Topic for the class:– Data wrangling
Unit _1 : Title-Data Evolution
Date & Time : 18.7.22 10.00 AM – 10.50 AM

Dr. Bhramaramba Ravi
Professor
Department of Computer Science and Engineering
GITAM School of Technology (GST)
Visakhapatnam – 530045
Email: **bravi@gitam.edu**

Unit1-syllabus

- **UNIT 1 Data Evolution 9 hours, P – 2 hours Data Evolution**: Data to Data Science – Understanding data: Introduction – Type of Data, Data Evolution – Data Sources. Preparing and gathering data and knowledge - Philosophies of data science - data all around us: the virtual wilderness - Data wrangling
-
- : from capture to domestication - Data science in a big data world - Benefits and uses of data science and big data - facets of data.
- https://www.coursera.org/learn/intro-analyticthinking-datascience- datamining?specialization=data-science-fundamentals

Data Wrangling Overview

- Defined as "having a long and complicated dispute."
  - Process of converting data into a format for use by conventional software.
  - A collection of strategies and techniques applied within a project strategy.
  - Not a pre-defined task; each case requires problem-solving.
  - Case study used to illustrate specific techniques and strategies.
- World Record Comparison in Athletics

Case study: best all-time performances in track and field

- World records are often compared based on their age or closeness to breaking.
  - Usain Bolt's 200 m dash world record was 12 years old when he broke it, while Usain Bolt's 100 m world record was less than a year old when he broke it in early 2008.
  - Age of a world record does not necessarily indicate strength, as Bolt's 19.19 sec 200 m mark was not worse than his 19.30 sec mark.
  - The percentage improvement of a mark over the second-best mark is often used as evidence of good performance.
  - However, this is not perfect due to the high variance of second-best performances

Common Heuristics

- Comparisons in Athletics
  - Armchair track and field enthusiasts often compare world records based on their age or closeness to breaking them.
  - Michael Johnson's 200 m dash world record was 12 years old when Usain Bolt broke it.
  - Usain Bolt's 100 m world record was less than a year old when he broke it in early 2008, then again at the 2008 Olympics and 2009 World Championships.
  - The age of a world record does not necessarily indicate strength, as Bolt's 19.19 sec mark for the 200 m was not worse than his 19.30 sec mark.
  - The percentage improvement of a mark over the second-best mark is often used as evidence of good performance, but this is not perfect due to the high variance of second-best performances.

IAAF Scoring Tables: A Historical Perspective

- The IAAF Scoring Tables of Athletics are the most widely accepted method for comparing performance between events in track and field.
  - The IAAF publishes an updated set of points tables every few years.
  - The tables are used in multidiscipline events like men's decathlon and women's heptathlon.
  - The scoring tables for individual events have little effect on competition, except for certain track and field meetings that award prizes based on the tables.
  - The 2008 Scoring Tables gave Usain Bolt's 2009 performance a score of 1374, indicating a dramatic change in his performance.
  - The 2011 tables, based on a relatively small set of the best performances in each event, could have affected the scores in the next update.
  - The 2008 and 2009 track and field seasons produced incredible 100 m performances, which affected the next set of scores, released in 2011.
  - The author aims to use all available data to generate a scoring method that is less sensitive to changes in the best performances and a good predictor of future performance levels.

Comparing Performances Using All Data

- Alltime-athletics.com provides a comprehensive set of elite track and field performances.
  - The site contains thousands of performances in all Olympic events.
  - The aim is to improve the robustness and predictive power of the IAAF's Scoring Tables.
  - Data collection involves web scraping and comparing scores with the IAAF Scoring Tables, available only in PDF.
  - Both web pages and PDFs are not ideal for programmatic parsing due to their messy HTML structure and page headers, footers, and numbers.
  - Two tasks are involved: wrangling the top performance lists from alltime-athletics.com and wrangling the IAAF Scoring Tables from the PDF.

