MAI4CAREU
Master programmes in Artificial Intelligence 4 Careers in Europe

University of Cyprus

HUMAN-CENTERED INTELLIGENT USER INTERFACES - MAI648

Marios Belk

2022

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

LAB 7

Modeling Human Factors with Machine Learning

CONTENTS

- Laboratory Introduction
- Scikit-learn Machine Learning Library
- Scenario
- Classification
- Example Code – Step-by-step
- Additional Topics – K-means Clustering

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

LAB 7

Laboratory Description

- In this laboratory students will learn how machine learning can assist in the modeling process of human factors
- Based on a hypothetical research scenario you are asked to download a Python script (.py) and a dummy data file (.csv)
- Python will be used utilizing the scikit-learn machine learning library for Python

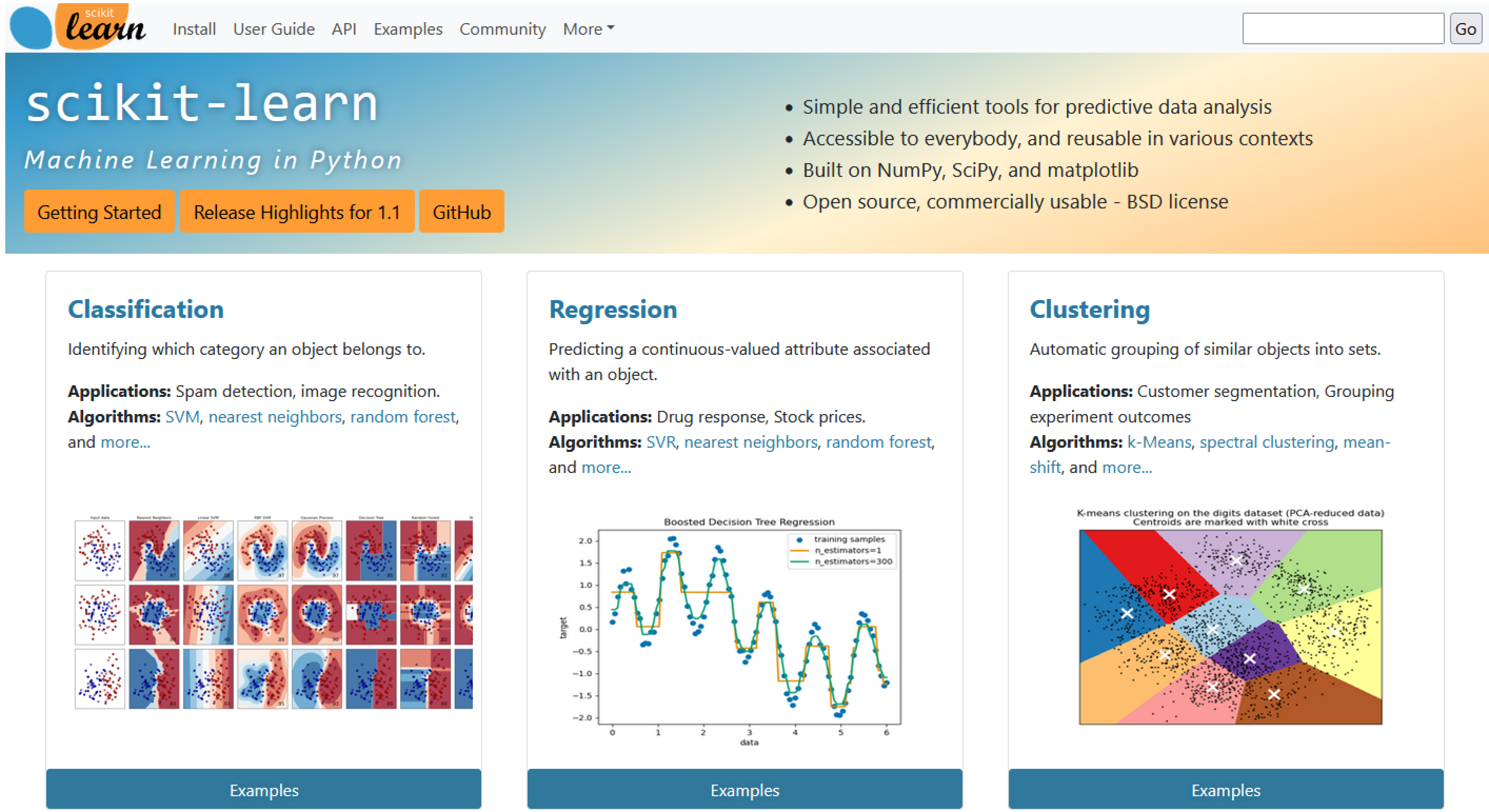
LAB 7

Scikit-learn Machine Learning Library

- Machine Learning in Python
  - Simple and efficient tools for data mining and data analysis
  - Accessible to everybody, and reusable in various contexts
  - Built on NumPy, SciPy, and matplotlib
  - Open source, commercially usable

- https://scikit-learn.org/stable/

LAB 7



scikit-learn

Machine Learning in Python

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

Clustering

Automatic grouping of similar objects into sets.

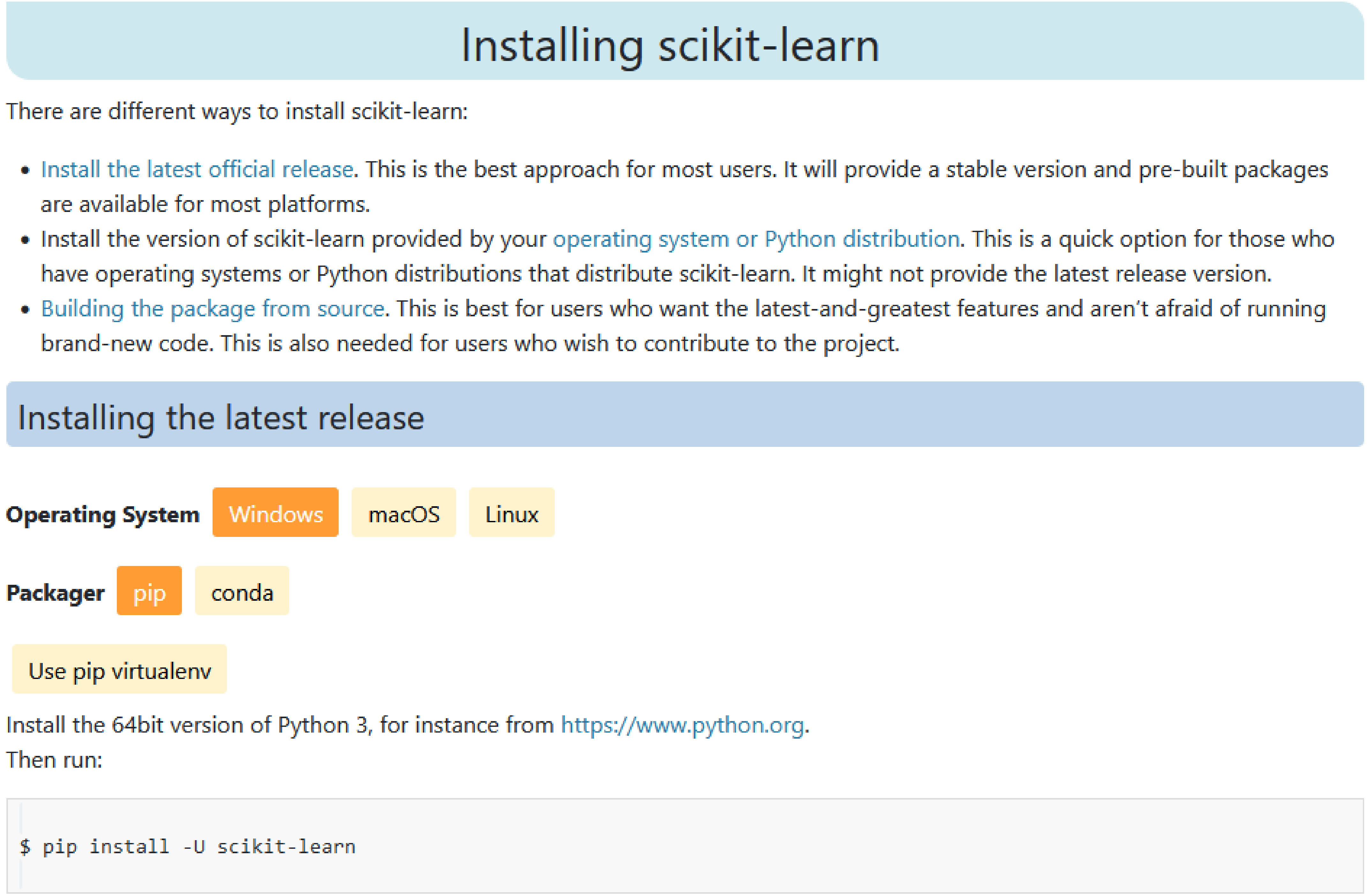
Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

