Creating a Matrix of Gene Expression (UMI Counts)

In [81]:

```
import csv
import gzip
import os
import scipy.io
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
```

In [1]:

```
# define MEX directory
matrix_dir = "filtered_feature_bc_matrix"
# read in MEX format matrix as table
mat_filtered = scipy.io.mmread(os.path.join(matrix_dir, "matrix.mtx.gz"))

# list of transcript ids, e.g. 'ENSG00000187634'
features_path = os.path.join(matrix_dir, "features.tsv.gz")
feature_ids = [row[0] for row in csv.reader(gzip.open(features_path, mode="rt

# list of gene names, e.g. 'SAMD11'
gene_names = [row[1] for row in csv.reader(gzip.open(features_path, mode="rt"

# list of feature_types, e.g. 'Gene Expression'
feature_types = [row[2] for row in csv.reader(gzip.open(features_path, mode="

# list of barcodes, e.g. 'AAACATACAAAACG-1'
barcodes_path = os.path.join(matrix_dir, "barcodes.tsv.gz")
barcodes = [row[0] for row in csv.reader(gzip.open(barcodes_path, mode="rt"),
```

In [2]:

```
# transform table to pandas dataframe and label rows and columns
matrix = pd.DataFrame.sparse.from_spmatrix(mat_filtered)
matrix.columns = barcodes
matrix.insert(loc=0, column="feature_id", value=feature_ids)
matrix.insert(loc=1, column="gene", value=gene_names)
matrix.insert(loc=2, column="feature_type", value=feature_types)

# display matrix
print(matrix)

# save the table as a CSV (note the CSV will be a very large file)
matrix.to_csv("filtered_matrix.csv", index=False)
```

```
            feature_id         gene     feature_type  AAACAAGTA
TCTCCCA-1  \
0      ENSG00000243485  MIR1302-2HG  Gene Expression
0
1      ENSG00000237613      FAM138A  Gene Expression
0
2      ENSG00000186092        OR4F5  Gene Expression
0
3      ENSG00000238009   AL627309.1  Gene Expression
0
4      ENSG00000239945   AL627309.3  Gene Expression
0
...                ...          ...              ...
...
36596  ENSG00000277836   AC141272.1  Gene Expression
0
36597  ENSG00000278633   AC023491.2  Gene Expression
0
36598  ENSG00000276017   AC007325.1  Gene Expression
0
36599  ENSG00000278817   AC007325.4  Gene Expression
1
36600  ENSG00000277196   AC007325.2  Gene Expression
0

       AAACAATCTACTAGCA-1  AAACAGAGCGACTCCT-1  AAACAGCTTTCAGAAG
-1  \
0                       0                   0
0
1                       0                   0
0
2                       0                   0
0
3                       0                   0
```

```
0
4                       0                   0
0
...                   ...                 ...
...
36596                   0                   0
0
36597                   0                   0
0
36598                   0                   0
0
36599                   0                   1
0
36600                   0                   0
0

       AAACAGGGTCTATATT-1  AAACCCGAACGAAATC-1  AAACCGGAAATGTTAA
-1  ...  \
0                       0                   0
0  ...
1                       0                   0
0  ...
2                       0                   0
0  ...
3                       0                   0
0  ...
4                       0                   0
0  ...
...                   ...                 ...
...  ...
36596                   0                   0
0  ...
36597                   0                   0
0  ...
36598                   0                   0
0  ...
36599                   1                   0
0  ...
36600                   0                   0
0  ...

       TTGTGGTATAGGTATG-1  TTGTGTATGCCACCAA-1  TTGTGTTTCCCGAAAG
-1  \
0                       0                   0
0
1                       0                   0
0
2                       0                   0
0
3                       0                   0
0
```

```
4                       0                   0
0
...                   ...                 ...
...
36596                   0                   0
0
36597                   0                   0
0
36598                   0                   0
0
36599                   0                   0
0
36600                   0                   0
0

       TTGTTAGCAAATTCGA-1  TTGTTCAGTGTGCTAC-1  TTGTTGTGTGTCAAGA
-1  \
0                       0                   0
0
1                       0                   0
0
2                       0                   0
0
3                       0                   0
0
4                       0                   0
0
...                   ...                 ...
...
36596                   0                   0
0
36597                   0                   0
0
36598                   0                   0
0
36599                   0                   0
0
36600                   0                   0
0

       TTGTTTCACATCCAGG-1  TTGTTTCATTAGTCTA-1  TTGTTTCCATACAACT
-1  \
0                       0                   0
0
1                       0                   0
0
2                       0                   0
0
3                       0                   0
0
4                       0                   0
```

```
0
...                   ...                 ...
...
36596                   0                   0
0
36597                   0                   0
0
36598                   0                   0
0
36599                   0                   0
0
36600                   0                   0
0

       TTGTTTGTGTAAATTC-1
0                       0
1                       0
2                       0
3                       0
4                       0
...                   ...
36596                   0
36597                   0
36598                   0
36599                   1
36600                   0

[36601 rows x 3525 columns]
```

