Introduction to Machine Learning in Python

G. Parkes, S. Senior, T. Rampat

Overview

1. An introduction to Machine Learning
2. Ski-kit Learn
3. Practical 1 – Decision Trees/K-Means
4. More Theory
5. Practical 2 – TBD

Assumptions

- Solid understanding of Python.
- No/Little understanding of Machine Learning.
- Exposure to Pandas and NumPy beneficial.
- Knowledge of a Python IDE like *spyder* will be beneficial.

Machine Learning – What is it?

- A subset of Computer Science and derived from Artificial Intelligence (AI)
- At it's heart a probability generator, with strong ties to statistics/mathematical optimisation.
- Improving the probability score taken as 'learning'. Uses this learnt data to 'predict' on new data.
- Examples of use: *Spam Filtering, Character Recognition, Computer Vision, Search Engines, Social Media*

Machine Learning – What is it?

$$f(\boldsymbol{X}) = \boldsymbol{y}$$

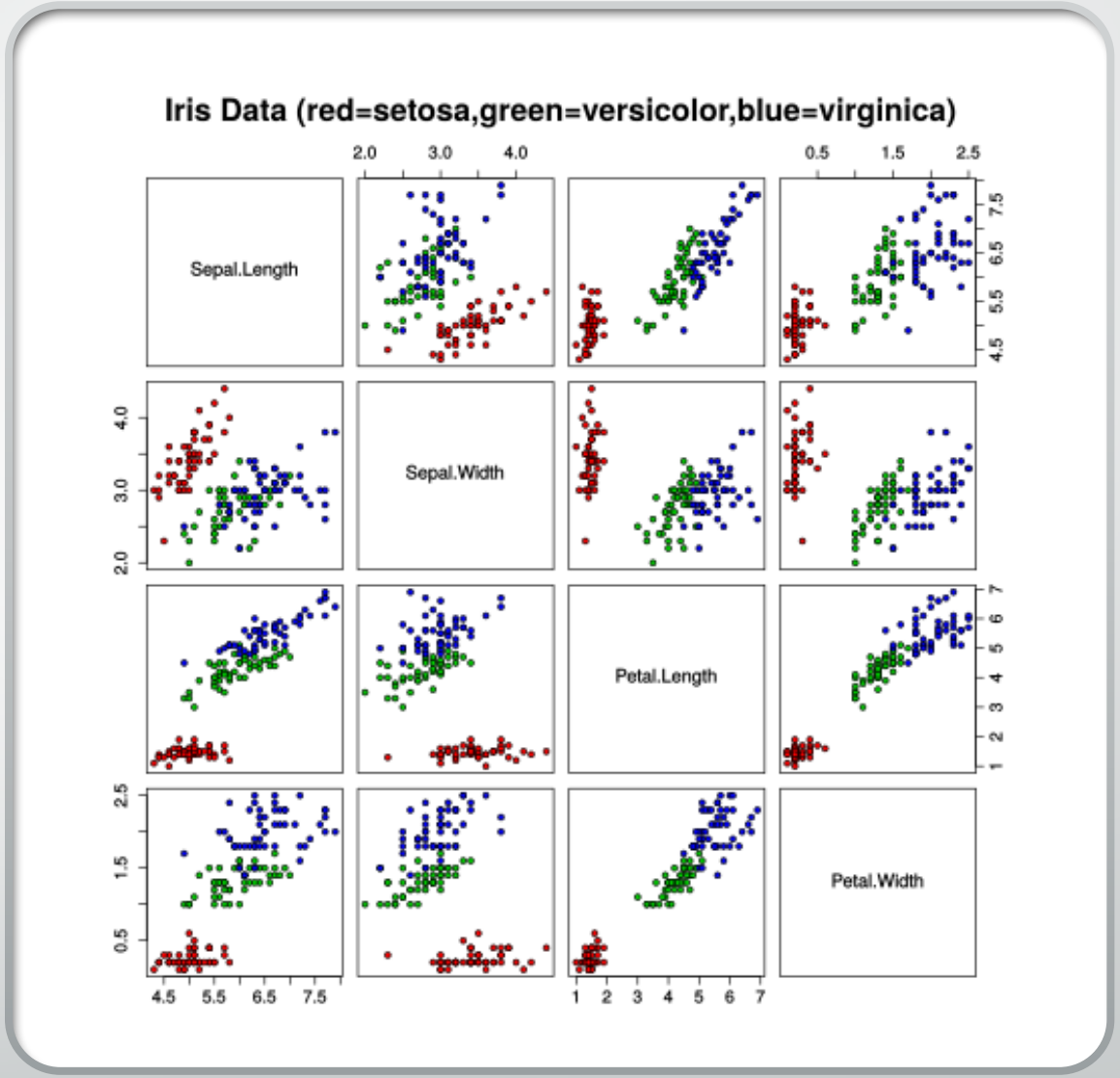
where *X* is a **matrix** of input, independent variables, and *y* is a **vector** of output, dependent variables *that classify X*.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	I. setosa
4.9	3.0	1.4	0.2	I. setosa
4.7	3.2	1.3	0.2	I. setosa
7.0	3.2	4.7	1.4	I. versicolor
6.4	3.2	4.5	1.5	I. versicolor
6.3	3.3	6.0	2.5	I. virginica

