BUSA3020-Week1

February 15, 2022

---

1 BUSA3020 Week 1 Lecture - Introduction

1.0.1 Unit Convenor & Lecturer

George Milunovich
george.milunovich@mq.edu.au

1.0.2 References

1. Python Machine Learning 3rd Edition by Raschka & Mirjalili - Chapter 1
   - Macquarie University Libarary Link
   - Sign in with your university email address
2. Various open-source material

1.0.3 Week 1 - Objectives

1. Introduce yourself
2. About the Unit - Course Outline and Assessment
3. Installing and Running Python
4. Learning How to open and read Jupyter Notebooks (.ipynb) with JupyterLab
5. Datasets Used & Basic Terminology
6. Running Python Code in Jupyter Notebooks
7. Mathematical Notation Used in Machine Learning
8. Different Types of Machine Learning
   - Supervised Learning
   - Unsupervised Learning
   - Reinforcement Learning

1.1 Installing Python and Running it

1.1.1 Installing Python

In BUSA3020 we will use Anaconda distribution of Python. - Go to Anaconda website https://www.anaconda.com/products/individual - Scroll down to the bottom of the page until you see "Anaconda Installers" as in the pic below

- Choose an installer for your OS, Python 3.9 and 64-Bit Installer

- Click on the saved installer and install to your machine
- Done!
- If unable to install see:
  - Step-by-step instructions https://docs.anaconda.com/anaconda/navigator/
  - Anaconda Navigator cheat sheet: [https://docs.anaconda.com/_downloads/9ee215ff15fde24bf01791d71
    Starter-Guide.pdf](https://docs.anaconda.com/_downloads/9ee215ff15fde24bf01791d719084950/Anac
    Starter-Guide.pdf)

1.1.2 Running Python

- If on Windows, click Start and type Anaconda
  - If using other type of OS follow similar instructions
- Click on Anaconda Navigator as in pic below
- Anaconda Navigator will open
- Click on "Launch" below **JupyterLab** package
- Download tutorial notes (Tutorial-Week1.ipynb) from iLearn
- In "File Explorer" on the LHS in JupyterLab navigate to the directory where you saved the lecture notes
- Select BUSA3020-Lecture-Week1.ipynb to open the file
- We can now proceed with the lecture

1.2 Machine Learning in Business Analytics - Introduction

Machine learning (ML) is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence.

Machine learning algorithms build a mathematical model based on **sample data**, also known as **training data**, in order to make predictions or decisions about **out-of-sample data** or **test data** without being explicitly programmed to do so.

Machine learning algorithms are used in a wide variety of business applications, for example:

- Predicting house prices based on house characteristics
- Deciding whether to grant a bank loan to a client
- Assessing whether or not to email a client with promotional information
- Which items will sell in a department store and how many
- Predict the appropriate level of inventory required for a retail store
- Etc.

References: https://en.wikipedia.org/wiki/Machine_learning

1.3 Some Basic Terminology, Datasets and Notation

1.3.1 Introduction to datasets we will use

We will use a number of different datasets in this unit, including the following: 1. The famous Iris dataset - https://archive.ics.uci.edu/ml/datasets/iris 2. Credit card default payments - https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients 3. House prices in Iowa (US) - will need to register for Kaggle (free) https://www.kaggle.com/c/house-prices-advanced-regression-techniques

Our data is typically organised in tables (dataframes) where - Columns represent features - also known as variables, attributes, measurements - Rows represent individual observations - also known as instances or examples

1.3.2 Iris Dataset

This is perhaps the best known database to be found in the pattern recognition literature.

Lets have a quick look at the Iris dataset in Python:

```
# python comments are made using the hash "#" symbol
# python code to read Iris data from the internet
# original data file has no column names so we assign them ourelves

import pandas as pd  # import pandas library
column_names = ['Sepal Length', 'Sepal Width', 'Petal Length', 'Petal Width', 'Class Label']  #
df = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data', na
df.to_excel('iris.xlsx')  # save for later use

print(df)
```

[ ]:

[ ]:

1.3.3 Python Counting

As we can see from the above printout, Python starts counting at 0. So we have 150 examples (observations) across rows 0 to 149 and 5 variables across columns 0 to 4, containing 4 features (explanatory variables) as well as the target variable in the last column.

1.3.4 Feature Information

1. Sepal Length in cm
2. Sepal Width in cm
3. Petal Length in cm
4. Petal Width in cm
5. Class:
    - Iris Setosa
    - Iris Versicolour
    - Iris Virginica

Predicted attribute: **class** of iris plant.

So the objective is to predict Iris class given the set of features: Sepal Length, Sepal Width, Petal Length and Petal Width.

1.3.5 Some ML Terminology

Machine learning is a branch of computer science and has its own terminology, which may be unfamiliar to students with statistics/econometrics backgrounds.

