

Chapter

Data Quality Measurement Based on Domain-Specific Information

Yury Chernov

Abstract

Over the past decades, the topic of data quality became extremely important in various application fields. Originally developed for data warehouses, it received a strong push with the big data concept and artificial intelligence systems. In the presented chapter, we are looking at traditional data quality dimensions, which mainly have a more technical nature. However, we concentrate mostly on the idea of defining a single data quality determinant, which does not substitute the dimensions but allows us to look at the data quality from the point of view of users and particular applications. We consider this approach, which is known as a fit-to-use indicator, in two domains. The first one is the test data for complicated multi-component software systems on the example of a stock exchange. The second domain is scientific research on the example of validation of handwriting psychology. We demonstrate how the fit-to-use determinant of data quality can be defined and formalized and what benefit to the improvement of data quality it can give.

Keywords: data quality, quality metrics, fit-to-use determinant, data warehouse, formalization, software application, test data, stock exchange, reference data, validation, handwriting psychology

1. Introduction

Over the past decades, data quality is getting more and more relevant and important both in science and practice. The topic is well known, therefore, multiple researchers and data practitioners have been intensively investigating different aspects of data quality. There are numerous publications and projects. Especially data-driven design, data-driven management, and the “big data” concept are drawing attention to the data quality issues. High quality is a prerequisite for success. The growth of BI tools (business intelligence) like Tableau or Power BI, market intelligence tools like NetBase Quid or Crunchbase Pro, A/B testing tools like Optimizely or HubSpot, etc. reflects the trend that decisions become more and more data-driven. Naturally, the requirements for data quality are permanently growing. Today, data quality is defined not only as a struggle against duplicates, outliers, missing data, corrupted text, or typos. It is a much more complicated concept.

However, the definition of data quality itself is not easy and always is a bit ambiguous. It depends on the viewpoint, aim, and context. Traditionally, data analysts define a set of data quality dimensions. Meanwhile, there are many dozens of them. Their

concepts often overlap and repeat themselves under different terms. These dimensions reflect different aspects of data quality. Most of them are well formalized and can be quantified. Quantifying is essential for planning improvement measures of data quality in different contexts.

The most popular context is natural data warehouse. Therefore, many dimensions appeared namely in this context. However, many other fields are not less influenced by data quality. The idea to develop a single quantitative indicator, which is perhaps more abstract, has been known for a long time. It seems reasonable and practical. Such a determinant is strongly domain-specific. It reflects specifics of the particular system and it can serve as a good instrument for comparison datasets.

2. Data quality dimensions

The standard approach to the definition of data quality in terms of the different aspects, which are traditionally termed dimensions, is reflected in numerous publications. Data quality dimensions were captured hierarchically in the very often referred study [1]. The study was based on a customer survey that treated data as a product. The authors defined four categories, which are still reasonable, although the study was done about 30 years ago. These categories formed the first level of the hierarchy. The data quality dimensions built the second level. Below we are following this approach and preserving the categories. However, the dimensions themselves have been modified, reflecting the development of data science and our perception of the topic. It is difficult to identify the actual source of a particular dimension. Many authors are speaking about the same dimensions often naming them differently. The review below is based mainly on several publications [2–8] that were analyzed explicitly. However, numerous additional publications, which the author studied, read, or skimmed, influenced this list as well.

2.1 Intrinsic category

2.1.1 Correctness

Data correctness or accuracy refers to the degree to which data represents real objects. In many cases to evaluate correctness, the data are compared to some reference sources. There can be different reference sources. For instance, a natural restriction to the data value (the age can be between 0 and 120 years), a certain rule (the sum of percentage values should be 100), a related database record, a calculated value, etc. When we try to formalize quality then correctness can be defined as a measure of the proximity of a data value to a referenced value, which is considered correct. If a reference source is available, an automated process of correctness evaluation can be established.

2.1.2 Validity

Validity refers to the degree to which data values comply with rules defined for the system. These can be external rules, for instance, regulation in the finance area, or internal system rules. Validity has associations with the correctness, completeness, and consistency of the data. However, data values can be valid but

not accurate, or they can be valid but not completed. Examples of non-valid data entities are a birth date, which is not in the range of valid dates, or a city, which is not in the list of cities.

2.1.3 Uniqueness/deduplication

Uniqueness means that no duplications or redundant information are overlapping across all the datasets of the system. It means that entities modeled in the system are captured and represented possibly only once within the proper component or the database segment. Uniqueness ensures that no entity exists more than once within the data. It possesses a unique key within the data set. For instance, in a master product table or person table, each product or person appears once and this entity is assigned a unique identifier. This identifier represents the product or person across the whole system. If additional instances of this product or person are created in different parts of the system, they preserve the unique identifier.

