Course 3 - Superstars in Music, Sports, and Entertainment

UCLA - Econ 19 - Fall 2018

François Geerolf

Contents

Introduction

Rosen [1981] writes:

> Performers of first rank comprise a limited handful out of these small totals and have very large incomes. There are also known to be substantial differences in income between them and those in the second rank, even though most consumers would have difficulty detecting more than minor differences in a "blind" hearing.

What Sherwin Rosen says is that there are very few differences in talents at the very top.

> The elusive quality of "box office appeal," the ability to attract an audience and generate a large volume of transactions, is the issue that must be confronted. Recognition that one's personal market scale is important, in the theory of income distribution has a long history, but the idea has not been developed very extensively in the literature.
>
> Rest assured that prospective impresarios will receive no guidance here on what makes for box office appeal, sometimes said to involve a combination of talent and charisma in uncertain proportions. In the formal model all that is taken for granted and represented by a single factor rather than by two, an index q labeled talent or quality.
>
> Albert Rees is a good introduction to the size distribution of income. The selectivity effects of differential talent and comparative advantage on the skew in income distributions are spelled out in my 1978 article, also see the references there. Melvin Reder's survey touches some of the issues raised here.
>
> Of course social scientists and statisticians have had a long standing fascination with rank-size relationships, as perusal of the many entries in the Encyclopedia of the Social Sciences will attest.

1 Statistical Distributions for Superstars

We use the methods we saw in course 2 and plot the log rank on the y axis against the log of the outcome of interest (revenues, number of views, number of sales, etc.) We show that many of these distributions associated to superstar phenomena display a Pareto-like behavior in the tail: this means that there are very many observations which deviate substantially from the mean, and that earnings and success accrue disproportionately to the very top.

1.1 Most-downloaded songs in the United Kingdom

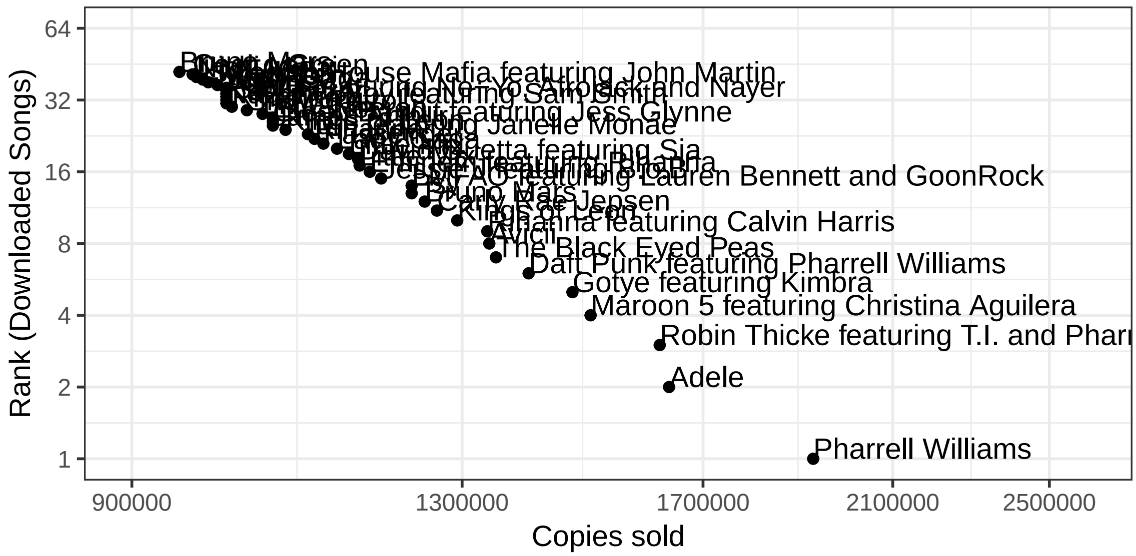
The data comes from the following Wikipedia entry: List of most-downloaded songs in the United Kingdom.

```
data <- "https://en.wikipedia.org/wiki/List_of_most-downloaded_songs_in_the_United_Kingdom" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

