Lab 2 Research Proposal

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 in the United States is spent on mobile applications. As digital applications become one of the principal media through which organizations engage with their customers, it becomes increasingly valuable to understand the drivers that lead to greater usage and better customer experience.

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 a linear model that assesses the importance (or lack of importance) of consumer rating on the number of downloads, with additional variables such as price, application size, and application category serving as controls. If we confirm the existence of a causal pathway, it would signal that application developers should invest heavily in improving their ratings, for instance by buying positive reviews from consumers. Our study may also be of interest to Google, who has a vested interest in preventing developers from unfairly inflating their ratings.

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 brand awareness from marketing campaigns, differences across customer bases, seasonality, and production value, among possible others. In spite of these limitations, our study should provide useful insight into the factors that cause an application's success.

The paper is structured as follows. Section 2 outlines our research question, the causal model we will use to contextualize our regression analysis, and the research design. Section 3 describes our data, the explanatory variables we will include in the modeling phase, and the transformations we will apply. Section 4 contains our statistical models, and Section 5 discusses their significance as well as their statistical validity. In Section 6, we present our conclusions and discuss the implications of our results.

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 higher consumer rating lead to greater success 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 also identify several omitted variables that could impact our results.

3.1 Simple Causal Model

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.

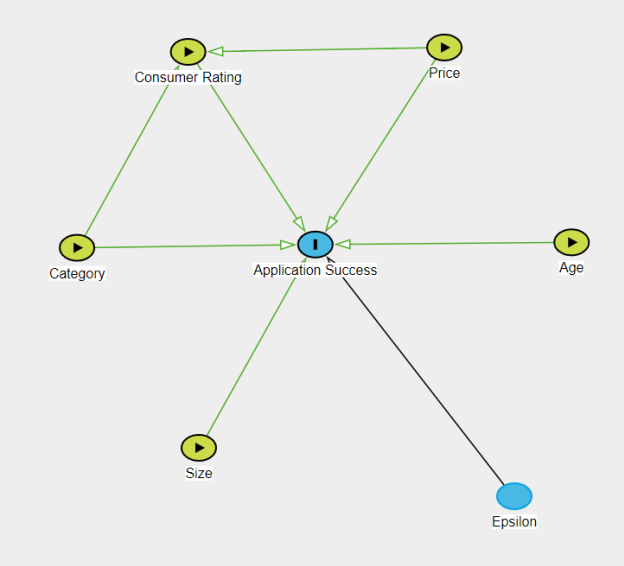


Figure 1: A hypothetical causal diagram for Google Play Store applications.

3.1.1 Application Success

The goal of our analysis is to identify causal factors of application success, and more specifically to assess whether consumer rating has a positive effect on application success. There are different ways in which we could operationalize application success; our study will 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 or social impact, it is effective in its simplicity and appropriate for this study.

3.1.2 Consumer Rating

The main independent variable of interest is consumer rating. We hypothesize that higher ratings should cause greater application success, because an application that has been highly rated is one that has been deemed worthwhile by other users. When consumers decide whether or not to download an application, they will likely trust the opinions of their peers and download applications 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 the application whereas consumer rating is decided once the 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 ratings and downloads both have positive effects on one another, however, our models may suffer from positive feedback. We will discuss the implications of this effect in Section 5.

3.1.3 Price

We expect price to also have a causal effect on application success. Overall, we believe that paid applications should be more successful than free applications because they will have more appealing features. Of course, this may not always be the case, as free applications could also have impressive features and paid applications

may lazily try to cash out on naive users.

We also anticipate a causal pathway from price to rating, for similar reasons as the pathway from price to success. Specifically, we believe that consumers are likely to rate paid applications positively since those applications have more desirable features, whereas free applications will be reviewed more negatively by comparison. Again, this relationship could lean in the opposite direction if our assumption about the connection between price and application quality is incorrect.

3.1.4 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 a smaller subset 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 that may lead users to feel frustrated, such as games and dating applications, may also receive more negative reviews on average.

3.1.5 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 (1200 to 1000), it would seem as if the older application was 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 far 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.

3.1.6 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.

3.1.7 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, the type of device (Android, iPhone, etc.) used to download an application, 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.

3.2 Omitted Variables

The true causal diagram is undoubtedly more complex than the one we have outlined above. We have identified several omitted variables that could affect our statistical models and any conclusions we infer from them, 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.

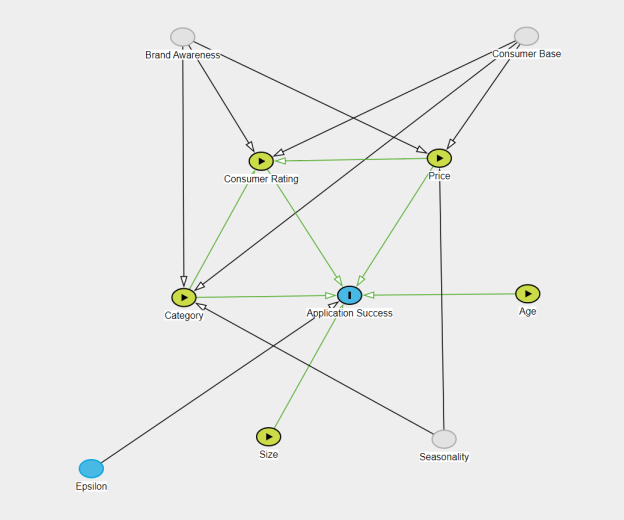


Figure 2: A revised version of the causal diagram including omitted variables.

Brand Awareness We believe that a lift in brand awareness leads to an increase in install count. We'd expect this is largely generated accumulated through marketing campaigns across varying channels. Many applications likely have higher install counts from being a large player within an industry as well as spending money to direct prospects to their digital product and/or directly to the app store. This would be possible to measure, but is not available within this dataset. With respect to our independent variables, we'd most expect this to be correlated with certain sublabels within the category variable. As such, for some sublabels this would cause a omitted variable bias of away from zero as the included variable may be more dramatic than otherwise would be.

Differences in Consumer Bases We think of this concept in terms of how the customer base engages with a product as well as who the customer base is. For example, some products are more easily translated into the size and interactive capabilities of a smartphone; likely incentivizing downloads for those types of apps. Some customer bases may also be substantially bigger acting as a massive customer funnel into accumulating downloads. Note, customer size is likely correlated with brand awareness, but we don't expect there to be perfect collinearity among them. Additionally, there may be demographic differences that lend themselves more to using mobile applications. In aggregate, we see this is a very vague bucketed concept that likely has interactions across both the dependent and indendent variables within our modeled causal graph.

