Annex Chapter 4

Juan Pablo Bertucci

28 November 2019

Input data

The data collected for the section defined is imported below

```
data=read_csv(file = "C:/Users/Juan Pablo/OneDrive - University of Illinois - Urbana/Fall 2019/CEE 508 -
data=as_tibble(data)

view(data)
```

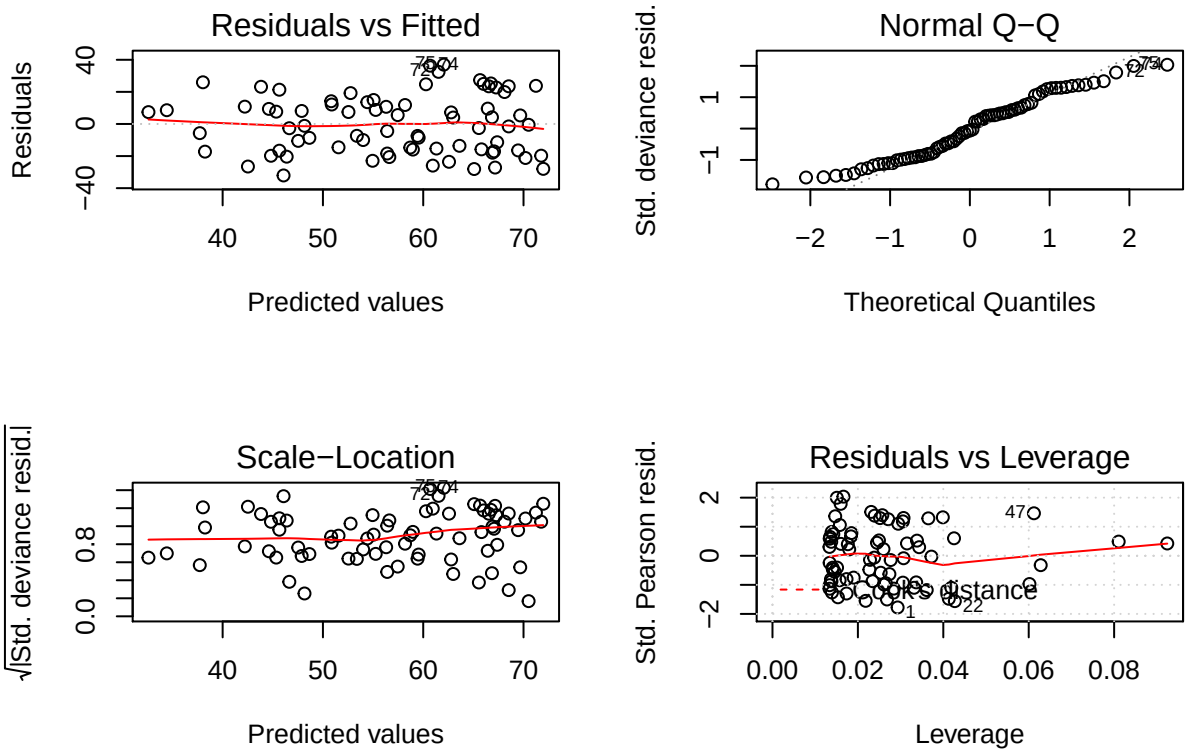
Linear Model

```
fit_lm=glm(pci~iri, data = data)

summary(fit_lm)
```

```
##
## Call:
## glm(formula = pci ~ iri, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -32.074  -16.218   -1.163   12.847   36.949
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   90.558      7.364  12.297  < 2e-16 ***
## iri          -10.442      2.199  -4.748 9.98e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 335.5585)
##
##     Null deviance: 32061  on 74  degrees of freedom
## Residual deviance: 24496  on 73  degrees of freedom
## AIC: 653
##
## Number of Fisher Scoring iterations: 2
```

```
par(mfrow=c(2,2))
plot(fit_lm)
grid()
```



Nearest Neighbors

```
fit_knn = train(
  pci ~ iri,
  data = data,
  method = "knn",
  trControl = trainControl(method='cv',number = 5)
  )

fit_knn
```

