Next-Generation sequencing: RNAseq

Sulev Kõks

Transcriptome

- … entire repertoire of transcripts, key link between information in DNA and phenotype
- Critical to explain how genotype affects phenotype
- Array technology – signals from hybridization
- Limitations:
  - background hybridization,
  - probes have different hybridization properties
  - limited with relevant probes on the array

RNA-seq

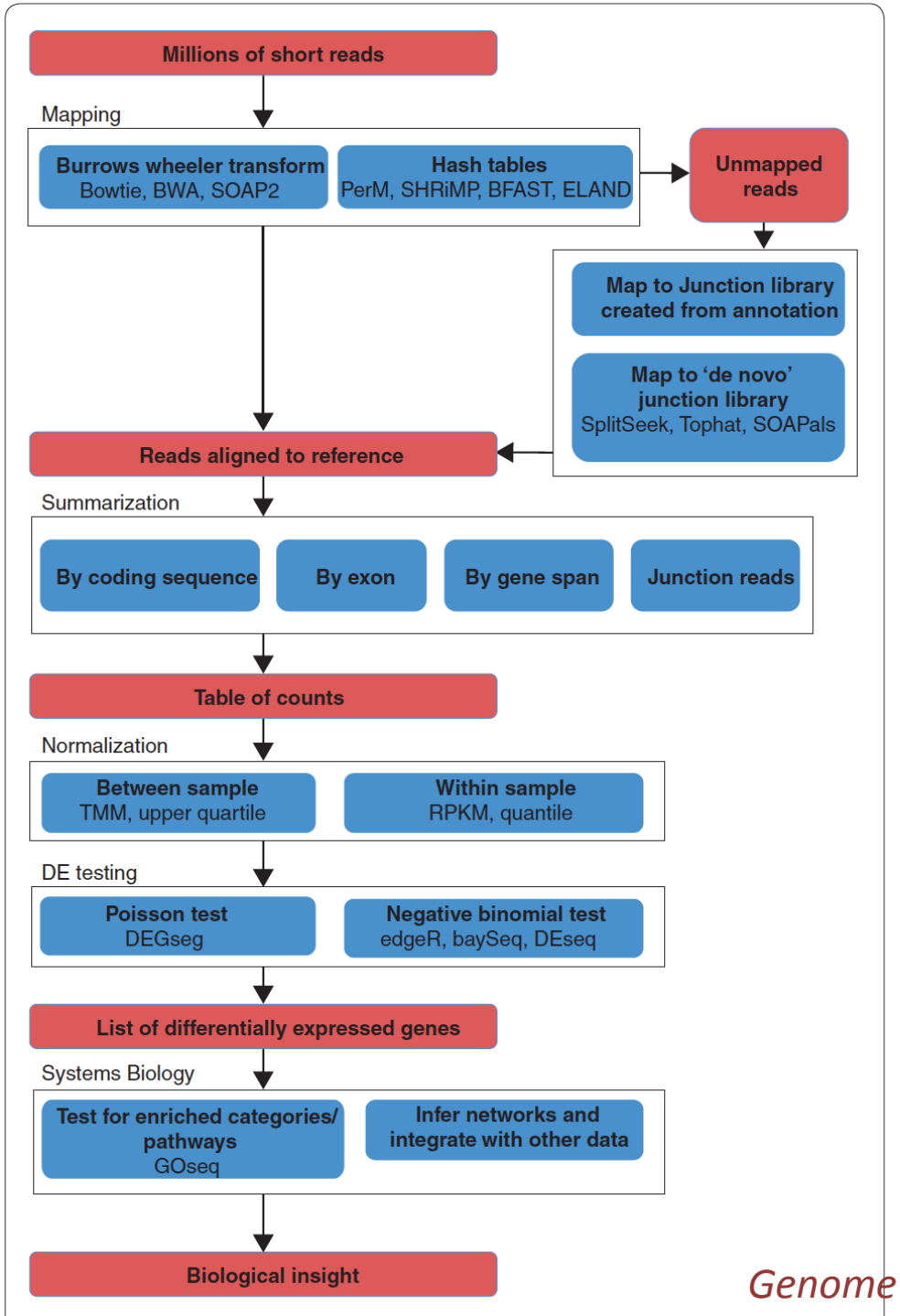
- Not limited to the prefabricated tools
- All species, all transcripts can be analyzed
- Better dynamic range, very good resolution
- Alternative splicing
- Fusions, junctions
- Gene expression regulation (eQTL)
- Allele specific expression (ASE)
- RNA isoforms

