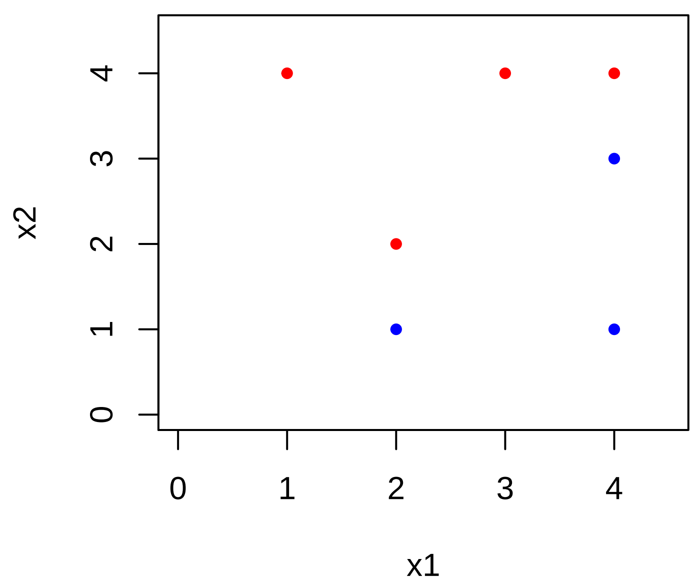
STATS 415 Homework 10 Solutions

April 5, 2018

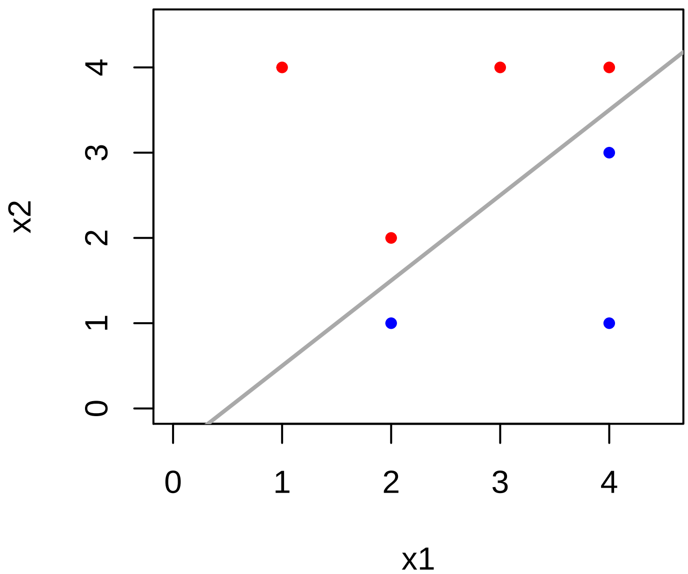
1. **Conceptual question 3, Section 9.7.** Here we explore the maximal margin classifier on a toy data set:

x1	x2	y
3	4	red
2	2	red
4	4	red
1	4	red
2	1	blue
4	3	blue
4	1	blue

(a) [1pt] We are given $n = 7$ observations in $p = 2$ dimensions. For each observation there is an associated class label. Sketch the observations.



(b) [2pt] Sketch the optimal separating hyperplane and provide the equation for this hyperplane.



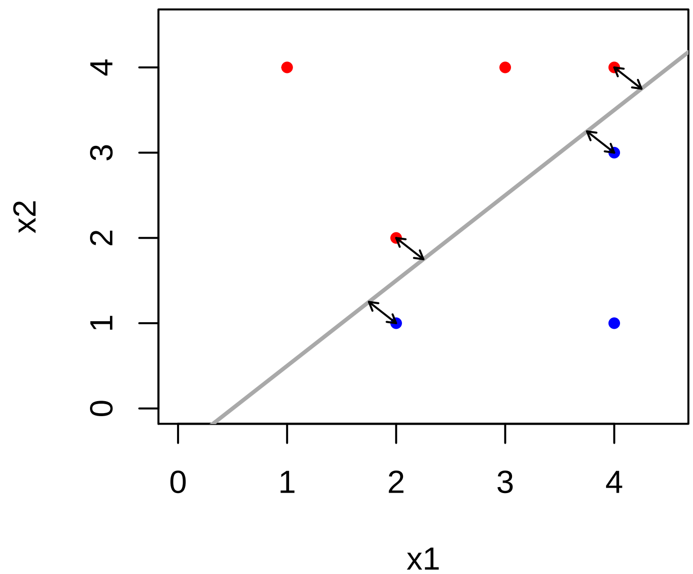
The equation of the hyperplane is $X_2 = -0.5 + X_1$.

(c) [2pts] Describe the classification rule for the maximal margin classifier. Provide the values for β_0, β_1, and β_2.

From above, we have that a suitable choice of βs is $\beta_0 = -0.5$, $\beta_1 = 1$, $\beta_2 = -1$. Recall the constraint that $\|\beta\| = 1$, so we divide by the norm, which, here, is 1.5.

