# Resolving Racial Health Disparities by Applying Statistics on Complex, Multidimensional Datasets

Dipro Ray, Sijia Huo, Liudmila Sergeevna Mainzer, Zeynep Madak-Erdogan

National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign

## BACKGROUND AND SIGNIFICANCE

Battling sophisticated socio-economic issues requires analyses of complex, multi-dimensional datasets composed of heterogeneous variables - binary, categorical, continuous etc. Public health issues, in particular, are complicated to deal with because of the difficulty in amassing large-scale data due to privacy issues - identifying information from health datasets need to be redacted, in compliance with the law - and access issues - datasets from medical institutions are in many different formats, and standardizing them or accumulating many such datasets is hard.

One such dataset is of breast cancer mortality for women in the U.S., which needs to be analyzed to find biological and socioeconomic factors responsible for racial health disparities. Breast cancer mortality rates for African-American women in the U.S. is 40% higher than for Caucasian women in the U.S., despite the same incidence rate [1][2]. Preliminary data reveals a 4 to 5 fold greater risk for tumors that express the estrogen receptor (ER), strongly suggesting that biological mechanisms operate in ER positive breasts tumors in black women that arm those tumors with a more aggressive phenotype [3].

The objective of the dataset's analysis is to cross-correlate the multidimensional dataset of blood metabolite measurements in a racially diverse population of women, to find factors that contribute to breast cancer risk disparities [4]. This would enable clinical translation in terms of targeting novel biomarkers and pathways, and facilitate developing biosensor-based companion diagnostic tools for early detection and individualized treatment.

## PROJECT GOALS

To combat the above problem of analyzing heterogeneous datasets, a statistical pipeline has been developed to harmonize data across various cohorts. The code in place analyzes numerical and discrete data, and applies statistical data standardization techniques commonly used on multidimensional datasets, like re-normalization, covariates identification and dimensionality reduction. It also accounts for missing attribute values in the dataset based on the user's preference. However, the pipeline, or workflow, that exists is not optimized - it has to be run line by line - and the written code is not streamlined, making it hard to use and unadaptive to the user's requirements.

The goals of the project are:

- To code an R package that implements the data harmonization pipeline in a flexible and adaptive manner allowing the user to choose steps of the workflow.
- To scale, automate and containerize the package so that it can be deployed through the cloud (like AWS)
- To publish the package on CRAN and make it open-source through Github.

The end result is envisioned as a fully automated, resource-efficient, open-source software useful for scientists to perform computations on public health data in real-time.

## THE STATISTICAL PIPELINE

The developed pipeline distinguishes itself from other similar purpose R packages in three ways:

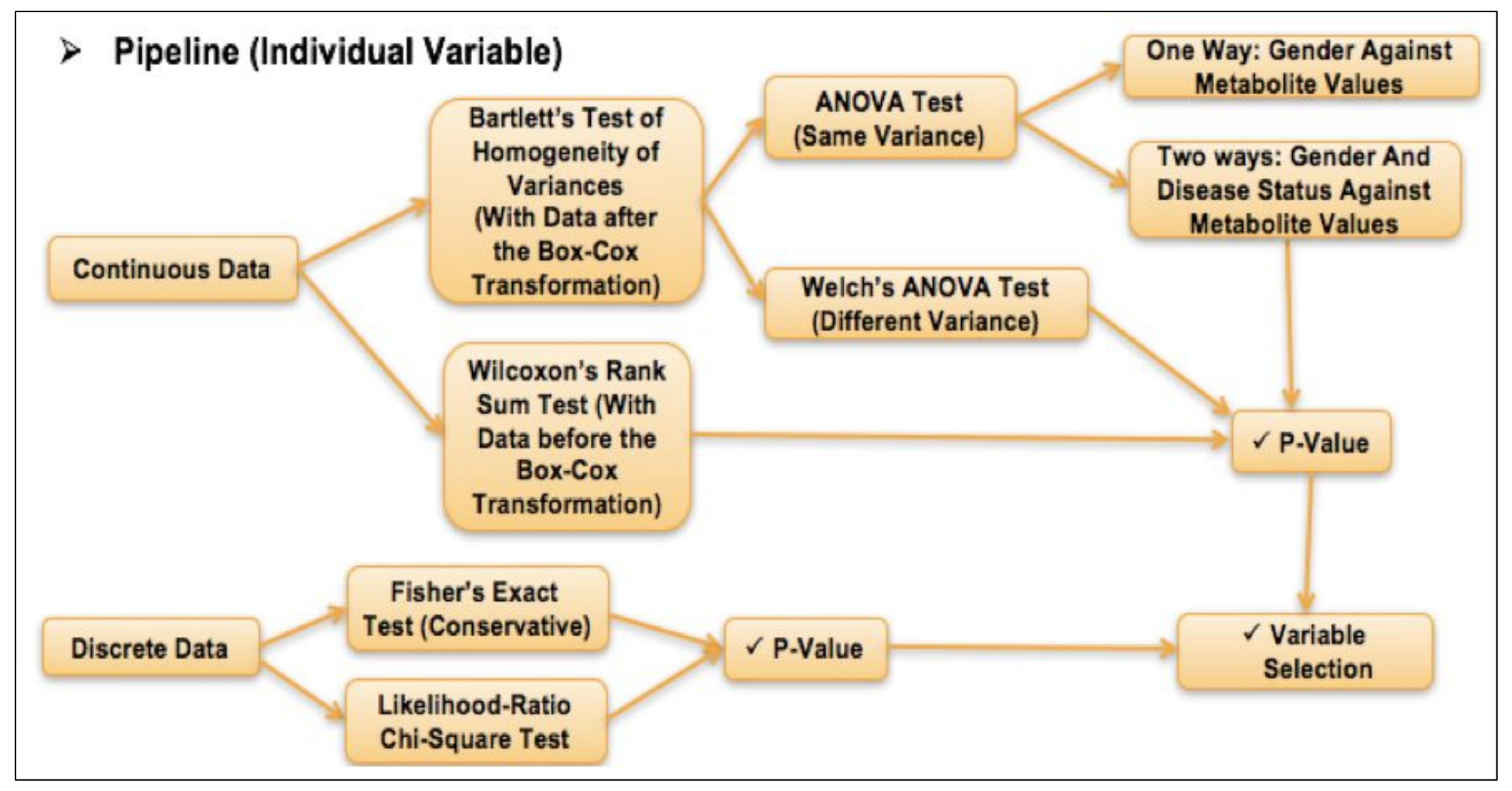
- Provides procedures to analyze both continuous (numerical) data types and discrete (nominal) data types.
- Adds functionality to deal with presence of significant missing data, by allowing the user to set a threshold for missing data percentage per variable
- Designed for scalability, keeping in mind that the input is a very large heterogeneous dataset

**Data Input, Setup, Normalization and Imputation**

After creating an R dataframe from the input data CSV, the data is divided into two sub-datasets: one for continuous variables, and the other for discrete variables, so that they can be processed separately. Since most statistical tests require variables to be normally distributed, we must check for normality (in reality, continuous data is often not normal). In our dataset, among 43 continuous variables, only 2 fail to reject the null hypothesis of normal distribution at a significance level of 0.05. The pipeline then transforms these variables into a normal distribution using the Box-Cox Transformation. We also standardize the data for easy processing later on. To deal with the problem of significant missing data, we filter out variables based on a threshold and imputate the dataset.

**Data Clustering and Analysis**

We compute the correlation between each variable to figure out which variables are highly correlated (so that we can reduce the size of the dataset to analyze). Since the multicollinearity in our dataset is high, we perform principal components analysis - a dimensionality reduction tool to reduce the dataset to a smaller set of less correlated variables retaining as much information as possible.


➢ Pipeline (Individual Variable)

Continuous Data

Bartlett's Test of Homogeneity of Variances (With Data after the Box-Cox Transformation)

ANOVA Test (Same Variance)

One Way: Gender Against Metabolite Values

Two ways: Gender And Disease Status Against Metabolite Values

Welch's ANOVA Test (Different Variance)

Wilcoxon's Rank Sum Test (With Data before the Box-Cox Transformation)

