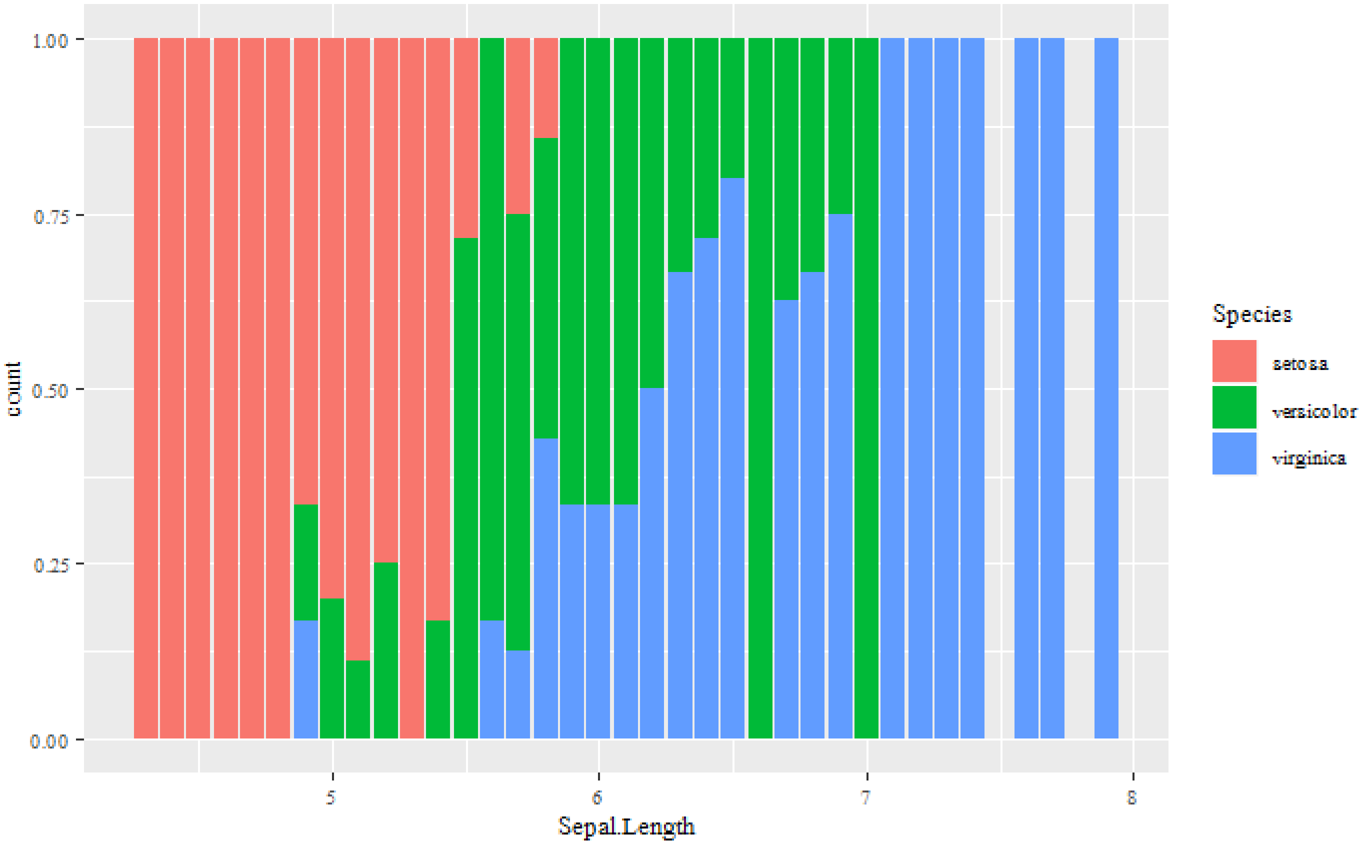
Hide

```
#Arrange elements side by side
ggplot(iris, aes(x = Sepal.Length,fill = Species)) +
geom_bar(position = "dodge")
```



