K-Fold cross validation

Guillermo Bustos-Pérez

How does a K-fold cross validation work?

k-fold cross validation is a basic measure to estimate the accuracy of a machine learning model. It consists of the following steps:

1) Randomly redistribute the dataset to aboid bias from the introduction of data.

2) Subset the dataset in different folders, each with equal number of data. Percentage of data in each folder depends on different factors such as the size of the dataset or analyst decision (10%, 20%, 25%, 50%, etc.).

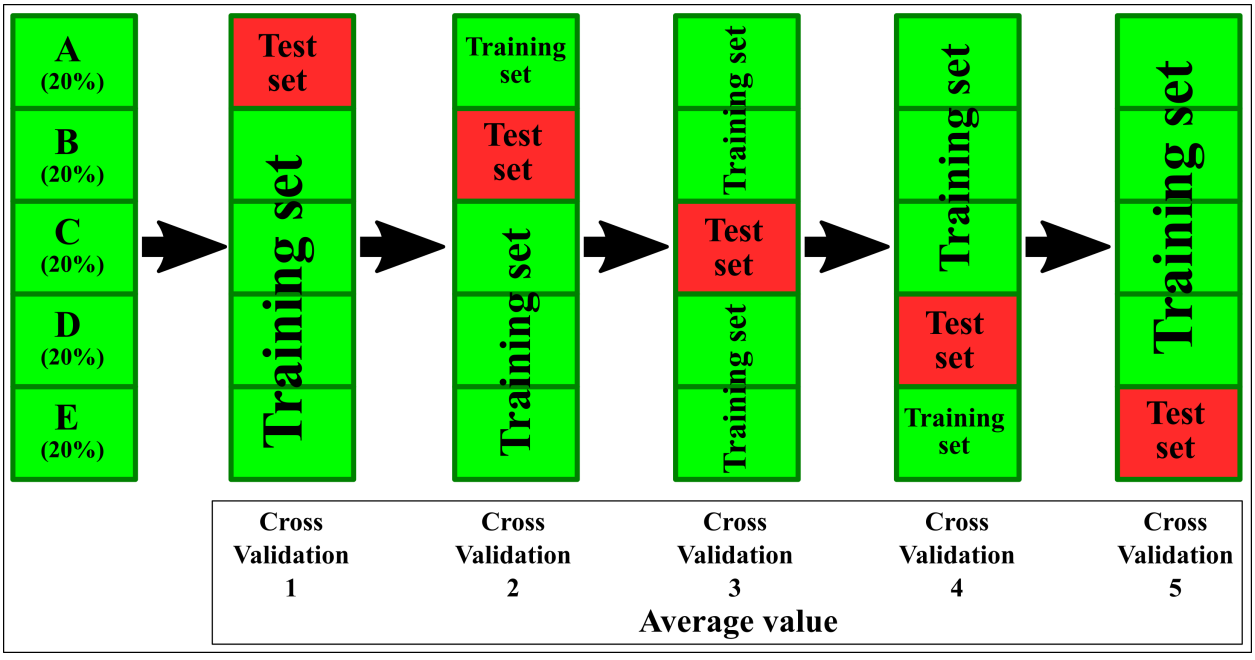
3) Retain first group as test datset and train the model with the rest of the groups. Evaluate the model using the test data set and retain the evaluation score.

4) Train a new model retaining the second group as test dataset. Evaluate and retain new evaluation score.

5) Repeat process subsequentially with all subgroups.

6) Calculate average of the evaluation scores from all subgroups. This concludes one cycle.

7) Since each cycle depends on the random redistribution of step 1, the cycles must be repeated. Number of repetition of cycles varies from 10 to 100 depending on analyst decision.



Althought many packages (such as the **caret** package) contain build in functions to carry out the cross validation is important to be able to replicate the code to acces internal information.

The following cell contains a basic code to perform K-fold cross validation dividing the data set into 5 subsets and repeating the cycles 50 times:

```
#### Create empty data frame to store K-Fold mean values ####
KCV <- data.frame(matrix(ncol = 1, nrow = 0))

#### Loop to obtain the values ####
repeat {

  # Sample and create subsets
  DB.Sampled <- DB.ML.2[sample(nrow(DB.ML.2)),]

x <- round(nrow(DB.Sampled) * 0.20, 0)

a <- DB.Sampled[1:x,]
b <- DB.Sampled[(x+1):(x*2) ,]
c <- DB.Sampled[((x*2)+1):(x*3) ,]
d <- DB.Sampled[((x*3)+1):(x*4) ,]
e <- DB.Sampled[((x*4)+1):(nrow(DB.Sampled)) ,]

Mean_Cross <- data.frame(Value = numeric())

# Cross subset a
Train <- rbind(b,c,d,e)
Good_Class_Model <- rpart(`Good Class` ~ .,
                          data = Train,
                          method = "class",
                          control = rpart.control(cp = 0))
a$pred <- predict(Good_Class_Model,
                  a,
                  type = "class")
Mean_Val <- mean(a$pred == a$`Good Class`)
Mean_Cross <- rbind(Mean_Cross, Mean_Val)
a <- a %>% select(-c(pred))

# Cross subset b
Train <- rbind(a,c,d,e)
Good_Class_Model <- rpart(`Good Class` ~ .,
                          data = Train,
                          method = "class",
                          control = rpart.control(cp = 0))
b$pred <- predict(Good_Class_Model,
                  b,
                  type = "class")
Mean_Val <- mean(b$pred == b$`Good Class`)
Mean_Cross <- rbind(Mean_Cross, Mean_Val)
b <- b %>% select(-c(pred))

# Cross subset c
Train <- rbind(a,b,d,e)
Good_Class_Model <- rpart(`Good Class` ~ .,
                          data = Train,
                          method = "class",
```

```
                          control = rpart.control(cp = 0))
c$pred <- predict(Good_Class_Model,
                  c,
                  type = "class")
Mean_Val <- mean(c$pred == c$`Good Class`)
Mean_Cross <- rbind(Mean_Cross, Mean_Val)
c <- c %>% select(-c(pred))

# Cross subset d
Train <- rbind(a,b,c,e)
Good_Class_Model <- rpart(`Good Class` ~ .,
                          data = Train,
                          method = "class",
                          control = rpart.control(cp = 0))
d$pred <- predict(Good_Class_Model,
                  d,
                  type = "class")
Mean_Val <- mean(d$pred == d$`Good Class`)
Mean_Cross <- rbind(Mean_Cross, Mean_Val)
d <- d %>% select(-c(pred))

# Cross subset e
Train <- rbind(a,b,c,d)
Good_Class_Model <- rpart(`Good Class` ~ .,
                          data = Train,
                          method = "class",
                          control = rpart.control(cp = 0))
e$pred <- predict(Good_Class_Model,
                  e,
                  type = "class")
Mean_Val <- mean(e$pred == e$`Good Class`)
Mean_Cross <- rbind(Mean_Cross, Mean_Val)
e <- e %>% select(-c(pred))

names(Mean_Cross)[1] <- "Cross_Val"
MeanK <- mean(Mean_Cross$Cross_Val) %>% as.data.frame()

KCV <- rbind(KCV, MeanK)
KCV <- na.omit(KCV)

if (nrow(KCV) == 50) {
  break
}
}
```