Chapter 9

RNA-Seq Experiment and Data Analysis

Hanquan Liang and Erliang Zeng

Abstract

With the ability to obtain tens of millions of reads, high-throughput messenger RNA sequencing (RNA-Seq) data offers the possibility of estimating abundance of isoforms and finding novel transcripts. In this chapter, we describe a protocol to construct an RNA-Seq library for sequencing on Illumina NGS platforms, and a computational pipeline to perform RNA-Seq data analysis. The protocols described in this chapter can be applied to the analysis of differential gene expression in control versus 17β-estradiol treatment of in vivo or in vitro systems.

Key words RNA-Seq, Next-generation sequencing, Data analysis, Bioconductor, Statistical analysis, Differentially expressed genes

1 Introduction

Recent advancements in next-generation sequencing (NGS) technologies enable sequencing of tens of millions to billions of cDNA fragments generated from RNA, offering great opportunity to directly quantify entire messenger RNA (mRNA) from a sample [1, 2]. In high-throughput transcript data a large number of transcripts are concurrently sensed using fragments of cDNAs (called reads), with the idea that abundance of a transcript is estimated by integrating the counts of reads likely to be produced from the transcript [3–6]. If the reads are uniquely associated with a transcript, estimating its expression value is relatively simple—roughly speaking, the total read counts associated with a transcript divided by the base length of the transcript with a fixed scaling gives an estimate [4]. There are many platforms that can be used for NGS and many pathways for data analysis. This chapter describes how to prepare an RNA-Seq library for sequencing on Illumina platforms [7], and how to use a computational pipeline for estimating expression of transcripts and for statistical analysis to discover differentially expressed genes (DEGs). Starting with total RNA, the library construction steps include mRNA purification and fragmentation,

Kathleen M. Eyster (ed.), *Estrogen Receptors: Methods and Protocols*, Methods in Molecular Biology, vol. 1366,
DOI 10.1007/978-1-4939-3127-9_9,

first-strand cDNA synthesis, and second-strand cDNA synthesis. The double-strand cDNA is then ligated to adapters, and enriched with PCR amplification. The computational pipeline for RNA-Seq data analysis consists of steps for data quality assessment, read aligning and gene expression estimating, and statistical analysis. The protocols described in this chapter can be applied to the analysis of differential gene expression in control versus 17β-estradiol treatment of in vivo or in vitro systems.

2 Materials

1. Samples of purified total RNA from control and 17β-estradiol treated cells or tissues.
2. Oligo (dT)25 magnetic beads and magnetic stand (Invitrogen).
3. Washing Buffer B, comes with Oligo (dT)25 magnetic beads reagent (Invitrogen).
4. Binding Buffer, comes with Oligo (dT)25 magnetic beads reagent (Invitrogen).
5. 10 mM Tris–HCl, pH 7.5 (Invitrogen).
6. 10× RNA Fragmentation Buffer (New England Biolabs).
7. 10× RNA Fragmentation Stop Solution (New England Biolabs).
8. Random hexamers (100 pmol/μL).
9. First-Strand Reaction Buffer (Invitrogen): 4 μL 5× first-strand buffer, 2 μL 100 mM dithiothreitol, 1 μL dNTP mix, 1 μL SuperScript II per reaction tube.
10. dNTP mix (10 mM each dATP, dCTP, dGTP, dTTP).
11. Second-strand reaction buffer: 51 μL nuclease-free water, 20 μL 5× second-strand reaction buffer, 2 μL dNTP mix, 5 μL *E. coli* DNA Polymerase I (10 U/μL) (New England Biolabs, NEB), 1 μL *E. coli* DNA Ligase (10 U/μL) (NEB), 1 μL *E. coli* RNase H (5 U/μL) (NEB) per reaction tube.
12. dA tailing mix: 5 μL 10× A-tailing buffer, 10 μL 1 mM dATP, 3 μL Klenow fragment (3′→5′ exo-) per reaction (NEB).
13. End repair reaction mix: 25 μL nuclease-free water, 10 μL 10× phosphorylation reaction buffer, 4 μL dNTP mix, 5 μL T4 DNA polymerase, 1 μL *E. coli* DNA polymerase I, Large (Klenow) Fragment, 5 μL T4 Polynucleotide Kinase (New England Biolabs).
14. T4 DNA ligation mix: 5 μL nuclease-free water, 3 μL 10× T4 DNA ligation buffer (NEB), 1 μL Illumina Adapters (Illumina), 1 μL T4 DNA ligase (NEB).
15. Universal Primer Mix (10 μM each forward & reverse).

