# tesina classificazione

Di Prospero, Iezzi, Valentini

9/9/2021

## Introduzione

Il seguente report analizza la probabilità di verificarsi un incendio boschivo in Algeria in base a parametri ambientali. La previsione che si vuole ottenere attraverso modelli di machine learning di apprendimento supervisionato permette di catalogare aree a rischio e, conseguentemente, attuare politiche tutelative e precauzionali.
Nell'eleborazione del progetto si analizzano i caratteri del fenomeno, si allenano differenti modelli su una parte dei dati a disposizione e se ne verifica la loro efficacia previsiva sulla restante porzione di dati. I modelli utilizzati sono quelli visti a lezione e altri non trattati nel corso come nearest neighbour, random forest and decision tree.

## DESCRIZIONE DEL DATASET E SELEZIONE DELLE VARIABILI DI INTERESSE:

Il set di dati "Algerian_forest_fires" include 246 osservazioni, in corrispondenza delle quali vengono rilevate 10 variabili esplicative ed una vartiabile di classe (categorica). I dati fanno riferimento a due regioni dell'Algeria, vale a dire la regione di Bejaia situata nel nord-est dell'Algeria e la regione di Sidi Bel-abbes situata nel nord-ovest dell'Algeria, e per ciascuna regione vengono rilevate 122 istanze. Per meglio descrivere la composizione del Dataset mostriamo le prime sei righe che lo caratterizzano:

```
library(knitr)
kable(head(Algerian_forest_fires_dataset_UPDATE))
```

| day | month | year | Temperature | RH | Ws | Rain | FFMC | DMC | DC | ISI | BUI | FWI | Classes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 06 | 2012 | 29 | 57 | 18 | 0.0 | 65.7 | 3.4 | 7.6 | 1.3 | 3.4 | 0.5 | not fire |
| 02 | 06 | 2012 | 29 | 61 | 13 | 1.3 | 64.4 | 4.1 | 7.6 | 1.0 | 3.9 | 0.4 | not fire |
| 03 | 06 | 2012 | 26 | 82 | 22 | 13.1 | 47.1 | 2.5 | 7.1 | 0.3 | 2.7 | 0.1 | not fire |
| 04 | 06 | 2012 | 25 | 89 | 13 | 2.5 | 28.6 | 1.3 | 6.9 | 0.0 | 1.7 | 0.0 | not fire |
| 05 | 06 | 2012 | 27 | 77 | 16 | 0.0 | 64.8 | 3.0 | 14.2 | 1.2 | 3.9 | 0.5 | not fire |
| 06 | 06 | 2012 | 31 | 67 | 14 | 0.0 | 82.6 | 5.8 | 22.2 | 3.1 | 7.0 | 2.5 | fire |

Le prime tre colonne non risutano essere necessarie ai fini dell'analisi, e pertanto si procede con la creazione dell'oggetto X che esclude tali colonne dal set di dati di riferimento e eliminiamo le righe che presenatno dati mancanti:

```
X <- Algerian_forest_fires_dataset_UPDATE[,4:14]
X  <- na.omit(X)
```

Mostriamo nuovamente le prime sei righe che caratterizzano il nuovo set di dati che contiene 243 osservazioni:

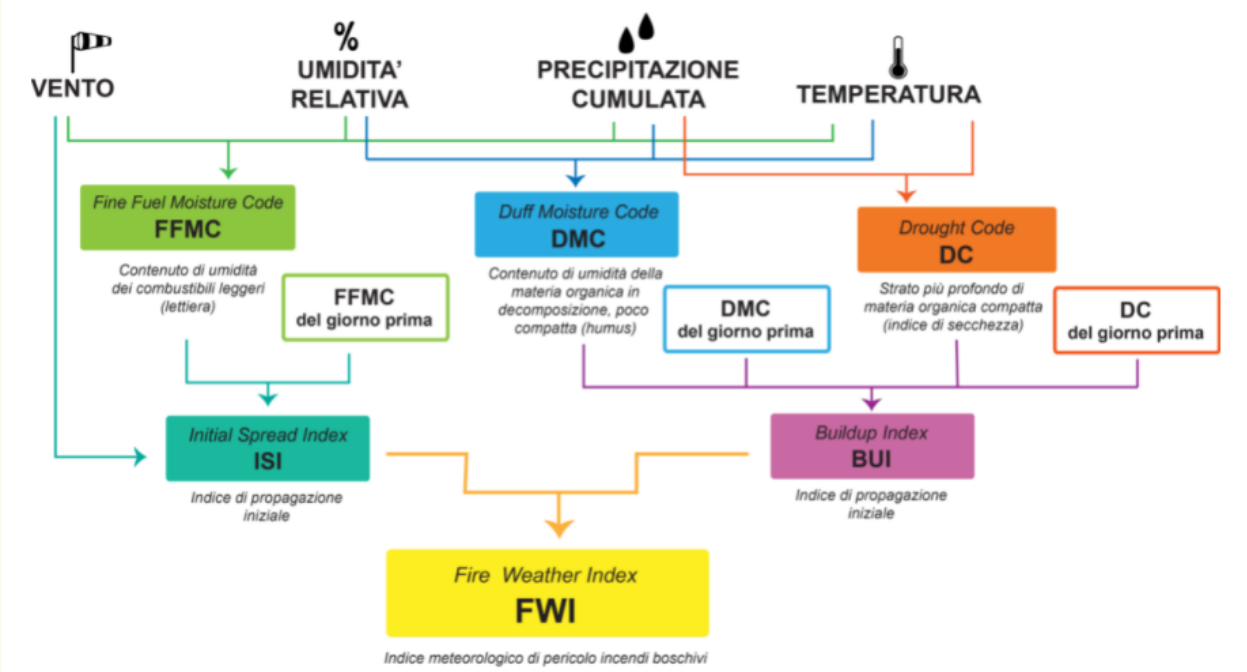
```
kable(head(X))
```

| Temperature | RH | Ws | Rain | FFMC | DMC | DC | ISI | BUI | FWI | Classes |
|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 57 | 18 | 0.0 | 65.7 | 3.4 | 7.6 | 1.3 | 3.4 | 0.5 | not fire |
| 29 | 61 | 13 | 1.3 | 64.4 | 4.1 | 7.6 | 1.0 | 3.9 | 0.4 | not fire |
| 26 | 82 | 22 | 13.1 | 47.1 | 2.5 | 7.1 | 0.3 | 2.7 | 0.1 | not fire |
| 25 | 89 | 13 | 2.5 | 28.6 | 1.3 | 6.9 | 0.0 | 1.7 | 0.0 | not fire |
| 27 | 77 | 16 | 0.0 | 64.8 | 3.0 | 14.2 | 1.2 | 3.9 | 0.5 | not fire |
| 31 | 67 | 14 | 0.0 | 82.6 | 5.8 | 22.2 | 3.1 | 7.0 | 2.5 | fire |

Per ogni osservazione osserviamo quindi le seguenti variabili:

- Temperature: temperatura massima registrata a mezzogiorno (espressa in gradi Celsius),
- RH: umidità relativa (espressa in percentuale)
- Ws: velocità del vento (espressa in km/h),
- Rain: pioggia caduta nell'arco di in un'intera giornata (espressa in mm),
- index from the FWI system (FFMC): contenuto idrico dei combustibili fini,
- index from the FWI system (DMC): contenuto idrico medio degli strati organici moderatamente profondi compattatati grossolanamente,
- index from the FWI system (DC): lo strato profondo di materia organica compattata
- index from the FWI system (ISI): strato profondo di materia organica compattata.
- index from the FWI system(BUI): media armonica dei contenuti di umidità dei due strati più profondi di combustibile,
- Fire Weather Index (FWI): possibilità di innesco di un incendio.
- Classes(la variabile d'interesse Y): variabile contenente le labels che identificano le classi di appartenenza dei gruppi. La variabile "Classes" si compone di due labels rappresentanti due classi:
- "Fire", ossia la label che identifica le aree boschive in cui si verificano incendi
- "not fire", ossia la label che identifica le aree boschive in cui non si verificano incendi. Con riferimento agli indicatori che compongono Fire Weather Index (FWI) occorre prendere in considerazione le seguenti informazioni:
- gli indicatori FFMC, DMC, DC rappresentano tre sottoindici primari,
- gli indicatori ISI, BUI rappresentano due sottoindici intermedi,
- l'indice FWI combina l'informazione derivata da ISI e BUI.

Di seguito viene riportata una rappressentazione grafica che meglio illustra la metodologia con cui vengono derivati i seguenti indicatori.




Il modo in cui gli indicatori che compongono Fire Weather Index (FWI) vengono derivati fa sospettare la presenza di un'elevata correlazione fra gli stessi. Pertanto, prima di procedere con la creazione del Training e Test set verifichiamo le correlazioni esistenti fra le variabili che compongono il set di dati considerato:

```
kable(cor(X[,1:10]), digits = 3)
```

| | Temperature | RH | Ws | Rain | FFMC | DMC | DC | ISI | BUI | FWI |
|---|---|---|---|---|---|---|---|---|---|---|
| Temperature | 1.000 | -0.651 | -0.285 | -0.326 | 0.677 | 0.486 | 0.376 | 0.604 | 0.460 | 0.567 |
| RH | -0.651 | 1.000 | 0.244 | 0.222 | -0.645 | -0.409 | -0.227 | -0.687 | -0.354 | -0.581 |
| Ws | -0.285 | 0.244 | 1.000 | 0.172 | -0.167 | -0.001 | 0.079 | 0.009 | 0.031 | 0.032 |
| Rain | -0.326 | 0.222 | 0.172 | 1.000 | -0.544 | -0.289 | -0.298 | -0.347 | -0.300 | -0.324 |
| FFMC | 0.677 | -0.645 | -0.167 | -0.544 | 1.000 | 0.604 | 0.507 | 0.740 | 0.592 | 0.691 |
| DMC | 0.486 | -0.409 | -0.001 | -0.289 | 0.604 | 1.000 | 0.876 | 0.680 | 0.982 | 0.876 |
| DC | 0.376 | -0.227 | 0.079 | -0.298 | 0.507 | 0.876 | 1.000 | 0.509 | 0.942 | 0.740 |
| ISI | 0.604 | -0.687 | 0.009 | -0.347 | 0.740 | 0.680 | 0.509 | 1.000 | 0.644 | 0.923 |
| BUI | 0.460 | -0.354 | 0.031 | -0.300 | 0.592 | 0.982 | 0.942 | 0.644 | 1.000 | 0.858 |
| FWI | 0.567 | -0.581 | 0.032 | -0.324 | 0.691 | 0.876 | 0.740 | 0.923 | 0.858 | 1.000 |

Dall'analisi delle correlazioni si può osservare che gli indicatori che compongono Fire Weather Index (FWI) tendono a presentare correlazioni alte; in particolare si evidenziano correlazioni prossime ad 1 per le seguenti variabili: * FWI ed ISI con correlazione pari a 0.923, * BUI e DMC con correlazione pari a 0.982, * BUI e DC con correlazione pari a 0.942.