✓ P-Value

Discrete Data

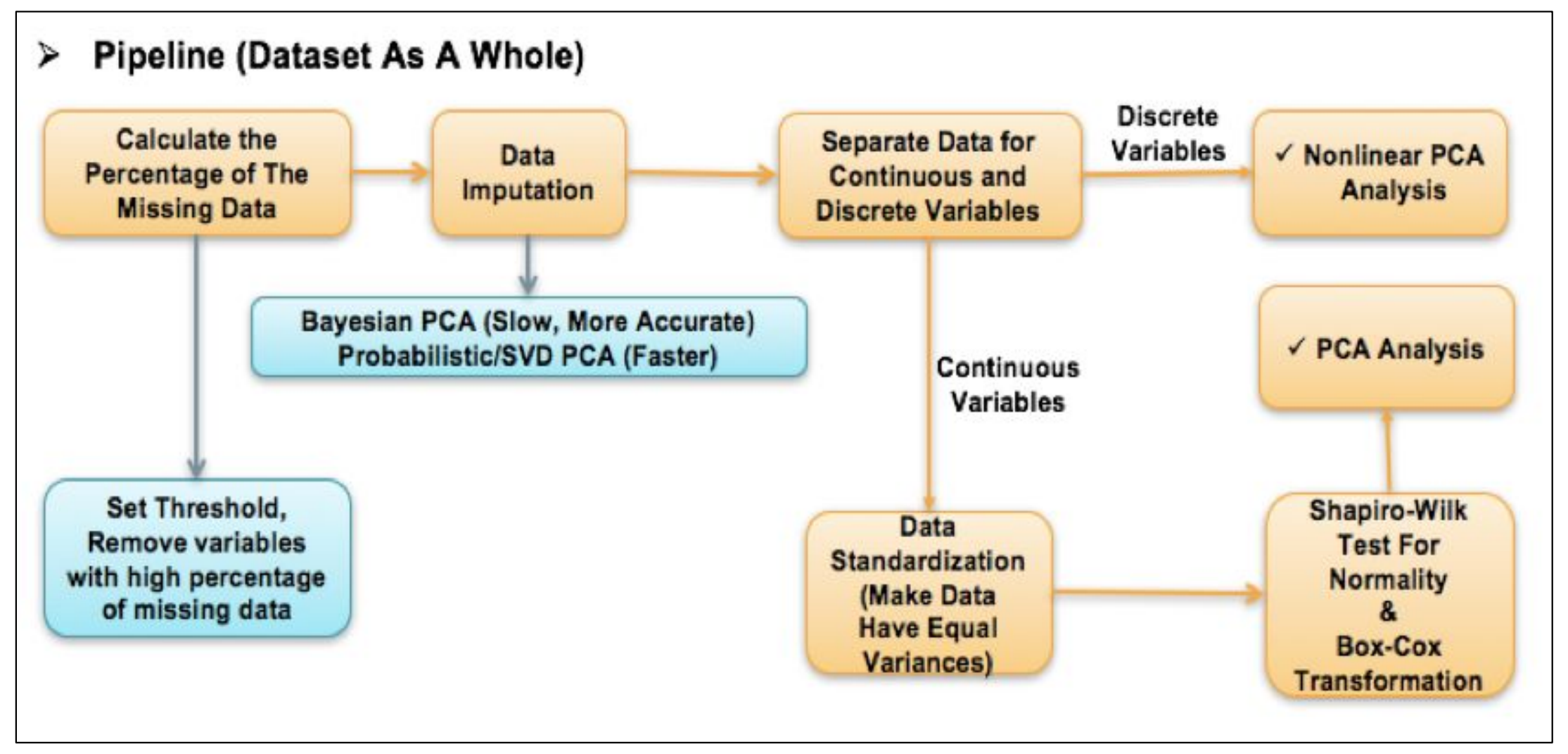
Fisher's Exact Test (Conservative)

Likelihood-Ratio Chi-Square Test

✓ P-Value

✓ Variable Selection

*Statistical Pipeline for an Individual Variable*


➢ Pipeline (Dataset As A Whole)

Calculate the Percentage of The Missing Data

Data Imputation

Separate Data for Continuous and Discrete Variables

Discrete Variables

✓ Nonlinear PCA Analysis

Bayesian PCA (Slow, More Accurate) Probabilistic/SVD PCA (Faster)

Continuous Variables

✓ PCA Analysis

Set Threshold, Remove variables with high percentage of missing data

Data Standardization (Make Data Have Equal Variances)

