# load_data

August 26, 2021

## 0.1 Load Kick Starter Data

- Load .csv files found from

[1]:
```
from keys import *
import json
import pandas as pd
import numpy  as np
from pandas.io.json import json_normalize
import os
from datetime import datetime
import matplotlib.pyplot as plt
import scipy.stats as stats
import statistics as st
import matplotlib
# Supporting functions Collapsed in next cell
font = {'family' : 'normal',
        'weight' : 'bold',
        'size'   : 22}

matplotlib.rc('font', **font)
```

[2]:
```
def read_csvs(directory, df_combined):
    """ Iterate through <directory> and read each .csv file in.

    Args:
        directory (string)     : Path to Folder w/ csv files
        df_combined (DataFrame):  Input with raw data read in so far

    Returns:
        df_combined (DataFrame): Input + raw data at <directory> concatenated␣
 ↪to end
    """
    for filename in os.listdir(directory):
        if filename.endswith('.csv'):

            csv_path    = os.path.join(directory, filename)
            df_file     = pd.read_csv(csv_path)
```

```
            df_combined = pd.concat([df_combined, df_file])
        else:
            continue # Skip if not .csv
    return df_combined

def ini_df_loop(type):
    if type == 'photo':
        df_loop = pd.DataFrame(columns=photo_keys)

    elif type == 'creator':
        df_loop = pd.DataFrame(columns=creator_keys)

    elif type == 'location':
        df_loop = pd.DataFrame(columns=location_keys)

    elif type == 'profile':
        df_loop = pd.DataFrame(columns=profile_keys)

    elif type == 'category':
        df_loop = pd.DataFrame(columns=category_keys)

    return df_loop

def expand_multi_level_cols(df_combined):
    """ Process Raw Data by Expanding Multi-Lvl

    Args:
        df_combined (DataFrame): Input DataFrame with raw unprocesseed nested␣
 ↪JSON columns

    Returns:
        df_out(DataFrame) : Output Dataframe with processed JSON Columns
    """

    types       = ["creator", "location", "profile", "category" ]
    n           = len(df_combined)
    df_out      = pd.DataFrame()
    for type_i in types:
        df          = df_combined[type_i]
        df_loop     = ini_df_loop(type_i)

        for i in range(0, n):
            try:
                item    = df[i]
                dic     = json.loads(item)
                x       = pd.json_normalize(dic)
                df_loop = df_loop.append(x, ignore_index=True)
```

```
            except:
                continue
        df_out = pd.concat([df_out, df_loop], axis=1)
        return df_out

def get_stats(feature):
    mu = round(st.mean(feature), 0)
    sigma = round(st.stdev(feature), 0)
    lower = mu - 3*sigma
    upper = mu + 3*sigma
    vc = feature.value_counts()
    if lower < 0:
        lower = 0
    range_str = f"Mean = {mu} , 6 sigma range = {lower} : {upper}"
    return mu, sigma, vc, range_str


def load_data(df_type, fname):
    # This is first file from june_2021 to start dataframe (removed from june␣
 ↪21 folder)
    df           = pd.read_csv(fname)
    df_expand    = expand_multi_level_cols(df)
    df_final     = df.drop(columns=["creator", "location", "profile",␣
 ↪"category"])
    df_final     = pd.concat([df_final, df_expand], axis=1)
    idx_1        = df_final['state'] == 'failed'
    idx_2        = df_final['state'] == 'successful'
    df_final     = df_final[idx_1 | idx_2]
    df_final     = wrangle(df_final)
    pkl_file     = 'df_' + df_type + '.pkl'
    df_final.to_pickle(pkl_file)
    return df_final

def wrangle(df):
    df_out = df.copy()

    # Replace empty goal rows with nan
    df_out['goal'] = df_out['goal'].replace('', np.nan)

    # remove rows without goal:
    df_out = df_out[ df_out['goal'] != np.nan ]

    # filter df to projects goal to raise $500 to $250,000
    df_out = df_out[(df_out['goal'] >= 500) & (df_out['goal']<= 250000 )]

    deadline                  = pd.to_datetime(df_out['deadline'], unit='s')
    launched                  = pd.to_datetime(df_out['launched_at'], unit='s')
```

```
    df_out['campaign_length'] = (deadline - launched).dt.days
    df_out['pledge_pct_goal'] = df_out['converted_pledged_amount'].
 ↪divide(df_out['goal'])

    return df_out

def eda(df):

    # Plot 2d Hist of Pledged and Goal
    y, x = df['converted_pledged_amount'], df['goal']

    x_mean, x_std, o, p = get_stats(x)
    y_mean, y_std, o, p = get_stats(y)

    fig, axs = plt.subplots(1, 1, figsize=(10, 10))
    axs.hist2d(x, y, bins=100, range=[[0, 50000], [0, 50000]])
    axs.set_title('Hist Map Pledged vs. Goal')
    axs.set_xlabel('Goal')
    axs.set_ylabel('Pledged')
    plt.tight_layout()

    # Plot 2d Hist of Pledged and % of Goal Funded
    y, x = df['pledge_pct_goal'], df['goal']

    x_mean, x_std, o, p = get_stats(x)
    y_mean, y_std, o, p = get_stats(y)

    fig, axs = plt.subplots(1, 1, figsize=(10, 10))
    axs.hist2d(x, y, bins=100, range=[[0, 50000], [0, 2]])
    axs.set_title('Hist Map Goal vs. % Goal Funded, n_bins =100')
    axs.set_xlabel('Goal')
    axs.set_ylabel('% of Goal Funded')
    plt.tight_layout()

    # Plot 2d Hist of Pledged and % of Goal Funded
    y, x = df['pledge_pct_goal'], df['goal']

    x_mean, x_std, o, p = get_stats(x)
    y_mean, y_std, o, p = get_stats(y)

    fig, axs = plt.subplots(1, 1, figsize=(10, 10))
    axs.hist2d(x, y, bins=50, range=[[0, 50000], [0, 2]])
    axs.set_title('Hist Map Goal vs. % Goal Funded, n_bins = 50')
    axs.set_xlabel('Goal')
    axs.set_ylabel('% of Goal Funded')
    plt.tight_layout()
```

```
# Plot Hists of Pledged and Goal in $
fig, axs = plt.subplots(2, 1, figsize=(20, 20))
y, x = df_test['converted_pledged_amount'], df_test['goal']

x_mean, x_std, o, p = get_stats(x)
y_mean, y_std, o, p = get_stats(y)

axs[0].hist(x, bins=300, range=[0, 50000])
axs[0].hist(y, bins=300, range=[0, 50000])
# axs[0].axis([0, 250, 0, 1000 ])
axs[0].set_title('Hist n_bins=300, Range: 0 - $50K')
axs[0].legend(['Goal','Pledged'])
axs[0].set_xlabel('Goal')
axs[0].set_ylabel('Count')

axs[1].hist(y, bins=200, range=[100, 50000])
axs[1].hist(x, bins=200, range=[100, 50000])
axs[1].set_title('Hist n_bins=200, Range: $100 - $50K')
axs[1].legend(['Goal','Pledged'])
axs[1].set_xlabel('Pledged')
axs[1].set_ylabel('Count')
plt.tight_layout()

# Plot Goal / Pledged Distributiuons

fig, axs = plt.subplots(2, 1, figsize=(20, 20))
features = ['goal','converted_pledged_amount']
for feat in features:
    mu, sigma, v, range_str = get_stats(df[feat])
    print(feat+ " " + range_str)
    x = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
    axs[0].plot(x, stats.norm.pdf(x, mu, sigma))
axs[0].axis([0, 100000, 0, 1.6*10**-5])
axs[0].set_title('Goal and Pledged Distributions')
axs[0].set_ylabel('Probability')
axs[0].set_xlabel('$')
axs[0].legend(features)

feat = 'pledge_pct_goal'
mu, sigma, v, range_str = get_stats(df[feat])
print(feat+ " " + range_str)
x = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
axs[1].plot(x, stats.norm.pdf(x, mu, sigma))
axs[1].axis([0, 20, 0, .07])
axs[1].set_ylabel('Probability')
axs[1].set_xlabel('% of Goal Funded')
```

```
    axs[1].set_title('Distribution: % of Goal Funded')
    axs[1].legend([feat])

    fig, axs = plt.subplots(1, 1, figsize=(20, 20))
    feat = 'campaign_length'
    mu, sigma, v, range_str = get_stats(df[feat])
    print(feat+ " " + range_str)
    x = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
    axs.plot(x, stats.norm.pdf(x, mu, sigma))
    axs.axis([0, 20, 0, .07])
    axs.set_ylabel('Probability')
    axs.set_xlabel('Campaign Length (Days)')
    axs.set_title('Distribution: Campaign Lengths')
    axs.legend([feat])


    # Plot Scatter of goal vs. pledge and goal vs. pct funded

    fig, axs = plt.subplots(1, 2, tight_layout=True, figsize=(20,10))

    x, y = df['goal']/1000, df['converted_pledged_amount'] /1000
    zeros, ones = np.zeros(len(x)), np.ones(len(x))
    xx = x.sort_values()

    axs[0].scatter(y = y, x=x)
    axs[0].set_title("Goal vs. Pledged Amount ($K)")
    axs[0].set_xlabel('Goal ($K)')
    axs[0].set_ylabel("Pledged ($K)")
    axs[0].fill_between(xx, 0, xx, color='red', alpha=0.2)
    axs[0].fill_between(xx, 100, xx, color='green', alpha=0.2)
    axs[0].axis([0, 100, 0, 100])


    x, y = df['goal']/1000, df['pledge_pct_goal']
    axs[1].scatter(y=y, x= x)
    axs[1].fill_between(xx, 1, 0, color='red', alpha=0.2)
    axs[1].fill_between(xx, 10, 1, color='green', alpha=0.2)
    axs[1].set_title("Goal vs. % of Goal Funded")
    axs[1].set_xlabel('Goal ($K)')
    axs[1].set_ylabel("% Goal Funded")
    axs[1].axis([0, 100, 0, 10])

def plot_train_test_target(df_train, df_test):
    y_train_count = df_train['state'].value_counts()
    y_test_count  = df_test['state'].value_counts()

    fig, ax = plt.subplots(1, 1, figsize=(10, 10))
```

```
    x = np.array([1,2])

    ax.bar(x, y_train_count, width=.5)
    ax.bar(x+.5, y_test_count, width=.5)
    ax.set_xticks([1, 2])
    ax.set_xticklabels(['Successful', 'Failed'], rotation = 45)
    ax.set_ylabel('Count')
    ax.set_title("Training & Test Target Class Count")
```

