Extinction risk meta-analysis — Urban 2015

> Urban (2015) conducted a meta-analysis of extinction risks and its relationship to climate change. He included 131 studies. In Figure 1, he plotted the number of studies reporting a certain overall proportion of extinction risk. The data (`data/Urban2015_data.csv`) is at a finer resolution than what needed for this figure. In fact, each study has been split into different lines according to the method and taxa used to compute the extinction risk. To reproduce Figure 1, you will need to coarse grain the data by grouping lines with the same author/year, and for each study compute the proportion of species at risk for extinction (sum the `N.Ext` for each study, and divide for the corresponding sum of `Total.N`). A close inspection of the original Figure shows that the data has been plotted in bins of unequal size (e.g., '0.5 < proportion < 1' is in one bin) so you will need to classify the various proportions into appropriate bins (0, 0-0.05, 0.05-0.1, . . . , 0.5-1) before plotting. A `ggplot2` version of Figure 1 of the original paper is reported in `data/Urban2015_figure1.pdf`. Reproduce the figure.

```r
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##     filter, lag
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
u2015 <- read.csv("../data/Urban2015_data.csv", sep = '\t', stringsAsFactors = FALSE)
```

Now we need to coarse-grain the data according to the study (`Author`, `Year`). In particular, we want to compute the `risk` by summing all the `N.Ext`, and the `total` by summing `Total.N`. This can be accomplished by either cycling through the data and build a new data frame, or using the `summarise` function of `dplyr`:

```r
by_study <- summarise(group_by(u2015, Author, Year), risk = sum(N.Ext), total = sum(Total.N))
# look at the results
by_study
```

```
## Source: local data frame [130 x 4]
## Groups: Author [?]
##
##       Author  Year  risk total
##        (chr) (int) (int) (int)
## 1     Albouy  2013    24    59
## 2    Anciaes  2006    15    98
## 3   Anderson  2009     1     6
## 4     Araujo  2004   135  2400
## 5   Bakkenes  2006    49  5240
## 6     Balint  2011     4    18
```

```
## 7   Bambach  2013    34   168
## 8  Beaumont  2002     5   288
## 9  Beaumont  2005     0   150
## 10 Beaumont  2007     4    27
## ..       ...   ...   ...   ...
```