Uniqueness can be monitored statically by periodic duplicate analyses of the data or dynamically when capturing new entities. Periodic checks of the data consistency are a typical task in every data warehouse. Dynamic verifications are often built into a database as triggers and restrictions on fields. If there is a combination of numerous databases, files, and other data collection facilities, then special procedures must be developed. When some problems are detected, analysts use data cleansing and deduplication procedures to address the issue. Formal uniqueness is rather easy to ensure. More complicated are the cases when the same data are named and defined differently – formal procedures can hardly help. Artificial intelligence methods of data analysis could be useful to identify logical duplication or overlapping.

2.1.4 Integrity (referential integrity)

When we assign unique identifiers to different objects (customers, products, etc.) within our system, we simplify the management of the data. At the same time, that automatically introduces the requirement, that this object identifier is used as a foreign key within the whole data set. This is referred to as referential integrity. Rules associated with referential integrity are constraints against duplication and non-consistency.

2.1.5 Reliability/consistency

Data reliability refers to two aspects. The first aspect relates to the functioning of different data sources in the system. It should be ensured, that regardless of what source collects the particular data or where it resides, this data cannot contradict a value, which resides in a different source or is collected by a different component of the system. The second aspect relates to the closeness of the initial data value to the subsequent data value.

2.1.6 Data decay

That is the measure of the rate of negative change to data. The old values taken from different sources become outdated with time. A source can be decommissioned and a new one not applied yet. For instance, biodata, mobile numbers, and emails of persons can be not valid anymore.

2.1.7 Objectivity

It reflects the extent to which information is unbiased, unprejudiced, and impartial.

2.1.8 Reputation

It means the extent to which users regard the information in terms of source and/or content.

2.2 Contextual category

2.2.1 Completeness

The dimension means that certain attributes should be assigned values. Completeness rules are based on the following three constrain levels:

- Mandatory attributes that require a value (for instance, a family name by a person data).
- Optional attributes, which may have a value based on some conditions (for instance, the education level of a person).
- Inapplicable attributes, which may not have a value (for instance, a maiden name for a single male).

Completeness or incompleteness can be measured through the amount of data that does not have values. The decisive is to which extent the system can perform its tasks with an uncompleted data set.

2.2.2 Data coverage

Data coverage actually reflects the second aspect of completeness, namely the completeness of records. It is the degree to which all required records in the dataset are present. Sometimes data coverage is understood as a measure of availability and comprehensiveness of data compared to the “total data universe.” However, this is not practical and could hardly be quantified.

2.2.3 Amount of data

It reflects the extent to which the volume or quantity of available data is appropriate for the tasks.

2.2.4 Effectiveness or usefulness

It reflects the capability of the data set to enable users to achieve specified goals or fulfill specified tasks with the accuracy and completeness required in the context of use. Sometimes this dimension is called the relevancy or reasonability of data.

2.2.5 Efficiency

Efficiency reflects the extent to which data can quickly meet the needs of users.

2.2.6 Timeliness (currency)

Timeliness has two aspects. As data currency, it refers to the degree to which data is up-to-date and to the extent to which data are correct despite possible time-related changes.

2.2.7 Timeliness (availability)

This second aspect of timeliness refers to the extent to which data are available in the expected time frame. It can be measured as the time difference between when information is expected and when it is available.

2.2.8 Credibility

It reflects the degree to which data values are regarded as true and believable by users and data consumers.

2.2.9 Ease of manipulation

The dimension reflects the extent to which data are easy to manipulate and apply to different formats.

2.2.10 Maintainability

Maintainability is the measure of the degree to which data can be easily updated, maintained, and managed.

2.3 Representational category

2.3.1 Interpretability

The degree to which data are presented in an appropriate language, symbols, and units of measure.

2.3.2 Consistency

Consistency reflects the plausibility of data values. That is the extent to which data is presented in the same format within a record, a data file, or a database and that semantic rules are preserved all over the system. Consistency is practically the measure of the equivalence of information in various data stores and applications.

2.3.3 Conciseness

This dimension reflects how compact information is. The extent to which it is compactly represented without losing completeness.

2.3.4 Conformance/alignment

This dimension refers to whether data are stored and presented in a format that is consistent with the domain values.

2.3.5 Usability

This dimension is rather generic. It reflects the extent to which information is clear and easily used. It includes as well understandability, that is, the degree to which data have attributes that enable them to be read and interpreted by users.

2.4 Access category

2.4.1 Availability/accessibility

The dimension reflects the ease, with which data can be consulted or retrieved by users or programs.

2.4.2 Confidentiality

The degree to which disclosure of data should be restricted to authorized users. Relates to the security dimension.

2.4.3 Security

The dimension reflects the degree to which access to information is appropriately restricted.

Traceability.

