Predictiction of Kickstart Projects

Daehyun Kim

Background

- The world’s largest funding platform for creative projects
- Backers
- Pledged vs. Goal
- Data from Kaggle

Business Questions

- Can we predict whether a crowdfunding project will be successful before release?

Variables

```
df.head(5)
```

name	category	main_category	currency	deadline	goal	launched	pledged	state	backers	country	usd pledged	usd_pledged_real	usd_goal_real
ıe Songs of Adelaide & Abullah	Poetry	Publishing	GBP	2015-10-09	1000.0	2015-08-11 12:12:28	0.0	failed	0	GB	0.0	0.0	1533.95
eeting From arth: ZGAC rts Capsule For ET	Narrative Film	Film & Video	USD	2017-11-01	30000.0	2017-09-02 04:43:57	2421.0	failed	15	US	100.0	2421.0	30000.00
Where is Hank?	Narrative Film	Film & Video	USD	2013-02-26	45000.0	2013-01-12 00:20:50	220.0	failed	3	US	220.0	220.0	45000.00
oshiCapital Rekordz eds Help to Complete Album	Music	Music	USD	2012-04-16	5000.0	2012-03-17 03:24:11	1.0	failed	1	US	1.0	1.0	5000.00
Community ilm Project: The Art of ghborhoo...	Film & Video	Film & Video	USD	2015-08-29	19500.0	2015-07-04 08:35:03	1283.0	canceled	14	US	1283.0	1283.0	19500.00

- 378302 rows

  16 variables

- Data from

  Kickstarter platform

Missing values

```
df.isna().sum()
```

```
ID                    0
name                  4
category              0
main_category         0
currency              0
deadline              0
goal                  0
launched              0
pledged               0
state                 0
backers               0
country               0
usd pledged        3797
usd_pledged_real      0
usd_goal_real         0
dtype: int64
```

Missing Values

```
df[df['usd pledged'].isna()].head()
```

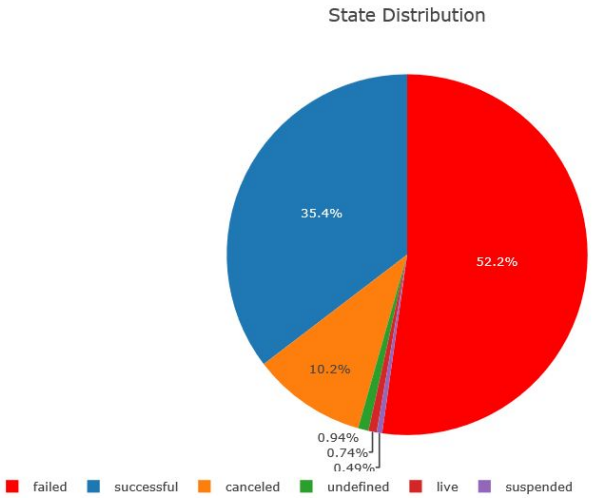
	name	category	main_category	currency	deadline	goal	launched	pledged	state	backers	country	usd pledged	usd_pledged_real	usd_goal_real
	'FIGHTERZ IE MURICA	Film & Video	Film & Video	USD	2014-09-20	6500.0	2014-08-06 21:28:36	555.00	undefined	0	N,0"	NaN	555.00	6500.00
	:an Woods - ameleon EP	Music	Music	AUD	2015-08-25	4500.0	2015-08-04 12:05:17	4767.00	undefined	0	N,0"	NaN	3402.08	3211.53
	e Making of ley Kelley's ebut Album	Music	Music	USD	2015-04-09	3500.0	2015-03-10 20:06:13	3576.00	undefined	0	N,0"	NaN	3576.00	3500.00
	· Side Down ebut Album	Music	Music	USD	2015-11-26	6000.0	2015-11-02 22:09:19	7007.80	undefined	0	N,0"	NaN	7007.80	6000.00
	se Goehring debut EP	Music	Music	USD	2016-03-21	3000.0	2016-02-23 03:09:49	3660.38	undefined	0	N,0"	NaN	3660.38	3000.00

