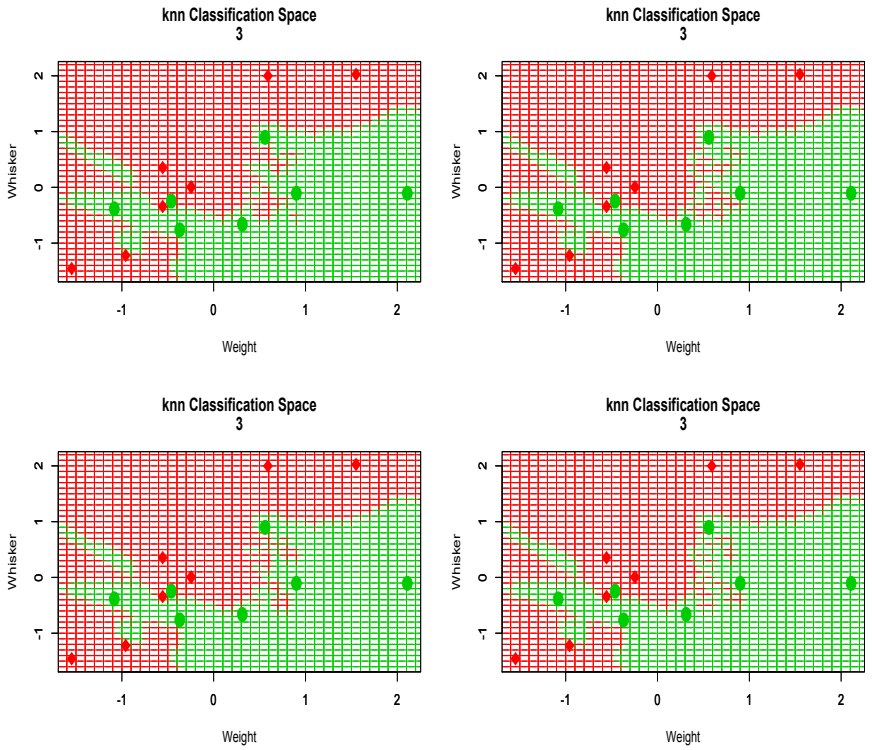
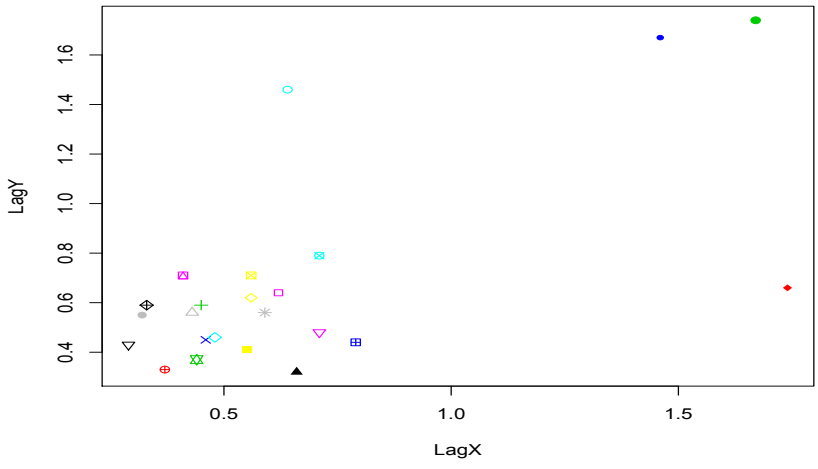
Marcus Crowder
CIS 3920 Data-Mining
KNN-5

5.1
After running the NewProbeKNN algorithm 4 times I was able to detect strange behavior around the circled regions. Strange because each graph looks slightly different and the blotches intrude between green points.



