Supplementary Information

Resampled dimensional reduction for feature representation in machine learning

Herdiantri Sufriyana, MD;[a,b] Yu-Wei Wu, PhD;[a,c] Emily Chia-Yu Su, PhD;[a,c,d,*]

[a] Graduate Institute of Biomedical Informatics, College of Medical Science and Technology, Taipei Medical University, 250 Wu-Xing Street, Taipei 11031, Taiwan.
[b] Department of Medical Physiology, Faculty of Medicine, Universitas Nahdlatul Ulama Surabaya, 57 Raya Jemursari Road, Surabaya 60237, Indonesia.
[c] Clinical Big Data Research Center, Taipei Medical University Hospital, 250 Wu-Xing Street, Taipei 11031, Taiwan.
[d] Research Center for Artificial Intelligence in Medicine, Taipei Medical University, 250 Wu-Xing Street, Taipei 11031, Taiwan.
* Corresponding author at: Clinical Big Data Research Center, Taipei Medical University Hospital, 250 Wu-Xing Street, Taipei 11031, Taiwan. Phone: +886-2-66382736 ext. 1515. Email address: emilysu@tmu.edu.tw

Introduction

In this Supplementary Information, we describe details on this study following chronological order of our analysis pipeline on data representation into new dimensions by resampling. We would use prelabor rupture of membranes (PROM) with 10-fold cross-validated principal components (PCs), as an example. There are three of six sections corresponding to some sections in the main text, which are respectively Introduction, Software and equipment, and Procedure. Along with this PDF document, we also provide R Markdown (.Rmd) containing the same texts with this document but including the programming codes for the data analysis in-between of these texts. The R Markdown are available in https://github.com/herdiantrisufriyana/resdimer. To get raw data, one need to request an access from the BPJS Kesehatan for their sample dataset published in August 2019. Up to this date, there are three sample datasets they published in February 2019, August 2019, and December 2020. For the first and second versions, a request is applied via https://e-ppid.bpjs-kesehatan.go.id/, while the third is applied via https://data.bpjs-kesehatan.go.id. To preprocess

the raw data into the input dataset of this protocol, follow the codes of the R Markdown in https://github.com/herdiantrisufriyana/medhist/tree/main/preprocessing.

Software and equipment

Programming environment

We set up a programming environment for this study. Bioconductor was utilized as described in the main text. There were 123 R packages which are 8 base packages, 29 other packages, and 109 dependencies.

Procedure

Step 1

The data source was a sample dataset of the whole health insurance database during 2015 and 2016 by cross-sectional design. Stratified random sampling was applied. The strata variable was constructed from 66,072 combinations of all the healthcare facilities (*n*=22,024) and category of family, which were: (1) a family of which members never visit the healthcare facilities; (2) a family of which members have visited only primary care; and (3) a family of which members have visited all levels of care. For each stratum, one to ten families were randomly included. This means only 10 families were randomly included if more than that number, resulting 586,969 families with 1,697,452 subjects.

We conducted non-essential data cleaning, e.g. revising the inconsistent name of states, estimating the healthcare identifiers, *et cetera*. These procedures were parts of our R package of medhist 0.1.0. No sampling was conducted.

After the non-essential data cleaning, we applied retrospective cohort design, as described in the main text. For pregnant women, we use several codes for determining delivery or immediately after delivery care. The 220 codes are described.

We conducted data preprocessing after defining the target population and sampling it retrospectively. Demographics were included as categorical variables for causal factor we used as variable of interests. Then, We computed a number of days for a code, or any code representing a causal factor, in the latest encounter before each visit.

To ensure PCs defined by derivation set only, we need to conduct data partition before continuing the downstream analysis. Therefore, PCs were not derived by involving validation set. Deriving PCs is relatively time-consuming. If one eventually uses this method for predictive modeling, it is efficient to conduct any filtering of predictors and data transformation before derivation of PCs. Therefore, more time can be saved and one may not need to run expensive computation.

All candidate predictors, including non-demographical causal factors, have non-zero variances. There were 460 candidate predictors fulfilling this criterion. We also showed in the same table that there are 426 candidate predictors without perfect separation.

We excluded the diagnosis/procedure codes that may leak the outcome information. We only used the existing codes in the training set to determine outcome-leaker codes based on the previous codes for determining delivery or immediately after delivery care. There were 54 codes that may leak the outcome. All of them were also irredundant.

We also determined causal factors as the candidate predictors. These can be an example how to conduct the data transformation on a variable represented by multiple codes of diagnosis and procedure. We combined these factors with other variables that assign a single code.

We inferred the nationwide historical rates given the day number from a code encounter to current visit for each candidate predictor, as described in the main text. This used irredundant candidate predictors with non-zero variances and no perfect separation in training set only.

Step 2 to 4

The historical rates of all candidate predictors were fitted to a principal component (PC) model. Only derivation set was used for the model fitting. We applied 10-fold cross validation to estimate weights for all candidate predictors in each PC.

Step 5 to 6

Variable-wise average and standard deviation and weights of the transformation was estimated for those at population level by averaging values of the respective metrics. These values were inferred from training set only. Then, for either training or validation set, data were standardized and transformed into PCs.

Step 7

We computed the existing number of events per variable (EPV) before being transformed into PCs. Using original predictors, we only get 40 EPV. However, if 50 EPV is needed, then only 75 PC is warranted from 372. The PCs were sorted by proportion of variance explained (PVE) from the highest to the lowest. Eventually, we only selected top PCs in any subset. In this example, we demonstrated selecting PCs in training set.