RNA-seq

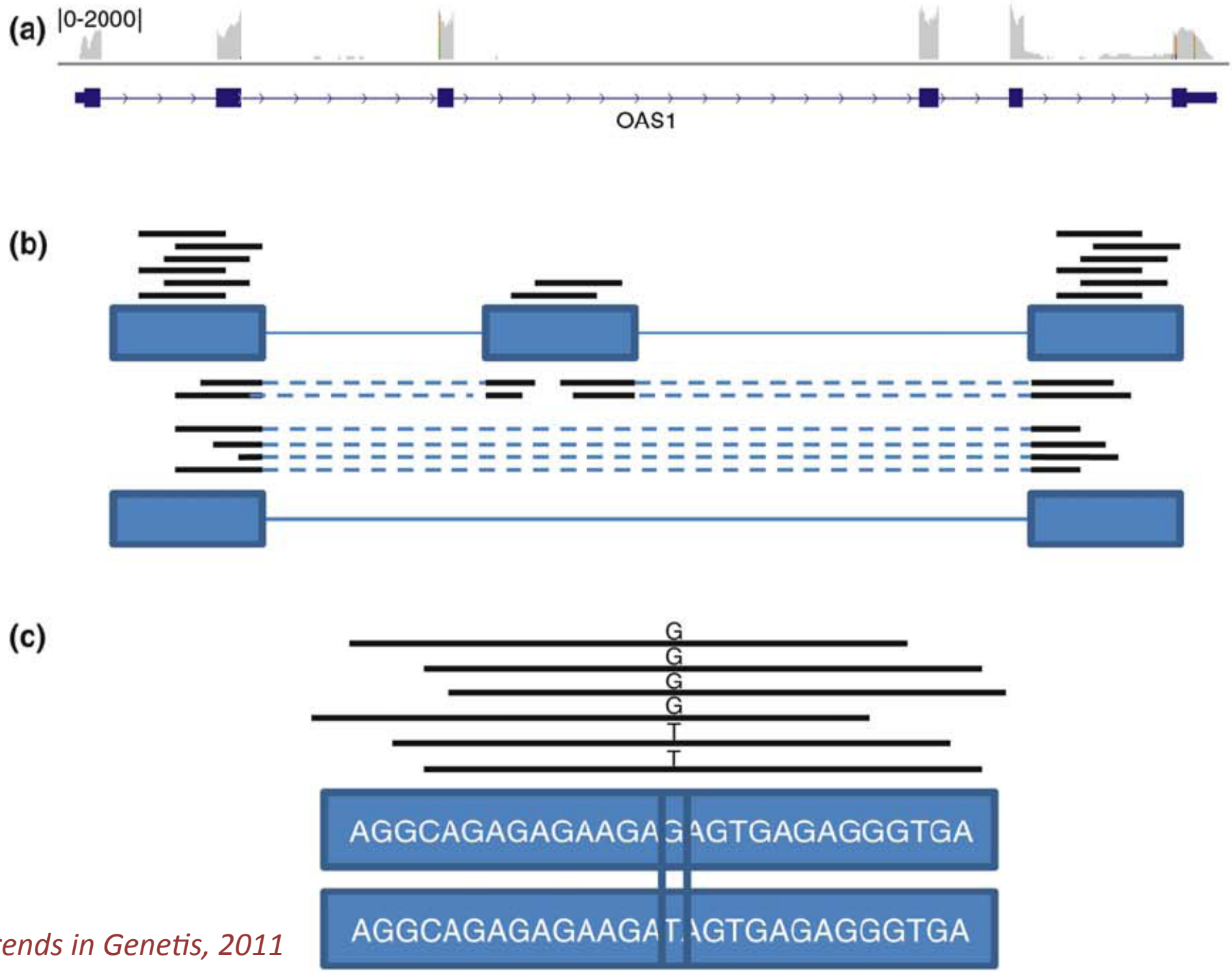
- You get reads
- Data are counts, number of sequences assigned to genes
- Fundamentally discrete, digital
- Other counts data
  - Ad clicks
  - Crime rate
  - Number of thunderstorms
  - ChIP-Seq
  - Mass-spec data
  - shRNA screening

