A Tidy Framework and Infrastructure to Systematically Assemble Spatio-temporal Indexes from Multivariate Data

H. Sherry Zhang[1] , Dianne Cook[1] , Ursula Laa[2] , Nicolas Langrené[3] , Patricia Menéndez[1]

[1] Department of Econometrics and Business Statistics, Monash University, Melbourne, Victoria, Australia
[2] Institute of Statistics, University of Natural Resources and Life Sciences, Vienna, Austria
[3] Department of Mathematical Sciences, BNU-HKBU United International College, Zhuhai, Guangdong, China

ABSTRACT

- indexes, useful, quantify severity, early monitoring,
- A huge number of indexes have been proposed by domain experts, however, a large majority of them are not being adopted, reused, and compared in research or in practice.
- One of the reasons for this is the plenty of indexes are quite complex and there is no obvious easy-to-use implementation to apply them to user's data.
- The paper describes a general pipeline framework to construct indexes from spatio-temporal data,
- This allows all the indexes to be constructed through a uniform data pipeline and different indexes to vary on the details of each step in the data pipeline and their orders.
- The pipeline proposed aim to smooth the workflow of index construction through breaking down the complicated steps proposed by various indexes into small building blocks shared by most of the indexes.
- The framework will be demonstrated with drought indexes as examples, but appliable in general to environmental indexes constructed from multivariate spatio-temporal data

1. Introduction

Indexes are commonly used to combine and summarize different sources of information into a single number for monitoring, communicating, and decision-making. Examples of those include the Air Quality Index, El Niño-Southern Oscillation Index, Consumer Price Index, QS University Rankings or Human Development Index among many others.

CONTACT: H. Sherry Zhang. Email: huize.zhang@monash.edu.

To construct an index, experts typically start by defining a concept of interest that requires measurement. This concept often lacks a direct measurable attribute or can only be measure as a composite of various processes, yet it holds social and public significance. To create an index, once the underlying processes involved are identified, relevant and available variables are then defined, collected, and combined using statistical methods into an index that aims at measure the process of interest. The construction process is often not straightforward, and decisions need to be made, such as the selection of variables to be included, which might depend on data availability and the statistical definition of the index to be used, among others. For instance, the indexes constructed from linear combination of variables need to decide on the weight assigned to each variable. Some indexes have a spatial and/or temporal components, and variables can be aggregated to different spatial resolutions and temporal scales, leading to various indexes for different monitoring purposes. Hence, all these decisions can result in different index values and have different practical implications.

To be able to test different decision choices systematically for an index, the index needs to be broken down into its fundamental building blocks to analyse the contribution and effect of each component. We call this process of breaking the index construction into different steps the index pipeline. Such decomposition of index components provides the means to standardise index construction via a pipeline and offers benefits for comparing among indexes and calculating index uncertainty. In social indexes for instance, the OECD handbook (OECD, European Union, and Joint Research Centre - European Commission 2008) has provided a set of steps and recommendations for constructing composite socio-economic indexes. However, there is still a need to extend these guidelines and methods to accommodate the inclusion of more methodological complex steps required for indexes in general.

In this work, we aim at providing statistical and computational methods to develop a data pipeline framework for constructing and customize indexes using data. As a companion the R package, tidyindex, is developed to construct and explore different. Furthermore, the tidyindex package can be used to explore the effects of changes on index definition, index construction steps, and construction methods. This work provides researchers actively developing new indexes or aiming at improving indexes with the tools to easily modify and evaluate indexes. It also helps index analysts diagnose indexes, carry out sensitivity analysis to assess small perturbations in the index, and identify potential weaknesses for methodology improvement.

The rest of the paper is structured as follows: Section 2 reviews the concept of data pipeline in R. The pipeline framework for index construction is presented in Section 3. Section 4 explains how to add new steps into the index methodology in the form of new building blocks inside the index construction pipeline. Examples are given in Section 5 to demonstrate the index construction with the pipeline built.

2. Tidy framework

The tidy framework consists of two key components: tidy data and tidy tools. The concept of tidy data (Wickham 2014) prescribes specific rules for organising data in an analysis, with observations as rows, variables as columns, and types of observational units as tables. Tidy tools, on the other hand, are concatenated in a sequence through which the tidy data flows, creating a pipeline for data processing and modelling. These

pipelines are data centric, meaning all the tidy tools or functions take a tidy data object as inputs and return a processed tidy data object, directly ready for the next operations to be applied. Also, the pipeline approach corresponds to the modular programming practice, which breaks down complex problems into smaller and more manageable pieces, as oppose to a monolithic design, where all the steps are predetermined and integrated into a single piece. The flexibility provided by the modularity makes it easier to modify certain steps in the pipeline and to maintain and extend the code base.