Si procede quindi con la creazione di un nuovo oggetto denominato "fireSet" in cui verranno prese in considerazione solamente le variabili "Temperature", "RH" , "Ws" , "Rain" e "FWI":

```
fireSet <- X[,-(5:9)]
```

Analiziamo più nel dettaglio le caratteristiche delle variabili considerate mediante la funzione str() e visualizziamo le principali statistiche descrittive delle nostre variabili:

```r
str(fireSet)
```

```
## tibble [243 x 6] (S3: tbl_df/tbl/data.frame)
##  $ Temperature: num [1:243] 29 29 26 25 27 31 33 30 25 28 ...
##  $ RH         : num [1:243] 57 61 82 89 77 67 54 73 88 79 ...
##  $ Ws         : num [1:243] 18 13 22 13 16 14 13 15 13 12 ...
##  $ Rain       : num [1:243] 0 1.3 13.1 2.5 0 0 0 0 0.2 0 ...
##  $ FWI        : num [1:243] 0.5 0.4 0.1 0 0.5 2.5 7.2 7.1 0.3 0.9 ...
##  $ Classes    : Factor w/ 2 levels "fire","not fire": 2 2 2 2 2 1 1 1 2 2 ...
##  - attr(*, "na.action")= 'omit' Named int [1:3] 123 124 168
##   ..- attr(*, "names")= chr [1:3] "123" "124" "168"
```

```r
kable(summary(fireSet[,1:5]))
```

| Temperature | RH | Ws | Rain | FWI |
|---|---|---|---|---|
| Min. :22.00 | Min. :21.00 | Min. : 6.00 | Min. : 0.000 | Min. : 0.000 |
| 1st Qu.:30.00 | 1st Qu.:52.50 | 1st Qu.:14.00 | 1st Qu.: 0.000 | 1st Qu.: 0.700 |
| Median :32.00 | Median :63.00 | Median :15.00 | Median : 0.000 | Median : 4.200 |
| Mean :32.15 | Mean :62.04 | Mean :15.49 | Mean : 0.763 | Mean : 7.035 |
| 3rd Qu.:35.00 | 3rd Qu.:73.50 | 3rd Qu.:17.00 | 3rd Qu.: 0.500 | 3rd Qu.:11.450 |
| Max. :42.00 | Max. :90.00 | Max. :29.00 | Max. :16.800 | Max. :31.100 |

Si procede poi con il partizionamento del Dataset ottenuto in Training e Test set mediante la funzione createDataPartition() facente parte della libreria "caret" ; in particolare il set di dati "fireSet" viene partizionato in modo tale che l'80% delle osservazioni entraranno a far parte del training set, mentre le rimanenti osservazioni (il 20%) entreranno a far parte del test set. Viene inoltre impostato un seed in modo tale da rendere i risultati dell'operazione di splitting replicabile.

```r
library(caret)

set.seed(123)
training.samples = createDataPartition(fireSet$Classes, p = .8, list = FALSE)
train.data <- fireSet[training.samples, ]
test.data <- fireSet[-training.samples, ]
```

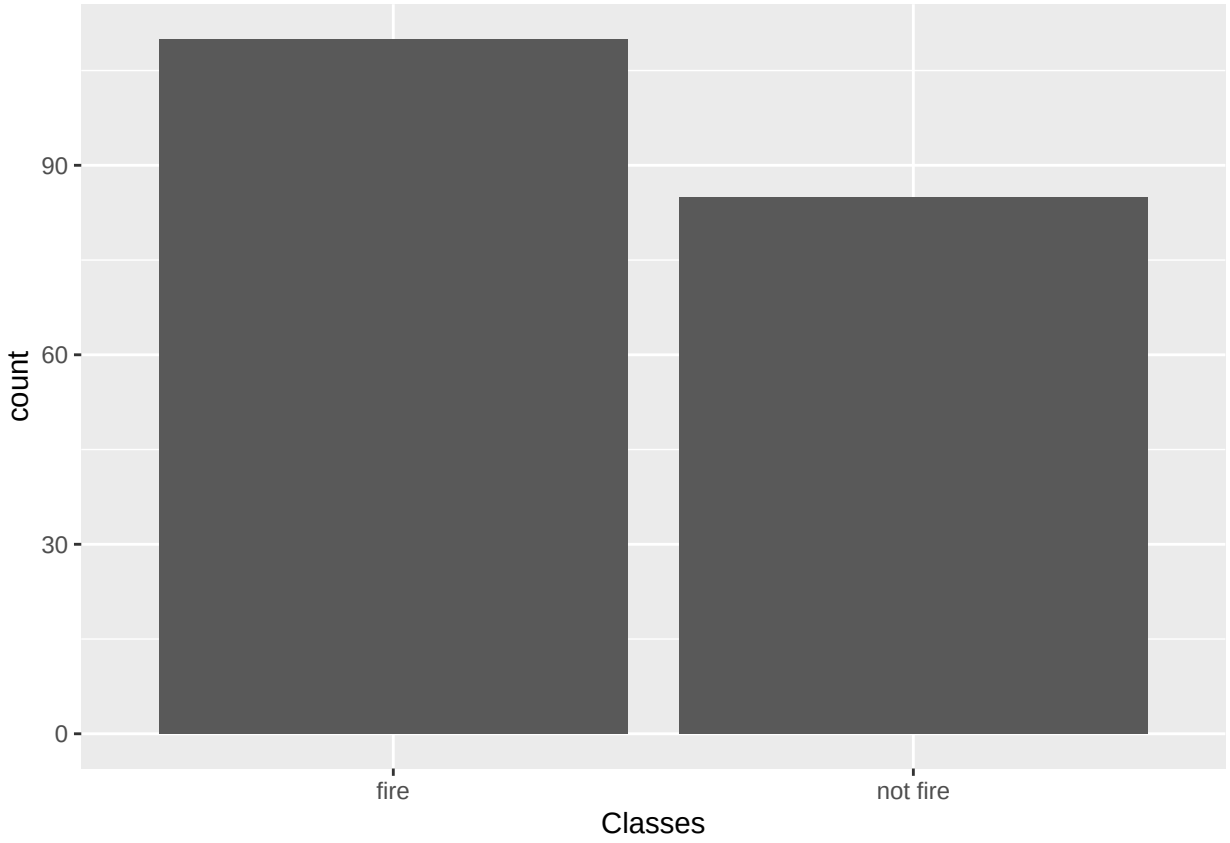
# ANALISI ESPLORATIVA

Il primo passo dell'analisi esplorativa dei dati selezionati consiste nel caricare le librerie che verranno impiegate(ossia, le librerie "ggord" e "psych") e nell'andare ad osservare la numerosità delle classi identificate, in modo tale da verificare se le stesse risultano essere bilanciate o meno.

```r
library(ggord)
library(psych)
```

Un primo approccio per verificare la numerosità delle classi consiste nell'andare ad effettuare una rappresentazione grafica mediante la funzione ggplot():

```
ggplot(train.data, aes(Classes)) + geom_bar()
```




La rappresentazione grafica ottenuta mostra che la classe “fire” risulta presentare una numerosità maggiore rispetto alla classe “not fire”. Per un analisi più approfondita viene inoltre impiegata la funzione summary.factor():

```
summary.factor(train.data$Classes)
```

```
##     fire not fire
##      110       85
```

Dall’output ottenuto si riscontrano 110 aree in cui si verifica un incendio boschivo (osservazioni classificate mediante label “fire”) ed 85 aree in cui non si verifica un incendio boschivo (osservazioni classificate mediante label “not fire”), con uno scarto di 25 osservazioni fra le due classi. Pertanto, si può conlcudere che le classi risultano essere abbastanza bilanciate. Procediamo ora a rappresentare in tabella le principali statistiche descrittive delle variabili presenti nel training set:

```
kable(summary(train.data[,1:5]))
```

