Generalized Price Indexes

Steve Martin

December 2, 2020

Generalized price indexes are a large family of price indexes with nice properties. Almost all bilateral price indexes used in practice are either generalized indexes (like the Laspeyres and Paasche index) or are nested generalized indexes (like the Fisher index). As it is trivial to turn a price index into its analogous quantity index by switching prices and quantities in the index-number formula, everything about generalized price indexes applies equally to generalizes quantity indexes.

Generalized indexes are natural candidates for making national statistics, as these indexes are consistent in aggregation. This justifies the hierarchical structure used by national statistical agencies for calculating and disseminating collections of price indexes. Generalized indexes also have other desirable properties: A generalized index is always positive, lies between the largest and smallest price change (mean value property), and only shows a change in price if prices change (identity property). Except in pathological cases, a generalized index is also a continuous function of prices and quantities, a proportional change in prices leads to a proportional change in the index (linear homogeneity), and the index is not affected by the currency in which prices are measured (homogeneity of degree zero)—see Balk (2008, Chapter 3) and ILO et al. (2004, Chapter 16) for more details. Many generalized indexes have additional nice properties not shared by the entire family.

One especially useful, although often overlooked, property of generalized indexes is that they all share the same mathematical structure, and can be manipulated in the same way. The `gpindex` package provides the computational tools to calculate and manipulate any generalized index using R. The purpose of this vignette is to show how this can be done mathematically, as most of the results in the literature apply to a narrow subset of generalized indexes, and how this can be done in R using the `gpindex` package.

Generalized means

All generalized indexes are based on the generalized mean. Given a set of price relatives and weights, any generalized price index is simply a generalized mean of price relatives. What distinguishes different generalized price indexes are the weights and the order of the generalized mean. See Bullen (2003, Chapter 3) for an extensive discussion of generalized means and their properties.

Given a collection of $i = 1, \ldots, n$ goods transacted at two points in time, period 0 (the base or reference period) and period 1 (the current or comparison period), let $p_t = (p_{1t}, p_{2t}, \ldots, p_{nt})$ be the vector of prices in period $t = 0, 1$, and let $w = (w_1, w_2, \ldots, w_n)$ be a vector of weights associated with each good. Suppose that each $p_i > 0$, $w_i > 0$, and $\sum_{i=1}^{n} w_i = 1$. The generalized price index of order r is simply the generalized

mean[1]

$$I_r\left(\frac{p_1}{p_0}, w\right) = \begin{cases} \left(\sum_{i=1}^n w_i \left(\frac{p_{i1}}{p_{i0}}\right)^r\right)^{1/r} & \text{if } r \neq 0 \\ \prod_{i=1}^n \left(\frac{p_{i1}}{p_{i0}}\right)^{w_i} & \text{if } r = 0. \end{cases}$$

A generalized index can equivalently be seen as the solution to the following prediction problem:

$$\min_I \begin{cases} \sum_{i=1}^n w_i \left(I^r - \left(\frac{p_{i1}}{p_{i0}}\right)^r\right)^2 & \text{if } r \neq 0 \\ \sum_{i=1}^n w_i \left(\log(I) - \log\left(\frac{p_{i1}}{p_{i0}}\right)\right)^2 & \text{if } r = 0. \end{cases}$$

All generalized indexes attempt to best predict a change in price for a good under quadratic loss. This emphasizes that generalized indexes are not just mathematically similar; they are conceptually similar as well.

Most price indexes used in practice correspond to a generalized index of order 1 (for arithmetic indexes), order 0 (for geometric indexes), or order -1 (for harmonic indexes)—i.e., most price indexes are based on one of the Pythagorean means. For example, setting $r = 1$ and $w_i = p_{i0}q_{i0}/\sum_{i=1}^n p_{i0}q_{i0}$ results in the Laspeyeres index, and setting $r = -1$ and $w_i = p_{i1}q_{i1}/\sum_{i=1}^n p_{i1}q_{i1}$ results in the Paasche index. More exotic price indexes are also special cases of the generalized index; for example, setting $r = 1 - \sigma$, where σ is the elasticity of substitution, and $w_i = p_{i0}q_{i0}/\sum_{i=1}^n p_{i0}q_{i0}$ results in the Lloyd-Moulton price index. The class of generalized price indexes is convenient because a large collection of seemingly different index-number formulas can be indexed by a single number.

