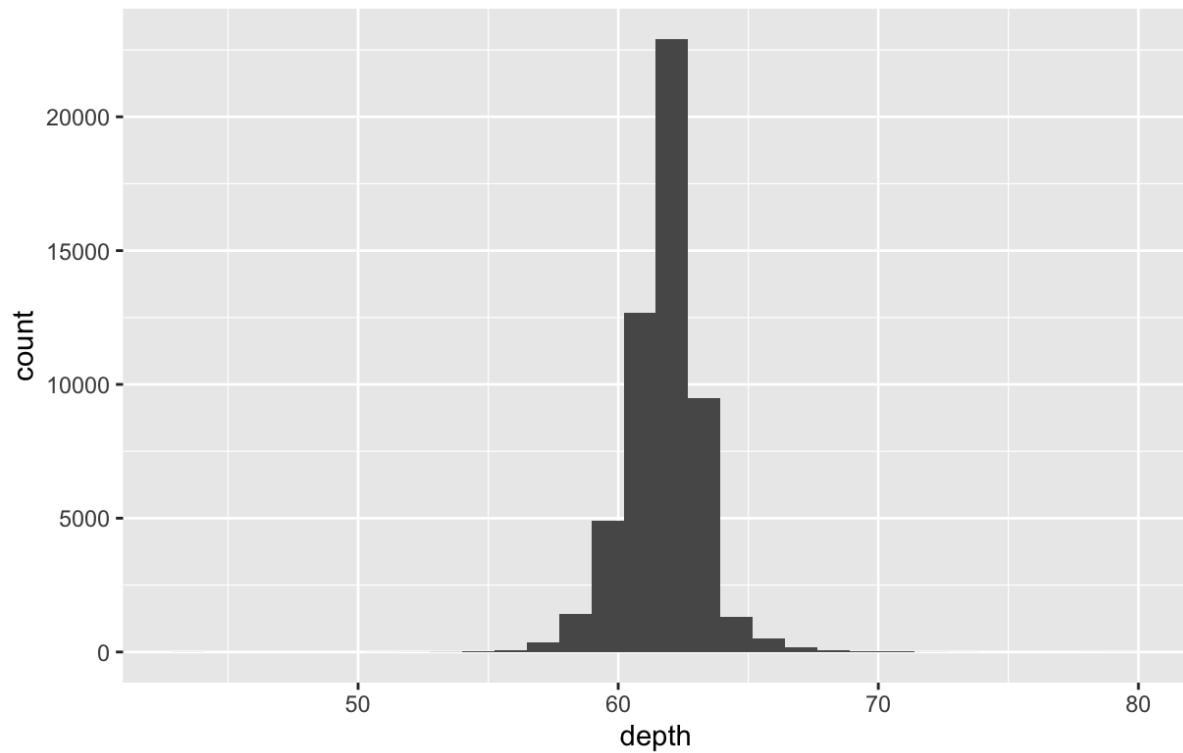
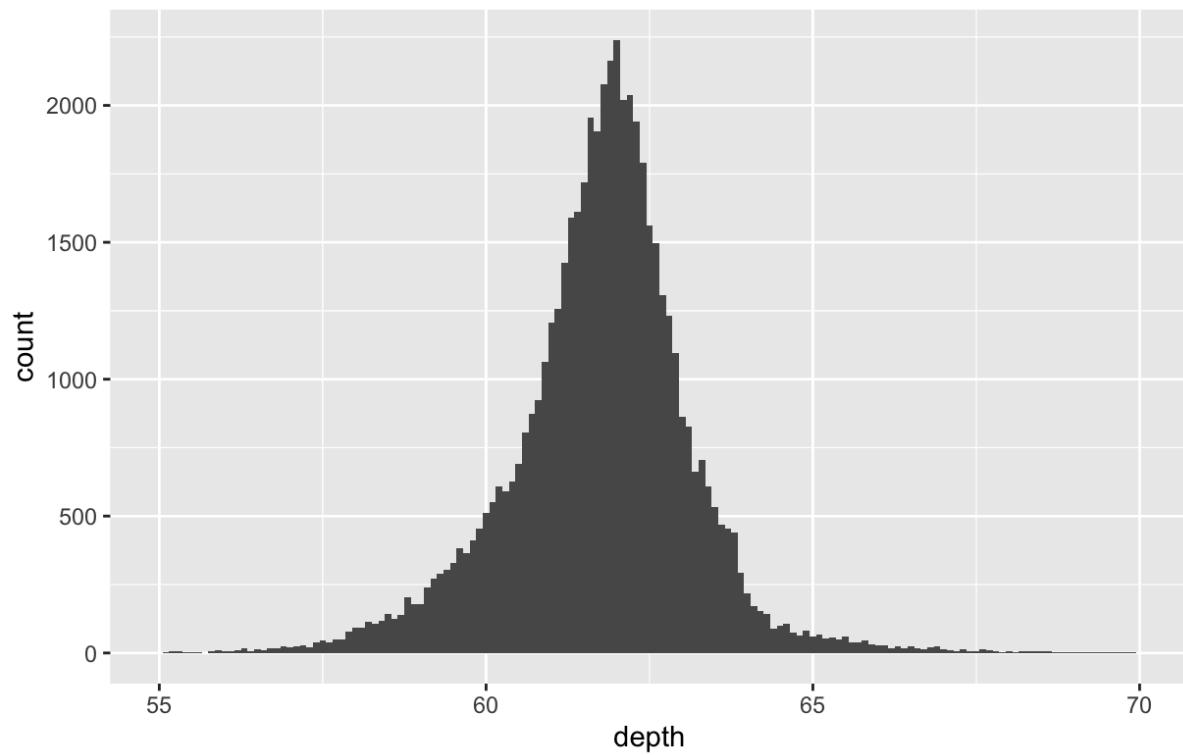


```
library(ggplot2)  
ggplot(diamonds, aes(depth)) +  
  geom_histogram()
```

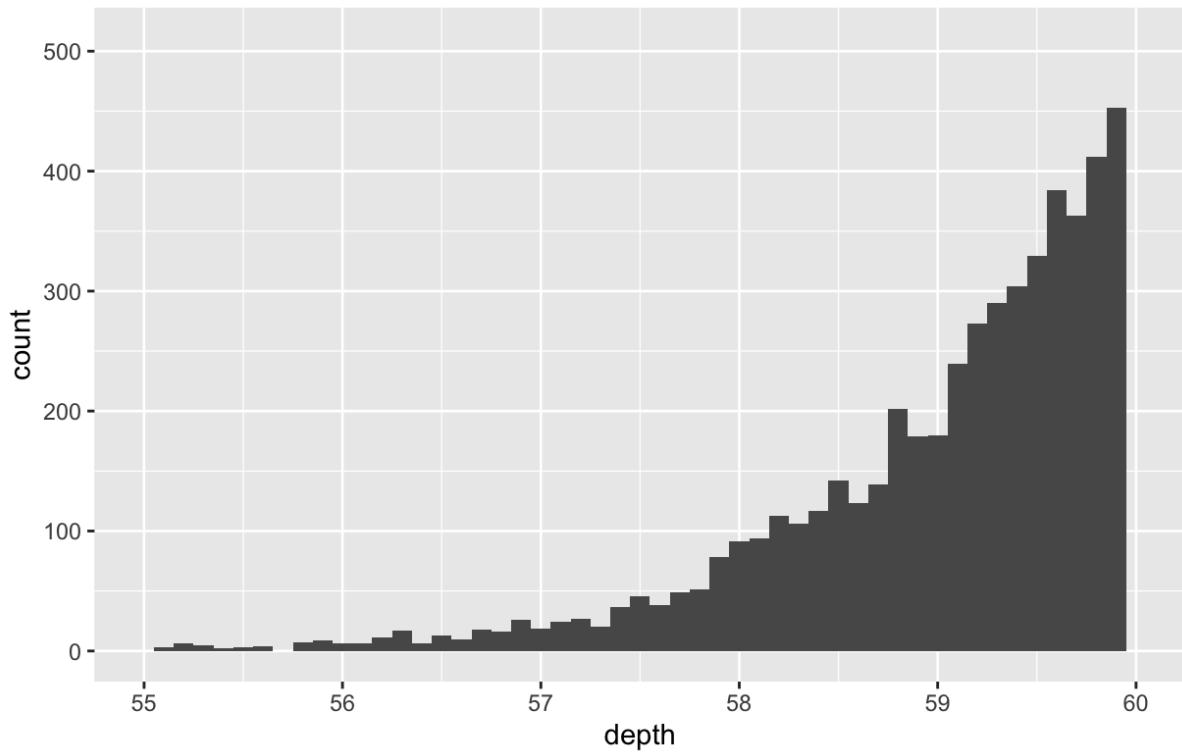


```
ggplot(diamonds, aes(depth)) +  
  geom_histogram(binwidth = 0.1) +  
  xlim(55, 70)
```

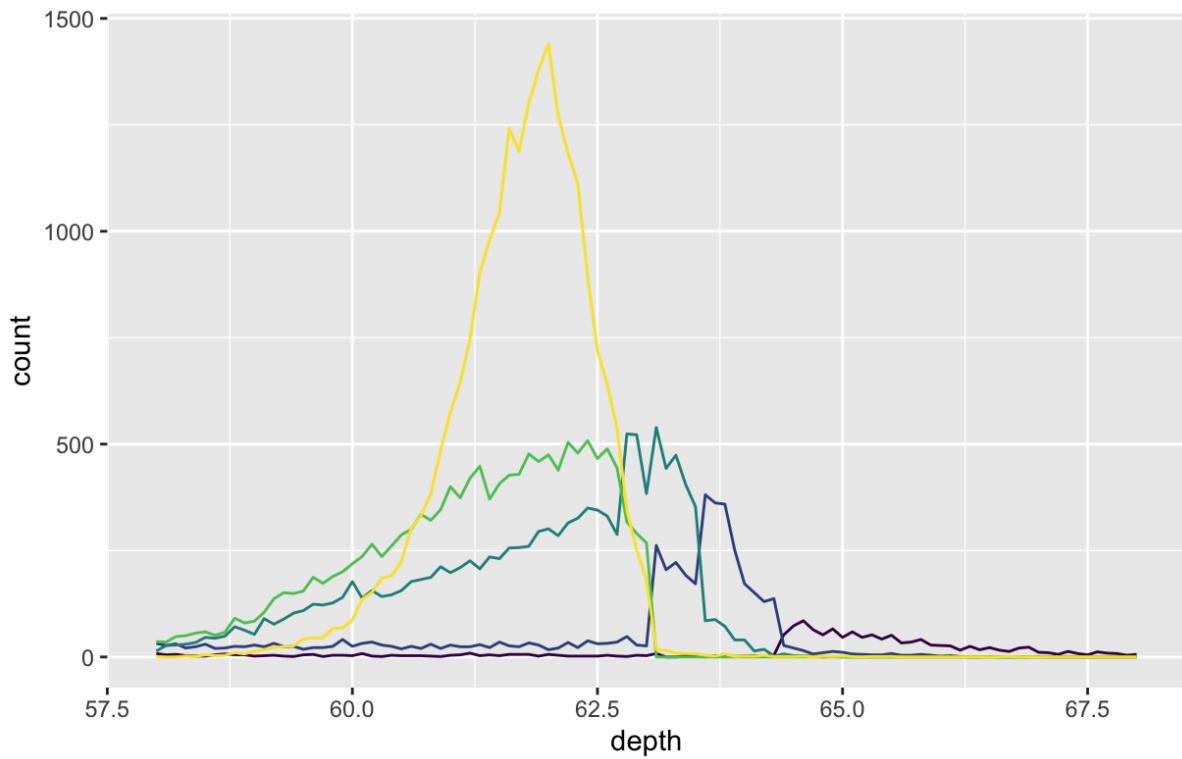


```
ggplot(diamonds, aes(depth)) +
```

```
geom_histogram(binwidth = 0.1) +  
  xlim(55, 60)
```

