STATISTICAL COMMISSION and Working Paper No. 11
ECONOMIC COMMISSION FOR EUROPE English only

CONFERENCE OF EUROPEAN STATISTICIANS

UN/ECE Work Session on Statistical Data Editing
(Cardiff, United Kingdom, 18-20 October 2000)

Topic II: Propagation of knowledge to users

A PROTOTYPE KNOWLEDGE BASE ON DATA EDITING AND IMPUTATION

Submitted by Statistics Canada[1]

Invited paper

I. INTRODUCTION

1. While developing and implementing editing techniques, statisticians acquire expertise that is worth sharing across statistical agencies. A public database seems to be a good vehicle to disseminate a set of techniques with related terminology. A glossary has been initiated in previous UN/ECE data editing workshops as a first step toward a knowledge base. Evaluations of edit and imputation systems can also contribute to build such a bank. The resulting product can make references to Euredit (2000) which is targetting methods rather than systems in their Internet database. Although a knowledge base can be developed at different levels, this paper targets an international infrastructure. It will present the key issues for the project: what should it contain, who should develop it and on which platform, and what protocol should be used for its maintenance, especially if users are encouraged to exchange their experiences.

2. Sections II to V give details on the proposed content: a glossary, evaluations of systems, practical experiences and technical papers. Sections VI and VII explain the practical aspects of the development and the maintenance of a knowledge base.

II. AN EDITING AND IMPUTATION GLOSSARY

3. The first aspect of a knowledge base on data editing and imputation is to make some definitions available to users. This would make sure all users have a common understanding of problems and methods before sharing expertise and experiences with techniques. A glossary was initiated by the UN/ECE Data Editing Group in the early nineties. Since then, the glossary has evolved to its current form (Winkler, 1999).

4. In many cases, terms can be linked together in some context. Ideally, the database should provide an easy way for the user to make a connection between these terms. The terms to be included in an editing and imputation glossary should cover specific aspects of the editing and imputation processes:

C **The basic components**. These are simple concepts that are used to develop methods and techniques. By their nature, basic components would often appear in the definition of some other glossary entries. Some examples are "data item", "imputation", "qualitative data", etc.
C **The methods and techniques**. They represent generic or specific algorithms to achieve editing or imputation tasks. Examples are "ratio editing", "donor imputation", "nearest neighbour", etc.

[1] Prepared by Claude Poirier.

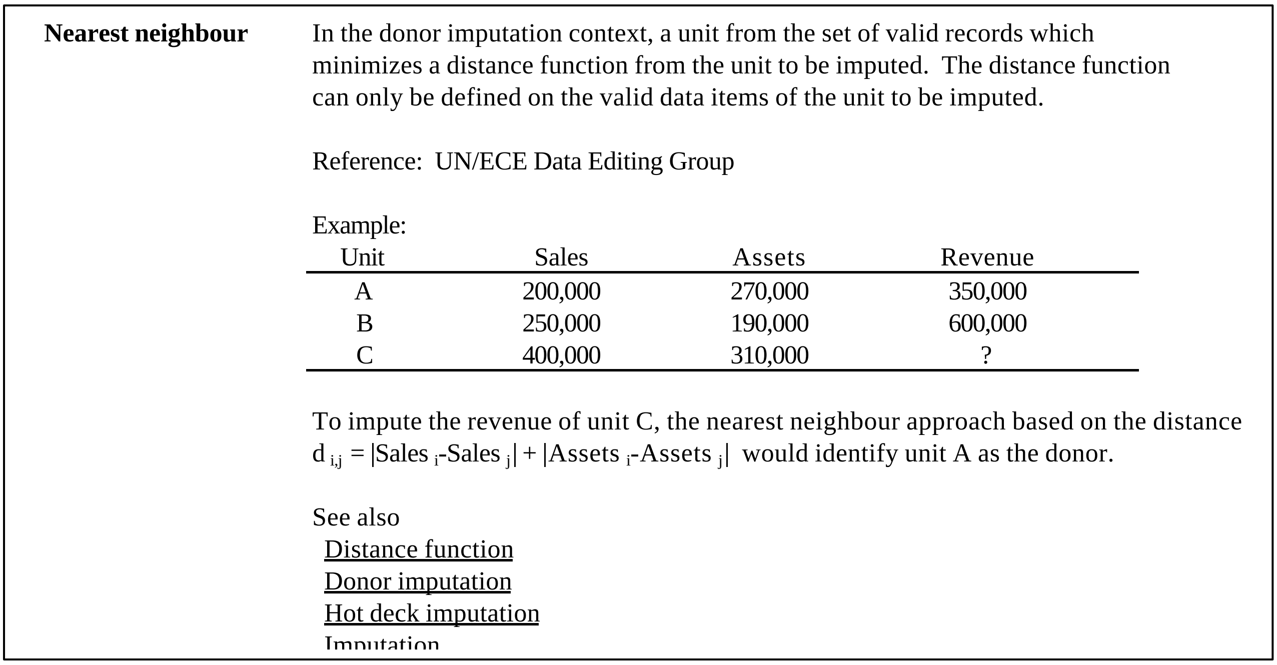
C **The principles**. They are approaches which drive the methods and techniques. Examples are "automated imputation", "minimum change", "computer assisted interviewing", etc.
C **Products or systems**. They include practical implementations of methods and techniques. Examples are "NIM", "GEIS", "StEPS", "Cherrypi", etc.

