A BRIEF GUIDE TO

DATA CLEANING

THE COMPANY

The Artists of Data Science

THE CURATOR

Harpreet Sahota

The Artists of Data Science
with Harpreet Sahota

THE ARTISTS
OF
DATA SCIENCE

WHY CLEAN YOUR DATA?

Knowing how to clean your data is advantageous for many reasons. Here are just a few:

- It prevents you from wasting time on wobbly or even faulty analysis
- It prevents you from making the wrong conclusions, which would make you look bad!
- It makes your analysis run faster. Correct, properly cleaned and formatted data speed up computation in advanced algorithms
-

WHAT THIS GUIDE IS FOR

This guide will take you through the process of getting your hands dirty with cleaning data. We will dive into the practical aspects and little details that make the big picture shine brighter.

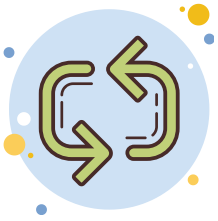
DATA CLEANING IS A 3-STEP PROCESS

STEP 1: FIND THE DIRT

Start data cleaning by determining what is wrong with your data.

STEP 2: SCRUB THE DIRT

Depending on the type of data dirt you're facing, you'll need different cleaning techniques. This is the most intensive step.



STEP 3: RINSE AND REPEAT

Once cleaned, you repeat steps 1 and 2.

The Artists of Data Science
with Harpreet Sahota

STEP 1: FIND THE DIRT

Start data cleaning by determining what is wrong with your data.

Look for the following:

- **Are there rows with empty values? Entire columns with no data? Which data is missing and why?**
- **How is data distributed? Remember, visualizations are your friends. Plot outliers. Check distributions to see which groups or ranges are more heavily represented in your dataset.**
- **Keep an eye out for the weird: are there impossible values? Like “date of birth: male”, “address: -1234”.**
- **Is your data consistent? Why are the same product names written in uppercase and other times in camelCase?**

Wear your detective hat and jot down everything interesting, surprising or even weird.

THE ARTISTS
OF
DATA SCIENCE

STEP 1: SCRUB THE DIRT

Knowing the problem is half the battle.

The other half is solving it.

How do you solve it, though?

One ring might rule them all, but one approach is not going to cut it with all your data cleaning problems.

Depending on the type of data dirt you're facing, you'll need different cleaning techniques.

Step 2 is broken down into eight parts:

- **Missing Data**
- **Outliers**
- **Contaminated Data**
- **Inconsistent Data**
- **Invalid Data**
- **Duplicate Data**
- **Data Type Issues**
- **Structural Errors**

The Artists of Data Science
with Harpreet Sahota

THE ARTISTS
OF
DATA SCIENCE

STEP 2.1: MISSING DATA

Sometimes you will have rows with missing values. Sometimes, almost entire columns will be empty.

What to do with missing data? Ignoring it is like ignoring the holes in your boat while at sea - you'll sink.

Start by spotting all the different disguises missing data wears. It appears in values such as 0, "0", empty strings, "Not Applicable", "NA", "#NA", None, NaN, NULL or Inf. Programmers before you might have put default values instead of missing data ("email@company.com").

When you have a general idea of what your missing data looks like, it is time to answer the crucial question:

"Is missing data telling me something valuable?"

There are 3 main approaches to cleaning missing data:

1. **Drop rows and/or columns with missing data. If the missing data is not valuable, just drop the rows (i.e. specific customers, sensor reading, or other individual exemplars) from your analysis. If entire columns are filled with missing data, drop them as well. There is no need to analyze the column "Quantity of NewAwesomeProduct Bought" if no one has bought it yet.**
2. **Recode missing data into a different format. Numerical computations can break down with missing data. Recoding missing values into a different column saves the day. For example, the column "payment_date" with empty rows can be recoded into a column "payed_yet" with 0 for "no" and 1 for "yes".**
3. **Fill in missing values with "best guesses." Use moving averages and backfilling to estimate the most probable values of data at that point. This is especially crucial for time-series analyses, where missing data can distort your conclusions.**

THE ARTISTS
OF
DATA SCIENCE

STEP 2.2: OUTLIERS

Outliers are data points which are at an extreme.

