A Better System of Life Tables for Data Science?

Brent Johnson

* 2018-02-12

Table of Contents

[Some concrete examples 3](#_Toc506233220)

[The Kimball distribution for replacement rates 5](#_Toc506233221)

[A replacement rate application 7](#_Toc506233222)

[Limitations 11](#_Toc506233223)

[Modifying the Kimball for a non-zero lower bound 12](#_Toc506233224)

[The Kimball distribution for price-band estimation 13](#_Toc506233225)

[Share your story 15](#_Toc506233226)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | mpg | cyl | disp | hp | drat | wt | qsec | vs | am | gear | carb |
| Mazda RX4 | 21.0 | 6 | 160 | 110 | 3.90 | 2.620 | 16.46 | 0 | 1 | 4 | 4 |
| Mazda RX4 Wag | 21.0 | 6 | 160 | 110 | 3.90 | 2.875 | 17.02 | 0 | 1 | 4 | 4 |
| Datsun 710 | 22.8 | 4 | 108 | 93 | 3.85 | 2.320 | 18.61 | 1 | 1 | 4 | 1 |
| Hornet 4 Drive | 21.4 | 6 | 258 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 | 3 | 1 |
| Hornet Sportabout | 18.7 | 8 | 360 | 175 | 3.15 | 3.440 | 17.02 | 0 | 0 | 3 | 2 |
| Valiant | 18.1 | 6 | 225 | 105 | 2.76 | 3.460 | 20.22 | 1 | 0 | 3 | 1 |

Table continues below

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | mpg | cyl | disp | | hp | | drat | wt | qsec | vs |
| **Mazda RX4** | 21 | 6 | 160 | | 110 | | 3.9 | 2.62 | 16.46 | 0 |
| **Mazda RX4 Wag** | 21 | 6 | 160 | | 110 | | 3.9 | 2.875 | 17.02 | 0 |
| **Datsun 710** | 22.8 | 4 | 108 | | 93 | | 3.85 | 2.32 | 18.61 | 1 |
| **Hornet 4 Drive** | 21.4 | 6 | 258 | | 110 | | 3.08 | 3.215 | 19.44 | 1 |
| **Hornet Sportabout** | 18.7 | 8 | 360 | | 175 | | 3.15 | 3.44 | 17.02 | 0 |
| **Valiant** | 18.1 | 6 | 225 | | 105 | | 2.76 | 3.46 | 20.22 | 1 |
|  | am | gear | | carb | |
| **Mazda RX4** | 1 | 4 | | 4 | |
| **Mazda RX4 Wag** | 1 | 4 | | 4 | |
| **Datsun 710** | 1 | 4 | | 1 | |
| **Hornet 4 Drive** | 0 | 3 | | 1 | |
| **Hornet Sportabout** | 0 | 3 | | 2 | |
| **Valiant** | 0 | 3 | | 1 | |

I’ve created an R package called [“kimball”](https://github.com/JohnsonBrent/kimball) to generate life tables based on the truncated normal distribution that I’ve found very helpful in my statistics applications. I’m going to describe it here with a couple of examples that demonstrate its usefulness. In short, the Kimball distribution is perfect for reflecting judgmental input about a random variable that one believes is positive and vaguely normal or Gaussian were it not for a lower bound (i.e., is left-truncated). But you can work with this distribution and see. Is it an improvement over others? Did it help you solve a key data science problem? If so, please respond back and tell me your story.

There are well established algorithms and R packages for implementing a truncated normal distribution, i.e., a bell-shaped density that’s truncated (or censored) on the left and/or right. One good implementation is in the [truncnorm](https://cran.r-project.org/web/packages/truncnorm/index.html) R package. This post introduces the Kimball distribution as an alternative formulation of the left-truncated normal distribution. What’s unique about the Kimball distribution is that its two parameters are completely intuitive. One specifies a Kimball distribution using just a mean and a maximum parameter–as in the distribution’s maximum likely value or the point beyond which which the probability of an observation is negligible. By contrast, a normal distribution (and other implementations of a truncated normal) is specified by a mean and standard deviation or variance parameter.