16. 2× Phusion High-Fidelity PCR Master Mix (Thermo Scientific).
17. QIAquick PCR Purification Kit (Qiagen).
18. RNeasy MinElute Cleanup Kit (Qiagen).
19. AMPure XP beads (Beckman Coulter).
20. Nuclease-free water.
21. 80 % ethanol.
22. Qubit Fluorometer (Invitrogen).
23. Thermal cycler.
24. Agilent 2100 Bioanalyzer (Agilent).

3 Methods

3.1 RNA-Seq Experiment

Before starting, it is highly recommended to assess the total RNA quality of the samples using an Agilent Bioanalyzer 2100. To achieve the best results, the RNA Integrity Number (RIN) estimated by the Bioanalyzer should be 8 or higher. Low quality samples may yield seemingly good libraries and good sequencing reads, but analysis of such data is challenging and the results may be misleading. In the second-strand cDNA synthesis reaction, the second strand of cDNA is synthesized and the RNA templates are removed.

In the mRNA purification steps, messenger RNA with poly-A tails will be captured, while other RNA components (rRNA, tRNA) will be removed from the samples. For mRNA fragmentation, the mRNA is cleaved into small pieces by heating in divalent metal cation buffer. The first-strand cDNA synthesis steps will generate the first strand of cDNA using reverse transcriptase and random primers. The second-strand cDNA synthesis steps generate the second strand of cDNA and removes the RNA templates from the reaction. In the end repair steps, the ends of double-strand cDNA fragments are converted into blunt ends. An adenine (A) base is then added to the 3′-end of blunt double strand cDNA to facilitate adapter ligation. The adaptor ligated cDNA is then enriched and amplified by PCR to achieve a sufficient amount of library. Finally the library is quantified and its quality is checked.

1. For mRNA purification, dilute 100–1000 ng of total RNA with nuclease-free water to a final volume of 50 μL (*see* **Note 1**).
2. Add 50 μL resuspended oligo-dT beads (prepared according to the manufacturer's recommendation if necessary) to each RNA sample.
3. Place the samples in a thermal cycler and heat at 65 °C for 5 min, and then 4 °C on hold. Remove the tubes from the thermal cycler after the temperature reaches 4 °C.
4. Incubate samples at room temperature for 5 min (*see* **Note 2**).

5. Place the samples on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully remove and discard supernatants without disturbing the beads (*see* **Note 3**).
6. Remove the tubes from the magnetic stand. Resuspend the beads with 200 μL washing buffer B.
7. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully remove and discard the supernatants without disturbing the beads.
8. Remove the tubes from the magnetic stand. Resuspend the beads with 50 μL 10 mM Tris–HCl buffer.
9. Place the tubes in a thermal cycler. Heat the samples at 80 °C for 2 min, and then 25 °C on hold. In this step, binding is disrupted and RNA is released into the supernatant.
10. Remove the tubes from the thermal cycler. Add 50 μL binding buffer and resuspend beads.
11. Incubate the samples at room temperature for 5 min. In this step, the mRNA rebinds to the poly-dT beads.
12. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully remove and discard the supernatants without disturbing the beads.
13. Remove the tubes from the magnetic stand. Resuspend the beads with 200 μL washing buffer B.
14. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully remove and discard the supernatants without disturbing the beads (*see* **Note 4**)
15. Remove the tubes from the magnetic stand. Resuspend the beads with 20 μL of nuclease-free water.
16. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully transfer 18 μL of the supernatant to new tubes.
17. For mRNA fragmentation, add 2 μL 10× RNA fragmentation buffer to the samples.
18. Place the tubes in a thermal cycler and incubate at 94 °C for 5 min. Immediately transfer the tubes to ice (*see* **Note 5**).
19. Add 2 μL 10× RNA fragmentation stop solution.
20. Clean up fragmented RNA using RNeasy MinElute columns following the manufacturer's instructions. Elute fragmented mRNA with 12 μL nuclease-free water.