```
data[[2]][, c(1, 2, 3, 7)] %>%
  as.tibble %>%
  head(10)
```

```
## # A tibble: 10 x 4
##      No. Artist                        Song                 `Copies sold[a]`
##    <int> <chr>                         <chr>                <chr>
##  1     1 Pharrell Williams             "\"Happy\""          1,922,000[3]
##  2     2 Adele                         "\"Someone Like Yo~ 1,637,000+[4]
##  3     3 Robin Thicke featuring T.I.~ "\"Blurred Lines\"" 1,620,000+
##  4     4 Maroon 5 featuring Christin~ "\"Moves Like Jagg~ 1,500,000+
##  5     5 Gotye featuring Kimbra        "\"Somebody That I~ 1,470,000+
##  6     6 Daft Punk featuring Pharrel~ "\"Get Lucky\""      1,400,000+
##  7     7 The Black Eyed Peas           "\"I Gotta Feeling~ 1,350,000+
##  8     8 Avicii                        "\"Wake Me Up\""     1,340,000+
##  9     9 Rihanna featuring Calvin Ha~ "\"We Found Love\"" 1,337,000+
## 10    10 Kings of Leon                 "\"Sex on Fire\""    1,293,000+
```

```
data[[2]][, c(1, 2, 3, 7)] %>%
  as.tibble %>%
  select("Artist", value = "Copies sold[a]") %>%
  mutate(value = value %>% substr(1, 9) %>% gsub(",", "", .) %>% as.numeric) %>%
  arrange(-value) %>%
  mutate(rank = 1:n()) %>%
  ggplot(aes(x = value, y = rank)) + geom_point() + theme_bw() +
  geom_text(aes(label = Artist), hjust = 0, vjust = 0) +
  scale_y_log10(breaks = 2^(seq(0, 10, 1)),
                limits = c(1, 64)) +
  scale_x_log10(breaks = seq(500000, 2600000, 400000),
                limits = c(900000, 2600000)) +
  ylab("Rank (Downloaded Songs)") + xlab("Copies sold")
```



1.2 Most-streamed songs on Spotify

The data comes from the following Wikipedia entry: List of most-streamed songs on Spotify.

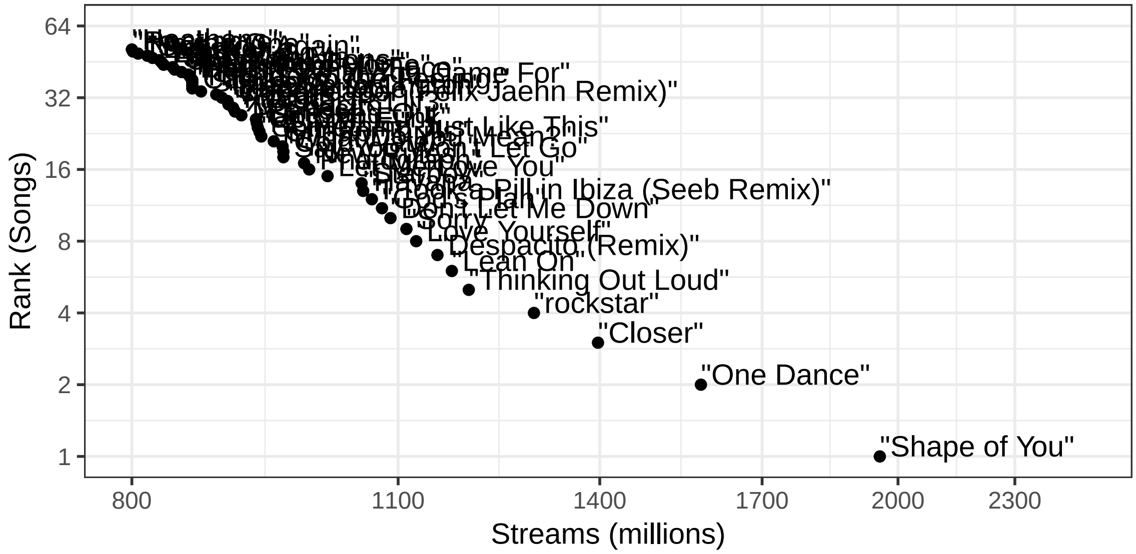
```
data <- "https://en.wikipedia.org/wiki/List_of_most-streamed_songs_on_Spotify" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

```
data[[1]][, c(1, 2, 5, 6)] %>%
  rename(song = "Song name", value = "Streams(millions)", date = "Date published") %>%
  as.tibble %>%
  head(10)
```

```
## # A tibble: 10 x 4
##    Rank  song                        value date
##    <chr> <chr>                       <chr> <chr>
##  1 1.    "\"Shape of You\""          1,957 6 January, 2017
##  2 2.    "\"One Dance\""             1,580 5 April, 2016
##  3 3.    "\"Closer\""                1,397 29 July, 2016
##  4 4.    "\"rockstar\""              1,294 15 September, 2017
##  5 5.    "\"Thinking Out Loud\""     1,197 21 June, 2014
##  6 6.    "\"Lean On\""               1,173 2 March, 2015
##  7 7.    "\"Despacito (Remix)\""     1,153 17 April, 2017
##  8 8.    "\"Love Yourself\""         1,124 9 November, 2015
##  9 9.    "\"Sorry\""                 1,111 23 October, 2015
## 10 10.   "\"Don't Let Me Down\""     1,090 5 February, 2016
```

```
data[[1]] %>%
  as.tibble %>%
  select(song = "Song name", value = "Streams(millions)") %>%
  mutate(value = value %>% substr(1, 9) %>% gsub(",", "", .) %>% as.numeric) %>%
  arrange(-value) %>%
  mutate(rank = 1:n()) %>%
  ggplot(aes(x = value, y = rank, label = song)) + geom_point() + theme_bw() +
  geom_text(aes(label = song), hjust = 0, vjust = 0) +
  scale_y_log10(breaks = 2^(seq(0, 10, 1)),
```

```r
                limits = c(1, 64)) +
  scale_x_log10(breaks = seq(800, 2500, 300),
                limits = c(800, 2500)) +
  ylab("Rank (Songs)") + xlab("Streams (millions)")
```