They usually have very high or very low values:

- **An antarctic sensor reading the temperature of 100°**
- **A customer who buys $0.01 worth of merchandise per year**

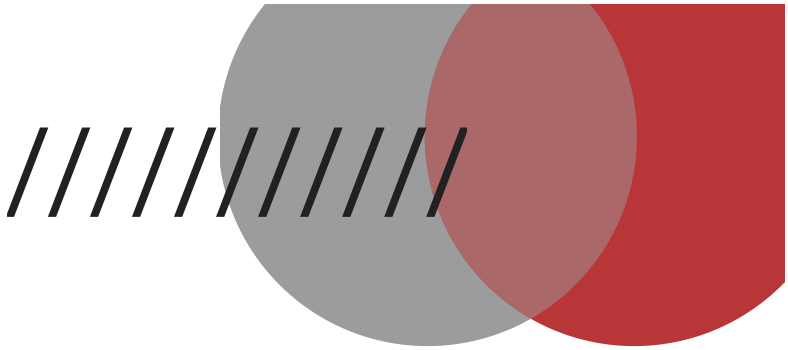
How to interpret those?

Outliers usually signify either very interesting behavior or a broken collection process.

Both are valuable information (hey, check your sensors, before checking your outliers), but proceed with cleaning only if the behavior is actually interesting.

There are three approaches to dealing with outliers:

1. **Remove outliers from the analysis. Having outliers can mess up your analysis by bringing the averages up or down and in general distorting your statistics. Remove them by removing the upper and lower X-percentile of your data.**
2. **Segment data so outliers are in a separate group. Put all the "normal-looking" data in one group, and outliers in another. This is especially useful for analysis of interest. You might find out that your highest paying customers, who actually buy 3 times above average, are an interesting target for marketing and sales.**
3. **Keep outliers, but use different statistical methods for analysis. Weighted means (which put more weight on the "normal" part of the distribution) and trimmed means are two common approaches of analyzing datasets with outliers, without suffering the negative consequences of outliers.**



STEP 2.3: CONTAMINATED DATA

Contaminated data is another red flag for your collection process.

Examples of contaminated data include:

- Wind turbine data in your water plant dataset.
- Purchase information in your customer address dataset.
- Future data in your current event time-series data.

The last one is particularly sneaky.

Imagine having a row of financial trading information for each day.

Columns (or features) would include the date, asset type, asking price, selling price, the difference in asking price from yesterday, the average asking price for this quarter.

The average asking price for this quarter is the source of contamination.

You can only compute the averages once the quarter is over, but that information would not be given to you on the trading date - thus introducing future data, which contaminates the present data.

With corrupted data, there is not much you can do except for removing it. This requires a lot of domain expertise.

When lacking domain knowledge, consult non-analytical members of your team. Make sure to also fix any leakages your data collection pipeline has so that the data corruption does not repeat with future data collection.

STEP 2.4: INCONSISTENT DATA

"Wait, did we sell 'Apples', 'apples', or 'APPLES' this month? And what is this 'monitor stand' for $999 under the same product ID?"

You have to expect inconsistency in your data.

Especially when there is a higher possibility of human error (e.g. when salespeople enter the product info on proforma invoices manually).

The best way to spot inconsistent representations of the same elements in your database is to visualize them.

Plot bar charts per product category.

Do a count of rows by category if this is easier.

When you spot the inconsistency, standardize all elements into the same format.

Humans might understand that 'apples' is the same as 'Apples' (capitalization) which is the same as 'appels' (misspelling), but computers think those three refer to three different things altogether.

Lowercasing as default and correcting typos are your friends here.

STEP 2.5: INVALID DATA

Similarly to corrupted data, invalid data is illogical.

For example, users who spend -2 hours on our app, or a person whose age is 170.

Unlike corrupted data, invalid data does not result from faulty collection processes, but from issues with data processing (usually during feature preparation or data cleaning).

