CMSC 3375 Homework 2

Isabelle Hu

2020-10-15

The goal of this exercise is to cluster the TCGA-HNSC RNA-seq data from the GDC.

```
knitr::opts_chunk$set(echo = TRUE, tidy=TRUE, tidy.opts=list(width.cutoff=60))
library(TCGAbiolinks)
library(rmarkdown)
```

```
DataDirectory <- paste0("./GDC/", gsub("-", "_", "HNSC"))
FileNameData <- paste0(DataDirectory, "_", "HTSeq_Counts", ".rda")
```

Query RNA-seq data from the GDC TCGA-HNSC project. Here we use teh raw counts in the HTSeq-counts files. This query returns 546 samples.

```
# query TCGA-HNSC RNA-seq data
query1 <- GDCquery(project = "TCGA-HNSC", data.category = "Transcriptome Profiling",
    data.type = "Gene Expression Quantification", workflow.type = "HTSeq - Counts")
```

Then we select the subset whose sample type is either Primary Solid Tumor (TP) or Solid Tissue Normal (NT). This filtering process removes 2 samples, returning 544 samples.

```
# get case names from query
samplesDown <- getResults(query1, cols = c("cases"))
# filter cases for primary solid tumor (TP) and solid tissue
# normal (NT)
a.cases <- TCGAquery_SampleTypes(barcode = samplesDown, typesample = c("TP",
    "NT"))
```

Query RNA-seq data from the GDC TCGA-HNSC project again of the subset, then download the data.

```
# query data subset
query2 <- GDCquery(project = "TCGA-HNSC", data.category = "Transcriptome Profiling",
    data.type = "Gene Expression Quantification", workflow.type = "HTSeq - Counts",
    barcode = a.cases)
```

```
# download data
GDCdownload(query = query2, directory = DataDirectory)
```

For data preparation, we first read the data downloaded and prepare it into an R object.

Then preprocess the data using Array Array Intensity correlation (AAIC), which returns a square symmetric matrix of spearman correlation among samples. I am not using a cutoff value here, but I visually inspected the matrix and boxplot of correlation samples by samples and did not identify any outliers with low correlation.

Then normalize the RNA transcripts, which includes within-lane normalization procedures and between-lane normalization procedures. This step returns all mRNA with mean across all samples, higher than the threshold defined quantile mean across all samples.

Finally, filter the data to keep higher than the threshold defined quantile mean across all samples.

```
# data preparation
dataPrep <- GDCprepare(query = query2, directory = DataDirectory)
dataProc <- TCGAanalyze_Preprocessing(object = dataPrep, cor.cut = 0,
    datatype = "HTSeq - Counts")
dataNorm <- TCGAanalyze_Normalization(tabDF = dataProc, geneInfo = geneInfoHT,
    method = "gcContent")
dataFilt <- TCGAanalyze_Filtering(tabDF = dataNorm, method = "quantile",
    qnt.cut = 0.05)
```

In order to associate clusters with biological information later, we add the sample type information into the data matrix for each sample.

