

ANN

Esther

2024-07-31

R Markdown

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:

```
#load the necessary packages
install.packages(c("neuralnet","keras","tensorflow"), dependencies = T)
```

```
## Installing packages into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
```

```
library(neuralnet)
install.packages("tidyverse")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::compute() masks neuralnet::compute()
```

```
## 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
```

```
#data preparation
```

```
iris<-iris %>%mutate_if(is.character, as.factor)
```

```
#Sample of the dataset
```

```
set.seed(123)
```

```
iris_sample <- iris %>% sample_n(10)
```

```
summary(iris_sample)
```

```
##   Sepal.Length   Sepal.Width   Petal.Length   Petal.Width   Species
##   Min.    :4.300   Min.    :2.500   Min.    :1.10   Min.    :0.100   setosa    :3
##   1st Qu.:5.125   1st Qu.:2.775   1st Qu.:2.05   1st Qu.:0.450   versicolor:3
##   Median :5.650   Median :3.000   Median :4.50   Median :1.350   virginica :4
```

```
## Mean      :5.670   Mean      :3.010   Mean      :3.89   Mean      :1.230
## 3rd Qu.:6.050   3rd Qu.:3.150   3rd Qu.:5.10   3rd Qu.:1.875
## Max.      :7.700   Max.      :3.800   Max.      :6.70   Max.      :2.200
```

```
# Train and test split
```

```
set.seed(254)
```

```
data_rows<-floor(0.80 * nrow(iris_sample))
```

```
data_rows
```

```
## [1] 8
```

```
train_indices<-sample(c(1:nrow(iris_sample)), data_rows)
```

```
train_indices
```

```
## [1] 7 4 8 5 9 1 10 2
```

```
train_data<-iris_sample[train_indices, ]
```

```
train_data
```

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width  Species
## 7           5.5         2.5         4.0         1.3 versicolor
## 4           4.4         3.2         1.3         0.2   setosa
## 8           5.5         2.6         4.4         1.2 versicolor
## 5           5.9         3.0         5.1         1.8  virginica
## 9           5.8         2.7         5.1         1.9  virginica
## 1           4.3         3.0         1.1         0.1   setosa
## 10          6.1         3.0         4.6         1.4 versicolor
## 2           5.0         3.3         1.4         0.2   setosa
```

```
test_data<-iris_sample[-train_indices,]
```

```
test_data
```

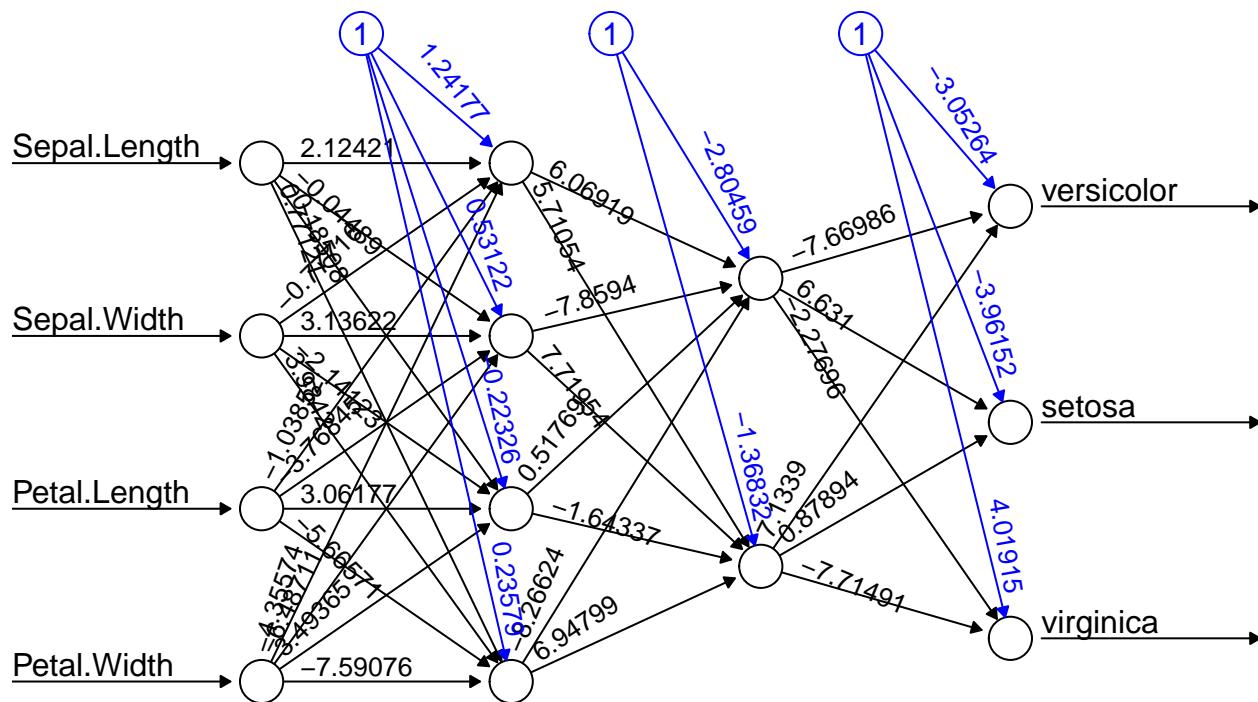
```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width  Species
## 3           7.7         3.8         6.7         2.2 virginica
## 6           6.5         3.0         5.2         2.0 virginica
```

