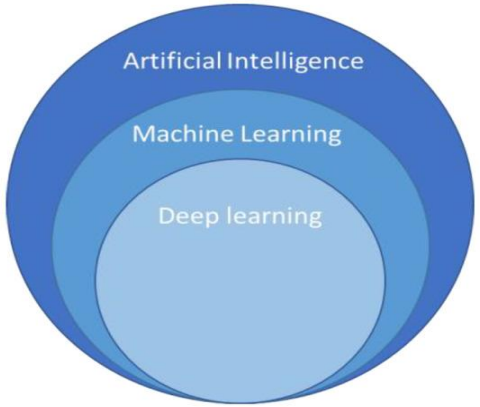
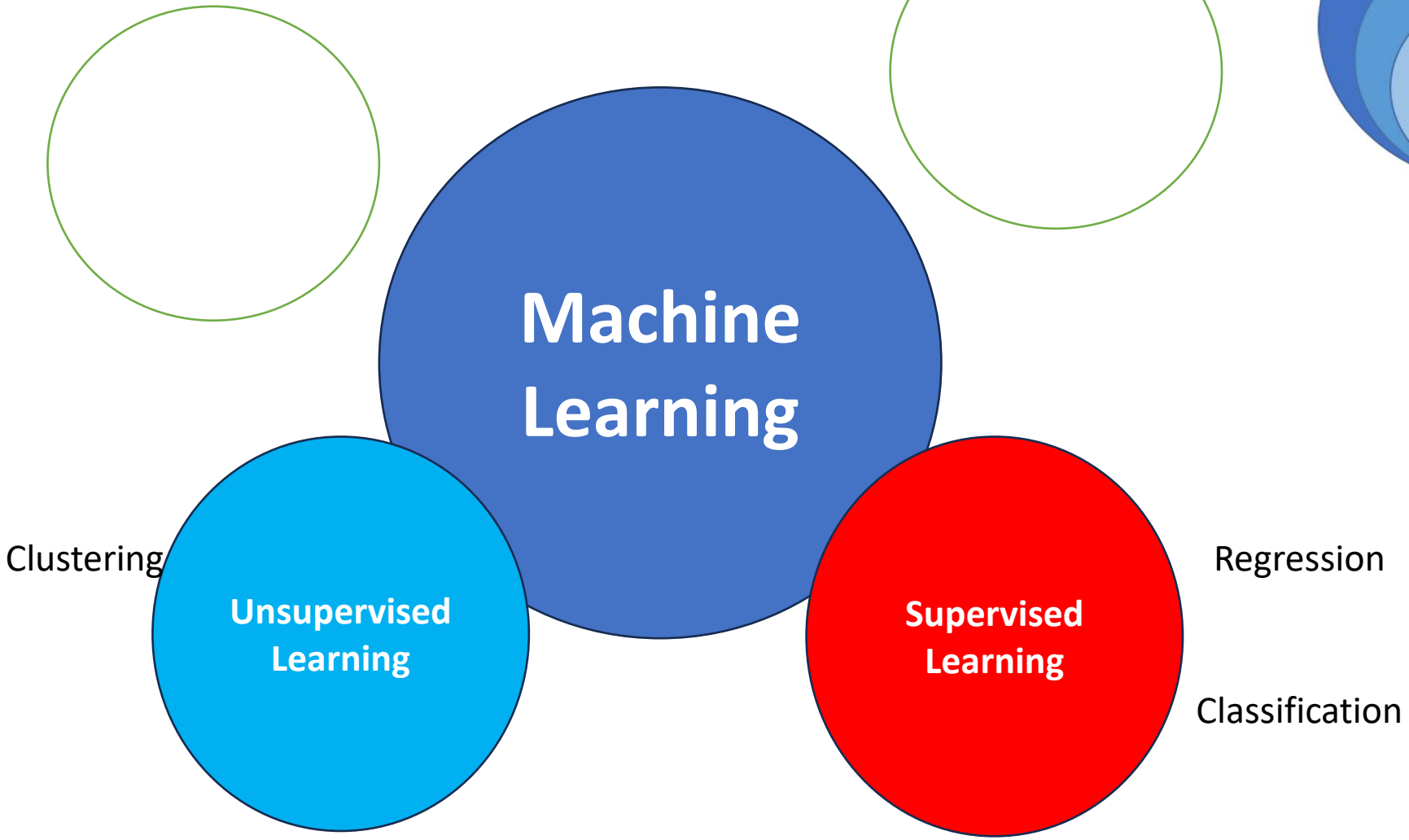
Model Behavior: Training a Machine Learning Network in Real-Time