Examples of using a pipeline approach for data analysis can be traced back to the interactive graphics literature, including Buja et al. (1988); Sutherland et al. (2000); Xie, Hofmann, and Cheng (2014); Wickham et al. (2009). Wickham et al. (2009) argues that whether made explicit or not, a pipeline has to be presented in every graphics program and making them explicit is beneficial for understanding the implementation and comparing between different graphic systems. While this comment is made in the context of interactive graphics program, it is also applicable generally to any data analysis workflow. More recently, the tidyverse suite (Wickham et al. 2019) takes the pipeline approach for general-purpose data wrangling and has gained popularity within the R community. The pipeline-style code can be directly read as a series of operations applied successively on tidy data object, offering a method to document the data wrangling process with all the computational details for reproducibility.

Since the success of tidyverse, more packages have been developed to analyze data using the tidy framework for domains specific applications, a noticeable example of which is `tidymodels` for building machine learning models (Kuhn and Wickham 2020). To create a tidy workflow tailored to a specific domain, developers first need to identify the fundamental building blocks to create a workflow. These components are then implemented as modules, which can be combined to form the pipeline. For example, in supervised machine learning models, steps such as data splitting, model training and model evaluation are commonly used in most workflow. In the `tidymodels`, these steps are correspondingly implemented as package `rsample`, `parsnip`, and `yardstick`, agnostic to the specific model chosen. The uniform interface in tidymodels frees analysts from recalling model-specific syntax for performing the same operation across different models, increasing the efficiency to work with different models simultaneously.

For constructing indexes, the pipeline approach adopts explicit and standalone modules that can be assembled in different ways. Index developers can choose the appropriate modules and arrange them accordingly to generate the data pipeline that is needed for their purpose. The pipeline approach provides many advantages:

- makes the computation more transparent, and thus more easily debugged.
- allows for rapidly processing new data to check how different features, like outliers, might affect the index value.
- provides the capacity to measure uncertainty by computing confidence intervals from multiple samples as generated by bootstrapping to original data.
- enables systematic comparison of surrogate indexes designed to measure the same phenomenon.
- it may even be possible to automate diagramatic explanations and documentation of the index.

Adoption of this pipeline approach would provide uniformity to the field of index development, research and application.

3. Details of the index pipeline

In constructing various indexes, the primary aim is to transform the data, often multivariate, into a univariate index. Spatial and temporal considerations are also factored into the process when observational units and time periods are not independent. However, despite the variations in contextual information for indexes in different fields, the underlying statistical methodology remains consistent across diverse domains. Each index can be represented as a series of modular statistical operations on the data. This allows us to decompose the index construction process into a unified pipeline workflow with a standardized set of data processing steps to be applied across different indexes.

An overview of the pipeline is presented in Figure 1, illustrating the nine available modules designed to obtain the index from the data. These modules include operations for temporal and spatial aggregation, variable transformation and combination, distribution fitting, benchmark setting, and index communication. Analysts have the flexibility to construct indexes by connecting modules according to their preferences.

Now, we introduce the notation used for describing pipeline modules. Consider multivariate spatio-temporal process,

$$\mathbf{x}(s;t) = \{x^1(s;t), x^2(s;t), \cdots, x^p(s;t)\} \qquad s \in D_s \subseteq \mathbb{R}^m, t \in D_t \subseteq \mathbb{R}^n \qquad (1)$$

where:

- $x^i(s,t)$ represents a variable of interest for example precipitation, $i = 1, \cdots, p$ and

- s represents the geographic locations in the space $D_s \subseteq \mathbb{R}^m$. Examples of geographic locations include a collection of countries, longitude and latitude coordinates or regions of interest and,

- t denotes the temporal order in $D_t \subseteq \mathbb{R}^n$. For instance, time measurements could be recorded hourly, yearly, monthly, quaterly or by season.

In what follows when geographic or temporal components of the $x(s,t)$ process are fixed we will be using suffix notation. For example, $x_s(t)$ represents the data for a fixed location s as a function of time t. While $x_t(s)$ denotes the spatial varying proccess for a fixed t. An overview of the notation for pipeline input, operation, and output is present in Table 1.

Table 1. An notation overview of the input, operation, and output of each pipeline module.

No.	Module	Input	Operation	Output
1	Temporal processing	$x_s^i(t)$	$f[x_s^i(t)]$	$\tilde{x}_s^i(t')$, where $t' \in D_{t'}$
2	Spatial processing	$x_t^i(s)$	$g[x_t^i(s)]$	$x_t^i(s')$, where $s' \in D_{s'}$????
3	Variable transformation	$x^i(s;t)$	$T[x^i(s;t)]$	$Tx^i(s;t)$????
4	Scaling	$x^i(s;t)$	$[x^i(s;t) - \alpha]/\gamma$	$Sx^i(s;t)$????

