Make Sure to Drop a Like—The Effect of Consumer Rating on Application Success

Oleg Ananyev, Oren Carmeli, Romain Hardy, Sam Rosenberg

Fall 2021

Contents

1 Introduction

In 2020, Statista reported a total of 220 billion mobile application downloads globally. Another study found that approximately 70% of all digital media spend in the United States is spent on mobile applications. As digital applications become the primary media through which organizations engage with their customers, it becomes increasingly valuable to understand the drivers that lead to application success.

The Google Play Store is one of the major hubs for Android mobile phone and tablet applications. Users can download any application on the Google Play Store for personal consumption across a wide range of categories. Making an application stand out in a sea of thousands of competing applications is no trivial task, however. To succeed, developers must carefully consider factors such as price, application size, and genre. Another variable that may be critical to an application's success is consumer rating. Today, smart algorithms play a key role in suggesting applications to consumers, creating a feedback loop that propels certain applications towards success and leaves others behind. An understanding of the relationship between consumer rating and application success would be incredibly valuable to developers seeking to create the next viral application.

The following study is a causal analysis of the relationship between consumer rating and application success. Using a data set scraped directly from the Google Play Store, we will build linear models that assess the importance (or lack of importance) of consumer rating on application success, with additional variables such as price, application size, and application category serving as controls. If our study supports the existence of a causal pathway, it would signal application developers to invest heavily in improving their ratings, for instance by interviewing customers and implementing new features. Our study may also be of interest to Google and other companies seeking to improve consumer engagement with their applications.

Given our prior beliefs on what factors motivate individuals to download applications, we believe there are omitted variables not included in our data set that may influence application success. These include product brand awareness, application store rankings, and the total addressable market, among possible others. In spite of these limitations, our study should provide useful insight into the causal factors of application success.

The paper is structured as follows. Section 2 outlines our research question, and Section 3 describes the causal theory we will use to contextualize our analysis. Section 4 describes the data we leverage for our models and Section 5 discusses our research design. Section 6 highlights our exploratory data analysis, followed by our statistical models in Section 7 and results in Section 8. Section 9 considers the limitations of our models, and Section 10 presents our conclusions.

2 Research Question

The goal of this study is to assess the causal factors of application success. Specifically, we seek to determine if there is a statistically significant relationship between consumer rating and application success. In the ensuing sections, we will answer the following question:

Does having a higher consumer rating score lead to more downloads for Google Play Store applications?

3 Causal Theory

Before we can discuss our data and research design, we must first describe the causal model which will serve as the reference point for our analysis. We identify five factors that bear causal influence on the success of an

application. These are (1) consumer rating, (2) price, (3) category, (4) age, and (5) size. Our proposed causal graph is shown below.
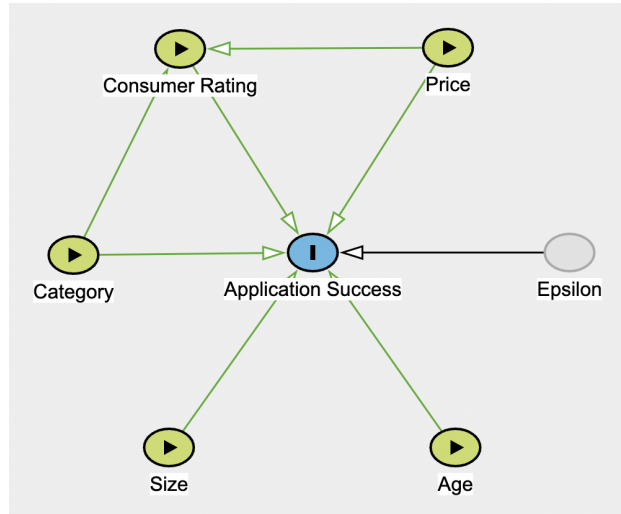

Figure 1: A hypothetical causal diagram for Google Play Store applications.

3.0.1   Application Success

Our objective is to assess whether consumer rating has a positive effect on application success. There are different ways in which we could operationalize success; we choose to use raw download count as a surrogate, since it directly measures the number of consumers that made the decision to download an application. Although this choice ignores other potential aspects of success, such as revenue and social impact, it is effective in its simplicity and appropriate for an exploratory study.

3.0.2   Consumer Rating

The leading independent variable of interest is consumer rating. We hypothesize that higher ratings improve application success, because an application that has been highly rated is one that has been deemed worthwhile by other users. When consumers decide which applications to download, they will likely trust the opinions of their peers and select those with positive reviews. Conversely, we expect applications with negative ratings to have less success, since other users have judged them poorly. There should not be causal pathways leading from consumer rating to any of the other explanatory variables, since they are determined during the development phase of an application whereas consumer rating is decided once an application is published to the Google Play Store.

Although we have not included it in the diagram, there is the possibility that a reverse causal pathway exists from application success to consumer rating. Successful applications are those that are enjoyable to a large number of consumers. Therefore, it is possible that users will rate applications with high download counts higher than applications with low download counts because they are primed to believe that such applications are better—otherwise, the successful applications would not have received so many downloads. Generally, we expect the reverse pathway to be weaker than the forward one. If rating and success have positive effects on one another, however, our models may suffer from positive feedback. We will discuss the implications of this effect in Section 9.

3.0.3   Price

We expect price to have a causal effect on application success. Generally, we believe that free applications will have higher download counts due to the lack of a monetary barrier. Of course, this may not always be the case, as paid applications could offer better features and more desirable experiences, thus encouraging

downloads. We also anticipate a causal pathway from price to consumer rating. In particular, we believe that consumers are more likely to rate paid applications positively than they are to rate free applications positively, since the former type offers better features.

3.0.4 Category

The category that an application belongs to is likely to affect its success. Certain categories of applications appeal to broad audiences and are more likely to find success than applications which appeal to smaller subsets of consumers. This interaction may not necessarily be so straightforward, however. If an application belongs to a popular category, then it also has to compete with other applications in the same category, which may in fact be detrimental to its success. Meanwhile, applications belonging to niche categories could have a greater chance of achieving success simply due to the fact that they have fewer competitors. Globally, we expect that the most successful applications will belong to popular categories, but that moderately successful applications will be spread across different categories.