Taken from Fisher's *Iris* data set

Iris Dataset

- Each species is shaded in different colour
- Clear distinct pattern emerges
- ML trained to learn these patterns
- Applies it to new information when available

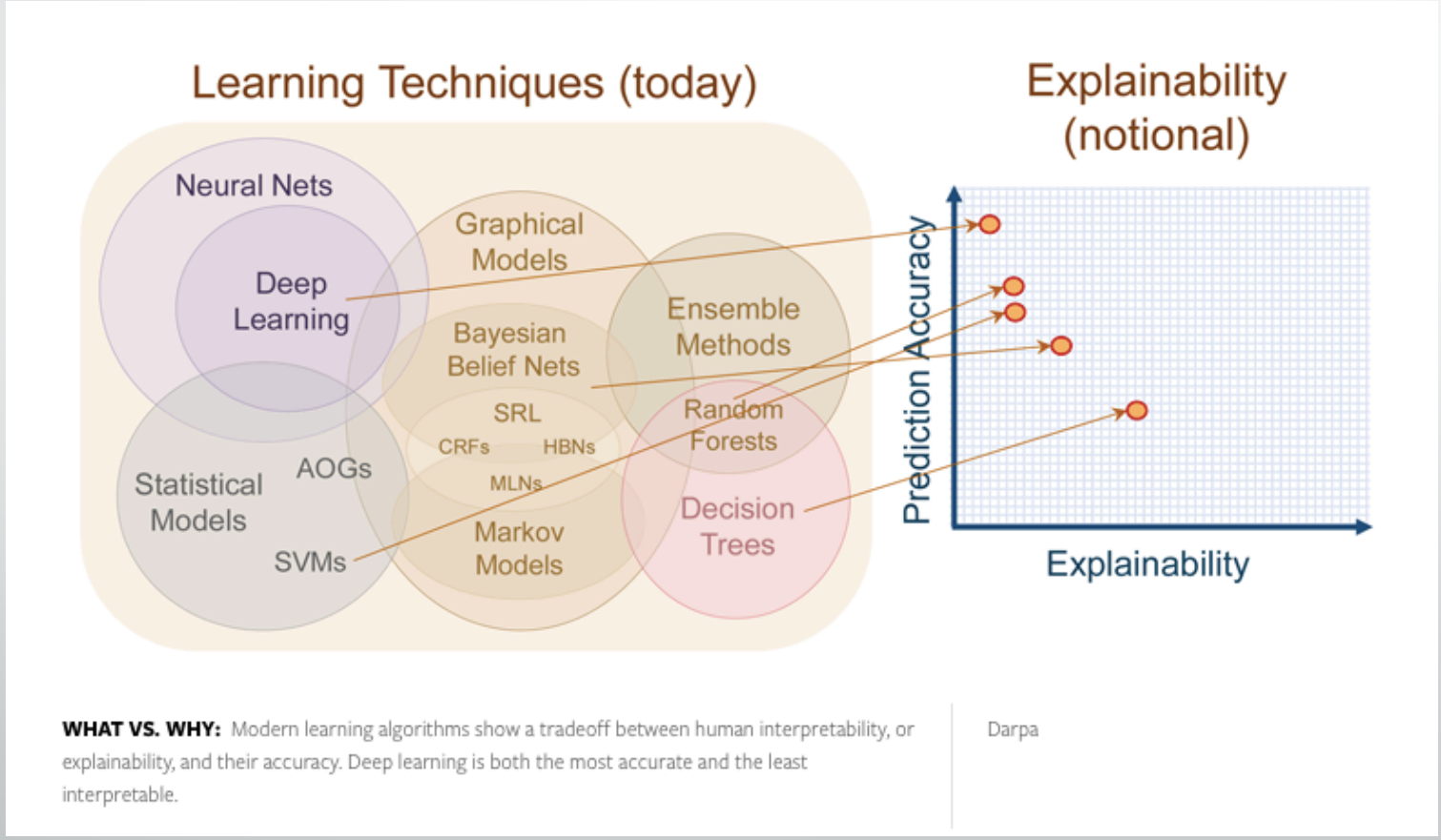


Machine Learning – Going slightly deeper

Three broad categories:

1. **Supervised Learning** – Presented with example inputs and corresponding outputs, given by a 'teacher'.
2. **Unsupervised Learning/Data Mining** – No labels given to inputs, computer infers features.
3. **Reinforcement Learning** – Interacts with a dynamic environment where it must perform a certain goal, feedback via rewards/punishments.

Understandability of Machine Learning



Taken from Nautil.us

Ski-kit Learn

- A package in Python programming language. Part of the Anaconda Distribution.
- Built on NumPy, SciPy and MatPlotLib libraries.
- Open source, commercially usable.
- Simple to set up for most applications, very powerful functionality.
- Highly accessible.

Ski-kit Learn

Name of Method	Example use	Algorithm
Classification	Spam Detection, Image Recognition	SVM, random forest, nearest neighbours
Regression	Drug response, stock prices	SVR, Ridge regression, LASSO