RNAseq workflow

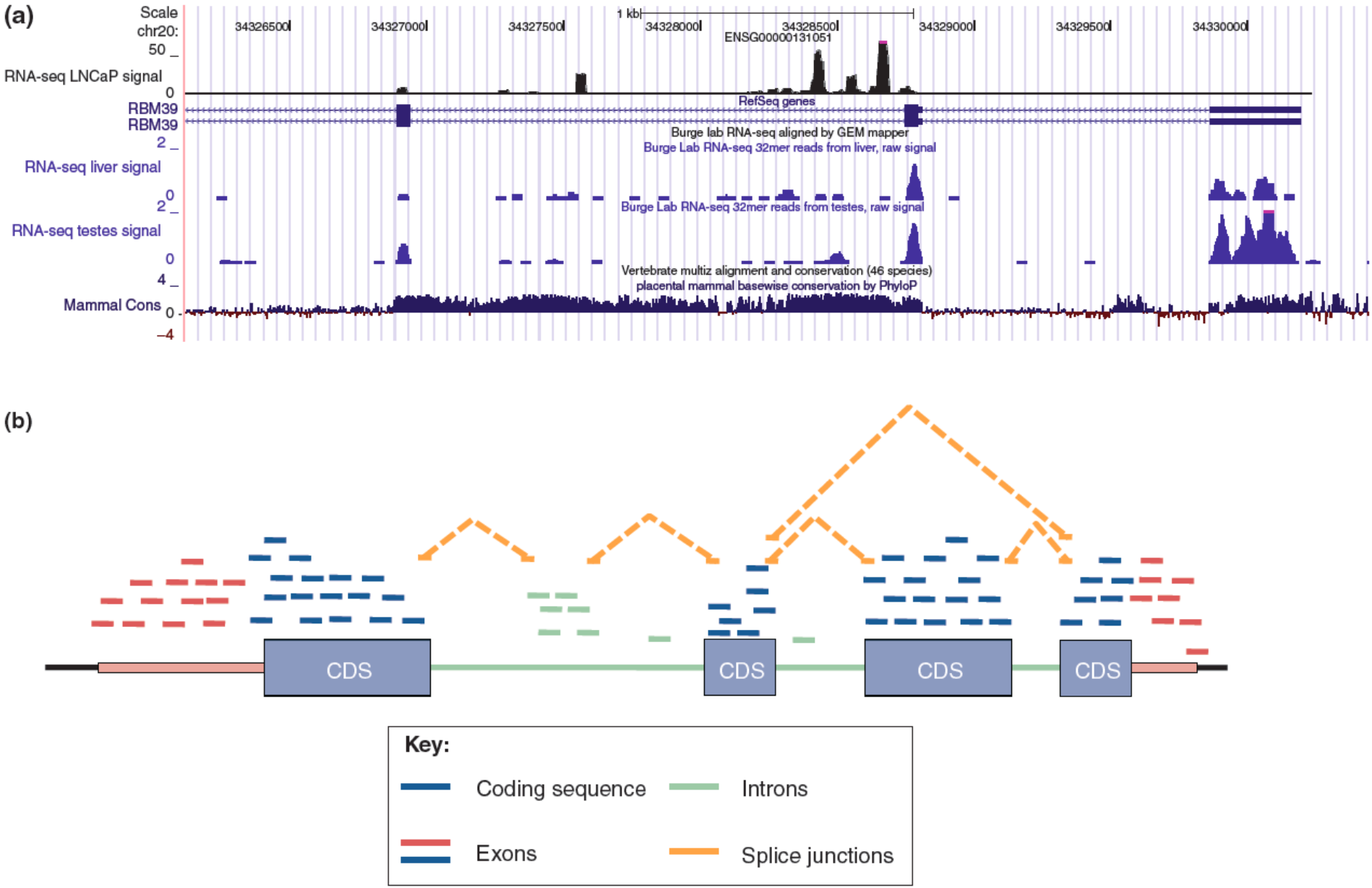
- RNA
- Fragmentation
- Library preparation
- Amplification
- Sequencing
- Informatics
- Statistical processing



