ONS-UNECE Machine Learning Group 2022

Quality of training data

Theme Group Report

Contributors:

Marco J.H. Puts (Theme Group Lead)
Abel da Silva
Loredana Di Consiglio
Inkyung Choi
David Salgado
Claire Clarke
Stirling Jones
Alison Baily

Contents:

1. Introduction

Statistical organizations are increasingly adopting machine learning (ML) in various work areas. ML helps to produce new statistics based on new data sources (e.g., sentiment index based on twitter data), increase timeliness of existing statistics (e.g., crop yield estimation based on satellite data during non-survey periods) and assist humans to perform their tasks more efficiently (e.g., suggesting human coders most likely SOC codes given a job description).

While there is a great interest in the ML algorithms and models[1] relatively little attention is given on the quality of the training data. When building a ML model, much focus is on trying different types of algorithms, combinations of pre-processing methods or hyperparameters, but the quality of the data set is often unquestioned. The issue of data quality is brought to attention or discovered belatedly when the ML model is not working as intended and this is usually after significant resources have already been invested. This disproportionate attention to this input data compared to ML algorithms is also observed in the industry that has been heavily using ML with data work being described as "*least incentivized aspects*" [6] and "*potentially the most under-valued and de-glamorised aspect of today's AI ecosystem*" [7].

The quality, in a broad sense, can be defined as a "totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs" (ISO-9000). Given that the ML model is built based on the training data, the quality of the data impacts directly the quality of the model itself. Statistical organizations have established systematic frameworks and implemented various processes to ensure the quality of the statistics they publish. With the growing interest in ML, the body of works on the ML-related quality in the context of official statistics has been expanding [13, 14]. This paper aims to focus on the quality issues around the training data, describing two sources of errors and how they are introduced in the ML processes.

Selecting data for training and testing is strongly connected to survey methodology. For this reason we will look at the total survey error model to investigate the errors that can be introduced when selecting a data set out of a population.

[1] In the report, algorithm refers to a "finite sequence of well-defined, computer-implementable instructions, typically to solve a class of specific problems or to perform a computation" while model refers to an "output of a machine learning algorithm that is run on the data set". Note that while an algorithm, as a set of instructions to be applied to a data set, exists prior to data, the model is obtained after applying the algorithm to the data set [1]

2. Machine learning in the context of total survey error

Total survey error model

In order to identify the sources of errors in the survey process, the total survey error model is used (see figure 1). The purpose of this section is to discuss only the parts of the total survey error model that are relevant to machine learning. In general, the model is divided into representation errors (errors associated with the sample) and measurement errors (errors unrelated to the sample).

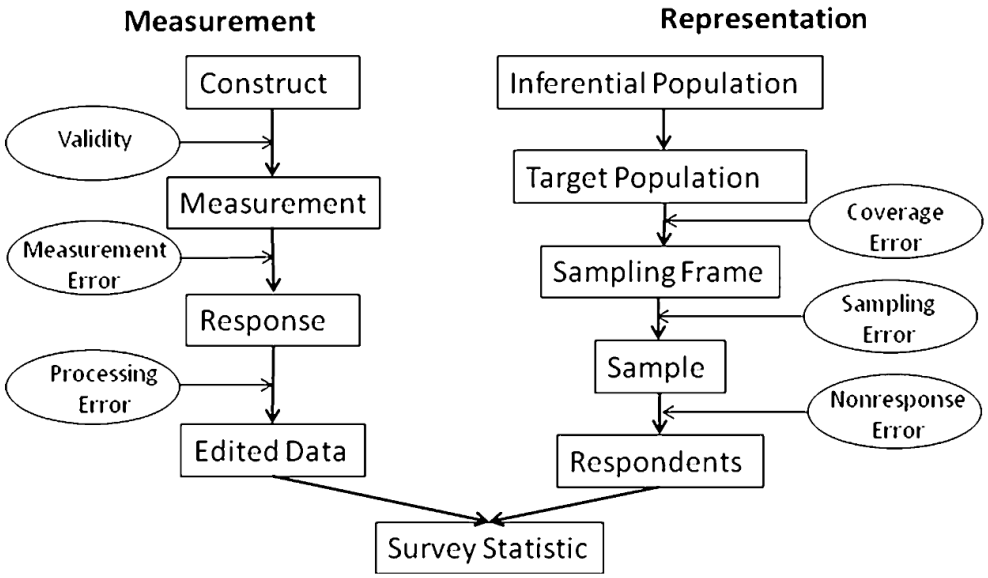

Figure1. The total survey error, according to Groves and Lyberg (2010).