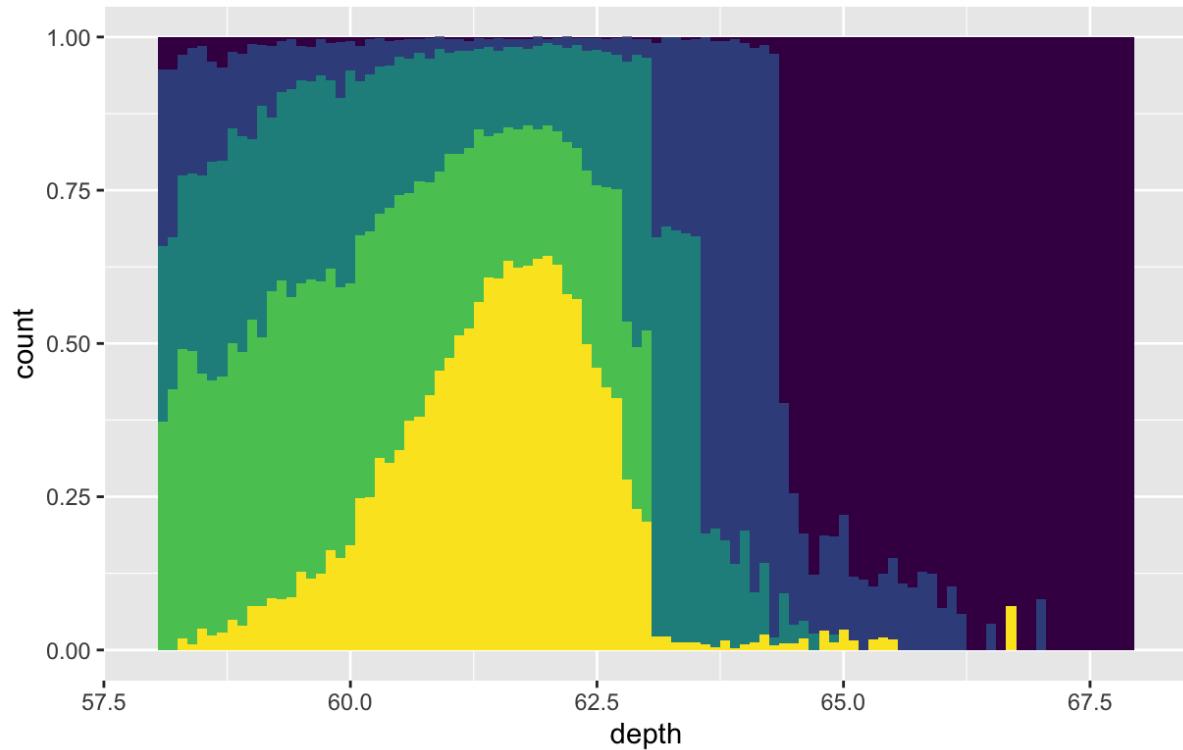


```
ggplot(diamonds, aes(depth)) +  
  geom_freqpoly(aes(colour = cut), binwidth = 0.1, na.rm = TRUE) +  
  xlim(58, 68) +  
  theme(legend.position = "none")
```

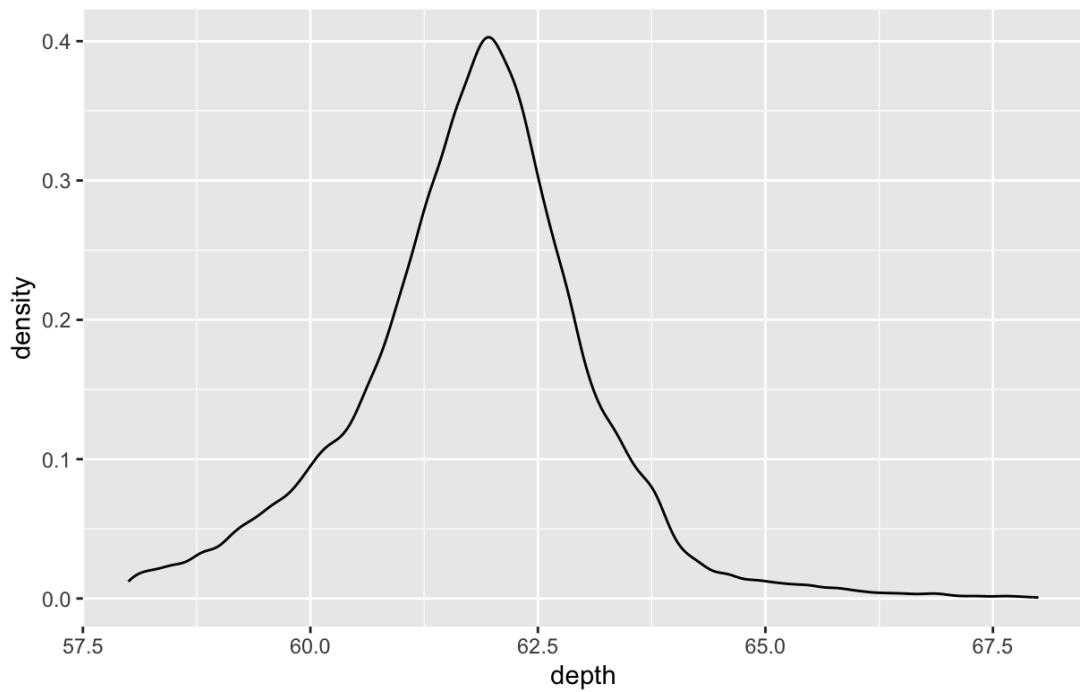


```
ggplot(diamonds, aes(depth)) +
```

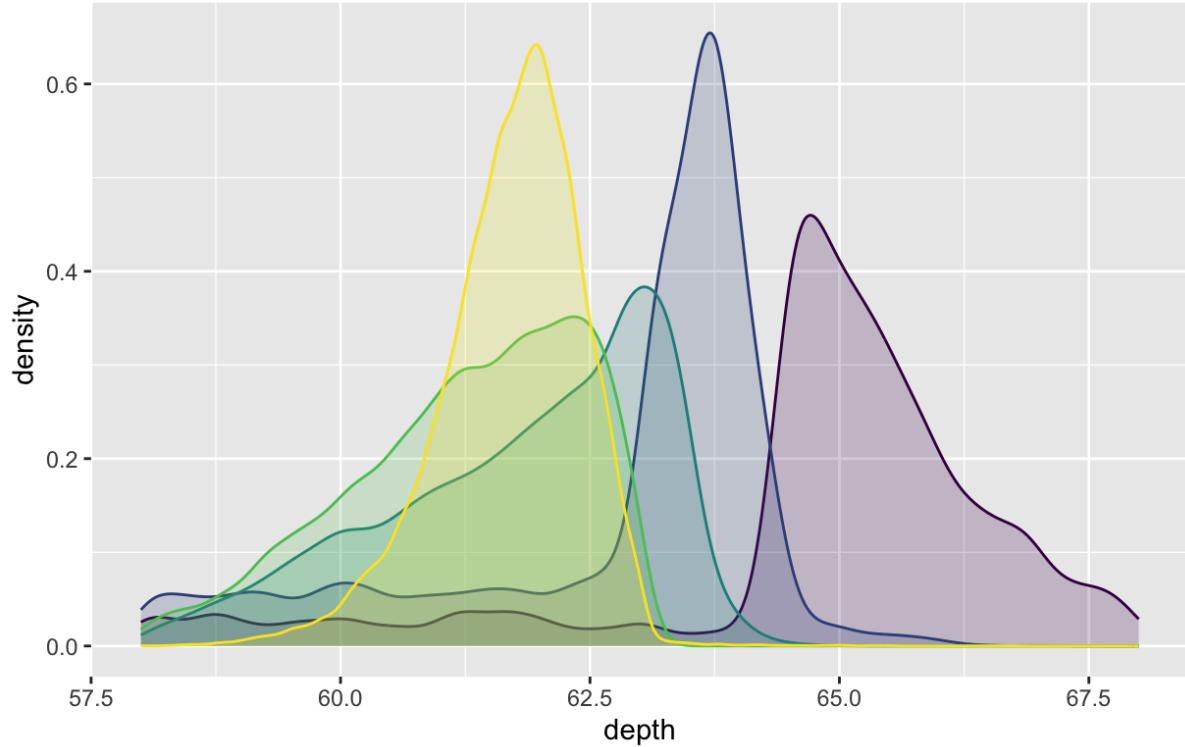
```
geom_histogram(aes(fill = cut), binwidth = 0.1, position = "fill",
               na.rm = TRUE) +
  xlim(58, 68) +
  theme(legend.position = "none")
```



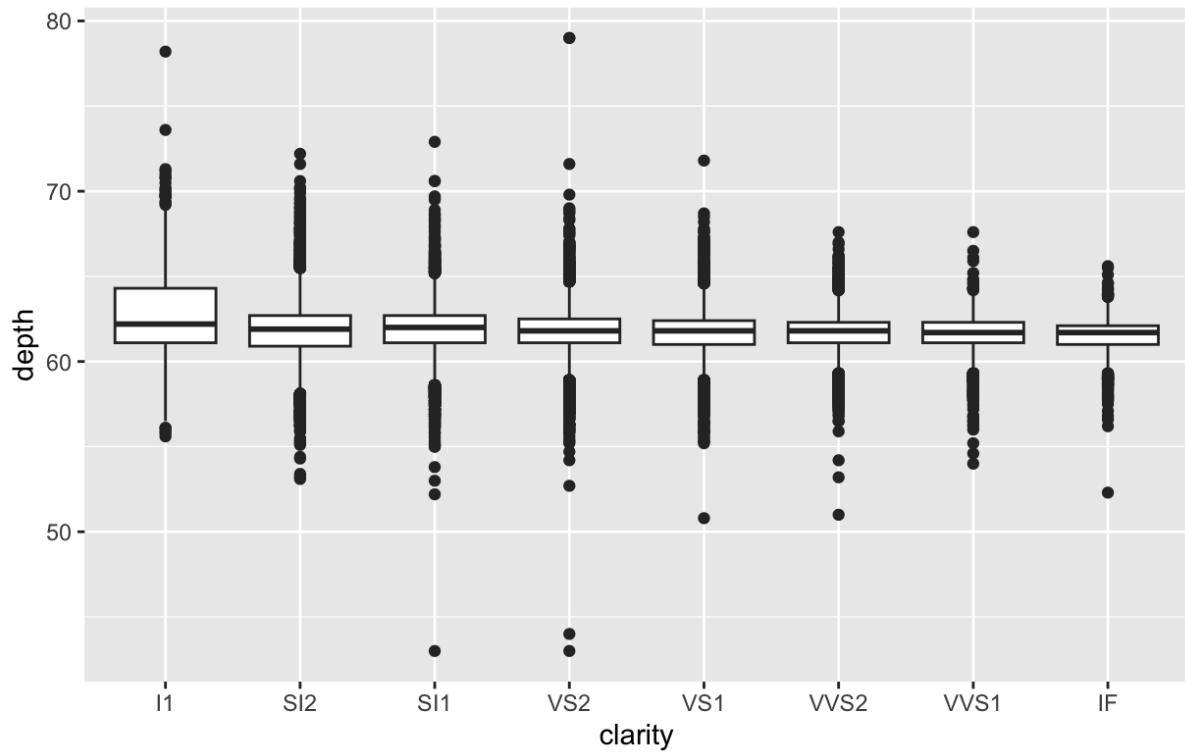
```
ggplot(diamonds, aes(depth)) +
  geom_density(na.rm = TRUE) +
  xlim(58, 68) +
  theme(legend.position = "none")
```



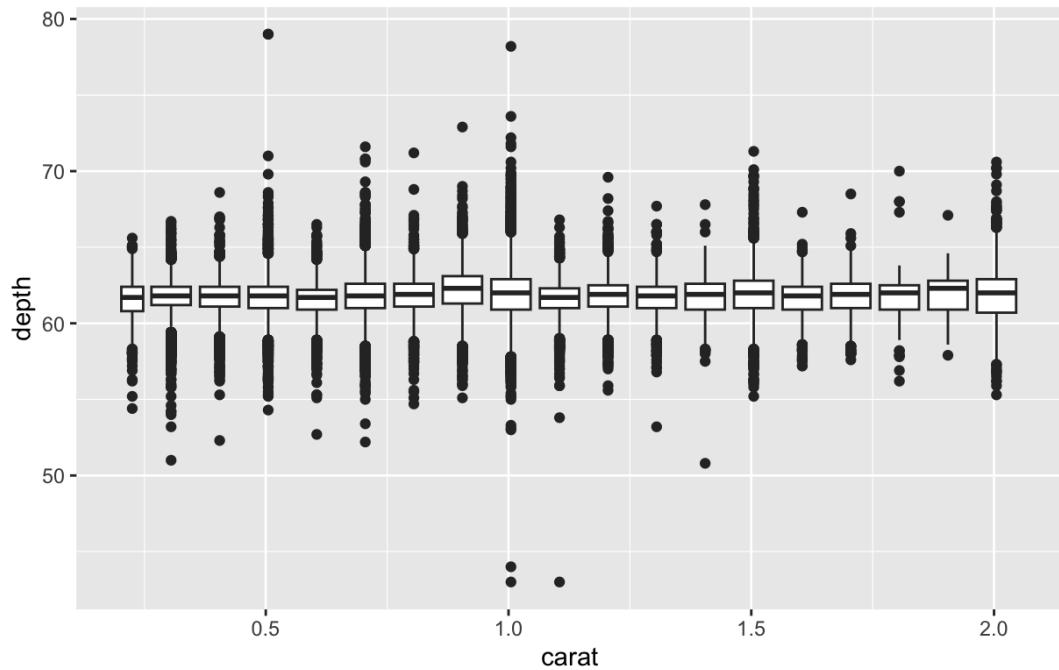
```
ggplot(diamonds,
aes(depth, fill = cut, colour = cut)) +
geom_density(alpha = 0.2, na.rm = TRUE) +
xlim(58, 68) +
theme(legend.position = "none")
```



