TNBC_Phenotype_Distribution

2022-10-12

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(RColorBrewer)
```

CSV

Read in the TNBC revised csv file.

```
TNBC <- readr::read_csv("/Users/henzhwang/Desktop/TNBC_training/MIBI-TNBC_scdata_counts_mm_matlab_revis
```

```
## Rows: 179194 Columns: 64
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): SITE_02, RECURRENCE_LABEL, mm
## dbl (61): sample_id, patient_id, AGE_AT_DX, STAGE, LATERAL, GRADE, Survival_...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Some statistics of the dataset

```
# Column names in the dataset
names(TNBC)
```

```
##  [1] "sample_id"             "patient_id"            "AGE_AT_DX"
##  [4] "STAGE"                 "SITE_02"               "LATERAL"
##  [7] "GRADE"                 "RECURRENCE_LABEL"      "Survival_days_capped"
## [10] "cluster_id"            "mm"                    "cell_type"
## [13] "ImageNb"              "cellLabelInImage"     "cellSize"
## [16] "cellRadius"           "centroidX"            "centroidY"
## [19] "majoraxis"            "eccentricity"         "Au"
## [22] "Background"           "betaCatenin"          "Ca"
## [25] "CD11b"                "CD11c"                "CD138"
## [28] "CD16"                 "CD20"                 "CD209"
## [31] "CD3"                  "CD31"                 "CD4"
## [34] "CD45"                 "CD45RO"               "CD56"
## [37] "CD63"                 "CD68"                 "CD8"
## [40] "dsDNA"                "EGFR"                 "Fe"
## [43] "FoxP3"                "H3K27me3"             "H3K9ac"
## [46] "HLA_Class_1"          "HLADR"                "IDO"
## [49] "Keratin17"            "Keratin6"             "Ki67"
## [52] "Lag3"                 "MPO"                  "Na"
## [55] "P"                    "p53"                  "panKeratin"
## [58] "PD1"                  "PDL1"                 "pS6"
## [61] "Si"                   "SMA"                  "Ta"
## [64] "Vimentin"
```

```
print("------------------------------------------------------------------------")
```

```
## [1] "------------------------------------------------------------------------"
```

```
# Chekc if patient_id and sample_id have the same number
count_patient <- n_distinct(TNBC$patient_id)
count_sample <- n_distinct(TNBC$sample_id)
dplyr::setequal(count_sample, count_patient)
```

```
## [1] TRUE
```

```
print(paste("There are equal number of", count_patient, "patients and samples in the dataset."))
```

```
## [1] "There are equal number of 39 patients and samples in the dataset."
```

```
print("------------------------------------------------------------------------")
```

```
## [1] "------------------------------------------------------------------------"
```

```
# Checking whether there are duplicates in cellLabelInImage column
dplyr::select(TNBC, cellLabelInImage) %>%
  duplicated() %>% sum()
```

```
## [1] 169164
```

```
print("------------------------------------------------------------------------")
```

```
## [1] "------------------------------------------------------------------------"
```

```
# Number of unique cell types and their names
## There are total of 16 different cells in the dataset
type_counts <- TNBC %>%
  group_by(mm) %>%
  #summarise(count = n_distinct(cellLabelInImage))
  summarise(n = n(), NumOfSamples = n_distinct(sample_id))
type_counts
```

```
## # A tibble: 16 x 3
##    mm               n NumOfSamples
##    <chr>        <int>        <int>
##  1 B            17084          31
##  2 CD3 T         1135          22
##  3 CD4 T         9918          36
##  4 CD8 T        13376          36
##  5 DC            2381          34
##  6 DC/Mono       1280          28
##  7 Endothelial    279          25
##  8 Epithelial   31871          39
##  9 Mac           5552          38
## 10 Mesenchymal  27698          39
## 11 Mono/Neu       835          36
## 12 Neu           1365          36
## 13 NK             285          26
## 14 Other        56603          39
## 15 Other immune  8669          36
## 16 T reg          863          22
```

