AMSI
VACATION
RESEARCH
SCHOLARSHIPS
2018-2019

Do the rich get richer on Reddit?

John Davey

Supervised by Dr Lewis Mitchell

The University of Adelaide

Vacation Research Scholarships are funded jointly by the Department of Education and Training and the Australian Mathematical Sciences Institute.

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Australian Government
Department of Education and Training

Abstract

The Matthew Effect is the widely observed and well documented phenomenon, often colloquially known by the adage “the rich get richer”. Previous research has identified the Matthew Effect in a wide range of contexts, including scientific collaboration networks, the internet, the making of new friends, population size, city growth and the most common English words and phrases, among others. This project established strong evidence to suggest the existence of the Matthew Effect in the scores of submissions and comments on the website Reddit, via analysis of data from the Reddit API and other massive data dumps.

1 Introduction

Our main goal in this project will be to establish the existence of the Matthew Effect in the scores of Reddit submissions and comments. Before we can achieve this, however, we must discuss both the Matthew Effect and Reddit.

1.1 The Matthew Effect / Preferential Attachment

“The rich get richer and the poor get poorer” is a common phrase used to describe the phenomenon where the ease of gaining more of a certain quantity grows as the current size of the quantity increases. In mathematics, this idea is captured by the Matthew Effect (or, in network science: Preferential Attachment).

To establish the existence of the Matthew Effect in some quantity, two considerations need to be made. First, if x is the quantity of interest, then we need the distribution of x to follow a power-law (or at least a similar heavy-tailed distribution), i.e. the probability density function (pdf) of x, $p(x)$, satisfies

$$p(x) \propto x^{-\beta}$$

for $\beta \in \mathbb{R}_{>0}$. Provided that this condition holds, we will also need to show that if y is an individual realisation of the quantity of interest, then the rate of change in y is proportional to the size of y, i.e.

$$\frac{\Delta y}{\Delta t} = A(t)y^{\gamma}$$

for some constant $\gamma \in \mathbb{R}_{>0}$ and some time dependent factor $A(t)$ [3]. Here $\frac{\Delta y}{\Delta t}$ is an approximate rate of change in y taken over a suitably small time step Δt.

The value of γ is used to classify the nature of the preferential attachment present in the quantity. If $\gamma = 1$ then we say that the quantity displays *linear* preferential attachment, whereas $\gamma < 1$ indicates *sublinear* preferential attachment (which is largely similar to linear preferential attachment), and $\gamma > 1$ suggests *superlinear* preferential attachment (which favours the growth of 'super-hubs' in the quantity) [3].

1.1.1 More information on power-laws

We will also briefly discuss two points about power-laws which will be useful later in the report. First, recall that power-laws are distributions of the form

$$p(x) \propto x^{-\beta}$$

for $\beta \in \mathbb{R}_{>0}$. This means that power-laws are undefined at $x = 0$. Hence, it is usual to set a value $x_{\min} > 0$ after which we will fit a power-law to experimental data, to ensure that the distribution is bounded.

The form of the power-law distribution also means that if

$$p(x) = Bx^{-\beta}$$

for $x \in \mathbb{R}_{>x_{\min}}$, $B \in \mathbb{R}_{>0}$, then

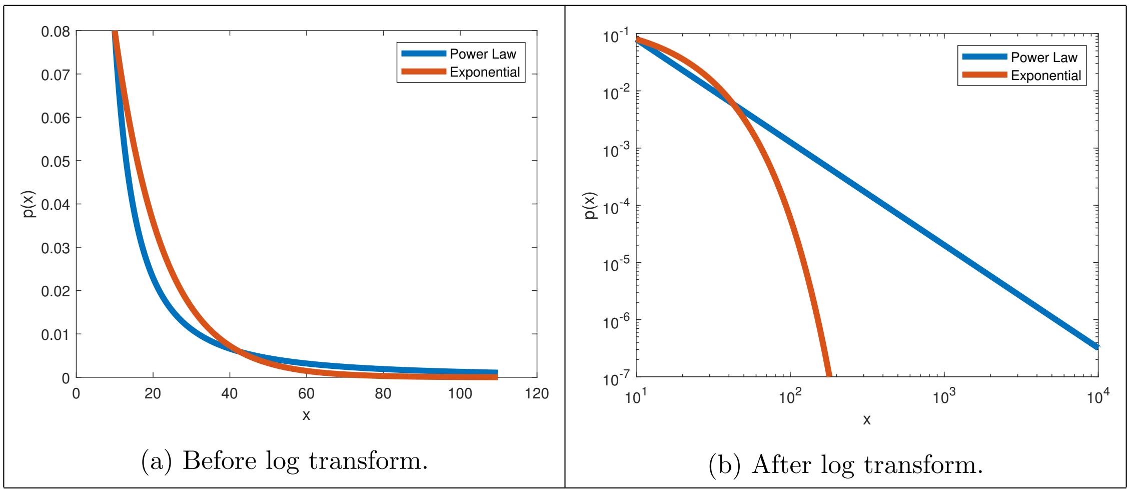
$$\log_{10}(p(x)) = \log_{10}(B) - \beta \log_{10}(x). \tag{1}$$

Equation 1 implies that on a log-log scale, power-law distributions should appear linear and decreasing. Indeed, Figure 1 shows that before taking a log transform, it can be hard to distinguish between two visually similar distributions (one being a power-law and one being an exponential). After the transform, however, it is immediately clear that the power-law is the blue distribution which is linear and decreasing, whereas the exponential distribution is still non-linear and decreases much more quickly.

1.2 General information about Reddit

Reddit[1] is a website where users can submit photos, links or plain text to themed communities called subreddits. Other users can rate these submissions (a.k.a. posts) with either an upvote or a downvote.

[1] www.reddit.com



(a) Before log transform.

(b) After log transform.

Figure 1: Plots of the pdfs of a power-law distribution and an exponential distribution before and after taking a log transform.

They can also choose to comment on posts, and these comments can also be upvoted or downvoted.

The score of a piece of Reddit content (a post/comment) is displayed next to the content, and is defined as the piece of content's number of upvotes minus its number of downvotes. Aside from being displayed alongside content, Reddit also uses the number of upvotes and downvotes of each piece of content to determine where amongst the other content it should be displayed by using internal ranking algorithms. A short description of the two default ranking algorithms for posts and comments follows[2].

