Nonlinear Regression Models

Contents

1 Non-linear Regression: nls()

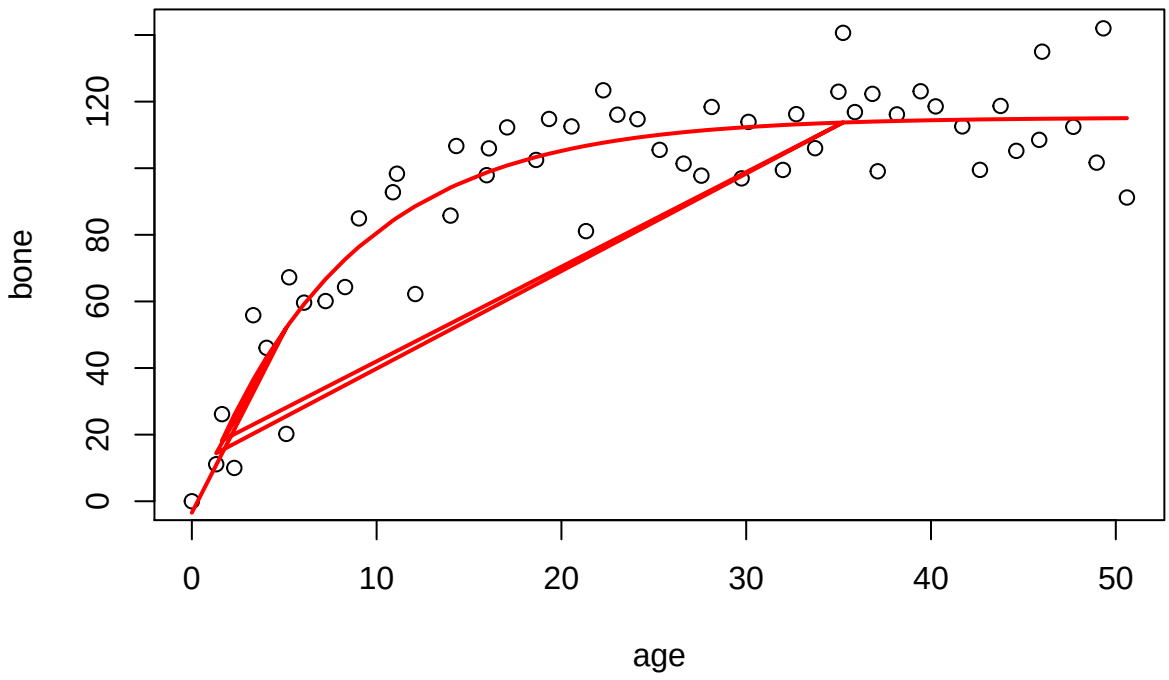
```
deer <- read.table("jaws.txt", header = TRUE)
head(deer)
```

```
##         age      bone
## 1  0.000000   0.00000
## 2  5.112000  20.22000
## 3  1.320000  11.11130
## 4 35.240000 140.65000
## 5  1.632931  26.15218
## 6  2.297635  10.00100
```

```
# first model
model1 <- nls(bone~a-b*exp(-c*age),start=list(a=120,b=110,c=0.064), data = deer)
summary(model1)
```

```
##
## Formula: bone ~ a - b * exp(-c * age)
##
## Parameters:
##   Estimate Std. Error t value Pr(>|t|)
## a 115.2528     2.9139   39.55  < 2e-16 ***
## b 118.6875     7.8925   15.04  < 2e-16 ***
## c   0.1235     0.0171    7.22 2.44e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.21 on 51 degrees of freedom
##
## Number of iterations to convergence: 5
## Achieved convergence tolerance: 2.383e-06
```