Extract needed columns

Extract columns that are useful for our analysis.

```
TNBC <- TNBC %>%
  dplyr::select(c(sample_id, patient_id, Survival_days_capped,
                  cluster_id, mm, cell_type, ImageNb, cellLabelInImage,
                  cellSize, cellRadius, centroidX, centroidY))
head(TNBC)
```

```
## # A tibble: 6 x 12
##   sample~1 patie~2 Survi~3 clust~4 mm    cell_~5 ImageNb cellL~6 cellS~7 cellR~8
##      <dbl>   <dbl>   <dbl>   <dbl> <chr>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1        1   30824    2612      34 B           5       1      10     211    7.82
## 2        1   30824    2612      11 B           5       1      17     184    7.20
## 3        1   30824    2612      31 B           5       1      18     277    8.94
## 4        1   30824    2612      33 B           5       1      47     564   13.0 
## 5        1   30824    2612      31 B           5       1      49     402   11.1 
## 6        1   30824    2612      31 B           5       1      65     705   14.7 
## # ... with 2 more variables: centroidX <dbl>, centroidY <dbl>, and abbreviated
## #   variable names 1: sample_id, 2: patient_id, 3: Survival_days_capped,
## #   4: cluster_id, 5: cell_type, 6: cellLabelInImage, 7: cellSize,
## #   8: cellRadius
```

Range of the spatial corrdinate x and y

First we want to find the range of the spatial corrdinate x and y in the whole dataset.

```
centroidX <- TNBC$centroidX
centroidY <- TNBC$centroidY

# Range for the whole dataset
range_wholeX <- c(min(centroidX), max(centroidX), median(centroidX), sd(centroidX))
range_wholeY <- c(min(centroidY), max(centroidY), median(centroidY), sd(centroidY))

print(paste("The range of centroid X in the whole dataset is (", range_wholeX[1], ",", range_wholeX[2],
```

```
## [1] "The range of centroid X in the whole dataset is ( 4.511905 , 2043.474 ), the median is 1054.774
```

```r
print(paste("The range of centroid X in the whole dataset is (", range_wholeY[1], ",", range_wholeY[2],
```

```
## [1] "The range of centroid X in the whole dataset is ( 3.514286 , 2044.483 ), the median is 1055.116
```

Now we want to find the range of the spatial corrdinate for each cell types in the dataset.

```r
# Find the range of each cell types in the dataset
## centroidX
cells_corrdX <- TNBC %>%
  group_by(mm) %>%
  summarise(n = n(), min = min(centroidX), max = max(centroidX), median = median(centroidX), sd = sd(ce
cells_corrdX
```

```
## # A tibble: 16 x 6
##    mm                n   min   max median    sd
##    <chr>         <int> <dbl> <dbl>  <dbl> <dbl>
##  1 B             17084  4.58 2043.  1069.  541.
##  2 CD3 T          1135  6.78 2043.  1425.  501.
##  3 CD4 T          9918  4.61 2043.  1324.  560.
##  4 CD8 T         13376  4.69 2043.  1110.  570.
##  5 DC             2381  4.57 2043.   949.  646.
##  6 DC/Mono        1280  9.00 2043.  1309.  499.
##  7 Endothelial     279 28.7  2023.  1310.  536.
##  8 Epithelial    31871  4.57 2043.  1118.  566.
##  9 Mac            5552  4.69 2043.  1194.  559.
## 10 Mesenchymal   27698  4.56 2043.  1029.  576.
## 11 Mono/Neu        835 10.0  2043.  1266.  547.
## 12 Neu            1365  8.70 2043.  1154.  569.
## 13 NK              285 15.8  2029.  1343.  523.
## 14 Other         56603  4.51 2043.   914.  588.
## 15 Other immune   8669  4.64 2043.  1056.  572.
## 16 T reg           863  9.92 2040.  1233.  512.
```

```r
## centroidY
cells_corrdY <- TNBC %>%
  group_by(mm) %>%
  summarise(n = n(), min = min(centroidY), max = max(centroidY), median = median(centroidY), sd = sd(ce
cells_corrdY
```

```
## # A tibble: 16 x 6
##    mm                n   min   max median    sd
##    <chr>         <int> <dbl> <dbl>  <dbl> <dbl>
##  1 B             17084  4.6  2044.  1144.  577.
##  2 CD3 T          1135 14.9  2044.  1070.  480.
##  3 CD4 T          9918  3.86 2044.  1107.  553.
##  4 CD8 T         13376  4.04 2044.  1096.  575.
##  5 DC             2381  5.20 2043.  1122.  619.
##  6 DC/Mono        1280  3.65 2043.  1004.  535.
##  7 Endothelial     279 20.4  2011.   902.  494.
##  8 Epithelial    31871  4.49 2044.   949.  558.
##  9 Mac            5552  5.91 2043.   977.  569.
## 10 Mesenchymal   27698  3.78 2044.   979.  580.
## 11 Mono/Neu        835 11.0  2044.  1257.  532.
## 12 Neu            1365  7.00 2043.  1214.  538.
## 13 NK              285 18.1  2042.  1041.  540.
## 14 Other         56603  3.51 2044.  1121.  595.
## 15 Other immune  8669  4.25 2044.  1041.  588.
## 16 T reg          863 10.0  2038.   973.  526.
```