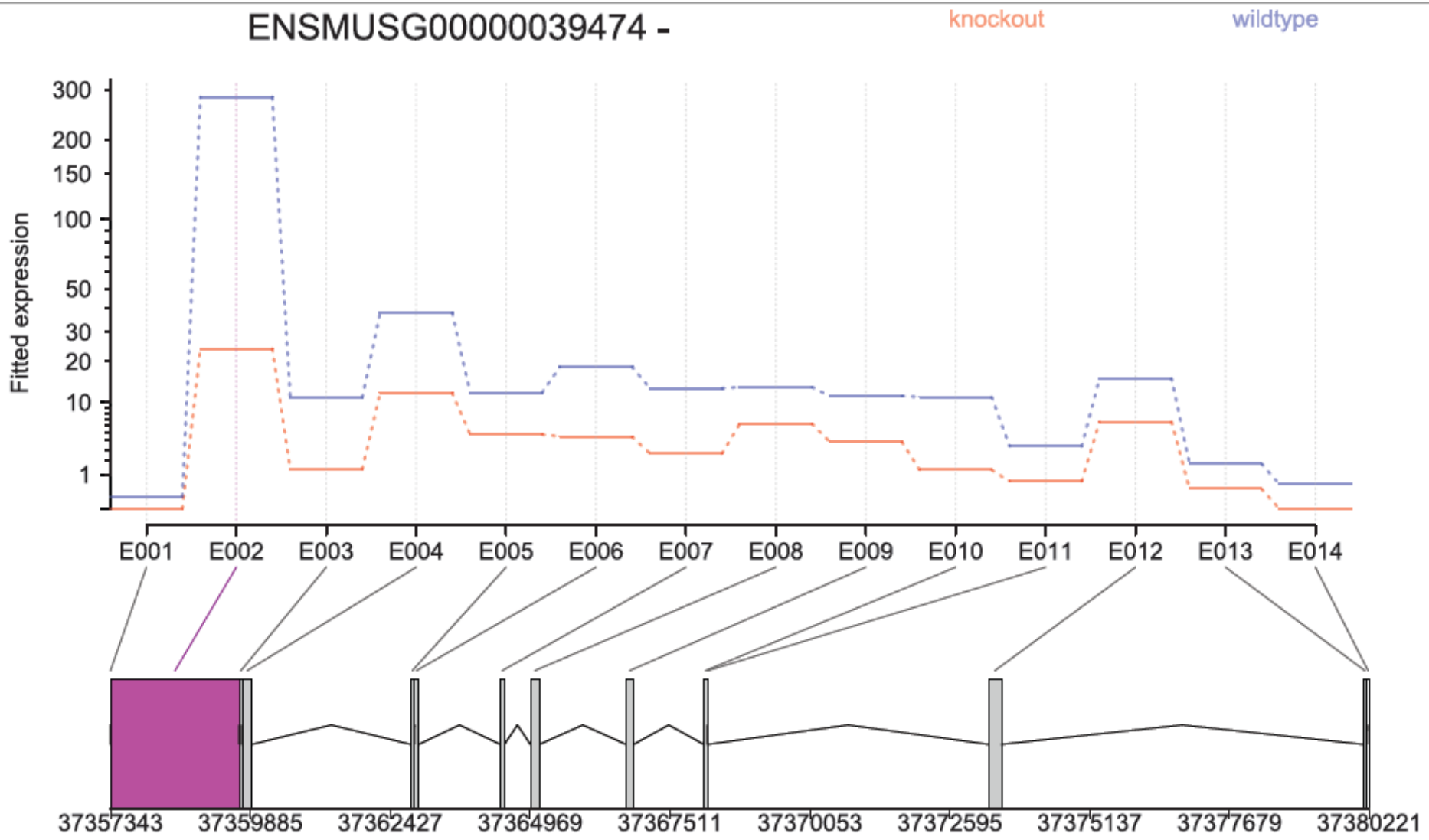
Genome Biology, 2010, 11:220



Trends in Genetis, 2011



Genome Biology, 2010, 11:220


ENSMUSG00000039474 -
knockout
wildtype
Fitted expression
300
200
150
100
50
30
20
10
1
E001
E002
E003
E004
E005
E006
E007
E008
E009
E010
E011
E012
E013
E014
37357343
37359885
37362427
37364969
37367511
37370053
37372595
37375137
37377679
37380221

