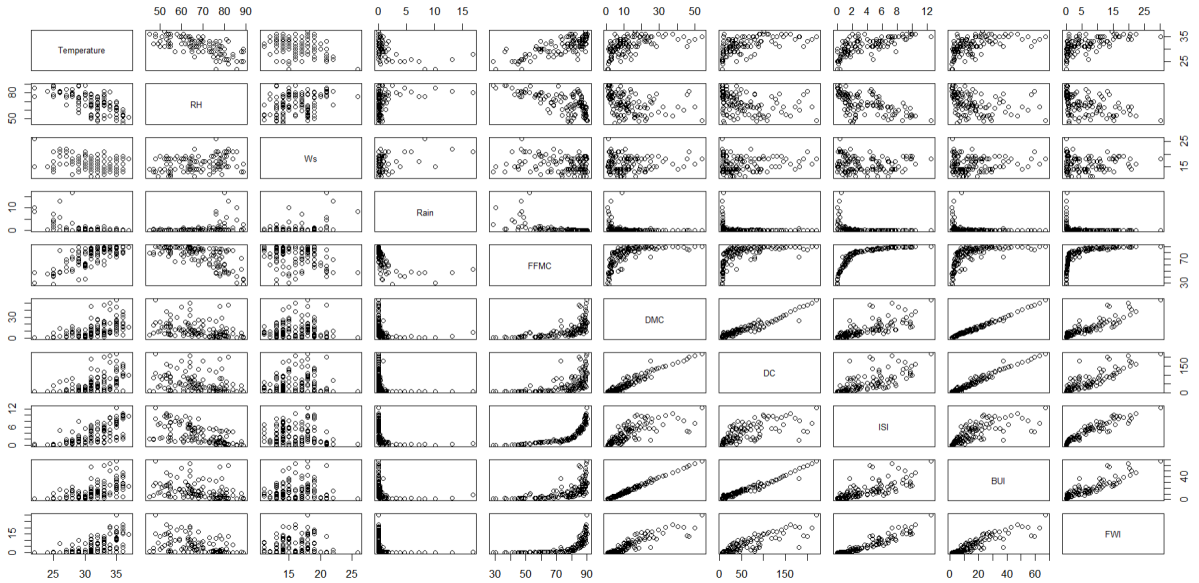
Incendios forestales
Análisis multivariado



ITBA

Ignacio de Achával
Leg. 60.696

18/12/2022

Contenidos

Descripción del caso de estudio

Conjunto de datos

El conjunto de datos a analizar consta de 122 observaciones diarias de la región de Bejaia (noreste de Argelia), compuesto por factores climáticos de la región y variables del sistema FWI (Fire weather index) y una variable de respuesta que indica si ocurrieron o no incendios forestales en la fecha. El mismo puede encontrarse en: https://archive.ics.uci.edu/ml/datasets/Algerian+Forest+Fires+Dataset++

Variables

Factores climáticos:

- Temperature: Temperatura (grados Celsius) registrada al mediodía en la fecha
- RH : humedad relativa (%)
- Ws: velocidad del viento (km/h)
- Rain: precipitaciones totales ocurridas en el día (mm)
- incendio: booleana.

Variables del sistema FWI:

- FFMC: Fine Fuel Moisture Code
- DMC (Duff Moisture Code)
- DC (Drought Code)
- ISI: Initial Spread Index
- BUI: Buildup Index
- FWI: Fire Weather Index

A continuación se presenta una estructura del sistema FWI, que muestra los factores climáticos asociados a cada índice:

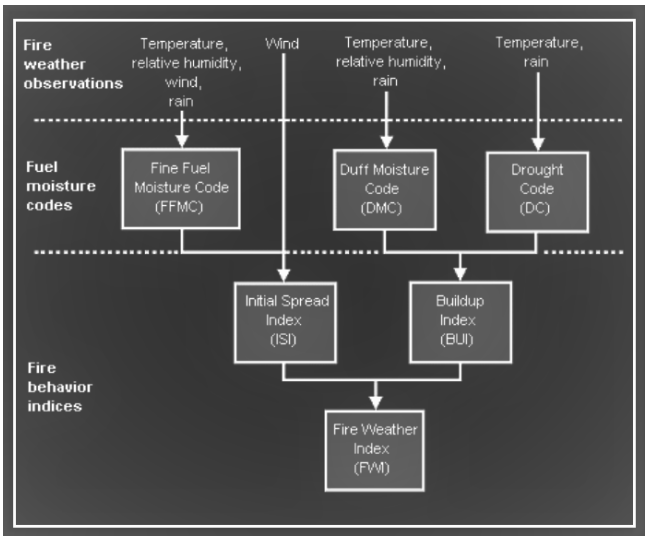


Figura 1. Diagrama FWI

Análisis exploratorio

Distribución de las variables

Se realizaron ensayos para la diferencia de medias de todas las variables numéricas del conjunto de datos contra la variable incendio, resultando todas ellas, a excepción de la variable Ws (velocidad del viento, km/h), significativamente distintas a un nivel de significancia del 5%,

Matriz de correlación e interpretación

Se construyó la siguiente matriz de correlación:

