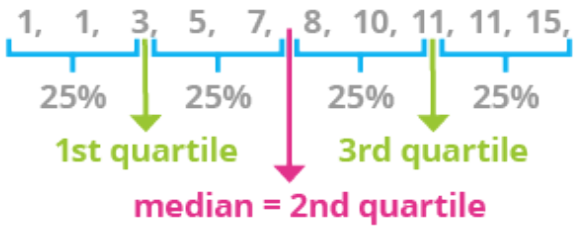
Machine Learning - 1100-ML0ENG (Ćwiczenia informatyczne Z-23/24)

Google Trends and the Stock Market - Classification

We see, that NN may be more useful as a classifier. Let's use again the Stock Market data. We mark the samples according to their **RealEstate** categorization. For those higher than 75% percentile, we give them label **High**; for those lower than 0.25 percentile, we label them as **Low**; otherwise, we label them **Median.** Note that even in the classification setting, the responses still must be numeric.




```
summary(google)
q3=quantile(google$RealEstate,0.75)
q1=quantile(google$RealEstate,0.25)
m=max(google$RealEstate)
```

```
temp=cut(google$RealEstate,c(0,q1-0.01,q3,m))
levels(temp)=c("Low","Median","High")
table(temp)
```

```
google$RealEstate=temp
```

We have prepared a dataset for classification. But this is not the end of the transformation. Now we need to transform it for neural network classification.

Normalization training and test datasets

```
google.norm.c<-as.data.frame(lapply(google[-3], normalize))
google.norm.c$RealEstate=google$RealEstate
```

```
set.seed(123)
split = sample.split(google.norm.c$RealEstate, SplitRatio = 0.7)
training_set = subset(google.norm.c, split == TRUE)
test_set = subset(google.norm.c, split == FALSE)
```

```
prop.table(table(google.norm.c$RealEstate))
prop.table(table(training_set$RealEstate))
```

