Automatic Versioning of Time Series Datasets: a FAIR Algorithmic Approach

Alba González–Cebrián*, Luke A. McGuinness*†,
Michael Bradford*, Adriana E. Chis*, and Horacio González–Vélez*
*Cloud Competency Centre, National College of Ireland. †DTSL, Ireland.
Email:{alba.gonzalez-cebrian,luke.mcguinness,michael.bradford,adriana.chis,horacio}@ncirl.ie

***Abstract*—As one of the fundamental concepts underpinning the FAIR (Findability, Accessibility, Interoperability, and Reusability) guiding principles, data provenance entails keeping track of each version for a given dataset from its original to its latest version. However, standard terms to determine and include versioning information in the metadata of a given dataset are still ambiguous and do not explicitly define how to assess the overlap of information between items along a versioning stream. In this work, we propose a novel approach for automatic versioning of time series datasets, based on the use of parameters from two dimensionality reduction approaches, namely *Principal Component Analysis* and *Autoencoders*. That is to say, we systematically detect and measure similarities *(information distances)* in datasets via dimensionality reduction, encode them as different versions, and then automatically generate provenance metadata via a FAIR versioning service using the W3C DCAT 3.0 nomenclature. We illustrate this approach with two time series datasets and demonstrate how the proposed parameters effectively assess the similarity between different data versions. Our results have shown that the proposed version similarity metrics are robust ($s^{(0,1)} = 1$) to the alteration of up to 60% of cells, the removal of up to 60% of rows, and the log-scale transformation of variables. In contrast, row-wise transformations (e.g. converting absolute values to a percentage of a second variable) yield minimal similarity values ($s^{(0,1)} < 0.75$). Our code and datasets are openly available to enable reproducibility.**

***Index Terms*—Data Provenance, Dimensionality Reduction, Information Distance, Principal Component Analysis, Findability, Accessibility, Interoperability, Reusability, Open Science, DCAT**

I. Introduction

Reproducibility and comparability have long been considered *sine qua non* within data-intensive scientific discovery [1], but an inadequate data organisation can easily imply that subsequent runs of the same experiment yield distinct results due to different data entities. If reproducibility and comparability are to be further improved, then dataset provenance should be recorded by keeping dataset lineage at a lower level of granularity i.e. systematically tracking the associated entities a.k.a. *versions*.

Versioning is typically considered as enabled when i) a dataset allows the access to all data, both retrospectively and prospectively, via user-definable interfaces and documented through metadata; and, ii) the history of changes is retained across entities where version labels are determined either by the modification time or by some user-defined method.

The importance of versioning has been acknowledged by initiatives such as the PROV-Template approach [2]. The recommendations of the Provenance Incubator Group ([PROV-XG]), include the definition of the following terms used by the PROV standard to define versioning: derivation, revision, specialisation, and alternate. Nonetheless, this framework is still ambiguous, e.g. a *revision* is defined as "a derivation (transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity) for which the resulting entity contains substantial content from the original". However, to the best of our knowledge, there is no technical definition of what "substantial" implies.

On the one hand, when it comes to scientific discovery, a useful framework for versioning is arguably an information-based one which defines labels according to the information distance from an initial dataset. Such a scenario is often intrinsic to the research method, due to the intermediate pre-processing steps followed by each research group.

On the other hand, information-based versioning is not necessarily aligned with the FAIR guiding principles. Based on four main pillars—Findability, Accessibility, Interoperability, and Reusability—the FAIR principles are an inter-sectional attempt to ease the access, use, and reuse of digital resources by both humans and machines [3]. They are being increasingly required to share and publish data science research. Indeed, the requirement of FAIR-compliant work has rightly permeated different stakeholder communities, but the current solutions offered to comply with them are not standardised.

> The aim of our research is to automatically generate standardised FAIR-compliant provenance metadata using an information-based *versioning service*, which efficiently detects and measures changes (information distances) in datasets and offers seamless interoperability between data catalogues.

The problem of parameterising data versioning has strong repercussions in many fields, but they can be specially critical in research. Reproducibility of results usually requires the analysis to use an identical version of the data. Hence, a reproducible research environment should provide computational tools together with the ability to automatically track the provenance of data, analyses and results, and to package them (or to point to persistent versions of them) for redistribution.

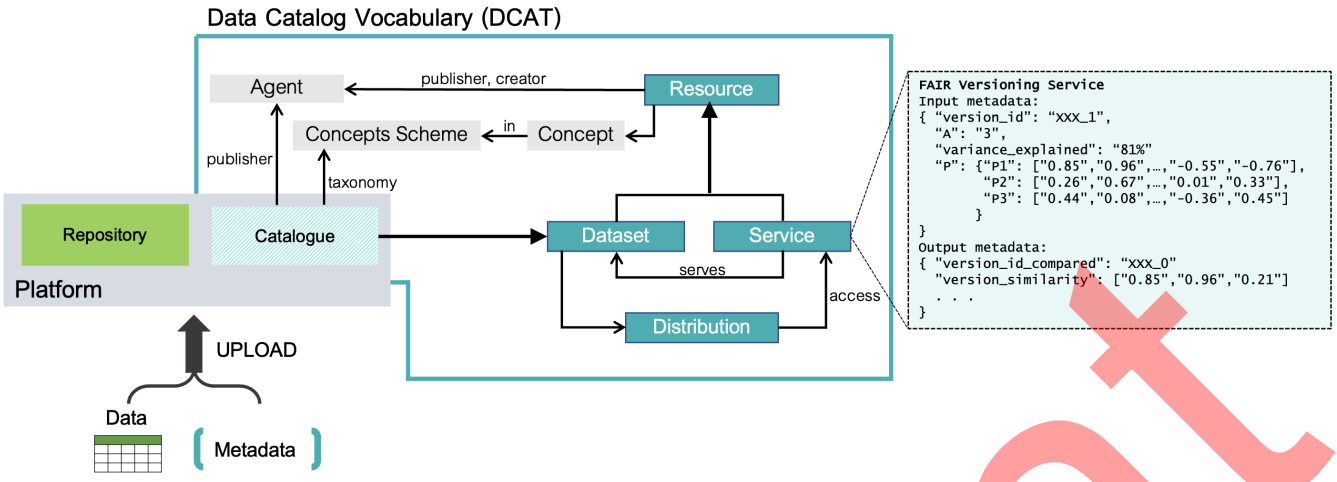

Fig. 1. High-level diagram representing the upload of a dataset to a platform, where the "Catalogue" follows the DCAT 3.0 nomenclature, which integrates the element "Services" interacting with the "Dataset" information, constituting a resource of the Catalogue. A box is representing the metadata required and produced by a FAIR versioning Service comparing the parameters from two different versions of a dataset.

Ensuring the convergence and interoperability of the data science community on this matter, still requires significant harmonisation aligned with the FAIR guiding principles [4].