Generalized means, and therefore generalized indexes, are calculated with the `mean_generalized()` function in the `gpindex` package. This function takes a single number `r` and returns a function to compute the corresponding generalized mean. For example, `mean_generalized(1)()` is the function to compute the usual arithmetic mean, and is mostly identical to the `weighted.mean()` function in the `stats` package, whereas `mean_generalized(0)()` is the geometric mean, and `mean_generalized(2)()` is the quadratic mean. Given their importance, the arithmetic, geometric, and harmonic means all have their own functions that simply call `mean_generalized()` with `r` taking the values 1, 0, and -1. Although there are many implementations of *unweighted* geometric and harmonic means for R, in most cases a *weighted* mean is needed for calculating a price index.

```
> library(gpindex)
> p1 <- price6[[2]]
> p0 <- price6[[1]]
> mean_arithmetic(p1 / p0) # Carli index

[1] 1.4

> mean_geometric(p1 / p0) # Jevons index

[1] 1.24192

> mean_harmonic(p1 / p0, 1:6) # Harmonic index with random weights

[1] 1.019744
```

---

[1]This is more general than the definition in Balk (2008, Chapter 3), where the weights are taken as base-period expenditure shares.

Named price indexes

A hallmark of the literature on price indexes is that most price indexes are associated with a name. Although naming is sometimes inconsistent, each name can be roughly mapped to a specific set of weights in the generalized mean of a specific order. For example, a Laspeyres index has weights given by base-period expenditure shares in a generalized mean of order 1, whereas a Paasche index has weights given by current-period expenditure shares in the a generalized mean of order -1. It is important to note that the mapping between names and weights is not necessarily one-to-one, as indexes can have the same weights but receive different names by virtue of having different orders. For example, both the Paasche index and the Palgrave index have the same weights; the difference is that the Paasche index is of order -1, whereas the Palgrave index is of order 1.[2]

The `index_weights()` function in `gpindex` takes the name of a price index and returns a function to compute the corresponding weights. Each of these functions depend on some combination of prices and quantities for the goods for which an index is being calculated; consequently, these functions are useful for "textbook" problems, but unhelpful for more complicated problems when weights are sourced separately from price data. For example, the `index_weights("Laspeyres")()` function takes information on base-period prices and quantities to calculate Laspeyres weights (i.e., base-period expenditure shares). Plugging these weights into an arithmetic mean of price relatives gives the regular Laspeyres index, whereas plugging these weights into a geometric mean produces a geometric Laspeyres index.

```
> q0 <- quantity6[[1]]
> index_weights("Laspeyres")(p0, q0) # period-0 expenditure shares

[1] 1.0 1.0 2.0 1.0 4.5 0.5

> mean_arithmetic(p1 / p0, index_weights("Laspeyres")(p0, q0)) # Laspeyres index

[1] 1.42

> sum(p1 * q0) / sum(p0 * q0) # Same as the manual calculation

[1] 1.42

> mean_geometric(p1 / p0, index_weights("Laspeyres")(p0, q0)) # Geometric Laspeyres index

[1] 1.329967
```

Note that the weights returned by `index_weights()` need not sum to 1, like all functions in `gpindex` that operate on weights, as this isn't always appropriate for some indexes (namely the Vartia-I index). The `weights_scale()` function scales a numeric vector so that it sums to 1, so unscaled weights in general pose no special challenges, and `mean_generalized()` scales the weights by default.

Naming for the weights follows Balk (2008) and ILO et al. (2004), although it is unlikely to be exhaustive given the great many index-number formulas that exist. Having `index_weights()` be a function factory, however, makes it easy to extend without changing the interface for how other weights are calculated.

Although most named generalized indexes can be calculated by combining the appropriate names in `index_weights()` with the appropriate order in `mean_generalized()`, there are a number of wrappers to

---

[2]This is made more confusing because the geometric index with current-period expenditure shares as weights is a geometric Paasche index, whereas the arithmetic Paasche index has different (hybrid) weights.

simplify doing this. For example, the `index_arithmetic()` function takes the name of an arithmetic index and returns a function to calculate it, essentially passing the appropriate function to calculate the weights to `mean_arithmetic()`.

```
> index_arithmetic("Laspeyres")(p1, p0, q0)
```

```
[1] 1.42
```

```
> index_geometric("Laspeyres")(p1, p0, q0)
```

```
[1] 1.329967
```

In the interest of completeness, there are also a number of wrappers for non-generalized indexes, like the Fisher index, in `gpindex`. The documentation page for `?price_index` has more details.

```
> q1 <- quantity6[[2]]
> index_fisher(p1, p0, q1, q0)
```

```
[1] 1.40105
```

```
> index_hlp(p1, p0, q1, q0) # Harmonic analog of the Fisher index
```

```
[1] 1.400924
```

Quote contributions

It is often useful to be able to decompose a price index into an additive contribution for each price relative. This is made possible by finding appropriate weights to transmute a generalized index into an arithmetic index. That is, for any number r, finding a collection of functions $\tilde{w}_{ir}$, $i = 1, \ldots, n$, of p_1/p_0 and w, such that $\sum_{i=1}^{n} \tilde{w}_{ir}(p_1/p_0, w) \equiv 1$ and

$$I_r\left(\frac{p_1}{p_0}, w\right) \equiv \sum_{i=1}^{n} \tilde{w}_{ir}\left(\frac{p_1}{p_0}, w\right) \frac{p_{i1}}{p_{i0}}.$$