LAB 7

Installing Scikit-learn

- https://scikit-learn.org/stable/install.html

- `pip install -U scikit-learn`



Installing scikit-learn

There are different ways to install scikit-learn:

- Install the latest official release. This is the best approach for most users. It will provide a stable version and pre-built packages are available for most platforms.
- Install the version of scikit-learn provided by your operating system or Python distribution. This is a quick option for those who have operating systems or Python distributions that distribute scikit-learn. It might not provide the latest release version.
- Building the package from source. This is best for users who want the latest-and-greatest features and aren't afraid of running brand-new code. This is also needed for users who wish to contribute to the project.

Installing the latest release

Operating System Windows macOS Linux

Packager pip conda

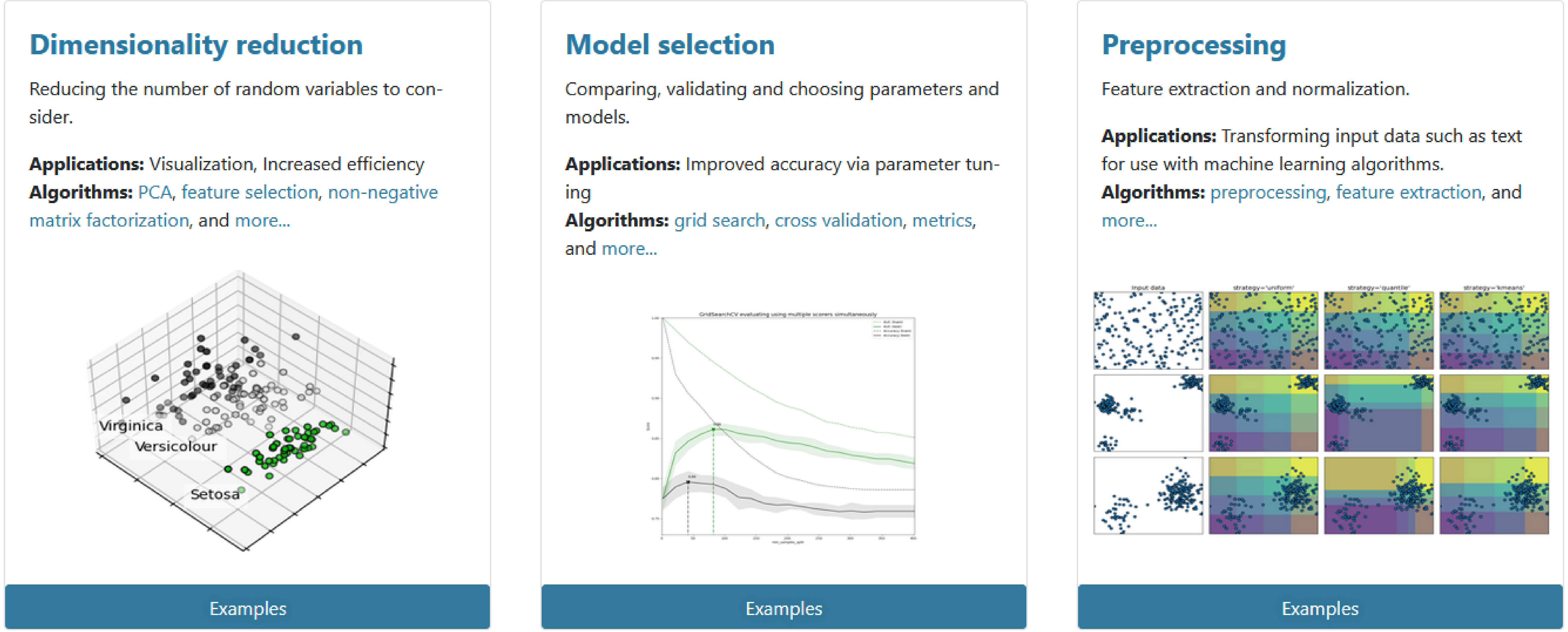
Use pip virtualenv

