

## exercise.and.assignment.M.2

### Codebook and data files

a. = article (news, journal)	c. = cheatsheet code. = .py or .ipynb	g = graphic
howTo. = <a href="#">explanandum</a>	py.M. exercise or assignment python file	r = reading

File Name	Purpose\Description
<a href="https://github.com/cosc-526/cosc.526.home.page/blob/main/code_notebook_cosc_526.ipynb">https://github.com/cosc-526/cosc.526.home.page/blob/main/code_notebook_cosc_526.ipynb</a>	<ul style="list-style-type: none"> <li>Codebook in Jupyter Notebook</li> <li>name = code.notebook.cosc.526.ipynb</li> <li><b>save your own copy!</b></li> </ul>
	Source data

**Note.1:** The codebook is formatted differently, and below highlights expected outcomes.

**Note.2:** The instructions below are an overview with additional details in the Notebook.

**Note.3:** Perform your work in your Notebook and generate outcomes for each code block. Export the Notebook as a .pdf for submission. If issues, submit an .ipynb file at the very minimum.

### => [exercise.M.2](#)

### Problem summary

#### Objectives:

##### 1. Import and manipula

##### Mean Imputation with Pandas:

- Use the fillna() method in Pandas to replace missing values with the mean of the column: df.fillna(df.mean(), inplace=True).
- This method replaces missing values with the mean of the corresponding column, providing a simple imputation strategy.

##### K-Nearest Neighbors (KNN) Imputation with scikit-learn:

- Utilize the KNNImputer class from scikit-learn to impute missing values based on the values of the nearest neighbors: imputer = KNNImputer(n\_neighbors=5); imputed\_data = imputer.fit\_transform(data).
- This approach imputes missing values by considering the values of the k nearest neighbors in the feature space.

##### Logistic Regression with scikit-learn:

- Import the necessary modules: from sklearn.linear\_model import LogisticRegression.
- Create an instance of the LogisticRegression class: logreg = LogisticRegression().

## => assignment.M.2

### Problem summary

#### Objectives:

2. Import and manipula

### a2.Problem 0 - Import, inspect, and view descriptive statistics

Import data and view descriptive statistics with the pandas library.

- grab data from **\*\*Github\*\*** URL, .csv. or kaggle api

#### a2.Problem.1 - Description =>

The United S

Task.0 - Expected outcome:

#### a2.Problem.2 - Description =>

Which v

Task.0 - Expected outcome:

#### a2.Problem.3 - Description =>

Which cou

Task.0 - Expected outcome:

#### a2.Problem.4 - Description =>

Task.0 - Expected outcome:

#### a2.Problem.5 - Description =>

Determine e

Task.0 - Expected outcome:

#### a2.Problem.6 - Description =>

Determine



Task.0 - Expected outcome:

## Additional resources

- [Jupyter Community Forum](#)
- Jupyter Notebook [documentation](#) (including [get started](#) guides).
- Install scientific [packages](#).
- Python Package Index ([pypi](#))