

## Perceptron

```
from sklearn.linear_model import Perceptron
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

data = arff.loadarff(r'C:\Users\reyna\Desktop\Travail\UQAC\Trimestre2-Hiver\DeepLearning\Travail2\RFID_Features_windows5.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
df = shuffle(df)
print(df.tail()) #To see the last lines

X = df.iloc[:, :188].values
y = df.iloc[:, 188].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 1)

sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
#y_train = np.where(y_train == 'Chambre_a1', 1, -1)
#y_test = np.where(y_test == 'Chambre_a1', 1, -1)

ppn = Perceptron(tol = 0.0001, max_iter=100, random_state=1)
ppn.fit(X_train_std, y_train)

print("Perceptron : ")
print(ppn.score(X_test_std, y_test))
```

```
PS C:\Users\reyna> C:\Users\reyna\AppData\Local\Programs\Python\Python39\python.exe c:/Users/reyna/Desktop/Travail/UQAC/Trimestre2-Hiver/DeepLearning/Travail2/perceptron.py
countNonZeroRSSI_1 absEnergy_1 absSumOfChanges_1 meanChange_1 meanRSSI_1 ... globalMinRSSI globalMaxRSSI globalAbsEnergy globalMeanAbsSumChanges class
3132 5.0 20490.0 4.0 0.4 -64.0 ... -69.0 -42.0 110228.0 44.05 Cuisine_c4
16382 0.0 0.0 0.0 0.0 0.0 ... -66.0 -60.0 70226.0 0.25 Sallemanger_a8
5011 5.0 19228.0 6.0 -0.4 -62.0 ... -69.0 -49.0 244117.0 60.50 Cuisine_j7
15089 0.0 0.0 0.0 0.0 0.0 ... -68.0 -56.0 164300.0 7.35 Salon_c5
22636 0.0 0.0 0.0 0.0 0.0 ... -68.0 -62.0 84562.0 0.25 Sallemanger_h18

[5 rows x 189 columns]
Perceptron :
0.3955416756407495
PS C:\Users\reyna> []
```

## Logistic Regression

```
from sklearn.linear_model import LogisticRegression
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

data = arff.loadarff(r'C:\Users\reyna\Desktop\Travail\UQAC\Trimestre2-Hiver\DeepLearning\Travail2\RFID_Features_windows5.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
df = shuffle(df)
print(df.tail()) #To see the last lines

X = df.iloc[:, :188].values
y = df.iloc[:, 188].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 1)

sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
#y_train = np.where(y_train == 'Chambre_a1', 1, -1)
#y_test = np.where(y_test == 'Chambre_a1', 1, -1)

lr = LogisticRegression(C = 1000.0, max_iter=10000, random_state=1)
lr.fit(X_train_std, y_train)
print("LR : ")
print(lr.score(X_test_std, y_test))
```

```

PS C:\Users\reyna> & C:\Users\reyna\AppData\Local\Programs\Python\Python39\python.exe c:/Users/reyna/Desktop/Travail/UQAC/Trimestre2-Hiver/DeepLearning/Travail2/LR.py
countNonZeroRSSI_1 absEnergy_1 absSumofChanges_1 meanChange_1 meanRSSI_1 minRSSI_1 ... globalMeanStdDev globalMinRSSI globalMaxRSSI globalAbsEnergy globalMeanAbsSumChanges c
9302 5.0 16021.0 4.0 0.0 -56.6 -57.0 ... 17.998 -69.0 -55.0 175422.0 57.50 Cuisine
16958 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.040 -69.0 -64.0 90942.0 0.10 Salleamange
112 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.482 -69.0 -54.0 158985.0 1.75 Chambr
e_a3
18897 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.209 -69.0 -60.0 84309.0 0.60 Salleamange
r_d2
4320 5.0 15143.0 7.0 -0.6 -55.0 -58.0 ... 3.147 -68.0 -48.0 75931.0 10.75 Cuisin
e_f6

[5 rows x 189 columns]
C:\Users\reyna\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=-1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
n_iter_i = _check_optimize_result(
LR :
0.9363558044367866

```

## SVM

```

from sklearn.svm import SVC
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

data = arff.loadarff(r'C:\Users\reyna\Desktop\Travail\UQAC\Trimestre2-Hiver\DeepLearning\Travail2\RFID_Features_windows5.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
df = shuffle(df)
print(df.tail()) #To see the last lines

X = df.iloc[:, :188].values
y = df.iloc[:, 188].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 1)

sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
#y_train = np.where(y_train == 'Chambre_a1', 1, -1)
#y_test = np.where(y_test == 'Chambre_a1', 1, -1)

svm = SVC(kernel = 'linear', C = 1.0, random_state=1)
svm.fit(X_train_std, y_train)
print("SVM : ")
print(svm.score(X_test_std, y_test))