5. Then, each entry of the glossary should ideally include common components to provide a complete explanation to the users. As detailed below, the proposed components are:

- **The term itself** in its regular form, with no abbreviation, unless it is known as an abbreviation or an acronym,
- **A definition**, possibly with several contexts to which it applies,
- **References** which give the sources for the definition: a person, a group of persons, an agency, key papers, books, Internet pages, etc.,
- **An example** which helps readers in understanding the practical uses, and
- **Related terms** or synonyms that can be connected to the term of interest and which give a bigger picture of the context.

6. Figure 1 gives an example of the format a glossary entry may take based on the above five components. Some terms may present several definitions because of multiple uses, or because they can be seen from different angles. In such a case, references for each definition should be available to guide the users.

Figure 1. Example of a glossary format



Nearest neighbour

In the donor imputation context, a unit from the set of valid records which minimizes a distance function from the unit to be imputed. The distance function can only be defined on the valid data items of the unit to be imputed.

Reference: UN/ECE Data Editing Group

Example:

Unit	Sales	Assets	Revenue
A	200,000	270,000	350,000
B	250,000	190,000	600,000
C	400,000	310,000	?

To impute the revenue of unit C, the nearest neighbour approach based on the distance $d_{i,j} = |Sales_i\text{-}Sales_j| + |Assets_i\text{-}Assets_j|$ would identify unit A as the donor.

See also
Distance function
Donor imputation
Hot deck imputation
Imputation

III. FUNCTIONAL EVALUATIONS OF EXISTING PRODUCTS

7. A second aspect to be included in a knowledge base is information about existing products or systems. A system is a practical implementation of specific methods and techniques. Poirier (1999)

evaluated the functionality of four edit and imputation systems. Ideally, a functional evaluation of an editing system should cover the following aspects.

III.1 System description

8. A description is mandatory to explain the key functions and the context in which the system performs. A length varying from 500 to 800 words seems reasonable to describe a system. A description that goes over that limit would probably be too complex for the goal of the knowledge base. The description should mention the person(s) or the institute(s) that developed the system, as well as any supporting material such as papers, web pages, etc.

III.2 Evaluation criteria

9. Since the description has no specific structure nor content, there is a need for specific criteria on which every system should be evaluated. The systems would be scored according to each individual criterion. As explained later in this section, a scaling range has to be established for this exercise.

A) TYPE OF DATA:

10. Since there are very few systems that process mixtures of data, it is important to mention what type of data the system can process. The evaluation should inform the users how well the system process both quantitative and qualitative data.

> Quantitative data: We say that a system processes quantitative data when it uses algebraic operators or functions to process many continuous numeric variables in the editing or imputation steps.
>
> Qualitative data: A system processes qualitative data when its editing or imputation functions deal with many variables describing attributes, properties or unordered categories. Although there exist qualitative variables with ordered values (e.g., classes defining small/large, poor/rich, young/old, etc.) represented by numerical characters sets, they usually should not be processed with algebraic functions.

