Reproducing Valverde

The aim of this script is to reproduce two of the plots in the the paper "Topology and Evolution of Technology Innovation Networks", which summarises the USPTO patent network. The first figure being reproduced looks at the number of patents over time, and the second is the in-degree distribution and fitting a extended power-law distribution to that.

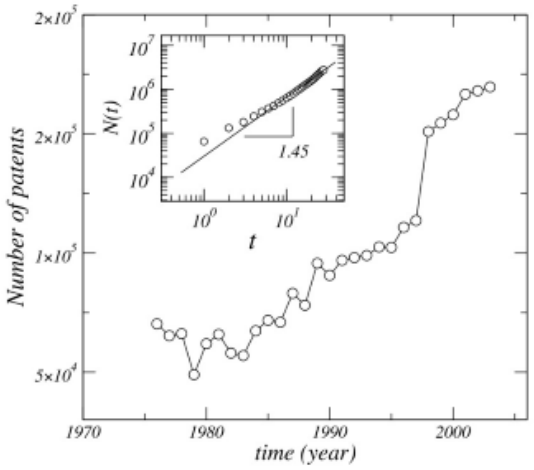


FIG. 2: Time evolution of the number of patents $N(t)$ in the USPTO dataset from 1973 to 2004. Inset: Cumulative number of patents on a log-log scale, showing a scaling $N(t) \sim t^{\theta}$.

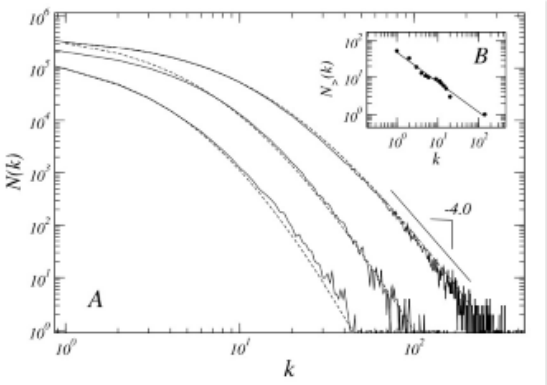


FIG. 3: (A) The in-degree distribution for the patent citation network follows an extended power-law distribution, $P_i(k) \sim (k+k_0)^{-\gamma}$. Three distributions are displayed for three different time windows, namely 1984 (leftmost), 1992 (center) and 2002 (rightmost). (B) The in-degree distribution for the subset of patents displayed in fig.1 f (for computer tomography) is roughly approximated by a scale free distribution. The leftmost point indicates the central hub in fig.1.

Figure 1:

The data being used for these plots in the cleaned concatonated patent data produced by the 'cleaned' script. We have extracted the year into a seperate feature as it is this which the data is summarised over. It contains a list of the patent numbers and date in which those patents were granted.

```
# Read data
data <- read_csv("../DataFiles/Cleaned/patent_cat.csv", progress = FALSE)
data$Year <- year(data$Date2)
str(data)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    5803302 obs. of  5 variables:
##  $ Patent: chr  "RE28671" "RE28672" "RE28673" "RE28674" ...
##  $ Date  : int  19760106 19760106 19760106 19760106 19760106 19760106 19760106 19760106 19760106 197
##  $ Order : int  8 7 3 5 13 2 11 8 3 4 ...
##  $ Date2 : Date, format: "1976-01-06" "1976-01-06" ...
##  $ Year  : num  1976 1976 1976 1976 1976 ...
```

