CSE 523 ML Project

Faculty mentor:

Prof. Mehul Raval

Project Number: 9

Project Title: Data-driven imputation scheme for human-subject-based dataset

Group name: Syntellect

Group member details:

ENROLLMENT NUMBER	**NAME**
Kashish Jethmalani	AU2140029
Srushti Thakar	AU2140117
Priyal Patel	AU2140204
Riya Patel	AU2140214

Progress report : (Week 4)

Date: 01-03-2024 to 7-03-2024

We had an online meeting with Srishti Sharma on Wednesday (06/03/24). She guided us on preprocessing the data, followed by some basic imputation techniques like local and global imputation and k means clustering. Further, we discussed how to measure the efficiency of our algorithm using standard measures such as percentage bias and raw bias.

Local imputation (using mean as imputation technique) : This method replaces missing values in a particular athlete's data with the mean value of that specific feature (e.g., age, height, weight) across all athletes.

Global imputation: Global imputation involves filling in missing values using statistical measures(mean) calculated across the entire dataset. Unlike local imputation, global imputation considers the dataset as a whole rather than individual instances.

Preprocessing the data

We followed the below steps to merge the data from different days into a structured format.

Step 1: Convert Vertical Jump Season 2.xlsx (multiple sheets) into a format similar to Vertical Jump Season 3.csv

Step 2: Merge the Vertical Jump Season 2.csv into Season 2 with the Polar.csv file using "date" as the merge key - This will now be a single CSV file with all modalities of data for each athlete date-wise.

Step 3: Repeat Step 2 for Season 3 data

Challenges faced while preprocessing the data

Firstly, we sorted vertical jump season 2.xlsx and removed unnecessary columns. Merging the data using the date as a standard column was challenging due to the data size. We tried implementing Python code to merge two sheets, but it didn't work.

Finally, we manually merged the Vertical Jump Season 2.csv and the Polar.csv files. It took almost 2 - 3 hours to preprocess the entire dataset.

Exploring the concept of K-means clustering:

K-means clustering is a popular unsupervised machine learning algorithm for partitioning a dataset into a predetermined number of clusters. The algorithm aims to group similar data points and discover underlying patterns.

1. Initialization: Choose K initial centroids randomly from the dataset, where K is the number of clusters.

2. Assignment: Assign each data point to the nearest centroid, forming K clusters.

3. Update: Recalculate the centroids as the mean of all data points assigned to each cluster.

4. Repeat steps 2 and 3 until convergence, where the centroids no longer change significantly or a maximum number of iterations is reached.

K-means clustering is used in large dataset imputation by following these steps:

1. Identify missing values: Determine which features are missing in your dataset.

2. Cluster the data: Apply k-means clustering to the dataset, excluding the features with missing values. This partitions the data into clusters based on the available features.

3. Assign missing values: For each missing value, replace it with the centroid of the cluster to which the data point belongs. This centroid serves as a representative value for the cluster.

4. Repeat if necessary: If there are still missing values after the initial imputation, repeat the process or use other imputation methods for the remaining missing values.

By leveraging the clustering structure of the data, k-means clustering helps in imputing missing values by providing reasonable estimates based on the characteristics of the data points within the same cluster. However, it's essential to note that k-means clustering has limitations, such as sensitivity to initial centroid selection and difficulties in handling non-linear data or clusters of varying sizes and densities.

Understanding standard measures such as percentage bias and raw bias:

In the context of evaluating a machine learning data value imputation model, measures like percentage bias and raw bias are used to assess the performance of the imputation process by comparing the imputed values to the true (or observed) values. Let's break down each measure:

1. Raw Bias:
   - Raw bias quantifies the average difference between the imputed values and the true values across the dataset.
   - It is calculated as the mean of the differences between the imputed values and the true values.

- Mathematically, it can be expressed as:

$$\text{Raw Bias} = \frac{1}{n}\sum_{i=1}^{n}(I_i - T_i)$$

where:

- I_i is the imputed value for observation i,
- T_i is the true value for observation i,
- n is the total number of observations.

2. Percentage Bias:

- Percentage bias measures the relative difference between the imputed and true values, expressed as a percentage of the true values.

- It is calculated as the mean of the percentage differences between the imputed and true values.

- Mathematically, it can be expressed as:

$$\text{Percentage Bias} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{I_i - T_i}{T_i} \times 100\right)$$

where:

- I_i is the imputed value for observation i,
- T_i is the true value for observation i,
- n is the total number of observations.

Both raw bias and percentage bias provide insights into the accuracy and performance of the imputation model:

- Raw Bias: It gives the average discrepancy between imputed and true values, providing a sense of the direction and magnitude of the errors in the imputation.

- Percentage Bias: It provides a relative measure of the errors, considering the scale of the actual values. This can be particularly useful when dealing with variables of different scales.

These measures can help evaluate the effectiveness of an imputation model and compare different imputation methods to choose the one that performs best for a given dataset and context.