[3]:
```
# just one df takes ~few mins to process, removing rest for now
# df_combined  = read_csvs(d_folder_1, df_combined)
# df_combined  = read_csvs(d_folder_2, df_combined)

# This is first/Second file from june_2021 to start dataframe (removed from␣
 ↪june 21 folder)
train_fname = './data/Kickstarter.csv'
test_fname  = './data/Kickstarter001.csv'
df_train    = load_data('train', train_fname)
df_test     = load_data('test', test_fname)
```

[4]:
```
df_train.shape, df_test.shape
```

[4]:
```
((2986, 51), (3012, 51))
```

[5]:
```
df_train.head()
```

[5]:
```
   backers_count                                              blurb  \
1            146  Help us finish a feature film about two strang…
2             41  Considered playfully mischievous and outlaws i…
3            466  A fantastical love story about two New Yorkers…
4             70  A story about a girl named Jane who is learnin…
5            166  A short film portraying the despair of a resea…

   converted_pledged_amount country country_displayable_name  created_at  \
1                     10120      US        the United States  1327602001
2                      3785      US        the United States  1447448673
3                     21634      US        the United States  1359739327
4                      3081      US        the United States  1330026300
5                      6115      GB       the United Kingdom  1408536568

  currency currency_symbol  currency_trailing_code current_currency  …  \
1      USD               $                    True              USD  …
2      USD               $                    True              USD  …
3      USD               $                    True              USD  …
4      USD               $                    True              USD  …
5      GBP               £                   False              USD  …
```

```
   is_superbacker  avatar urls  \
1            None     NaN  NaN
2            None     NaN  NaN
3            None     NaN  NaN
4            None     NaN  NaN
5            None     NaN  NaN

                                       avatar.thumb  \
1  https://ksr-ugc.imgix.net/assets/005/910/605/a…
2  https://ksr-ugc.imgix.net/assets/007/087/961/6…
3  https://ksr-ugc.imgix.net/assets/005/987/612/0…
4  https://ksr-ugc.imgix.net/assets/005/925/157/3…
5  https://ksr-ugc.imgix.net/assets/008/591/512/2…

                                       avatar.small  \
1  https://ksr-ugc.imgix.net/assets/005/910/605/a…
2  https://ksr-ugc.imgix.net/assets/007/087/961/6…
3  https://ksr-ugc.imgix.net/assets/005/987/612/0…
4  https://ksr-ugc.imgix.net/assets/005/925/157/3…
5  https://ksr-ugc.imgix.net/assets/008/591/512/2…

                                      avatar.medium  \
1  https://ksr-ugc.imgix.net/assets/005/910/605/a…
2  https://ksr-ugc.imgix.net/assets/007/087/961/6…
3  https://ksr-ugc.imgix.net/assets/005/987/612/0…
4  https://ksr-ugc.imgix.net/assets/005/925/157/3…
5  https://ksr-ugc.imgix.net/assets/008/591/512/2…

                                      urls.web.user  \
1     https://www.kickstarter.com/profile/1939666693
2  https://www.kickstarter.com/profile/roguecodan…
3      https://www.kickstarter.com/profile/488675737
4   https://www.kickstarter.com/profile/shakyballoon
5     https://www.kickstarter.com/profile/1567687816

                                      urls.api.user campaign_length  \
1  https://api.kickstarter.com/v1/users/193966669…              49
2  https://api.kickstarter.com/v1/users/101884698…              30
3  https://api.kickstarter.com/v1/users/488675737…              29
4  https://api.kickstarter.com/v1/users/220105304…              29
5  https://api.kickstarter.com/v1/users/156768781…              30

   pledge_pct_goal
1         1.012000
2         1.261667
3         1.081700
```

```
4         1.027000
5         2.038333

[5 rows x 51 columns]
```

[6]:
```
df_train['spotlight'].iloc[0]
```

[6]:
```
True
```

[7]:
```
df_test.head()
```

[7]:
```
   backers_count                                              blurb  \
0             44  "Tripping the Light Fantastic" cDc's new evoca…
1            102  A suspenseful adaptation of Edgar Allan Poe's …
2             17  Hi,  I am looking to create these lovely Fan A…
3             35  A sci-fi short about the relationship between …
4            101  Kinetic Makeover is a new dance work by Milka …

   converted_pledged_amount country country_displayable_name  created_at  \
0                      2154      US        the United States  1375716180
1                      6131      US        the United States  1374106946
2                       467      US        the United States  1518674951
3                      1261      DE                  Germany  1605711443
4                      3600      US        the United States  1362009545

  currency currency_symbol  currency_trailing_code current_currency  …  \
0      USD               $                    True              USD  …
1      USD               $                    True              USD  …
2      USD               $                    True              USD  …
3      EUR               €                   False              USD  …
4      USD               $                    True              USD  …

   is_superbacker  avatar  urls  \
0            None     NaN   NaN
1            None     NaN   NaN
2            None     NaN   NaN
3            None     NaN   NaN
4            None     NaN   NaN

                                      avatar.thumb  \
0  https://ksr-ugc.imgix.net/assets/006/736/731/d…
1  https://ksr-ugc.imgix.net/assets/007/786/258/6…
2  https://ksr-ugc.imgix.net/assets/020/242/497/d…
3  https://ksr-ugc.imgix.net/assets/027/910/766/7…
4  https://ksr-ugc.imgix.net/assets/006/064/734/c…

                                      avatar.small  \
```

```
0  https://ksr-ugc.imgix.net/assets/006/736/731/d…
1  https://ksr-ugc.imgix.net/assets/007/786/258/6…
2  https://ksr-ugc.imgix.net/assets/020/242/497/d…
3  https://ksr-ugc.imgix.net/assets/027/910/766/7…
4  https://ksr-ugc.imgix.net/assets/006/064/734/c…

                                      avatar.medium  \
0  https://ksr-ugc.imgix.net/assets/006/736/731/d…
1  https://ksr-ugc.imgix.net/assets/007/786/258/6…
2  https://ksr-ugc.imgix.net/assets/020/242/497/d…
3  https://ksr-ugc.imgix.net/assets/027/910/766/7…
4  https://ksr-ugc.imgix.net/assets/006/064/734/c…

                                      urls.web.user  \
0    https://www.kickstarter.com/profile/concordance
1     https://www.kickstarter.com/profile/1943040097
2     https://www.kickstarter.com/profile/1266698815
3  https://www.kickstarter.com/profile/guardianai…
4       https://www.kickstarter.com/profile/23511873

                                      urls.api.user  campaign_length  \
0  https://api.kickstarter.com/v1/users/649433872…                30
1  https://api.kickstarter.com/v1/users/194304009…                33
2  https://api.kickstarter.com/v1/users/126669881…                29
3  https://api.kickstarter.com/v1/users/209606685…                24
4  https://api.kickstarter.com/v1/users/23511873?…                28

   pledge_pct_goal
0         1.077000
1         1.021833
2         0.518889
3         1.261000
4         1.200000

[5 rows x 51 columns]
```

[8]:
```
df_train.to_excel('df_train.xlsx')
df_test.to_excel('df_test.xlsx')
```

[9]:
```
df_train.columns
```

[9]:
```
Index(['backers_count', 'blurb', 'converted_pledged_amount', 'country',
       'country_displayable_name', 'created_at', 'currency', 'currency_symbol',
       'currency_trailing_code', 'current_currency', 'deadline',
       'disable_communication', 'friends', 'fx_rate', 'goal', 'id',
       'is_backing', 'is_starrable', 'is_starred', 'launched_at', 'name',
       'permissions', 'photo', 'pledged', 'slug', 'source_url', 'spotlight',
```

```
       'staff_pick', 'state', 'state_changed_at', 'static_usd_rate', 'urls',
       'usd_exchange_rate', 'usd_pledged', 'usd_type', 'id', 'name', 'slug',
       'is_registered', 'is_email_verified', 'chosen_currency',
       'is_superbacker', 'avatar', 'urls', 'avatar.thumb', 'avatar.small',
       'avatar.medium', 'urls.web.user', 'urls.api.user', 'campaign_length',
       'pledge_pct_goal'],
      dtype='object')
```