Plot 1: Patents granted by year

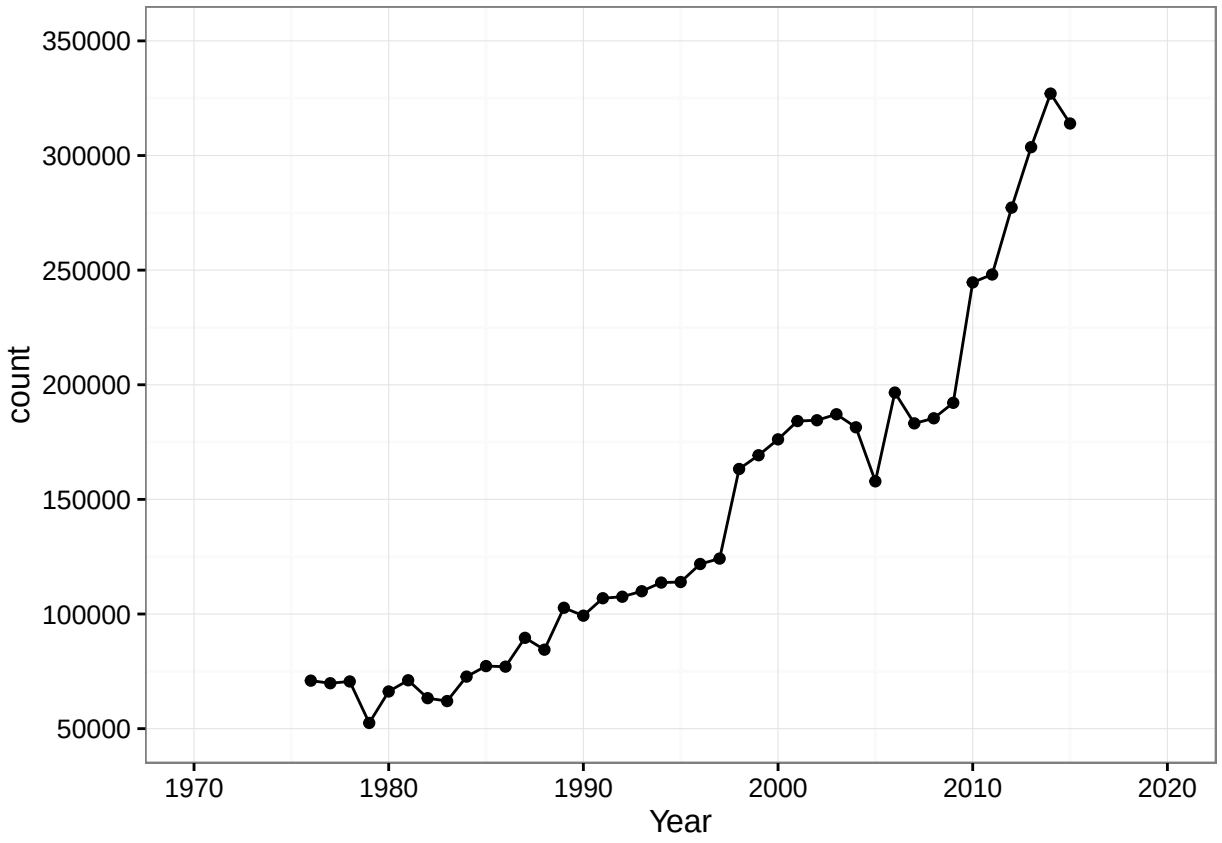
The first plot counts the number of patents granted by year, there is an inset which looks at this as a cumulative sum on a log-log scale.

In order to process the data into a tidy format processable by ggplot, we summarise it counting the number of entries (patents) for each year. There are a few outliers (1) so this is then filtered to only allow only patents

within that range.

```
# Summarise results
year_counts <- data %>% group_by(Year) %>% summarise(count = n()) %>% filter(Year %in% 1976:2015)
```

```
(gg1 <- ggplot(data = year_counts, aes(x = Year, y = count)) +
    geom_line() +
    geom_point(shape = 19) +
    scale_x_continuous(limits = c(1970,2020), minor_breaks = seq(1975, 2015, 10) ) +
    scale_y_continuous(limits = c(5e4, 3.5e5), breaks = seq(5e4, 35e4, 5e4)) +
    theme_bw()
 )
```



Notes on plot1

From our reproduction of the first figure we can make some observations:

- Although exact numbers aren't given by cross referencing the two graphs certainly **follow the same shape**.
  - In terms of exact numbers the region leading **up to 1990 seems identical** but **after this point there may be larger numbers in our data**, e.g. in our data 1997 reaches 1.25e5 but there's is much closer to 1.18 ish. A fairly constant gap of around this ammount seems to persist until the end of the data.
  - I have parsed foreign patents but on an earlier encarnation when foreign patents were missed there was a large discrepency in number of patents on this graph. So there could be a systematic parsing error or difference in what is being parsed. Either way I believe this my parsing to be

more accurate but an analysis into the differences between this graph and other uspto data sources for patent counts has been conducted elsewhere to check for systematic problems.

