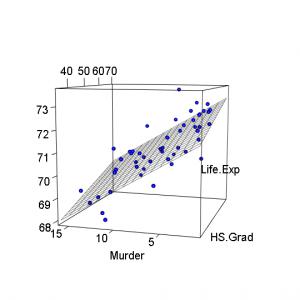
**Logistic** [**Regression**](https://www.r-exercises.com/2017/01/15/multiple-regression-part-1/)

En los siguientes ejercicios vamos a cubrir la regresión logística en R.

Como dice Andrew Ng, las redes neuronales no son más que una pila de múltiples modelos de regresión logística.

Este proyecto proporcionará una guía paso a paso para construir un modelo de regresión logística utilizando R. En este caso, usaremos una 'regresión logística binomial', ya que la variable de decisión solo puede tener dos valores. Sin embargo, también podemos predecir una variable de decisión con más de tres valores. Este tipo de regresión se llama "Regresión logística multinomial". En este proyecto, predecimos la probabilidad de lluvia en el aeropuerto SEATAC en Seattle, USA.

**Exercise 1**

a. Cargar el set de datos RainSeattle2016.csv.

b. Utilice las funciones **str()** y **head()** para explorar el set de datos.

c. Realice una breve descripción de las variables.

**Exercise 2**

Identifique las variables con valores ausentes y valores “NA”.

**Exercise 3**

1. Categorizar dos columnas en el set de datos: Rain y Season. Primero Rain es convertida a factor con 2 categorías, 0 y 1. 0 indicará no lluvia, mientras que 1 indicará lluvia durante el día. Season debe ser convertida a factor: con 4 categorías ('Fall', 'Soring', 'Summer', 'Winter'). Sugerencia: utilice Ordered = FALSE para indicar a R que las categorías no son ordinales.
2. Adicionalmente necesitamos convertir DATE de tipo de dato carácter a tipo de dato date (usamos la función as.Date()).

**Exercise 4**

Divida el set de datos en datos de entrenamiento y datos para testear el modelo. Utilice el 80% de los datos para entrenamiento.

**Exercise 5**

Realizar un análisis exploratorio de los datos (EDA), utilizar la función **featurePlot()** del paquete caret.

Ajustar un modelo de regresión logística binomial para predecir la probabilidad de lluvia en el aeropuerto SEATAC en Seattle, USA. Utilizar las siguientes variables: Season+Ave.Wind+SNOW+TAVG+TMAX+TMIN+WDF5+WSF5.

**Exercise 6**

Generar un modelo usando el algoritmo stepwise (mixto, forward and backward).

**Exercise 7**

Utilizar la función **anova()** y el parámetro test='Chisq' para verificar el efecto global de las variables AveWind y Season en el modelo final.

**Exercise 8**

Generar un modelo final (el mejor modelo) basado en los resultados de los ejercicios 6 y 7. Utilizar la función **anova()** para calcular la significancia de la diferencia de residuos entre el modelo de interés y el modelo nulo (sin predictores -> Ho).

**Exercise 9**

Calcular un intervalo de confianza del 95% para los **odds ratios** (razón de probabilidades) de las dos variables predictoras.

**Exercise 10**

Realizar predicciones con nuestro modelo utilizando los datos de entrenamiento o **train** y de validación o **test**. Si la probabilidad de lluvia es >0.5, asumimos que lloverá ese día y si la probabilidad es <0.5, podemos asumir que no lloverá ese día. Imprima en pantalla las predicciones basadas en este umbral.

**Exercise 11**

Uno de los indicadores de rendimiento más importantes de un modelo es la matriz de confusión. Calcular la matriz de confusión para las predicciones del ejercicio anterior.

**Exercise 12**

Calcular la exactitud de nuestras predicciones.

**Exercise 13**

Graficar la curva ROC y calcular el área bajo la curva como una medida de la predicción de nuestro modelo. Como regla general, un modelo con buena capacidad predictiva debe tener un AUC más cercano a 1 (= 1 idealmente).

**Exercise 14**

Establezca la función del modelo logit y la función para la probabilidad de lluvia. Graficar la curva de probabilidad para el modelo logit.

**Exercise 15 (optional)**

Generar un nuevo modelo de regresión logística con k-fold cross validation usando el paquete caret.

**Exercise 16 (optional)**

Generar un nuevo modelo de regresión logística con k-fold cross validation y el algoritmo Random Forest (“Bosques Aleatorios”) usando el paquete caret.