Week 2 InClass Activity

Akram Almohalwas

Week 2 In-Class Activity:

```
library(ggplot2)
theme_update(plot.title = element_text(hjust = 0.5)) # Centers the Title#
head(iris)

##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa

set.seed(1234567)
newiris<-iris[sample(nrow(iris),150,replace = F),]

head(newiris)

##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 85           5.4         3.0          4.5         1.5 versicolor
## 109          6.7         2.5          5.8         1.8  virginica
## 136          7.7         3.0          6.1         2.3  virginica
## 5            5.0         3.6          1.4         0.2     setosa
## 113          6.8         3.0          5.5         2.1  virginica
## 70           5.6         2.5          3.9         1.1 versicolor

attach(newiris)
summary(newiris[,1:4])

##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  

normalize<-function(x) {return((x-min(x))/(max(x)-min(x)))}
summary(normalize(newiris[,2]))

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  0.0000  0.3333  0.4167  0.4406  0.5417  1.0000
```

```
iris.norm <-cbind(as.data.frame(lapply(newiris[,1:4],normalize)),Species)
head(iris.norm)

##   Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 1    0.3055556   0.4166667   0.59322034  0.58333333 versicolor
## 2    0.6666667   0.2083333   0.81355932  0.70833333  virginica
## 3    0.9444444   0.4166667   0.86440678  0.91666667  virginica
## 4    0.1944444   0.6666667   0.06779661  0.04166667     setosa
## 5    0.6944444   0.4166667   0.76271186  0.83333333  virginica
## 6    0.3611111   0.2083333   0.49152542  0.41666667 versicolor

summary(iris.norm[,1:4])

##   Sepal.Length     Sepal.Width      Petal.Length     Petal.Width
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.2222   1st Qu.:0.3333   1st Qu.:0.1017   1st Qu.:0.08333
##  Median :0.4167   Median :0.4167   Median :0.5678   Median :0.50000
##  Mean   :0.4287   Mean   :0.4406   Mean   :0.4675   Mean   :0.45806
##  3rd Qu.:0.5833   3rd Qu.:0.5417   3rd Qu.:0.6949   3rd Qu.:0.70833
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000

i=1:dim(iris.norm)[1]
set.seed(9876)

150*0.7

## [1] 105

i.train<-sample(i,105,replace=F)

iris.train=iris.norm[i.train, ]
iris.test = iris.norm[-i.train, ]

head(iris.train)

##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 128    0.4444444   0.4166667    0.5423729   0.5833333 versicolor
## 55     0.5555556   0.5416667    0.6271186   0.6250000 versicolor
## 19     0.1666667   0.1666667    0.3898305   0.3750000 versicolor
## 84     0.3333333   0.1250000    0.5084746   0.5000000 versicolor
## 62     0.6388889   0.4166667    0.5762712   0.5416667 versicolor
## 41     0.5555556   0.3333333    0.6949153   0.5833333  virginica

dim(iris.train)

## [1] 105   5

head(iris.test)

##    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 2     0.6666667   0.2083333   0.81355932  0.70833333  virginica
## 3     0.9444444   0.4166667   0.86440678  0.91666667  virginica
```

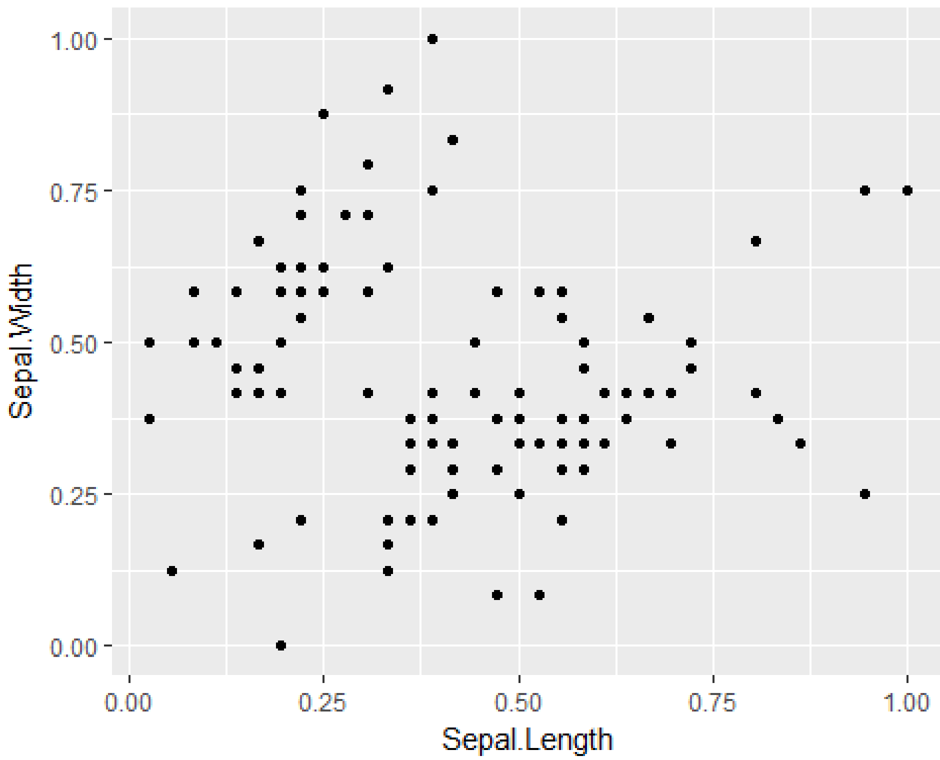
```
## 4     0.1944444   0.6666667   0.06779661  0.04166667     setosa
## 11    0.5555556   0.1250000   0.57627119  0.50000000 versicolor
## 16    0.3611111   0.4166667   0.59322034  0.58333333 versicolor
## 21    0.6666667   0.4166667   0.71186441  0.91666667  virginica
```

```r
dim(iris.test)
```