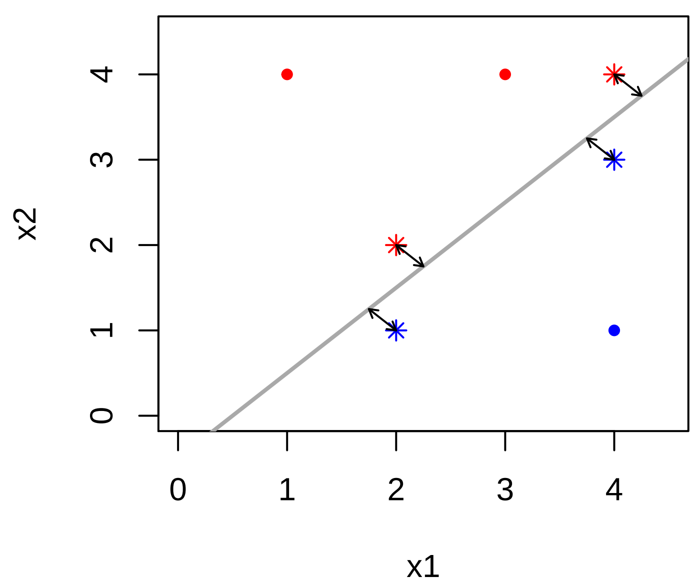
The classification rule states that if $\frac{2}{3}X_1 - \frac{2}{3}X_2 - \frac{1}{3} > 0$, classify to "blue". Otherwise, classify to "red". Equivalently, if $\frac{2}{3}X_2 - \frac{2}{3}X_1 + \frac{1}{3} > 0$, classify to "red".

(d) [1pt] On your sketch, indicate the margin for the maximal margin hyperplane.



The margin is the length of the arrows in the picture above.

(e) [1pt] Indicate support vectors for the maximal margin classifier.

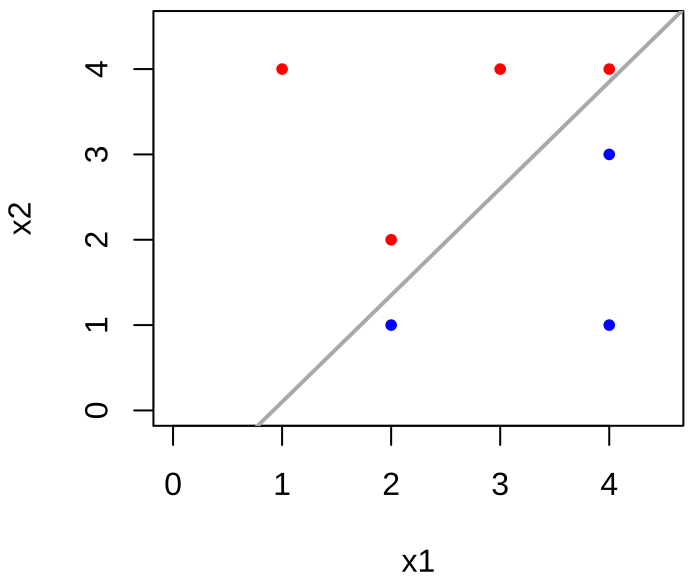


The support vectors are indicated by stars.

(f) [1pt] Argue that a slight movement of the seventh observation would not affect the maximal margin hyperplane.

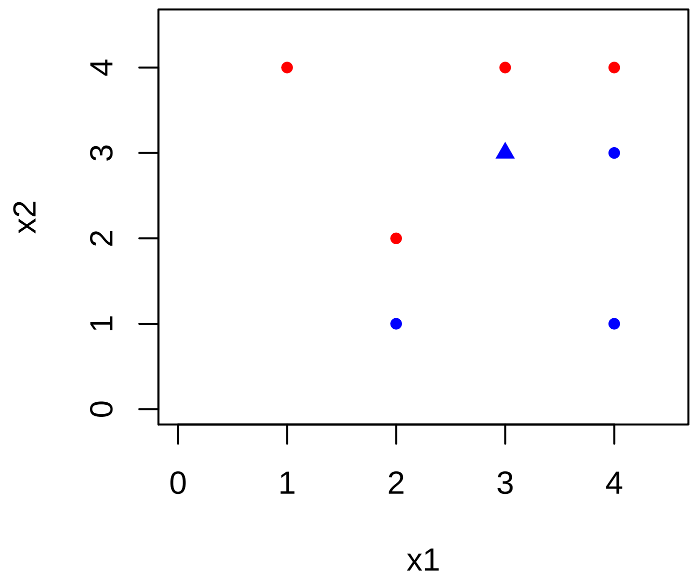
The maximal margin hyperplane is determined exclusively by support vectors; the seventh observation is not a support vector.

(g) [2pt] Sketch a hyperplane that is not the optimal separating hyperplane, and provide the equation for this hyperplane.



The gray line in the figure above is clearly a separating hyperplane; however, its margin (the smallest distance from a support vector to the hyperplane) is clearly much smaller than the optimal hyperplane's. The equation of the hyperplane shown above is $0.51X_2 - 0.64X_2 + 0.58 = 0$.

(h) [1pt] Draw an additional observation on the plot so that the two classes are no longer separable by a hyperplane.



The new observation is the blue triangle at $X_1 = 3$, $X_2 = 3$.

