Investment Assignment

By Ayush Mandawara

Imports

Loading Data to Dataframes

Understanding the Data

Feature Extraction

In companies dataframe, the columns of interest are

In round2 dataframe, the columns of interest are

Null Value Analysis - round2

Treating missing values - round2

Boxplot to perform outlier analysis for `raised_amount_usd` column

round2 missing values treated

Null Value Analysis - Companies

Treating missing values - companies

companies missing value treated

Permalink Analysis - round2 vs companies

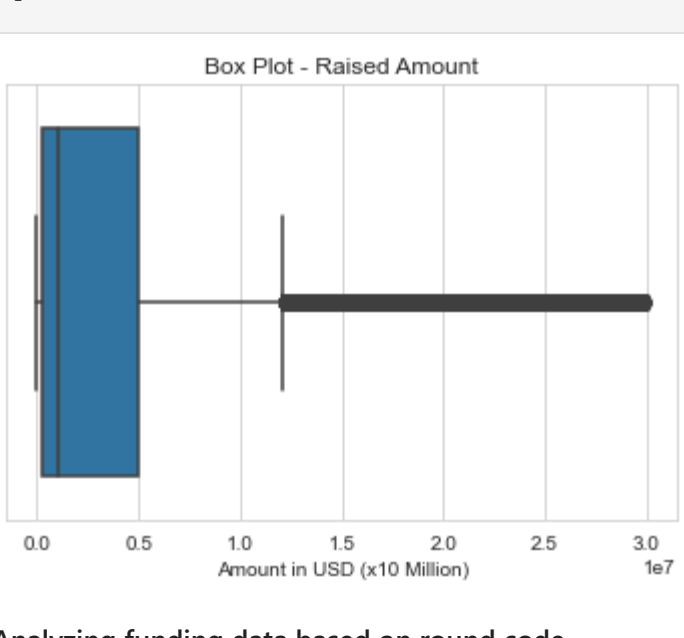
Merging round2 & companies dataframe after treatment

Understand the Data Set

Funding Analysis

In [64]:

```
# Box Plot of raised_amount_usd after removing outliers
rounds2_df_outlier_removed = rounds2_df_treated[rounds2_df_treated.raised_amount_usd < rounds2_df_treated.raise
amount_box_plot = sns.boxplot(x = 'raised_amount_usd', data=rounds2_df_outlier_removed)
amount_box_plot.set(xlabel='Amount in USD (x10 Million)')
plt.title('Box Plot - Raised Amount')
plt.savefig('boxplot_outlier_removed_raised_amount_usd.png')
plt.show()
```



Analyzing funding data based on round code

We know that apart from funding round type, funding round code (Series A, B, C..) also influences the amount of investment received.
It is known that 70% data is missing for round code, we just want to see if there is any pattern in the data which does exist
To get a rough idea,