21. For first-strand cDNA synthesis, add 1 μL random hexamers. Incubate at 70 °C for 10 min, and quick-chill on ice.
22. Add 8 μL first-strand cDNA reaction mixture to each reaction.
23. Incubate at 25 °C for 10 min, 42 °C for 50 min, 70 °C for 15 min, hold at 4 °C (*see* **Note 6**).
24. For second-strand cDNA synthesis, add 80 μL of second-strand reaction mixture to each reaction.
25. Place the tubes in a thermal cycler at 16 °C for 2 h.
26. Purify the double-stranded cDNA using the QIAquick PCR Purification Kit following the manufacturer's directions and elute in 50 μL nuclease-free water.
27. For the end repair steps, add 50 μL of end repair reaction mix to each double-stranded cDNA sample.
28. Incubate at 20 °C for 30 min.
29. Purify end-repaired double-stranded cDNA using the QIAquick PCR Purification Kit and elute in 32 μL nuclease-free water.
30. To carry out the dA tailing step, add 18 μL of dA tailing mix to each 32 μL of end-repaired cDNA.
31. Incubate at 37 °C for 30 min.
32. Purify dA-tailed double-stranded cDNA using QIAquick PCR Purification Kit and elute in 20 μL nuclease-free water.
33. To ligate the adapters, add 10 μL of T4 DNA ligation mix to each dA-tailed sample (20 μL) (*see* **Note 7**).
34. Incubate the samples in a thermal cycler at 30 °C for 10 min.
35. Add 1.2× (42 μL) resuspended AMPure XP beads to each reaction and mix thoroughly by pipetting. Incubate at room temperature for 10 min.
36. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully remove and discard the supernatants without disturbing the beads.
37. With the tubes on the magnetic stand, add 200 μL of 80 % freshly prepared ethanol.
38. Incubate at room temperature for 30 s, and then discard all supernatant without disturbing the beads.
39. Repeat ethanol wash one more time.
40. Keep the tubes on the magnetic stand, leave lids open and air-dry the beads for 10 min (but do not over dry).
41. Remove tubes from magnetic stand. Suspend beads with 52 μL nuclease-free water, and then incubate at room temperature for 2 min.

42. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully transfer 50 μL of the supernatant to new tubes.
43. Perform size selection by purifying the adapter-ligated DNA with 1× (50 μL) AMPure XP beads (**steps 35–40**) (*see* **Note 8**).
44. Remove tubes from magnetic stand. Suspend beads with 25 μL nuclease-free water, and then incubate at room temperature for 2 min.
45. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully transfer 23 μL of the supernatant to new tubes.
46. For PCR amplification of the cDNA samples, add 2 μL Universal Primer Mix and 25 μL of 2× Phusion High-Fidelity PCR Master Mix to each adapter-ligated cDNA (23 μL).
47. Place the samples in a thermal cycler and program the cycles as follows (*see* **Note 9**):
    (a) 98 °C for 30 s
    (b) 98 °C for 10 s
    (c) 60 °C for 30 s
    (d) 72 °C for 30 s
    (e) Go to **step** (**b**) for 14×
    (f) 72 °C for 5 min
    (g) Hold at 10 °C
48. Purify the PCR-enriched library with 1× (50 μl) AMPure XP beads (**steps 35–40**).
49. Remove tubes from magnetic stand. Suspend beads with 22 μL nuclease-free water, and then incubate at room temperature for 2 min.
50. Place the tubes on a magnetic stand for 5 min or until the solution is clear. Keeping the tubes on the magnetic stand, carefully transfer 20 μL of the supernatant to new tubes.
51. Use an Agilent Bioanalyzer to check the quality of the library and to estimate library size (*see* **Note 10**).
52. Quantify library using Qubit Assay (Invitrogen) by following the manufacturer's documents.
53. Based on library size and concentration, dilute library to 20 nM. Pool libraries if necessary (*see* **Note 11**).
54. Store the library at −20 °C.
55. Perform sequencing by following the manufacturer's protocol. This library is designed for sequencing on the Illumina platform. Alternatively, send the samples out for sequencing.