Next we want to find the range of spatial coordinate for each cell types in each sample.

```
# Find range of corrdinate for each cell types in each sample
## centroidX
cells_corrdX_perSample <- TNBC %>%
  group_by(sample_id, mm) %>%
  summarise(n = n(), min = min(centroidX), max = max(centroidX), median = median(centroidX), sd = sd(ce
cells_corrdX_perSample
```

```
## # A tibble: 523 x 7
## # Groups:   sample_id [39]
##    sample_id mm              n     min   max median    sd
##        <dbl> <chr>       <int>   <dbl> <dbl>  <dbl> <dbl>
##  1         1 B             734   34.6  2022.  1062.  321.
##  2         1 CD4 T         152  100.   2043.  1566.  516.
##  3         1 CD8 T         147    9.66 2041.  1147.  615.
##  4         1 DC              1  238.    238.   238.   NA
##  5         1 Epithelial     20   44.7  2042.  1742.  584.
##  6         1 Mac            10  347.   2003.  1520.  528.
##  7         1 Mesenchymal   289   36.4  2042.  1074.  518.
##  8         1 Mono/Neu        2 1540.   1938.  1739.  281.
##  9         1 Neu             2 1159.   1572.  1365.  292.
## 10         1 NK              4  610.   1141.   644.  255.
## # ... with 513 more rows
```

```
## centroidY
cells_corrdY_perSample <- TNBC %>%
  group_by(sample_id, mm) %>%
  summarise(n = n(), min = min(centroidY), max = max(centroidY), median = median(centroidY), sd = sd(ce
cells_corrdY_perSample
```