It reflects to which extent data lineage is available. That is the possibility to identify the source of data and transformations they have passed.

3. The fit-for-use and domain-specific data quality determinant

Traditional dimensions of the data quality are good, since they reflect different aspects of data and are rather formal, that is, they can be in most cases automatically evaluated. However, they, first, are often derived from the data warehouse concept [9, 10] and are not always suitable in a different context. Secondly, they are good for homogeneous software systems, where they can be rather easily applied. However, they cannot be directly used for distributed heterogeneous systems, which is often the case, or for special applications, such as scientific research. Both examples we are presenting in the following text.

That is why already long ago they were speaking about the generalized fit-for-use data quality determinant [11, 12], which is close to the view of data users. That was summarized in [2]: “In general, data can be considered of high quality if the data is fit to serve a purpose in a given context.” A data user can be a person, a group of people, an organization, or a software system. We consider this indicator the most important in many practical cases. Often it dominates even the formal nonconformity of a product by quality management.

Fit-for-use is a rather subjective concept. However, in the data quality context, we can provide the required formalism to make it quantitative. To enable that, we need to define a good metric. Such a metric cannot be universal—it is always context-dependent or domain-specific. However, the requirements for a data quality metric can be generic. Every good metric should answer them. In the next paragraph, we are looking at such requirements.

3.1 Requirements for Data quality metrics

In [13] authors formulated the set of requirements, which are appropriate for domain-specific data quality metrics. It includes five basic requirements:

- Normalization
- Cardinality
- Adaptivity
- Scalability (in the original publication they call it “Ability of being aggregated”)
- Interpretability

Normalization should be adequate to assure that results can be interpreted and compared. That means the metric determinants should be on the same scale, which is preferably a relative one. That is important since we use data quality metrics to compare different data sets to each other and select an optimal one (our application case on testing data), to understand the trend of changes in time, or to evaluate the fitness of data for the deduced results of a scientific study (our application case on validation of handwriting analysis).

Cardinality in our context means that the metric should be highly differentiated, that is, it should ensure many possible values and not restrict itself to a rough evaluation. The sensitivity of the metrics should be good enough to capture even small differences.

Adaptability means that the metric must be easily adapted to a particular application. It should be tied to business-oriented goals. That requirement is actually the basis for the fit-to-use data quality determinant.

Scalability means that it should be possible to measure the whole system as well as its components or sub-systems. It can concern, for instance, different layers of data.

Interpretability means that the metric should be clear and simple. That means mean that a user understands the metric and that it is comprehensible and meaningful. In particular, simple metrics should be easily formalized and possibly automatically deduced from the system.

These requirements are rather technical ones. Many of the data quality dimensions mentioned above do answer them. However, sometimes and especially by fit-to-use determinants, some compromises are necessary. Metrics are needed for quantifying data quality in order to answer questions regarding the data sets and to work out the measures to improve data quality in particular domains.

4. Application case 1. Reference data of the end-to-end test system of the stock exchange

The current application case is based on the author's experience at the Swiss Stock Exchange [14]. However, the model is rather generic and it could be valid for any other financial stock exchange or other applications.

4.1 Reference data

The application case covers the quality of the reference data for the test system of the Swiss Stock Exchange. The requirements for the correctness and reliability of the system are extremely high. That is why testing plays a crucial role during the delivery of new functions into production. The controlled testing is carried out at least at four levels: component, integration, system, and end-to-end testing (unit testing is done by developers before they officially release the code). By saying "controlled" I mean that the software is delivered and built in a code control system, it has an official version number, and is installed in a controlled testing environment either automatically (DevOps) or by environment supporting stuff, that is, not by developers themselves.

Test data fully reflects the production system and consists of three parts:

- Reference (static) data remain stable—the changes can be done on a daily basis, not in real-time modus
- Configuration data, mainly technical configuration of different components, that is, IP addresses, the distribution of components over servers, timeouts, etc. They enable the system to run in the testing environment. Configuration data naturally differ from the production system. For instance, some components can share the same servers to save expensive hardware and licenses, which are not allowed in production, where reliability is the major criterion.
- Trading data are generated in real-time for testing purposes. That is done mostly using test automation scripts. However, manual intervention is as well possible.

The reference data is very important. It defines the quality of the whole testing. In the current application case, we are speaking about end-to-end testing, which is rather business than technical oriented. Therefore, the major users and customers of the test system and correspondingly end-to-end testers and business experts.

The test system reflects almost fully the real production configuration. It consists of two dozen components, which are distributed among many application servers (Linux and Windows) and multiple databases (Oracle, MS SQL, Postgres, MySQL, SQLite, and 4D). All components can be divided into three categories:

- Upstream components ensure the interface to the customers and enable the data entering.
- Trading engines, where the on-book and off-book operations take place.
- Downstream systems, such as a data warehouse or a supervision component.