3.2 RNA-Seq Data Analysis

A functional laptop/desktop with 4GB or larger of RAM is required for running the computational pipeline. The computational pipeline for analysis of RNAseq data described here includes a series of steps. First, some simple quality control checks need to be performed to ensure that the raw data is of good quality before analyzing the RNA-Seq data [8]. FastQC is a computational tool that provides a QC report which can spot problems originating either in the sequencer or in the starting library material [9]. Second, the R and Bioconductor (a set of packages that run in R) programs will be installed. R and Bioconductor will be used to perform most of the analyses [10, 11]. R is a free, very powerful statistics environment [10]. The sequence files will be read and the sequence reads will be mapped to a reference genome. Gene expression values will be estimated by calculating either the raw read count number or Reads Per Kilobase of transcript per Million reads mapped (RPKM). Statistical analyses will be performed to discover DEGs.

1. For the quality control checks, download an appropriate version of FastQC from http://www.bioinformatics.babraham.ac.uk/projects/fastqc/ (*see* **Note 12**).
2. Install FastQC as instructed (*see* **Note 13**).
3. Perform basic operations in FastQC, including opening a sequence file, evaluating results, and saving a report (*see* **Note 14**).
4. Go to R project website (http://www.r-project.org/), download and install an appropriate R version (R Version 3.1.2 and up is recommended).
5. Start R on your computer.
6. After starting R, paste or type following commands to install Bioconductor (if the Bioconductor program has not been installed before). The installation will take a few minutes.

```
source("http://bioconductor.org/biocLite.R")
biocLite()
```

7. Set working director to your local folder. In the following command, change "path to your local folder" to your specific working directory. For example, I set my working directory as *setwd("/Users/ezeng/Documents/Teaching/RNA-Seq")*. Note that my data subdirectory is "*/Users/ezeng/Documents/Teaching/RNA-Seq/*data." The data subdirectory contains RNA-Seq FASTQ files from the study of human estrogen receptors, as well as the corresponding human reference genome sequence (FASTA) and annotation (GFF) file (*see* **Note 15**).

```
setwd("path to your local folder")
```

8. Install and load *QuasR* [12]. *QuasR* is a versatile NGS mapping and post-processing pipeline for RNA-Seq data analysis.

It uses *Rbowtie* for ungapped alignments and *SpliceMap* for spliced alignments. Install and load *QuasR* package.

```
biocLite("QuasR")
library(QuasR)
```

9. To read sequence files and map sequence reads to the reference genome, first read sequence file information. The FASTQ files are organized in the provided samples.txt file in data subdirectory (*see* **Note 16**). To import samples.txt, we run the following commands from R.

```
samples <- read.delim("data/samples.txt")
```

10. Set environment.

```
write.table(samples[,1:2], "data/QuasR_samples.txt",
    row.names=FALSE, quote=FALSE, sep="\t")
sampleFile <- "./data/QuasR_samples.txt"
genomeFile <- "./data/Homo_sapiens.GRCh38.dna.top-
    level.fa"
# Note: all output data will be written to subdirec-
    tory 'results'
dir.create("results")
# Defines location where to write results
results <- "./results"
# Defines number of CPU cores to use
cl <- makeCluster(1)
```

11. Use single command to index reference, align all samples, and generate BAM files (*see* **Note 17**).

```
proj <- qAlign(sampleFile, genome=genomeFile, max-
    Hits=1, splicedAlignment=FALSE, alignmentsDir=
    results, clObj=cl, cacheDir=results)
```

