GeneLab URR analysis notebook

This notebook contains analyses of RNA-seq RSEM unnormalized gene counts, and DESeq2 normalized gene counts generated from Universal Reference RNA.

Table of Content

- Total Expressed Genes (Annotated Genes)
- Comparative analyses from different NovaSeq runs

GeneLab RNA-seq pipeline

Explore the RNA-seq pipeline.

Setting up the notebook

In [1]:

```python
# Import python packages
import os
import pandas as pd
import numpy as np
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

Total Expressed Genes (Annotated Genes)

Unnormalized Counts

In [2]:

```python
os.chdir("URR_Compare_Analysis")
os.listdir()
```

Out[2]:

```
['.DS_Store',
 'LPkit_SampleTable.csv',
 'NS_SampleTable.csv',
 'NumNonZeroGenes.csv',
 'star_alignment.csv',
 'URR_Compare_Analysis.html',
 'RSEM_Unnormalized_Counts.csv',
 'Normalized_Counts.csv',
 '.ipynb_checkpoints']
```

In [3]:

```python
# Get NumNonZeroGenes.csv
totgenes_file = os.listdir()[3]
totgenes_table = pd.read_csv(totgenes_file, index_col=0)
totgenes_table.index.rename("Sample", inplace=True)
pd.set_option("max_columns", None)
totgenes_table.head()
```

Out[3]:

	Number of genes with non-zero counts
Sample	
FS_20190404_HRep1	25217
FS_20190404_HRep2	24548

	Number of genes with non-zero counts
Sample	
FS_20190404_HRep3	24595
FS_20190404_Rep10	28473
FS_20190404_Rep11	25218

In [4]:

```python
# Get RSEM_Unnormalized_Counts.csv
unnorm_file = os.listdir()[6]
unnorm_cutoff = pd.read_csv(unnorm_file).rename(columns={"Unnamed: 0": "Genes"})
unnorm_cutoff.head()
```

Out[4]:

	Genes	FS_20190404_HRep1	FS_20190404_HRep2	FS_20190404_HRep3	FS_20190404_
0	ENSMUSG00000000001	5373.0	4574.0	4647.0	1
1	ENSMUSG00000000003	0.0	0.0	0.0	
2	ENSMUSG00000000028	1755.0	1376.0	1434.0	
3	ENSMUSG00000000031	3181.0	2704.0	2581.0	
4	ENSMUSG00000000037	68.0	72.0	63.0	

In [5]:

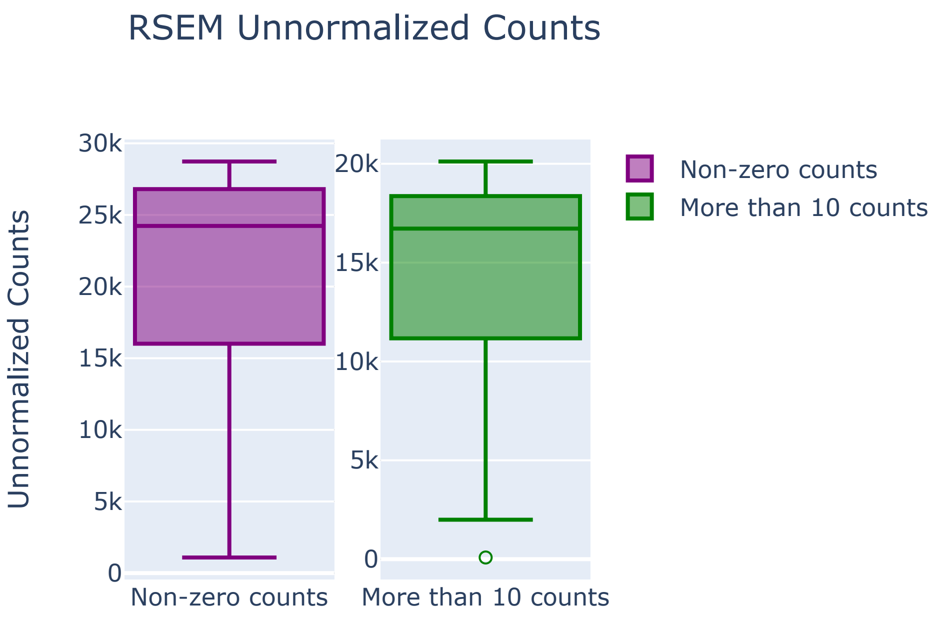
```python
# Unnormalized counts cutoff > 10
unnorm_cutoff = unnorm_cutoff.set_index(keys="Genes")
unnorm_cutoff = unnorm_cutoff[unnorm_cutoff.index.str.contains("ENSMUSG")]
unnorm_cutoff = unnorm_cutoff[unnorm_cutoff > 10]
unnorm_10 = unnorm_cutoff.count().to_frame(name="Number of genes with more than 10 counts'
unnorm_10.index.rename("Sample", inplace=True)
unnorm_10.head()
```

Out[5]:

	Number of genes with more than 10 counts
Sample	
FS_20190404_HRep1	17074
FS_20190404_HRep2	16730
FS_20190404_HRep3	16669
FS_20190404_Rep10	19635
FS_20190404_Rep11	16985

In [6]:

```python
# Box plot of unnormalized counts >0 and >10 for all samples
data = totgenes_table.melt()
data10 = unnorm_10.melt()
fig = go.Figure()
fig = make_subplots(rows=1, cols=2)
fig.add_trace(go.Box(y=data["value"], quartilemethod="inclusive", name="Non-zero counts",
                row=1, col=1)# "exclusive","inclusive", or "linear" by default
fig.add_trace(go.Box(y=data10["value"], quartilemethod="inclusive", name="More than 10 cou
             row=1, col=2)
fig.update_traces(boxpoints="suspectedoutliers", jitter=0.1, textsrc="inside", width=0.5)
fig.update_layout(title_text="RSEM Unnormalized Counts", title_x=0.4, yaxis_title="Unnorma
                 boxmode="overlay", hovermode="x unified", width=500, height=400)
fig.show()
```



Comparative analyses from different NovaSeq runs

Total Expressed Genes (Annotated Genes)

In [7]:

```
# Go to URR_Compare_Analysis
os.listdir()
```

Out[7]:

```
['.DS_Store',
 'LPkit_SampleTable.csv',
 'NS_SampleTable.csv',
 'NumNonZeroGenes.csv',
 'star_alignment.csv',
 'URR_Compare_Analysis.html',
 'RSEM_Unnormalized_Counts.csv',
 'Normalized_Counts.csv',
 '.ipynb_checkpoints']
```

In [8]:

```
# Get NumNonZeroGenes.csv
totgenes_file = os.listdir()[3]
totgenes_table = pd.read_csv(totgenes_file, index_col=0)
totgenes_table.index.rename("Sample", inplace=True)
pd.set_option("max_columns", None)
totgenes_table.head()
```

Out[8]:

	Number of genes with non-zero counts
Sample	
FS_20190404_HRep1	25217
FS_20190404_HRep2	24548
FS_20190404_HRep3	24595
FS_20190404_Rep10	28473
FS_20190404_Rep11	25218

In [9]:

```
# Get LPkit_SampleTable.csv
kit_file = os.listdir()[1]
kit_table = pd.read_csv(kit_file).rename(columns={"Unnamed: 0":"Sample", "condition":"Libr
kit_table = kit_table.sort_values(by="Sample", ascending=True)
kit_table.head()
```

Out[9]:

	Sample	Library kit
0	FS_20190404_HRep1	Illumina_TruSeq_Stranded_totRNA_Gold
1	FS_20190404_HRep2	Illumina_TruSeq_Stranded_totRNA_Gold
2	FS_20190404_HRep3	Illumina_TruSeq_Stranded_totRNA_Gold
3	FS_20190404_Rep1	Illumina_TruSeq_Stranded_totRNA_Gold
4	FS_20190404_Rep10	Illumina_TruSeq_Stranded_totRNA_Gold

In [10]:

```
# Get NS_SampleTable.csv
ns_file = os.listdir()[2]
ns_table = pd.read_csv(ns_file).rename(columns={"Unnamed: 0":"Sample", "condition":"NovaSe
ns_table.head()
```

Out[10]:

	Sample	NovaSeqRun
0	FS_20190404_HRep1	FS
1	FS_20190404_HRep2	FS
2	FS_20190404_HRep3	FS
3	FS_20190404_Rep1	FS
4	FS_20190404_Rep2	FS

In [11]:

```
# Get Uniquely mapped reads from star_alignment.csv
aligned_file = os.listdir()[4]
aligned_table = pd.read_csv(aligned_file).rename(columns={"Category":"Sample"})
depth_table = aligned_table.filter(["Sample", "Uniquely mapped"]).rename(columns={"Uniquel
depth_table["Uniquely mapped reads"] = depth_table["Uniquely mapped reads"]/ 1000000
depth_table = depth_table.rename(columns={"Uniquely mapped reads": "Uniquely mapped reads 
depth_table.head()
```

Out[11]:

	Sample	Uniquely mapped reads (M)
0	FS_20190404_HRep1	40.272953
1	FS_20190404_HRep2	34.034887
2	FS_20190404_HRep3	34.851778
3	FS_20190404_Rep1	110.888770
4	FS_20190404_Rep10	115.690366

In [12]:

```
# Non-zero unnormalized counts of different library kits used
kit_summary = kit_table.merge(totgenes_table, on="Sample")
kit_summary = kit_summary.set_index(keys="Sample")
kit_summary.head()
```

Out[12]:

	Library kit	Number of genes with non-zero counts

	Library kit	Number of genes with non-zero counts
Sample		
FS_20190404_HRep1	Illumina_TruSeq_Stranded_totRNA_Gold	25217
FS_20190404_HRep2	Illumina_TruSeq_Stranded_totRNA_Gold	24548
FS_20190404_HRep3	Illumina_TruSeq_Stranded_totRNA_Gold	24595
FS_20190404_Rep1	Illumina_TruSeq_Stranded_totRNA_Gold	28129
FS_20190404_Rep10	Illumina_TruSeq_Stranded_totRNA_Gold	28473

In [13]:

```
# Non-zero unnormalized counts of different NovaSeq runs
ns_summary = ns_table.merge(kit_summary, on="Sample")
ns_summary
```

Out[13]:

	Sample	NovaSeqRun	Library kit	Number of genes with non-zero counts
0	FS_20190404_HRep1	FS	Illumina_TruSeq_Stranded_totRNA_Gold	25217
1	FS_20190404_HRep2	FS	Illumina_TruSeq_Stranded_totRNA_Gold	24548
2	FS_20190404_HRep3	FS	Illumina_TruSeq_Stranded_totRNA_Gold	24595
3	FS_20190404_Rep1	FS	Illumina_TruSeq_Stranded_totRNA_Gold	28129
4	FS_20190404_Rep2	FS	Illumina_TruSeq_Stranded_totRNA_Gold	26941
...	...	...	...	...
149	RR10_KDN_UPX_20220104_7	RR10_KDN_UPX	QIAseq_UPX_mRNA	14842
150	RR10_KDN_UPX_20220104_8	RR10_KDN_UPX	QIAseq_UPX_mRNA	14774
151	RR23_LVR_LNG_20220112_2_Xp	RR23_LVR_LNG	Illumina_TruSeq_Stranded_totRNA_Gold	28463
152	RR23_LVR_LNG_20220112_3_Xp	RR23_LVR_LNG	Illumina_TruSeq_Stranded_totRNA_Gold	26158
153	PI_HRT_20220112_Xp	PI_HRT	Illumina_TruSeq_Stranded_totRNA_Gold	24734

154 rows × 4 columns

In [14]:

```
# Non-zero unnormalized counts of different NovaSeq runs
depth_summary = ns_summary.merge(depth_table, on="Sample")
depth_summary = depth_summary.filter(["Sample", "NovaSeqRun", "Number of genes with non-ze
depth_summary.head()
```

Out[14]:

	Sample	NovaSeqRun	Number of genes with non-zero counts	Uniquely mapped reads (M)
0	FS_20190404_HRep1	FS	25217	40.272953
1	FS_20190404_HRep2	FS	24548	34.034887
2	FS_20190404_HRep3	FS	24595	34.851778
3	FS_20190404_Rep1	FS	28129	110.888770
4	FS_20190404_Rep2	FS	26941	72.993442

In [15]:

```
# Group by NS run
```

```python
ind_run = dict(list(depth_summary.groupby("NovaSeqRun")))
ind_run
```

Out[15]:

```
{'FS':                Sample NovaSeqRun  Number of genes with non-zero counts   \
 0   FS_20190404_HRep1         FS                                 25217
 1   FS_20190404_HRep2         FS                                 24548
 2   FS_20190404_HRep3         FS                                 24595
 3    FS_20190404_Rep1         FS                                 28129
 4    FS_20190404_Rep2         FS                                 26941
 5    FS_20190404_Rep3         FS                                 27068
 6    FS_20190404_Rep4         FS                                 27455
 7    FS_20190404_Rep5         FS                                 26860
 8    FS_20190404_Rep6         FS                                 27144
 9    FS_20190404_Rep7         FS                                 28065
 10   FS_20190404_Rep8         FS                                 28421
 11   FS_20190404_Rep9         FS                                 27759
 12  FS_20190404_Rep10         FS                                 28473
 13  FS_20190404_Rep11         FS                                 25218
 14  FS_20190404_Rep12         FS                                 25485
 15  FS_20190404_Rep13         FS                                 27567
 16  FS_20190404_Rep14         FS                                 28147
 17  FS_20190404_Rep15         FS                                 28424
 18  FS_20190404_Rep16         FS                                 26099
 19  FS_20190404_Rep17         FS                                 23202
 20  FS_20190404_Rep18         FS                                 26323
 21  FS_20190404_Rep19         FS                                 27128
 22  FS_20190404_Rep20         FS                                 26497

     Uniquely mapped reads (M)
 0                   40.272953
 1                   34.034887
 2                   34.851778
 3                  110.888770
 4                   72.993442
 5                   76.767071
 6                   89.686997
 7                   79.059419
 8                   84.585888
 9                   88.462019
 10                  95.498190
 11                  87.566747
 12                 115.690366
 13                  45.072266
 14                  47.183193
 15                  76.889819
 16                 141.445416
 17                 121.429975
 18                  67.321401
 19                  95.642894
 20                  78.873987
 21                  84.886688
 22                 120.532743  ,
 'PI_FF_20211026':                Sample      NovaSeqRun  Number of genes with non-zero co
unts  \
 138  PI_FF_20211026_1  PI_FF_20211026                                 24574
 139  PI_FF_20211026_2  PI_FF_20211026                                 24160
 140  PI_FF_20211026_3  PI_FF_20211026                                 24188

      Uniquely mapped reads (M)
 138                  37.173529
 139                  31.926943
 140                  33.138743  ,
 'PI_FF_20211124':                Sample      NovaSeqRun  Number of genes with non-zero co
unts  \
 141  PI_FF_20211124_1  PI_FF_20211124                                 21979
```

```
142  PI_FF_20211124_2  PI_FF_20211124                                 22183

     Uniquely mapped reads (M)
141                  23.961512
142                  26.511764  ,
'PI_GS':                   Sample NovaSeqRun  Number of genes with non-zero counts  \
23   PI_GS_20190422_Rep8      PI_GS                                 28545
24   PI_GS_20190422_Rep9      PI_GS                                 27504
25  PI_GS_20190422_Rep10      PI_GS                                 27400

    Uniquely mapped reads (M)
23                 112.353609
24                  89.063295
25                  87.982191  ,
'PI_HRT':                  Sample NovaSeqRun  Number of genes with non-zero counts  \
153  PI_HRT_20220112_Xp     PI_HRT                                 24734

     Uniquely mapped reads (M)
153                  65.852981  ,
'PI_Rad_HPC':                          Sample  NovaSeqRun  \
97  PI_Rad_HPC_20210225_Rep1  PI_Rad_HPC
98  PI_Rad_HPC_20210225_Rep2  PI_Rad_HPC

    Number of genes with non-zero counts  Uniquely mapped reads (M)
97                                 27226                  80.995246
98                                 26550                  71.985778  ,
'RF_LVR_SLS':                          Sample  NovaSeqRun  \
60  RF_LVR_SLS_20200528_Rep3  RF_LVR_SLS
61  RF_LVR_SLS_20200528_Rep4  RF_LVR_SLS
62  RF_LVR_SLS_20200528_Rep5  RF_LVR_SLS
63  RF_LVR_SLS_20200528_Rep6  RF_LVR_SLS
64  RF_LVR_SLS_20200528_Rep7  RF_LVR_SLS
65  RF_LVR_SLS_20200528_Rep8  RF_LVR_SLS
66  RF_LVR_SLS_20200528_Rep9  RF_LVR_SLS

    Number of genes with non-zero counts  Uniquely mapped reads (M)
60                                 27863                 112.960576
61                                 26606                  76.112090
62                                 27355                  73.127886
63                                 27368                  85.572623
64                                 28308                  77.245352
65                                 27865                  70.986349
66                                 27706                  73.385536  ,
'RF_SPL':                   Sample NovaSeqRun  Number of genes with non-zero counts  \
57  RF_SPL_20200213_Rep1     RF_SPL                                 25292
58  RF_SPL_20200213_Rep2     RF_SPL                                 27234
59  RF_SPL_20200213_Rep6     RF_SPL                                 26231

    Uniquely mapped reads (M)
57                  60.263281
58                  60.070304
59                  59.919477  ,
'RR10_KDN_UPX':                          Sample    NovaSeqRun  \
143  RR10_KDN_UPX_20220104_1  RR10_KDN_UPX
144  RR10_KDN_UPX_20220104_2  RR10_KDN_UPX
145  RR10_KDN_UPX_20220104_3  RR10_KDN_UPX
146  RR10_KDN_UPX_20220104_4  RR10_KDN_UPX
147  RR10_KDN_UPX_20220104_5  RR10_KDN_UPX
148  RR10_KDN_UPX_20220104_6  RR10_KDN_UPX
149  RR10_KDN_UPX_20220104_7  RR10_KDN_UPX
150  RR10_KDN_UPX_20220104_8  RR10_KDN_UPX

     Number of genes with non-zero counts  Uniquely mapped reads (M)
143                                 16439                  14.041522
144                                 13554                   6.058263
145                                 16012                  21.257030
```

```
146                                 14724                   7.484766
147                                 14189                  12.430669
148                                 14143                   6.223526
149                                 14842                   9.257538
150                                 14774                   8.399899  ,
'RR23_LVR_LNG':                          Sample    NovaSeqRun  \
151  RR23_LVR_LNG_20220112_2_Xp  RR23_LVR_LNG
152  RR23_LVR_LNG_20220112_3_Xp  RR23_LVR_LNG

     Number of genes with non-zero counts  Uniquely mapped reads (M)
151                                 28463                 171.437695
152                                 26158                  74.171619  ,
'RR6_CLN_LNG':                        Sample   NovaSeqRun  \
38  RR6_CLN_LNG_20190718_Rep1  RR6_CLN_LNG
39  RR6_CLN_LNG_20190718_Rep2  RR6_CLN_LNG
40  RR6_CLN_LNG_20190718_Rep3  RR6_CLN_LNG