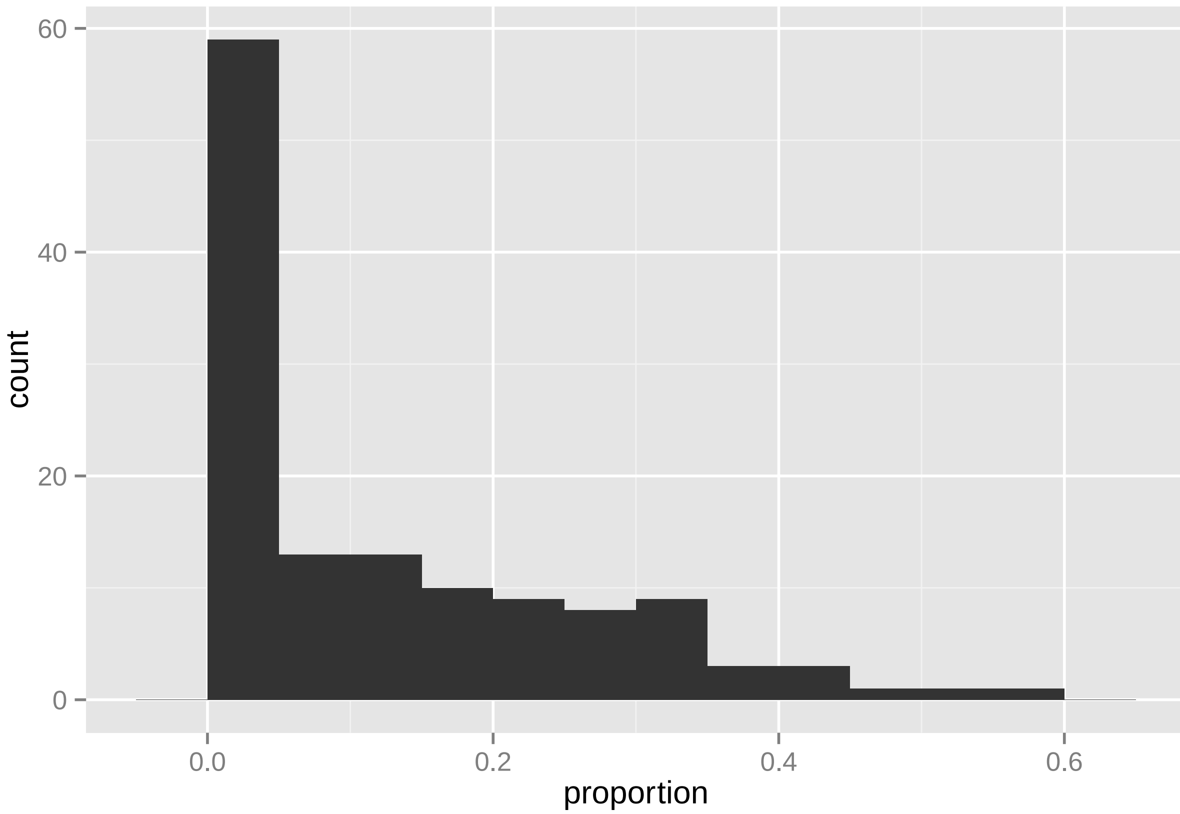
Now let's add a column expressing the proportion of species at risk of extinction:

```
by_study <- mutate(by_study, proportion = risk / total)
by_study
```

```
## Source: local data frame [130 x 5]
## Groups: Author [109]
##
##      Author  Year  risk total  proportion
##       (chr) (int) (int) (int)       (dbl)
## 1    Albouy  2013    24    59 0.406779661
## 2   Anciaes  2006    15    98 0.153061224
## 3  Anderson  2009     1     6 0.166666667
## 4    Araujo  2004   135  2400 0.056250000
## 5  Bakkenes  2006    49  5240 0.009351145
## 6    Balint  2011     4    18 0.222222222
## 7   Bambach  2013    34   168 0.202380952
## 8  Beaumont  2002     5   288 0.017361111
## 9  Beaumont  2005     0   150 0.000000000
## 10 Beaumont  2007     4    27 0.148148148
## ..       ...   ...   ...   ...         ...
```

We can plot the data using `geom_bar` and adjust the `binwidth`.

```
ggplot(data = by_study, aes(proportion)) + geom_bar(binwidth = .05)
```



However, it looks slightly different from Figure 1 in the publication given the unequal width of the bins. In order to exactly reproduce the figure, we can construct a column `risk_bin` that classifies the proportion of species at risk into bins of variable length. This can be accomplished using the function `findInterval`. For example:

```
head(cbind(by_study$proportion, findInterval(by_study$proportion, c(0, 0.000001, seq(0.05, 0.5, by = 0.0
```

```
##             [,1] [,2]
## [1,] 0.406779661   10
## [2,] 0.153061224    5
## [3,] 0.166666667    5
## [4,] 0.056250000    3
## [5,] 0.009351145    2
## [6,] 0.222222222    6
```

Let's add this column to the table `by_study`:

```
by_study$risk_bin <- findInterval(by_study$proportion, c(0, 0.000001, seq(0.05, 0.5, by = 0.05)))
```

And transform this into a factor, with the right labels. First, build the labels for each bin:

```
leftbound <- c(0, seq(0.0, 0.45, by = 0.05), 0.5)
leftbound
```

```
##  [1] 0.00 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50
```

```r
rightbound <- c(0, seq(0.05, 0.5, by = 0.05), 1)
rightbound
```

```
##  [1] 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 1.00
```

```r
label_risk_bin <- paste(leftbound, rightbound, sep = "-")
label_risk_bin
```