B) EDITING FUNCTIONS:

11. The process of detecting and handling errors in data includes the definition of a consistent set of requirements, their verification on given data, and the elimination or substitution of data which is in contradiction with the defined requirements. The following functions are the most frequently used and should serve as evaluation criteria.

> Data verification: The system can verify the validity of the data. This can be done at different levels (data item, questionnaire section, whole record) with various techniques (answer code, list of valid values, data format, range edit, relationship between variables, comparison with other records, historical comparisons, etc.). The process may suggest various actions like follow-ups, manual corrections, etc.
>
> On-line correction: The system allows a direct access to a database in order to modify values with the help of follow-up results, historical data, editor judgement or any other source of information which needs interpretation. A good system would allow concurrent users and would keep track of the changes everybody makes.
>
> Error localization: For the records in error, the system automatically identifies the fields which need to be modified in order to create good records. For a record in error, the error localization can be performed independently of the imputation process, or as part of an imputation method. A good error localization function should work with almost no manual intervention.

Minimum changes: As part of the error localization process, the system identifies the minimum number of fields to be imputed, or it minimizes the overall magnitude of the changes, or else, it minimizes other metrics. A good implementation should be able to solve the optimization problem even with high numbers of edit rules and variables.

User-defined changes: The system allows the user to specify which fields have to be modified for specific combinations of edit failures. This approach is the opposite of the automated error localization explained above. Ideally, the system will be flexible enough to allow a wide variety of user-defined rules.

Outlier detection: The system identifies data values that lie in the tail of the statistical distribution of the variable. Outliers may have to be imputed, or at least must not be used in the process of imputing other units because it is too different from the other values. A good function would let the user define the outlying criteria depending on the variable characteristics.

C) IMPUTATION FUNCTIONS:

12. The imputation process consists of replacing missing or unusable values with other values. Post-imputation rules may be required to make sure the process provides valid values. Following are some of the most popular imputation techniques the system should be evaluated against.

Deterministic imputation: The system can identify the situations where there is only one possible value that can be used to impute a given field. It is desired that such a function works not only with independent edit rules, but also with cross-edit rules.

Donor imputation: Here, a unit (donor) is identified from the set of valid records, and its data are used for imputation purposes. The donor selection can be anywhere between the random process and the nearest neighbour approach. A good implementation would allow the use of constraints or specific matches to reduce the set of potential donors for each record to be imputed. It would also verify that a minimum number of donors are available to impute the units in error.

Imputation by estimators: Various estimator functions, also called models, can be used to impute data: Mean, ratio, trend, regression, etc. They can be applied to a mixture of current data, administrative data and historical data. Similarly to the donor approach, the estimators should allow the definition of models within sub-populations, with constraints on the number of records required to estimate the model.

Multiple imputation: The idea of this approach is to generate several imputation runs based on a unique method, in order to produce variability and quality indicators. The implementation has to offer general models, with a user-defined number of replications of the process.

D) GENERAL CRITERIA:

13. Other criteria have to be considered in the evaluation of systems. They are mostly related to the features and tools that make the system accessible to inexperienced users.

Graphical user interface: This refers to a menu structure which helps in the setting up of functions and the submission of computer jobs. The interface should be preferably mouse-driven and intuitive from the user=s point of view. Functions that are similar should be implemented similarly in the interface.

User-friendliness: How easy and how quickly can the user learn the system? Is it interesting to use? Examples of good practices are the transfer of settings among versions of the system, among job submissions and among repetitive processes, the choice of non-aggressive colours that help differentiate concepts, functions or outputs, the parameter inputs specified in an intuitive order, the outputs being easy to read, etc.

On-line help: An on-line help provides information on the spot. Ideally, it should provide a description and some instructions on the active module by default. It should also offer a navigation option to obtain information on any module.

On-line tutorial: A tutorial is an important component of the training material. It should offer an opportunity to play with live data as opposed simply providing a static show.