Agenda

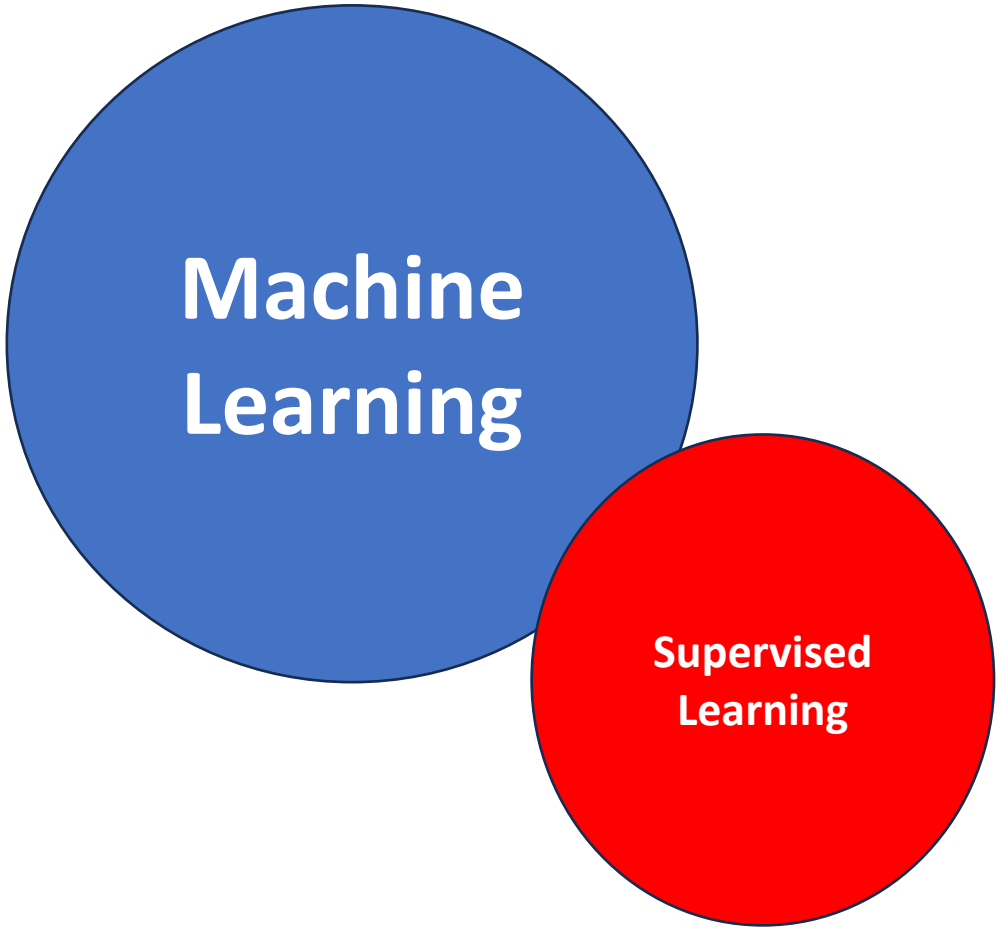
- Supervised Machine Learning Classification
- When to use it (Cases)
- About R
- Credit Card dataset
- Neural Networks

Machine Learning





Supervised Learning



Binary $\begin{bmatrix} 1 & 0 \end{bmatrix}$

Multinomial $\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$ *if red*

$\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$ *if green*

$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$ *if blue*

Classification

When to use it (Cases)

- Document classification
- Spam filtering
- Medical diagnostic test
- Customer behavior prediction
- Fraud detection

R and Data Science process

Data Story - Credit Card Dataset

Supervised ML Model (Classification)

Supervised ML Model (Classification)