Seasonality Seasonality can influence consumer demand towards specific app categories. For example, health applications are likely more popular during the beginning of the year when new year resolutions are top of mind. We'd expect this to be largely related to the category of the application and likely a factor that does not fit well within this research design. We believe it would need to be modeled and descibed through time series data whereas our dataset is based on a snapshot in time.

4 Data

To answer our research question, we will leverage publicly available data about applications available on the Google Play Store. This data was randomly scraped from the Google Play Store interface and uploaded to Kaggle.com 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, the fields in our data set do not map exactly onto these variables, so we will approximate them using the following operationalizations:

Dependent variable

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

Independent variables

- Consumer rating — `rating` (the average consumer rating for the 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+) `size` — The memory space occupied by the application.
- Age — `current_version` (the current version number of the application) and `last_updated` (the date when the application was last updated)
- Size — `size` (the download size of the application, in units of MB)

5 Research Design

The goal of this analysis is to determine the effect of consumer ratings on the success of Google Play Store applications. We hypothesize that higher consumer ratings lead to greater application success. To moderate and refine our analysis, we include 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 (listed above). Although these mappings are sometimes imperfect, we believe they are sufficient for a meaningful analysis.

Before we proceed with building any 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 installations and consumer ratings. The third and final model explores interactions between `rating` 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 variables, we built a few rules to filter clean data based on some logical conclusions. Specifically it included; removing duplicates, removing records with a null review count, and removing records with a consumer rating value of larger than 5. Our research question focuses on consumer rating, thus we elected to only keep records with a valid value for it. We intuitively do not believe the consumer rating score is representative if there are less than 100 reviews, but, we wanted to first understand the distribution before making a potentially dramatic reduction in sample size of the data set. After performing these initial operations, the cleaned data set includes 7,226 rows (i.e applications) with 24 metadata columns (11 of which we built). We split up the exploration into 2 sections based on variable type; numeric versus categorical.

In summary, the EDA

6.1 Numeric Variables

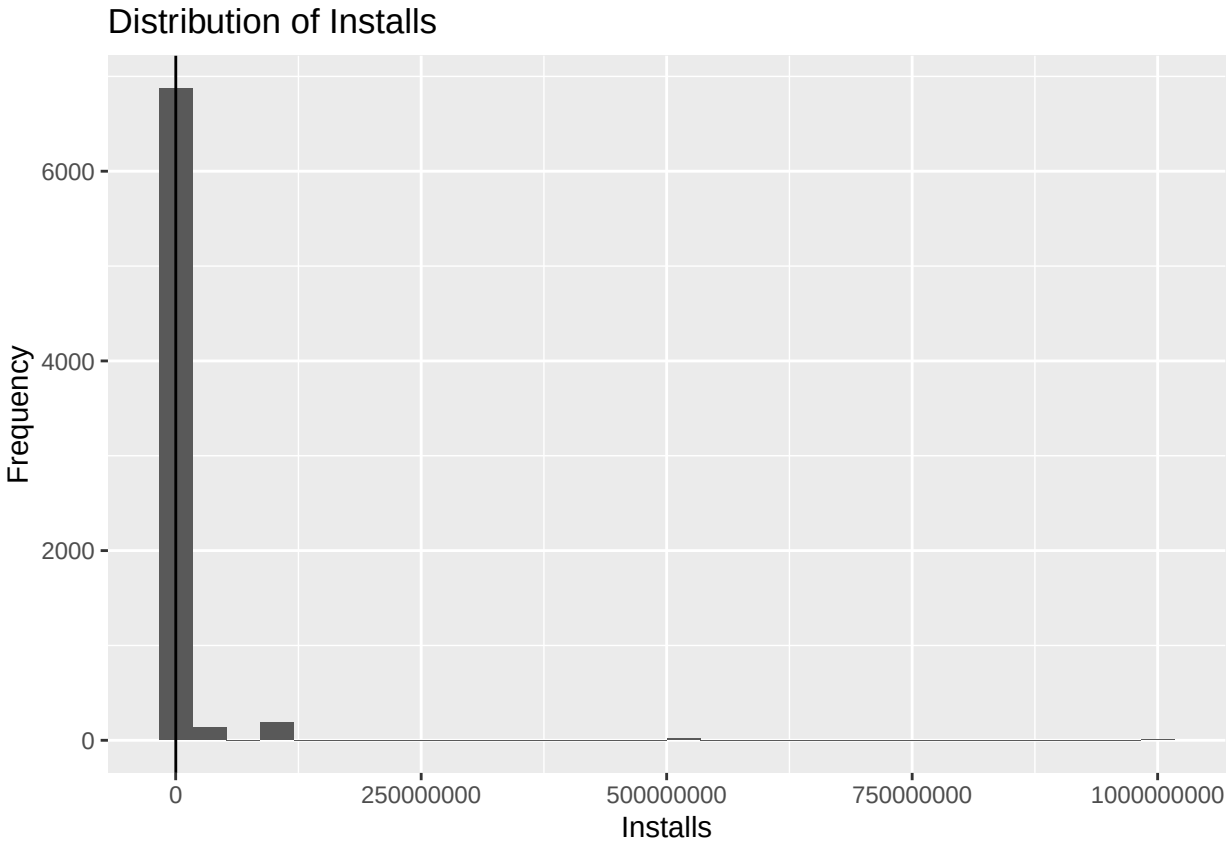
For the numeric variables, we wanted to understand the distribution of each as well as examine the correlations/covariances among them. The distribution is used to measure the quality of the data, understand outliers, and evaluate the need for variable transformations. Understanding correlations help us by highlighting variables that may strongly explain the variance of our dependent variable as well as quantifying the collinearity among our independent variables (which we want to avoid). When evaluating the correlation coefficients we used the rule of thumbs where $>= 0.9$ is great and $>= 0.6$ is okay.