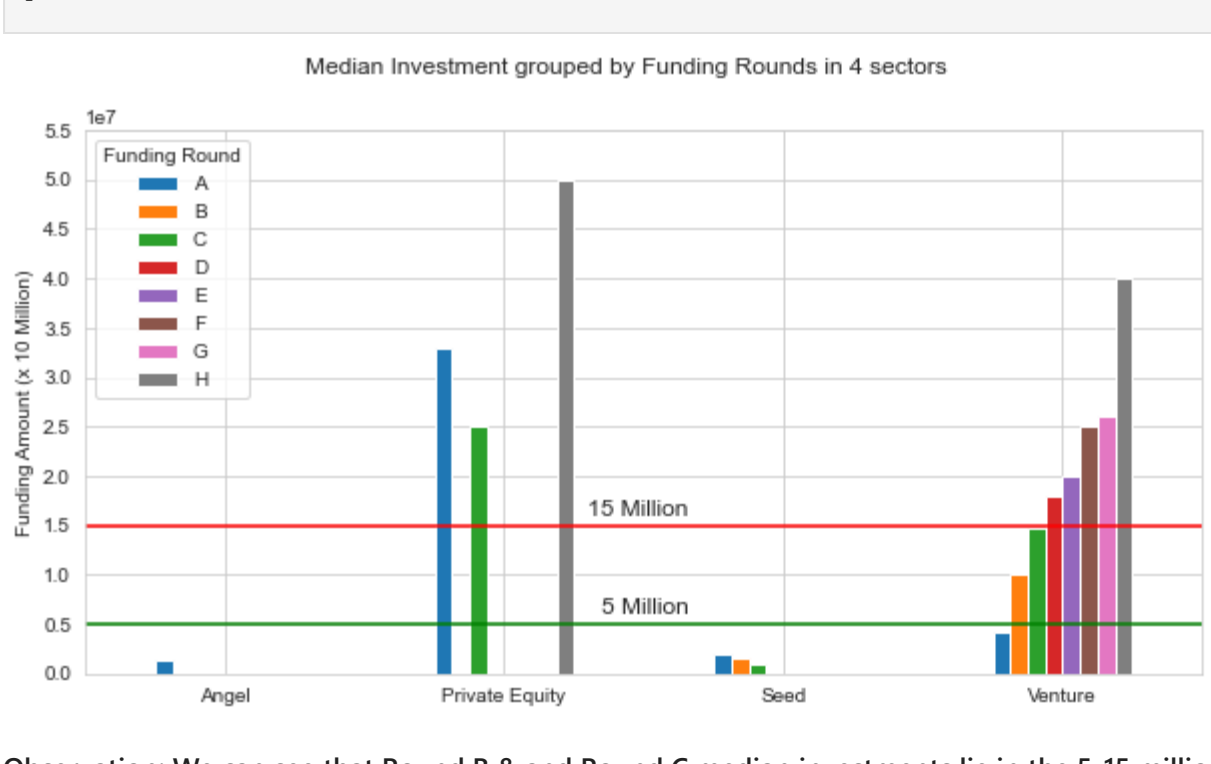
- we will use the original dataset (rounds2),
- to see median investments of each investment type, '
- further categorized by round code

In [65]:

```
fig, ax1 = plt.subplots()
rounds2_funding_round_type = rounds2[rounds2['funding_round_type'].isin(required_investment_types)]
rounds2_funding_round_type.head()

median_piv_rounds2_df = rounds2_funding_round_type.pivot_table(
    index='funding_round_type',
    values='raised_amount_usd',
    columns='funding_round_code',
    aggfunc='median'
)
median_piv_rounds2_df.loc[:, :].plot(kind='bar', figsize=(10,5), legend=True, ax=ax1)
plt.suptitle('Median Investment grouped by Funding Rounds in 4 sectors')
ax1.set_ylabel('Funding Amount (x 10 Million)')
ax1.set_xlabel('')
ax1.set_xticklabels(['Angel', 'Private Equity', 'Seed', 'Venture'], rotation=0)
plt.yticks(np.arange(0, max(ax1.get_yticks()), 5000000))
plt.axhline(y=15000000, color='r', linestyle='-')
plt.axhline(y=5000000, color='g', linestyle='-')
plt.text(1.35, 6000000, '5 Million', fontsize = 12)
plt.text(1.3, 16000000, '15 Million', fontsize = 12)
ax1.legend(title='Funding Round')
ax1.annotate('15 Million', xy=(5, 5))

plt.savefig('median_investments_by_rounds_from_original_rounds_df.png')
plt.show()
```



Observation: We can see that Round B & and Round C median investments lie in the 5-15 million bracket set by Spark Funds
Combining the information from outlier anlysis for raised amount usd, & Funding Analysis we conclude that,

- `Median` is a better representative for this particular dataset.
- As per business requirement, (5-15 Million USD investment per round), `Venture Funding` is the most appropriate type of funding for Spark Funds

Representative Values of Investments for Each of these Funding Types

Description	Value
Representative funding amount of venture type	5000000
Representative funding amount of angel type	400000
Representative funding amount of seed type	275000
Representative funding amount of private equity type	20000000
Best Funding type for Range between 5-15 Million	Venture Funding