- The data has a **jump at the year 1997-1998**.
- Valverde's data ends at 2004, **after 2005** the number of patents becomes much **more erratic** but also **climbs very steeply**.
- Worth noting is that there are 3 different formats in which the data was stored, a propriotary format from 1976:2001, an sgml format from 2001:2004 and an xml format since then (although there have been minor itterations to this format since).
  - In a previous version of the analysis the method for parsing sgml and propriotary had systematic differences so there was a step at the year 2001. The current parsing method uses the same method for each data type to ensure consistency. Having said this the erratic behaviour starts at the same time xml is begun to be used.

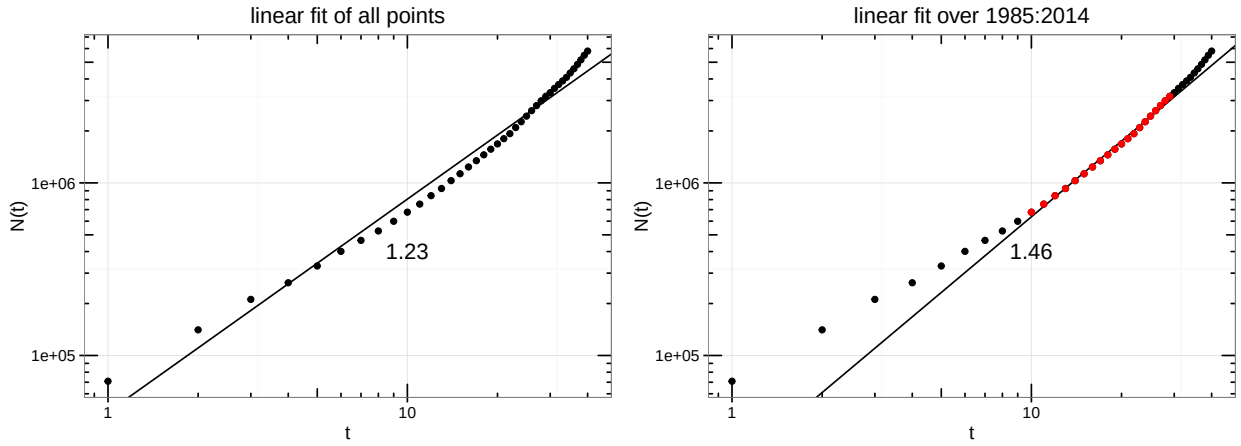
Plot1 inset

The inset of plot 1, shows the "Cumulative number of patents on a log-log scale, showing a scaling: $N(t) \sim t^{\theta}$"

```
year_counts$cum_count <- cumsum(year_counts$count)
year_counts$rel_year <- year_counts$Year - 1975
```

```
gg1b <- ggplot(data = year_counts, aes(x = rel_year, y = cum_count)) +
    geom_point() +
    scale_x_log10() +
    scale_y_log10() +
    theme_bw() +
    labs(x = "t", y = "N(t)")
```

```
lm1 <- lm(log10(cum_count) ~ log10(rel_year), data = year_counts)

gg1b.lm1 <-
    gg1b +
    geom_abline(intercept = lm1$coefficients[1], slope = lm1$coefficients[2]) +
    ggtitle("linear fit of all points") +
    annotate(label = round(lm1$coefficients[2],2), x = 10, y = 4e5, geom = "text", size = 5) +
    annotation_logticks(sides = "tblr")

t <- year_counts %>% filter(Year %in% 1985:2004)
lm2 <- lm(log10(cum_count) ~ log10(rel_year), data = t)
gg1b.lm2 <-
    gg1b +
    geom_abline(intercept = lm2$coefficients[1], slope = lm2$coefficients[2]) +
    ggtitle("linear fit over 1985:2014") +
    geom_point(data = t, col = "red") +
    annotate(label = round(lm2$coefficients[2],2), x = 10, y = 4e5, geom = "text", size = 5) +
    annotation_logticks(sides = "tblr")

grid.arrange(gg1b.lm1, gg1b.lm2, ncol=2)
```



Notes on plot1 inset

- To address the claim that this is an approxmately linear relationship, it does appear to not be linear.
  - Having said that the claim is that it is close to linear which is fairly accurate, the R-squared for the linear fit of all of the data is 0.9800419
- In fact it also seems like the axes were stretched to make the linear fit look nicer in the plot. Finally to get the fit found in the paper you have to ignore the first set of points (we found 1985:2004 although its conceievable a larger range was used with the differences in patent counts in our analysis) and finally this relationship doesn't hold into the new data since 2004.