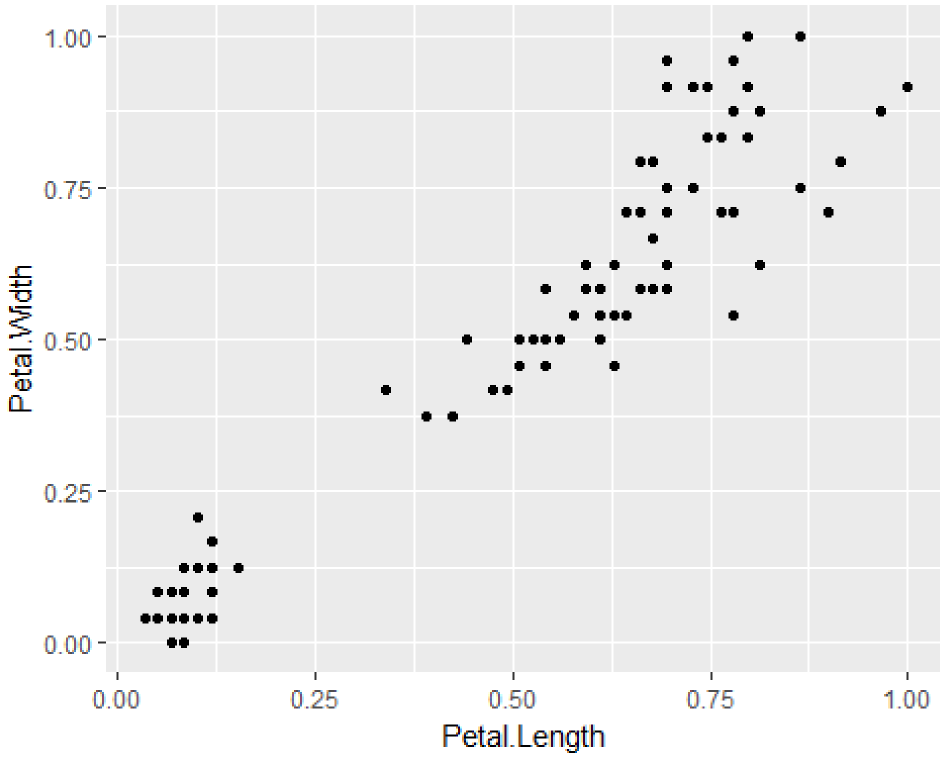
```
## [1] 45  5
```

Plots:

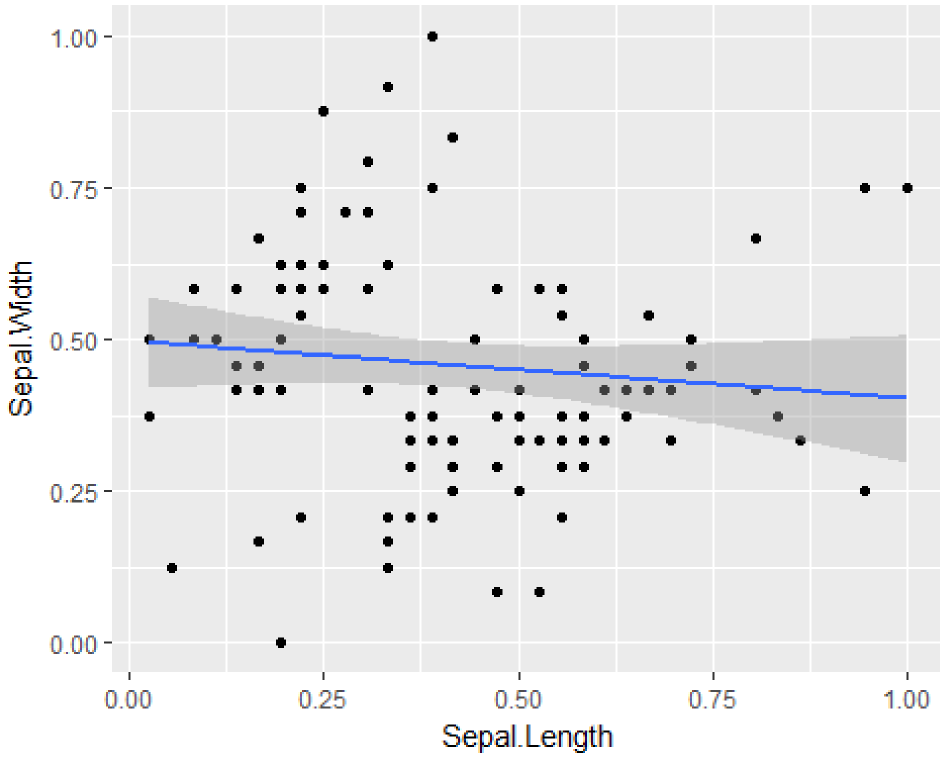
```r
library(ggplot2)
qplot(Sepal.Length,Sepal.Width,data=iris.train)
```