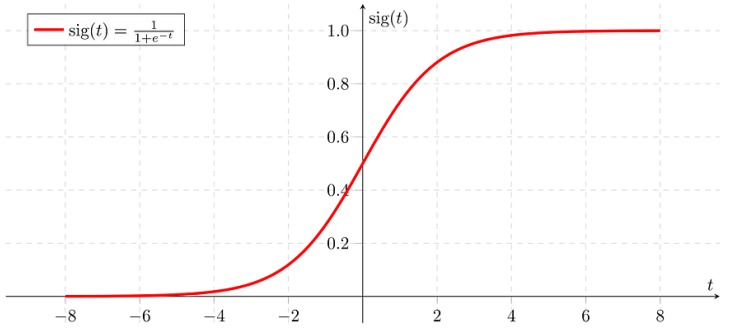
Cross Validation

Random Split:

Classification model assumptions

There is minimal or no multicollinearity among the independent variables.

The observations to be independent of each other.

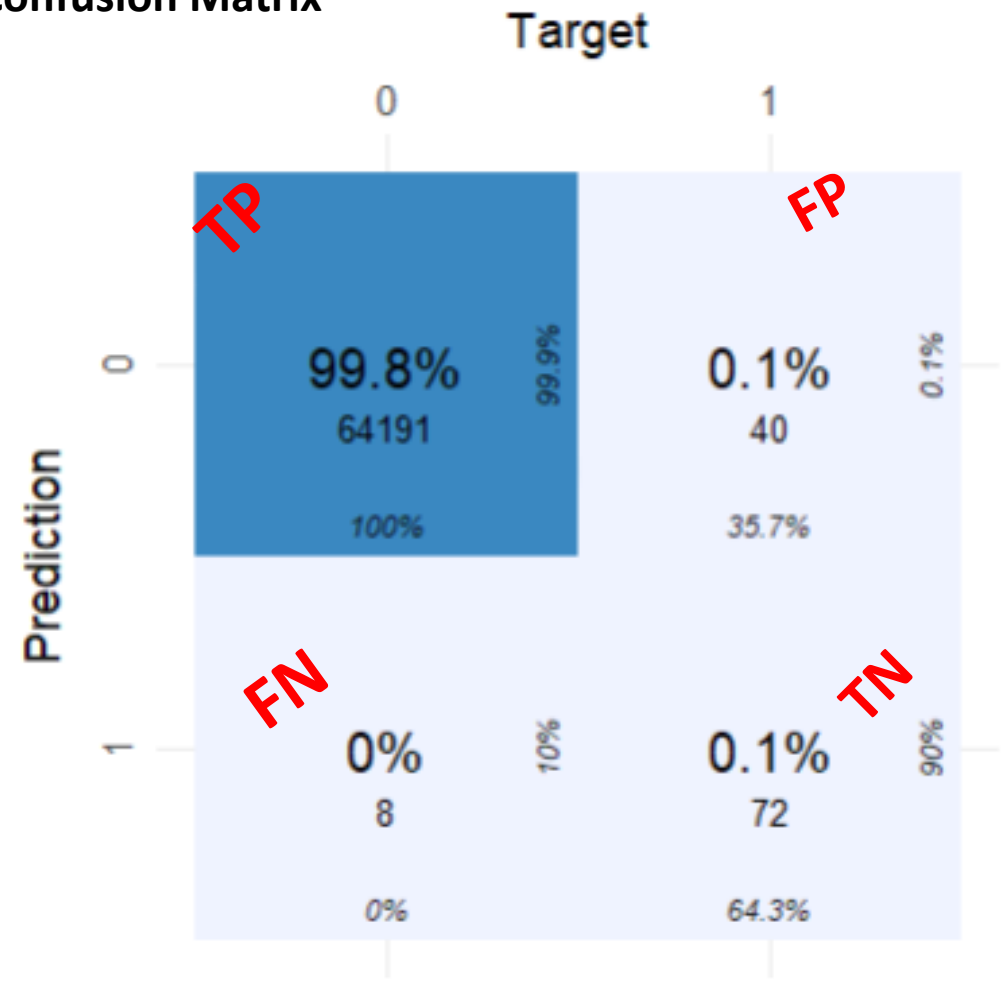


Model evaluation measures