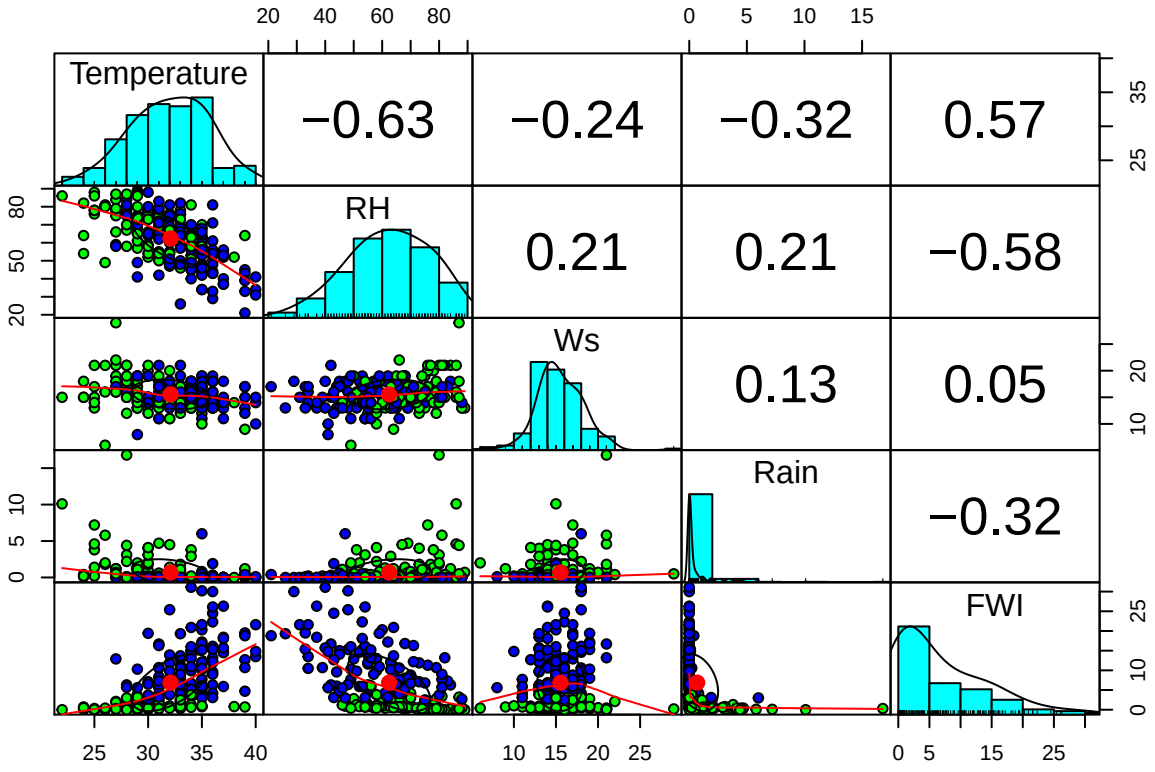
| Temperature | RH | Ws | Rain | FWI |
|---|---|---|---|---|
| Min. :22.00 | Min. :21.00 | Min. : 6.00 | Min. : 0.0000 | Min. : 0.000 |
| 1st Qu.:29.50 | 1st Qu.:53.00 | 1st Qu.:14.00 | 1st Qu.: 0.0000 | 1st Qu.: 0.700 |
| Median :32.00 | Median :64.00 | Median :15.00 | Median : 0.0000 | Median : 4.200 |

| Temperature | RH | Ws | Rain | FWI |
|---|---|---|---|---|
| Mean :32.09 | Mean :62.43 | Mean :15.54 | Mean : 0.6887 | Mean : 6.872 |
| 3rd Qu.:35.00 | 3rd Qu.:74.50 | 3rd Qu.:17.00 | 3rd Qu.: 0.4500 | 3rd Qu.:11.450 |
| Max. :40.00 | Max. :89.00 | Max. :29.00 | Max. :16.8000 | Max. :31.100 |

Nella rappresentazione seguente ottenuta tramite il comando pairs.panels() della libreia psych, nella parte sopra la diagonale possiamo notare le correlazioni tra le variabili e non troviamo valori alti da lasciar pensare a problemi di multicollinearità. Nella parte sotto la diagonale, invece, troviamo i vari scatterplot tra le variabili in cui le nostre osservazioni sono divise per colore in base alla label di appartenenza. Lungo la diagonale sono presnti gli istogrammi e le relative funzioni di densità delle variabili. In particolare:

- Possiamo notare che la rappresentazione dell'istogramma relativo alla temperatura massima tende a presentare una leggera asimmetria negativa (asimmetria a sinistra), in quanto la forma della distibuzione di densità è caratterizzata da una coda allungata verso sinistra. Inoltre, possiamo notare anche la presenza di un picco ampio nella parte centrale della distibuzione, il che può suggerire che la distribuzione può essere multimodale.
- Possiamo notare che la rappresentazione dell'istogramma relativo alla percentuale di umidità tende a presentare un asimmetria negativa, in quanto anche in questo caso la forma della distibuzione di densità è caratterizzata da una coda allungata verso sinistra. Inoltre, possiamo notare anche la presenza di un picco ampio nella parte centrale della distribuzione, il che suggerisce che la distribuzione può essere multimodale.
- Possiamo notare che la rappresentazione dell'istogramma relativo alla velocità del vento risulta essere quasi simmetrica. Inoltre, possiamo notare anche la presenza di un picco ampio nella parte centrale della distribuzione, il che suggerisce che la distribuzione risulta essere anche in questo caso multimodale.
- Possiamo notare che nella rappresentazione dell'istogramma relativo alla pioggia caduta la maggior parte dei valori tendono a concentrarsi nella parte a sinistra dove è presente un unico picco ampio e che lascia pensare a una distribuzione unimodale.
- Dalla rappresentazione dell'istogramma emerge che l'indicatore "FWI" tende ad assumere principalmente valori pari a zero o leggermente maggiori.

```
pairs.panels(train.data[1:5],gap=0, bg=c("blue","green")[train.data$Classes],pch=21)
```

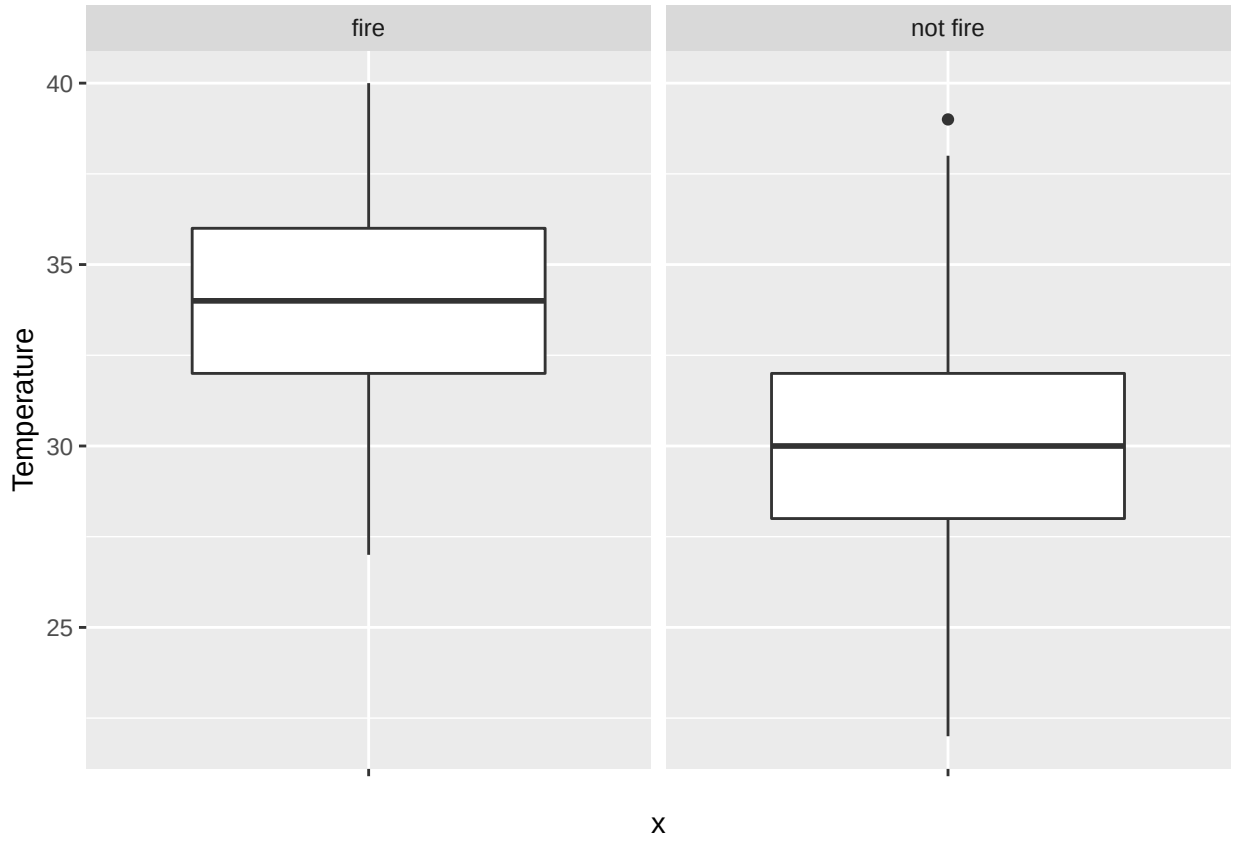



L'analisi esplorativa procede considerando la distribuzione della variabili esplicative considerate. In particolare, analizziamo la distribuzione di tali variabili mediante l'utilizzo di Boxplot categorizzati, in modo tale da:

- verificare quanta variabilità c'è nei dati,
- comprendere se le distribuzioni sono simmetriche oppure asimmetriche,
- confrontare la forma delle distribuzioni della variabile di riferimanto divisa per classi sotto-collettivi)per comprendere le differenze tra i gruppi
- identificare eventuali "outliers".