2. **Data Analysis**

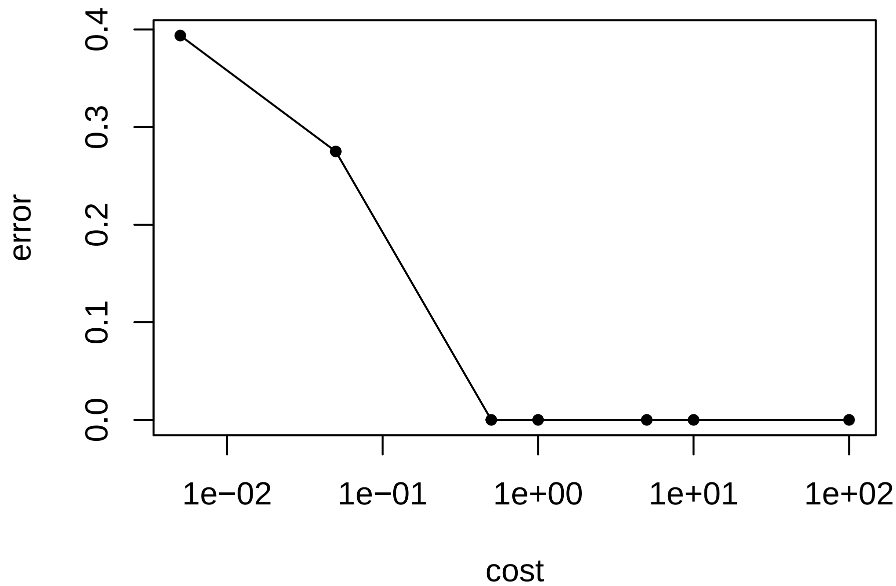
```
library(e1071)
set.seed(45678)
attach(crabs)

blueMale = which(sp == "B" & sex == "M")
orangeMale = which(sp == "O" & sex == "M")
blueFemale = which(sp == "B" & sex == "F")
orangeFemale = which(sp == "O" & sex == "F")
train_id = c(sample(blueMale, size = trunc(0.80 * length(blueMale))),
sample(orangeMale, size = trunc(0.80 * length(orangeMale))),
sample(blueFemale, size = trunc(0.80 * length(blueFemale))),
sample(orangeFemale, size = trunc(0.80 * length(orangeFemale))))
crabs_train = crabs[train_id, ]
crabs_test = crabs[-train_id,]
```

(a) [5pts] Fit a linear support vector machine to the data with various values of `cost`, in order to predict Species from the five numerical measurements. Omit the variable Sex for this homework. Report the cross-validation errors associated with different values of `cost`. Comment on your results and make some relevant plots.

```
costVals <- c(.005, .05, .5, 1, 5, 10, 100)
svm1 <- tune(svm, as.factor(sp) ~ FL + RW + CL + CW + BD, data = crabs_train,
             kernel = "linear",
             ranges = list("cost" = costVals))
kable(svm1$performances[, c("cost", "error")])
```

cost	error
5e-03	0.39375
5e-02	0.27500
5e-01	0.00000
1e+00	0.00000
5e+00	0.00000
1e+01	0.00000
1e+02	0.00000

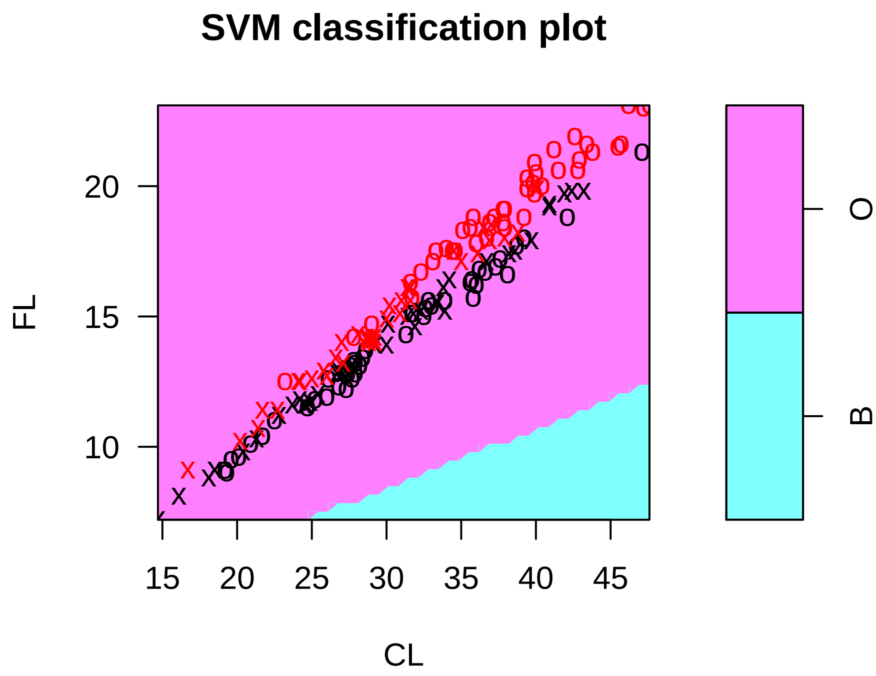
```r
with(svm1$performances, plot(error ~ cost, type = "o", pch = 20, log = "x"),
     xlab = "log(cost)")
```