Category is also a predictor of consumer rating. Due to stylistic and functional differences between application categories, it is likely that they are reviewed against different criteria. For example, a consumer reviewing a mobile game may place emphasis on the graphics, the fluidity of the controls, and the balance of the game mechanics, among others. A lifestyle application, on the other hand, will probably be judged on completely different features, such as ease of use, relevance in every day life, and usefulness. If review criteria depend on application category, then consumer ratings assuredly do too. It is difficult to predict in advance what categories are positively or negatively associated with consumer ratings, though we expect categories with narrower consumer bases to receive harsher ratings. Categories of applications that can lead users to feel frustrated, such as games, social media, and dating applications, may also receive more negative reviews on average.

3.0.5 Age

If we are to interpret application success in terms of the number of downloads an application accumulates, then the age of an application is necessarily an influential factor. Applications can only receive more downloads as time passes, so the longer an application remains on the Google Play Store, the more downloads it is likely to have. As an example of why this is important, consider two applications: one that was uploaded one year ago, and one that was uploaded one week ago. The older application receives 100 downloads a month for the whole year while the newer application receives 1,000 downloads in one week. If we were to only compare the raw download counts (1,200 to 1,000), it would seem as if the older application is more successful. By bringing age into the analysis, we are able to compare the applications by download rate instead of count, thus realizing that the newer application is more successful than its older counterpart.

We do not expect age to have causal effects on the other explanatory variables, though it might be possible to argue that consumer rating and price are affected by age. Unless our analysis clearly demonstrates otherwise, we will assume these effects are negligible.

3.0.6 Size

We foresee two opposing sides to the relationship between application size and success. The first is a negative effect; given the limited space available on mobile and tablet devices, users may be more likely to download smaller applications. Alternatively, application size could be an indicator of production quality, in which case users may prefer larger applications over smaller ones. We hypothesize that the first effect takes precedence.

3.0.7 Epsilon

Epsilon represents variables which may influence application success but are independent from the other variables in our causal graph. For instance, these could include geographical location or the time of day at which a download occurs. Crucially, we assume that there do not exist any directed paths from epsilon to any of the five explanatory variables. This guarantees independence and is necessary for ordinary least-squares regression to be valid.

4   Data

To answer our research question, we will leverage public data about applications available on the Google Play Store. This data was randomly scraped from the Google Play Store interface and uploaded to Kaggle in 2019. It contains key information about sampled applications, such as downloads, file size, consumer rating, category, and price. In total, the data contains records of about 10,000 applications.

For the modeling phase, we will use ordinary least-squares (OLS) regression. OLS regression is the plug-in estimator for the best linear predictor of a dependent random variable given the joint distribution of a set of independent random variables. Although OLS regression is often used with the goal of making predictions on new data, we will instead use it to answer a causal question about the relationship between variables. By interpreting model coefficients within the context of our causal theory, we will develop a statistically valid argument that addresses the research question.

In our case, the dependent variable is application success and the independent variables are the predictors included in our causal model—consumer rating, price, category, age, and size. Unfortunately, these variables do not map exactly onto the fields in our data set, so we will need to make certain approximations. The features that most closely correspond to our dependent and independent variables are listed below.

4.1   Dependent Variable

- Application Success — `installs` (the accumulated number of downloads since the application was uploaded to the Google Play Store)

4.2   Independent Variables

- Consumer Rating — `rating` (the average consumer rating of an application, out of 5)
- Price — `price` (the price of the application) and `type` (the price type of the application, free or paid)
- Category — `category` (the category tagged for the application, i.e Lifestyle, Game) and `content_rating` (the official content rating given to the application, i.e Teen, Everyone, Mature 17+)
- Age — `current_version` (the current version of the application) and `last_updated` (the number of years since the application was last updated)
- Size — `size` (the download size of the application, in units of MB)

5   Research Design

We seek to test our hypothesis that positive consumer ratings lead to greater success for Google Play Store applications. To moderate and refine our analysis, we will incorporate four additional control variables: price, category, age, and download size. Price lets us differentiate between paid and free applications, while category lets us differentiate between different genres and target audiences. Age is included to account for the fact that applications which have been available in the store for a long time have an innate advantage over applications which were uploaded recently; with age as a variable, we can directly compare applications which were uploaded simultaneously. Finally, we include download size as it could be an indicator of production quality.

Our data set offers a cross-sectional view of Google Play Store applications in 2019. Since not every feature maps directly to the variables we have defined in our causal framework, we make certain approximations, as listed above. Although these mappings are sometimes imperfect, we believe they are sufficient for a meaningful analysis.

Before we proceed with building statistical models, we will conduct a thorough exploratory analysis of the data. We will note important patterns and trends in the data set, filter problematic entries and outliers, and justify necessary variable transformations. From there, we will build three models of increasing complexity and interpret the model coefficients, verify underlying assumptions, and discuss possible limitations. The first model will estimate how `installs` depends on `rating`, and will serve as a baseline for further analysis. The second model introduces control variables from our causal theory which we hypothesize have an effect on

success and consumer ratings. The third and final model explores interactions between ratings and other explanatory variables. As justification for adding specific covariates, we will provide visualizations and conduct statistical tests that demonstrate their significance.

6 Exploratory Data Analysis

Prior to exploring the data, we will create rules to filter and clean records based on logical conditions. This procedure involves removing duplicate records, removing records with null review counts, and removing records with consumer ratings greater than 5. Since our research question focuses on consumer rating, we elect to only keep records with valid values for that field. Intuitively, we do not believe that consumer rating is a suitable predictor if there are fewer than 100 ratings in total, so we only keep applications that exceed this amount. Although this step removes almost 25% of the data, it is acceptable given that the initial data set contains approximately 10,000 records.

After these initial operations, the cleaned data set contains 7,226 records (distinct applications) and 24 metadata columns, 11 of which we built. We split up the exploration into two sections based on the type of the variable, numeric and categorical.

6.1 Numeric Variables

We want to understand the underlying distribution of each numeric variable as well as examine the correlations and covariances among them. The distributions are used to measure the quality of the data, identify outliers, and evaluate the need for variable transformations. In addition, understanding the correlations between variables helps us to highlight those that might explain the variance of our dependent variable, and to quantify the level of collinearity between different features.

6.1.1 Distributions

Table 1: Distribution statistics for numeric variables

| Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
installs	7,226	7,646,081.000	45,741,633.000	1	10,000	1,000,000	1,000,000,000
size	7,226	24.296	24.006	1	5.8	35	100
reviews	7,226	279,760.700	1,758,072.000	1	97	35,930.2	44,893,888
rating	7,226	4.171	0.551	1	4	4.5	5
price	7,226	1.139	17.948	0	0	0	400
current_version	7,226	5.241	25.529	0	1.1	3.7	858

After the cleaning step, all feature values appear valid; there are no negative values, null values, or values close to infinity. Notably, only two fields contain zero values, `price` and `current_version`. This will be important when deciding whether or not to apply logarithmic transformations. We also note that we have treated `current_version` (the version number of an application) as a metric variable. Although this is technically incorrect, we feel it is justified given that this feature is ordinal in scale and has consistent intervals.