Install the 64bit version of Python 3, for instance from https://www.python.org.
Then run:

```
$ pip install -U scikit-learn
```

LAB 7

Documentation of scikit-learn - https://scikit-learn.org/stable/index.html



Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: PCA, feature selection, non-negative matrix factorization, and more...

Examples

Model selection

Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

Examples

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...

Examples

LAB 7

Hypothetical Scenario of an IUI Experiment

- Existing studies have shown a correlation between cognitive style differences (field dependent-independent) and specific eye-gaze metrics
- Consider the hypothetical scenario that you have run an experimental study in which users interacted with a graphical authentication task in which they were asked to create a secret picture password

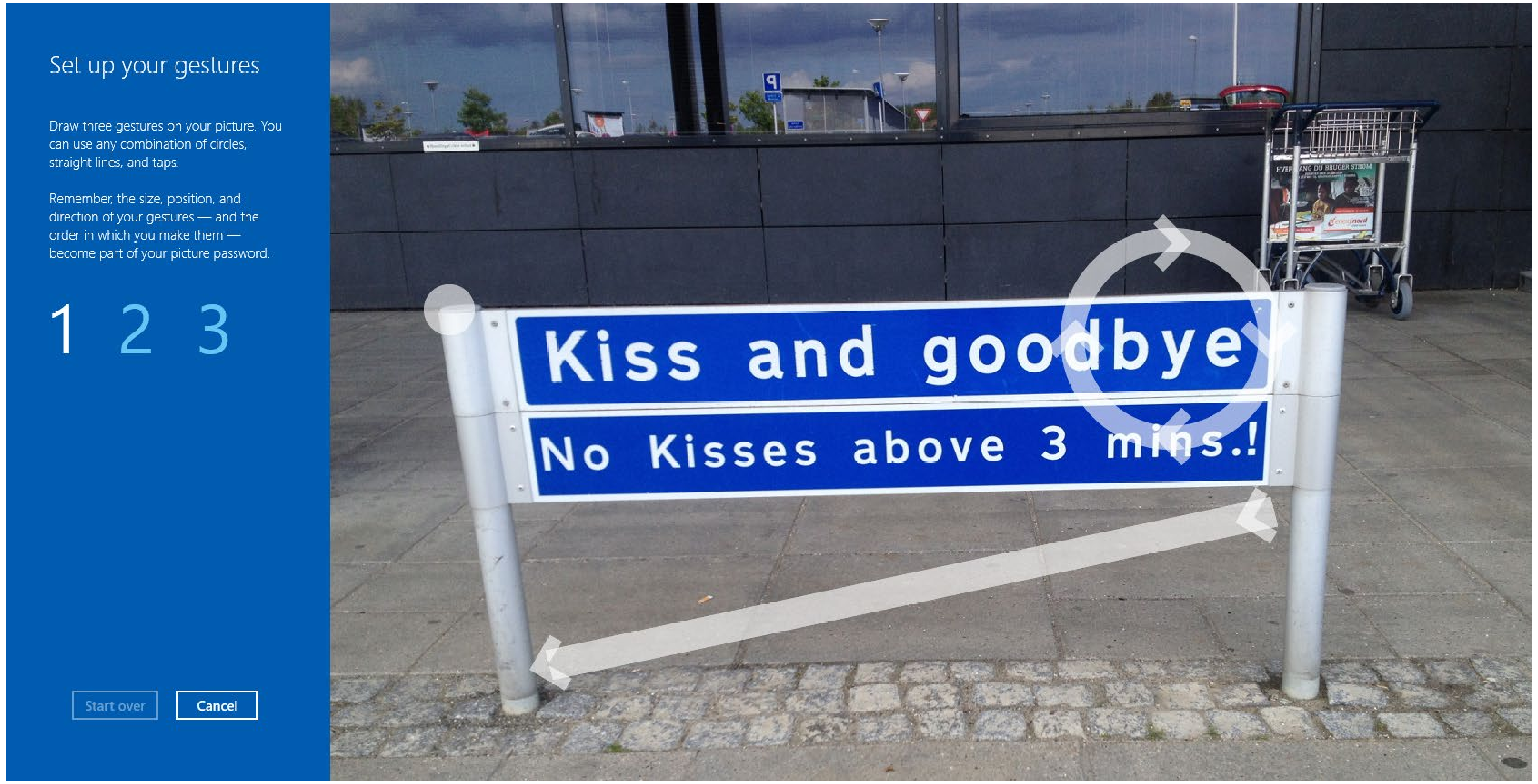
Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Hypothetical Scenario of an IUI Experiment