Let us walk through an example:

You are preparing a report for your CEO about the average time spent in your recently launched mobile app.

Everything works fine, the activities time looks great, except for a couple of rogue examples.

You notice some users spent -22 hours in the app. Digging deeper, you go to the source of this anomaly.

In-app time is calculated as finish_hour - start_hour. In other words, someone who started using the app at 23:00 and finished at 01:00 in the morning would have for their time_in_app -22 hours (1 - 23 = - 22).

Upon realizing that, you can correct the computations to prevent such illogical data.

Cleaning invalid data mostly means amending the functions and transformations which caused the data to be invalid. If this is not possible, we remove the invalid data.

THE ARTISTS
OF
DATA SCIENCE

STEP 2.6: DUPLICATE DATA

Duplicate data means the same values repeating for an observation point.

This is damaging to our analysis because it can either deflate/inflate our numbers (e.g. we count more customers than there actually are, or the average changes because some values are more often represented).

There are different sources of duplicate data:

- Data are combined from different sources, and each source brings in the same data to our database.
- The user might submit information twice by clicking on the submit button.
- Our data collection code is off and inserts the same records multiple times.

There are three ways to eliminate duplicates:

1. Find the same records and delete all but one.
2. Pairwise match records, compare them and take the most relevant one (e.g. the most recent one)
3. Combine the records into entities via clustering (e.g. the cluster of information about customer Harpreet Sahota, which has all the data associated with it).

The Artists of Data Science
with Harpreet Sahota

STEP 2.7: DATA TYPE ISSUES

Depending on which data type you work with (DateTime objects, strings, integers, decimals or floats), you can encounter problems specific to data types.

2.7.1 Cleaning string

Strings are usually the messiest part of data cleaning because they are often human-generated and hence prone to errors.

The common cleaning techniques for strings involve:

- Standardizing casing across the strings
- Removing whitespace and newlines
- Removing stop words (for some linguistic analyses)
- Hot-encoding categorical variables represented as strings
- Correcting typos
- Standardizing encodings

Especially the last one can cause a lot of problems. Encodings are the way of translating between the 0's and 1's of computers and the human-readable representation of text.

And as there are different languages, there are different encodings.

Everyone has seen strings of the type ����. Which meant our browser or computer could not decode the string. It is the same as trying to play a cassette on your gramophone. Both are made for music, but they represent it in different ways.

When in doubt, go for UTF-8 as your encoding standard.

STEP 2.7: DATA TYPE ISSUES

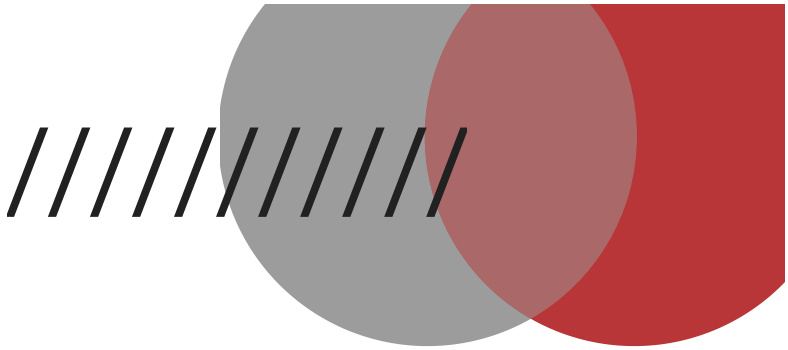
2.7.2 Cleaning date and time

Dates and time can be tricky. Sometimes the error is not apparent until doing computations (like the activity duration example above) on date and times.

The cleaning process involves:

-
- Making sure that all your dates and times are either a DateTime object or a Unix timestamp (via type coercion). Do not be tricked by strings pretending to be a DateTime object, like "24 Oct 2019". Check for data type and coerce where necessary.

- Internationalization and time zones. DateTime objects are often recorded with the time zone or without one. Either of those can cause problems. If you are doing region-specific analysis, make sure to have DateTime in the correct timezone. If you do not care about internationalization, convert all DateTime objects to your timezone.