Clustering	Customer segmentation, grouping experiment outcomes	K-Means, Spectral Clustering, mean-shift
Dimensionality Reduction	Visualization, increased efficiency	PCA, Feature Selection, non-negative matrix factorisation
Model Selection	Improved accuracy via parameter tuning	Grid search, cross validation, metrics
Preprocessing	Transforming data (i.e text) for use in ML algorithms	Feature Extraction

Taken from skikit-learn.org

Practical 1.1 – Decision Trees

1. Enter the folder ‘Practical-1-1-DT’
2. Open ‘apples.py’, run the example, and make changes to the features, labels, and trying your own pieces of fruit. Try and understand what is happening.
3. A ‘tree.dot’ file will be created from running ‘apples.py’. Use `$ bash apples.sh $` to create a .png to visualise the decision tree.

Practical 1.2 – Classification & Iris Dataset

1. Enter the folder 'Practical-1-1-DT'
2. Open 'iris.py', run the example, and make changes to the code, such as features, labels etc.
3. Complete Tasks 1 and 2 by implementing iris_dataset_2 and 3.
4. For a challenge, complete Task 3 by using another dataset.

Attempt all of the tasks before looking at the solutions-iris.py file.

Summary

This work was supported by an EPSRC Doctoral Training Centre grant (EP/L015382/1)