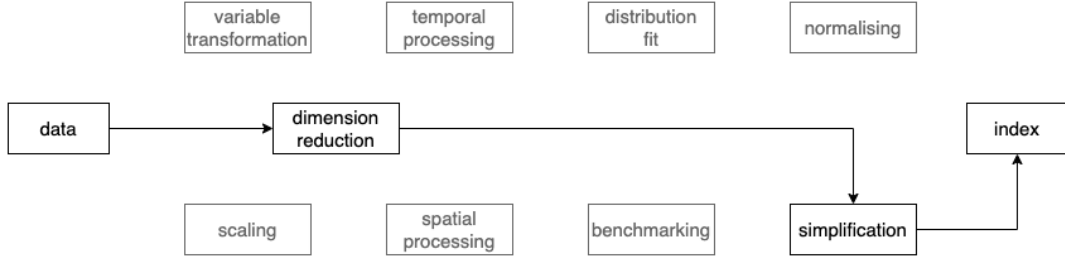| No.	Module	Input	Operation	Output
5	Dimension reduction	$\mathbf{x}(s;t)$, where $\mathbf{x} \subseteq \mathbb{R}^q,\ q \leq p$	$h[\mathbf{x}(s;t)]$	$\mathbf{y}(s;t)$, where $\mathbf{y} \subseteq \mathbb{R}^d$ and $d < q$
6	Distribution fit	$x^i(s;t)$	$F[x^i(s;t)]$, where F is the CDF	$p^i(s;t)$, where $p(.) \in [0,1]$ is a probability value
7	Normalising	$x^i(s;t)$	$\Phi^{-1}[x^i(s;t)]$	$z^i(s;t)$, where z has a normal distribution
8	Benchmarking	$x^i(s;t)$	$u[x^i(s;t)]$	$b^i(s;t)$
9	Simplification	$x^i(s;t)$	$v[x^i(s;t)]$	$A^i(s;t) \in \{a_1, a_2, \cdots, a_j\}$

Figure 1. Diagram of pipeline modules for index construction. The highlighted path illustrates one possible construction using the dimension reduction and simplification module.

3.1. *Temporal processing*

The temporal processing module takes as input argument a single variable $x_s^i(t)$ at location s as a function of time. In this step the original time series can be transformed or summarized into a new one via time aggregation, the transformation is represented by the function f, $\tilde{x}_s^i(t') = f[x_s^i(t)]$ where t' refers to the new temporal resolution after aggregation. An example of a temporal processing done in the computation of the Standardized Precipitation Index (SPI), consists of summing the monthly precipitation series over a rolling time window of size k. That is also known as the time scale. For SPI, the choice of the time scale k is used to control the accumulation period for the water deficit, enabling the assessment of drought severity across various types (meteorological, agricultural, and hydrological).

3.2. *Spatial processing*

The spatial processing module takes a single variable with a fixed temporal dimension, $x_t(s)$, as input. This step transforms the variable from the original spatial dimension s into the new dimension $s' \in D_{s'}$ through $x^{\#}(s;t) = g[x(s;t)]$. The change of spatial dimension allows for the alignment of variables collected from different measurements, such as in-situ stations and satellite imagery, or originating from different resolutions. This also includes the aggregation of variables into different levels, such as city, state, and countries scales.

5

3.3. *Variable transformation*

Variable transformation takes input of a single variable $x^i(s;t)$ and reshapes its distribution using the function $T[.]$ to produce $x^i(s;t)$. When a variable has a skewed distribution, transformations such as log, square root, or cubic root can adjust the distribution towards normality. For example, in Human Development Index (HDI), a logarithmic transformation is applied to the variable Gross National Income per capita (GNI), to reduce its impact on HDI, particularly for countries with high GNI values.

3.4. *Scaling*

Unlike variable transformation, scaling maintains the distributional shape of the variable. It includes techniques such as centering, z-score standardisation, and min-max standardisation and can be expressed as $[x^i(s;t) - \alpha]/\gamma$. In Human Development Index (HDI), the three dimensions (health, education, and economy) are converted into the same scale (0-1) using min-max standardisation.
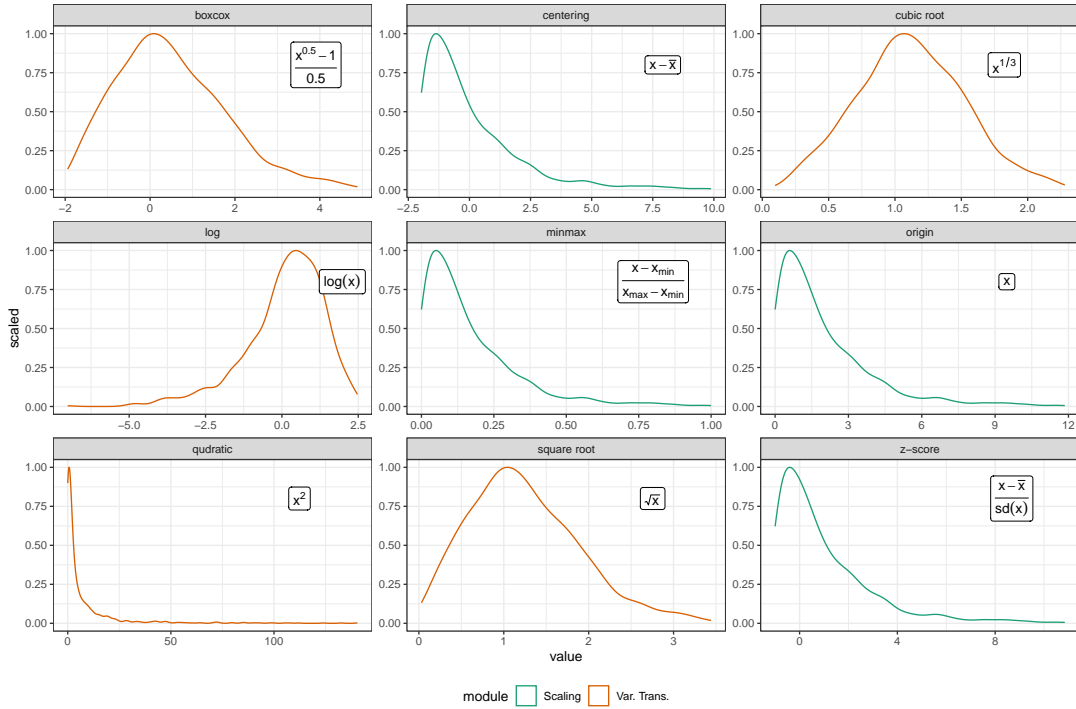