Documentation: A well documented system should offer a detailed methodological description of its functions as well as a system's documentation which details the data flow and its processing, and a complete user's documentation.

Diagnostic reports: The diagnostics include information and statistics that help users to monitor how well the process is performing. Examples are: the number of records or data items which failed edits, the number of records in a donor pool, etc. Good diagnostic reports have to be easy to read, with enough statistics to understand what happened but not too much that the useful information is drawn into a mass of numbers.

Integration: This feature refers to the flow of data across various survey steps, including the editing and imputation processes. A well integrated suite of systems would allow the processing of data without requiring any reformat or pre/post processor. For instance, the input of an edit and imputation system usually comes from a collection/capture system, and its output goes to an estimation system. This creates a need for an integrated suite. A record linkage process may also be required to get auxiliary information for the data editing and imputation. A whole data stream developed on a unique platform would be desired, but the level of complexity of each step often justifies the choice of different platforms.

Reusable code: Reusability refers to the possibility of using the system to process data from various censuses, surveys or administrative sources. We also refer to this as the generalization of the function, which allows the development of specific applications simply by changing input parameters.

Portability: A portable system is relatively easy to install because it requires simple foundation softwares, if any are required. For instance, a set of executable files which need a simple copy to a Windows environment is portable. If the product can be installed on other platforms as well (UNIX, MVS, etc.), it is a bonus.

Flexibility: A system built from individual and self-contained modules is flexible. These modules should be embeddable in another survey stream whenever needed. This allows the user to replace some modules by his own customized function, or to add additional modules easily. An open source code may be considered here to facilitate these additions.

User support: Users' support via telephone and/or e-mail is desired if it cannot be in person.

Acquisition cost: If applicable, the acquisition cost should be mentioned in an evaluation.

III.3 How to rate a product?

14. To make the evaluation easier to understand, a standard scoring should be defined for the set of criteria. Although the scoring is a subjective process, some scales can be proposed to try to standardize everybody's interpretation. A 0 to 10 rating represents a possibility, but it may be too refined: How do we distinguish 6 from 7 ? The author proposes the following 0 to 3 rating to qualify the functions and its quality, flexibility, efficiency and reliability: (3) is given to a function when its implementation offers the sub-functions or the options being required by a wide range of survey applications. This does not mean, however, that no improvement can be made to the function; (2) is given to an implementation having a less complete set of options; (1) means the implementation offers a partial functionality; (0) is assigned when the functionality is not offered at all.

IV. SHARING EXPERIENCES

15. A knowledge base should also provide a means to share experiences with specific techniques and specific products. Statisticians regularly study techniques and compare them against each other in various contexts. Papers documenting these studies should be made available through a knowledge base. As mentioned above, Euredit (2000) plans to share evaluations of techniques and methods.

16. As for the existing products or packages, statistical agencies spend large amount of resources in evaluating and developing applications based on them. The idea of sharing good and bad experiences with specific products would help many users. For any specific product, typical users would be interested to learn from colleagues about:

- the technical and statistical expertise they needed to initiate a first application,
- the money and human investment required in setting up the hardware/software environment,
- the installation of the product itself,
- the number of employees involved in developing the application(s),
- the required training with the product and/or with any foundation software,
- the sources of data being processed,
- the volume of data being processed,
- the development of pre-processors and post-processors,
- the changes that had to be done to other systems to integrate the product into a data stream,
- the methods that had to be modified because of the available functionality,
- the turn around time for the investigation,
- the difficulties encountered during the development and the production phases,
- the overall comparison with former systems, and
- any comments related to the evaluation criteria listed in Section 3.2 D.

V. TECHNICAL PAPERS

17. The fourth aspect of a knowledge base is to offer a repository of technical papers. The UN/ECE (1997, 1999, 2000) posted on its Internet site all the documents produced at the corresponding workshops on statistical data editing. These documents are available for each work session but there is no search engine to access them by topics, key words, or authors. A knowledge base should ideally include such functions.