Shapiro-Wilk Test For Normality & Box-Cox Transformation

*Statistical Pipeline for the Whole Dataset*

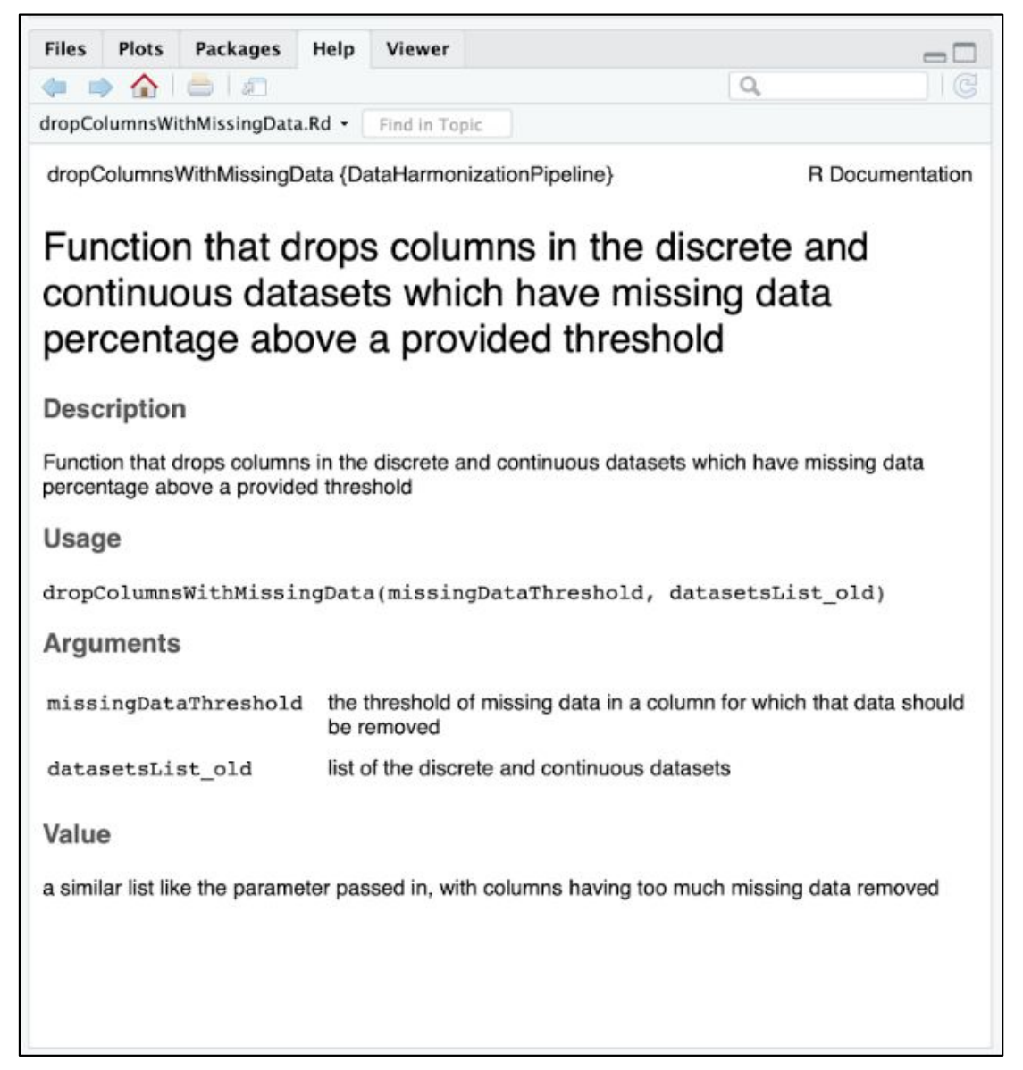
## THE R CODE PACKAGE

(https://github.com/ncsa/DataHarmonizationPipeline)

The design of the code was motivated by software needs.

Most importantly, the code is designed to be **modular**. Each step/block of the pipeline exists as separate R functions, with similar input and output datasets, making it easy to modify the pipeline by calling different functions in a different order from the "driver" file, appropriately for each analysis. Also, a user could, then, easily choose which steps of the pipeline they would want to be implemented.