    Number of genes with non-zero counts  Uniquely mapped reads (M)
38                                 27484                  82.853300
39                                 27002                  68.550399
40                                 26636                  73.374007  ,
'RR6_LVR_SPL':                         Sample   NovaSeqRun  \
41   RR6_LVR_SPL_20190805_Rep3  RR6_LVR_SPL
42   RR6_LVR_SPL_20190805_Rep4  RR6_LVR_SPL
43   RR6_LVR_SPL_20190805_Rep9  RR6_LVR_SPL
44  RR6_LVR_SPL_20190805_Rep10  RR6_LVR_SPL
45  RR6_LVR_SPL_20190805_Rep13  RR6_LVR_SPL
46  RR6_LVR_SPL_20190805_Rep14  RR6_LVR_SPL

    Number of genes with non-zero counts  Uniquely mapped reads (M)
41                                 28731                  95.641397
42                                 28133                 104.343357
43                                 27875                  93.640154
44                                 27517                  90.053019
45                                 27392                  85.881207
46                                 27002                  86.680351  ,
'RR6_TMS_DSKN':                         Sample    NovaSeqRun  \
35  RR6_TMS_DSKN_20190628_Rep1  RR6_TMS_DSKN
36  RR6_TMS_DSKN_20190628_Rep2  RR6_TMS_DSKN
37  RR6_TMS_DSKN_20190628_Rep3  RR6_TMS_DSKN

    Number of genes with non-zero counts  Uniquely mapped reads (M)
35                                 27178                  87.336601
36                                 26803                  79.650733
37                                 27614                  93.389118  ,
'RR7_KDN_SKN':                         Sample   NovaSeqRun  \
47   RR7_KDN_SKN_20190909_Rep1  RR7_KDN_SKN
48   RR7_KDN_SKN_20190909_Rep2  RR7_KDN_SKN
49   RR7_KDN_SKN_20190909_Rep3  RR7_KDN_SKN
50   RR7_KDN_SKN_20190909_Rep4  RR7_KDN_SKN
51   RR7_KDN_SKN_20190909_Rep5  RR7_KDN_SKN
52   RR7_KDN_SKN_20190909_Rep6  RR7_KDN_SKN
53   RR7_KDN_SKN_20190909_Rep7  RR7_KDN_SKN
54   RR7_KDN_SKN_20190909_Rep8  RR7_KDN_SKN
55   RR7_KDN_SKN_20190909_Rep9  RR7_KDN_SKN
56  RR7_KDN_SKN_20190909_Rep10  RR7_KDN_SKN

    Number of genes with non-zero counts  Uniquely mapped reads (M)
47                                 26669                  58.771134
48                                 26778                  58.922385
49                                 26928                  65.582509
50                                 24600                  61.024954
51                                 25151                  59.019873
52                                 25111                  55.183371
53                                 24950                  61.833689
54                                 24961                  58.939524
```

```
 55                                 25282                  59.381925
 56                                  7785                   3.839760  ,
 'RR9_RR5_MHU2':                         Sample    NovaSeqRun  \
 26  RR9_RR5_MHU2_20190522_Rep1  RR9_RR5_MHU2
 27  RR9_RR5_MHU2_20190522_Rep2  RR9_RR5_MHU2
 28  RR9_RR5_MHU2_20190522_Rep3  RR9_RR5_MHU2
 29  RR9_RR5_MHU2_20190522_Rep4  RR9_RR5_MHU2
 30  RR9_RR5_MHU2_20190522_Rep5  RR9_RR5_MHU2
 31  RR9_RR5_MHU2_20190522_Rep6  RR9_RR5_MHU2
 32  RR9_RR5_MHU2_20190522_Rep7  RR9_RR5_MHU2
 33  RR9_RR5_MHU2_20190522_Rep8  RR9_RR5_MHU2
 34  RR9_RR5_MHU2_20190522_Rep9  RR9_RR5_MHU2

     Number of genes with non-zero counts  Uniquely mapped reads (M)
 26                                 26331                  54.975307
 27                                 25778                  52.599008
 28                                 25680                  60.889496
 29                                 25965                  55.978324
 30                                 26014                  58.339829
 31                                 25100                  55.400545
 32                                 25742                  52.250636
 33                                 27184                  55.831453
 34                                 26164                  49.281373  ,
 'RRRM1_LVR':                           Sample NovaSeqRun  \
 99   RRRM1_LVR_RR6_20210318_Rep1  RRRM1_LVR
 100  RRRM1_LVR_RR6_20210318_Rep2  RRRM1_LVR
 101      RRRM1_LVR_20210318_Rep1  RRRM1_LVR
 102      RRRM1_LVR_20210318_Rep2  RRRM1_LVR
 103      RRRM1_LVR_20210318_Rep3  RRRM1_LVR
 104      RRRM1_LVR_20210318_Rep4  RRRM1_LVR
 105      RRRM1_LVR_20210318_Rep5  RRRM1_LVR
 106      RRRM1_LVR_20210318_Rep6  RRRM1_LVR
 107      RRRM1_LVR_20210318_Rep7  RRRM1_LVR

      Number of genes with non-zero counts  Uniquely mapped reads (M)
 99                                  24254                  47.040706
 100                                 23912                  48.326822
 101                                 24672                  48.409954
 102                                 25135                  38.875188
 103                                 24071                  44.860812
 104                                 25177                  51.146926
 105                                 24190                  43.066764
 106                                 24209                  41.793269
 107                                 25073                  49.907092  ,
 'RiboTest_RiboZeroGold':                                          Sample             Nova
SeqRun  \
 112  RiboTest_RiboZeroGold_20210402_100ng_RNA_1  RiboTest_RiboZeroGold
 113  RiboTest_RiboZeroGold_20210402_100ng_RNA_2  RiboTest_RiboZeroGold
 114  RiboTest_RiboZeroGold_20210402_500ng_RNA_1  RiboTest_RiboZeroGold
 115  RiboTest_RiboZeroGold_20210402_500ng_RNA_2  RiboTest_RiboZeroGold

      Number of genes with non-zero counts  Uniquely mapped reads (M)
 112                                 24214                  49.649754
 113                                 24372                  47.243508
 114                                 23027                  43.804980
 115                                 23320                  48.423480  ,
 'RiboTest_RiboZeroPlus':                                          Sample             Nova
SeqRun  \
 108  RiboTest_RiboZeroPlus_20210402_100ng_RNA_1  RiboTest_RiboZeroPlus
 109  RiboTest_RiboZeroPlus_20210402_100ng_RNA_2  RiboTest_RiboZeroPlus
 110  RiboTest_RiboZeroPlus_20210402_500ng_RNA_1  RiboTest_RiboZeroPlus
 111  RiboTest_RiboZeroPlus_20210402_500ng_RNA_2  RiboTest_RiboZeroPlus