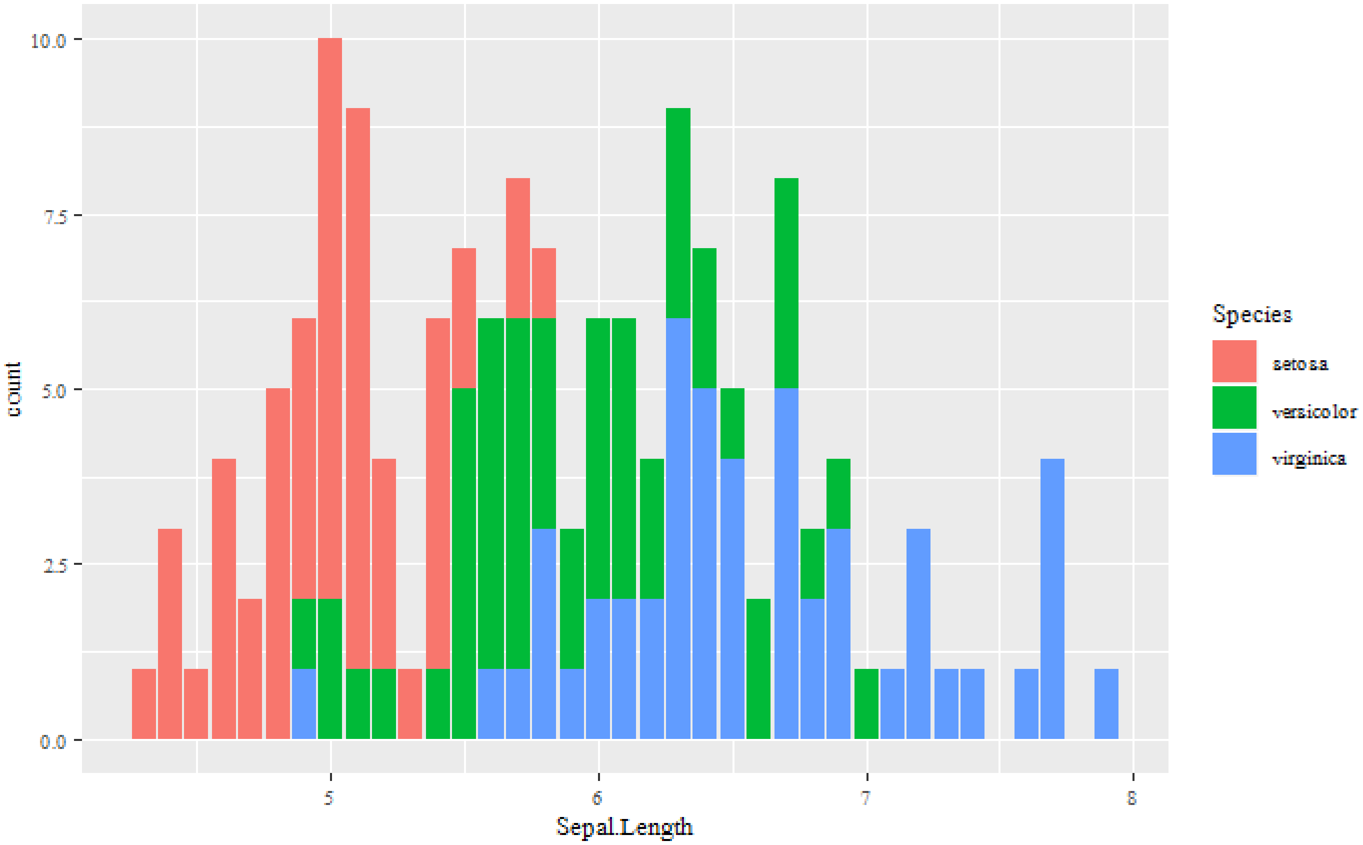
Hide

```
#Stack elements on top of one another, normalize height
ggplot(iris, aes(x = Sepal.Length,fill = Species)) +
geom_bar(position = "fill")
```



Hide

```
#Stack elements on top of one another
ggplot(iris, aes(x = Sepal.Length,fill = Species)) +
geom_bar(position = "stack")
```



Hide

```
#error : ggplot(iris, aes(x = Sepal.Length,y=Sepal.Width,col = Species)) +
#geom_point(position = "jitterdodge")
#Each position adjustment can be recast as a function with manual width and height arguments
#s + geom_bar(position = position_dodge(width = 1))
table(iris$Sepal.Length,iris$Species)
```

```
     setosa versicolor virginica
4.3       1          0         0
4.4       3          0         0
4.5       1          0         0
4.6       4          0         0
4.7       2          0         0
4.8       5          0         0
4.9       4          1         1
5         8          2         0
5.1       8          1         0
5.2       3          1         0
5.3       1          0         0
5.4       5          1         0
5.5       2          5         0
5.6       0          5         1
5.7       2          5         1
5.8       1          3         3
5.9       0          2         1
6         0          4         2
6.1       0          4         2
6.2       0          2         2
6.3       0          3         6
6.4       0          2         5
6.5       0          1         4
6.6       0          2         0
6.7       0          3         5
6.8       0          1         2
6.9       0          1         3
7         0          1         0
7.1       0          0         1
7.2       0          0         3
7.3       0          0         1
7.4       0          0         1
7.6       0          0         1
7.7       0          0         4
7.9       0          0         1
```