Plot2: In degree distribution

```
degree_distribution_all <- data %>% group_by(Year, Order) %>% summarise(count = n())
degree_distribution_all <- degree_distribution_all %>%
    group_by(Year) %>%
    mutate(group_total = sum(count), freq = count / group_total)
degree_distribution <- filter(degree_distribution_all, Year %in% c(1984, 1992, 2002, 2012))
```

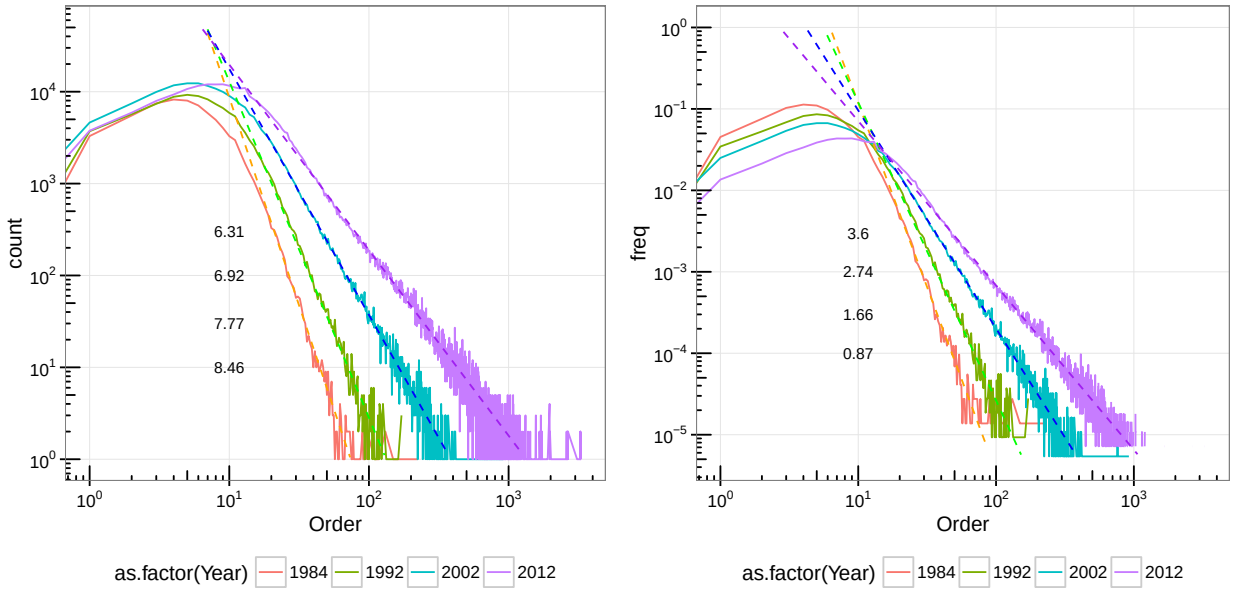
```
lm_func <- function(x, y, year, data = degree_distribution, xmin = 20, xmax = 200) {
    dat <- data %>% filter(Year == year, Order > xmin, Order < xmax)
    dat$Order[dat$Order == 0] <- NA

    model <- switch(y,
           "count" = lm(data = dat, formula = log10(count) ~ log10(Order)),
           "freq" = lm(data = dat, formula = log10(freq) ~ log10(Order)))
    list(func = model$coefficients[2] * log10(x) + model$coefficients[1], coef = model$coefficients)
}
```

```
gg2a <- ggplot(data = degree_distribution, aes(x = Order, y = count, col = as.factor(Year))) +
    geom_line(size = 0.5) +
    scale_x_log10(
        breaks = scales::trans_breaks("log10", function(x) 10^x),
        labels = scales::trans_format("log10", scales::math_format(10^.x))
    ) +
    scale_y_log10(
        breaks = scales::trans_breaks("log10", function(x) 10^x),
    labels = scales::trans_format("log10", scales::math_format(10^.x)),
```

```
    limits = c(1, 5e4)
    ) +
    annotation_logticks() +
    theme_bw() +
    theme(panel.grid.minor = element_blank()) +
    theme(legend.position = "bottom") +
    stat_function(fun = function(x) lm_func(x, y = "count", year = 1984, xmax = 50)$func,
                  geom = 'line', colour = 'orange', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "count", year = 1984, xmax = 50)$coef[1],2),
             x = 10, y = 1e1, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "count", year = 1992)$func,
                  geom = 'line', colour = 'green', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "count", year = 1992)$coef[1],2),
             x = 10, y = 3e1, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "count", year = 2002)$func,
                  geom = 'line', colour = 'blue', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "count", year = 2002)$coef[1],2),
             x = 10, y = 1e2, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "count", year = 2012)$func,
                  geom = 'line', colour = 'purple', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "count", year = 2012)$coef[1],2),
             x = 10, y = 3e2, geom = "text", size = 3)
```

```
gg2a2 <- ggplot(data = degree_distribution, aes(x = Order, y = freq, colour = as.factor(Year))) +
    geom_line(size = 0.5) +
    scale_x_log10(
        breaks = scales::trans_breaks("log10", function(x) 10^x),
        labels = scales::trans_format("log10", scales::math_format(10^.x))
    ) +
    scale_y_log10(
        breaks = scales::trans_breaks("log10", function(x) 10^x),
    labels = scales::trans_format("log10", scales::math_format(10^.x)),
        limits = c(5e-6, 1e0)
    ) +
    annotation_logticks() +
    theme_bw() +
    theme(panel.grid.minor = element_blank()) +
    theme(legend.position = "bottom") +
    stat_function(fun = function(x) lm_func(x, y = "freq", year = 1984, xmax = 50)$func,
                  geom = 'line', colour = 'orange', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "freq", year = 1984, xmax = 50)$coef[1],2),
             x = 10, y = 3e-3, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "freq", year = 1992)$func,
                  geom = 'line', colour = 'green', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "freq", year = 1992)$coef[1],2),
             x = 10, y = 1e-3, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "freq", year = 2002)$func,
                  geom = 'line', colour = 'blue', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "freq", year = 2002)$coef[1],2),
             x = 10, y = 3e-4, geom = "text", size = 3) +
    stat_function(fun = function(x) lm_func(x, y = "freq", year = 2012)$func,
                  geom = 'line', colour = 'purple', linetype = "dashed") +
    annotate(label = round(lm_func(x = NA, y = "freq", year = 2012)$coef[1],2),
             x = 10, y = 1e-4, geom = "text", size = 3)
```