[10]:
```
df_train.describe()
```

[10]:

| | backers_count | converted_pledged_amount | created_at | deadline |
|---|---|---|---|---|
| count | 2986.000000 | 2.986000e+03 | 2.986000e+03 | 2.986000e+03 |
| mean | 251.333557 | 2.321144e+04 | 1.462994e+09 | 1.469696e+09 |
| std | 1650.811663 | 1.923184e+05 | 9.698903e+07 | 9.735865e+07 |
| min | 0.000000 | 0.000000e+00 | 1.252853e+09 | 1.257597e+09 |
| 25% | 20.000000 | 1.210000e+03 | 1.392803e+09 | 1.400898e+09 |
| 50% | 51.000000 | 3.960000e+03 | 1.451499e+09 | 1.457544e+09 |
| 75% | 123.000000 | 1.030050e+04 | 1.542982e+09 | 1.552259e+09 |
| max | 58561.000000 | 8.324792e+06 | 1.621672e+09 | 1.623888e+09 |

| | fx_rate | goal | id | launched_at | pledged |
|---|---|---|---|---|---|
| count | 2986.000000 | 2986.000000 | 2.986000e+03 | 2.986000e+03 | 2.986000e+03 |
| mean | 1.015933 | 12424.128637 | 1.073281e+09 | 1.466987e+09 | 2.354764e+04 |
| std | 0.204322 | 26360.233990 | 6.250572e+08 | 9.752979e+07 | 1.722938e+05 |
| min | 0.049112 | 500.000000 | 1.132300e+05 | 1.252857e+09 | 0.000000e+00 |
| 25% | 1.000000 | 2000.000000 | 5.383573e+08 | 1.398272e+09 | 1.196250e+03 |
| 50% | 1.000000 | 5000.000000 | 1.080353e+09 | 1.454608e+09 | 4.008500e+03 |
| 75% | 1.000000 | 10000.000000 | 1.606345e+09 | 1.549569e+09 | 1.044614e+04 |
| max | 1.399194 | 250000.000000 | 2.147380e+09 | 1.622570e+09 | 6.840648e+06 |

| | state_changed_at | static_usd_rate | usd_exchange_rate | usd_pledged |
|---|---|---|---|---|
| count | 2.986000e+03 | 2986.000000 | 2986.000000 | 2.986000e+03 |
| mean | 1.469696e+09 | 1.016046 | 1.015824 | 2.318778e+04 |
| std | 9.735864e+07 | 0.212263 | 0.212425 | 1.922753e+05 |
| min | 1.257597e+09 | 0.041017 | 0.040899 | 0.000000e+00 |
| 25% | 1.400898e+09 | 1.000000 | 1.000000 | 1.210000e+03 |
| 50% | 1.457544e+09 | 1.000000 | 1.000000 | 3.973500e+03 |
| 75% | 1.552259e+09 | 1.000000 | 1.000000 | 1.030575e+04 |
| max | 1.623888e+09 | 1.714466 | 1.715913 | 8.332854e+06 |

| | campaign_length | pledge_pct_goal |
|---|---|---|
| count | 2986.000000 | 2986.000000 |
| mean | 31.096450 | 2.007789 |
| std | 11.741924 | 5.634798 |
| min | 1.000000 | 0.000000 |
| 25% | 26.000000 | 1.000000 |
| 50% | 30.000000 | 1.067954 |

```
75%          32.000000         1.374125
max          90.000000       143.630000
```

[11]:
```
df_test.describe()
```

[11]:
```
       backers_count  converted_pledged_amount    created_at      deadline  \
count    3012.000000              3.012000e+03  3.012000e+03  3.012000e+03
mean      209.942895              1.950811e+04  1.466301e+09  1.472903e+09
std      1030.168095              1.496255e+05  9.003002e+07  8.995423e+07
min         0.000000              0.000000e+00  1.257914e+09  1.261285e+09
25%        13.000000              8.750000e+02  1.403686e+09  1.409191e+09
50%        44.000000              3.385000e+03  1.455134e+09  1.461042e+09
75%       111.000000              9.888000e+03  1.537137e+09  1.543340e+09
max     26828.000000              5.333792e+06  1.622055e+09  1.623881e+09

       friends      fx_rate           goal            id  is_backing  \
count      0.0  3012.000000    3012.000000  3.012000e+03         0.0
mean       NaN     1.016105   12599.093363  1.060122e+09         NaN
std        NaN     0.209793   25636.410992  6.255584e+08         NaN
min        NaN     0.009032     500.000000  9.349030e+05         NaN
25%        NaN     1.000000    2000.000000  5.075874e+08         NaN
50%        NaN     1.000000    5000.000000  1.058390e+09         NaN
75%        NaN     1.000000   12000.000000  1.588047e+09         NaN
max        NaN     1.408205  250000.000000  2.146215e+09         NaN

       is_starred   launched_at  permissions       pledged  state_changed_at  \
count         0.0  3.012000e+03          0.0  3.012000e+03      3.012000e+03
mean          NaN  1.470126e+09          NaN  2.189289e+04      1.472903e+09
std           NaN  9.011799e+07          NaN  1.456891e+05      8.995422e+07
min           NaN  1.258693e+09          NaN  0.000000e+00      1.261285e+09
25%           NaN  1.406295e+09          NaN  8.587500e+02      1.409191e+09
50%           NaN  1.458050e+09          NaN  3.488660e+03      1.461042e+09
75%           NaN  1.540542e+09          NaN  1.012830e+04      1.543340e+09
max           NaN  1.622891e+09          NaN  5.333793e+06      1.623881e+09

       static_usd_rate  usd_exchange_rate   usd_pledged  campaign_length  \
count      3012.000000        3012.000000  3.012000e+03      3012.000000
mean          1.013750           1.013198  1.949528e+04        31.901726
std           0.216422           0.215582  1.491901e+05        12.087291
min           0.008848           0.008942  0.000000e+00         3.000000
25%           1.000000           1.000000  8.750000e+02        28.000000
50%           1.000000           1.000000  3.379000e+03        30.000000
75%           1.000000           1.000000  9.938929e+03        34.000000
max           1.715575           1.702670  5.333793e+06        90.000000

       pledge_pct_goal
count      3012.000000
```

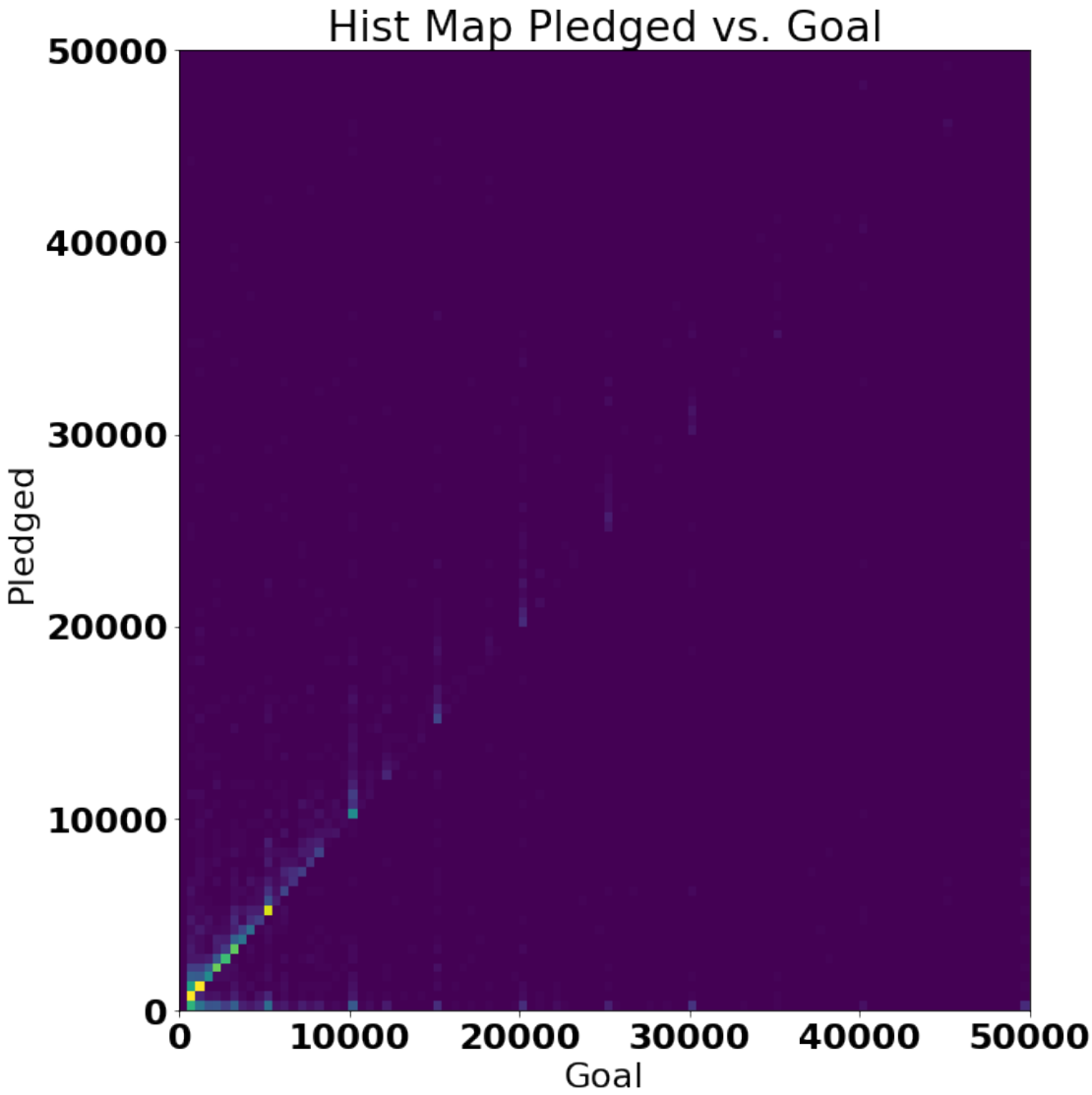
```
mean          1.876503
std           5.742793
min           0.000000
25%           0.199688
50%           1.053000
75%           1.364273
max         152.510000
```

