

# Nifty with Data: Can a Business Intelligence Analysis Sourced from Open Data form a Nifty Assignment?

Matthew Love  
Dept. of Computing  
Sheffield Hallam University  
Sheffield S1 1WB, UK  
m.love@shu.ac.uk

Charles Boisvert  
Dept. of Computing  
Sheffield Hallam University  
Sheffield S1 1WB, UK  
c.boisvert@shu.ac.uk

Elizabeth Uruchurtu  
Dept. of Computing  
Sheffield Hallam University  
Sheffield S1 1WB, UK  
e.uruchurtu@shu.ac.uk

Ian Ibbotson  
Better with Data Society  
<http://betterwithdata.co>  
ianibbo@gmail.com

## ABSTRACT

This paper describes an assignment investigating the relationship between weather conditions and levels of air pollution. The case study illustrates aspects of finding and accessing Open Data, sections of the Extract-Transform-Load processes of data warehousing, building an analytic cube, and application of data mining tools. It is intended to aid tutors and students of databases by providing a study that gives a practical case example that gives an overview of how several topics in the area of data collection and analysis integrate together.

As well as being an interesting assignment in its own right, the case study raises a number of questions about what makes a good assignment for students learning to handle data (some of the usual nifty criteria have to be adapted), and about the use of Open Data in student assignments.

## CCS Concepts

•Information systems → Information integration; Decision support systems; Data mining; •Social and professional topics → Student assessment;

## Keywords

Nifty assignments; Open Data; Business Intelligence; Computer Science Education

## 1. INTRODUCTION

This paper proposes a *nifty assignment* in data mining. We consider the sources of data used, to study

whether Open Data can form the basis of more such assignments, and if so how. We then discuss how assignments differ between databases and other domains in computer science, particularly programming, and consequently how that affects criteria of quality or *niftiness*.

In the next sections, we describe the nifty assessment criteria and explain why use them as a standard for quality of assessment. We then propose an assignment which outlines a number of topics related to finding and accessing Open Data, merging sources, and analysing the data using self-service and data mining tools.

Once the assignment is clear, we will reconsider it against the nifty criteria, but also consider how the criteria themselves apply to the area of data mining which has few assignments proposed. Finally, we will consider whether the basis of this assignment, the use of Open Data as a source of data to analyse, can be extended to different cases and examples, and if so how.

## 2. QUALITY IN ASSIGNMENT

The notion and the need for the *nifty assignments* repository has been solidly defended by Nick Parlante [10]:

Assignments play a crucial role in what my students take away from a course, but I'm always amazed at what an error prone and time consuming process it is to put together a good assignment.

Since 1999, Parlante has been maintaining a repository of assignments, coordinating regular additions at SIGCSE - most recently in [11]. The repository has become a reference, analysed for example by Layman et al. [8] to find the social context of scenarios, or by Fincher et al. [5] to evaluate repositories.

Given the popularity and established character of the nifty assignment repository, its criteria constitute a good framework to start evaluating assignments in

areas cognate to computer science, including data mining. Here are the criteria as proposed on the Nifty Assignments website [9]:

- **Nifty** – Nifty Assignments often have a playful sort of "fun factor" to them. They are very visual, or they build a game, or they have entertaining output. The assignments invite the students to play around with the material. Of course we shouldn't regard this as a requirement. Not all CS fits into the "game with blinking lights" motif.
- **Topical** – most Nifty Assignments fit into the curriculum and difficulty range that makes sense for most schools (typically CS0-CS2). This is just a practical bias, where we want to promote assignments that can work for the greatest number of students. Platform independence is also desirable, and we try to avoid dependencies on non-portable or non-standard libraries. At present, Java is a great language to make your Nifty Assignment adoptable by the widest audience.
- **Scalable** – many Nifty Assignments operate at two levels. First and most importantly, there's the mainstream part of the assignment that is nifty, meaningful, and effective for the average student. Beyond that, many Nifty Assignments have an open-ended aspect where advanced students can take the assignment beyond its original boundaries.
- **Adoptable** – for an ideal Nifty Assignment, the author has put together materials that make the assignment easy for another instructor to adopt: handouts (.rtf, .doc, or .html formats), starter source code, data files, and other ancillary materials. Here again, platform independence, use of open, vendor-neutral languages, libraries, etc., is a plus. Although they might lack glamour, high-quality materials are appreciated. It's easy to think of Tic-Tac-Toe or whatever as an assignment, but there's a big gap between the idea and having all the materials tested and ready to go. In this way, Nifty Assignments can complete that last step for the community, making the idea concretely available for the whole community.
- **Inspirational and thought provoking** – sometimes a Nifty Assignment is just thought provoking about what is possible in an assignment, inspiring people to work out their own assignments more than being something a lot of people adopt.