Getting Ready to Wrangle Data

- Advocates for a deliberate approach to data collection.
  - Encourages thorough exploration before writing code or implementing strategies.
  - Provides insights to aid in effective data wrangling.
  - Highlights the potential messiness of messy data.
  - Describes steps to determine data type, actions needed, and potential issues.

Messy Data in Data Science

- Types of Messy Data
  - Each data set is unique, making it challenging to parse and use efficiently.
  - Data scraping involves programmatically pulling selected elements from sources not designed for programmatic access.
  - Corrupted data can be found in poorly formatted or corrupted files, often due to disk errors or other low-level problems.
  - Common corrupted file formats include PST, an email archive.
  - Poorly designed databases can lead to inconsistencies in data sources, such as unmatched database values or keys, in scope, depth, APIs, or schemas.
  - As of 2016, there is still no Era of Clean Data, raising questions about its potential.

Web Scraping for Track and Field Project

- Web scraping is a useful tool for tracking and field data.
- The process involves examining raw HTML and imagining the task as a wrangling script.
- The raw data is what any code will see, and it's crucial to understand how to deal with it.
- The first step in wrangling is to look at the raw data, such as header lines and other material at the top of the page.
- The main goal is to capture the top marks at this point.
- A wrangling script can recognize an athlete's performance by testing each line of the file.
- Document structure, particularly in HTML or XML, can provide clues about where valuable data starts.
- The data is preceded by a tag, which is often on the page or only right before the data set starts.
- The tag is used to denote the beginning of the data set for each of the events.
- The text parser in the scripting language can read the tag, separate the columns into fields, and store each text field in a variable or data structure.

Data Wrangling Challenges and Uncertainties

- Understanding the starting point of valuable data within each HTML page is crucial.
  - Character sequences in raw HTML can cause confusion and potential errors.
  - These sequences are often HTML representations of characters like ü or é.
  - It's important to double-check everything, both manually and programmatically.
  - A quick scroll through post-wrangle, clean data files can reveal obvious mistakes.
  - Extra tab characters can interfere with parsing algorithms, including standard R packages and Excel imports.
  - Every case requires careful consideration of potential parsing errors.
  - Awareness is the most important aspect of data wrangling.

Wrangling Data and File Analysis

- Wrangling scripts start at the beginning of the file and finish at the end, but unexpected changes can occur in the middle.
  • It's crucial to examine the wrangled data file(s) at the beginning, end, and some places in the middle to ensure the expected state.
  • Nonstandard lists of best performances can be found at the bottom of the pages.
  • The HTML tag that denotes the beginning of the desired data is closed at the end of the main list.
  • This tag closure is a good way to end the parsing of the data set.
  • If the wrangling script ignores the end of the useful data set, it may collect nonstandard results at the bottom of the page or fail to know what to do when the data stops fitting the established column format.
  • Looking at the end of the wrangled data file is crucial to determine if the data wrangling was successful.
  • The data scientist should decide which aspects of wrangling are most important and ensure they are completed properly.

Data Wrangling Plan for Track and Field Performance Data

- The process involves imagining oneself as a wrangling script, parsing through raw data, and extracting necessary parts.
  • A potential solution is to download all web pages containing all Olympic track and field events and parse them using HTML structure.
  • However, this requires a list of web addresses for individual events to download programmatically.
  • Each page has a unique address that needs to be copied or typed manually, which could be time-consuming.
  • The author decided not to go with web scraping, instead opting for the post-HTML, already rendered web page version.
  • The author would visit each of the 48 web pages, select all text, copy the text, and paste the text into a separate flat file.
  • This method eliminates the need for translating HTML or scripting the downloading of the pages.
  • The choice of data wrangling plan should depend on all the information discovered during initial investigation.
  • The author suggests pretending to be a wrangling script, imagine what might happen with the data, and then write the script later.

Data Wrangling Techniques and Tools

- Data wrangling is an abstract process with uncertain outcomes.
  - No single tool can clean messy data.
  - Tools are good for various tasks, but no single tool can wrangle arbitrary data.
  - No one application can read arbitrary data with an arbitrary purpose.
  - Data wrangling requires specific tools in specific circumstances.