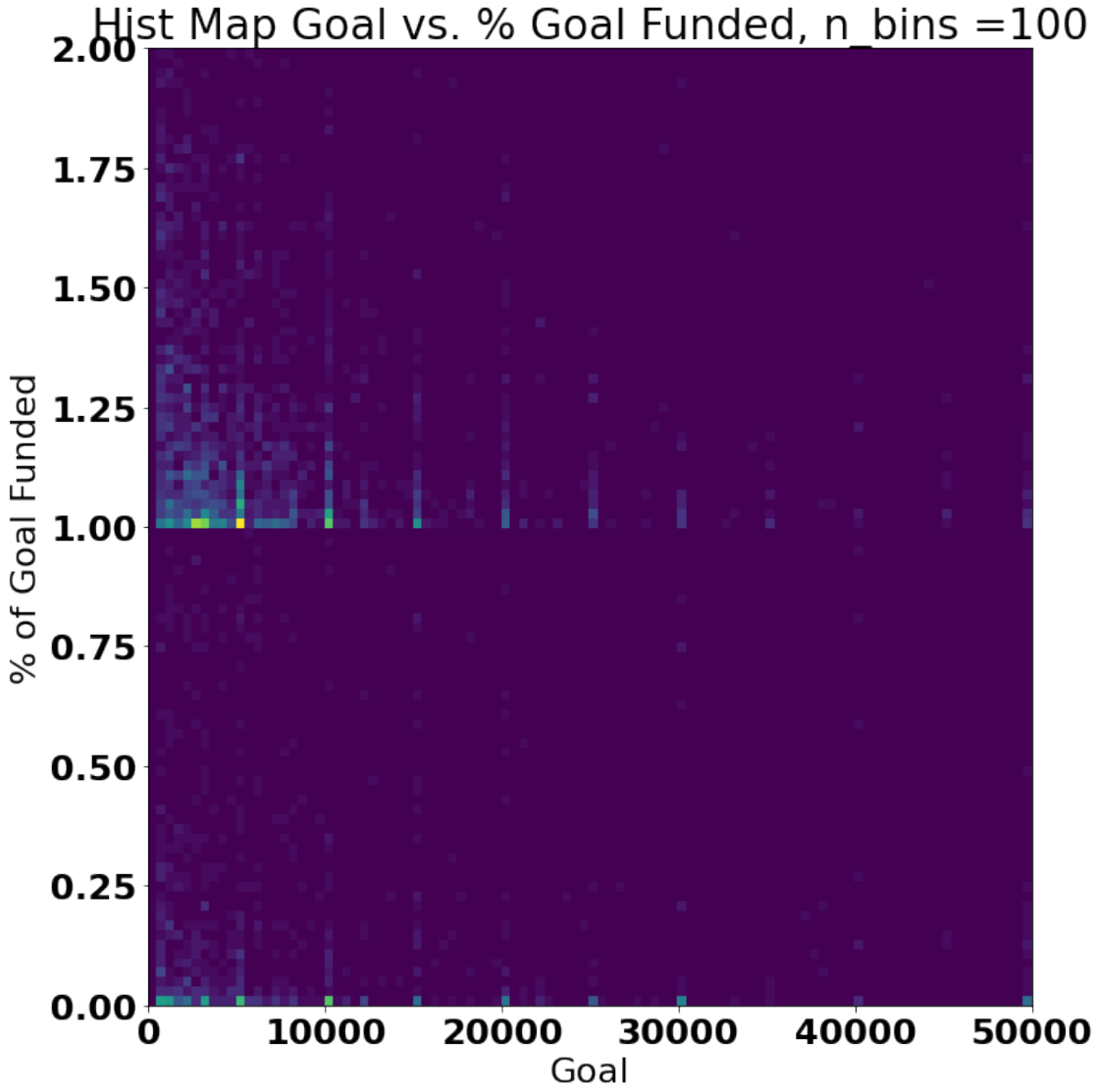
[12]:
```
df_train['currency'].unique()
```

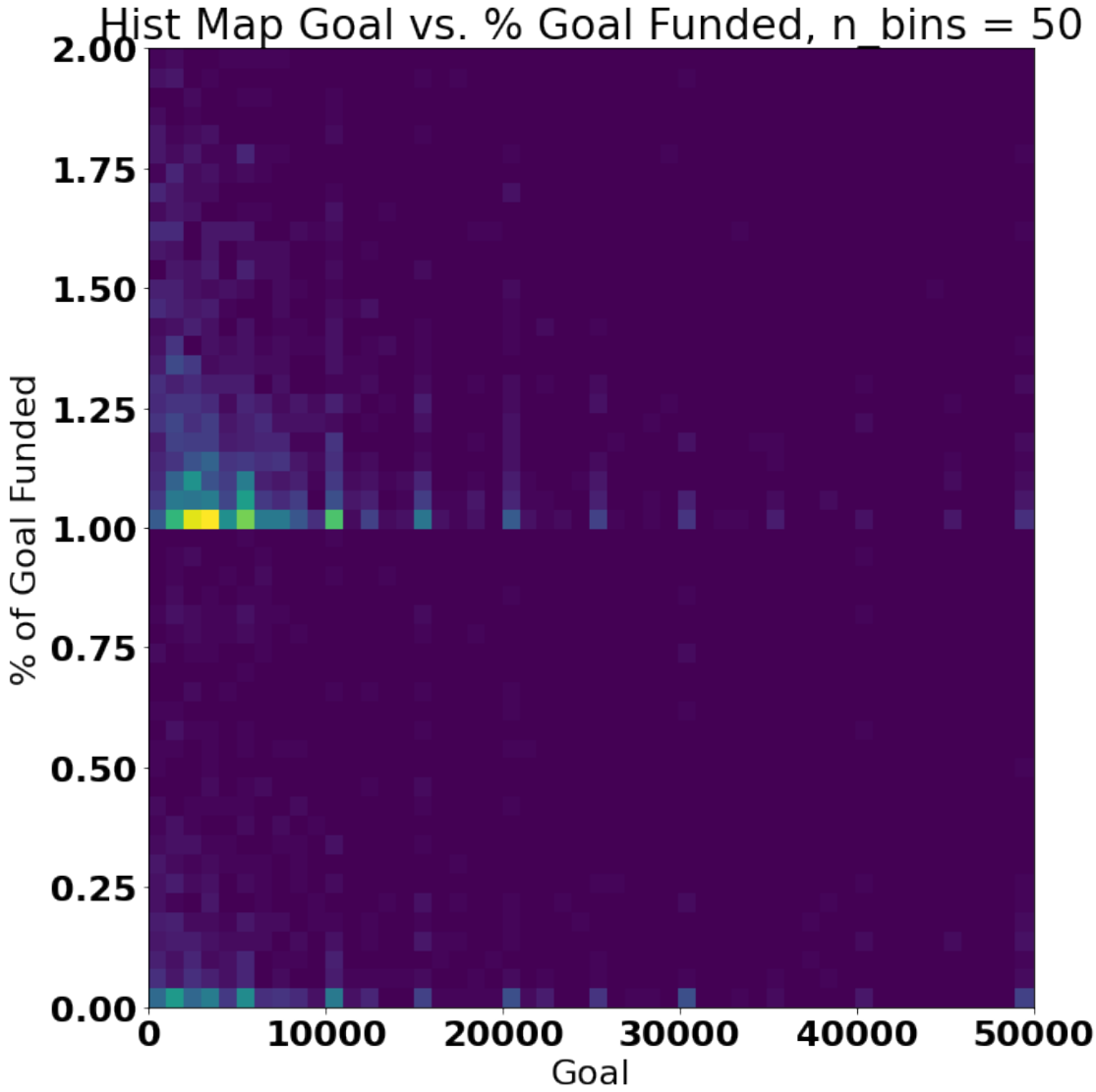
[12]:
```
array(['USD', 'GBP', 'EUR', 'CAD', 'HKD', 'MXN', 'NOK', 'DKK', 'CHF',
       'AUD', 'NZD', 'SGD', 'SEK'], dtype=object)
```