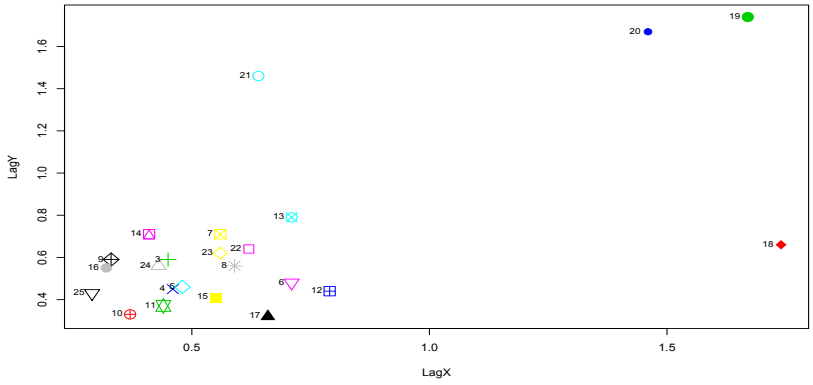
5.2 Pick a Dataset of your own with two continuous x-variable and a class variable then run through the steps shown in LN5.
I choose my daily return range to use for this example. As I wanted to see how nice it would look on a graph

Plot of My Lag1 and Lag

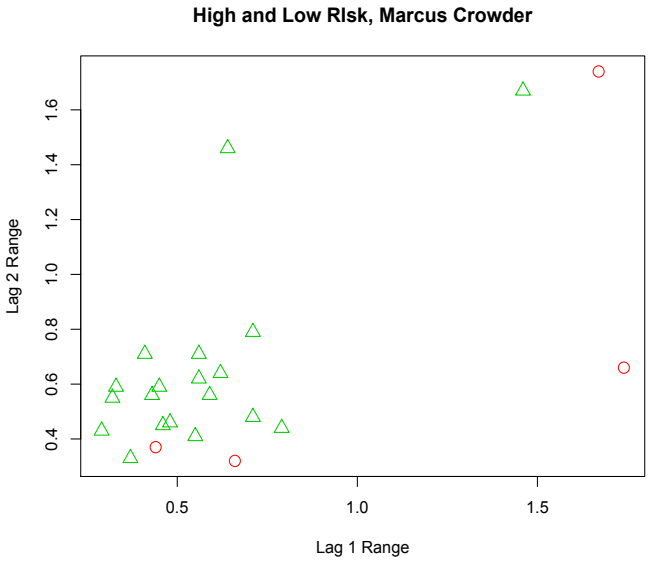


plot(Lag1, Lag2, pch=(1:25), col=(1:25))

> plot(LagX, LagY, pch=(1:25), col=(1:25), cex=2)
> text(LagX, LagY, labels=c(1:25), cex=.8, pos=2)