If such functions exists, they must be unique.[3] With this decomposition, the contribution for a price relative is simply $\tilde{w}_{ir}(p_1/p_0, w)(p_{i1}/p_{i0} - 1)$.[4]

Balk (2008, Chapter 4) shows how this can be done for a geometric price index using logarithmic means. His approach can be generalized using the generalized logarithmic mean (Bullen, 2003, p. 385), which is defined for any $a > 0$ and $b > 0$ as

$$\mathcal{L}_r(a, b) = \begin{cases} \left(\frac{a^r - b^r}{r(a-b)}\right)^{1/(r-1)} & r \notin \{0, 1\}, a \neq b \\ \frac{a-b}{\log(a) - \log(b)} & r = 0, a \neq b \\ \frac{1}{\exp(1)}\left(\frac{a^a}{b^b}\right)^{1/(a-b)} & r = 1, a \neq b \\ a & a = b. \end{cases}$$

---

[3] Consider the case of $n = 2$, and suppose $p_{11}/p_{10} \neq p_{21}/p_{20}$. If there are positive numbers w_1 and w_2 such that $w_1 p_{11}/p_{10} + (1 - w_1)p_{21}/p_{20} = w_2 p_{11}/p_{10} + (1 - w_2)p_{21}/p_{20}$, then this implies that $w_1 = w_2$. Any function $\tilde{w}_{ir}(p_1/p_0, w)$ must then be unique for all $p_{11}/p_{10} \neq p_{21}/p_{20}$.

[4] As any arithmetic index can be represented as a basket-style index, $\tilde{w}_{ir}(p_1/p_0, w)/p_{i0}$ gives a collection of implied "quantities" such that the basket-style index using these quantities agrees with the generalized index.

The proof is simple, and relies on the identity

$$0 \equiv \sum_{i=1}^{n} w_i \mathcal{L}_r \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-1} \left[\frac{p_{i1}}{p_{i0}} - I_r \left(\frac{p_1}{p_0}, w \right) \right],$$

from which it follows that

$$\tilde{w}_{ir} \left(\frac{p_1}{p_0}, w \right) = w_i \mathcal{L}_r \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-1} \Big/ \sum_{i=1}^{n} w_i \mathcal{L}_r \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-1}.$$

Balk also shows that any arithmetic price index can be written as a geometric price index with a similar change in weights, which is useful for calculating multiplicative contributions. This result can again be generalized as finding functions $\tilde{w}_{is}$ such that

$$\sum_{i=1}^{n} w_i \frac{p_{i1}}{p_{i0}} \equiv I_s \left(\frac{p_1}{p_0}, \tilde{w}_s \left(\frac{p_1}{p_0}, w \right) \right)$$

using the generalized logarithmic mean.

As any price index of order r can be expressed as an arithmetic index, and any arithmetic index can be expressed as an index for order s, these results can be combined to turn any index of order r into an index of order s with a suitable change of weights. Although these weights can be expressed using the generalized logarithmic mean, it is easier to express them with the more general extended mean (Bullen, 2003, p. 393)

$$\mathcal{E}_{rs}(a, b) = \begin{cases} \left(\frac{s(a^r - b^r)}{r(a^s - b^s)} \right)^{1/(r-s)} & r \neq s, r \neq 0, s \neq 0, a \neq b \\ \left(\frac{a^r - b^r}{r(\log(a) - \log(b))} \right)^{1/r} & r \neq 0, s = 0, a \neq b \\ \left(\frac{a^s - b^s}{s(\log(a) - \log(b))} \right)^{1/s} & r = 0, s \neq 0, a \neq b \\ \frac{1}{\exp(1)^{1/r}} \left(\frac{a^{a^r}}{b^{b^r}} \right)^{1/(a^r - b^r)} & r = s \neq 0, a \neq b \\ \sqrt{ab} & r = s = 0, a \neq b \\ a & a = b. \end{cases}$$

As with the generalized logarithmic mean, this satisfies the identity

$$0 \equiv \sum_{i=1}^{n} w_i \mathcal{E}_{rs} \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-s} \times \begin{cases} \left[\left(\frac{p_{i1}}{p_{i0}} \right)^s - I_r \left(\frac{p_1}{p_0}, w \right)^s \right] & \text{if } s \neq 0 \\ \left[\log \left(\frac{p_{i1}}{p_{i0}} \right) - \log \left(I_r \left(\frac{p_1}{p_0}, w \right) \right) \right] & \text{if } s = 0, \end{cases}$$

from which it follows that

$$\tilde{w}_{irs} \left(\frac{p_1}{p_0}, w \right) = w_i \mathcal{E}_{rs} \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-s} \Big/ \sum_{i=1}^{n} w_i \mathcal{E}_{rs} \left(\frac{p_{i1}}{p_{i0}}, I_r \left(\frac{p_1}{p_0}, w \right) \right)^{r-s}.$$

