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/jpb6/OneDrive - University of Illinois - Urbana/Fall 2019/CEE 508 - Pavement Evaluation and Rehabilitation/Term Project/DATA/irivspci2.csv",col\_names = T)  
data=as\_tibble(data)

## Models

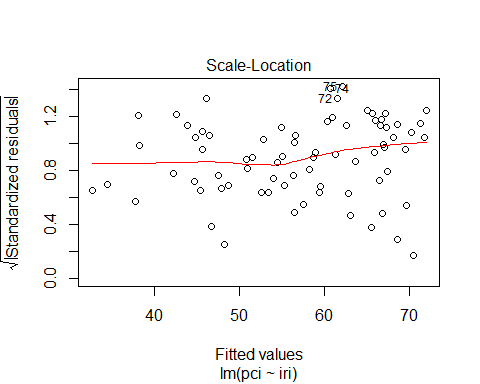
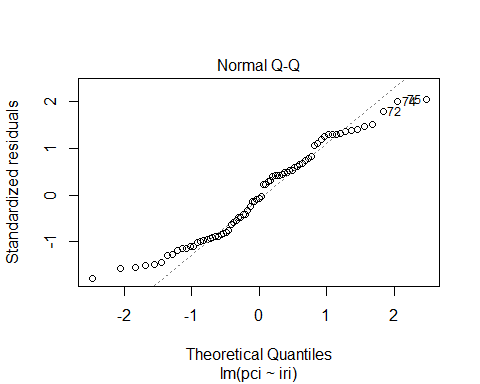
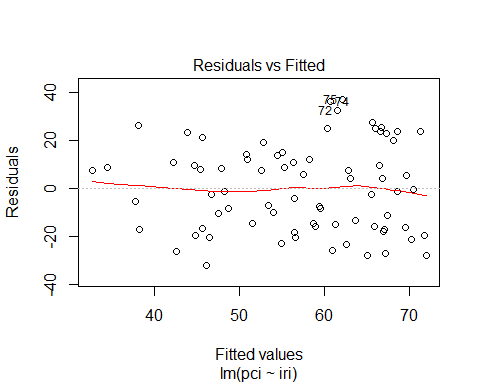
We try first a linear model, then a nearest neighbour model, and then a random forest model to analyse possible machine learning approaches.

### Linear Model

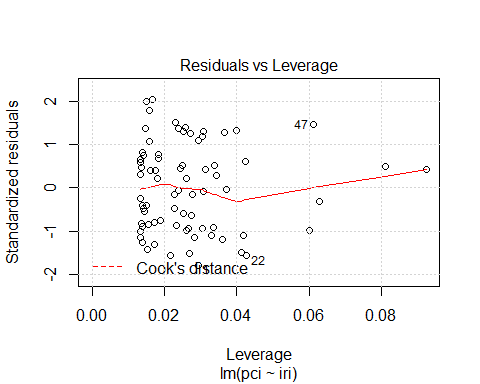
fit\_lm=lm(pci~iri, data = data)  
  
summary(fit\_lm)

##   
## Call:  
## lm(formula = pci ~ iri, data = data)  
##   
## 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  
##   
## Residual standard error: 18.32 on 73 degrees of freedom  
## Multiple R-squared: 0.236, Adjusted R-squared: 0.2255   
## F-statistic: 22.54 on 1 and 73 DF, p-value: 9.979e-06

plot(fit\_lm)



grid()



### Nearest Neighbors

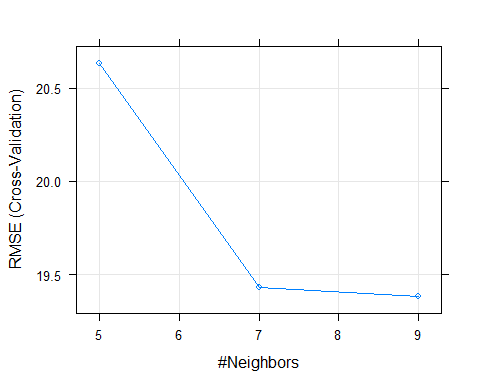
fit\_knn = train(  
 pci ~ iri,  
 data = data,  
 method = "knn",  
 trControl = trainControl(method='cv',number = 5)  
 )  
  
fit\_knn

## k-Nearest Neighbors   
##   
## 75 samples  
## 1 predictor  
##   
## No pre-processing  
## Resampling: Cross-Validated (5 fold)   
## Summary of sample sizes: 60, 60, 59, 60, 61   
## Resampling results across tuning parameters:  
##   
## k RMSE Rsquared MAE   
## 5 20.63490 0.1610891 17.76946  
## 7 19.42948 0.2047556 17.09264  
## 9 19.38052 0.1936146 17.05062  
##   
## 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 20.63490 0.1610891 17.76946 2.19431 0.1222830 2.416988  
## 2 7 19.42948 0.2047556 17.09264 1.57143 0.1528751 1.779502  
## 3 9 19.38052 0.1936146 17.05062 1.68053 0.1266733 1.707211

plot(fit\_knn)



### Random Forest

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: 59, 60, 61, 60, 60   
## Resampling results across tuning parameters:  
##   
## mtry RMSE Rsquared MAE   
## 2 15.12385 0.6035703 12.84597  
## 10 13.11961 0.6075963 10.51618  
## 19 12.75191 0.6296628 10.07108  
##   
## RMSE was used to select the optimal model using the smallest value.  
## The final value used for the model was mtry = 19.

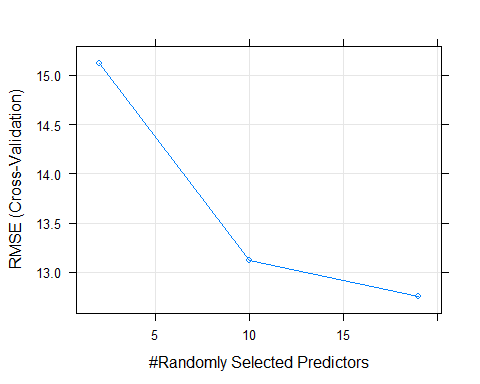
fit\_rf$results

## mtry RMSE Rsquared MAE RMSESD RsquaredSD MAESD  
## 1 2 15.12385 0.6035703 12.84597 2.092102 0.06572844 1.222715  
## 2 10 13.11961 0.6075963 10.51618 1.785553 0.04762010 1.180347  
## 3 19 12.75191 0.6296628 10.07108 1.644128 0.05001152 1.100144

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: 19  
##   
## Mean of squared residuals: 199.0633  
## % Var explained: 53.43

plot(fit\_rf)



## Predicted vs. Observed Values

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