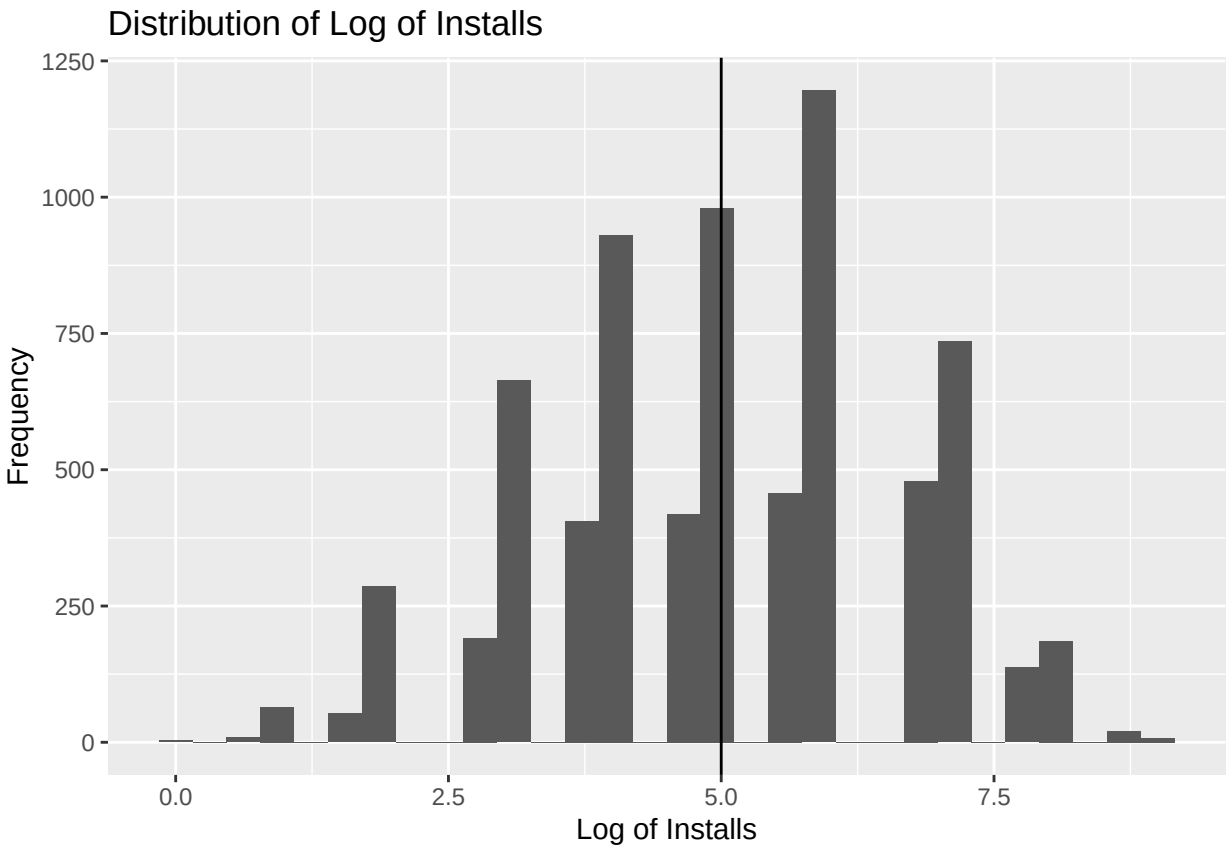
6.1.1 Distribution of Variables

```
##     installs               size           reviews             rating
##  Min.   :         1   Min.   :  0.85   Min.   :       1   Min.   :1.000
##  1st Qu.:     10000   1st Qu.:  5.80   1st Qu.:      97   1st Qu.:4.000
##  Median :    100000   Median : 15.00   Median :    2039   Median :4.300
##  Mean   :   7646081   Mean   : 24.30   Mean   :  279761   Mean   :4.171
##  3rd Qu.:   1000000   3rd Qu.: 35.00   3rd Qu.:   35930   3rd Qu.:4.500
##  Max.   :1000000000   Max.   :100.00   Max.   :44893888   Max.   :5.000
##      price
##  Min.   :  0.000
##  1st Qu.:  0.000
##  Median :  0.000
##  Mean   :  1.139
##  3rd Qu.:  0.000
##  Max.   :400.000
```

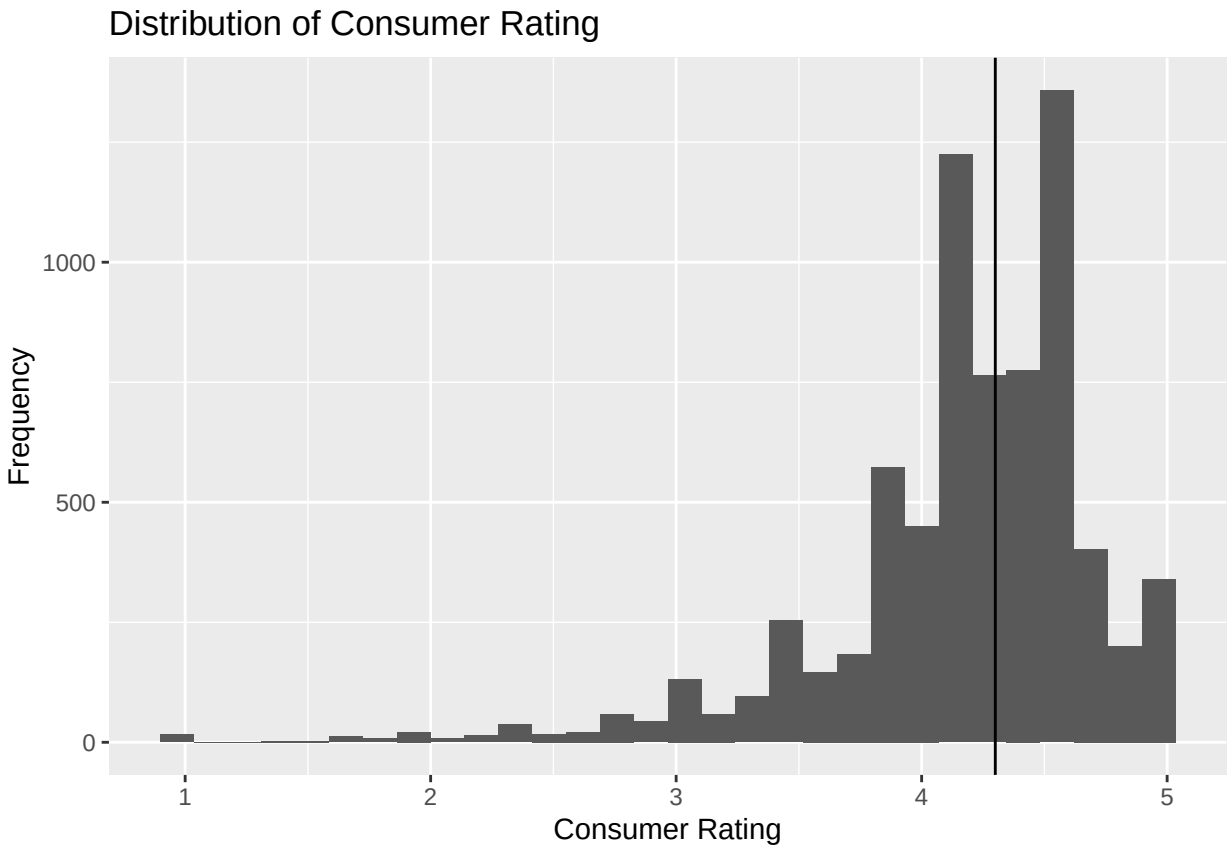
The column values all seem valid given there were no negative values (i.e negative review count) or a number near infinity. Notably, none contained any zero values which would be an important note if we'd want to take the log transformation of a variable.