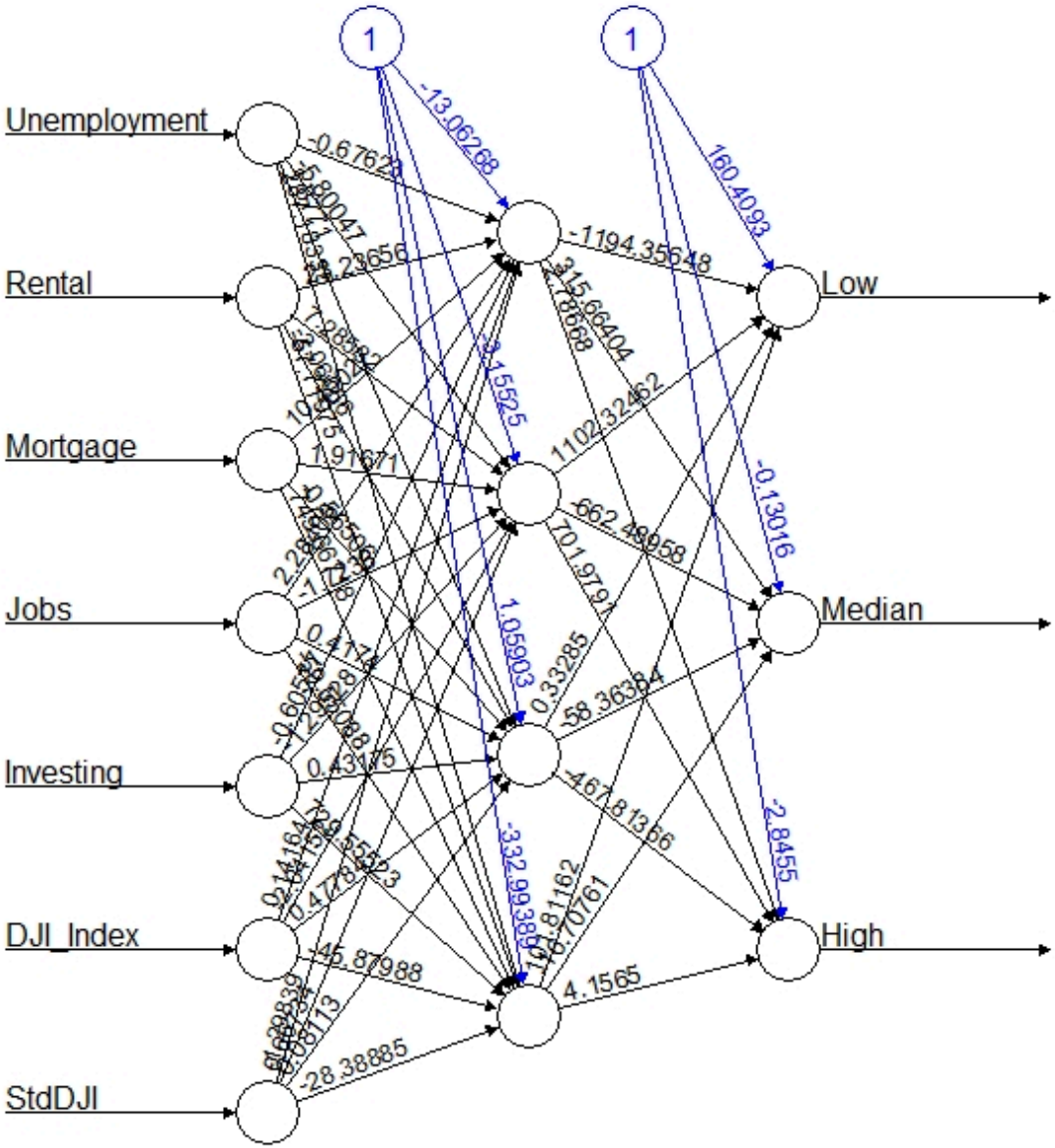
Since we have 3 values as labels for the target variable, we add 3 additional columns.

```
training_set$Low = training_set$RealEstate == "Low"
training_set$Median = training_set$RealEstate == "Median"
training_set$High = training_set$RealEstate == "High"
training_set=training_set[-8]
```

Neural network model

We use non-linear output and display intermediate results every 5,000 iterations.

```
set.seed(123)
google.nn.c=
neuralnet(Low+Median+High~Unemployment+Rental+Mortgage+Jobs+Investing+DJI_Index+StdD
JI,
 data = training_set,
 hidden=4,
 linear.output=FALSE,
 lifesign="full", lifesign.step=5000)
plot(google.nn.c)
```



Error: 8.012577 Steps: 94065

Evaluating model

```
net1.predict = compute(google.nn.c, test_set[-8])
str(net1.predict)
net1.predict$net.result
```

we obtained one column each for low, median and high.

We choose the best column, with the highest value

```
best.column=apply(net1.predict$net.result, 1, which.max)
```

and finally

```
net.prediction = c("Low","Median","High")[best.column]
```

and confiusion matrix

```
table(test_set$RealEstate, net.prediction)
```

In order to order the confusion matrix, the factor obtained must first be ordered

```
ordered.net.prediction <- factor(net.prediction, levels=c("Low","Median","High"),
ordered=TRUE)
table(test_set$RealEstate, ordered.net.prediction)
```

Last modified: środa, 26 maja 2021, 6:35

Accessibility settings

Przetwarzanie danych osobowych

Platformą administruje Komisja ds. Doskonalenia Dydaktyki wraz z Centrum Informatyki Uniwersytetu Łódzkiego Więcej

Informacje na temat logowania

Na platformie jest wykorzystywana metoda logowania za pośrednictwem Centralnego Systemu Logowania.

Studentów i pracowników Uniwersytetu Łódzkiego obowiązuje nazwa użytkownika i hasło wykorzystywane podczas logowania się do systemu USOSweb.

Deklaracja dostępności