1.2.1 Default post sorting algorithm

Reddit's default post sorting algorithm is called 'Hot'. It is defined in the following way: consider a Reddit post with u upvotes and d downvotes, that was posted at time t_p (note that all times here are in UNIX time: the number of seconds since midnight on January 1st, 1970 UTC). Then the post's score, s, will be $s = u - d$.

We define t_s as $t_s := t_p - 1134028003$. Here, 1134028003 is a reference date (07:46:43 on 8/12/2005 UTC). This is likely the exact date and time at which the Hot algorithm was first launched. This means that t_s is the difference (in seconds) between the post's submission time and the reference time in 2005.

[2] Reddit's code is open source, so the sorting algorithms can be viewed directly via the relevant section in their Github repository.

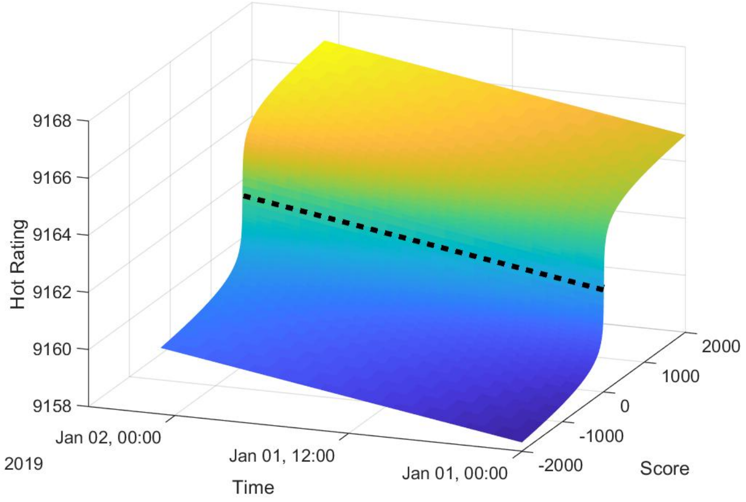


Figure 2: A surface plot of Reddit's 'Hot' algorithm. The score ranges from -2000 to 2000, while time ranges from midnight on January 1st 2019 to midnight on January 2nd 2019.

Finally, we define $\tilde{s}$ as $\tilde{s} := \max\{1, |s|\}$. The post's Hot rating, h, is then

$$h = \operatorname{sign}(s) \cdot \log_{10}(\tilde{s}) + t_s/45000. \tag{2}$$

Figure 2 is a surface plot of the Hot rating versus time and score. Notice that the Hot rating increases linearly for more recent posts. This means that more recent posts are favored over older posts. Also, the dependence on score is logarithmic in base 10, so, for example, the first 10 upvotes have the same effect as the next 100.

1.2.2 Default comment sorting algorithm

The default comment sorting algorithm is called 'Best'. It is the lower bound of the 95 % Wilson score interval [2] for the true proportion of upvotes to total votes. To define it precisely, let $z_{0.95}$ be the value such that $\mathbb{P}(X < z_{0.95}) = 0.95$ where $X \sim N(0,1)$. Then for a comment with u upvotes and d downvotes, the comment's 'Best' rating b is

$$b = \frac{\hat{p} + \frac{z_{0.95}^2}{2(u+d)}}{1 + \frac{z_{0.95}^2}{u+d}} - \frac{z_{0.95}}{1 + \frac{z_{0.95}^2}{u+d}} \sqrt{\frac{\hat{p}(1-\hat{p})}{u+d} + \frac{z_{0.95}^2}{4(u+d)^2}}. \tag{3}$$

Figure 3 shows the 95% Wilson score intervals for two hypothetical comments. While Comment 1 has a higher sample proportion of upvotes to total votes than Comment 2 (marked by the blue crosses on the plot), since Comment 2 has many more total votes, the lower bound for its Wilson score interval

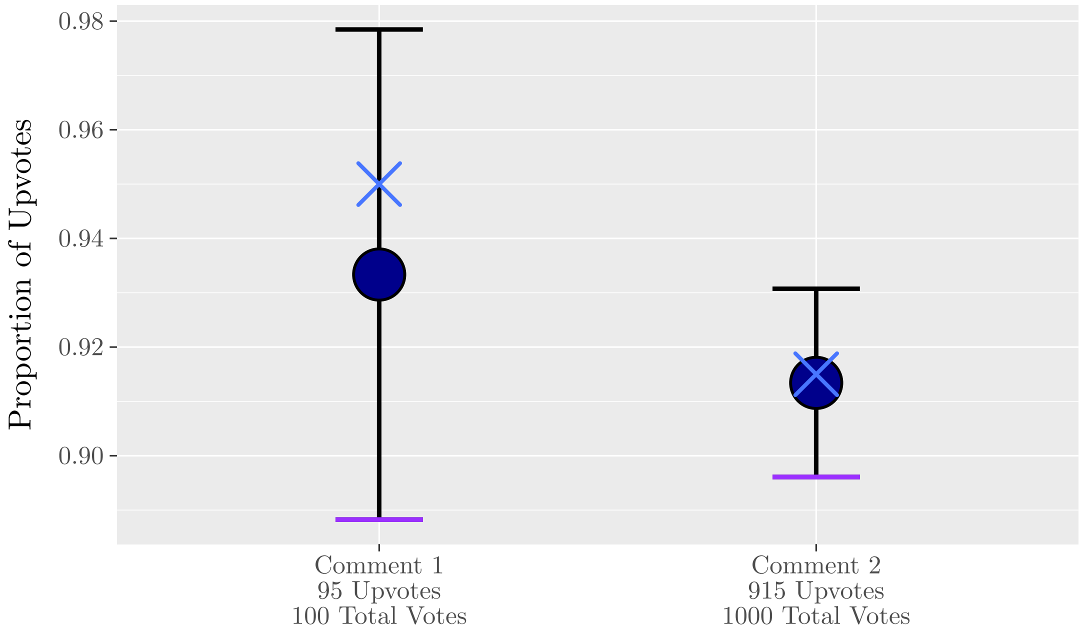


Figure 3: Comparing the 95% Wilson score intervals for two hypothetical comments. The blue crosses mark the sample upvote proportions, while the purple lines mark the two comment's 'Best' ratings. Comment 2 would be ranked higher than Comment 1 even though Comment 1 had a higher sample upvote proportion.

(marked by the purple horizontal lines) is higher and hence Comment 2 has a higher 'Best' rating.