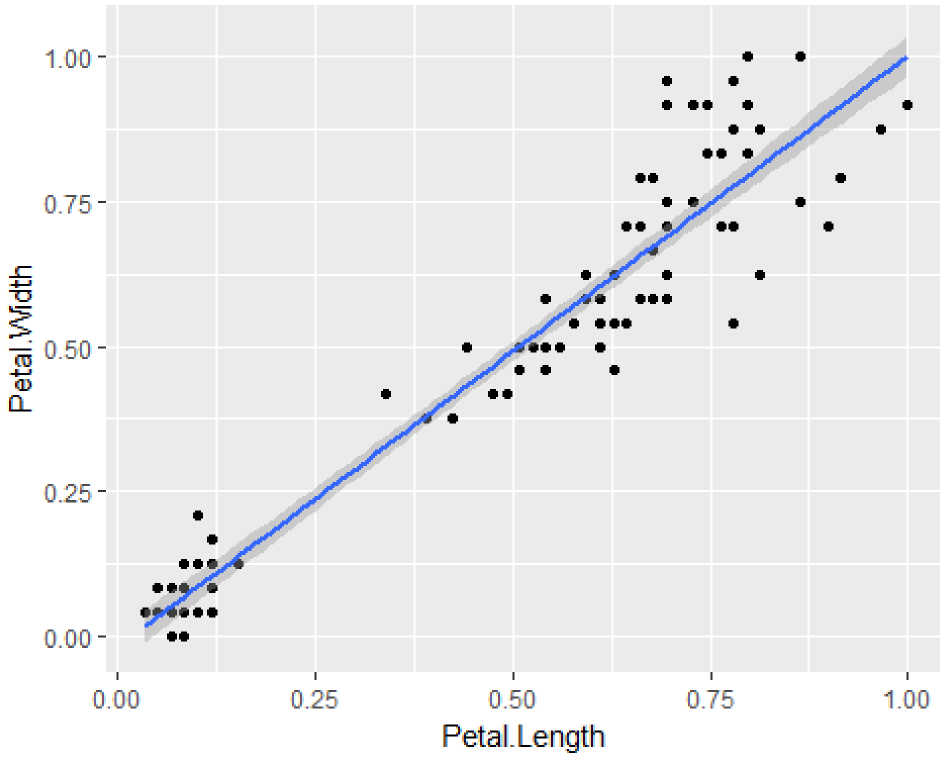

```r
qplot(Petal.Length,Petal.Width,data=iris.train)
```



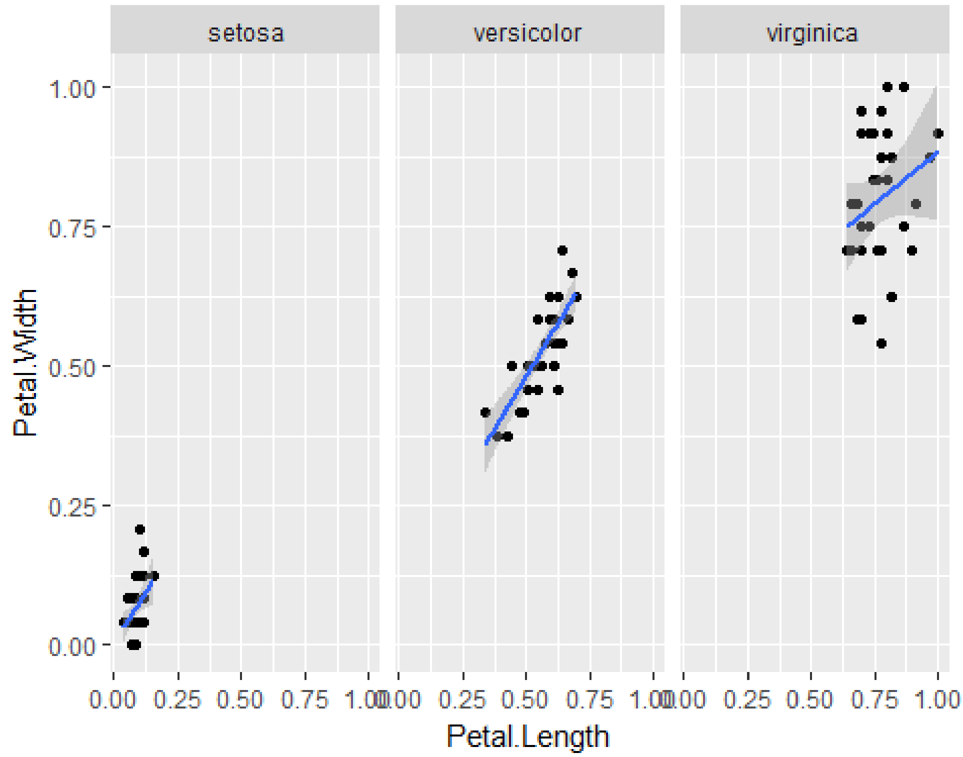

```
qplot(Sepal.Length,Sepal.Width,data=iris.train,geom
=c("point","smooth"),method="lm")

## Warning: Ignoring unknown parameters: method
```



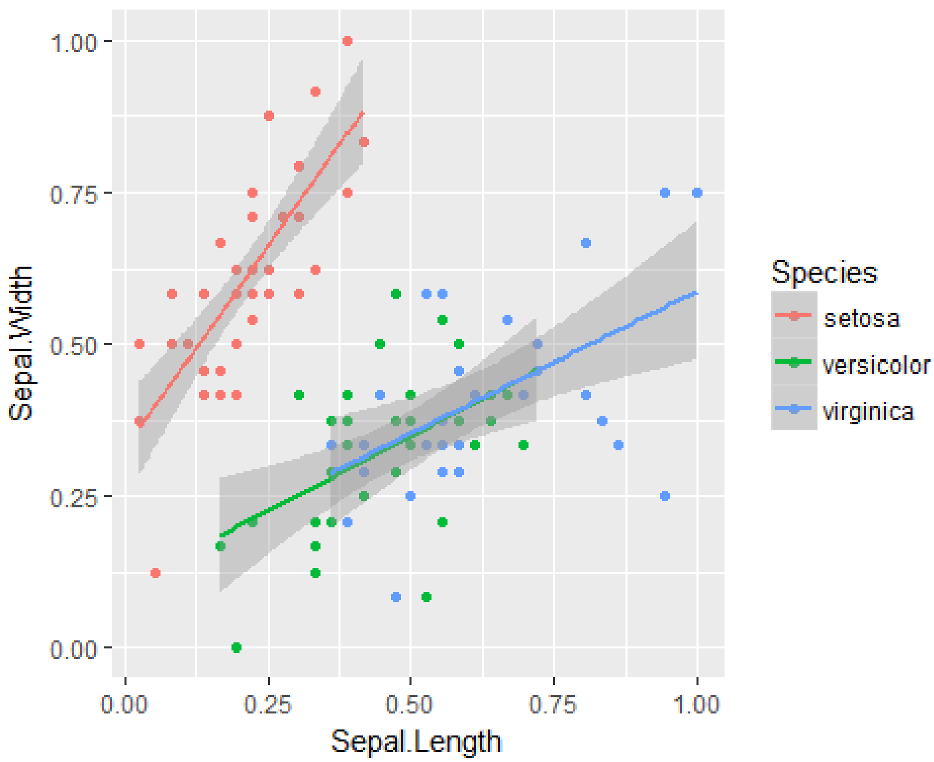

```
qplot(Petal.Length,Petal.Width,data=iris.train,geom
=c("point","smooth"),method="lm")

## Warning: Ignoring unknown parameters: method
```