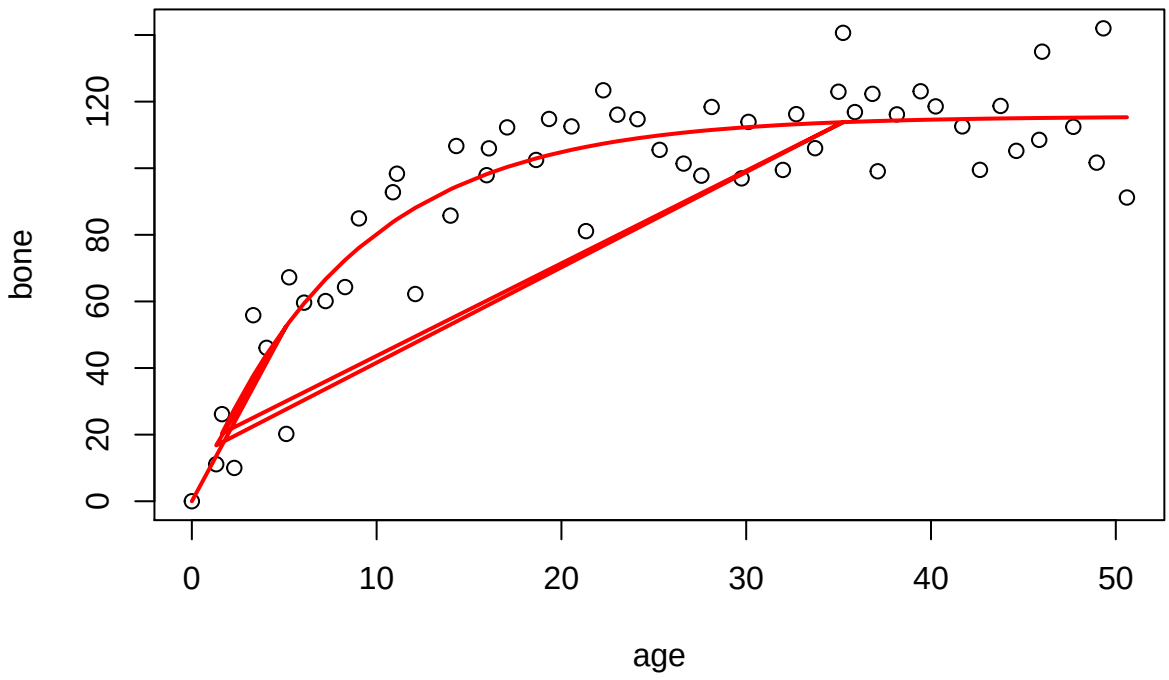
```
plot(bone ~ age, data = deer)
lines(deer$age, fitted(model1), col = "red", lwd = 2)
```




```
# second model
model2 <- nls(bone~a*(1-exp(-c*age)),start=list(a=120,c=0.064), data = deer)
summary(model2)
```

```
##
## Formula: bone ~ a * (1 - exp(-c * age))
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## a 115.58056    2.84365  40.645  < 2e-16 ***
## c   0.11882    0.01233   9.635 3.69e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.1 on 52 degrees of freedom
##
## Number of iterations to convergence: 5
## Achieved convergence tolerance: 1.369e-06
```

```
plot(bone ~ age, data = deer)
lines(deer$age, fitted(model2), col = "red", lwd = 2)
```




```
# comparing two models
anova(model1, model2)
```

```
## Analysis of Variance Table
##
## Model 1: bone ~ a - b * exp(-c * age)
## Model 2: bone ~ a * (1 - exp(-c * age))
##   Res.Df Res.Sum Sq Df  Sum Sq F value Pr(>F)
## 1     51     8897.3
## 2     52     8929.1 -1 -31.843  0.1825  0.671
```