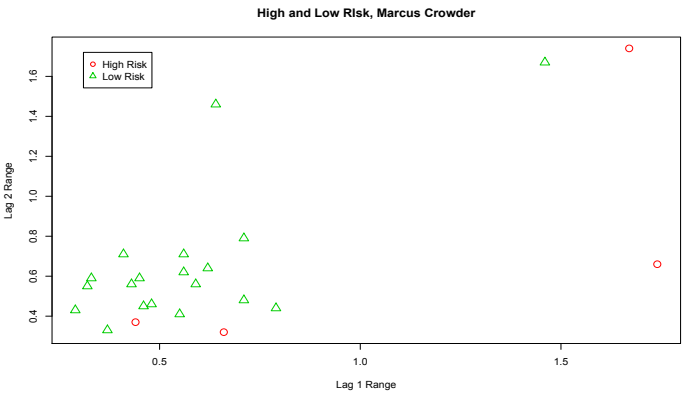


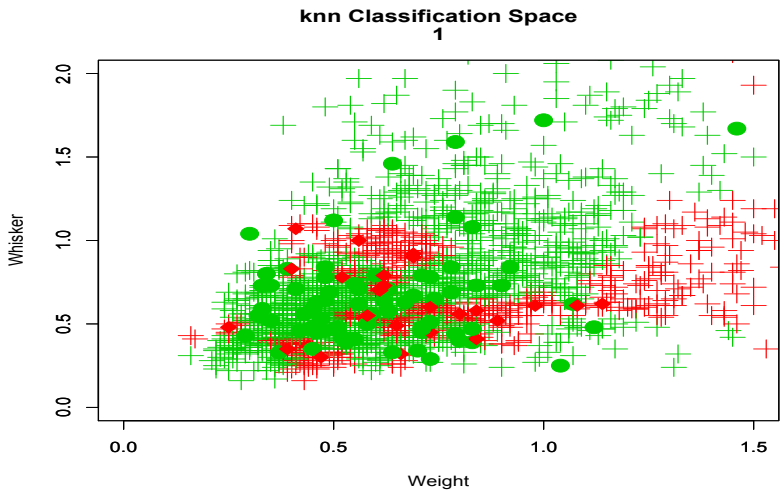
I turned the High Risk Marks into Red Circles and the Green Triangles as Low Risk to Prettify my graph
> plot(LagX, LagY, pch=(1:25), col=(1:25), cex=2)



Then I upped my previous example by adding a legend to the graph

> plot(LagX, LagY, pch=(1:25), col=(1:25), cex=2)
> text(LagX, LagY, labels=c(1:25), cex=.8, pos=2)





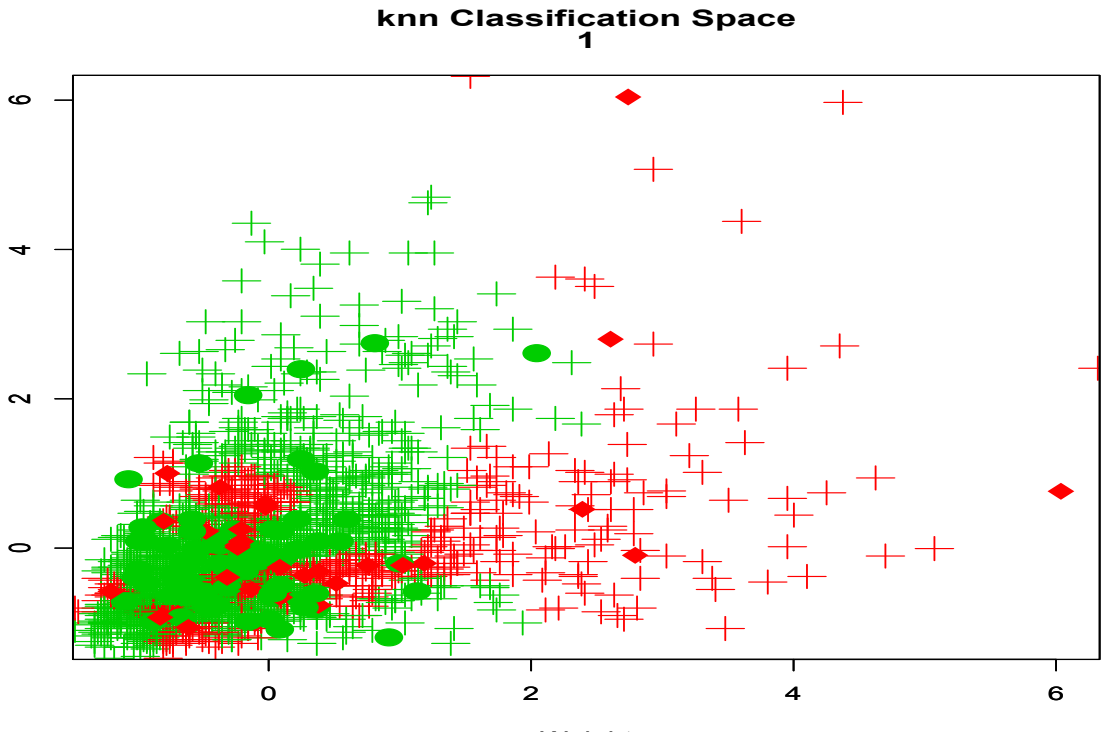
In order to make the Knn Classification look presentable with my data points I had to use the fix(ProbeKnn) function to adjust to xlim and ylim of the plot regions. In addition to that I wanted to see how the graphs would look with a lot more plot points.

```
plot(ProbeX[,1],ProbeX[,2],main=c("knn Classification Space", k),
 xlab="Weight",ylab="Whisker",cex=2, pch=3,col=ProbeColor, xlim=c(0,1.5),ylim=c(0,2))

plot(TrainX[,1],TrainX[,2],ann=FALSE,pch=c(cbind(TrainY)+17),col=c(cbind(TrainY)+17),xlim=c
 (0,1.5),ylim=c(0,2),cex=2)
```