---

Filtering data frame to only contain `Venture` funding

In [66]:

```
venture_fund = master_frame[master_frame.funding_round_type == 'venture']
venture_fund.head()
```

Out[66]:

	permalink	funding_round_type	raised_amount_usd	category_list	country_code
0	/organization/-fame	venture	10000000.0	Media	IND
1	/organization/-qounter	venture	5000000.0	Application Platforms|Real Time|Social Network...	USA
3	/organization/-the-one-of-them-inc-	venture	3406878.0	Apps|Games|Mobile	USA
4	/organization/0-6-com	venture	2000000.0	Curated Web	CHN
5	/organization/004-technologies	venture	5000000.0	Software	USA

Since we have only picked venture funding type for futhere analysis, we can drop `funding_round_type` column from this data frame

In [67]:

```
venture_fund = venture_fund.drop('funding_round_type', axis=1)
```

In [68]:

```
venture_fund.head()
```

Out[68]:

	permalink	raised_amount_usd	category_list	country_code
0	/organization/-fame	10000000.0	Media	IND
1	/organization/-qounter	5000000.0	Application Platforms|Real Time|Social Network...	USA
3	/organization/-the-one-of-them-inc-	3406878.0	Apps|Games|Mobile	USA
4	/organization/0-6-com	2000000.0	Curated Web	CHN
5	/organization/004-technologies	5000000.0	Software	USA

In [69]:

```
venture_fund.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 55494 entries, 0 to 114946
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   permalink          55494 non-null  object 
 1   raised_amount_usd  55494 non-null  float64
 2   category_list      55494 non-null  object 
 3   country_code       55494 non-null  object 
dtypes: float64(1), object(3)
memory usage: 2.1+ MB
```

In [70]:

```
venture_fund.describe()
```

Out[70]:

	raised_amount_usd
count	5.549400e+04
mean	1.110852e+07
std	8.217658e+07
min	0.000000e+00
25%	2.000000e+06
50%	5.000000e+06
75%	1.050000e+07
max	1.760000e+10

Country Analysis

In [71]:

```
venture_fund.head()
```

Out[71]:

	permalink	raised_amount_usd	category_list	country_code
0	/organization/-fame	10000000.0	Media	IND
1	/organization/-qounter	5000000.0	Application Platforms|Real Time|Social Network...	USA
3	/organization/-the-one-of-them-inc-	3406878.0	Apps|Games|Mobile	USA
4	/organization/0-6-com	2000000.0	Curated Web	CHN
5	/organization/004-technologies	5000000.0	Software	USA

Countries which received the Highest Investment

- Group by `country_code`
- Aggreate over `raised_amount_usd` to get Total Funding received by Country
- Sort Values from highest total funding to lowest

In [72]:

```
top_9_countries = venture_fund.groupby(['country_code'])['raised_amount_usd'].agg(['count', 'sum']).sort_values
top_9_countries
```

Out[72]:

country_code	count	sum
USA	41247	4.629720e+11
CHN	2087	4.245042e+10
GBR	2303	2.148563e+10
IND	992	1.523186e+10
CAN	1346	1.003333e+10
FRA	985	7.599537e+09
DEU	786	7.486960e+09
ISR	910	7.477515e+09
JPN	318	3.583677e+09

Observation: The top 9 most heavily invested countries (in terms of Total Amount Invested) are USA, China, United Kingdom, India, Canada, France, Germany, Israel and Japan.
We need to identify the top 3 English speaking countries so that Spark Funds can communicate with companies easily.
As per this reference, we can add a column to identify English Speaking countryies.

