Практикум поMicroarrays

17 мая 2021 г.

1 Задание

Необходимо выполнить манипуляции с данными об исследовании экспрессии генов на основании статьи Hyun Goo Woo et al. "Identification of a Cholangiocarcinoma-Like Gene Expression Trait in Hepatocellular Carcinoma"[1].

Авторы работы рассматривают основные виды рака печени у взрослых (HCC - Hepatocellular carcinoma и CC - cholangiocarcinoma). Утверждается, что существует комбинированное заболевание, которое предполагает фенотипическое пересечение между этими опухолями. Авторы статьи применили интегративный онкогеномный подход для клинических и функциональных последствий для комбинированного типа опухолей. Было выполнено исследование, посвещенное экспрессии генов.

В данной работе необходимо провести похожие исследования с рассмотренными в статье данными:

1. Загрузить данные для анализа (Affymetrix) из открытых источников: ArrayExpress и GEO для E-GEOD-15765;
2. Оценить данные, отсеить выборсы, произвести нормализацию;
3. Выполнить поиск дифференциально экспрессирующихся генов;
4. Провести кластеризацию профилей дифференциально экспрессирующихся генов, построить тепловые карты;
5. Проанализировать обогащённость генов метаболическими путями из баз данных GO, KEGG и Reactome;
6. Сделать выводы.

2 Решение

2.1 Загрузка данных

Вначале загрузим необходимые библиотеки:

```
> library(ArrayExpress)
> library(oligo)
> library(limma)
> library(AnnotationDbi)
> library(reactome.db)
> setwd("C:/microarrayslab")
```

Теперь загрузим сами данные.

```
> #geod15765 <- getAE("E-GEOD-15765", type = "full")
> #save(geod15765, file="geod15765.RData")
> load("geod15765.RData")
> aeset <- ae2bioc(mageFiles = geod15765) # A-AFFY-37
```

Далее необходимо посмотреть, какие факторы поставлены в соответствие образцам. Это именно те факторы, которые должны присутствовать в анализе.

```
> colnames1 <- colnames(phenoData(aeset))
> fac <- colnames1[grep("Factor", colnames1)]
> fac
```

```
[1] "Factor.Value..TISSUE."
```

Фактор получился всего один - он отвечает за тип злокачественного новообразования. Теперь посмотрим на данные, которые соответствуют этому фактору.

```
> groups <- phenoData(aeset)[, fac]
> head(pData(groups))
```

	Factor.Value..TISSUE.
GSM395663.CEL	hepatocellular carcinoma
GSM395728.CEL	cholangiocarcinoma
GSM395672.CEL	hepatocellular carcinoma
GSM395723.CEL	combined hepatocellular carcinoma and cholangiocarcinoma
GSM395706.CEL	hepatocellular carcinoma
GSM395717.CEL	hepatocellular carcinoma

2.2 Проверка качества данных

Для проверки качества данных построим модель данных.

```
> dataPLM <- fitProbeLevelModel(aeset)
```

Затем посмотрим на графики контроля качества для массивов Affymetrix: NUSE показывает нормализованные немасштабированные стандартные ошибки, а RLE показывает относительные значения логарифма значений экспрессии. Итак, вначале рассмотрим NUSE:

```
> nuse <- NUSE(dataPLM, type="stats")
> nuse[, 1:5]
```

	GSM395663.CEL	GSM395728.CEL	GSM395672.CEL	GSM395723.CEL	GSM395706.CEL
0%	0.9483260	0.9464485	0.9432030	0.9381795	0.9412987
25%	0.9849854	0.9934357	0.9821402	0.9863062	0.9831642
50%	0.9970127	1.0094214	0.9928456	1.0003685	0.9944144
75%	1.0114944	1.0345462	1.0064939	1.0201490	1.0084584
100%	1.2629980	1.6467113	1.6201418	1.5025273	1.4667223

```
> png("NUSE.png", width = 720)
> par(mar=c(8, 4.1, 4.1, 2.1))
> NUSE(dataPLM, las = 2)
> dev.off()
```