Some of these criteria apply straightforwardly to a data mining problem. But we will see that the peculiar process of procuring data, merging sources, and analyzing it, represents a large volume of work that



Figure 1: Nitrogen Dioxide diffusion tube vs. 'Groundhog' pollution automated station

may require to redefine what nifty means in the context of teaching and learning Business Intelligence.

### 3. CASE STUDY: AIR POLLUTION IN A MAJOR CITY

All the ancillary documents, data and scripts are available online<sup>1</sup>.

Air pollution kills people. It is estimated that in the UK 29,000 people die early every year due to breathing difficulties at times of low air quality [1]. The UK government has imposed targets for reducing the quantities and/or frequencies of the main pollutants. Local Authorities are responsible for monitoring and publishing pollution levels in their areas.

Sheffield City Council uses two types of monitoring devices in the city: diffusion tubes and fully automated processing units. Both types of devices are illustrated in Fig. 1. There are around 160 diffusion tube devices and six fully automated processing stations. The diffusion tubes have the advantage of being spread throughout the city area. However, they give data only when sent in for analysis, and typically this is once every six to eight weeks per tube. The results are aggregated to an annual level prior to publication. The six automated processing stations, named 'Groundhogs', measure a variety of pollutants, and one also measures temperature and air pressure. Between three and eight readings are taken per hour. After a short delay, the public can access the log; the council occasionally may correct or delete readings from it. Although some of the stations have been operating since 2000, there are a number of gaps in the data logs. In addition, the stations are occasionally moved (usually to help investigate new pollution 'hot spot' concerns).

The council maintains a web site<sup>2</sup> that provides informative descriptions on pollutants commonly found in air, including relevant related images and additional information<sup>3</sup>.

Readers (and their students) are invited to visit the

<sup>1</sup><http://aces.shu.ac.uk/AirQuality>

<sup>2</sup><https://www.sheffield.gov.uk/environment/air-quality/monitoring.html>

<sup>3</sup><http://sheffieldairquality.gen2training.co.uk/sheffield/index.html>

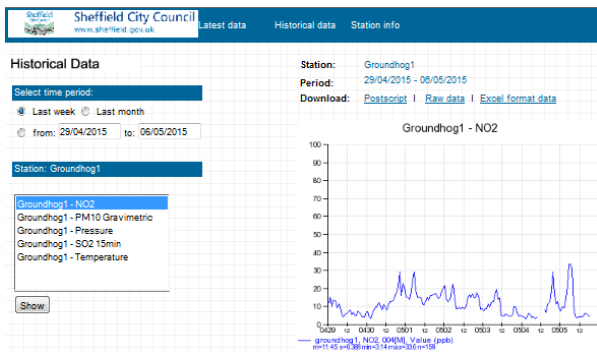


Figure 2: Council automated station results

latter website and then select 'Station info' (see Figure 2). This page illustrates a very common problem with data sourced from the Internet. The information is presented as textual descriptions, with no obvious way of automatically deriving further information. For example, readers are told that Groundhog1 is at 'Orphanage Road, Firhill' but it would take a human-based Web search to find the geographical location and then further searches to discover the nature of the location (residential or industrial area, proximity to main road, etc).

Tutors may use this to introduce a discussion on why data on the internet is not necessarily considered as *Open Data*. Moreover, if the user clicks on the 'latest data' tab, and then click on any of the Groundhogs (Groundhog1 is often the best choice), they will notice that navigation to this page is not designed for automation. The user must click on a visual map to select the page. Notice too, that the URL for data pages does not reflect the name of the Groundhog being visited. These issues might be used to lead students to reflect on how, in an age of Internet-sourced data, URLs can and should be designed to allow for automated discovery by data harvesting tools.