In [73]:

```
top_9_countries['English'] = [True, False, True, True, True, False, False, True,  False]
```

In [74]:

```
top_9_countries
```

Out[74]:

country_code	count	sum	English
USA	41247	4.629720e+11	True
CHN	2087	4.245042e+10	False
GBR	2303	2.148563e+10	True
IND	992	1.523186e+10	True
CAN	1346	1.003333e+10	True
FRA	985	7.599537e+09	False
DEU	786	7.486960e+09	False
ISR	910	7.477515e+09	True
JPN	318	3.583677e+09	False

In [75]:

```
top_9_countries[top_9_countries['English'] == True]
```

Out[75]:

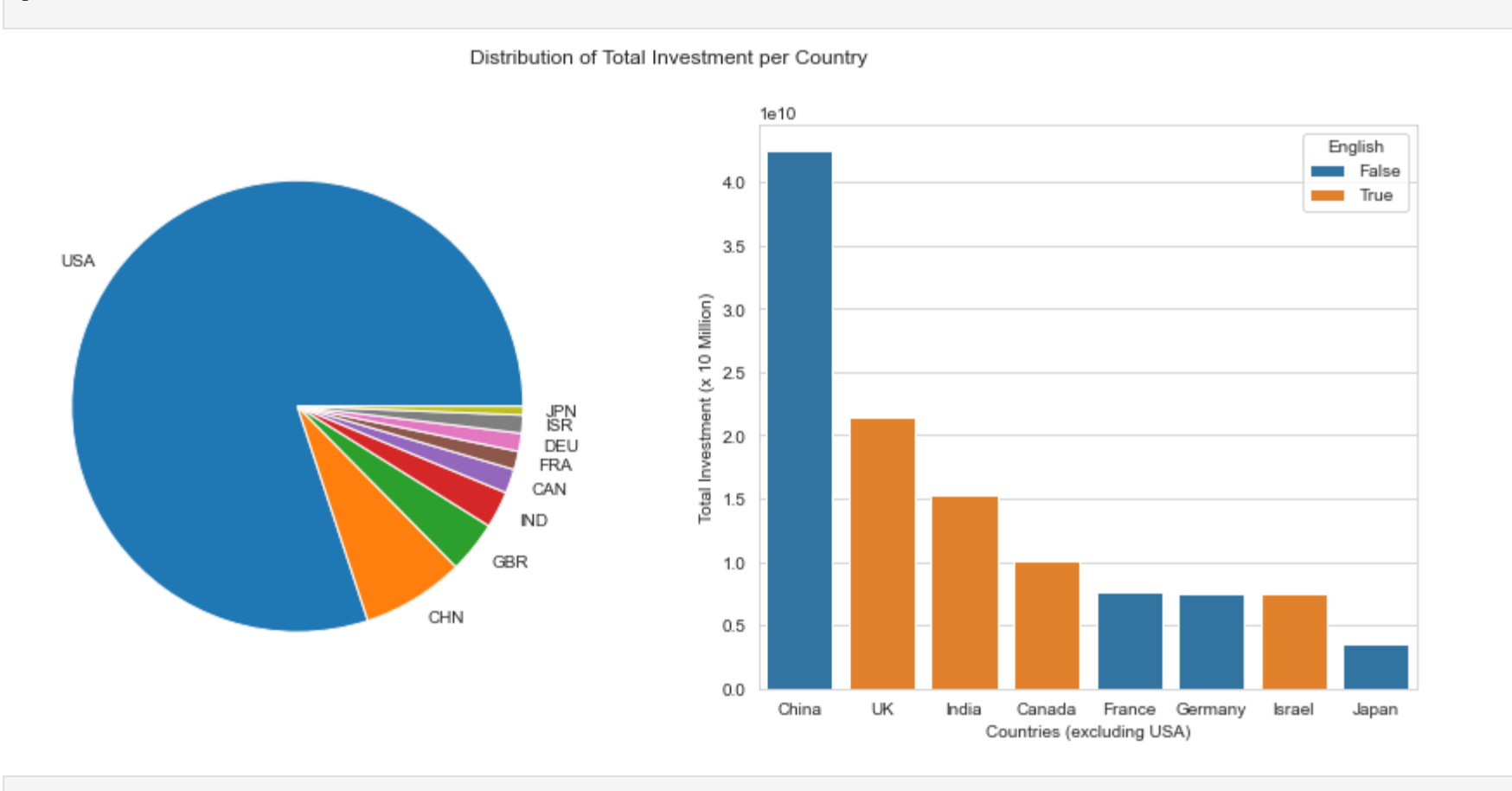
country_code	count	sum	English
USA	41247	4.629720e+11	True
GBR	2303	2.148563e+10	True
IND	992	1.523186e+10	True
CAN	1346	1.003333e+10	True
ISR	910	7.477515e+09	True

