HW06

Zixin Ouyang

10/26/2017

```
wisc_train<-read.csv('wisc-trn.csv')
wisc_test<-read.csv('wisc-tst.csv')
```

```
library(caret)
```

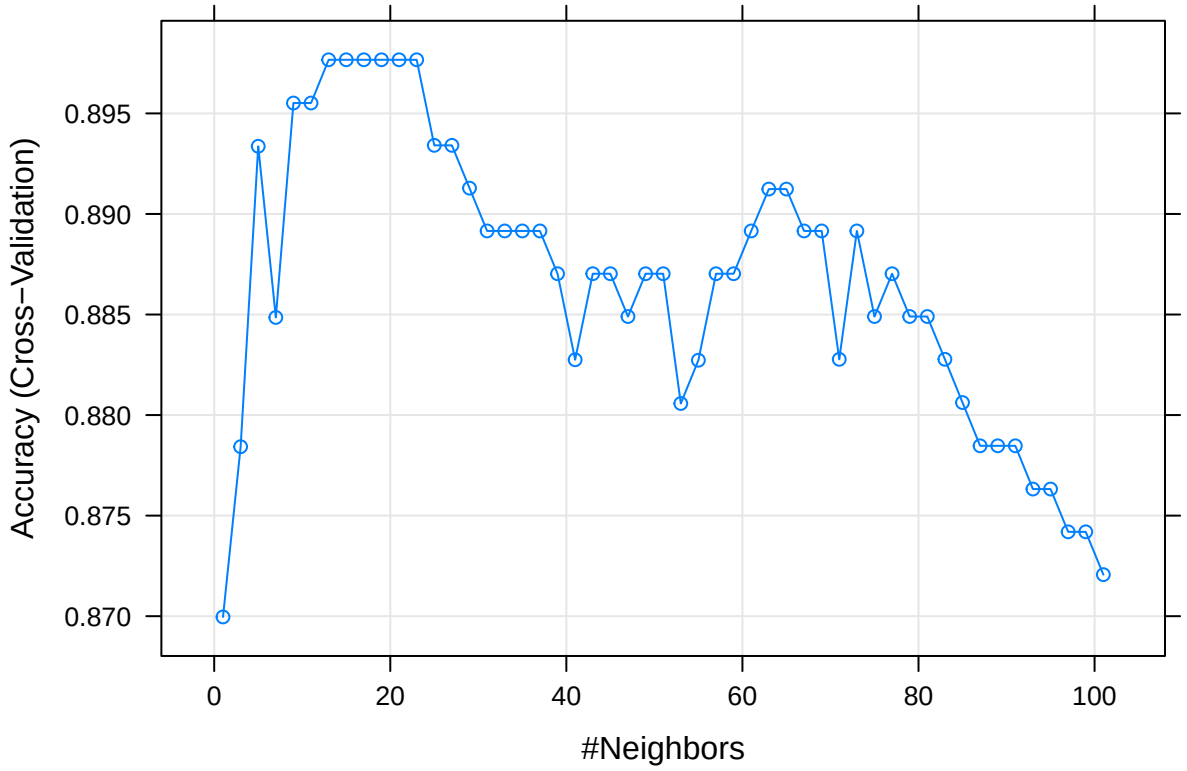
```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

Exercise 1

