DEMO ARXIV TEMPLATE

A PREPRINT

H. Sherry Zhang
Department of Econometrics and Business Statistics
Monash University
Melbourne, VIC
huize.zhang@monash.edu

Collaborators
Department of Econometrics and Business Statistics
Monash University
Melbourne, VIC

November 28, 2022

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

***K*eywords** indexes • data pipeline • software design

1 Introduction

Why index is useful, why people care about indexes

incorporate the following in why using index: multiple pieces of information (variables) that need to be taken into account

Many concepts relevant to decision making cannot be directly measured, however, they are crucial for resource allocation, early prevention, and other operational purpose. For example, fire authorities would be interested to quantify fire risk since bushfires can have a huge impact on monetary loss, health, and the local ecosystem. Climatologists would be interested in monitoring the change in global climate since variability in atmospheric and oceanic conditions has a direct impact on global weather and climate. Usually this concept of interest is associated with more than one variables and these variables need to be integrated to make decisions on the subject matter. A common approach to quantify concepts like these is to construct an index using these relevant variables. This allows researchers to compare the quantity of interest across entities (i.e. countries, regions) and also cross time.

Define what is an index, what is not

In this article, an index is defined as a tool to quantify a concept of interest that does not have a direct measure. The concept of interest doesn't have a direct measure can because it is impractical to measure at the population level. For example, it would be nearly impossible to include all the available stocks in the market to characterise stock market behavior, so indexes like Dow Jones Industrial Average, S&P 500, and Nasdaq Composite select a representative set of stocks to measure the overall market behavior. Also belonging to this category are the economic indexes like the Consumer Price Index, where price changes of a basket of items are weighted to measure inflation. The lack of direct measure could also because the concept itself is an unobservable human construction, rather than a physical quantity that can be measured. Many natural hazard and social concepts falls into this category. This includes drought indexes constructed from meteorological, agricultural, hydrological, and social-economic variables, e.g. Standardised Precipitation Index (SPI) (McKee et al. 1993) and Aggregated Drought Index (ADI) (Keyantash and Dracup 2004) among others. Social development indexes like Human Development Index (United Nations Development Programme 2022) and Global Liveability Index (Economist Intelligence Unit 2019) measure various aspects of the quality of human capital and urban life.

still need to tweak the tone a bit: "they are called index, they are not the index we will talk about"

Despite many quantity having the term *index* in their name, they cannot be technically classified as indexes according to the definition given above. The reason for these quantities to lose their index memberships is that they are variables can be accurately measured given the instrument precision. This includes quantities like precipitation of the driest month or percentage of days when maximum temperature is below 10th percentile. They are measures of precipitation and percentage of days under specific conditions (dries month, maximum temperature below 10th percentile). They are variables, or indicators, that can be used to construct indexes but are not indexes themselves. Similarly, a set of remote sensing indexes are not indexes, since they are measures of electromagnetic wave reflectance. This includes Normalized Difference Vegetation Index (NDVI) (Tucker 1979), derived from the ratio of difference over sum on two segments in the spectrum, also called band: near-infrared (NIR) and red. So are the "indexes" derived from NDVI, e.g. Vegetation Condition Index (Kogan 1995). Notice that this does not exclude all the construction derived from remotes sensor variables to be valid indexes. For example, Vegetation Drought Response Index (Brown et al. 2008) is a valid index since it integrates climate, satellite, and biophysical variables to quantify vegetation stress.

What is the challenges with current index construction

see if there is any paper describing this type of pains

useful to reference tidy data and tidy model that makes the workflow on modelling tidy somewhere in introduction

Currently, index construction lacks a standardised workflow. It is often up to researchers or research institutions to decide whether to provide open source code on the new indexes, what would be the best user interface for other researchers to use the new indexes, and how easily the new indexes can be compared with other existing indexes. This makes the computation lack transparency and indexes cumbersome to experiment with:

- Researchers who wish to validate the indexes calculated from large institutes need to reinvent the wheels themselves since the source code used for computing is often not available for public consumption;
- Open-source code provided by research groups has a narrow margin for exploring other options outside the provided;
- Similar steps used by different indexes are difficult to spot since the design of the user interface for indexes often includes all the steps under a single function call; and
- It is generally hard to inspect intermediate results during the index construction if users wish to check the output of a certain step.

what can be done if people adopt this pipeline/ why it is beneficial?