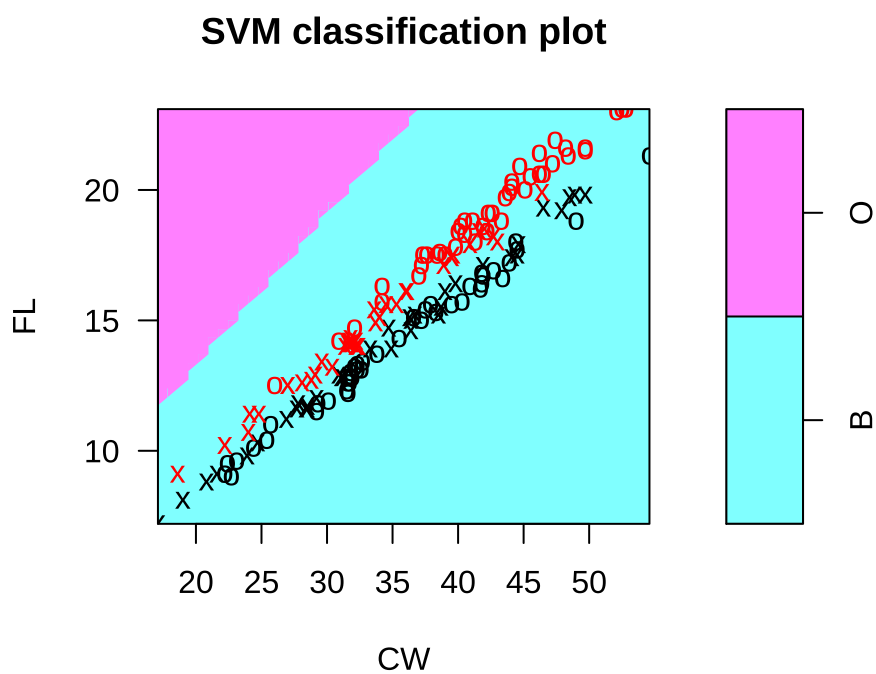
The optimal value of cost (per 10-fold CV) is 0.5. This gives the widest margin with the lowest training error. Since the training error is zero, this suggests that the classes are linearly separable within the training data.

We can also examine pairwise SVM plots.

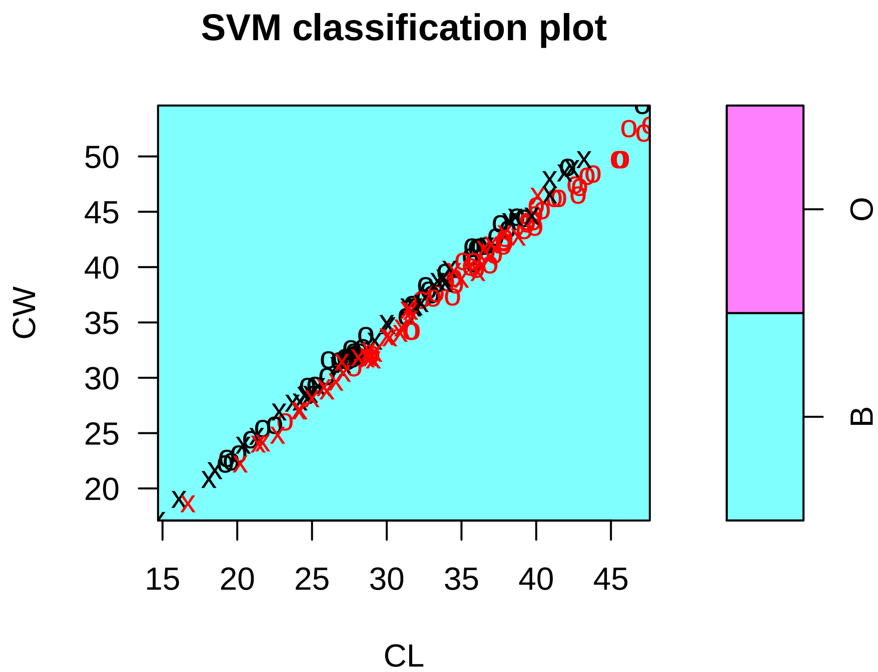
```r
plot(svm1$best.model, data = crabs_train, formula = FL ~ CL)
```



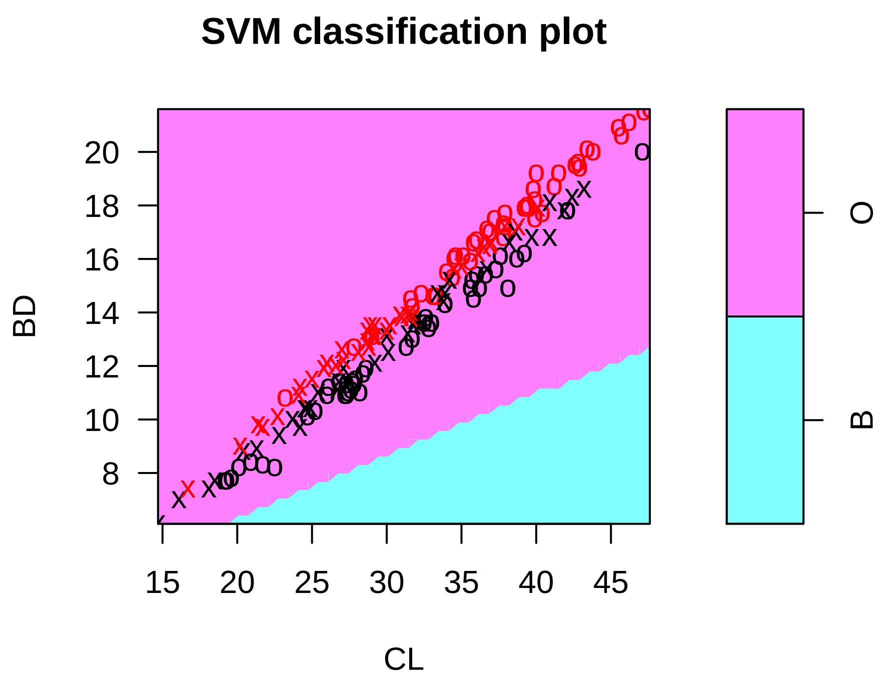

```
plot(svm1$best.model, data = crabs_train, formula = FL ~ CW)
```