In the normal distribution, the mean parameter is straightforward. But the standard deviation (or variance) parameter is less so. Of course one learns the definition of a standard deviation in Statistics 101. It’s defined as the square root of the average squared deviation between each observed value and the mean. But most students are seldom able to think about observed phenomena in such terms. Few data scientists have a good intuitive sense for square roots of average squared deviations. In fact, other than when one of the necessary parameters is the mean, parameter interpretability isn’t a common feature among most probability distributions. The parameters of most common probability distributions don’t always have an easy real-world, behavioral interpretation. However, all else equal, I much prefer the ones that do.

Enter the Kimball distribution. First articulated in 1947 by Bradford Kimball[[1]](#footnote-1) (note he didn’t name the distribution after himself. I’m doing that.) and then revisited by Oates and Spencer (1962)[[2]](#footnote-2), the Kimball has largely been forgotten. What makes it compelling and worth reconsidering is that it’s entirely specified by just a mean and a maximum. (In its original form, the Kimball distribution also assumes a minimum of zero–something I relax in one of my later examples below, suggesting yet a third intuitive parameter for those applications that need it). Like all distributions, the Kimball has a variance–which one can calculate by minimizing a function determined by its mean and maximum values. (See Oates & Spencer’s (1962) equation #7 for the variance formula). But for the Kimball, the variance is a descriptive term only. It’s not a direct parameter and one doesn’t need to articulate a variance (or standard deviation) to specify the distribution. Rather, one specifies the Kimball distribution simply in terms of its mean and maximum.

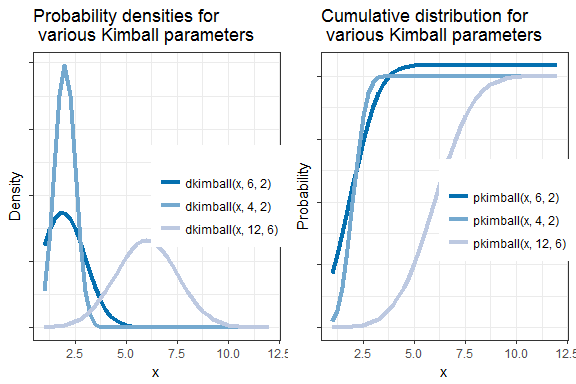
## Some concrete examples

Here below are several examples of the Kimball distribution’s probability density. Following the naming convention for other probability distributions in R, I use the dkimball() function to compute the probability density for a given set of parameters.

library(dplyr)  
library(kimball)  
x <- seq(from=1, to=12, by=.25)  
density1 <- dkimball(x, 6, 2)  
density2 <- dkimball(x, 4, 2)  
density3 <- dkimball(x, 12, 6)  
combined1 <- data.frame(x, density1, density2, density3)

And I use pkimball() for the corresponding cumulative distribution function or CDF:

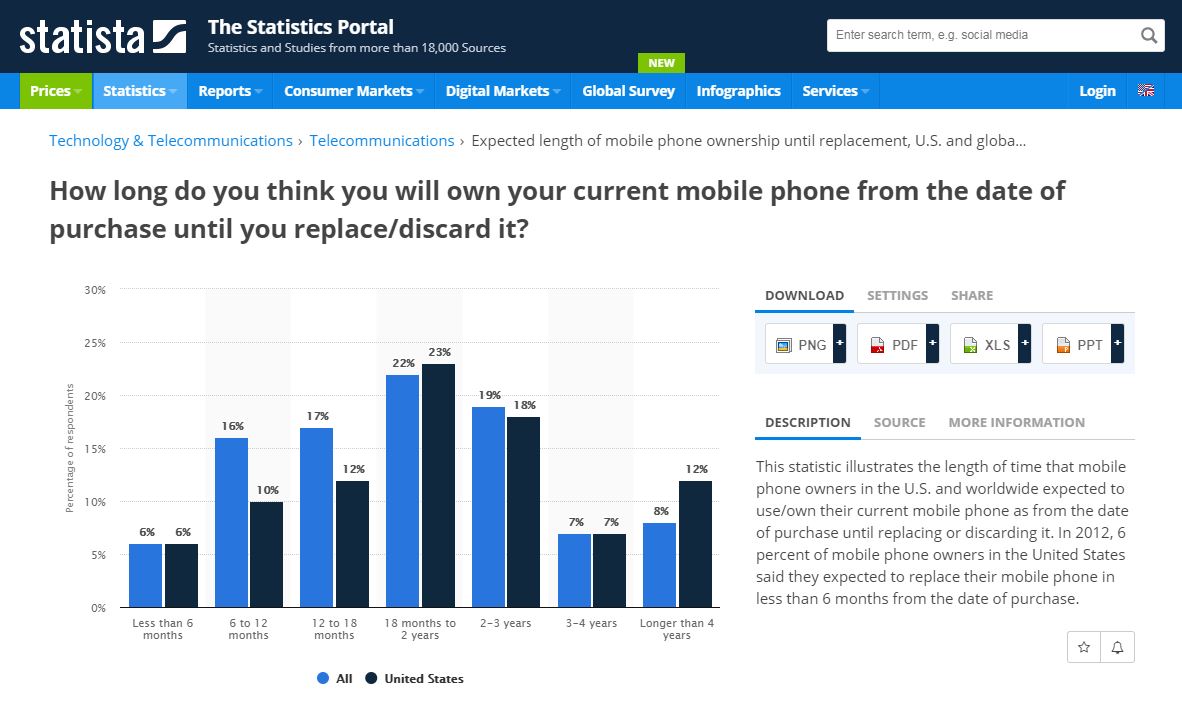
cumulative1 <- pkimball(x, 6, 2)  
cumulative2 <- pkimball(x, 4, 2)  
cumulative3 <- pkimball(x, 12, 6)  
combined2 <- data.frame(x, cumulative1, cumulative2, cumulative3)