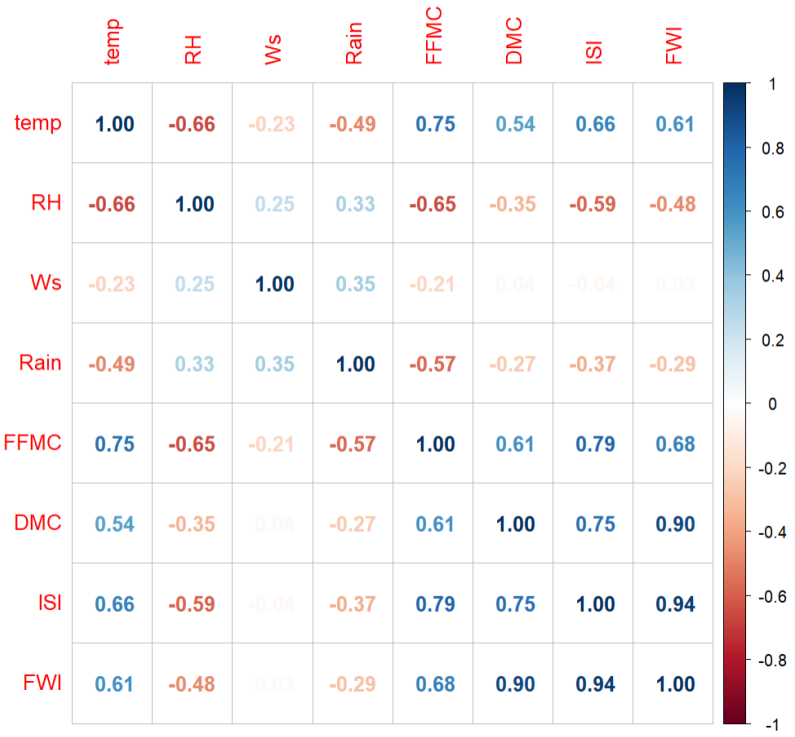


Figura 2. Matriz de correlación de variables explicativas

De la misma se observa que las variables del sistema FWI se hallan fuertemente correlacionadas (positivamente) entre sí, siendo, de los factores climáticos, la temperatura el factor que muestra una correlación más fuerte con ellos, también positiva. A su vez, la lluvia, la velocidad del viento y la humedad relativa guardan correlaciones positivas entre sí, y una correlación negativa con las variables del sistema FWI, a excepción de la velocidad del viento. La temperatura se correlaciona negativamente con el resto de los factores climáticos.

Reducción de dimensión

Sobre las variables previamente mencionadas, se realizó un análisis de componentes principales, tomando dim = 3, explicando con ello el 83.8% de la varianza. Al graficar en tres dimensiones, coloreando contra el grupo incendio, aparecen dos grupos diferenciables.

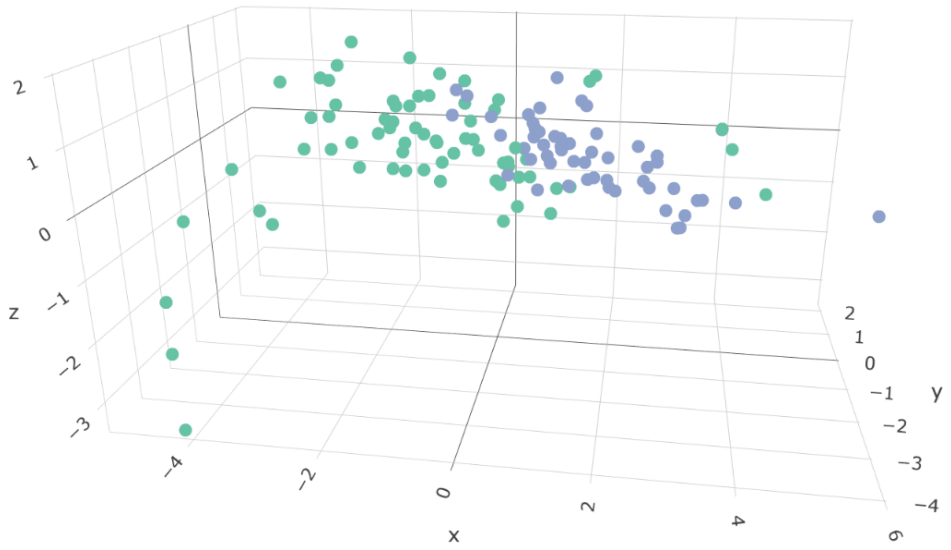


Figura 3. Scatterplot 3d de componentes principales

Relación entre componentes hallados y variables originales

	temp	RH	Ws	Rain	FFMC	DMC	ISI	FWI
PC1	0.3895326	-0.3340097	-0.09379566	-0.2607381	0.418011734	0.3682012	0.426342389	0.4096903
PC2	0.1494872	-0.2058769	-0.67903096	-0.4359866	0.109382601	-0.3491463	-0.201341122	-0.3312024
PC3	-0.2048005	0.7016786	-0.07750395	-0.6027672	-0.008224047	0.2828228	-0.007557066	0.1274962

Tabla 1. Componentes principales y variables originales

PC1 - Este componente separa los aglomerados en función de los indicadores ISI, FFMC y FWI.

PC2 - Este componente disminuye para las observaciones de días lluviosos y ventosos.

PC3 - Este componente podría describir días húmedos no lluviosos.

Análisis de disimilaridad: Incendios

Introducción

Para el siguiente estudio se construyeron dos subconjuntos de datos: uno correspondiente a las observaciones diarias en que se registran incendios; el otro, a aquellos días que no presentaron incendios. El objetivo principal será analizar la similaridad entre los incendios para poder detectar incendios atípicos con respecto a las variables que los describen.

La medida de distancia utilizada fue la distancia de Mahalanobis, calculada entre cada observación con respecto al resto de las observaciones de la muestra. Para ello se definió la función mahalanobis_dist() presente en el anexo.

Días de incendio

Matriz de distancia

Una vez obtenida la matriz de distancias de mahalanobis, la volcamos en el siguiente mapa de calor:

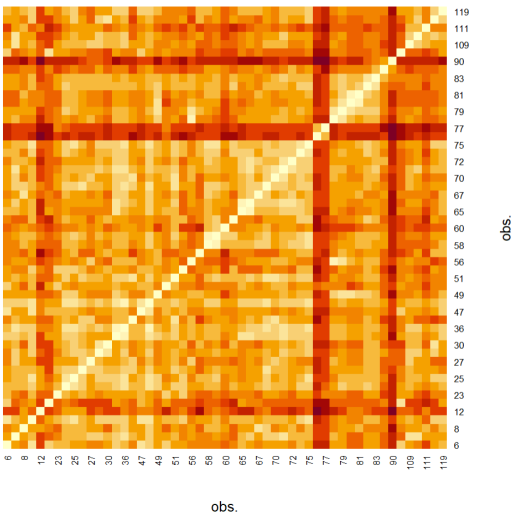


Figura 4. Matriz de distancia de mahalanobis entre observaciones de incendios

Se registra que las observaciones 76, 77 y 90 tienen una distancia de mahalanobis con el resto de las observaciones mayor a lo que el resto de las observaciones presenta; esto es: están alejadas.

Al revisar cada caso particularmente, las observaciones 76 y 77 presentaron niveles de precipitaciones que constituyen valores atípicos para el nivel de precipitaciones

observado en días de incendio (típicamente nulo), como se observa en el histograma de la variable:

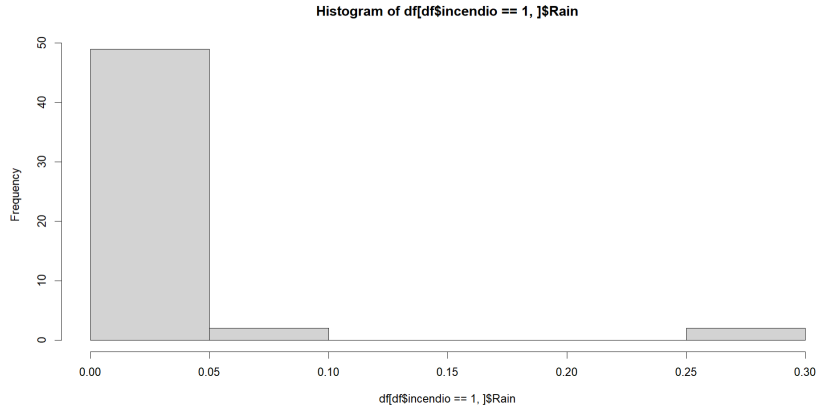


Figura 5. Histograma de precipitaciones (mm)
para observaciones de incendios

De forma similar, la observación 90 alcanza valores atípicos para los índices ISI y FWI.

Escalamiento multidimensional

Se realizó un escalamiento multidimensional sobre la muestra de incendios. Para una mayor interpretabilidad en el plano, fueron seleccionadas 2 dimensiones. Las coordenadas obtenidas volcadas en el plano son:

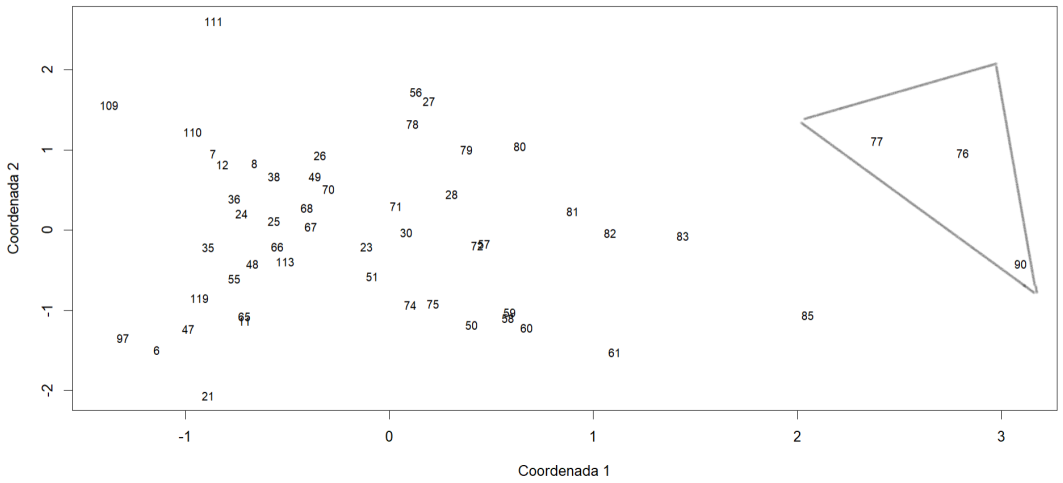


Figura 6. Scatterplot de coordenadas MDS (incendios)

Efectivamente, las observaciones 76, 77 y 90 resultan atípicas. Al momento de clasificar, las obs. 76 y 77 serán removidas del conjunto de entrenamiento, dado que pueden ser representativas de un día sin incendio. No así la observación 90, pues, aunque atípica en tanto incendio, lo es más como no incendio.