This paper proposes a data pipeline for index construction. By recognising the common steps shared by many indexes, we develop a pipeline that breaks down index construction into multiple modules and allow operations in various modules to be combined like building blocks to construct indexes. The pipeline approach is general while adaptable to most index construction. It allows indexes to be created, studied, and compared in a structured tidy form and enables statistical analysis of indexes to be performed easily: More specifically, it enables researchers to 1) validate the indexes calculated from external organisations, 2) unify various indexes under the same framework for computing, 3) swap or adjust individual steps in the index construction to study their contribution, 4) calculate uncertainty on indexes through bootstrap or others, 5) enhance existing indexes through comparing and studying their statistical properties, and finally, 6) propose new indexes from combining different steps in existing indexes.

who would benefit from this paper

This work is of interest to researchers actively developing new indexes since it encourages new indexes to be delivered in an easy-to-reproduce design. It would also provide analysts who wish to compute a range of indexes in their analysis a uniform interface to build relevant indexes from raw data. For statisticians and software developing engineers, this work frames the process of index construction in a more user-oriented workflow and could motivate similar research for other process in scientific computing.

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 include a new building block in each pipeline module. Examples are given in Section 5 to demonstrate the index construction with the pipeline built.

2 Data pipeline

Think about if there is another word for data pipeline

Why you should care about pipeline

Data pipeline is not a new concept to computing. It refers to a set of data processing elements connected in series, where the output of one element is the input of the next one. Wickham et al. (2009) argues that whether made explicit or not, the pipeline has to be presented in every graphics program. The paper also argues that breaking down graphic rendering into steps is beneficial for understanding the implementation and comparing between different graphic systems. The discussion on pipeline construction is well documented in early interactive graphics software: Buja et al. (1988), Sutherland et al. (2000), and Xie, Hofmann, and Cheng (2014) and their pipeline steps include non-linear transformation, variable standardization, randomization and dimension reduction.

What is pipeline, its underlying software design philosophy, and how these are reflected in R

One of the most commonly known pipeline examples is perhaps the Unix pipeline where programs can be concatenated with `|` to flow the output from the last program into the next program, i.e.

```
command 1 | command 2 | command 3 | ...
```

To solve a complex problem, the Unix system builds simple programs that do one thing well and work well together. This design is also reflected in the tidyverse ecosystem in R. To solve a complicated data problem using tidyverse, analysts typically build the solution using a collection of tools from the tidyverse toolbox. The data object can flow smoothly from one command to the next, safeguarded by the tidy data format (Wickham 2014), which prescribes three rules on how to lay out tabular data. The tidyverse tools also embrace a strong human-centered design where function names are intuitive and easy to reference through autocomplete. With the tidyverse design principle in mind, the tidymodel suite enables analysts to build machine learning models through the data pipeline. It includes typical tasks required in machine learning like data resampling, feature engineering, model fitting, model tuning, and model evaluation. An advantage of tidymodel pipeline over separate software for individual models is that analysts no longer need to write model-specific syntax to work with each model, but pipeline-specific syntax that is applicable to all the models implemented in tidymodel. This allows users to easily experiment with a collection of machine learning models.

Constructing indexes would also benefit from pipeline and embracing the aforementioned design philosophy.

In index construction, data pipeline is often presented in a workflow diagram in the research paper to illustrate how the raw data is transformed into the final indexes. This agrees with Wickham's argument on the presence of the data pipeline, however, more often than not, the pipeline is not made explicit in the software. Often the time, all the steps are lumped into a single wrapper function, rather than being split into smaller, modulated functions. This increases the cost of maintaining and understanding the code base, gives analysts little freedom to customise the indexes for specific needs, and hinders reusing existing code for building new indexes. A pipeline approach unites a range of indexes under a single data pipeline and analysts can compose indexes from pipeline steps like building Legos from individual bricks. In this workflow, analysts are not limited by indexes that have been already proposed and can easily combine pipeline steps to compose novel indexes. Analysis of the indexes (i.e. calculation of uncertainty) is also feasible by adding external code into the pipeline.