After scratching my head and getting things wrong for a while I finally found my Probelag variable.

```
> test.X = cbind(Lag1,Lag2)[104:2000,]
> ProbeLag = test.X
> ProbeKnn(train.X, ProbeLag, train.Risk, 1)
```



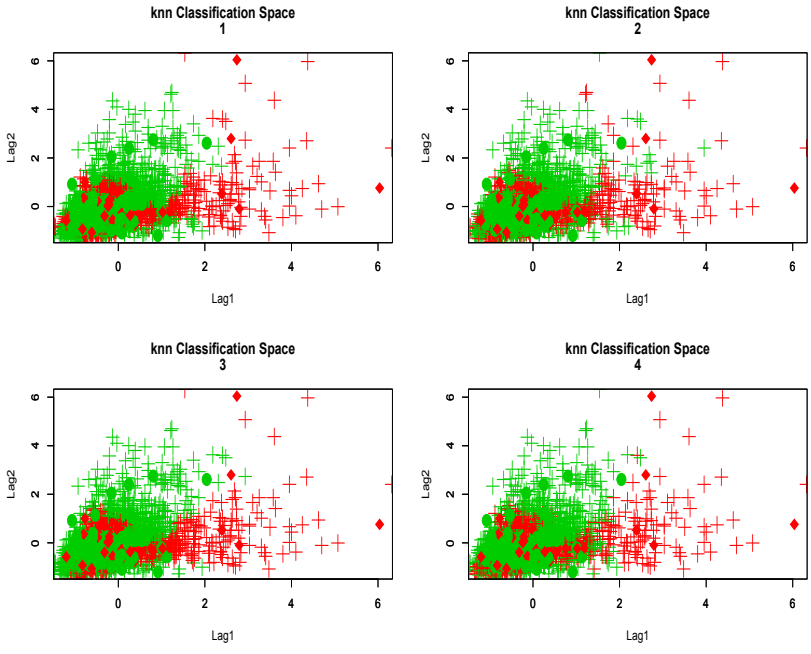
After a couple failed attempts and bad commands I was able to standardize my Lag and Train variables with the apply function

> StLagTrainX = apply(train.X, 2, scale)

> StProbeLag = apply(ProbeLag, 2, scale)

> NewProbeKnn(StLagTrainX, StProbeLag, train.Risk, 1)

Finally I loaded 4 examples of the NewProbe Knn with different variations of K there are some slight variations on the streaks especially on the lower left corner for each variation of k most noticeably in K=4 on the bottom right Graph