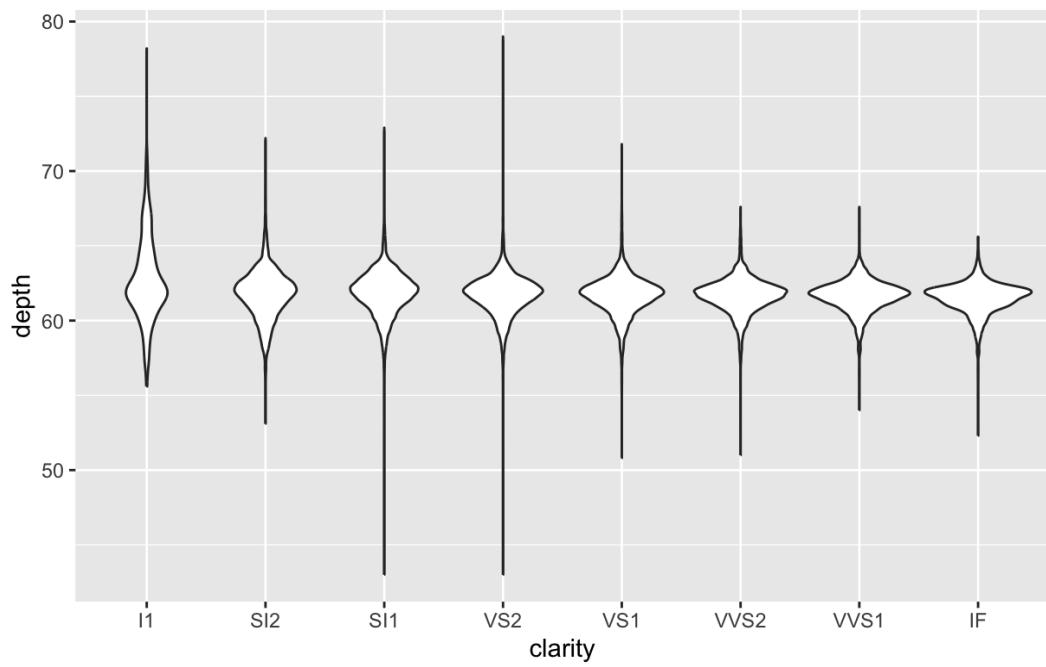
```
ggplot(diamonds, aes(clarity, depth)) +
geom_boxplot()
```



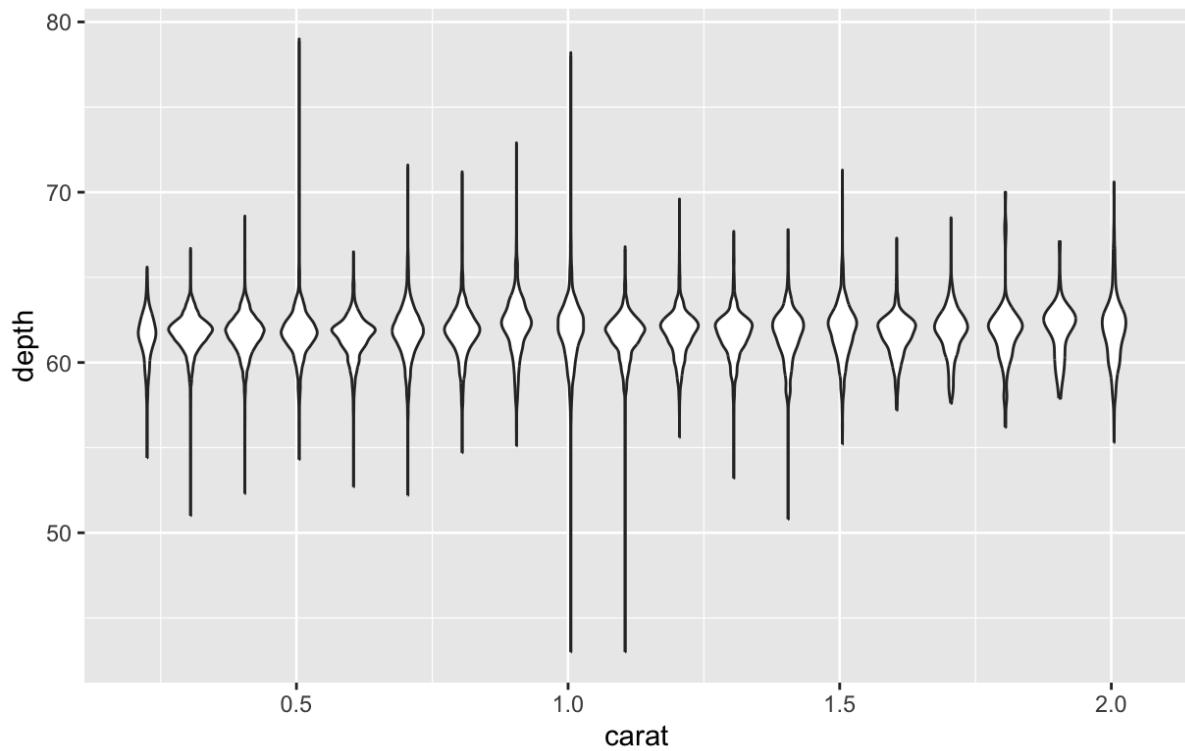
```
ggplot(diamonds, aes(carat, depth)) +
  geom_boxplot(aes(group = cut_width(carat, 0.1))) +
  xlim(NA, 2.05)
```



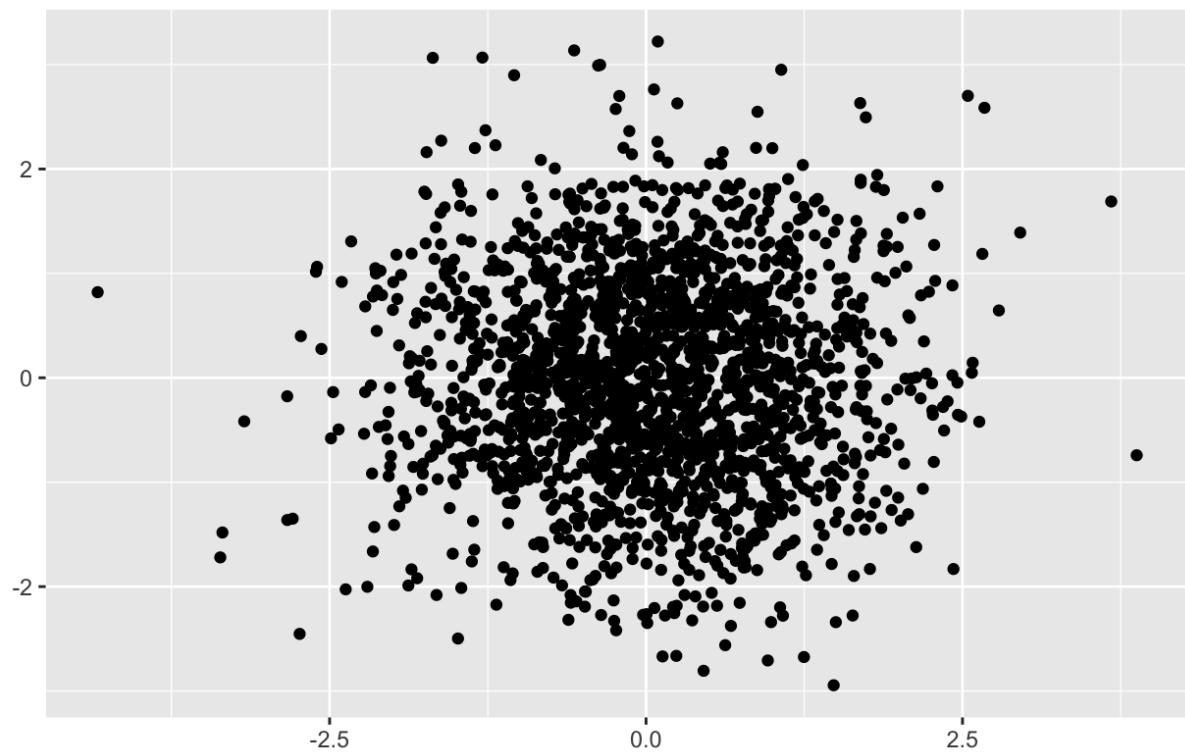
```
ggplot(diamonds, aes(clarity, depth)) +
  geom_violin()
```



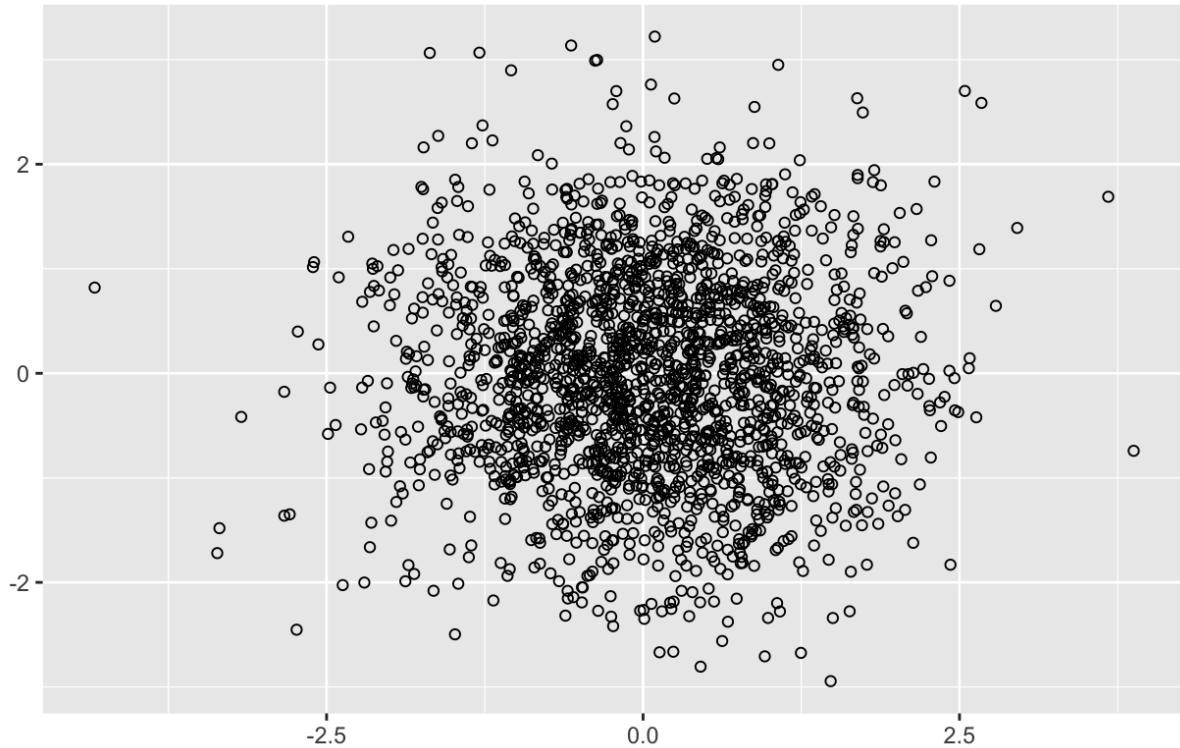
```
ggplot(diamonds, aes(carat, depth)) +  
  geom_violin(aes(group = cut_width(carat, 0.1))) +  
  xlim(NA, 2.05)
```



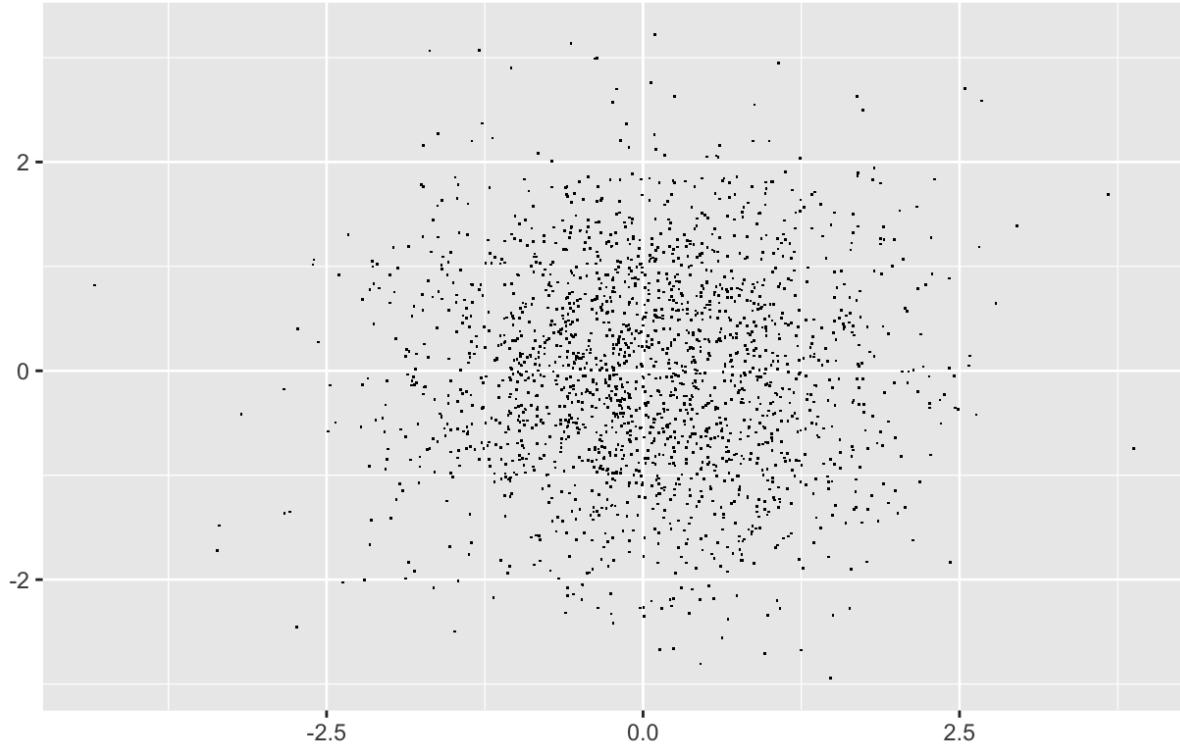
```
df <- data.frame(x = rnorm(2000), y = rnorm(2000))  
norm <- ggplot(df, aes(x, y)) + xlab(NULL) + ylab(NULL)  
norm + geom_point()
```



```
norm + geom_point(shape = 1) # Hollow circles
```