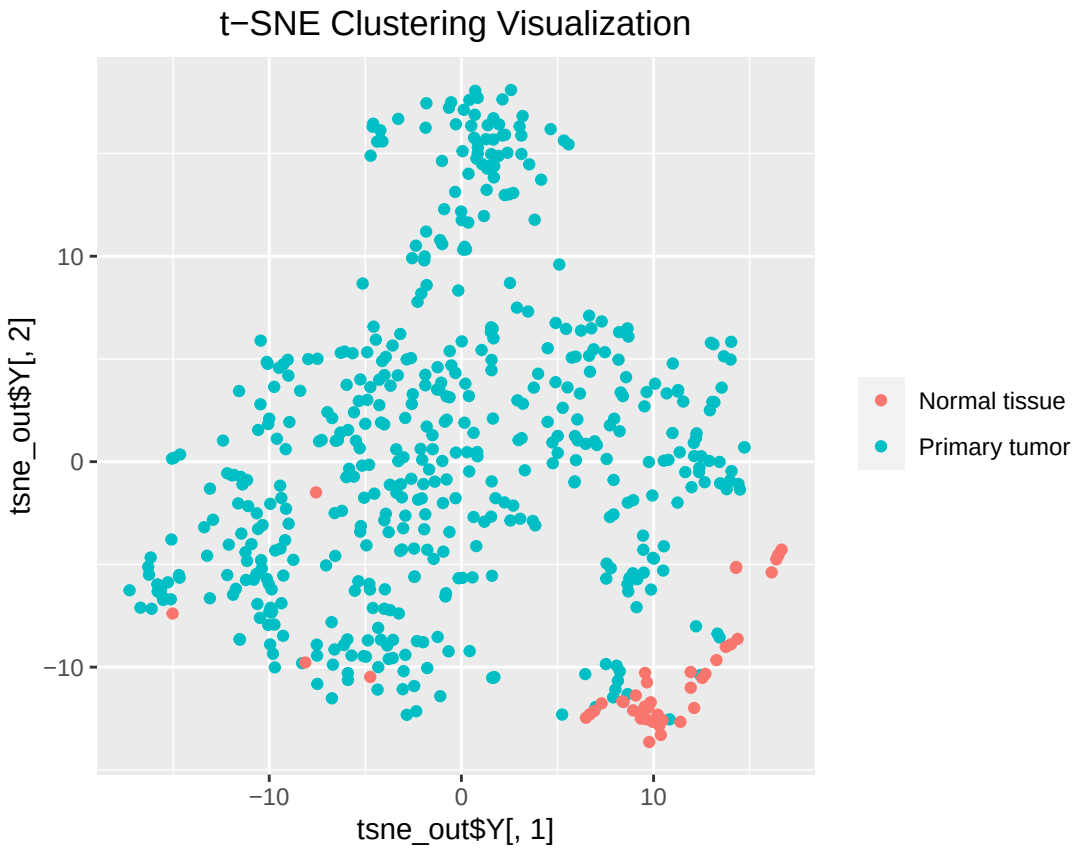
```
# get case names for primary solid tumor (TP) and solid
# tissue normal (NT) subsets
samplesNT <- TCGAquery_SampleTypes(colnames(dataFilt), typesample = c("NT"))
samplesTP <- TCGAquery_SampleTypes(colnames(dataFilt), typesample = c("TP"))

# transpose the filtered dataset and add a new column
# indicating the sample type
dataFilt_t_new <- cbind(t(dataFilt), c(samplesTP, samplesNT))
colnames(dataFilt_t_new)[ncol(dataFilt_t_new)] <- "Type"
dataFilt_t_new[samplesTP, "Type"] <- "Primary tumor"
dataFilt_t_new[samplesNT, "Type"] <- "Normal tissue"
```

Use the t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm to visualize clustering in the two-dimensional embedding space. The sample types are indicated by color in the figure. I found that not performing PCA dimension reduction prior to t-SNE gave the best clustering result. The two sample types are somewhat separated in the figure, but not very clearly separated with large margins.