2 Generalized Additive Model

Data Source: House Prices https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

```
# importing the dataset
train <- read.csv("train_cleaned.csv", header = TRUE)

library(splines)
library(gam)
fit.gam <- gam( log(SalePrice) ~
                  s(MSSubClass , df = 5) + s(LotArea , df = 5) +
                  s(OverallQual , df = 5) + s(OverallCond , df = 5) +
                  s(YearBuilt , df = 5) + s(YearRemodAdd , df = 3) +
                  s(MasVnrArea , df = 5) + s(BsmtFinSF1 , df = 5) +
```

```
                  s(BsmtFinSF2 , df = 5) + s(BsmtUnfSF , df = 5) +
                  s(X1stFlrSF , df = 5) + s(X2ndFlrSF , df = 5) +
                  s(BsmtFullBath , df = 1) + s(FullBath , df = 1) +
                  s(BedroomAbvGr , df = 5) + s(KitchenAbvGr , df = 4) +
                  s(TotRmsAbvGrd , df = 1) + s(Fireplaces , df = 2) +
                  s(GarageCars , df = 2) + s(GarageArea , df = 2) +
                  s(WoodDeckSF , df = 2) + s(ScreenPorch , df = 4) +
                  s(PoolArea , df = 5) + s(MoSold , df = 1), data = train
)

summary(fit.gam)
```

```
##
## Call: gam(formula = log(SalePrice) ~ s(MSSubClass, df = 5) + s(LotArea,
##     df = 5) + s(OverallQual, df = 5) + s(OverallCond, df = 5) +
##     s(YearBuilt, df = 5) + s(YearRemodAdd, df = 3) + s(MasVnrArea,
##     df = 5) + s(BsmtFinSF1, df = 5) + s(BsmtFinSF2, df = 5) +
##     s(BsmtUnfSF, df = 5) + s(X1stFlrSF, df = 5) + s(X2ndFlrSF,
##     df = 5) + s(BsmtFullBath, df = 1) + s(FullBath, df = 1) +
##     s(BedroomAbvGr, df = 5) + s(KitchenAbvGr, df = 4) + s(TotRmsAbvGrd,
##     df = 1) + s(Fireplaces, df = 2) + s(GarageCars, df = 2) +
##     s(GarageArea, df = 2) + s(WoodDeckSF, df = 2) + s(ScreenPorch,
##     df = 4) + s(PoolArea, df = 5) + s(MoSold, df = 1), data = train)
## Deviance Residuals:
##       Min        1Q    Median        3Q       Max
## -1.052457 -0.049861  0.005207  0.065379  0.449203
##
## (Dispersion Parameter for gaussian family taken to be 0.0148)
##
##     Null Deviance: 232.8007 on 1459 degrees of freedom
## Residual Deviance: 20.327 on 1372 degrees of freedom
## AIC: -1919.091
##
## Number of Local Scoring Iterations: 5
##
## Anova for Parametric Effects
##                            Df  Sum Sq Mean Sq   F value    Pr(>F)
## s(MSSubClass, df = 5)       1   0.111   0.111    7.4807  0.006317 **
## s(LotArea, df = 5)          1  17.960  17.960 1212.2022 < 2.2e-16 ***
## s(OverallQual, df = 5)      1 133.630 133.630 9019.5669 < 2.2e-16 ***
## s(OverallCond, df = 5)      1   0.460   0.460   31.0371 3.043e-08 ***
## s(YearBuilt, df = 5)        1   8.424   8.424  568.5784 < 2.2e-16 ***
## s(YearRemodAdd, df = 3)     1   0.433   0.433   29.2182 7.618e-08 ***
## s(MasVnrArea, df = 5)       1   2.391   2.391  161.4031 < 2.2e-16 ***
## s(BsmtFinSF1, df = 5)       1   4.449   4.449  300.2847 < 2.2e-16 ***
## s(BsmtFinSF2, df = 5)       1   0.432   0.432   29.1291 7.969e-08 ***
## s(BsmtUnfSF, df = 5)        1   2.727   2.727  184.0290 < 2.2e-16 ***
## s(X1stFlrSF, df = 5)        1   1.831   1.831  123.5587 < 2.2e-16 ***
## s(X2ndFlrSF, df = 5)        1  15.685  15.685 1058.6925 < 2.2e-16 ***
## s(BsmtFullBath, df = 1)     1   0.130   0.130    8.8000  0.003064 **
## s(FullBath, df = 1)         1   0.017   0.017    1.1700  0.279584
## s(BedroomAbvGr, df = 5)     1   0.079   0.079    5.3254  0.021165 *
## s(KitchenAbvGr, df = 4)     1   0.575   0.575   38.7808 6.291e-10 ***
## s(TotRmsAbvGrd, df = 1)     1   0.086   0.086    5.8167  0.016006 *
```

```
## s(Fireplaces, df = 2)      1   0.805   0.805   54.3240 2.927e-13 ***
## s(GarageCars, df = 2)      1   1.138   1.138   76.7920 < 2.2e-16 ***
## s(GarageArea, df = 2)      1   0.052   0.052    3.5237  0.060708 .
## s(WoodDeckSF, df = 2)      1   0.056   0.056    3.7464  0.053125 .
## s(ScreenPorch, df = 4)     1   0.281   0.281   18.9666 1.429e-05 ***
## s(PoolArea, df = 5)        1   0.016   0.016    1.1124  0.291754
## s(MoSold, df = 1)          1   0.003   0.003    0.2186  0.640148
## Residuals               1372  20.327   0.015
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                          Npar Df   Npar F     Pr(F)
## (Intercept)
## s(MSSubClass, df = 5)         4    3.955 0.0033887 **
## s(LotArea, df = 5)            4   12.727 3.570e-10 ***
## s(OverallQual, df = 5)        4    4.799 0.0007591 ***
## s(OverallCond, df = 5)        4    3.005 0.0175310 *
## s(YearBuilt, df = 5)          4   17.347 6.817e-14 ***
## s(YearRemodAdd, df = 3)       2    2.178 0.1137076
## s(MasVnrArea, df = 5)         4    1.589 0.1747111
## s(BsmtFinSF1, df = 5)         4   27.922 < 2.2e-16 ***
## s(BsmtFinSF2, df = 5)         4    1.573 0.1791396
## s(BsmtUnfSF, df = 5)          4    1.329 0.2569960
## s(X1stFlrSF, df = 5)          4   43.357 < 2.2e-16 ***
## s(X2ndFlrSF, df = 5)          4    2.630 0.0329743 *
## s(BsmtFullBath, df = 1)       0    7.128 1.240e-07 ***
## s(FullBath, df = 1)           0    0.055 9.328e-08 ***
## s(BedroomAbvGr, df = 5)       4    2.779 0.0256844 *
## s(KitchenAbvGr, df = 4)       2    1.755 0.1733034
## s(TotRmsAbvGrd, df = 1)       0    0.705 1.395e-06 ***
## s(Fireplaces, df = 2)         1    0.687 0.4071841
## s(GarageCars, df = 2)         1    2.410 0.1207652
## s(GarageArea, df = 2)         1    1.386 0.2393656
## s(WoodDeckSF, df = 2)         1    1.503 0.2204145
## s(ScreenPorch, df = 4)        3    1.689 0.1674292
## s(PoolArea, df = 5)           4    2.579 0.0358894 *
## s(MoSold, df = 1)             0 -204.076
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3 References

https://data-flair.training/blogs/r-nonlinear-regression/

MH Kutner, CJ Nachtsheim, J Neter, W Li (2005), Applied linear statistical models.

Gareth James, Daniela Witten, Trevor Hastie Robert Tibshirani (2013), An Introduction to Statistical Learning with Applications in R.