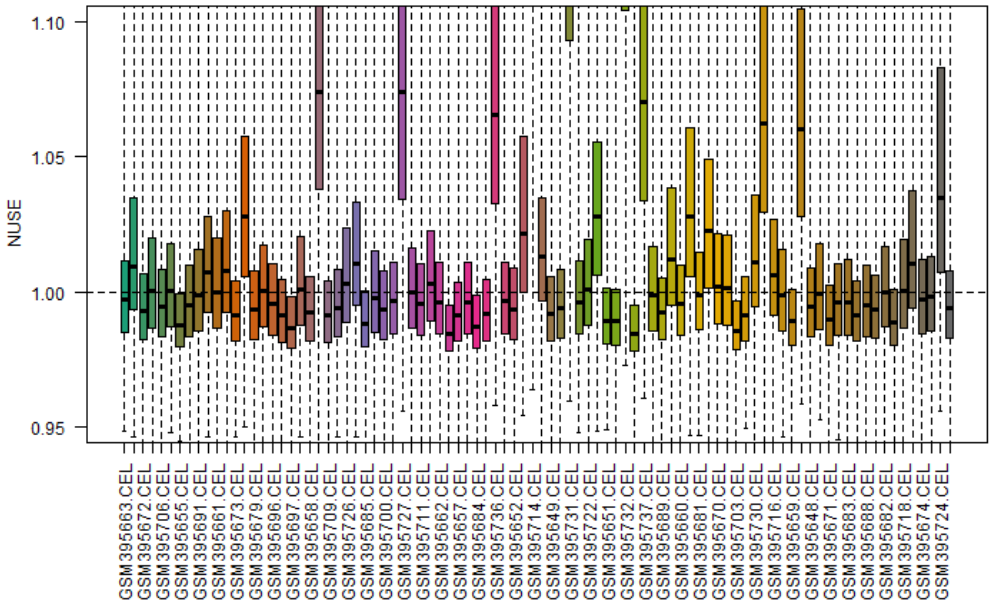


Рис. 1: NUSE-plot

Аналогично рассмотрим RLE:

```
> rle <- RLE(dataPLM, type = "stats")
> rle[, 1:5]
```

	GSM395663.CEL	GSM395728.CEL	GSM395672.CEL	GSM395723.CEL	GSM395706.CEL
0%	-4.49989367	-8.8644396	-6.76866417	-8.31372413	-6.55812908
25%	-0.25549230	-0.2778927	-0.20385685	-0.28274340	-0.27781808
50%	-0.03407394	0.0596426	0.01078127	-0.01937574	-0.05127737
75%	0.28310768	0.4158990	0.22723836	0.32594139	0.26831059
100%	8.72385561	7.9664275	6.90632814	9.28864883	9.44486821

Выведем боксплоты:

```
> png("RLE.png", width = 720)
> par(mar=c(8, 4.1, 4.1, 2.1))
> RLE(dataPLM, las = 2)
> dev.off()
```

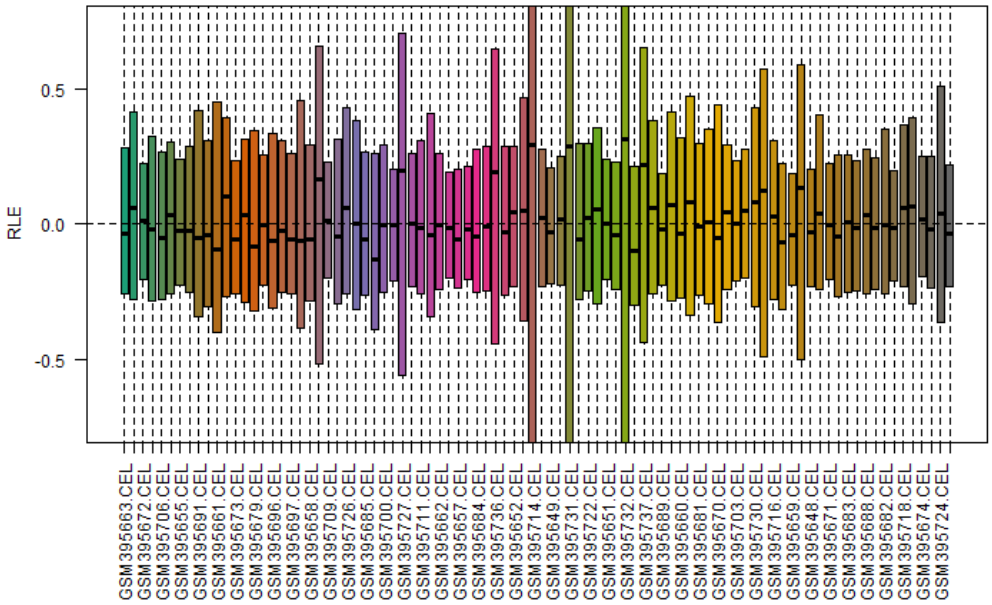


Рис. 2: RLE-plot

По обоим графикам видим выбросы для следующих данных:

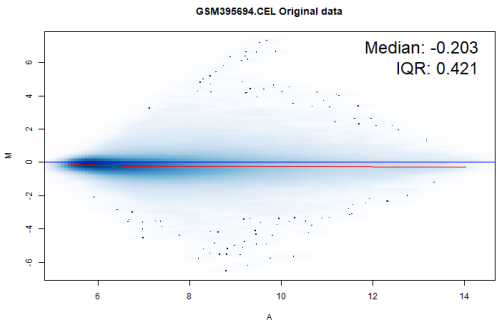
Название	Номер в массиве
GSM395733.CEL	22
GSM395727.CEL	31
GSM395736.CEL	41
GSM395714.CEL	45
GSM395731.CEL	49
GSM395732.CEL	55
GSM395737.CEL	57
GSM395734.CEL	70
GSM395721.CEL	74
GSM395724.CEL	89

Уберем выбросы из дальнейшего рассмотрения.