RNAseq informatics

- QC
- Mapping/pairing
- Annotation
- Statistical analysis
  - edgeR, Deseq, Deseq2, Bayseq
- Biological annotation

Count table

- From sequence alignment and annotation
- `summarizeOverlaps` in *GenomicRanges* package
- `featureCounts` in Rsubread package
- *easyRNASeq* package
- `htseq-counts` Python software

Count table

```
> head(D)
      wt07 wt08 wt09 wt10 ko15 ko16 ko17 ko18
Xkr4  1271  716 1831 1203 1246  818  808  639
Rp1     79   66  134   97  161   99   57   91
Sox17  181   96  190  147  215   73  169   97
Mrpl15 491  328  712  547  441  327  405  448
Lypla1 402  247  491  485  432  311  217  282
Tcea1  542  342  639  441  362  331  323  362
```

edgeR

- Robinson, MD, McCarthy, DJ, Smyth, GK (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 26, 139–140.
- Pairwise comparison
- Uses negative binomial model fitting
- Quantile-adjusted conditional maximum likelihood (qCML) method for dispersion estimation
- Differential expression (DE) is determined using exact test

edgeR

- Multiple factors
- Generalized linear models (GLMs), extension of classical linear model
- Cox-Reid profile-adjusted likelihood (CR) method for dispersion estimation
- DE detection with GLM likelihood test

edgeR

```
library(edgeR)
library(Hmisc)
raw <- read.delim("http://cost-sysgenet.iop.kcl.ac.uk/3wed/session5/countswfship.txt",header=TRUE, sep = "\t")
head(raw)
describe(raw) #how much uniqe Symbols?
raw[duplicated(raw[,1]),] #this list could be long!
ureads<-raw[!duplicated(raw[,1]),]
describe(ureads) #all symbols have to be unique
names(raw)
d <- ureads[, 2:9] #removes first column, 2-8 are stored
head(ureads)
head(d)
rownames(d) <- ureads[, 1]
head(d) #correct format
describe(d)
write.table(ureads, file="filecounts.txt", sep="\t")
```

edgeR

- `D <- read.delim("filecounts.txt", row.names = "Symbol")`
- `group <- factor(c( "wt", "wt", "wt", "wt", "ko", "ko","ko","ko"))`
- `d <- DGEList(counts=D, group = group, lib.size = colSums(D))`
- `d <- calcNormFactors(d)`
- `d <- estimateCommonDisp(d)`
- `d <- estimateTagwiseDisp(d)`
- `de.com<-exactTest(d, pair=c("wt", "ko"))`
- `topTags(de.com, n = 30)`

glm functionality

- `design <- model.matrix(~group)`
- `d <- estimateGLMCommonDisp(d, design)`
- `d <- estimateGLMTrendedDisp(d, design)`
- `d <- estimateGLMTagwiseDisp(d, design)`
- `fit <-glmFit(y, design)`
- `lrt <- glmLRT(fit, coef=2)`
- `topTags(lrt)`

Quick start

```
x <- read.delim("fileofcounts.txt", row.names = "Symbol”)
group <- factor(c(1,1,2,2))
y <- DGEList(counts=x,group=group)
y <- calcNormFactors(y)
y <- estimateCommonDisp(y)
y <- estimateTagwiseDisp(y)
et <- exactTest(y)
topTags(et)
```