      Number of genes with non-zero counts  Uniquely mapped reads (M)
 108                                 16832                  19.791358
 109                                 18834                  32.755994
```

```
110                                 22855                  57.839715  ,
111                                 22741                  54.113288  ,
'UPX_ALSDA100':                             Sample    NovaSeqRun  \
134  UPX_ALSDA100_20210806_TecRep1  UPX_ALSDA100   
135  UPX_ALSDA100_20210806_TecRep2  UPX_ALSDA100   
136  UPX_ALSDA100_20210806_TecRep3  UPX_ALSDA100   
137  UPX_ALSDA100_20210806_TecRep4  UPX_ALSDA100   

     Number of genes with non-zero counts  Uniquely mapped reads (M)  
134                                 23799                  97.064195  
135                                 21729                  76.338304  
136                                 21411                  71.400526  
137                                 21744                  75.452050  ,
'UPX_test':                                     Sample NovaSeqRun  \
67      UPX_test_10_ng_ERCC_20201020_tRep1   UPX_test   
68      UPX_test_10_ng_ERCC_20201020_tRep2   UPX_test   
69      UPX_test_10_ng_ERCC_20201020_tRep3   UPX_test   
70   UPX_test_10_ng_no_ERCC_20201020_tRep1   UPX_test   
71   UPX_test_10_ng_no_ERCC_20201020_tRep2   UPX_test   
72   UPX_test_10_ng_no_ERCC_20201020_tRep3   UPX_test   
73       UPX_test_1_ng_ERCC_20201020_tRep1   UPX_test   
74       UPX_test_1_ng_ERCC_20201020_tRep2   UPX_test   
75       UPX_test_1_ng_ERCC_20201020_tRep3   UPX_test   
76    UPX_test_1_ng_no_ERCC_20201020_tRep1   UPX_test   
77    UPX_test_1_ng_no_ERCC_20201020_tRep2   UPX_test   
78    UPX_test_1_ng_no_ERCC_20201020_tRep3   UPX_test   
79      UPX_test_20_ng_ERCC_20201020_tRep1   UPX_test   
80      UPX_test_20_ng_ERCC_20201020_tRep2   UPX_test   
81      UPX_test_20_ng_ERCC_20201020_tRep3   UPX_test   
82   UPX_test_20_ng_no_ERCC_20201020_tRep1   UPX_test   
83   UPX_test_20_ng_no_ERCC_20201020_tRep2   UPX_test   
84   UPX_test_20_ng_no_ERCC_20201020_tRep3   UPX_test   
85     UPX_test_500_pg_ERCC_20201020_tRep1   UPX_test   
86     UPX_test_500_pg_ERCC_20201020_tRep2   UPX_test   
87     UPX_test_500_pg_ERCC_20201020_tRep3   UPX_test   
88  UPX_test_500_pg_no_ERCC_20201020_tRep1   UPX_test   
89  UPX_test_500_pg_no_ERCC_20201020_tRep2   UPX_test   
90  UPX_test_500_pg_no_ERCC_20201020_tRep3   UPX_test   
91      UPX_test_50_pg_ERCC_20201020_tRep1   UPX_test   
92      UPX_test_50_pg_ERCC_20201020_tRep2   UPX_test   
93      UPX_test_50_pg_ERCC_20201020_tRep3   UPX_test   
94   UPX_test_50_pg_no_ERCC_20201020_tRep1   UPX_test   
95   UPX_test_50_pg_no_ERCC_20201020_tRep2   UPX_test   
96   UPX_test_50_pg_no_ERCC_20201020_tRep3   UPX_test   

    Number of genes with non-zero counts  Uniquely mapped reads (M)  
67                                 17111                   3.894863  
68                                 17089                   3.943846  
69                                 17100                   3.808051  
70                                 17421                   5.345093  
71                                  1089                   0.025915  
72                                 17377                   5.002224  
73                                 12244                   3.329272  
74                                 12829                   4.375843  
75                                 13711                   5.861074  
76                                 12188                   4.492884  
77                                 12705                   4.673743  
78                                 12356                   4.122470  
79                                 17230                   3.217031  
80                                 18107                   4.148696  
81                                 18361                   4.340296  
82                                 17631                   3.599179  
83                                 18233                   4.156169  
84                                 17363                   3.116021  
85                                 11000                   4.803449  
86                                 10053                   3.897370  
```

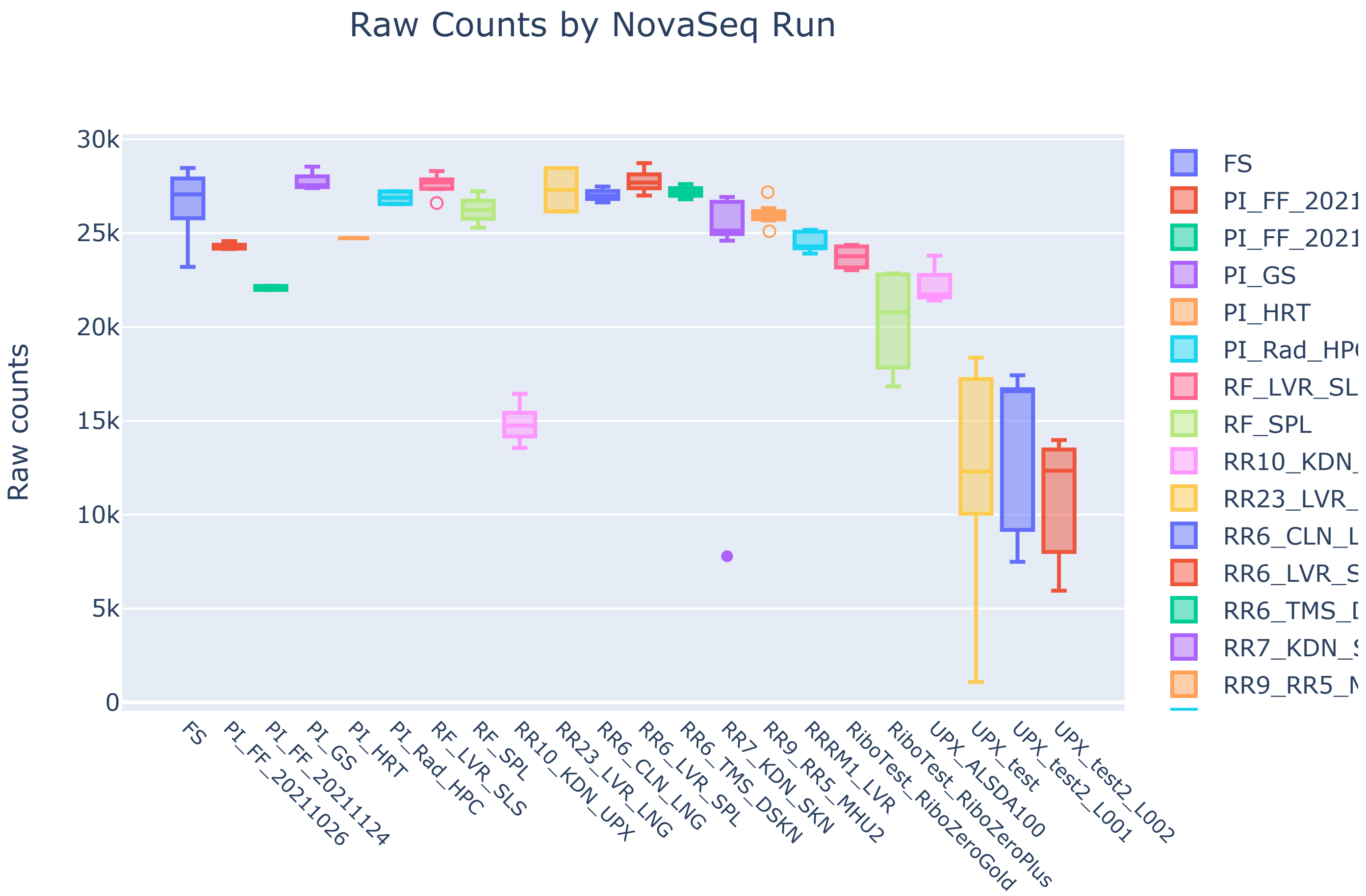
```
87                                  10174                   3.945274
88                                  10097                   3.183167
89                                  11091                   4.639503
90                                  10954                   4.200025
91                                   5256                   3.746737
92                                   5181                   3.883702
93                                   5990                   5.213186
94                                   4664                   4.175198
95                                   4378                   3.799869
96                                   3924                   3.199142  ,
'UPX_test2_L001':                                               Sample      NovaSeqRun  \
116   UPX_test2_L001_S1_Qiagen_008X_20210415_1_10ng  UPX_test2_L001
117   UPX_test2_L001_S1_Qiagen_008X_20210415_2_10ng  UPX_test2_L001
118   UPX_test2_L001_S1_Qiagen_008X_20210415_3_10ng  UPX_test2_L001
119  UPX_test2_L001_S3_Illumina_RZP_20210415_1_10ng  UPX_test2_L001
120  UPX_test2_L001_S3_Illumina_RZP_20210415_2_10ng  UPX_test2_L001
121  UPX_test2_L001_S3_Illumina_RZP_20210415_3_10ng  UPX_test2_L001
122   UPX_test2_L001_S5_No_Ribo_Dep_20210415_1_10ng  UPX_test2_L001
123   UPX_test2_L001_S5_No_Ribo_Dep_20210415_2_10ng  UPX_test2_L001
124   UPX_test2_L001_S5_No_Ribo_Dep_20210415_3_10ng  UPX_test2_L001