  Procediamo con la costruzione del Boxplot categorizzato relativo alla variabile "Temperature", la quale indica la temperatura massima rilevata a mezzogiorno in corrispondenza delle aree considerate (espressa in gradi Celsius):

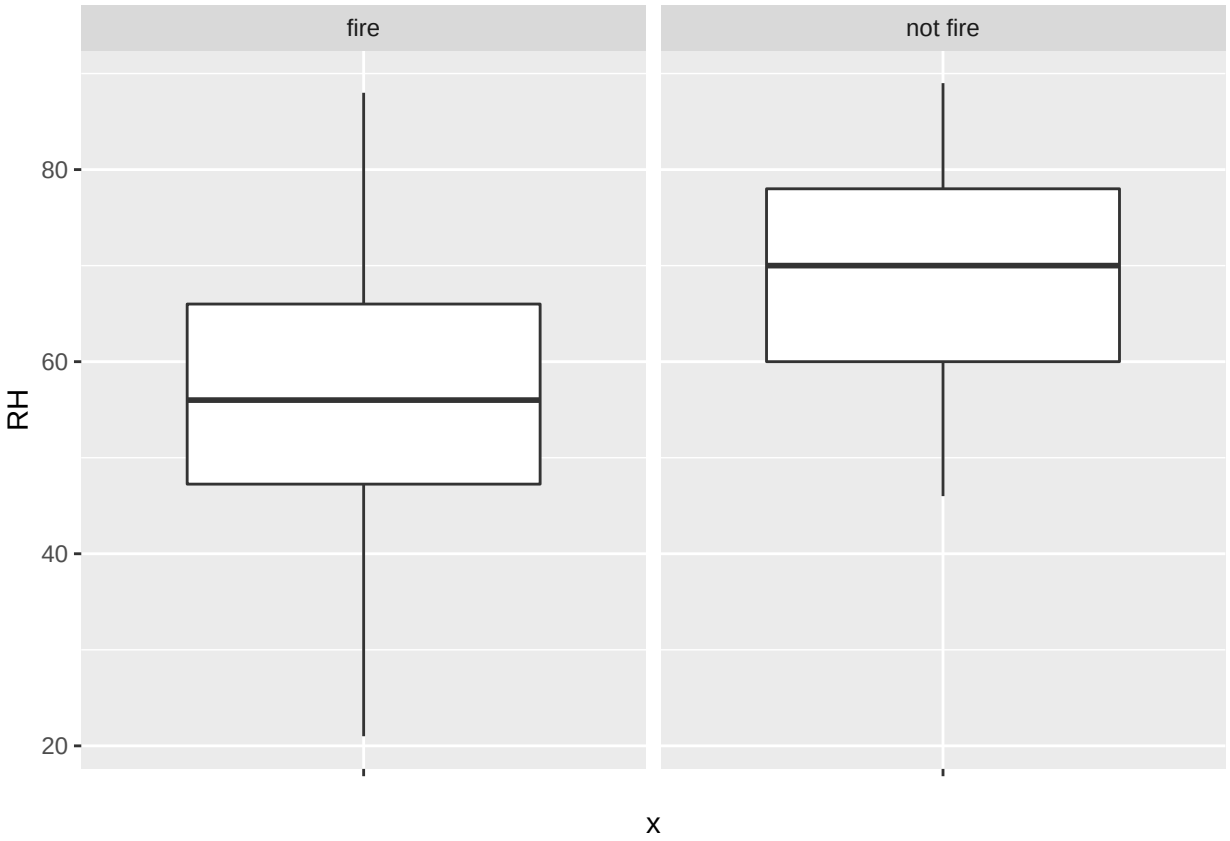
```
ggp_Temp <- ggplot(train.data, aes(x = " ", y = Temperature)) +
 geom_boxplot() +
 facet_grid(facets = ~train.data$Classes, as.table = TRUE)
ggp_Temp
```




Da questo grafico si può notare che nelle aree in cui si registrano incendi boschivi si rileva un valore mediano di temperatura massima più alto rispetto alle aree in cui non ri registrano incendi boschivi. Sia il primo ed il terzo quartile che la mediana risultano essere più alti nel boxplot a sinistra, ossia quello che descrive la distribuzione della temperatura media massima delle aree boschive in cui si registrano incendi. Inoltre, con riferimento al 50% dei valori centrali (rappresentati nel grafico dall'altezza della scatola, che corrisponde al range interquartile) si può notare che le due distribuzioni presentano sostanzialmente la stessa variabilità (dispersione), e che i range dei due gruppi tendono a sovrapporsi.Inoltre, si nota anche che le due distribuzioni risultano essere simmetriche: infatti, il primo ed il terzo quartile sono alla stessa distanza dalla mediana (la linea della mediana si trova esattamente a metà della scatola). Infine, possiamo osservare che relativamente al boxplot a destra si osserva che è presente un potenziale outlier, che nel grafico è indicato con un punto nero.

Procediamo con la costruzione del Boxplot relativo alla variabile "RH", la quale indica la quantità umidità relativa rilevata in corrispondenza delle aree considerate (espressa in percentuale):

```
ggp_RH <- ggplot(train.data, aes(x = " ", y = RH)) +
  geom_boxplot() +
  facet_grid(facets = ~train.data$Classes, as.table = TRUE)
ggp_RH
```

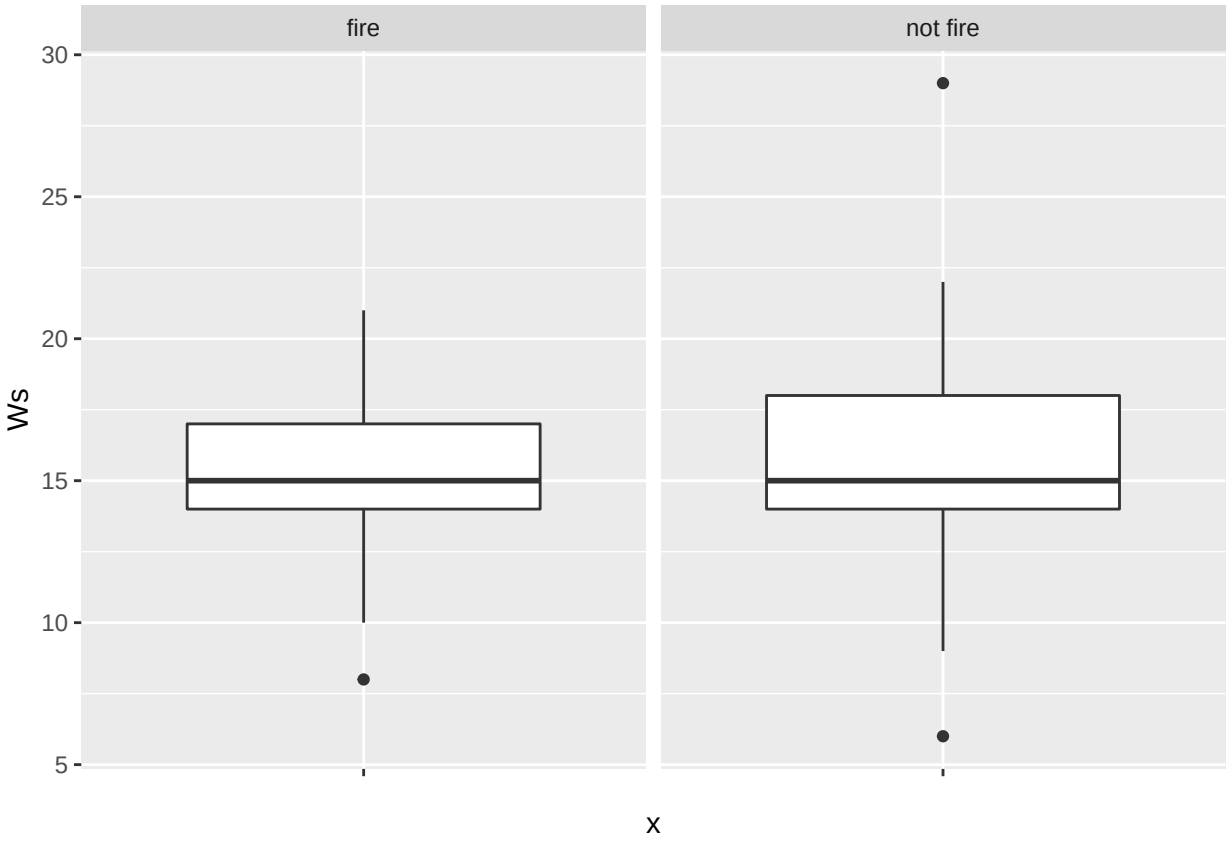



Da questo grafico si può notare che nelle aree in cui non si registrano incendi boschivi si rileva un valore mediano relativo alla percentuale di umidità più alta rispetto alle aree in cui si registrano incendi boschivi e pari a 70. Sia il primo ed il terzo quartile che la mediana risultano essere più alti nel boxplot a destra, ossia quello che descrive la distribuzione della percentuale di umidità delle aree boschive in cui non si registrano incendi. Inoltre, con riferimento al 50% dei valori centrali, si può notare che le due distribuzioni presentano sostanzialmente la stessa variabilità (dispersione), e che i range dei due gruppi tendono a sovrapporsi. Inoltre, si nota anche che le due distribuzioni tendono a presentare una leggera asimmetria:

- la distribuzione relativa alle aree in cui si registrano incendi boschivi tende ad essere asimmetrica a destra, in quanto il terzo quartile è leggermente più lontano dalla mediana di quanto non lo sia il primo quartile.
- la distribuzione relativa alle aree in cui non si registrano incendi boschivi tende ad essere asimmetrica a sinistra, in quanto la mediana risulta essere leggermente più vicina al terzo quartile rispetto al primo quartile. Infine, possiamo osservare che entrambi i boxplot non mostrano la presenza di possibili potenziali outliers.

Procediamo con la costruzione del Boxplot categorizzato relativo alla variabile "Ws", la qualeindica la velocità del vento rilevata in corrispondenza delle aree boschive considerate (espressa in km/h):