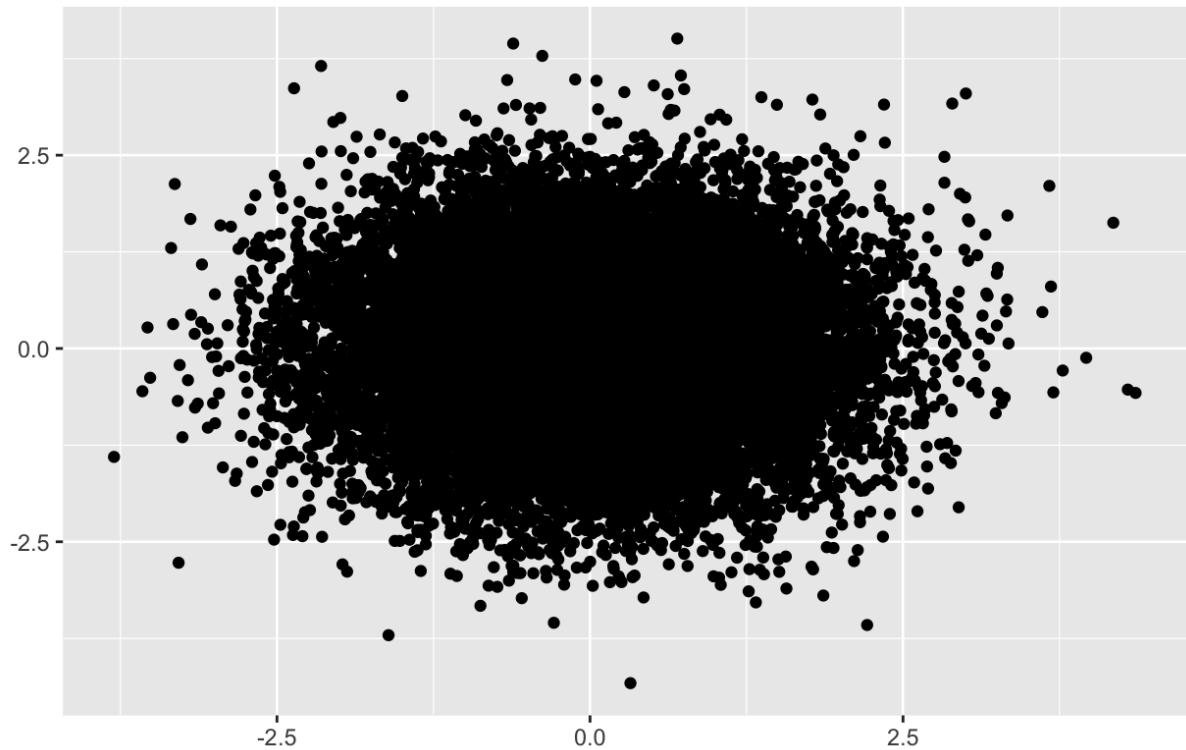


```
norm + geom_point(shape = ".") # Pixel sized
```

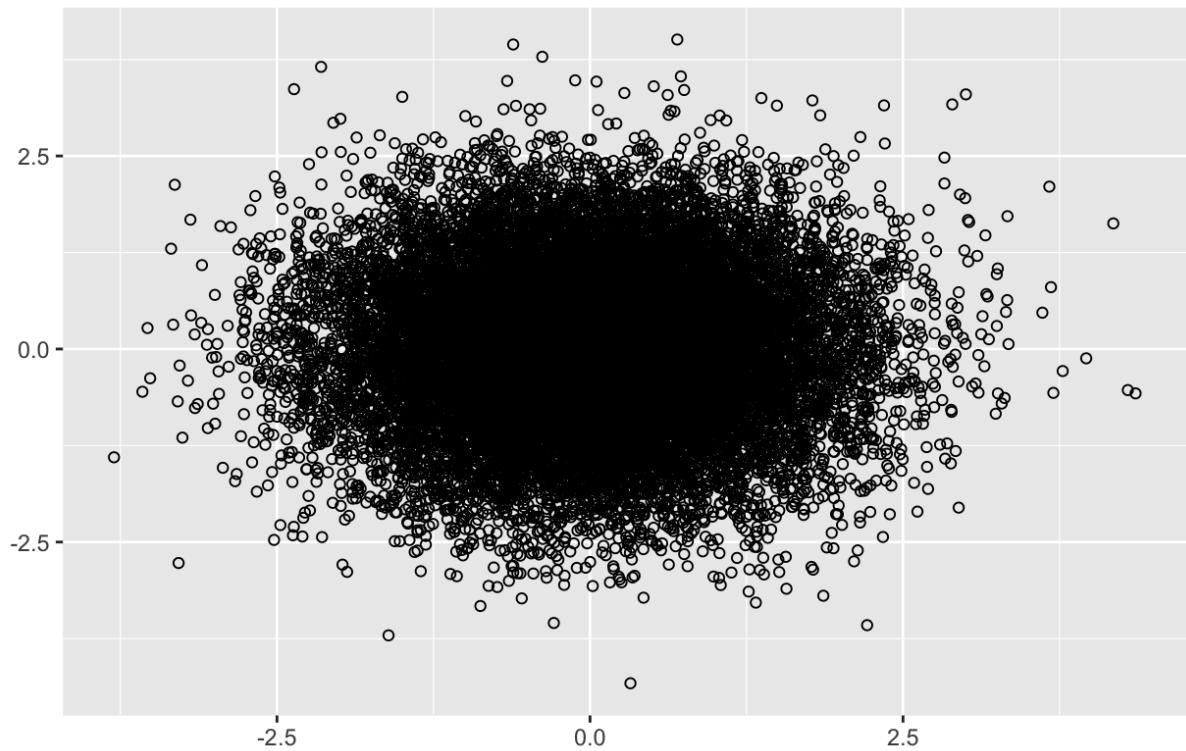


```
df <- data.frame(x = rnorm(20000), y = rnorm(20000))  
norm <- ggplot(df, aes(x, y)) + xlab(NULL) + ylab(NULL)
```

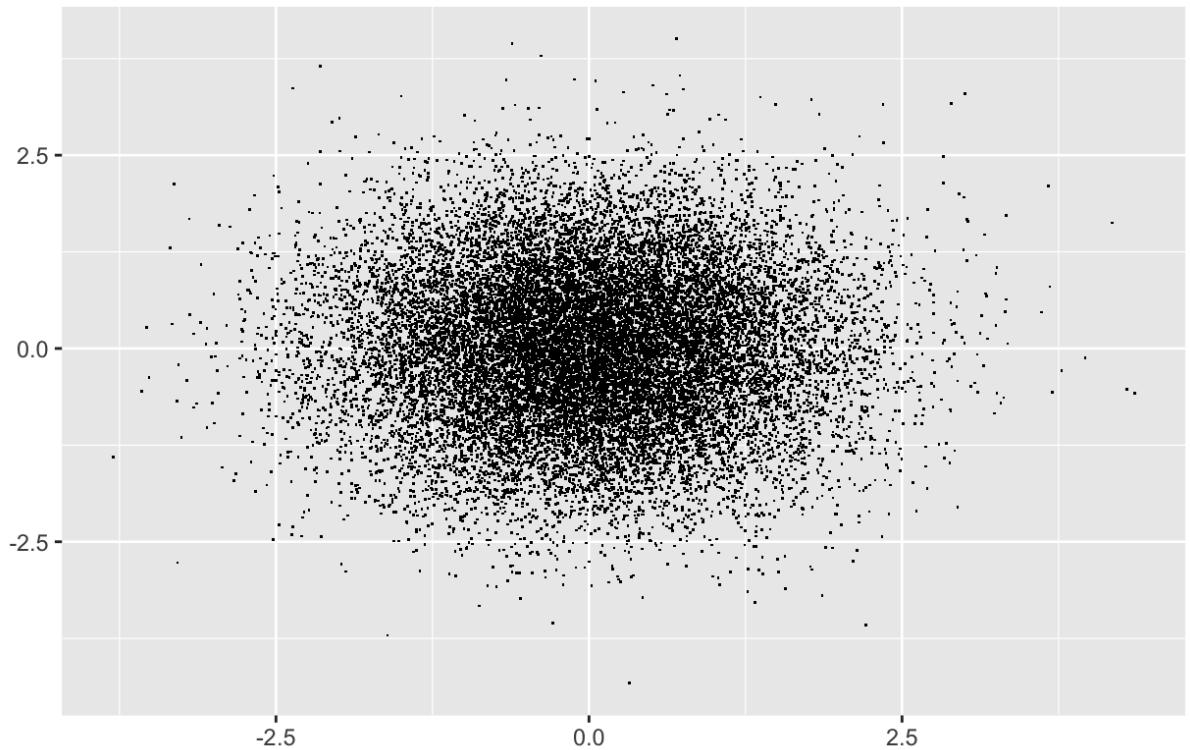
norm + geom\_point()



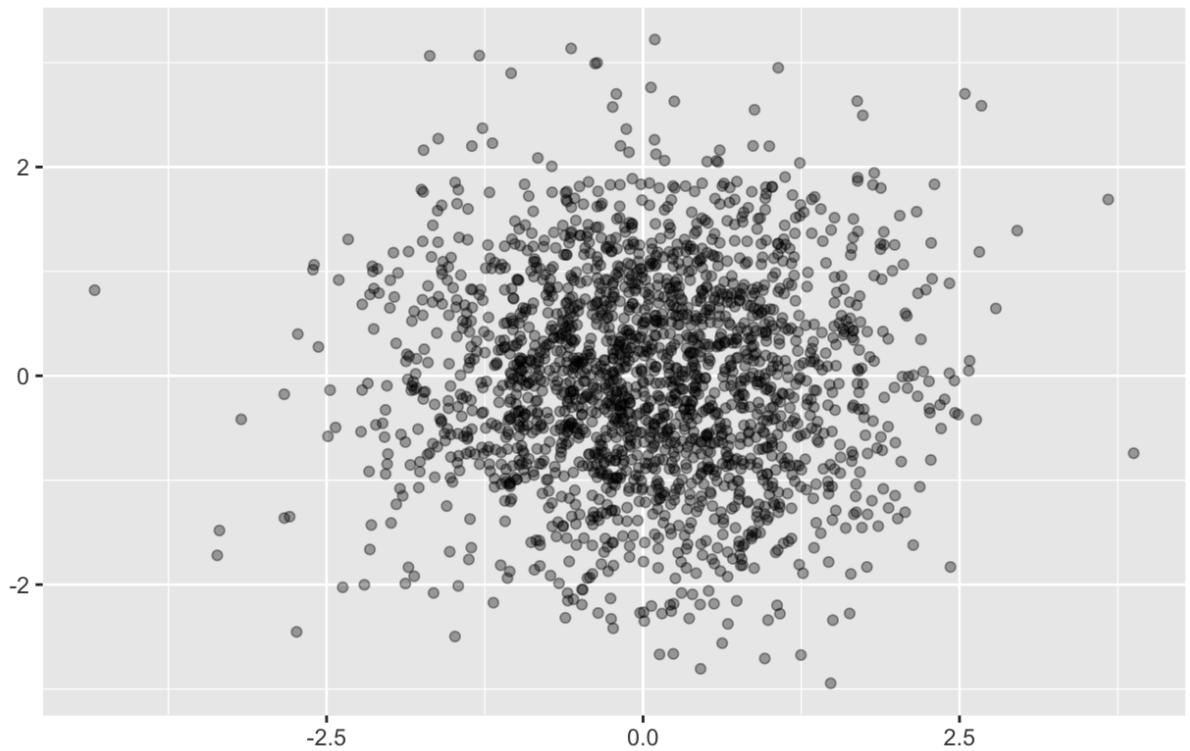
norm + geom\_point(shape = 1) # Hollow circles



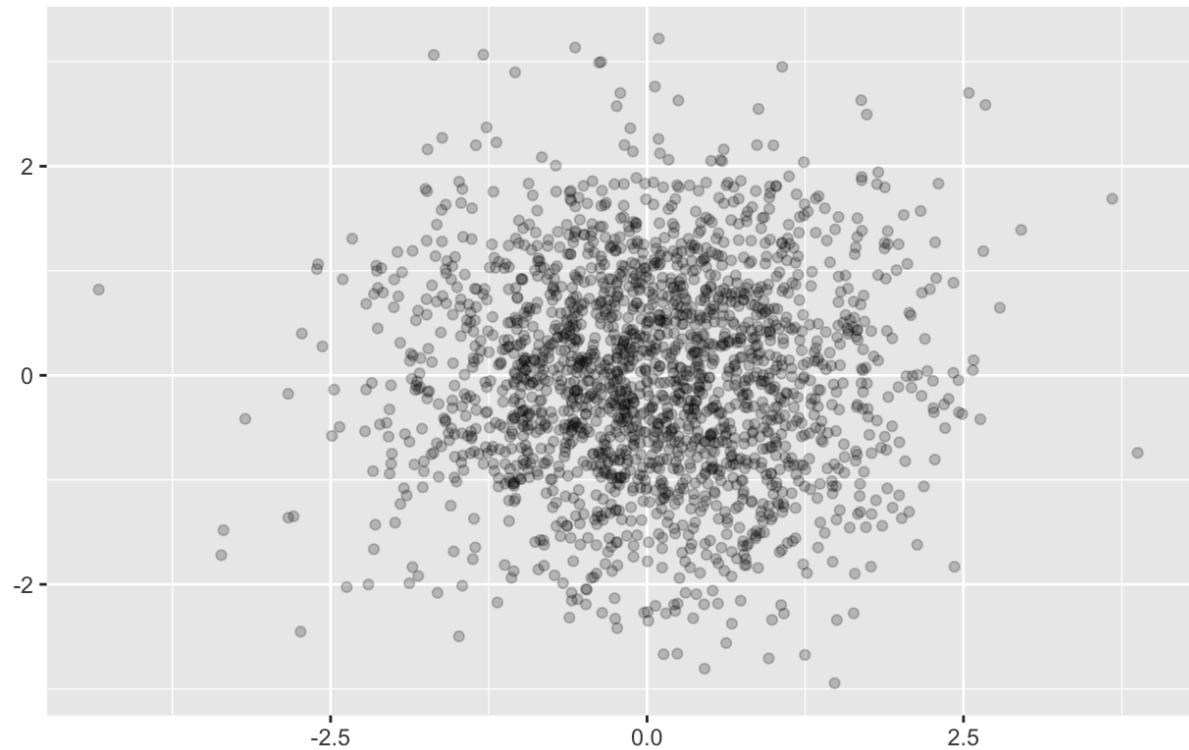
```
norm + geom_point(shape = ".") # Pixel sized
```