A sample is randomly selected from a sampling frame during the survey process. It is important that this sampling frame covers the entire target population, but sometimes it does not. In some cases, the sampling frame may contain elements that are not present in the target population (overcoverage), while in other cases it may miss elements present in the target population (undercoverage). These coverage errors could result in errors in the final estimate (the survey statistic in figure 1). The selection of the sample is another source of error. The sample size, for example, has an impact on the sampling error. Taking the entire population as a sample, it is evident that the sampling error is zero. As the sample size decreases, this error increases. Thus, a small sample size will result in a larger sampling error.

A survey is designed based on a construct. Simply stated, a construct describes the relationship between measurables and the phenomenon we wish to measure. A flawed construct will ultimately lead to an error. The relationship between our measurements and the target statistic determines whether we are on the right track. It is very common for our

target variable to be latent and we must translate it into a measurable. Consumer confidence is an example of this. The confidence of a consumer is latent and we operationalize it with a set of questions about spending behavior in the past and the future.

Errors may also occur during the measurement process. These errors are known as measurement errors. There is the possibility that a respondent does not know the exact answer to a question, or that a sensor has a certain degree of drift, which makes the result less accurate. These errors are referred to as measurement errors.

Total survey error in machine learning

Experts in machine learning are often concerned with optimizing their performance indicators when they are training their models. For instance, when training a classifier, the confusion matrix is consulted and certain scores are calculated on the basis of this confusion matrix.

Poor performance may result from a variety of factors, ranging from a training set that does not follow the same distribution as the population to poorly annotated data. In this paragraph, we will examine how different errors in the training set affect the performance of machine learning models. The current text focuses on classifications. It is important to note, however, that regression problems are not immune to these kinds of errors.

There are two stages involved in machine learning: the learning stage and the prediction stage. In the learning stage, the model will be able to determine a certain relationship in the data. In the prediction stage, the model predicts the values of items in a population in accordance with a set of features used during training. Different types of errors can occur at both stages. Our next step will be to look at the errors in the learning and predicting stages and match them in the total survey error framework.

Learning Stage

During the learning stage, the machine learning model is trained to accomplish the task of predicting a dependent variable based on a set of independent variables, called features. From an estimation point of view, one could reflect on such a process as estimating a set of parameters, defining the machine learning model. This set of parameters will have a certain uncertainty, which lead to (co-)variances or biases in the estimated parameters. In the next two paragraphs, we will try to investigate the different type of errors that can arise during the estimation process of these parameters.

To do so, we will use the total survey error framework, which was developed to identify the different kinds of error that can appear when we try to make an estimate based on a sample. The total survey model consists of two parts: measurement and representation. Measurement errors are mostly concerned with the relevance and correctness of the data, whereas the representation errors are concerned with the relationship between population, frame, and sample.

Measurement errors

Data is collected, prepared, and, in cases where there is no dependent variable available, annotated. The collected data contains independent variables that should be related to the dependent variable. This is the point at which the quality of the model has already been determined. Based on the strength of the relationship between the independent variables and the dependent variable, the model will perform better. When the relationship between independent variables and the dependent variable is not so strong, this will result in a larger variance in the final model.

Even when a model appears to perform quite well, it is possible that the relationship discovered by the machine learning model has no basis in reality (the problem of spurious correlations). Due to this, it is likely that the trained model will have a low degree of external validity. A machine learning model is formed by only looking at the data in the training set - the sample from the sampling frame that is used to train the model. This means that, by default, the model is only valid for that part of the population. This is called **internal validity**. **External validity** refers to the fact that the model is also valid for the complete population.