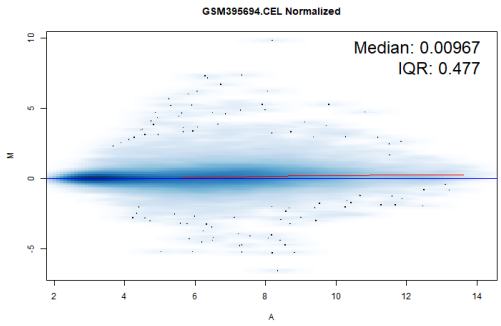
```
> extremes <- c(22, 31, 41, 45, 49, 55, 57, 70, 74, 89)
> new_aeset <- aeset[, -extremes]
> new_groups <- groups[-extremes, ]
```

Теперь отнормируем данные и построим MA-plots для исходных и нормированных данных для случайно выбранных экземпляров.

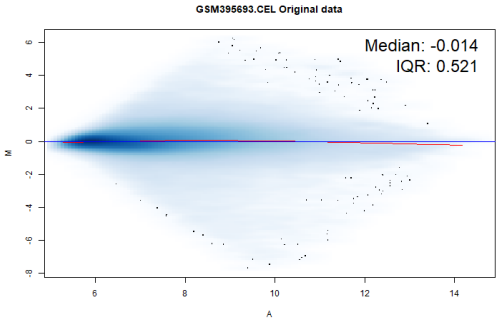
```
> normed_new_aeset <- rma(new_aeset)
> arrays <- c(16, 32, 48, 64, 80)
> for (a in arrays)
+ {
+   png(paste("Original, №", a, ".png", sep=""), width = 720)
+   MAplot(new_aeset, which = a, main = "Original data")
+   dev.off()
+   png(paste("Normed, №", a, ".png", sep=""), width = 720)
+   MAplot(normed_new_aeset, which = a, main = "Normalized")
+   dev.off()
+ }
```