The website, however, allows data (of any user-selected range) to be downloaded, in a choice of *PostScript*, *Raw* (i.e. comma separated values) or *Excel* formats. Students should — one 'hog', and one pollution type at a time — download CSV files. The Nitrogen Dioxide files (NO<sub>2</sub>) total about 15 MBytes. Downloads may be inspected using a text editor (avoiding Microsoft Notepad, as the End of Lines are not compatible), when it can be seen that there are date (YYMMDD), time (HH:MM) and NO<sub>2</sub> reading, roughly 3 to 8 readings per hour.

### 3.1 Integration of further data sources

One of the principles of Data Warehousing when used for analytic purposes (as opposed to 'data store housing' for safe custody of data) is to try to give added context to facts, through Dimension descriptors added from other sources.

Groundhog1 lists temperature and air pressure readings. But other factors may influence pollution formation and/or dispersal as well. Obvious factors are wind strength and humidity. Wind direction is also a factor (a complex one: if the monitor is west of a pollution source then a strong west wind will increase measurement values; if the monitor was directly east of the source, then the same wind would remove the pollution from the area of the monitor).

Detailed historic weather data is commercially valuable, and rarely available for free download. Sheffield is fortunate in having a local enthusiast who had monitored and published readings at five-minute intervals for all the desired measures. Unfortunately the data is published in PDF format, with documents of around 200 pages per month of data. The tool *Bytescout PDF viewer*<sup>4</sup> can be used to extract all pages into one CSV file. Readers are requested to contact the data owner —details online— to get permissions to use the data for non-commercial purposes (any commercial use of the data could cause the site to be closed).

### 3.2 The Data Warehouse, and ETL processes

All data values are then uploaded into a Microsoft SQL Server with Business Intelligence database. This software is free (for academic use) to install from Microsoft Dreamspark<sup>5</sup> onto university teaching systems and student laptops. Alternatively, students can have 150-day free use of the same software from the *Microsoft Azure cloud platform*<sup>6</sup>. *Microsoft Azure* has convenient setup options for *SQL Server Business Intelligence*.

Once the data is loaded into tables on the *SQL Server* it needs to be transformed into formats suitable for data analysis. The case study demonstrates a realistic but manageable number of steps that can be found in many Extract-Transform-Load systems of Data Warehouses (the scripts used for ETL are available). Tutors can use the ETL process to contrast the Server's menu-driven wizard approach for uploading files into tables, with SQL scripts that do the same tasks. Students are not always aware that SQL has commands that allow for manipulation of database structure (as opposed to manipulation of data values), but quickly start to see the value of relatively short scripts that can be reused across multiple uploads. More complexity comes from the 'Sheffield Weather Page' data being at five minute frequency, while the Groundhog readings vary between twelve and twenty minute frequencies. Further SQL scripts first summarize the respective Groundhog and the Weather data into hourly readings (taking the means of readings

<sup>4</sup><https://bytescout.com/products/pdfmultitool/index.html>

<sup>5</sup><https://www.dreamspark.com>

<sup>6</sup><http://azure.microsoft.com>

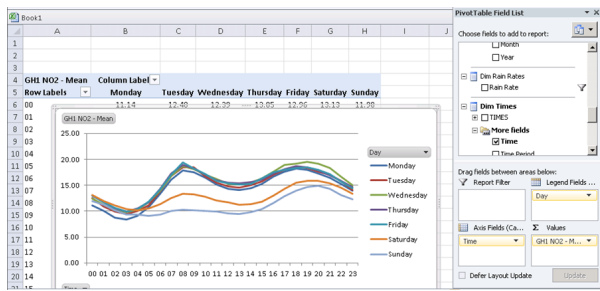


Figure 3: Self-service display of data: Nitrogen Dioxide levels per hour on days of week

within each hour, except for wind direction where the most frequent wind direction was taken), and then integrate these into a single observations table. Students should open a second database on the same server, copy the observations table (via a short script command) into it, and create descriptive tables that give informative names and attributes for the Groundhogs, and descriptive category names range limits for each of the weather attributes (for example *dimWindSpeed*: No wind = 0 kph; very light breeze = 1-3 kph, through to strong winds 20 kph and over). A script then creates a data 'Star' based on Kimball's designs [7], with a single Facts table linked to relevant rows in each of the Dimension tables. Having two databases on the same server, one for ETL data acquisition and preparation, and one for storage of the integrated Star of facts and dimension tables, helps students see for themselves the concept of a Data Staging area as described throughout Kimball's work. Just as Kimball describes, all the messy processes happen, hidden from end-user view, in the Staging area. Clean, usable, subject-structured data is then published to data marts.

