HW02

Zixin Ouyang

9/22/2017

```
hw02_train<-read.csv('hw02-train-data.csv')
hw02_test<-read.csv('hw02-test-data.csv')
```

Exercise 1

```
fit_poly = function(degree = 1, data) {
lm(y ~ poly(x, degree = degree), data = data)
}
```

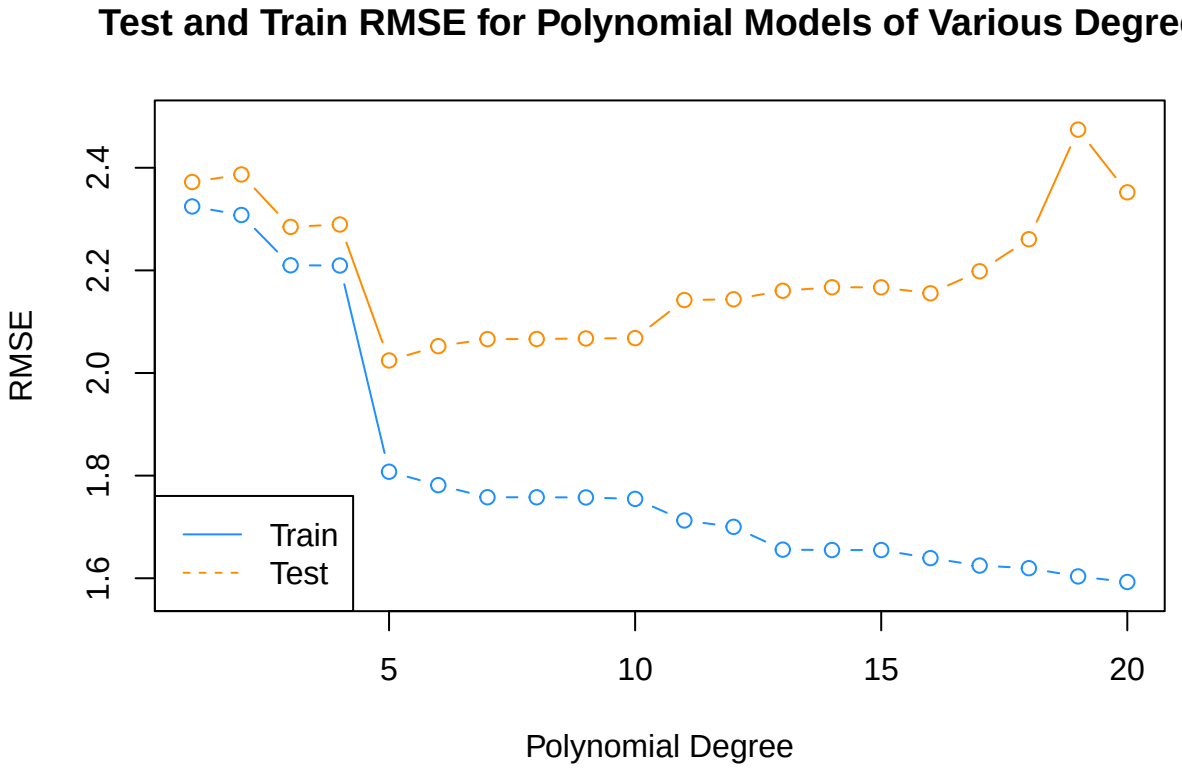
```
rmse = function(actual, predicted) {
  sqrt(mean((actual - predicted) ^ 2))
}
```

```
get_rmse = function(model, data, response) {
  rmse(actual = data[, response],
       predicted = predict(model, data))
}
```

```
degrees = 1:20
model = lapply(degrees, fit_poly, data = hw02_train)
```

```
trn_rmse = sapply(model, get_rmse, data = hw02_train, response = "y")
tst_rmse = sapply(model, get_rmse, data = hw02_test, response = "y")
```

```
plot(degrees, trn_rmse, type = "b",
     ylim = c(min(c(trn_rmse, tst_rmse)) - 0.02,
              max(c(trn_rmse, tst_rmse)) + 0.02),
     col = "dodgerblue",
     xlab = "Polynomial Degree",
     ylab = "RMSE",
     main = "Test and Train RMSE for Polynomial Models of Various Degrees")
lines(degrees, tst_rmse, type = "b", col = "darkorange")
legend("bottomleft", c("Train", "Test"), lty = c(1, 2),
       col = c("dodgerblue", "darkorange"))
```




```
which.min(tst_rmse)
```

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

The model with degree of polynomial equal to 5 has lowest test RMSE, so it performs the best. Underfitting models: models with degree of polynomial less than 5. Overfitting models: models with degree of polynomial larger than 5.

Exercise 2

```
library(FNN)
k = seq(5, 50, by = 5)
```

```
make_knn_pred = function(k = 1, training, predicting) {
  pred = FNN::knn.reg(train = training["x"],
                      test = predicting["x"],
                      y = training$y, k = k)$pred
  act  = predicting$y
  rmse(predicted = pred, actual = act)
}
```

```
knn_trn_rmse = sapply(k, make_knn_pred,
                      training = hw02_train,
                      predicting = hw02_train)
```

```
knn_tst_rmse = sapply(k, make_knn_pred,
                      training = hw02_train,
                      predicting = hw02_test)
```

```
best_k = k[which.min(knn_tst_rmse)]
```

```
fit_status = ifelse(k < best_k, "Over", ifelse(k == best_k, "Best", "Under"))
```

```
knn_results = data.frame(
  k,
  round(knn_trn_rmse, 2),
  round(knn_tst_rmse, 2),
  fit_status)
colnames(knn_results) = c("K", "Train RMSE", "Test RMSE", "Fitting")
```

```
knitr::kable(knn_results)
```

K	Train RMSE	Test RMSE	Fitting
5	1.65	2.16	Over
10	1.70	2.08	Over
15	1.79	2.05	Best
20	1.93	2.06	Under
25	2.02	2.14	Under
30	2.28	2.36	Under
35	2.60	2.67	Under
40	2.96	2.99	Under
45	3.27	3.29	Under
50	3.58	3.57	Under