12. Get alignment summary report.

```
alignstats <- alignmentStats(proj)
alignstats
```

13. Enumerate the number of reads in each FASTQ file and how many of them aligned to the reference. For *QuasR* this step can be omitted because the *qAlign* function generates this information automatically.

```
biocLite("ShortRead")
biocLite("Rsamtools")
library(ShortRead)
library(Rsamtools)
Nreads <- countLines(dirPath="./data", pattern=".
    fastq$")/4
bfl <- BamFileList(alignments(proj)$genome$FileName,
    yieldSize=50000, index=character())
```

```
Nalign <- countBam(bfl)
read_statsDF <- data.frame(FileName=names(Nreads), Nreads=Nreads, Nalign=Nalign$records, Perc_Aligned=Nalign$records/Nreads*100)
write.table(read_statsDF, "results/read_statsDF.xls", row.names=FALSE, quote=FALSE, sep="\t")
```

14. To estimate the gene expression values, calculate either raw read count number or RPKM. To do this, get annotation data from GFF.

```
biocLite("rtracklayer")
biocLite("GenomicRanges")
library(rtracklayer)
library(GenomicRanges)
gff <- import.gff("./data/ref_GRCh38_top_level.gff3", asRangedData=FALSE)
seqlengths(gff) <- end(ranges(gff[which(elementMetadata(gff)[,"type"]=="chromosome"),]))
subgene_index <- which(elementMetadata(gff)[,"type"] == "exon")
gffsub <- gff[subgene_index,] # Returns only gene ranges
ids <- gsub("Parent=|\\..*", "", elementMetadata(gffsub)$group)
gffsub <- split(gffsub, ids) # Coerce to GRangesList
```

15. Store annotation rangers in *TranscriptDb* databases, which make many operations more robust and convenient.

```
biocLite("GenomicFeatures")
library(GenomicFeatures)
txdb <- makeTranscriptDbFromGFF(file="data/ref_GRCh38_top_level.gff3",
format="gff3",
dataSource="NCBI",
species="Homo sapiens")
saveDb(txdb, file="./data/GRCh38.sqlite")
txdb <- loadDb("./data/GRCh38.sqlite")
eByg <- exonsBy(txdb, by="gene")
```

16. Read counting with *qCount* from *QuasR*.

```
countDF <- qCount(proj, txdb, reportLevel="gene", orientation="any")
write.table(countDF, "results/countDFgene.xls", col.names=NA, quote=FALSE, sep="\t")
write.table(countDF[,2:5], "./results/countDF", quote=FALSE, sep="\t", col.names = NA)
```

17. Perform a simple RPKM normalization (*see* **Note 18** for an alternative way to calculate RPKM).

```
returnRPKM <- function(counts, gffsub) {
geneLengthsInKB <- sum(width(reduce(gffsub)))/1000
millionsMapped <- sum(counts)/1e+06
rpm <- counts/millionsMapped
rpkm <- rpm/geneLengthsInKB
return(rpkm)
}
countDFrpkm <- apply(countDF, 2, function(x)
    returnRPKM(counts=x, gffsub=eByg))
```

18. Check the sample reproducibility by computing a correlating matrix and plotting it as a tree (*see* **Note 19**).

```
biocLite("ape")
library(ape)
d <- cor(rpkmDFgene, method="spearman")
hc <- hclust(dist(1-d))
plot.phylo(as.phylo(hc), type="p", edge.col=4, edge.
    width=3, show.node.label=TRUE, no.margin=TRUE)
```

19. To perform statistical analyses to discover differentially expressed genes (DEGs), define the *colAg()* function (*see* **Note 20**).

```
colAg <- function(myMA=myMA, group=c(1,1,1,2,2,2,3,
    3,4,4), myfct=mean, …) {
myList <- tapply(colnames(myMA), group, list)
names(myList) <- sapply(myList, paste, collapse=
    "_")
myMAmean <- sapply(myList, function(x) apply(myMA
    [, x,
drop=FALSE], 1, myfct, …))
return(myMAmean)
}
```

20. Compute mean values for replicates using function *colAg()* (*see* **Note 21**).

```
countDFrpkm_mean <- colAg(myMA=rpkmDFgene, group=
    c(1,1,2,2), myfct=mean)
```