Días sin presencia de incendios

Matriz de distancia

Una vez obtenida la matriz de distancias de mahalanobis, esta vez sobre los días en que no se presentaron incendios, la volcamos en el siguiente mapa de calor:

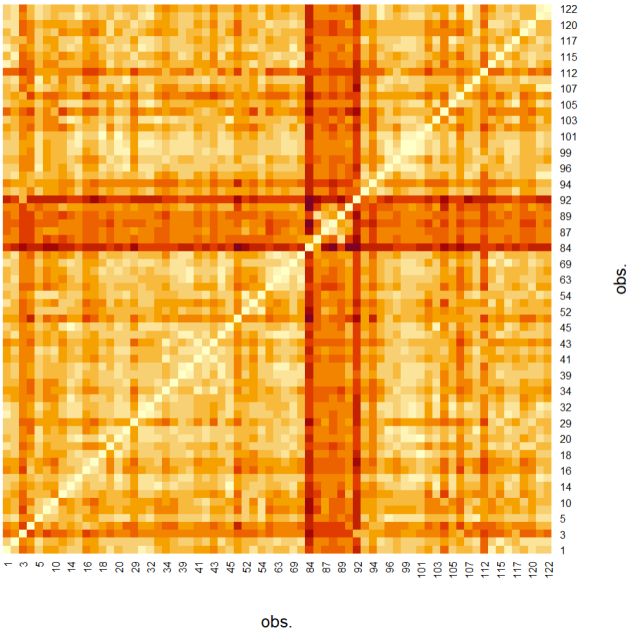


Figura 7. Matriz de distancia de mahalanobis entre observaciones (no incendios)

Se destacan en particular las observaciones entre la 84 y la 92, siendo estas las más distantes del resto del conjunto.
El análisis de cada caso en particular arrojó que la observación 92 presentó precipitaciones muy por fuera de lo regular, mientras que la observación 84 presenta un valor atípico para el índice ISI, siendo más probable clasificarlo incorrectamente entre los días de incendio, pues P(incendio | ISI=30) > P(¬incendio | ISI=30).

Escalamiento multidimensional

Al realizar el escalamiento multidimensional, confirmamos que efectivamente las observaciones se encuentran en los extremos de la nube de puntos.

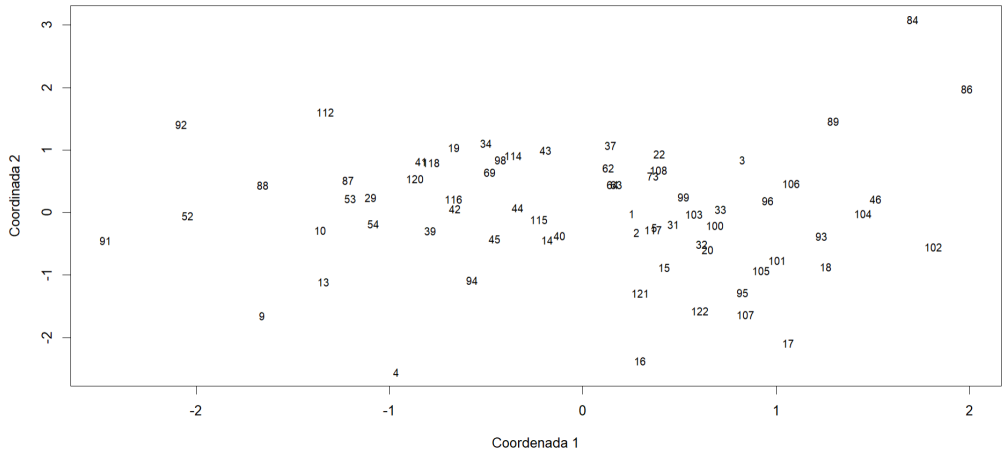


Figura 8. Scatterplot de coordenadas MDS (no incendios)

Conclusiones

El análisis permitió encontrar tanto días donde se presentaron incendios como días en los que no, que resultaron ser atípicos dentro de su respectivo grupo. Al momento de realizar un análisis predictivo sobre el conjunto de datos, se removerán las observaciones 76, 77 y 84.

Escalamiento multidimensional: variables explicativas

Se realizó un escalamiento multidimensional sobre la matriz de correlación de las variables explicativas para lograr un mejor entendimiento de cómo inciden en los índices FWI los factores climáticos registrados. Se tomaron 2 dimensiones para mejorar la interpretabilidad.

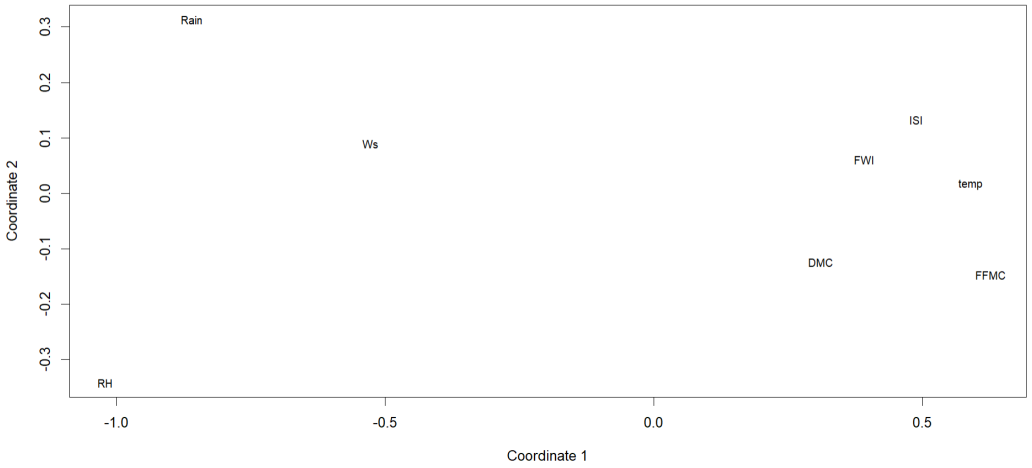


Figura 9. Scatterplot de coordenadas MD
(variables explicativas)

Volcando las coordenadas obtenidas en un scatterplot, de igual manera que podía ya intuirse en la matriz de correlación, la temperatura parece ser un factor clave para el cálculo de los estadísticos del sistema FWI.