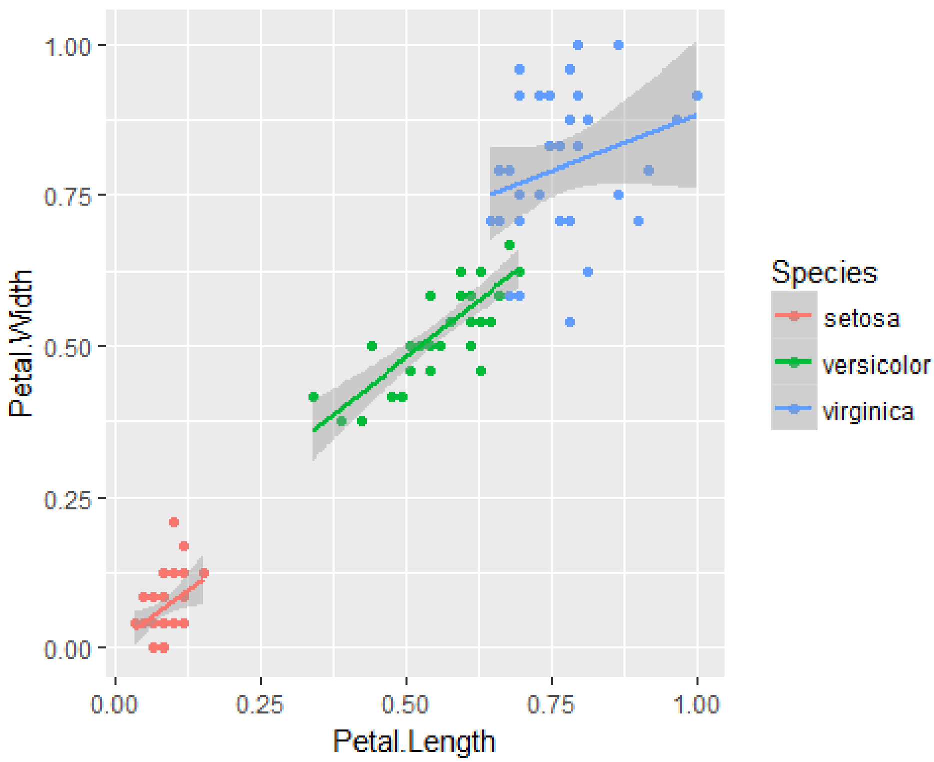

```r
qplot(Petal.Length,Petal.Width, data=iris.train,facets=.~Species,geom
=c("point","smooth"),method="lm")

## Warning: Ignoring unknown parameters: method
```




```
qplot(Sepal.Length,Sepal.Width,color=Species,data=iris.train,geom
=c("point","smooth"),method="lm")

## Warning: Ignoring unknown parameters: method
```




```
qplot(Petal.Length,Petal.Width,color=Species,data=iris.train,geom
=c("point","smooth"),method="lm")

## Warning: Ignoring unknown parameters: method
```



1) create a predictive model that uses Sepal length and Species to predict Sepal Width

a) Create a linear model with one numerical predictor

```
lm10<-lm(Sepal.Width~Sepal.Length,data=iris.train)
```

b) Create a linear model with one categorical predictor.

```
lm11<- lm(Sepal.Width~Species,data=iris.train)
```