1.3 Most-viewed YouTube videos

The data comes from the following Wikipedia entry: List of most-viewed YouTube videos.

```r
data <- "https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

We then display the first rows of the data:

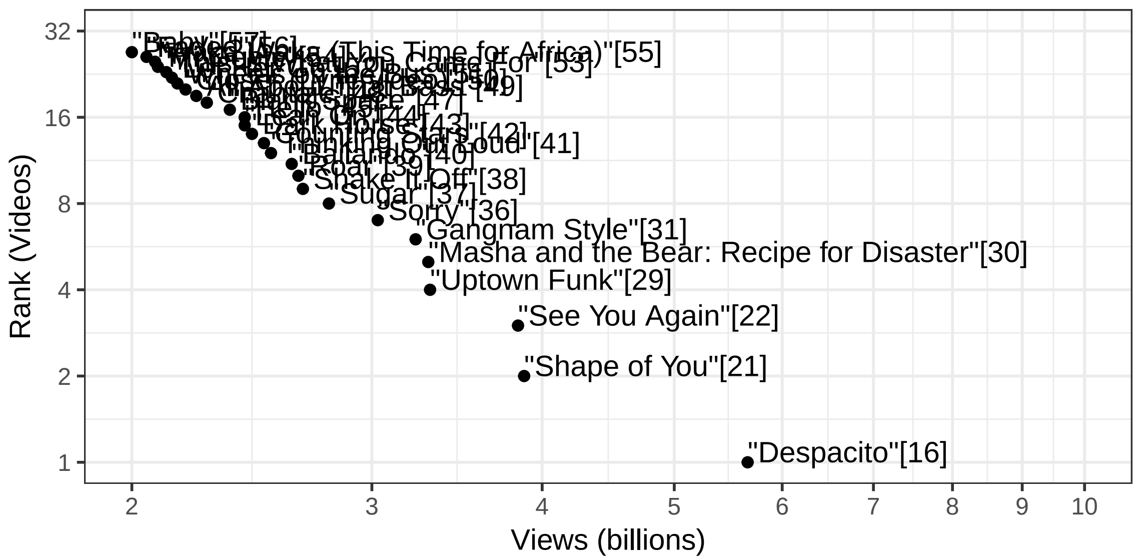
```r
data[[1]][, c(1, 2, 4)] %>%
  as.tibble %>%
  rename(video = "Video name", value = "Views (billions)") %>%
  head(10)
```