The reference data in the production system is maintained partly by the system customers (banks and other trading organizations) and partly by the market supporting staff of the stock exchange. The data changes are maintained and distributed on a daily base—the reference data maintenance is not a real-time functionality. It is entered using different tools and interfaces and then is transferred into a central data repository, from which is distributed among all relevant system components. In components, the reference data can be enriched to enable more tests.

Test reference data must ensure complete test coverage. To do that, the testware is maintained in Jira and consists of the test requirements that reflect the system functional requirements, test specifications (or test plans), and test cases, which are the elements of test plans and test executions—the results of the testing for a specific project, a test cycle or a sprint. A big portion of the test cases or some steps of them are automated. Several examples of test cases regarding reference data are:

- List a new product/security of type bond.
- Update particular trading parameters of security of type share.
- Change the delisting date of security of type derivative.
- Add a new trading organization.
- Update trading access of an existing trading participant.
- Add a new market holiday.

Test cases are the major objects, which are relevant for the data quality evaluation. The testware includes both the new functionality and the regression testing, which should ensure that the current functions are not affected by new versions of software. Test data must, first, cover all existing business requirements and, secondly, additional functions that are technically possible, but are not used yet in production. They can in principle be activated later, and therefore, must be as well checked. Additionally, the data must support so-called negative test cases to test the reaction of the system to the wrongly entered data.

Reference data must be tested, since they first together with the trading data are provided to the end customers and, secondly, they are the base for the generation of the trading data (orders, trades, transaction reports, indices, etc.)

The data quality evaluation assumes that the test cases are designed properly, that is, they cover all needed functions, business-relevant cases, and configurations. A test case may reflect in this context either a business case or a certain business configuration.

4.2 Data quality determinant for reference test data

Assuming that the test case design is appropriate, we can define the following usability quality metrics.

$$q = \frac{\sum_1^n c_i b_i}{\sum_1^n c_i} \quad (1)$$

where q - data quality determinant; n - number of test cases; c_i - the weight of i -th test case; and b_i - the indicator of test case coverage by the test data (0 or 1).

The model is simple, but it is practically very useful. It fulfills the requirements mentioned above. Maybe just cardinality is fulfilled partly since the differentiability of the model is not perfect, we can get the same value of q for different weight and coverage combinations. To improve the differentiability, the determinant must be done more complex. But that will reduce the clearness and simplicity, that is, will deteriorate interpretability.

The value of the quality determinant is used to compare different test data sets to each other and to ensure the required test quality. Test data sets differ when they are applied in different test environments or at different project phases. The last is probably the most important. Therefore, if we see that the determinant in the current project (project phase) is lower than in the previous project (project phase), it is a clear requirement for additional data enrichment.

Two aspects are important: the definition of the test case weights and the evaluation (preferable automatic) of the test coverage indicator.

4.2.1 Test case weights

The test case weights are assigned by a test designer or automatically. The automation is based on the assignment of the same weight to all test cases in a certain group, typically in the same test area or test suite. Formally, weight is defined on the continuous interval from 0 to 1. Practically the following values are used: 1.0 (required), 0.75 (important), 0.5 (quite important), 0.25 (not important), 0 (not relevant). The following factors influence the weight:

- Business relevance
- Test automation
- Test case complexity
- Execution effort

Business relevance: Test cases that are more important for business should have higher weights. This aspect of test case prioritizing is covered in the publications as customer requirement-based techniques [15–18]. For instance, the issuing of a new share happens on the stock exchange four-six times a year, and at the same time, banks are listing hundreds of derivatives and structured products daily. The last case is much more important and test-relevant. Another example is that adding a new trading participant (of which there are several hundred) has a higher weight than adding a new clearing organization, which could happen once in several years. Business relevance depends on the current project. The new functions that are being introduced by the current project may have higher priority over the regression test cases, and they receive lower priority when the project is over since they become regression ones.

Test automation: Automated regression test cases have higher priority over manual ones since they check the basic functions and must be always successful. Another reason is that their execution is quicker and simpler. Therefore, they should get a weight value of 1.0.

Test case complexity: Simpler test cases should have a higher weight since the data for them could be easier maintained. Generally, a good design should lead to simple and unambiguous test cases. That is, very complex ones are in any case a bit “suspicious” and probably require re-design. They can be, for instance, broken into several simple ones.

Execution effort: Like with the complexity, test cases that require less effort should have higher priority and correspondingly higher weight by the evaluation of the test-data quality.

The initial setting of the weights requires a big effort. However, it should be generally done once and then just be maintained when new test cases are developed, or/and the system functionally is changed. That happens not very often—typically twice a year with big releases.

4.2.2 Indicator of test case coverage

The indicator of test coverage has only two values—0 or 1. When a test case cannot be executed because of the missed data then it gets the value 0. When the test case can be executed or has another problem, like a bug in the software or not implemented functionality, the indicator gets the value of 1.

The value can be assigned manually, like the weight or automatically based on the results of the test execution. For every test cycle (sprint) a test execution dashboard is defined in Jira. It includes the planned test cases and the results of the execution. The results can be passed, failed, not applicable, etc. If the result is “blocked” that means that the test case is blocked by the missing data. This information can be retrieved and used for evaluation.

4.2.3 Testware status

The current snapshot of the major end-to-end test specifications of the Swiss Stock exchange is shown in **Table 1**. The total number of test cases is 2473. Of course, that changes with new development, new projects, and corresponding versions of components.

Most test cases in the first two specifications are automated. According to the above-described logic, they get the weight value of 1.0. In the rest of the testware, some 30% of test cases have as well weight 1.0; approximately 30% - weight 0.75;

Test specification	No test cases
On-book trading	799
Off-book trading	234
Reference data	523
Post trading	137
Clearing & Settlement	186
DWH & Billing	264
Instrument submission	196
Market monitoring sanity	134

Table 1.
Testware volumes.

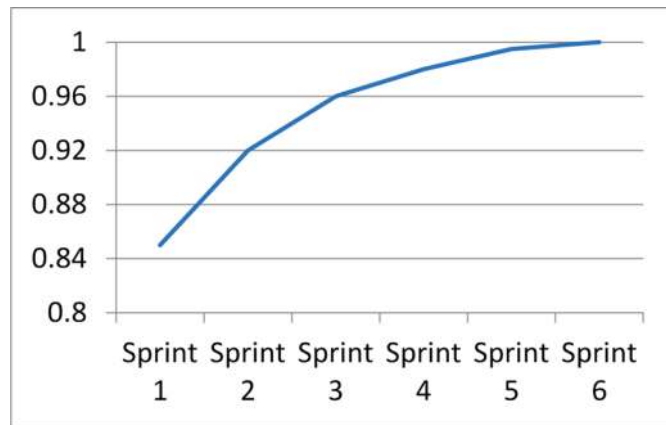


Figure 1.
Data quality dynamics.

20% - weight 0.5; and 20% - weight 0.25. This can as well differ from project to project. When a project includes software changes even in less important areas, they should be tested more thoroughly and their weight becomes higher. On the other hand, if important functionality is not affected, the test case may get a lower weight. The sum of weights in the example from **table 1** is 1929. That gives us, for instance, for the case with 20% uncovered test cases (with the weight of 1.0), what is realistic at the initial phases of a project, $Q = 0.86$.

The typical dynamic of the data quality (data quality determinant) along test cycles or sprints (when a project is being done with agile methodology) is shown in **Figure 1**.

5. Application case 2. Validation of handwriting psychology

The second application case relates to the area of scientific research. Data quality is not often treated formally by research experiments in psychology. Although it plays a very important role in the success and reliability of the results. An approach to quantify and model this was done by the author in the area of validation studies of handwriting analysis [19].

5.1 Handwriting analysis as a psychometric instrument

Handwriting analysis (handwriting psychology) is one of the so-called projective techniques for psychological assessment. It is based on the evaluation of a person's handwriting and deducing from it a range of personality traits. It is traditionally used for recruitment and in some specific areas where the typically used self-assessment tools are not applicable. For instance, in forensic psychology.

The technique has certain unique features and advantages over the mostly used questionnaire-based psychological tests. First, it allows the wide coverage of personal characteristics. Secondly, it excludes social desirability, which is typical for self-assessment questionnaires. However, handwriting psychology, like other projective methods, is not sufficiently validated. That often makes its usage controversial. Historically, the validation studies of handwriting analysis were based on expert procedures, involving specialists with their manual and often subjective evaluations.

In the last years, many validation studies were done with software, which does not completely substitute the experts, but rather assists them to make their evaluations more objective and reliable. One of these approaches we are discussing is below. It is based on the HSDetect system [20, 21] for handwriting analysis. The system includes statistically evaluated relations between some 800 handwriting signs and about 400 personality traits and behavior patterns.

5.2 Validation of handwriting analysis

In psychometrics, traditionally three major quality criteria are required: objectivity, reliability, and validity.

The objectivity of a psychometric test ensures that the testing person does not influence the result. In the case of handwriting analysis, the testing person is an involved expert or a computer program, which evaluates the written sample.

Reliability denotes the accuracy and precision of the procedure. The results should remain the same when the test and its evaluation are repeated under the same conditions. The typical methods for the assessment of reliability are test-retest, parallel evaluation, and split-half methods. In the context of the handwriting analysis, we consider three major components, namely, the handwriting signs, personality traits, and the relation of the signs to the traits, which we call graphometric functions. They are the objects of the quality assessment. In the traditional procedure, an expert evaluates the handwriting of the subject and interprets it in terms of personality traits, compiling a textual report. This procedure is rather subjective and that was the major objection and the root cause of the controversy. The analyzed studies were done with the computer-aided application HSDetect. This ensures objectivity and reliability [20].

Validity is the primary criterion. Objectivity and reliability are important, but they are just the prerequisites for validity. A test is then valid when it really measures what it is supposed to measure. It is always a challenge to practically define against which reference the test should be validated. Theoretically, a psychometric test should be validated against the psychological features. However, how are those obtained? In most cases, only indirectly, since a direct self-assessment is subjective and a proper expert evaluation is extremely difficult to set up. That is why a typical approach is to check the test against other psychometric tests, which are considered valid. This approach is used in statistical experiments, the data quality of which we are investigating in the current study.

The comparison between two psychometric tests is the comparison between two statistical rows – the results of the validated instrument and the reference instrument. Both tests must include the evaluation of the same subjects (involved persons). In our case, all subjects execute the reference test and provide samples of their handwriting. The handwriting is evaluated by the handwriting experts and HSDetect. Therefore, we can say that the input data for both tests are different, the output is the same - evaluated values of so-called test scales or, in other words, psychological constructs. When the results agree, we can say that the instrument under investigation (handwriting analysis) demonstrates good validity against the reference instrument. Researches very often check the agreement using correlation or another statistical method. In most of our experiments, the direct correlation does not work well and we used a special method consisting of four steps:

- Mapping of original quantitative test scale onto a simpler scale with only three values (high, medium, and low) – scale transformation.

- Assignment of all points of both the reference test and handwriting analysis to one of the three mentioned values.
- Calculation of the number of agreements (both tests have the same value) and disagreements.
- Evaluation of the statistical significance of the agreement or disagreement using the binomial distribution.

In some cases, we did use the correlation, either the product–moment correlation or the lognormal one.

5.3 Data quality determinant for the validation analysis

Data quality of a psychological test has two aspects. The first is the data of the experiment itself. Let us call it the experiment component. The second one is the distribution of subjects involved in the test along different categories – age, sex, education, profession, etc. – subject component. Both aspects are important because they both influence the meaningfulness of the test results. If we, say, make our experiments only with students, the results may be not significant for retired persons.

5.3.1 Experiment component

For the experiment component, we define the following three quality parameters:

- S - The number of subjects involved in the experiment, or in other words, the sample size.
- O – Outliers, their presence, and quantity.
- N - Normality of the row distribution.

They are briefly discussed below. By formalizing, variables S and O get a value of 1, when the quality requirement is fulfilled, or 0, when not. Variable N reflects the relation of normally distributed test scales (or test dimensions) to the total number of test scales.

Sample size: It was mentioned above that we are using the binomial check to decide the statistical consistency of the result. Typically, power analysis [22] is used to evaluate the required sample size. The standard for psychology levels of $\alpha = 0.05$ (type I error of 5%) and $\beta = 0.2$ (type II error) and the medium effect size of 0.5, results in this case in the minimal sample size = 49. Therefore, when the number of subjects is more than 48, we assume this data quality component as fulfilled, that is $S = 1$, otherwise $S = 0$.

The sample size should not be maximal big, but rather it should be optimal with adequate statistical power. It is a critical step in the design of an experiment. Involving too many participants makes a study expensive. If the study is underpowered, it is statistically inconclusive, although its results may be interesting.

Outliers: The outliers are those points of the statistical sample that are distant from other observations. This happens either due to measurement variability or due to the experiment error. Often, outliers are excluded from the data set. In this case,

they may become the subject of special analysis. The exclusion of outliers leads to a reduction in the sample size. On the other hand, that improves the experiment results. Therefore, there is always a trade-off between the result and its reliability.

In our context, there are two types of outliers. The first one means the deviation from the normal distribution of the statistical row (here is the relation to the third quality parameter N). The second type relates to the results of comparison of handwriting analysis to the psychological test. The removal of “bad” points, which mostly contribute to the disagreement, may improve the resulting evaluation. The criterion may be the proportion of improvement to the proportion of change. Say, if we remove 10% of points and that gives us 40% of improvement, we can consider the excluded points as outliers. When the improvement is 5%, the “bad” points are not outliers.

Parameter $O = 1$, when there are no outliers, and $O = 0$, if some outliers exist and were not removed.

Normality: When a random variable is normally distributed that enables many additional methods of statistical analysis, for example, correlation analysis, variance analysis, regression modeling, or ANOVA. That is why the sample must be always checked for normality. There are many methods, the most powerful of which is the Shapiro–Wilk test.