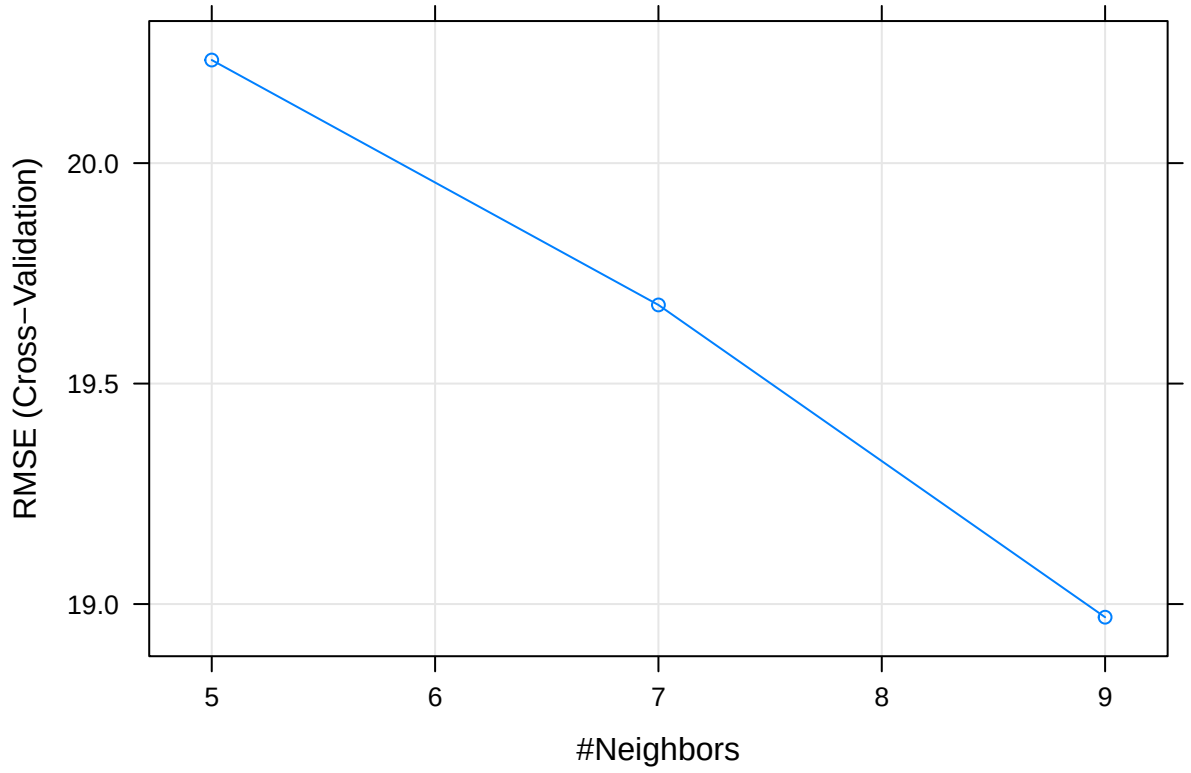
```
## k-Nearest Neighbors
##
## 75 samples
##  1 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 60, 60, 59, 61, 60
## Resampling results across tuning parameters:
##
##   k  RMSE      Rsquared   MAE
##   5  20.23381  0.1606776  17.43074
```

```
##   7  19.67836  0.1662957  17.21735
##   9  18.97016  0.2072669  16.60279
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 9.
```

```
fit_knn$results
```

```
##   k     RMSE  Rsquared      MAE   RMSESD RsquaredSD    MAESD
## 1 5 20.23381 0.1606776 17.43074 2.790018 0.09777436 1.826948
## 2 7 19.67836 0.1662957 17.21735 2.664373 0.06346023 1.930777
## 3 9 18.97016 0.2072669 16.60279 2.429816 0.08283956 2.016656
```

```
plot(fit_knn)
```




```
fit_rf = train(
  pci ~ iri+st,
  data = data,
  method = "rf",
  trControl = trainControl(method='cv',number = 5)
  )

fit_rf
```

```
## Random Forest
##
## 75 samples
##  2 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 60, 60, 61, 59, 60
## Resampling results across tuning parameters:
##
##   mtry  RMSE      Rsquared   MAE
##    2    15.72291  0.5799634  13.27749
##   10    13.73331  0.5990931  11.32807
##   19    13.77414  0.5890353  11.19452
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 10.
```