     Number of genes with non-zero counts  Uniquely mapped reads (M)
116                                 16682                  14.087898
117                                 16385                  14.548883
118                                 16996                  17.584845
119                                  7489                  13.358559
120                                  9188                  29.790044
121                                  8203                  15.754161
122                                 16654                  15.155427
123                                 17427                  17.572372
124                                 16653                  13.089206  ,
'UPX_test2_L002':                                              Sample      NovaSeqRun  \
125   UPX_test2_L002_S1_Qiagen_05X_20210415_1_10ng  UPX_test2_L002
126   UPX_test2_L002_S1_Qiagen_05X_20210415_2_10ng  UPX_test2_L002
127   UPX_test2_L002_S1_Qiagen_05X_20210415_3_10ng  UPX_test2_L002
128    UPX_test2_L002_S2_Qiagen_1X_20210415_1_10ng  UPX_test2_L002
129    UPX_test2_L002_S2_Qiagen_1X_20210415_2_10ng  UPX_test2_L002
130    UPX_test2_L002_S2_Qiagen_1X_20210415_3_10ng  UPX_test2_L002
131  UPX_test2_L002_S3_No_Ribo_Dep_20210415_1_10ng  UPX_test2_L002
132  UPX_test2_L002_S3_No_Ribo_Dep_20210415_2_10ng  UPX_test2_L002
133  UPX_test2_L002_S3_No_Ribo_Dep_20210415_3_10ng  UPX_test2_L002

     Number of genes with non-zero counts  Uniquely mapped reads (M)
125                                 12342                   6.287190
126                                 13472                   9.021345
127                                 13976                  11.122008
128                                  8013                  15.886517
129                                  6126                   9.574508
130                                  5950                   6.696182
131                                 13410                   4.821391
132                                  8655                   1.303264
133                                 13629                   4.788189  }
```

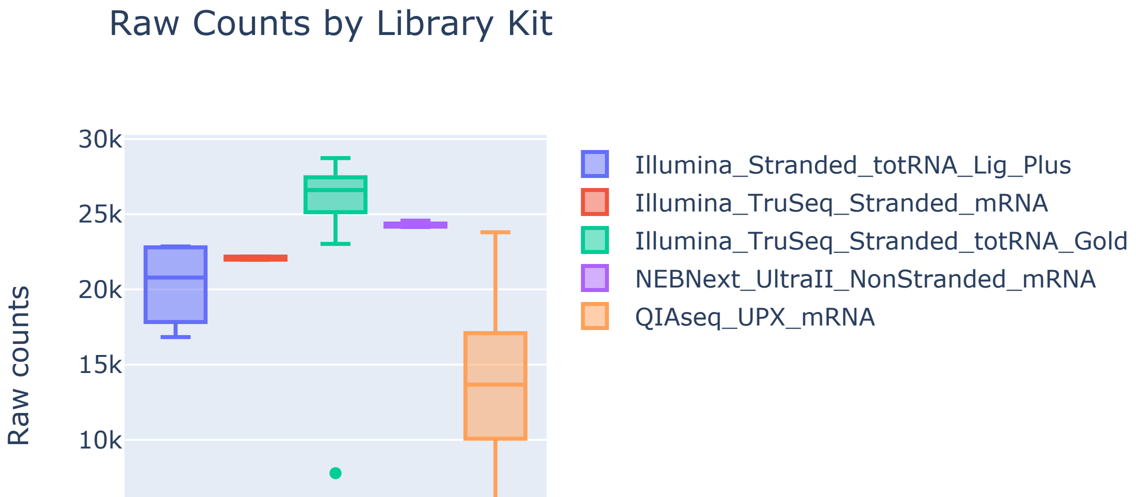
In [16]:

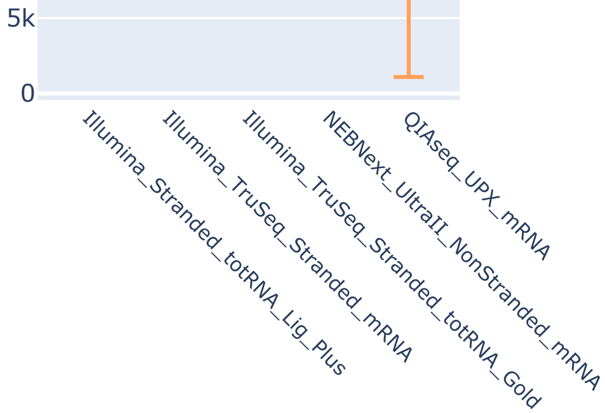
```python
# Box plot grouped by uniquely mapped reads for each NS run on same graph
fig = go.Figure()
fig = make_subplots(rows=1, cols=1)
for i in ind_run:
    fig.add_trace(go.Box(y=ind_run[i]["Number of genes with non-zero counts"], quartilemet
                  name=i), row=1, col=1)
fig.update_traces(boxpoints="suspectedoutliers", jitter=0.1, textsrc="inside", width=0.75)
fig.update_layout(title_text="Raw Counts by NovaSeq Run", title_x=0.4, yaxis_title="Raw co
                 boxmode="group", hovermode="x unified", width=800, height=500)
fig.update_xaxes(tickangle = 45, tickfont = {"size": 10})
fig.show()
```