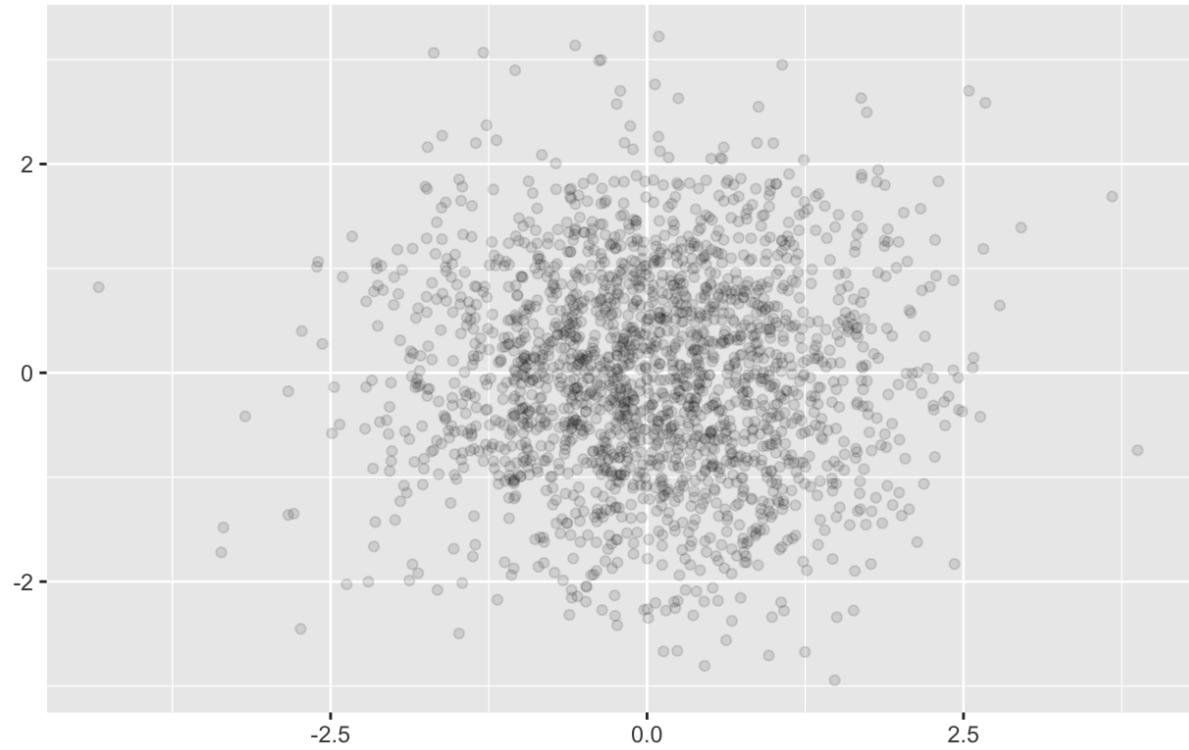
```
norm + geom_point(alpha = 1 / 3)
```



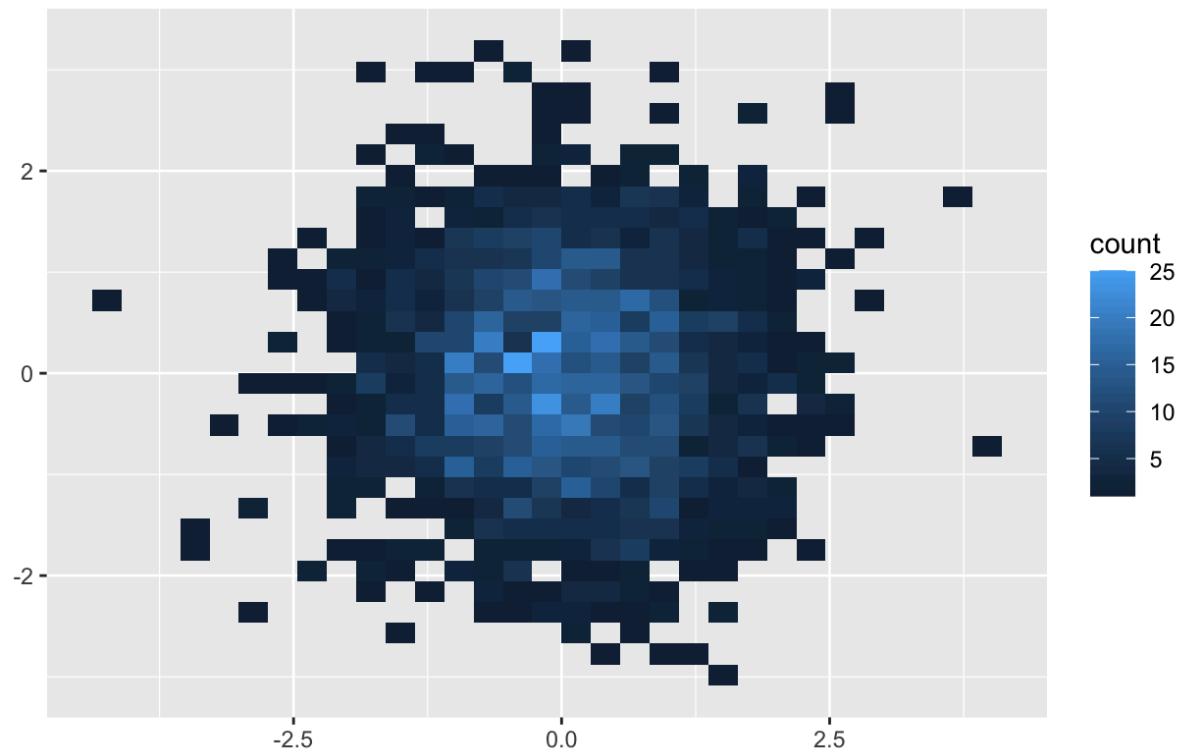
norm + geom\_point(alpha = 1 / 5)



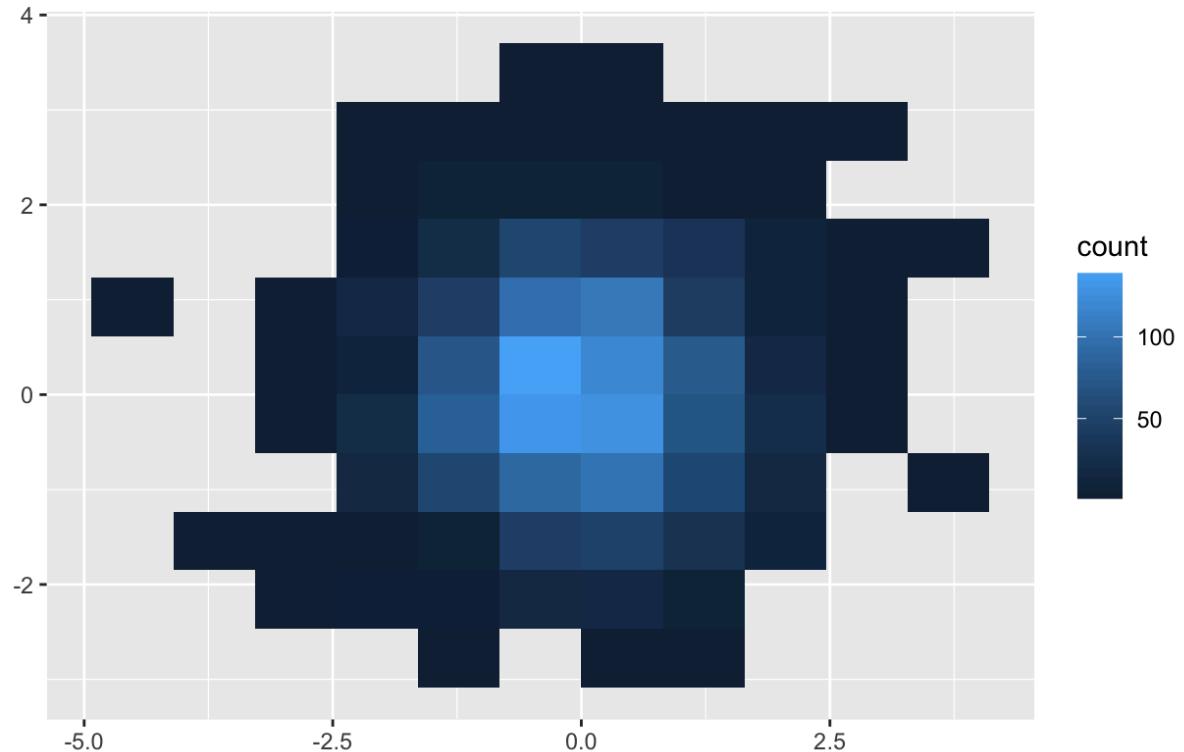
norm + geom\_point(alpha = 1 / 10)



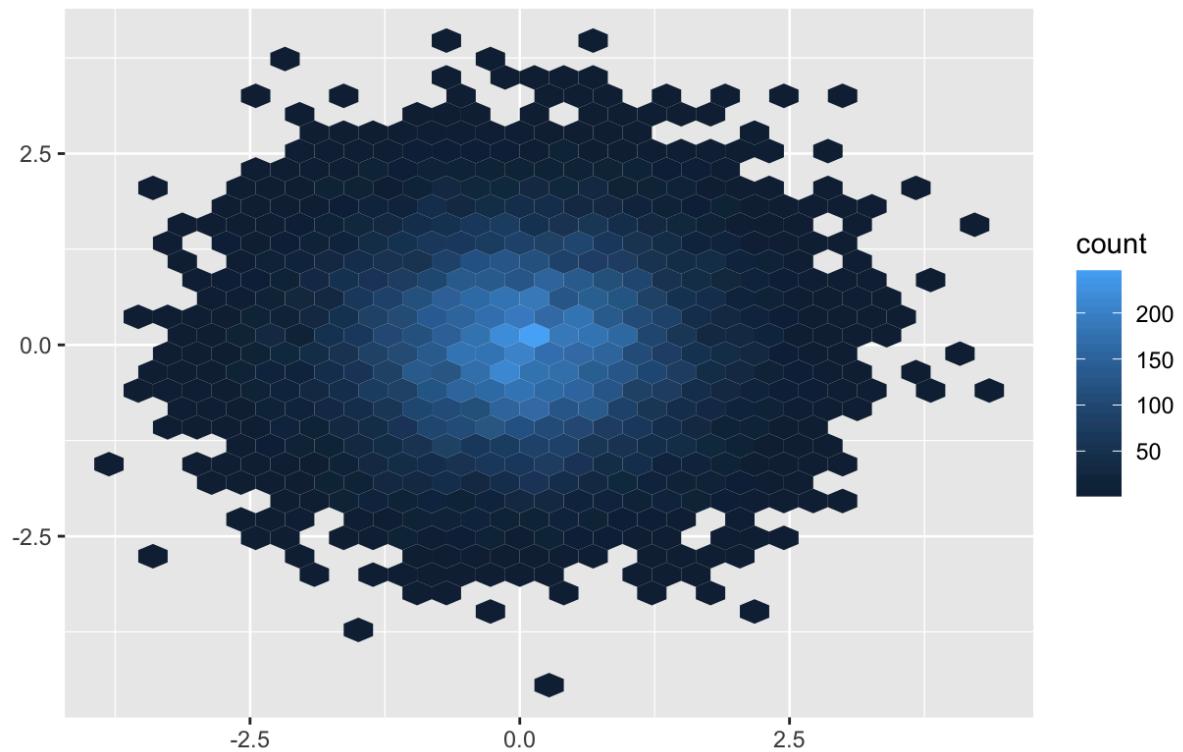
norm + geom\_bin2d()



