Lab Activity 4 – Support Vector Machine
Apply SVM to predict BBC news category

Bonus Points (Continuous assessment)

This activity will contribute **bonus points** to your overall grade.

You are required to submit your solution of this assignment on BOOSTCAMP, respecting the deadline given in class. Submit your own work. Cheating will not be tolerated and will be penalized.

You may work in groups of up to 4 members.

Support Vector Machines (SVMs)

Support Vector Machines (SVMs) are a class of supervised learning models widely used for classification and regression tasks.
The key idea of SVM is to find the optimal hyperplane that best separates the data points of different classes in a high-dimensional space.

For text classification, SVMs are particularly effective because text data is often represented in a high-dimensional feature space (e.g., via bag-of-words or TF-IDF vectors), and SVMs can efficiently handle such sparsity while maximizing the margin between classes.

SVMs can use **different kernels** (linear, polynomial, radial basis function, etc.) to capture both linear and non-linear relationships between features. In Natural Language Processing (NLP), SVMs are frequently applied for tasks such as sentiment analysis, spam detection, topic classification, and more.

Learning Objectives:

By the end of this activity, you will be able to:

- Understand the concept and working principle of Support Vector Machines.
- Apply text preprocessing techniques for NLP tasks using NLTK.
- Prepare text data for supervised learning by combining title and content, tokenizing, lemmatizing, and removing stop words.
- Convert textual data into numerical representations using TF-IDF vectorization.
- Train and evaluate an SVM model for text classification tasks.
- Interpret model outputs and evaluate classification performance using appropriate metrics.

Use Case 4.1: Classify wine reviews using SVM

We will revisit the Wine Review use case, where an SVM model was applied.
Refer back to this example to verify how the SVM was implemented for classifying wine reviews.
The source code for this task was provided in Lab Activity 2.2.

Use Case 4.2: BBC news category classification using SVM (Bonus Points)

BBC News dataset includes:

- **category**
  This is the **target variable** (the label).
  It tells you what type of news article it is (e.g., *business, politics, sport, tech, entertainment*).
- **filename**
  The original filename of the text document (e.g., 163.txt).
  Each file contains one news article. It's just an identifier, not useful for classification itself.
- **title**
  The headline of the news article.
  Example: *"US data sparks inflation worries"*.
- **content**
  The body text of the news article.
  Example: *"Wholesale prices in the US rose at the fastest …"*.

We will work with BBC news for 2004-2005 available at :
https://www.kaggle.com/hgultekin/bbcnewsarchive

Main Objective of this Activity:
This task involves a **supervised text classification problem**. The input is the news article text (a combination of the title and content), and the output is the corresponding **category label**.

- The title and content are merged into a single field called *raw_text* to represent the full article.
- The objective is to train a machine learning model capable of predicting the correct category based on this text.

Instructions:

Text Preprocessing with NLTK:
This procedure consists of the following points:

- Combining the title and the content of the article in one field.
- Bringing the text to lowercase, breaking it into word tokens.
- We will leave only letter words (thus removing punctuation and numbers).

- Apply lemmatization.
- Delete stop words

You have to use the NLTK library for text preprocessing.

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
```

You may use the following:

```
nltk.download('punkt') #this tokenizer divides a text into a list of sentences
nltk.download('wordnet') #it is a large word database of English Nouns, Adjectives,
Adverbs and Verbs
nltk.download('stopwords') #commonly used word
nltk.download('omw-1.4') #provides access to open wordnets in a variety of
languages, all linked to a collaborative Interlingual Index
```

SVM Modeling:

This procedure consists of the following:

- Split the data into train and test.
- Vectorize using `TfidfVectorizer() from sklearn.feature_extraction.text`
- Train a Support Vector Machine classifier on the training data.
- Evaluate on the test set.

Do not forget to add internal comments for interpretation and explanation in each step.

GOOD LUCK!