# Predicting App Success on Google Play Store

BY: LAUREL STEWART, YUNJIU LI

## Objective and Motivation

**Goal:** Predict App Success (installs per month) using app level data from google play store

#### **Motivation:**

- 1. Android dev cost = high, revenue = uncertain.
- Users spend lesser than iOS users.
- 3. Device fragmentation makes support costly.
- 4. Use data from Google Play Store to understand what drives app success.

#### **Data Collection**

 Data Source: Combination of preexisting datasets (Kaggle) and custom web scraping using Apify and google-play-scraper, originally more than 40 columns.

#### • Scope:

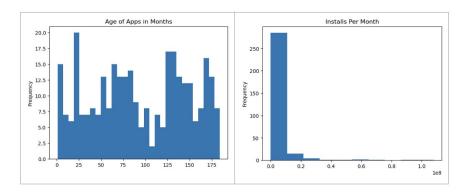
- 1. 316 apps across 10 popular categories.
- Excluded: Google-owned apps, major media brands, social networks
- **Dataframe:** 27 columns, cleaned and encoded.

Removed in processing	Feature	Description  App title/name					
	title						
	description	Full app description					
	descriptionHTML	HTML formatted description					
	summary	Brief app summary					
х	installs	Install count range					
x	minInstalls	Minimum number of installs					
	realInstalls	Actual install count					
	score	App rating score					
	ratings	Number of ratings					
	reviews	Number of reviews					
Х	histogram	Rating distribution histogram					
	price	App price					
	free	Whether app is free					
	currency	Price currency					
	sale	Whether app is on sale					
	saleTime	Sale duration/timing					
	originalPrice	Original price before sale					
	saleText	Sale description text					
	offersIAP	Offers in-app purchases					
	inAppProductPrice	In-app purchase pricing					
х	developer	Developer name					
х	developerId	Developer identifier					

developerEmail	Developer contact email
developerWebsite	Developer website URL
developerAddress	Developer address
privacyPolicy	Privacy policy URL
genre	App genre/category
genreld	Genre identifier
categories	App categories
icon	App icon image
headerImage	Header/banner image
screenshots	App screenshot images
video	Promotional video
videolmage	Video thumbnail image
contentRating	Age/content rating
contentRatingDescription	Content rating details
adSupported	Whether app shows ads
containsAds	Contains advertisements
released	Release date
lastUpdatedOn	Last update date
updated	Update timestamp
version	App version number
comments	User comments/reviews
appld	Unique app identifier
url	Google Play Store URL
	developerWebsite developerAddress privacyPolicy genre genreld categories icon headerImage screenshots video videoImage contentRating contentRatingDescription adSupported containsAds released lastUpdatedOn updated version comments appld

## Data Cleaning and Pre-Processing

- Removed columns (Missing Values):
  - redundant or unavailable data (e.g., sale time and original price).
- Feature adjustments:
  - Create InstallsPerMonth to normalize across different app ages
- Boolean & Categorical Conversion:
  - Convert T/F -> 1/0
  - One-hot encoded genre and other categories