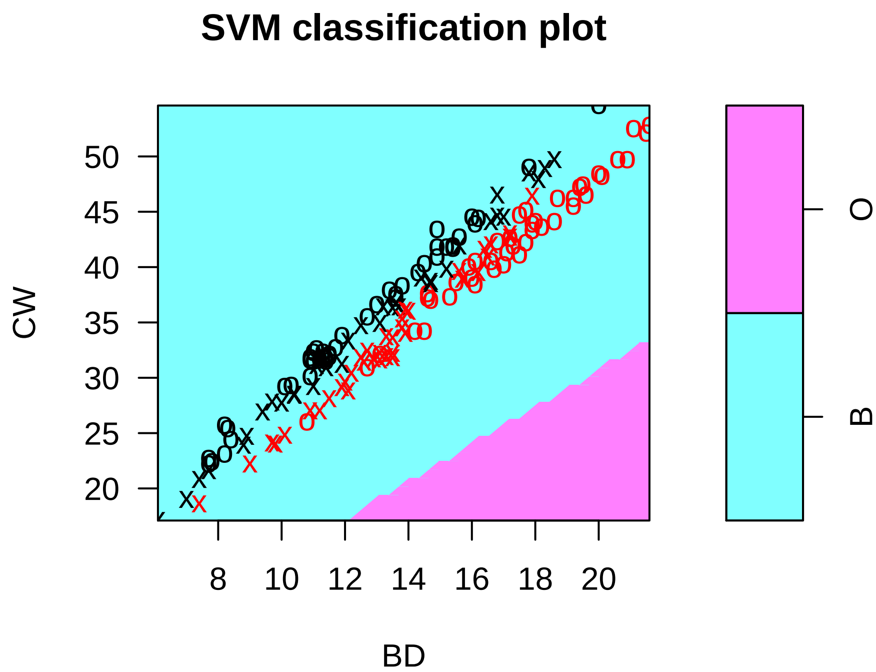

```
plot(svm1$best.model, data = crabs_train, formula = CW ~ CL)
```




```r
plot(svm1$best.model, data = crabs_train, formula = BD ~ CL)
```




```r
plot(svm1$best.model, data = crabs_train, formula = CW ~ BD)
```



The plots confirm our hypothesis that the classes are linearly separable: in particular, the classes are separable when examining FL, CL, CW, and BD, pairwise. Note that the decision boundary does not seem to separate points because of how projection is done.

Test error:

```
testPred <- predict(svm1$best.model, crabs_test)
table(testPred, crabs_test$sp)
```

```
##
## testPred  B  O
##        B 20  0
##        O  0 20
```

Test error is also zero.

(b) [5pts] Fit nonlinear SVMs with radial and polynomial kernels, with different values of `gamma` and `degree` and `cost`. Report the cross-validation errors associated with different values of `cost`. Comment on your results and make some relevant plots.

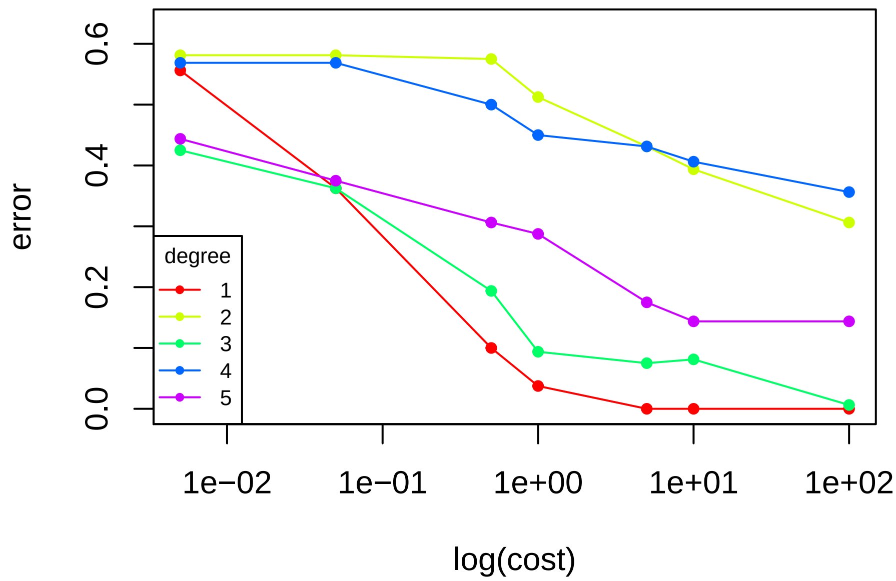
```
svm2 <- tune(svm, as.factor(sp) ~ FL + CL + CW + BD + RW, data = crabs_train,
             kernel = "polynomial",
             ranges = list("cost" = costVals, degree = 1:5))
```

```
kable(svm2$performances[, -4])
```

cost	degree	error
5e-03	1	0.55625
5e-02	1	0.36250
5e-01	1	0.10000
1e+00	1	0.03750
5e+00	1	0.00000