Figure 2. Comparison of the module scaling (green) and variable transformation (orange). While both modules change the variable range, scaling maintains the same distributional shape, which is not the case with variable transformation.

3.5. *Dimension reduction*

Dimension reduction takes the multivariate information $\mathbf{x}(s;t)$, where $\mathbf{x} \subseteq \mathbb{R}^q$, $q \leq p$, as the input. It summarises the high-dimensional information into a lower dimension representation $\mathbf{y}(s;t)$, where $y \subseteq \mathbb{R}^d$ and $d < q$, as the output. The transformation can be based on domain-specific knowledge, originating from theories describing the underlying physical processes, or guided in statistical methods. For example, the Standardized

Precipitation-Evapotranspiration Index (SPEI) calculates the difference D between precipitation (P) and potential evapotranspiration (PET), using a water balance model ($D = P - \text{PET}$). This is the only step that differs from the Standardized Precipitation Index (SPI).

3.6. *Distribution fit*

Distribution fit applies the cumulative distribution function F of a distribution on the variable $x^i(s;t)$ to obtain the probability values $p^i(s;t) \in [0,1]$. In SPEI, many distributions, including log-logistic, pearson III, lognormal, and general extreme distribution, are candidates for the aggregated series. Different fitting methods, and different goodness of fit tests may be used to compare the distribution choice on the index value.

3.7. *Normalising*

3.8. *Benchmarking*

Benchmark sets a value $b^i(s,t)$ for comparing against the original variable $x^i(s;t)$. This benchmark can be a fixed value consistently across space and time or determined by the data through the function $u[x^i(s;t)]$. Once a benchmark being set, observations can be highlighted for adjustments in other modules or can serve targets for monitoring and planning.

3.9. *Simplification*

Simplification takes a continuous variable $x^i(s;t)$ and categorises it into a discrete set $A^i(s;t) \in \{a_1, a_2, \cdots, a_j\}$ through a piecewise function,

$$
v[x_i(s;t)] = \begin{cases}
a_0 & C_1 \leq x^i(s;t) < C_0 \\
a_1 & C_2 \leq x^i(s;t) < C_1 \\
a_2 & C_3 \leq x^i(s;t) < C_2 \\
\cdots & \\
a_z & C_z \leq x^i(s;t)
\end{cases}
\tag{2}
$$

This is typically used at the end of the index pipeline to simplify the index to communicate to the public the severity of the concept of interest measured by the index. An example of simplification is to map the calculated SPI to four categories: mild, moderate, severe, and extreme drought.

4. Software design

5. Examples

This section uses the example of drought and social indexes to show the analysis made possible with the index pipeline. The drought index example computes two indexes

with various time scales and distributions simultaneously using the pipeline framework to understand the flood and drought events in Queensland. The social index example focuses on the dimension reduction in Global Gender Gap Index to explore the impact of weight changes in linear combination on index value and country ranking.

5.1. *Every distribution, every scale, every index all at once*

A common task for drought researchers is to compute indexes at different parameter combinations. This can be used to identify the spatial and temporal extent of drought events, recommend the best parameter choice, or compare the effectiveness of indexes for monitoring drought. The example below computes two indexes: SPI and SPEI, at various time scales and fitted distributions, for stations in the state of Queensland in Australia. The purpose of the example is to demonstrate the interfaces the tidyindex package built to allow easy computing at different parameter combinations.