3 A pipeline for building statistical indexes

3.1 How does the pipeline constructin of an index look like?

Consider a commonly used drought index: Standardized Precipitation-Evapotranspiration Index (SPEI) (Vicente-Serrano, Beguería, and López-Moreno 2010). Its construction involves: 1) calculating potential evapotranspiration (PET) from average temperature, `tavg`, and its difference, `d`, with precipitation, `prcp`, 2) aggregating difference series with a time scale, 3) fitting the aggregated series with a chosen distribution to get density values, and 4) converting the density values to normal quantiles. Under the pipeline approach, SPEI will be constructed as:

```
DATA %>%
  trans_pet(method = thornthwaite, var = tavg) %>%
  aggregate(scale = 12, var = d) %>%
  normalise(dist = gamma, fit_method = lmom) %>%
  augment(var = .agg_prcp)
```

Change the examples to two scales values, compare how indexes look like

Here each command corresponds to one step described above with additional arguments specific to the step. The pipeline construction produces the same index value as command like `spei(...)`, as shown in Figure 1, but with additional benefit:

- Multiple scales and multiple distributions can be fitted using the `c(...)` syntax to compare index values constructed from different parameterisations;
- Intermediate results can be checked after each step; and
- Additional steps and analysis can be wired into the workflow for index diagnostics and customised user need.

everyone is providing single command function, this is also the same for spi

may not worthwhile if only one index, as long as start changing

just want to calculate index, single function is fine

Figure 1: Standardised Precipitation Evapotranspiration Index (SPEI) calculated with two distribution fits in Step 3 described above: generalised logistic (`glogist`) and generalised extreme value (`gev`) distribution, using the `wichita` data from the package `SPEI`.

3.2 Pipeline steps for constructing indies

An overview of the pipeline is given in Figure 2 to illustrate the construction from raw data to the final indexes. The pipeline includes modules for operations in the spatial, temporal, and multivariate aspect of the data: temporal processing, spatial aggregation, variable transformation, scaling, dimension reduction, and normalising, as well as modules for comparing and communicating indexes (benchmarking and simplification). These steps are not required to be arranged in a fixed order and not all the steps have to appear when constructing each index. While the pipeline starts with the raw data, steps prior to this, i.e. define a useful concept to construct the index, collect high quality data relevant for measuring the concept, are also cricial to the sucess of an index. The framework proposed in this paper assumes the objective of an index has been defined and relevant variables have been collected and quality controlled.

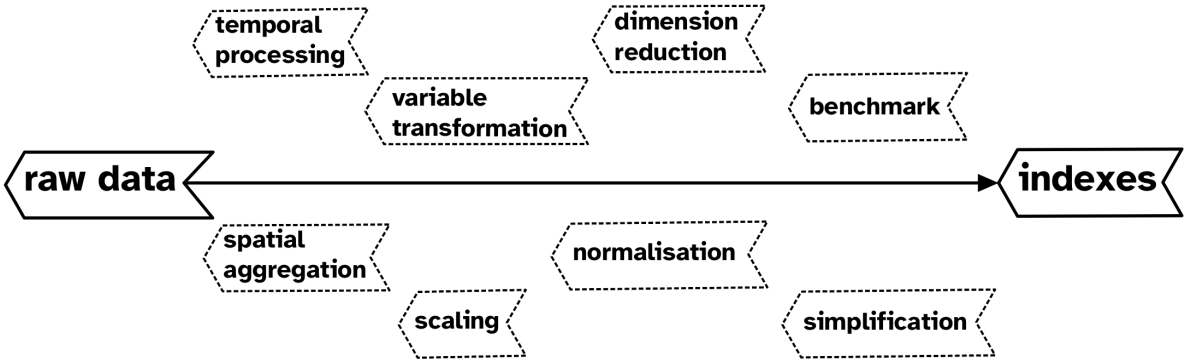

Figure 2: Diagram of pipeline steps for index construction.