Regarding the total survey error, this issue is loosely related to the construct validity of the model. In implicit terms, the machine learning expert assumes a strong relationship between the independent variables and the dependent variable, which is closely related to the construct. This could lead to a larger uncertainty during the predicting phase.

The problem of model validity arises very often in machine learning. The models are created in a very data driven fashion, and one is often not that concerned about the real relationship between the dependent and independent variables. Validating these models afterwards is difficult. This leads to a ground truth problem which is hard to solve. One way of dealing with it is by using a continuous validation process, even when the model is already in production [2]. The model is continuously validated by finding proof for the correctness of the model over and over again.

Another problem can arise when annotating data. The quality of the annotated data is determined by the difficulty of the annotation task as well as the quality of the annotator. It may be beneficial to consult extra data when the annotation task is challenging. For instance, when identifying crops in a field based on satellite images, it will be extremely hard to identify the crops on the satellite images per se, leading to larger uncertainties in the estimated parameters. Obtaining ground truth measurements (by visiting the crop fields) is necessary for the annotation process to be successful. In terms of the total survey error, this problem is related to the measurement error.

The measurement error also depends on the experience of the annotator, and sometimes on perception biases, the quality of the annotation will vary. Let us assume that an annotator is asked to identify the sentiment of sentences in a corpus. A person's annotation will be influenced by his or her experiences, culture, and even emotions. In fact, it is possible that the annotation is influenced by perception. As an example, visual illusions demonstrate the possibility that images can be perceived incorrectly when performing visual tasks.

Representation errors

In machine learning, we often receive data sets of unknown origin. Our task is to develop a model that performs the task as effectively as possible. Even when the final model performs

extremely well, there are several factors to consider in determining its quality. Essentially, these factors relate to the extent to which the data is representative of the population. Clemmensen and Kjaersgaard (2022) provide a thorough discussion of this topic. For now, we will consider two types of error: the coverage error (whether the sampling frame covers the entire population) and the sampling error (how well does the sample reflect the items in the sampling frame).

It would be ideal if the frame from which we can sample cases covered the entire target population. Each unit in the frame corresponds to a unit in the population and vice versa. In this situation, we have a clear understanding of how features are distributed throughout the population. The situation changes when we have over- or under-coverage. In certain conditions, coverage problems will lead to coverage bias: the model will eventually not focus on the right features due to the lack of certain cases.

For instance, in the case of the platform economy use case, the business register was used to identify as many URLs as possible of company websites in the Netherlands. Approximately 20% of these companies were not included in the frame. Coverage bias may result from the absence of these websites. It was assumed that this coverage bias is negligible. This is because it would be strange for a platform company that is based on its own website not to have a website. In contrast, if we were to examine, for example, construction companies, we would find a different picture. It is unlikely that many construction companies will have a website because their success does not depend on having one.

In order to create a training set (and a test set), a sampling mechanism must be used. From the perspective of sampling, a random sample will provide a valid representation of the entire population. It should be noted, however, that this is not always the case. When identifying rare events, it is better to make sure that we cover many of the features that really distinguish the different classes. Whatever choice we make, this may lead to biases in the model.

Another source of sampling error comes from taking a finite sample from a finite population. With an increasing sample size, the (co)variance of the parameters will decrease, leading to a more stable model. Thus, when creating a large number of models based on independent samples, these models will all tend to the same parameters when we increase the number of samples in the training set.

Predicting stage

During the predicting stage, the estimated parameters are used in the model to predict all other items in the population. So, instead of using the dependent variable which was used during the learning stage, we now use the model to derive the dependent variable for the unobserved set of the target population based on the independent variables.

The different errors, discussed during the previous paragraph, will affect the quality of the results during the prediction stage. Table I gives a short overview of how bias and variance are introduced in the final predictions of the model based on errors in the model.

Table I. Errors introduced during the learning stage leading to variance or bias during the predicting stage.