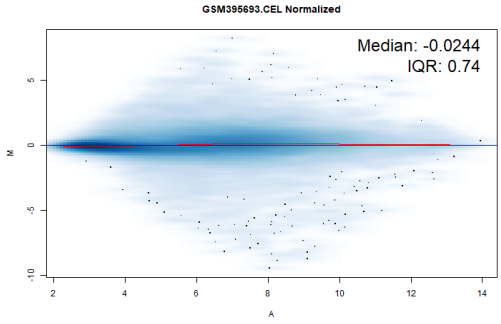
(a) Экземпляр №16, исходные данные



(b) Экземпляр №16, нормированные данные

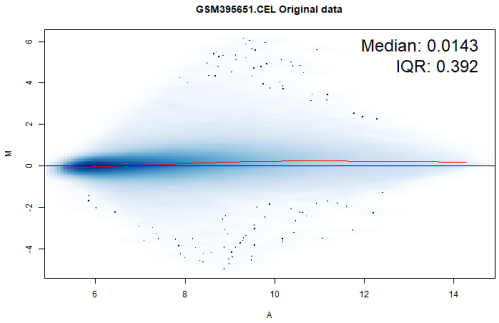


(c) Экземпляр №32, исходные данные

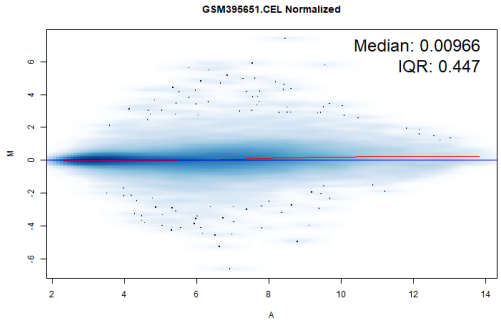


(d) Экземпляр №32, нормированные данные

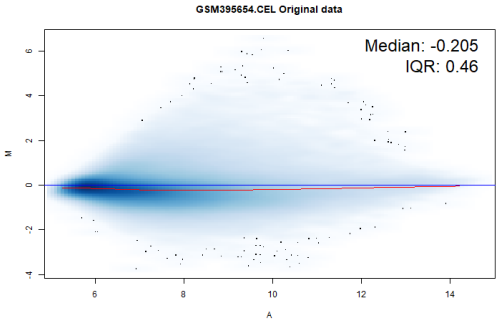
Рис. 3: MA-plots, ч.1



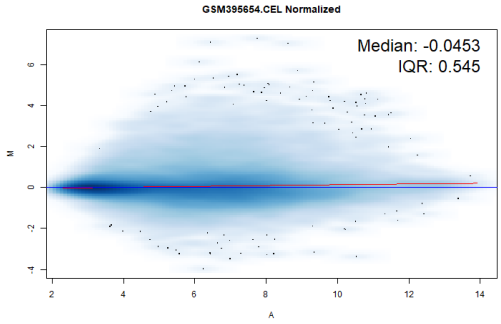
(a) Экземпляр №48, исходные данные



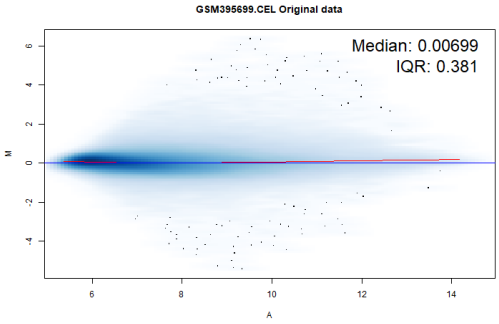
(b) Экземпляр №48, нормированные данные



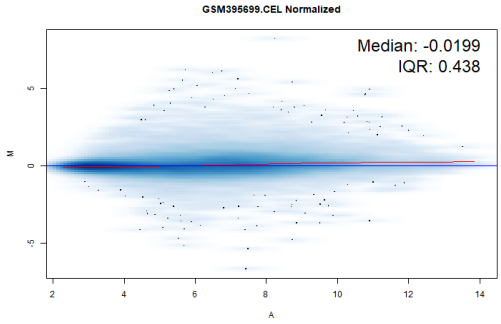
(c) Экземпляр №64, исходные данные



(d) Экземпляр №64, нормированные данные



(e) Экземпляр №80, исходные данные



(f) Экземпляр №80, нормированные данные

Рис. 4: MA-plots, ч.2

2.3 Исследование дифференциальной экспрессии

Создадим матрицу дизайна и переименуем столбики, чтобы было короче:

- cholangiocarcinoma → cc
- combined hepatocellular carcinoma and cholangiocarcinoma → hc_cc,

- hepatocellular carcinoma → hc

```
> factorname <- factor(new_groups$Factor.Value..TISSUE.)
> designmat <- model.matrix(~0 + factorname)
> colnames(designmat) <- c("cc", "hc_cc", "hc")
> head(designmat)
```

```
  cc hc_cc hc
1  0     0  1
2  1     0  0
3  0     0  1
4  0     1  0
5  0     0  1
6  0     0  1
```

```
> print(is.fullrank(designmat))
```

```
[1] TRUE
```

Также создадим матрицу контраста:

```
> contrmat <- makeContrasts(hc_cc - cc, hc_cc - hc, levels = designmat)
> head(contrmat)
```

```
       Contrasts
Levels  hc_cc - cc hc_cc - hc
  cc            -1          0
  hc_cc          1          1
  hc             0         -1
```

```
> print(is.fullrank(contrmat))
```

```
[1] TRUE
```

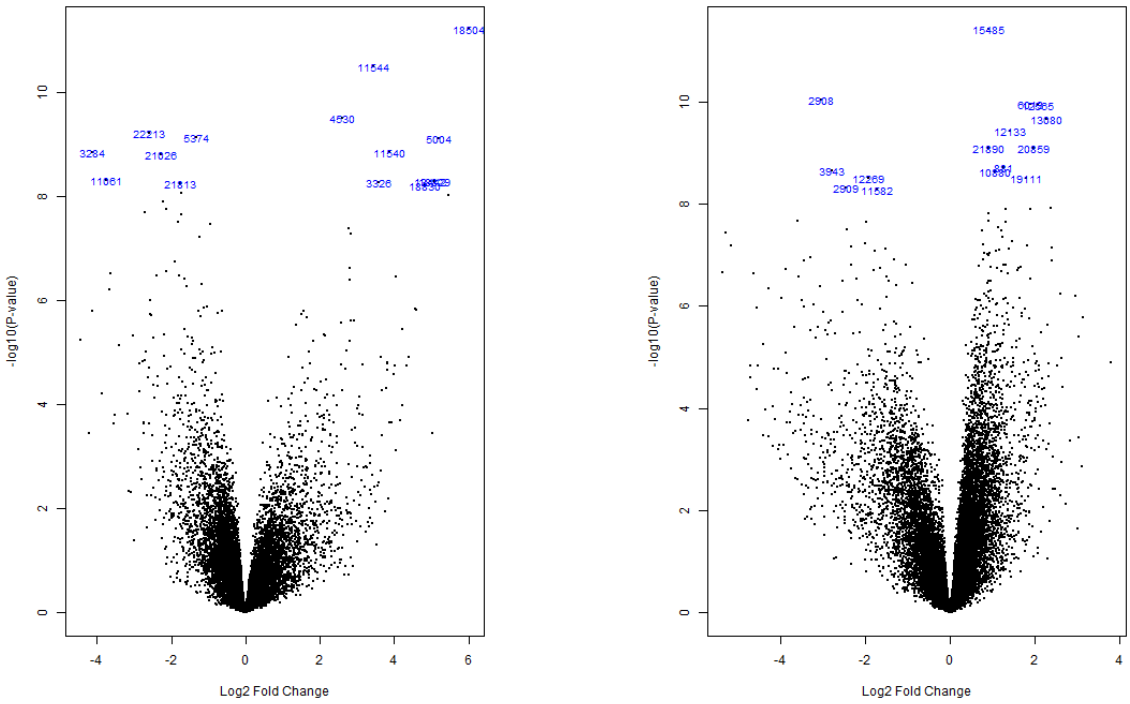
Теперь проведем анализ с помощью 'eBayes': обучим модели и вычислим набор статистик, чтобы понять зависимость факторов от профилей экспрессии.

```
> fitted <- lmFit(normed_new_aeset, designmat)
> contrfit <- contrasts.fit(fitted, contrmat)
> new_fitted <- eBayes(contrfit)
> head(new_fitted$coefficients)
```

```
           Contrasts
             hc_cc - cc  hc_cc - hc
  1007_s_at -0.82273863  1.85490510
  1053_at   -0.28737574 -0.23776832
  117_at    -0.31216182 -0.15047989
  121_at     0.07323826  0.48548548
  1255_g_at -0.08039222  0.06378417
  1294_at   -0.56144671 -0.35203948
```