Plot - Total Investment in Country

In [76]:

```
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(15,6))

plt.suptitle('Distribution of Total Investment per Country')
top_9_countries['sum'].plot(kind='pie', ax=ax1)
ax2 = sns.barplot(x='country_code', y='sum', hue='English', data=top_9_countries[1:].reset_index(), dodge=False
ax1.set_ylabel('')
ax2.set_xlabel('Countries (excluding USA)')
ax2.set_xticklabels(['China', 'UK', 'India', 'Canada', 'France', 'Germany', 'Israel', 'Japan'])
ax2.set_ylabel('Total Investment (x 10 Million)')
fig.savefig('total_investment_distribution.png')
plt.show()
```



In [77]:

```
top3_countries = top_9_countries[top_9_countries['English'] == True][:3].index.tolist()
top3_countries
```

Out[77]: `['USA', 'GBR', 'IND']`

Observation: The top3 English Speaking countries based on highest total investment amount are

- United States of America (USA)
- United Kingdom (UK)
- India (IND)

Analysing the Top 3 English-Speaking Countries

Description	Value
Top English speaking country	USA
Second English speaking country	UK
Third English speaking country	IND

Mapping Primary Category to Main Sector

Loading and understanding the mapping dataset

In [78]:

```
mapping = pd.read_csv('mapping.csv')
mapping.head()
```

Out[78]:

	category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others	Social, Finance, Analytics, Advertising
0	NaN	0	1	0	0	0	0	0	0	0
1	3D	0	0	0	0	0	1	0	0	0
2	3D Printing	0	0	0	0	0	1	0	0	0
3	3D Technology	0	0	0	0	0	1	0	0	0
4	Accounting	0	0	0	0	0	0	0	0	1

In [79]:

```
mapping.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 688 entries, 0 to 687
Data columns (total 10 columns):
 #   Column                                   Non-Null Count  Dtype 
---  ------                                   --------------  ----- 
 0   category_list                            687 non-null    object
 1   Automotive & Sports                      688 non-null    int64 
 2   Blanks                                   688 non-null    int64 
 3   Cleantech / Semiconductors               688 non-null    int64 
 4   Entertainment                            688 non-null    int64 
 5   Health                                   688 non-null    int64 
 6   Manufacturing                            688 non-null    int64 
 7   News, Search and Messaging               688 non-null    int64 
 8   Others                                   688 non-null    int64 
 9   Social, Finance, Analytics, Advertising  688 non-null    int64 
dtypes: int64(9), object(1)
memory usage: 53.9+ KB
```

In [80]:

```
# checking if multiple sectors belong to a category
any(mapping.sum(axis=1) > 1)
```

Out[80]: `False`

In [81]:

```
# checking if there are any rows where no column has true value
sum(mapping.sum(axis=1))
```

Out[81]: `688`

From the dataset, we can see that

- Each sector has one category
- If a category belongs to a main sector, the cell value is set to 1 Else it is set to 0
- The data is in Wide Format see #6

Since we need to map the categories to respective sectors, we can convert it to long format and only keep entries where the value is one.