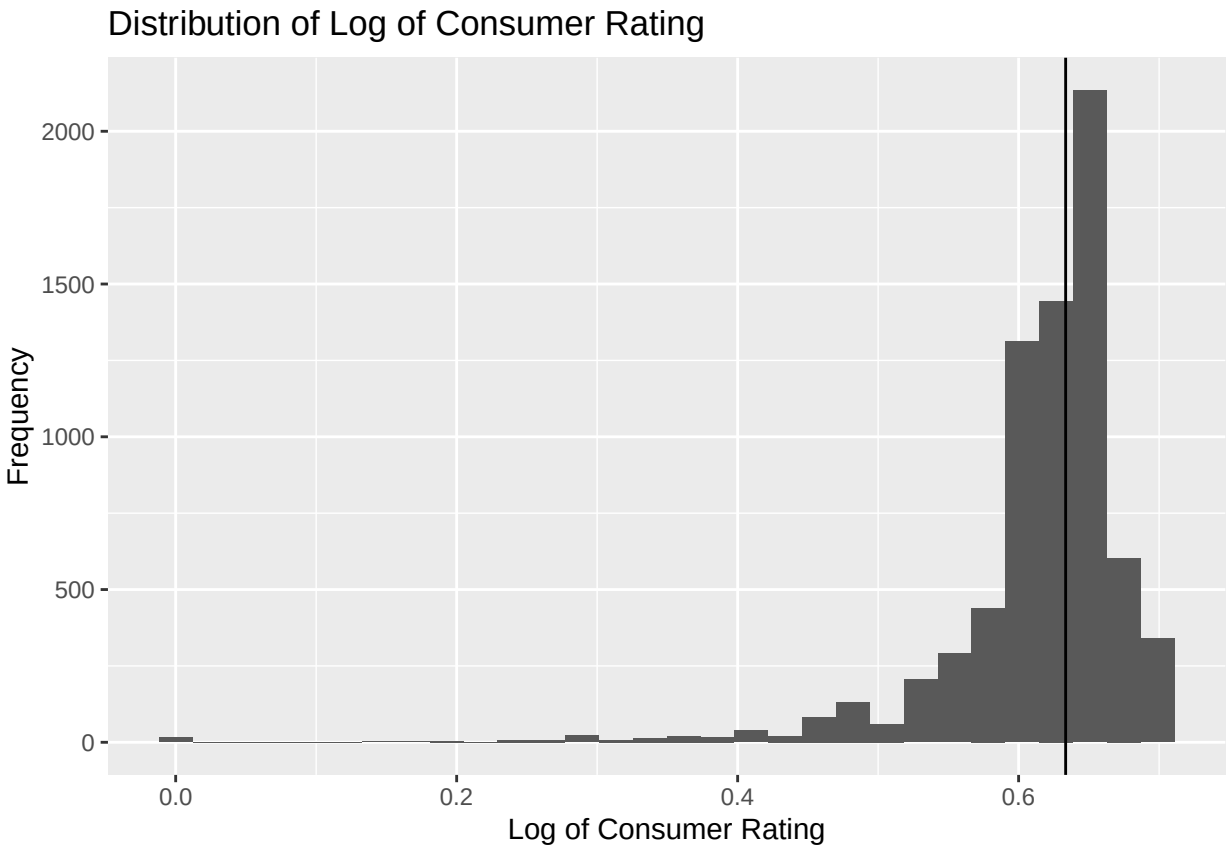
Based on the difference across medians and means, size and rating count appear to have more normal distributions while the others look more conducive to having a long tail in either end of their spectrum. Notably, we expect there to be a reverse causal link between install count and review count (i.e high install count leads to review count). Based on this we elected to not include review count as an independent variable. Based on the distribution our initially idea of only keeping applications with at least 100 reviews seems sensible as it will only remove 25% of the data which is okay given the initial size of over 7k apps.


Distribution of Installs
Frequency
6000
4000
2000
0
0
250000000
500000000
750000000
1000000000
Installs