From the differences between their medians and means, `size` and `reviews` appear to have approximately normal distributions, while the others have strong right or left skews. Since we expect a causal pathway to exist from `installs` to `reviews` (i.e. high install counts lead to high review counts), we do not include `reviews` as a predictor in our analysis.

Application Success

Application success is measured using the `installs` feature, which represents the raw download count of an application. It is important to note here that `installs` is a binned feature. The bins start at 1 and scale upwards logarithmically; 1+, 5+, 10+, 100+, 500+, etc. For example, a value of 100+ means that the application has between 100 and 499 downloads. In the cleaning step, we remove the + sign and convert

`installs` to a metric variable. This conversion is valid because the feature is ordinal and there is a measurable distance between bins. Given that this distance scales logarithmically, we can also claim that the distance between bins is consistent. Although there is precision error due to the fact that binning obscures the true download count, we believe that `installs` can be treated as a metric variable in practice. The raw distribution has a strong right skew of $\tilde{\mu}_3 = 15.2$; the maximum is 10^9 and the median is 10^5. Unsurprisingly, applying a logarithmic transformation causes the resulting distribution to resemble a normal distribution, which makes it an appropriate transformation for future modeling.
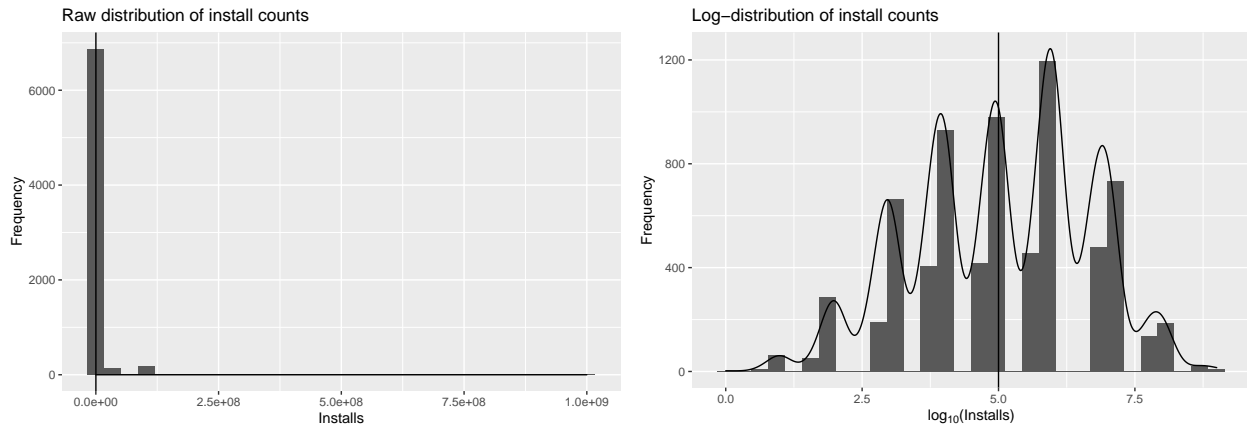

Consumer Rating

Consumer rating is measured using the `rating` feature, which represents the average consumer rating of an application. This feature appears to have a distribution that is approximately normal, but that is negatively skewed towards larger values ($\tilde{\mu}_3 = -1.75$). Average ratings range between 0 and 5, with a median of 4.3. Applying a logarithmic transformation only weakly affects the distribution, so we opt to keep the raw feature instead.
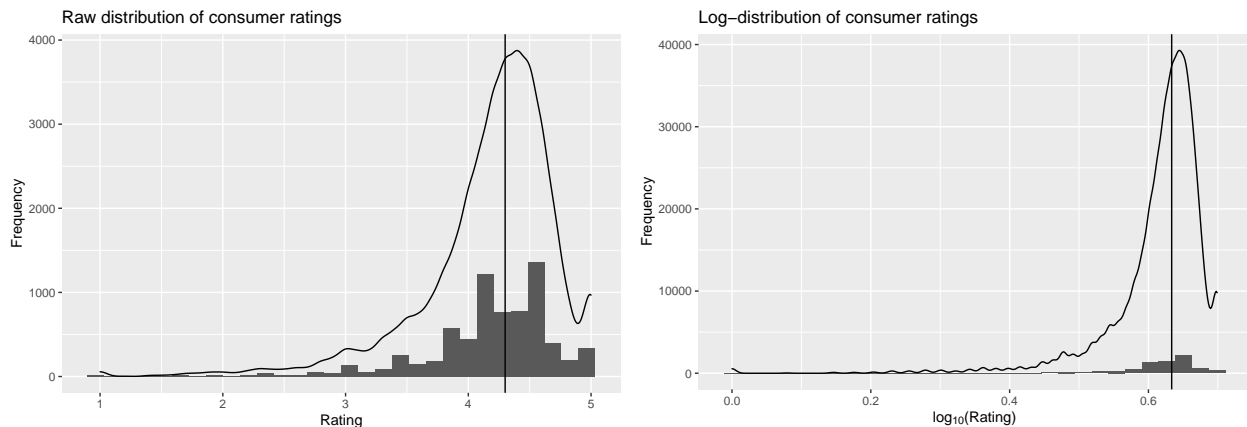
Size

Download size is measured using the `size` feature. `size` has a distribution that is slightly closer to normal than `installs`, though it still has a noticeable right-leaning tail ($\tilde{\mu}_3 = 1.36$). The maximum download size is 100 MB, and the median is 15 MB. Applying a logarithmic transformation shifts the distribution closer to normality. Although the log-distribution fluctuates near the middle, we still find it more appropriate than the raw distribution for modeling.