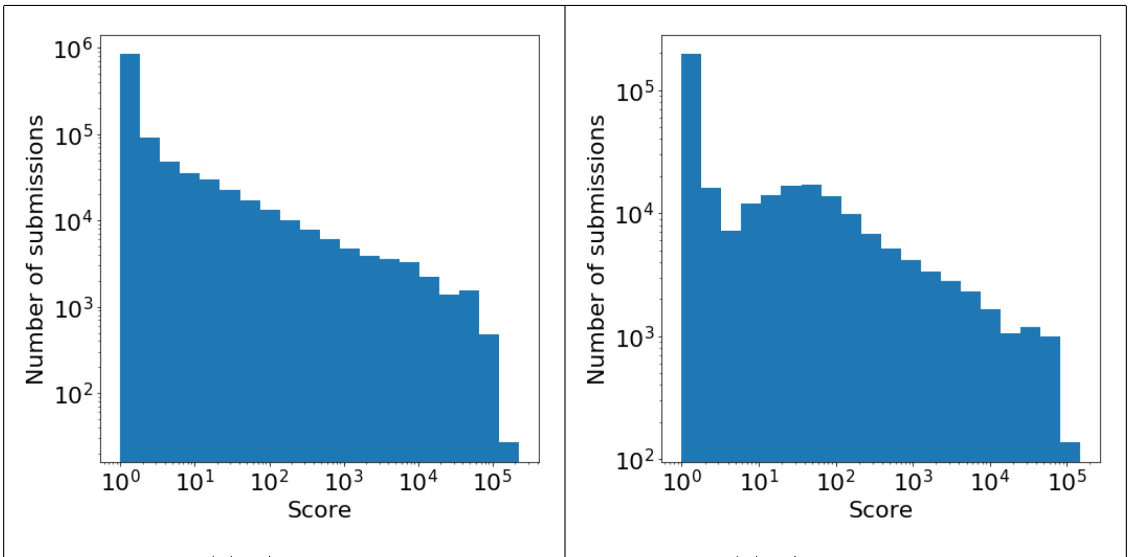
It is also worth noting that the default comment ranking algorithm has no direct time dependence. However, since comments are nested within posts, the time dependence in the default post ranking algorithm likely affects comments too.

2 Score distributions of Reddit comments and submissions

To establish the existence of the Matthew Effect in the scores of Reddit posts or comments, recall that the first step is to show that the scores follow a power-law distribution (or related heavy-tailed distribution). Since Reddit makes all of its submission and comment data freely available to the public, large collections of historical Reddit data have been created. We will use the website 'pushshift.io'[3], which is one example of a massive[4] Reddit data dump.

[3] https://pushshift.io/

[4] For an idea of the scale of this data set, the file containing Reddit content from December 2018 (which can be found here) is over 12.5 GB!



(a) r/gaming.

(b) r/todayilearned.

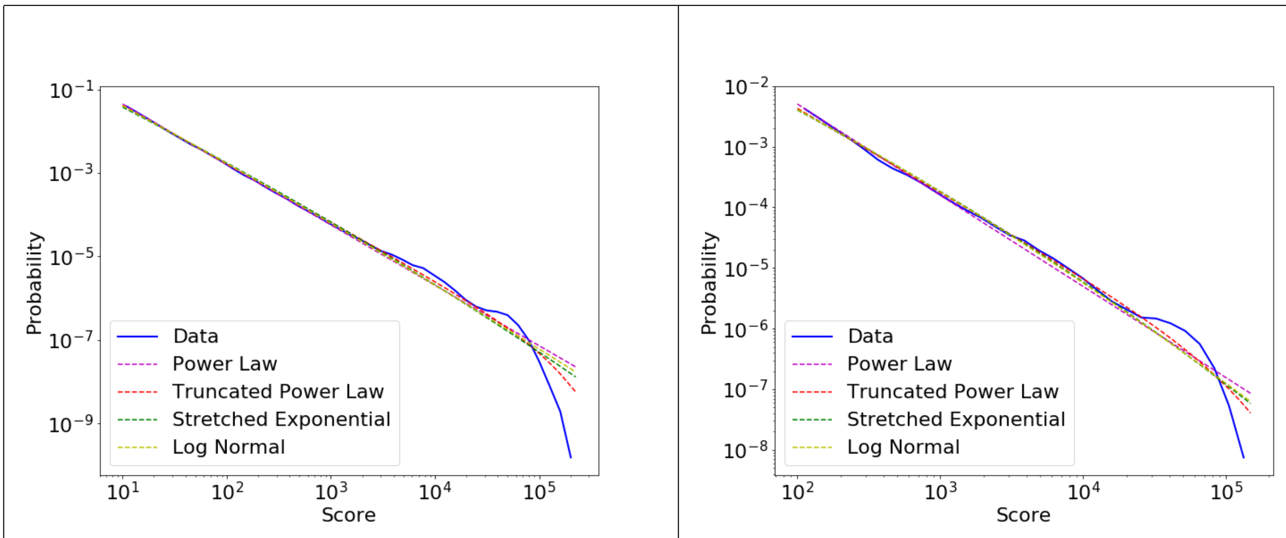
Figure 4: Log-log histograms of the scores of posts to r/gaming and r/todayilearned.

We can use the Python package `psaw`[5] to access the data from 'pushshift.io' within Python. `psaw` allows us to access the metadata of all of the comments and posts to any subreddit over a user specified length of time. We therefore write Python code to generate our own data sets of Reddit score data for a sample of subreddits. After cleaning this data, we can produce log-log histograms of the submissions and comment data for different subreddits. Note that in this case, cleaning the data mostly consisted of removing unnecessary data about each piece of content, because pushshift data contains extra information like the content's post time and the type of content (image/link/plaintext).

Figure 4 shows log-log histograms of the scores of posts to r/gaming and r/todayilearned, respectively. Both histograms show a very strong drop off in the tail, which is likely due to the time dependence in the post ranking formula (Equation 2). This formula causes older posts to drop in rank and hence be less visible to users and therefore less likely to be upvoted.

The decreasing linear trend that we are hoping to observe is reasonably strong after the first bin for the posts from r/gaming, however the posts from r/todayilearned have a much more interesting distribution for the first two decades, before it starts to follow the decreasing linear trend (excluding the drop in the tail). Note that while we examined several extra subreddits, only r/gaming and

[5] `psaw`'s documentation can be found here.



(a) Fitted distributions for r/gaming posts.