STEP 2.8: STRUCTURAL ERRORS

Even though we treated data issues comprehensively, there is a class of problems with data, which arise due to structural errors.

Structural errors arise during measurement, data transfer, or other situations.

Structural errors can lead to inconsistent data, data duplication, or contamination.

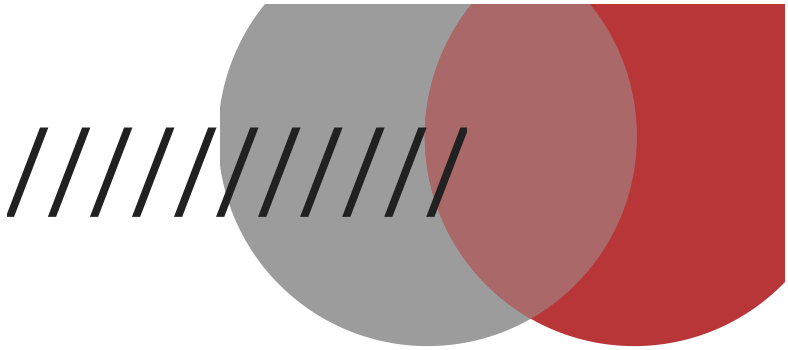
But unlike the treatment advised above, you are not going to solve structural errors by applying cleaning techniques to them.

Because you can clean the data all you want, but at the next import, the structural errors will produce unreliable data again.

Structural errors are given special treatment to emphasize that a lot of data cleaning is about preventing data issues rather than resolving data issues.

So you need to review your engineering best practices.

Check your ETL pipeline and how you collect and transform data from their raw data sources to identify where the source of structural errors is and remove it.



STEP 3: RINSE AND REPEAT

Once cleaned, you repeat steps 1 and 2.

This is helpful for three reasons:

1. **You might have missed something. Repeating the cleaning process helps you catch those pesky hidden issues.**
2. **Through cleaning, you discover new issues. For example, once you removed outliers from your dataset, you noticed that data is not bell-shaped anymore and needs reshaping before you can analyze it.**
3. **You learn more about your data. Every time you sweep through your dataset and look at the distributions of values, you learn more about your data, which gives you hunches as to what to analyze.**

Data scientists spend 80% of their time cleaning and organizing data because of the associated benefits.

Or as the old machine learning wisdom goes:

Garbage in, garbage out.

All algorithms can do is spot patterns.

And if they need to spot patterns in a mess, they are going to return "mess" as the governing pattern.

Clean data beats fancy algorithms any day.

But cleaning data is not in the sole domain of data science. High-quality data are necessary for any type of decision-making.

From startups launching the next Google search algorithm to business enterprises relying on Microsoft Excel for their business intelligence - clean data is the pillar upon which data-driven decision-making rests.

THE ARTISTS
OF
DATA SCIENCE

AUTOMATE YOUR DATA CLEANING

By now it is clear how important data cleaning is.

But it still takes way too long. And it is not the most intellectually stimulating challenge.

To avoid losing time, while not neglecting the data cleaning process, data practitioners automate a lot of repetitive cleaning tasks.

Mainly there are two branches of data cleaning that you can automate:

- Problem discovery. Use any visualization tools that allow you to quickly visualize missing values and different data distributions.

- Transforming data into the desired form. The majority of data cleaning is running reusable scripts, which perform the same sequence of actions. For example: 1) lowercase all strings, 2) remove whitespace, 3) break down strings into words.

Whether automation is your cup of tea or not, remember the main steps when cleaning data:

1. Identify the problematic data
2. Clean the data
3. Remove, encode, fill in any missing data
4. Remove outliers or analyze them separately
5. Purge contaminated data and correct leaking pipelines
6. Standardize inconsistent data
7. Check if your data makes sense (is valid)
8. Deduplicate multiple records of the same dataForesee and prevent type issues (string issues, DateTime issues)
9. Remove engineering errors (aka structural errors)
10. Rinse and repeat

Keep a list of those steps by your side and make sure your data gives you the valuable insights you need.

The Artists of Data Science
with Harpreet Sahota