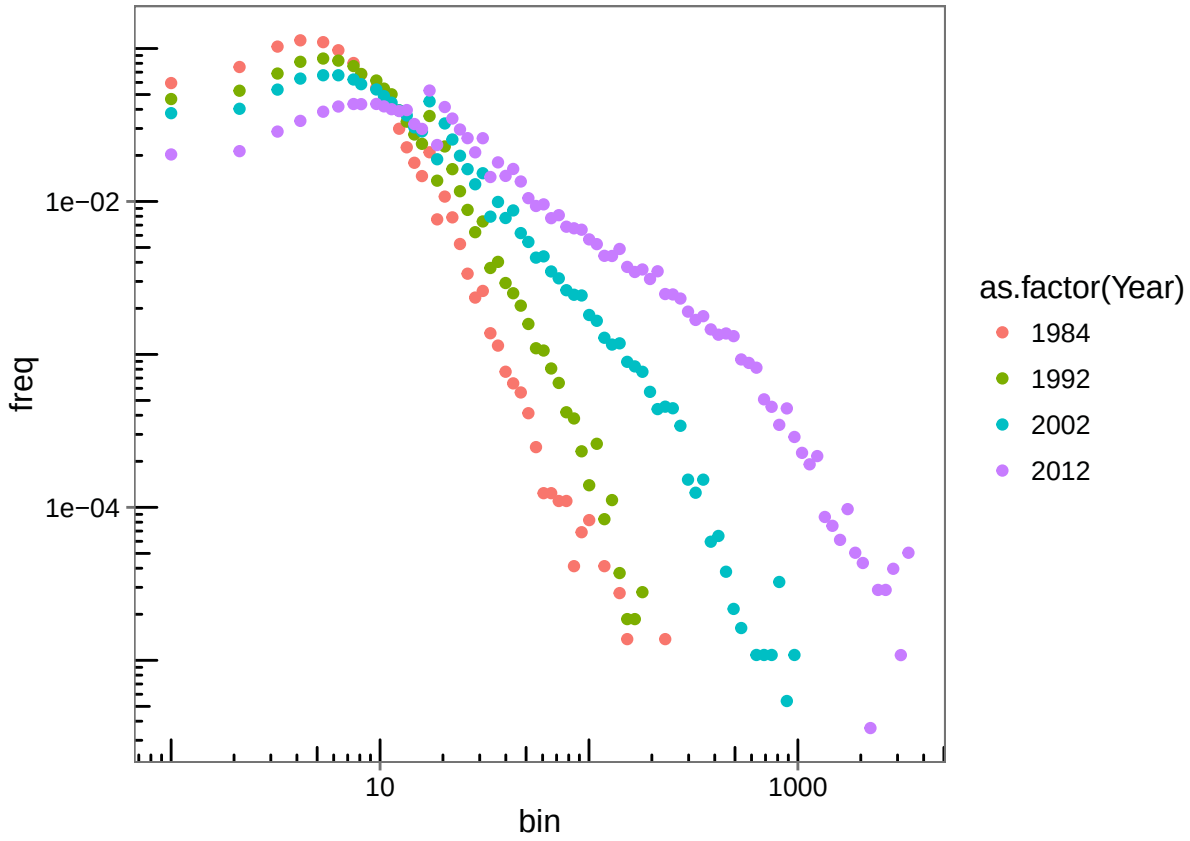
```
grid.arrange(gg2a, gg2a2, ncol = 2)
```