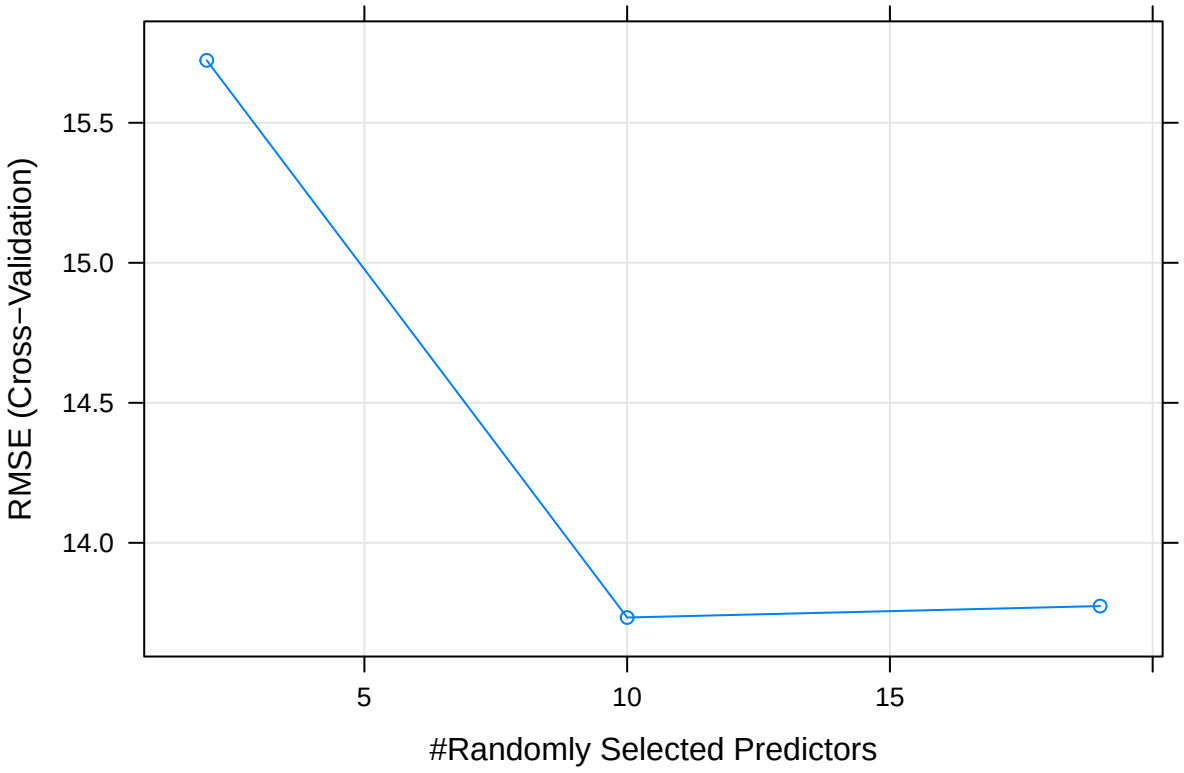
```
fit_rf$results
```

```
##   mtry     RMSE  Rsquared      MAE   RMSESD RsquaredSD    MAESD
## 1    2 15.72291 0.5799634 13.27749 1.258105  0.1642321 0.461746
## 2   10 13.73331 0.5990931 11.32807 2.290333  0.1484029 2.176807
## 3   19 13.77414 0.5890353 11.19452 2.289887  0.1400407 2.156963
```

```
fit_rf$finalModel
```