Raw distribution of download sizes   Log–distribution of download sizes
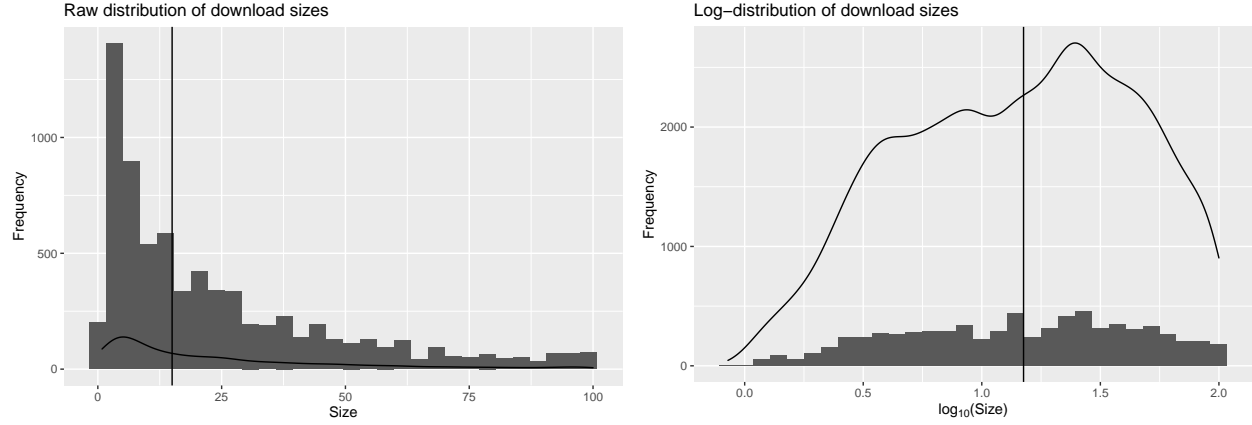
Price

Application price is measured using the `price` feature. In total, 92.7% of applications in the data set are free to download; the median price is 0, and the skew is $\tilde{\mu}_3 = 21.6$. As a result, we believe that neither the raw distribution nor the log-distribution of `price` are desirable for modeling, as they deviate too far from normality. Instead, we will transform `price` into an indicator variable that takes value 1 for paid applications and 0 for free applications.

Raw distribution of prices   Log–distribution of prices
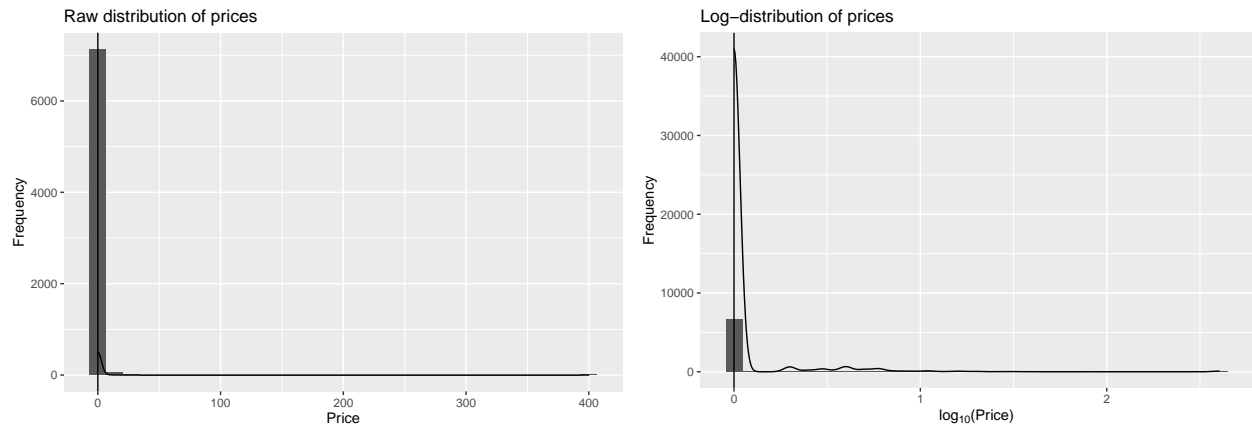
Age

The age of an application is measured using the proxy features `current_version` and `last_updated`. We notice that that `current_version` is skewed right ($\tilde{\mu}_3 = 15.7$), with a maximum value of 858 and a median of 2. Similarly, `last_updated` is skewed right ($\tilde{\mu}_3 = 2.16$), with a maximum value of 8.22 and a median of 0.253. Due to their strong positive skews, we apply logarithmic transformations to both features before the modeling phase.
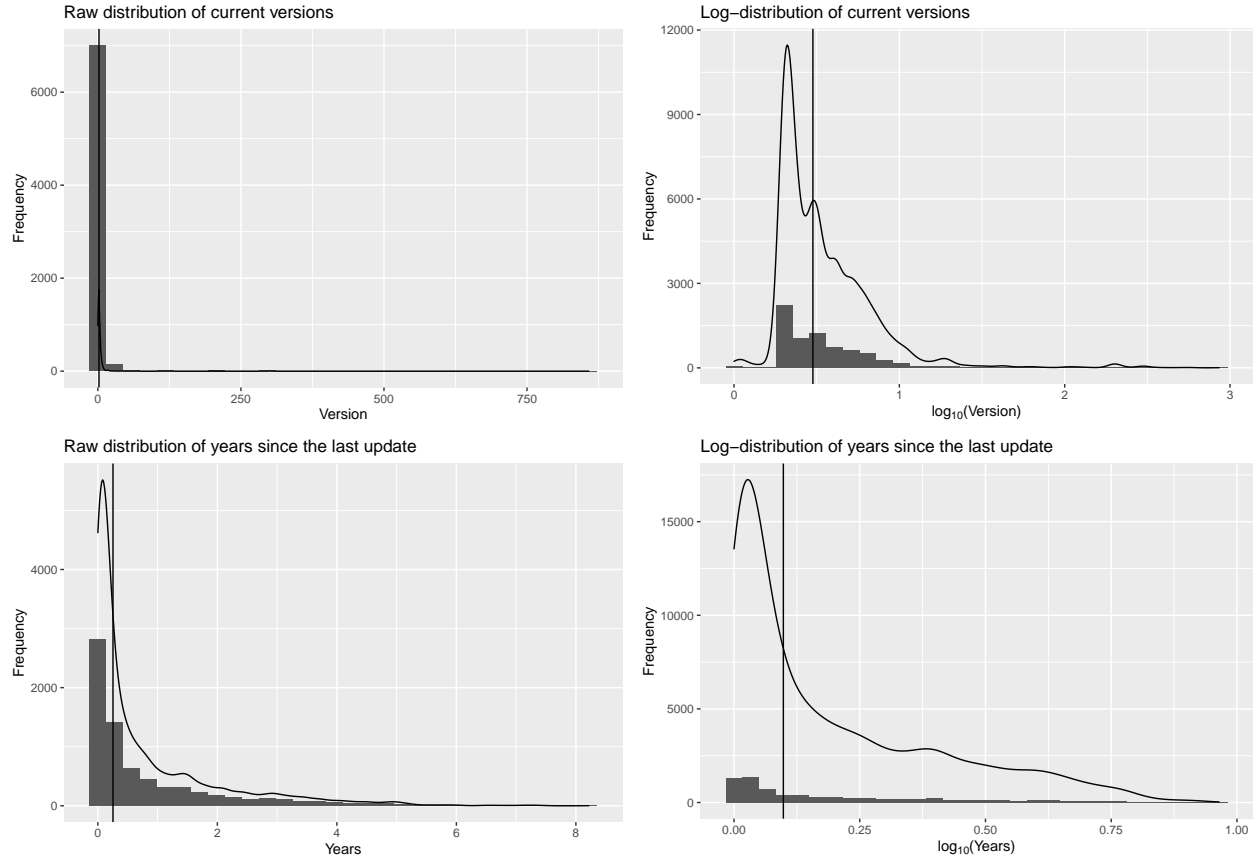
6.1.2 Correlations