Kimball (1947), followed by Oates and Spencer (1962), developed this distribution to estimate life tables. For example, the kimball density could describe the distribution of human lifespans. In this later application (life tables and mortality rates) the Kimball competes with the much more commonly used [Gompertz distribution](https://en.wikipedia.org/wiki/Gompertz_distribution). Similarly, the Kimball can be applied to an estimate of the probability of a machine part’s failure, for example. In this case, it’s an alternative to the [Weibull](https://en.wikipedia.org/wiki/Weibull_distribution) distribution. But again, the Gompertz distribution’s two parameters (eta and b) and the Weibull’s parameters (lambda and k) are somewhat opaque. They are inputs to computing mean and variance (or standard deviation) and they don’t have an easy interpretation..

## The Kimball distribution for replacement rates

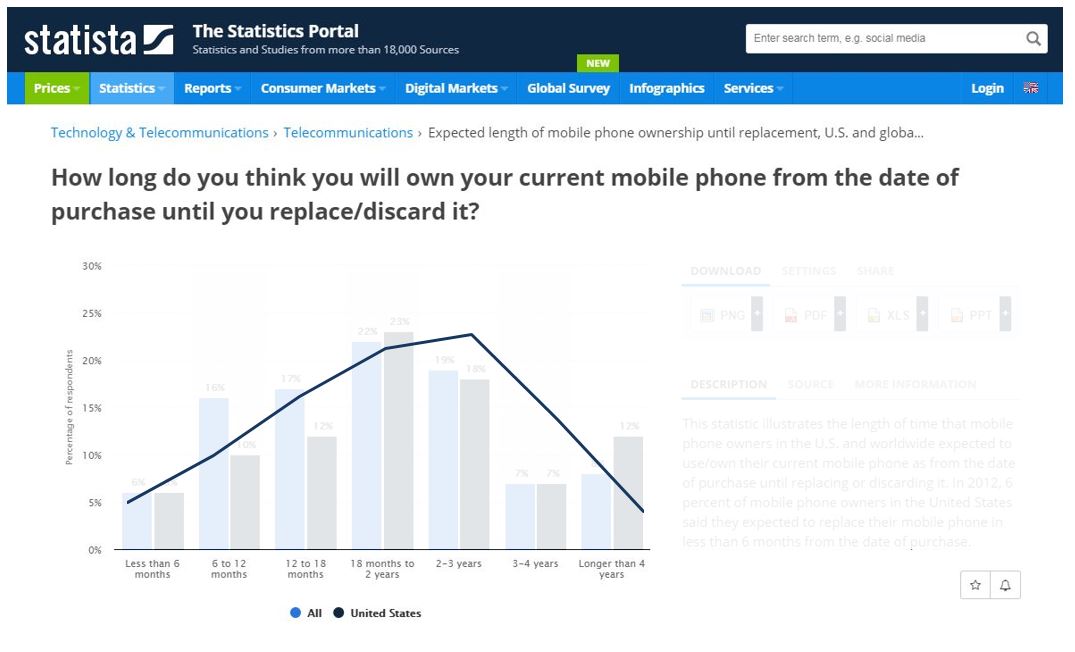
The benefit of a distribution with interpretable parameters is that it lends itself to estimation via easily elicited inputs. One can examine easily collected data or judgments and use these to estimate the Kimball’s parameters. For example, per the simple, publicly available survey data below, a mobile phone has a median expected life of between 1.5 to 2 years (in both the US and in all countries). Looking a little more closely and mentally projecting the curve below further to the right (since the last couple of years are grouped in the chart), it appears that perhaps the expected maximum life of a mobile phone is around 6 years.