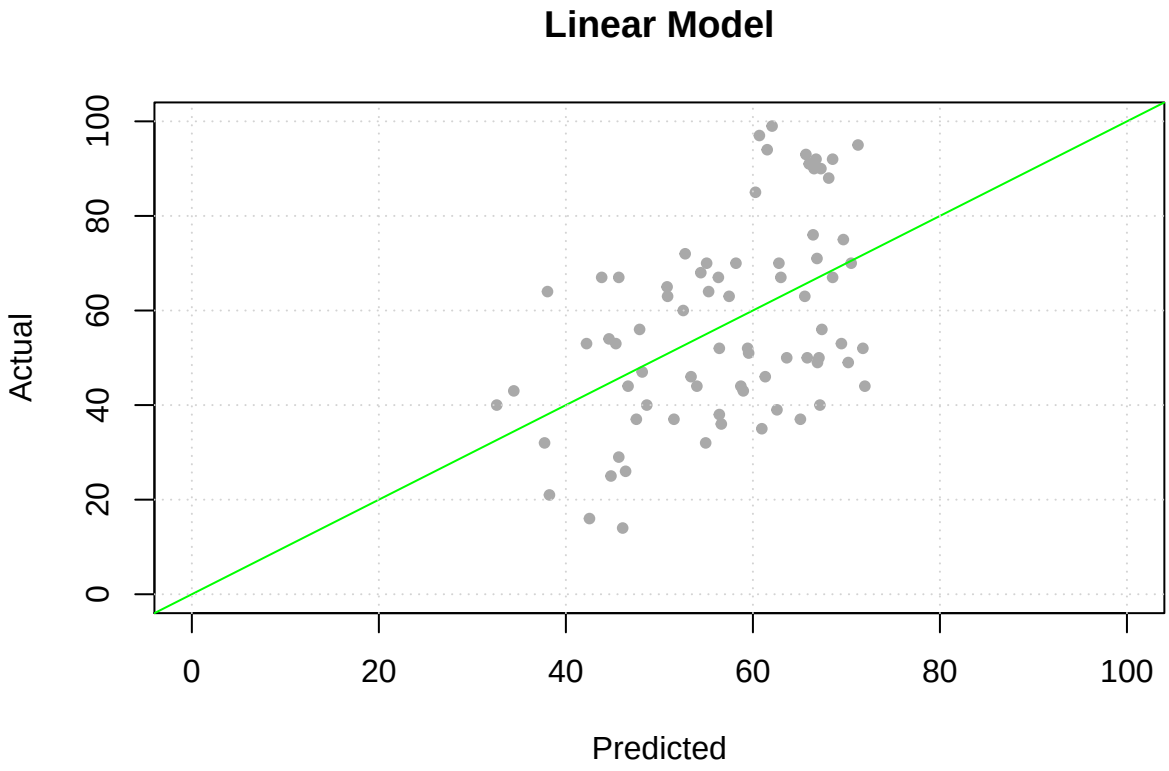
```
##
## Call:
##  randomForest(x = x, y = y, mtry = param$mtry)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 10
##
##           Mean of squared residuals: 197.4057
##                     % Var explained: 53.82
```