c) Create a linear model with ine numerical and one categorical predictor

```
lm12<-lm(Sepal.Width~Sepal.Length+Species,data=iris.train)

summary(lm10)

## 
## Call:
## lm(formula = Sepal.Width ~ Sepal.Length, data = iris.train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.47871 -0.11388 -0.02268  0.10757  0.53966 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   0.49708    0.03981  12.486   <2e-16 ***
```

```
## Sepal.Length -0.09448    0.08413  -1.123    0.264
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1904 on 103 degrees of freedom
## Multiple R-squared:  0.0121, Adjusted R-squared:  0.002504
## F-statistic: 1.261 on 1 and 103 DF,  p-value: 0.2641
```

```
summary(lm11)
```

```
##
## Call:
## lm(formula = Sepal.Width ~ Species, data = iris.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.47863 -0.10363  0.00130  0.08701  0.39637
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.60363    0.02421  24.929  < 2e-16 ***
## Speciesversicolor -0.27398    0.03548  -7.722 8.14e-12 ***
## Speciesvirginica  -0.18827    0.03607  -5.220 9.47e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1512 on 102 degrees of freedom
## Multiple R-squared:  0.3826, Adjusted R-squared:  0.3705
## F-statistic:  31.6 on 2 and 102 DF,  p-value: 2.085e-11
```

```
summary(lm12)
```

```
##
## Call:
## lm(formula = Sepal.Width ~ Sepal.Length + Species, data = iris.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38185 -0.06879 -0.00589  0.09074  0.28051
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.47141    0.02755  17.109  < 2e-16 ***
## Sepal.Length       0.63792    0.09148   6.973 3.27e-10 ***
## Speciesversicolor -0.43518    0.03732 -11.662  < 2e-16 ***
## Speciesvirginica  -0.45918    0.04895  -9.381 2.08e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1249 on 101 degrees of freedom
```

```
## Multiple R-squared:  0.5833, Adjusted R-squared:  0.5709
## F-statistic: 47.12 on 3 and 101 DF,  p-value: < 2.2e-16
```

```r
anova(lm10)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##               Df Sum Sq  Mean Sq F value Pr(>F)
## Sepal.Length   1 0.0457 0.045692  1.2611 0.2641
## Residuals    103 3.7320 0.036233
```

```r
anova(lm11)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##            Df Sum Sq Mean Sq F value    Pr(>F)
## Species     2 1.4453 0.72267  31.604 2.085e-11 ***
## Residuals 102 2.3324 0.02287
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(lm12)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##               Df  Sum Sq Mean Sq F value  Pr(>F)
## Sepal.Length   1 0.04569 0.04569  2.9313 0.08995 .
## Species        2 2.15766 1.07883 69.2103 < 2e-16 ***
## Residuals    101 1.57436 0.01559
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

MSEs:

a) MSE of lm10:

```r
MSE10<- sum(lm10$residuals^2)/length(Sepal.Length)
MSE11<- sum(lm11$residuals^2)/length(Sepal.Length)
MSE12<- sum(lm12$residuals^2)/length(Sepal.Length)

# pred10<- predict(lm10,newdata =
as.data.frame(Sepal.Length),type="response")
# MSE.test10<- sum((Sepal.Length-pred10)^2)/length(Sepal.Length)
# MSE.test10

c(MSE10,MSE11,MSE12)
```

```
## [1] 0.02488013 0.01554916 0.01049573
```

2) create a predictive model that uses Sepal length and Species to predict Sepal Width

a) Create a linear model with one numerical predictor

```
lm20<-lm(Petal.Width~Petal.Length,data=iris.train)
```

b) Create a linear model with one categorical predictor.

```
lm21<- lm(Petal.Width~Species,data=iris.train)
```

c) Create a linear model with ine numerical and one categorical predictor

```
lm22<-lm(Petal.Width~Petal.Length+Species,data=iris.train)

summary(lm20)

##
## Call:
## lm(formula = Petal.Width ~ Petal.Length, data = iris.train)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.235689 -0.050225 -0.008559  0.048354  0.267541
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.01903    0.01499  -1.269    0.207
## Petal.Length  1.02145    0.02781  36.727   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08576 on 103 degrees of freedom
## Multiple R-squared:  0.9291, Adjusted R-squared:  0.9284
## F-statistic:  1349 on 1 and 103 DF,  p-value: < 2.2e-16

summary(lm21)

##
## Call:
## lm(formula = Petal.Width ~ Species, data = iris.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25651 -0.02819 -0.02137  0.05515  0.20182
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.06303    0.01388   4.542 1.53e-05 ***
## Speciesversicolor  0.46515    0.02034  22.872  < 2e-16 ***
## Speciesvirginica   0.73514    0.02067  35.558  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08668 on 102 degrees of freedom
## Multiple R-squared:  0.9282, Adjusted R-squared:  0.9268
## F-statistic: 659.6 on 2 and 102 DF,  p-value: < 2.2e-16
```

```
summary(lm22)
```

```
##
## Call:
## lm(formula = Petal.Width ~ Petal.Length + Species, data = iris.train)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.261340 -0.032759 -0.003784  0.037882  0.203619
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.01648    0.01531   1.076 0.284280
## Petal.Length         0.56984    0.10989   5.186 1.11e-06 ***
## Speciesversicolor    0.19185    0.05575   3.441 0.000843 ***
## Speciesvirginica     0.34224    0.07798   4.389 2.81e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07741 on 101 degrees of freedom
## Multiple R-squared:  0.9433, Adjusted R-squared:  0.9416
## F-statistic: 560.3 on 3 and 101 DF,  p-value: < 2.2e-16
```

```
anova(lm10)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##               Df Sum Sq  Mean Sq F value Pr(>F)
## Sepal.Length   1 0.0457 0.045692  1.2611 0.2641
## Residuals    103 3.7320 0.036233
```

```
anova(lm11)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##            Df Sum Sq Mean Sq F value    Pr(>F)
## Species     2 1.4453 0.72267  31.604 2.085e-11 ***
## Residuals 102 2.3324 0.02287
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(lm12)
```

```
## Analysis of Variance Table
##
## Response: Sepal.Width
##               Df  Sum Sq Mean Sq F value  Pr(>F)
## Sepal.Length   1 0.04569 0.04569  2.9313 0.08995 .
## Species        2 2.15766 1.07883 69.2103 < 2e-16 ***
## Residuals    101 1.57436 0.01559
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

KNN Classification Model:

```
library(class)
# k = sqrt(n)
sqrt(150)

## [1] 12.24745

model.1<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=1)
model.1

##  [1] virginica  virginica  setosa     versicolor versicolor virginica
##  [7] setosa     virginica  virginica  versicolor virginica  versicolor
## [13] versicolor versicolor virginica  setosa     virginica  versicolor
## [19] setosa     setosa     versicolor virginica  setosa     setosa
## [25] versicolor versicolor virginica  versicolor virginica  virginica
## [31] versicolor versicolor virginica  setosa     versicolor virginica
## [37] virginica  virginica  setosa     versicolor setosa     versicolor
## [43] versicolor virginica  setosa
## Levels: setosa versicolor virginica

length(model.1)

## [1] 45

table(model.1)

## model.1
##     setosa versicolor  virginica
##         11         17         17

aa<- table(iris.test[,5],model.1)
aa

##             model.1
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor      0         16         0
##   virginica       0          1        17

aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]
```