The above shows a raw plot as shown in valverde and a normalised density plot where the count has been divided by the total. We have judged the section of linear power law distribution by eye and fitted a linear model to get the gradient. In this case in the range of 20-200 with the exception of 1984 which uses 50 as its maxmimum order. We can see how a linear model fits well in this region and that values of gradient are higher than that produced by valverde, due to the presumed difference of including foreign patents.

Typically however heavy tailed plots like above are done using histograms to mitigate the effects of very rare events in the heavy tail. Below we create such a histogram, however there are artefacts due to the fact that order is discrete, as the bins change from containing one order to two etc.

```
test <- 1*10^seq(0,log10(4000), length.out = 100)
degree_distribution$bin <- cut(degree_distribution$Order,
                               breaks = c(0,test),
                               include.lowest = TRUE,
                               labels = test) %>% as.character() %>% as.numeric()
hist <- degree_distribution %>% group_by(Year, bin) %>% summarise(count = sum(count))
hist <- hist %>% group_by(Year) %>% mutate(group_total = sum(count), freq = count / group_total)

hist2 <-dplyr::filter(hist, Year %in% c(1984,1992,2002,2012))
(gg2a3 <- ggplot(data = hist2, aes(x = bin, y = freq, group = Year, colour = as.factor(Year))) +
    geom_point() +
    scale_y_log10() +
    scale_x_log10() +
    theme_few() +
    annotation_logticks()
)
```



Notes on plot2

- In the above histogram plot we can see an apparent "knee" in the linear fit, especially visible in the 2002, and 2012 years.
- We see the effect where average number of citations is increasing continue into 2012
- The big difference between our data and valverde's is that up to a peak of 7-9 the average citations is increasing whereas valerde' only decreases from a peak of 1 citation.
- Our data has lower maximum counts and higher maximum order (1e3 rather than 1e2)
  - This could indicate that valverde doesn't include **foreign citations** whereas we do.

```
## pdf
##   2
## pdf
##   2
## pdf
##   2
## pdf
##   2
## pdf
##   2
## pdf
##   2
## pdf
##   2
```