Нарисуем графики-вулканы для разных коэффициентов:

```
> png("volcano1.png", height = 720)
> volcanoplot(new_fitted, coef = "hc_cc - cc", highlight = 15)
> dev.off()
> png("volcano2.png", height = 720)
> volcanoplot(new_fitted, coef = "hc_cc - hc", highlight = 15)
> dev.off()
```



(a) hc_cc-cc case

(b) hc_cc-hc case

Рис. 5: Volcano plots

Теперь отфильтруем гены по p-value сверхпредставленности: для первого случая $p_{val} = 0.05$, для второго - 0.001 (по информации статьи). Также отрисуем тепловые карты профилей тех генов, которые по результатам фильтрации будут признаны дифференциально экспрессирующимися.

```
> p_thresh_1 = 0.05
> res1 <- topTable(new_fitted, coef = "hc_cc - cc", number = nrow(normed_new_aeset))
> filtered_res1 <- res1[res1$adj.P.Val < p_thresh_1, ]
> head(filtered_res1)
```

```
               logFC    AveExpr         t      P.Value    adj.P.Val        B
219140_s_at  6.005432 11.600843  8.075676 6.053991e-12 1.348648e-07 16.45103
212158_at    3.449072 10.578168  7.705429 3.188901e-11 3.551957e-07 14.92793
205003_at    2.617243  7.788540  7.195518 3.094820e-10 2.298110e-06 12.84031
91826_at    -2.591255  4.991326 -7.050758 5.872518e-10 2.850422e-06 12.25117
205847_at   -1.319275  5.587869 -7.004908 7.189698e-10 2.850422e-06 12.06500
205477_s_at  5.197282 12.168530  6.990031 7.677215e-10 2.850422e-06 12.00463

> p_thresh_2 = 0.001
> res2 <- topTable(new_fitted, coef = "hc_cc - hc", number = nrow(normed_new_aeset))
> filtered_res2 <- res2[res2$adj.P.Val < p_thresh_2, ]
> head(filtered_res2)

               logFC    AveExpr         t      P.Value    adj.P.Val        B
216113_at    0.9176531  2.883135  8.177502 3.829256e-12 8.530434e-08 17.09433
203381_s_at -3.0411880 12.431643 -7.469327 9.159073e-11 6.539065e-07 14.12608
206487_at    1.8887096  4.036296  7.430086 1.090987e-10 6.539065e-07 13.96236
213183_s_at  2.0624626  3.797693  7.413600 1.174138e-10 6.539065e-07 13.89361
213700_s_at  2.2649238  4.865064  7.273371 2.190831e-10 9.761027e-07 13.30966
212748_at    1.4033476  5.933957  7.151591 3.759768e-10 1.395939e-06 12.80393
```

Посмотрим, какое количество проб разных категорий получилось:

```
> hc_cc_cc <- c(nrow(filtered_res1), nrow(filtered_res1[filtered_res1$logFC <= 0, ]),
+               nrow(filtered_res1[filtered_res1$logFC > 0, ]))
> hc_cc_hc <- c(nrow(filtered_res2), nrow(filtered_res2[filtered_res2$logFC <= 0, ]),
+               nrow(filtered_res2[filtered_res2$logFC > 0, ]))
> resframe <- rbind(hc_cc_cc, hc_cc_hc)
> colnames(resframe) <- c("total", "<=0", ">0")
> resframe

       total <=0  >0
hc_cc_cc   369 227 142
hc_cc_hc   245  82 163
```