21. Calculate $\log_2$ fold changes.

```
countDFrpkm_mean <- cbind(countDFrpkm_mean, log2ratio=
    log2(countDFrpkm_mean[,2]/countDFrpkm_mean[,1]))
countDFrpkm_mean <- countDFrpkm_mean[is.
    finite(countDFrpkm_mean[,3]),]
degs2fold <- countDFrpkm_mean[countDFrpkm_mean[,3]
    >= 1 | countDFrpkm_mean[,3] <= -1,]
write.table(degs2fold, "./results/degs2fold.xls",
    quote=FALSE, sep="\t", col.names = NA)
degs2fold <- read.table("./results/degs2fold.xls")
```

22. Perform statistical analysis with DESeq library (*see* **Note 22**). Note that DESeq is expected to use raw count data [13].

```
biocLite("DESeq")
library(DESeq)
countDF <- read.table("./results/countDF")
conds <- samples$Factor
# Creates object of class CountDataSet derived from
   eSet class
cds <- newCountDataSet(countDF, conds)
# Estimates library size factors from count data.
cds <- estimateSizeFactors(cds)
# Estimates the variance within replicates
cds <- estimateDispersions(cds)
# Calls DEGs with nbinomTest
res <- nbinomTest(cds, "Control", "Treatment")
res <- na.omit(res)
res2fold <- res[res$log2FoldChange >= 1 |
   res$log2FoldChange <= -1,]
res2foldpadj <- res2fold[res2fold$padj <= 0.05,]
```

23. Perform statistical analysis with edgeR library (*see* **Note 22**). Note that edgeR is also expected to use raw count data [14].

```
biocLite("edgeR")
library(edgeR)
countDF <- read.table("./results/countDF")
# Constructs DGEList object
y <- DGEList(counts=countDF, group=conds)
# Estimates common dispersion
y <- estimateCommonDisp(y)
# Estimates tagwise dispersion
y <- estimateTagwiseDisp(y)
# Computes exact test for the negative binomial
   distribution.
et <- exactTest(y, pair=c("Control", "Treatment"))
topTags(et, n=4)
edge <- as.data.frame(topTags(et, n=50000))
edge2fold <- edge[edge$logFC >= 1 | edge$logFC <=
   -1,]
edge2foldpadj <- edge2fold[edge2fold$FDR <= 0.05,]
```

24. Perform statistical analysis with edgeR using generalized linear models (glms) (*see* **Note 22**).

```
library(edgeR)
countDF <- read.table("./results/countDF")
```

```
# Constructs DGEList object
y <- DGEList(counts=countDF, group=conds)
# Filtering and normalization
keep <- rowSums(cpm(y)>1) >= 2; y <- y[keep,]
y <- calcNormFactors(y)
# Design matrix
design <- model.matrix(~0+group, data=y$samples);
   colnames(design) <- levels(y$samples$group)
# Estimates common dispersions
y <- estimateGLMCommonDisp(y, design, verbose=TRUE)
# Estimates trended dispersions
y <- estimateGLMTrendedDisp(y, design)
# Estimates tagwise dispersions
y <- estimateGLMTagwiseDisp(y, design)
# Fit the negative binomial GLM for each tag
fit <- glmFit(y, design)
# Contrast matrix is optional
contrasts <- makeContrasts(contrasts="AP3-TRL",
   levels=design)
# Takes DGEGLM object and carries out the likelihood
   ratio test
lrt <- glmLRT(fit, contrast=contrasts[,1])
edgeglm <- as.data.frame(topTags(lrt, n=length
   (rownames(y))))
# Filter on fold change and FDR
edgeglm2fold <- edgeglm[edgeglm$logFC >= 1 |
   edgeglm$logFC <= -1,]
edgeglm2foldpadj <- edgeglm2fold[edgeglm2fold$FDR
   <= 0.05,]
```

25. Heatmap of top-ranking DEGs (*see* **Note 23**).

```
biocLite("lattice")
biocLite("gplots")
library(lattice)
library(gplots)
y <- countDFrpkm[rownames(edgeglm2foldpadj)[1:20],]
colnames(y) <- targets$Factor
y <- t(scale(t(as.matrix(y))))
y <- y[order(y[,1]),]
levelplot(t(y), height=0.2, col.regions=colorpanel
   (40, "darkblue", "yellow", "white"), main=
   "Expression Values (DEG Filter: FDR 5 %, FC > 2)",
   colorkey=list(space="top"), xlab="", ylab="Gene
   ID")
```

