CMSC 435 Assignment 2

Fall 2020
(individual work; 10 pts total)

This assignment asks you to implement and evaluate four algorithms for the imputation of missing values using two provided datasets. You will also evaluate and compare quality of the imputed values by comparing them with the corresponding values in the "complete" dataset.

<u>Datasets</u>

There are three datasets: the original dataset without missing values and two derived datasets where the missing values were introduced at two different amounts:

- *dataset_complete.csv* file is the complete dataset. It includes 9 features (including the class/predicted features) and 8795 objects.
- *dataset_missing01.csv* and *dataset_missing20.csv* files include the same dataset with 1% and 20% of missing values, respectively.

You will impute the missing values in each of the latter two datasets and compare these imputed values to the true/correct values that are available in the *dataset_complete.csv* file to evaluate and compare quality of different imputation algorithms. The three files are in the comma separated value (CSV) format. The first line in each file defines the names of features and the remaining lines include the values of the corresponding 8795 objects. The first 8 features are numeric and continuous with values in [0, 1] interval. The last, class feature is symbolic and binary with values Yes and No. The missing values are represented by ?. There are no missing values in the class feature.

<u>Algorithms for missing data imputation</u>

You will implement four algorithms for the imputation of missing values and apply each of them on the corresponding two datasets that have missing values: *dataset_missing01.csv* and *dataset_missing20.csv*.

<u>Algorithm 1. *Mean imputation*</u>

Missing value for a specific feature and object is imputed with the mean value computed using the complete values of this feature.

Example

	F1	F2	F3	Class
Object 1	0.40256	0.14970	?	No
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	0.11169	No
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	0.08910	No

To impute the missing value for feature F3 from object 1, we compute the mean of all complete values of F3: *mean* = (0.11169 + 0.13986 + 0.0891) / 3 = 0.11355.

	F1	F2	F3	Class
Object 1	0.40256	0.14970	0.11355	No
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	0.11169	No
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	0.08910	No

The imputed values **must not** be used to compute the means. This means that all missing values for a given feature are imputed with the same mean value.

	F1	F2	F3	Class
Object 1	0.40256	0.14970	0.11355	No
Object 2	0.41139	0.30140	0.11355	Yes
Object 3	0.24752	0.32148	0.11169	No
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	0.08910	No

Algorithm 2. *Conditional mean imputation*

Missing value for a specific feature and object is imputed with the mean value computed using the complete values of this feature for objects that satisfy a condition defined by the class feature. For instance, a missing value for an object 1 for which class = No is imputed based on the mean value computed using all objects for which class = No.

Example

	F1	F2	F3	Class
Object 1	0.40256	0.14970	?	**No**
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	**0.11169**	**No**
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	**0.08910**	**No**

For object 1 for which class = No, the missing value for the feature F3 is imputed as
$mean_{\text{No}}$ = (0.11169 + 0.0891) / 2 = 0.10039
For object 2 for which class = Yes, the missing value for the feature F3 is imputed as
$mean_{\text{Yes}}$ = 0.13986 / 1 = 0.13986
The two imputed values are

	F1	F2	F3	Class
Object 1	0.40256	0.14970	**0.10039**	**No**
Object 2	0.41139	0.30140	0.13986	Yes
Object 3	0.24752	0.32148	**0.11169**	**No**
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	**0.08910**	**No**

Algorithm 3. *Hot deck imputation*

Missing values for features that have missing values in a given object are imputed with the values for the same features copied from another, the most similar object. First, similarity of a given object that has missing values with every other object in the dataset is computed using the Manhattan distance. The object with the smallest distance is assumed to be the most similar and its values are used for the imputation. If that object is missing some of the values that should be imputed then the second most similar object is used to impute the **remaining** missing values, and so on. In other words, you should use the first complete value that you find by screening objects by their increasing values of the distance. Given two objects $\mathbf{x} = \{x_1, x_2, \dots x_i, \dots, x_n\}$ and $\mathbf{y} = \{y_1, y_2, \dots, y_i, \dots, y_n\}$, the Manhattan distance is calculated as $d(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{m} |x_i - y_i|}{m}$ where x_i and y_i are values of feature i for objects **x** and **y**, respectively, n is the number of features (excluding the class future), and $m \leq n$ is the number of features that do not have missing values in either of the two objects. In other words, the distance is computed using the *m* features that have complete values. Note that you **must not** use the class feature in the calculation of the distance.