```
In [17]:  # Group by library kits
          kit_run = dict(list(ns_summary.groupby("Library kit")))
```

```
In [18]:  # Box plot grouped by library preparation for each NS run on same graph
          fig = go.Figure()
          for i in kit_run:
              fig.add_trace(go.Box(y=kit_run[i]["Number of genes with non-zero counts"], quartilemet
                        name=i))
          fig.update_traces(boxpoints="suspectedoutliers", jitter=0.1, textsrc="inside", width=0.75)
          fig.update_layout(title_text="Raw Counts by Library Kit", title_x=0.12, yaxis_title="Raw c
                           boxmode="group", hovermode="x unified", width=600, height=500)
          fig.update_xaxes(tickangle = 45, tickfont = {"size": 10})
          fig.show()
```





In [19]:

```
# Box plot grouped by uniquely mapped reads for samples from all NS runs on same graph
fig = go.Figure()
fig.add_trace(go.Box(x=depth_summary["Uniquely mapped reads (M)"], y=depth_summary["Number
              name="Non-zero counts", marker_color="purple"))
fig.update_traces(boxpoints="suspectedoutliers", jitter=0.3, textsrc="inside", width=1)
fig.update_layout(title_text="Raw Counts by Aligned Uniquely Mapped Reads", title_x=0.5, x
                 boxmode="group", hovermode="x unified", width=600, height=500)
fig.update_xaxes(tickangle = 45, tickfont = {"size": 10})
fig.show()
```

DESeq2 Normalized Data

In [ ]:

```
# Change to R kernel
getwd()
```

In [1]:

```r
# Go to Normalized_Counts.csv and SampleTable.csv directory
work_dir="URR_Compare_Analysis"
setwd(file.path(work_dir))
```

In [2]:

```r
## Install and load ggfortify and ggplot if not already installed
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install("tidyverse")
BiocManager::install("ggfortify")
BiocManager::install("plotly")
library(tidyverse)
library(ggfortify)
library(plotly)
```

```
'getOption("repos")' replaces Bioconductor standard repositories, see
'?repositories' for details

replacement repositories:
    CRAN: https://cran.r-project.org


Bioconductor version 3.14 (BiocManager 1.30.16), R 4.1.2 (2021-11-01)

Warning message:
"package(s) not installed when version(s) same as current; use `force = TRUE` to
  re-install: 'tidyverse'"
Old packages: 'BiocManager', 'blob', 'brew', 'callr', 'class', 'cli', 'clipr',
  'cluster', 'colorspace', 'commonmark', 'cpp11', 'crayon', 'curl',
  'data.table', 'DBI', 'desc', 'devtools', 'digest', 'evaluate', 'fansi',
  'farver', 'foreign', 'formatR', 'GenomeInfoDb', 'gert', 'gh', 'gitcreds',
  'glue', 'gtable', 'htmltools', 'httr', 'IRkernel', 'isoband', 'jsonlite',
  'knitr', 'locfit', 'magrittr', 'markdown', 'MASS', 'Matrix', 'matrixStats',
  'mgcv', 'nlme', 'nnet', 'openssl', 'packrat', 'pbdZMQ', 'pillar', 'pkgload',
  'processx', 'ps', 'purrr', 'RColorBrewer', 'Rcpp', 'RcppArmadillo', 'RCurl',
  'readr', 'rmarkdown', 'roxygen2', 'rpart', 'rprojroot', 'rsconnect',
  'RSQLite', 'rstudioapi', 'rversions', 'S4Vectors', 'spatial', 'stringi',
  'stringr', 'survival', 'sys', 'testthat', 'tibble', 'tidyselect', 'tinytex',
  'usethis', 'uuid', 'viridisLite', 'vroom', 'waldo', 'xfun', 'XML', 'yaml',
  'zip'

'getOption("repos")' replaces Bioconductor standard repositories, see
'?repositories' for details

replacement repositories:
    CRAN: https://cran.r-project.org


Bioconductor version 3.14 (BiocManager 1.30.16), R 4.1.2 (2021-11-01)

Warning message:
"package(s) not installed when version(s) same as current; use `force = TRUE` to
  re-install: 'ggfortify'"
Old packages: 'BiocManager', 'blob', 'brew', 'callr', 'class', 'cli', 'clipr',
  'cluster', 'colorspace', 'commonmark', 'cpp11', 'crayon', 'curl',
  'data.table', 'DBI', 'desc', 'devtools', 'digest', 'evaluate', 'fansi',
  'farver', 'foreign', 'formatR', 'GenomeInfoDb', 'gert', 'gh', 'gitcreds',
  'glue', 'gtable', 'htmltools', 'httr', 'IRkernel', 'isoband', 'jsonlite',
  'knitr', 'locfit', 'magrittr', 'markdown', 'MASS', 'Matrix', 'matrixStats',
  'mgcv', 'nlme', 'nnet', 'openssl', 'packrat', 'pbdZMQ', 'pillar', 'pkgload',
  'processx', 'ps', 'purrr', 'RColorBrewer', 'Rcpp', 'RcppArmadillo', 'RCurl',
  'readr', 'rmarkdown', 'roxygen2', 'rpart', 'rprojroot', 'rsconnect',
  'RSQLite', 'rstudioapi', 'rversions', 'S4Vectors', 'spatial', 'stringi',
  'stringr', 'survival', 'sys', 'testthat', 'tibble', 'tidyselect', 'tinytex',
  'usethis', 'uuid', 'viridisLite', 'vroom', 'waldo', 'xfun', 'XML', 'yaml',
  'zip'