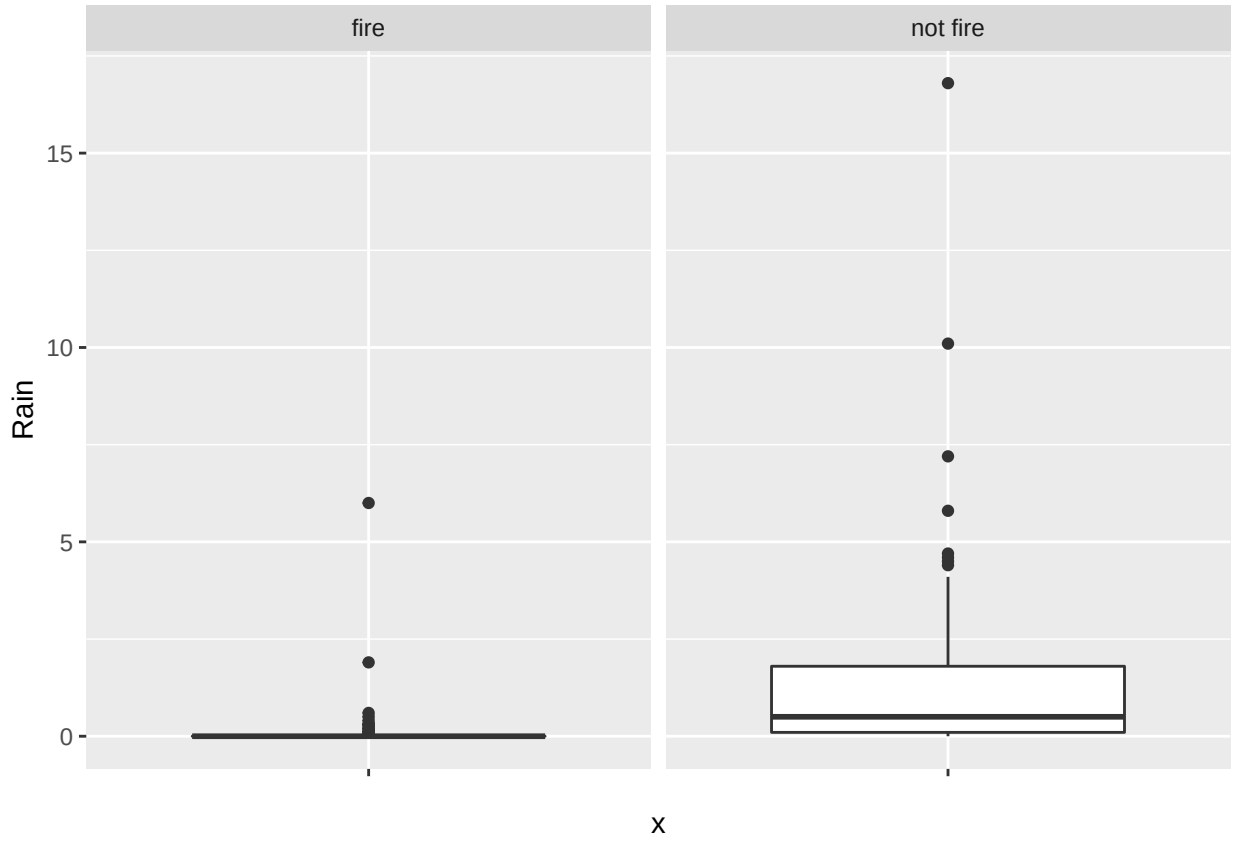
```
ggp_Ws <- ggplot(train.data, aes(x = " ", y = Ws)) +
  geom_boxplot() +
  facet_grid(facets = ~train.data$Classes, as.table = TRUE)
ggp_Ws
```




Da questo grafico si può notare che i valori mediani relativi alla velocità del vento sono gli stessi per entrambe le distribuzioni e pari a 15 Km/h. Il terzo quartile risulta essere più alto nel boxplot a sinistra, ossia quello che descrive la distribuzione della velocità del vento nelle aree boschive in cui non si registrano incendi, mentre il primo quartile si posiziona alla stessa altezza in entrambi i boxplot. Inoltre, con riferimento al 50% dei valori centrali, si può notare che la distribuzione relativa alla velocità del vento nelle aree boschive in cui non si verificano incendi presenta una maggiore variabilità (dispersione) rispetto alla distribuzione relativa alla velocità del vento nell aree boschive in cui si verificano incendi, ed inolte si nota anche che i range dei due gruppi si sovrappongono. Le due distribuzioni presentano una asimmetria a destra, in quanto il terzo quartile è più lontano dalla mediana di quanto non lo sia il primo quartile. Tale asimmetria risulta essere più marcata nella distribuzione relativa alla velocità del vento nelle aree in cui non si verificano incendi boschivi (boxplot a sinistra). Infine, possiamo osservare che entrambe le distribuzioni mostrano la presenza di possibili potenziali outliers (uno nel box plot di destra e 2 nel boxplot di sinistra).

Procediamo con la costruzione del Boxplot categorizzato relativo alla variabile “Rain”, la quale indica la quantità di pioggia caduta nel corso di un intera giornata in corrispondenza delle aree boschive considerate (espressa in mm):

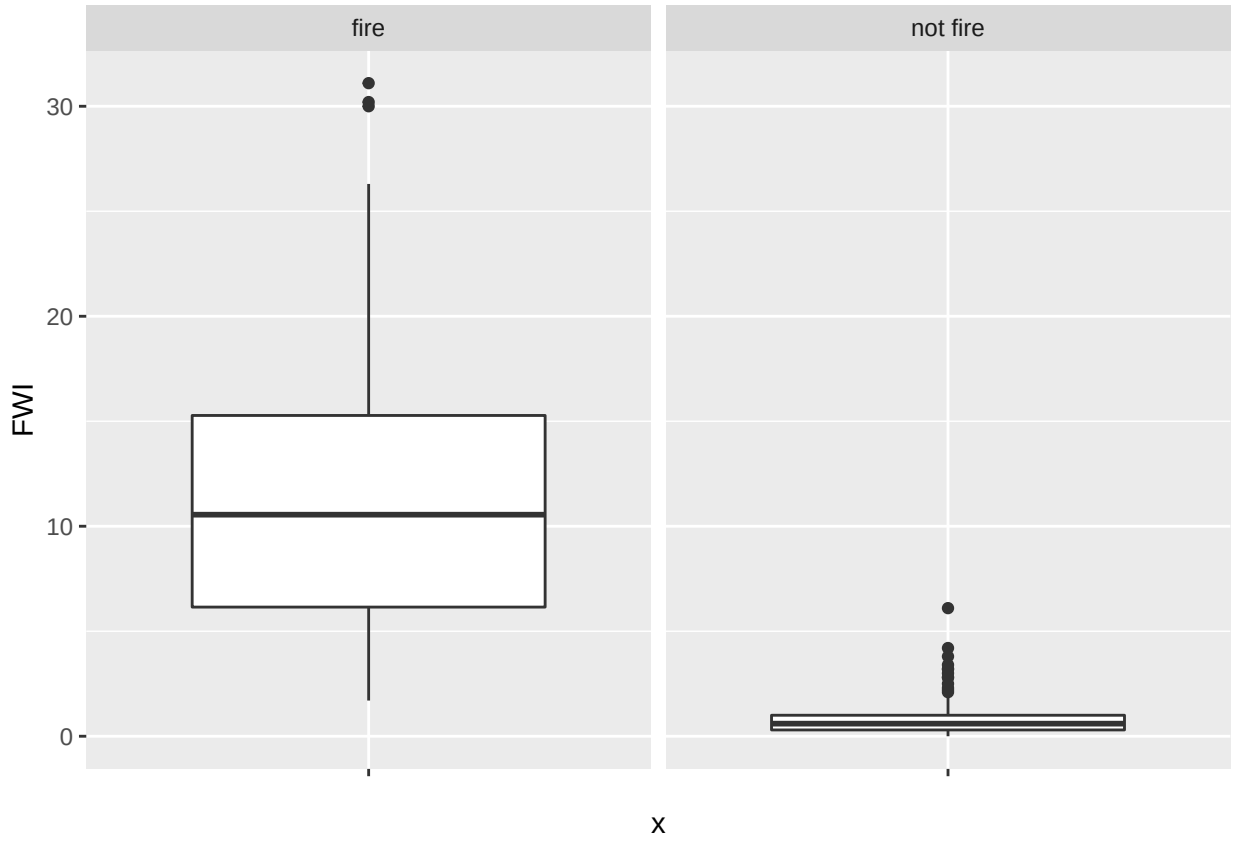
```
ggp_Rain <- ggplot(train.data, aes(x = " ", y = Rain)) +
  geom_boxplot() +
  facet_grid(facets = ~train.data$Classes, as.table = TRUE)
ggp_Rain
```




Da questo grafico si può notare che nelle aree in cui non si registrano incendi boschivi si rileva un valore mediano relativo alla pioggia caduta nell'arco di una giornata leggermente maggiore rispetto alle aree in cui si registrano incendi boschivi. Per quanto concerne la distribuzione del livello di pioggia caduta nell'arco di una giornata relativa alle aree in cui si verificano gli incendi boschivi i valori si concentrano sullo zero. Nello specifico, per quanto concerne tale distribuzione il primo ed il terzo quartile coincidono con la mediana. Inoltre, si può notare che si registra una variabilità nella distribuzione del livello di pioggia esclusivamente per quanto riguarda il colletivo dato dalle aree in cui non si verificano incendi (boxplot di destra). Inoltre, si nota anche che le due distribuzioni tendono a presentare una leggera asimmetria: la distribuzione relativa alle aree in cui non si registrano incendi boschivi tende inolte ad essere asimmetrica a destra, in quanto il terzo quartile è più lontano dalla mediana di quanto non lo sia il primo quartile. Infine, possiamo osservare che entrambi i boxplot mostrano la presenza di possibili potenziali outliers i quali tendono ad essere più numerosi rispetto a quelli rilevati nelle distribuzioni precedenti.

Procediamo infine con la costruzione del Boxplot categorizzato relativo alla variabile "FWI", la quale fornisce una indicazione circa la possibilità di innesco di un incendio rilevata in corrispondenza delle aree boschive considerate:

```
ggp_FWI <- ggplot(train.data, aes(x = " ", y = FWI)) +
  geom_boxplot() +
  facet_grid(facets = ~train.data$Classes, as.table = TRUE)
ggp_FWI
```




Da questo grafico si può notare che nelle aree in cui si registrano incendi boschivi si rileva un valore mediano relativo all'indicatore "FWI" più alto rispetto alle aree in cui si registrano incendi boschivi e pari a circa 10. Sia il primo ed il terzo quartile che la mediana risultano essere più alti nel boxplot a sinistra, ossia quello che descrive la distribuzione dell'indice "FWI" nelle aree boschive in cui si registrano incendi boschivi. Inoltre, con riferimento al 50% dei valori centrali, si può notare che la distribuzione relativ all'indice FWI nelle aree boschive in cui si verificano incendi presenta una maggiore variabilità (dispersione) rispetto alla distribuzione relativa all'indice "FWI" nelle aree boschive in cui non si verificano incendi, la quale presenta un elevato livello di concentrazione (la distanza fra i quartili e la mediana e quasi nulla). Con riferimento al 50% dei valori centrali si nota inoltre che i range dei due gruppi non si sovrappongono. Le due distribuzioni sono simmetriche in quanto il primo ed il terzo quartile sono alla stessa distanza dalla mediana. Infine, possiamo osservare che entrambe le distribuzioni mostrano la presenza di possibili potenziali outliers.