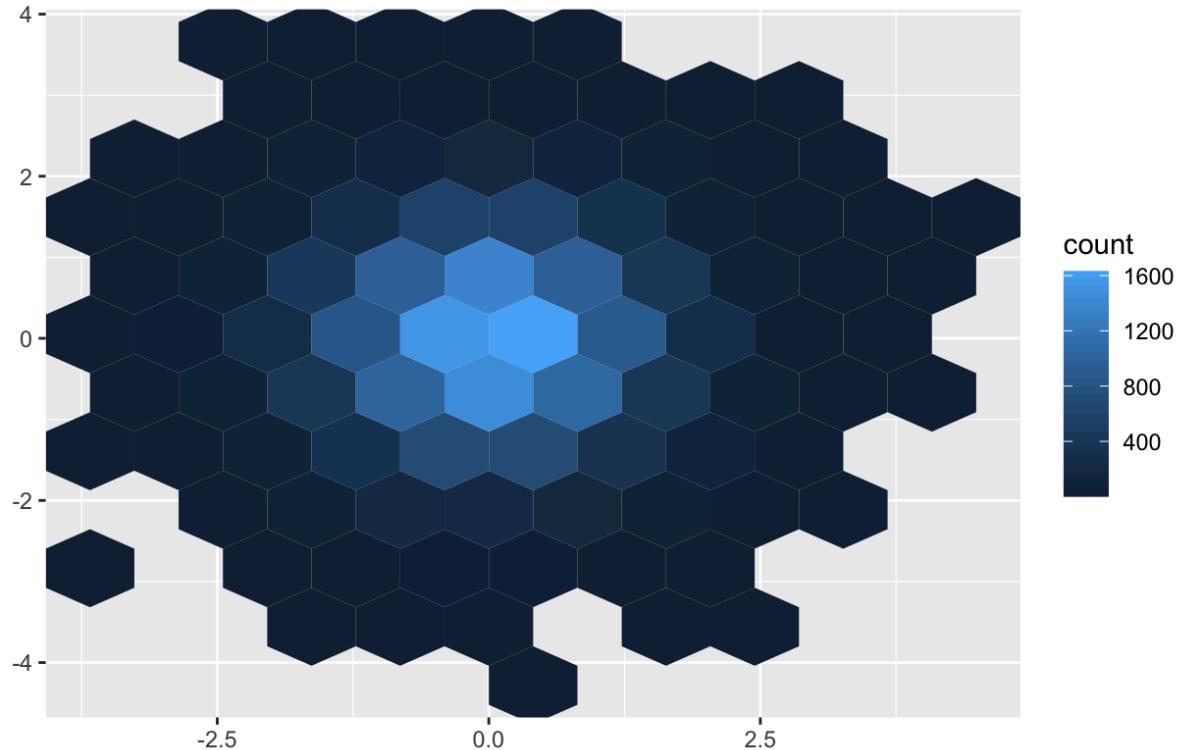
norm + geom\_bin2d(bins = 10)



norm + geom\_hex()



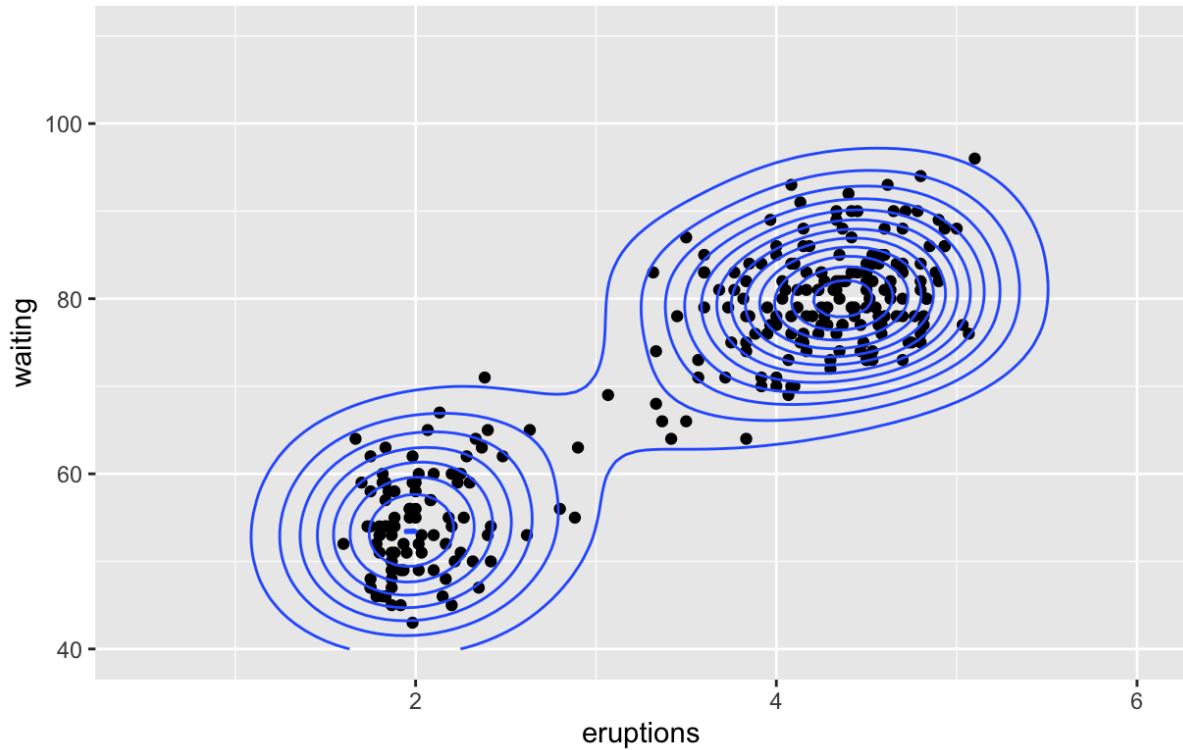
norm + geom\_hex(bins = 10)



```
library("ggplot2")
m <- ggplot(faithful, aes(x = eruptions, y = waiting)) +
  geom_point() +
  xlim(0.5, 6) +
  ylim(40, 110)
```

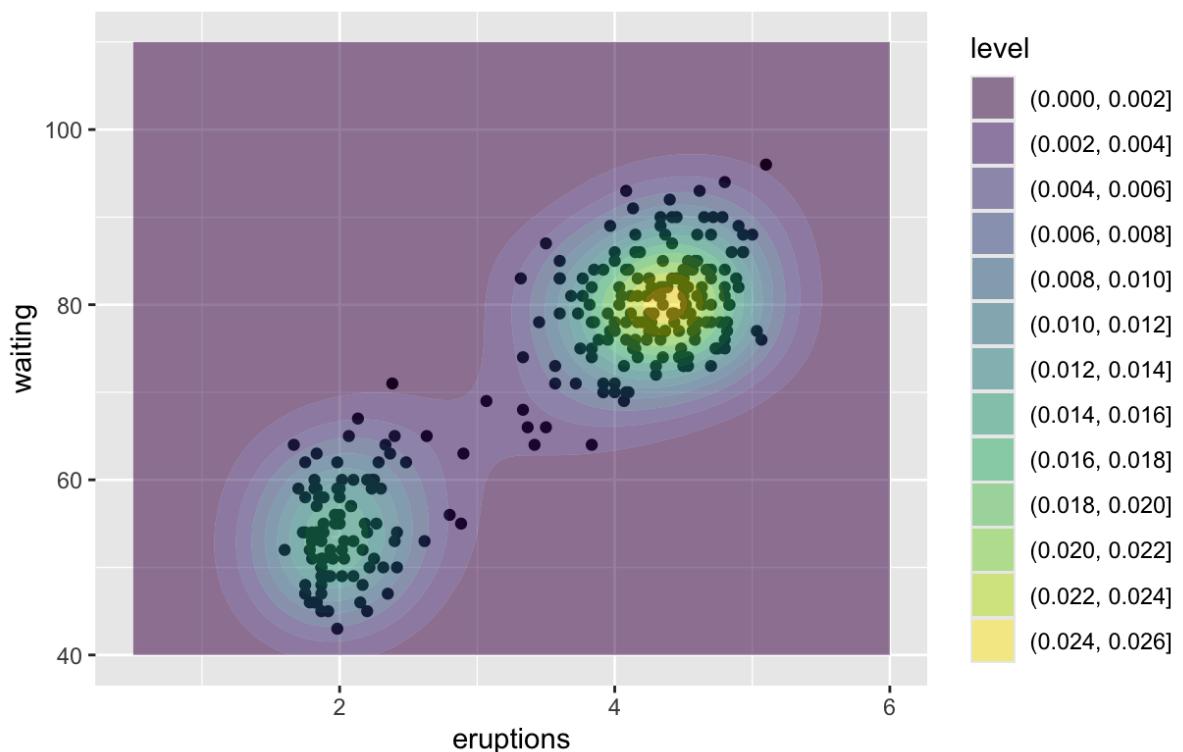
# contour lines

m + geom\_density\_2d()



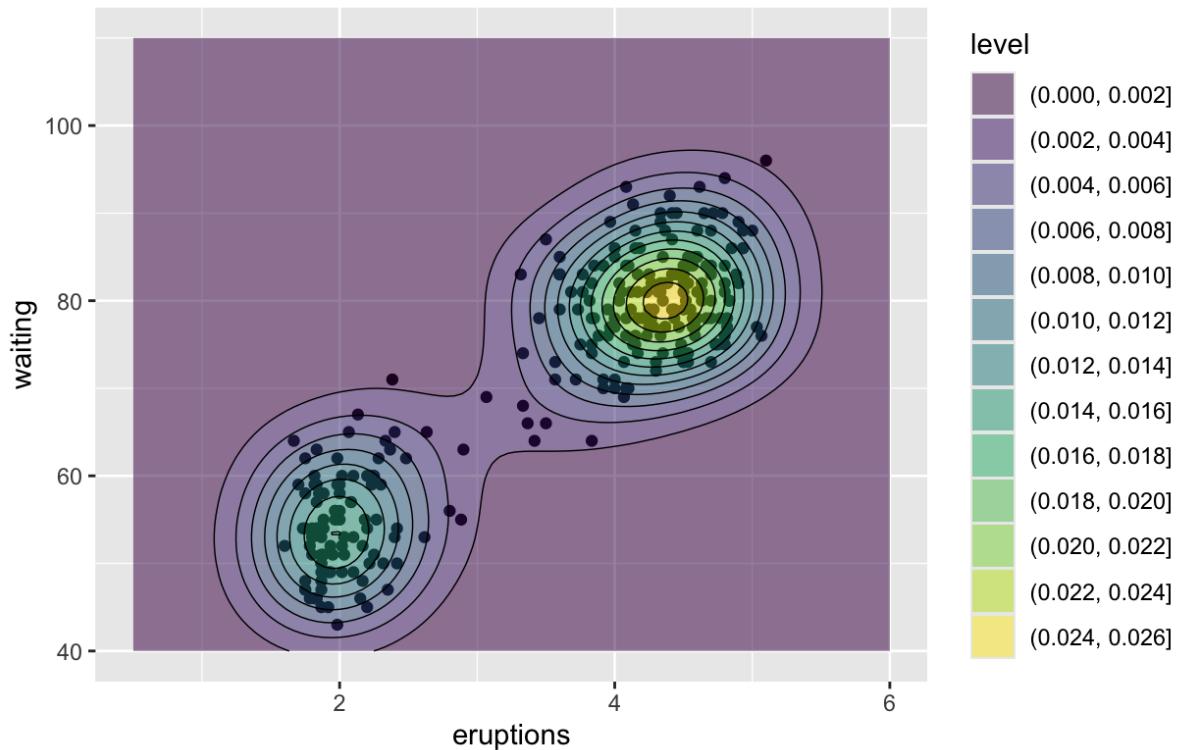
# contour bands

m + geom\_density\_2d\_filled(alpha = 0.5)

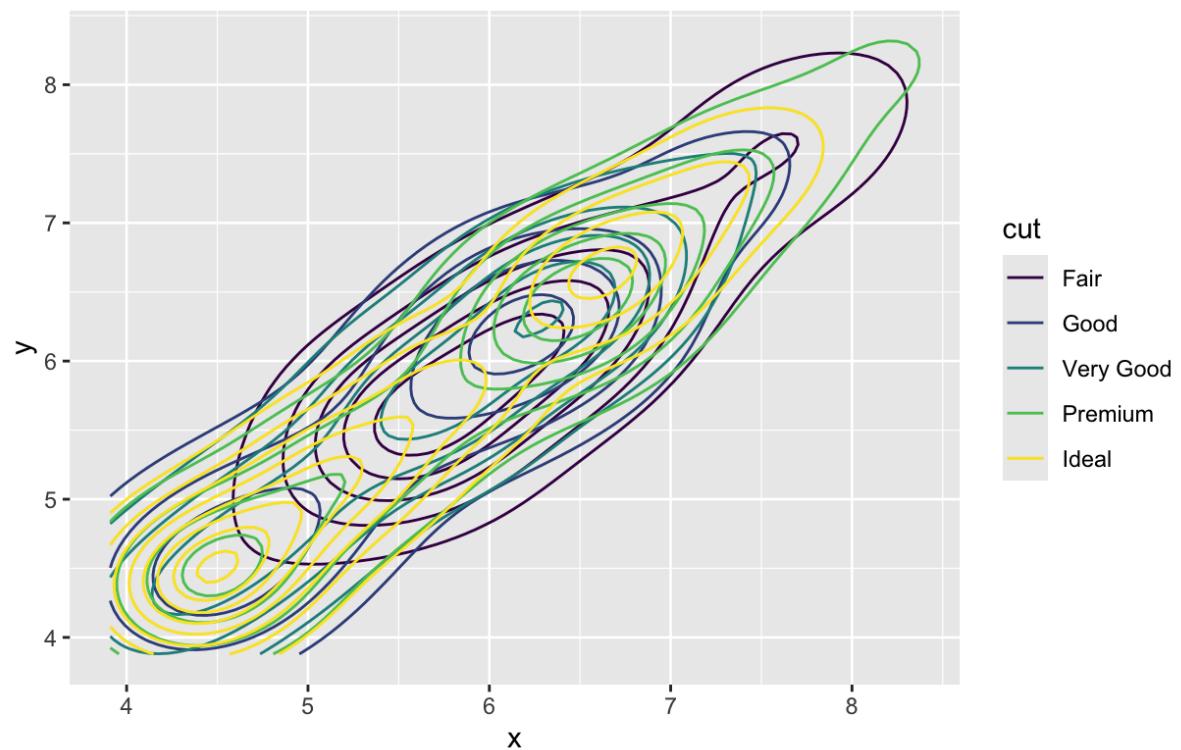


# contour bands and contour lines

```
m + geom_density_2d_filled(alpha = 0.5) +
  geom_density_2d(linewidth = 0.25, colour = "black")
```

