

# Aprendizaje máquina para la toma de decisiones: Tarea 2

Guillermo Alberto Garcia Candanosa - A01034958

2021-06-27

## Tarea 2: Magic Loop

En esta tarea implementarás un magic loop.

El objetivo del magic loop es que puedas generar diferentes modelos con base en un mismo algoritmo a través de diferentes configuraciones a sus hiper parámetros, así como generar diferentes modelos utilizando diferentes algoritmos, por ejemplo: árboles, random forest, regresión logística, knn, etc.

- Para cada algoritmo que quieres probar:
- Para cada configuración de hiperparámetros a probar en el algoritmo:
- Imprime el algoritmo y la configuración de hiperparámetros a correr

### ¿Qué debes hacer?

Crea un *magic loop* que itere sobre 3 diferentes algoritmos: k-nn, árboles, *random forest*.

Para cada algoritmo define al menos 2 valores diferentes para 2 de sus hiper parámetros.

En lugar de correr los algoritmos imprime el algoritmo que estás “ocupando” y los hiper parámetros asociados.

Puedes implementar el *magic loop* en R o en Python.

Por ejemplo: Si tienes un árbol con los hiper parámetros: profundidad de 5 y 7 y número de elementos para ser hoja como 8 y 13; y un random forest con los hiper parámetros: número de árboles 300 y 500 y profundidad de 6 y 9, lo que el *magic loop* deberá imprimir es:

```
árbol, profundidad: 5, num elementos para hoja: 8
árbol, profundidad: 5, num elementos para hoja: 13
árbol, profundidad: 7, num elementos para hoja: 8
árbol, profundidad: 7, num elementos para hoja: 13
random forest, num de árboles: 300, profundidad: 6
random forest, num de árboles: 300, profundidad: 9
random forest, num de árboles: 500, profundidad: 6
random forest, num de árboles: 500, profundidad: 9
```

### ¿Qué debes entregar?

Un script `.r` o `.Rmd` o `.ipynb` o `py` o excel (con macro) con el código de tu *magic loop*. -> Cambiaré la parte de tu código en donde defines los algoritmos y los hiper parámetros para probar mi propia configuración <-.

¿Cuándo se entrega?

Máximo **domingo 27 de junio del 2021 a las 23:59:59 CST** por correo a [liliana.millan@tec.mx](mailto:liliana.millan@tec.mx) con el subject `tarea_2`.

---

## Realización

### Declaración de algoritmos

```
# Hago un vector con los algoritmos que estaré tratando.
algoritmos <- c("k-nn", "árbol", "random forest")

# Hago una lista de vectores con los hiperparámetros que trataré para knn.
hiper_knn <- list(c("k: 4", "distancia: euclidea"),
                 c("k: 4", "distancia: Manhattan"),
                 c("k: 6", "distancia: euclidea"),
                 c("k: 6", "distancia: Manhattan"))

# Hago una lista de vectores con los hiperparámetros que trataré para árboles.
hiper_arbol <- list(c("profundidad: 5", "num elementos para hoja: 8"),
                  c("profundidad: 5", "num elementos para hoja: 13"),
                  c("profundidad: 7", "num elementos para hoja: 8"),
                  c("profundidad: 7", "num elementos para hoja: 13"))

# Hago una lista de vectores con los hiperparámetros que trataré para random forest.
hiper_rf <- list(c("num de árboles: 300", "profundidad: 6"),
                c("num de árboles: 300", "profundidad: 9"),
                c("num de árboles: 500", "profundidad: 6"),
                c("num de árboles: 500", "profundidad: 9"))

# Hago una lista de listas de vectores con los hiperparámetros que trataré.
hiper_list <- list(hiper_knn, hiper_arbol, hiper_rf)
```

### Magic Loop

```
# Declaro una variable numérica n igual a 0.
n <- as.numeric(0)

# Construyo un for anidado que imprima un string en donde se concatenen cada una
# de las combinaciones de algoritmos con sus respectivos hiperparámetros.
for (a in algoritmos){
  n = n + 1
  for (h in 1:4){
    print(paste(a, " ", hiper_list[[n]][[h]][1], " ", hiper_list[[n]][[h]][2]))
  }
}
```

```
## [1] "k-nn , k: 4 , distancia: euclidean"
## [1] "k-nn , k: 4 , distancia: Manhattan"
## [1] "k-nn , k: 6 , distancia: euclidean"
## [1] "k-nn , k: 6 , distancia: Manhattan"
## [1] "árbol , profundidad: 5 , num elementos para hoja: 8"
## [1] "árbol , profundidad: 5 , num elementos para hoja: 13"
## [1] "árbol , profundidad: 7 , num elementos para hoja: 8"
## [1] "árbol , profundidad: 7 , num elementos para hoja: 13"
## [1] "random forest , num de árboles: 300 , profundidad: 6"
## [1] "random forest , num de árboles: 300 , profundidad: 9"
## [1] "random forest , num de árboles: 500 , profundidad: 6"
## [1] "random forest , num de árboles: 500 , profundidad: 9"
```