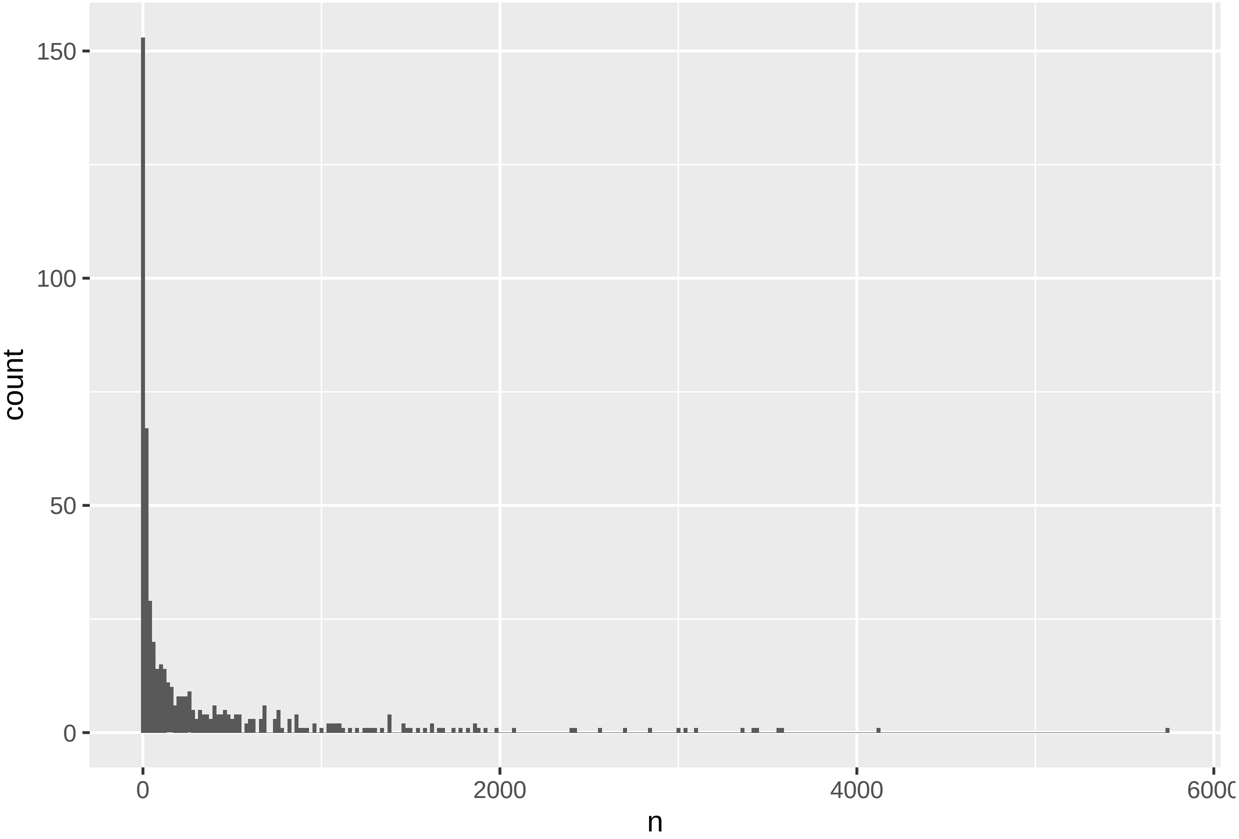
```
## # A tibble: 523 x 7
## # Groups:   sample_id [39]
##    sample_id mm              n    min   max median    sd
##        <dbl> <chr>       <int>  <dbl> <dbl>  <dbl> <dbl>
##  1         1 B             734   6.73 2043.  1793.  525.
##  2         1 CD4 T         152  12.7  2000.  1005.  635.
##  3         1 CD8 T         147  12.1  1998.  1074.  624.
##  4         1 DC              1 412.    412.   412.   NA
##  5         1 Epithelial     20   6.44  818.   240.  284.
##  6         1 Mac            10 418.   1767.  1100.  461.
##  7         1 Mesenchymal   289   8.05 2038.   596.  538.
##  8         1 Mono/Neu        2 170.   1415.   793.  880.
##  9         1 Neu             2 988.   1686.  1337.  494.
## 10         1 NK              4 301.   1086.   872.  349.
## # ... with 513 more rows
```

Phenotype distribution for each sample

Want to make a heatmap of the phenotype distribution where sample number vs. cell types. We first want to plot a distribution plot of the number of cells in the dataset.