(b) Fitted distributions for r/todayilearned posts.

Figure 5: Comparing the fitted distributions for the r/gaming and r/todayilearned post data.

r/todayilearned have been included in this report for illustration purposes.

We can use the Python package `powerlaw`[6] to fit a number of potential distributions to a data set that we suspect might follow a power-law distribution (the full list of candidate distributions which are fitted is: power-law, truncated power-law, exponential, stretched exponential and log-normal[7]). The package uses the method of maximum likelihood to fit these distributions to the data set (more information about this method can be found in [1]), which can then be plotted to produce images like those in Figure 5. Note that while there are methods to estimate the best possible value of x_{min}, for simplicity we will instead choose our x_{min} value manually based on plots of the data.

Visually, the fit for all of the candidate distributions appears to be quite good, except in the tail where none are able to fully capture the dramatic drop. However, visually assessing which distribution best fits the data is highly subjective. The power-law package is also able to perform log-likelihood ratio tests to more rigorously determine which of the candidate distributions fits best.

We will briefly outline the idea of this test. Informally, the hypotheses for this type of test are as

[6] `powerlaw`'s documentation can be found here.

[7] See Appendix A for a definition of each of these distributions.

follows.

H_0 : The data is more likely to follow the null distribution.

versus

H_a : The data is more likely to follow the alternative distribution.

If the observed log-likelihood ratio is positive, this indicates that the null distribution is a better fit than the alternative distribution, and conversely a negative log-likelihood ratio implies that the alternative distribution is a better fit than the null distribution. Underlying statistics allow us to determine the theoretical distribution of the log-likelihood ratio, and hence allow us to construct a p-value that we can use to retain or reject the null hypothesis. More details on this type of testing can be found in [1].

Table 2 summarises the application of the log-likelihood ratio test to the r/gaming post data set, with the fitted power-law as the null distribution. It is therefore clear that at the 95% level, the truncated power-law and log-normal distributions are significantly better than the power-law, while there is insufficient evidence to determine whether the power-law is a better fit than the stretched exponential.

We can then run a second set of tests, this time using the truncated power-law as the null distribution. The results of these further tests are summarised in Table 2. This suggests that the truncated power-law is significantly better than both the stretched exponential and log normal distributions at the 95% level. Hence, of the 5 candidate distributions which were examined, the truncated power-law is the best fit.

A truncated power-law is the distribution obtained when you multiply the kernel of a power-law by the kernel of an exponential distribution, resulting in a distribution of the form

$$p(x) \propto x^{-\beta} e^{-\lambda x}$$

for $\beta, \lambda \in \mathbb{R}_{>0}$. They behave similarly to a normal power-law except for very large values of x, where they drop faster than a power-law. Nevertheless, the close relationship between power-laws and truncated power-laws places them in the 'related heavy-tailed distribution' category, so observing their presence should be enough to satisfy to first condition of the Matthew Effect.

Distribution	Truncated Power Law	Exponential	Stretched Exponential	Log Normal
Log Likelihood Ratio	-125.43	35815.85	9.99	-70.63
Associated p value	0.0	0.0	0.67	1.00e-12

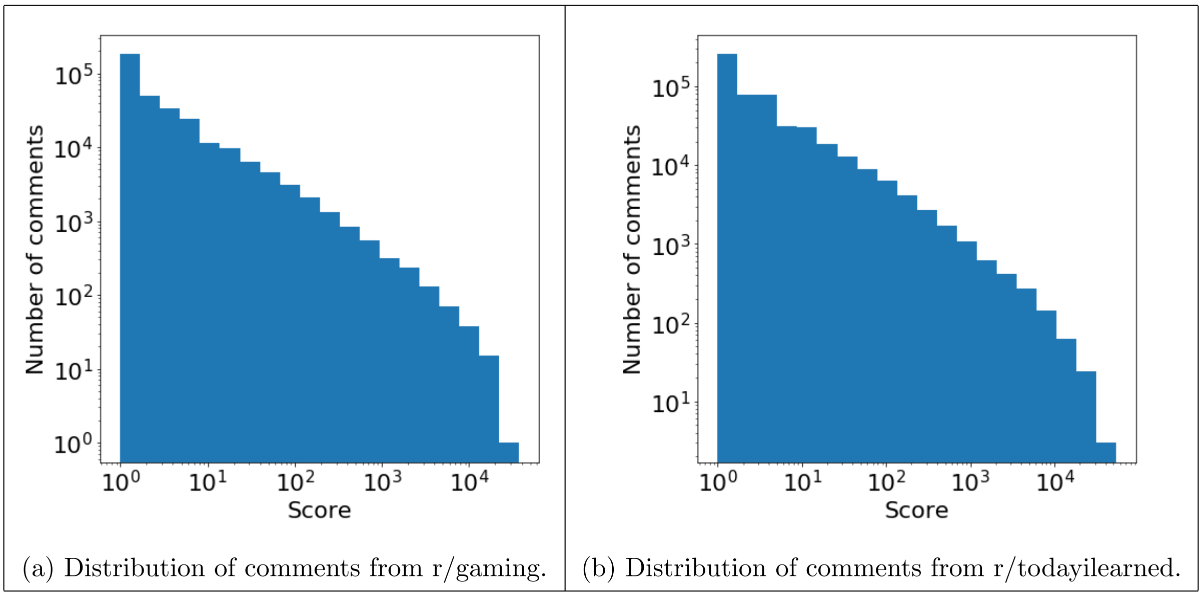
Table 1: Summary of the first set of tests for r/gaming. A positive log-likelihood ratio indicates that the null distribution is more likely, while a negative log-likelihood ratio indicates that the alternative distribution is more likely.

Distribution	Stretched Exponential	Log Normal
Log Likelihood Ratio	54.80	135.43
Associated p value	7.87e-15	3.82e-17

Table 2: Summary of the second set of tests for r/gaming.

Note that this form of testing is not a direct test on whether the score data does or does not follow a given distribution. Such testing is possible - one method uses K-S statistics and bootstrapping [1]. However for our purposes, this is likely unnecessary, since it is clear that the truncated power-laws fit our data well over the majority of the data's range.