In [82]:

```
# Extracting value & id list
value_vars = list(mapping.columns[1:])
id_vars = list(np.setdiff1d(mapping.columns, value_vars))

# print(value_vars, ':', id_vars)

# using pd.melt to convert from wide to long format (see #7)
category_sector_mapping = pd.melt(mapping, id_vars = id_vars, value_vars=value_vars)
```

In [83]:

```
category_sector_mapping.head(9)
```

Out[83]:

	category_list	variable	value
0	NaN	Automotive & Sports	0
1	3D	Automotive & Sports	0
2	3D Printing	Automotive & Sports	0
3	3D Technology	Automotive & Sports	0
4	Accounting	Automotive & Sports	0
5	Active Lifestyle	Automotive & Sports	0
6	Ad Targeting	Automotive & Sports	0
7	Advanced Materials	Automotive & Sports	0
8	Adventure Travel	Automotive & Sports	1

In [84]:

```
# keeping only those entries where value is one
category_sector_mapping = category_sector_mapping[category_sector_mapping.value == 1]
```

In [85]:

```
category_sector_mapping.head()
```

Out[85]:

	category_list	variable	value
8	Adventure Travel	Automotive & Sports	1
14	Aerospace	Automotive & Sports	1
45	Auto	Automotive & Sports	1
46	Automated Kiosk	Automotive & Sports	1
47	Automotive	Automotive & Sports	1

In [86]:

```
# dropping value column
category_sector_mapping.drop('value', axis=1, inplace=True)
category_sector_mapping.head()
```

Out[86]:

	category_list	variable
8	Adventure Travel	Automotive & Sports
14	Aerospace	Automotive & Sports
45	Auto	Automotive & Sports
46	Automated Kiosk	Automotive & Sports
47	Automotive	Automotive & Sports

In [87]:

```
category_sector_mapping.rename(columns={'variable': 'sector'}, inplace=True)
```

In [88]:

```
category_sector_mapping.head()
```

Out[88]:

	category_list	sector
8	Adventure Travel	Automotive & Sports
14	Aerospace	Automotive & Sports
45	Auto	Automotive & Sports
46	Automated Kiosk	Automotive & Sports
47	Automotive	Automotive & Sports

Sectors have been mapped to respective Category

In [89]:

```
venture_fund.head()
```

Out[89]:

	permalink	raised_amount_usd	category_list	country_code
0	/organization/-fame	10000000.0	Media	IND
1	/organization/-qounter	5000000.0	Application Platforms|Real Time|Social Network...	USA
3	/organization/-the-one-of-them-inc-	3406878.0	Apps|Games|Mobile	USA
4	/organization/0-6-com	2000000.0	Curated Web	CHN
5	/organization/004-technologies	5000000.0	Software	USA

Normalizing category_list

We can see that our funding dataset contains a category_list which has multiple categories in it separated by `|`.
As per the business rule given by Spark Funds, we will only keep the primary category.

In [90]:

```
venture_fund.category_list = venture_fund.category_list.apply(lambda x: str(x).lower().split('|')[0])
venture_fund.head()
```

Out[90]:

	permalink	raised_amount_usd	category_list	country_code
0	/organization/-fame	10000000.0	media	IND
1	/organization/-qounter	5000000.0	application platforms	USA
3	/organization/-the-one-of-them-inc-	3406878.0	apps	USA
4	/organization/0-6-com	2000000.0	curated web	CHN
5	/organization/004-technologies	5000000.0	software	USA

In [91]:

```
category_sector_mapping.category_list = category_sector_mapping.category_list.apply(lambda x: str(x).lower())
```

In [92]:

```
category_sector_mapping.head()
```

Out[92]:

	category_list	sector
8	adventure travel	Automotive & Sports
14	aerospace	Automotive & Sports
45	auto	Automotive & Sports
46	automated kiosk	Automotive & Sports
47	automotive	Automotive & Sports

In [93]:

```
set(category_sector_mapping.category_list) - set(venture_fund.category_list)
```