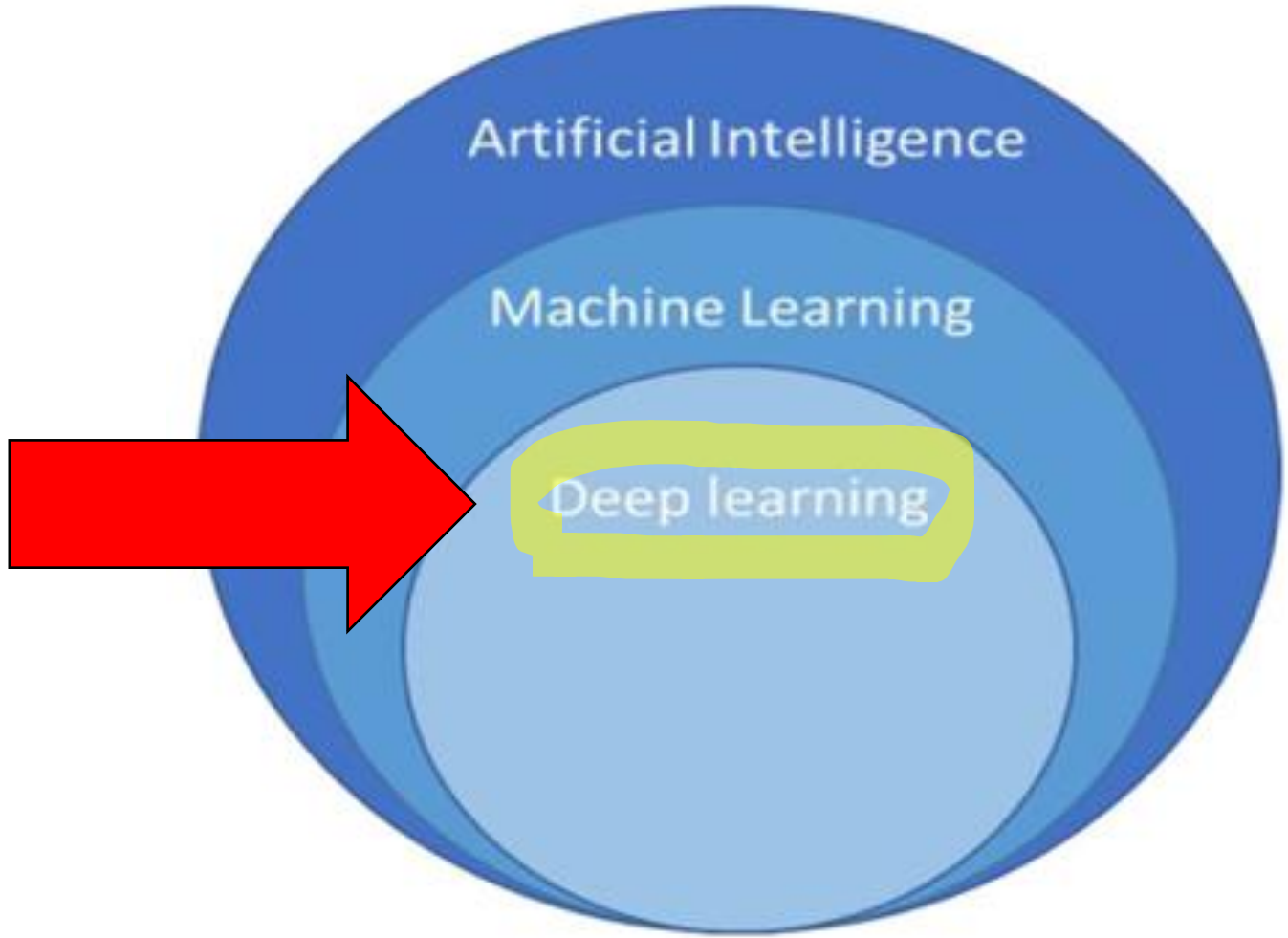
- Confusion Matrix- used to describe the performance of a classification model
- Accuracy – simply measures how often the classifier correctly predicts
- ROC AUC score shows how well the classifier distinguishes positive and negative classes. It can take values from 0 to 1