cost	degree	error
1e+01	1	0.00000
1e+02	1	0.00000
5e-03	2	0.58125
5e-02	2	0.58125
5e-01	2	0.57500
1e+00	2	0.51250
5e+00	2	0.43125
1e+01	2	0.39375
1e+02	2	0.30625
5e-03	3	0.42500
5e-02	3	0.36250
5e-01	3	0.19375
1e+00	3	0.09375
5e+00	3	0.07500
1e+01	3	0.08125
1e+02	3	0.00625
5e-03	4	0.56875
5e-02	4	0.56875
5e-01	4	0.50000
1e+00	4	0.45000
5e+00	4	0.43125
1e+01	4	0.40625
1e+02	4	0.35625
5e-03	5	0.44375
5e-02	5	0.37500
5e-01	5	0.30625
1e+00	5	0.28750
5e+00	5	0.17500
1e+01	5	0.14375
1e+02	5	0.14375

```r
with(subset(svm2$performances, degree == 1),
     plot(error ~ cost, type = "o", col = rainbow(5)[1], pch = 20,
          log = "x", xlab = "log(cost)", ylim = c(0, max(svm2$performances$error) + .05)))
with(subset(svm2$performances, degree == 2),
     lines(error ~ cost, type = "o", col = rainbow(5)[2], pch = 20))
with(subset(svm2$performances, degree == 3),
     lines(error ~ cost, type = "o", col = rainbow(5)[3], pch = 20))
with(subset(svm2$performances, degree == 4),
     lines(error ~ cost, type = "o", col = rainbow(5)[4], pch = 20))
with(subset(svm2$performances, degree == 5),
     lines(error ~ cost, type = "o", col = rainbow(5)[5], pch = 20))
legend("bottomleft", legend = 1:5, lty = 1, pch = 20, col = rainbow(5),
       title = "degree", cex = .7)
```



The best polynomial SVM according to 10-fold CV has degree 1 and cost 5. This produces a classifier that is identical to the (linear) support vector classifier above!

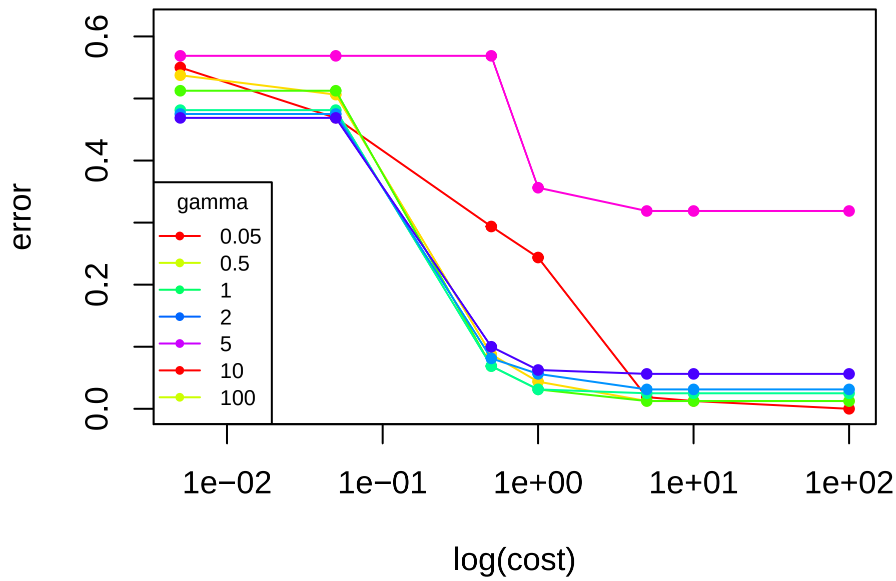
```
gammaVals <- c(.05, .5, 1, 2, 5, 10, 100)
svm3 <- tune(svm, as.factor(sp) ~ FL + CL + CW + BD + RW, data = crabs_train,
             kernel = "radial",
             ranges = list("cost" = costVals, gamma = gammaVals))
```

```
kable(svm3$performances[, -4])
```

cost	gamma	error
5e-03	5e-02	0.55000
5e-02	5e-02	0.46875
5e-01	5e-02	0.29375
1e+00	5e-02	0.24375
5e+00	5e-02	0.01875
1e+01	5e-02	0.01250
1e+02	5e-02	0.00000
5e-03	5e-01	0.53750
5e-02	5e-01	0.50625
5e-01	5e-01	0.08750
1e+00	5e-01	0.04375
5e+00	5e-01	0.01250
1e+01	5e-01	0.01250
1e+02	5e-01	0.01250
5e-03	1e+00	0.51250
5e-02	1e+00	0.51250
5e-01	1e+00	0.06875
1e+00	1e+00	0.03125
5e+00	1e+00	0.01250

cost	gamma	error
1e+01	1e+00	0.01250
1e+02	1e+00	0.01250
5e-03	2e+00	0.48125
5e-02	2e+00	0.48125
5e-01	2e+00	0.06875
1e+00	2e+00	0.03125
5e+00	2e+00	0.02500
1e+01	2e+00	0.02500
1e+02	2e+00	0.02500
5e-03	5e+00	0.47500
5e-02	5e+00	0.47500
5e-01	5e+00	0.08125
1e+00	5e+00	0.05625
5e+00	5e+00	0.03125
1e+01	5e+00	0.03125
1e+02	5e+00	0.03125
5e-03	1e+01	0.46875
5e-02	1e+01	0.46875
5e-01	1e+01	0.10000
1e+00	1e+01	0.06250
5e+00	1e+01	0.05625
1e+01	1e+01	0.05625
1e+02	1e+01	0.05625
5e-03	1e+02	0.56875
5e-02	1e+02	0.56875
5e-01	1e+02	0.56875
1e+00	1e+02	0.35625
5e+00	1e+02	0.31875
1e+01	1e+02	0.31875
1e+02	1e+02	0.31875

```r
with(subset(svm3$performances, gamma == gammaVals[1]),
     plot(error ~ cost, type = "o", col = rainbow(length(gammaVals))[1], pch = 20,
          log = "x", xlab = "log(cost)", ylim = c(0, max(svm3$performances$error) + .05)))
invisible(lapply(2:length(gammaVals), function(i) {
  with(subset(svm3$performances, gamma == gammaVals[i]),
       lines(error ~ cost, type = "o", col = rainbow(length(gammaVals))[i], pch = 20))
}))
legend("bottomleft", legend = gammaVals, lty = 1, pch = 20, col = rainbow(5),
       title = "gamma", cex = .7)
```