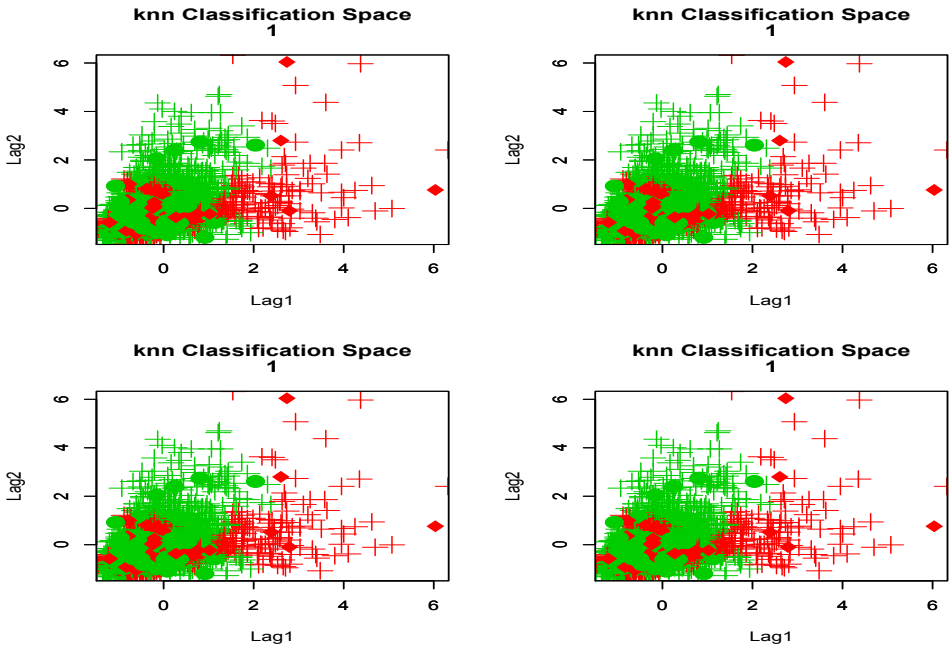
```
> par(mfrow=c(2,2))
> NewProbeKnn(StLagTrainX, StProbeLag, train.Risk, 1)
NULL
> NewProbeKnn(StLagTrainX, StProbeLag, train.Risk, 2)
NULL
> NewProbeKnn(StLagTrainX, StProbeLag, train.Risk, 3)
NULL
> NewProbeKnn(StLagTrainX, StProbeLag, train.Risk, 4)
NULL
```

I used these commands to load my data in.

```
MKCMarket <- read.csv("~/Downloads/MKC.csv")
> Lag1 <- MKCMarket$Lag1Range
> Lag2 <- MKCMarket$Lag2Range
> Risk = MKCMarket$Risk
> train.X = cbind(Lag1,Lag2)[3:103,]
> test.X = cbind(Lag1,Lag2)[104:204,]
> train.Risk=Risk[03:103]
> test.Risk=Risk[104:204]
```

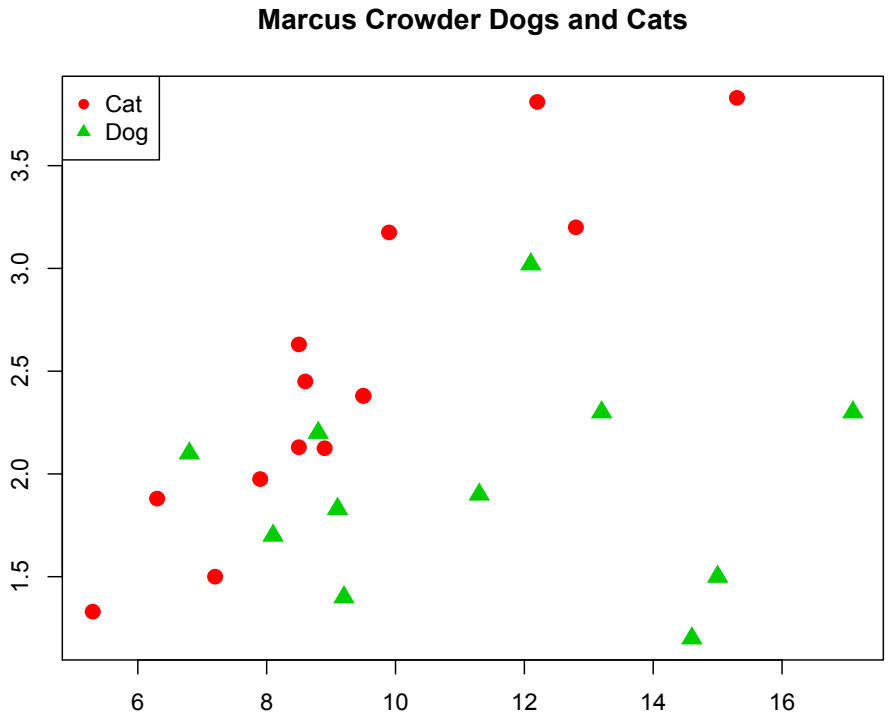
5.3

There does not seem to be any differences that I could put my finger on for the



repeat NewProbeKNN graphs when K=1.