Aside from `log_reviews`, none of the other numeric features have strong correlations with `installs`. The high correlation coefficient between `log_reviews` and `log_installs` supports our hypothesis that there is a causal path from `installs` to `reviews`; indeed, consumers will generally only review applications once they have downloaded them. Since one of our goals is to build an efficient model, we would ideally like to see stronger correlations between the independent variables and the dependent variable. There may be latent predictive power in the interactions between variables, however. Notably, we observe that the correlation coefficient between `rating` and `log_installs` increases by a factor of two when we exclude applications with fewer than 100 reviews. This agrees with our assumption that consumer rating bears causal influence on application success, and motivates our choice to remove applications with low review counts. Lastly, the correlation plots do not indicate high levels of collinearity among the numeric variables.
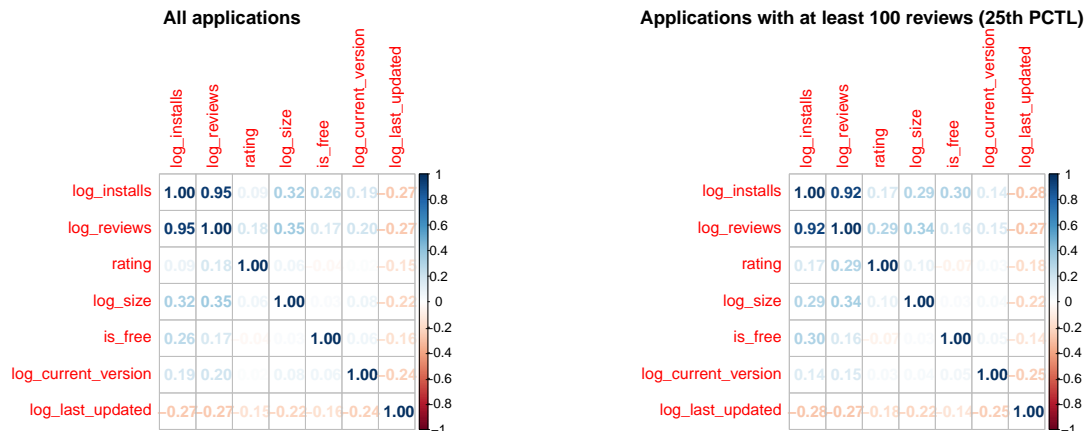
6.2 Categorical Variables

For the categorical features, we want to aggregate the frequency and mean of `log_installs` by feature sublabel. We use the mean as a measure of central tendency rather than the median given the approximately normal distribution of `log_installs`. To identify which categorical features have the largest dispersion across sublabels, we built a quantile table using the sublabel averages of `log_installs` as the datapoints for each categorical feature. We choose to exclude sublabels with fewer than 100 applications to reduce noise.

Table 2: Quantile summary table

Variable	0%	25%	50%	75%	100%	Diff (Max - Min)	Diff (Max - Min) / Min
category	3.7747	4.6795	4.8993	5.2020	6.0252	2.2505	0.5962
content_rating	4.8948	5.1170	5.3555	5.6133	5.8939	0.9991	0.2041
current_version	4.1225	4.8891	5.1031	5.3976	5.6553	1.5328	0.3718
android_version	4.2383	4.8487	4.9869	5.0170	5.2442	1.0059	0.2373

Table 3: Application category summary table

Label	Count Apps	Log of Install Count (Avg)	Variable
GAME	934	6.0252	category
PHOTOGRAPHY	220	5.6902	category
SHOPPING	154	5.6543	category
VIDEO_PLAYERS	107	5.3847	category
SPORTS	230	5.2027	category
COMMUNICATION	197	5.2017	category
SOCIAL	162	5.0700	category
HEALTH_AND_FITNESS	190	5.0181	category
TRAVEL_AND_LOCAL	143	4.9550	category
PRODUCTIVITY	223	4.9096	category
FAMILY	1571	4.8891	category
NEWS_AND_MAGAZINES	157	4.8004	category
TOOLS	604	4.7813	category
DATING	141	4.7436	category
PERSONALIZATION	268	4.7023	category
BOOKS_AND_REFERENCE	141	4.6110	category
FINANCE	260	4.5524	category
LIFESTYLE	265	4.4823	category
BUSINESS	221	4.0993	category
MEDICAL	272	3.7747	category

Table 4: Content rating summary table

Label	Count Apps	Log of Install Count (Avg)	Variable
Everyone 10+	293	5.8939	content_rating
Teen	810	5.5198	content_rating
Mature 17+	326	5.1911	content_rating
Everyone	5794	4.8948	content_rating

From the quantile summary table, all four categorical features have at least a 20% dispersion across the minimum and maximum sublabels. This indicates that these features might possess predictive power as inputs to regression models. Notably, the `category` column seems to have the largest dispersion across groups, implying that it could be greatly impactful. We have omitted the variable `android_version` as it is difficult to interpret.

Overall, this analysis motivates using `category` and `content_rating` as explanatory variables in regression models. To ease interpretability and limit the number of variables, we will use binned and binary versions of these variables, isolating the sublabels with the highest frequencies.