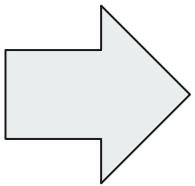
Project Length Variable

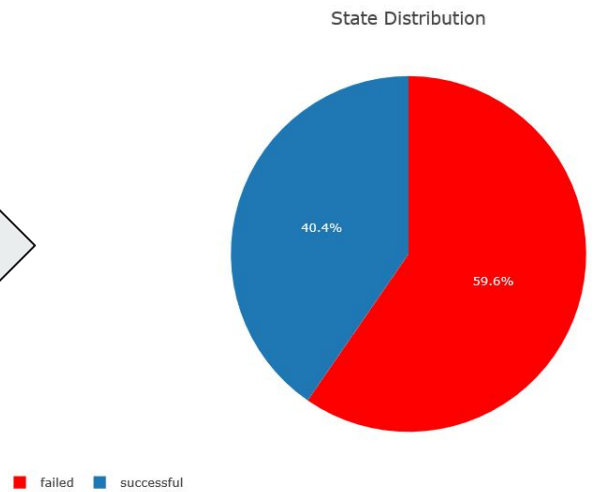
```
df[['deadline','launched','project_length']].head(5)
```

	deadline	launched	project_length
0	12/4/2009	11/25/2009	10
1	12/13/2011	11/7/2011	37
2	3/16/2012	1/25/2012	52
3	11/12/2016	11/11/2016	2
4	7/19/2011	7/12/2011	8