4 Notes

1. Although it is possible to start with 100 ng total RNA or even less, a smaller quantity of starting material will yield suboptimal results, e.g., inefficient adapter ligation, lower library yield, and inaccurate quantification.
2. This heating step denatures the RNA and disrupts secondary structures.
3. Allow beads to fully pellet against magnetic stand. Do not allow the beads to dry.
4. Take care to remove all supernatant; fragmentation will be affected if there is contamination with residual washing buffer.
5. A shorter incubation time will result in longer fragments. Fragment size can be checked by running 1 μL of isolated RNA on Agilent BioAnalyzer.
6. The 42 °C incubation is for reverse transcription. The reverse transcriptase is deactivated during the 70 °C incubation.
7. If multiple samples will be sequenced in the same lane, use adapters with different index. Take care to avoid cross contamination.
8. Library size (including ds cDNA insert and adapters) can be adjusted by changing the volume ratio of bead:sample. Increased bead to DNA ratio recovers more shorter fragments, while keeping the long fragments. When only longer fragments are desired, user should try lower the ratio, thus to remove short fragments. However, lower ratios usually result in lower yields.
9. Too many PCR cycles may introduce bias (certain sequences get more representation in the RNA-Seq data) and higher duplication rate (same sequence get sequenced multiple times).
10. Bioanalyzer electropherograms of libraries described in this protocol usually show a peak starting from 200 bp to 500 bp, with a summit at around 260–280 bp. Pay attention to any adapter/dimer peaks, which show up approximately 60 bp or 120 bp if they are present. Adapters have negative effects for library quantification and cluster generation. If necessary, perform one more round of purification to remove adapters. Average library size can be estimated by Bioanalyzer software (refer to the manufacturer's documents).
11. It is highly recommended to have sufficient index diversity in a pooled library. Low diversity in the index sequences would result in unbalanced signals and low base-calling quality, which makes de-multiplexing difficult. Illumina provides software (Illumina Experiment Manager, http://support.illumina.com/sequencing/sequencing_software/experiment_manager/

downloads.html) to help check index compatibility for pooling.

12. Based on your operating system, there are Windows, Linux, and MAC versions available for downloading. Users can also download source code and build FastQC from scratch.
13. Detailed installation and setup instructions are available on the website http://www.bioinformatics.babraham.ac.uk/projects/fastqc/INSTALL.txt. FastQC is a java application. In order to run, your system must have a suitable Java Runtime Environment (JRE) installed.
14. Detailed information about basic FastQC operations can be found on http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/2%20Basic%20Operations/. Documentation about analysis modules is available on http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/3%20Analysis%20Modules/. It is important to notice that although the analysis results appear to give a pass/fail result, these evaluations must be taken in the context of what you expect from your library. A "normal" sample as far as FastQC is concerned is random and diverse. Users should treat the summary evaluations as pointers to their own concentration, if the experiments are expected to produce libraries that are biased in particular ways. An example report of good Illumina data can be found at http://www.bioinformatics.babraham.ac.uk/projects/fastqc/good_sequence_short_fastqc.html, and an example report of bad Illumina data can be found at http://www.bioinformatics.babraham.ac.uk/projects/fastqc/bad_sequence_fastqc.html. It is recommended to perform additional data trimming and filtering if needed.
15. Download human reference genome sequence (FASTA) from ftp://ftp.ensembl.org/pub/release-78/fasta/homo_sapiens/dna/Homo_sapiens.GRCh38.dna.toplevel.fa.gz, unzip it and store it to the data subdirectory. Download the annotation (GFF) file from ftp://ftp.ncbi.nih.gov/genomes/Homo_sapiens/GFF/ref_GRCh38_top_level.gff3.gz, unzip it, and store it to the data subdirectory.
16. The file samples.txt is a tab-delimited file that records information about experiments. An example of the content in samples.txt is as following:

FileName	SampleName	Factor
Seq1.fastq	Sample1	Control
Seq2.fastq	Sample2	Control
Seq3.fastq	Sample3	Treatment
Seq4.fastq	Sample4	Treatment

17. In this command, *splicedAlignment* should be set to *TRUE* when reads are >=*50 nt* long. In this example, the read length is short and less than 50 nt, so *splicedAlignment* was set to *FALSE*.