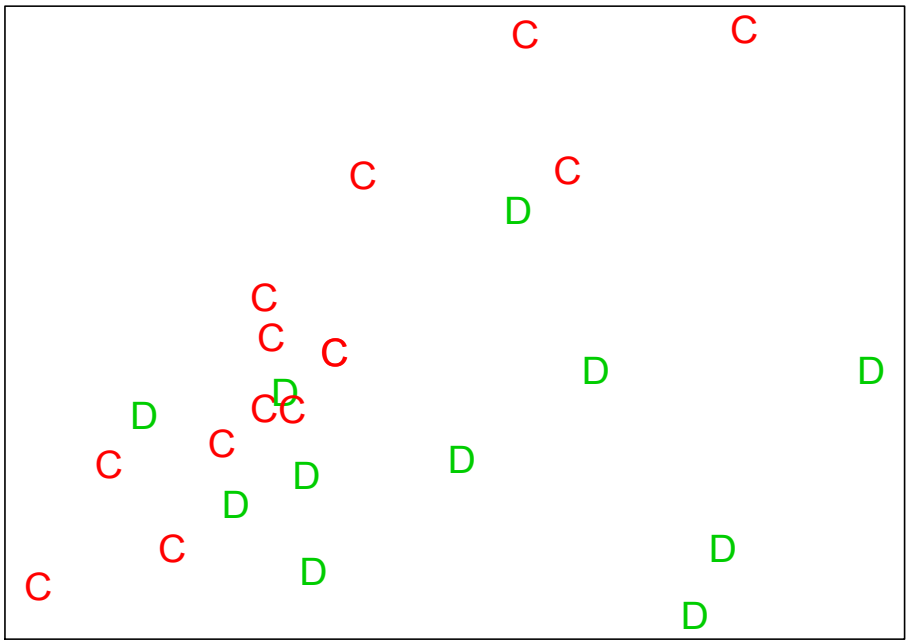
5.4 This took some



googling but I was able to come up with a plot that Added a Red Solid Red Circle and Solid Green Triangle. By adding +15 to the CatDog PCH. Increasing. Cat has a value of 1 and Dogs have a value of 2 placing each Cat Dogs point at pch 16 and pch 17 and by looking at the R index we find that those values produce a solid Circle and Solid Traingle the Color is added by the Col concatenation. To remove the x and y labels I just left a blank string. (to remove the axis numbering I would have to use xant = 'n' and yant = 'n')
http://www.sthda.com/english/wiki/r-plot-pch-symbols-the-different-point-shapes-available-in-r
Website for R reference

```
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog)+15, col=(c(catdog)+1),
cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> legend("topleft", legend=c("Cat", "Dog"), pch=c(1,2)+15, col=c(2,3))
```