Supervised Classification model

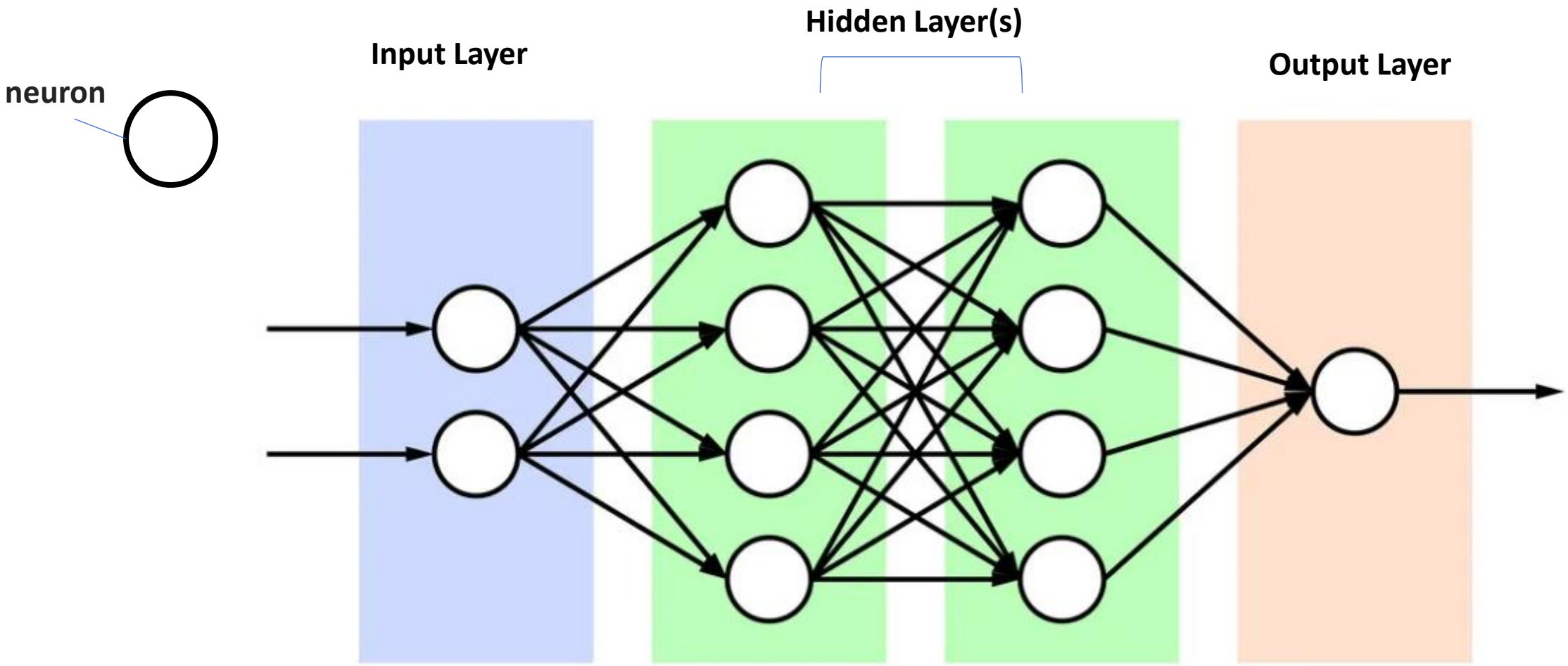
Confusion Matrix



Evaluation	Scores
Accuracy	.9
Recall	.64
Precision	.9
ROC AUC	.96
F 1 score	.75


Artificial Intelligence
Machine Learning
Deep learning

Neural Network

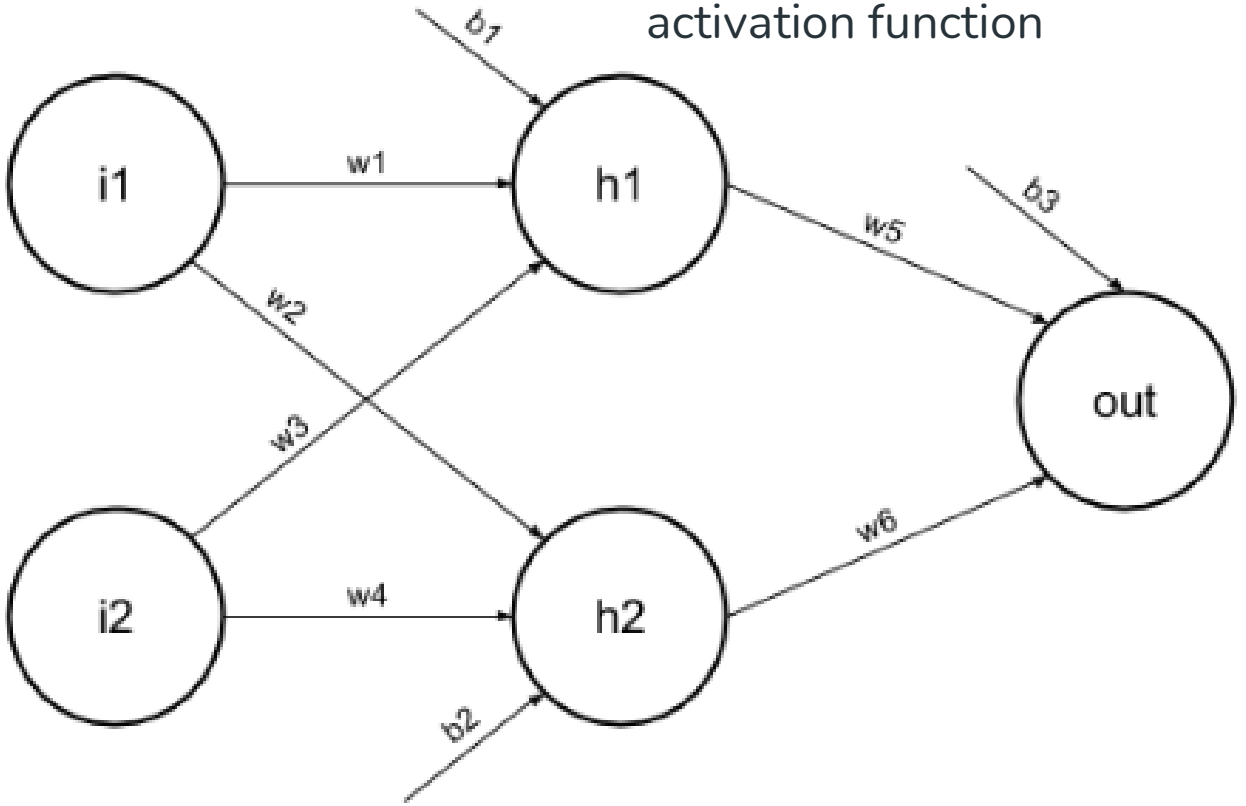


Neural Network

Basis

weights

activation function



Back to the **Credit Card Dataset story**