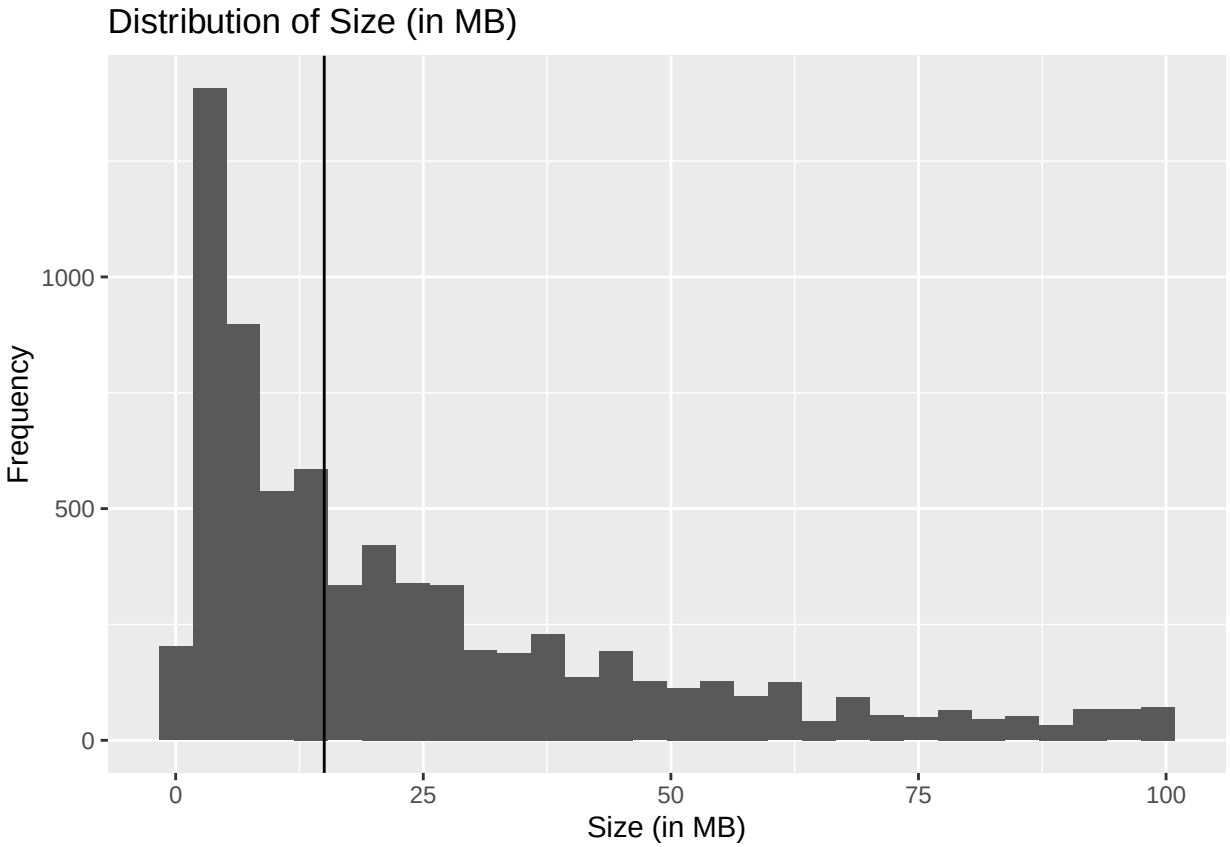


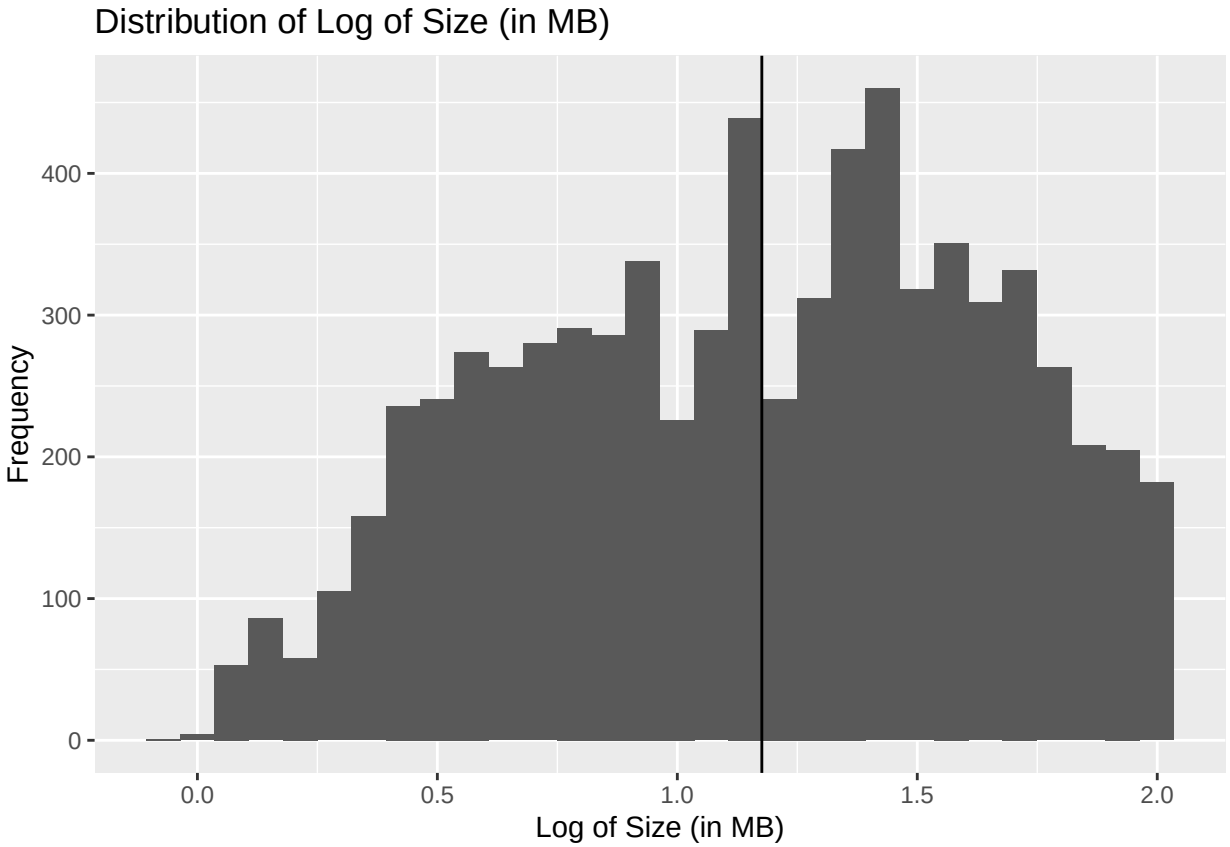
Application success — `installs`: Important to note is the installs count variable is binned. The bins start at one and scale logarithmically; 1+, 5+, 10+, 100+, 500+, 1000+, etc (i.e 100+ means there are between 100 and 499 installs for that app). We simply removed the `+` and made treated it as a metric variable. We know the variable is ordinal as there is a clear ordering to it. We concluded it is metric because there is a measurable distance between groups. And, given the distance between groups scales logarithmically, we can claim there is a constant distance across groups. There is some error with precision as the bin hides the true value, but we believe this variable can be considered metric in practice. There is a heavy tail towards the end of the spectrum as the maximum is 1 billion with a median of 100,000. Unsurprisingly, taking a log of the variable allows resemble a normal distribution making this an appropriate variable transformation for the model.