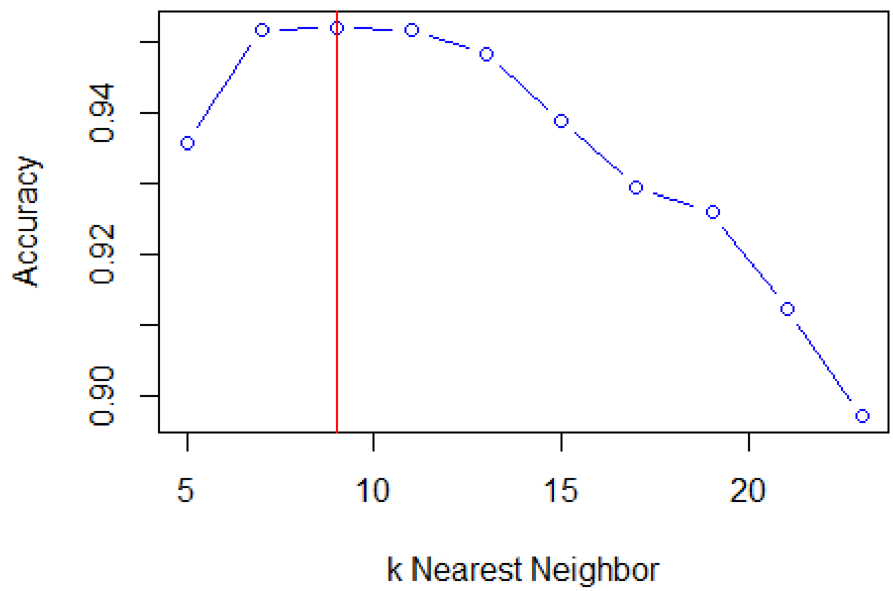
```
## [1] 1
```

Plotting k vs Accuracy:

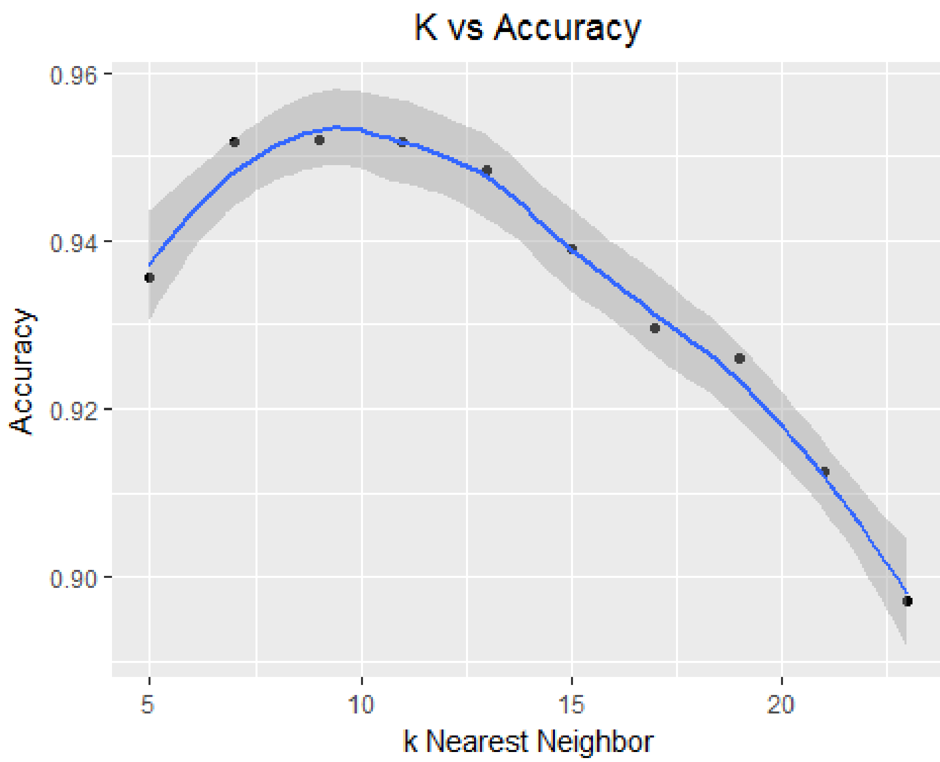
```
##########################################################################
# install.packages("caret")
# install.packages("e1071")
library(caret)

## Loading required package: lattice

library(e1071)
trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
set.seed(3333)
knn_fit <- train(Species ~., data = iris.train, method = "knn",
                 trControl=trctrl,
                 preProcess = c("center", "scale"),
                 tuneLength = 10)
knn_fit

## k-Nearest Neighbors
##
## 105 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 94, 95, 95, 94, 95, 94, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy   Kappa
##    5  0.9356229  0.9025906
##    7  0.9516835  0.9266172
##    9  0.9519865  0.9272019
##   11  0.9516835  0.9266782
##   13  0.9482828  0.9214128
##   15  0.9388889  0.9073712
##   17  0.9295623  0.8931353
##   19  0.9259259  0.8877337
##   21  0.9124579  0.8672833
##   23  0.8973064  0.8441926
##
## Accuracy was used to select the optimal model using  the largest value.
## The final value used for the model was k = 9.

tt<-data.frame(knn_fit[4])
str(tt)

## 'data.frame':    10 obs. of  5 variables:
##  $ results.k         : int  5 7 9 11 13 15 17 19 21 23
```

```
##  $ results.Accuracy  : num  0.936 0.952 0.952 0.952 0.948 ...
##  $ results.Kappa     : num  0.903 0.927 0.927 0.927 0.921 ...
##  $ results.AccuracySD: num  0.0685 0.0653 0.0491 0.0494 0.0623 ...
##  $ results.KappaSD   : num  0.1037 0.0994 0.0744 0.0749 0.0946 ...
```

```
plot(tt[,1],tt[,2],type="b",col="blue",xlab="k Nearest
Neighbor",ylab="Accuracy")
abline(v=9,col="red")
```




```
qplot(tt[,1],tt[,2],xlab="k Nearest
Neighbor",ylab="Accuracy",geom=c("point","smooth"),main="K vs Accuracy")

## `geom_smooth()` using method = 'loess'
```




```
##########################################################################
model.2<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=13)
model.2

##  [1] virginica  virginica  setosa     versicolor versicolor virginica
##  [7] setosa     virginica  virginica  versicolor virginica  versicolor
## [13] versicolor versicolor virginica  setosa     virginica  versicolor
## [19] setosa     setosa     versicolor virginica  setosa     setosa
## [25] versicolor versicolor virginica  versicolor virginica  virginica
## [31] versicolor versicolor virginica  setosa     versicolor virginica
## [37] virginica  virginica  setosa     versicolor setosa     versicolor
## [43] versicolor virginica  setosa
## Levels: setosa versicolor virginica

aa<- table(iris.test[,5],model.2)
aa

##             model.2
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor      0         16         0
##   virginica       0          1        17

aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]

## [1] 1
```

```
model.3<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=25)
model.3
```

```
##  [1] virginica  virginica  setosa     versicolor versicolor virginica
##  [7] setosa     virginica  virginica  versicolor virginica  versicolor
## [13] versicolor versicolor virginica  setosa     versicolor versicolor
## [19] setosa     setosa     versicolor virginica  setosa     setosa
## [25] versicolor versicolor virginica  versicolor versicolor virginica
## [31] versicolor versicolor virginica  setosa     versicolor virginica
## [37] virginica  virginica  setosa     versicolor setosa     versicolor
## [43] versicolor virginica  setosa
## Levels: setosa versicolor virginica
```

```
aa<- table(iris.test[,5],model.3)
aa
```

```
##             model.3
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor      0         16         0
##   virginica       0          3        15
```

```
aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]
```

```
## [1] 3
```

```
model.3n<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=30)
model.3n
```

```
##  [1] virginica  virginica  setosa     versicolor versicolor virginica
##  [7] setosa     virginica  virginica  versicolor virginica  versicolor
## [13] versicolor versicolor virginica  setosa     virginica  versicolor
## [19] setosa     setosa     versicolor virginica  setosa     setosa
## [25] versicolor versicolor virginica  versicolor versicolor virginica
## [31] versicolor versicolor virginica  setosa     versicolor virginica
## [37] virginica  virginica  setosa     versicolor setosa     versicolor
## [43] versicolor virginica  setosa
## Levels: setosa versicolor virginica
```

```
aa<- table(iris.test[,5],model.3n)
aa
```

```
##             model.3n
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor      0         16         0
##   virginica       0          2        16
```

```
aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]
```

```
## [1] 2
```

```r
model.4<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=55)
model.4
```

```
##  [1] virginica  virginica  setosa     versicolor versicolor virginica 
##  [7] setosa     virginica  virginica  versicolor virginica  versicolor
## [13] versicolor versicolor virginica  setosa     versicolor versicolor
## [19] setosa     setosa     versicolor versicolor setosa     setosa    
## [25] versicolor versicolor virginica  versicolor versicolor virginica 
## [31] versicolor versicolor virginica  setosa     versicolor virginica 
## [37] virginica  virginica  setosa     versicolor setosa     versicolor
## [43] versicolor versicolor setosa    
## Levels: setosa versicolor virginica
```

```r
aa<-table(iris.test[,5],model.4)
aa
```

```
##             model.4
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor      0         16         0
##   virginica       0          5        13
```

```r
aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]
```

```
## [1] 5
```

```r
model.5<-
knn(train=iris.train[,1:4],test=iris.test[,1:4],cl=iris.train[,5],k=100)
model.5
```

```
##  [1] versicolor versicolor setosa     setosa     versicolor versicolor
##  [7] setosa     setosa     setosa     versicolor versicolor versicolor
## [13] setosa     setosa     setosa     setosa     setosa     setosa    
## [19] setosa     setosa     setosa     setosa     setosa     setosa    
## [25] setosa     setosa     setosa     versicolor setosa     versicolor
## [31] setosa     setosa     setosa     setosa     setosa     versicolor
## [37] setosa     setosa     setosa     setosa     setosa     setosa    
## [43] setosa     versicolor setosa    
## Levels: setosa versicolor virginica
```