Out[93]:

```
{'0notechnology',
 '0tural language processing',
 '0tural resources',
 '0vigation',
 'a0lytics',
 'alter0tive medicine',
 'aquaculture',
 'big data a0lytics',
 'biomass power generation',
 'boating industry',
 'building owners',
 'business a0lytics',
 'can0bis',
 'career ma0gement',
 'casual games',
 'charities',
 'chi0 internet',
 'cloud ma0gement',
 'contact ma0gement',
 'digital rights ma0gement',
 'digital sig0ge',
 'direct advertising',
 'document ma0gement',
 'educatio0l games',
 'email newsletters',
 'energy ma0gement',
 'energy storage',
 'event ma0gement',
 'farmers market',
 'fi0nce',
 'fi0nce technology',
 'fi0ncial exchanges',
 'fi0ncial services',
 'field support services',
 'film distribution',
 'flash sales',
 'fleet ma0gement',
 'flowers',
 'fmcg',
 'forums',
 'fuel cells',
 'funeral industry',
 'gay & lesbian',
 'gover0nce',
 'green',
 'group buying',
 'handmade',
 'hedge funds',
 'homeless shelter',
 'ict',
 'identity ma0gement',
 'impact investing',
 'indians',
 'innovation ma0gement',
 'intellectual asset ma0gement',
 'investment ma0gement',
 'it ma0gement',
 'jour0lism',
 'k-12 education',
 'knowledge ma0gement',
 'landscaping',
 'lead ma0gement',
 'lifestyle products',
 'low bid auctions',
 'medical professio0ls',
 'micro-enterprises',
 'minerals',
 'mobile a0lytics',
 'mobile social',
 'mobile video',
 'monetization',
 'moneymaking',
 'multi-level marketing',
 'musicians',
 'nan',
 'organic',
 'performance marketing',
 'perso0l branding',
 'perso0l data',
 'perso0l fi0nce',
 'perso0l health',
 'perso0lization',
 'pervasive computing',
 'postal and courier services',
 'predictive a0lytics',
 'private social networking',
 'productivity',
 'professio0l networking',
 'professio0l services',
 'project ma0gement',
 'promotio0l',
 'property ma0gement',
 'q&a',
 'realtors',
 'renewable tech',
 'risk ma0gement',
 'shoes',
 'smart grid',
 'social media ma0gement',
 'social media platforms',
 'social recruiting',
 'speech recognition',
 'subscription service',
 'supply chain ma0gement',
 'task ma0gement',
 'universities',
 'university students',
 'unmanned air systems',
 'usability',
 'user experience design',
 'vending and concessions',
 'veteri0ry',
 'waste ma0gement',
 'wholesale',
 'women'}
```

We can see that there is some problem in category_list column of the mappings dataset.

- the character `na` seems to have been replaced by `0`

In [94]:

```
category_sector_mapping[category_sector_mapping.category_list.str.contains('0')]
```

Out[94]:

	category_list	sector
1616	energy ma0gement	Cleantech / Semiconductors
1847	0notechnology	Cleantech / Semiconductors
1848	0tural language processing	Cleantech / Semiconductors
1849	0tural resources	Cleantech / Semiconductors
2048	waste ma0gement	Cleantech / Semiconductors
2262	digital rights ma0gement	Entertainment
2263	digital sig0ge	Entertainment
2287	educatio0l games	Entertainment
2320	event ma0gement	Entertainment
2772	alter0tive medicine	Health
2841	can0bis	Health
3189	medical professio0ls	Health
3269	perso0l health	Health
3416	veteri0ry	Health
4231	chi0 internet	News, Search and Messaging
4242	cloud ma0gement	News, Search and Messaging
4519	jour0lism	News, Search and Messaging
4602	0vigation	News, Search and Messaging
4674	professio0l networking	News, Search and Messaging
5026	document ma0gement	Others
5059	enterprise 2.0	Others
5097	fleet ma0gement	Others
5126	gover0nce	Others
5188	intellectual asset ma0gement	Others
5205	it ma0gement	Others
5210	knowledge ma0gement	Others
5216	lead ma0gement	Others
5331	perso0l data	Others
5334	perso0lization	Others
5363	professio0l services	Others
5364	project ma0gement	Others
5449	supply chain ma0gement	Others
5453	task ma0gement	Others
5526	a0lytics	Social, Finance, Analytics, Advertising
5563	big data a0lytics	Social, Finance, Analytics, Advertising
5583	business a0lytics	Social, Finance, Analytics, Advertising
5595	career ma0gement	Social, Finance, Analytics, Advertising
5649	contact ma0gement	Social, Finance, Analytics, Advertising
5777	fi0nce	Social, Finance, Analytics, Advertising
5778	fi0nce technology	Social, Finance, Analytics, Advertising
5779	fi0ncial exchanges	Social, Finance, Analytics, Advertising
5780	fi0ncial services	Social, Finance, Analytics, Advertising
5857	identity ma0gement	Social, Finance, Analytics, Advertising
5874	innovation ma0gement	Social, Finance, Analytics, Advertising
5888	investment ma0gement	Social, Finance, Analytics, Advertising
5952	mobile a0lytics	Social, Finance, Analytics, Advertising
6018	perso0l branding	Social, Finance, Analytics, Advertising
6020	perso0l fi0nce	Social, Finance, Analytics, Advertising
6038	predictive a0lytics	Social, Finance, Analytics, Advertising
6053	promotio0l	Social, Finance, Analytics, Advertising
6054	property ma0gement	Social, Finance, Analytics, Advertising
6079	risk ma0gement	Social, Finance, Analytics, Advertising
6116	social media ma0gement	Social, Finance, Analytics, Advertising