This research puts forward dimensionality reduction techniques as the basis of a quantitative approach to measure and parameterise the versioning process of data. Namely, we compare the performance of parameters extracted from a Principal Component Analysis (PCA) model [5]–a well-established statistical technique to discard redundant variables [6] ergo reducing dimensionality in datasets–and Autoencoders [7], [8]. Both techniques deal differently with the same mathematical goal: modelling a low-dimensional representation of the original dataset, which can be useful to retain a reduced number of parameters as descriptors of the numerical information within the data [9].

Our results show that values of PCA-based version similarity metrics are consistently higher than the Autoencoder-based metrics when altered and original versions are compared. Moreover, the Least Significant Difference (LSD) intervals for the PCA-based version similarity values indicate a perfect match between the altered versions and the original one for changes up to 60% of cells and up to the deletion of the 60% of rows (Cases I and III). Besides, even if non-linear transformations are applied to the totality of variables, the resulting versions still have a perfect match with the original version parameters (column-wise transformation from Case II). On the contrary, transformations affecting the correlation pattern are the ones yielding the minimal similarity values ($s^{(0,1)} < 0.75$) for row-wise transformation from Case II).

The paper is structured as follows. Section II critically compares some related approaches and then frames our contribution. Section III explains the chosen methods for dimensionality reduction. Section IV presents the results obtained assuming different scenarios of data versioning using two Open Data time series: *Dublin Footfall* and *Air Quality*. Finally, Section IV discusses the main conclusions of our work, along with avenues for further research. In compliance with Open Science principles, the code and data used are openly available from https://github.com/SMARDY-NCI/ADV.

II. Related Work

In data science, provenance is typically determined via the amount of information describing all the elements and their relationships, that contribute to the existence of a datum [10]. While adequate data provenance affords researchers access to experimental reproducibility, knowledge reuse, and data quality assessment, datasets are often released without any provenance information in their metadata. There are two traditionally accepted approaches to recording data provenance in metadata [11]. A prospective one where metadata includes annotations describing typically *ab-initio* the different versions; and an inversion one, where derivations of datasets can be determined via data queries. While both methods have merits in their own right, neither has been standardised in terms of the FAIR principles, a *sine qua non* for open data.

While there have been some attempts to extract and annotate automatically different code versions from source scripts, documented methods for versioning data mostly focus on how datasets are processed but rarely deal directly with datasets contents and their structure [12]. On the other hand, computational research has long studied the changes of datasets from an infrastructure perspective, forming different versions via a structured graph approach with an information retrieval perspective to store datasets [13] or as part of large distributed experiments with a general metadata storage and management layer for parallel file systems [14].

We contend that such approaches do not represent the final goal of the FAIR data principles. Further efforts should be

carried out to define more quantitative and objective versioning and provenance vocabularies. In fact, this critical intersection between FAIRness, provenance, and versioning is aligned with the definition of a proper environment tackling data provenance, provided by W3C [15], stating that such environment should:

- allow objects referring to versions of as they evolve over time, or to temporal information statements of when the object was created, modified, or accessed. In particular it should provide for a representation of how one version (or parts thereof) was derived from another version (or parts thereof);
- include a standard way to represent a procedure which has been enacted; and
- include a way to determine commonality of derivation in two resources.

We argue that data provenance should be *automatically* defined and extracted in more specific FAIR terms to enable experimental reproducibility [16]. The importance of data provenance in data FAIRness is mentioned in several principles, namely, *Interoperability Principle I3* and *Reproducibility Sub-Principle R1.2*. *Principle I3* states that "*(meta)data include qualified references to other (meta)data*", referring to the fact that data digital resources are often interlinked and metadata should refer to these relationships between resources. Some "upper ontologies" (e.g. SIO-biomedical research [17]) defining relationships can be used as-is, or as a starting-point for a new and more specific relationships. Hierarchical relationships between data can ease the interpretation of the intent of the new relationship, and also improve the interoperability.

Furthermore, efficient automatic metadata extraction generation has remained an open problem [18], particularly in connection with data provenance. Specifically, *Reproducibility Sub-Principle R1.2* requires that "*(meta)data are associated with detailed provenance*", which includes all the transformations of processes that have been applied to an original data object. Some of the tools focused on making the construction of FAIR metadata easier include the CEDAR workbench [19], CERN's CASTOR [20], and the knowledge models in the Data Stewardship Wizard [4].

As an RDF vocabulary designed to facilitate interoperability between data catalogues, the Data Catalog Vocabulary (DCAT) [21] describes datasets and data services in a catalogue to allow consumption and aggregation of metadata. While there have been initial guidelines suggestions to make DCAT vocabularies FAIR compliant [22], to the best of our knowledge, none of the aforementioned tools integrate FAIR-compliant data versioning as part of DCAT metadata in a quantitative, measurable, and automatic manner.

A. Contribution

In this work, we are proposing quantitative and measurable data versioning, as a systematic approach to distance reporting about the similarity between data versions, as part of a versioning streamline for time series datasets using dimensionality reduction.

Figure 1 illustrates the integration of the version-similarity parameters using the DCAT 3.0 nomenclature in our approach by explicitly documenting the *Dataset* metadata and the *Service* element. The FAIR-versioning service returns the version-similarity values and provides new information to be integrated as terms within the Catalogue.

In the short-term, the contributions of this work involve the assessment and selection of a candidate set of parameters that present a parsimonious and informative behaviour as a function of the distance between altered and original versions. Afterwards, the selected set of parameters for data versioning will have to be articulated with other tools as part of a data marketplace ecosystem. Thus, middle-term contributions derived from this work will involve the integration of the selected parameters with provenance and metadata standards.

Finally, in the long-term, there are several benefits of having informative and quantitative data versioning protocols. First of all, having an assessment on how numerically similar are the final versions of the datasets used in different studies, could facilitate a proper comparison between research studies. Moreover, one of the issues that prevents attaining full reproducibility of research results, involves data owners rejecting requests to share data. However, assessing the similarity between data versions through a set of parameters, instead of doing so directly with the data, could be better accepted among the data owners.

III. Methodology

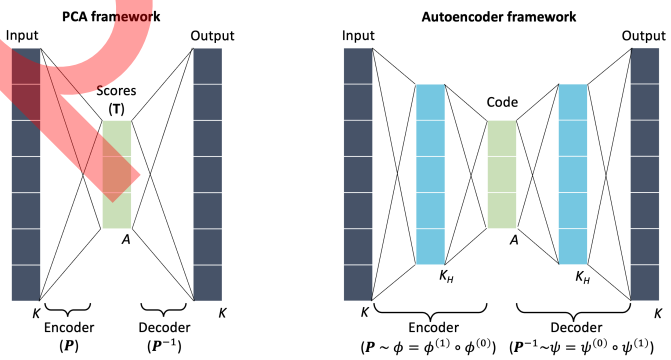

Fig. 2. Visual representation establishing a parallelism between the PCA (left) and the Autoencoder (right) model frameworks, and how the encoding and decoding steps ($\mathbf{P}$ and $\mathbf{P}^{-1}$ terms, respectively) are represented in each model.