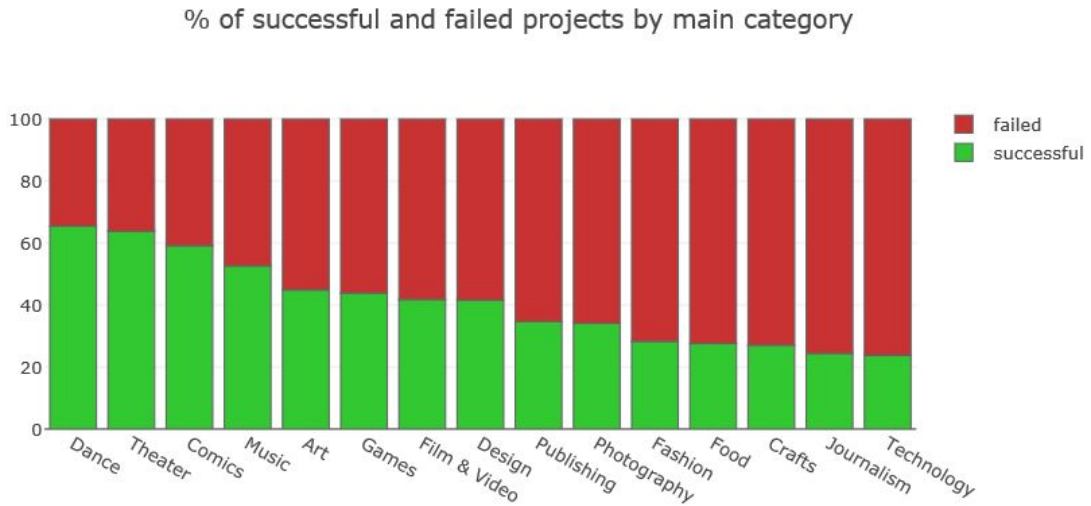
Distribution of State



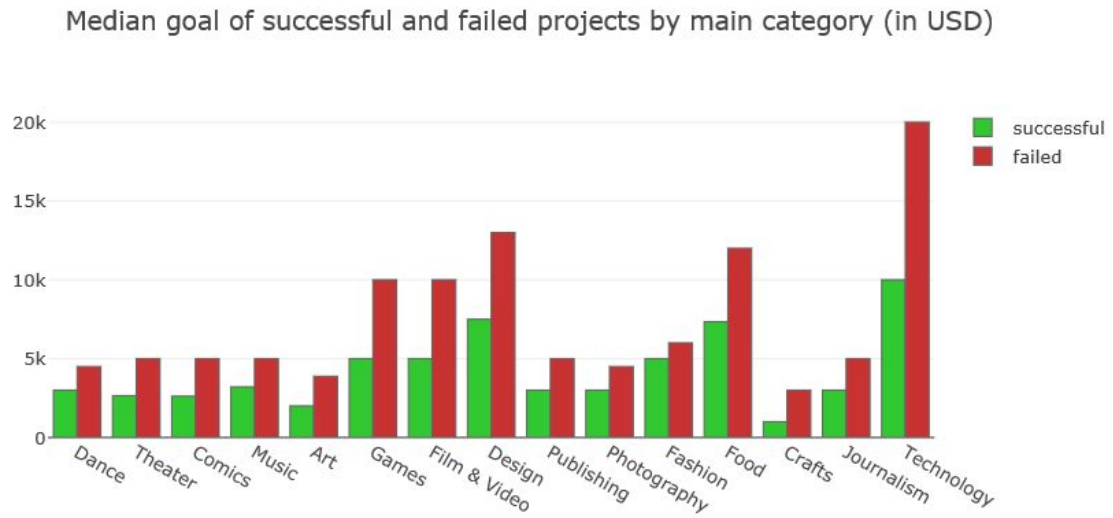




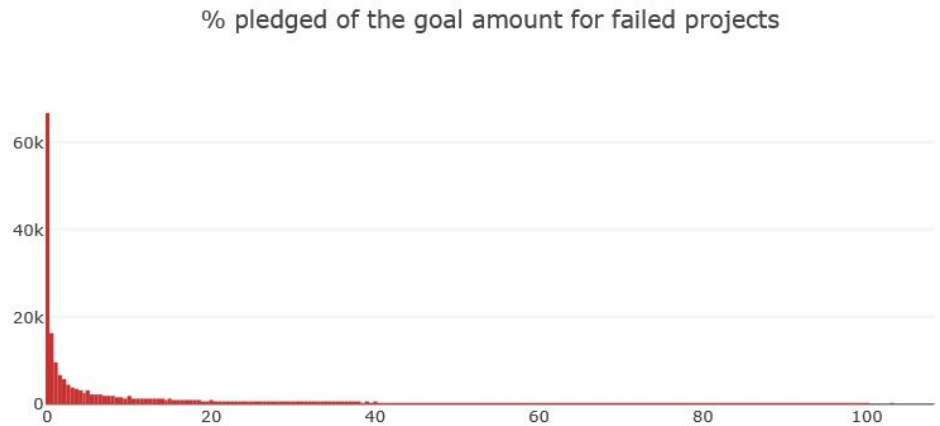
State by Main Category



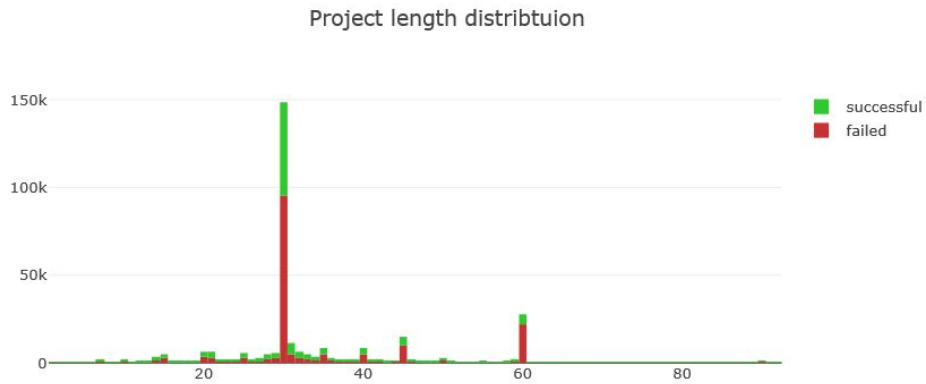
Goal of projects by Main Category



Pledged vs. Goal for Failed Projects



Project Length Distribution



Mean days for failed projects: 35.17
Mean days for successful projects: 32.16

Classifier Models

- Random Forest
  - A large number of decision trees picking the most common outcome as the final outcome
  - Good for multi-class
  - Much faster
- Support Vector Machine (linear)
  - Kernel trick to transform data and find an optimal boundary between possible outputs.
  - Good for two-class
- Scikit-Learn (python)

Variables in Model

```
df_features.head(5)
```

	main_category	currency	state	backers	country	usd_pledged_real	usd_goal_real	project_length
0	Publishing	GBP	failed	0	GB	0.0	1533.95	59
1	Film & Video	USD	failed	15	US	2421.0	30000.00	60
2	Film & Video	USD	failed	3	US	220.0	45000.00	45
3	Music	USD	failed	1	US	1.0	5000.00	30
4	Film & Video	USD	canceled	14	US	1283.0	19500.00	56

Dummy Variables for categories

```
# Categorial columns to numerical using dummy variables
df_features = pd.get_dummies(df_features)
```

```
df_features.head(5)
```

ID	state	backers	usd_pledged_real	usd_goal_real	project_length	main_category_Art	main_category_Comics	main_category_Crafts	main_category_Da
620302213	1	6	100.00	0.01	10	1	0	0	
9572984	0	0	0.00	0.15	52	0	0	0	
1379346088	1	7	16.41	0.49	2	1	0	0	
219760504	0	0	0.00	0.50	8	0	0	0	
69101025	1	2	522.81	0.55	33	0	0	0	

5 rows × 57 columns

Train vs Test split

```
# Split the data to train and test
df_train, df_valid = train_test_split(df_features,
                                      test_size = 0.25,
                                      random_state=2018)
```