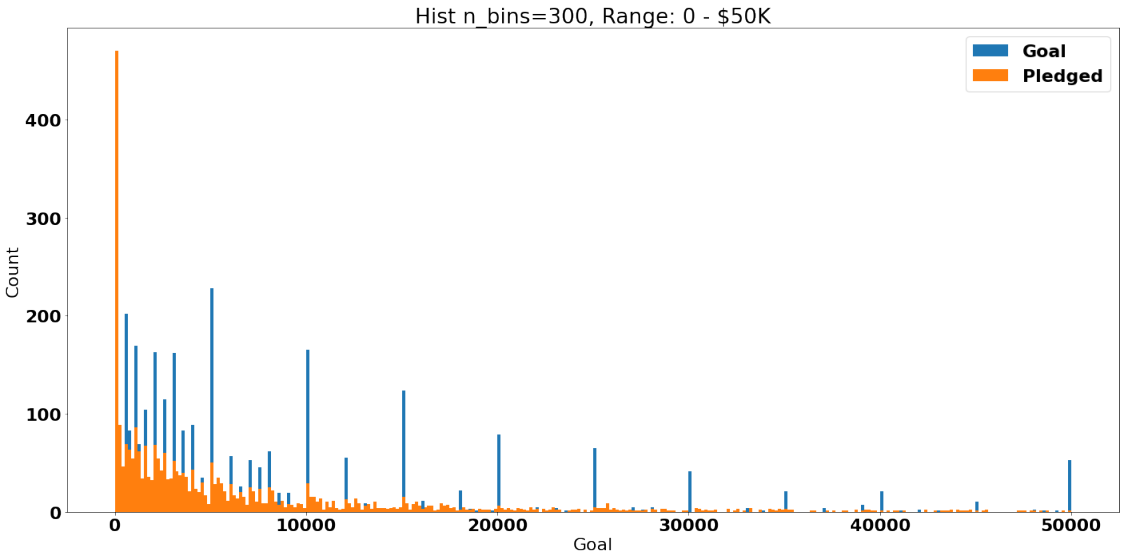
[13]:
```
eda(df_train)
```

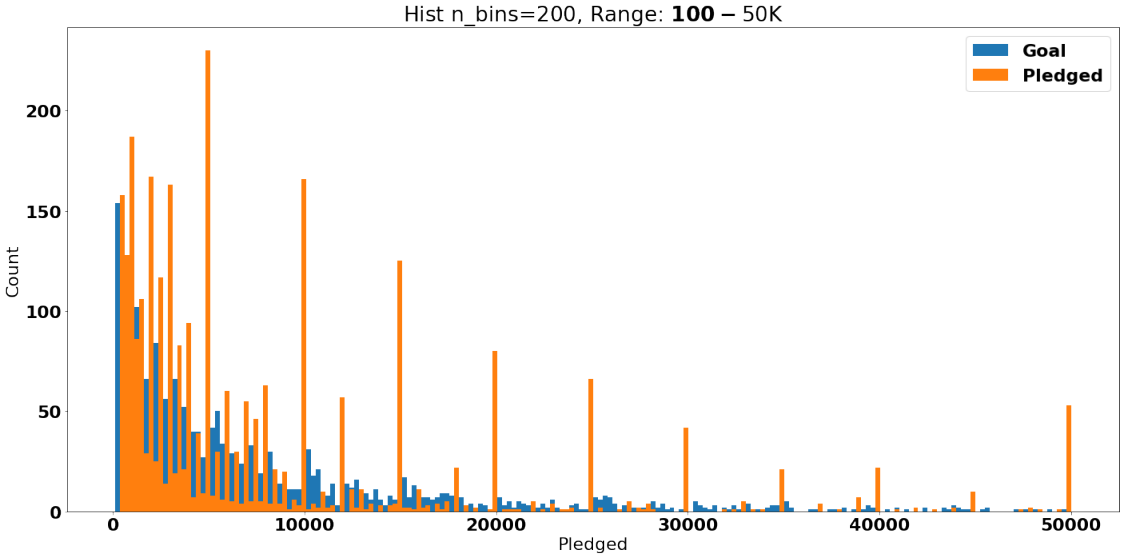
```
findfont: Font family ['normal'] not found. Falling back to DejaVu Sans.
findfont: Font family ['normal'] not found. Falling back to DejaVu Sans.
findfont: Font family ['normal'] not found. Falling back to DejaVu Sans.

goal Mean = 12424.0 , 6 sigma range = 0 : 91504.0
converted_pledged_amount Mean = 23211.0 , 6 sigma range = 0 : 600165.0
pledge_pct_goal Mean = 2.0 , 6 sigma range = 0 : 20.0
campaign_length Mean = 31.0 , 6 sigma range = 0 : 67.0
```

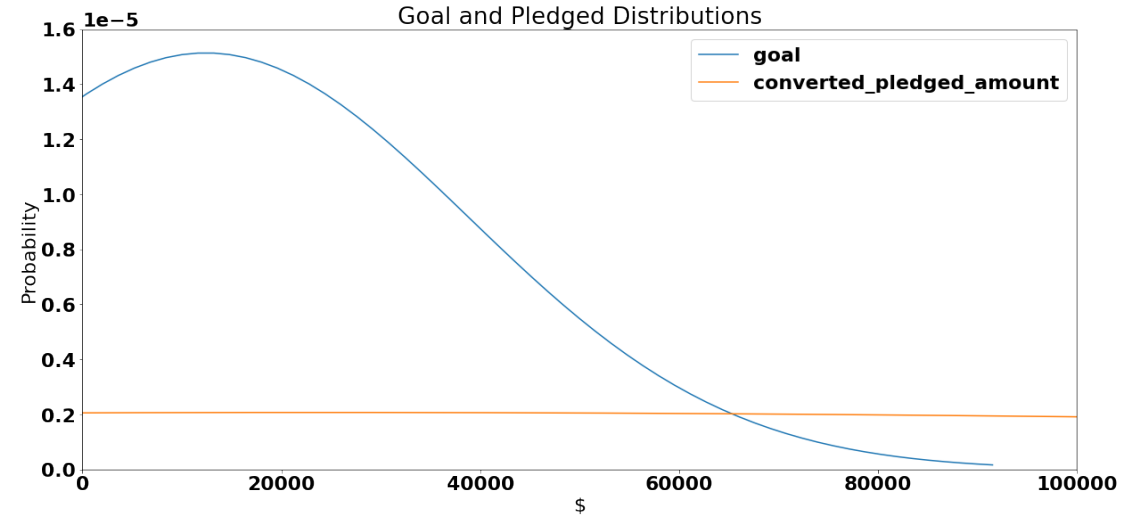

Hist Map Pledged vs. Goal
50000
40000
30000
20000
10000
0
Pledged
0
10000
20000
30000
40000
50000
Goal

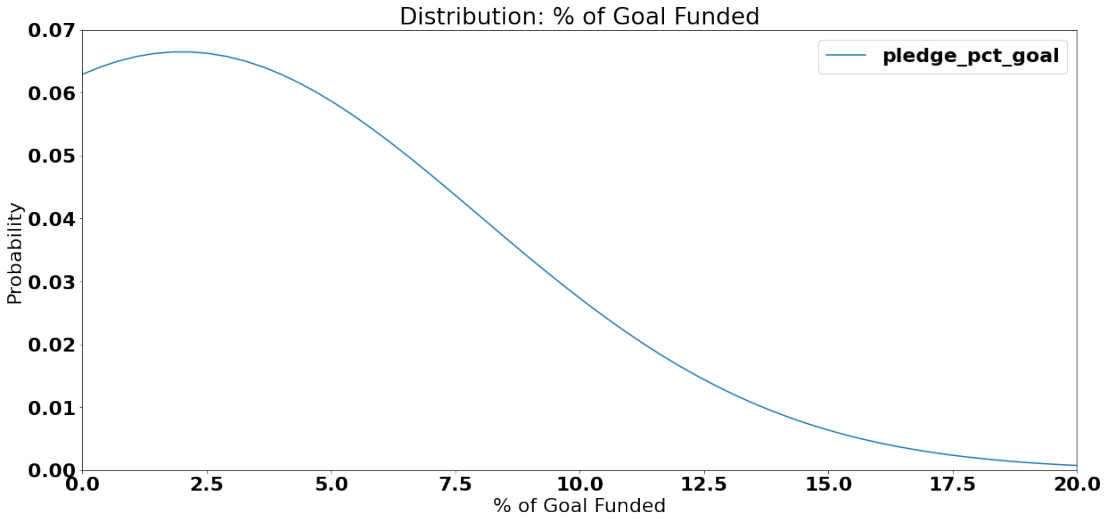

Hist Map Goal vs. % Goal Funded, n_bins =100
% of Goal Funded
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
10000
20000
30000
40000
50000
Goal

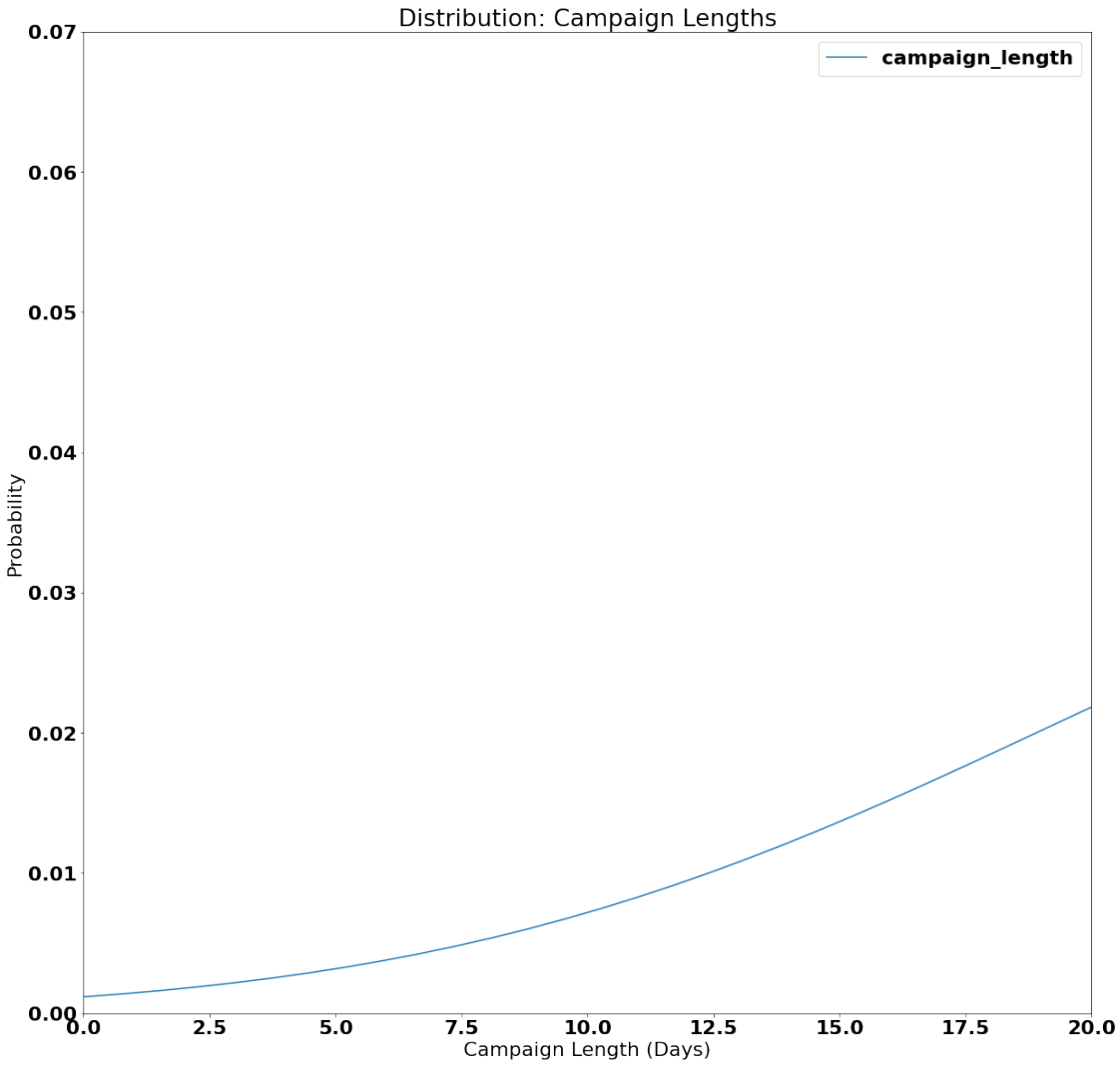

Hist Map Goal vs. % Goal Funded, n_bins = 50
% of Goal Funded
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
10000
20000
30000
40000
50000
Goal

