Teaching & Learning Notes for this Lesson, GBenoît

Updated GB, Nov 11, 2023.

Building on Paul's pirate motivated explorations, this week's notebook is all about EDA (Agenda point #1). It is designed to review some main points of the lesson presented in the asynch and to provide examples using different data sets (#2 and #3).

It's intended, too, to present to you important related topics that you might want to use, particularly issues of **encoding** (#4). This section may seem extraneous but it's important to be aware of processing data sets you may not have encountered, concept of encoding/decoding, various legacy encoding systems, and file-size reduction (UTF-x *v* ASCII). This will lead, too, to the importance of visualization, fonts, and that *data* are separate from the *presentation* of those data. Details from the bit- through collection-level data are important in both **exploratory data analysis**, **data mining**, **machine-learning**, **compression algorithms**, **data packets**, and **computational-efficiency**

Optionally you may want to review real-world examples spawn from student questions about integrating SQL, demo of a common technique (configuration files), and Unicode's code points, hex, and glyph names conversions in Python. Finally, there are optional examples from [my] work; other instructors may want to present their own work products or research: Salton's general vector model, various full-text corpora projects used in full-text, sentient analysis, natural language processing, and a compression technique using bit signatures, common in biomedical work and networking.

1. What is exploratory data analysis (or EDA)?

The answer depends where you work. The original perspective was from the world of statistics, driven particularly the flagship journal *The American Statistician*. In one famous edition of that journal, the lead article suggested that visuals be used to help (a) to explore and (b) to analyze data. That scandalous opinion led to the common work features of data cleansing and learning about the data set's utility in a project, tempering and qualifying our research questions by what the data might provide, and creating the set of charts and plots we find in every spreadsheet program and computing software package.

2. Visualization

The roots of those plots and visualizations in understanding and presenting data are offered later in the course. A lesson is coming up about visualization. For the moment, note that exploring data requires an understanding of statistics, one's research area (the "problematic"), the specific question(s) one wants to ask, using visuals to uncover oddities and confirm our understanding, before progressing to analysis.

3. Bigger Picture: The relationships between ...

visualization, statistics, data science, public information sources, ethics, philosophy and our coding enjoy a lively engagement in research papers and corporate sales pitches. Take a look at the *AS* (vol. 72, 1, 2018) special edition on Data Science. From the computing world this intersection of efforts appears as a lead article in *ACM* on whether computer science student should be taught graphic design and many questions of EDA in K-12, College, CS curricula.

Data Scientists, then, should be versed in the industry- and research- literature of their domains, along with an understanding of the aesthetics (more than just plotting), the interative, bi-directional flow of data and interpretation, and an open-minded approach to research questions, ethical use of statistics, and transparency in presenting the results.

4. The main purpose of EDA is to ...

1 get to know our data before making assumptions,
2 before finalizing a qualified researchable question,
3 identify "errors" in the data themselves
4 appropriateness or applicability of the data to the questions
5 range and domain of the data
6 and ultimately allow us to extract groups for statistical testing to identify "interesting events"
7 potentially interesting data sub-sets are subjected to further statistical tests and
8 use plots to explore the data as well as later present, describe, explain, and explore questions in a visual language
9 interpreted by the data scientist and a domain expert
10 for insights into the data - new knowledge - with the correct statistical and domain background to warrant our interpretations.

Contents:

- Encoding/Decoding
- Unicode (Class Discussion)
- Byte-Order - remember you don't know what your data might be ... codecs
- User-preferences in encoding
- ASCII versus UTF-8 when reading data

5 Student Questions:

- Pre request added points about MySQL + Python connection example, config.ini
- Using unicodedata.name

6 END OF THE LESSON

7 Advanced Demos

- General Vector Model in Info Retrieval
- Corpora (Brown, Warwick, Reuters, others) and the NLT
- Bit Signatures on-the-job

Encoding and SQL and File Reading

Before any EDA attempt we must know the state of the data - the structure as well as the data themselves. Hence in a worst case scenario, it's necessary to convert all the data to binary and reconvert 'em all to something shareable, such as UTF-8. Note, tho, in late 2023, some text editors (like TextMate) are defaulting to UTF-16BE ("be" means "big end", reading data from left to right [https://www.freecodecamp.org/news/what-is-endianness-big-endian-vs-little-endian/]).

There's also the role of **aesthetics** in contemporary programming. Visualizations are used in exploring the data before a project and used in presenting the results to clients/end-users. They're presented, too, to the public in digital boards. The aesthetic part of the end-user experience is shown to impact trustworthiness and comprehensibility of the data.

Taking data down to "semantic tokens" and applying some weighting scheme in a matching algorithm leads us to large language models, machine learning, retrieval engines, and compression. For example, while retrieval engines usually rely on some model of human language use, evidenced by token frequency and distribution, chat bots and inference engines use the probability of a token's appearance given a previous context (starting with the simplest Bayes's Theorem to the back-propogation loop of ANNs and the knowing introduction of error in genetic algorithms, Markov Models, and other weighting schemes).

In some areas of IR, such as the biological sciences, it's common to create topical signatures for parts of documents to locate candidate matches to queries. These are binary matches; but using fuzzy set models and/or other context to infer moves us more towards AI.