Redes Neuronales

Edgar Jiménez

27 de febrero de 2016

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

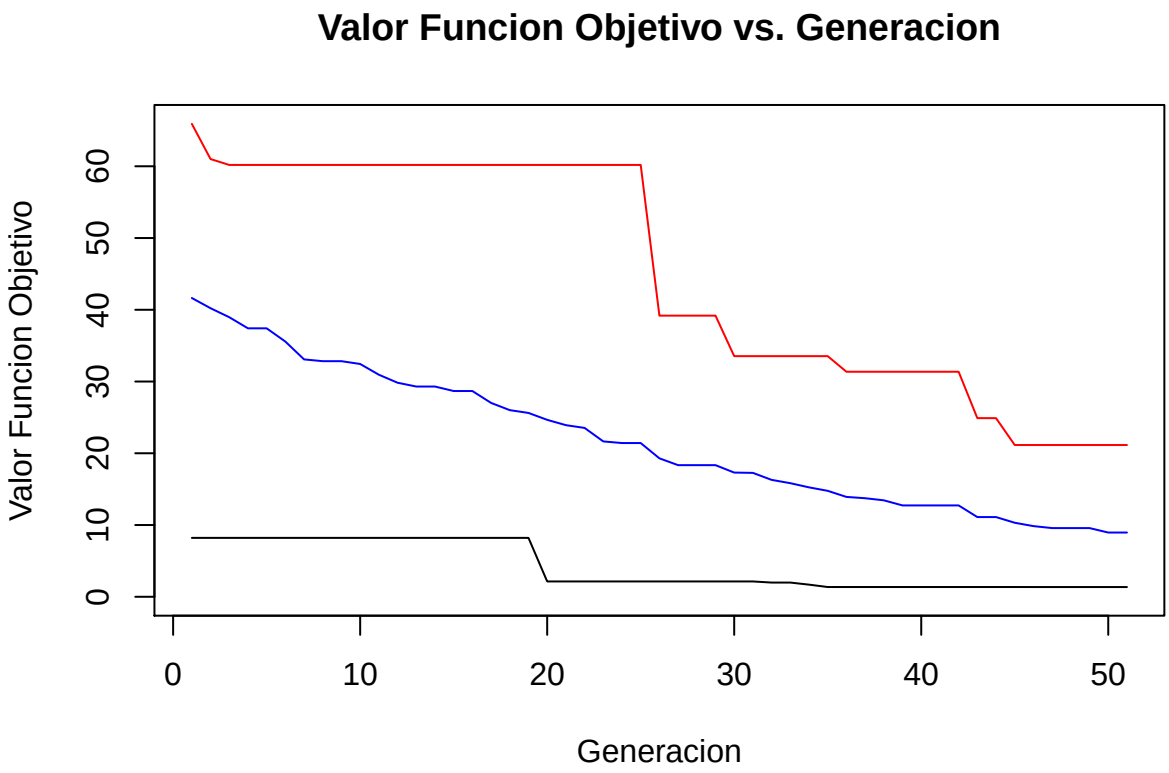
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
diff.evol <- function(fn, N, LB, UB, mu, ngen, Cr = 0.1, F = 0.8){

  Xl = t(matrix(rep(LB,mu),N,mu))

  Xu = t(matrix(rep(UB,mu),N,mu))

  D = Xu - Xl

  Xp = Xl + matrix(runif(mu * N), mu, N) * D

  fp = apply(Xp, 1, fn)

  #Auxiliares de almacenamiento

  Mejorfp = matrix(0, 1, ngen)

  Peorfp = matrix(0, 1, ngen)

  Promediofp = matrix(0, 1, ngen)

  gen = 0

  while(gen <= ngen){

    for (i in 1:N){

      r1 = 1 + floor(mu * runif(mu))

      r2 = 1 + floor(mu * runif(mu))

      r3 = 1 + floor(mu * runif(mu))

      X0 = Xp[r1,] + F * (Xp[r2,] - Xp[r3,])

      f0 = apply(X0,1,fn)

    }

    #Calcuylo aleatorio

    rand = runif(mu)
```

```
    RXp = Xp

    Rfp = fp

    #Se muta a los individuos donde el RAND sea menor que Cr

    k = which(rand < Cr)

    RXp[k,] = X0[k,]

    Rfp[k] = f0[k]

    #Se seleccionan los mejores individuos

    k = which(Rfp < fp)

    Xp[k,] = RXp[k,]

    fp[k] = Rfp[k]

    #Avanzo a la siguiente generación

    gen = gen + 1

    #Calculo el mejor de la función Objetivo

    Mejorfp[gen] = fp[which.min(fp)]

    #Calculo el peor de la función Objetivo

    Peorfp[gen] = fp[which.max(fp)]

    #Calculo el promedio de la función Objetivo

    Promediofp[gen] = mean(fp)

  }


  #Grafico

  plot(Peorfp, ylim = c(0,Peorfp[1]), type="l", col='red', xlab = "Generacion", ylab = "Valor Funcion O

  lines(Mejorfp, type="l", col='black', lwd = 1)

  lines(Promediofp, type="l", col='blue', lwd = 1)

  list(x.opt = Xp[which.min(fp), ], f.opt = fp[which.min(fp)])
}
```

You can also embed plots, for example:

```r
f <- function(x) sum((x-1/3)^2)

diff.evol(fn = f, N = 5, LB = rep(-5,5), UB = rep(5,5), mu = 30, ngen = 50, F = 0.2)
```




```
## $x.opt
## [1] 0.02214589 0.31483412 0.08537890 0.09827683 1.40106727
##
## $f.opt
## [1] 1.353969
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.