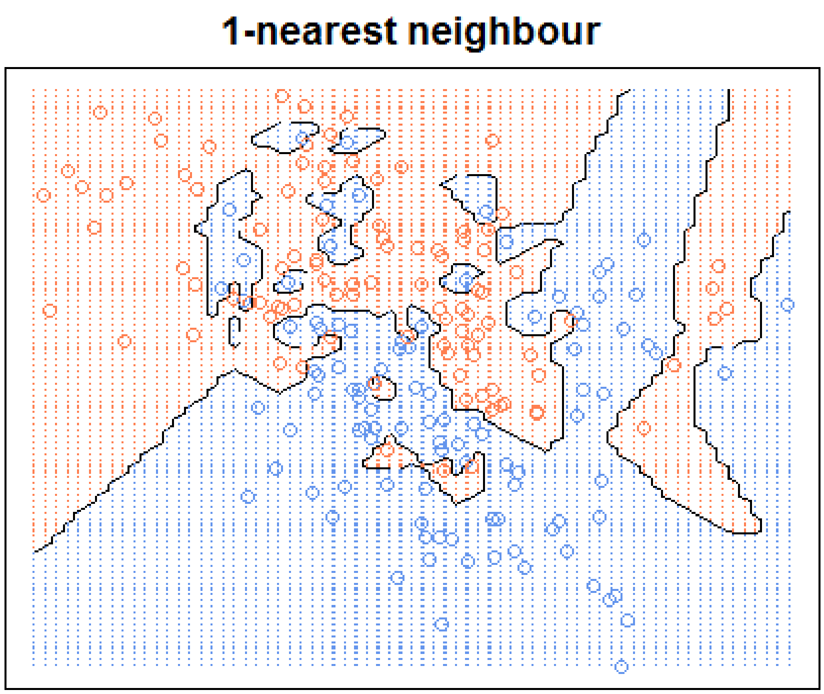
```r
aa<-table(iris.test[,5],model.5)
aa
```

```
##             model.5
##              setosa versicolor virginica
##   setosa         11          0         0
##   versicolor     12          4         0
##   virginica      11          7         0
```

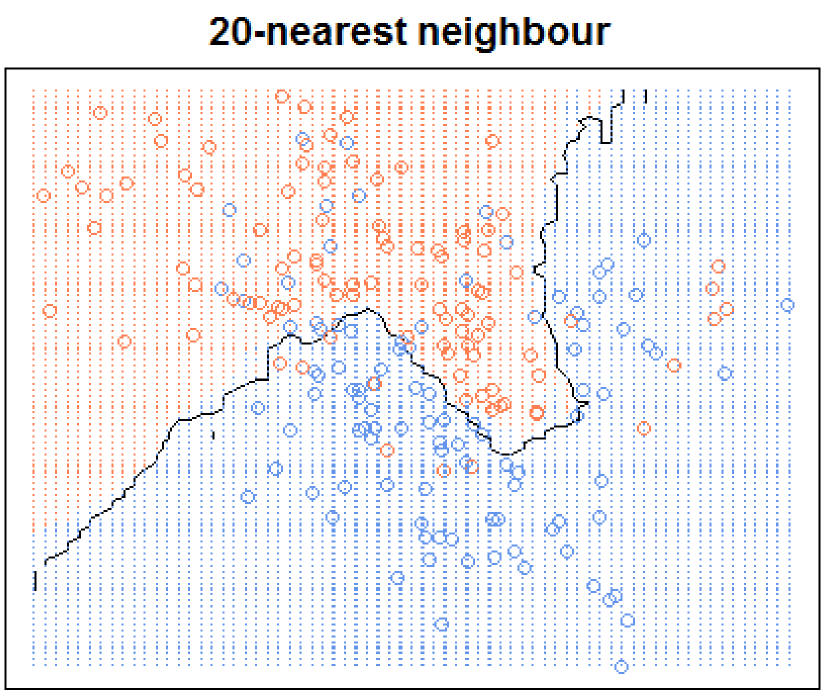
```r
aa[1,2]+aa[1,3]+aa[2,1]+aa[2,3]+aa[3,1]+aa[3,2]

## [1] 30
```

Boundary of Nearest Neighbor:

```r
library(ElemStatLearn)
require(class)
x <- mixture.example$x
g <- mixture.example$y
xnew <- mixture.example$xnew
mod15 <- knn(x, xnew, g, k=1, prob=TRUE)
prob <- attr(mod15, "prob")
prob <- ifelse(mod15=="1", prob, 1-prob)
px1 <- mixture.example$px1
px2 <- mixture.example$px2
prob15 <- matrix(prob, length(px1), length(px2))
par(mar=rep(2,4))
contour(px1, px2, prob15, levels=0.5, labels="", xlab="", ylab="", main=
          "1-nearest neighbour", axes=FALSE)
points(x, col=ifelse(g==1, "coral", "cornflowerblue"))
gd <- expand.grid(x=px1, y=px2)
points(gd, pch=".", cex=1.2, col=ifelse(prob15>0.5, "coral",
"cornflowerblue"))
box()
```



```r
mod15 <- knn(x, xnew, g, k=20, prob=TRUE)
prob <- attr(mod15, "prob")
prob <- ifelse(mod15=="1", prob, 1-prob)
px1 <- mixture.example$px1
px2 <- mixture.example$px2
prob15 <- matrix(prob, length(px1), length(px2))
par(mar=rep(2,4))
contour(px1, px2, prob15, levels=0.5, labels="", xlab="", ylab="", main=
          "20-nearest neighbour", axes=FALSE)
points(x, col=ifelse(g==1, "coral", "cornflowerblue"))
gd <- expand.grid(x=px1, y=px2)
points(gd, pch=".", cex=1.2, col=ifelse(prob15>0.5, "coral",
"cornflowerblue"))
box()
```




```r
mod15 <- knn(x, xnew, g, k=50, prob=TRUE)
prob <- attr(mod15, "prob")
prob <- ifelse(mod15=="1", prob, 1-prob)
px1 <- mixture.example$px1
px2 <- mixture.example$px2
prob15 <- matrix(prob, length(px1), length(px2))
par(mar=rep(2,4))
contour(px1, px2, prob15, levels=0.5, labels="", xlab="", ylab="", main=
          "50-nearest neighbour", axes=FALSE)
points(x, col=ifelse(g==1, "coral", "cornflowerblue"))
gd <- expand.grid(x=px1, y=px2)
points(gd, pch=".", cex=1.2, col=ifelse(prob15>0.5, "coral",
```

```r
"cornflowerblue"))
box()
```