Two different approaches are compared in this work, namely the parameters extracted by a PCA model and by an Autoencoder. This set of parameters would be integrated as part of the metadata, enabling a quantitative and standard framework for data comparison that could be performed based on a set of metadata fields.

A. Principal Component Analysis

Let $\mathbf{X}$ be a matrix with N observations on K variables. After some pre-processing such as mean-centering and/or

unit variance scaling, a PCA model is estimated. This is done by compressing the high-dimensional $\mathbf{X}$ matrix into a low-dimensional subspace of dimension A (with $A \leq rank(\mathbf{X})$). PCA is based on the bi-linear decomposition of $\mathbf{X}$ in $\mathbf{X} = \mathbf{T}\mathbf{P}^\top + \mathbf{E}$, where $\mathbf{T}$ is an $N \times A$ matrix of *scores* and $\mathbf{P}$ is a $K \times A$ matrix of *loadings*.

The A columns of the loading matrix $\mathbf{P}$ are the *loading* vectors $\mathbf{p}_a$, with $a = 1, 2, \ldots, A$. The *score* matrix $\mathbf{T}$ can be considered as a collection of row vectors $\boldsymbol{\tau}^\top$ (scores of an observation) or column vectors $\mathbf{t}_a$ (latent variables, with $\mathbf{t}_a = \mathbf{X}\mathbf{p}_a$ and $a = 1, 2, \ldots, A$). The score matrix can be obtained as $\mathbf{T} = \mathbf{X}\mathbf{P}$, that is, as the projection of the $\mathbf{X}$ matrix on the $A-$dimensional space of the PCA model (i.e., columns of $\mathbf{P}$ matrix). Analogously, given an observation $\mathbf{x}$ of the original $K-$dimensional space, its projection $\boldsymbol{\tau}$ onto the subspace of the model can be obtained using the projection matrix $\mathbf{P}$ as well by $\boldsymbol{\tau} = \mathbf{P}^\top\mathbf{x}$. From the scores matrix one can recall the explained part of $\mathbf{X}$ in the PCA model as $\hat{\mathbf{X}} = \mathbf{T}\mathbf{P}^\top$. Thus, the original data can be decomposed by the part explained (i.e., predicted) by the model (signal or $\hat{\mathbf{X}}$) and the error not considered in any of the A latent variables (noise or $\mathbf{E}$).

B. Autoencoders

Autoencoders are a specific type of feed-forward neural networks where the input is compressed into a lower-dimensional code, and then reconstructed to obtain the output. The code is a compression of the input, which is the reason why Autoencoders are mainly a dimensionality reduction algorithm whose mathematical framework enables them to include non-linearities in the encoding and decoding process.

Mathematically, the Autoencoder uses the encoding function ($\phi : X \rightarrow F$) to obtain a representation of observations in $\mathbf{X}$ of a lower dimensionality. Then, the low-dimensional representation, $\mathbf{F}$, is decodified by applying the function $\psi : F \rightarrow X$. The goal of the Autoencoder is finding the set of coefficients that will yield a loss value as low as possible:

$$\phi, \psi = \arg\min_{\phi,\psi} ||\mathbf{X} - (\psi \circ \phi)\mathbf{X}||^2 \quad (1)$$

where the second element of the subtraction refers to the reconstruction ($\hat{\mathbf{X}}$) of the original input, $\mathbf{X}$.

The optimisation problem formulated in the previous expression is the same one as for the PCA model. Both the Autoencoder and the PCA will fit a set of parameters that will try to reconstruct the original matrix $\mathbf{X}$, minimizing the loss of information, measured as the mean squared error between the input and the reconstructed output.

C. Versioning parameters

As can be seen in the previous definition of the algorithms, there is a commonality between PCA and Autoencoders [9]. In fact, a PCA model can be regarded as a single-layer, linear case of an Autoencoder (Fig. 2). Based on this conceptual overlap, we decided to search for a comparable set of coefficients that could be useful to track the difference between different versions of a dataset.

It is important to remark that this versioning framework would be based on the numerical information shared by different versions of the same data. This assumption of a hierarchical relationship between the compared datasets is the basis of the proposed comparison based on the parameters of a low-dimensional space which, by definition, should retain the characteristic signal of the data.

The proposed metric to measure the similarity between versions (0) and (1) of a dataset ($s_a^{(0,1)}$), is the correlation between pairs of homologous loading vectors:

$$s_a^{(0,1)} = corr(\mathbf{p}_a^{(0)}, \mathbf{p}_a^{(1)}) \quad a \in 1, ..., A \quad (2)$$

This correlation can be used to keep track of the differences on how each model mathematically defines the latent variables (the scores in the context of PCA, and coded information in the context of Autoencoders). Fig. 3 illustrates the comparison between two versions of the same dataset, yielding a vector of A differential parameters obtained using the equation above.

In order to work with the same number of parameters for the PCA and for the Autoencoder, the weights used to calculate the code layer will be the ones compared across different data versions. Hence, the correlation will be calculated between pairs of the deepest encoding vectors, i.e., $\mathbf{p}_{a,AE} = \boldsymbol{\phi}_{a,K_H}^{(N_L)}$, with N_L being the number of hidden layers and K_H being the number of nodes of the previous hidden layer.

IV. Results

All analyses presented in this paper have been performed using R version 4.2.0 (2022-04-22), and tested locally on a MacBook Pro (2021) with Apple M1 Pro, 8-core CPU, 14-core GPU, 16GB of RAM, and 512GB SSD. For longer running computations, we have used a 40-core Intel Xeon Processor E5-2650 v3 server running Ubuntu 20.04.4. All code and data used are openly available from https://github.com/SMARDY-NCI/ADV.

In this work, the outcomes of both approaches are evaluated on two open data repositories:

- The *Dublin Footfall* dataset has pedestrian footfall counts registered in the city of Dublin from January 1st to April 3rd 2022 ($N = 288, K = 30, A = 4$). This dataset contains the counts of pedestrians passing by 30 streets in Dublin [23].
- The *Air Quality* dataset ($N = 827, K = 13, A = 6$) contains the hourly averaged measurements of a gas multi-sensor device deployed on the field in an Italian city [24], [25].

Both datasets contain time series data, which is a frequent type of resource in the Internet Of Things paradigm. Hence, we consider that although this work focuses on this particular type of data, it is still a relevant and significant problem encountered by the data science community.