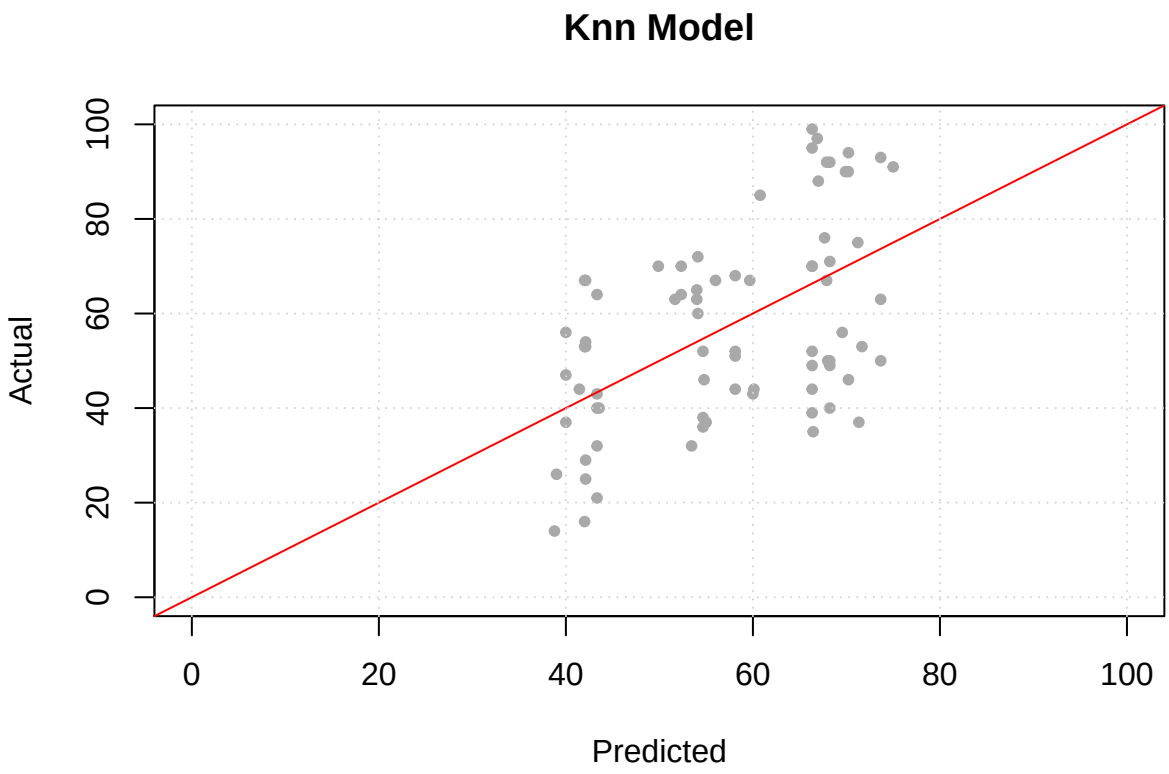
```
plot(fit_rf)
```

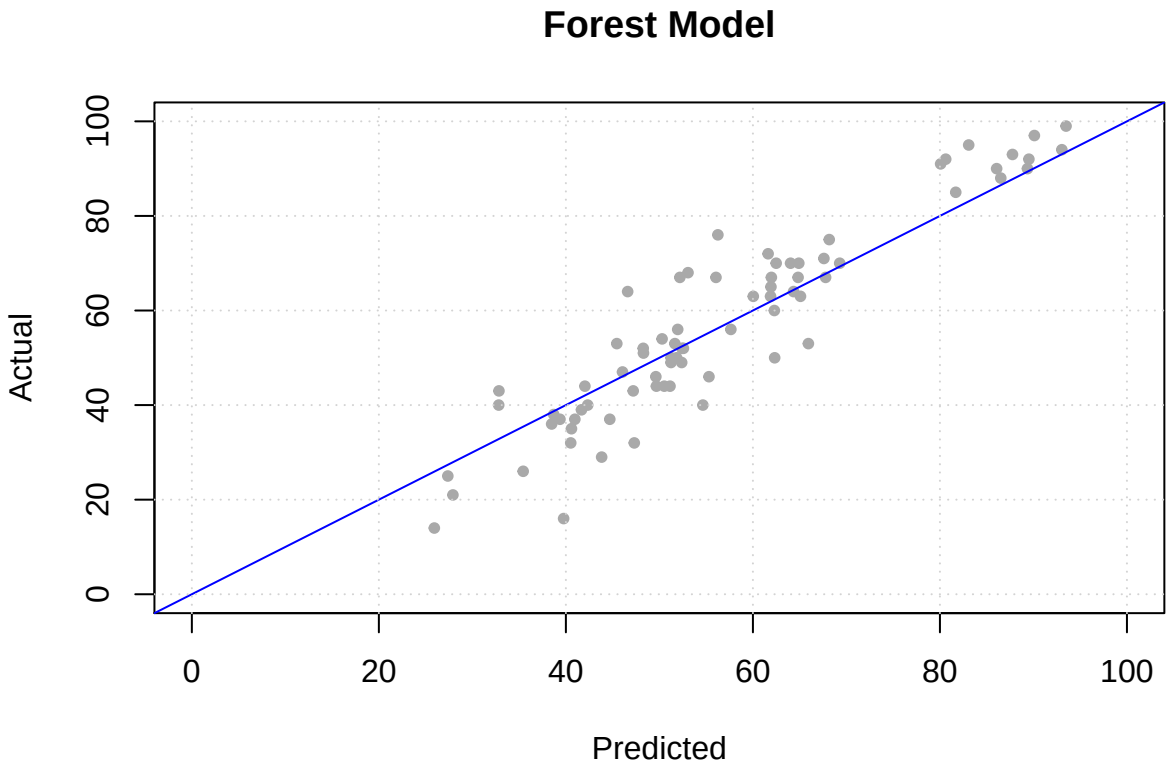


Predicted vs. Observed Values

The predicting quality of each model is shown below contrasting the observed and predicted values.


Linear Model
Actual
Predicted
0
20
40
60
80
100


Knn Model
Actual
Predicted
0
20
40
60
80
100


Forest Model
Actual
Predicted
0
20
40
60
80
100