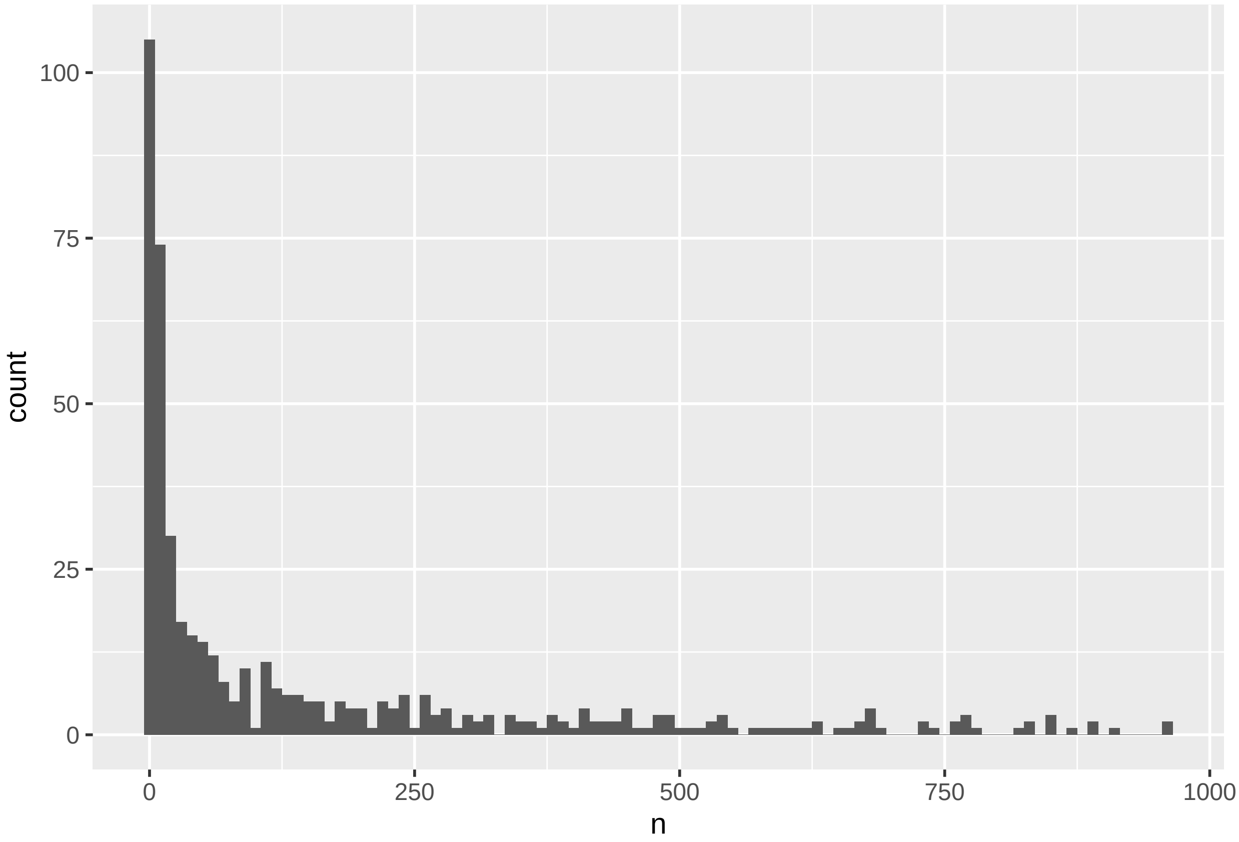
```
# hist 1
cell_counts <- cells_corrdX_perSample
```

```
ggplot(cell_counts, aes(x = n)) +
  geom_histogram(binwidth = 20)
```



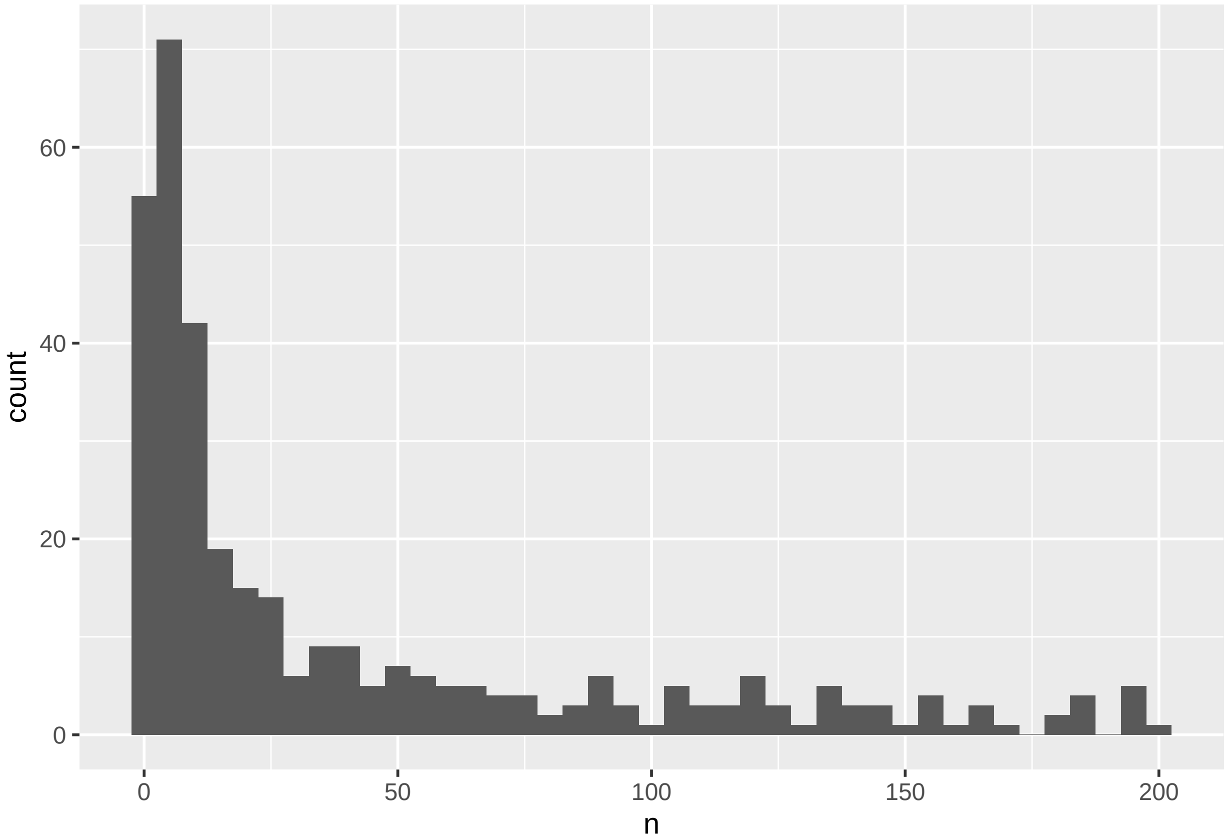
```
## We notice that majority of the distribution is less than 1000 count, we then want to draw a new hist

# hist 2
cell_counts$n[cell_counts$n >= 1000] <- NA

ggplot(cell_counts, aes(x = n)) +
  geom_histogram(binwidth = 10)
```

```
## Warning: Removed 55 rows containing non-finite values (stat_bin).
```