# ANALISI DISCRIMINANTE LINEARE

Carichiamo la libreria MASS e procediamo alla analisi discriminate sul nostro training set:

```
library(MASS)

LDA <- lda(Classes~., data = train.data)
```

Mostriamo l'output ottenuto:

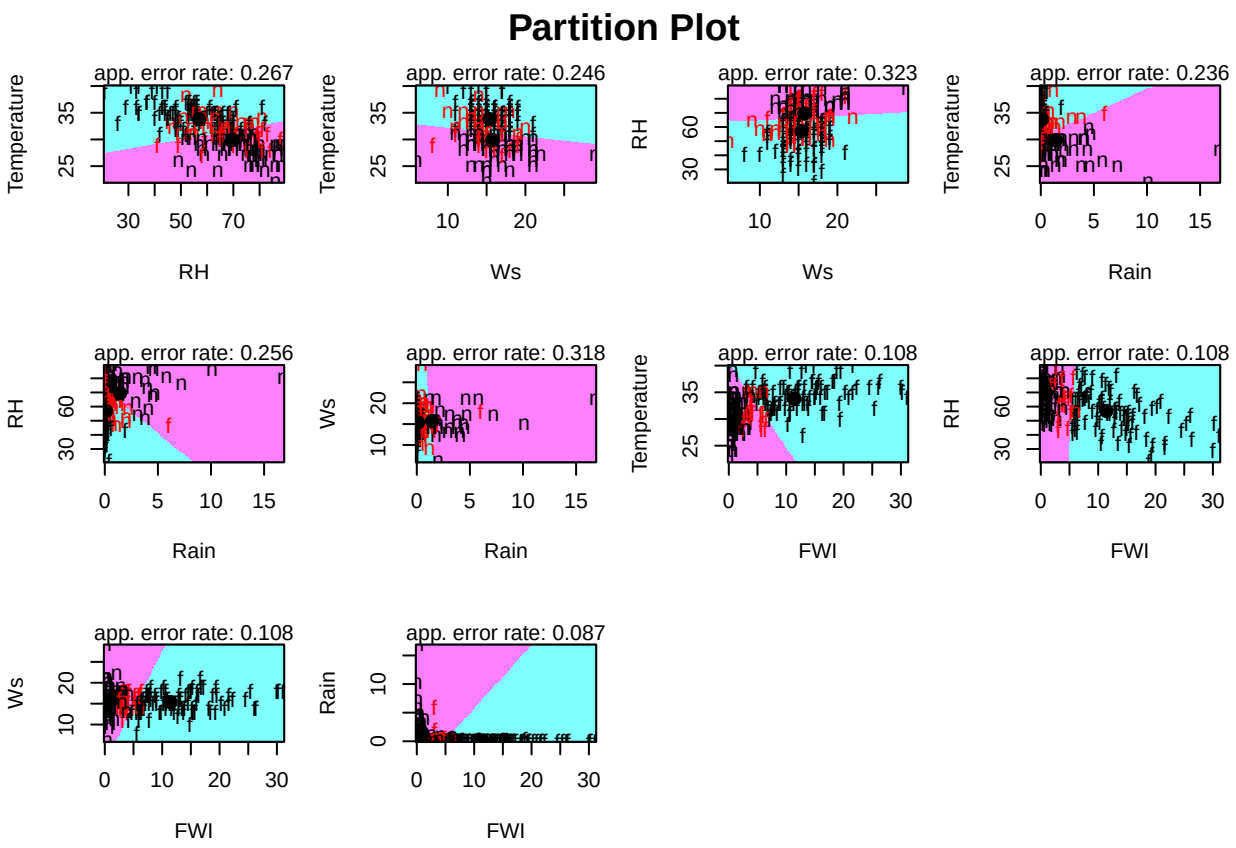
```
LDA
```

```
## Call:
## lda(Classes ~ ., data = train.data)
##
## Prior probabilities of groups:
##      fire  not fire
## 0.5641026 0.4358974
##
## Group means:
##          Temperature       RH       Ws      Rain   FWI
## fire        33.80000 56.93636 15.38182 0.1090909 11.44
## not fire    29.87059 69.54118 15.75294 1.4388235  0.96
##
## Coefficients of linear discriminants:
##                     LD1
## Temperature -0.10127899
## RH          -0.01148306
## Ws           0.04174576
## Rain         0.12904198
## FWI         -0.17295902
```

Nella prima parte dell'output (Prior probabilities of groups) la funzione definisce a priori delle probabilità: queste vengono definite considerando le frequenze relative delle classi. Group means: sono le medie di gruppo (cioè le coordinate dei centroidi dei gruppi). Sono in 5 dimensioni perchè sono 5 le variabili che sto considerando. Coefficients of linear discriminants: in corrispondenza di ogni variabile troviamo la nuova variabile generata dalla compinazione lineare delle variabili originarie (LD1). I valori riportati per ogni variabile sono gli scores della funzione discriminante. Gli scores sono i coefficienti della combinazione lineare e quindi gli autovettori che definiscono la funzione discriminante. Nel caso che stiamo affrontando abbiamo 2 gruppi, per cui considerando il numero di variabili disponibili, il numero massimo di funzioni discriminanti è pari a G-1, motivo per cui abbiamo un' unica funzione disciminante.

Carichiamo adesso al libreria klaR, nella quale è presente la funzione partimat() che ci permette di visualizzare per ogni coppia di variabili come opera il classficatore lineare.

```
library(klaR)
partimat(train.data$Classes ~. , data=train.data[,1:5],method="lda")
```




Utilizziamo ora la funzione predict() per testare il modello sul nostro test set:

```
predictions <- predict(LDA, test.data)
names(predictions)
```

```
## [1] "class"     "posterior" "x"
```

Andando a vedere i nomi che caratterizzano l'output della previsione vediamo che otteniamo 3 variabili di riferimento:

- "class" rappresenta la classe prevista per ogni unità statistica;
- "posterior" rappresenta le probabilità a posteriori che una determinata unità statistica appartenga ad una determinata classe;
- "x" rappresenta le coordinate delle unità statistiche date dalla funzione discriminante (LD1)

Calcoliamo ora l'accuracy del modello sul test set:

```
mean(predictions$class==test.data$Classes)
```

```
## [1] 0.8958333
```

Come possiamo vedere otteniamo un valore di accuracy pari a 0.89 circa, ovvero la probabilità di commettere un errore di classificazione è pari a circa il 10%.

# CONFRONTO CLASSIFICATORI

In questa sezione della nostra analisi andremo ad effettuare un operazione di confronto tra diversi classificatori al fine di stabilire quale nel nostro caso di analisi garantisce una migliore accuratezza di previsione. Svolgiamo un operazione di cross-validazione per poter andare a scegliere il modello migliore. Questo perchè alcuni dei classificatori che analizzeremo richiedono un parametro di Tuning, e quindi di conseguenza la procedura viene rilanciata più volte sotto varie condizioni di inizializzazione, per poter poi scegliere la parametrizzazione migliore. Questo è possibile farlo mediante la funzione trainControl():

- con il metodo "repeatedcv" indichiamo alla funzione di adottare un approccio di K-fold CV, in cui fissiamo a 10 il numero di blocchi.
- con repeats=3 indichiamo alla funzione che la procedura deve essere ripetuta 3 volte: la ripartizione dei blocchi è quindi una ripartizione casuale e non viene effettuata in sequenza, ma le varie unità vengono partizionate in 10 blocchi in maniera casuale.

```
control <- trainControl(method="repeatedcv", number=10, repeats=3)
```

L'oggetto "control" conterrà il "disegno di controllo performance" delle procedure. Procediamo ora tramite la funzione train() a lanciare i avri classficatori. Il setting di questa funziona prevede la specificarzione dela variabile di riferimento, nel nostro caso Classes, il metodo di classificazione, il Dataset da considerare, il "trControl" per specificare alla funzione il disegno di che vogliamo eseguire per allenare il modello.

I classficatori che andremo ad usare sono:

- rpart: è la regressione per alberi di classificazione (alberi di regressione nel caso di variabili quantitative). Questo metodo prevedere la segmentazione dello spazio dei predittori in un numero semplice di regioni. Le fasi prevedono la suddivisione dello spazio dei predittori in R regioni distinte e non sovrapposte; per ogni osservazione che cade nella regione j-esima si fa sempre la stessa previsione che è la media delle variabili (dato che la var. dipendente è qualitativa, le previsioni finali nelle foglie dell'albero saranno caratterizzate dalla moda e non dalla media). Più ramificazioni o nodi ha l'albero è più sarà flessibile il modello. Il procedimento di splitting è di tipo top-down (parte che tutti appartengono ad una sola regione) e sfrutta la minimizzazione del Sum of Squares of Error.
- lda: Linear Discriminant Analysis permette di ottenere una funzione matematica utilizzabile per assegnare, il più correttamente possibile, ulteriori unità statistiche ad uno dei gruppi conosciuti a priori. Più nel dettaglio, i pesi da attribuire alle variabili osservate sulla funzione lineare sono ottenuti attraverso la funzione di Fisher che contempla la devianza tra i gruppi e dentro i gruppi; tanto più alto è il risultato e tanto più robusto è il modello ottenuto (maggiore è la devianza TRA gruppi meglio è). Questo modello differisce dalla cluster analysis perchè la prima ha un carattere previsivo mentre quest'ultima esplorativo.
- svm lineare e radiale: il Support Vector Machine è un classificatore lineare (o non lineare sfruttando i kernel) binario non probabilistico che sfrutta un iperpiano. Il miglior iperpiano è quello più distante tra i punti più vicini delle due (o più) classi e con margine maggiore fra questi punti affinchè l'errore di generalizzazione commesso dal classificatore sia minimo. Il margine è un area simmetrica all'iperpiano che incorpora l'incertezza di classificazione (in essa si possono trovare valori non correttamente classificati); un approccio hard non ammette errori (epsilon=0) mentre uno soft li comprende. Trovare un trade-off tra i due è necessario per un modello ottimale. Tale modello è concettualmente vicino alla rete neurale ma presenta vantaggi computazionali grazie ad un algoritmo più efficiente (programmazione quadratica convessa con vincoli di uguaglianza, il valore del parametro può oscillare entro un limite) ed una eguale flessibilità per funzioni non lineari complesse.
- knn: K-nearest neighbors è l'algoritmo più semplice tra quelli utilizzati nell'apprendimento automatico. Tale modello classifica le osservazioni basandosi sulle caratteristiche degli oggetti vicini a quello considerato, quindi l'oggetto in esame viene classificato in base al voto di pluralità dei suoi vicini risultando assegnato alla classe più comune nell'area di esame.