```
## # A tibble: 10 x 3
##    `#`   video                                              value
##    <chr> <chr>                                              <chr>
##  1 1.    "\"Despacito\"[16]"                                5.66
##  2 2.    "\"Shape of You\"[21]"                             3.88
##  3 3.    "\"See You Again\"[22]"                            3.84
##  4 4.    "\"Uptown Funk\"[29]"                              3.31
##  5 5.    "\"Masha and the Bear: Recipe for Disaster\"[30]" 3.30
##  6 6.    "\"Gangnam Style\"[31]"                            3.23
##  7 7.    "\"Sorry\"[36]"                                    3.03
##  8 8.    "\"Sugar\"[37]"                                    2.79
##  9 9.    "\"Shake It Off\"[38]"                             2.67
## 10 10.   "\"Roar\"[39]"                                     2.65
```

These videos are available on Youtube:
- Luis Fonsi - Despacito ft. Daddy Yankee
- Ed Sheeran - Shape of You

- Wiz Khalifa - See You Again ft. Charlie Puth
- etc.

```
data[[1]][, c(1, 2, 4)] %>%
  as.tibble %>%
  select(video = "Video name", value = "Views (billions)") %>%
  mutate(value = value %>% substr(1, 9) %>% gsub(",", "", .) %>% as.numeric) %>%
  arrange(-value) %>%
  mutate(rank = 1:n()) %>%
  ggplot(aes(x = value, y = rank)) + geom_point() + theme_bw() +
  geom_text(aes(label = video), hjust = 0, vjust = 0) +
  scale_y_log10(breaks = 2^(seq(0, 10, 1)),
                limits = c(1, 32)) +
  scale_x_log10(breaks = seq(1, 10, 1),
                limits = c(2, 10)) +
  ylab("Rank (Videos)") + xlab("Views (billions)")
```



1.4 Highest paid American television stars

The data comes from the following Wikipedia entry: List of highest paid American television stars.

Network primetime salaries per episode

```
data <- "https://en.wikipedia.org/wiki/List_of_highest_paid_American_television_stars" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

We then display the first rows of the data:

```
data[[1]][, c(1, 2, 3, 4)] %>%
  as.tibble %>%
  head(10)
```

```
## # A tibble: 10 x 4
##    Name                 Program               Role             Salary
##    <chr>                <chr>                 <chr>            <chr>
##  1 Peter Dinklage       Game of Thrones       Tyrion Lannister £2,000,000[~
##  2 Nikolaj Coster-Wal~ Game of Thrones       Jaime Lannister  £2,000,000[~
```

```
##   3 Lena Headey        Game of Thrones       Cersei Lannister £2,000,000[~
##   4 Emilia Clarke      Game of Thrones       Daenerys Targar~ £2,000,000[~
##   5 Kit Harington      Game of Thrones       Jon Snow         £2,000,000[~
##   6 Charlie Sheen      Two and a Half Men    Charlie Harper   $1.8 million
##   7 Ray Romano         Everybody Loves Raym~ Raymond Barone   $1.7 million
##   8 Kelsey Grammer     Frasier               Frasier Crane    $1.6 million
##   9 Tim Allen          Home Improvement      Tim Taylor       $1.25 milli~
##  10 James Gandolfini   The Sopranos          Tony Soprano     $1 million
```