The `gpindex` package has functions for calculating the generalized logarithmic mean and extended mean. The `logmean_generalized()` function takes a parameter `r` and returns a function to calculate the corresponding generalized logarithmic mean. Similarly, `mean_extended()` takes a pair of parameters `r` and `s` and returns the appropriate extended mean. The `weights_transmute()` function wraps these up to calculate the weights to turn a generalized index of order r into a generalized index of order s. (There are other uses

for logarithmic means, like constructing the weights for the Vartia-I and Vartia-II indexes.)

```
> mean_harmonic(p1 / p0)

[1] 1.125967

> weights_transmute(-1, 1)(p1 / p0)

[1] 0.7401048 0.2960419 0.6831737 1.2687511 0.6343755 1.1101572

> mean_arithmetic(p1 / p0, weights_transmute(-1, 1)(p1 / p0))

[1] 1.125967

> mean_geometric(p1 / p0, weights_transmute(-1, 0)(p1 / p0))

[1] 1.125967
```

This procedure for transmuting a generalized index is the same as constructing the hybrid weights that allow a Paasche index to have an arithmetic, rather than harmonic, calculation.

```
> hybrid <- index_weights("HybridPaasche")(p0, q1)
> transmuted <- weights_transmute(-1, 1)(p1 / p0, index_weights("Paasche")(p1, q1))
> all.equal(weights_scale(hybrid), weights_scale(transmuted))

[1] TRUE
```

Quote contributions for an index are easy to calculate by transmuting the weights for the index to arithmetic weights with `weights_transmute(r, 1)()`. The `contributions()` function saves from having to do the manual calculation.

```
> contributions(-1)(p1 / p0)

[1]  0.03127685  0.12510741  0.04330641 -0.08042619  0.05361746 -0.04691528

> all.equal(sum(contributions(-1)(p1 / p0)), mean_harmonic(p1 / p0) - 1)

[1] TRUE
```

Price-updating weights

Price indexes are usually calculated for many points in time, not just two, to construct a time-series that gives the evolution of prices over time with respect to a fixed base period. One way to do this with a bilateral index is to simply chain the index values for successive periods together, so that the index in period t is

$$I_r \left(\frac{p_{t-1}}{p_0}, w \right) \times I_r \left(\frac{p_t}{p_{t-1}}, w \right).$$

This saves from having to hold onto the price information in the base period for the life of the index, and only the index value for the previous period and most recent price relatives are needed to compute the next index value in the time series.

6

In general it is not the case that the product of generalized means is the generalized mean of a product. But, it is the case that there is a collection of function $\hat{w}_{ir}$ that factor the generalized mean into two terms

$$I_r\left(\frac{p_t}{p_0}, w\right) \equiv I_r\left(\frac{p_s}{p_0}, w\right) \times I_r\left(\frac{p_t}{p_s}, \hat{w}_r\left(\frac{p_s}{p_0}, w\right)\right),$$

where

$$\hat{w}_{ir} = w_i\left(\frac{p_s}{p_0}\right)^r \bigg/ \sum_{i=1}^{n} w_i\left(\frac{p_s}{p_0}\right)^r.$$

When $r = 0$, $\hat{w}_{ir} = w_i$ so that no adjustment to the weights is needed; a geometric index can always be chained together over time. When $r = 1$, the weights given by $\hat{w}_{ir}$ are sometimes called price-updated weights, and this adjustment is essential for being able to chain an arithmetic index over time. What's important to note, however, is that this works for any generalized index.

The `weights_factor()` function takes a number `r` and returns a function to calculate the updated weights needed to chain an index of order `r`. The `weights_update()` function covers the special case of `r = 1`.

```
> p2 <- price6[[3]]
> weights <- index_weights("Laspeyres")(p0, q0)
> mean_harmonic(p2 / p0, weights) # Harmonic Laspeyres index

[1] 1.134594

> mean_harmonic(p1 / p0, weights) * mean_harmonic(p2 / p1, weights) # Too small

[1] 1.071457

> mean_harmonic(p1 / p0, weights) *
+   mean_harmonic(p2 / p1, weights_factor(-1)(p1 / p0, weights))

[1] 1.134594
```

References

Balk, B. M. (2008). *Price and Quantity Index Numbers*. Cambridge University Press.

Bullen, P. S. (2003). *Handbook of Means and their Inequalities*. Springer Science+Business Media.

ILO, IMF, OECD, Eurostat, UN, and Bank, W. (2004). *Consumer price index manual: Theory and practice*. International Monetary Fund.