Notation

Let $\mathbf{x}(\mathbf{s};\mathbf{t})$ denote the raw data with spatial, temporal, and multivariate aspect: the spatial dimension $\mathbf{s} = (s_1, s_2, \cdots, s_n)'$ is defined in the 2D space: $\mathbf{s} \in \mathcal{D}_s \subseteq \mathbb{R}^2$, the temporal dimension $\mathbf{t} = (t_1, t_2, \cdots, t_J)'$ is defined in the 1D space: $\mathbf{t} \in \mathcal{D}_t \subseteq \mathbb{R}$. When more than one variable are involved, the multivariate data can also be written as: $\mathbf{x}(\mathbf{s};\mathbf{t}) = (x_1(\mathbf{s};\mathbf{t}), x_2(\mathbf{s};\mathbf{t}), \cdots, x_P(\mathbf{s};\mathbf{t}))'$.

Temporal processing

The construction of an index sometimes needs to consider information from neighbouring time periods. Temporal processing is a general operator on the time dimension of the data in the form of

$$f_\psi(x(\mathbf{s};\mathbf{t})),$$

where $\psi \in \Psi \subseteq \mathbb{R}^{d_\psi}$ is the parameters associated with the temporal operation and d_ψ is the number of parameter of ψ. An typical example of temporal processing is aggregation in the drought index SPI to measure the lack of precipitation for meteorological drought. In SPI, monthly precipitation is aggregated by a time scale parameter k: $x(s_i; t_{j'}) = \sum_{j=j'-k+1}^{j'} x(s_i; t_j)$, where j' is the new time index after the aggregation. In this notation, each spatial location is seaprately aggregated and precipitation is ummed from k month back, $j' - k + 1$, to the current, j' period to create the aggregated series, indexed by j'.

The choice of time scales parameter k can result in variation in the calculated index values: a small k of 3 or 6 month produces the index more sensitive to individual months, while a large k of 24 or 36, an equivalent to a 2- or 3-year aggregation, gives dryness information relative to the long term condition. As will be shown in section [SECTION EXAMPLE], this variation may even lead to conflicting conclusions on the dry/wet condition of the area, highlighting the importance to account for index uncertainty when interpreting index values for decision making.

Spatial aggregation

Sometimes, information from neighbouring spatial locations can also be borrowed when constructing indexes. This involves interpolating data from observed locations to the unobserved and aggregating data into different levels. In the pipeline, spatial aggregation can be written as

$$g_\theta(x(\mathbf{s};\mathbf{t})),$$

where $\theta \in \Theta \subseteq \mathbb{R}^{d_\theta}$ is the spatial operation parameters and d_θ is the number of parameter of θ. [GIVEN AN EXAMPLE]

collect ground and satellite image -> merge

Variable transformation

Variable transformation and scaling, in the next section, are both variable pre-processing steps to prepare the data with desire statistical property to work with. This can be a more stablised variance, a normal distribution that satisfies modelling assumptions, or a data range that required by the algorithms. Variable tranformation is a general notion of a functional transformation on the variable:

$$h_\tau(x(\mathbf{s};\mathbf{t})),$$

where $\tau \in T \subseteq \mathbb{R}^{d_\tau}$ is the parameter in the transformation, if any, and d_τ is the number of parameter of τ. Typical examples of variable transformation are log, quadratic, square root, and box-cox transformation.

Scaling

While scaling can be seen as a specific type of variable tranformation, it is separated into own step to make the scaling operation explicit in the pipeline. The key difference between the two steps is that variable transformation typically changes the shape of the data while scaling only changes the data scale and can usually be written in the form of

$$[x(s_i;t_j) - \alpha]/\gamma.$$

For example, a z-score standardisation can be written in the above form with $\alpha = \bar{x}(s;t)$ and $\gamma = \sigma(s;t)$, a min-max standardisation uses $\alpha = \min[x(s_i,t_j)]$ and $\gamma = \max[x(s_i,t_j)] - \min[x(s_i,t_j)]$. Figure 3 shows a collection of variable pre-processing operations and uses color to differentiate whether the operation is a variable transformation or a scaling step. All the scaling operations in green results in the same shape as the original data.