Análisis predictivo

Introducción

Se implementó una batería de modelos predictivos (árbol de decisión, random forest, red neuronal, máquina de soporte vectorial) sobre el conjunto de datos con el objetivo de clasificar los días en los cuales, a partir de las variables analizadas, se produciría un incendio. Salvo que se explicite lo contrario, todos los modelos fueron entrenados sobre el mismo conjunto de entrenamiento, previamente definido mediante un muestreo aleatorio de semilla fija contra la variable de respuesta, y correspondiendo a un 70% de las observaciones de la muestra. El testeo fue realizado sobre el conjunto restante del 30% de las observaciones.
Para todos los casos se utilizaron la totalidad de los predictores del conjunto de datos.

Árbol de decisión

Se ejecutó un árbol de decisión con los parámetros por defecto de la librería rpart. Las predicciones del modelo sobre el conjunto de testeo arrojaron la siguiente matriz de confusión:

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 19  2
         1  1 13

               Accuracy : 0.9143
                 95% CI : (0.7694, 0.982)
    No Information Rate : 0.5714
    P-Value [Acc > NIR] : 9.733e-06

                  Kappa : 0.8235

 Mcnemar's Test P-Value : 1

            Sensitivity : 0.9500
            Specificity : 0.8667
         Pos Pred Value : 0.9048
         Neg Pred Value : 0.9286
             Prevalence : 0.5714
         Detection Rate : 0.5429
   Detection Prevalence : 0.6000
      Balanced Accuracy : 0.9083

       'Positive' Class : 0
```

Figura 10. Matriz de confusión
(Árbol de decisión)

Random forest

Se ejecutó un random forest de parámetros (ntree=100, mtry=2). Las predicciones del modelo sobre el conjunto de testeo arrojaron la siguiente matriz de confusión:

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  1
         1  0 14

               Accuracy : 0.9714
                 95% CI : (0.8508, 0.9993)
    No Information Rate : 0.5714
    P-Value [Acc > NIR] : 8.492e-08

                  Kappa : 0.9412

 Mcnemar's Test P-Value : 1

            Sensitivity : 1.0000
            Specificity : 0.9333
         Pos Pred Value : 0.9524
         Neg Pred Value : 1.0000
             Prevalence : 0.5714
         Detection Rate : 0.5714
   Detection Prevalence : 0.6000
      Balanced Accuracy : 0.9667

       'Positive' Class : 0
```

Figura 11. Matriz de confusión
(random forest)

Red neuronal

Se entrenó mediante cross-validation (k=10), sobre el total de la muestra, una red neuronal y se ajustaron los hiper parámetros de tamaño y decay de acuerdo a la siguiente grilla: (size=c(5,10,20),decay=c(0.001,0.01,0.1)).
El accuracy medio de los modelos resultantes fue 0.948834. El tiempo de entrenamiento fue considerablemente mayor al de los demás modelos. La red final contó con 10 neuronas escondidas y un parámetros de decay=0.1, y su estructura es:

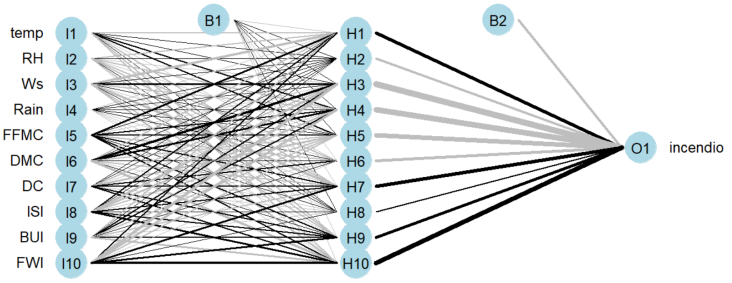


Figura 12. Red neuronal

Máquina de soporte vectorial

Se entrenó una máquina de soporte vectorial de kernel lineal y ajustando el parámetro de costo iterando sobre los siguientes valores: c(0.001, 0.01, 0.1, 0.5, 1). El parámetro de costo finalmente fue 1. Las predicciones del modelo sobre el conjunto de testeo arrojaron la siguiente matriz de confusión:

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 19  0
         1  1 15

               Accuracy : 0.9714
                 95% CI : (0.8508, 0.9993)
    No Information Rate : 0.5714
    P-Value [Acc > NIR] : 8.492e-08

                  Kappa : 0.9421

 Mcnemar's Test P-Value : 1

            Sensitivity : 0.9500
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 0.9375
             Prevalence : 0.5714
         Detection Rate : 0.5429
   Detection Prevalence : 0.5429
      Balanced Accuracy : 0.9750

       'Positive' Class : 0
```

Figura 13. Matriz de confusión
(Máquina de soporte vectorial)

También se entrenaron máquinas de soporte vectorial con kernels polinómicos, radiales y sigmoideos, todos ellos arrojando desempeños inferiores a los de los modelos presentes en el informe.

Conclusiones

Todos los modelos demostraron un muy buen desempeño, encontrándose por encima del criterio de no información y con accuracies por encima del 90%. Cabe destacar que los intervalos de confianza del 95% para el accuracy de los modelos se superponen, por lo que no es posible afirmar que alguno tenga una precisión más alta que la de los demás. Sin embargo, el árbol de decisión resultó ser el modelo que menor especificidad (indicador que más valoramos por la naturaleza del problema) alcanzó, seguido por el random forest, habiendo alcanzado la mayor especificidad la máquina de soporte vectorial, motivo por el cual lo conservamos.

Conclusiones generales y próximos pasos

Una variedad de modelos de clasificación pueden alcanzar muy buenos resultados para la predicción de incendios forestales a partir de los factores climáticos de la región y los índices elaborados.
Podría resultar fructífero atender a la problemática como un trabajo de series temporales multivariadas. Si podemos considerar la pregunta *¿Qué combinación de factores concluye en un incendio forestal?*, como resuelta a partir de la evaluación de cualquiera de los modelos descritos anteriormente (o alguna infinidad de otros) frente a un punto de *forecast* dado, la siguiente cuestión sobre la cual deberemos preocuparnos es: *¿Cuándo volverán a combinarse los factores de manera que se produzca un incendio forestal?*

Anexo

Código R utilizado

```
require(dplyr)
require(ggplot2)
require(funModeling)
require(ltm)
require(corrplot)
require(plotly)
require(caret)
require(nnet)
require(NeuralNetTools)
require(randomForest)
require(rpart)
require(rpart.plot)
require(e1071)

#lectura datos
df <- read.csv("Algerian_forest_fires_dataset_UPDATE.csv",sep=";")
df <- df[1:122,5:ncol(df)-1]

colnames(df) <- c("temp","RH","Ws","Rain","FFMC","DMC","DC","ISI","BUI","FWI","incendio"
)
#limpieza

df$temp <- as.numeric(df$temp)
df$RH <- as.numeric(df$RH)
df$Ws <- as.numeric(df$Ws)
df$Rain <- as.numeric(df$Rain)
df$FFMC <- as.numeric(df$FFMC)
df$DMC <- as.numeric(df$DMC)
df$DC <- as.numeric(df$DC)
df$ISI <- as.numeric(df$ISI)
df$BUI <- as.numeric(df$BUI)
df$FWI <- as.numeric(df$FWI)
df$incendio <- ifelse(df$incendio=="fire   ",1,0)

funModeling::df_status(df)

# df <- df %>% filter(!is.na(DC),!is.na(FWI))
```

```
#EDA
df %>% ggplot(aes(x=temp,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=RH,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=Ws,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=log(Rain),fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=FFMC,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=DMC,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=ISI,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=BUI,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% ggplot(aes(x=FWI,fill=factor(incendio)))+geom_density(alpha=0.8)
df %>% summary()

#ensayos para la diferencia de medias. nivel de significación=0.05
t.test(df[df$incendio==1,]$temp,df[df$incendio==0,]$temp) #significativo
t.test(df[df$incendio==1,]$RH,df[df$incendio==0,]$RH) #significativo
t.test(df[df$incendio==1,]$Ws,df[df$incendio==0,]$Ws) #no significativo
t.test(df[df$incendio==1,]$Rain,df[df$incendio==0,]$Rain) #significativo
t.test(df[df$incendio==1,]$FFMC,df[df$incendio==0,]$FFMC) #significativo
t.test(df[df$incendio==1,]$DMC,df[df$incendio==0,]$DMC) #significativo
t.test(df[df$incendio==1,]$ISI,df[df$incendio==0,]$ISI) #significativo
t.test(df[df$incendio==1,]$BUI,df[df$incendio==0,]$BUI) #significativo
t.test(df[df$incendio==1,]$FWI,df[df$incendio==0,]$FWI) #significativo


df_num <- df %>% select(temp,RH,Ws,Rain,FFMC,DMC,DC,ISI,BUI,FWI)
plot(df_num)
df_num
cor(df_num)
corrplot(cor(df_num),method="number")

heatmap(cor(df_num),method="number",symm=T,Colv=NA,Rowv=NA)

biserial.cor(df$temp,df$incendio)
biserial.cor(df$RH,df$incendio)
biserial.cor(df$Rain,df$incendio)
biserial.cor(df$FFMC,df$incendio)
biserial.cor(df$DMC,df$incendio)
biserial.cor(df$DC,df$incendio)
biserial.cor(df$ISI,df$incendio)
biserial.cor(df$BUI,df$incendio)
biserial.cor(df$FWI,df$incendio)
```

```
df %>%
ggplot(aes(y=temp,x=ISI,color=factor(incendio)))+geom_point()+ggtitle("Temperatura vs
ISI")


#reducción de dimensión
df
df_num <- df %>% dplyr::select(temp,RH,Ws,Rain,FFMC,DMC,ISI,FWI)
df_num

X <- scale(df_num)
S <- cov(X)
S
eig <- eigen(S)

lambda <- eig$values
sum(eig$values[1:2])/sum(eig$values) #explican 75.4% de la varianza, bien

V <- X %*% eig$vectors[,1:2]
V <- data.frame(V)
V

plot(V$X1,V$X2,col=factor(df$incendio),pch=19,main = "PCA",xlab = "Componente
1",ylab="Componente 2")

j <- rbind(
  eig$vectors[,1],
  eig$vectors[,2])

j <- as.data.frame(j)
colnames(j) <- colnames(df_num)
rownames(j) <- c("eig1","eig2")

j

sum(eig$values[1:3])/sum(eig$values) #explican 83.8% de la varianza, bien

V3 <- X %*% eig$vectors[,1:3]
V3 <- data.frame(V)
V3

plot_ly(V3, type='scatter3d', mode='markers', color = as.factor(df$incendio),
          x = V$X1, y = V$X2, z = V$X3,
          marker = list(size=2))
```