- During interaction, users wore a state-of-the-art eye tracking device which measured specific eye-gaze metrics (fixations count and duration)
- Users also completed a paper-and-pencil test (Group Embedded Figures Test) to highlight their cognitive style
- You are provided with a dataset that includes the users' aggregated eye gaze metrics and their respective cognitive style group

Picture Password Creation Task



Set up your gestures

Draw three gestures on your picture. You can use any combination of circles, straight lines, and taps.

Remember, the size, position, and direction of your gestures — and the order in which you make them — become part of your picture password.

1 2 3

Start over Cancel

Windows^{TM} Picture Gesture Authentication

LAB 7

Eye-Gaze Metrics

- Fixation duration: total duration of fixations of an individual within an area of interest (AOI), considering visits and revisits to the AOI
  - sum, mean, max, and std.
- Fixation count: total number of fixations of an individual within each AOI, considering visits and revisits to the AOI
- Gaze-transition entropy (Krejtz et al., 2015)
  - Transition entropy measures how random transition are among AOIs

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Classification Experiment

- We would like to train a classifier to predict the users' cognitive styles (Field Dependent *vs.* Field-Independent)

- Use the eye-gaze metrics as input data

- Compare the classification results with a baseline model (dummy classifier)

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

LAB 7

Classification Experiment

- In the next slides we will show an example tested with a classifier based on Logistic Regression

- You can further test other classifiers using the scikit-learn library (Naïve Bayes, k-Nearest Neighbors, Classification and Regression Trees, and Support Vector Machines)
  - https://scikit-learn.org/stable

LAB 7

Python Script and Data

- Please download the following files for this laboratory. The instructor will guide you through the code aiming to get familiar on how machine learning can be used to model human factors within intelligent user interfaces.

  - Python script: MAI648.labs.07.py
  - Data file: MAI648.labs.07.csv

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

Loading Python Libraries

```
# Import libraries
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.dummy import DummyClassifier
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_auc_score
from sklearn.metrics import roc_curve
```

Reading Data using Pandas

```
DIRECTORY = 'Combination_of_features/'

filename = DIRECTORY + 'data_second_X.csv'
data = pd.read_csv(filename, header=0)
data = data.dropna() #Drop missing values
```