The state of Queensland in Australia is frequently affected by natural disaster events such as flood and drought, which can have significant impacts on its agricultural industry. This study uses daily data from Global Historical Climatology Network Daily (GHCND), accessed via the package `rnoaa` to examine drought/flood condition in Queensland. Daily data is average into monthly and stations are excluded if monthly data contains missings, which is required for calculating both SPI and SPEI. This gives 29 stations with complete records from 1990 January to 2022 April.
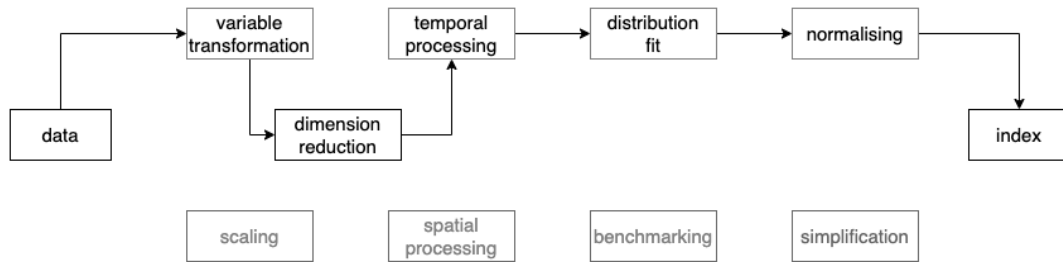

Figure 3. Diagram of pipeline steps for index construction. will need to be updated with better design and the distribution fitting step.

The function `compute_indexes()` can be used to collectively compute multiple indexes. The tidyindex offers wrapper functions, with the prefix idx_, that simplify the calculation of commonly used indexes by combining a set of pipeline steps into a single function. For example, the function `idx_spei()` includes the five steps previously described in **?@sec-toy-example** (variable transformation, dimension reduction, temporal aggregation, distribution fit, and normalise). Each `idx_xxx()` function specifies the relevant parameters relevant to the index: the thornthwaite method is used to calculate PET in SPEI, with the average temperature (`tavg`) and latitude (`lat`) used as inputs. The SPEI is computed at four time scales (6, 12, 24, and 36 months) and fitted with two distributions (Log-logistic and General Extreme Value (GEV)). The SPI is also computed at the same four time scales and uses the default gamma distribution to fit the aggregated series.

```
.scale <- c(6, 12, 24, 36)
idx <- queensland %>%
```

```
init(id = id, time = ym) %>%
compute_indexes(
  spei = idx_spei(
    .pet_method = "thornthwaite", .tavg = tavg, .lat = lat,
    .scale = .scale, .dist = c(gev(), loglogistic())),
  spi = idx_spi(.scale = .scale)
)
```

The output from `compute_indexes()` contains index values and associated parameter in a long tibble. It includes the original variables (`id`, `ym`, `prcp`, `tmax`, `tmin`, `tavg`, `long`, `lat`, and `name`), index parameters (`.idx`, `.scale`, `.method`, and `.dist`), intermediate variables (`pet`, `.agg`, and `.fitted`), and the final index (`.index`). This data can be visualised across space or time, or simultaneously, to explore the wet/dry condition in Queensland. Figure 4 visualises the spatial distribution of SPI at two periods (2010 October - 2011 March and 2019 October - 2020 March) with significant natural disaster events: 2010/11 Queensland flood and 2019 Australia drought, which contributes to the notorious 2019/20 bushfire. Figure 5 displays the sensitivity of the SPEI series for one particular station, Texas post office, at different time scales and fitted distributions. These two plots demonstrate some possibilities to explore the indexes after they are computed from `compute_indexes()`.
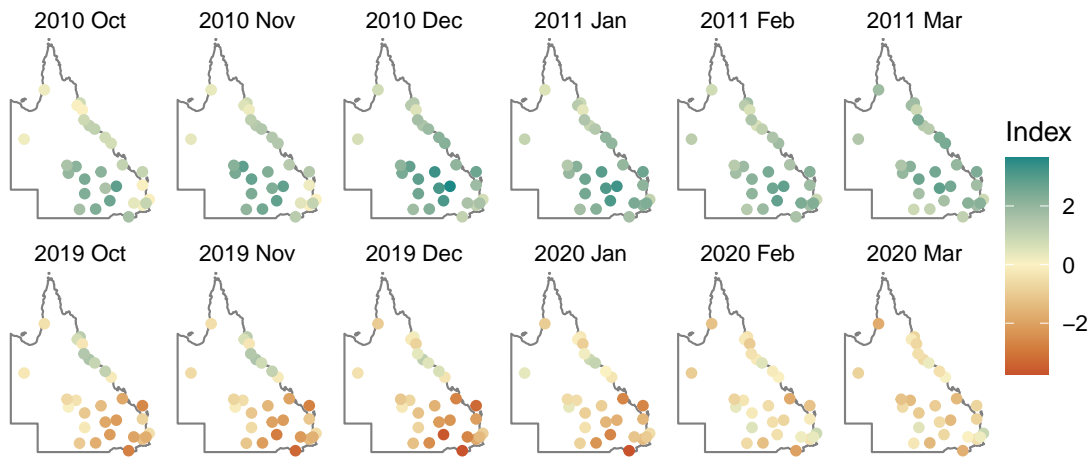