[](https://www.statista.com/statistics/241036/ownership-time-length-of-current-mobile-phone-until-replacement/)

It’s easy to fit a corresponding truncated normal probability density to these data above using the dkimball function:

time <- c(0, .5, 1, 1.5, 2, 3, 4, 5, 6)  
kimball.dens <- dkimball(time,6,2)   
  
# group last 3 periods to match the given chart  
char.time <- data.frame(time=c(0, .5, 1, 1.5, 2, 3, "4+"))  
kimball.dens.sum <- c(kimball.dens[1:6],sum(kimball.dens[7:9]))   
  
#normalize due to irregular intervals in 'time'  
kimball.dens.sum <- kimball.dens.sum / sum(kimball.dens.sum) #

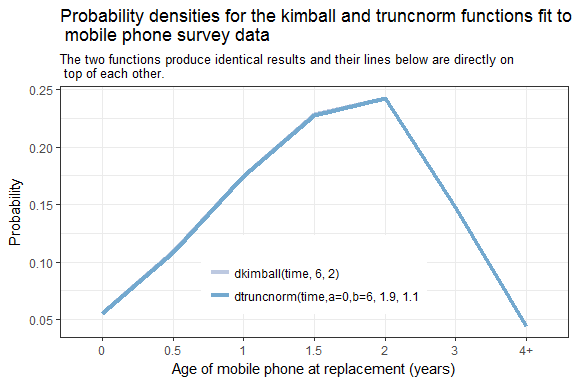
Here’s the above Kimball density overlayed with the observed survey data:

[](https://www.statista.com/statistics/241036/ownership-time-length-of-current-mobile-phone-until-replacement/)

Note that I’m not using a formal parametric estimation method, such as maximum likelihood, to fit the kimball distribution to these survey data. I could indeed do so if I wanted to take the time. however, the parameters for the kimball distribution are already given to me in the observed statistics in the original image above. The parameters for the kimball are straightway handed to me in the way one typically describes such events. The Kimball’s parameters reflect the everyday language one uses to describe the age of durable products and other phenomena.

One can compare the Kimball density to the density from the truncnorm package–another implementation of the truncated normal distribution. Optimizing dtruncnorm() to achieve the same fit as dkimball() above yields a mean and standard deviation for dtruncnorm of 1.9 and 1.1, respectively. As seen in the image below, the two distributions mimic each other exactly. Their lines are indistinguishable. They’re both truncated normal distributions. But, again, one can judgmentally estimate the Kimball version of the truncated normal distribution. For the Truncnorm version it requires other methods to obtain the best-fitting standard deviation.

library(truncnorm)  
truncnorm.dens <- dtruncnorm(time,a=0,b=6, 1.9, 1.1)   
  
# group last 3 periods to match the given chart  
truncnorm.dens <- c(truncnorm.dens[1:6],sum(truncnorm.dens[7:9]))  
  
# normalize due to irregular intervals in 'time'  
truncnorm.dens <- truncnorm.dens / sum(truncnorm.dens)



## A replacement rate application

To extend the mobile phone example above, I sometimes use the kimball distribution to estimate replacement sales of consumer or business products If one knows, say, the total number of mobile phones sold in a given year, the kimball distribution (applied retrospectively) enables an estimate of the replacement sales volume. And one can also apply the Kimball distribution prospectively to easily forecast likely replacement sales in the future. For example, here below is publicly [available data](https://www.statista.com/statistics/271539/worldwide-shipments-of-leading-smartphone-vendors-since-2007/) showing world-side historic mobile phone sales (in millions).[[3]](#footnote-3)

|  |  |
| --- | --- |
| year | shipments |
| 2007 | 122.36 |
| 2008 | 151.40 |
| 2009 | 173.50 |
| 2010 | 304.70 |
| 2011 | 494.60 |
| 2012 | 725.30 |
| 2013 | 1019.50 |
| 2014 | 1301.70 |
| 2015 | 1437.20 |
| 2016 | 1470.60 |
| 2017 | 1468.10 |