Fraud detection

Using neuralnet package

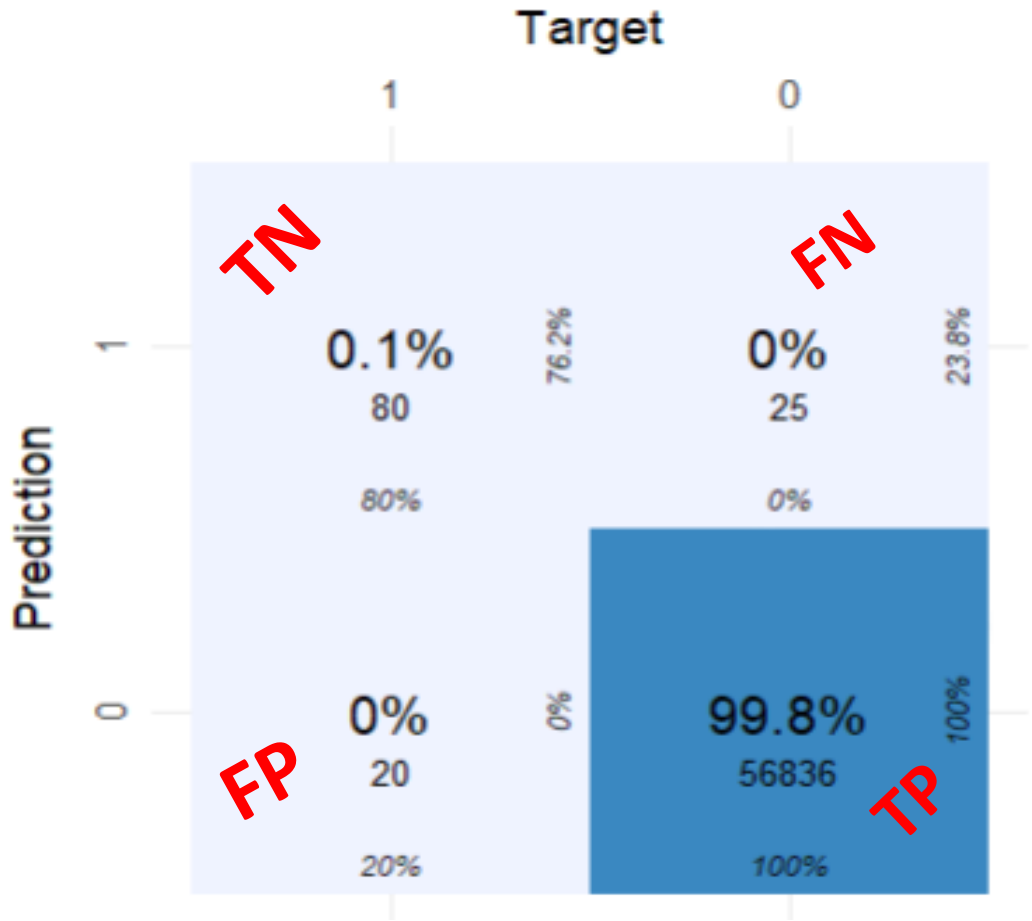
```
library(neuralnet)
split <- createDataPartition(y = df$Class, p=  0.8, list = F)

creditcard.training <- df[split,]
creditcard.test <- df[ split,]

creditcart.training.two <- creditcard.training%>% mutate_at(c(1:30),
funs(c(scale(.))))


nn_model <- neuralnet(Class ~ ., data = creditcart.training.two, hidden =
c(5,2), linear.output = F)
```

Using neuralnet package

```
library(neuralnet)

# plot our neural network

plot(nn_model, rep ="best")

#scale the test dataset
setcreditcart.test.two <- creditcard.test %>% mutate_at(c(1:30),
funs(c(scale(.))))

#create the predicted values
predicted.nn.values <- neuralnet::compute(nn_model, creditcart.test.two)
```