- **Feature** (x) = Predictor = Input = Independent Variable = Explanatory Variable = a column in the data matrix $\mathbf{X}$
- **Target** (y) = **Label** (in classification) = Output = Dependent Variable = Response Variable
- **Example** = Observation from a sample, i.e. a sample is a collection of examples
- **Training** = Model fitting, for parameteric models like linear regression this refers to parameter estimation

Therefore a **labeled dataset** is a dataset which contains data on the label/target ($\mathbf{y}$), i.e. Dependent Variable.

Examples:

1. Predict whether a product will sell or not, i.e. target or dependent variable takes values True (1) of False (0), on the basis of Features or Predictors such as Price, Color, Country of Production, etc.
2. Predict the price of house, target numeric value e.g. Price = \$500k, given a set of features (predictors) such as Number of Bedrooms, Size, Suburb, etc.

1.4 Mathematical Notation

In contrast to Python, our mathematical notation starts at 1, e.g. rows 1 - 150 (instead 0 to 149 as in Python). *We must keep this in mind all the time.*

So our features will be stored in a (150×4) matrix $\mathbf{X} \in \mathbb{R}^{150\times 4}$

$$\mathbf{X} = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & x_1^{(3)} & x_4^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & x_4^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(150)} & x_2^{(150)} & x_3^{(150)} & x_4^{(150)} \end{pmatrix}$$

So a typical element is $x_m^{(n)}$, where the subscript $m = 1 \dots 4$ represents columns and the superscript $n = 1 \dots 150$ represents rows. For instance, $x_2^{(4)} = 3.1$ - Sepal Width, observation 4.

We can also represent columns of $\mathbf{X}$ as column vectors. For instance, column 2 which contains Sepal Width can be written as $\mathbf{x_2} = \begin{pmatrix} x_2^{(1)} \\ x_2^{(2)} \\ \vdots \\ x_2^{(150)} \end{pmatrix} = \begin{pmatrix} 3.5 \\ 3.0 \\ \vdots \\ 3.0 \end{pmatrix}$.

Similarly, rows of $\mathbf{X}$ are row vectors. Row 3, for example, becomes $\mathbf{x^{(3)}} = \begin{pmatrix} x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & x_4^{(3)} \end{pmatrix} = \begin{pmatrix} 4.7 & 3.2 & 1.3 & 0.2 \end{pmatrix}$.

Lastly, we store target variables (class labels) in a column vector $y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(150)} \end{pmatrix} = \begin{pmatrix} \text{Iris-setosa} \\ \text{Iris-setosa} \\ \vdots \\ \text{Iris-virginica} \end{pmatrix}$.

1.5 Introduction to Different Types of ML Algorithm

Broadly, there are three distinct types of machine learning: 1. Supervised Learning 2. Unsupervised Learning 3. Reinforecement Learning

---

1.6 Supervised Learning

1.6.1 Making Predictions with Supervised Learning

Goal of Supervised Learning: learn a model from labeled training data that allows us to make predictions about unseen or future data. - **Supervised** means that we have a set of training data in which the labels (values of dependent variable) are known.

There are two tasks in supervised learning: - **Classification** - **Regression**

1.6.2 Classification

The problem of predicting the categorical class labels of new instances, based on a set of features. For example, predict whether a product will sell or not sell (True/False).

We can also distinguish between:
- Binary Classification: Classification tasks with two classes, e.g. True/False
- Multi-class Classification: Classification tasks with more than two classes, e.g. when investing in shares Buy/Sell/Hold

Example: Classification

- Predict whether a bank loan (y) will be repaid (+) or will default (-)
- Prediction made on the basis of 2 features: borrower's income (x_1) and borrower's age (x_2)

1.6.3 Regression

The task of predicting the outcome is a continuous variable, e.g. House Price, on the basis of a set of features - explanatory variables. - There are many algorithms for doing this. - We are most familiar with Linear Regression from basic statistics/econometrics courses.

Example: Regression - Use a linear regression to predict an exam mark (y) on the basis of the time spent studying (x)

1.7 Unsupervised Learning

In supervised learning we - Know what the target (dependent) variable in regression analysis (label in classification) is, e.g. exam mark - Have the values of the target (dependent) variable before we train a model, our training data contains exam marks for a sample of students

In unsupervised learning - Unlabeled data, i.e. we dont even know what the dependent variable is - Data has unknown structure which we wish to discover

1.7.1 Clustering

Clustering is a type of unsupervised learning where we attempt to group a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters) without having any prior knowledge of their group memberships.

Example: Clustering - Marketers aim to discover customer groups based on their age (x_1) and income (x_2) in order to develop targeted marketing programs.

1.8 Reinforcement Learning

The goal is to develop a system (agent) that improves its performance based on interactions (feedback) with the environment. The agent can observe the environment, perform actions and get rewards in return (or penalties - negative rewards). The agent must learn by itself the best strategy to get the most rewards over time either via an exploratory trial-and-error approach or deliberative planning. Thus, reinforcement learning is concenred with leanring to choose a series of actions that maximizes the total reward, which could be earned either immediately after taking an action of via delayed feedback.

Example: Reinforcement Learning
A chess program where the agent decides upon a series of moves depending on the state of the board (the environment) and the reward can be defined as a win or lose at the end of the game.

[ ]: