

RNA-Seq analysis mini-project

Rogelio Castro

Section 1:

```
library(DESeq2)

metaFile <- "GSE37704_metadata.csv"
countFile <- "GSE37704_featurecounts.csv"

colData = read.csv(metaFile, row.names = 1)
head(colData)
```

```
              condition
SRR493366 control_sirna
SRR493367 control_sirna
SRR493368 control_sirna
SRR493369      hoxa1_kd
SRR493370      hoxa1_kd
SRR493371      hoxa1_kd
```

```
countData = read.csv(countFile, row.names=1)
head(countData)
```

	length	SRR493366	SRR493367	SRR493368	SRR493369	SRR493370
ENSG00000186092	918	0	0	0	0	0
ENSG00000279928	718	0	0	0	0	0
ENSG00000279457	1982	23	28	29	29	28
ENSG00000278566	939	0	0	0	0	0
ENSG00000273547	939	0	0	0	0	0
ENSG00000187634	3214	124	123	205	207	212

	SRR493371
ENSG00000186092	0
ENSG00000279928	0
ENSG00000279457	46
ENSG00000278566	0
ENSG00000273547	0
ENSG00000187634	258

Q1. Complete the code below to remove the troublesome first column from countData

```
# Note we need to remove the odd first $length col
countData <- as.matrix(countData[,-1])
head(countData)
```

	SRR493366	SRR493367	SRR493368	SRR493369	SRR493370	SRR493371
ENSG00000186092	0	0	0	0	0	0
ENSG00000279928	0	0	0	0	0	0
ENSG00000279457	23	28	29	29	28	46
ENSG00000278566	0	0	0	0	0	0
ENSG00000273547	0	0	0	0	0	0
ENSG00000187634	124	123	205	207	212	258

Q2. Complete the code below to filter countData to exclude genes (i.e. rows) where we have 0 read count across all samples (i.e. columns). Tip: What will rowSums() of countData return and how could you use it in this context?

```
zerocounts <- rowSums(countData) == 0
head(zerocounts)
```

ENSG00000186092	ENSG00000279928	ENSG00000279457	ENSG00000278566	ENSG00000273547
TRUE	TRUE	FALSE	TRUE	TRUE
ENSG00000187634				
FALSE				

```
newcounts <- countData[!zerocounts, ]
head(newcounts)
```

	SRR493366	SRR493367	SRR493368	SRR493369	SRR493370	SRR493371
ENSG00000279457	23	28	29	29	28	46

ENSG00000187634	124	123	205	207	212	258
ENSG00000188976	1637	1831	2383	1226	1326	1504
ENSG00000187961	120	153	180	236	255	357
ENSG00000187583	24	48	65	44	48	64
ENSG00000187642	4	9	16	14	16	16

```
nrow(newcounts)
```

```
[1] 15975
```

3. Setup and run DESeq:

Already loaded DESeq2 at the beginning of the project

```
dds <- DESeqDataSetFromMatrix(countData = newcounts,
                              colData = colData,
                              design = ~condition)
```

Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in design formula are characters, converting to factors

```
dds <- DESeq(dds)
```

estimating size factors

estimating dispersions

gene-wise dispersion estimates

mean-dispersion relationship

final dispersion estimates

fitting model and testing

```
res <- results(dds)
```

```
res = results(dds, contrast=c("condition", "hoxa1_kd", "control_sirna"))
res
```

log2 fold change (MLE): condition hoxa1_kd vs control_sirna

Wald test p-value: condition hoxa1 kd vs control sirna

DataFrame with 15975 rows and 6 columns

	baseMean	log2FoldChange	lfcSE	stat	pvalue
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
ENSG00000279457	29.9136	0.1792571	0.3248216	0.551863	5.81042e-01
ENSG00000187634	183.2296	0.4264571	0.1402658	3.040350	2.36304e-03
ENSG00000188976	1651.1881	-0.6927205	0.0548465	-12.630158	1.43989e-36
ENSG00000187961	209.6379	0.7297556	0.1318599	5.534326	3.12428e-08
ENSG00000187583	47.2551	0.0405765	0.2718928	0.149237	8.81366e-01
...
ENSG00000273748	35.30265	0.674387	0.303666	2.220817	2.63633e-02
ENSG00000278817	2.42302	-0.388988	1.130394	-0.344117	7.30758e-01
ENSG00000278384	1.10180	0.332991	1.660261	0.200565	8.41039e-01
ENSG00000276345	73.64496	-0.356181	0.207716	-1.714752	8.63908e-02
ENSG00000271254	181.59590	-0.609667	0.141320	-4.314071	1.60276e-05
	padj				
	<numeric>				
ENSG00000279457	6.86555e-01				
ENSG00000187634	5.15718e-03				
ENSG00000188976	1.76549e-35				
ENSG00000187961	1.13413e-07				
ENSG00000187583	9.19031e-01				
...	...				
ENSG00000273748	4.79091e-02				
ENSG00000278817	8.09772e-01				
ENSG00000278384	8.92654e-01				
ENSG00000276345	1.39762e-01				
ENSG00000271254	4.53648e-05				

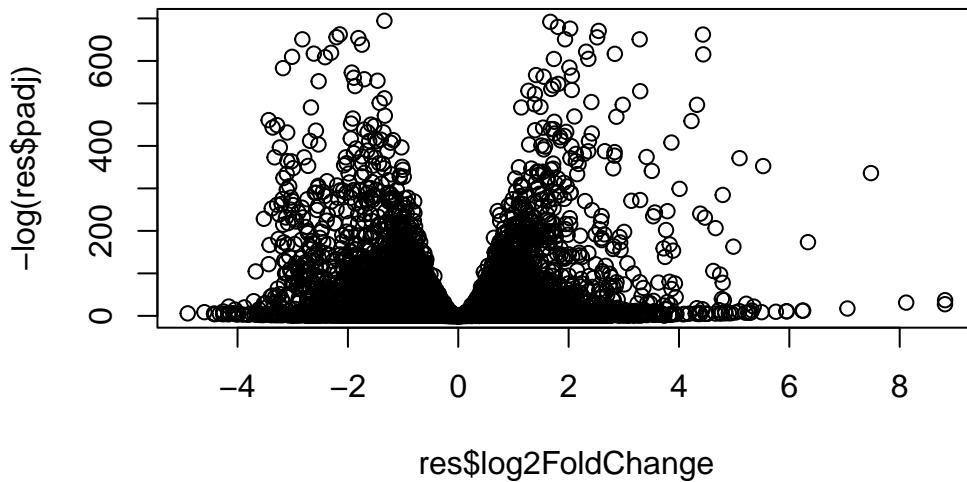
Q. Call the `summary()` function on your results to get a sense of how many genes are up or down-regulated at the default 0.1 p-value cutoff.

```
summary(res)
```

out of 15975 with nonzero total read count
adjusted p-value < 0.1

```
LFC > 0 (up)      : 4349, 27%
LFC < 0 (down)    : 4396, 28%
outliers [1]      : 0, 0%
low counts [2]    : 1237, 7.7%
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

```
plot( res$log2FoldChange, -log(res$padj) )
```



4. Annotate results:

I need to add annotation to my results including gene symbols and ENTREZ IDs etc. For this I will use the **AnnotationDbi** package.

```
library(AnnotationDbi)
library("org.Hs.eg.db")
```

```
columns(org.Hs.eg.db)
```

```
[1] "ACCNUM"      "ALIAS"      "ENSEMBL"    "ENSEMBLPROT" "ENSEMBLTRANS"
[6] "ENTREZID"    "ENZYME"     "EVIDENCE"   "EVIDENCEALL" "GENENAME"
[11] "GENETYPE"    "GO"         "GOALL"      "IPI"          "MAP"
[16] "OMIM"        "ONTOLOGY"   "ONTOLOGYALL" "PATH"         "PFAM"
[21] "PMID"        "PROSITE"    "REFSEQ"     "SYMBOL"       "UCSCKG"
[26] "UNIPROT"
```

```
res$symbol = mapIds(org.Hs.eg.db,
  keys=rownames(res),
  keytype="ENSEMBL",
  column="SYMBOL",
  multiVals="first")
```

'select()' returned 1:many mapping between keys and columns

```
res$entrez = mapIds(org.Hs.eg.db,
  keys=rownames(res),
  keytype="ENSEMBL",
  column="ENTREZID",
  multiVals="first")
```

'select()' returned 1:many mapping between keys and columns

```
head(res)
```

log2 fold change (MLE): condition hoxa1_kd vs control_sirna

Wald test p-value: condition hoxa1 kd vs control sirna

DataFrame with 6 rows and 8 columns

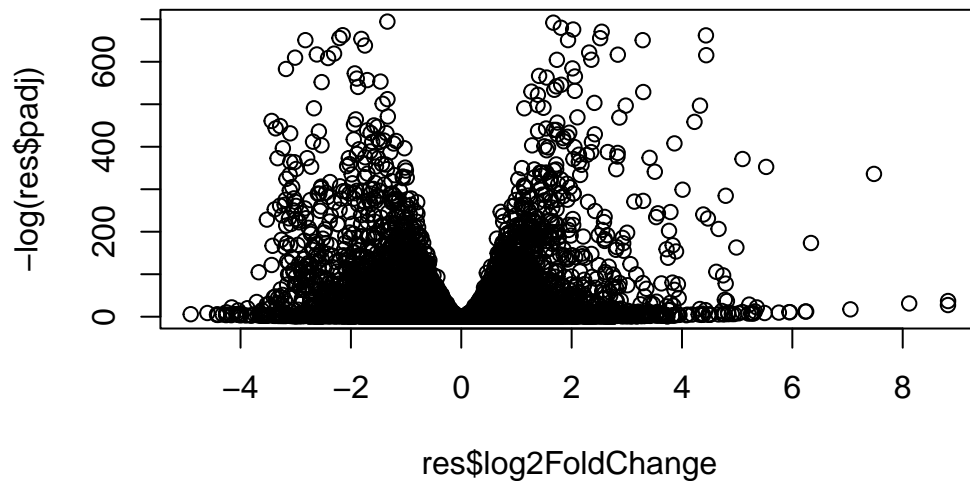
	baseMean	log2FoldChange	lfcSE	stat	pvalue
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
ENSG00000279457	29.9136	0.1792571	0.3248216	0.551863	5.81042e-01
ENSG00000187634	183.2296	0.4264571	0.1402658	3.040350	2.36304e-03
ENSG00000188976	1651.1881	-0.6927205	0.0548465	-12.630158	1.43989e-36
ENSG00000187961	209.6379	0.7297556	0.1318599	5.534326	3.12428e-08
ENSG00000187583	47.2551	0.0405765	0.2718928	0.149237	8.81366e-01

ENSG00000187642	11.9798	0.5428105	0.5215599	1.040744	2.97994e-01
	padj	symbol	entrez		
	<numeric>	<character>	<character>		
ENSG00000279457	6.86555e-01	NA	NA		
ENSG00000187634	5.15718e-03	SAMD11	148398		
ENSG00000188976	1.76549e-35	NOC2L	26155		
ENSG00000187961	1.13413e-07	KLHL17	339451		
ENSG00000187583	9.19031e-01	PLEKHN1	84069		
ENSG00000187642	4.03379e-01	PERM1	84808		

Q. Finally for this section let's reorder these results by adjusted p-value and save them to a CSV file in your current project directory.

```
res = res[order(res$pvalue),]
write.csv(res, file = "deseq_results.csv")
```

```
plot(res$log2FoldChange, -log(res$padj))
```



#Pathway Analysis:

```

library(pathview)
library(gage)
library(gageData)

data(kegg.sets.hs)
data(sigmet.idx.hs)

# Focus on signaling and metabolic pathways only
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]

foldchanges = res$log2FoldChange
names(foldchanges) = res$entrez
head(foldchanges)

```

```

      1266      54855      1465      51232      2034      2317
-2.422719  3.201955 -2.313738 -2.059631 -1.888019 -1.649792

```

Run gage:

```

# Get the results:
keggres = gage(foldchanges, gsets = kegg.sets.hs)

head(keggres$less)

```

	p.geomean	stat.mean	p.val
hsa04110 Cell cycle	8.995727e-06	-4.378644	8.995727e-06
hsa03030 DNA replication	9.424076e-05	-3.951803	9.424076e-05
hsa03013 RNA transport	1.375901e-03	-3.028500	1.375901e-03
hsa03440 Homologous recombination	3.066756e-03	-2.852899	3.066756e-03
hsa04114 Oocyte meiosis	3.784520e-03	-2.698128	3.784520e-03
hsa00010 Glycolysis / Gluconeogenesis	8.961413e-03	-2.405398	8.961413e-03

	q.val	set.size	exp1
hsa04110 Cell cycle	0.001448312	121	8.995727e-06
hsa03030 DNA replication	0.007586381	36	9.424076e-05
hsa03013 RNA transport	0.073840037	144	1.375901e-03
hsa03440 Homologous recombination	0.121861535	28	3.066756e-03
hsa04114 Oocyte meiosis	0.121861535	102	3.784520e-03
hsa00010 Glycolysis / Gluconeogenesis	0.212222694	53	8.961413e-03


```
pathview(gene.data = foldchanges, pathway.id = "hsa04110")
```

Info: Working in directory /Users/rogeliocastro/Documents/Classes UCSD/BIMM 143/RNA-Seq_Min

The diagram illustrates the Cell Cycle pathway, showing the progression from G1 to S to G2 to M phases. Key components and interactions include:

- G1 Phase:** Growth factor withdrawal leads to GSK3β activation. Growth factor stimulation leads to MAPK signaling pathway. TGFβ signaling involves Smad2,3 and Smad4. Cyclins (CycD, CycE, CycA) and CDKs (CDK4, CDK2, CDK1) are regulated. Checkpoint proteins like p16, p15, p18, p19, p27, p57, p21, and p21 are involved.
- S Phase:** DNA replication and DNA biosynthesis. Key proteins include PCNA, Cdc25A, Cdc25B, Cdc25C, Cdc25D, Cdc25E, Cdc25F, Cdc25G, Cdc25H, Cdc25I, Cdc25J, Cdc25K, Cdc25L, Cdc25M, Cdc25N, Cdc25O, Cdc25P, Cdc25Q, Cdc25R, Cdc25S, Cdc25T, Cdc25U, Cdc25V, Cdc25W, Cdc25X, Cdc25Y, Cdc25Z, Cdc25AA, Cdc25AB, Cdc25AC, Cdc25AD, Cdc25AE, Cdc25AF, Cdc25AG, Cdc25AH, Cdc25AI, Cdc25AJ, Cdc25AK, Cdc25AL, Cdc25AM, Cdc25AN, Cdc25AO, Cdc25AP, Cdc25AQ, Cdc25AR, Cdc25AS, Cdc25AT, Cdc25AU, Cdc25AV, Cdc25AW, Cdc25AX, Cdc25AY, Cdc25AZ, Cdc25BA, Cdc25BB, Cdc25BC, Cdc25BD, Cdc25BE, Cdc25BF, Cdc25BG, Cdc25BH, Cdc25BI, Cdc25BJ, Cdc25BK, Cdc25BL, Cdc25BM, Cdc25BN, Cdc25BO, Cdc25BP, Cdc25BQ, Cdc25BR, Cdc25BS, Cdc25BT, Cdc25BU, Cdc25BV, Cdc25BW, Cdc25BX, Cdc25BY, Cdc25BZ, Cdc25CA, Cdc25CB, Cdc25CC, Cdc25CD, Cdc25CE, Cdc25CF, Cdc25CG, Cdc25CH, Cdc25CI, Cdc25CJ, Cdc25CK, Cdc25CL, Cdc25CM, Cdc25CN, Cdc25CO, Cdc25CP, Cdc25CQ, Cdc25CR, Cdc25CS, Cdc25CT, Cdc25CU, Cdc25CV, Cdc25CW, Cdc25CX, Cdc25CY, Cdc25CZ, Cdc25DA, Cdc25DB, Cdc25DC, Cdc25DD, Cdc25DE, Cdc25DF, Cdc25DG, Cdc25DH, Cdc25DI, Cdc25DJ, Cdc25DK, Cdc25DL, Cdc25DM, Cdc25DN, Cdc25DO, Cdc25DP, Cdc25DQ, Cdc25DR, Cdc25DS, Cdc25DT, Cdc25DU, Cdc25DV, Cdc25DW, Cdc25DX, Cdc25DY, Cdc25DZ, Cdc25EA, Cdc25EB, Cdc25EC, Cdc25ED, Cdc25EE, Cdc25EF, Cdc25EG, Cdc25EH, Cdc25EI, Cdc25EJ, Cdc25EK, Cdc25EL, Cdc25EM, Cdc25EN, Cdc25EO, Cdc25EP, Cdc25EQ, Cdc25ER, Cdc25ES, Cdc25ET, Cdc25EU, Cdc25EV, Cdc25EW, Cdc25EX, Cdc25EY, Cdc25EZ, Cdc25FA, Cdc25FB, Cdc25FC, Cdc25FD, Cdc25FE, Cdc25FF, Cdc25FG, Cdc25FH, Cdc25FI, Cdc25FJ, Cdc25FK, Cdc25FL, Cdc25FM, Cdc25FN, Cdc25FO, Cdc25FP, Cdc25FQ, Cdc25FR, Cdc25FS, Cdc25FT, Cdc25FU, Cdc25FV, Cdc25FW, Cdc25FX, Cdc25FY, Cdc25FZ, Cdc25GA, Cdc25GB, Cdc25GC, Cdc25GD, Cdc25GE, Cdc25GF, Cdc25GG, Cdc25GH, Cdc25GI, Cdc25GJ, Cdc25GK, Cdc25GL, Cdc25GM, Cdc25GN, Cdc25GO, Cdc25GP, Cdc25GQ, Cdc25GR, Cdc25GS, Cdc25GT, Cdc25GU, Cdc25GV, Cdc25GW, Cdc25GX, Cdc25GY, Cdc25GZ, Cdc25HA, Cdc25HB, Cdc25HC, Cdc25HD, Cdc25HE, Cdc25HF, Cdc25HG, Cdc25HH, Cdc25HI, Cdc25HJ, Cdc25HK, Cdc25HL, Cdc25HM, Cdc25HN, Cdc25HO, Cdc25HP, Cdc25HQ, Cdc25HR, Cdc25HS, Cdc25HT, Cdc25HU, Cdc25HV, Cdc25HW, Cdc25HX, Cdc25HY, Cdc25HZ, Cdc25IA, Cdc25IB, Cdc25IC, Cdc25ID, Cdc25IE, Cdc25IF, Cdc25IG, Cdc25IH, Cdc25II, Cdc25IJ, Cdc25IK, Cdc25IL, Cdc25IM, Cdc25IN, Cdc25IO, Cdc25IP, Cdc25IQ, Cdc25IR, Cdc25IS, Cdc25IT, Cdc25IU, Cdc25IV, Cdc25IW, Cdc25IX, Cdc25IY, Cdc25IZ, Cdc25JA, Cdc25JB, Cdc25JC, Cdc25JD, Cdc25JE, Cdc25JF, Cdc25JG, Cdc25JH, Cdc25JI, Cdc25JJ, Cdc25JK, Cdc25JL, Cdc25JM, Cdc25JN, Cdc25JO, Cdc25JP, Cdc25JQ, Cdc25JR, Cdc25JS, Cdc25JT, Cdc25JU, Cdc25JV, Cdc25JW, Cdc25JX, Cdc25JY, Cdc25JZ, Cdc25KA, Cdc25KB, Cdc25KC, Cdc25KD, Cdc25KE, Cdc25KF, Cdc25KG, Cdc25KH, Cdc25KI, Cdc25KJ, Cdc25KK, Cdc25KL, Cdc25KM, Cdc25KN, Cdc25KO, Cdc25KP, Cdc25KQ, Cdc25KR, Cdc25KS, Cdc25KT, Cdc25KU, Cdc25KV, Cdc25KW, Cdc25KX, Cdc25KY, Cdc25KZ, Cdc25LA, Cdc25LB, Cdc25LC, Cdc25LD, Cdc25LE, Cdc25LF, Cdc25LG, Cdc25LH, Cdc25LI, Cdc25LJ, Cdc25LK, Cdc25LL, Cdc25LM, Cdc25LN, Cdc25LO, Cdc25LP, Cdc25LQ, Cdc25LR, Cdc25LS, Cdc25LT, Cdc25LU, Cdc25LV, Cdc25LW, Cdc25LX, Cdc25LY, Cdc25LZ, Cdc25MA, Cdc25MB, Cdc25MC, Cdc25MD, Cdc25ME, Cdc25MF, Cdc25MG, Cdc25MH, Cdc25MI, Cdc25MJ, Cdc25MK, Cdc25ML, Cdc25MM, Cdc25MN, Cdc25MO, Cdc25MP, Cdc25MQ, Cdc25MR, Cdc25MS, Cdc25MT, Cdc25MU, Cdc25MV, Cdc25MW, Cdc25MX, Cdc25MY, Cdc25MZ, Cdc25NA, Cdc25NB, Cdc25NC, Cdc25ND, Cdc25NE, Cdc25NF, Cdc25NG, Cdc25NH, Cdc25NI, Cdc25NJ, Cdc25NK, Cdc25NL, Cdc25NM, Cdc25NN, Cdc25NO, Cdc25NP, Cdc25NQ, Cdc25NR, Cdc25NS, Cdc25NT, Cdc25NU, Cdc25NV, Cdc25NW, Cdc25NX, Cdc25NY, Cdc25NZ, Cdc25OA, Cdc25OB, Cdc25OC, Cdc25OD, Cdc25OE, Cdc25OF, Cdc25OG, Cdc25OH, Cdc25OI, Cdc25OJ, Cdc25OK, Cdc25OL, Cdc25OM, Cdc25ON, Cdc25OO, Cdc25OP, Cdc25OQ, Cdc25OR, Cdc25OS, Cdc25OT, Cdc25OU, Cdc25OV, Cdc25OW, Cdc25OX, Cdc25OY, Cdc25OZ, Cdc25PA, Cdc25PB, Cdc25PC, Cdc25PD, Cdc25PE, Cdc25PF, Cdc25PG, Cdc25PH, Cdc25PI, Cdc25PJ, Cdc25PK, Cdc25PL, Cdc25PM, Cdc25PN, Cdc25PO, Cdc25PP, Cdc25PQ, Cdc25PR, Cdc25PS, Cdc25PT, Cdc25PU, Cdc25PV, Cdc25PW, Cdc25PX, Cdc25PY, Cdc25PZ, Cdc25QA, Cdc25QB, Cdc25QC, Cdc25QD, Cdc25QE, Cdc25QF, Cdc25QG, Cdc25QH, Cdc25QI, Cdc25QJ, Cdc25QK, Cdc25QL, Cdc25QM, Cdc25QN, Cdc25QO, Cdc25QP, Cdc25QQ, Cdc25QR, Cdc25QS, Cdc25QT, Cdc25QU, Cdc25QV, Cdc25QW, Cdc25QX, Cdc25QY, Cdc25QZ, Cdc25RA, Cdc25RB, Cdc25RC, Cdc25RD, Cdc25RE, Cdc25RF, Cdc25RG, Cdc25RH, Cdc25RI, Cdc25RJ, Cdc25RK, Cdc25RL, Cdc25RM, Cdc25RN, Cdc25RO, Cdc25RP, Cdc25RQ, Cdc25RR, Cdc25RS, Cdc25RT, Cdc25RU, Cdc25RV, Cdc25RW, Cdc25RX, Cdc25RY, Cdc25RZ, Cdc25SA, Cdc25SB, Cdc25SC, Cdc25SD, Cdc25SE, Cdc25SF, Cdc25SG, Cdc25SH, Cdc25SI, Cdc25SJ, Cdc25SK, Cdc25SL, Cdc25SM, Cdc25SN, Cdc25SO, Cdc25SP, Cdc25SQ, Cdc25SR, Cdc25SS, Cdc25ST, Cdc25SU, Cdc25SV, Cdc25SW, Cdc25SX, Cdc25SY, Cdc25SZ, Cdc25TA, Cdc25TB, Cdc25TC, Cdc25TD, Cdc25TE, Cdc25TF, Cdc25TG, Cdc25TH, Cdc25TI, Cdc25TJ, Cdc25TK, Cdc25TL, Cdc25TM, Cdc25TN, Cdc25TO, Cdc25TP, Cdc25TQ, Cdc25TR, Cdc25TS, Cdc25TT, Cdc25TU, Cdc25TV, Cdc25TW, Cdc25TX, Cdc25TY, Cdc25TZ, Cdc25UA, Cdc25UB, Cdc25UC, Cdc25UD, Cdc25UE, Cdc25UF, Cdc25UG, Cdc25UH, Cdc25UI, Cdc25UJ, Cdc25UK, Cdc25UL, Cdc25UM, Cdc25UN, Cdc25UO, Cdc25UP, Cdc25UQ, Cdc25UR, Cdc25US, Cdc25UT, Cdc25UU, Cdc25UV, Cdc25UW, Cdc25UX, Cdc25UY, Cdc25UZ, Cdc25VA, Cdc25VB, Cdc25VC, Cdc25VD, Cdc25VE, Cdc25VF, Cdc25VG, Cdc25VH, Cdc25VI, Cdc25VJ, Cdc25VK, Cdc25VL, Cdc25VM, Cdc25VN, Cdc25VO, Cdc25VP, Cdc25VQ, Cdc25VR, Cdc25VS, Cdc25VT, Cdc25VU, Cdc25VV, Cdc25VW, Cdc25VX, Cdc25VY, Cdc25VZ, Cdc25WA, Cdc25WB, Cdc25WC, Cdc25WD, Cdc25WE, Cdc25WF, Cdc25WG, Cdc25WH, Cdc25WI, Cdc25WJ, Cdc25WK, Cdc25WL, Cdc25WM, Cdc25WN, Cdc25WO, Cdc25WP, Cdc25WQ, Cdc25WR, Cdc25WS, Cdc25WT, Cdc25WU, Cdc25WV, Cdc25WW, Cdc25WX, Cdc25WY, Cdc25WZ, Cdc25XA, Cdc25