Most psychological tests have a rich normative base and they are generally normally distributed. Whether our current experiments follow the statistical population of the taken psychological test or not is not important. Therefore, we consider only the handwriting variables, which are the subject of research. In the presented model, normality in general often cannot be distinctly defined, since every test has several scales and the check is done for every particular scale and its handwriting model. Therefore, formally, the normality should be a vector and N, as mentioned above, represents the ratio of normally distributed scales to the total scales.

5.3.2 Subject component

In the experiments related to handwriting analysis, such as biodata as sex, age, handedness, education, and profession, are important, since they may influence the handwriting signs. However, in the current application case, we consider only two parameters:

- X – the sex of subjects (two values: female and male).
- A – the age of subjects (four groups are defined: below 30 years old, from 30 to 45, from 46 to 60, and above 60).

In a good experiment, both parameters should be close to uniform distribution to more or less equally represent subject categories. In this case, X and A get a value of 1. If the distribution is far from uniform, they are set to 0.

5.3.3 Data quality determinant

The determinant model is as follows:

$$q = a_S S + a_O O + a_N N + a_X X + a_A A \quad (2)$$

where a_i are corresponding weights.

		16PF-R	NEO-FFI	PVQ	EQ-i2.0
Subjects	No	57	62	22	11
	Age 1 (<30)	15	9	6	0
	Age 2 (30–45)	17	43	8	0
	Age 3 (46–60)	15	10	6	11
	Age 4 (>60)	10	0	2	0
	Sex Male	12	27	2	3
	Sex Female	45	35	20	8
Test scales	No	16	5	10	16
	Normally distributed	12	2	7	12

Table 2.
Raw data for the estimation of quality determinant.

The defined components of data quality are not equally important. That we can solve through the standard approach—assigning different weights. Their values were defined by experts, and therefore, are rather subjective. However, that allows the comparison of different experiments. In our case, we assign the weights, so that the sum is 1.0. The experiment component gets a weight of 0.6, while the subject component is 0.4. The number of subjects is as well more important than outliers and normality. This logic results in the following weights $a_s = 0.36$, $a_O = 0.2$, $a_N = 0.21$, $a_X = 0.11$, and $a_A = 0.11$. The absolute value of the weights is not extremely important, since our aim is mostly to compare different experiments to each other.

The presented model does satisfy four requirements for the good metrics that were formulated above. Namely normalization, adaptivity, scalability, and interpretability. Only cardinality cannot be assured.

The input data for the evaluation of the quality parameters are shown in **Table 2**. We consider four studies on the validation of handwriting analysis against the following psychometric tests [20, 23]: Cattell’s 16 personality factors test (revised) 16PF-R, NEO five-factor inventory by Costa & McCrae, portrait values questionnaire (PVQ) by Schwartz, and the emotional quotient inventory (EQ-i 2.0).

The data quality determinant was calculated based on model (2) and defined above weights. The results are presented in **Table 3**.

The data quality evaluation for different validation experiments demonstrates big differences. It can be a good indicator of the required experiment improvements. For instance, the removal of outliers when the sample size is big enough can be a proper way to improve the statistical power and the data quality. That may improve

Experiment	S	O	N	X	A	q
16PF-R	1	1	0.75	1	1	0.95
NEO-FFI	1	0	0.40	0	0	0.44
PVQ	0	0	0.70	1	1	0.26
EQ-I2.0	0	1	0.75	0	0	0.37

Table 3.
Data quality determinant.

the normality of the data as well. On the other side, outliers may deliver important additional information, and, if their influence on the data quality is not that strong, they should remain in the sample.

In any case, data quality should be the important parameter for the evaluation of the reliability of the whole experiment. How to do that formally is not yet clear. That is the point for further research.

6. Conclusion

The domain-specific information is a very important factor when we are trying to define data quality. Traditional dimensions of quality reflect technical and formal aspects of the data. They are doubles useful and define the requirements for data quality. However, they are not sufficient. The real attitude of data users and the added value of the data quality is reflected in fit-for-use determinants.

In the current work, we formulate the requirements for the data quality metric and analyze two application cases with the fit-to-use determinants. They demonstrate a rather practical than theoretical approach. However, the presented results can be useful in finding ways to control data improvement.

In the presented application examples, the amount of data was small. The preparation of raw data and the estimation of the determinant value were done offline of the data. A universal data quality determinant is practically useful when it can be derived automatically from original data. The testware for the stock exchange reference data is stored in one of the test management systems (in our case, that is, Jira). The corresponding queries from the database could be easily developed and integrated with the test data management. In the second example, the required data was as well automatically derived from the experiment databases. That is a good basis for a generic system for data quality estimation. It can include the calculation engine with different models and adapters for a particular application. Their role is to retrieve the data and convert it into a generic structure.