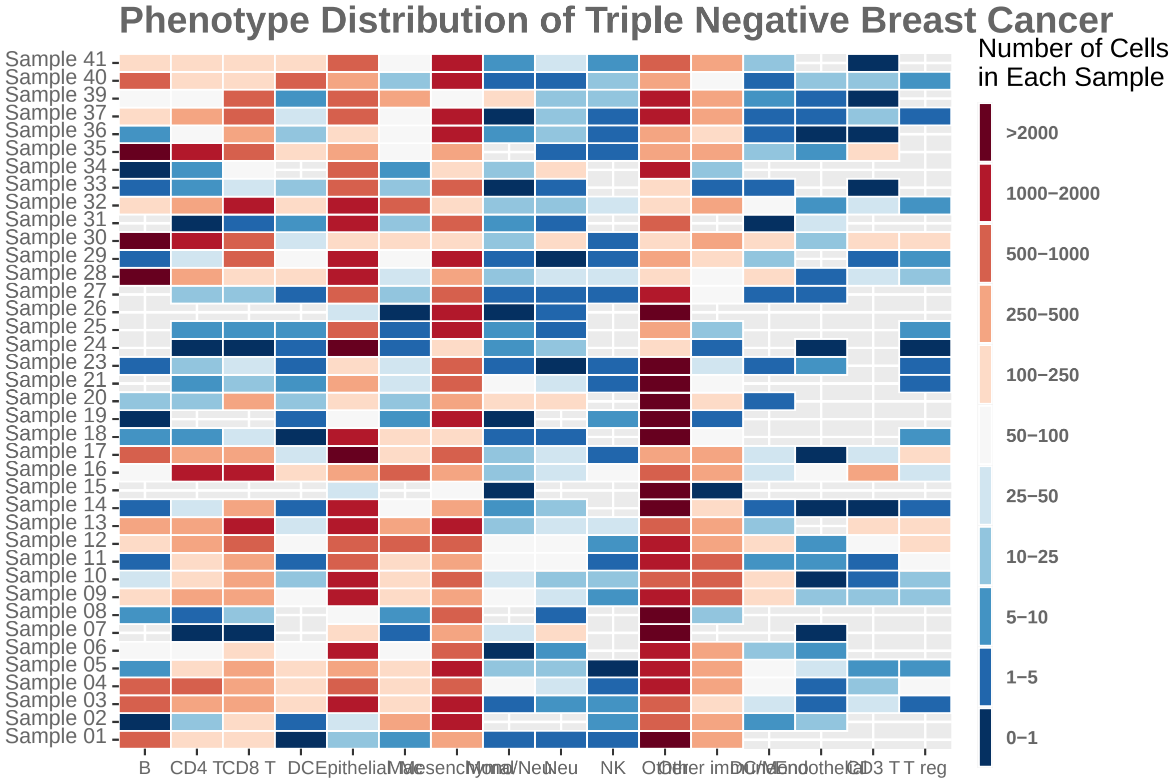
```
## We notice that again the majority of the distribution is less than 200, we want to take a closer loo

# hist 3
cell_counts$n[cell_counts$n >= 200] <- NA

ggplot(cell_counts, aes(x = n)) +
  geom_histogram(binwidth = 5)
```

```
## Warning: Removed 178 rows containing non-finite values (stat_bin).
```