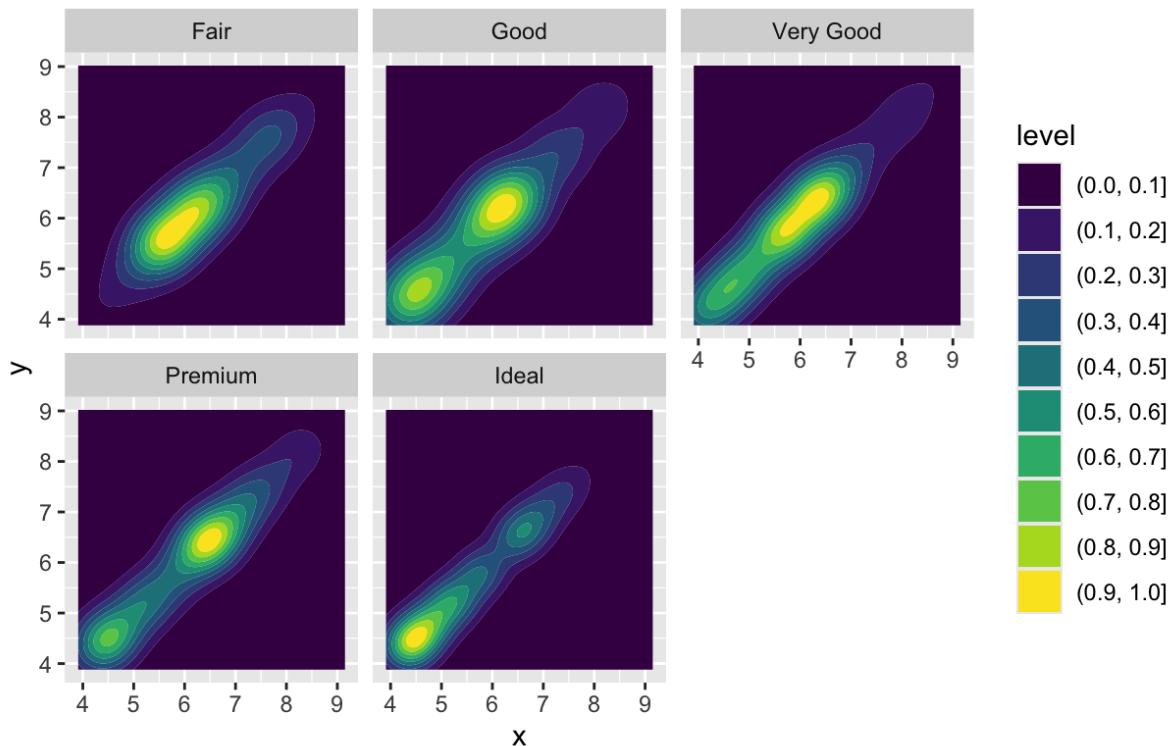
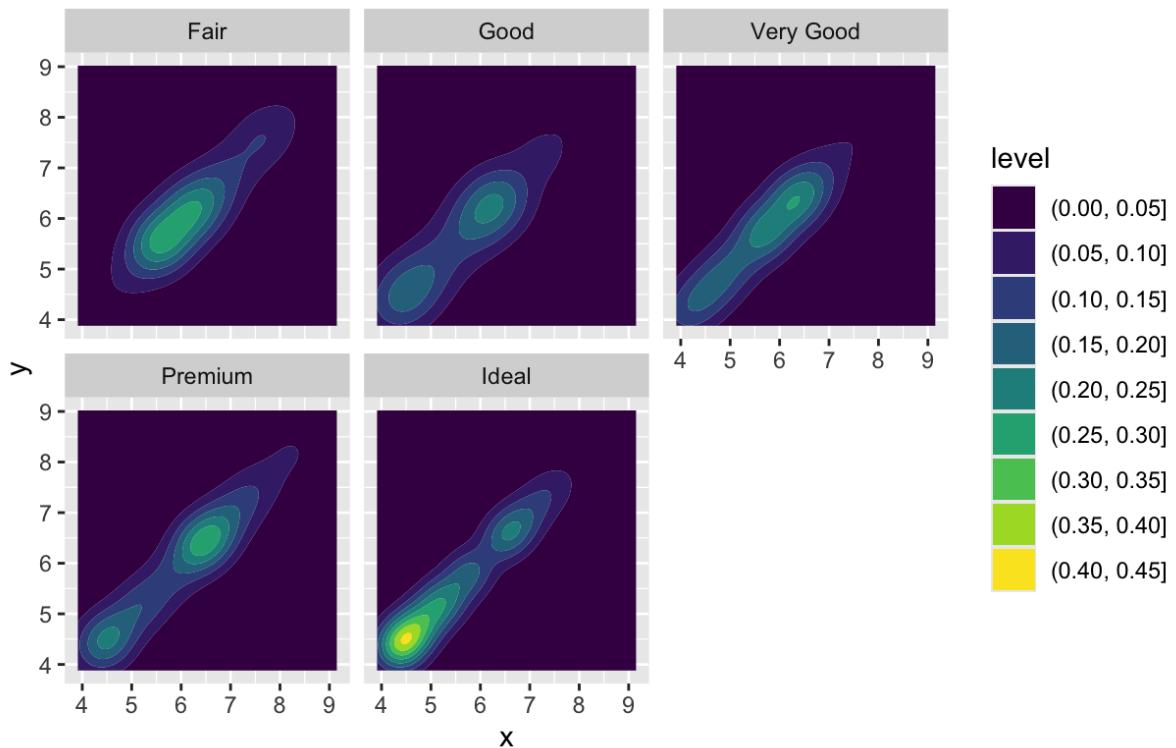


```
set.seed(4393)
dsmall <- diamonds[sample(nrow(diamonds), 1000), ]
d <- ggplot(dsmall, aes(x, y))
```



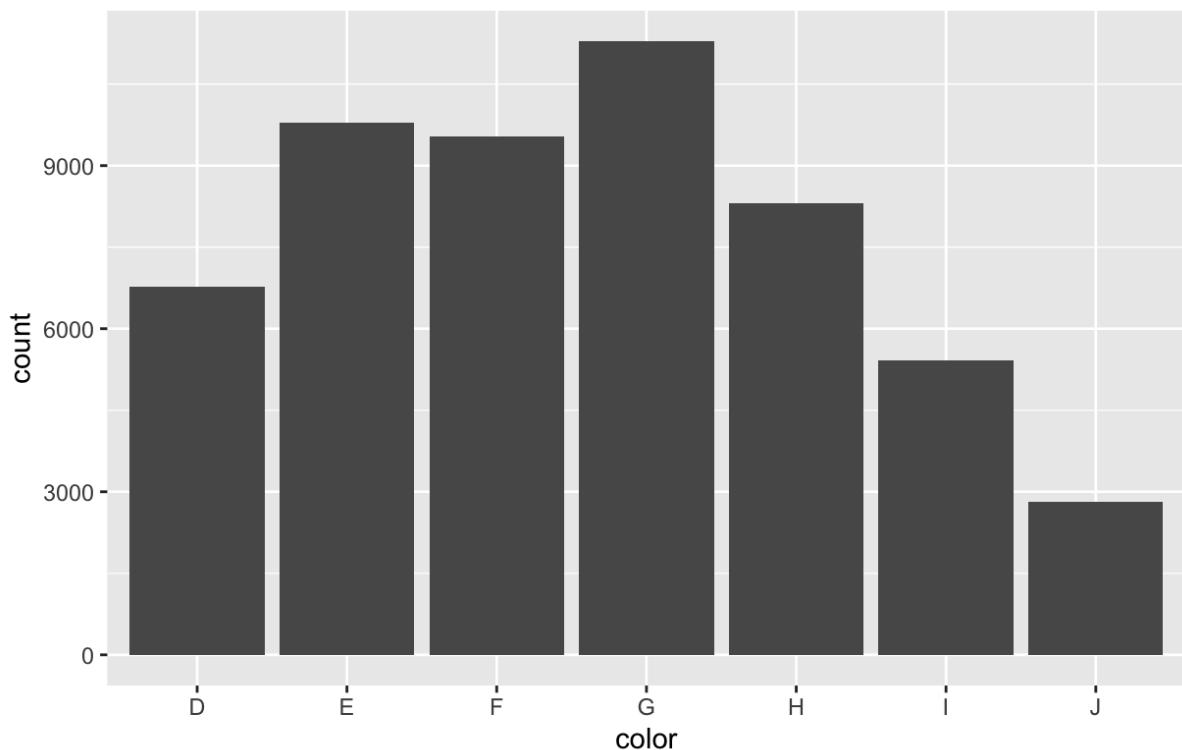
```
d + geom_density_2d(aes(colour = cut))
d + geom_density_2d_filled() + facet_wrap(vars(cut))
```

```
d + geom_density_2d_filled(contour_var = "ndensity") + facet_wrap(vars(cut))
```

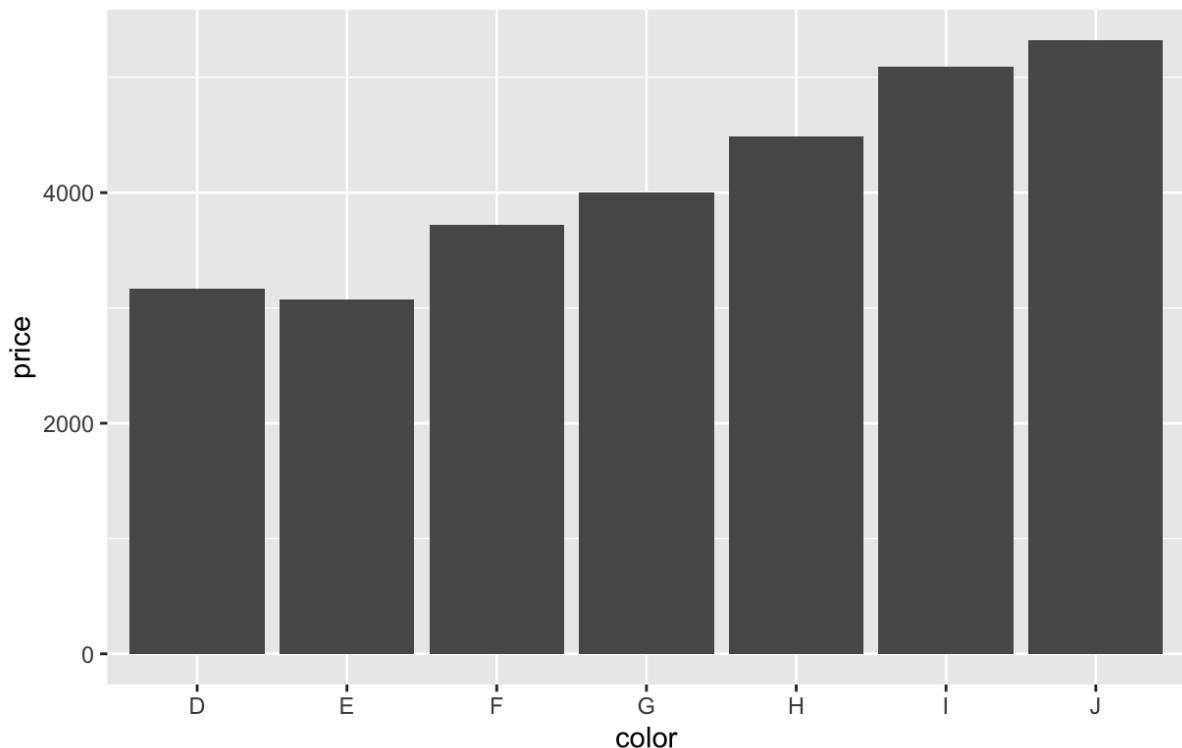


```
ggplot(diamonds, aes(color)) +
```