The code is also written to be very **readable** and easily understandable to scientists who wish to use it. The written code and syntax is as per standard R conventions, and the in-code documentation through comments is capable of being viewed separately through Roxygen2 (a documentation system for R).



dropColumnsWithMissingData {DataHarmonizationPipeline} R Documentation

Function that drops columns in the discrete and continuous datasets which have missing data percentage above a provided threshold

Description

Function that drops columns in the discrete and continuous datasets which have missing data percentage above a provided threshold

Usage

```
dropColumnsWithMissingData(missingDataThreshold, datasetsList_old)
```

Arguments

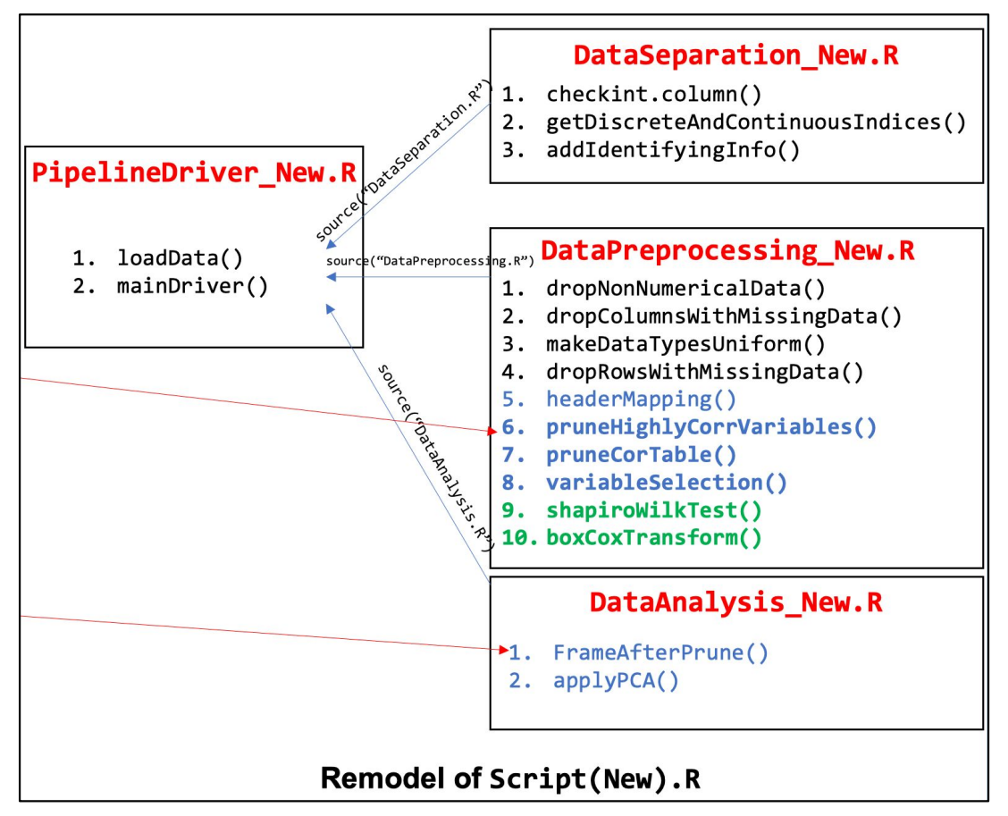
| | |
|---|---|
| missingDataThreshold | the threshold of missing data in a column for which that data should be removed |
| datasetsList_old | list of the discrete and continuous datasets |

Value

a similar list like the parameter passed in, with columns having too much missing data removed

*An Example of Roxygen2 based Documentation.*

*The R Markdown (.Rd) files for documentation are automatically generated by the package using the documentation within the source code files.*


PipelineDriver_New.R

1. loadData()
2. mainDriver()

source("DataSeparation.R")

DataSeparation_New.R

1. checkint.column()
2. getDiscreteAndContinuousIndices()
3. addIdentifyingInfo()

source("DataPreprocessing.R")

DataPreprocessing_New.R

1. dropNonNumericalData()
2. dropColumnsWithMissingData()
3. makeDataTypesUniform()
4. dropRowsWithMissingData()
5. headerMapping()
6. pruneHighlyCorrVariables()
7. pruneCorTable()
8. variableSelection()
9. shapiroWilkTest()
10. boxCoxTransform()

