Neural Network Notes

library(datasets)  
library(tidyverse)

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --  
## v dplyr 1.1.2 v readr 2.1.4  
## v forcats 1.0.0 v stringr 1.5.0  
## v ggplot2 3.4.2 v tibble 3.2.1  
## v lubridate 1.9.2 v tidyr 1.3.0  
## v purrr 1.0.1   
## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()  
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(caret)

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

library(grid)  
library(gridExtra)

##   
## Attaching package: 'gridExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine

library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(rattle)

## Loading required package: bitops  
## Rattle: A free graphical interface for data science with R.  
## Version 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

library(MASS)

##   
## Attaching package: 'MASS'  
##   
## The following object is masked from 'package:dplyr':  
##   
## select

library(caTools)  
library(nnet)  
library(NeuralNetTools)  
library(neuralnet)

##   
## Attaching package: 'neuralnet'  
##   
## The following object is masked from 'package:dplyr':  
##   
## compute

## Neural Network (feedfoward)

set.seed(3920)  
irisam=sample.split(iris$Species,SplitRatio = 0.7)  
iris.train=subset(iris,irisam==TRUE)  
iris.test=subset(iris,irisam==FALSE)

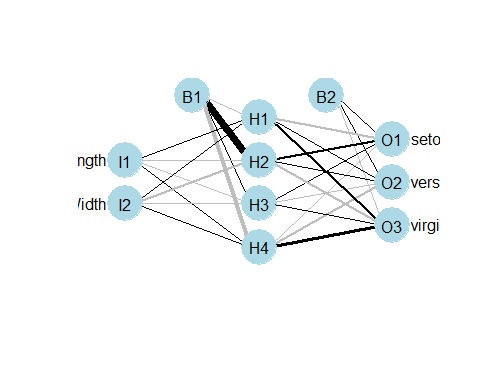
irisnn = nnet(Species~Petal.Length + Petal.Width, iris.train, size=4)

## # weights: 27  
## initial value 123.861467   
## iter 10 value 37.803479  
## iter 20 value 9.034768  
## iter 30 value 8.814433  
## iter 40 value 8.557308  
## iter 50 value 8.406768  
## iter 60 value 8.214940  
## iter 70 value 8.167614  
## iter 80 value 7.949111  
## iter 90 value 6.935585  
## iter 100 value 6.138059  
## final value 6.138059   
## stopped after 100 iterations

summary(irisnn)

## a 2-4-3 network with 27 weights  
## options were - softmax modelling   
## b->h1 i1->h1 i2->h1   
## -19.51 7.39 11.73   
## b->h2 i1->h2 i2->h2   
## 202.10 -23.83 -58.42   
## b->h3 i1->h3 i2->h3   
## 5.66 -5.56 -5.74   
## b->h4 i1->h4 i2->h4   
## -113.89 18.24 8.02   
## b->o1 h1->o1 h2->o1 h3->o1 h4->o1   
## 21.29 -53.72 50.26 7.23 -12.47   
## b->o2 h1->o2 h2->o2 h3->o2 h4->o2   
## 12.69 2.03 19.32 -17.00 -64.33   
## b->o3 h1->o3 h2->o3 h3->o3 h4->o3   
## -34.82 51.24 -68.29 10.78 76.93

plotnet(irisnn)

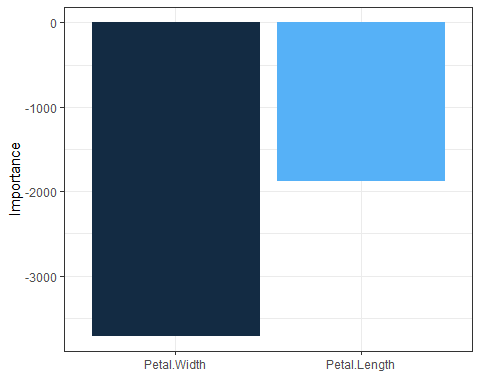


predictions = predict(irisnn,iris.test,type="class")  
table(iris.test$Species,predictions)

## predictions  
## setosa versicolor virginica  
## setosa 15 0 0  
## versicolor 0 14 1  
## virginica 0 0 15

olden(irisnn)

## Warning in olden.default(wts\_in, x\_names, y\_names, skip\_wts = skip\_wts, :  
## Results for first response variable only, use out\_var argument to change



model1 = neuralnet(  
 Species~Petal.Length+Petal.Width,  
data=iris.train,  
hidden=c(4,1),  
linear.output = FALSE  
)  
plot(model1,rep="best")