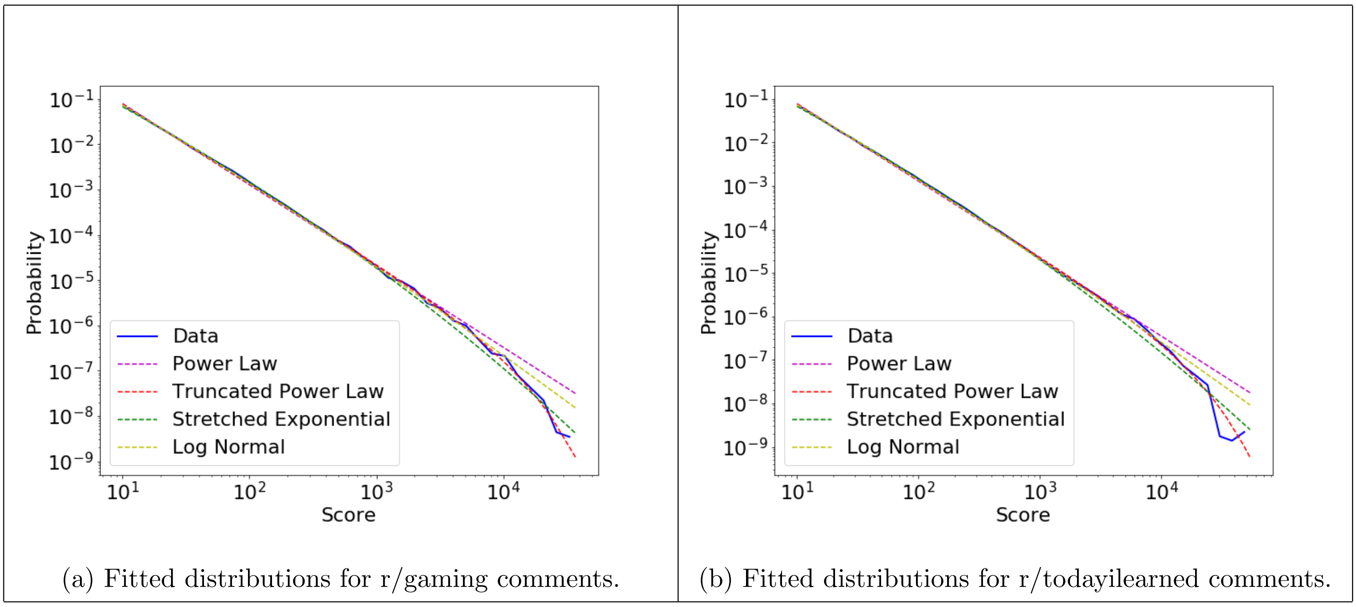
We can repeat the procedure above for comments. Figures 6 and 7 show the log-log histograms and fitted distributions for r/gaming and r/todayilearned comments. Interestingly, the comment data appears to follow the decreasing linear trend much more closely.



(a) Distribution of comments from r/gaming.

(b) Distribution of comments from r/todayilearned.

Figure 6: Log-log histograms of the scores of comments from r/gaming and r/todayilearned.



(a) Fitted distributions for r/gaming comments.

(b) Fitted distributions for r/todayilearned comments.

Figure 7: Comparing the fitted distributions for the r/gaming and r/todayilearned comment data.

According to the log-likelihood ratio tests, the truncated power-law is also the best fit for the distribution of comment scores from both subreddits.

3 Time resolved data

Given that our work in the previous section shows that the score distributions of posts and comments from several subreddits are plausibly distributed as truncated power-laws, we now focus on the second criteria for establishing the existence of the Matthew Effect. Recall that this second condition was that the rate of change in score for a piece of content is proportional to the content's current score, s, i.e.

$$\frac{\Delta s}{\Delta t} = A(t)s^{\gamma}.$$

While in reality the coefficient $A(t)$ will be time dependent (since posts and comments have a limited lifespan before they are hidden by the ranking algorithms), we will simplify this expression by treating A as a constant. Under this assumption, after taking a log transform the relationship becomes

$$\log_{10}\left(\frac{\Delta s}{\Delta t}\right) = \log_{10}(A) + \gamma \log_{10}(s).$$

Hence, provided that γ is positive, if the above relationship holds, then on log-log axes we should observe a linear increasing trend.

To establish this relationship requires time resolved data about Reddit content. This essentially means that we need to track content, which is possible by using the Python package `praw`[8]. `praw` allows us to access submission and comment streams via the Reddit API, and to check the current score of a post/comment using the content's unique ID. Hence, we write a script that tracks a given number of posts/comments over 18 hours at approximately 1 hour intervals. 18 hours was chosen as the tracking lifetime since after some experimentation, it appears that most posts and comments have only negligible changes in score after that length of time.

One complication is that most subreddits do not receive large numbers of new posts every hour. Many will only have 5-50 new posts created each hour, so attempting to track large numbers of posts can take significantly more that 18 hours, because the script will need to spend several hours searching for posts as well. To avoid this, instead of getting time resolved data at an individual subreddit level, we can just track large numbers of posts made to r/all. As the name suggests, r/all compiles posts from all of the subreddits on Reddit, meaning that there are many more new posts submitted to r/all than any other subreddit. Tracking comments, however, does not tend to have this problem, since new comments are made far more frequently than new posts.

Figure 8 is a score versus time plot of 2,000 posts to r/all. However, analysing the raw time resolved data of individual posts or comments will not be sufficient, since the process of Reddit content being

[8] `praw`'s documentation can be found here.

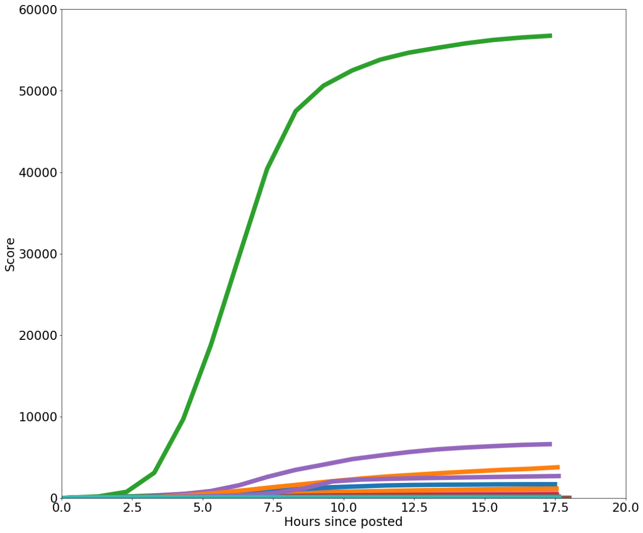


Figure 8: Score vs time for 2000 posts to r/all.

upvoted and downvoted is inherently stochastic. Instead, we will first need to apply a technique to try to account for this stochastic nature. There are two main methods which have been employed previously; averaging and cumulation [3]. In this project we will employ the former.

