UNIVERSIDADE FEDERAL DE MINAS GERAIS

DISCIPLINA: Redes Neurais Artificial
PROFESSORES: Frederico Gualberto Ferreira Coelho
ALUNO: Antonio Carlos da Anunciação – 2018019443

TRABALHO PRATICO 1 – Regressão Polinomial:

Obter aproximações polinomiais utilizando 10 amostras para a função geradora $\boldsymbol{f_g(x) = 0.5x^2 + 3x + 10}$ com $\boldsymbol{x \in [-15, 10]}$ somadas com um ruído gaussiano $\boldsymbol{N(0, 4)}$, o grau do polinônimo variando entre $\boldsymbol{p = 1}$ a $\boldsymbol{p = 8}$. Para cada aproximação, mostre um gráfico com a função geradora, as amostras e o polinômio obtido.

Polinomios para Números de Amostas igual a 10:

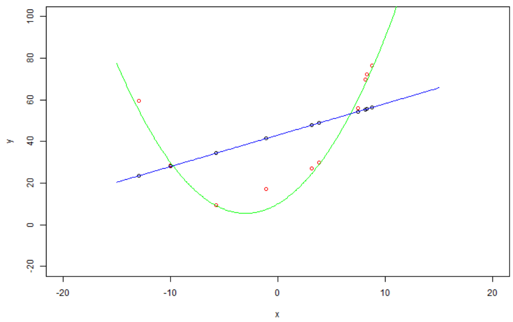

FIG.01: Polinomio de Grau 1

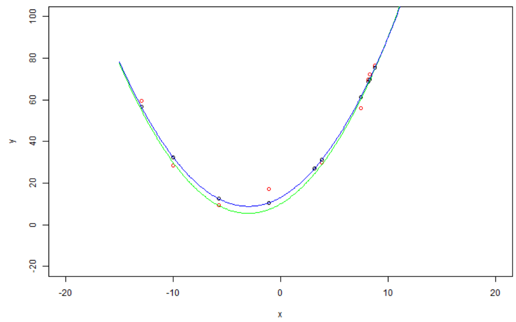

FIG.02: Polinomio de Grau 2

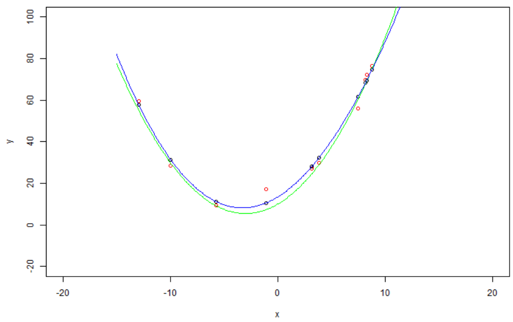

FIG.03: Polinomio de Grau 3

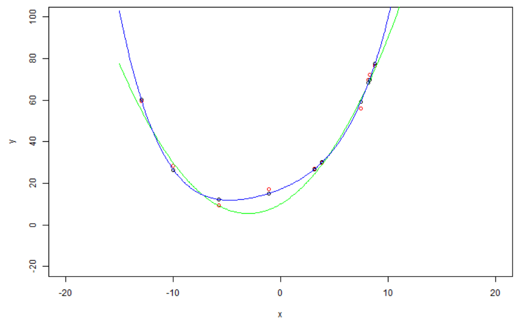

FIG.04: Polinomio de Grau 4

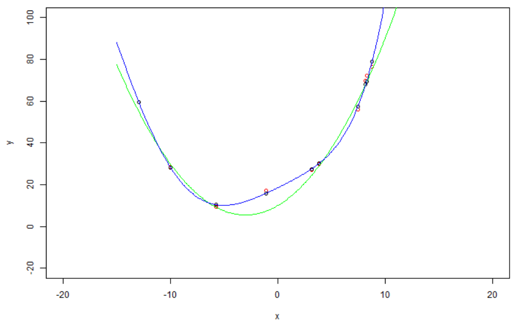

FIG.05: Polinomio de Grau 5

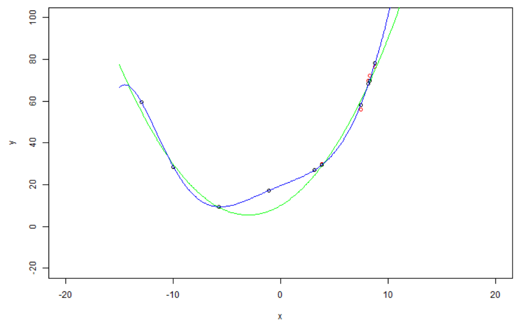

FIG.06: Polinomio de Grau 6

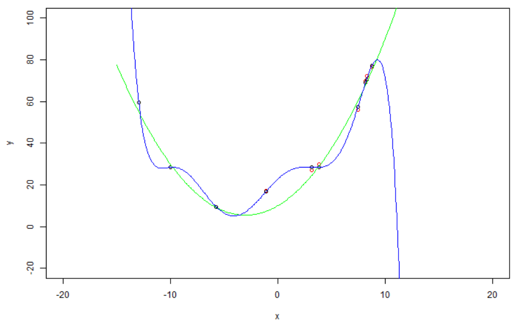

FIG.07: Polinomio de Grau 7

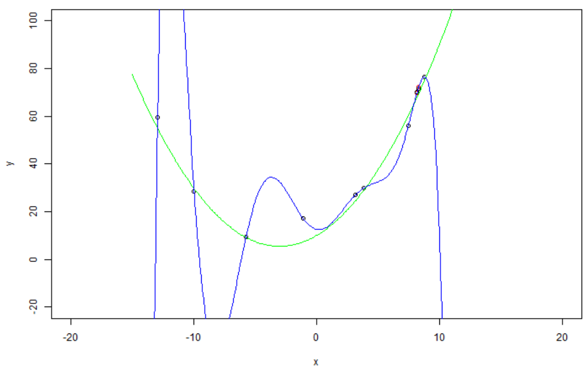

FIG.08: Polinomio de Grau 8

Baseado nos graficos podemos inferir que ocorreram ambos, Overfitting e Underfitting para nossas aproximações para amostras de tamanho 10. A **FIG.01** é um exemplo de **underfitting**, ou seja, nossa aproximação simplificou demais os dados da nossa curva geradora, não se adaptando bem e se tornando uma simples regressão linear dos dados, que é claramente ruim dado o intervalo de da analise. A **FIG.07** e **FIG.08** podemos considerar que ocorreu o **overfitting**, uma vez que existe um sobre-ajuste da curva, sendo forçada a passar em uma quantidade de pontos que em muitos casos são ruidos e não refletem a natureza real do problema, nesses tipos de ajuste a aproximação perde sua capacidade de generalização .

Repetindo o processo agora para um tamanho de amostra igual a 100.

Polinomios para Números de Amostas igual a 100:

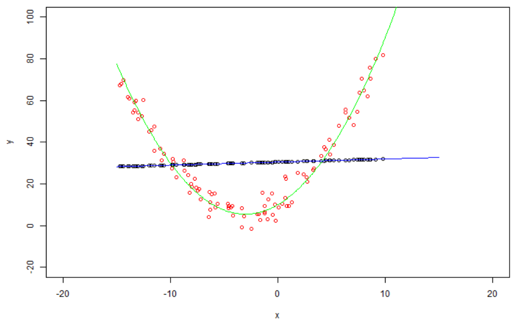

FIG.09: Polinomio de Grau 1

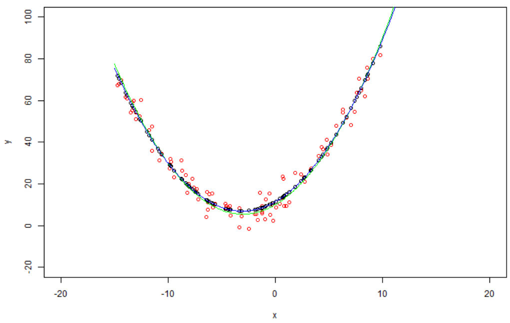

FIG.10: Polinomio de Grau 2

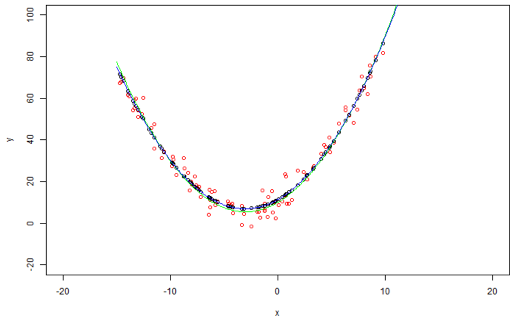

FIG.11: Polinomio de Grau 3

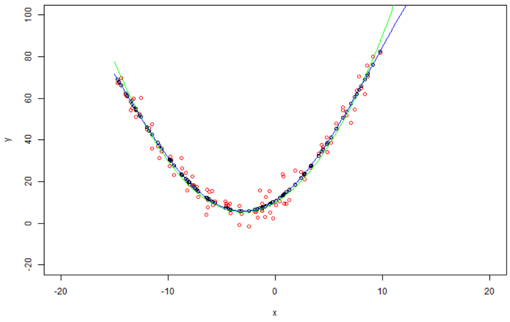

FIG.12: Polinomio de Grau 4

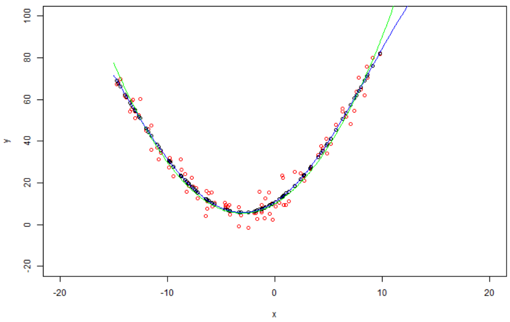

FIG.13: Polinomio de Grau 5

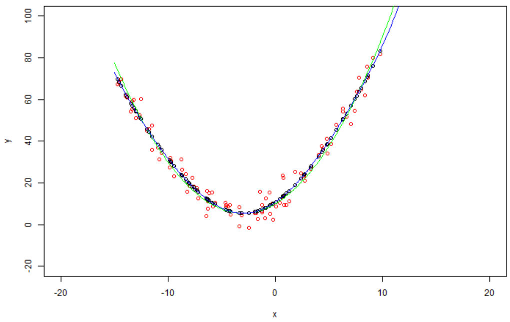

FIG.14: Polinomio de Grau 6

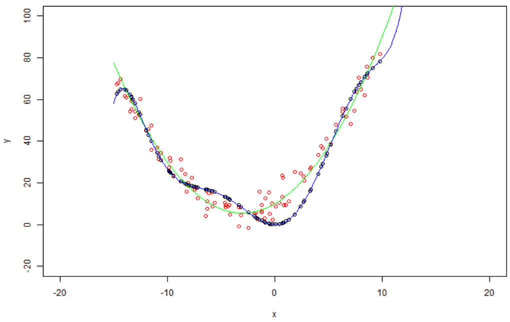

FIG.15: Polinomio de Grau 7

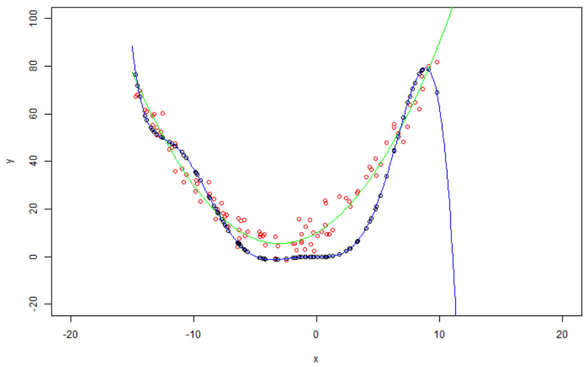

FIG.16: Polinomio de Grau 8

Aumentando consideravelmente a quantidade de pontos da amostra ainda podemos observar o fenomeno de Underfitting na **FIG.09** apartir dai podemos inferir que o Underfitting é influenciado pela tentativa de representar dados não-linear em forma linear, por exemplo tentar aproximar uma parabola por uma reta. Para o caso do **overfitting** um aumento na quantidade de pontos disponiveis para calcular nossa aroximação suavizou as oscilações da função aproximadora, mas como podemos ver nas **FIG.15** e **FIG.16** isto não reduziu o efeito de perda de capacidade de generalização dessas ultimas funções, sendo essas consideradas aproximações ruins.