Figure 4. Spatial distribution of Standardized Precipitation Index (SPI-12) in Queensland, Australia during two major flood and drought events: 2010/11 and 2019/20. The map shows a continous wet period during the 2010/11 flood period and a mitigated drought situation, after its worst in 2019 December and 2020 Janurary, likely due to the increased rainfall in February from the meteorological record.
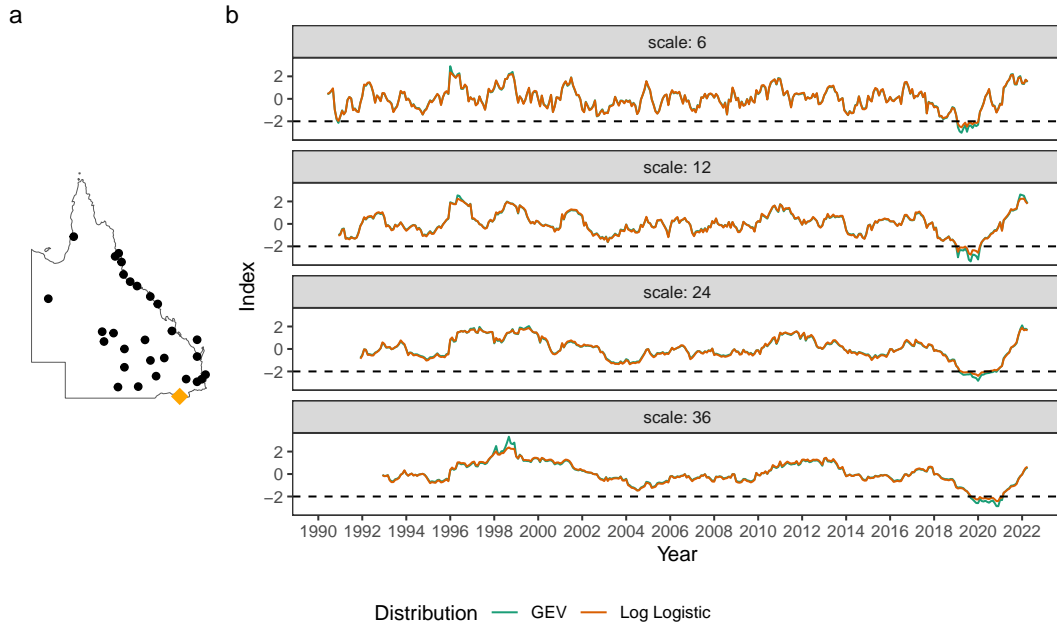
Figure 5. Time series plot of Standardized Precipitation-Evapotranspiration Index (SPEI) at the Texas post office station (highlighted by a diamond shape in panel a). The SPEI is calculated at four time scales (6, 12, 24, and 36 months) and fitted with two distributions (Log Logistic and GEV). The dashed line at -2 represents the class "extreme drought" by the SPEI. A larger time scale gives a smoother index series, while also takes longer to recover from an extreme situation as seen in the 2019/20 drought period. The SPEI values from two distribution fits mostly agree, while GEV can results in more extreme values, i.e. in 1998 and 2020.

5.2. *Does a minor change in variable weights cause a tornado?*

The Global Gender Gap Index (GGGI), published annually by the World Economic Forum, measures gender parity by assessing relative gaps between men and women in four key areas: Economic Participation and Opportunity, Educational Attainment, Health and Survival, and Political Empowerment (World Economic Forum 2023). The index, compiled from 14 variables expressed as female-to-male ratios, first aggregates these variables into four dimensions through linear combinations. These dimensions are then combined though another linear combination. Figure 6 illustrates this pipeline construction by applying the dimension reduction module twice on the data to generate the index. Table 2 presents the The variable weights (`V-weight`) and dimension weights (`D-weight`) used in the two dimension reduction. In the table, the variable weights are computed as the inverse of the standard deviation of each variable, scaled to sum to 1. This weights ensures that a one percentage point change in the standard deviation of each variable to contribute equally to the index. Dimension weights are equal across the four dimensions and the last column, `weight`, multiples the variable and dimension weights to produce a single set of weight. The 2023 GGGI data is available from the Global Gender Gap Report 2023 in the country's economy profile and can be accessed in R via the `tidyindex` package as `gggi` and Table 2 as `gggi_weights`.
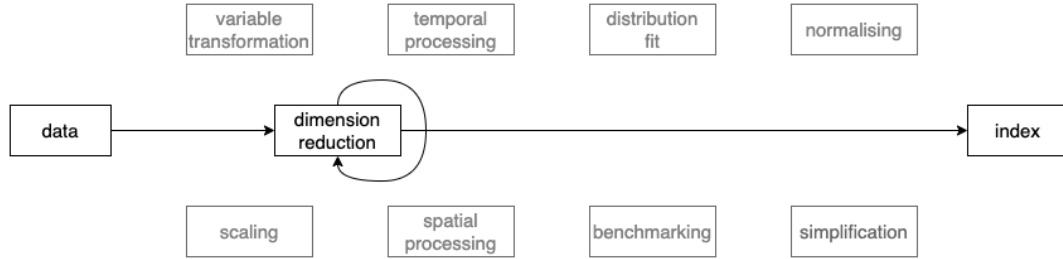

Figure 6. Index pipeline for the Global Gender Gap Index (GGGI). The index is constructed as applying the module dimension reduction twice on the data.