Especially useful is a universal data quality determinant and the corresponding automatic procedure can be for artificial intelligence models. Their outcome strongly depends not only on the data quantity but as well on the quality of the training and test data. To avoid the famous GIGO (garbage in, garbage out) effect, data quality should be properly managed at all levels.

Author details

Yury Chernov
QADAS, Zurich, Switzerland

*Address all correspondence to: y.chernov@gmx.ch

IntechOpen

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Wang RY, Strong DM. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*. 1996;12(4):5-33. DOI: 10.1080/07421222.1996.11518099
- [2] Data VS. Quality management. In: Kunosic S, Zerem E, editors. *Scientometrics Recent Advances*. London: IntechOpen; 2019. pp. 1-15. DOI: 10.5772/intechopen.86819
- [3] Eppler MJ. *Managing Information Quality*. 2nd ed. Berlin: Springer Verlag; 2003. p. 398
- [4] Batini C, Scannapieco M. *Data Quality: Concepts, Methodologies and Techniques*. 6th ed. Berlin: Springer Verlag; 2006. p. 281
- [5] Pipino LL, Lee YW, Wang RY. Data quality assessment. *Communications of the ACM*. 2002;45:211-218
- [6] Redman TC. *Data Quality for the Information Age*. Boston: Artech House; 1996. p. 332
- [7] McGilvray D. *Executing Data Quality Projects: Ten Steps to Quality Data and Trusted Information*. Burlington: Morgan Kaufmann; 2008. p. 352
- [8] Wand Y, Wang RY. Anchoring data quality dimensions in ontological foundations. *Communications of the ACM*. 1996;39:86-95
- [9] Kimball R, Ross M. *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. 3rd ed. New York: John Wiley & Sons; 2013. p. 600
- [10] Jarke M. *Fundamentals of Data Warehouses*. Berlin, Heidelberg: Springer Verlag; 2003. 219 p. DOI: 10.1007/978-3-662-05153-5
- [11] Juran JM, Godfrey AB. *Juran's Quality Handbook*. 7th ed. New York: McGraw-Hill; 2016. p. 992
- [12] Redman TC. Data quality management past, present, and future: Towards a management system for Data. In: Salig S, editor. *Handbook of Data Quality: Research and Practice*. Berlin, Heidelberg: Springer Verlag; 2013. DOI: 10.1007/978-3-642-36257-6
- [13] Heinrich B, Kaiser M, Klier M. How to measure data quality? A metric-based approach. In: *Proceedings of the International Conference on Information Systems (ICIS 2007)*; 9-12 December 2007. Montreal, Quebec, Canada: AIS; 2007
- [14] Chernov Y. Test-Data quality as a success factor for end-to-end testing. An approach to formalisation and evaluation. In: *Proceedings of 5th International Conference on Data Management*; 24-26 July 2016. Lisbon, Portugal: Lisbon SCITEPRESS; 2016. pp. 95-101
- [15] Roonquangsuwan S, Daengdej J. A test prioritization method with practical weight factors. *Journal of Software Engineering*. 2010;4(3):193-214
- [16] Zwang X, Xu B, Nie C, Shi L. An approach for optimizing test suite based on testing requirement reduction. *Journal of Software*. 2007;18:821-831
- [17] Srikanth H, Williams L. On the economics of requirement-based test case prioritization. In: *Proceedings of the 7th International Workshop on Economic-Driven Software Engineering Research*; 15-21 May 2005. St. Louis, Missouri, USA. New York: ACM; 2005. pp. 1-3
- [18] Elbaum S, Malishevsky A, Rothermel G. Test case prioritization:

A family of empirical studies. IEEE Transactions on Software Engineering. 2002;**28**:159-182

[19] Chernov Y. Data quality metrics and reliability of validation experiments for psychometric instruments. In: Proceedings of 15th European Conference on Psychological Assessment; 7-10 July 2019. Brussels, Belgium: Brussel: ECPA; 2016. p. 37

[20] Chernov Y. In: Chernov Y, Nauer MA, editors. Formal Validation of Handwriting Analysis, Handwriting Research. Validation and Quality. Berlin: Neopubli; 2018. pp. 38-69

[21] Chernov Y. Компьютерные методы анализа почерка [Computer Methods of Handwriting Analysis]. Zurich: IHS Books; 2021. p. 232

[22] Cohen J. Statistical Power Analysis for the Behavioral Sciences. New Jersey: Lawrence Erlbaum Associates; 1988. p. 567

[23] Chernov Y, Caspers C. Formalized computer-aided handwriting psychology: Validation and integration into psychological assessment. Behavioral Sciences. 2021;**10**(1):27