Código Fonte

```
rm(list=ls())
library('corpcor')

# Geração dos dados sinteticos:
N <- 10 # Tamanho da Amostra
n <- 8 # Maior grau da regressão polinomial

x <- runif(n = N, min = -15, max = 10)
yr <- (0.5*x^2+3*x+10) + rnorm(n = length(x), mean = 0, sd = 4)

xgrid <- seq(-15, 15, 0.1)
ygrid <- 0.5*xgrid^2+3*xgrid+10

H <- c(1)
Hgrid <- c(1)

for(i in 1:n) {
  H <- cbind(x^i, H)
  w <- pseudoinverse(H) %*% yr
  yhat <- H %*% w
  Hgrid <- cbind(xgrid^i, Hgrid)
  yhatgrid <- Hgrid %*% w

  nome <- paste(as.character(i))
  img <- paste("regPol_p",nome,"_N",N,'.png',sep = "",collapse = "")
  png(file = img, width=720, height=500)

  plot(x, yr, col='red', xlim=c(-20,20), ylim=c(-20,100), xlab = 'x', ylab = 'y')
  par(new = T)
  plot(x, yhat, col='black', xlim=c(-20,20), ylim=c(-20,100), xlab = '', ylab = '', axes = F)
  par(new = T)
  plot(xgrid, ygrid, col='green', type = 'l', xlim=c(-20,20), ylim=c(-20,100), xlab = '', ylab = '', axes = F)
  par(new = T)
  plot(xgrid, yhatgrid, col='blue', type = 'l', xlim=c(-20,20), ylim=c(-20,100), xlab = '', ylab = '', axes = F)

  dev.off()
}
```