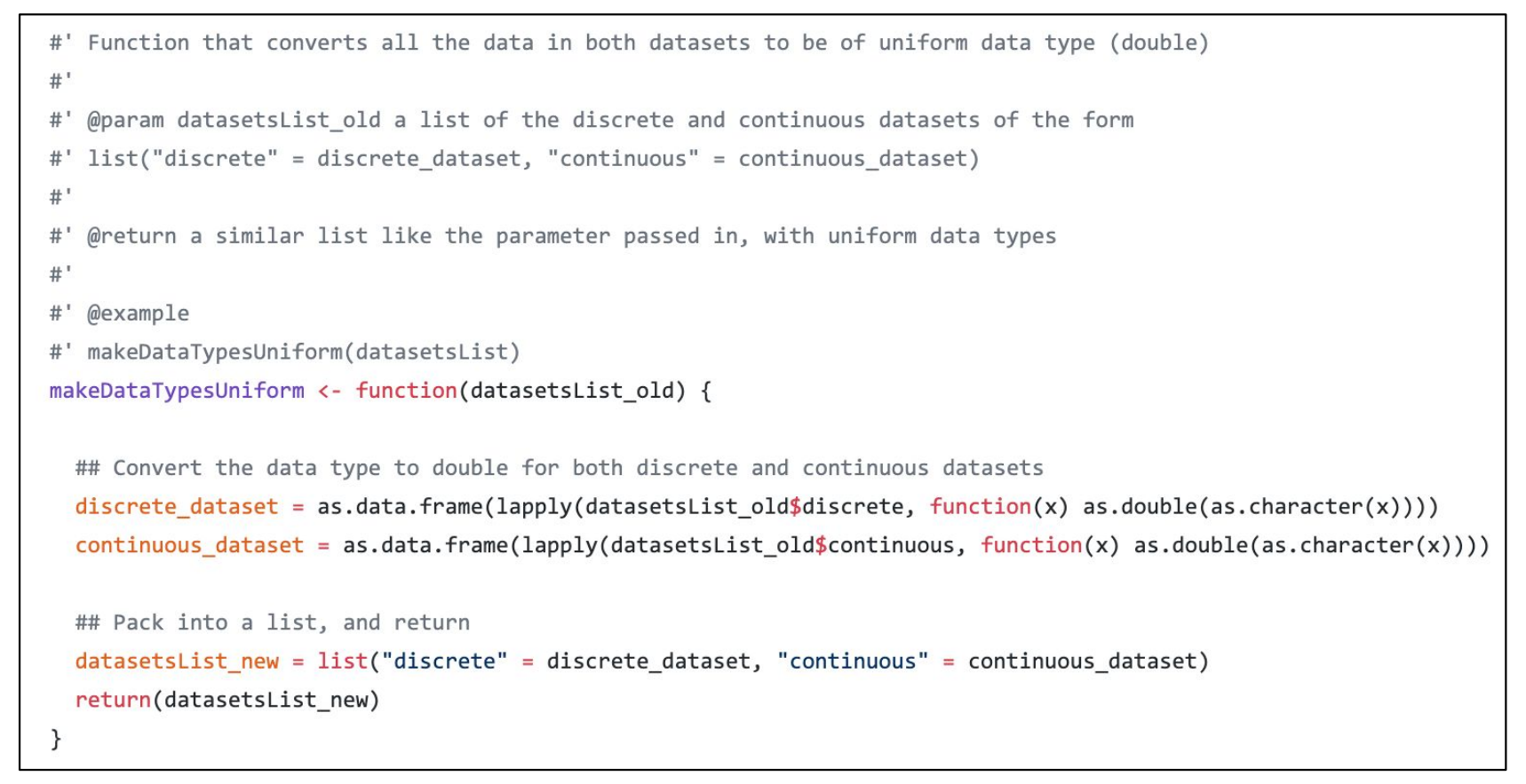
source("DataAnalysis.R")