18. An alternative way to calculate RPKM, you will see same results stored in *countDFrpkm* and *rpkmDFgene*

```
rpkmDFgene <- t(t(countDF[,-1]/countDF[,1] *1000)/
   colSums(countDF[,-1]) *1e6)
```

19. The *plotMDS* function from *edgeR* is a more robust method for this task.

20. The *colAg()* function is a convenience function for applying a variety of computations on any combination of column aggregates in a matrix or data frame.
    How to run the function:

```
myMA <- matrix(rnorm(100000), 10000, 10, dimnames=
   list(1:10000, paste("C", 1:10, sep="")))
```

```
colAg(myMA=myMA, group=c(1,1,1,2,2,2,3,3,4,4), myfct=
   mean)[1:4,]
```

21. In this example, four samples are assumed to be analyzed, of which, two are samples for controls, and the other two are group with treatment (the same as example samples.txt in **Note 16**). Users need change the code accordingly, based on their own experimental designs. For example, if the experiment has three groups, each having three replicates, then the command should like this:

```
countDFrpkm_mean <- colAg(myMA=rpkmDFgene, group=
   c(1,1,1,2,2,2,3,3,3), myfct=mean)
```

22. There are different ways to discover DEGs using statistical analysis. **Steps 19**, **20**, and **21** include commands to discover DEGs using three methods. Users can select all of the three methods, and compare the sets of three DEGs using visualization tool such as Venn diagram. Note that the final DEGs as obtained from **steps 19**, **20**, and **21** require twofold change or larger and an adjusted p-value less than 0.05. Users may change these parameters accordingly.

23. These commands generate a heatmap of top 20 DEGs resulting from **step 21**. Users can change related parameters to generate heatmaps resulting from **steps 19** and **20** as well.

Acknowledgments

We thank Dr. Thomas Girke at the University of California Riverside for sharing his R scripts.

References

1. Wang Z, Gerstein M, Snyder M (2009) RNA-Seq: a revolutionary tool for transcriptomics. Nat Rev Genet 10(1):57–63
2. Ozsolak F, Milos PM (2011) RNA sequencing: advances, challenges and opportunities. Nat Rev Genet 12:87–98
3. Marioni JC, Mason CE, Mane SM, Stephens M, Gilad Y (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. Genome Res 18(9): 1509–1517
4. Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B (2008) Mapping and quantifying mammalian transcriptomes by RNA-seq. Nat Methods 5(7):621–628
5. Twine NA, Janitz K, Wilkins MR, Janitz M (2011) Whole transcriptome sequencing reveals gene expression and splicing differences in brain regions affected by Alzheimer's disease. PLoS One 6(1), e16266
6. Eksi R, Li HD, Menon R et al (2013) Systematically differentiating functions for alternatively spliced isoforms through integrating RNA-seq data. PLoS Comput Biol 9(11), e1003314
7. http://www.illumina.com/applications/sequencing/rna/mrna-seq.html
8. Leggett RM, Ramirez-Gonzalez RH, Clavijo BJ, Waite D, Davey RP (2013) Sequencing quality assessment tools to enable data-driven informatics for high throughput genomics. Front Genet 4:288
9. Andrews S (2010) FastQC: a quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
10. R: A language and environment for statistical computing. http://www.r-project.org/
11. Gentleman RC, Carey VJ, Bates DM et al (2004) Bioconductor: open software development for computational biology and bioinformatics. Genome Biol 5:R80
12. Gaidatzis D, Lerch A, Hahne F, Stadler MB (2014) QuasR: quantification and annotation of short reads in R. Bioinformatics pii, btu781
13. Anders S, Huber W (2010) Differential expression analysis for sequence count data. Genome Biol 11:R106
14. Robinson MD, McCarthy DJ, Smyth GK (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 26: 139–140