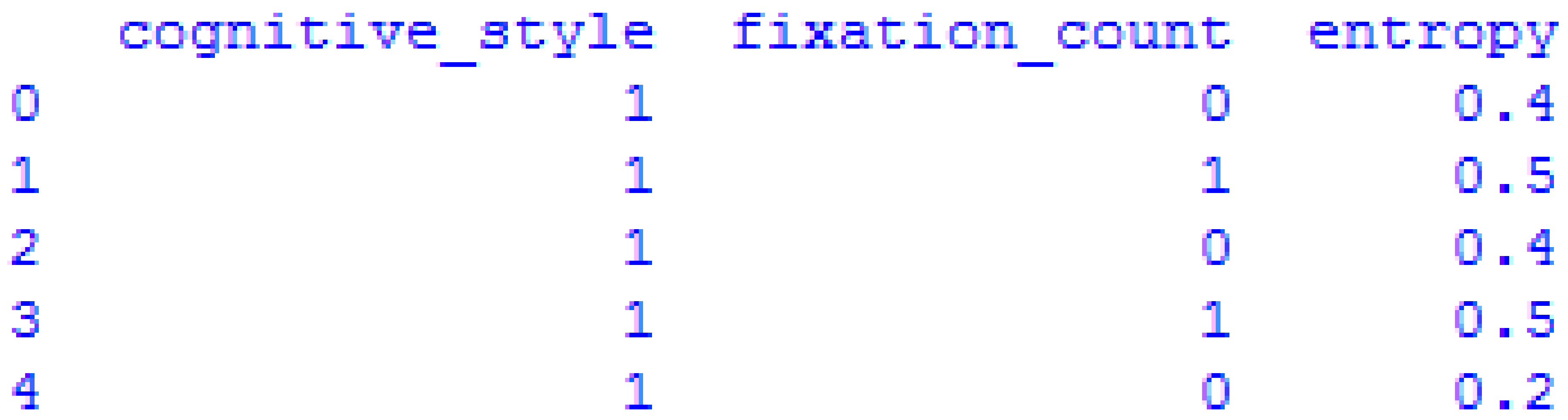
Other pandas commands to read other data formats:

```
pd.read_excel('file.xlsx',sheet_name='Sheet1',index_col=None,
na_values=['NA'])

pd.read_sas('myfile.sas7bdat')
…
```

https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.dropna.html

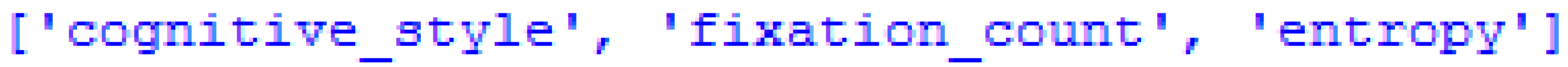
Exploring Data Frames

```
# List first 5 records
data.head()
```



```
   cognitive_style  fixation_count  entropy
0                1               0      0.4
1                1               1      0.5
2                1               0      0.4
3                1               1      0.5
4                1               0      0.2
```

```
# List data columns
list(data.columns)
```



```
['cognitive_style', 'fixation_count', 'entropy']
```

LAB 7

Select values for the X,Y data frames

```
# Select all values in column 1 and 2
X = data.iloc[:,[1,2]].values
# Select all values in column 0
y = data.iloc[:,0].values
```

iloc: selecting start and end index from the data frame

LAB 7

Training and Testing

```
# Split arrays or matrices into random train and test subsets
X_train, X_test, y_train, y_test = train_test_split(X, y,
random_state=0)
```

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

LAB 7

Classification

```
# Creating an instance of the classifier
classifier = LogisticRegression(random_state=0)

# Train the classifier
classifier.fit(X_train, y_train)

# Use the classifier to predict
y_pred = classifier.predict(X_test)
```

Accuracy of Classification Models

- In classification, accuracy estimation is measured through the confusion matrix
  - shows the actual and predicted labels from a classification problem

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive Count (TP)	False Positive Count (FP)
	Negative	False Negative Count (FN)	True Negative Count (TN)

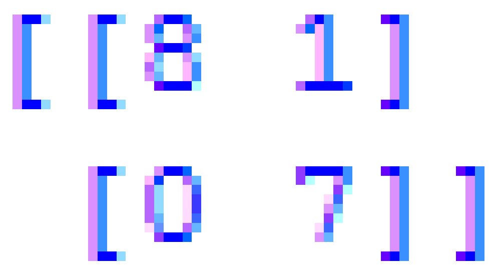
Classification Accuracy and Report

```
# Calculating the confusion matrix
confusion_matrix_res = confusion_matrix(y_test, y_pred)

# Classification accuracy
classifier.score(X_test, y_test)

# Creating an instance of the classifier
print(classification_report(y_test, y_pred))
```

LAB 7

Classification Accuracy

```
# Calculating the confusion matrix
confusion_matrix_res = confusion_matrix(y_test, y_pred)

# Classification accuracy
classifier.score(X_test, y_test)
```