### 3.3 Creation of Data Cube from Data Star

SQL Server with Business Intelligence provides a facility for defining Data Cubes for fast analytic processing. Cubes can source their data directly from the uploaded CVS files, but students quickly appreciate the simplicity of sourcing from the Star created in the previous step. Refreshes in the data values in the Star (or even alterations in the design of the Star) can quickly be pulled through into the Cube. By default, when used for self-serve reporting (Figure 3) cubes automatically report totals (sums) of data value, aggregated over the user-selected timeframe (or geographic distribution, etc.). For example, selecting NO2 (Nitrogen Dioxide) would automatically report on the total readings ever, or totals per year, or totals per month, or per day, or even per hour, depending on what date-range the user happened to select. Users usually start at the top level -the most aggregated- and then 'drill down' for more details.

In the current case study the averages of pollution

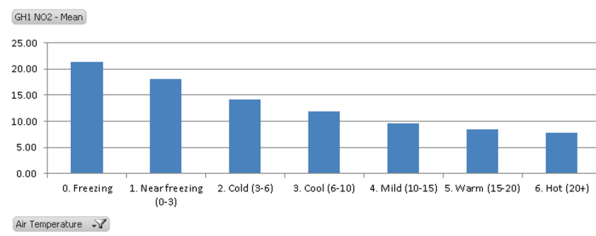


Figure 4: Average NO2 levels for categories of temperature

values are much more relevant than the totals. It is a lot easier to compare calendar months of data if an averages are used, as this eliminates that some months are longer than others. Peak values within any selected time frame are also of 'headline' interest, but users need to treat information with caution as a peak may well be caused by a local factor such as a badly tuned lorry or tractor passing upwind of the monitor station).

The 'Calculated Measures' facility of the Data Cube was used to set formulas to report the means of each of the numeric measures. The formula for mean is as simple as 'Sum of NO2 divided by Count of NO2': the cube automatically applies the context of level of drilling for all selected dimensions. Setting up medians is beyond the scope of this simplified case study, but students can discuss how median values can be used to ignore the effect of outlier readings. Many texts on Data Warehousing utilize Inmon's term of subject-oriented [6]. In simple case studies students often cannot see the difference between the data sources and the DW subject-orientation. One differentiator is that business rules can be encoded into the data or data presentation within the cubes. For NO2 air pollution, 40 mg is a threshold for concern, and 100 mg is a threshold for serious concern. Facilities within the Data Cube were used to encode these levels into colors of presentation. Key Performance Indicators, with 'traffic light' colors and 'trend' arrows could also be set up. The threshold values can usefully be explained to students as examples of Business Metadata, contrasting with Technical Metadata (such as field data types) more often seen in tutorials.

### 3.4 Self-Service Data Exploration

Microsoft's preferred self-service data exploration tool is Excel. Indeed, a single button click from within the cube development tool will open the cube in Excel. Data is presented via Pivot Tables (readers should note that the Azure cloud platform does not contain Excel. Users can set end-points to allow their local copy of Excel to link to the cloud server. Alternatively users can simply install 30-day trial copies of Office onto Azure).

Students will very quickly (within minutes) start making discoveries about the data. For example, Fig. 3 shows

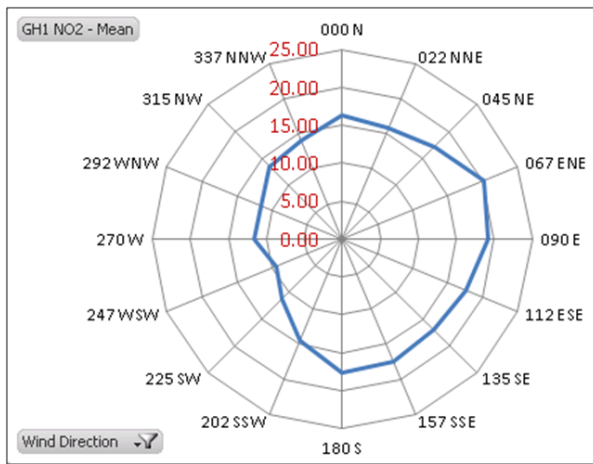


Figure 5: Average NO2 levels for source direction of wind (Groundhog 1 monitor)

NO2 pollution levels varying across time for each day of the week.