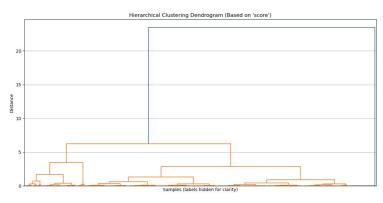


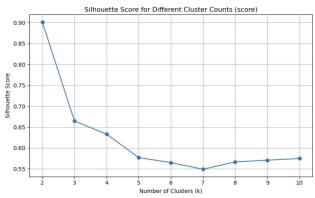
## Feature Engineering

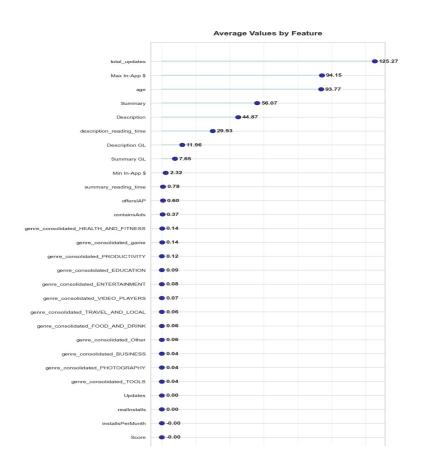
- Key Features Created
- Installs per month = installs/age.
- 2. App age(in months),
- 3. Update frequency
- Format Changes
- 1. Boolean -> 0/1,
- genre -> dummy variables,
- Tried but removed
- 1. Readability scores of descriptions
- 2. Clustering-based features (didn't help)

```
# Feature Engineering
df['log_installsPerMonth'] = np.log1p(df['installsPerMonth']) # Log-transformed target
df['iap_price_range'] = df['maxInAppProductPrice'] - df['minInAppProductPrice'] # Price range
df['score_to_age_ratio'] = df['score'] / (df['age'] + 1) # Score normalized by age
df['readability_blend'] = (df['description_readability_score'] + df['summary_readability_score']) /
2 # Combined readability
df['update_density'] = df['total_updates'] / (df['age'] + 1) # Updates per unit age
df['score_reading_interact'] = df['score'] * df['description_reading_time'] # Interaction term
```

## Feature Engineering







### Data Analysis

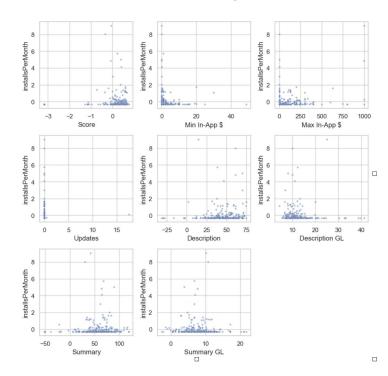
#### Visuals:

- Histograms of installs/month.
- Box plots of installs by genre.

#### Key Insights:

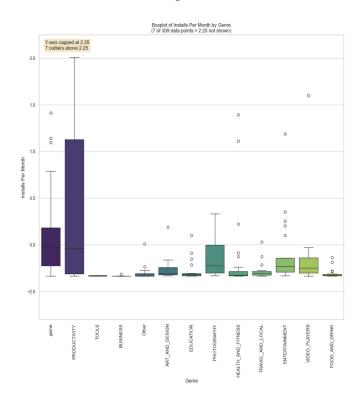
- Target variable highly skewed.
- Significant variance across genres.
- Correlations revealed data leakage (realInstalls).

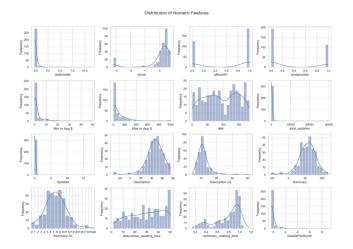
#### Scatter Plots of Features vs. Target Variable