Common dispersion

```
> topTags(de.com, n = 30)
Comparison of groups:  ko-wt
                   logFC    logCPM        PValue           FDR
8430422H06Rik  8.015244  5.387959 1.184613e-123 2.845441e-119
Trpm8          6.620645  6.927891 1.245334e-113 1.495646e-109
Bhlhe41       -6.126986  7.548720 1.191209e-104 9.537617e-101
Olfr979        7.236486  4.419463 5.041463e-101  3.027399e-97
4930523O13Rik  5.883185  7.093512  2.209414e-96  1.061402e-92
C330011F03     6.035645  5.752135  3.465725e-96  1.387445e-92
Pm20d2         5.561375 10.486827  1.769165e-94  6.070762e-91
Vmn1r171      -5.646375  6.765767  1.082993e-91  3.251686e-88
Olfr430        6.004094  4.155623  1.782413e-83  4.757063e-80
Srpx2          5.150773  5.855470  8.540829e-79  2.051507e-75
3110009F21Rik -5.351927  4.790677  1.556060e-77  3.397869e-74
E2f7           4.866379  7.933176  5.122344e-77  9.892152e-74
2610028E06Rik  5.107441  5.485900  5.353787e-77  9.892152e-74
Rec8          -5.040380  5.515219  2.308980e-75  3.961550e-72
Camsap3        4.747788  7.998184  1.481414e-74  2.372238e-71
Serpina10      5.312468  4.191038  8.434838e-73  1.266280e-69
Clec1b        -4.705556  6.545123  2.298688e-72  3.247911e-69
Cyb5r2        -4.917459  4.930940  9.245887e-70  1.233812e-66
Sesn2          4.486981 10.141481  1.343283e-68  1.698192e-65
Gzmb           5.353975  3.578984  7.757540e-68  9.316805e-65
Dusp27         4.733449  5.277616  8.329278e-68  9.527108e-65
Sprr2f         4.981788  3.974004  5.292900e-66  5.778884e-63
```

Tagwise dispersion

```
> topTags(de.com, n = 30)
Comparison of groups:  ko-wt
                  logFC    logCPM        PValue           FDR
Trpm8          6.623517  6.927891 5.996474e-203 1.440353e-198
Camsap3        4.748797  7.998184 3.082161e-156 3.701675e-152
Cyb5r2        -4.919149  4.930940  1.261147e-84  1.009758e-80
Zfyve27       -2.540399  8.174247  1.785921e-61  1.072446e-57
Gm16197        3.831465  4.812604  7.140771e-60  3.430426e-56
Rec8          -5.043743  5.515219  1.082678e-59  4.334321e-56
Srpx2          5.144765  5.855470  2.530585e-57  8.683523e-54
Gpr179         3.075715  5.077707  3.130638e-57  9.399741e-54
Ccdc120        3.595592  6.619875  1.911575e-55  5.101782e-52
C330011F03     6.029116  5.752135  5.018891e-48  1.205538e-44
Olfr979        7.224089  4.419463  2.766252e-45  6.040488e-42
2610028E06Rik  5.105053  5.485900  5.894676e-45  1.179918e-41
Tktl2          4.696103  3.094668  9.778474e-42  1.806761e-38
Bhlhe41       -6.128780  7.548720  1.124925e-40  1.930051e-37
Col5a2         3.579486  6.161568  3.079329e-37  4.931033e-34
Lyn           -3.629375  6.866210  8.036579e-37  1.206491e-33
Gcnt7          4.152053  3.759391  2.189751e-34  3.093990e-31
Sprr2f         4.971388  3.974004  1.245822e-33  1.662480e-30
Zc3h13         2.901991  7.144156  3.237456e-33  4.092826e-30
Ccl28          2.754599  4.199223  3.974586e-33  4.773478e-30
Wfs1          -2.840692  5.575801  7.109956e-33  8.132435e-30
Vmn1r171      -5.640312  6.765767  1.282055e-32  1.399771e-29
Cox8c          4.075096  2.027039  2.443657e-32  2.552027e-29
```

DESeq

- Extends the glm model used in *edgeR*
- Negative binomial distribution
- Slight differences in variance estimation
- Offers variance stabilizing transformation
- `nbinomTest()`
- `fitNbinomGLMs()`

DEseq2

- Further development of DESeq package
- Data-driven prior distributions for $\log_2$ FCs (moderates genes with low counts)
- glm is used more widely
- Standard DE analysis is wrapped into single function `DESeq()`
- Wald test for DE
- Several options for QC plots

Conclusions

- edgeR and DESeq both use very similar assumptions and use glm
- Benjamini-Hochberg adjustment for false discovery rate
- Differences are quite minor
- edgeR is similar to *limma*
- DESeq offers more graphical outputs

THANK YOU FOR ATTENTION