```
#model training
```

```
model<-neuralnet( Species ~ Sepal.Length +Sepal.Width+Petal.Length +Petal.Width, data = train_data, hidden = 10, test.data = test_data)
```

```
#plot the model
```

```
plot(model, rep = 'best')
```



Error: 0.018628 Steps: 76

```
#predicting with the model
```

```
pred<-predict(model, test_data)
pred
```

```
##           [,1]      [,2]      [,3]
## 3 0.02214252 0.06515439 0.9466858
## 6 0.01714247 0.10296964 0.9181845
```

```
labels<-c("setosa", "versicolor", "virginica")
labels
```

```
## [1] "setosa"      "versicolor" "virginica"
```

```
#mapping prediction to labels
```

```
prediction_label<-data.frame(max.col(pred))>%mutate(pred=labels[max.col(pred)])>% select(2) >%
unlist()
table(test_data$Species, prediction_label)
```

```
##           prediction_label
##           virginica
## setosa              0
## versicolor          0
## virginica           2
```

```
summary(test_data)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## Min.   :6.5   Min.   :3.0   Min.   :5.200 Min.   :2.00  setosa   :0
## 1st Qu.:6.8   1st Qu.:3.2   1st Qu.:5.575 1st Qu.:2.05  versicolor:0
## Median :7.1   Median :3.4   Median :5.950 Median :2.10  virginica :2
## Mean   :7.1   Mean   :3.4   Mean   :5.950 Mean   :2.10
## 3rd Qu.:7.4   3rd Qu.:3.6   3rd Qu.:6.325 3rd Qu.:2.15
```

```
## Max. :7.7 Max. :3.8 Max. :6.700 Max. :2.20
```

```
#accuracy of the model
```

```
check= as.numeric(test_data$Species) == max.col(pred)
check
```

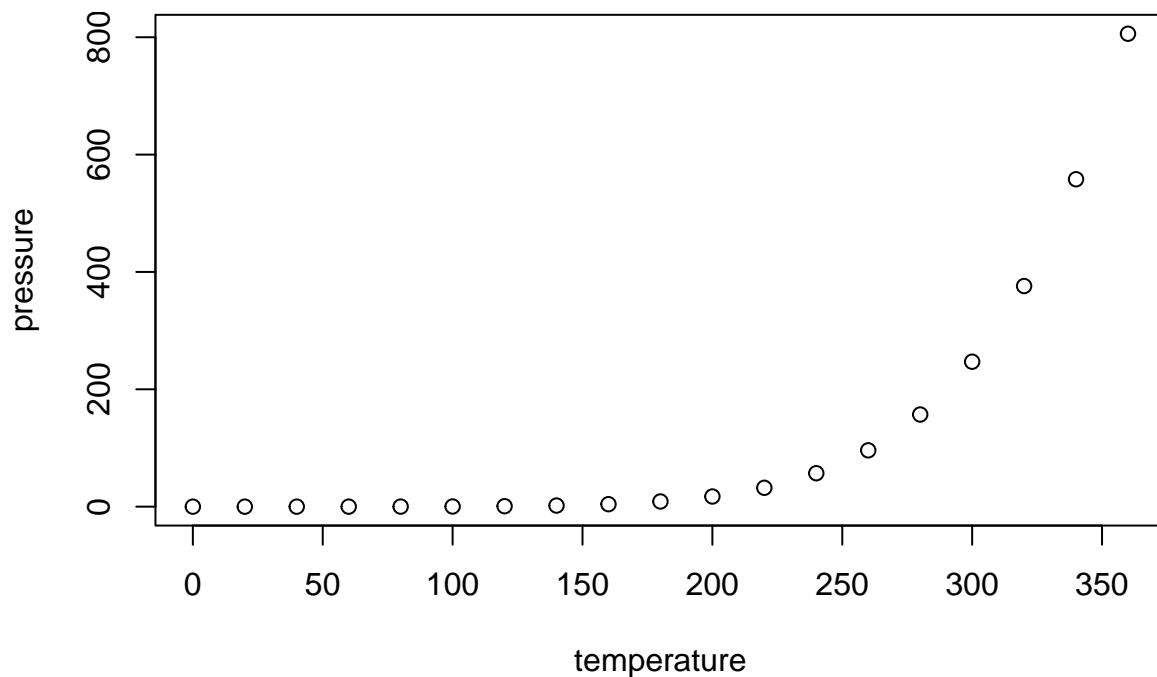
```
## [1] TRUE TRUE
```

```
accuracy<-(sum(check)/nrow(test_data))*100
print(accuracy)
```

```
## [1] 100
```

Including Plots

You can also embed plots, for example:



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