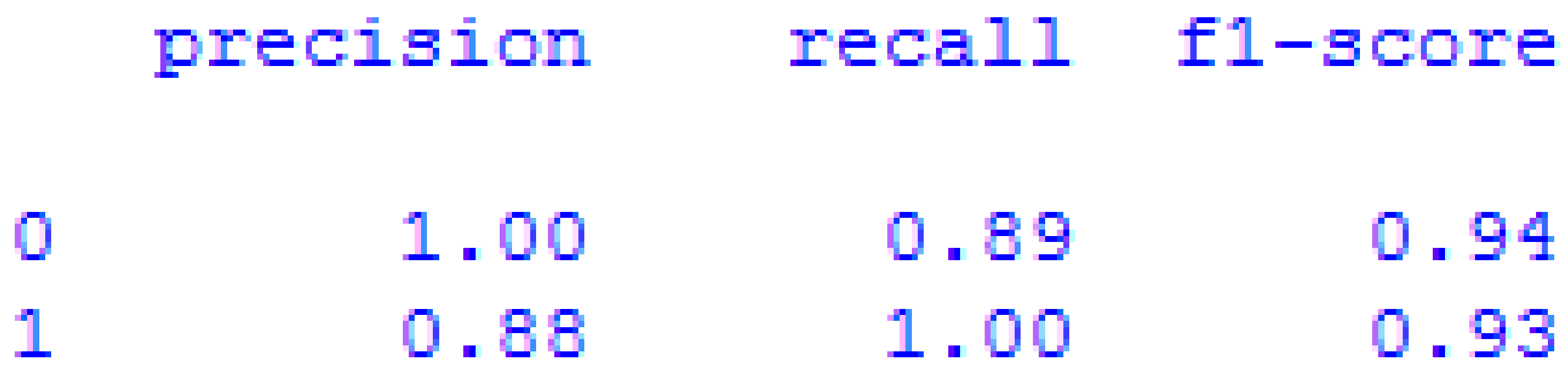


```
[[8 1]
 [0 7]]
```

LAB 7

Classification Report

```
# Creating an instance of the classifier
print(classification_report(y_test, y_pred))
```



```
          precision    recall  f1-score

       0       1.00      0.89      0.94
       1       0.88      1.00      0.93
```

LAB 7

Dummy Classifier

```
# Baseline dummy classifier
dummy = DummyClassifier(strategy='most_frequent', random_state=100,
constant=None)
dummy.fit(X_train, y_train)

y_dummy_pred = dummy.predict(X_test)

print('Accuracy of dummy classifier: {:.2f}'.format(dummy.score(X_test,
y_test)))
print(classification_report(y_test, y_dummy_pred))
```

Other Topics – Unsupervised Learning

- Unlabeled data
- We might not know what we are looking for
- Clustering: splitting data items into groups
    - Explore data
    - Put similar items in the same group
    - Distance between data items

K-means Clustering

- Main steps
  - Choose initial cluster centers
  - Assign data item to cluster
  - Recalculate cluster center

- K-means clustering in scikit-learn

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

LAB 7

K-means Clustering

```
# Library
from sklearn import cluster

# Create cluster instance
k_means = cluster.Kmeans(n_clusters=3)

# Fit the model on the data
km = k_means_fit(DATA)
```

LAB 7

Source and Further Readings

- MIT OpenCoursware. Introduction to Machine Learning - https://www.youtube.com/watch?v=h0e2HAPTGF4
- Documentation of scikit-learn - https://scikit-learn.org/stable/index.html
- Python documentation - https://docs.python.org/3/
- Adaptive and Interactive Systems Course, Cognitive Systems, Open University of Cyprus, and Department of Computer Science, University of Cyprus

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423

MAI4CAREU

Master programmes in Artificial Intelligence 4 Careers in Europe

Thank you.

Co-financed by the European Union
Connecting Europe Facility

This Master is run under the context of Action
No 2020-EU-IA-0087, co-financed by the EU CEF Telecom
under GA nr. INEA/CEF/ICT/A2020/2267423