Distribution of Consumer Rating
Frequency
1000
500
0
1
2
3
4
5
Consumer Rating

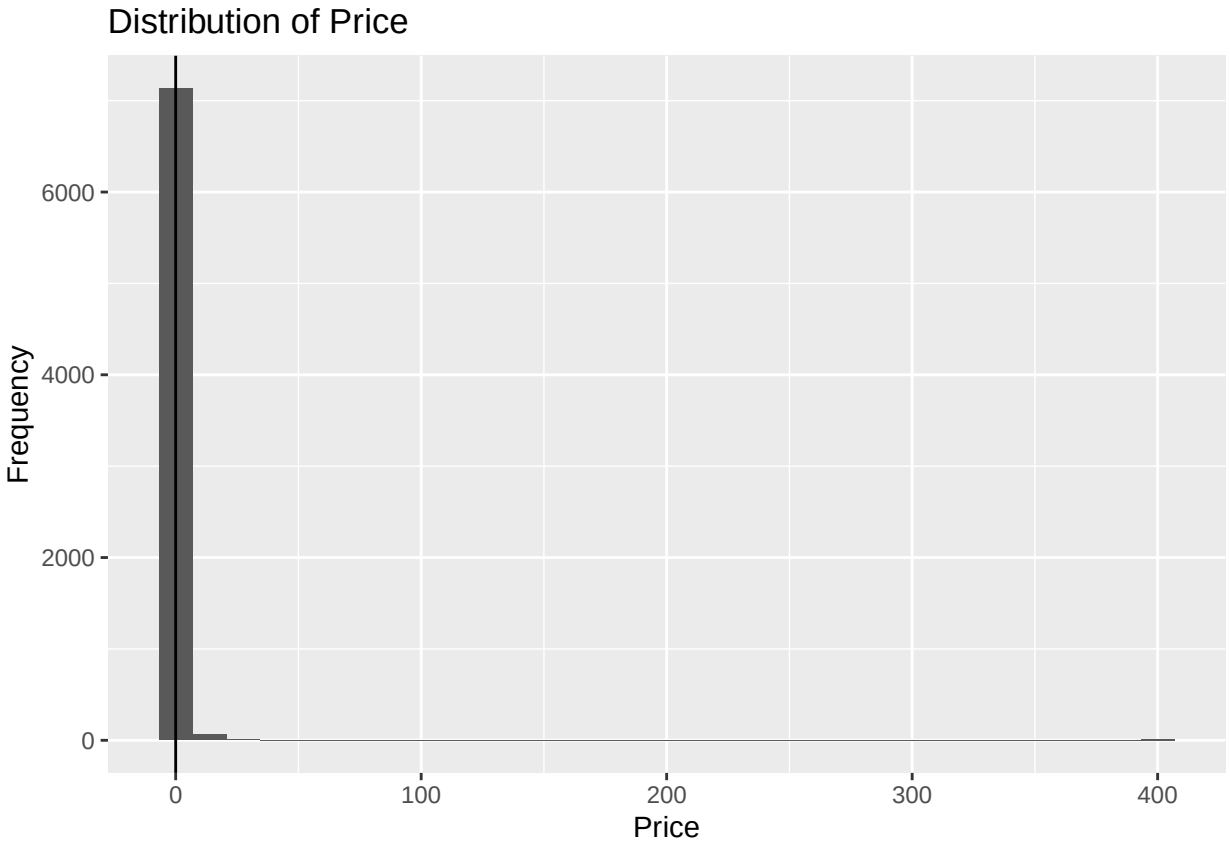


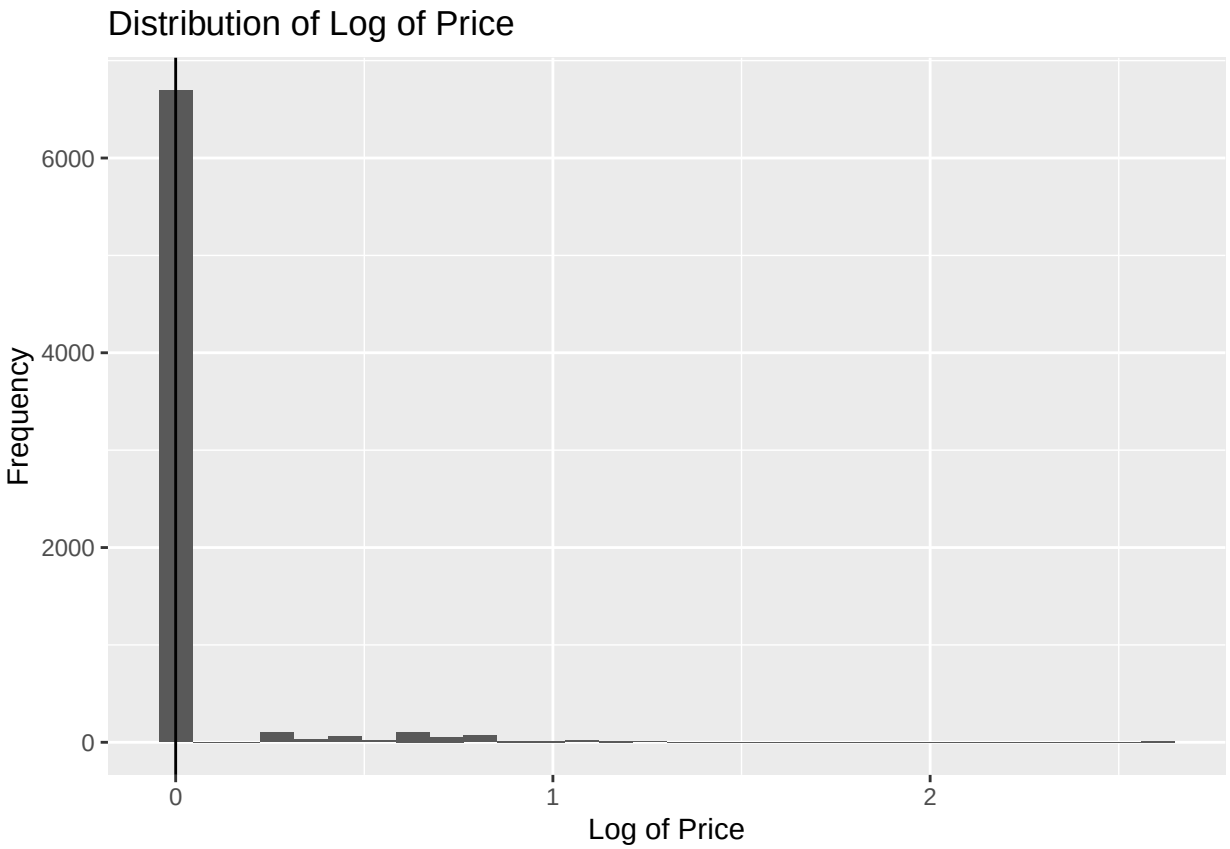
Consumer rating — `rating`: The rating variable appears to have a distribution that appears approximately normal that is skewed towards the right of the valid values between 0 and 5 with a median of 4.3.