7 Statistical Models

To model the effect of consumer ratings on application success (as understood by the number of downloads an application receives), we will use ordinary least squares regression. Our first model will regress `log_installs`—the logarithmic transform of `installs`—against `rating`. We will then create two more explanatory models to expand on this baseline model, including additional control variables (Section 4.2) and interaction terms. Based on the results of our exploratory data analysis, we have also decided to apply transformations to some of our variables. Our focus is to maximize the predictive power R^2 while maintaining models that are easily interpretable. The three models we will test are:

$$\text{log\_installs} = \beta_0 + \beta_1(\text{rating}) + \epsilon \tag{1}$$

$$\begin{aligned}
\text{log\_installs} = \beta_0 &+ \beta_1(\text{rating}) \\
&+ \beta_2(\text{log\_size}) + \beta_3(\text{log\_current\_version}) \\
&+ \beta_4(\text{log\_last\_updated}) + \beta_5(\text{is\_free}_{\text{TRUE}}) \\
&+ \beta_6(\text{is\_family\_category}_{\text{TRUE}}) + \beta_7(\text{is\_game\_category}_{\text{TRUE}}) \\
&+ \beta_8(\text{is\_tools\_category}_{\text{TRUE}}) + \beta_9(\text{is\_content\_everyone}_{\text{TRUE}}) \\
&+ \epsilon
\end{aligned} \tag{2}$$

$$\begin{aligned}
\text{log\_installs} = \beta_0 &+ \beta_1(\text{rating}) \\
&+ \beta_2(\text{log\_size}) + \beta_3(\text{log\_current\_version}) \\
&+ \beta_4(\text{log\_last\_updated}) + \beta_5(\text{is\_free}_{\text{TRUE}}) \\
&+ \beta_6(\text{is\_content\_everyone}_{\text{TRUE}}) + \beta_7(\text{is\_family\_category}_{\text{TRUE}}) \\
&+ \beta_8(\text{is\_game\_category}_{\text{TRUE}}) + \beta_9(\text{is\_tools\_category}_{\text{TRUE}}) \\
&+ \beta_{10}(\text{rating} \times \text{is\_family\_category}_{\text{TRUE}}) + \beta_{11}(\text{rating} \times \text{is\_game\_category}_{\text{TRUE}}) \\
&+ \beta_{12}(\text{rating} \times \text{is\_tools\_category}_{\text{TRUE}}) + \epsilon
\end{aligned} \tag{3}$$

As justification for including additional covariates in the second and third models, we also conduct F-tests with the null hypothesis that the smaller model is the correct population model. Comparing the first and second models yields $F = 288$ and $p = 0$, allowing us to reject the null. Comparing the second and third models yields $F = 11.7$ and $p = 1.27 \times 10^{-7}$, also allowing us to reject the null. Therefore, our inclusion of these covariates is cogent.

8 Results

In this section, we present the results of the OLS regression models described above. We discuss the statistical significance of the different coefficients, as well as their meaning in a practical sense. The dependent variable for each of these models is `log_installs`, the logarithmic transformation of application install count.

A few observations stand out from Table 5. First, all model coefficients are statistically significant within a threshold of 0.1 excluding two, `is_family_category` in Model 2 and `is_game_category` in Model 3. These variables are derived from `category`, which denotes the category of an application. The coefficient estimate for `is_family_category` also stands out in the sense that it is close to zero, indicating that the Family category is a poor standalone predictor of application success. In Model 3, the coefficient for `is_game_category` is an intercept term and is thus less relevant.

Table 5: Results of three regression models

	Dependent variable:		
	log_installs		
	(1)	(2)	(3)
rating	0.252***	0.141***	0.137***
	(0.037)	(0.036)	(0.045)
log_size		0.701***	0.702***
		(0.038)	(0.038)
log_current_version		0.769***	0.768***
		(0.083)	(0.083)
log_last_updated		−1.083***	−1.072***
		(0.085)	(0.085)
is_free		1.407***	1.409***
		(0.055)	(0.055)
is_family_category		0.026	1.113***
		(0.043)	(0.382)
is_game_category		0.877***	−0.837
		(0.056)	(0.570)
is_tools_category		0.296***	−0.765*
		(0.063)	(0.438)
rating:is_family_category			−0.260***
			(0.092)
rating:is_game_category			0.402***
			(0.137)
rating:is_tools_category			0.264**
			(0.110)
is_content_everyone		−0.209***	−0.207***
		(0.043)	(0.043)
Constant	3.968***	2.139***	2.150***
	(0.149)	(0.176)	(0.209)
Observations	7,226	7,226	7,226
R^2	0.007	0.248	0.251
Adjusted R^2	0.007	0.247	0.250
Residual Std. Error	1.598 (df = 7224)	1.392 (df = 7216)	1.389 (df = 7213)
F Statistic	54.454*** (df = 1; 7224)	263.871*** (df = 9; 7216)	201.697*** (df = 12; 7213)

Note: *p<0.1; **p<0.05; ***p<0.01

Although the explanatory power of Model 3 ($R^2 = 0.251$) is slightly greater than that of Model 2 ($R^2 = 0.247$), we believe that the latter best represents the relationships among the different variables, as it contains fewer extraneous terms. In fact, all coefficients in Model 2 except for `is_family_category` have a p-value less than 0.01. This means that the probability of making a type I error for any of these coefficients (i.e. rejecting the null hypothesis assuming it is true) is less than 1%, providing strong evidence that these coefficients meaningfully improve our ability to predict application success.

In all three models, we observe that `rating` has a positive influence on `log_installs`. Assuming all other variables are held constant, we predict that a one unit increase in consumer rating corresponds to a 14.1% increase in the number of downloads. Notably, our estimates for the `rating` coefficient in Model 2 are lower than in Model 1, due to omitted-variable bias. As we mentioned in Section 9, including additional omitted variables may drive down our estimates even further.

As predicted, free applications are more successful than paid applications, with free applications receiving 1.41% more downloads on average. We also notice that age influences application success; increasing the current version by 1% increases the number of downloads by 0.769%, and increasing the number of years since the last update by 1% decreases the number of downloads by 1.08%.

Although the models generally agree with our hypotheses, some of them were incorrect. For instance, we expected size to negatively affect application success, with larger applications being installed less frequently. Surprisingly, the opposite is true; increasing download size by 1% leads to a 0.701% increase in the number of downloads. This could be because size is an indicator of production quality. Also surprising is the fact that, holding other variables constant, newly updated applications are installed less, which could be due to bias or a flaw in our methodology. One possible explanation is that newly updated applications are inherently less likely to be installed, since they have not had as much time to reach consumers. Finally, Model 3 shows that the interaction between `rating` and `category` may take precedence over the standalone variables. For example, we observe that the relationship between rating and success is stronger for games than for family applications. Whereas a one unit increase in rating causes a 40.2% increase in downloads for games, the same change causes a 26.0% decrease for family applications.