A common methodological approach was used for both datasets. Firstly, an exploratory analysis was carried out to obtain clean matrices without any potential artifacts, such as missing values or outliers. Next, a PCA was fitted with

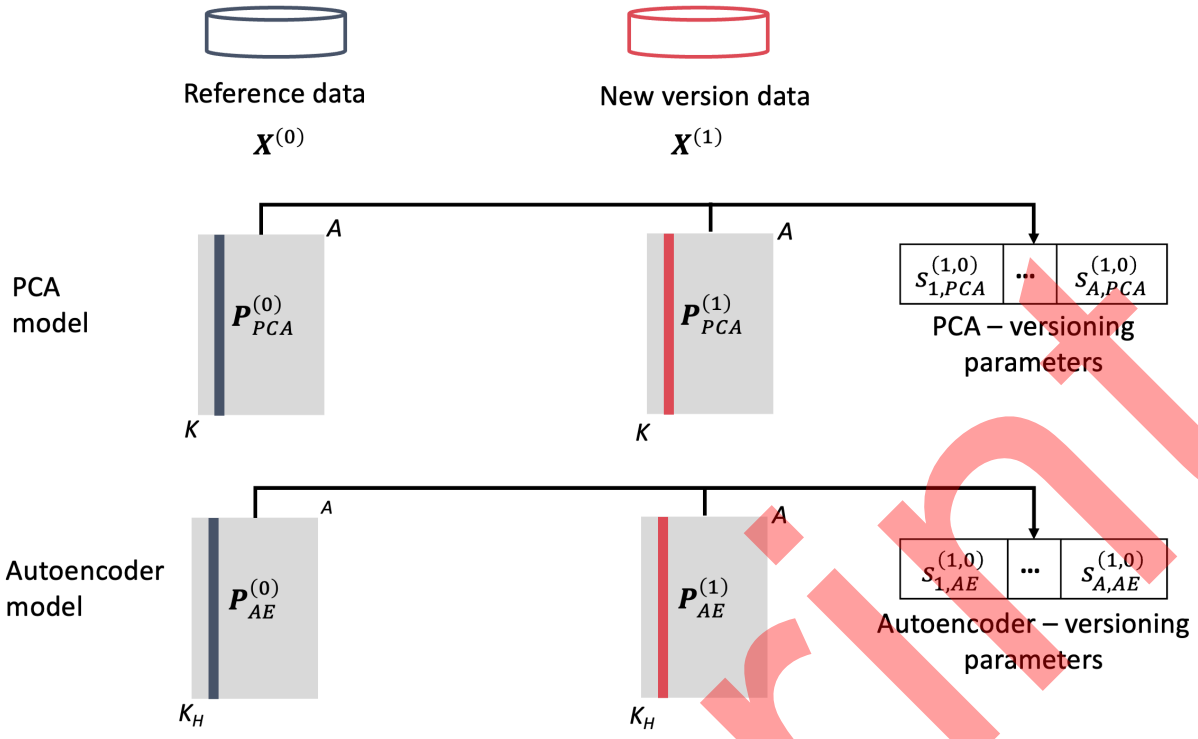

Fig. 3. Calculation of versioning parameters comparing the reference data ($\mathbf{X}^{(0)}$) and a newer version ($\mathbf{X}^{(1)}$) by means of a version-similarity vector $\mathbf{s}^{(1,0)}_{1,\ldots,A}$, based on the comparison of PCA or Autoencoder models of each version.

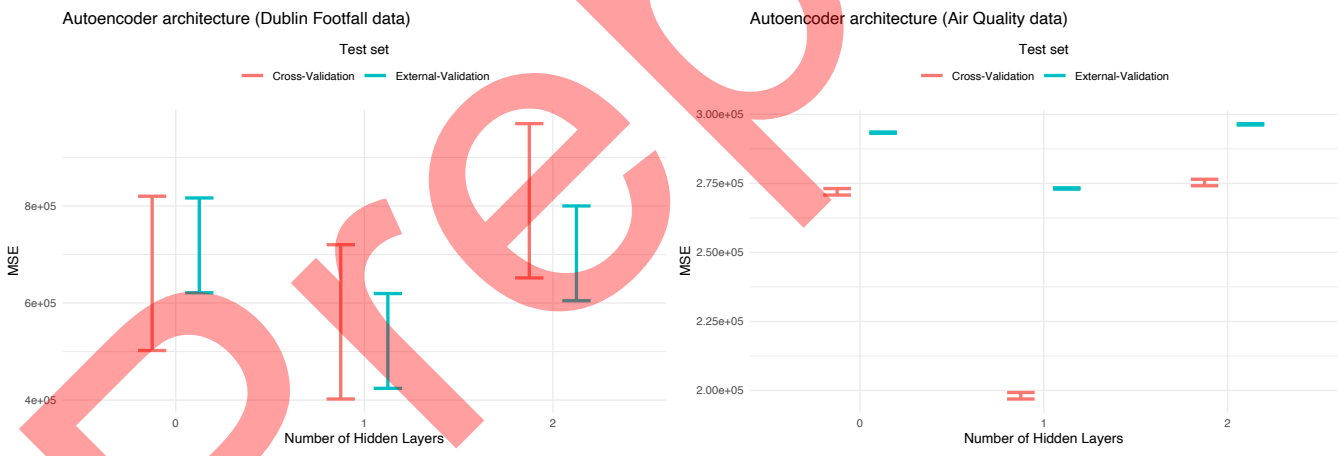

Fig. 4. Average MSE and LSD intervals as a function of the number of hidden layers of the Autoencoder between the input and the latent layers (and between the latent and the output layers), for the *Dublin Footfall* (left) and the *Air Quality* (right) datasets.

each dataset, in order to estimate the latent dimension A. The latent dimension was selected considering a trade-off between the explained variability and the information added by each principal component. In this case, principal components were added until at least 80% of the total variability was explained, and ceased to be added when new principal components explained no more than 2-3% of the variability. Table I shows the cumulative explained variance for each dataset as a function of the number of principal components. Following the aforementioned criteria, the latent dimension was set to $A = 4$ for the *Dublin Footfall* dataset, and to $A = 3$ for the *Air Quality* dataset.

This outcome yielded by the optimization of the PCA model, was used to determine also the latent dimension of the Autoencoder, i.e., A was used as the number of nodes of the "code" layer. The parameters considered in this optimization were the number of hidden layers of the neural network architecture, their number of nodes and also their activation

TABLE I
CUMULATIVE VARIANCE EXPLAINED BY EACH PRINCIPAL COMPONENT OF THE PCA MODELS (%)

Data set	Principal Components (A)				
	PC 1	*PC 2*	*PC 3*	*PC 4*	*PC 5*
Dublin Footfall	69.2019	75.8691	79.8526	83.3379	85.7898
Air Quality	71.4170	85.8391	93.0434	95.5752	97.0972

functions. The loss metric used as the objective function to fit the model parameters was the Mean Squared Error (MSE). To assess the uncertainty in the MSE values, a double cross-validation procedure was followed. This double cross-validation set aside an eternal-validation set with 10% of observations. Then, with the remaining 90%, a k-folds cross-validation division is applied, dividing the set in k sets. Then, in each round of the cross-validation scheme, models are trained with $k-1$ folds, and tested with the k left-out cross-validation set and with the previous external-validation set.