As the name suggests, the method of averaging intends to average out the stochastic fluctuations in our time resolved data to retrieve the underlying trend. The time resolved data which was collected takes the form of two matrices. One matrix contains the scores of the content and the other contains the UNIX time stamps at which the score data was collected. Each row of the matrix represents an individual piece of content. Hence, taking the row-wise difference of the score matrix and dividing it element-wise by the row-wise difference of the timestamp matrix will give us an approximation for $\frac{\Delta s}{\Delta t}$. Since we have taken row-wise differences of both matrices, the matrix of rates will have one less column than the matrices of scores and times. If we discard the last row of the scores matrix, however, then scores and rates matrices will have the same dimensions, allowing us to pair the data in these matrices element-wise. We discard the last column of the score matrix since the content's rate of change at the last time step should be negligible.

We then collate all of the pairs of scores and rates of change in score, before binning them by score into groups of roughly 100. We then find the average score and average rate of change of score within each bin. This process will give us vectors of average scores and average rates. After taking log transforms of these vectors, we want to recover an increasing linear trend, as discussed earlier.

When this process is applied to the time resolved data from Figure 8, Figure 9 is the result. Here a linear regression has been fitted as the black line, with the grey shaded area around the line representing a 95% confidence interval for the regression. The coefficients of the line of best fit and corresponding r^2 value are also included. Of note is the slope of the regression, since this is our approximation for γ. In this case, a strong positive linear trend is present, with $\gamma = 0.827$. R calculates the 95% confidence interval for γ as $(0.74, 0.91)$, suggesting the presence of slightly sublinear preferential attachment.

Note that the assumption plots for each linear model included in this section can be found in Appendix B. For the linear model in Figure 9, the Normal Q-Q plot indicates that normality is likely reasonable given the small dataset. The majority of the data on the Normal Q-Q plot follows the roughly linear trend that we would expect, however there is a clear deviation in the tails. Homoskedasticity and linearity also appear to be justified, since aside from a few points, the scatter on the Residuals versus Fitted plot appears to be random, and its variance appears to be roughly constant. Independence is also likely to be justified, since knowledge of the score of one post/comment should not affect our knowledge of the score of another post/comment.

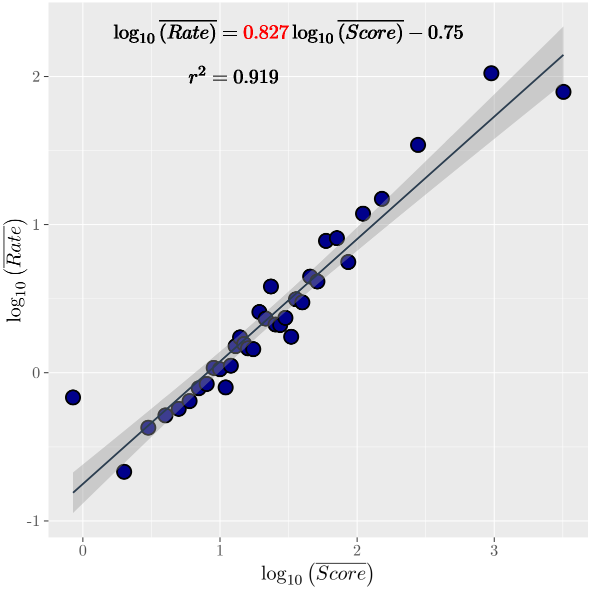
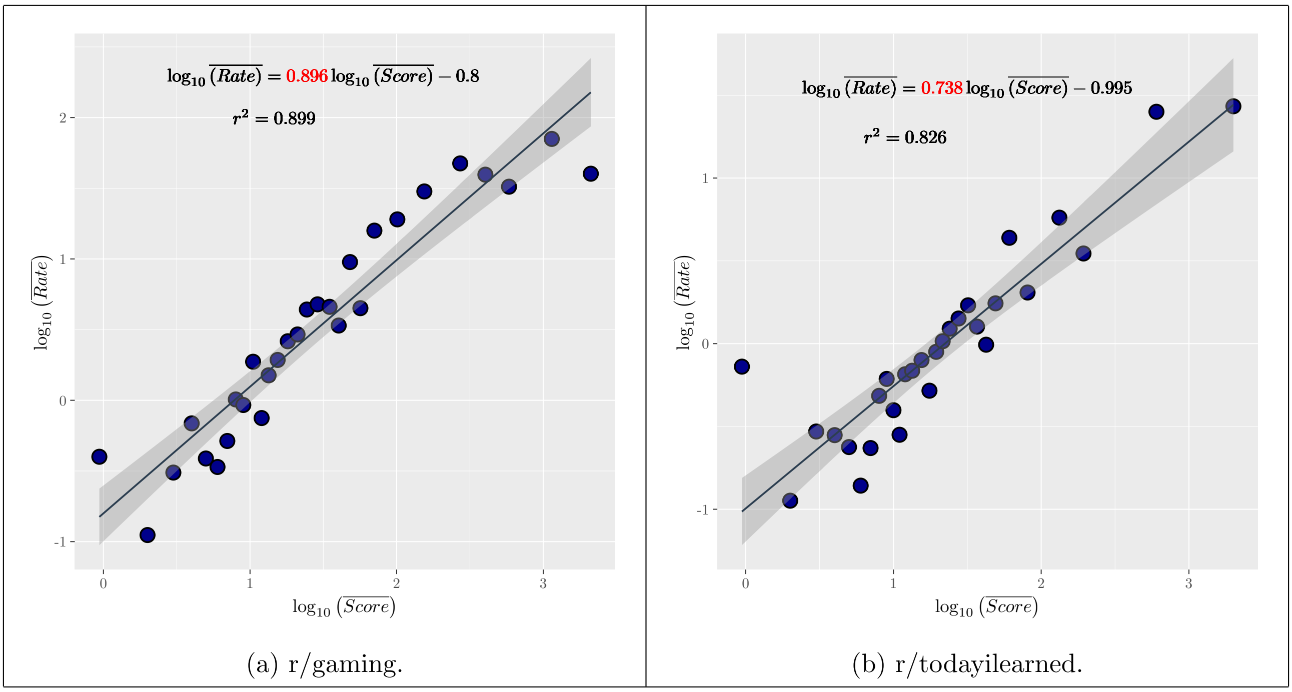


Figure 9: Average score vs average rate of change in score for posts to r/all after log transformation.