```
set.seed(1337)
knn_mod = train(
  class ~ .,
  data = wisc_train,
  method = "knn",
  trControl = trainControl(method = "cv", number = 5),
  tuneGrid = expand.grid(k = seq(1, 101, by = 2))
)
```

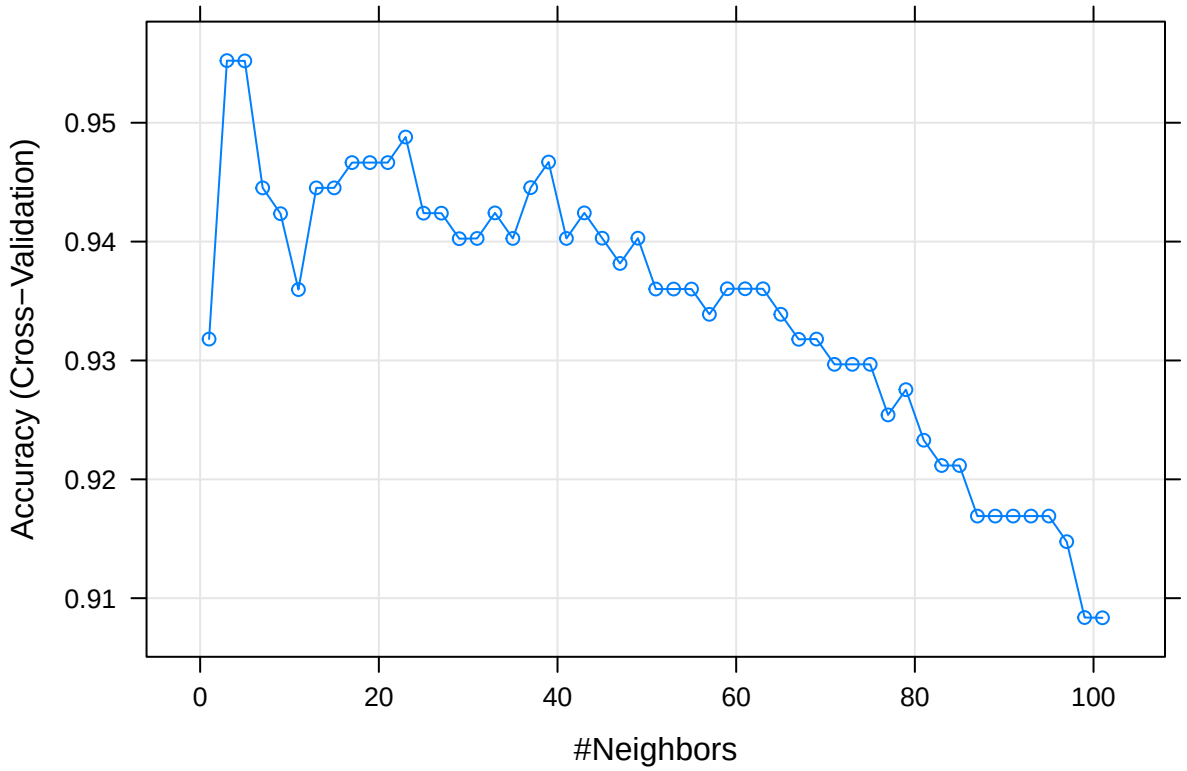
```
plot(knn_mod)
```



Exercise 2

```
set.seed(1337)
knn_mod2 = train(
  class ~ .,
  data = wisc_train,
  method = "knn",
  trControl = trainControl(method = "cv", number = 5),
  preProcess = c("center", "scale"),
  tuneGrid = expand.grid(k = seq(1, 101, by = 2))
)
```

```
plot(knn_mod2)
```



Exercise 3

```
library(randomForest)
```

```
set.seed(1337)
rf_mod = train(
  class ~ .,
  data = wisc_train,
  method = "rf",
  trControl = trainControl(method = "cv", number = 5),
  tuneGrid = expand.grid(mtry = seq(1, 10, by = 1))
)
```

mtry	Accuracy
1	0.9402654
2	0.9466484
3	0.9402654
4	0.9487989
5	0.9403111
6	0.9403111
7	0.9381835
8	0.9424388
9	0.9403111
10	0.9381835

Exercise 4

```
get_best_result = function(caret_fit) {
  best = which(rownames(caret_fit$results) == rownames(caret_fit$bestTune))
  best_result = caret_fit$results[best, ]
  rownames(best_result) = NULL
  best_result
}
```

```
test_pred1 = predict(knn_mod, newdata = wisc_test)
```

```
test_pred2 = predict(knn_mod2, newdata = wisc_test)
```

```
prob = predict(rf_mod, newdata = wisc_test, type = "prob")[10, 2]
```

```
test_pred3 = predict(rf_mod, newdata = wisc_test)
result = confusionMatrix(test_pred3, wisc_test$class, positive = "M")
```

```
calc_acc = function(actual, predicted) {
  mean(actual == predicted)
}
```

```
calc_acc(actual = wisc_test$class, predicted = test_pred3)
```

```
## [1] 0.93
```

Question	Answer
(a)	23
(b)	0.8976664
(c)	0.86
(d)	3
(e)	0.9552276
(f)	0.88
(g)	KNN is performing better with predictor scaling
(h)	4
(i)	0.04
(j)	0.875
(k)	0.9666667
(l)	the random forest is performing better