VI. THE DEVELOPMENT OF A KNOWLEDGE BASE

18. Given the proposed four components of a knowledge base, the structure of a prototype had been thought of and a first implementation was recently initiated. The first step of the development was to determine the platform and the foundation software. Some questions needed answers: Should the base be developed as a stand-alone package that can be distributed to interested persons, or should it be a centralized package that offers concurrent access? Should it use database management systems, help file technologies, web technologies, or a mixture of these?

19. To make choices, the goals had to be established: The knowledge base should be accessible by the international community, without having to send individual copies to each user. In terms of features, it should be developed with a dynamic structure that allows modules to be updated and new entries to be added with regular technology, without having to recompile the product for each modification. Its foundation software should be well known to make its maintenance easy, with simple tools. It must allow the navigation from item to item while offering active links between related concepts. Although it has a

modular structure, it should offer print options that allow the printing of individual modules as well as the whole content. It should also offer a search engine that can extract information corresponding to keywords.

20. These desired features lead the development toward a centralized package being accessible from the Internet. The Web technology had to stay simple to be compliant with a wide range of browser versions. The standard HTML language was then chosen.

21. The first prototype of the knowledge base is to be loaded with materials from the UN/ECE Data Editing Workgroup: an editing and imputation glossary, an evaluation of software, and a collection or papers published on the Internet.

VII. THE MAINTENANCE OF A KNOWLEDGE BASE

22. All the components of a knowledge base have to be kept up-to-date to make sure the product is evolving. Definitions of glossary entries may be missing or incomplete. Some others may be improved with additional information or with applications into more contexts. Some terms, techniques and principles issued from past or new experiments may also be added to the glossary. As for the evaluation of systems, additional opinions on existing systems would always be interesting to share so that users would be encouraged to provide results of their own experiments. Of course, information about systems that are not already covered would be more than welcome. This may include new systems or new versions of systems. Finally, relevant technical papers have to be added to the base regularly.

23. In order to achieve a continuous evolution with the considerations mentioned above, the knowledge base has to include, on one end, a tool to collect comments, proposals, modifications and additions to its content. At the other end, a team of experts has to be identified to analyse the proposals and decide how to use them for the updating process. They would have to establish contacts with specialists on specific topics. Another responsibility would be to make sure users' recommendations are serious and realistic, and that opinions are not conflicting with their professional interests. What must be avoided is a base that is perceived by system developers as a promotional tool to present only the good side of their "revolutionary" products.

24. These experts have to come from several agencies to most likely provide unbiased opinions for the updating process. The UN/ECE work group would provide good candidates for a maintenance committee.

VIII. CONCLUDING REMARK

25. A knowledge base is a tool that provides information about the editing and imputation processes. The proposed content includes a glossary of terms, a repository of evaluations of systems, the description of practical experiences and a set of technical papers. Standards have to be adopted to manage the content, especially to ensure glossary entries have a common format and that systems are evaluated against a same set of criteria. Although the prototype developed from the above recommendations targets an international audience, it is understood that similar products can be developed within statistical agencies. The maintenance of the base is a key aspect for product quality . There are many examples of good implementations that quickly deteriorated because of a lack of maintenance.

REFERENCES

Euredit. *"The development and evaluation of new methods for editing and imputation"*. Internet page, www.cs.york.ac.uk/euredit, University of York, United Kingdom, 2000.

Poirier, C. *"A Functional Evaluation of Edit and Imputation Tools"*. Proceedings of the Workshop on Data Editing, UN-ECE, Italy (Rome), 1999.

UN/ECE. *"List of Documents"*. Internet page of the 1997 Workshop on Statistical Data Editing, www.unece.org/stats/documents/1997.10.sde.htm, Czech Republic (Prague), 1997.

UN/ECE. "*List of Documents*". Internet page of the 1999 Workshop on Statistical Data Editing, www.unece.org/stats/documents/1999.06.sde.htm, Italy (Rome), 1999.

UN/ECE. *"List of Documents"*. Internet page of the 2000 Workshop on Statistical Data Editing, www.unece.org/stats/documents/2000.10.sde.htm, United Kingdom (Cardiff), 2000.

Winkler, B. *"Draft Glossary of Terms Used in Data Editing"*. Proceedings of the Workshop on Data Editing, UN-ECE, Italy (Rome), 1999.