This image prompted a lot of discussion as to the timing of the apparent peak times for pollution (the effect of driving?) and the clear difference between Saturday and Sunday versus the rest of the week. It can also be discovered that freezing or near-freezing days are associated with high NO2 pollution levels (Fig. 4), and that low east winds measured by Groundhog1 coincide with worse pollution (Fig. 5). Students can self-serve discover other relationships between the data. Some are obvious (winter months tend to have colder days), but students do get to experience the concept of a data analyst exploring the data themselves. Many students do not know that displays other than line graphs and bar charts are available, and useful discussions can be held about using comparative percentages as a means of spotting patterns or exceptions.

### 3.5 Data mining

Many students of databases get a few introductory classes on Data Mining, but may not get to build and use a data mining facility for themselves. Having got the pollution and weather data into SQL Server, the same environment can be used to develop mining reports within a few minutes and with no further coding. For example, using the SQL Server Business Intelligence suite, a Clustering algorithm has identified ten clusters of weather data. The darker clusters, for example Cluster 9, contain a high proportion of bad pollution days, while the lighter ones (for example 3 and 6) contain hardly any bad pollution days (Figure 6). Cluster 9, the cluster with a large proportion of High NO2 readings, can be understood by analyzing the characteristics of the cluster in more detail. An analysis shows that these are low pressure days, high/very high humidity but not actually raining, and

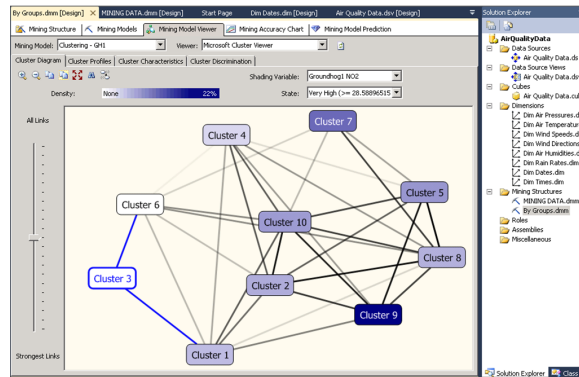


Figure 6: Result of running the Cluster data mining tool

cold or near freezing temperatures. In other words: murky, dry winter mornings. More plots are available including Association Rules, Decision Trees, Neural Nets, Regression and Naive Bayes.

The default settings for each of the analysis shown produce interpretable results quickly, then fine-tuning the parameters (controlling the number of clusters, for example) can improve data interpretation. Discussing with students what the parameters do can help students progress to more unsupervised learning. A frequent discussion point is whether Categories can then be fed back into the Data Warehouse, to fine-tune the Dimension attributes.

## 4. EVALUATING NIFTINESS

This case study demonstrates the application of Data Warehousing and Data Mining tools, using data gathered from Internet sources. But it does not fit with the intended nifty assignment criteria straightaway. Let's consider the assignment against the *nifty* criteria:

- **Nifty.** Business intelligence doesn't lend itself easily to the 'game with blinking lights motif'. But the use of a local subject raises the students' interest.
- **Topical.** Most assignments in the nifty repository are about programming in one form or another. This one isn't, and so it may appear to ignore that criterion. However, it supports 'Data Management Systems', one of the core knowledge areas in the ACM computer science curriculum [12]. We would argue that Data Management quality assignments are, in fact, particularly difficult to produce, and that therefore there is a real need to provide examples and ideas in this area.

It is also expected that the assignment discussed here will provide ideas to broaden the scope of nifty assignments. We could find only one other example of nifty assignment solely intended for Data Management, [3].