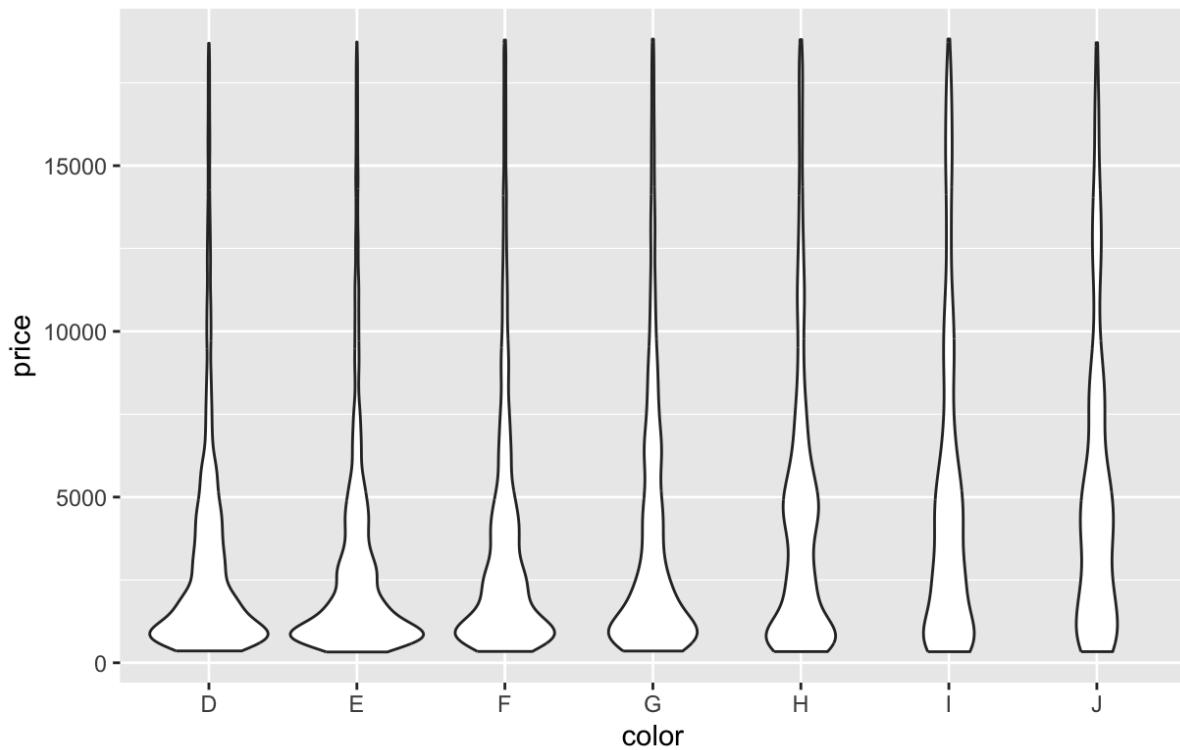
```
geom_bar()
```



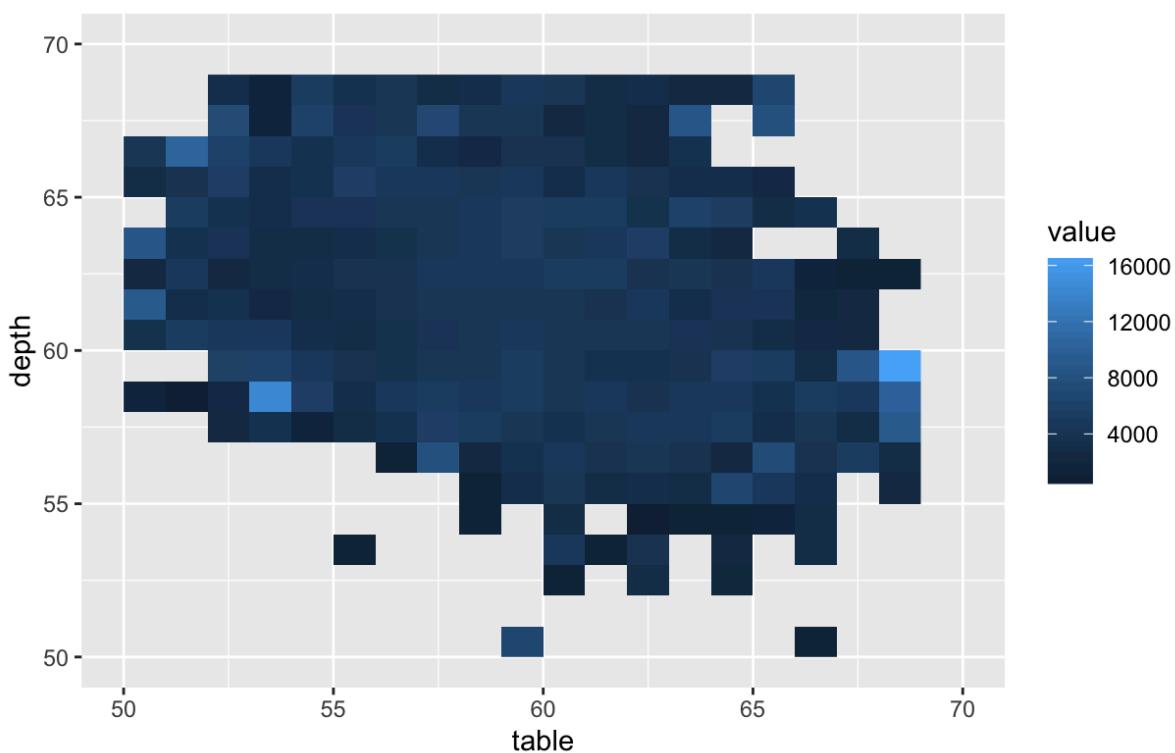
```
ggplot(diamonds, aes(color, price)) +  
  geom_bar(stat = "summary_bin", fun.y = mean)
```



```
ggplot(diamonds, aes(color, price)) +  
  geom_violin()
```

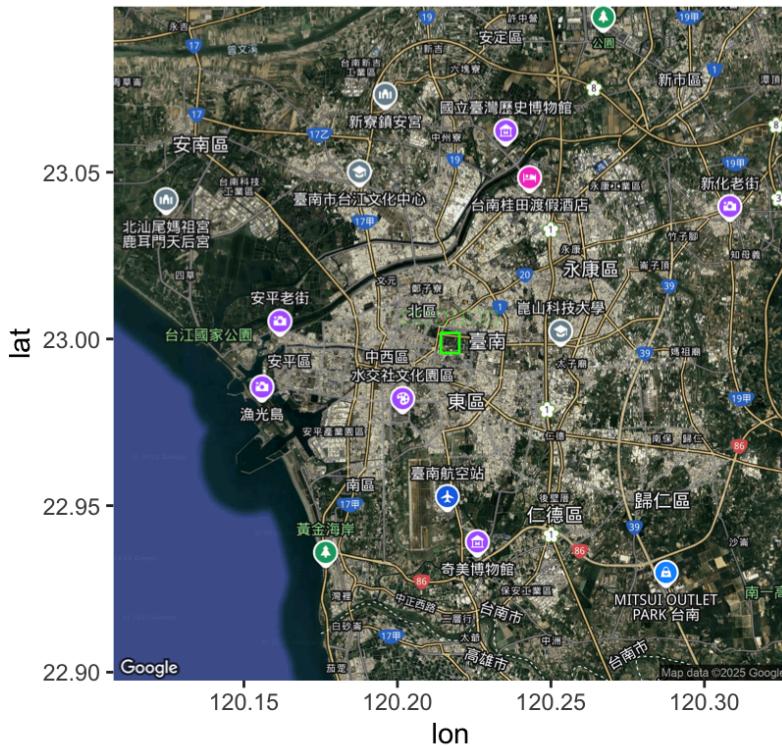
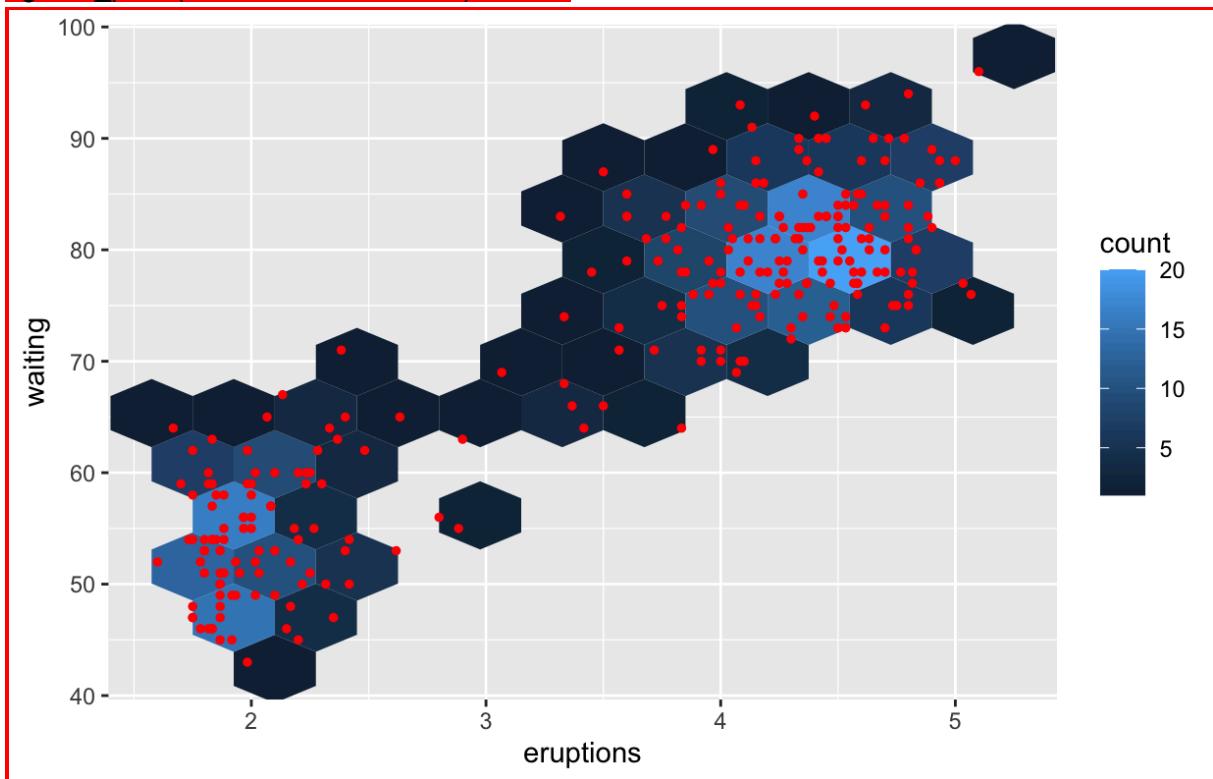


```
ggplot(diamonds, aes(table, depth, z = price)) +
  geom_raster(binwidth = 1, stat = "summary_2d", fun = mean,
  na.rm = TRUE) +
  xlim(50, 70) +
  ylim(50, 70)
```



```
ggplot(faithful, aes(x = eruptions, y = waiting)) +
  geom_hex(bins = 10) +
```

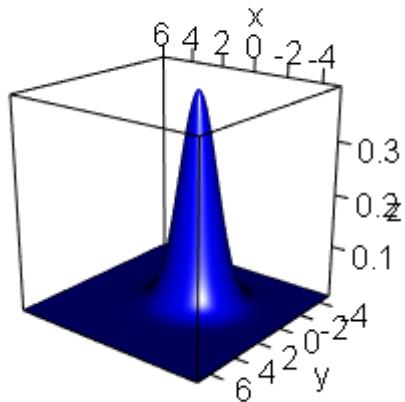
```
geom_point(color = "red", size = 1) 期末考
```



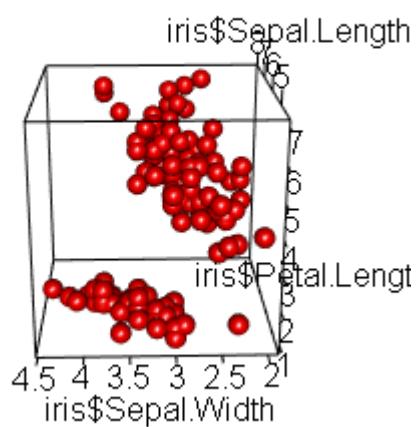
```
x <- seq(-5, 6, by = 0.1)
y <- seq(-5, 7, by = 0.1)
z <- outer(x, y, function(x, y) dnorm(sqrt(x^2 + y^2)))
```

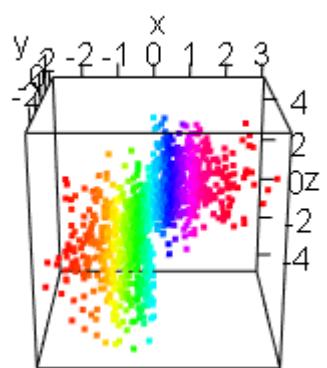
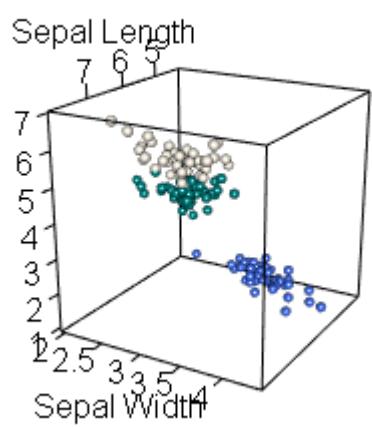
```
# Create a 3D surface plot
persp3d(x, y, z, col = "blue")
```

```
# add animation  
play3d(spin3d(axis = c(0, 0, 1)), duration = 10)
```



```
data(iris)  
  
# plot the 3D graph  
plot3d(iris$Sepal.Length, iris$Sepal.Width, iris$Petal.Length, col = "red", size = 3, type = "s")  
# add animations  
play3d(spin3d(axis = c(0, 0, 1)), duration = 10)
```





✓ —