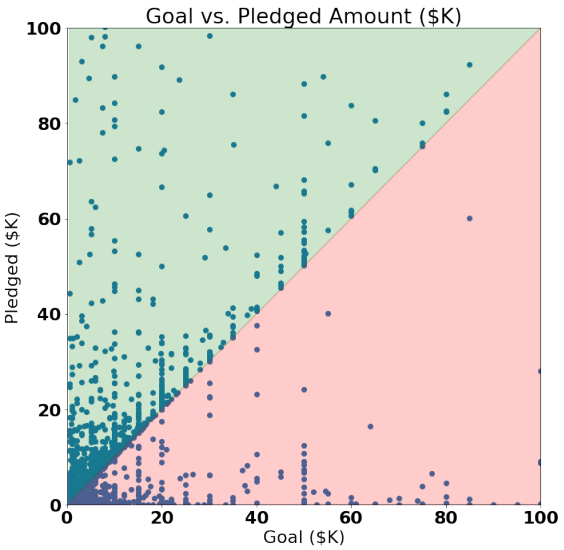

Hist n_bins=300, Range: 0 - $50K
Goal
Pledged
Count
0
100
200
300
400
0
10000
20000
30000
40000
50000
Goal

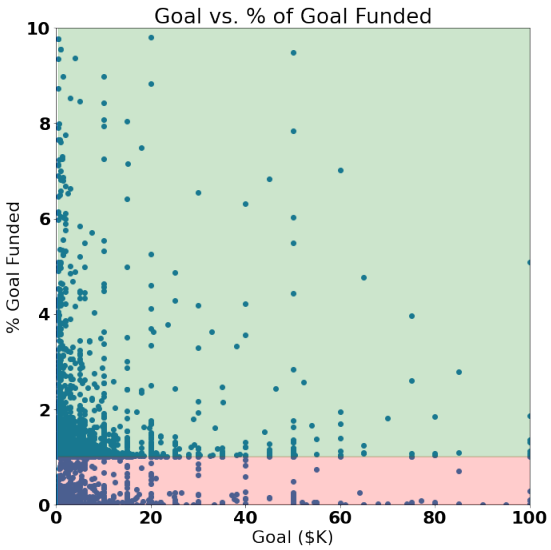

Hist n_bins=200, Range: 100 – 50K
Goal
Pledged
Count
0
50
100
150
200
0
10000
20000
30000
40000
50000
Pledged


Goal and Pledged Distributions
1e−5
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Probability
goal
converted_pledged_amount
0
20000
40000
60000
80000
100000
$


Distribution: % of Goal Funded
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Probability
pledge_pct_goal
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
% of Goal Funded

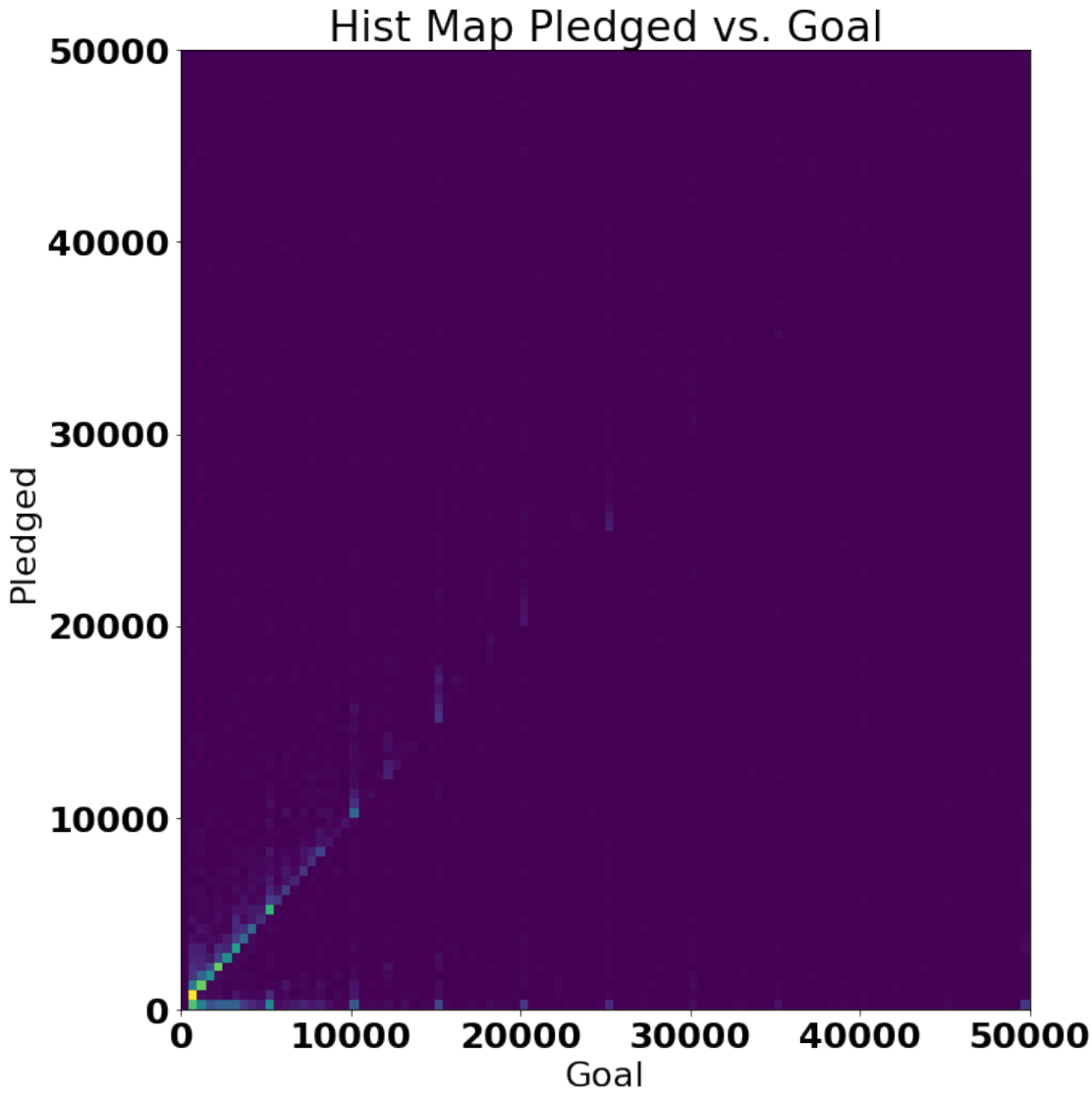

Distribution: Campaign Lengths
campaign_length
Probability
Campaign Length (Days)
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