- rf: è un modello derivato da gli alberi di decisione, risolve l'overfitting che caratterizza gli alberi di decisione.

```
set.seed(123)
fit.cart <- train(Classes~., data=train.data, method="rpart", trControl=control)

set.seed(123)
fit.lda <- train(Classes~., data=train.data, method="lda", trControl=control)

set.seed(123)
fit.svm <- train(Classes~., data=train.data, method="svmLinearWeights2", trControl=control)

set.seed(123)
fit.svmR <- train(Classes~., data=train.data, method="svmRadial", trControl=control)

set.seed(123)
fit.knn <- train(Classes~., data=train.data, method="knn", trControl=control)

set.seed(123)
fit.rf <- train(Classes~., data=train.data, method="rf", trControl=control)
```

Cosideriamo tutti gli output ottenuti e li organizziamo in una lista (mettendoli quindi tutti insieme) in modo tale da favorire in un unico comando il confronto della performance dei veri classificatori. Effettuiamo ciò mediante la funzione resamples()che ci permette di mettere in un unica lista gli output dei vari classificatori per poter sfuttare poi la funzione summary(), che restituirà una serie di indici utili per effettuare un primo confronto.

```
results <- resamples(list(CART=fit.cart, LDA=fit.lda,
        SVM_Linear=fit.svm, SVM_Radial=fit.svmR, KNN=fit.knn, RF=fit.rf))
summary(results)
```

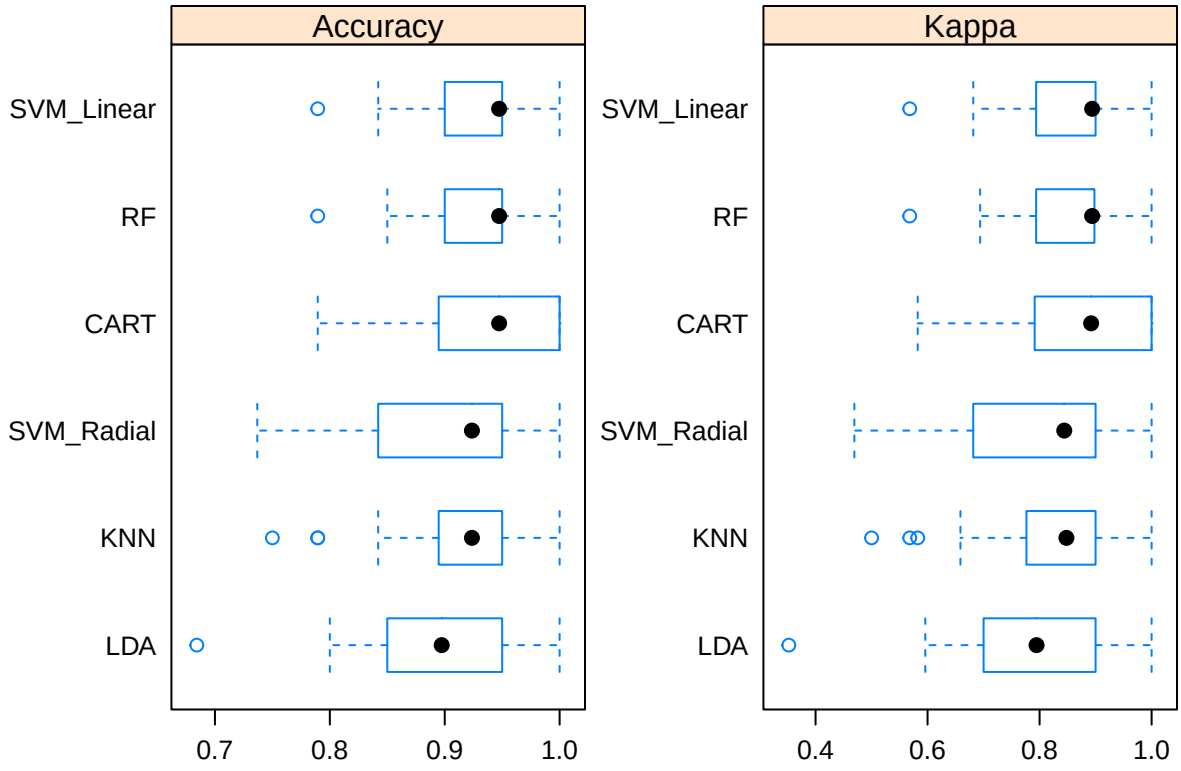
```
##
## Call:
## summary.resamples(object = results)
##
## Models: CART, LDA, SVM_Linear, SVM_Radial, KNN, RF
## Number of resamples: 30
##
## Accuracy
##                  Min.   1st Qu.    Median      Mean   3rd Qu. Max. NA's
## CART       0.7894737 0.8947368 0.9473684 0.9299123 1.0000000    1    0
## LDA        0.6842105 0.8500000 0.8973684 0.9007018 0.9500000    1    0
## SVM_Linear 0.7894737 0.9000000 0.9473684 0.9365789 0.9500000    1    0
## SVM_Radial 0.7368421 0.8440789 0.9236842 0.8973684 0.9493421    1    0
## KNN        0.7500000 0.8947368 0.9236842 0.9145614 0.9500000    1    0
## RF         0.7894737 0.9000000 0.9473684 0.9333333 0.9500000    1    0
##
## Kappa
##                  Min.   1st Qu.    Median      Mean   3rd Qu. Max. NA's
## CART       0.5824176 0.7912088 0.8920141 0.8573894 1.0000000    1    0
## LDA        0.3522727 0.7000000 0.7945943 0.8007560 0.9000000    1    0
## SVM_Linear 0.5681818 0.7948558 0.8938547 0.8714314 0.9000000    1    0
## SVM_Radial 0.4692737 0.6861732 0.8440766 0.7934090 0.8984637    1    0
```

```
## KNN         0.5000000 0.7783757 0.8479175 0.8262054 0.8994898    1    0
## RF          0.5681818 0.7948558 0.8938547 0.8637080 0.8979592    1    0
```

Number of resamples: 30 indica che abbiamo un numero di ricampionamenti 3 * 10, perchè abbiamo fatto 3 prove utilizzando un metodo Key-fold-cv con K=10. Di conseguenza per ogni blocco otteniamo dei risultati, e per cui si genera una sorta di distribuzione per gli indici considerati. Gli indici presi in considerazione sono: l'indice di Accuracy e l'indice Kappa. L'indice Kappa è un indice di concordanza utilizzabile in classificazioni qualitative (in questo caso dicotomica) che valuta il livello di concordanza tra due modelli, esso può andare da 0 a 1. Lavora sulla matrice di confusione calcolando la proporzione di concordanza dovuta al caso.
Trovata tale proporzione si procede ad applicare la formula di Cohen che vede al numeratore della frazione la differenza tra la proporzione osservata (quando i due modelli concordano, in questo caso osservato e previsto) e la proporzione di concordanza dovuta al caso, al denominatore 1 meno la proporzione di concordanza dovuta al caso.
In sintesi è sufficiente che Kappa sia maggiore di 0.8 affinchè vi sia un accordo quasi perfetto tra i due valutatori.è un indice che viene utilizzato per capire se i risultati della classificazione sono casuali o meno. L'indice Accuracy rappresenta invece in numero di osservazioni correttamente classificate rispetto al totale. Dato che abbiamo tanti risultati è possibile creare una distribuzione di questi indici. Di questa distribuzione, col comando summary osserviamo: il minimo, il primo quartile, la mediana, la media, il terzo quartile, ed il valore massimo per ogni classificatore considerato. In questo modo otteniamo delle info. che ci permettono di effettuare un primo confronto dei classificatori rispetto a questi indici.

E' possibile anche migliorare la visualizzazione delle performance dei classificatori mediante delle rappresentazioni grafiche. In particolare utiliziamo una rappresentazione grafica di tipo Box Plot ottenuti mediante la funzione bwplot():

```
scales <- list(x=list(relation="free"), y=list(relation="free"))
bwplot(results, scales=scales)
```



Nel'output vengono rappresentate le distribuzioni ottenute. Tutte le distibuzioni caratterizzanti la performance vengono rappresentate per riga: Ogni riga (Box plot) corrisponde ad un classificatore. I classificatori vengono ordinati dall'alto verso il basso secondo la loro performance. In questo caso sembrerebbe che il Support-Vector-Machine lineare ed il Random Forest, con le rispettive parametrizzazioni scelte nel momento in cui sono state impostate le relative funzioni, tendono a fornire le performance migliori, in quanto presentano una minore variabilità rispetto alle distribuzioni delle performance relative agli altri classificatori . Qundi anche in media (o in mediana) tendono a mostrare indici di performance (rappresentati in termini di Accuracy e Kappa) migliori degli altri classificatori.

La valutazione dei classificatori considerati prosegue con una verifica delle performance a livello di **Test set** , in modo tale da testare le capacità previsive dei classificatori su nuove osservazioni non considerate in fase di learning (tuning) mediante la funzione predict() alla quale vengono passati come argomenti:

- il modello Fittato (ad esempio il modello cart, contenuto nell'oggetto "fit.cart"),
- i dati su cui testare il modello ("test.data").

```
# CART
pCART <- predict(fit.cart, test.data)
# LDA
pLDA <- predict(fit.lda, test.data)
# SVM Linear
pSVM_L <- predict(fit.svm, test.data)
# SVM Radial
pSVM_R <- predict(fit.svmR, test.data)
# KNN
pKNN <- predict(fit.knn, test.data)
# Random Forest
pRF <- predict(fit.rf, test.data)
```

Per ciascuna metodologia di classificazione viene successivamente calcolata la relativa matrice di confusione mediante la funzione confusionMatrix(), alla quale vengono passati come argomenti:

- le labels osservate all'interno del Test set ("test.data$Classes"),
- le labels previste dai classificatori.