(Why the sample_id is outlier for B cell?) (How well the sequencing for the sample_id?)

```
# Heatmap
#sample_number <- paste("Sample", sort(unique(TNBC$sample_id)))
#sample_number <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4")

TNBC$sample_id[TNBC$sample_id < 10] <- paste0("0", TNBC$sample_id[TNBC$sample_id < 10])

TNBC %>%
  group_by(sample_id, mm) %>%
  mutate(n = n()) %>%
  mutate(sample_id = paste("Sample", sample_id)) %>%
  mutate(sample_id = factor(sample_id, levels = rev(sort(unique(sample_id))))) %>%
  select(c(sample_id, mm, n)) %>%
  #as.data.frame() %>%

  mutate(mm = factor(mm, levels = rev(sort(unique(mm))))) %>%
  mutate(count = cut(n, breaks = c(-1, 1.1, 5.1, 10.1,
                                   25.1, 50.1, 100.1, 250.1, 500.1, 1000.1, 2000.1, max(n, na.rm = TRUE
                     labels = c("0-1", "1-5", "5-10", "10-25", "25-50",
                                "50-100", "100-250", "250-500", "500-1000", "1000-2000", ">2000"))) %>%
  mutate(count = factor(as.character(count), levels = rev(levels(count)))) %>%

  ggplot(mapping = aes(x = mm, y = sample_id, fill = count)) +
    geom_tile(colour = "white", size = 0.3) +
    guides(fill=guide_legend(title = "Number of Cells \nin Each Sample"))+
    labs(x="", y="", title = "Phenotype Distribution of Triple Negative Breast Cancer")+
    scale_y_discrete(expand = c(0, 0)) +
    scale_x_discrete(expand = c(0, 0)) +

    #scale_fill_manual(values = rev(brewer.pal(11, "RdGy")), na.value = "azure4") + #RdGy RdBu
    scale_fill_manual(values = brewer.pal(11, "RdBu"), na.value = "azure4") +
    #scale_fill_manual(values=c("#d53e4f", "#f46d43", "#fdae61", "#fee08b",
    #                           "#e6f598", "#abdda4", "#ddf1da"), na.value = "grey90") +

    #theme_grey(base_size=6)+
  # #theme options
  #   theme(
  #   #bold font for legend text
  #   legend.text=element_text(face="bold"),
  #   #set thickness of axis ticks
  #   axis.ticks=element_line(size=0.4),
  #   #remove plot background
  #   plot.background=element_blank(),
  #   #remove plot border
  #   panel.border=element_blank()
  #   )
    theme_grey(base_size = 10)+
    theme(legend.position = "right", legend.direction = "vertical",
        legend.title = element_text(colour = "black"),
        legend.margin = margin(grid::unit(0, "cm")),
        legend.text = element_text(colour = "grey40", size = 7, face = "bold"),
        legend.key.height = grid::unit(0.8, "cm"),
        legend.key.width = grid::unit(0.2, "cm"),
        axis.text.x = element_text(size = 7, colour = "grey40"),
        axis.text.y = element_text(vjust = 0.2, colour = "grey40"),
        axis.ticks = element_line(size = 0.4),
        plot.background = element_blank(),
        panel.border = element_blank(),
        plot.margin = margin(0.7, 0.4, 0.1, 0.2, "cm"),
        plot.title = element_text(colour = "grey40", hjust = 0, size = 14, face = "bold")
     )
```



Reference: https://www.royfrancis.com/a-guide-to-elegant-tiled-heatmaps-in-r-2019/

Next:

To explore whether we can colour the cell in the tiff file based on cell type. Chap 11 Moedern Stats Modern Bio