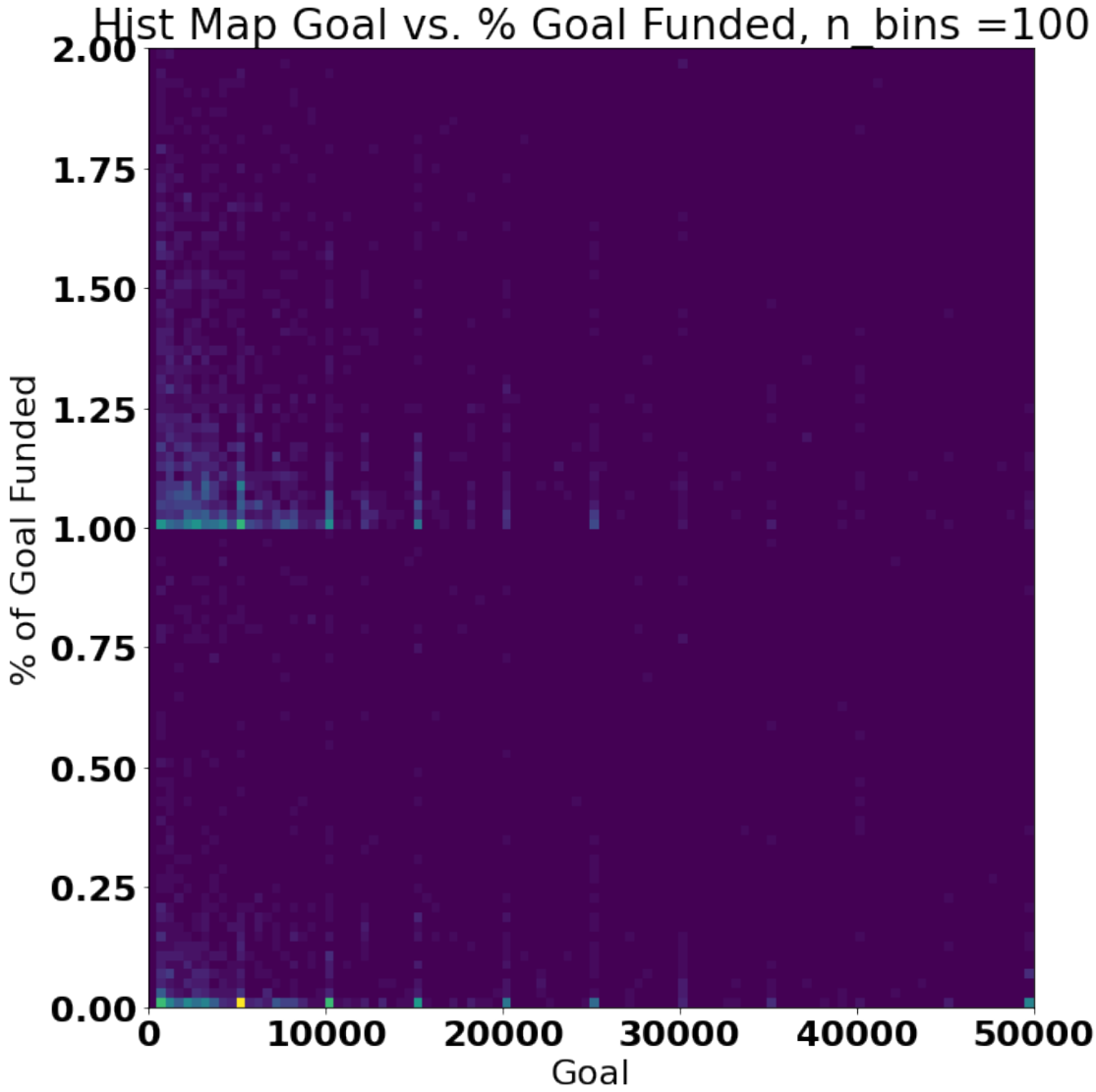

Goal vs. Pledged Amount ($K)
Pledged ($K)
Goal ($K)
0
20
40
60
80
100

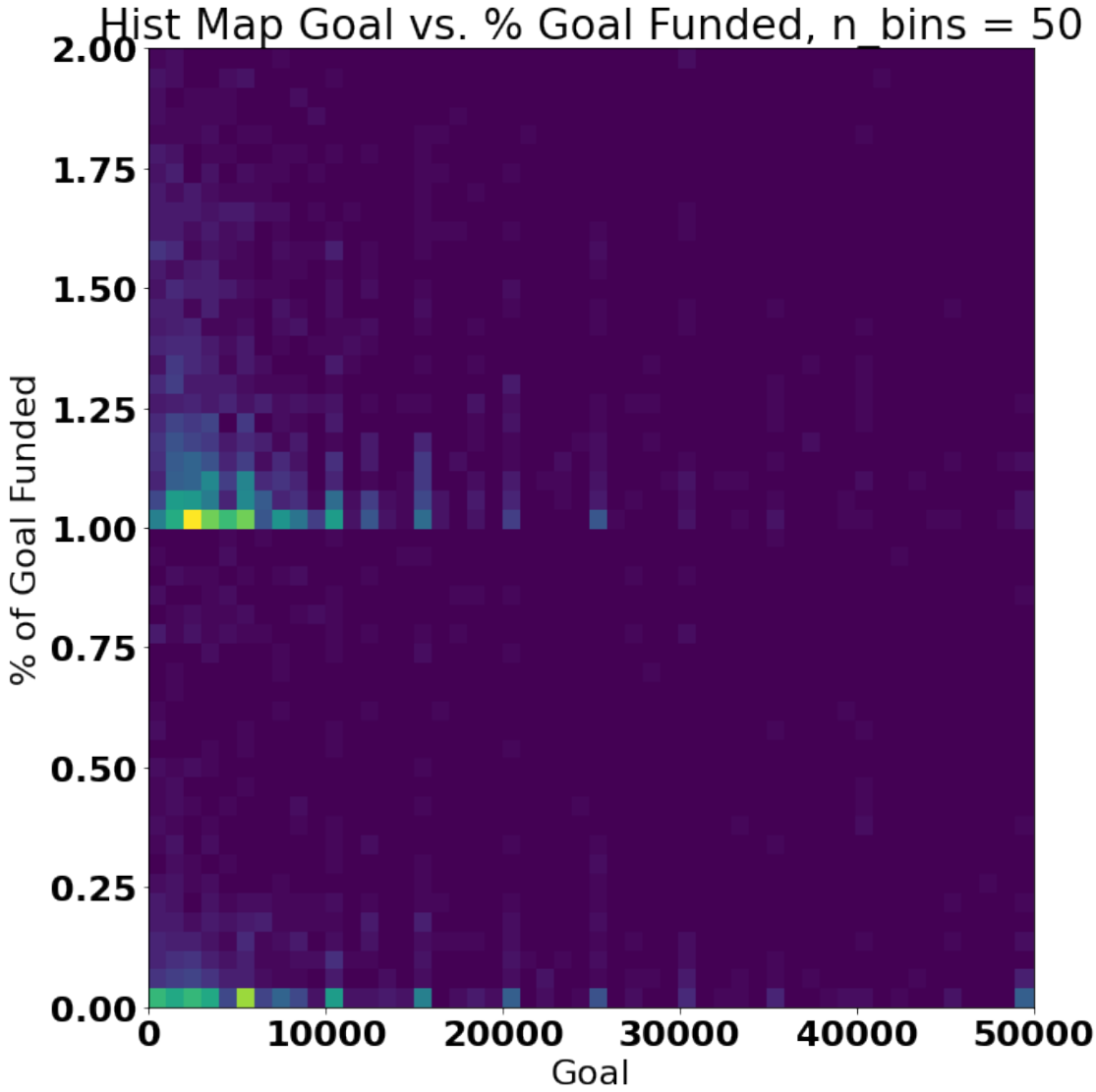

Goal vs. % of Goal Funded
% Goal Funded
Goal ($K)
0
2
4
6
8
10
0
20
40
60
80
100

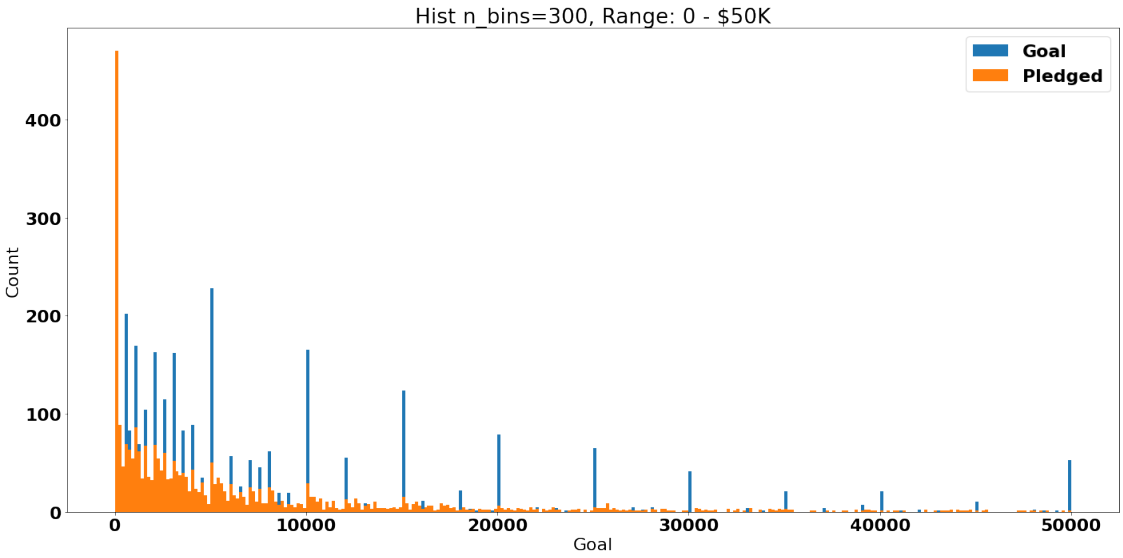
[14]:
```
eda(df_test)
```

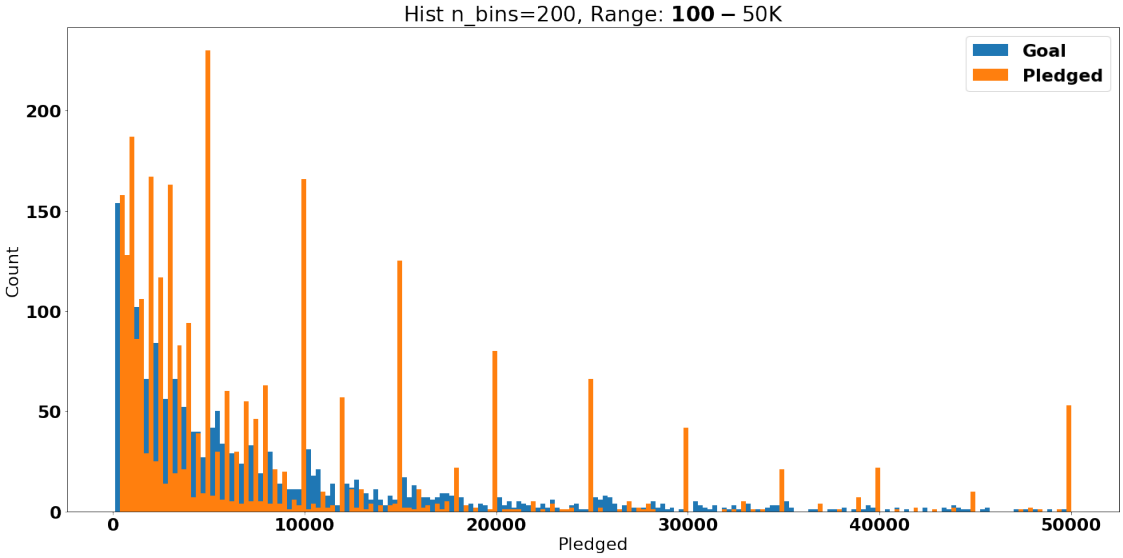
```
goal Mean = 12599.0 , 6 sigma range = 0 : 89507.0
converted_pledged_amount Mean = 19508.0 , 6 sigma range = 0 : 468383.0
pledge_pct_goal Mean = 2.0 , 6 sigma range = 0 : 20.0
campaign_length Mean = 32.0 , 6 sigma range = 0 : 68.0
```