Table 2. Weights used to compute the Global Gender Gap Index

Variable	Dimension	V-weight	D-weight	Weight
Labour force participation	Economy	0.199	0.25	0.050
Wage equality for similar work	Economy	0.310	0.25	0.078
Estimated earned income	Economy	0.221	0.25	0.055
Legislators senior officials and managers	Economy	0.149	0.25	0.037
Professional and technical workers	Economy	0.121	0.25	0.030
Literacy rate	Education	0.191	0.25	0.048
Enrolment in primary education	Education	0.459	0.25	0.115
Enrolment in secondary education	Education	0.230	0.25	0.058
Enrolment in tertiary education	Education	0.121	0.25	0.030
Sex ratio at birth	Health	0.693	0.25	0.173
Healthy life expectancy	Health	0.307	0.25	0.077
Women in parliament	Politics	0.310	0.25	0.078
Women in ministerial positions	Politics	0.247	0.25	0.062
Years with female head of state	Politics	0.443	0.25	0.111

The index can be reproduce with tidyindex as:

```
gggi %>%
  init(id = country) %>%
  add_meta(gggi_weights, var_col = variable) %>%
  dimension_reduction(
    index_new = aggregate_linear(
      ~labour_force_participation:years_with_female_head_of_state,
      weight = weight))
```

After initialising the `gggi` object and attaching the `gggi_weights` as meta-data, a single linear combination step within the dimension reduction module is applied to the 14 variables (from column `labour_force_participation` to `years_with_female_head_of_state`), using the weight specified in the `weight` column of the attached metadata. While computing the index from the original 14 variables, it remains unclear how the missing values are handled within the index, which impacts 68 out of the total 146 countries. However, after aggregating variables into the four dimensions, where no missing values exist, the index is reproducible for all the countries.

A linear combination can also be interpreted as a linear projection of multivariate information with a weight vector. Projecting data from higher to lower dimension unavoidably leads to information loss and the weights used require careful examination to understand their effects on the index and implications for interpretation and decision-making. We use a visualisation method called "tour" to animate the data projection across different sets of weights, aiming to understand how changes on the politics weight will affect the index value. We select a set countries for this exercise, which includes:

1) Top-ranked countries with GGGI > 0.85,
2) Countries ranked between 57 and 62, with GGGI values from 0.72 to 0.73, and
3) Low-ranked countries with GGGI < 0.6.

We run a local tour [TODO: reference to local tour] to slightly vary the weight of the politics dimension from the original 0.25 while keeping the weights constant for the other three dimensions.

In Figure 7, six frames have been chosen from the animation available at https://vimeo.com/847874016. When the weight of politics is reduced (Frame 1 and 6 vs. Frame 12), the difference in GGGI values between the top Nordic countries and mid- or low-ranked countries narrows, suggesting a smaller variation among countries in achieving gender parity. Conversely, when the weight of politics increases (Frame 18, 24, and 29 vs. Frame 12), nearly all the countries in the three categories experience a decrease in GGGI values. A noticeable exception to this trend is Bangladesh, where its index value moves in the same direction as the politics weight. This leads to its index value being almost similar to those of the top-ranked Nordic countries when the politics dimension is assigned a weight of 0.52 in Frame 29.