Having established strong evidence of sublinear preferential attachment in Reddit posts, we can also apply the same techniques to comments. Since these are easier to track, we can get time resolved data for comments at the subreddit level. Figure 10 shows the result of the averaging technique applied to



(a) r/gaming.

(b) r/todayilearned.

Figure 10: Average score vs average rate of change in score for comments to r/gaming and r/todayilearned after log transformation.

time resolved comment data from r/gaming and r/todayilearned respectively. Both plots show strong linear trends in the averaged data. For r/gaming, $\gamma = 0.896$, while for r/todayilearned $\gamma = 0.738$. Both cases, therefore, also suggest the presence of sublinear preferential attachment.

While we have only shown two subreddits in this report, this process can be easily adapted to examine any subreddit. As discussed earlier, smaller subreddits are harder to collect time resolved data on, but in theory the Python scripts that were in applied in this report should work on any subreddit.

4 Summary

By using the Python packages `psaw` and `powerlaw`, we have been able to establish that the score distributions of both posts and comments to r/gaming and r/todayilearned are plausibly distributed as truncated power-laws.

By gathering time resolved data using `praw`, we were also able to establish the proportionality between a piece of content's current score and its rate of increase in score for both subreddits.

This strongly suggests the existence of sublinear preferential attachment for both of these subreddits. The same process can easily be applied to more subreddits to broaden the scope of this study.

One piece of future research that is not strictly related to the topic of this report is to create a Markov chain to simulate Reddit content. It would be interesting to try and account for the strange shapes that were observed in the log-log histograms earlier in the report, like the log-log histogram of scores for posts to r/todayilearned.

5 References

[1] A. Clauset, C.R. Shalizi, and M.E.J. Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703, 2009.

[2] Edwin Wilson. Probable Inference, the Law of Succession, and Statistical Inference. *Journal of the American Statistical Association*, 22:209–212, 1927.

[3] Matjaž Perc. The Matthew effect in empirical data. *Journal of the Royal Society Interface*, 11, 2014.

Appendices

A Distribution definitions

Name of distribution:	Kernel of pdf:
Power-law	$x^{-\alpha}$
Truncated power-law	$e^{-\lambda x}x^{-\alpha}$
Exponential	$e^{-\lambda x}$
Stretched Exponential	$x^{\beta-1}e^{-\lambda x^{\beta}}$
Log-normal	$\frac{1}{x}\exp\left(-\frac{(\ln(x)-\mu)^2}{2\sigma^2}\right)$

Table 3: The kernels of each of the distributions used in the log-likelihood ratio tests [1].

B Assumption plots

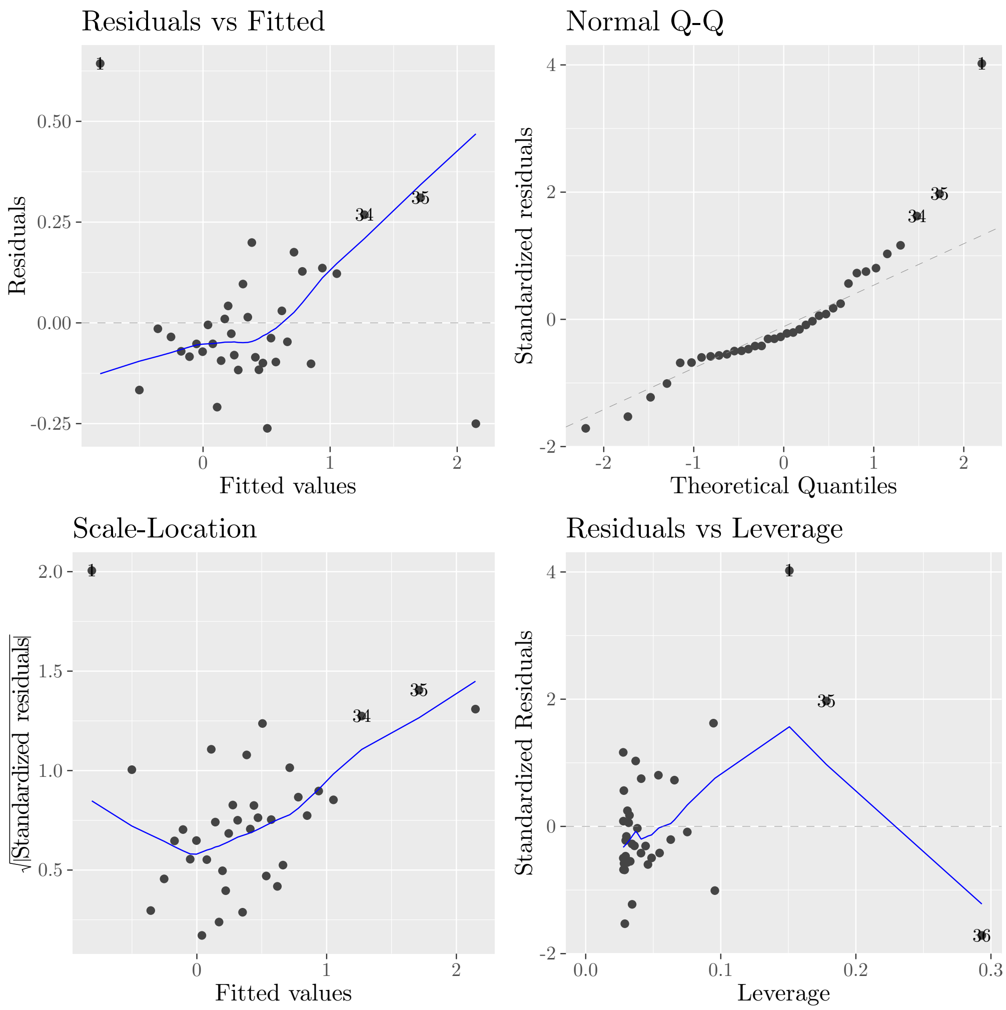


Figure 11: Assumptions for the r/all linear model.