The optimal gamma is 0.05 and the best choice of cost is 100, chosen by 10-fold cross-validation. This produces a training error of 0. We compute test error:

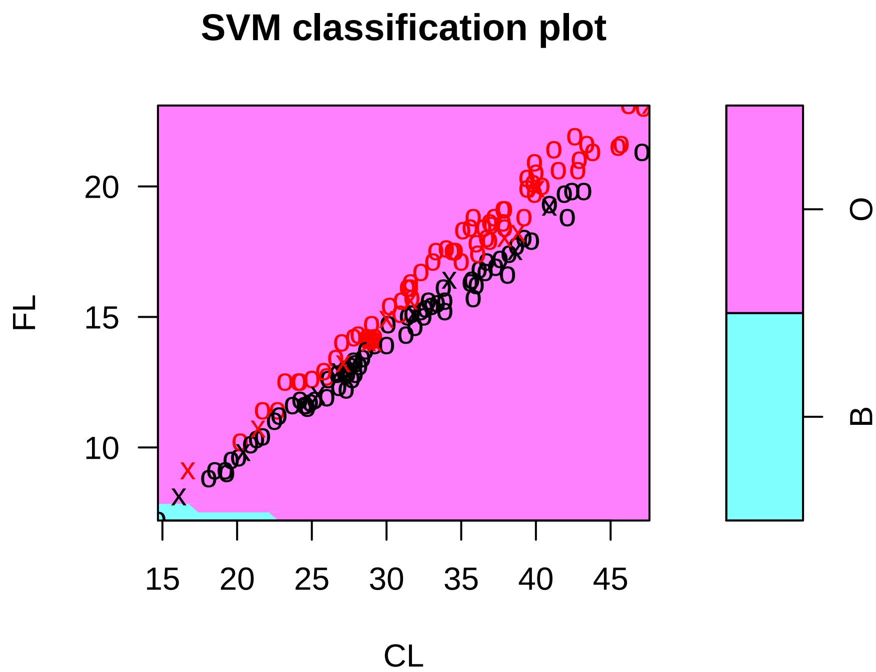
```
testPred <- predict(svm3$best.model, crabs_test)
table(testPred, crabs_test$sp)
```

```
##
## testPred  B  O
##        B 20  0
##        O  0 20
```

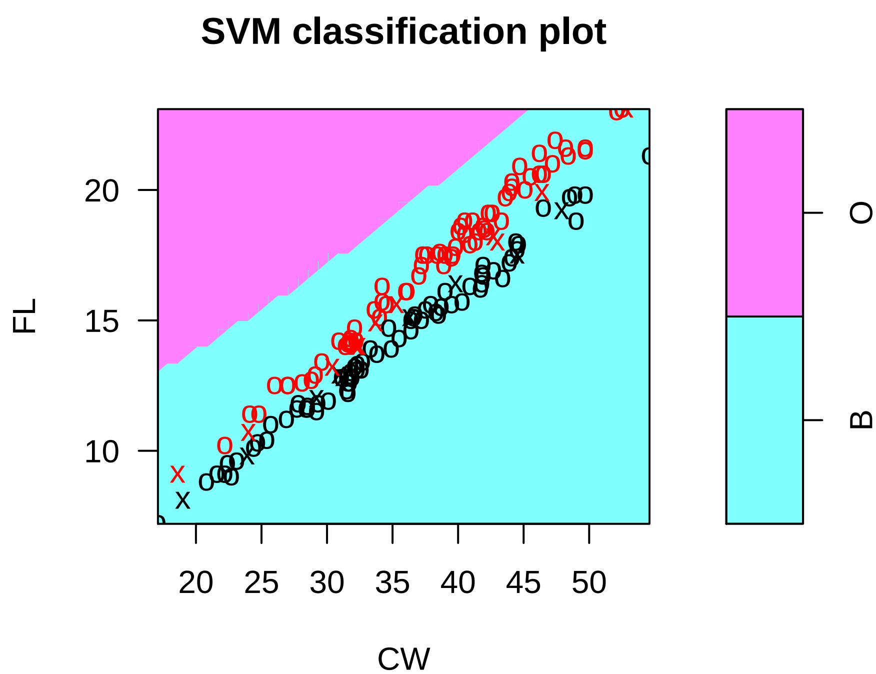
As above, we have zero test error.

We reproduce the plots from above.