Отрисуем heatmaps:

```
> res1_rownames <- rownames(filtered_res1)
> res2_rownames <- rownames(filtered_res2)
> hmap1 <- exprs(normed_new_aeset[res1_rownames, ])
> colnames(hmap1) <- unlist(pData(new_groups)[1])
> colnames(hmap1)[colnames(hmap1) ==
+                 "combined hepatocellular carcinoma and cholangiocarcinoma"] <-"combined"
> png("heatmap1.png", width = 1080, height = 1080)
> heatmap(hmap1, margins = c(10, 10))
> dev.off()
> hmap2 <- exprs(normed_new_aeset[res2_rownames, ])
```

```
> colnames(hmap2) <- unlist(pData(new_groups)[1])
> colnames(hmap2)[colnames(hmap2) ==
+                 "combined hepatocellular carcinoma and cholangiocarcinoma"] <-"combined"
> png("heatmap2.png", width = 1080, height = 1080)
> heatmap(hmap2, margins = c(10, 10))
> dev.off()
```

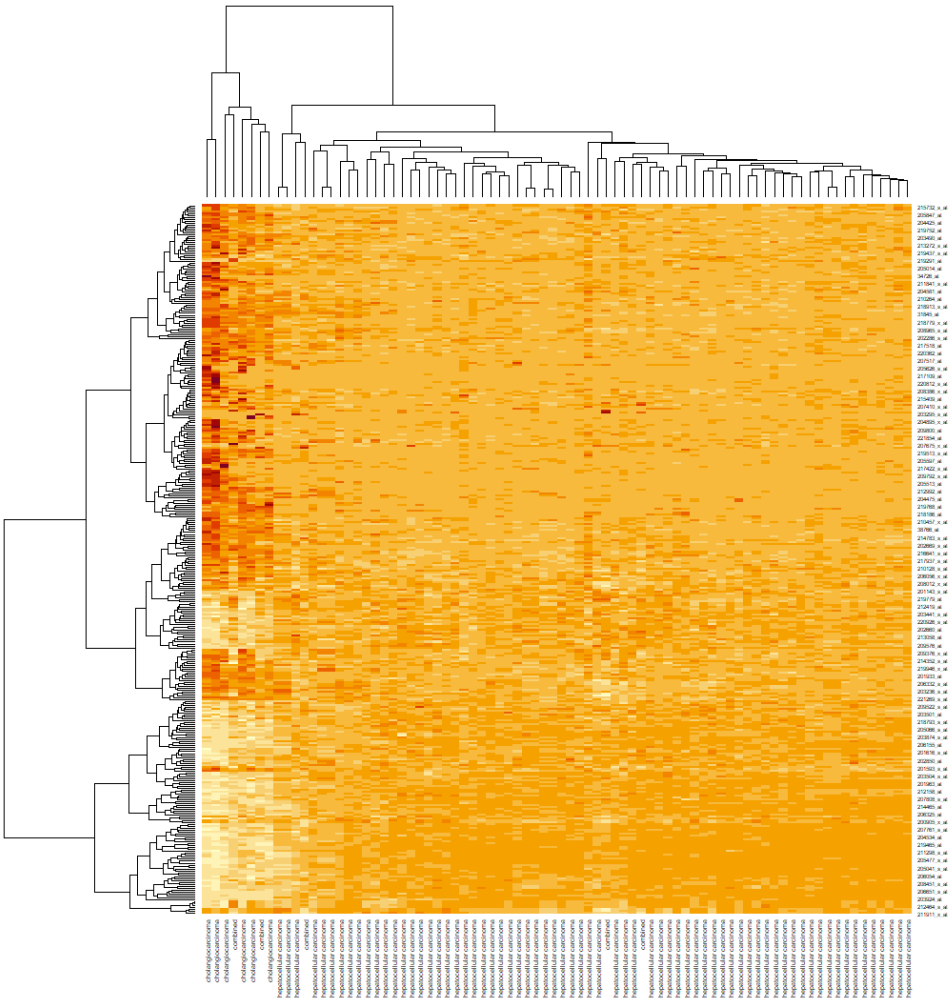


Рис. 6: Heatmap case 1: hc_cc-cc

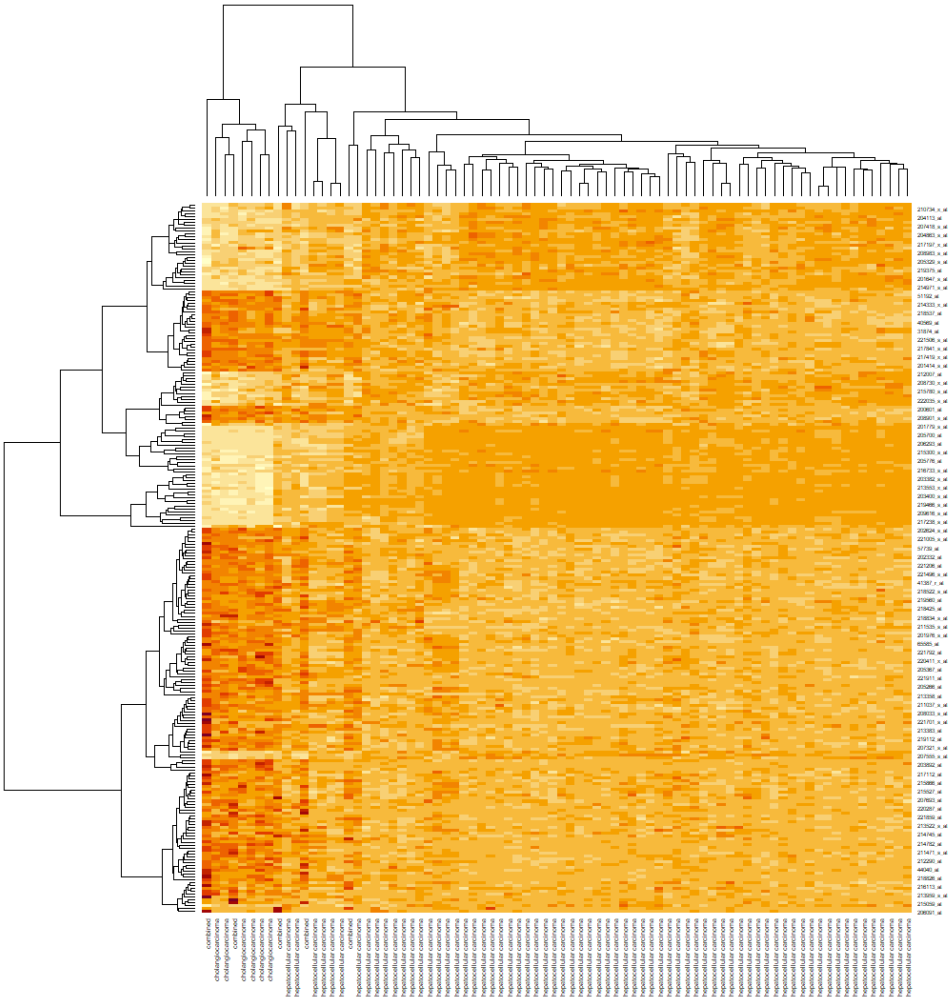


Рис. 7: Heatmap case 2: hc_cc-hc

2.4 Анализ обогащенности

Сохраним данные о генах, признанных дифференциально экспрессирующимися, для последующего анализа в онлайн-утилите DAVID.

```
> write(rownames(res1), file = 'background1.txt')
> write(rownames(filtered_res1), file = "genes1.txt")
> write(rownames(res2), file = 'background2.txt')
> write(rownames(filtered_res2), file = "genes2.txt")
```

Далее будем проводить анализ обогащенности путями KEGG, Reactome и GO.

Для этого выполним следующие действия:

1. Перейдем во вкладку "Upload"в левой части сайта. На первом шаге загрузим файл ‘genes1.txt‘. На втором шаге выберем формат ‘AFFYMETRIX_3PRIME_IVT_ID‘. На третьем шаге выберем ‘Gene List‘. Нажмем ‘Submit‘.

2. Вернемся на вкладку ‘Upload‘ и повторим ту же процедуру с файлом ‘background.txt‘, на третьем шаге укажем его как ‘Background‘.