Fig. 5. Loss criteria as a function of the epochs obtained during the training of the reference Autoencoders, (i.e., with the version (0) data, without any artifacts applied), for the *Dublin Footfall* (top panel) and the *Air Quality* (bottom panel) dataset.

Figure 4 shows the results trying several configurations of the Autoencoder's architecture. For both datasets, the optimal architecture minimizing the MSE is achieved with one hidden layer. For the *Dublin Footfall* dataset (Fig. 4, above), the MSE values are considerably more overlapped than for the *Air Quality* dataset. The overlap between all three cross-validation MSE LSD intervals, means that there were not statistically significant differences between MSE values yielded by each architectures. However, the MSE values obtained with the external-validation set, served to double-check the analysis with results that removed the validation-set as a source of variability. Indeed, when the width of the LSD intervals is reduced, the differences between the MSE values become more significant, pointing more clearly that one hidden layer is associated with significantly lower MSE values.

This approach was also followed for the *Air Quality* dataset (Fig. 4, below). In this second case, the differences between architectures were clearly pointing, both with the cross-validation and the external-validation MSE, that one hidden layer was yielding the lowest MSE values. As a result, a single hidden layer was chosen for the architecture of the Autoencoder for both the *Dublin Footfall* and the *Air Quality* dataset. It is important to mention as well, that the chosen activation function the rectified linear ("ReLu") one. This activation function returns the maximum value between zero and its input value, ensuring that its output will always be a positive number, which has good repercussions to deal with vanishing-gradient effects, improving the efficiency during the training process of the Autoencoders.

Finally, once the architecture of the Autoencoders was optimised, the number of epochs used for the Autoencoders' training was set in order to ensure the convergence of the loss criteria. As it can be seen in Fig. 5, around 20 epochs were enough to ensure the convergence, but 50 epochs were finally set to give some flexibility in case modified versions of the datasets required more iterations until model convergence.

In each one of the following experiments, sampling approaches were applied to measure the uncertainty on the model estimates, varying the subset of observations used to fit both models. A number of ten hold-out repetitions was used to vary training and validation samples. Afterwards, an ANOVA test [26] was carried out and LSD intervals (calculated at a 95% of confidence level) were obtained to assess the statistical significance (or not) of the differences between the versioning parameters' values in each case. Two factors could be considered in the ANOVA: the *Method* (PCA or Autoencoder), and, if applicable, the *Artifact level*, which could vary depending on each case, i..e, percentage of missing cells, or deleted rows. The *Repetition* was also considered as a random factor in the ANOVA, since a different sample was being used to fit the models in each hold-out sample. When the *Artifact level* factor was present, the random factor *Repetition* was modeled as a factor nested on the *Artifact level* factor.

A. Case I: imputation of missing data

In this case, different levels of missing data completely at random (MCAR) were simulated. This type of missingness assumes that a random percentage of cells in the matrix is empty due to completely random effects [27]. Among the approaches to impute missing values, in this case the algorithm Trimmed Squares Regression was used, given its good performance with several types of datasets and because of its code availability [28]. Missing data are very frequent artifacts in the original versions of datasets. In fact, in both original

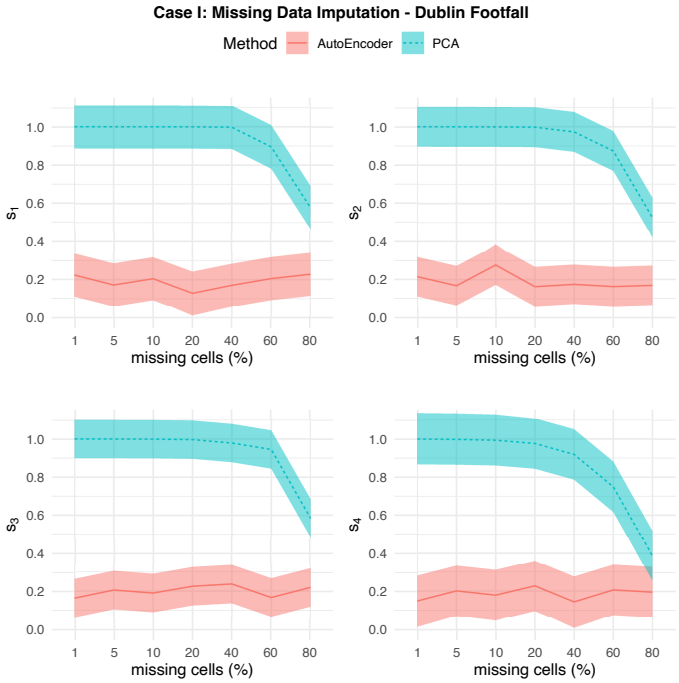

Fig. 6. Averaged version-similarity scores (solid and dashed lines) and their LSD intervals (shadowed area) as a function of the method and the percentage of generated missing data for each one of the four loading and encoding vectors extracted from the *Dublin Footfall* dataset.

datasets included in this work, missing data were present, but they were not imputed to avoid the potential distortions in the results if comparisons were performed between two imputed datasets. Thus, the "reference" dataset version had to be free of any potential distortion, and only complete cases (i.e., the original datasets were pre-processed to initially remove rows containing missing data values) were considered when developing both the PCA and Autoencoder models.

The results in Fig. 6 and Fig. 7, show that the similarity scores between the original and the new versions remain stable over the range of missing data percentages. However, the similarity values for the Autoencoder parameters are significantly lower than the similarity values yielded by the PCA model. One possible explanation is that the higher flexibility of the Autoencoder (with more parameters to be trained), results also in higher instability when the information used to fit the model slightly changes. To avoid this effect, some pre-processing steps could be studied to make more robust the performance of Autoencoders.

Nevertheless, the similarity values yielded by the PCA model are able to capture the common information shared between the original complete dataset and the new versions obtained by imputing generated missing values. The LSD intervals of the similarity values, contain the correlation value of 1.0 up to a 40% of missing values. From that percentage of missingness, the average similarity values start to decay, and from a 60% of missing values, the LSD intervals are not overlapped anymore with previous similarity scores. Note that, results with the *Air Quality* dataset (Fig. 7), show much more stability in the different repetitions performed, which results also in thinner LSD intervals.