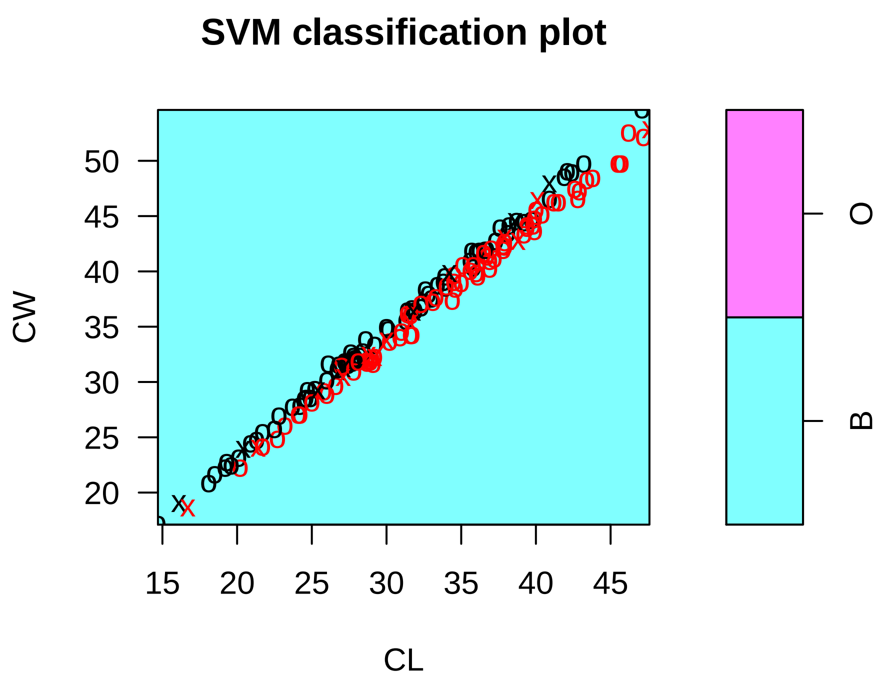
```
plot(svm3$best.model, data = crabs_train, formula = FL ~ CL)
```



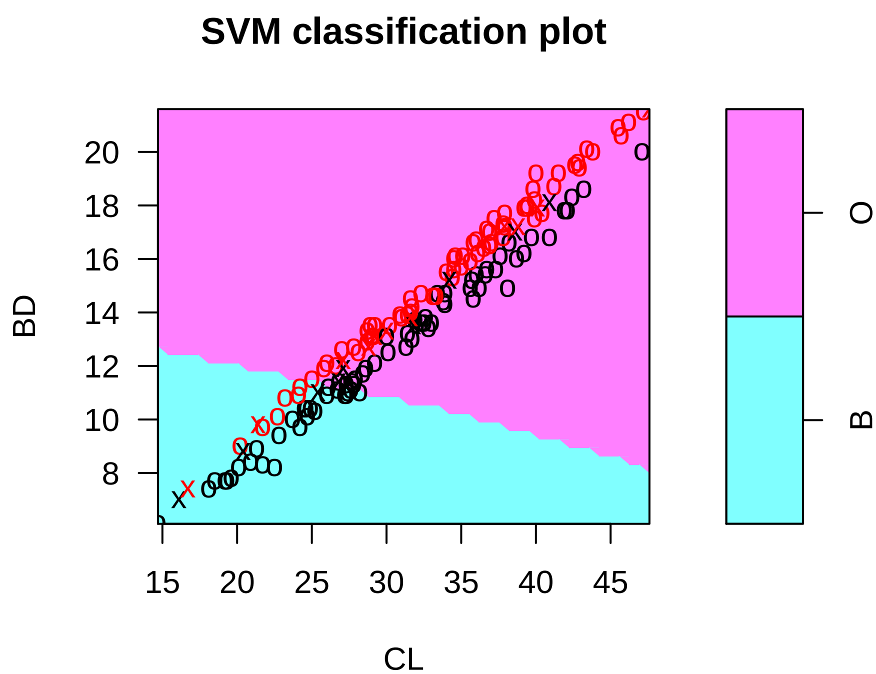

```
plot(svm3$best.model, data = crabs_train, formula = FL ~ CW)
```



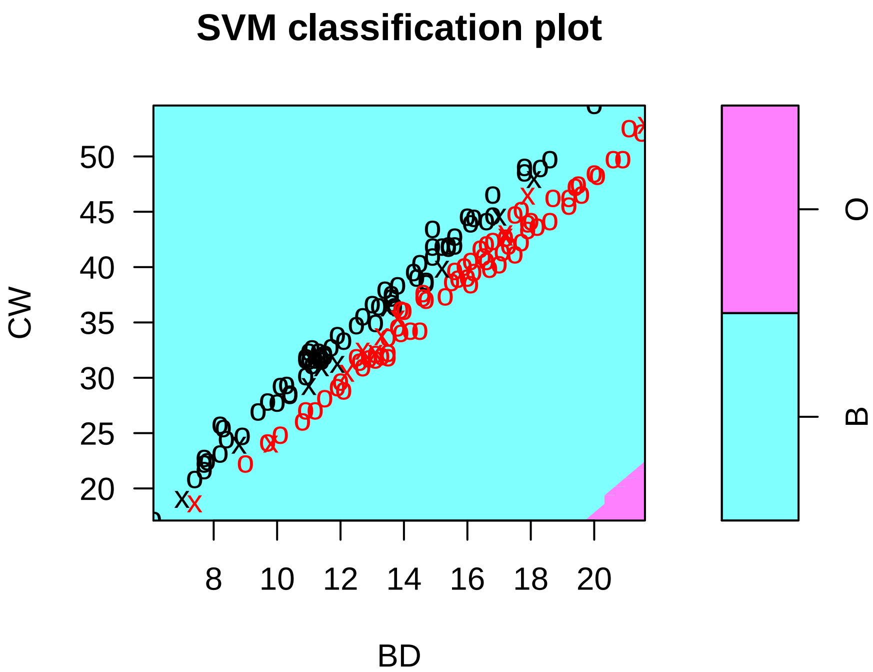

```
plot(svm3$best.model, data = crabs_train, formula = CW ~ CL)
```




```
plot(svm3$best.model, data = crabs_train, formula = BD ~ CL)
```




```
plot(svm3$best.model, data = crabs_train, formula = CW ~ BD)
```



We see that the SVM boundary is near-linear, which supports the previous claim that the classes are linearly separable.