.

File Format Converters Overview

- Converting from HTML, CSV, PDF, TXT to other file formats.
  - PDF format is not ideal for data analysis.
  - File format converters can convert PDFs to other formats like text files and HTML.
  - Unix application pdf2txt and pdf2html are useful for data scientists.
  - Numerous file format converters are available, many free or open source.
  - Google search can help determine if a file format is easily convertible.

Proprietary Data Wranglers in 2016

- Numerous companies offer data wrangling services at a cost.
  - Many software products claim to be capable of this, but many are limited.
  - Some proprietary products can convert existing data into desired data.
  - The cost of these tools may be worthwhile for early project completion.
  - The industry is young and rapidly changing, making it difficult to conduct a comprehensive survey.

Scripting: Using the Plan, But Then Guess and Check

- Imagine being a script and reading through files to understand the complexity of the task.
  • Use simple tools like the Unix command line for simpler tasks like extracting lines, converting occurrences of a word, and splitting files.
  • For more complex operations, a scripting language like Python or R is recommended.
  • Writing a wrangling script is not a well-orchestrated affair; it involves trying various techniques until finding the best one.
  • The most important capability to strive for when choosing scripting languages or tools is the ability to load, manipulate, write, and transform data quickly.
  • Make informed decisions about how best to wrangle the data or guess and check if it's more time efficient.
  • Consider the manual-versus-automate question: can you wrangle manually in a shorter time than you can write a script?
  • Stay aware of the status of the data, the script, the results, the goals, and what each wrangling step and tool is gaining you.

Common Pitfalls in Wrangling Scripts

- Misuse of messy data can lead to omissions.
  - Even with careful consideration, risks exist.
  - Observance and thorough consideration are crucial.
  - Symptoms of a wrangling script falling into a pitfalls are provided.

Data Incompatibilities in Windows/Mac/Linux

- Major operating systems still have disagreements on line endings in text files.
  - Unix and Linux have used line feed (LF) denotation for new lines since the 1970s.
  - Mac OS before version 9.0 used carriage return (CR) character for new lines.
  - Mac OS joined Unix derivatives in using line feed since 1999, but Microsoft Windows uses a hybrid CR+LF line ending.
  - Improper line ending parsing can lead to various problems.
  - Each programming language has its own capabilities for reading different file types.
  - OS file formats include : more lines of text than expected, too few lines of text, and interspersed weird-looking characters.

Understanding Escape Characters in Text Processing

- Understanding Escape Characters
  - Unix or Linux shells use the character * as a wildcard, representing all files in the current directory.
  - The backslash character, '*', removes this special meaning, representing only the simple asterisk character.
  - These escape characters can occur in text files and text/string variables in programming languages.

Examples of Escape Characters

- A text file with three lines of text, followed by a line containing no characters and the third line with tab characters in between the letters, would be read by Python and R.
  - The line breaks have been replaced by 'n' and the tab characters by 't'.
  - A single string variable can represent the data from an entire file, with each line separated by the escaped newline character.

Using Escape Characters Within Quotations or Quotations Within Quotations

- The text itself contains quotation marks, which need to be escaped within the string variable.
  - For example, a text file containing emails could be encoded as a string variable, escaping the internal quotations.

Implications of Complex Escapes

- Complex escapes can be confusing, especially when dealing with many nested quotation marks and newlines.
  - Symptoms of escaping problems include lines or strings being too long or short, trying to read a file line by line but end up with one long line, finding text inside quotation marks that doesn't belong there, and getting errors while reading or writing files.

Outliers in Data Analysis

- Incorrect data can sneak into projects without causing an error or making itself obvious.
  - Summary statistics and exploratory graphs can help catch these errors.
  - Checking the range of values—minimum to maximum—could catch the error.
  - Plotting histograms of all data can help check for errors and gain awareness about the data sets.
  - Generating statistical or visual summaries can prevent errors and promote awareness.
  - Techniques like basic descriptive statistics, summaries, and diagnostics can be used to ensure successful data assessment.

THANK YOU