1.5 Highest grossing concert tours

The data comes from the following Wikipedia entry: List of highest-grossing concert tours. We first input the data from the Wikipedia page, using the `rvest` package to extract tables from the html source code:

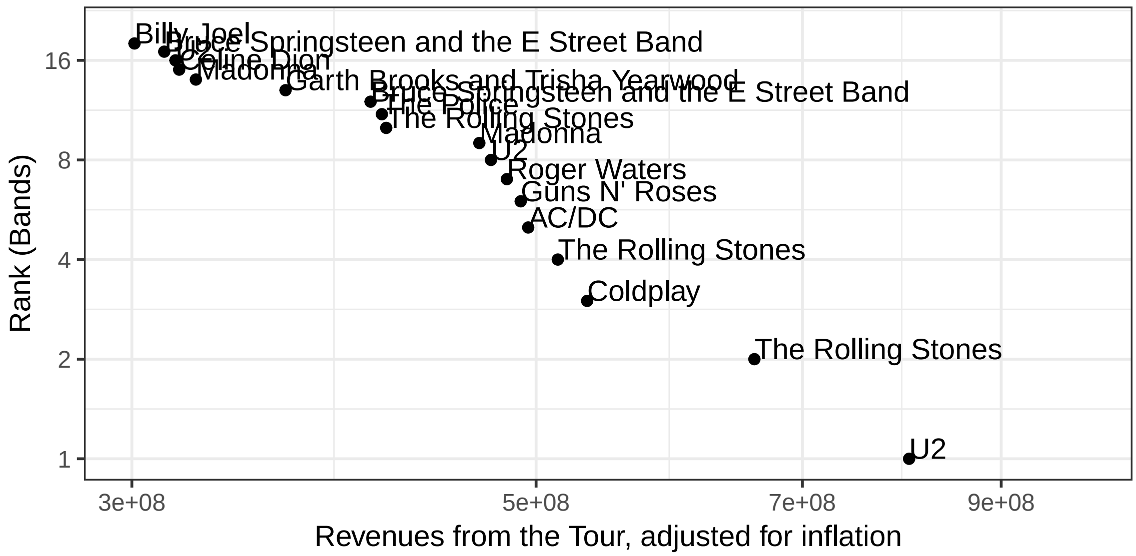
```
data <- "https://en.wikipedia.org/wiki/List_of_highest-grossing_concert_tours" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

We then display the first rows of the data:

```
data[[1]][, c(1, 2, 3, 4)] %>%
  rename(Gross = "Gross adjusted for inflation(2018 $)") %>%
  as.tibble %>%
  head(10)
```

```
## # A tibble: 10 x 4
##     Rank `Actual gross` Gross        Artist
##    <int> <chr>          <chr>        <chr>
##  1     1 $736,421,584   $801,130,818 U2
##  2     2 $558,255,524   $658,868,741 The Rolling Stones
##  3     3 $523,033,675   $533,331,898 Coldplay
##  4     4 $480,900,000   $490,368,636 Guns N' Roses
##  5     5 $458,673,798   $481,869,587 Roger Waters
##  6     6 $441,121,000   $495,041,025 AC/DC
##  7     7 $408,000,000   $465,399,721 Madonna
##  8     8 $389,047,636   $472,277,371 U2
##  9     9 $364,300,000   $364,300,000 Garth Brooks and Trisha Yearwood
## 10    10 $362,000,000   $411,460,278 The Police
```

```
data[[1]][, c(1, 2, 3, 4)] %>%
  as.tibble %>%
  rename(gross = "Gross adjusted for inflation(2018 $)") %>%
  mutate(gross = gross %>% substr(2, 13) %>% gsub(",", "", .) %>% as.numeric) %>%
  arrange(-gross) %>%
  mutate(rank = 1:n()) %>%
  ggplot(aes(x = gross, y = rank, label = Artist)) + geom_point() + theme_bw() +
  geom_text(aes(label = Artist), hjust = 0, vjust = 0) +
  scale_y_log10(breaks = 2^(seq(0, 10, 1))) +
  scale_x_log10(breaks = 10^8*seq(1, 10, 2),
                limits= c(3*10^8, 10^9)) +
  ylab("Rank (Bands)") + xlab("Revenues from the Tour, adjusted for inflation")
```



1.6 Highest grossing concert tours

The data comes from the following Wikipedia entry: List of highest paid film actors. We first input the data from the Wikipedia page, using the `rvest` package to extract tables from the html source code:

```
data <- "https://en.wikipedia.org/wiki/List_of_highest_paid_film_actors" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

We then display the first rows of the data:

```
data[[1]] %>%
  select(-Ref) %>%
  as.tibble %>%
  head(15)
```

```
## # A tibble: 15 x 5
##    Actor         Film                          Year Salary   `Total income`
##    <chr>         <chr>                        <int> <chr>    <chr>
##  1 Keanu Reeves  The Matrix ReloadedThe Matr~  2003 $30,000~ $156,000,000
##  2 Bruce Willis  The Sixth Sense               1999 $14,000~ $100,000,000
##  3 Tom Cruise    Mission: Impossible 2         2000 ""       $100,000,000
##  4 Tom Cruise    War of the Worlds             2005 ""       $100,000,000
##  5 Will Smith    Men in Black 3                2012 ""       $100,000,000
##  6 Robert Down~  Iron Man 3                    2013 ""       $75,000,000
##  7 Sandra Bull~  Gravity                       2013 $20,000~ $70,000,000+
##  8 Tom Hanks     Forrest Gump                  1994 ""       $70,000,000
##  9 Tom Cruise    Mission: Impossible           1996 ""       $70,000,000
## 10 Harrison Fo~  Indiana Jones and the Kingd~  2008 ""       $65,000,000
## 11 Jack Nichol~  Batman                        1989 $6,000,~ $60,000,000
## 12 Leonardo Di~  Inception                     2010 ""       $59,000,000
## 13 Robert Down~  Captain America: Civil War    2014 $40,000~ $40,000,000+
## 14 Robert Down~  Avengers: Age of Ultron       2014 ""       $40,000,000
## 15 Johnny Depp   Pirates of the Caribbean: O~  2011 $35,000~ $55,000,000
```

1.7 Largest sports contracts

The data comes from the following Wikipedia entry: List of largest sports contracts.

```
data <- "https://en.wikipedia.org/wiki/List_of_largest_sports_contracts" %>%
  read_html %>%
  html_table(header = TRUE, fill = TRUE)
```

We then display the first rows of the data:

```
data[[1]] %>%
  select(Player, Sport, length = "Length of contract" , value = "Contract value (USD)") %>%
  as.tibble %>%
  head(10)
```

```
## # A tibble: 10 x 4
##    Player             Sport      length               value
##    <chr>              <chr>      <chr>                <chr>
##  1 Canelo Álvarez     Boxing     5 years (2018-2023)  $365,000,000
##  2 Giancarlo Stanton Baseball   13 years (2014-2027) $325,000,000
##  3 Alex Rodriguez1R   Baseball   10 years (2008-2017) $275,000,000
##  4 Alex Rodriguez2R   Baseball   10 years (2001-2010) $252,000,000
##  5 Miguel Cabrera     Baseball   8 years (2016-2023)  $247,000,000
##  6 Robinson Cano      Baseball   10 years (2014-2023) $240,000,000
##  7 Albert Pujols      Baseball   10 years (2012-2021) $240,000,000
##  8 James Harden       Basketball 6 years (2017-2023)  $228,000,000
##  9 Joey Votto         Baseball   10 years (2014-2024) $225,000,000
## 10 David Price        Baseball   7 years (2016-2022)  $217,000,000
```