Example

	F1	F2	F3	Class
Object 1	0.40256	0.14970	?	No
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	0.11169	No
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	0.08910	No

To impute missing value for feature F3 from object 1, we compute distances to every other object

$d(obj1, obj2) = (|0.40256 - 0.41139| + |0.14970 - 0.30140|)/2 = 0.0803$
$d(obj1, obj3) = (|0.40256 - 0.24752| + |0.14970 - 0.32148|)/2 = 0.1634$
$d(obj1, obj4) = |0.40256 - 0.24609|/1 = 0.1565$
$d(obj1, obj5) = |0.14970 - 0.58306|/1 = 0.4334$
Since object 2 that is the most similar to object 1 has a missing value for the feature F3, the second nearest object 4 is used and the missing value is imputed as follows

	F1	F2	F3	Class
Object 1	0.40256	0.14970	0.13986	No
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	0.11169	No
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	0.08910	No

The imputed values **must not** be used to compute the distances. In other words, all missing values for each feature are imputed based on the distances that use the dataset before the imputation. This ensures that the errors inherent in the imputed values are not propagated to compute the imputation.

Algorithm 4. *Conditional hot deck imputation*
Missing values for features with missing values in a given object are imputed with the values for the same features copied from another, most similar object that satisfies a condition defined by the class feature. For instance, a missing value for an object 1 for which class = No is imputed based on similarity only to the objects for which class = No. The calculation of the similarities follows the unconditional hot deck imputation.

Example

	F1	F2	F3	Class
Object 1	0.40256	0.14970	**?**	**No**
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	**0.11169**	**No**
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	**0.08910**	**No**

To impute missing value for feature F3 from object 1, we first compute the two distances to the objects that share the same value of the class feature
$d(obj1, obj3) = (|0.40256 - 0.24752| + |0.14970 - 0.32148|)/2 = 0.1634$
$d(obj1, obj5) = |0.14970 - 0.58306|/1 = 0.4334$
The missing value is imputed with the value for the same feature F3 from the closest object 3 as follows.

	F1	F2	F3	Class
Object 1	0.40256	0.14970	**0.11169**	**No**
Object 2	0.41139	0.30140	?	Yes
Object 3	0.24752	0.32148	**0.11169**	**No**
Object 4	0.24609	?	0.13986	Yes
Object 5	?	0.58306	**0.08910**	**No**

Like for the unconditional hot deck imputation, the imputed values **must not** be used to compute the distances.

Calculation of the imputation error

You will use the two datasets that were imputed with the four methods to calculate the corresponding eight imputation errors. You will evaluate quality of these imputations based on the Mean Absolute Error (MAE) between the imputed values and the corresponding complete values that are available in the *dataset_complete.csv* file. This dataset should be used only to calculate MAE values, not to perform the imputations. The MAE values should be used to judge and compare the quality of each imputation.

Given the imputed values $\mathbf{x} = \{x_1, x_2, \dots x_i, \dots, x_N\}$ computed from a dataset that has missing values and the corresponding complete values $\mathbf{t} = \{t_1, t_2, \dots, t_i, \dots, t_N\}$ in the complete dataset, MAE is defined as

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |x_i - t_i|$$

where N is the total number of missing values, x_i is a the imputed value in the dataset that has missing values, x_i and t_i are values for the same object and same feature in the two datasets, and $|\cdot|$ denotes the absolute value.