In GGGI, the index value has a direct interpretation as the percentage of gender gap that has been closed. When countries exhibit closer index values, or consistent decreasing index values, it can be directly interpreted as country's progress towards gender parity. Analysts must carefully consider whether such interpretation is reasonable and review the index construction process.
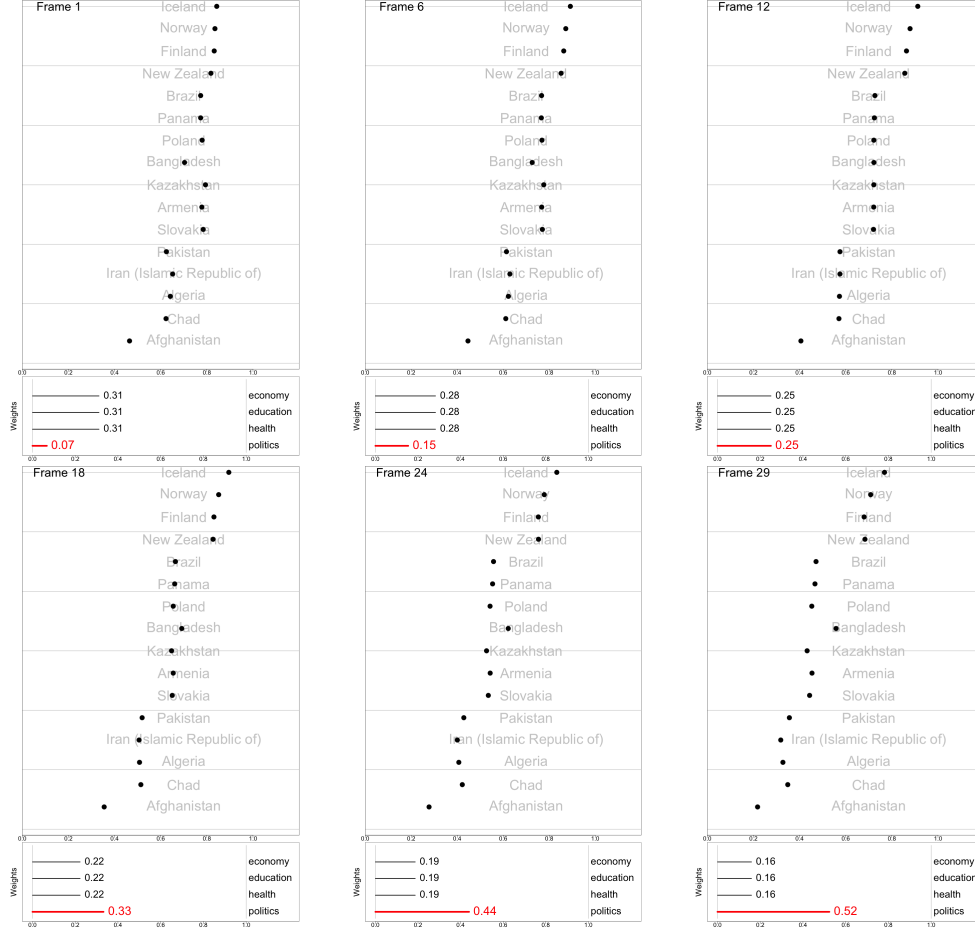
Figure 7. Six frames selected from varying the weights of the politics dimension in Global Gender Gap Index (GGGI). The top panel shows the GGGI value against the country, ranked by its original index value in Frame 12. The bottom panel displays the weight used to produce the index value in the top panel, with each frame corresponds to a set of weights. Countries selected includes 1) top-ranked countries with GGGI > 0.85, 2) countries ranked between 57 and 62 with GGGI from 0.72 to 0.73, and 3) low-ranked countries with GGGI < 0.6. Compared to Frame 12 where equal weigts are used for the four dimension, a reduced weight in politics (Frame 1 and 6) shows narrower gaps between top and mid- or lower- ranked countries, while an increase in the politics weight (Frame 18, 24, and 29) leads to a systematic decrease of GGGI values across all the countries, except for Bangladesh. Full animation is available at https://vimeo.com/847874016.

13

6. Conclusion

The paper presents a data pipeline with nine modules for constructing and analysing indexes. The pipeline increases transparency in the practice for index analysts to experiment with different index design and parameter choices to better design and apply their indexes. The significance of this work is its ability to provide a universal framework for index construction, which can be applied across different domains.

Examples have been given in the drought indexes and human development index to demonstrate computing of indexes with different parameters combinations and how alternative index design can provide insights to understand distinctive country characteristics that could sometimes be overlooked. The accompanied package, tidyindex, is not meant to provide comprehensive implementation for all indexes across all domains. Instead, it demonstrates implementing individual pipeline steps that are versatile to multiple indexes and composing new indexes from existing steps. Domain experts are welcomed to adopt the pipeline approach to develop specialised packages for specific-domains indexes.

Future work: - integrate more complex dimension reduction methods to calculate weights - strengthen the spatial processing module

Reference

Buja, A, D Asimov, C Hurley, and JA McDonald. 1988. "Elements of a Viewing Pipeline for Data Analysis." In *Dynamic Graphics for Statistics*, 277–308. Wadsworth, Belmont.

Kuhn, Max, and Hadley Wickham. 2020. *Tidymodels: A Collection of Packages for Modeling and Machine Learning Using Tidyverse Principles.* https://www.tidymodels.org.

OECD, European Union, and Joint Research Centre - European Commission. 2008. *Handbook on Constructing Composite Indicators: Methodology and User Guide.* OECD. https://doi.org/10.1787/9789264043466-en.

Sutherland, Peter, Anthony Rossini, Thomas Lumley, Nicholas Lewin-Koh, Julie Dickerson, Zach Cox, and Dianne Cook. 2000. "Orca: A Visualization Toolkit for High-Dimensional Data." *Journal of Computational and Graphical Statistics* 9 (3): 509–29. https://www.jstor.org/stable/1390943.

Wickham, Hadley. 2014. "Tidy Data." *Journal of Statistical Software* 59 (September): 1–23. https://doi.org/10.18637/jss.v059.i10.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the Tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Michael Lawrence, Dianne Cook, Andreas Buja, Heike Hofmann, and Deborah F. Swayne. 2009. "The Plumbing of Interactive Graphics." *Computational Statistics* 24 (2): 207–15. https://doi.org/10.1007/s00180-008-0116-x.

World Economic Forum. 2023. "The Global Gender Gap Report 2023." https://www3.weforum.org/docs/WEF_GGGR_2023.pdf.

Xie, Yihui, Heike Hofmann, and Xiaoyue Cheng. 2014. "Reactive Programming for Interactive Graphics." *Statistical Science* 29 (2): 201–13. https://www.jstor.org/stable/43288470?seq=1.