We can fix this problem by changing all `0` s back to `na`, except for the entry where the number is valid i.e Enterprise 2.0

In [95]:

```
category_sector_mapping.category_list = category_sector_mapping.category_list.apply(lambda x: str(x).replace("(
category_sector_mapping.head()
```

Out[95]:

	category_list	sector
8	adventure travel	Automotive & Sports
14	aerospace	Automotive & Sports
45	auto	Automotive & Sports
46	automated kiosk	Automotive & Sports
47	automotive	Automotive & Sports

In [96]:

```
# checking how many category list values cannot be mapped.
set(venture_fund.category_list) - set(category_sector_mapping.category_list)
```

Out[96]:

```
{'adaptive equipment',
 'biotechnology and semiconductor',
 'english-speaking',
 'enterprise hardware',
 'greentech',
 'natural gas uses',
 'nightlife',
 'product search',
 'racing',
 'rapidly expanding',
 'retirement',
 'specialty retail'}
```

Categoy List has been cleaned and normalized in both the dataframes.

Merging mapping data to venture fund data further analysis

In [97]:

```
category_sector_df = pd.merge(venture_fund, category_sector_mapping, on='category_list')
```

In [98]:

```
category_sector_df.head()
```

Out[98]:

	permalink	raised_amount_usd	category_list	country_code	sector
0	/organization/-fame	10000000.0	media	IND	Entertainment
1	/organization/90min	15000000.0	media	GBR	Entertainment
2	/organization/90min	5800000.0	media	GBR	Entertainment
3	/organization/90min	18000000.0	media	GBR	Entertainment
4	/organization/all-def-digital	5000000.0	media	USA	Entertainment

In [99]:

```
category_sector_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 55480 entries, 0 to 55479
Data columns (total 5 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   permalink          55480 non-null  object 
 1   raised_amount_usd  55480 non-null  float64
 2   category_list      55480 non-null  object 
 3   country_code       55480 non-null  object 
 4   sector             55480 non-null  object 
dtypes: float64(1), object(4)
memory usage: 2.5+ MB
```

In [100…

```
venture_fund.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 55494 entries, 0 to 114946
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   permalink          55494 non-null  object 
 1   raised_amount_usd  55494 non-null  float64
 2   category_list      55494 non-null  object 
 3   country_code       55494 non-null  object 
dtypes: float64(1), object(3)
memory usage: 4.1+ MB
```

In [101…

```
len(venture_fund) - len(category_sector_df)
```

Out[101… `14`

Observation: We lost 14 values due to merge (~0.02%)

In [102…

```
# checking if any sector is missing
any(category_sector_df.sector.isnull())
```

Out[102… `False`

Sector Analysis

In [103…

```
top3_countries_sectors = category_sector_df[category_sector_df.country_code.isin(top3_countries)]
```

In [104…

```
top3_countries_sectors.head()
```