This parsimonious behaviour showing a monotonic decrease of the similarity values along with the increase of the missingness (the *Artifact level*), is desirable and useful to track the differences between versions. Moreover, the quantification of the uncertainty enables a quantitative assessment on the "substantial" differences between versions. Retaking the *revision* term discussed in the Introduction, in this case it could be applied for versions obtained from a 60% of missing values, arguing that "substantial" differences had been found between the new and the original versions, i.e.: LSD intervals of the similarity indices do not contain anymore the 1.0 similarity value.

B. Case II: transformation of values

In this case, two types of transformations were applied to the original dataset. The first one was a row-wise transformation applied to the *Dublin Footfall* dataset, expressing all the variables as percentages, i.e., re-expressing the number of pedestrians per day as the fraction of the total number of pedestrians measured that day. In this case, error bars are used to represent the results, since all rows had to be affected by the transformation, and there were not different levels of the generated artifact.

Figure 8 shows the similarity indices for each principal component. In comparison to the Case I, Case II shows a noticeable decrease in the similarity between versions of the *Dublin Footfall* dataset, with a closer agreement between the values of the Autoencoder-based similarity and of the PCA-based similarity. This result exemplifies a case in which the differences on the numerical information between the original and the new version would be clearly substantial. Moreover, it is also worth to mention that versions obtained in Case II are far more different from the original dataset than versions yielded by Case I (and also by Case III).

The second transformation was a column-wise transformation: a logarithmic (reference) transformation. This is a type of non-linear transformation usually applied to normalize skewed data. The *Air Quality* dataset contains information about the concentration of several compounds and other positive magnitudes, making this dataset a good candidate for such logarithmic transformations. In this case, the experiment contemplated different percentages of transformed columns. Moreover, ten repetitions were performed at each percentage of transformed columns, changing the affected columns and therefore, the resulting datasets.

Figure 9 illustrates that version-similarity parameters extracted from the PCA model are more stable over the range of transformed columns than the ones extracted from the Autoencoder. This is probably due to the fact that PCA is accounting the covariance structure of the dataset. Thus, as far as the transformation does not distort the correlation pattern (as it happened in the row-wise transformation case, Fig. 8), PCA will be robust to such transformations. This is why similarity

Fig. 7. Averaged version-similarity scores (solid and dashed lines) and their LSD intervals (shadowed area) as a function of the method and the percentage of generated missing data for each one of the three loading and encoding vectors extracted from the *Air Quality* dataset.

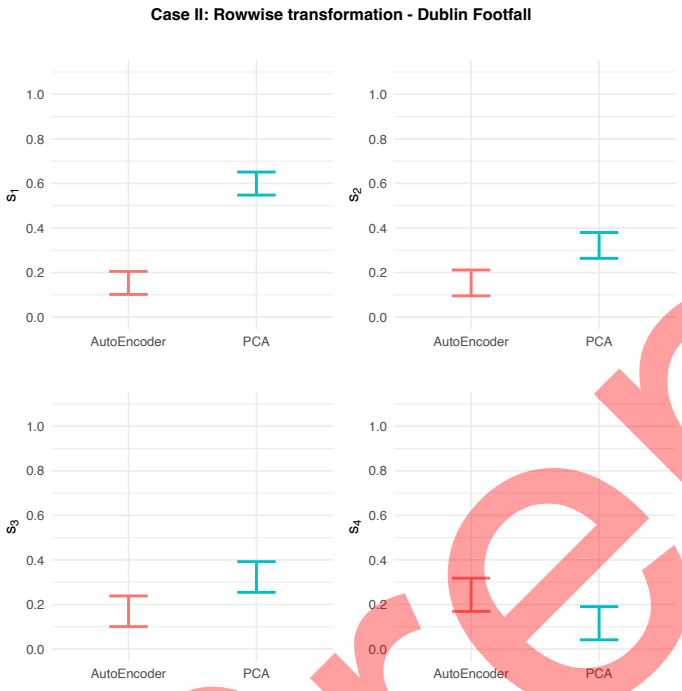

Fig. 8. Averaged version-similarity scores and their LSD intervals as a function of the method for each one of the four loading and encoding vectors extracted from the *Dublin Footfall* set after converting the footfall counts to percentages relative to the total counts of each day.

values in Fig. 9 are higher than the ones obtained in Fig. 8, with version-similarity values not showing any substantial difference between altered and reference datasets. In contrast, Autoencoder results show once more a lower correlation value and, in this particular case, a less parsimonious behaviour along the range of transformed columns.

C. Case III: subsetting of rows

In this final scenario, a different percentage or rows were removed from the dataset, to analyse whether or not there was a sample size effect on the proposed set of statistical parameters.

Results from Fig. 10 resemble the ones from Fig. 6. Once again, both similarity metrics show a steady behaviour as a function of the percentage of deleted rows, and the PCA similarity metric also shows a major agreement between the original and the new versions obtained by sub-setting rows. Only for the similarity metric obtained for the fourth latent variable (s_4), there is a substantial difference between the 60% and the 80% of removed rows.

This outcome could be a useful insight to determine if studies based on different observations of a dataset, could be comparable or not. Another way to look at this case of study could be the symmetric scenario, when a dataset is updated with new observations. In this case, tracking the agreement between the original and the new updated version, could serve to check that any substantial changes are affecting the new information.

Finally, results from Fig. 11 illustrate a very similar scenario, with the PCA versioning parameters being significantly more similar to the reference values than the Autoencoders' versioning parameters. Besides, Case III with the *Air Quality* dataset also resembles the results from the column-wise transformation in Case II, with the versioning parameters relative to the three latent variables, showing stability over the whole range of deletion percentages. Moreover, the similarity values s_1 for the Autoencoder are the highest ones among the range of covered scenarios in this work, suggesting that row-wise deletion is one of the operations least affecting the model's coefficients.

V. Conclusions and Future Work

In this work, we have proposed a similarity index to quantitatively assess the difference between data versions. The concept of versioning is a pillar of the data provenance information, which is a concept deeply embedded in the FAIR principles guidelines and philosophy. Nonetheless, the available terms to include this information in the data and metadata are still ambiguous and do not offer any insight about the changes on the numerical information between different data versions.