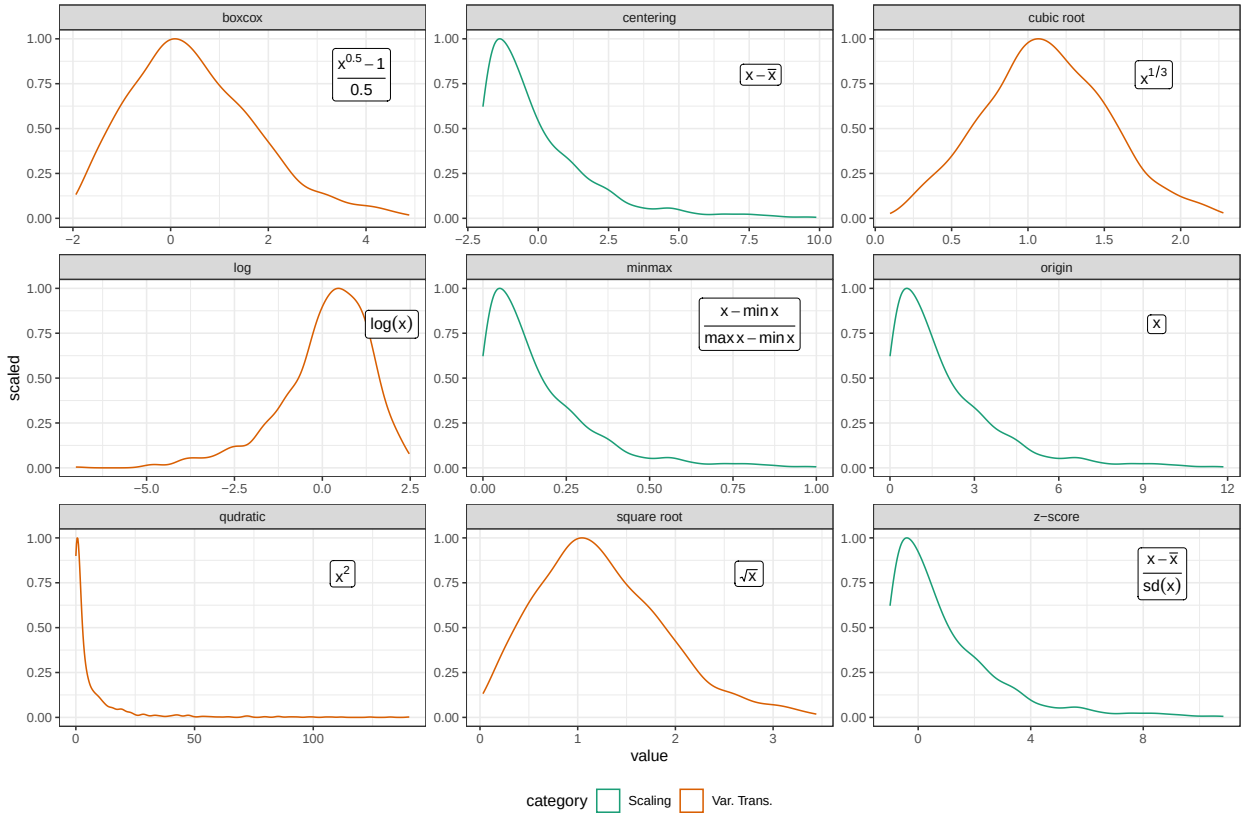

Figure 3: Comparison of scaling and variable transformation.

Dimension reduction

While multivariate information can be collected on a concept, they need to be summarised into a single number for the final index. Dimension reduction step

$$x_{p^*}(\mathbf{s};\mathbf{t}) \rightarrow x_p(\mathbf{s};\mathbf{t}),$$

where $p^* = 1, 2, \cdots, P^*$ and $p = 1, 2, \cdots, P$, reduces the variable dimension from P to P^*. The most commonly used dimension reduction technique Principal Component Analysis (PCA), also called Empirical Orthogonal Function (EOF) in earth science, can be written as $x_{p^*}(\mathbf{s};\mathbf{t}) = \sum_{p=1}^{P} \lambda_p x_p(\mathbf{s};\mathbf{t})$ where λ_p is the loading of the PC1, derived from maximising the data variance given the constraint $\sum_{p=1}^{P} \lambda_p^2 = 1$.

weighting

Distribution fit

modell fit?