2 Other examples

2.1 Salaries at the University of California (UC)

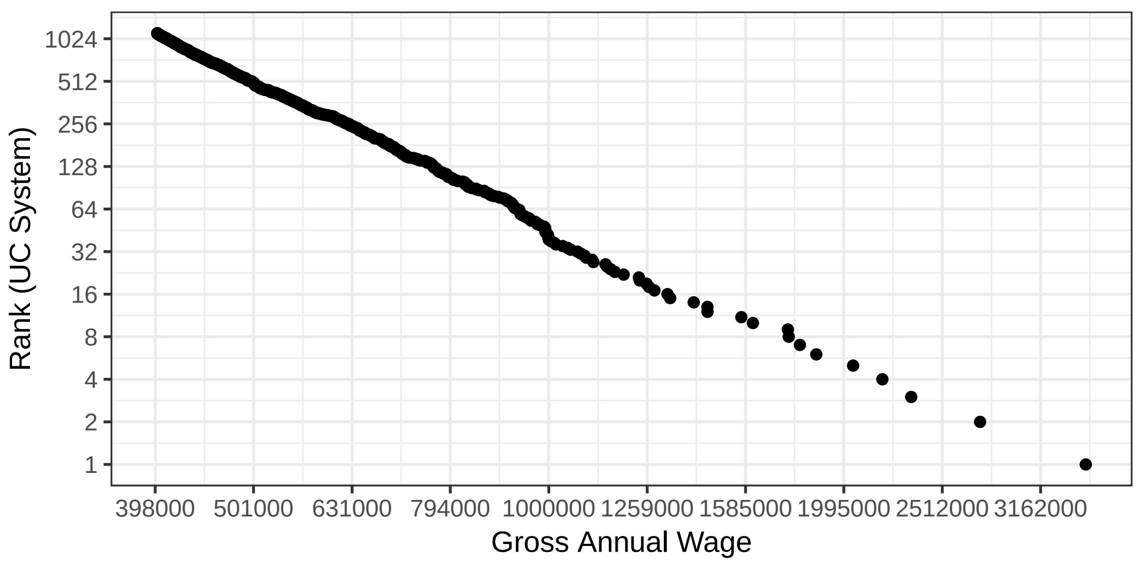
The salaries at the University of California are public and available at this website: https://ucannualwage.ucop.edu/wage/

```
ucop.2015 <- read.csv("https://raw.githubusercontent.com/raleighlittles/UC-Employee-Salaries/master/UCO
  # use clean_names() from janitor package
  clean_names() %>%
  mutate_all(. %>% paste %>% gsub("'", "", .)) %>%
  mutate_at(vars(ends_with("pay")), funs(as.numeric)) %>%
  arrange(-x_gross_pay) %>%
  mutate(rank = 1:n())
```

The distribution in the tail at the University of California is really well approximated by a Pareto distribution. Below is the plot of wages higher than $400K.

```
ucop.2015 %>%
  rename(gross = x_gross_pay, name = x_last_name) %>%
  filter(gross >= 400000) %>%
  ggplot(aes(x = gross, y = rank, label = name)) + geom_point() + theme_bw() +
  scale_y_log10(breaks = 2^(seq(0, 10, 1))) +
  scale_x_log10(breaks = 2^(seq(10, 100, 1))) +
  ylab("Rank (UC System)") + xlab("Gross Annual Wage")
```

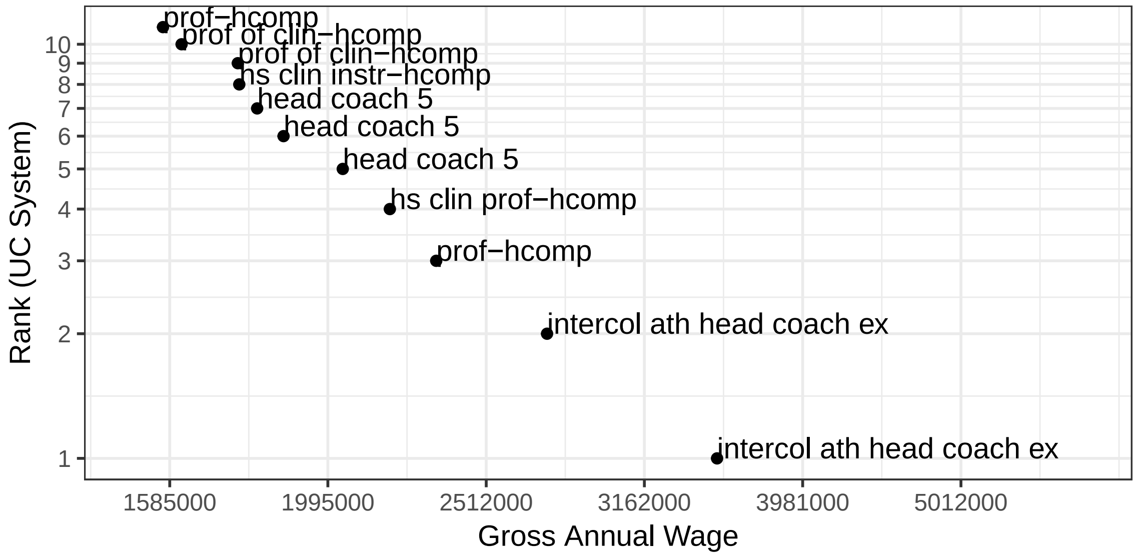

```
ucop.2015 %>%
  rename(gross = x_gross_pay, name = x_last_name) %>%
  filter(gross >= 400000) %>%
  ggplot(aes(x = gross, y = rank, label = name)) + geom_point() + theme_bw() +
  scale_y_log10(breaks = 2^(seq(0, 10, 1))) +
  scale_x_log10(breaks = round(10^(seq(5, 7, 0.1))/1000)*1000) +
  ylab("Rank (UC System)") + xlab("Gross Annual Wage")
```



Below is a zoom on the distribution of wages higher than 1.5 million annual. You can see that the highest paid superstars on campus are the Head Coaches, and superstar physicians. Again, it is striking that these distributions are very well approximated by a Pareto distribution.

```
ucop.2015 %>%
  rename(gross = x_gross_pay, name = x_title) %>%
  mutate(name = tolower(name)) %>%
  filter(gross >= 1500000) %>%
  ggplot(aes(x = gross, y = rank, label = name)) + geom_point() + theme_bw() +
  geom_text(aes(label = name), hjust = 0, vjust = 0) +
  scale_y_log10(breaks = seq(1, 10, 1)) +
```

```
scale_x_log10(breaks = round(10^(seq(5, 6.7, 0.1))/1000)*1000,
              limits= c(1500000, 6000000)) +
ylab("Rank (UC System)") + xlab("Gross Annual Wage")
```



2.2 Market for Executive Officers in large firms

Again, from Sherwin Rosen:

> Such considerations are important for understanding the market for executive officers in large firms. Unusually good information on executive compensation is available from public proxy statements circulated to stockholders by requirement of the Securities and Exchange Commission. Examination of these statements is instructive. They reveal Superstar-scale rewards that are highly concentrated among the top half-dozen executives in these firms. More detailed study indicates that the top incomes vary systematically with the size of the organization. Large firms pay executives more than smaller firms do. Even the occasional, well-publicized dollar-a-year man falls in line once stock options, pensions, and other forms of deferred compensation are properly accounted. The value to the organization of good top-level decisions and avoidance of bad decisions is abundantly clear once the nature of control of resources on such a vast scale is considered.
>
> Common use of the term Officer for corporate executives Suggests certain parallels with the military. A good or bad decision by a platoon leader does not have much effect on the overall fortunes of war, but the same cannot be said of decisions made by the chief strategists. The value of extra talent is much larger at the top of the organizational hierarchy than at the bottom because those decisions percolate through the enterprise, and they have much further to travel in a larger enterprise than in a smaller one.

References

Sherwin Rosen. The Economics of Superstars. *The American Economic Review*, 71(5):845–858, 1981. ISSN 0002-8282. URL http://www.jstor.org/stable/1803469.