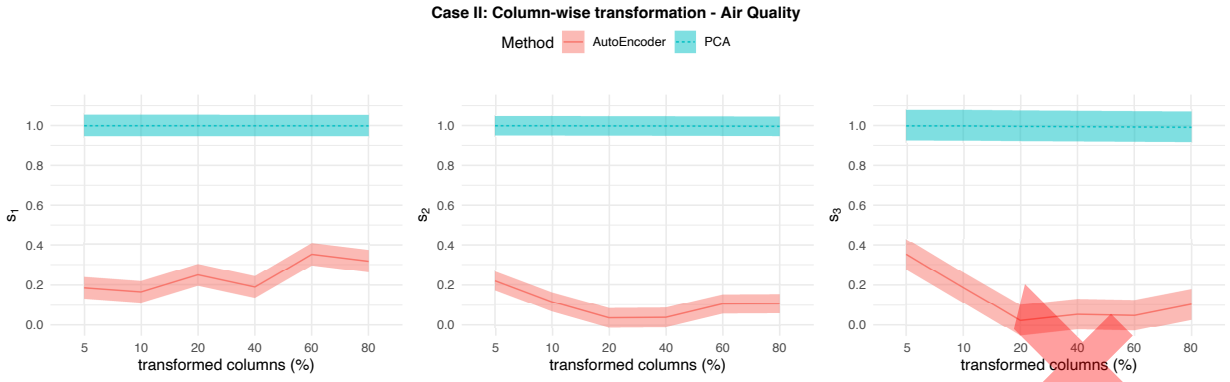

Fig. 9. Averaged version-similarity scores (solid and dashed lines) and their LSD intervals (shadowed area) as a function of the method for each one of the three loading and encoding vectors extracted from the *Air Quality* dataset after converting some air quality parameters to logarithmic scale.

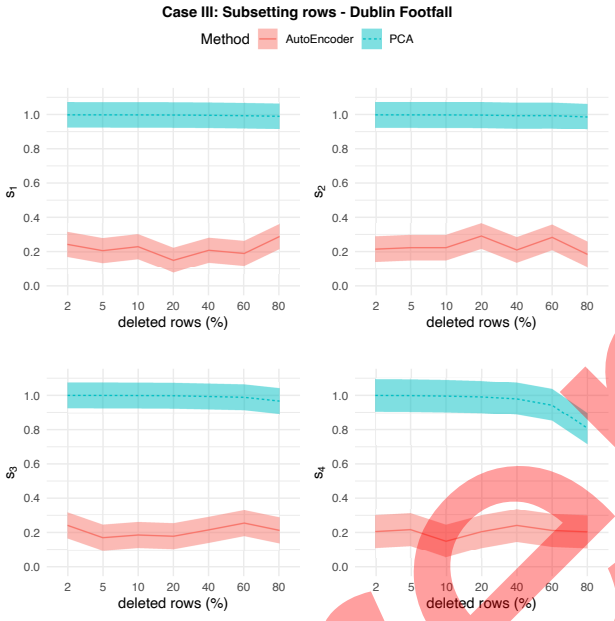

Fig. 10. Averaged version-similarity scores (solid and dashed lines) and their LSD intervals (shadowed area) as a function of the method and the percentage of rows deleted from the training set for each one of the four loading and encoding vectors extracted from the *Dublin Footfall* dataset.

To address this issue, we have formulated a similarity metric to track changes on the information of different data versions. Two different dimensionality-reduction models were used to compute the similarity metric: Autoencoders and Principal Component Analysis. The goal was to analyse the potential of this approach to quantitatively assess the differences between data versions. Three versioning scenarios were considered, generating: (*I*) cellwise differences by missing data imputation, (*II*) rowwise and columnwise transformations by re-expressing rows and columns in different units, and (*III*) sample size reduction by selecting subsets of observations.

The results illustrated the behaviour of the proposed similarity metrics to track changes in the information. The PCA-based similarity index showed a better agreement in general between the artificially generated versions and the original dataset. Moreover, the differences between Cases I-III and Case II, also showed how the proposed versioning framework reflects the nature of the transformations, objectively assessing a major or minor versioning based on the difference between the numerical information.

In our FAIR-compliant versioning service, the nomenclature is also aligned with the Dublin Core metadata [29], which includes elements to define the versions and relationships between datasets. Thus, some future work will include the seamless integration between standard versioning tags and the set of the proposed versioning parameters to form complete data workflows using Open Science repositories e.g. Zenodo. Another area deserving further research is to investigate the implications for comparability when similarity results are extracted from different data versions.

In conclusion, this work has arguably presented some promising results about the use of a quantitative, measurable and automatic versioning system of time series datasets. Further research is needed to contemplate other versioning scenarios (e.g. new versions based on subsets of variables, combinations of several transformations, etc.), more data types, and most importantly, to articulate how the proposed versioning parameters can be included terms of provenance metadata standards.

ACKNOWLEDGMENTS

This work has been developed under the auspices of the following European research projects: i) "EUREKA SMARDY Marketplace for Technology Transfer of R&I Data, Software, and Results" (EUREKA ID: E!13437) https://smardy-project.eu funded in Ireland from 2021 to 2024 via the International Research Fund of Enterprise Ireland; ii) "CHIST-ERA SPuMoNI Smart Pharmaceutical Manufacturing" (CHIST-ERA BDSI Call 2017) http://www.spumoni.eu funded in Ireland from 2019 to 2022 via the Irish Research Council; and, iii) "ERASMUS+ TrainRDM TrainRDM – Open Science and Research Data Management Innovative and Distributed Training Programme" (2020-1-RO01-KA203-080170) https://rdmtraininghub.eu funded from 2020 to 2023.

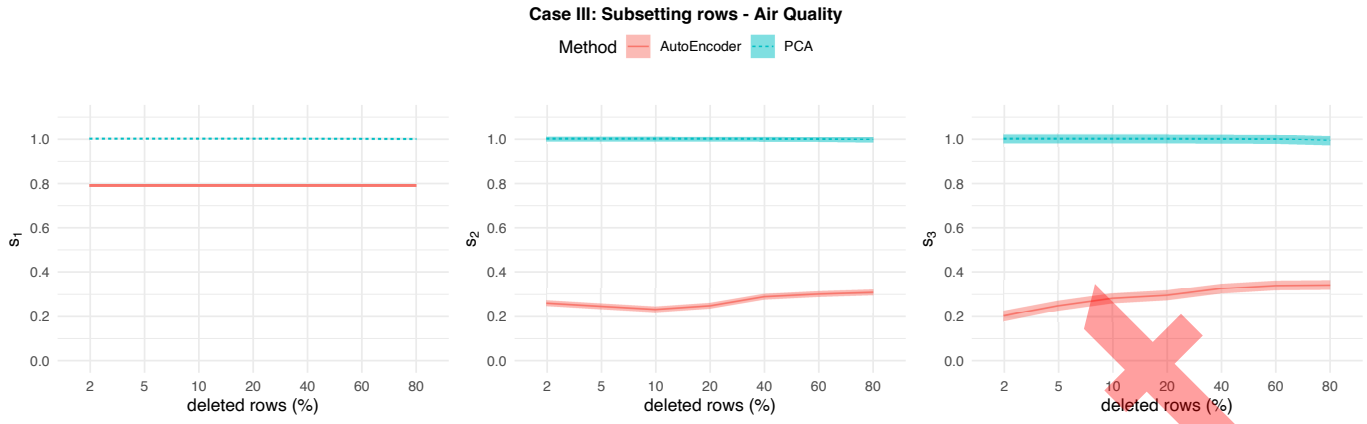

Fig. 11. Averaged version-similarity scores (solid and dashed lines) and their LSD intervals (shadowed area) as a function of the method and the percentage of rows deleted from the training set for each one of the three loading and encoding vectors extracted from the *Air Quality* dataset.

