Depuración y calidad de datos en R (II)

SISTEMAS INTELIGENTES PARA LA GESTIÓN DE LA EMPRESA

CURSO 2016-2017

Documentación adicional

Missing values https://www.rstudio.com/rviews/2016/11/30/missing-values-data-science-and-r/

- MICE https://www.r-bloggers.com/imputing-missing-data-with-r-mice-package/
- Amelia https://www.rstudio.com/rviews/2016/11/30/missing-values-data-science-and-r/
- Simputation https://cran.r-project.org/web/packages/simputation/vignettes/intro.html

Tratamiento de ruido

- NoiseFiltersR https://cran.r-project.org/web/packages/NoiseFiltersR/index.html

Predicción

- Caret http://topepo.github.io/caret/

MICE

Multivariate Imputaton by Chained Equations

- Software para imputar valores perdidos o incompletos basado en FCS (*fully conditional specification*)
  - Sustituir valores perdidos por ”valores más probables”, estimados mediante inferencia a partir del resto del dataset

Facilita:

- Crear modelos predictivos para imputar valores perdidos
- Utilizar datasets con diferentes tipos de imputaciones para el análisis (*pooling*)

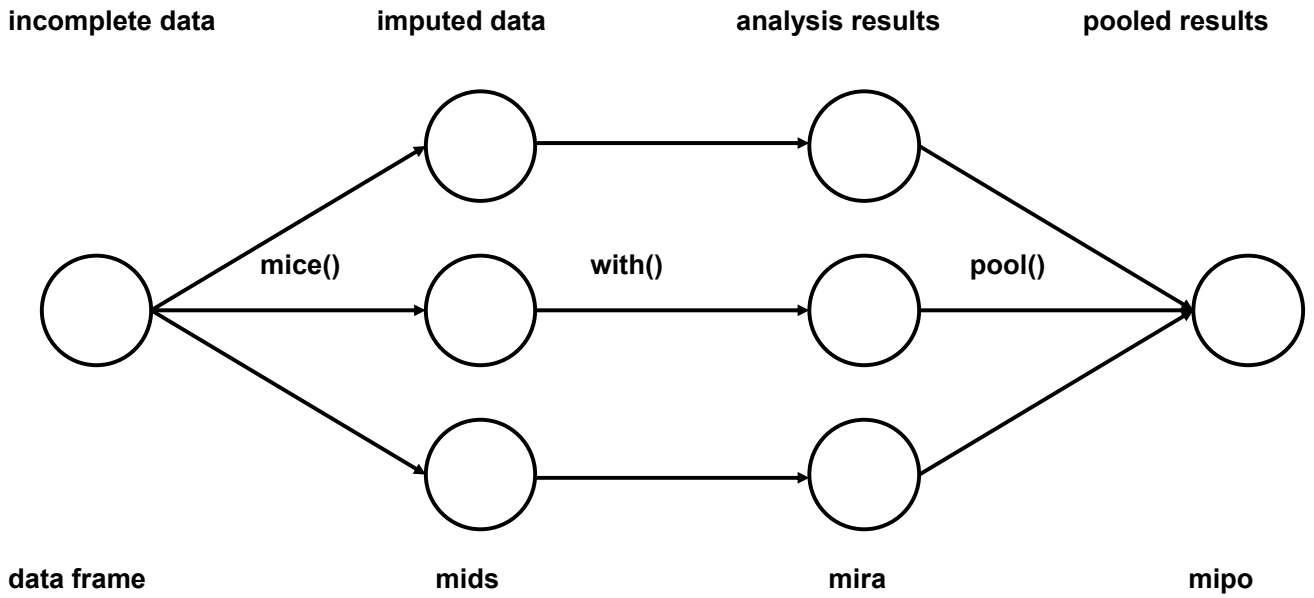
Idealmente, el modelo de predicción debe:

- Tener en cuenta el proceso que generó los valores perdidos
- Mantener las relaciones entre los datos
- Manejar incertidumbre de esas relaciones

Alternativas

- Amelia, caret (kNN)

MICE



van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software, 45*(3), 1 - 67. doi:http://dx.doi.org/10.18637/jss.v045.i03

MICE

Selección del modelo de imputación

1. MAR (*Missing at Random*) vs MNAR (*Missing Not at Random*)
2. Forma del modelo de imputación (estructura, distribución del error)
3. Conjunto de variables que se usarán como predictores
4. Imputar variables derivadas de variables incompletas
5. Orden de imputación de las variables
6. Imputaciones iniciales y número de iteraciones
7. Número de *datasets* a la salida

NoiseFiltersR

Implementación de algoritmos de preprocesamiento para tratamiento de ruido de clase en problemas de clasificación

Eliminan las observaciones identificadas como ruidosas o modifican la clase asignada

Métodos basados en distancia (vecindario) o clasificación (frecuentemente mediante *ensembles* y *cross validation*)

Sintaxis

- Dataset
- Fórmula describiendo la etiqueta con ruido y las clases que se utilizarán para calcular la probabilidad de ruido

Salida

- Datos sin ruido
- Vector de índices eliminados

NoiseFiltersR

Métodos

- AENN: All-k Edited Nearest Neighbors
- BBNR: Blame Based Noise Reduction
- C45ensembles: Classical Filters based on C4.5
- CNN: Condensed Nearest Neighbors
- CVCF: Cross-Validated Committees Filter
- DROP: Decremental Reduction Optimization Procedures
- dynamicCF: Dynamic Classification Filter
- edgeBoostFilter: Edge Boosting Filter
- EF: Ensemble Filter
- ENG: Editing with Neighbor Graphs
- ENN: Edited Nearest Neighbors
- EWF: Edge Weight Filter
- GE: Generalized edition
- HARF: High Agreement Random Forest
- hybridRepairFilter: Hybrid Repair-Remove Filter
- INFCC: Iteraative Noise Filter based on the Fusion of Classifiers
- IPF: Iterative Partitioning Filter
- ModeFilter: Mode Filter
- ORBoostFilter: Outlier Removal Boosting Filter
- PF: Partitioning Filter
- PRISM: Preprocessing Instances that Should be Misclassified
- RNN: Reduced Nearest Neighbors
- saturationFilter: Saturation Filters
- TomekLinks: TomekLinks

- summary.filter: Summary method for class filter