And here’s how one might work with these data in practice and apply the kimball distribution to discover a richer story. I start this below by defining a function to put lagged values of total shipments into separate variables/columns which I name shipments1, shipments2, etc. And I save these lagged mobile phone shipment values inside a data frame called lagged.shipments. But note that the first variable inside lagged.shipments isn’t a lagged value per se but is the current period’s sales. When it comes to products like mobile phones, there’s usually some small fraction of product that gets replaced within the same year they were purchased (perhaps due to breakage or returns).

I next take the replacement densities computed above using the survey data and insert this inside the mapply() function. The mapply function multiplies my 7-column matrix of lagged sales history (lagged.shipments) by the 7-element vector of replacement probabilities (replacement.schedule).

# create a function to lag variable x a total of n times and name it with a prefix--keeping   
# the original variable. This function creates n lags. Hence, it returns n+1 columns (including original).  
f <- function(x, n, pad, prefix="lag") {  
 if(!missing(pad)) {  
 X <- c(rep(pad, n), x)  
 }  
 y <- data.frame(embed(X, n+1))  
 names(y) <- c(gsub('.\*\\$', '', deparse(substitute(x))), paste(prefix, seq(1:(n)), sep=""))  
 return(y)  
}  
  
# apply above function and create SIX lagged shipment values (plus current period)  
lagged.shipments <- f(smartphone$shipments, 6, 0, "shipments")  
  
# define vector of replacement probabilities given avg life=2 and maxlife=6  
time <- seq(0, 6)  
kimball.dens <- dkimball(time,6,2)  
replacement.schedule <- kimball.dens / sum(kimball.dens)  
  
# compute estimateed replacement sales  
smartphone$replacements <- rowSums(mapply(`\*`, lagged.shipments, replacement.schedule))

The result is an estimate of replacement mobile phone sales. Note that in the table below I also calculate replacement sales as a percentage of total sales (which is just replacements / shipments). And I compute first-time sales by subtraction (or shipments - replacements).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| year | shipments | replacements | replacement share | first-time sales |
| 2007 | 122.36 | 10.17 | 8.3% | 112.19 |
| 2008 | 151.40 | 44.81 | 29.6% | 106.59 |
| 2009 | 173.50 | 99.01 | 57.1% | 74.49 |
| 2010 | 304.70 | 153.5 | 50.4% | 151.2 |
| 2011 | 494.60 | 225.58 | 45.6% | 269.02 |
| 2012 | 725.30 | 350.17 | 48.3% | 375.13 |
| 2013 | 1019.50 | 535.43 | 52.5% | 484.07 |
| 2014 | 1301.70 | 770.72 | 59.2% | 530.98 |
| 2015 | 1437.20 | 1027.11 | 71.5% | 410.09 |
| 2016 | 1470.60 | 1248.93 | 84.9% | 221.67 |
| 2017 | 1468.10 | 1388.65 | 94.6% | 79.45 |

Applying the kimball function like this allows the following observations:

* Fewer mobile phones are being sold to first-time buyers over time.
* Replacement sales are a growing share of the market. That is, an increasing share of customers are on their second, third, or fourth, etc. mobile phone.
* The market for mobile phones is close to saturation. Unless something new happens, sales growth of mobile phones is likely to continue to slow–just like it did starting in 2017.

Granted, these observations rest on a lot of assumptions. I’m first assuming that the mobile phone replacement rate was constant over this entire time period. A longer average life earlier in time would result in larger pool of first-time buyers today and the market might not be quite so close to saturation after all. In addition, the potential pool of first-time buyers could possibly expand if the price and feature sets of mobile phones improved such that they became more affordable or more attractive.

The point of this example is to show how the Kimball distribution allows an easy way to incorporate judgmental AND analytic inputs about a consumer or business products’s replacement rate–all in the context of easily understood parameters–the mean and maximum.

## Limitations

There are some limitations and parameter domain restrictions on the Kimball-related functions in this package. The Kimball functions will generate warnings and/or fail to provide good estimates if the user violates the restrictions.

Domain restrictions on the parameters are manifest in a key ingredient to the derivation of the Kimball: the ratio between the maximum and the mean parameter. By definition, this maximum-to-mean ratio is bounded on the lower end by 1 since the mean population value must lie below the maximum population value. Likewise, there’s a boundary on the upper end of the ratio (a maximum well in excess of the mean) beyond which the Kimball’s functions become undefined. Oates & Spencer’s (1962) tables place the domain for the maximum-to-mean ratio between 1.22 and 4.88. In other words, the population maximum ought not be lower than 22 percent above the population mean and no higher than nearly 5 times the mean.

Oates & Spencer (1962) say that in most survey or engineering applications, the user will rarely need to fit a truncated normal to populations with a maximum-to-mean ratio outside this range. And these restrictions rarely pose a problem in practice. A Kimball distribution fit with a maximum/mean ratio below 2 is nearly symmetric and starts to resemble a non-truncated normal distribution. In this case, dnorm() (R’s implementation of normal or Gaussian density) may be more appropriate than dkimball. Moreover, dnorm() is most certainly more efficient. And at the other extreme, attempting to fit a Kimball distribution with a maximum/mean ratio above 4 results in a highly skewed distribution. In that case, a negative exponential distribution (e.g., dexp()) or gamma distribution (e.g., dgamma()) is likely more appropriate and more efficient than dkimball.

The Kimball functions will still attempt to return a value if the maximum-to-mean ration exceeds the recommended domain (1.22 < x < 4.88), but the warning should encourage the user to check the result.

My implementation adopts yet another assumption expressed by Oates & Spencer (1962). In the Kimball distribution, the maximum parameter is defined as the point at which it’s expected that only 1 unit out of 1,000 exists in the population. This is an intuitive assumption in practice, but it creates the condition wherein the Kimball’s cumulative distribution doesn’t resolve to precisely 1.0 but to some small value in excess of 1. Therefore, technically speaking, the Kimball distribution function is only an *approximate* probability distribution. Kimball (1947) even refers to his work merely as a “system of life tables” and not a formal distribution.

Finally, I don’t allow negative population values–and hence, no negative mean or maximum parameter values. The assumed population domain for the Kimball is strictly positive.

## Modifying the Kimball for a non-zero lower bound

The original Kimball distribution also assumes a minimum population value of zero (x >= 0) and a lower bound other than zero is not permitted. This assumption goes back to Kimball’s (1947) work application to life tables in which a zero lower bound is a reasonable assumption. After all, no species ever lives a *negative* number of months or years. And in my implementation, zero is the default minimum when specifying any of the Kimball functions. However, I include an option to over-ride this minimum assumption. Within each function’s options, one can specify a non-zero (but still positive) minimum if desired. Here are some examples:

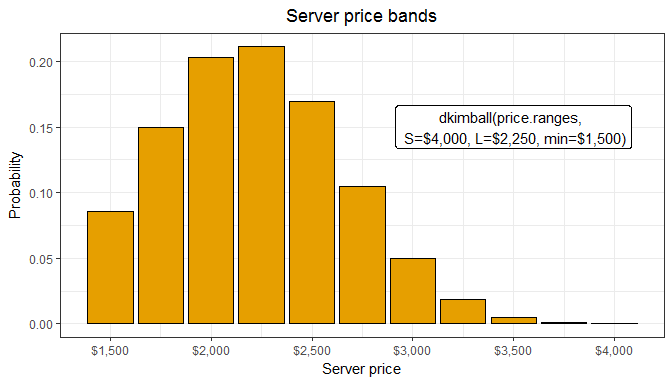
x <- seq(from=1, to=10, by=2)  
dkimball(x, 6, 3, min=1)  
#> [1] 2.447264e-02 4.884809e-01 2.215891e-02 2.284458e-06 5.352441e-13  
  
x <- seq(from=100, to=1000, by=100)  
pkimball(x, 1000, 300, 100)  
#> [1] 0.6081542 0.9003049 1.1732047 1.3801143 1.5074440 1.5710406 1.5968199  
#> [8] 1.6053002 1.6075639 1.6080542

Removing the restriction of a zero minimum makes the Kimball distribution a natural fit for yet more creative applications.

## The Kimball distribution for price-band estimation

Another task with which I am sometimes faced is to estimate the distribution of prices for a given product. For example, a client may provide me with research (or an assumption) telling me the average price of a computer server is around $2,250, the minimum price for a stripped down version is $1,500 and the maximum price for a richly configured workplace server–one with lots of memory and processors–is $4,000. The client shares all this information with a desire to know the likely share of servers sold in $250 price increments. Experience tells me that such price distributions often resemble a truncated normal distribution. That is, there’s a sharp lower bound below which no vendor prices their products. Also, most customers aren’t interested in the lowest-price products. Such products often don’t have enough features. And at the other extreme, there’s often a long tail reflecting a gradual decline in the number of products sold at ever increasing prices and feature sets. This reflects the fact that highly priced products are often geared towards niche uses. The Kimball function is ideal given this scenario and the limited information that I usually have about the population. The dkimball() function solves this problem rather easily. Here’s the Kimball density with precisely these parameters:

# specify the ranges  
price.ranges <- seq(from=1500, to=4000, by=250)  
  
# compute the densities  
price.densities <- dkimball(price.ranges, 4000, 2250, 1500)  
price.probabilities <- price.densities / sum(price.densities)



And here’s a table version of the above probabilities and price bands. For example, I’ve estimated that 21% of all servers are sold in the $2250 - $2,500 price range.

|  |  |
| --- | --- |
| price | probabilties |
| $1,500 | 8.6% |
| $1,750 | 15.0% |
| $2,000 | 20.3% |
| $2,250 | 21.1% |
| $2,500 | 17.0% |
| $2,750 | 10.5% |
| $3,000 | 5.0% |
| $3,250 | 1.9% |
| $3,500 | 0.5% |
| $3,750 | 0.1% |
| $4,000 | 0.0% |

Of course its entirely possible that a big server manufacturer in this product example only sells servers in excess of $3,000 suggesting some other distribution. But without such disproving information (in fact with only a marginal amount of judgmental information), a single-mode distribution, covering only a positive range, and that is skewed right, i.e., is leaning left (which is most definitely the case since the mean is closer to the minimum than it is to the maximum), the Kimball distribution is a logical choice.

There are a number of other distributions that I could fit to achieve this same shape. A Weibull, gamma, lognormal, or even an alternative implementation of the truncated normal, could all be specified with parameters such that they have a shape similar to the above. However, I don’t have a complete set of observed data for all price bands. I don’t even have information on any single price band. Such lack of data prevents me from using a parametric estimation approach, such as maximum likelihood, which I’d need to fit these other distributions. Instead, I’m left to judge the disribution. But none of these other distributions contain interpretable parameters that are as easily judged. By contrast, the Kimball distribution’s parameters (a mean, minimum, and maximum) are easily obtained. The very words used to describe the price band problem are precisely the Kimball distribution’s necessary and sufficient parameters.

## Share your story

The Kimball package and system of life tables enables the data scientist to easily fit a wide range of data using easily observed and judgmental inputs. I keep the Kimball close at hand when I need a probability density with intuitive and interpretable parameters.

Strange that such a helpful set of functions should remain so obscure for nearly 75 years. Perhaps putting it in an R package can help remedy that and put the knowledge into more hands. If you find the Kimball package helpful, if you apply it to something new and different from the above examples, I’d sure like to hear your story.

Have you got an application for which the Kimball distribution works particularly well?

1. Kimball, Bradford F., “A System of Life Tables for Physical Property Based on the Truncated Normal Distribution”. (1947). *Econometrica*, Vol. 15, No. 4 (Oct), pp. 342-360 [↑](#footnote-ref-1)
2. Oates, Thomas A. and Milton H. Spencer, “A System of Retirement Frequencies for Depreciable Assets” (1962). *The Accounting Review*, Vol. 37, No. 3 (July), pp. 452-459 [↑](#footnote-ref-2)
3. I’m going to assume in this example that mobile phone shipments equal mobile phone sales. In practice, however, there’s a small amount of inventory in the channel and shipments in a given time period usually don’t precisely match end-user sales for that same time period. [↑](#footnote-ref-3)