Distribution fit can be seen as model fitting in its simplest term. It can be represented by

$$F_\eta(x(\mathbf{s};\mathbf{t})),$$

where $\eta \in H \subseteq \mathbb{R}^{d_\eta}$ is the distribution parameter and d_η is the number of parameter of η. A distribution fit typically aims at finding the distribution that best fit the data. Analysts may start from a pool of candidate distributions with a chosen fitting method and goodness of fit measure. While it is useful to find the ultimate best distribution to fits the data, from a probabilistic perspective, the fitting procedure itself has uncertainty associated with the data fed and parameter chosen. A reasonable alternative is to understand how much the index values can vary given different distributions, fitting methods, and goodness of fit tests, and whether these variations are negligible in a given application.

Normalising

This step maps the univariate series into a different scale, typically for the ease of comparison across regions. For example, a normal scale, [0, 1], or [0, 100] may be favored for reporting certain indexes. In drought indexes, i.e. SPI or SPEI, the quantiles from the fitted distribution is converted into the normal scale via the normal reverse CDF function: $\Phi^{-1}(.)$.

Normalising is usually used in the end of the pipeline and its main difference from the scaling step is that here the change of scale also changes the distribution of the variable. While being commonly used, this step can get criticism from analysts for forcing the data into the decided scale, which can be either unnecessary or inaccurately exaggerate or downplay the outliers. Also, the use of normal scale needs to be interpreted with caution. Figure 4 illustrates the normal density not being directly proportional to its probability of occurrence. This is concerning, especially at the extreme values, since a small difference in the tail density can have magnitudes of difference in its probability of occurrence.

Benchmarking

Benchmarking sets a constant value to allow the constructed index to be compared across time. Here we denote it with $u[x(s_i, t_j)]$ where u is a scalar of interest in the index constructed, could be a constant or a function of the data, i.e. mean.

Simplification

In public communication, indexes are often delivered in categorical grades, along with its underlying numerical values. The simplification step, sometimes can also be called discretisation, prescribes how the continuous index values are converted into the discrete grades. This process can be written with the piece-wise function:

$$\begin{cases} C_0 & c_1 \leq (s_i;t_j) < c_0 \\ C_1 & c_2 \leq x(s_i;t_j) < c_1 \\ C_2 & c_3 \leq x(s_i;t_j) < c_2 \\ \cdots & \\ C_z & c_z \leq x(s_i;t_j) \end{cases}$$

where $C_0, C_1, \cdots, C_z$ are the categories and $c_0, c_1, \cdots, c_z$ are the threshold value in each category. In SPI, drought are sorted into four categories: mild drought: $[-0.99, 0]$; moderate drought: $[-1.49, -1]$; severe drought: $[-1.99, -1.5]$, and extreme drought: $[-\infty, -2]$. In this case, C_0, C_1, C_2, C_3 are the drought categories: mild, moderate, severe, and extreme drought ($z = 3$) and $c_0 = 0, c_1 = -1, c_2 = -1.5, c_3 = -2$ are the cutoff value for each class.