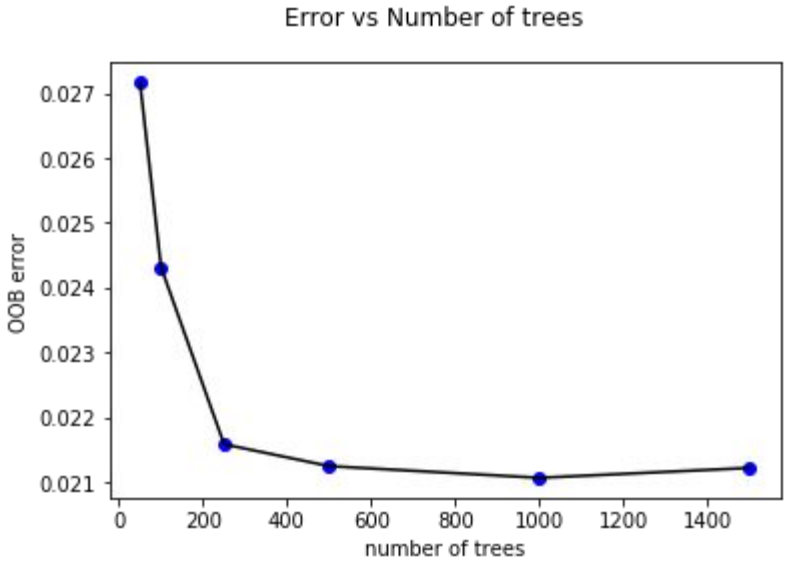
```
df_train['state'].value_counts()
```

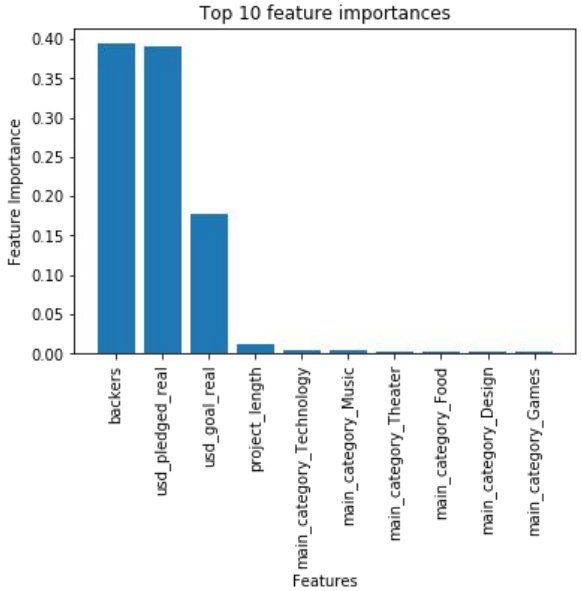
```
0    148174
1    100342
Name: state, dtype: int64
```

```
df_valid['state'].value_counts()
```

```
0    49348
1    33491
Name: state, dtype: int64
```

Random Forest





Random Forest

```
class_names = ['failed','successful']
conf_df = pd.DataFrame(conf_mat, class_names, class_names)
conf_df
```

	failed	successful
failed	47819	1529
successful	251	33240

```
conf_df_pct = conf_df/conf_df.sum(axis=0)
round(conf_df_pct, 5)
# Very Successful results
```

	failed	successful
failed	0.99478	0.04398
successful	0.00522	0.95602

```
# Class-level performance: 0.978
f1_score(y_true=y_test,
         y_pred=y_pred_test,
         average='macro')
```

```
0.9778257287006349
```

```
# Overall performance across all classes: 0.979
f1_score(y_true=y_test,
         y_pred=y_pred_test,
         average='micro')
```

```
0.9785125363657214
```

Support Vector Machine

```
conf_df_svm = pd.DataFrame(conf_mat_svm, class_names, class_names)
conf_df_svm
```

	failed	successful
failed	49329	19
successful	1	33490

```
conf_df_pct_svm = conf_df_svm/conf_df_svm.sum(axis=0)
round(conf_df_pct_svm, 5)
# Very Successful results
```

	failed	successful
failed	0.99998	0.00057
successful	0.00002	0.99943

```
# Overall performance across all classes: 0.9998
f1_score(y_true=y_test,
         y_pred=y_pred_test_svm,
         average='micro')
```

```
0.9997585678243339
```

```
# Overall performance across all classes: 0.99997
f1_score(y_true=y_test,
         y_pred=y_pred_test_svm,
         average='macro')
```

```
0.9997494065576775
```

Speed of Model Fitting

Random Forest	Support Vector Machine
9.5 min	59.5 min

Findings

- Some categories are more likely to be successful
- Projects with higher goal amount are more likely to fail
- Most failing projects are pledged less than 20% of the goal amount
- Backers, Pledged amount, Goal amount biggest contributions in the model

Challenges

- Not enough features from data
- Most important features are not in your control
- Long model fitting time

Recommendations & Future Research

- Get other features that are more in project owner's control
  - and exclude backers and pledged from model
  - or cap on the contribution from each feature or Weighted contribution
- Consider aggregating the categories into fewer groups
  - countries, main project categories
- Try other classfier algorithms
- Try regression for number outcome

Appendix

https://www.kaggle.com/kemical/kickstarter-projects/

Questions