3. Выберем ‘Functional Annotation Tool‘ в правой части страницы. В открывшемся меню выберем Gene Ontology: GOTERM_BP_DIRECT, GOTERM_CC_DIRECT, GOTERM_MF_DIRECT; Pathways: KEGG_PATHWAY, REACTOME_PATHWAY. Выберем ‘Functional Annotation Chart‘.

4. Скачаем открывшуюся таблицу по ссылке справа - ‘Download File‘. Сохраним файл как ‘ALL_res1.txt‘.

5. Аналогичную процедуру проделаем для файлов ‘genes2.txt‘ и ‘background2.txt‘.

Теперь загрузим результаты для дальнейшего анализа.

```
> res1_david <- read.table("ALL_res1.txt", sep = "\t", header = T)
> res1_david$Genes <- NULL
> res2_david <- read.table("ALL_res2.txt", sep = "\t", header = T)
> res2_david$Genes <- NULL
```

Для кейса №1 в статье указано, что CC-обогащение связно с функциями развития или дифференцировки, а также с метастазами/адгезией. Выделим такие результаты среди экземпляров в загруженном файле.

```
> match1 <- c("develop", "differ", "metastas", "adh")
> matched_res1 <- res1_david[grep(paste(match1, collapse = "|"), res1_david$Term), ]
> matched_res1[, c("Term", "PValue", "FDR")]
```

	Term	PValue	FDR
13	GO:0008544~epidermis development	2.444295e-05	0.009783292
32	hsa04514:Cell adhesion molecules (CAMs)	5.740529e-04	0.059414476
77	GO:0007160~cell-matrix adhesion	8.739801e-03	0.341278570
96	GO:0030216~keratinocyte differentiation	2.397800e-02	0.738245846
99	GO:0034116~positive regulation of heterotypic cell-cell adhesion	2.621698e-02	0.776927861
120	GO:0005925~focal adhesion	4.255197e-02	0.564780676
122	GO:0050839~cell adhesion molecule binding	4.369990e-02	0.996396396
179	GO:0010977~negative regulation of neuron projection development	7.085616e-02	0.990105133