Distribution of Size (in MB)
Frequency
1000
500
0
0
25
50
75
100
Size (in MB)



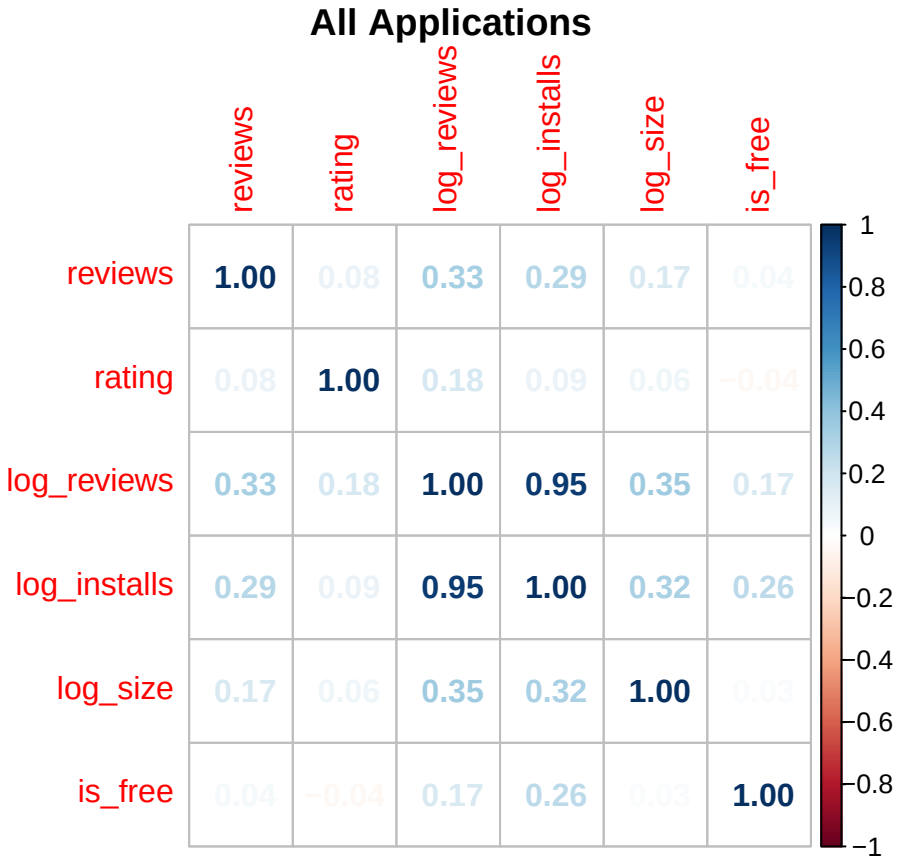
Size — `size`: The size variable had slightly more normal distribution than installs count, but still had a heavy tail towards the end of its spectrum. The log of size more closely resembles a normal distribution leading us to believe it will be an appropriate variable transformation. Note, the log(size) value does not have a very pronounced high density point near the mean/median, but we believe it still is appropriate when comparing it to the distribution of the non-logged version.

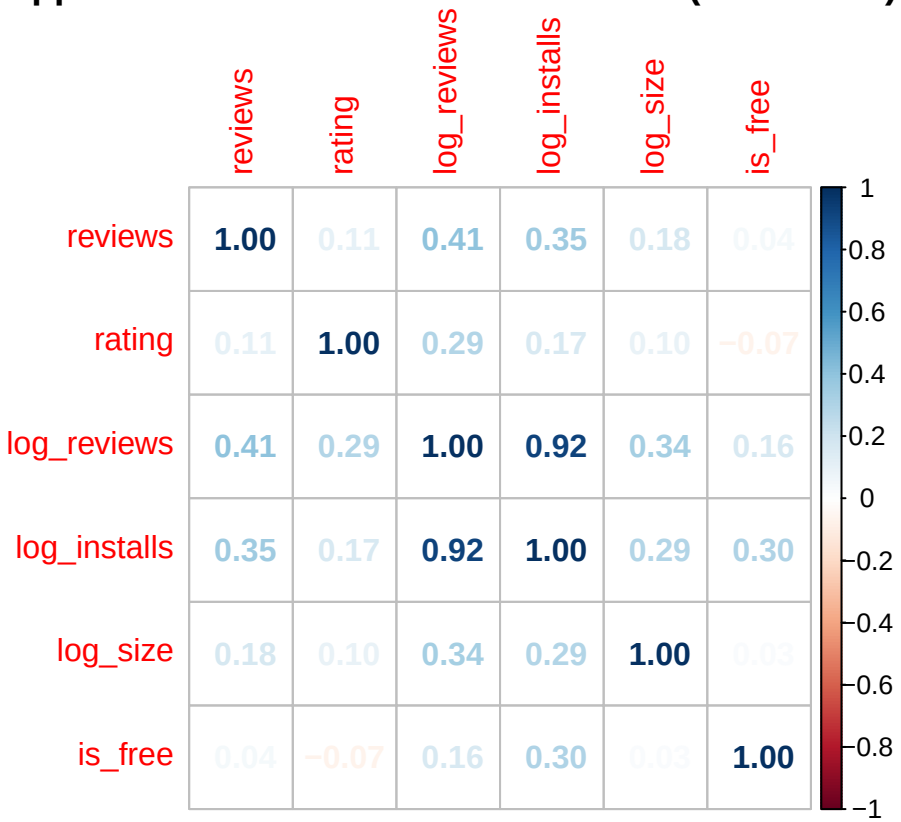

Distribution of Price
Frequency
6000
4000
2000
0
0
100
200
300
400
Price



Price — `price`: 93% of the applications are free. This is why believe both distributions of price and log(price) are undesirable where neither mimic a normal distribution at all. This finding lead to us exploring the concept of a binary metric variable where 1 means it is paid and 0 means it is not paid.

6.1.2 Correlation of Variables





Aside from the reviews variable, none of the other numeric variables have a strong correlation with install count. The high coefficient of 0.9 between log_reviews and log_installs supports the hypothesis of a reverse causal link between reviews and installs. With the goal of building an efficient model, we'd like to see stronger correlation coefficients between the variables, but there may be predictive power in the interaction between them. Of note, we do see the coefficient between rating and log_installs increase by 2x when we exclude apps with less than 100 reviews. This agrees with our hypothesis that rating does have an influence on install count and that the rating value is more valid as it accrues at least 100 reviews. Lastly, the numeric variables seem to support the statistical assumption of there not being perfect collinearity among independent variables.

6.2 Categorical Variables

To explore the categorical variables we wanted to aggregate the apps by variable sublabel and look at the frequency and mean log of install count across each. We are okay using mean as a measure of central tendency opposed to the median given the approximately normal distribution of the log of install count variable. To understand an which of the categorical columns has the largest dispersion across sublabels we calculated the a quantile table based on the `Log of Install Count (Avg)` variable. Note, we excludes sublabels with a count of less than 100 apps.