Deep learning (neural networks) model

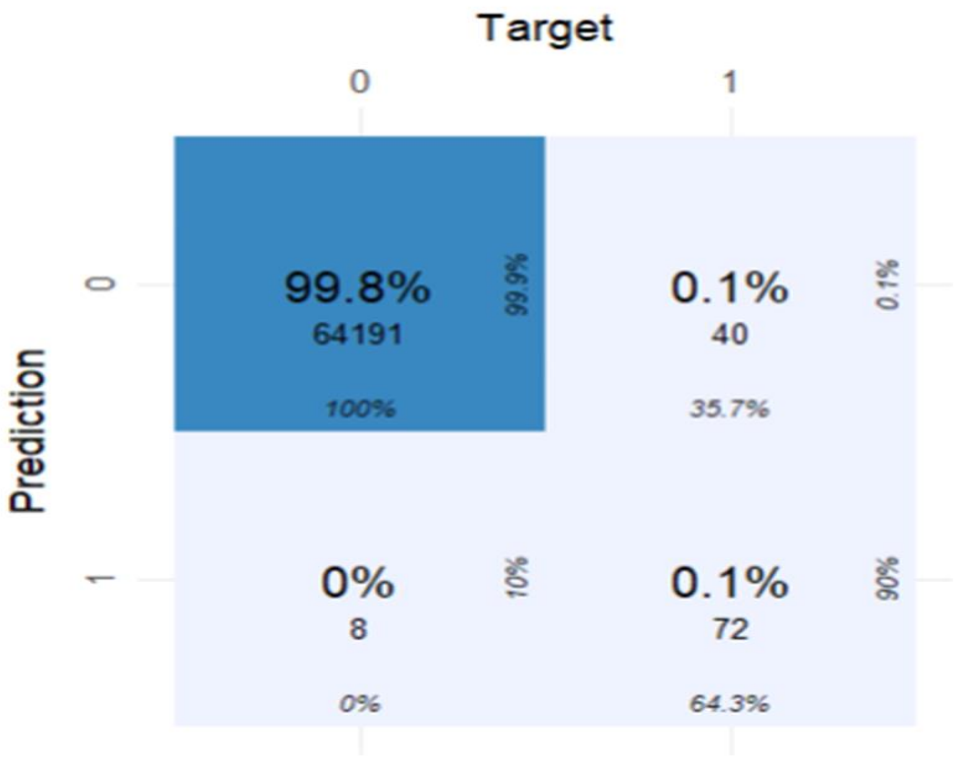
Confusion Matrix



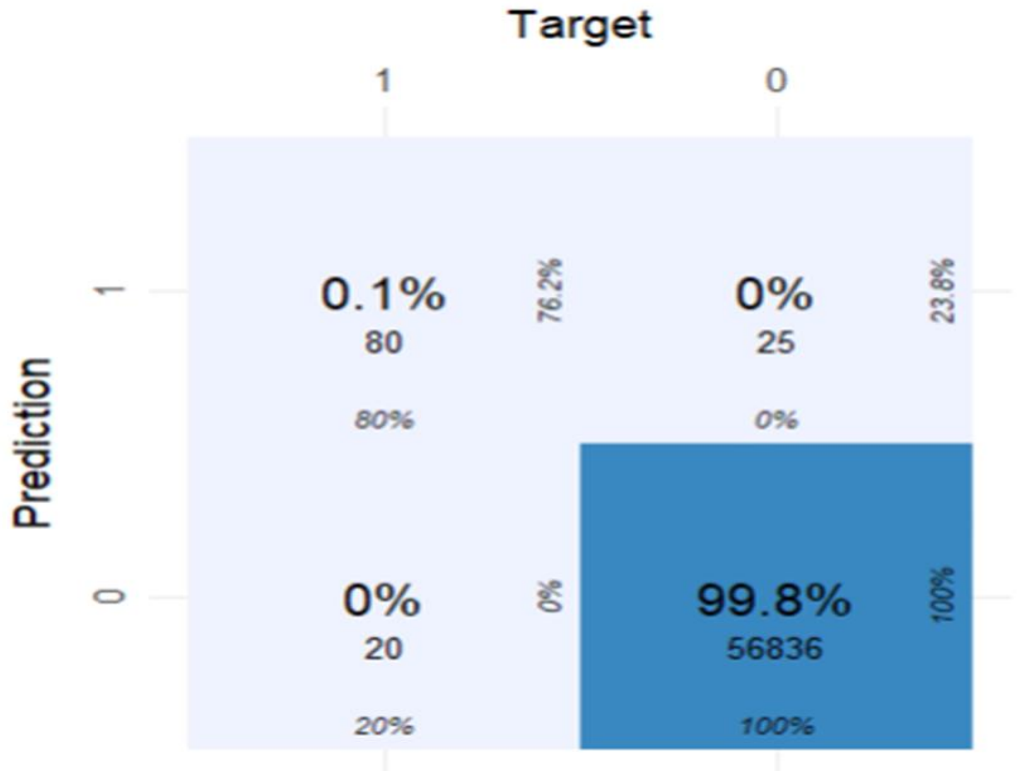
Evaluation	Scores
Accuracy	.99
Recall	.99
Precision	.99
ROC AUC	.89
F 1 score	.91

Confusion matrix

Supervised Classification model



Deep learning (neural networks) model



Model Evaluation Metrics

Supervised Classification model

Evaluation	Scores
Accuracy	.90
Recall	.64
Precision	.90
ROC AUC	.96
F 1 score	.75

Deep learning (neural networks) model

Evaluation	Scores
Accuracy	.99
Recall	.99
Precision	.99
ROC AUC	.89
F 1 score	.91

What did we learn?