La matrice di confusione mette a confronto la labels osservate (che sono contenute nella variabile "Classes" del test set) con le labels previste (contenute per es. nell'oggetto pCART). Da tale matrice vengono inoltre derivati una serie di indicatori di performance, i quali considerano determinati aspetti riguardanti le performance del classificatore considerato.

```
# CART
cmCART <- confusionMatrix(test.data$Classes, pCART)
cmCART
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       26        1
##   not fire    2       19
##
##                Accuracy : 0.9375
##                  95% CI : (0.828, 0.9869)
```

```
##     No Information Rate : 0.5833
##     P-Value [Acc > NIR] : 4.015e-08
##
##                   Kappa : 0.8723
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.9286
##             Specificity : 0.9500
##          Pos Pred Value : 0.9630
##          Neg Pred Value : 0.9048
##              Prevalence : 0.5833
##          Detection Rate : 0.5417
##    Detection Prevalence : 0.5625
##       Balanced Accuracy : 0.9393
##
##        'Positive' Class : fire
##
```

Con riferimento al modello CART (classification and regression Trees) la matrice di confusione mostra che il numero di osservazioni correttamente classificate (TP ed FP riportati nella diagonale principale) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN ossia gli elementi fuori dalla diagonale principale della matrice). L'incide di accuratezza (Accuracy) conferma quanto appena affermato: infatti si può osservare che più del 93% delle osservazioni viene correttamente classificato (ossia vi è concordanza tra le classi osservate e le classi previste). Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity), ossia il rapporto fra i TP e la somma dei TP e FN, risulta essere pari a 0.9286; ciò sta ad indicare che il classificatore classifica correttamente il 92% delle aree(osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity), ossia il rapporto fra i TN e la somma dei TN e FP, risulta essere pari a 0.95; ciò sta ad indicare che il classificatore classifica correttamente il 95% delle aree(osservazioni) in cui non si registrano incendi

```
# LDA
cmLDA <- confusionMatrix(test.data$Classes, pLDA)
cmLDA
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       23        4
##   not fire    1       20
##
##                Accuracy : 0.8958
##                  95% CI : (0.7734, 0.9653)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : 6.84e-09
##
##                   Kappa : 0.7917
##
```

```
##  Mcnemar's Test P-Value : 0.3711
##
##             Sensitivity : 0.9583
##             Specificity : 0.8333
##          Pos Pred Value : 0.8519
##          Neg Pred Value : 0.9524
##              Prevalence : 0.5000
##          Detection Rate : 0.4792
##    Detection Prevalence : 0.5625
##       Balanced Accuracy : 0.8958
##
##        'Positive' Class : fire
##
```

Con riferimento al modello LDA (Linear Discriminant Analysis) la matrice di confusione mostra che anche in questo caso il numero di osservazioni correttamente classificate (TP ed FP) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN). Ciò viene ulteriormente confermato dall'indice di accuratezza (Accuracy) il quale indica che l'89% delle osservazioni viene correttamente classificato. Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity) risulta essere pari a 0.9583 ; ciò sta ad indicare che il classificatore classifica correttamente il 95% circa delle aree (osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity) risulta essere pari a 0.8333; ciò sta ad indicare che il classificatore classifica correttamente l'83% circa delle aree (osservazioni) in cui non si registrano incendi boschivi.

```
# SVM_Linear
cmSVM_L <- confusionMatrix(test.data$Classes, pSVM_L)
cmSVM_L
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       26        1
##   not fire    2       19
##
##                Accuracy : 0.9375
##                  95% CI : (0.828, 0.9869)
##     No Information Rate : 0.5833
##     P-Value [Acc > NIR] : 4.015e-08
##
##                   Kappa : 0.8723
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.9286
##             Specificity : 0.9500
##          Pos Pred Value : 0.9630
##          Neg Pred Value : 0.9048
##              Prevalence : 0.5833
##          Detection Rate : 0.5417
##    Detection Prevalence : 0.5625
```

```
##       Balanced Accuracy : 0.9393
##
##        'Positive' Class : fire
##
```

Con riferimento al modello SVM (Linear Support Vector Machine) la matrice di confusione mostra ancora una volta che il numero di osservazioni correttamente classificate (TP ed FP) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN). Ciò viene confermato dall'indice di accuratezza (Accuracy) il quale indica che il 93% circa delle osservazioni viene correttamente classificato. Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity) risulta essere pari a 0.9286 ; ciò sta ad indicare che il classificatore classifica correttamente il 92% circa delle aree (osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity) risulta essere pari a 0.9500 ; ciò sta ad indicare che il classificatore classifica correttamente il 95% delle aree (osservazioni) in cui non si registrano incendi boschivi.

```r
# SVM_Radial
cmSVM_R <- confusionMatrix(test.data$Classes, pSVM_R)
cmSVM_R
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       25        2
##   not fire    2       19
##
##                Accuracy : 0.9167
##                  95% CI : (0.8002, 0.9768)
##     No Information Rate : 0.5625
##     P-Value [Acc > NIR] : 8.116e-08
##
##                   Kappa : 0.8307
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.9259
##             Specificity : 0.9048
##          Pos Pred Value : 0.9259
##          Neg Pred Value : 0.9048
##              Prevalence : 0.5625
##          Detection Rate : 0.5208
##    Detection Prevalence : 0.5625
##       Balanced Accuracy : 0.9153
##
##        'Positive' Class : fire
##
```

Con riferimento al modello SVM (Radial Support Vector Machine) la matrice di confusione mostra ancora una volta che il numero di osservazioni correttamente classificate (TP ed FP) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN). L'indice di accuratezza (Accuracy) indica che l'91% delle osservazioni viene correttamente classificato. Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity) risulta essere pari a 0.9259 ; ciò sta ad indicare che il classificatore classifica correttamente l'92% circa delle aree (osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity) risulta essere pari a 0.9048; ciò sta ad indicare che il classificatore classifica correttamente l'90% circa delle aree (osservazioni) in cui non si registrano incendi boschivi.

```
# KNN
cmKNN <- confusionMatrix(test.data$Classes, pKNN)
cmKNN
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       25        2
##   not fire    4       17
##
##                Accuracy : 0.875
##                  95% CI : (0.7475, 0.9527)
##     No Information Rate : 0.6042
##     P-Value [Acc > NIR] : 3.814e-05
##
##                   Kappa : 0.7433
##
##  Mcnemar's Test P-Value : 0.6831
##
##             Sensitivity : 0.8621
##             Specificity : 0.8947
##          Pos Pred Value : 0.9259
##          Neg Pred Value : 0.8095
##              Prevalence : 0.6042
##          Detection Rate : 0.5208
##    Detection Prevalence : 0.5625
##       Balanced Accuracy : 0.8784
##
##        'Positive' Class : fire
##
```

Con riferimento al modello KNN (Keyniar Neist Neightbor) la matrice di confusione mostra ancora una volta che il numero di osservazioni correttamente classificate (TP ed FP) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN). L'indice di accuratezza (Accuracy) indica che l'87.5% circa delle osservazioni viene correttamente classificato. Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity) risulta essere pari a 0.8621 ; ciò sta ad indicare che il classificatore classifica correttamente l'86% circa delle aree (osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity) risulta essere pari a 0.8947; ciò sta ad indicare che il classificatore classifica correttamente l' 89% circa delle aree (osservazioni) in cui non si registrano incendi boschivi.

```
# RANDOM FOREST
cmRF <- confusionMatrix(test.data$Classes, pRF)
cmRF
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction fire not fire
##   fire       27        0
##   not fire    2       19
##
##                Accuracy : 0.9583
##                  95% CI : (0.8575, 0.9949)
##     No Information Rate : 0.6042
##     P-Value [Acc > NIR] : 1.617e-08
##
##                   Kappa : 0.9144
##
##  Mcnemar's Test P-Value : 0.4795
##
##             Sensitivity : 0.9310
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 0.9048
##              Prevalence : 0.6042
##          Detection Rate : 0.5625
##    Detection Prevalence : 0.5625
##       Balanced Accuracy : 0.9655
##
##        'Positive' Class : fire
##
```

Con riferimento al modello Random Forest la matrice di confusione mostra ancora una volta che il numero di osservazioni correttamente classificate (TP ed FP) è nettamente superiore rispetto al numero di osservazioni erroneamente classificate (FP e FN). L'indice di accuratezza (Accuracy) indica che il 95% circa delle osservazioni viene correttamente classificato. Con riferimento alle due possibili tipologie di errore che si possono commettere osserviamo che:

- l'indice di sensitività (Sensitivity) risulta essere pari a 0.9310; ciò sta ad indicare che il classificatore classifica correttamente il 93% circa delle aree (osservazioni) in cui si registrano incendi boschivi.
- l'indice di specificità (Specificity) risulta essere pari a 1; ciò sta ad indicare che il classificatore classifica correttamente tutte le aree (osservazioni) in cui non si registrano incendi boschivi.

Infine andiamo a riportare in una tabella i valori massimi di accuracy e i complementari errori di classficazione sia per il train che per il test set per ognuno dei classificatori utilizzati:

| ModelType | TrainAcc | Train_misscl_Er | TestAcc | Test_misscl_Er |
|---|---|---|---|---|
| CART | 0.930 | 0.070 | 0.938 | 0.062 |
| LDA | 0.901 | 0.099 | 0.896 | 0.104 |
| SVM_L | 0.937 | 0.063 | 0.938 | 0.062 |
| SVM_R | 0.897 | 0.103 | 0.917 | 0.083 |
| KNN | 0.915 | 0.085 | 0.875 | 0.125 |
| Random forest | 0.933 | 0.067 | 0.958 | 0.042 |

Analizzando i risultati notiamo che tutti i classificatori utilizzati sono altamente performanti. In particolare sul train il valore migliore lo otteniamo tramite l'utilizzo del Random Forest, mentre sul test set il valore migliore è otteneuto tramite l'utilizzo dl Support Vector Machine lineare.