'getOption("repos")' replaces Bioconductor standard repositories, see
'?repositories' for details

replacement repositories:
    CRAN: https://cran.r-project.org


Bioconductor version 3.14 (BiocManager 1.30.16), R 4.1.2 (2021-11-01)

Warning message:
“package(s) not installed when version(s) same as current; use `force = TRUE` to
  re-install: 'plotly'”
Old packages: 'BiocManager', 'blob', 'brew', 'callr', 'class', 'cli', 'clipr',
  'cluster', 'colorspace', 'commonmark', 'cpp11', 'crayon', 'curl',
  'data.table', 'DBI', 'desc', 'devtools', 'digest', 'evaluate', 'fansi',
  'farver', 'foreign', 'formatR', 'GenomeInfoDb', 'gert', 'gh', 'gitcreds',
  'glue', 'gtable', 'htmltools', 'httr', 'IRkernel', 'isoband', 'jsonlite',
  'knitr', 'locfit', 'magrittr', 'markdown', 'MASS', 'Matrix', 'matrixStats',
  'mgcv', 'nlme', 'nnet', 'openssl', 'packrat', 'pbdZMQ', 'pillar', 'pkgload',
  'processx', 'ps', 'purrr', 'RColorBrewer', 'Rcpp', 'RcppArmadillo', 'RCurl',
  'readr', 'rmarkdown', 'roxygen2', 'rpart', 'rprojroot', 'rsconnect',
  'RSQLite', 'rstudioapi', 'rversions', 'S4Vectors', 'spatial', 'stringi',
  'stringr', 'survival', 'sys', 'testthat', 'tibble', 'tidyselect', 'tinytex',
  'usethis', 'uuid', 'viridisLite', 'vroom', 'waldo', 'xfun', 'XML', 'yaml',
  'zip'

── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
✔ ggplot2 3.4.0      ✔ purrr   0.3.4
✔ tibble  3.1.6      ✔ dplyr   1.0.10
✔ tidyr   1.2.1      ✔ stringr 1.4.0
✔ readr   2.1.2      ✔ forcats 0.5.2
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()

Attaching package: ‘plotly’


The following object is masked from ‘package:ggplot2’:

    last_plot


The following object is masked from ‘package:stats’:

    filter


The following object is masked from ‘package:graphics’:

    layout

```

In [3]:

```r
# Import table with samples and respective groups and column numbers
samp_group <- read.csv(Sys.glob("LPkit_SampleTable.csv"), header = TRUE, row.names = 1, st
samp_group1 <- read.csv(Sys.glob("NS_SampleTable.csv"), header = TRUE, row.names = 1, stri
colnames(samp_group)[1] <- "LibraryKit"
colnames(samp_group1)[1] <- "NovaSeqRun"
samp_group2 <- merge(samp_group, samp_group1, by.x=0, by.y=0)
```

In [4]:

```
head(samp_group1)
```

A data.frame: 6 × 1

	NovaSeqRun
	<fct>
FS_20190404_HRep1	FS
FS_20190404_HRep2	FS
FS_20190404_HRep3	FS
FS_20190404_Rep1	FS
FS_20190404_Rep2	FS
FS_20190404_Rep3	FS

In [5]:

```
head(samp_group2)
```

A data.frame: 6 × 3

	Row.names	LibraryKit	NovaSeqRun
	<I<chr>>	<fct>	<fct>
1	FS_20190404_HRep1	Illumina_TruSeq_Stranded_totRNA_Gold	FS
2	FS_20190404_HRep2	Illumina_TruSeq_Stranded_totRNA_Gold	FS
3	FS_20190404_HRep3	Illumina_TruSeq_Stranded_totRNA_Gold	FS
4	FS_20190404_Rep1	Illumina_TruSeq_Stranded_totRNA_Gold	FS
5	FS_20190404_Rep10	Illumina_TruSeq_Stranded_totRNA_Gold	FS
6	FS_20190404_Rep11	Illumina_TruSeq_Stranded_totRNA_Gold	FS

In [6]:

```
# Import normalized counts table
normCounts <- read.csv(Sys.glob("Normalized_Counts.csv"), header = TRUE, row.names = 1, st
normCounts <- normCounts +1
```

In [7]:

```
# PCA plot, all samples grouped by NS run
## Indicate PCA plot size
size_var <- 3
alpha_var <- 0.3
exp_raw <- log2(normCounts)
PCA_raw <- prcomp(t(exp_raw), scale = FALSE)

NS <- autoplot(PCA_raw, data = samp_group2, colour = 'NovaSeqRun', shape = 'NovaSeqRun', s
     theme_bw() + theme(axis.text.x = element_text(size=12), axis.text.y = element_text(si
     theme(legend.title = element_text(size=8), legend.text = element_text(size=6)) +
     guides(alpha = guide_legend(order = 1), size = guide_legend(order = 2)) +
     ggtitle("Grouped by NovaSeq Run") + theme(plot.title = element_text(hjust = 0.5)) +
     scale_shape_manual(values = rep(17:22, len = 22))
ggplotly(NS, tooltip = c("text", "size"), width = 700, height = 750)
```

In [8]:

```
# PCA plot, all samples grouped by LP kit
## Indicate PCA plot size
size_var <- 3
alpha_var <- 0.3
exp_raw <- log2(normCounts)
PCA_raw <- prcomp(t(exp_raw), scale = FALSE)

kit <- autoplot(PCA_raw, data = samp_group2, colour = 'LibraryKit', shape = 'LibraryKit',
      theme_bw() + theme(axis.text.x = element_text(size=12), axis.text.y = element_text(s
      theme(legend.title = element_text(size=8), legend.text = element_text(size=8)) +
      guides(alpha = guide_legend(order = 1), size = guide_legend(order = 2)) +
      ggtitle("Grouped by Library Kit") + theme(plot.title = element_text(hjust = 0.5)) +
      scale_shape_manual(values = rep(17:22, len = 22))
ggplotly(kit, tooltip = c("text", "size"), width = 700, height = 400)
```