- **Scalable.** Frequently, Data Management assignments stay within reach of the ‘average’ student by radically limiting the scope of the problem and the data to study. One characteristic of this example is that rather than taking conveniently prepared datasets, it proposes to experience some of the common difficulties met by database professionals when collecting data from non-traditional sources.

The present case study, and online support documentation, is intended so that tutors can use the resources described to give overview presentations of the topic without needing to overcome problematic barriers.

The case study involves more data than might normally be used for overviews of the subject and is ‘rich’ enough in content to help highlight genuine issues, but remains sufficiently structured and simple as to not become overwhelming. This introduces and discusses a number of ‘real-world’ issues, particularly around the Extract-Transform-Load procedures of Data Warehousing, while keeping the study to a scale that is feasible for quick comprehension by students. Largely, each of the steps can be done by taking default options of off-the-shelf tools, and mistakes in the design can be recovered simply by re-running relevant steps. Of course, there is a risk that students may ignore the alternatives to defaults. However, our experience is that it is very helpful to be able to see the ‘end-to-end picture’ at a relatively early stage. Better students are then able to revisit the pieces to see their connection with their mainstream database and data analysis studies, and develop a fuller mastery of the tools available.

- **Adoptable.** To make our work *adoptable*, all the tools used and data used, were selected to be available for free use in academic contexts; although note that the weather data could only be used with permission. Links and resources - assignment information, software tools and data are available online
- **Inspirational.** The theme of pollution in our city is one that interests and motivates our own local students. Many raise questions that the data might be of value in relation to public interests such as health or traffic. So the inspirational value of the assignment is in part due to the richness of this data set for local students. This of course will not apply to every potential user of this data.

The last point raises the question whether the ideas developed in this assignment would be adaptable to a wider range of students and of situations. To this end, we will return to the data sources of our assignment

and consider whether we can exploit a movement to help increase the availability of public data: Open Data.

## 5. OPEN DATA: SELF-EVIDENT MATERIAL?

This time, Open Data (according to the *Open Data Handbook* [2], ‘data that can be freely used, re-used and redistributed by anyone’) provided a valuable case study. It might be expected that the data liberation movement offers many further examples, relevant to the locality and interests of our students at more institutions. As Atenas and Havemann [4] put it, in an educational context, Open Data becomes an Open Educational Resource by very definition.

### 5.1 How not to assess with Open Data

In practice however, Open Data is not a simple choice. Open data does not become usable for student assessment the moment it is publicly released. We may hope that its availability will lead to insightful analyses and applications; yet that process is anything but self-evident.

The release of Sheffield pollution data on a convenient, clear website is the result of several years lobbying local authorities to engage with liberating data. As part of this process, the *Better with Data Society*<sup>7</sup>, a local Open Data group, welcomed students to apply their skills to the data released. The only result has been to show how unsuited that data was to student work. Only one student eventually pursued Open Data, and that was an investigation as to why *Better with Data* was making so little apparent progress!

The pollution data assignment presented above works precisely because a lot of work has been done to identify, cleanse and ensure the availability of the data, with the result that the students are proposed a carefully chosen, set of sources, confusing enough to be challenging, yet within reach of the students in the time available.

All this highlights the need for better methods of publishing data for machine discovery and consumption.

### 5.2 How the Air Pollution assignment was born

Such a better method may form an alternative start for the assignment: to investigate the data in the machine-readable form of linked data. The impetus for the assignment presented here is actually the *Air Quality+* database, which allows over-the-web interrogation of the Sheffield pollution measurements as linked data.

Two of this paper’s authors developed *Air Quality+* and an editor for SPARQL queries. The database holds each *Groundhog* and the sensor measurements for the *Groundhog* stations and diffusion tubes. For example,

<sup>7</sup><http://betterwithdata.co/>

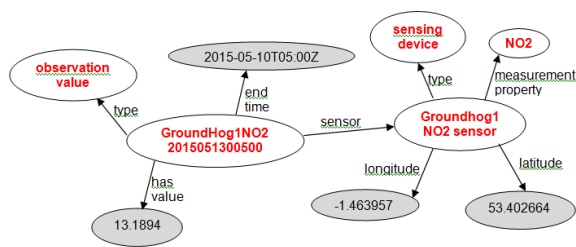


Figure 7: A subset of the *Sheffield Air Quality+* database. Data points in bold text are URIs

as figure 7 shows, the data about the `Groundhog1 NO2` sensor records what it measures, its location, the type of device it is, and actual measured values with their date and time.