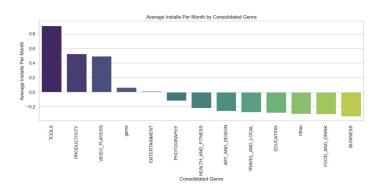


0

## Data Analysis







## Data Analysis

#### Correlation Matrix of Numeric Features

realInstalls	1.00	0.05	0.02	0.01	-0.09	0.33	0.23	-0.01	-0.01	0.03	0.02	0.02	-0.01	-0.07	0.05	0.94
Score	0.05	1.00	0.29	0.21	0.05	0.15	0.39	0.03	0.03	0.14	-0.05	0.07	-0.08	0.36	0.04	0.06
offersIAP	0.02	0.29	1.00	0.31	0.39	0.47	0.03	0.04	0.05	-0.05	-0.01	0.05	-0.06	0.34	0.03	0.00
containsAds	0.01	0.21	0.31	1.00	-0.11	0.06	0.17	-0.05	-0.05	0.08	-0.10	0.09	-0.08	0.23	0.11	-0.02
Min In-App \$	-0.09	0.05	0.39	-0.11	1.00	0.25	-0.22	-0.02	-0.01	-0.01	-0.04	0.04	-0.05	0.01	-0.05	-0.05
Max In-App \$	0.33	0.15	0.47	0.06	0.25	1.00	0.07	0.00	0.00	-0.16	0.09	0.03	-0.05	0.12	-0.02	0.27
age	0.23	0.39	0.03	0.17	-0.22	0.07	1.00	-0.04	-0.05	0.10	0.00	0.12	-0.10	0.28	0.12	0.14
total_updates	-0.01	0.03	0.04	-0.05	-0.02	0.00	-0.04	1.00	1.00	0.07	-0.05	-0.01	0.01	-0.03	0.00	0.01
Updates	-0.01	0.03	0.05	-0.05	-0.01	0.00	-0.05	1.00	1.00	0.07	-0.05	-0.01	0.01	-0.04	-0.00	0.01
Description	0.03	0.14	-0.05	0.08	-0.01	-0.16	0.10	0.07	0.07	1.00	-0.90	0.27	-0.26	-0.05	0.04	0.06
Description GL	0.02	-0.05	-0.01	-0.10	-0.04	0.09	0.00	-0.05	-0.05	-0.90	1.00	-0.20	0.20	0.02	-0.01	-0.01
Summary	0.02	0.07	0.05	0.09	0.04	0.03	0.12	-0.01	-0.01	0.27	-0.20	1.00	-0.98	0.06	-0.08	0.03
Summary GL	-0.01	-0.08	-0.06	-0.08	-0.05	-0.05	-0.10	0.01	0.01	-0.26	0.20	-0.98	1.00	-0.04	0.23	-0.02
description_reading_time	-0.07	0.36	0.34	0.23	0.01	0.12	0.28	-0.03	-0.04	-0.05	0.02	0.06	-0.04	1.00	0.28	-0.09
summary_reading_time	0.05	0.04	0.03	0.11	-0.05	-0.02	0.12	0.00	-0.00	0.04	-0.01	-0.08	0.23	0.28	1.00	0.00
installsPerMonth	0.94	0.06	0.00	-0.02	-0.05	0.27	0.14	0.01	0.01	0.06			-0.02		0.00	1.00
ealth	galls s	offere	dontains Contains	Ads Min In A	Nat In A	<sup>5</sup> 63	otal Jud	ates upo	descrip	ion Jescription	GL SUM	Summan	Sunnary Sunnary	reading.	ine states of the	Inth
											,	escription	Summan	, lu	,	

- 0.75 - 0.50 - 0.25 - 0.00 - -0.25 - -0.50 - -0.75

## Modeling Approaches

#### **Models Tested:**

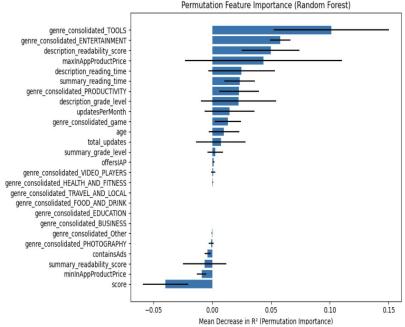
- Linear Regression
- Random Forest Regressor
- Gradient Boosting Regressor

#### Initial Results (with leakage):

 Unrealistically high accuracy due to reallnstalls included.

#### **After Fixing:**

- Gradient Boosting performed best (R<sup>2</sup> = 0.1112).
- Linear & RF underperformed (R<sup>2</sup> < 0).</li>



Model	R² Score	RMSE			
Linear Regression	-0.081 2	1.173 3			
Random Forest Regressor	-0.198 3	1.235 2			
Gradient Boosting Regressor	0.1112	1.063 8			

## **Key Findings**

- Most apps don't get many downloads.
- App genre affects installs, but not in a clear or consistent way.
- Apps with frequent updates and in-app purchases tend to do better.
- Readability of the app description doesn't seem to matter.
- Some variables (like total installs) can accidentally give away the answer and hurt the model.

•

#### Conclusions & Lesson Learned

- Web-scraped data can be messy—cleaning and validation are essential.
- Some engineered features added noise instead of value.
- To predict real success, we need more context—like marketing budgets or team size.
- Big app hits are hard to predict with limited features alone.

•

## Thanks!