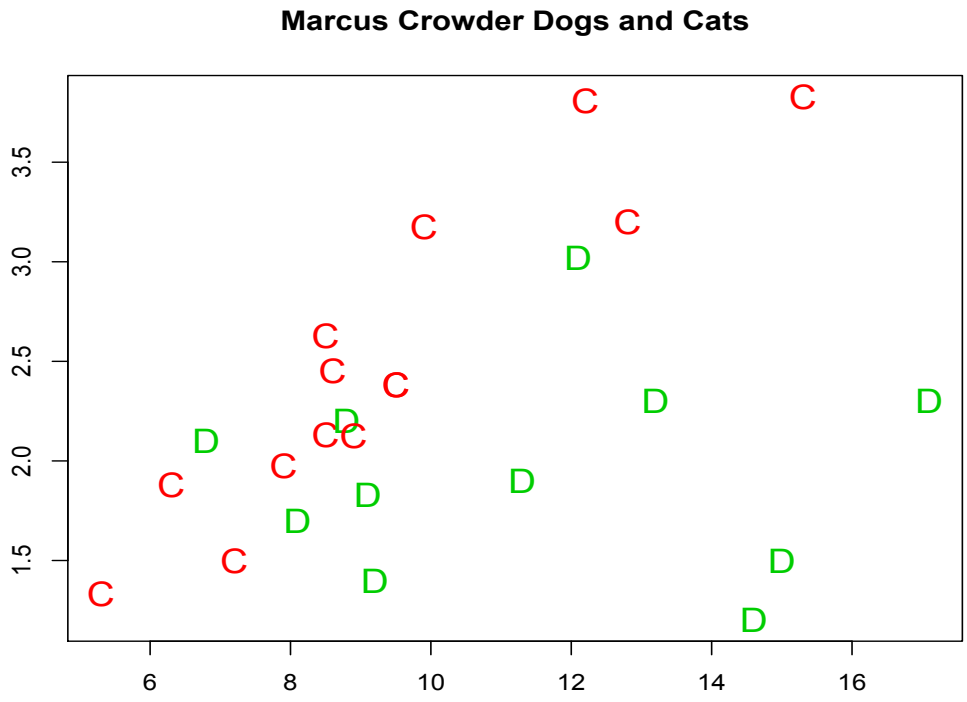
5.5 This section was even more challenging but I was able to find a neat function. The strtrim() function took apart the CatDog factor and it removes the first letter from each string in the concatenation.




```
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1),
cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "")

> strtrim(catdog, 1)
 [1] "D" "C" "D" "C" "D" "D" "C" "C" "C" "C" "D" "D" "C" "C" "D" "D" "C" "C"
"C" "C" "D" "D" "C" "C" "D"
```

Here is me trying a bunch of functions and plots to make my graph =).



```
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= """, labels =
FALSE)
+
+
+
+
+
+ )
+ ""
Error: unexpected string constant in:
")
""
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "", labels =
FALSE)
Warning messages:
1: In plot.window(...) : "labels" is not a graphical parameter
2: In plot.xy(xy, type, ...) : "labels" is not a graphical parameter
3: In box(...) : "labels" is not a graphical parameter
4: In title(...) : "labels" is not a graphical parameter
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "")
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "", xaxt = 'n',
yaxt = 'n')
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "")
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog), col=(c(catdog)+1), cex=1.5, main = "Dogs and Cats", xlab="Weight", ylab="Whisker Length", xaxt = 'n',
yaxt = 'n', ann=FALSE)
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog), col=(c(catdog)+1), cex=1.5, main = "Dogs and Cats", xlab="Weight", ylab="Whisker Length", xaxt = 'n',
yaxt = 'n')
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> legend("topleft", legend=c("Cat", "Dog"), pch=c(1,2), col=c(2,3))
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog), col=(c(catdog)+15), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog), col=(c(catdog)+16), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog)+15, col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> legend("topleft", legend=c("Cat", "Dog"), pch=c(1,2), col=c(2,3))
> legend("topleft", legend=c("Cat", "Dog"), pch=c(1,2)+15, col=c(2,3))
> plot(Dogs$Weight, Dogs$Whisker, pch=c(catdog)+15, col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab="")
> legend("topleft", legend=c("Cat", "Dog"), pch=c(1,2)+15, col=c(2,3))
> plot(Dogs$Weight, Dogs$Whisker, pch=strtrim(catdog,1), col=(c(catdog)+1), cex=1.5, main = "Marcus Crowder Dogs and Cats", xlab="", ylab= "")
> cls(catdog)
```