In [1]:
```
# pip install pyathena
```

In [2]:
```
# pip install PyAthena[Pandas]
```

import sqlalchemy

In [3]:
```
from urllib.parse import quote_plus
from sqlalchemy.engine import create_engine
from sqlalchemy.sql.expression import select
from sqlalchemy.sql.functions import func
from sqlalchemy.sql.schema import Table, MetaData
import sqlalchemy
```

import athena

In [4]:
```
from pyathena import connect
```

In [5]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
%config InlineBackend.figure_format='retina'
import seaborn as sns
```

In [6]:
```
database = 'dsoaws'
table = 'amazon_reviews_tsv'
bucket = 'data-science-on-aws22'
```

create connection engine

In [7]:
```
engine = create_engine("awsathena+rest://AKIAQUEPHTPDTCLPBSHY:rcpmIAV4pUgVxOmL7F8PLDdhn5
                       "default?s3_staging_dir=s3://data-science-on-aws22/athena/staging
```

In [8]:
```
# conn_str = "awsathena+rest://{aws_access_key_id}:{aws_secret_access_key}@athena.us-eas
#             "{schema_name}?s3_staging_dir={s3_staging_dir}"

# engine = create_engine(conn_str.format(
#     aws_access_key_id=quote_plus("AKIAQUEPHTPDTCLPBSHY"),
#     aws_secret_access_key=quote_plus("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"),
#     region_name="us-east-1",
#     schema_name="default",
#     s3_staging_dir=quote_plus("s3://{0}/path/to/").format(bucket)))
#     session_token = 'kiane'
```

the sql statement

In [9]:
```
sql_statement="""
SELECT DISTINCT product_category from {0}.{1}
ORDER BY product_category
""".format(database,table)
```

In [10]:
```
pd.read_sql(sql_statement, con=engine)
```

Out[10]:

	product_category
0	Apparel
1	Automotive

2	Baby
3	Beauty
4	Books
5	Camera
6	Digital_Ebook_Purchase
7	Digital_Music_Purchase
8	Digital_Software
9	Digital_Video_Download
10	Digital_Video_Games
11	Electronics
12	Furniture
13	Gift Card
14	Grocery
15	Health & Personal Care
16	Home
17	Home Entertainment
18	Home Improvement
19	Jewelry
20	Kitchen
21	Lawn and Garden
22	Luggage
23	Major Appliances
24	Mobile_Apps
25	Mobile_Electronics
26	Music
27	Musical Instruments
28	Office Products
29	Outdoors
30	PC
31	Personal_Care_Appliances
32	Pet Products
33	Shoes
34	Software
35	Sports
36	Tools
37	Toys
38	Video
39	Video DVD

40	Video Games
41	Watches
42	Wireless

Which product categories are the highest rated by average rating?

In [11]:

```
sql2 = """SELECT product_category, AVG(star_rating) AS avg_star_rating
FROM {0}.{1}
GROUP BY product_category
ORDER BY avg_star_rating DESC
""".format(database,table)
```

In [12]:

```
pd.read_sql(sql2, con=engine)
```

Out[12]:

	product_category	avg_star_rating
0	Gift Card	4.731363
1	Digital_Music_Purchase	4.642891
2	Music	4.436624
3	Books	4.341658
4	Grocery	4.312219
5	Digital_Ebook_Purchase	4.308775
6	Video DVD	4.302017
7	Tools	4.261769
8	Musical Instruments	4.251103
9	Automotive	4.246302
10	Shoes	4.241260
11	Outdoors	4.240019
12	Sports	4.229365
13	Luggage	4.223391
14	Toys	4.211735
15	Kitchen	4.207424
16	Digital_Video_Download	4.201208
17	Video	4.191511
18	Beauty	4.187224
19	Home Improvement	4.182270
20	Home	4.178399
21	Baby	4.162683
22	Health & Personal Care	4.161833
23	Jewelry	4.144090
24	Pet Products	4.143653

25	Watches	4.138283
26	Camera	4.127015
27	Apparel	4.105229
28	Lawn and Garden	4.093177
29	PC	4.086444
30	Furniture	4.083949
31	Office Products	4.072539
32	Video Games	4.059893
33	Electronics	4.035507
34	Mobile_Apps	3.981594
35	Personal_Care_Appliances	3.977402
36	Home Entertainment	3.902123
37	Wireless	3.891779
38	Digital_Video_Games	3.853126
39	Mobile_Electronics	3.763163
40	Major Appliances	3.716185
41	Software	3.567035
42	Digital_Software	3.539330

In [13]:

```
result = pd.read_sql(sql2, con=engine)
```

Set the size of plot canvas

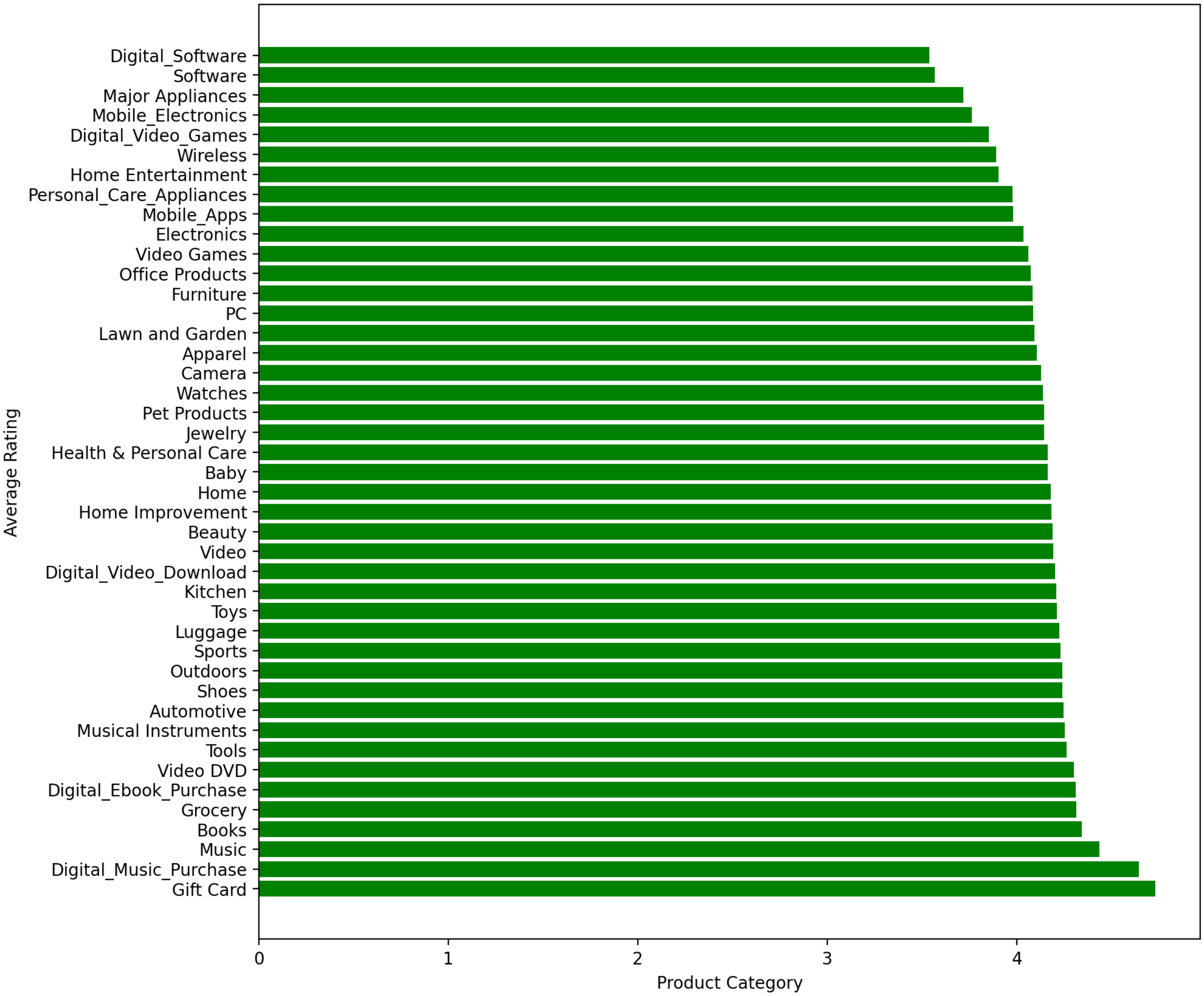
In [14]:

```
plt.rcParams['figure.figsize'] = [10, 10]
```

In [15]:

```
plt.barh(result['product_category'],result['avg_star_rating'], color ='green')

plt.xlabel("Product Category")
plt.ylabel("Average Rating")
plt.show()
```



Which product categories have the most reviews?

In [16]:

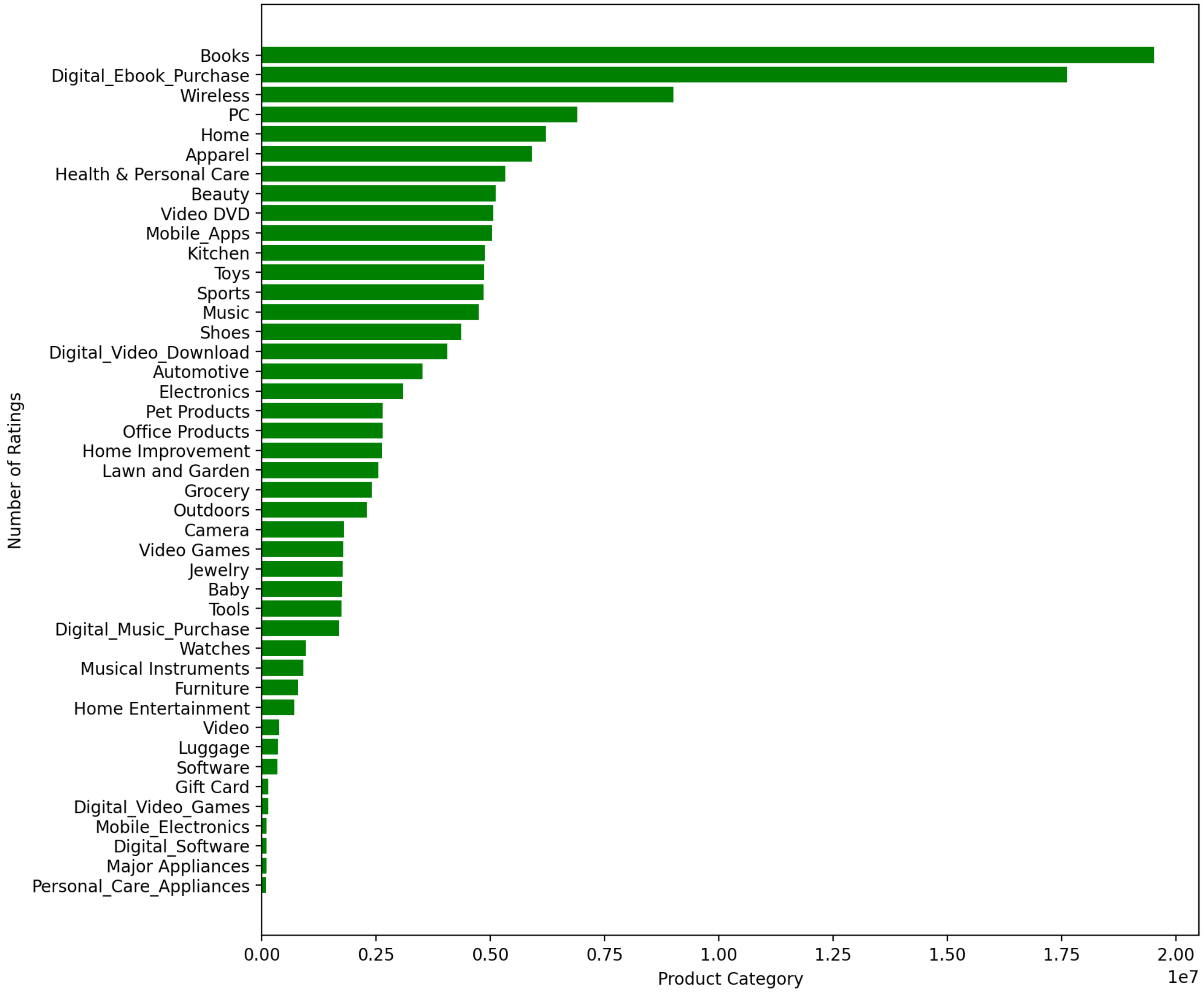
```
sql3 = """SELECT product_category, COUNT(star_rating) AS count_star_rating
FROM {0}.{1}
GROUP BY product_category
ORDER BY count_star_rating
""".format(database,table)

result2 = pd.read_sql(sql3, con=engine)
```

In [17]:

```
plt.barh(result2['product_category'],result2['count_star_rating'], color ='green')

plt.xlabel("Product Category")
plt.ylabel("Number of Ratings")
plt.show()
```



When did each product category become available in the Amazon catalog

I need to check first the column schema

In [18]:

```
sql_test = """SELECT *
FROM {0}.{1}
LIMIT 3
""".format(database,table)

result_all = pd.read_sql(sql_test, con=engine)
result_all
```

Out[18]:

	marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	sta
0	US	12076615	RQ58W7SMO911M	0385730586	122662979	Sisterhood of the Traveling Pants (Book 1)	Books	
1	US	12703090	RF6IUKMGL8SF	0811828964	56191234	The Bad Girl's Guide to Getting What You Want	Books	
2	US	12257412	R1DOSHH6AI622S	1844161560	253182049	Eisenhorn (A	Books	

Warhammer
40,000
Omnibus)

Then run the query after investigation.

In [21]:
```
sql4 = """SELECT product_category, MIN(EXTRACT(YEAR FROM CAST(review_date AS DATE))) AS
FROM {0}.{1}
GROUP BY product_category
""".format(database,table)

result4 = pd.read_sql(sql4, con=engine)
result4
```

Out[21]:

	product_category	release_year
0	Wireless	1998
1	Personal_Care_Appliances	2000
2	PC	1999
3	Home Improvement	1999
4	Mobile_Apps	2010
5	Gift Card	2004
6	Tools	1999
7	Beauty	2000
8	Automotive	1999
9	Major Appliances	2000
10	Books	1995
11	Office Products	1998
12	Musical Instruments	1999
13	Digital_Music_Purchase	2000
14	Digital_Ebook_Purchase	1999
15	Electronics	1999
16	Sports	1997
17	Home	1998
18	Mobile_Electronics	2001
19	Baby	1999
20	Digital_Software	2008
21	Jewelry	2001
22	Music	1995
23	Digital_Video_Games	2006
24	Health & Personal Care	1999
25	Video Games	1997
26	Furniture	2000

27	Video	1995
28	Luggage	2002
29	Shoes	1999
30	Home Entertainment	1998
31	Outdoors	1999
32	Apparel	2000
33	Camera	1998
34	Pet Products	1998
35	Lawn and Garden	1999
36	Digital_Video_Download	2000
37	Video DVD	1996
38	Software	1998
39	Kitchen	2000
40	Watches	2001
41	Toys	1997
42	Grocery	1999

What is the breakdown of star ratings (1–5) per product category?

In [23]:

```
sql5 = """SELECT product_category, star_rating, COUNT(*) AS count_reviews
FROM {0}.{1}
GROUP BY product_category, star_rating
ORDER BY product_category ASC, star_rating DESC,count_reviews
""".format(database,table)

result5 = pd.read_sql(sql5, con=engine)
result5
```

Out[23]:

	product_category	star_rating	count_reviews
0	Apparel	5	3320566
1	Apparel	4	1147237
2	Apparel	3	623471
3	Apparel	2	369601
4	Apparel	1	445458
...	...	...	...
210	Wireless	5	4824783
211	Wireless	4	1501327
212	Wireless	3	815205
213	Wireless	2	598330
214	Wireless	1	1262376

215 rows × 3 columns

In [28]:

```
sql5_2 = """SELECT star_rating, COUNT(*) AS count_reviews,
FROM {0}.{1}
GROUP BY star_rating
ORDER BY star_rating DESC
""".format(database,table)

result5_2 = pd.read_sql(sql5_2, con=engine)
result5_2
```

Out[28]:

	star_rating	count_reviews
0	5	93200812
1	4	26223470
2	3	12133927
3	2	7304430
4	1	12099639