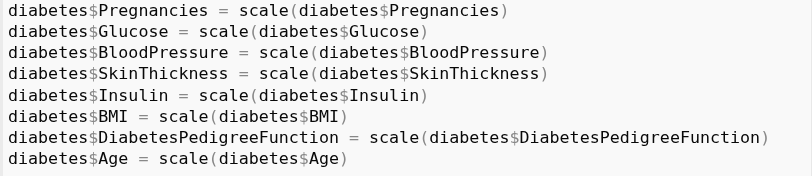
Analizando Dataset “diabetes.xlsx” para identificar la precisión del modelo de red Neuronal

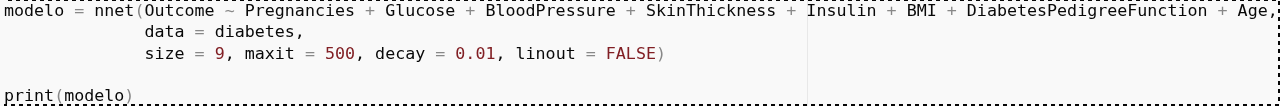
Paso 1.

Escalamos los datos de las variables predictoras para que así tengan una escala similar y nuestro modelo de red neuronal pueda aprender correctamente.

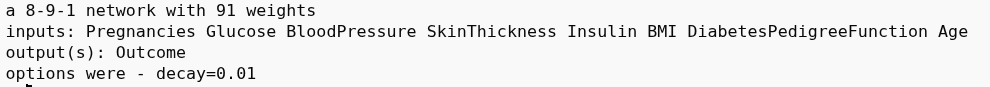


Paso 2.

Creamos y entrenamos nuestro modelo y lo guardamos en la variable “modelo” la variable Outcome es la primera variables ya que es la que queremos predecir las otras 8 son las predictoras, definimos un “size = 9” que serán las neuronas ocultas, agregamos un valor de regulación “decay = 0.01”, también colocamos un “maxin = 500” que significa que el modelo realizara 500 iteraciones para aprender y definimos “linout = FALSE” ya que la salida de la variable “Outcome” es categórica es decir que solo puede ser 0 y 1. y mostramos el modelo con: print(modelo)



Analizando el resultado del print(modelo).



Este resultado: a **8**-**9**-1 network, nos dice lo siguiente.

8 neuronas de entrada que representan las 8 variables.

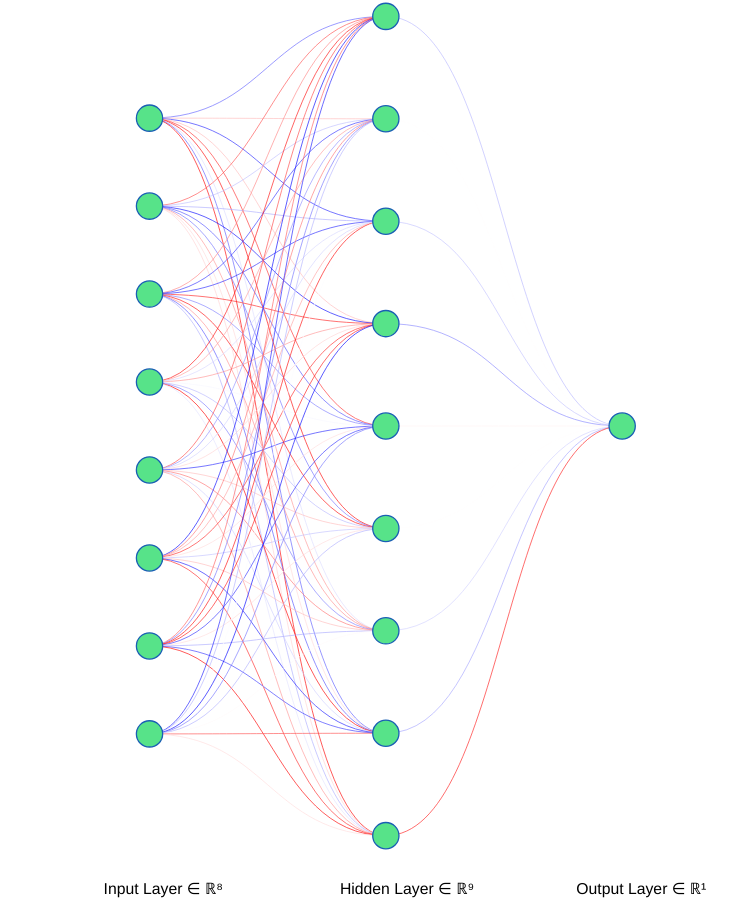
9 neuronas en la capa oculta, que es la que definimos en “size = 9”.

1 neurona de salida que es Outcome.

with 91 weights:

esto significa que en toda la red, hay 91 pesos o conexiones

Representación gráfica de la red neuronal:



Paso 3:

Ahora usamos el modelo ya entrenado para hacer predicciones, esto se logra utilizando la función predict().



Paso 4.

Comparamos lo que predijo el modelo con los valores reales.

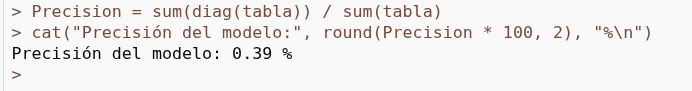


para verificar el porcentaje de precisión de nuestro modelo pasamos al

Paso 5.



en nuestro caso nos dio un resultado de:



nos dio una precisión del 0.39% aunque cabe recalcar que el modelo cambia su precisión cada vez que se ejecuta debido a la aleatoriedad de los pesos en la red neuronal.