Error learning stage	**Bias predicting stage**	**Variance predicting stage**
Construct validity	Relationships between dependent and independent variables changes in time (Real concept drift)	Independent variables do not completely predict dependent variable
Measurement error	Systematic misinterpretations by annotator	Random misinterpretations by the annotator
Coverage error	• Cases not missing at random • The sampling frame changes in time (Covariate shift)	
Sampling error	• Sample too small to find a good estimate of parameters • Sample was not representative for frame	• Different samples produces different estimates

As can be seen, many errors result in bias. This is primarily due to the fact that errors introduced during the learning stage are corrected during the prediction stage. Consider the case of a model that is trained on a probability sample of (a very small) number of cases, for example. There is no doubt that this results in a large sampling variance in the parameters. In the absence of more than one training set, the error is fixed, which results in bias when the model is used for prediction.

Measurement error is another interesting type of error. In the training material, it is unclear what the ground truth is, which leads to several types of uncertainty. It is possible for the annotator to display random behavior (if he or she is unaware of the value, a random value will be annotated) or systematic error. In the first case, there will be an increase in variance, in the second case, there will be an increase in bias.

Different sources of error are added by concept drift. Concept drift can occur when the relationship between the independent variable and the dependent variable changes (real concept drift), when the underlying distribution of the independent variables changes (covariate shift) and when the proportions of the different classes (in case of classification) changes (prior probability shift).

Whenever independent variables that are found to be related to each other do not have a stable basis in reality, the variables will lose their relevance rather quickly. This is called real concept drift. Let us suppose we train a model to identify Dutch users on social media. Assume the model finds Dutch first names to be positive features and non-Dutch names to be negative features. This appears to be a reasonable feature. However, Dutch residents are increasingly adopting first names with other origins (one of the authors is named Marco, despite not being Italian). Moreover, migration means that Dutch citizens will come from other countries, with different names. The first name may be a strong feature right now, but it will become a weaker feature in the future. As a result, the implicit construct that the model assumes will be invalidated by this concept drift. (See the report of the Model Retraining Theme Group for more information on the scope of drift.)

Concept drift can also be attributed to coverage, called covariate shift. Consider the case in which the composition of the frame changes. As a result of the loss of representativeness of the training set, the model's performance will be degraded.

3. Human Annotation Process

Many machine learning applications source their training data through human annotators, who may or may not be experts. In the latter case, the crowdsourcing approach may be used. It is an interesting option for applications where there is more than one right answer for each example, such as natural language processing tasks. Indeed it can be cheaper and faster than relying on experts. It can also represent the diversity of opinion by assigning many annotators to each example, since these annotators may disagree. For example, in their natural language processing example experiment, Soberon et al. (2013) assign 15 annotators per sentence to extract medical relations from a set of sentences. Then the question is how to best exploit the expected disagreements among the annotators.

To this end, Inel et al. (2014) have proposed the CrowdTruth methodology that includes many disagreement metrics. In this methodology, an example is called a media unit and an annotator is called a worker. The task consists in checking if an example satisfies some properties, where the result of checking a specific property is called an annotation. In a closed task, these properties are known beforehand, i.e. before the examples are examined by the annotators. In an open task this is not the case.

Describing the CrowdTruth metrics is simpler for closed tasks, where there are three kinds. Following Inel et al. (2014), define an annotation vector as the 0-1 vector comprising the annotations of a media unit by a worker. Also define a media unit vector as the sum of all the annotation vectors for the media unit.

1. The first kind of metric is at the level of each *media unit*, including the unit-annotation score and the unit clarity score. The unit-annotation score is the cosine similarity between a media unit vector and an annotation unit vector, where the latter vector is the 0-1 vector with a single 1 for the annotation. The unit clarity score is the largest unit-annotation score for a media unit.

2. The second kind of metric is at the *worker level*, including the average number of annotations per unit, the worker-worker disagreement and the worker-unit disagreement. The worker-unit disagreement is the cosine similarity between a worker's annotation vector for a given media unit and the media unit vector minus the worker's annotation vector. It measures the disagreement between the worker and the other workers regarding the media unit. The worker-worker disagreement compares two workers across all the media units that they both annotate. It is based on the worker-worker agreement that is defined as a ratio, where the numerator is the total of the common annotations across the common media units, and the denominator is the total of the annotations across all the media units annotated by the first worker. The worker-worker disagreement is then equal to one minus the worker-worker agreement.

3. The third kind of metric is at the *annotation level*, including the annotation frequency, the annotation clarity score, the annotation similarity score and the annotation ambiguity score. The annotation frequency is the number of media units where it is selected by at least one worker. The annotation clarity score is the maximum unit-annotation score across all the media units. For two annotations, the annotation similarity score is related to the conditional probability that the second annotation is selected given that the first annotation is selected in a media unit. For an annotation, the annotation ambiguity score is the maximum similarity score between this annotation and any other annotation. It is expected to be small for a clearly defined annotation.

Soberon et al. (2013) apply these metrics and define additional metrics when extracting medical relations from sentences.

Dumitrache et al. (2018) define improved metrics to account for the interdependent nature of the media unit ambiguity, annotation ambiguity and worker performance. To this end they define quality scores for the media units, annotations and workers. Using these quality scores as weights, they propose weighted versions of the above described disagreement metrics. For example they weigh the different annotations by their quality score, when computing the agreement between a worker and a media unit. Ultimately, Dumitrache et al. (2018) obtain a system of fixed point equations that are solved iteratively.

The disagreement metrics serve to improve the quality of the annotations by identifying the low quality workers, who are also called spam workers or simply spammers, and removing

all the related annotations. For example, Soberon et al. (2013) identify the spammers based on a threshold on the number of annotations, the worker-unit disagreement or the worker-worker agreement, separately, or after combining all these metrics linearly.

4. Representativity

Machine learning models are trained to predict a certain target variable for each individual in a population. This means that we train the model on a sample of data and use the prediction for the rest of the population. Frequently, we use found data as a training set. It is important to note that this data is derived from a non-probability sample, which is usually not representative of the entire population. In this regard, an important question is whether the final model can be generalized to the entire population. Or is it only valid for a smaller subpopulation within that population?

When examining the training set, a number of problems can arise. In the case of classifications, the classes may be present in different proportions in the training set than in the population. Almost all classification models implicitly assume that the classes in the population are represented in the same proportions as in the training set, resulting in a bias ([15], [16]). Another source of error may arise due to the fact that the features in the training set are not equally distributed as those in the finite population. In this case, we might refer to a model that is not representative of the population.

ML models are best viewed as estimates of parameters or coefficients. There will be a sampling error associated with these parameters, which can be measured through the covariance matrix of these coefficients. Whenever the sample size is small, the sampling error will be large, and since the model will be fixed, the large error will be visible as a bias. To obtain a model with only a small sampling error, we need to select the right sample size.

Statistical organizations have extensive experience and expertise in representative sampling which should be utilized to ensure representativeness. What this representativeness should look like in practice is still open for discussion. Suppose we have a data set with a large class imbalance. If we trained a model on a “representative” data set, one where the proportions of classes is the same as in the population, the model would never learn this class in the right way. It would be more ‘convenient’ for the model to learn that all items belong to the major class. On the other hand, the model is trained based on the distribution of features in the training set, and disregarding this distribution completely will not lead to a well trained model. For a discussion of representativity, see [1].

5. Use Case: ABS Intelligent Coder

The Australian Bureau of Statistics (ABS) has been expanding the use of its ABS Intelligent Coder to a range of ABS collections. The ABS Intelligent Coder is an SVM-based classifier that can be trained to predict most statistical classifications from free text responses to surveys and from other data sources. In this use case the ABS Intelligent Coder was being

trained to predict ANZSCO – the Australian and New Zealand Standard Classification of Occupations – 4-digit codes from an employee/employer business survey.

The coder was initially trained using a previous cycle's pre-coded data. However, this did not perform well in the training or validation steps. Attempts were made to adjust the hyperparameters of the model, but none had more than a minimal impact on model performance. It was then observed that the text responses in the survey were long and verbose, so several approaches were tried to shorten, split, or otherwise duplicate / reduce these responses. This improved the performance of the model but not by a substantial amount. The management information provided by the coder suggested that it was overfitting, but the survey sample was not especially large (approximately 55,000 records), so reducing the number of examples seemed counterintuitive.