DataAnalysis_New.R

1. FrameAfterPrune()
2. applyPCA()

Remodel of Script(New).R

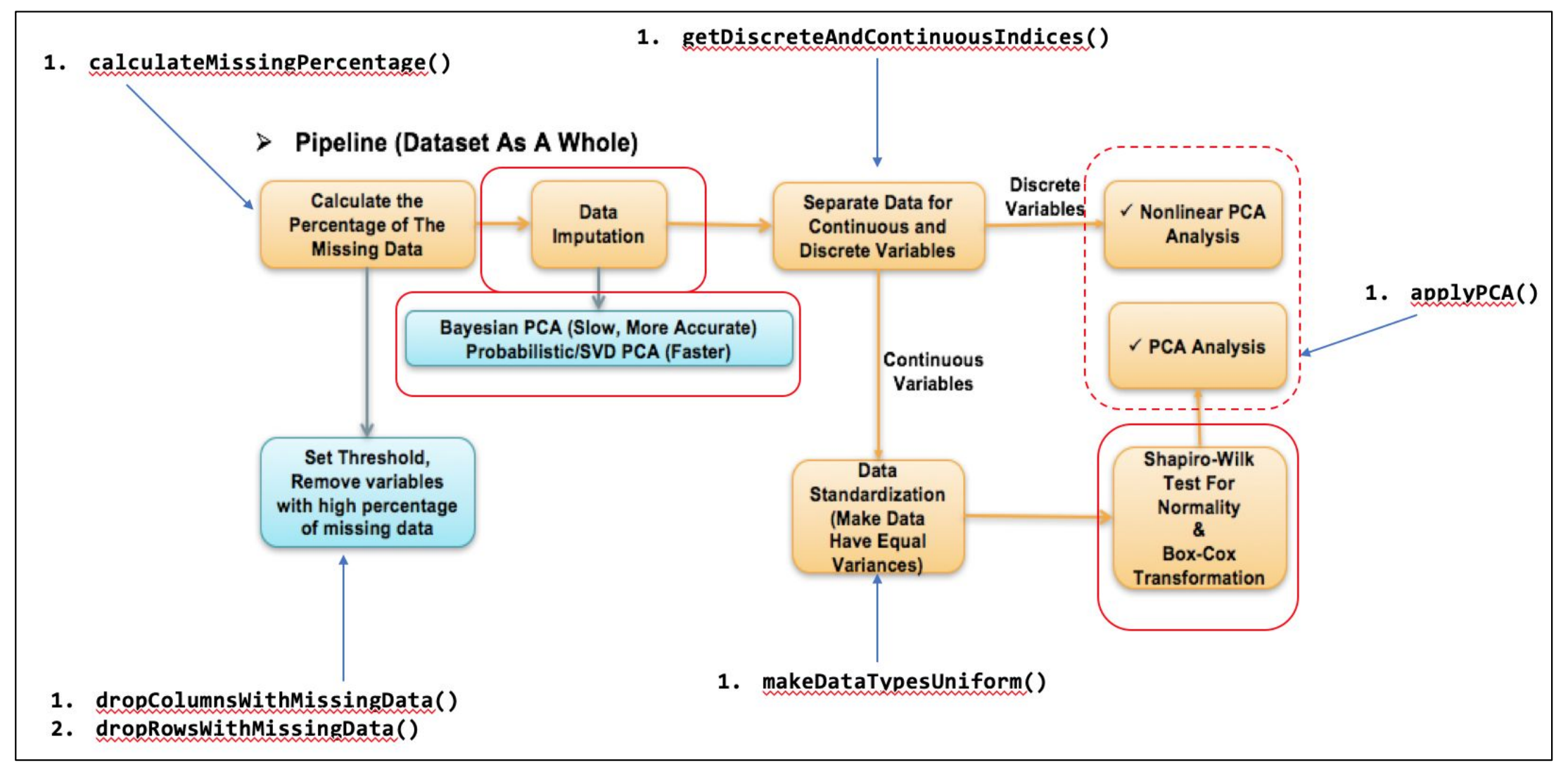
*Categorization of Functions into Files based on Functionality.*


```
#' Function that converts all the data in both datasets to be of uniform data type (double)
#'
#' @param datasetsList_old a list of the discrete and continuous datasets of the form
#' list("discrete" = discrete_dataset, "continuous" = continuous_dataset)
#'
#' @return a similar list like the parameter passed in, with uniform data types
#'
#' @example
#' makeDataTypesUniform(datasetsList)
makeDataTypesUniform <- function(datasetsList_old) {

  ## Convert the data type to double for both discrete and continuous datasets
  discrete_dataset = as.data.frame(lapply(datasetsList_old$discrete, function(x) as.double(as.character(x))))
  continuous_dataset = as.data.frame(lapply(datasetsList_old$continuous, function(x) as.double(as.character(x))))

  ## Pack into a list, and return
  datasetsList_new = list("discrete" = discrete_dataset, "continuous" = continuous_dataset)
  return(datasetsList_new)
}
```