Example

Incomplete dataset

	F1	F2	F3	F4	Class
Object 1	0.40256	0.14970	0.16870	?	No
Object 2	0.41139	0.30140	0.47033	?	Yes
Object 3	0.24752	0.32148	0.41167	0.11169	No
Object 4	0.24609	?	?	0.13986	Yes
Object 5	?	0.58306	0.52568	0.08910	No

Dataset where values were imputed using the unconditional mean imputation

	F1	F2	F3	F4	Class
Object 1	0.40256	0.14970	0.16870	0.11355	No
Object 2	0.41139	0.30140	0.47033	0.11355	Yes
Object 3	0.24752	0.32148	0.41167	0.11169	No
Object 4	0.24609	0.33891	0.39409	0.13986	Yes
Object 5	0.32689	0.58306	0.52568	0.08910	No

Complete dataset

	F1	F2	F3	F4	Class
Object 1	0.40256	0.14970	0.16870	0	No
Object 2	0.41139	0.30140	0.47033	0.14175	Yes
Object 3	0.24752	0.32148	0.41167	0.11169	No
Object 4	0.24609	0.21359	0.24071	0.13986	Yes
Object 5	0.70541	0.58306	0.52568	0.08910	No

Given the above imputation, the MAE is calculated as follows.

$$MAE = \frac{1}{5}(|0.11355 - 0| + |0.11355 - 0.14175| + |0.33891 - 0.21359| + |0.39409 - 0.24071| + |0.32689 - 0.70541|) = 0.1598$$

The MAE values must be computed with precision of **four digits** after the decimal point.
The imputed values must be computed with precision of **five digits** after the decimal point.

Implementation

Your code must perform imputation, display the eight values of MAE on the screen and save the eight imputed datasets in the csv format. The imputed datasets should be named as follows:
Vnumber_missing01_imputed_mean.csv
Vnumber_missing01_imputed_mean_conditional.csv
Vnumber_missing01_imputed_hd.csv
Vnumber_missing01_imputed_hd_conditional.csv
Vnumber_ missing20_imputed_mean.csv
Vnumber_missing20_imputed_mean_conditional.csv
Vnumber_missing20_imputed_hd.csv
Vnumber_missing20_imputed_hd_conditional.csv
where *Vnumber* is your V number, e.g., *V12345678_missing01_imputed_mean.csv*

The MAE values should be displayed on the screen in the following format

```
MAE_01_mean = 0.1234
MAE_01_mean_conditional = 0.5678
MAE_01_hd = 0.1234
MAE_01_hd_conditional = 0.5678
MAE_20_mean = 0.1234
MAE_20_mean_conditional = 0.5678
MAE_20_hd = 0.1234
MAE_20_hd_conditional = 0.5678
```

You must use Java or Python 3 to implement all computations including loading the datasets from the csv files, coding the four imputation methods, calculation of the MAE values, printing the MAE values on the screen, and saving of the eight imputed datasets.

If you use Python 3:

- Your code should be in one source code (.py) file. You may define any number of classes and functions, but everything must be in that file.
- You are only allowed to use NumPy (https://www.numpy.org/) and pandas (https://pandas.pydata.org) as imported libraries. They may help you with accessing the csv files and working with the data.
- Your program will be tested on **Python 3.8.5** with **NumPy1.19.0** and **pandas 1.1.1** installed (Which are the latest versions).
- This python file must successfully run on the above python environment and produce the above-mentioned outputs with the required precision. It should be run by executing the below command in the location where three input csv files are located.

  ```
  Python3 a2.py
  ```

  For this, you need to make sure you read the input csv files from the current working directory.

If you use Java:

- All your code should be in one source code (.java) file. You may define any number of classes and functions, but everything must be in that file.
- Your program will be tested on **Java 14** (which is the latest version).
- This java file must successfully compile and run with above version of Java and produce the above-mentioned outputs with the required precision. It should be run by executing the below command in the location where three input csv files are located.

  ```
  javac a2.java
  java a2
  ```

  For this, you need to make sure your main class is named `a2,` you don't declare a package in `a2.java`, and you read the input csv files from the current working directory.