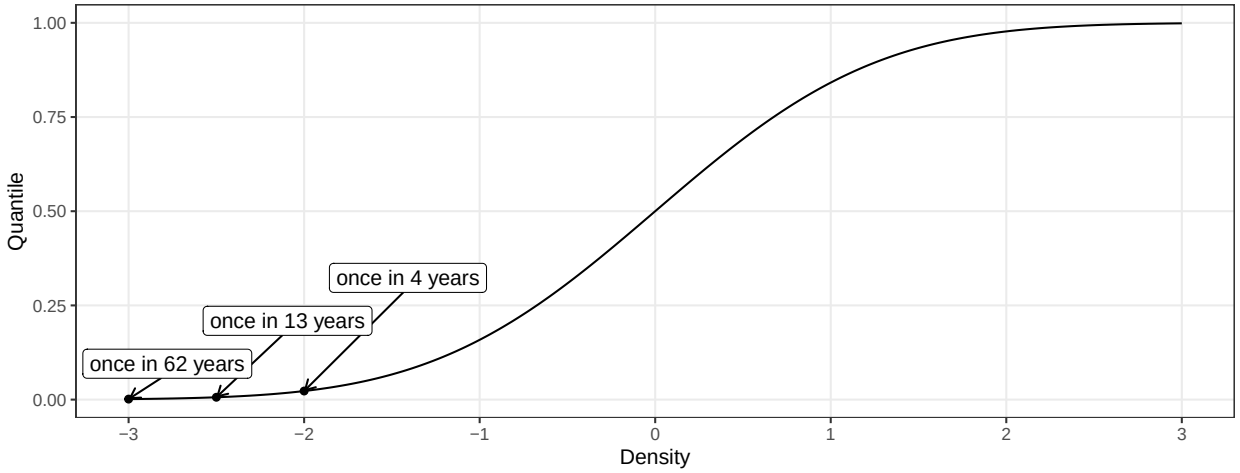

Figure 4: Scatterplot of normal quantiles against their density values. THree tail density values are highlighted with its probability of occurence labelled. Probability is calculated assuming monthly data: with a density of -2, the probability of occurrence is 1/pnorm(-2)/12 = 4 years. The non-linear relationship between the two quantities suggests normalised indexes need to be interpreted with caution since a slight change in the tail distribution can result in magnitudes of difference in its probability of occurrence.

Discretise the continuous index into a few labelled categories. For communicating the severity of natural hazard to general public.

uniform workflow to work with index construction.

- illustration
- math notation
- benefit of the pipeline approach
  - index diagnostic
  - uncertainty

4 Incorporating new buliding blocks into the pipeline

5 Examples

5.1 Constructing Standardised Precipitation Index (SPI)

- a basic workflow and congruence with results in the `SPEI` pkg
- allow multiple distribution fit
- allow bootstrap uncertainty

5.2 Calculating SPEI with raster data

Reference

Brown, Jesslyn F., Brian D. Wardlow, Tsegaye Tadesse, Michael J. Hayes, and Bradley C. Reed. 2008. "The Vegetation Drought Response Index (VegDRI): A New Integrated Approach for Monitoring Drought Stress in Vegetation." *GIScience & Remote Sensing* 45 (1): 16–46. https://doi.org/10.2747/1548-1603.45.1.16.

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.

Economist Intelligence Unit. 2019. "The Global Liveability Index 2019." The Economist. https://www.cbeinternational.ca/pdf/Liveability-Free-report-2019.pdf.

Keyantash, John A., and John A. Dracup. 2004. "An Aggregate Drought Index: Assessing Drought Severity Based on Fluctuations in the Hydrologic Cycle and Surface Water Storage." *Water Resources Research* 40 (9). https://doi.org/10.1029/2003WR002610.

Kogan, F. N. 1995. "Application of Vegetation Index and Brightness Temperature for Drought Detection." *Advances in Space Research*, Natural Hazards: Monitoring and Assessment Using Remote Sensing Technique, 15 (11): 91–100. https://doi.org/10.1016/0273-1177(95)00079-T.

McKee, Thomas B, Nolan J Doesken, John Kleist, et al. 1993. "The Relationship of Drought Frequency and Duration to Time Scales." In *Proceedings of the 8th Conference on Applied Climatology*, 17:179–83. 22. Boston, MA, USA.

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.

Tucker, Compton J. 1979. "Red and Photographic Infrared Linear Combinations for Monitoring Vegetation." *Remote Sensing of Environment* 8 (2): 127–50. https://doi.org/10.1016/0034-4257(79)90013-0.

United Nations Development Programme. 2022. "Human Development Report 2021-22." New York. http://report.hdr.undp.org.

Vicente-Serrano, Sergio M., Santiago Beguería, and Juan I. López-Moreno. 2010. "A Multiscalar Drought Index Sensitive to Global Warming: The Standardized Precipitation Evapotranspiration Index." *Journal of Climate* 23 (7): 1696–1718. https://journals.ametsoc.org/view/journals/clim/23/7/2009jcli2909.1.xml.

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

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.

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.