Отдельно посмотрим на результаты пути Reactome (поскольку короткого описания для таких путей не представлено):

```
> react_annotations1 <- select(reactome.db,
+                          gsub(":.*","",
+                              res1_david[res1_david$Category ==
+                                       "REACTOME_PATHWAY", ]$Term),
+       keytype = "PATHID",
+       c("PATHNAME"))
> react_annotations1 <- react_annotations1[complete.cases(react_annotations1), ]
> react_annotations1$PATHNAME <- tolower(react_annotations1$PATHNAME)
> matched_res1_react <- react_annotations1[grep(paste(match1, collapse = "|"),
+                                       react_annotations1$PATHNAME),]
> matched_res1_react # результатов не нашлось...

[1] PATHID   PATHNAME
<0 rows> (or 0-length row.names)
```

Аналогичную процедуру проведем со вторым файлом. В данном случае для HCC обогащенность наиболее проявляется в функциях, связанных с метаболизмом и иммунитетом.

```
> match2 <- c("metabolism", "immun")
> matched_res2 <- res2_david[grep(paste(match2, collapse = "|"), res2_david$Term), ]
> matched_res2[, c("Term", "PValue", "FDR")]

[1] Term   PValue FDR
<0 rows> (or 0-length row.names)
```

Отдельно рассмотрим Reactome:

```
> react_annotations2 <- select(reactome.db,
+                          gsub(":.*","",
+                              res2_david[res2_david$Category ==
+                                       "REACTOME_PATHWAY", ]$Term),
+                         keytype = "PATHID",
+                         c("PATHNAME"))
> react_annotations2 <- react_annotations2[complete.cases(react_annotations2), ]
> react_annotations2$PATHNAME <- tolower(react_annotations2$PATHNAME)
> matched_res2_react <- react_annotations2[grep(paste(match2, collapse = "|"),
+                                       react_annotations2$PATHNAME),]
> matched_res2_react

      PATHID                                    PATHNAME
1 R-HSA-975634 homo sapiens: retinoid metabolism and transport
```

Выведем более подробную информацию:

```
> res2_david[res2_david$Term == paste(matched_res2_react$PATHID,
+                                matched_res2_react$PATHID, sep=":"), ]
```

```
        Category                    Term Count      X.     PValue
29 REACTOME_PATHWAY R-HSA-975634:R-HSA-975634     4 1.869159 0.02157339
   List.Total Pop.Hits Pop.Total Fold.Enrichment Bonferroni Benjamini FDR
29        115       36      6843        6.611594  0.9987636         1   1
```

Рассмотрим отдельно ген TP53, упомянутый в статье. Скачаем все связанные с ним GO-пути и посмотрим, нашли ли мы что-то похожее.

```
> TP53 <- read.csv("TP53.csv", header = T)
> TP53_GO <- TP53$Accession
> rbind(res1_david[substring(res1_david$Term, 1, 10) %in% TP53_GO, ],
+ res2_david[substring(res2_david$Term, 1, 10) %in% TP53_GO, ])[, c("Term", "PValue")]
```

```
                                        Term     PValue
160 GO:0060333~interferon-gamma-mediated signaling pathway 0.06596447
87                  GO:0012501~programmed cell death 0.09525254
92                      GO:0042493~response to drug 0.09537280
```

3 Выводы

Итак, мы рассмотрели анализ дифференциальной экспрессии генов и провели анализ обогащенности, а также построили тепловые карты, которые, как и предполагалось авторами статьи, практически не показывают больших различий между комбинированным вариантом опухоли и отдельно взятыми CC и HCC.

Также мы получили 8 дифференциально экспрессируемых обогащенных путей для CC (из баз GO и KEGG), на типах которых делался акцент в статье. Все эти образцы имеют скорректированный p-value меньше 0.1. Для HCC был получен только один путь в базы Reactome, имеющий значение $p-value \approx 0.02$.

Отдельно был рассмотрен ген TP53, который указан авторами как значительно обогащенный ($p < 0.036$). Авторы подчеркнули, что TP53 может играть ключевую роль в развитии CLHCC - CC signature-expressing HCC. Известно также, что мутации гена TP53 присутствуют при многих типах рака. В нашем исследовании мы также нашли несколько путей GO, имеющих связь с этим геном:

- GO:0060333 interferon-gamma-mediated signaling pathway
- GO:0012501 programmed cell death
- GO:0042493 response to drug

Список литературы

[1] Hyun Goo Woo, Jeong-Hoon Lee, Jung-Hwan Yoon, Chung Yong Kim, Hyo-Suk Lee, Ja June Jang, Nam-Joon Yi, Kyung-Suk Suh, Kuhn Uk Lee, Eun Sung Park, Snorri S. "Thorgeirsson and Yoon Jun Kim Identification of a Cholangiocarcinoma-Like Gene Expression Trait in Hepatocellular Carcinoma"(2010). | doi: 10.1158/0008-5472.CAN-09-2823