# ST 762 HW-1
# Solution by Hao Hu

rm(list=ls())
setwd("/stat.ad.ncsu.edu/Redirect/hhu5/Desktop/ST 762 TA/hw1")

data <- read.table("pasture.txt")
x <- data[,1]; y <- data[,2]

weibull1 <- function(x,b1,b2,b3,b4){
  y <- b1 - b2*exp(-exp(b3 + b4*log(x)))
  y
}
data

time  yield
1   9   8.93
2  14  10.80
3  21  18.59
4  28  22.33
5  42  39.35
6  57  56.11
7  63  61.73
8  70  64.62
9  79  67.08

# Plot lines to guess initial
par(mfrow=c(1,1))
plot(data,xlim=c(0,100),ylim=c(0,100))
xgrid <- seq(from = 0.01, to = 100, by= 0.01)
  # From extreme x we can guess b1 and b2
  # A reasonable guess would be b1 - b2 = 5, b1 = 70
  # Let's try to find b4 first?
lines(xgrid,weibull1(xgrid,70,65,-10,2),col=2)
  # red line: seems to low
lines(xgrid,weibull1(xgrid,70,65,-10,3),col=3)
  # Green line, to high
lines(xgrid,weibull1(xgrid,70,65,-10,2.5),col=4)
  # Blue lines seems better, so b4 close to 2.5
  # Now let's seek b3
lines(xgrid,weibull1(xgrid,70,65,-9,2.5),col=5)
  # Sky blue one seems to be higher
  # So maybe b3 between -9 to -10?
lines(xgrid,weibull1(xgrid,70,65,-9.5,2.5),col=6)
  # Purple one looks great
legend("topleft",c("Real","b3=-10,b4=2","b3=-10,b4=3",
  "b3=-10,b4=2.5","b3=-9,b4=2.5","b3=-9.5,b4=2.5"),
  col = c(1, 2, 3, 4, 5, 6), lwd = 1, merge = TRUE,
  lty = c(NA, 1, 1, 1, 1, 1), pch = c(1, NA, NA, NA, NA, NA))

title("Guess initial values")
```r
# Set up data and parameters
thedat <- data.frame(x,y)
maxiter <- 20
bstart <- list(b1=70,b2=65,b3=-9.5,b4=2.5)
olsfit <- nls(y ~ weibull1(x,b1,b2,b3,b4),thedat,bstart)
summary(olsfit)

# Extract coefficients
beta <- coef(olsfit)

# Plot
par(mfrow=c(1,1))
```

The script sets up a dataset `thedat` with variables `x` and `y`, defines the maximum number of iterations `maxiter` and starting parameter values `bstart`. It then fits a nonlinear least squares model `olsfit` using the `nls` function with the `weibull1` function as the model formula. The `summary` function is used to print the model summary, including the estimated parameters and their standard errors. The coefficients are then extracted, and the plot is set up with `par(mfrow=c(1,1))` to display a single graph.
```r
plot(data, xlim=c(0,100), ylim=c(0,100))
y.pred <- weibull1(xgrid, beta[1], beta[2], beta[3], beta[4])
lines(xgrid, y.pred, col=3)
title("Pasture Data")
legend("topleft", c("Real", "predict"), col=c(1,3),
      lwd = 1, lty = c(NA, 1), pch = c(1, NA), merge = TRUE)
```