It is worthwhile to note that none of these models have R^2 values that would traditionally be accepted in hard science fields such as physics or molecular biology. However, since our aim is to support the existence of a causal relationship in a social science field, low R^2 values may not be problematic. Given the extreme variability between types of applications, the lack of precision in the dependent variable, and other possible external factors, we still find that these models offer a useful description of how consumer rating affects application success.

9   Model Limitations

9.1   Statistical Limitations

In the following section, we assess the five assumptions of the classic linear model: independence and identical distributions (I.I.D.), no perfect collinearity, linear conditional expectations, homoskedastic errors, and normally distributed errors.

9.1.1   I.I.D.

According to the Kaggle authors, this data set was collected by randomly scraping the Google Play Store. Since no clusters of applications were specifically targeted, we can reasonably use the entire set of applications on the Google Play Store as our reference population. We recognize that applications likely have some degree of interdependence, especially within categories. For example, the success of one application likely has a negative impact on other applications of the same type. Due to the large size of this data set, however, we expect any dependencies to be negligible. We also have reason to believe that the data are identically distributed, as they are drawn from the same population of applications. One could argue that since the Google Play Store changes over time, the distribution also shifts in response. Because the authors make no specific mention of the time frame across which the data was collected, we will assume that they originate

from a cross-sectional snapshot of the Google Play Store and that no shifts in the underlying distribution occurred during the sampling process.

9.1.2 No Perfect Collinearity

We can immediately conclude that the variables included in our models are not perfectly collinear, as otherwise the regressions above would have failed. We can also assess near perfect collinearity for these variables by observing the robust standard errors returned by the regression model. In general, highly collinear features will have large standard errors. Since the standard errors of the coefficients are small relative to their magnitude, we can reasonably conclude that they are not nearly collinear.
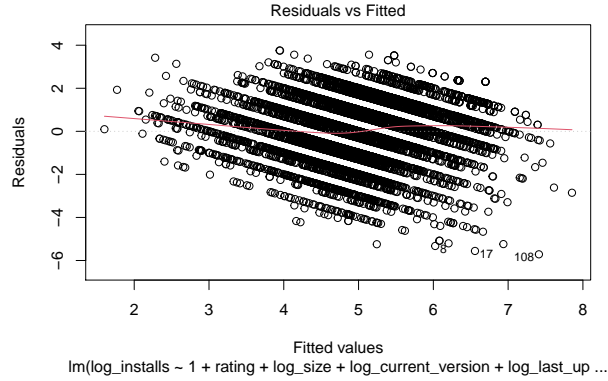
9.1.3 Linear Conditional Expectations

To verify the assumption of linear conditional expectations, we seek to show that there is no relationship between the model residuals and the predictors. That is, the model does not systematically underpredict or overpredict in certain regions of the input space. In the figures below, we plot the relationship between the model residuals and metric-scale predictors. The residuals are generally well-centered around zero, although the model seems to underpredict when `rating` and `log_current_version` are high. The last plot shows the model residuals as a function of the model predictions. Here, the model seems to slightly underpredict in the right region and overpredict in the left region. Despite some inconsistencies, we do not find enough evidence to reject the assumption of linear conditional expectations.
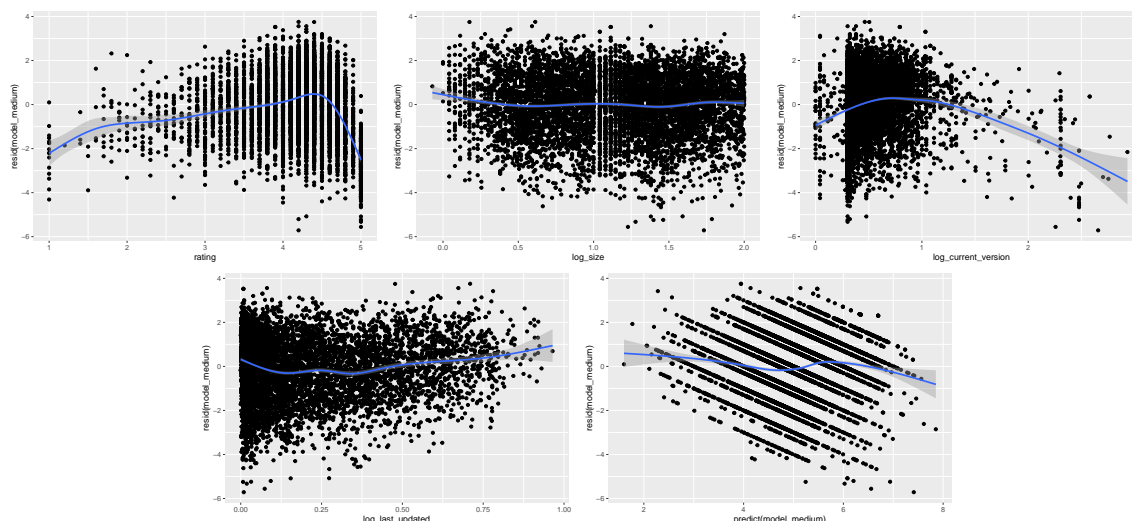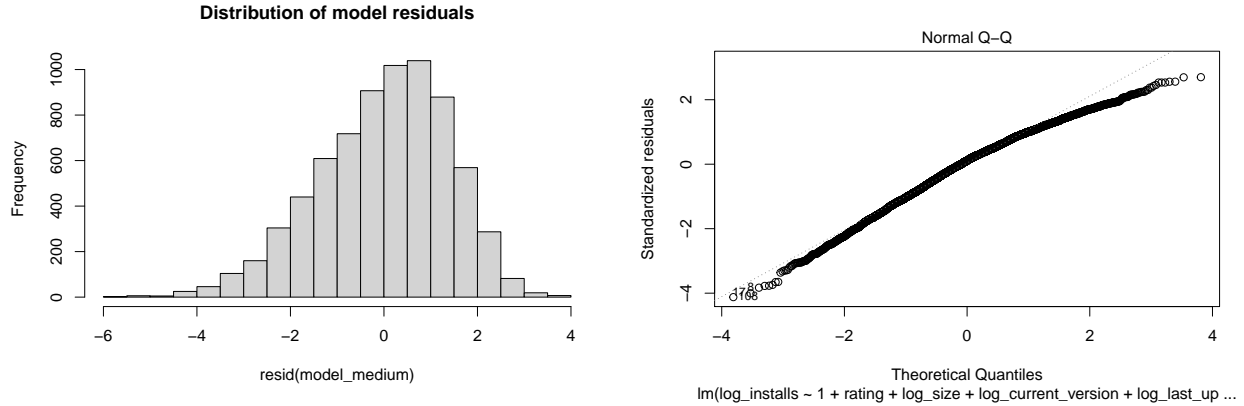

