A short guide to data analysis (good practice)

Inês Dutra

Oct 9th, 2018

1) Univariate analysis: start by looking at each one of your variables. You may need to clean and preprocess data before you start using methods and models. Things you should look at for each variable:

- is the variable of one single type? (number, nominal, ordinal, boolean, etc)
  - check if numbers should be treated as numbers. Sometimes numbers are just codes and should not be treated as numbers by the methods and algorithms.
  - If the variable is actually numeric and it is continuous or discrete, it may be necessary to arrange them in intervals/groups. This is particularly true for variables such as age, height or weight. Be aware that arranging variable values into groups may depend on the goal of your study. For example, age may be put in different groups if we are studying patients mammograms or if we are studying prostate cancer (age ranges for medical examinations vary according to the patient condition). The same is true for values such as blood pressure, glucose or similar.
  - Special attention should be given to variables of type "date". Different software may read in dates in different formats. For example, portuguese and english formats are different, nominal variables such as gene names may have names that can be interpreted as dates - MARCH1 may be converted to 1-Mar. Dates can also cause trouble when some of the values are read as numbers (excel is particularly bad for doing that).

- If all values have the same value, this variable won't discriminate and won't help in the analysis. You can remove that variable from your study.

- Are there missing values?
  - Look for N/A
  - Be careful with blank (empty) cells or zeros
    - empty cells may mean that the value was not observed
    - zero may have a meaning in the domain context
  - If a variable has more than 70% of values missing, you may consider removing it
  - Be aware that some software will automatically fill up missing values. Weka, for example, fill up missing values with the mean, if the variable is numerical, and with the

mode, if the variable is categorical. This is true only for some methods. Not all methods fill up missing values. In the case of Weka you need to check the tab “Capabilities” in order to know what kind of variables the method will accept (for the independent variables and for the class variable). RapidMiner and R allows you to read in values according to specified types given by the user.

- Look for inconsistencies: impossible values (**noise**), out of range (be careful to not discard rare, but possible values – **outliers**). You can try to fix incorrect values. If you can not do it, the best thing to do is to remove the row with the incorrect value from your data.

- When handling categorical (nominal) variables, you may need to group together values that may have the same meaning (for example, “unknown”, “?”).

- Some systems will convert variables on reading. Weka and R, for example, do that. For some packages, R assumes that nominal variables should be **binarized**. Other packages assume that numerical variable values should be **normalized** or **standardized**.

2) Bivariate analysis: once you cleaned and preprocessed your data, you may start looking for correlations. Be aware that most correlation methods look for **linear** dependencies between pairs of variables.

- Linear correlations: Pearson, Spearman, Kendall
- Non linear correlations: polynomial, logarithmic
- (Multiple) Mutual information
- Entropy
- Distances (column-wise or row-wise)

3) Feature Selection

- dimensionality reduction (most common method: PCA – Principal Component Analysis)
- Filtered-based
- Wrapped-based
- Embedded
- Mostly used algorithms: Relief, CFS, Winnow

4) Learning

- Unsupervised
  - Clustering
  - Association Rules
- Supervised (do prediction)
  - Classification
    - Mathematical models
      - Logistic Regression
      - Linear discrimination analysis (LDA)
      - Support Vector Machines
      - Neural Networks
    - Search-based
      - Decision Trees
      - Random Forests
      - Bayesian Networks
      - Inductive Logic Programming
      -
  - Regression