Eventually, after careful, manual examination of the data, it was determined that issues with the quality of the previous cycle data – particularly the labels – were the likely cause of the problems. For example, some records with similar text responses were coded to different occupation categories. This is a common problem with complex classifications such as ANZSCO where the difference between, for example, a Chef (ANZSCO 3513) and a Fast-food Cook (ANZSCO 8511) may be quite difficult to determine given the information available. A method for detecting and resolving these kinds of issues with labels could have saved considerable time and resources spent in discovering the true source (nature) of the problem.

Current approaches to training the coder use latest Census response data (2021) as this is a large and rich data source with approximately 24 million records. Python programs are used to clean, filter, manipulate and perturbate the raw dataset to improve its use in machine learning. The programs remove: stop words and noise (nonsensical words), apply normalisation, tokenisation, remove duplicate words, and remove records with less than three characters. The programs will also remove records with codes that do not match legacy coder indices. In its current design the ABS Intelligent Coder has limited processing power with 10 CPUs and 198 GB of RAM. As this limits training files to approximately 650,000 records a weighted sampling approach is then used to reduce the size of the training dataset whilst ensuring classification and response range coverage. While still in development testing, this approach has thus far yielded much more suitable training and validation outcomes.

This new approach will be used for the development of multiple cross-cutting applications including classifying responses to ANZSIC (Industry), ANZSCO (Occupation), SACC (Classification of Countries), ASCL (Language), ASCRG (Religion) and ASCED (Education) to support the broader migration of ABS legacy auto coder users to the ABS Intelligent Coder.

6. Conclusion

The quality of an ML model depends for a great part on the quality of the training data. This starts with the selection of features and the (implicit) assumption that these features describe

the dependent variable and that the training set represents the target population. Explainable AI could shed light on the first problem, whereas survey methodology helps in investigating the latter problem. Many of the errors introduced by these uncertainties are discussed in this paper.
The total survey error model gives many insights into the different kinds of errors that can occur when training a machine learning model. One the most surprising conclusions is that most of the errors introduced during training will lead to biases during prediction.

References

1. Clemmensen, L.H. and Kjærsgaard R.D. (2022) Data Representativity for Machine Learning and AI Systems
2. Yu, B. (2020) Veridical data science
3. Zhang, L. (2012) Topics of Statistical Theory for Register-Based Statistics and Data Integration.
4. Meertens, Q.A. (2021) Misclassification bias in statistical learning https://pure.uva.nl/ws/files/59712159/Thesis.pdf
5. Kraff N.J., Wurm, W., and Taubenböck, T. (2020) Uncertainties of Human Perception in Visual Image Interpretation in Complex Urban Environments
6. Sambasivan, N. et. al. (2021) "Everyone wants to do the model works, not the data work": Data Cascades in High-Stakes AI
7.Data Excellence for AI: Why Should You Care
8. UNECE (2022) Machine Learning for Official Statistics
9. Soberón, G., Aroyo, L., Welty, C., Inel, O., Lin, H., & Overmeen, M. (2013). Measuring crowd truth: Disagreement metrics combined with worker behavior filters. In Proceedings of CrowdSem2013 Workshop, ISWC2013. Available at http://ceur-ws.org/Vol-1030/paper-07.pdf.
10. Inel, O., Khamkham, K., Cristea, T., Dumitrache, A., Rutjes, A., van der Ploeg, J., Romaszko, L., Aroyo, L., and Sips, R.-J. (2014). Crowdtruth: Machine-human computation framework for using disagreement in gathering annotated data. In Proceedings of The Semantic Web–ISWC 2014, Springer, pp. 486–504. Available at https://link.springer.com/content/pdf/10.1007/978-3-319-11915-1_31.pdf.
11. Aroyo, L., & Welty, C. (2015). Truth Is a Lie: Crowd Truth and the Seven Myths of Human Annotation. The AI Magazine, 36(1), 15-24, https://doi.org/10.1609/aimag.v36i1.2564