Table 1: 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 2: SubLabel Summary Table (Category Variable)

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 3: SubLabel Summary Table (Content Rating Variable)

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

Based on the quantile summary table, all 4 categorical variable have at least a 20% dispersion across the minimum and maximum sublabels. This indicates that they might all have some predictive power and be worth exploring as inputs. Notably, the category column seems to have the largest dispersion across groups indicating it may be the most impactful. We've omitted the tables for the `current_version` and `android_version` as they are less readable. Overall, this supports our hypothesis of testing these as inputs. We also want to play around with creating binned/binary variables from them to isolate high frequency sublabels or those with high average log install count.

7 Statistical Models

> Lab 2 Intructions: You will next build a set of models to investigate your research question, documenting your decisions. Here are some things to keep in mind during your model building process: 1. *What do you want to measure*? Make sure you identify one, or a few, variables that will allow you to derive conclusions relevant to your research question, and include those variables in all model specifications. How are the variables that you will be modeling distributed? Provide enough context and information about your data for your audience to understand whatever model results you will eventually present. 2. What covariates help you achieve your modeling goals? Are there problematic covariates? either due to *collinearity*, or because they will absorb some of a causal effect you want to measure? 3. What *transformations*, if any, should you apply to each variable? These transformations might reveal linearities in the data, make our results relevant, or help us meet model assumptions. 4. Are your choices supported by exploratory data analysis (*EDA*)? You will likely start with some general EDA to *detect anomalies* (missing values, top-coded variables, etc.). From then on, your EDA should be interspersed with your model building. Use visual tools to *guide* your decisions. You can also leverage statistical *tests* to help assess whether variables, or groups of variables, are improving model fit. At the same time, it is important to remember that you are not trying to create one perfect model. You will create several specifications, giving the reader a sense of how robust (or sensitive) your results are to modeling choices, and to show that you're not just cherry-picking the specification that leads to the largest effects. At a minimum, you need to estimate at least three model specifications: The first model you include should include *only the key variables* you want to measure. These variables might be transformed, as determined by your EDA, but the model should include the absolute minimum number of covariates (usually zero or one covariate that is so crucial it would be unreasonable to omit it). Additional models should each be defensible, and should continue to tell the story of how product features contribute to product success. This might mean including additional right-hand side features to remove omitted variable bias identified by your casual theory; or, instead, it might mean estimating a model that examines a related concept of success, or a model that investigates a heterogeneous effect. These models, and your modeling process should be defensible, incremental, and clearly explained at all points. Your goal is to choose models that encircle the space of reasonable modeling choices, and to give an overall understanding of how these choices impact results.

Based on our exploratory data analysis we decided on various interaction terms and transformations to use in our linear model. Our focus was to maximize its prediction accuracy (R^2) as well as maintaining a model that is explainable.

```
model_small  <- lm(log_installs ~ 1 + rating, data = data_clean)
model_medium <- lm(log_installs ~ 1 + rating + log_size + log_current_version +
                     log_last_updated + is_free + is_family_category +
                     is_game_category + is_tools_category + is_content_everyone,
                   data = data_clean)
model_large  <- lm(log_installs ~ 1 + rating + log_size + log_current_version +
                     log_last_updated + is_free + is_content_everyone +
                     rating * is_family_category + rating * is_game_category +
                     rating * is_tools_category,
                   data = data_clean)

# plot(model_small)
# plot(model_medium)
# plot(model_large)
```

```
stargazer(
  model_small,
  model_medium,
  model_large,
  header = FALSE,
  type = "latex",
  se = list(get_robust_se(model_small), get_robust_se(model_medium),
            get_robust_se(model_large)),
  column.sep.width = "3pt",
  font.size = "small"
)
```

8 Results

Lab 2 Intructions: You should display all of your model specifications in a regression table, using a package like stargazer to format your output. It should be easy for the reader to find the coefficients that represent key effects near the top of the regression table, and scan horizontally to see how they change from specification to specification. Make sure that you display the most appropriate standard errors in your table. In your text, comment on both *statistical significance and practical significance*. You may want to include statistical tests besides the standard t-tests for regression coefficients. Here, it is important that you make clear to your audience the practical significance of any model results. How should the product change as a result of what you have discovered? Are there limits to how much change you are proposing? What are the most important results that you have discovered, and what are the least important?

9 Model Limitations

9.1 Statistical Limitations

Lab 2 Intructions: As a team, evaluate all of the large sample model assumptions. However, you do not necessarily want to discuss every assumption in your report. Instead, highlight any assumption that might pose significant problems for your analysis. For any violations that you identify, describe the statistical consequences. If you are able to identify any strategies to mitigate the consequences, explain these strategies. Note that you may need to change your model specifications in response to violations of the large sample model.

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

Table 4:

	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

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 any of the predictor variables. That is, the model does not systematically underpredict or overpredict in certain regions of the input space. Figures show the relationships between the model residuals and metric-scale predictors. The residuals are generally well-centered around zero, although the model seems to underpredict when `log_reviews` is high and `rating` is low. The fourth plot shows the model residuals as a function of the model predictions. Here, the model seems to underpredict in the left-most and right-most regions, and slightly overpredict in the middle. Overall, there are no strong non-linear relationships between the model residuals and the input features, so we do not find enough evidence to reject the assumption of linear conditional expectation.

9.1.4 Homoskedastic Errors

When assessing homoskedastic errors, we seek to determine if there is a relationship between the variance of the model residuals and the predictors. If the homoskedastic assumption is satisfied, then we should observe a lack of relationship; conversely, if the data are heteroskedastic then the conditional variance will depend on the predictors. The first plot 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 data, although the right-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 our 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.

9.1.5 Normally Distributed Errors

When assessing the normality of the error distribution, we seek to determine if the model residuals are approximately Gaussian. If so, then the sample quantiles of the residuals should closely match the theoretical quantiles of a normal distribution in a Q-Q plot. Below, we plot the Q-Q plot associated with our model. In general, the residuals seem to follow a normal distribution, as the middle quantiles match the corresponding

theoretical quantiles. However, the tails of the residual distribution are fatter than expected; the first quantiles occur at smaller than expected values, and the last quantiles occur at larger than expected values. Overall, the assumption of normally distributed errors seems imperfect but reasonably justified.

Because our data fails to meet the assumption of homoskedasticity, we adopt the large-sample assumptions (assumptions 1 and 2) instead. Specifically, we report robust standard errors rather than non-adjusted errors in our results.

9.2 Structural Limitations

Lab 2 Intructions: What are the most important *omitted variables* that you were not able to measure and include in your analysis? For each variable you name, you should *reason about the direction of bias* caused by omitting this variable and whether the omission of this variable calls into question the core results you are reporting. What data could you collect that would resolve any omitted variables bias?

10 Conclusion

Lab 2 Intructions: Make sure that you end your report with a discussion that distills key insights from your estimates and addresses your research question

11 Appendix