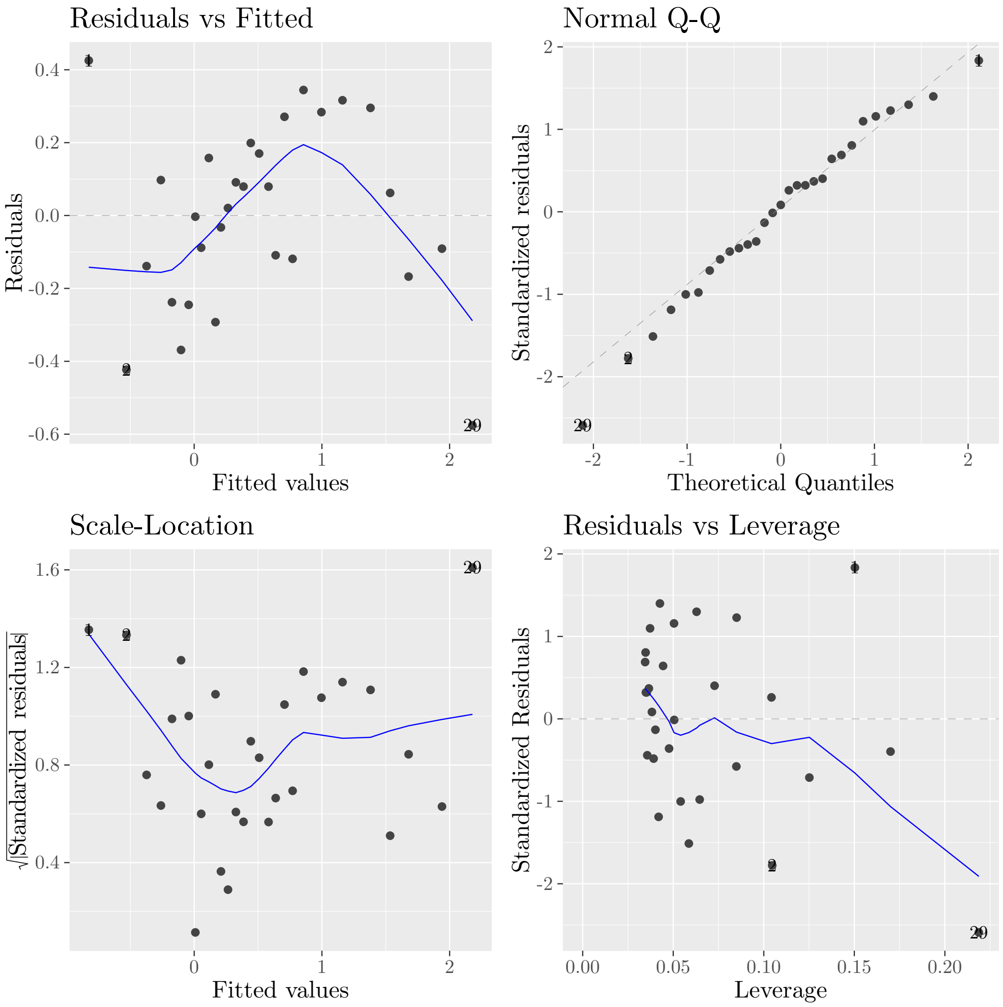


Figure 12: Assumption plots for the r/gaming linear model.

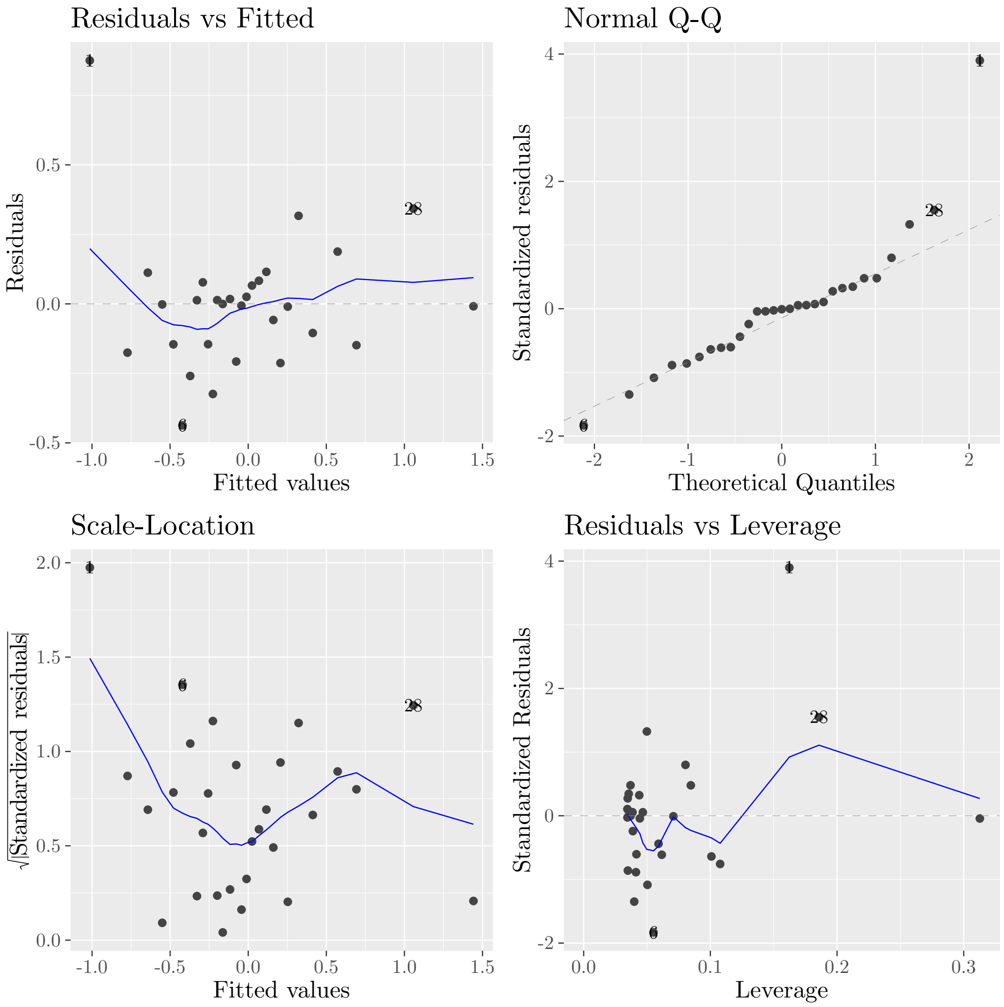


Figure 13: Assumption plots for the r/todayilearned linear model.