References

[1] T. Hey, S. Tansley, K. Tolle, and J. Gray, *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Redmond: Microsoft Research, 2009, ISBN: 978-0-9825442-0-4.

[2] L. Moreau, P. Groth, J. Cheney, T. Lebo, and S. Miles, "The rationale of PROV," *Journal of Web Semantics*, vol. 35, pp. 235–257, 2015.

[3] M. D. Wilkinson, M. Dumontier, I. Jan Aalbersberg, G. Appleton, M. Axton *et al.*, "The FAIR guiding principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 160018, pp. 1–9, 2016.

[4] A. Jacobsen, R. de Miranda Azevedo, N. Juty, D. Batista *et al.*, "FAIR principles: Interpretations and implementation considerations," *Data Intelligence*, vol. 2, no. 1-2, pp. 10–29, 2020.

[5] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37–52, 1987.

[6] I. T. Jolliffe, "Discarding variables in a Principal Component Analysis. I: artificial data," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 21, no. 2, pp. 160–173, 1972.

[7] G. E. Hinton and R. S. Zemel, "Autoencoders, minimum description length and Helmholtz free energy," in *NIPS'93*. Denver: Morgan Kaufmann, Nov. 1993, pp. 3–10.

[8] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel, "The Helmholtz Machine," *Neural Computation*, vol. 7, no. 5, pp. 889–904, 1995.

[9] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE Journal*, vol. 37, no. 2, pp. 233–243, 1991.

[10] B. Pérez, J. Rubio, and C. Sáenz-Adán, "A systematic review of provenance systems," *Knowledge and Information Systems*, vol. 57, pp. 495–543, 2018.

[11] Y. L. Simmhan, B. Plale, and D. Gannon, "A survey of data provenance in e-Science," *ACM SIGMOD Record*, vol. 34, no. 3, pp. 31–36, 2005.

[12] J. F. Pimentel, J. Freire, L. Murta, and V. Braganholo, "A survey on collecting, managing, and analyzing provenance from scripts," *ACM Computing Surveys*, vol. 52, no. 3, pp. 1–38, 2019.

[13] S. Bhattacherjee, A. Chavan, S. Huang, A. Deshpande, and A. Parameswaran, "Principles of dataset versioning: Exploring the recreation/storage tradeoff," *Proceedings of the VLDB Endowment*, vol. 8, no. 12, pp. 1346–1357, 2015.

[14] D. Zhao, C. Shou, T. Maliky, and I. Raicu, "Distributed data provenance for large-scale data-intensive computing," in *CLUSTER'13*. Indianapolis: IEEE, Sep. 2013, pp. 1–8.

[15] Y. Gil, J. Cheney, P. Groth, O. Hartig, S. Miles, L. Moreau, and P. Pinheiro da Silva, "Provenance XG final report," World Wide Web Consortium (W3C), W3C Incubator Group Report XGR-prov-20101214, Dec. 2010. [Online]. Available: https://www.w3.org/2005/Incubator/prov/XGR-prov-20101214/

[16] L. A. McGuinness, A. González-Cebrián, A. E. Chis, M. Bradford, and H. González-Vélez, "Automated data versioning using statistical machine learning," in *MDS'22*. San Diego: SIAM, Sep. 2022, (Poster).

[17] M. Dumontier, C. J. Baker, J. Baran, A. Callahan *et al.*, "The Semanticscience Integrated Ontology (SIO) for biomedical research and knowledge discovery," *Journal of Biomedical Semantics*, vol. 5, no. 14, pp. 1–11, 2014.

[18] J. Greenberg, "Metadata extraction and harvesting," *Journal of Internet Cataloging*, vol. 6, no. 4, pp. 59–82, 2004.

[19] R. S. Gonçalves, M. J. O'Connor, M. M. Romero, A. L. Egyedi *et al.*, "The CEDAR workbench: An ontology-assisted environment for authoring metadata that describe scientific experiments," in *ISWC 2017*, ser. LNCS, vol. 10588. Vienna: Springer, Oct. 2017, pp. 103–110.

[20] G. Lo Presti, O. Barring, A. Earl, R. M. Garcia Rioja *et al.*, "CASTOR: A distributed storage resource facility for high performance data processing at CERN," in *MSST 2007*. San Diego: IEEE, Sep. 2007, pp. 275–280.

[21] R. Albertoni, D. Browning, S. Cox, A. N. González-Beltrán, A. Perego, and P. Winstanley, "Data Catalog Vocabulary (DCAT)," World Wide Web Consortium (W3C), W3C Working Draft Version 3, May 2022. [Online]. Available: https://www.w3.org/TR/vocab-dcat-3/

[22] S. Cox, A. N. González-Beltrán, B. Magagna, and M. V. Marinescu, "Ten simple rules for making a vocabulary FAIR," *PLoS Computational Biology*, vol. 17, no. 6, pp. e1 009 041:1–15, 2021.

[23] Dublin City Council, "Pedestrian footfall," DCC Transport City Centre Projects, Dublin, Dataset, 2022, Created on: 2020-08-14. Last updated: 2022-05-04. Retrieved: 2022-05-30. [Online]. Available: https://data.smartdublin.ie/dataset/dublin-city-centre-footfall-counters

[24] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. Di Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario," *Sensors and Actuators B: Chemical*, vol. 129, no. 2, pp. 750–757, 2008.

[25] UC Irvine Machine Learning Repository, "Air quality data set," Center for Machine Learning and Intelligent Systems, Irvine, Dataset, 2016, Donated on: 2016-03-23.Retrieved: 2022-05-30. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Air+Quality

[26] G. E. P. Box, J. S. Hunter, and W. G. Hunter, *Statistics for Experimenters: Design, Innovation, and Discovery*, 2nd ed. Hoboken: John Wiley & Sons, 2005, ISBN: 978-0-471-71813-0.

[27] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*, 3rd ed. Hoboken: John Wiley & Sons, 2019, ISBN: 978-0-470-52679-8.

[28] A. Folch-Fortuny, F. Arteaga, and A. Ferrer, "PCA model building with missing data: New proposals and a comparative study," *Chemometrics and Intelligent Laboratory Systems*, vol. 146, pp. 77–88, 2015.

[29] DCMI Usage Board, "DCMI Metadata Terms," Dublin Core Metadata Initiative, DCMI Recommendation, Jan. 2020. [Online]. Available: https://www.dublincore.org/specifications/dublin-core/dcmi-terms/