9.1.4 Homoskedastic Errors

When assessing homoskedastic errors, we seek to determine if the variance of the model residuals depends on the predictors. If the homoskedastic assumption is satisfied, then we should observe a lack of relationship (i.e. constant error variance across the input space); conversely, if the data are heteroskedastic then the conditional variance will depend on the predictors (i.e. non-constant error variance across the input space). The plot below is an eyeball test of homoskedasticity, showing the model residuals as a function of the model predictions. We notice that the spread of the residuals is mostly consistent throughout the input space, although the left-hand side is somewhat narrower. As a more concrete assessment, we also perform a Breush-Pagan test with the null hypothesis that there are no heteroskedastic errors in the model. Since the p-value falls below a significance threshold of 0.001, we find enough evidence to reject the null hypothesis. In response to this failed assumption, we report robust standard errors (adjusted for heteroskedasticity) instead of non-adjusted errors.

Residuals vs Fitted

Fitted values
lm(log_installs ~ 1 + rating + log_size + log_current_version + log_last_up ...

9.1.5 Normally Distributed Errors

Here, we seek to determine if the model residuals are normally distributed. Below, we show a histogram and a Q-Q plot of the model residuals. We notice that the left tail is slightly fatter than expected, while the right tail is slightly thinner than expected. However, because the residuals generally seem to follow a normal distribution, we can reasonably justify this assumption and conclude that our estimates are unbiased.

Distribution of model residuals

Normal Q–Q

Theoretical Quantiles
lm(log_installs ~ 1 + rating + log_size + log_current_version + log_last_up ...

9.2 Structural Limitations

The true causal diagram is undoubtedly more complex than the one we have outlined in Section 3. We have identified a few omitted variables that could affect our statistical models, shown in the second causal diagram below. We discuss the relationships these omitted variables have with our existing variables and the ways in which they could bias our results.

9.2.1 Brand Awareness

Brand awareness is the measure of how memorable and recognizable a brand is to its target audience. In the context of our research question, we understand this variable as the percentage of consumers that have the ability to download an application from the Google Play store and are familiar with the application brand. We believe brand awareness to be positively correlated with both application success and consumer rating. Greater brand awareness widens the acquisition funnel for applications, leading to more downloads. Organizations that publish highly rated applications likely have more disposable income for marketing and hence better awareness among consumers. Omitting this variable from our models creates a positive bias that pushes the `rating` coefficient away from zero, causing us to overestimate the relationship between consumer rating and application success.

9.2.2 Application Rankings

An internal factor that could affect application success is the business logic defined by Google, specifically how the company ranks applications and lets users explore products. We believe this variable can be operationalized through a numeric value representing an application's rank within its respective category. Application rankings should positively influence success, as we expect consumers to discover highly ranked applications more easily. Assuming Google uses rating as a metric for defining an application's rank, ranking and rating will also be positively correlated. As a result, the omitted-variable bias is positive, pushing the coefficient for `rating` further from zero.

9.2.3 Total Addressable Market

The total addressable market relates to the innate opportunity contained in an application (measured in revenue or in the number of available consumers), which varies according to its functionality. In general, we believe this variable is positively correlated with application success, as larger markets tend to have greater potential to drive downloads. We do not anticipate a strong association with consumer rating, though we do expect a positive one; the bigger the market, the more crucial it is to build a differentiated product that offers quality features. Omitting this variable also creates a positive bias that inflates the coefficient for `rating` and leads us to overestimate its effect on application success.
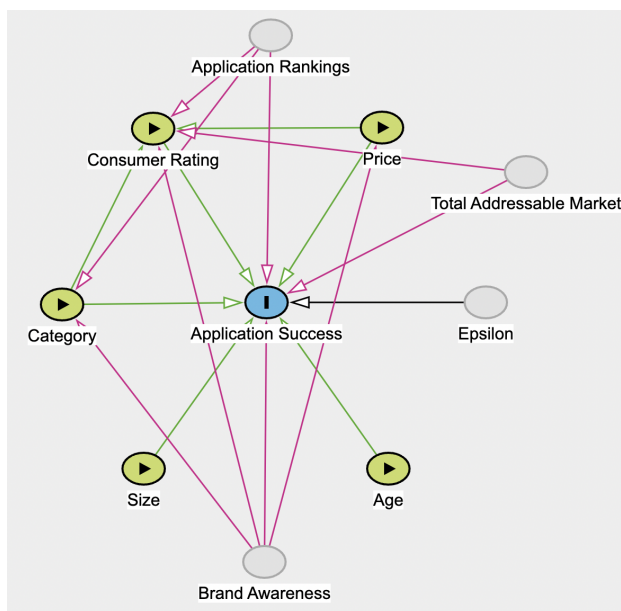

Figure 2: A revised version of the causal diagram including omitted variables.

10 Conclusion

The goal of this analysis was to investigate the causal factors of application success. Specifically, we sought to assess our hypothesis that consumer rating has a positive influence on application success, as understood by the number of downloads an application receives. Our causal model included a total of five independent variables, though we identified additional omitted variables that could have impacted our results. We used a cross-sectional data set from Kaggle, containing records of 10,000 applications from the Google Play Store in 2019. After exploring, cleaning, and transforming the data, we created three regression models and interpreted their coefficients.

The models we produced confirm that applications with higher consumer ratings are more successful than those with poor ratings. It is important to note, however, that we have not proven the existence of a causal

link, but merely provided evidence in favor of one. Whatever the case, it is clear that other factors besides rating also contribute to application success. For instance, we also noticed that free applications, applications with many versions, and large applications perform better than their counterparts. Unfortunately, our models have limited predictive power (adjusted $R^2 = 0.247$), showing that our analysis has room for improvement. Further studies may seek to bring in additional variables in an effort to capture new aspects of the problem. New data could also help mitigate the issues we faced, such as omitted-variable bias, irregularities in the data, and imprecision in the outcome variable. We believe that this study is valuable to application developers, as it identifies key variables that may influence success. Our findings may also be useful to Google and other moderators of mobile applications, who share an interest in understanding the dynamics of consumer environments.

11 Appendix

1. Lab 2 Repository