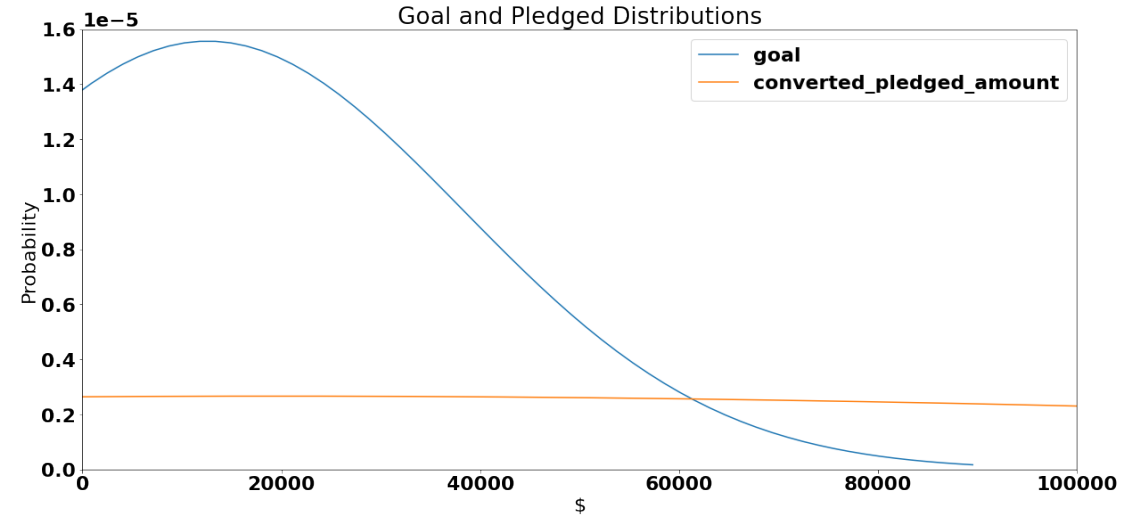


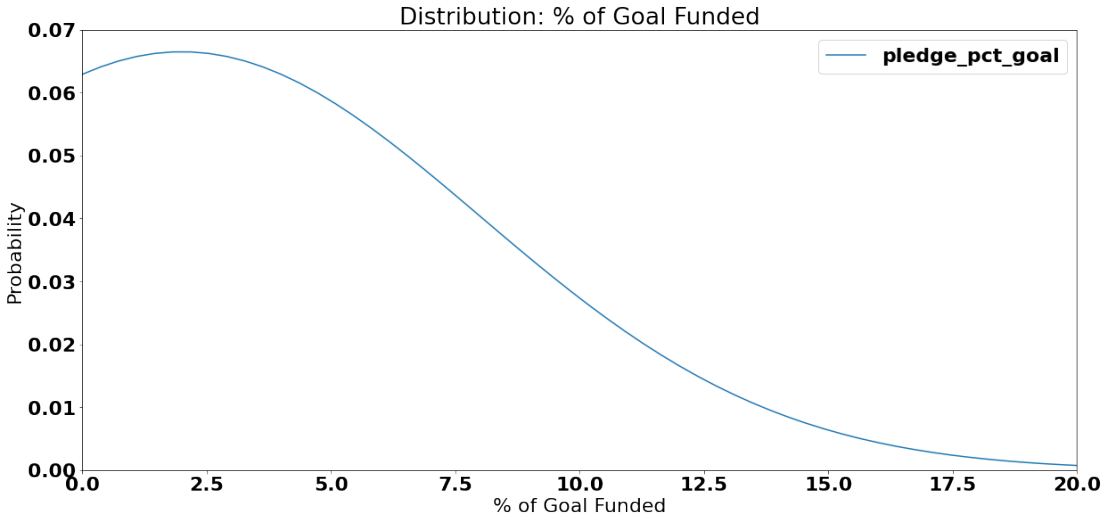

Hist Map Goal vs. % Goal Funded, n_bins =100
% of Goal Funded
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
10000
20000
30000
40000
50000
Goal

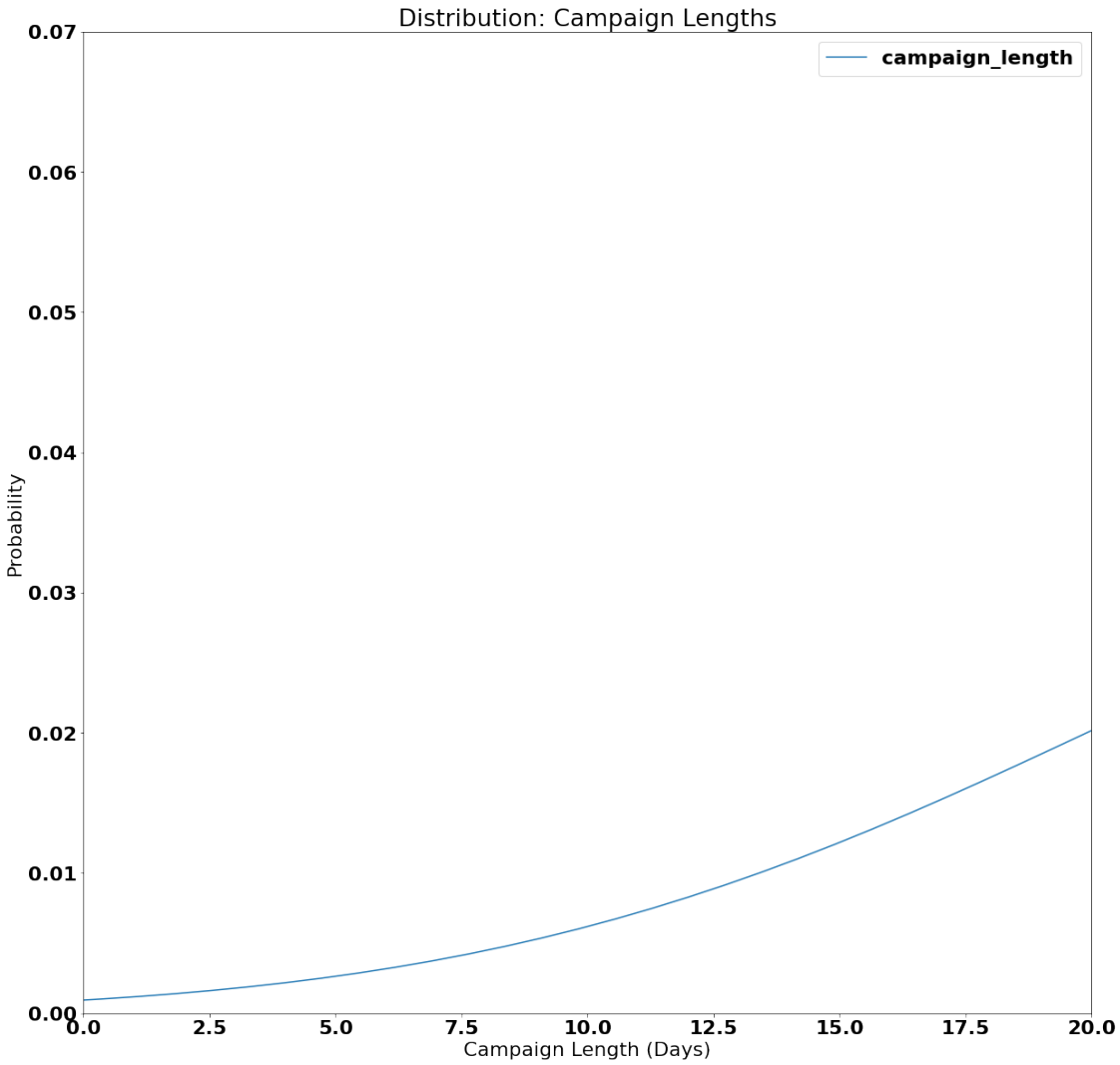

Hist Map Goal vs. % Goal Funded, n_bins = 50
% of Goal Funded
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
10000
20000
30000
40000
50000
Goal

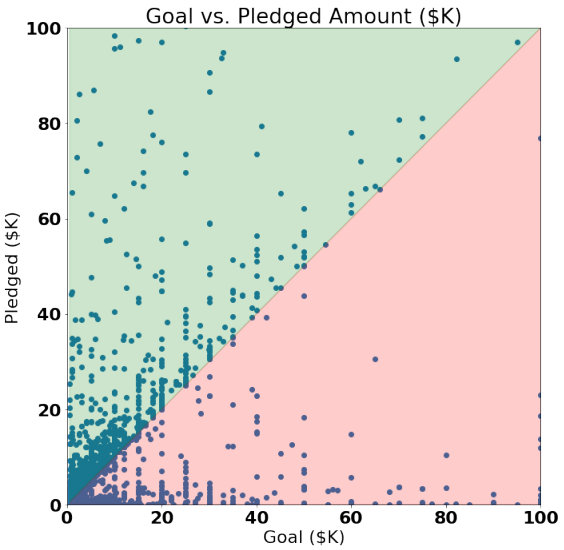

Hist n_bins=300, Range: 0 - $50K
Goal
Pledged
Count
0
100
200
300
400
0
10000
20000
30000
40000
50000
Goal

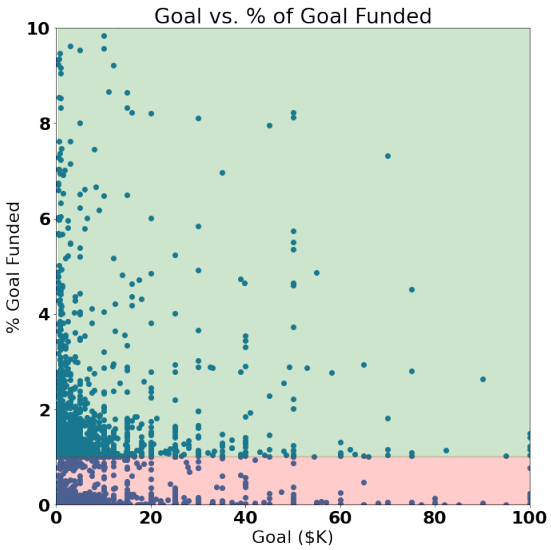

Hist n_bins=200, Range: 100 – 50K
Goal
Pledged
Count
0
50
100
150
200
0
10000
20000
30000
40000
50000
Pledged


Goal and Pledged Distributions
1e−5
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Probability
goal
converted_pledged_amount
0
20000
40000
60000
80000
100000
$


Distribution: % of Goal Funded
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Probability
pledge_pct_goal
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
% of Goal Funded


Distribution: Campaign Lengths
campaign_length
Probability
Campaign Length (Days)
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0


Goal vs. Pledged Amount ($K)
Pledged ($K)
Goal ($K)
0
20
40
60
80
100


Goal vs. % of Goal Funded
% Goal Funded
Goal ($K)
0
2
4
6
8
10
0
20
40
60
80
100

[15]:
```
plot_train_test_target(df_train, df_test)
```