```
j3 <- rbind(
  eig$vectors[,1],
  eig$vectors[,2],
  eig$vectors[,3])

j3 <- as.data.frame(j3)
colnames(j3) <- colnames(df_num)
rownames(j3) <- c("PC1","PC2","PC3")

j3

V3
#mahalanobis y MDS
cov1 <- function(x){
  x <- as.matrix(scale(x,scale = F))
  xt <- t(x)
  cov <- 1/(nrow(x)-1) * xt %*% x

  return(cov)
}

mahalanobis_dist <- function(x){
  x <- as.matrix(x)
  S <- cov1(x)
  S_1 <- solve(S)
  matriz <- matrix(NA,ncol=nrow(x),nrow=nrow(x))

  for(i in 1:nrow(x)){
          for(j in i:nrow(x)){
          a <- as.numeric(x[i,])-as.numeric(x[j,])
          matriz[i,j] = sqrt(t(a) %*% S_1 %*% a)
          b <- as.numeric(x[j,])-as.numeric(x[i,])
          matriz[j,i] = sqrt(t(b) %*% S_1 %*% b)
          }
  }
  return(matriz)
}

incendios <- df_num[df$incendio==1,]
incendios

m <- mahalanobis_dist(incendios)
row.names(m) <- row.names(incendios)
```

```
colnames(m) = row.names(incendios)
heatmap(m, Colv = NA, Rowv = NA, symm = T,main = "Matriz de distancia (mahalanobis)
entre incendios",ylab="obs.",xlab="obs.")
#los incendios 76, 77 Y 90 son los que menos se parecen a los demás, de hecho, son
parecidos a otros días donde no hubo incendios

df[65:90,]
hist(df[df$incendio==1,]$Rain)
hist(df[df$incendio==0,]$Rain)
#dichos días registraron volúmenes atípicos de lluvia para los días en que hay incendios
hist(df[df$incendio==1,]$FWI)
hist(df[df$incendio==0,]$FWI)



fit <- cmdscale(m,eig=TRUE, k=2)
fit # view results

x <- fit$points[,1]
y <- fit$points[,2]

df$incendio <- factor(df$incendio)
plot(x, y, xlab="Coordenada 1", ylab="Coordenada 2",
        type="n",pch=19)
text(x, y, labels = colnames(m), cex=.8)

#no incendios
no_incendios <- df_num[df$incendio==0,]
no_incendios

m2 <- mahalanobis_dist(no_incendios)
row.names(m2) <- row.names(no_incendios)
colnames(m2) = row.names(no_incendios)
heatmap(m2, Colv = NA, Rowv = NA, symm = T,ylab="obs.",xlab="obs.")

#los registros 84 y 92 son los más distintos con respecto de otros días de no incendio
df[80:95,]
hist(df[df$incendio==0,]$ISI) #el registro 84 presenta un ISI elevado (outlier), similar al
que se ve en los días de incendio.
hist(df[df$incendio==1,]$ISI)

hist(df$Rain) #el día 92 registró valores atípicos de lluvia.

fit <- cmdscale(m2,eig=TRUE, k=2)
fit
```

```
x <- fit$points[,1]
y <- fit$points[,2]

df$incendio <- factor(df$incendio)
plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2",
        main="Metric MDS", type="n",pch=19,xlim = c(-5,5),ylim=c(-6,6))
text(x, y, labels = colnames(m2), cex=.8)

### todo

m3 <- mahalanobis_dist(df_num)
fit <- cmdscale(m3,eig=TRUE, k=2)
fit

x <- fit$points[,1]
y <- fit$points[,2]
x
y

p <- cbind(x,y,df$incendio)
colnames(p) <- c("x","y","fire")
p <- as.data.frame(p)
p
plot(p$x, p$y, xlab="Coordinate 1", ylab="Coordinate 2",
        main="Metric MDS",pch=19,col=factor(p$fire))

#no parecen diferenciarse demasiado los grupos con 2 dimensiones, veremos qué tal
añadiendo una dimensión más

fit <- cmdscale(m3,eig=TRUE, k=3)
fit

x <- fit$points[,1]
y <- fit$points[,2]
z <- fit$points[,3]
x
y
z
p <- cbind(x,y,z,df$incendio)
colnames(p) <- c("x","y","z","fire")
p <- as.data.frame(p)
p
glimpse(p)
```

```
plot_ly(p, type='scatter3d', mode='markers',
        x = x, y = y, z = z,color=factor(p$fire),size=5)

###
#MDS matriz de correlación (variables)
cor <- cor(df_num)
cor <- 1-cor
cor
fit <- cmdscale(cor, k=2)
fit
x <- fit[,1]
y <- fit[,2]
plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2",
        main="Metric MDS", type="n")
text(x, y, labels = colnames(df_num), cex=.8)

cor <- cor(df_num)
cor <- 1-cor
cor
fit <- cmdscale(cor, k=3)
fit
x <- fit[,1]
y <- fit[,2]
z <- fit[,3]

dd <- cbind(x,y,z)
dd <- as.data.frame(dd)
colnames(dd) <- c("x1","x2","x3")
rownames(dd) <- rownames(fit)
dd
plot_ly(dd, type='scatter3d', mode='markers',
        x = x, y = y, z = z) #hmmmm