*A Code Excerpt.*


1. calculateMissingPercentage()

1. getDiscreteAndContinuousIndices()

➢ Pipeline (Dataset As A Whole)

Calculate the Percentage of The Missing Data

Data Imputation

Separate Data for Continuous and Discrete Variables

Discrete Variables

✓ Nonlinear PCA Analysis

1. applyPCA()

Bayesian PCA (Slow, More Accurate) Probabilistic/SVD PCA (Faster)

Continuous Variables

✓ PCA Analysis

Set Threshold, Remove variables with high percentage of missing data

Data Standardization (Make Data Have Equal Variances)

Shapiro-Wilk Test For Normality & Box-Cox Transformation

1. makeDataTypesUniform()

1. dropColumnsWithMissingData()
2. dropRowsWithMissingData()

*Breakdown of Pipeline into Functions, to support Modularity*

## FURTHER WORK

**Package Github Documentation**

Along with the package's documentation, extensive documentation has been added to the Github repository to make the package easily understandable for users. The README.md file lists installation instructions and library dependencies and the more_info.md file lists the purpose of each file and the reasons for splitting up the functions in which they have been split up.

**Scalability**

To make the package scalable for operation on very large datasets, a copy of the package in Apache SparkR has been made, making use of SparkR's many parallelized functions.

## FUTURE DIRECTIONS

With the pipeline in place, modified for scalability, the next step would be to containerize the package (through Docker), so that the code is system independent, and deploy it through the cloud, for ease of install and run for the user.

Since the package is designed based on and tested on the breast cancer mortality dataset, it has to be tested on more datasets to ensure accuracy. Also, an input file standard has to be developed for the package to ensure the software is generic enough for any health dataset.

Since the pipeline can now handle large datasets, it is possible to add more features within the analysis. In relation to our dataset, geospatial data is one such important data point that can be added. We are testing how we can best split areas into geographical sections (as opposed to zipcodes) to incorporate it within our datasets.

## ACKNOWLEDGEMENTS

I would like to express my gratitude to NCSA for giving me the opportunity to work on this project, as a part of the NCSA Genomics group, through the NCSA Students Pushing Innovation (SPIN) internship program. Many thanks to my mentors Prof. Mainzer and Prof. Madak-Erdogan for helping me immensely over the course of a year for the development of the package. And thank you to Sijia Huo, a previous SPIN Intern on the same project, who helped develop much of the statistical pipeline and starter code base.

## REFERENCES

[1] DeSantis CE, Fedewa SA, Goding Sauer A, Kramer JL, Smith RA, Je- mal A. Breast cancer statistics, 2015: Convergence of incidence rates between black and white women.. CA Cancer J Clin. 2016; 66(1):31-42. doi: 10.3322/caac.21320. PubMed PMID: 26513636

[2] Whitman S, Orsi J, Hurlbert M. The racial disparity in breast cancer mortality in the 25 largest cities in the United States.. Cancer Epidemiol. 2012;36(2):e147-51. doi: 10.1016/j.canep.2011.10.012. PubMed PMID: 22443886.

[3] Kish JK, Yu M, Percy-Laurry A, Altekruse SF. Racial and ethnic disparities in cancer survival by neighborhood socioeconomic status in Surveillance, Epidemiology, and End Results (SEER) Registries. J Natl Cancer Inst Monogr. 2014;2014(49):236-43. doi: 10.1093/jncimonographs/lgu020.PubMed PMID: 25417237; PMCID: PMC4841168.

[4] Claudia R. Baquet M.D., M.P.H. Patricia Commiskey M.A. Socioeconomic factors and breast carcinoma in multicultural women. American Cancer Society. https://doi.org/10.1002/(SICI)1097-0142(20000301)88:5