Educators interested in linked data could consider accessing the pollution data and sensor information via the the SPARQL editor<sup>8</sup>; this can be a worthwhile prequel to the assignment above. But our choice has been to not offer such a neat, ready-made, machine-readable access to the data for our students, precisely because the journey of discovering a set of open data and struggling to make the data into a usable form, is itself a learning experience. As Sooriamurthi [13] said of his assignment: “it’s not a calendar, it’s a journey”. Having traveled that journey, the path we map for our students is not shorter, but offers more opportunities for discovery.

## 6. CONCLUSIONS: NIFTINESS WITH DATA

The assignment discussed in this paper demonstrates the application of Data Warehousing and Data Mining tools, using data gathered from internet sources. Rather than taking conveniently prepared data sets, it shows some of the common difficulties met by database professionals when collecting data from non-traditional sources.

The same difficulties force us to redefine what nifty assignment might mean in a data mining context: the ‘fun factor’ is difficult to find, but the work of educators is in identifying and preparing data sets appropriate to the level and the interests of the students. At the same time, we show that there is value in not preparing data too carefully, because this is precisely the expertise that students need to develop.

Finally, the reader should be aware that the opportunities offered by Open Data with students are not easily realised: providing an assignment which offers both an appropriate challenge for all students and open-ended opportunities for the better ones, requires discovering the data and considering its possibilities in detail before letting students investigate it.

<sup>8</sup><http://www.boisvert.me.uk/opendata>

## 7. REFERENCES

- [1] Estimating local mortality burdens associated with particulate air pollution. public health england, 2014. [https://www.gov.uk/government/uploads/system/uploads/attachment\\_data/file/332854/PHE\\_CRCE\\_010.pdf](https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/332854/PHE_CRCE_010.pdf). Accessed: 2016-01-12.
- [2] Open data handbook. <http://opendatahandbook.org/en/what-is-open-data>. Accessed: 2016-01-12.
- [3] N. F. Angel, N. Young, and A. Dollman. Creating a database from scratch: Smarter solutions, inc. case study: nifty assignment. *Journal of Computing Sciences in Colleges*, 31(2):185–187, 2015.
- [4] J. Atenas, L. Havemann, and E. Priego. Open data as open educational resources: Towards transversal skills and global citizenship. *Open Praxis*, 7(4):377–389, 2015.
- [5] S. Fincher, M. Kölling, I. Utting, N. Brown, and P. Stevens. Repositories of teaching material and communities of use: nifty assignments and the greenroom. In *Proceedings of the 6th international workshop on Computing education research*, pages 107–114. ACM, 2010.
- [6] W. H. Inmon. *Building the data warehouse*. John Wiley & sons, 2005.
- [7] R. Kimball. *The data warehouse lifecycle toolkit: expert methods for designing, developing, and deploying data warehouses*. John Wiley & Sons, 1998.
- [8] L. Layman, L. Williams, and K. Slaten. Note to self: make assignments meaningful. *ACM SIGCSE Bulletin*, 39(1):459–463, 2007.
- [9] N. Parlante. Nifty assignments. <http://nifty.stanford.edu>. Accessed: 2016-01-12.
- [10] N. Parlante, J. Popyack, S. Reges, S. Weiss, S. Dexter, C. Gurwitz, J. Zachary, and G. Braught. Nifty assignments. In *ACM SIGCSE Bulletin*, volume 35, pages 353–354. ACM, 2003.
- [11] N. Parlante, J. Zelenski, P.-M. Osera, M. Stepp, M. Sherriff, L. Tychonievich, R. Layer, S. J. Matthews, A. Obourn, D. R. Raymond, et al. Nifty assignments. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 673–674. ACM, 2015.
- [12] M. Sahami, S. Roach, E. Cuadros-Vargas, and D. Reed. Computer science curriculum 2013: reviewing the strawman report from the acm/ieee-cs task force. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 3–4. ACM, 2012.
- [13] R. Sooriamurthi. Introducing abstraction and decomposition to novice programmers. In *ACM SIGCSE Bulletin*, volume 41, pages 196–200. ACM, 2009.