#modelos predictivos

#conjunto de entrenamiento y testeo
data <- df %>% dplyr::select(temp,RH,Ws,Rain,FFMC,DMC,DC,ISI,BUI,FWI,incendio)
data <- data[-c(76,77,84),] #removemos aquellos días que son atípicos en su respectiva
categoría
data$incendio <- factor(data$incendio)
set.seed(8);part = createDataPartition(y=data$incendio, p=0.7,list=F)
entreno = data[part,]
testeo = data[-part,]
```

```
#modelo I: árbol
set.seed(8);arbol = rpart(incendio~.,entreno,method="class")
# rpart.plot(arbol)
pred=predict(arbol,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

#modelo II: rf
set.seed(8);RF = randomForest(incendio~.,entreno,ntree=100,mtry=1)
pred=predict(RF,testeo,type='class')
confusionMatrix(pred,testeo$incendio)

#modelo III: red neuronal
set.seed(8);modelo=train(incendio~.,data,maxit=1000,MaxNWts=2000,method="nnet",
                 trControl=trainControl(method="cv",10), #method = "cv", cross-validation,
folds=10
                 tuneGrid=expand.grid(size=c(5,10,20),decay=c(0.001,0.01,0.1))) #tune grid
for hyper-parameters
plotnet(modelo)
modelo
mean(modelo$resample[,"Accuracy"]) #accuracies de los modelos a los que tomo la
media

#modelo IV: svm
svm = svm(incendio~.,entreno,kernel="linear")
svm
pred=predict(svm,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

##svm - ajuste de hiperparametros
set.seed(123);tune_l = tune.svm(incendio~., data=entreno, kernel="linear",
                          cost = c(0.001, 0.01, 0.1, 0.5, 1))
summary(tune_l)
tune_l$best.model

svm <- tune_l$best.model
pred=predict(svm,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

###utilizando únicamente los componentes principales

data <- cbind(V3,df$incendio)
data <- as.data.frame(data)
colnames(data) <- c("X1","X2","X3","incendio")
```

```
data <- data[-c(76,77,84),] #removemos aquellos días que son atípicos en su respectiva categoría
data$incendio <- factor(data$incendio)
set.seed(8);part = createDataPartition(y=data$incendio, p=0.7,list=F)
entreno = data[part,]
testeo = data[-part,]

#modelo I: árbol
set.seed(8);arbol = rpart(incendio~.,entreno,method="class")
# rpart.plot(arbol)
pred=predict(arbol,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

#modelo II: rf
set.seed(8);RF = randomForest(incendio~.,entreno,ntree=100,mtry=1)
pred=predict(RF,testeo,type='class')
confusionMatrix(pred,testeo$incendio)

#modelo III: red neuronal
set.seed(8);modelo=train(incendio~.,data,maxit=1000,MaxNWts=2000,method="nnet",
                 trControl=trainControl(method="cv",10), #method = "cv", cross-validation, folds=10
                 tuneGrid=expand.grid(size=c(5,10,20),decay=c(0.001,0.01,0.1))) #tune grid for hyper-parameters
plotnet(modelo)
modelo
mean(modelo$resample[,"Accuracy"]) #accuracies de los modelos a los que tomo la media

#modelo IV: svm
svm = svm(incendio~.,entreno,kernel="linear")
svm
pred=predict(svm,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

##svm - ajuste de hiperparametros
set.seed(123);tune_l = tune.svm(incendio~., data=entreno, kernel="linear",
                          cost = c(0.001, 0.01, 0.1, 0.5, 1))
summary(tune_l)
tune_l$best.model

svm <- tune_l$best.model
pred=predict(svm,testeo,type="class")
confusionMatrix(pred,testeo$incendio)
```

```
results <- princomp(df_num)

#visualize results of PCA in biplot
biplot(results)


###utilizando sólo factores climáticos e índice FWI

data <- df %>% dplyr::select(temp,RH,Ws,Rain,FWI,incendio)
data <- data[-c(76,77,84),] #removemos aquellos días que son atípicos en su respectiva categoría
data$incendio <- factor(data$incendio)
set.seed(8);part = createDataPartition(y=data$incendio, p=0.7,list=F)
entreno = data[part,]
testeo = data[-part,]

#modelo I: árbol
set.seed(8);arbol = rpart(incendio~.,entreno,method="class")
# rpart.plot(arbol)
pred=predict(arbol,testeo,type="class")
confusionMatrix(pred,testeo$incendio)

#modelo II: rf
set.seed(8);RF = randomForest(incendio~.,entreno,ntree=100,mtry=1)
pred=predict(RF,testeo,type='class')
confusionMatrix(pred,testeo$incendio)

#modelo III: red neuronal
set.seed(8);modelo=train(incendio~.,data,maxit=1000,MaxNWts=2000,method="nnet",
                 trControl=trainControl(method="cv",10), #method = "cv", cross-validation, folds=10
                 tuneGrid=expand.grid(size=c(5,10,20),decay=c(0.001,0.01,0.1))) #tune grid for hyper-parameters
plotnet(modelo)
modelo
mean(modelo$resample[,"Accuracy"]) #accuracies de los modelos a los que tomo la media

#modelo IV: svm
svm = svm(incendio~.,entreno,kernel="linear")
svm
pred=predict(svm,testeo,type="class")
```

```
confusionMatrix(pred,testeo$incendio)

##svm - ajuste de hiperparametros
set.seed(123);tune_l = tune.svm(incendio~., data=entreno, kernel="linear",
                                cost = c(0.001, 0.01, 0.1, 0.5, 1))
summary(tune_l)
tune_l$best.model

svm <- tune_l$best.model
pred=predict(svm,testeo,type="class")
confusionMatrix(pred,testeo$incendio)
```