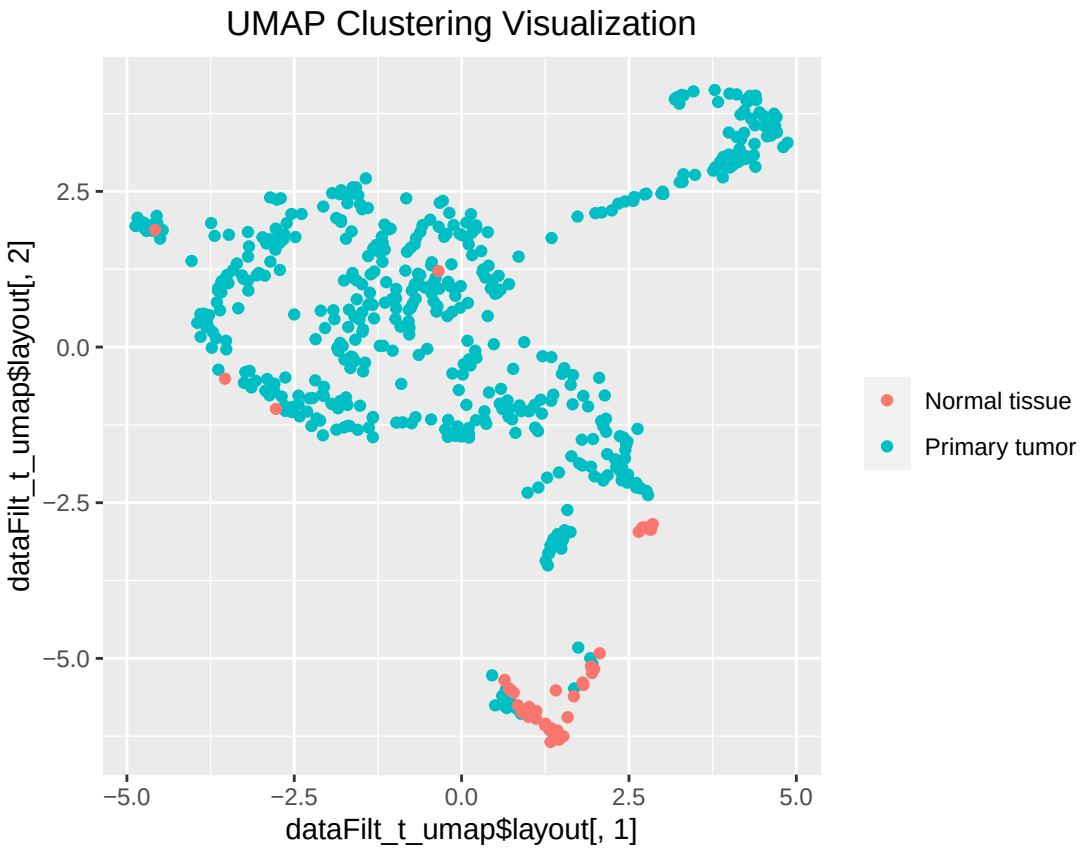
```
library(Rtsne)
tsne_out <- Rtsne(t(dataFilt), pca = FALSE, check_duplicates = FALSE)

library(ggplot2)
p <- qplot(x = tsne_out$Y[, 1], y = tsne_out$Y[, 2], col = factor(dataFilt_t_new[,
    ncol(dataFilt_t_new)]), geom = "point", asp = 1)
p + ggtitle("t-SNE Clustering Visualization") + theme(legend.title = element_blank(),
    plot.title = element_text(hjust = 0.5))
```



Use the Uniform Manifold Approximation and Projection (UMAP) algorithm to visualize clustering in 2D. The UMAP method resulted in more clearly separated clusters than the t-SNE method above. One of the clusters corresponds to (although not perfectly) the normal tissue samples.

```
library(umap)
dataFilt_t_umap <- umap(t(dataFilt))
p2 <- qplot(x = dataFilt_t_umap$layout[, 1], y = dataFilt_t_umap$layout[,
    2], col = factor(dataFilt_t_new[, ncol(dataFilt_t_new)]),
    geom = "point", asp = 1)
p2 + ggtitle("UMAP Clustering Visualization") + theme(legend.title = element_blank(),
    plot.title = element_text(hjust = 0.5))
```


UMAP Clustering Visualization
dataFilt_t_umap$layout[, 2]
2.5
0.0
−2.5
−5.0
−5.0
−2.5
0.0
2.5
5.0
dataFilt_t_umap$layout[, 1]
Normal tissue
Primary tumor