INTRODUCTION TO ML TOOLS

AARHUS UNIVERSITY
DEPARTMENT OF BIOSCIENCE

9 MAY 2019 | XIAODONG DUAN
POSTDOC

SCIKIT-LEARN

AARHUS UNIVERSITY
DEPARTMENT OF BIOSCIENCE

9 MAY 2019 | XIAODONG DUAN
POSTDOC

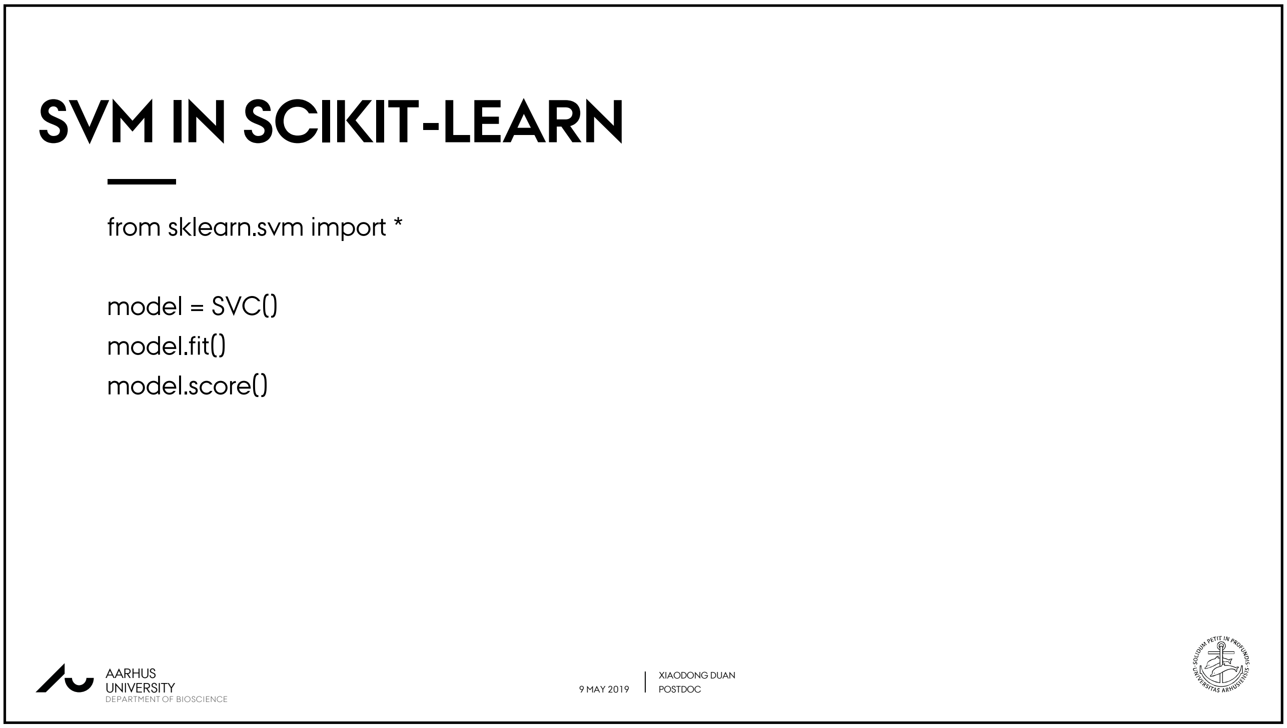
SCIKIT-LEARN

1. Open source machine learning tools for Python.
2. It is built on NumPy and Matplotlib.
   - NumPy: matrix operations
   - Matplotlib: visualization
3. Plenty of machine learning methods:
   - Feature extraction
   - SVM
   - PCA
   - ...

PCA IN SCIKIT-LEARN

```
from sklearn import decomposition

pca_f = decomposition.PCA(10)
pca_f.fit()
pca_f.transform()
```



SVM IN SCIKIT-LEARN

```
from sklearn.svm import *

model = SVC()
model.fit()
model.score()
```

KERAS

KERAS

1. It is a high-level tool in Python for neural networks.
2. You can choose TensorFlow, CNTK or Theano as backend.
3. It is easy and fast to build a neural network

KERAS

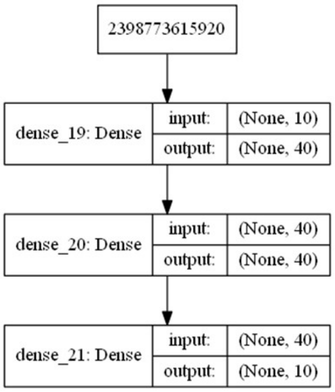
1. **Sequential**: a linear stack of layers.
   - compile(): configure the model for training.
   - fit(): train the model on the given data.
   - evaluate(): test on the given data and return the loss value on metrics value.
   - predict(): return the predictions for the given simples.
2. Core layer:
   - Dense(): fully connected layer
     - Activation functions: softmax, relu, sigmoid
3. keras.utils.to_categorical(): convert a scalar class number to a vector
   - Suppose we have 3 classes: [1 0 0], [0 1 0], [0 0 1]

KERAS

```
import numpy as np
import keras
from keras.utils import *
from keras.models import Sequential
from keras.layers import Dense

model = Sequential()
model.add(Dense(40,input_dim=10, activation='relu'))
model.add(Dense(40, activation='relu'))
model.add(Dense(10, activation='softmax'))
model.compile(loss=keras.losses.categorical_crossentropy,
          optimizer=keras.optimizers.Adadelta(),
          metrics=['accuracy'])
plot_model(model, to_file='model.png', show_shapes=True)
```

KERAS



AARHUS
UNIVERSITY