```

```

PS C:\Users\reyna> & C:\Users\reyna\AppData\Local\Programs\Python\Python39\python.exe c:/Users/reyna/Desktop/Travail/UQAC/Trimestre2-Hiver/DeepLearning/Travail2/SVM.py
countNonZeroRSSI_1 absEnergy_1 absSumofChanges_1 meanChange_1 meanRSSI_1 minRSSI_1 ... globalMinRSSI globalMaxRSSI globalAbsEnergy globalMeanAbsSumChanges class
30211 0.0 0.0 0.0 0.0 0.0 0.0 ... -68.0 -41.0 131203.0 1.40 Salleamanger r2
25917 0.0 0.0 0.0 0.0 0.0 0.0 ... -66.0 -47.0 71164.0 0.30 Salleamanger l17
16306 0.0 0.0 0.0 0.0 0.0 0.0 ... -68.0 -60.0 82566.0 0.30 Salleamanger a6
24906 0.0 0.0 0.0 0.0 0.0 0.0 ... -66.0 -51.0 71196.0 0.45 Salleamanger k13
14884 0.0 0.0 0.0 0.0 0.0 0.0 ... -69.0 -58.0 185506.0 7.40 Salon_b1

[5 rows x 189 columns]
SVM :
0.8517122550075382

```

## Decision Tree

```
from sklearn.tree import DecisionTreeClassifier
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

data = arff.loadarff(r'C:\Users\reyna\Desktop\Travail\UQAC\Trimestre2-Hiver\DeepLearning\Travail2\RFID_Features_windows5.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
df = shuffle(df)
print(df.tail()) #To see the last lines

X = df.iloc[:, :188].values
y = df.iloc[:, 188].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 1)

sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
#y_train = np.where(y_train == 'Chambre_a1', 1, -1)
#y_test = np.where(y_test == 'Chambre_a1', 1, -1)

tree = DecisionTreeClassifier(criterion = 'entropy', max_depth= 20, random_state=1)
tree.fit(X_train_std, y_train)
print("tree : ")
print(tree.score(X_test_std, y_test))
```

```
PS C:\Users\reyna> & C:/Users/reyna/AppData/Local/Programs/Python/Python39/python.exe c:/Users/reyna/Desktop/Travail/UQAC/Trimestre2-Hiver/DeepLearning/Travail2/arbres.py
countNonZeroRSSI 1 absEnergy 1 absSumofChanges 1 meanChange 1 meanRSSI 1 ... globalMinRSSI globalMaxRSSI globalAbsEnergy globalMeanAbsSumChanges class
20076 0.0 0.0 0.0 0.0 0.0 0.0 ... -66.0 -59.0 81113.0 0.15 Salleamanger_e16
25114 0.0 0.0 0.0 0.0 0.0 0.0 ... -68.0 -60.0 81536.0 0.50 Salleamanger_k18
26272 0.0 0.0 0.0 0.0 0.0 0.0 ... -69.0 -53.0 125399.0 25.55 Salleamanger_m7
18584 0.0 0.0 0.0 0.0 0.0 0.0 ... -69.0 -60.0 83541.0 0.05 Salleamanger_d2
7196 5.0 19988.0 4.0 0.8 -63.2 ... -69.0 -42.0 246834.0 54.65 Cuisine_l11

[5 rows x 189 columns]
tree :
0.9543398664656472
```

## K Nearest Neighbors

```
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

data = arff.loadarff(r'C:\Users\reyna\Desktop\Travail\UQAC\Trimestre2-Hiver\DeepLearning\Travail2\RFID_Features_windows5.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
df = shuffle(df)
print(df.tail()) #To see the last lines

X = df.iloc[:, :188].values
y = df.iloc[:, 188].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 1)

sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
#y_train = np.where(y_train == 'Chambre_a1', 1, -1)
#y_test = np.where(y_test == 'Chambre_a1', 1, -1)

knn = KNeighborsClassifier(n_neighbors= 5, p= 2, metric= 'minkowski')
knn.fit(X_train_std, y_train)
print("k nearest neighbors : ")
print(knn.score(X_test_std, y_test))
```

```

PS C:\Users\reyna> & C:\Users\reyna\AppData\Local\Programs\python\python39\python.exe c:/Users/reyna/Desktop/Travail/UQAC/Trimestre2-River/DeepLearning/Travail2/kNearestNeighbors.py
countNonZeroRSSI_1 absEnergy_1 absSumofChanges_1 meanChange_1 meanRSSI_1 ... globalMinRSSI globalMaxRSSI globalAbsEnergy globalMeanAbsSumChanges class
18502 0.0 0.0 0.0 0.0 0.0 ... -69.0 -65.0 91426.0 0.55 Sallemanger_c18
27160 0.0 0.0 0.0 0.0 0.0 ... -68.0 -46.0 116375.0 9.10 Sallemanger_n8
2288 5.0 16949.0 7.0 -0.2 -58.2 ... -69.0 -52.0 106465.0 35.50 Cuisine_f2
17545 0.0 0.0 0.0 0.0 0.0 ... -69.0 -61.0 88689.0 0.25 Sallemanger_b15
11220 5.0 18883.0 4.0 -0.4 -61.4 ... -69.0 -58.0 193539.0 4.75 Cuisine_a18

[5 rows x 189 columns]
k nearest neighbors :
0.6283652810682748

```