Deliverables

1. Java or Python source code in a single .java or .py file. The file **must be named** a2.java or a2.py.
2. A pdf document (**named** with your last name, e.g., Smith.pdf) with answers to the following four questions

   2a. What are the MAE values for the eight results? You should copy the output from the screen.

   2b. Which of the considered four imputation methods is the most accurate for the *dataset_missing01.csv* dataset? Briefly explain **why** this method is better than the other three methods.

2c. Are the results for the same algorithm on the two datasets the same/worse/better (e.g. mean on *dataset_missing01* vs. mean on *dataset_missing20*)? Briefly explain **why**.

2d. Which of the two unconditional methods (mean vs. hot deck) is faster, i.e., requires fewer computations? Briefly explain **why**. Give computational complexity of both methods as a function of the number of objects *n*, and use it to support your explanation.

Notes

– Include your **name, class number and title** at the top of the pdf file. Use separate and **clearly marked and numbered paragraphs** for answers to each of the four questions. We will deduct points if you do not follow these instructions.

– Do not procrastinate and start early – this assignment requires a substantial amount of time and effort. Late submissions will be subject to deductions: 15% in first 12 hours and 30% for between 12 and 48 hours. We will not accept submissions that are over 48 hours late.

– We will check your source code, verify if it runs correctly, validate the results on the screen and in the files, and mark the answers to the four questions.

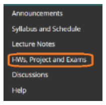
– We will **deduct** points if the files names and/or the outputs on the screen do not follow the above-defined format.

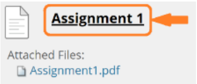
– We will check for **plagiarism**. Write your own code and provide your own answers.

Due Date

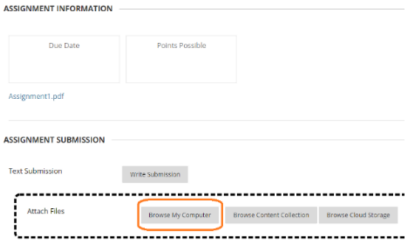
Your assignment must be received by 12:30 pm Eastern Time (beginning of the lecture) on September 22 (Tuesday), 2020. Submission, which include the two files, must be done via the class web page in Blackboard. 4-step instructions are included below.

1. Go to the section “HWs, Projects and Exams” and select Assignment 2 by clicking on the assignment title

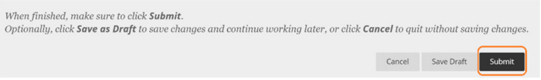




2. Select your submission files by clicking on “Browse My Computer”.

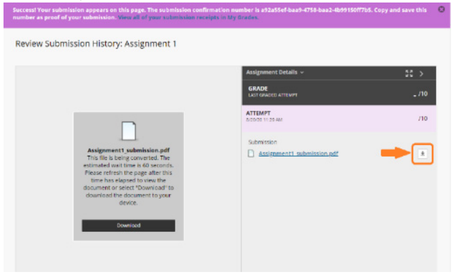


3. [Important!] Your file(s) will be submitted only after you click the submit button.



*When finished, make sure to click **Submit**.*
*Optionally, click **Save as Draft** to save changes and continue working later, or click **Cancel** to quit without saving changes.*

4. [Important!] Make sure you download your submission and double check if the correct files are sent.



Success! Your submission appears on this page. The submission confirmation number is a92a55ef-baa9-4758-baa2-4b99150ff7b5. Copy and save this number as proof of your submission. View all of your submission receipts in My Grades.

Review Submission History: Assignment 1

Assignment1_submission.pdf
This file is being converted. The estimated wait time is 60 seconds. Please refresh the page after this time has elapsed to view the document or select "Download" to download the document to your device.