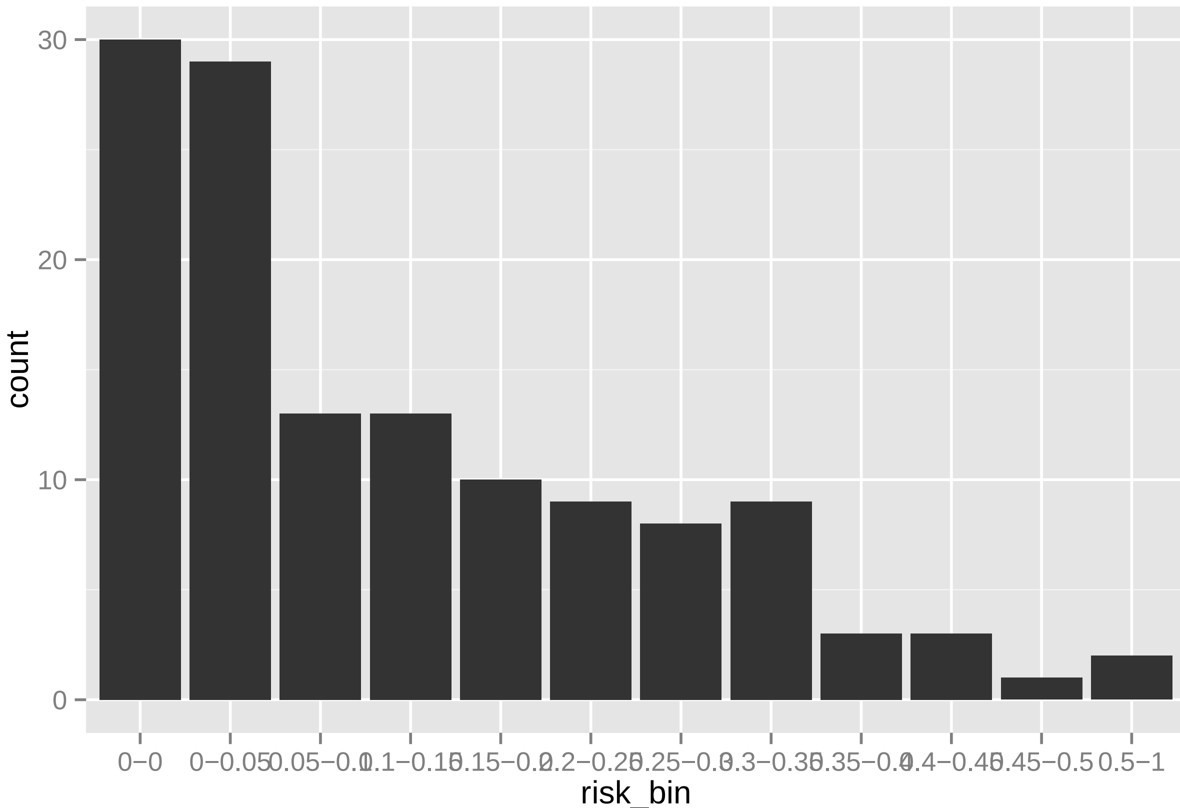
```
##  [1] "0-0"      "0-0.05"   "0.05-0.1" "0.1-0.15" "0.15-0.2" "0.2-0.25"
##  [7] "0.25-0.3" "0.3-0.35" "0.35-0.4" "0.4-0.45" "0.45-0.5" "0.5-1"
```

```r
# now transform bin_risk into factors
by_study$risk_bin <- factor(by_study$risk_bin, levels = 1:12, labels = label_risk_bin)
# see the result
by_study
```

```
## Source: local data frame [130 x 6]
## Groups: Author [109]
##
##      Author  Year  risk total  proportion risk_bin
##       (chr) (int) (int) (int)       (dbl)   (fctr)
## 1    Albouy  2013    24    59 0.406779661 0.4-0.45
## 2   Anciaes  2006    15    98 0.153061224 0.15-0.2
## 3  Anderson  2009     1     6 0.166666667 0.15-0.2
## 4    Araujo  2004   135  2400 0.056250000 0.05-0.1
## 5  Bakkenes  2006    49  5240 0.009351145   0-0.05
## 6    Balint  2011     4    18 0.222222222 0.2-0.25
## 7   Bambach  2013    34   168 0.202380952 0.2-0.25
## 8  Beaumont  2002     5   288 0.017361111   0-0.05
## 9  Beaumont  2005     0   150 0.000000000      0-0
## 10 Beaumont  2007     4    27 0.148148148 0.1-0.15
## ..      ...   ...   ...   ...         ...      ...
```