1. Basic Exploration:

In [4]:

```
#Start from here, above code was to create csv
matrix = pd.read_csv("filtered_matrix.csv")
```

In [52]:

```
matrix.loc[matrix["gene"] == "TFF3"]
```

Out[52]:

	feature_id	gene	feature_type	AAACAAGTATCTCCCA-1	AAACAATCTACTA
34246	ENSG00000160180	TFF3	Gene Expression	145	

1 rows × 3525 columns

In [31]:

```
matrix.loc[matrix["gene"] == "AKR1C2"]
```

Out[31]:

	feature_id	gene	feature_type	AAACAAGTATCTCCCA-1	AAACAATCTA
17578	ENSG00000151632	AKR1C2	Gene Expression	1	

1 rows × 3525 columns

2. Identifying Unexpressed Genes:

In [33]:

```
blank_list = []
for i in range(36601):
    if i%100 == 0:
        print(i)
    if matrix.iloc[i][3:].sum() == 0:
        blank_list.append(i)
```

```
34900
35000
35100
35200
35300
35400
35500
35600
35700
35800
35900
36000
36100
36200
36300
36400
36500
36600
```

In [59]:

```
print(len(blank_list))
blank_list[0:10]
```

```
14060
```

Out[59]:

```
[0, 1, 2, 4, 5, 7, 10, 12, 13, 19]
```

In [ ]:

```
#Ideas:
#Filter out zeros for all samples
#Summary statistics for each row
#Getting the top 5 most expressed genes for each coordinate and creating some
#Apply Decision Trees, RFCs, or Gradient Boosting onto each sample using Dr.
#Include all of this in paper
```

3. Creating Separate Tables for Genes and Expressions:

In [60]:

```
all_expr = matrix.drop(matrix.columns[[0,1,2]], axis=1)
```

In [61]:

```
all_expr
```

Out[61]:

	AAACAAGTATCTCCCA-1	AAACAATCTACTAGCA-1	AAACAGAGCGACTCCT-1	AAACA
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
...	...	...	...	
36596	0	0	0	
36597	0	0	0	
36598	0	0	0	
36599	1	0	1	
36600	0	0	0	

36601 rows × 3522 columns

In [62]:

```
all_names = matrix[matrix.columns[[0,1,2]]]
```

In [63]:

```
all_names
```

Out[63]:

	feature_id	gene	feature_type
0	ENSG00000243485	MIR1302-2HG	Gene Expression
1	ENSG00000237613	FAM138A	Gene Expression
2	ENSG00000186092	OR4F5	Gene Expression
3	ENSG00000238009	AL627309.1	Gene Expression
4	ENSG00000239945	AL627309.3	Gene Expression
...	...	...	...
36596	ENSG00000277836	AC141272.1	Gene Expression
36597	ENSG00000278633	AC023491.2	Gene Expression
36598	ENSG00000276017	AC007325.1	Gene Expression
36599	ENSG00000278817	AC007325.4	Gene Expression
36600	ENSG00000277196	AC007325.2	Gene Expression

36601 rows × 3 columns

In [65]:

```
filtered_expr = all_expr.drop(index=blank_list)
filtered_expr.head()
```

Out[65]:

	AAACAAGTATCTCCCA-1	AAACAATCTACTAGCA-1	AAACAGAGCGACTCCT-1	AAACAGCT
3	0	0	0	
6	0	0	0	
8	0	0	0	
9	0	0	0	
11	0	0	0	

5 rows × 3522 columns

In [66]:

```
filtered_names = all_names.drop(index=blank_list)
filtered_names.head()
```

Out[66]:

	feature_id	gene	feature_type
3	ENSG00000238009	AL627309.1	Gene Expression
6	ENSG00000241860	AL627309.5	Gene Expression
8	ENSG00000286448	AP006222.2	Gene Expression
9	ENSG00000236601	AL732372.1	Gene Expression
11	ENSG00000235146	AC114498.1	Gene Expression

4. Summary Statistics and Analysis: ¶

In [68]:

```
expr_stats = filtered_expr.apply(pd.DataFrame.describe, axis=1)
```

In [69]:

```
expr_stats.head(20)
```

Out[69]:

	count	mean	std	min	25%	50%	75%	max
3	3522.0	0.008518	0.091912	0.0	0.0	0.0	0.0	1.0
6	3522.0	0.009086	0.094899	0.0	0.0	0.0	0.0	1.0
8	3522.0	0.002839	0.053217	0.0	0.0	0.0	0.0	1.0
9	3522.0	0.006246	0.082324	0.0	0.0	0.0	0.0	2.0
11	3522.0	0.000568	0.033700	0.0	0.0	0.0	0.0	2.0
14	3522.0	0.049972	0.225602	0.0	0.0	0.0	0.0	2.0
15	3522.0	0.000568	0.023826	0.0	0.0	0.0	0.0	1.0
16	3522.0	0.100227	0.326619	0.0	0.0	0.0	0.0	3.0
17	3522.0	0.016184	0.130624	0.0	0.0	0.0	0.0	2.0
18	3522.0	0.002555	0.050493	0.0	0.0	0.0	0.0	1.0
22	3522.0	0.175468	0.450806	0.0	0.0	0.0	0.0	4.0
23	3522.0	0.180011	0.444562	0.0	0.0	0.0	0.0	4.0
24	3522.0	0.883021	1.075112	0.0	0.0	1.0	1.0	9.0
25	3522.0	0.360875	0.630190	0.0	0.0	0.0	1.0	5.0
26	3522.0	0.074106	0.274681	0.0	0.0	0.0	0.0	2.0
27	3522.0	0.022147	0.150991	0.0	0.0	0.0	0.0	2.0
28	3522.0	0.166951	0.420246	0.0	0.0	0.0	0.0	4.0
29	3522.0	2.651618	2.176632	0.0	1.0	2.0	4.0	14.0
30	3522.0	1.361726	2.262091	0.0	0.0	1.0	2.0	40.0
32	3522.0	1.477002	1.562177	0.0	0.0	1.0	2.0	12.0

In [77]:

```
expr_stats.loc[filtered_names["gene"] == "TFF3"]
```

Out[77]:

	count	mean	std	min	25%	50%	75%	max
34246	3522.0	90.235094	80.550117	0.0	38.0	69.5	118.0	920.0

In [83]:

```
TFF3_expr = np.array(filtered_expr.loc[filtered_names["gene"] == "TFF3"])
```

In [90]:

```
'''# the histogram of the data
n, bins, patches = plt.hist(TFF3_expr, bins=20, density=True, facecolor='g',


plt.xlabel('UMI Counts')
plt.ylabel('Probability')
plt.title('Histogram of TFF3')
plt.text(60, .025, r'$\mu=100,\ \sigma=15$')
plt.xlim(0, 180)
plt.ylim(0, 0.03)
plt.grid(True)
plt.show()'''
```

Out[90]:

```
"# the histogram of the data\nn, bins, patches = plt.hist(TFF3_
expr, bins=20, density=True, facecolor='g', alpha=0.75)\n\n\npl
t.xlabel('UMI Counts')\nplt.ylabel('Probability')\nplt.title('H
istogram of TFF3')\nplt.text(60, .025, r'$\\mu=100,\\ \\sigma=1
5$')\nplt.xlim(0, 180)\nplt.ylim(0, 0.03)\nplt.grid(True)\nplt.
show()"
```

In [ ]: