Bladder Cancer RNA-Seq Analysis

Zhengpei Cheng

2023-04-02

Introduction

Introduction to biological question

Bladder cancer is a malignant tumor derived from transitional epithelium, and its incidence rate ranks ninth among all cancers. With the development of sequencing technology, people can study the expression of genes in tumor tissues more comprehensively, which provides new ideas for the diagnosis and treatment of bladder cancer. In this project, we downloaded RNA-Seq gene expression data from the TCGA-BLCA dataset to analyze the differentially expressed genes between normal and tumor samples of bladder cancer

Description of the experimental design

The experimental design includes downloading RNA-Seq gene expression data from the TCGA-BLCA dataset, preprocessing the data to filter out low-expressed genes and remove duplicates. Then, we divided the samples into two groups, normal and tumor samples, and conducted differential gene expression analysis on the two groups.

Description of the data preprocessing steps.

We downloaded the RNA-Seq gene expression data of the TCGA-BLCA dataset using the TCGAbiolinks package. After downloading, we obtained a SummarizedExperiment object. We extracted mRNA data from the object by filtering the gene type. For mRNA data, we extracted the gene expression counts matrix. We added the gene symbol to the counts matrix to obtain the mRNA data frame. Next, we removed duplicates and filtered out low-expressed genes. Finally, we read in the mRNA data frame for subsequent analysis.

```
query <- GDCquery(project = "TCGA-BLCA",
                  data.category = "Transcriptome Profiling",
                  data.type = 'Gene Expression Quantification',
                  experimental.strategy = "RNA-Seq",
                  workflow.type = "STAR - Counts")
```

```
GDCdownload(query)
testdata <- GDCprepare(query = query)  # SummarizedExperiment object
```

```
## |                                                  |   0%                      |
```

```
test_mRNA <- testdata[rowData(testdata)$gene_type == "protein_coding",] # extract mRNA
```

Exploratory data analysis.

Quality control

First, used addPerCellQC() to filter low quality cells and filter low quality genes that mean of reads < 10

```
test_mRNA
```

```
## class: RangedSummarizedExperiment
## dim: 16435 427
## metadata(1): data_release
## assays(6): counts stranded_first ... fpkm_unstrand fpkm_uq_unstrand
## rownames(16435): ENSG00000000003.15 ENSG00000000419.13 ...
##   ENSG00000288658.1 ENSG00000288675.1
## rowData names(10): source type ... hgnc_id havana_gene
## colnames(427): TCGA-FJ-A871-01A-11R-A352-07
##   TCGA-DK-AA6L-01A-11R-A39I-07 ... TCGA-GC-A3OO-01A-11R-A22U-07
##   TCGA-XF-A8HC-01A-11R-A36F-07
## colData names(240): barcode patient ... sum detected
```

```
table(qcfilt$discard)
```

```
##
## FALSE  TRUE
##   427     4
```

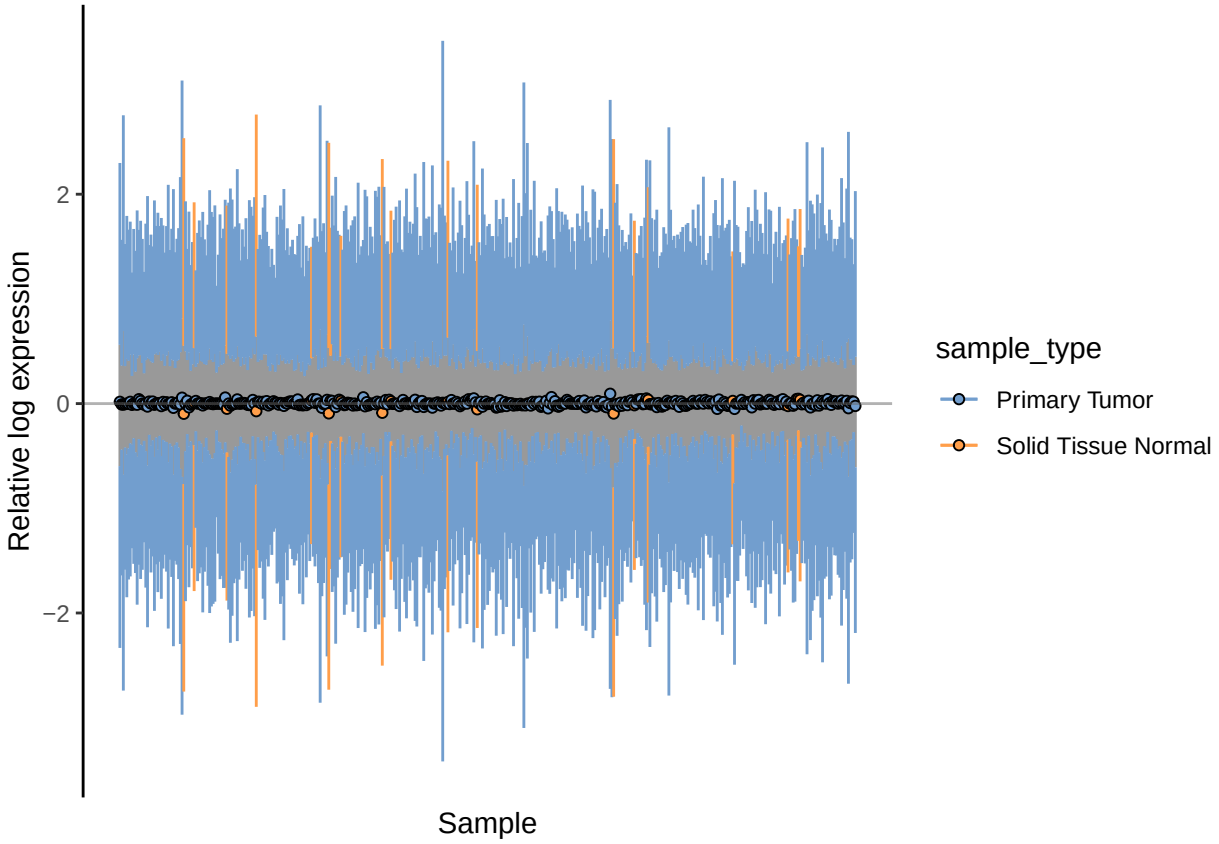
```
table(filter)
```

```
## filter
## FALSE  TRUE
##  3527 16435
```

Normalization

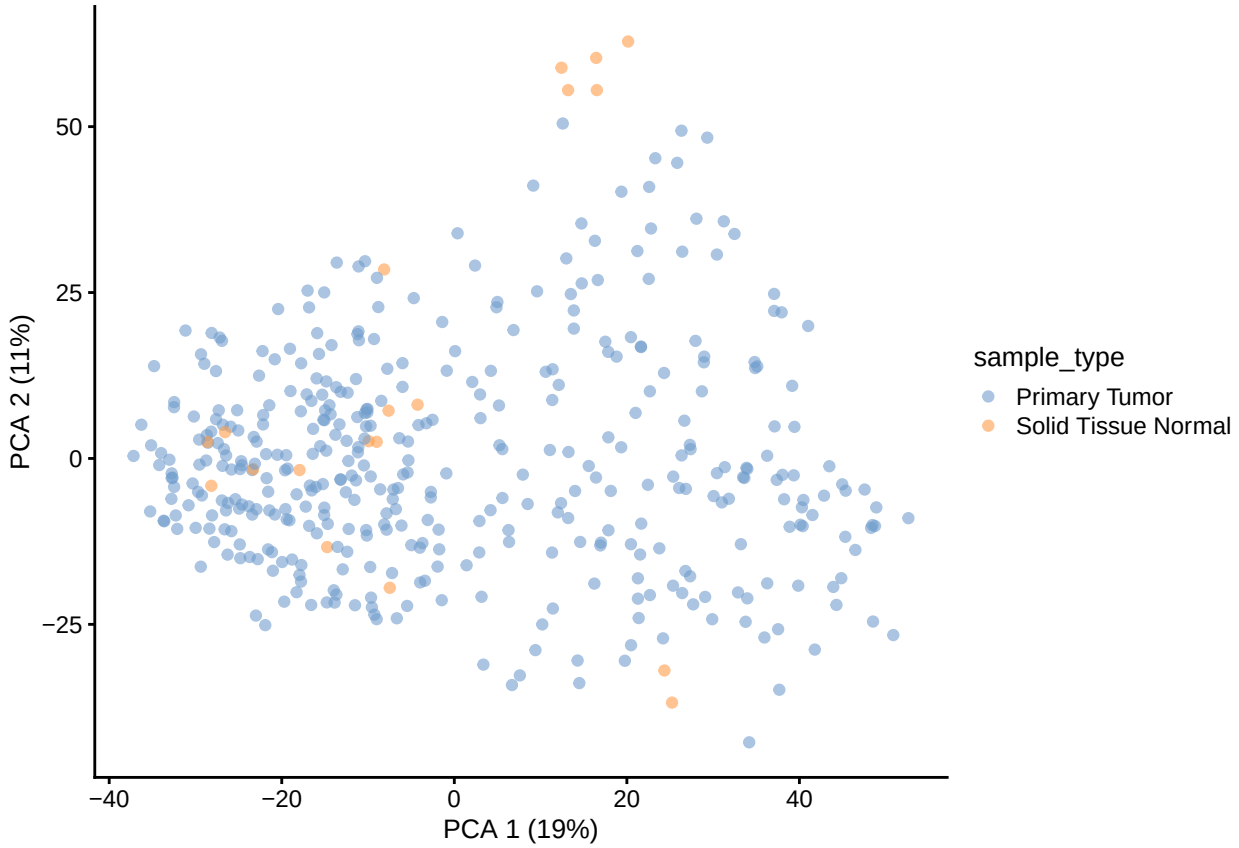
Normalization by TMM and draw RLE plot to measure the quality of normalization. It shows that TMM normalization has a good performance.

```
dge<-test_mRNA
tmm_factors <- calcNormFactors(assay(dge,"counts"), method = "TMM")
scales <- colSums(assay(dge,"counts")) * tmm_factors
tmm <- t(t(assay(dge,"counts"))/scales * mean(scales))
assay(dge, "logcounts") <- log1p(tmm)
scater::plotRLE(as(dge,"SingleCellExperiment"),colour_by = "sample_type",exprs_values = "logcounts")
```

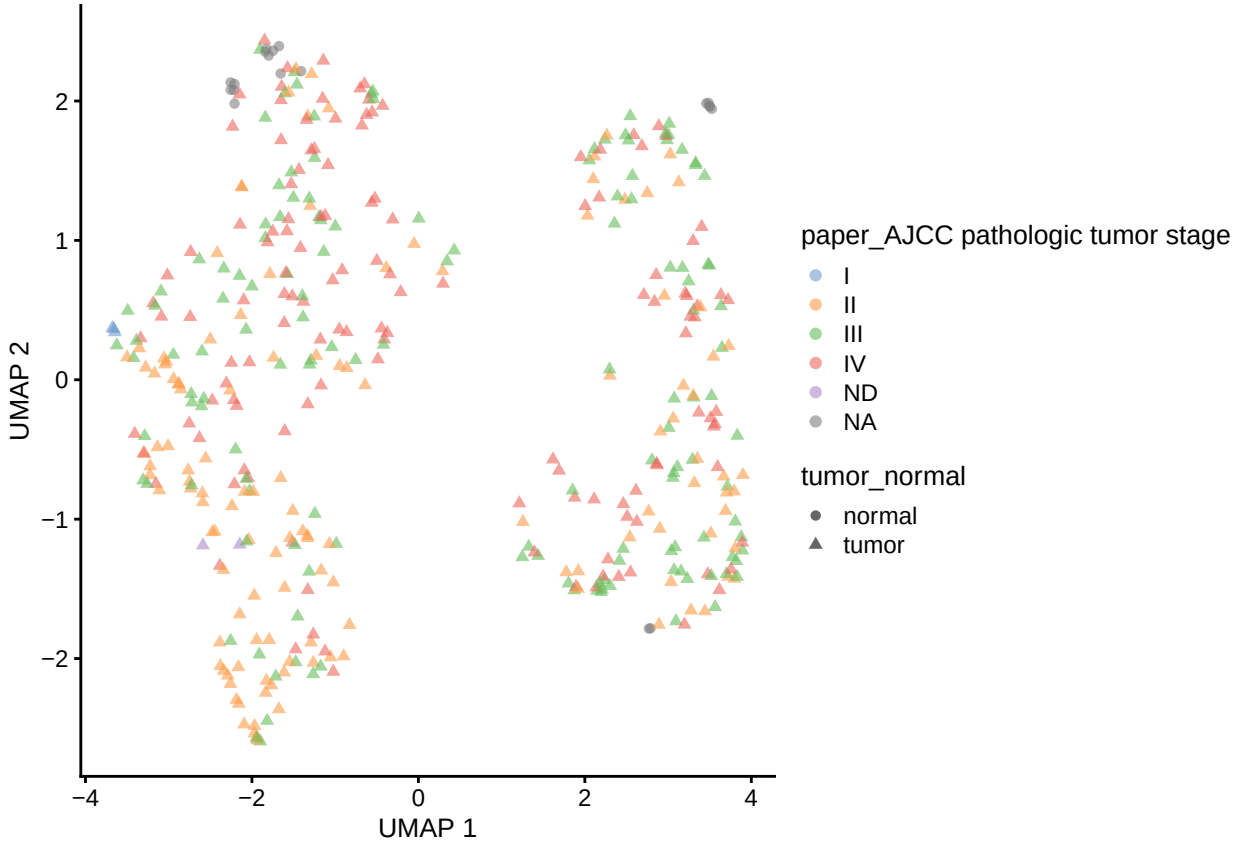


Principal component analysis (PCA)

Do some exploratory data analysis by principle component analysis and also draw some pictures of PCA, and UMAP. It seems that those samples are divided into 2 clusters. However, there are more than 200 dimensions in this data set, I just chose if "sample_type" and "paper_AJCC pathologic tumor stage" are associated with this result. It seems that those 2 dimensions are not related with 2 clus-

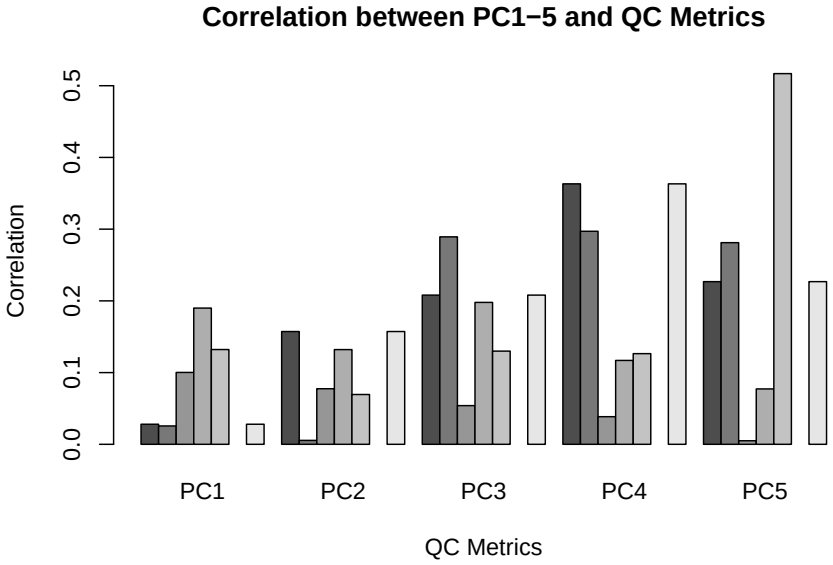

PCA 2 (11%)
50
25
0
−25
−40
−20
0
20
40
PCA 1 (19%)
sample_type
Primary Tumor
Solid Tissue Normal

ters.


UMAP 2
2
1
0
−1
−2
−4
−2
0
2
4
UMAP 1
paper_AJCC pathologic tumor stage
I
II
III
IV
ND
NA
tumor_normal
normal
tumor

Correlation of principal components with quality control indicators

Besides, to find the correlation between principle components 1-5(PC1-5) and interested quality control metrics, I chose "sample_type", "detected", "sum", "race", "gender", "pack_years_smoked". Finally, it seems that they do not have high correlation with PC1-5.



Because I want to extract data from summarizedexperiment object, I extract expression matrix, removed duplicates and filtered out low-expressed genes again.

Devided samples into 'tumor' group and 'normal' group. If the 14th and 15th string of rt's colnames is "01", that means tumor sample. If it is "11", that means normal sample

```
dim(rt)
```

```
## [1] 16434   427
```

```
head(rt[,1:2])
```

```
##          TCGA-FJ-A871-01A-11R-A352-07 TCGA-DK-AA6L-01A-11R-A39I-07
## TSPAN6                           2291                          4390
## DPM1                             1711                          2073
## SCYL3                             328                           675
## C1orf112                          285                           397
## FGR                                43                           717
## CFH                               355                           169
```

```
group <- factor(ifelse(as.integer(substr(colnames(rt),14,15))<10,'tumor','normal'),
                levels = c('normal','tumor'))
table(group)
```

```
## group
## normal  tumor
##     19    408
```

Application of statistical models to answer the biologicalquestion.

Differential gene expression analysis

I set cutoff of logFC equal to 1 and threshold of False Discovery Rate (FDR) equal to 0.05.Then calculate the normalization factors (by TMM), the dispersion estimates, and the design matrix for "tumor" and "normal" group, we can fit our log-linear model. After that, I built a contrast matrix for them to find the differential expression genes between them. Used the likelihood ratio test to identify differentially expressed genes, via the glmLRT function. To identify which genes are differentially expressed, I used the topTags function, specifying a p-value cutoff.

```
# Differential gene expression analysis
logFC_cutoff=1
padj=0.05
dge <- DGEList(counts=rt,group=group)
dge$samples$lib.size <- colSums(dge$counts)
# Normalization
dge <- calcNormFactors(dge,method = "TMM")
# Design matrix
design <- model.matrix(~0+factor(group))
colnames(design) <- c('normal','tumor')
rownames(design)<-colnames(dge)
colnames(design)<-levels(group)
# estimate Divergence
dge <- estimateGLMCommonDisp(dge,design)
dge <- estimateGLMTrendedDisp(dge, design)
dge <- estimateGLMTagwiseDisp(dge, design)
fit <- glmFit(dge, design)
fit1 <- glmLRT(fit, contrast=c(-1,1))
# correct P value and extract gene
DEG1=topTags(fit1, n=nrow(rt))#n=nrow(rt)
DEG=as.data.frame(topTags(fit1,n=100))
```

```
DEG=as.data.frame(DEG)
DEG1=as.data.frame(DEG1)
```

Then I filtered genes which FDR<0.05 and absolute values of logFC>1

Biological interpretation of the results

From the differentially expressed genes, we can see that many of them have been previously reported in bladder cancer studies. For example, KIF18B and MEN1 are known to be over-expressed in bladder cancer, while MYOM1 is known to be under-expressed. Other genes, such as PLOD1, have also been associated with bladder cancer progression;KLHL41 stabilizes skeletal muscle sarcomeres by nonproteolytic ubiquitination

```
head(significant)
```

```
##              logFC     logCPM       LR       PValue          FDR
## MYOM1    -4.508849  2.6916941 407.4633 1.307068e-90 2.148036e-86
## KLHL41   -4.808690 -0.0438148 395.2625 5.919087e-88 4.863714e-84
## PPP1R12B -3.900942  6.7044694 364.4976 2.952954e-81 1.617628e-77
## PYGM     -4.635656  2.1761752 349.4665 5.537423e-78 2.275050e-74
## CLEC3B   -4.255142  1.4596345 346.9759 1.930613e-77 6.318279e-74
## SLMAP    -2.567311  6.3572242 346.6209 2.306783e-77 6.318279e-74
```

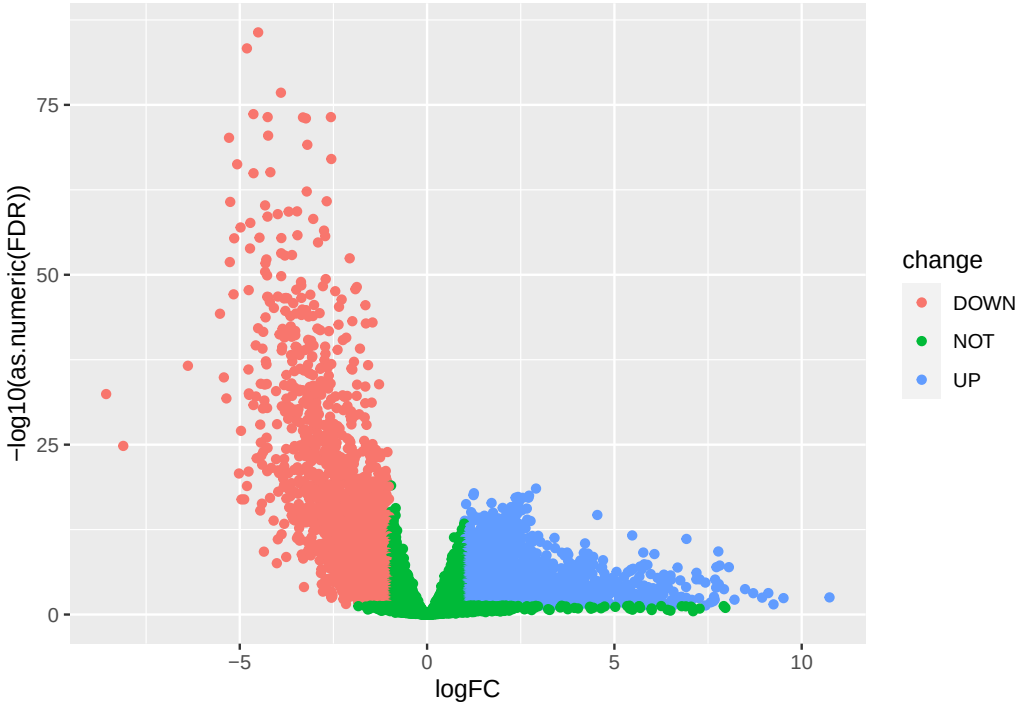
```
head(upregulation)
```

```
##           logFC   logCPM       LR       PValue          FDR
## KIF18B 2.907806 4.833981 87.60089 8.008877e-21 2.886357e-19
## MEN1   1.250202 6.046035 84.37720 4.088345e-20 1.405604e-18
## CASP2  1.239375 5.964202 83.05901 7.963995e-20 2.665586e-18
## EME1   2.722345 2.225569 82.68032 9.645673e-20 3.202364e-18
## XRCC2  2.428574 3.203921 81.79155 1.512234e-19 4.921198e-18
## TROAP  2.708089 4.839479 81.23655 2.002541e-19 6.450507e-18
```

```
head(downregulation)
```

```
##              logFC     logCPM       LR       PValue          FDR
## MYOM1    -4.508849  2.6916941 407.4633 1.307068e-90 2.148036e-86
## KLHL41   -4.808690 -0.0438148 395.2625 5.919087e-88 4.863714e-84
## PPP1R12B -3.900942  6.7044694 364.4976 2.952954e-81 1.617628e-77
## PYGM     -4.635656  2.1761752 349.4665 5.537423e-78 2.275050e-74
## CLEC3B   -4.255142  1.4596345 346.9759 1.930613e-77 6.318279e-74
## SLMAP    -2.567311  6.3572242 346.6209 2.306783e-77 6.318279e-74
```

Finally, used volcano plot to visualize differential expression gene and used heatmap to visualize different gene expression between tumor and normal samples. Since there are too many differential expression genes, I random select 100 genes from them to draw heatmap.




```
## NULL
```