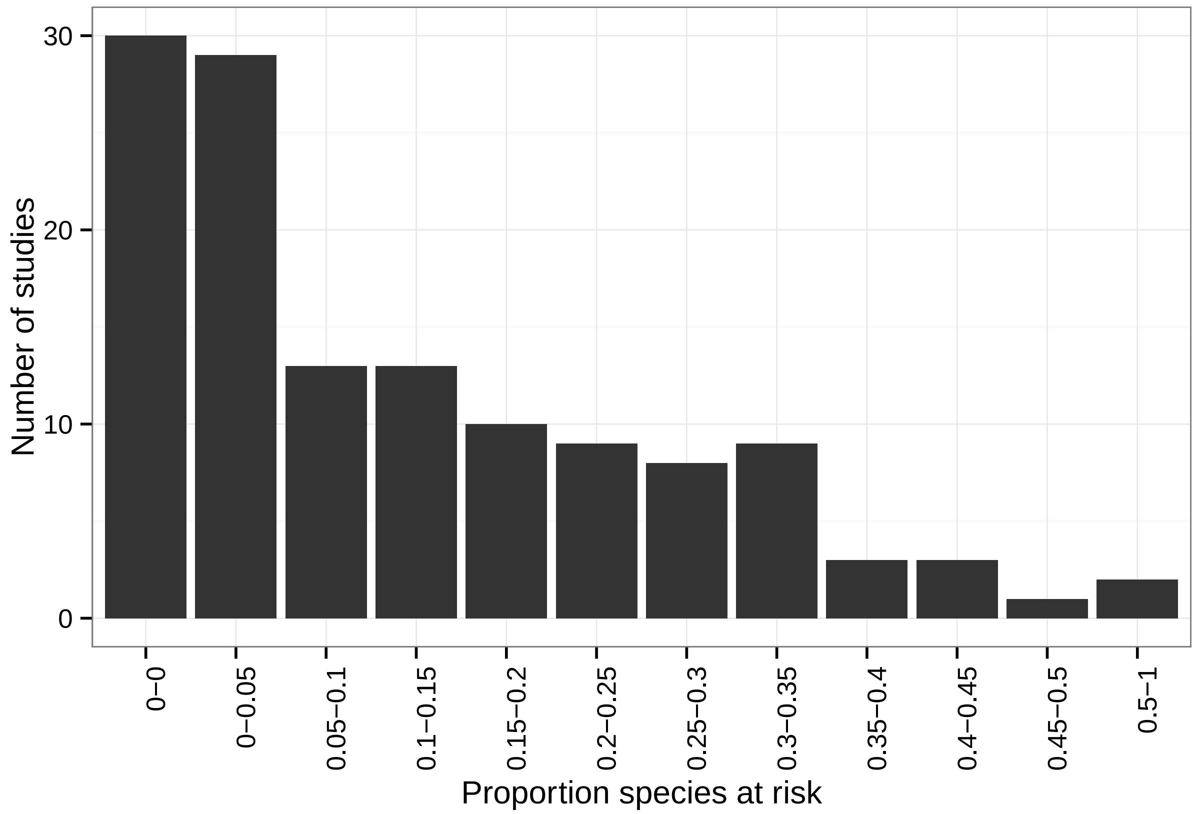
After all this data reshaping, we are ready to plot!

```r
pl <- ggplot(data = by_study, aes(x = risk_bin)) + geom_bar()
pl
```



Make the graph prettier:

```
pl <- pl + theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  xlab("Proportion species at risk") +
  ylab("Number of studies")
pl
```



Ready to save the plot using `ggsave`:

```
ggsave(pl, file = "../data/Urban2015_figure1.pdf")
```

```
## Saving 6.5 x 4.5 in image
```