مقدمه ای بر بیوانفورماتیک

پروژه پایانی

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال اول ۰۱-۰۰

اساتید : سرکار خانم دکتر کوهی و جناب آقای دکتر شریفی

نام و نام خانوادگی : افشار :)

۱ مقدمه

در این پروژه، قصد داریم به تحلیل داده های ریزآرایه ی لوکمی (لوسمی) حاد مغزاستخوان بپردازیم. لوکمی حاد میلوئیدی (Acute Myeloid Leukemia) یا به اختصار AML یکی از انواع سرطان خون است. این نوع لوکمی سلول‌های مغز استخوان یا میلوسیت‌ها را تحت تأثیر قرار می‌دهد و روندی حاد دارد. در این بیماری مغز استخوان، میلوبلاست‌ها (نوعی گلبول سفید)، گلبول‌های قرمز یا پلاکت‌های غیرطبیعی می‌سازد. سلول‌های لوسمیک نابالغ اغلب بلاست نامیده می‌شوند که به تولید مثل و تجمع ادامه می‌دهند. در لوکمی میلوئیدی منشا سلول‌های سرطانی رده میلوسیت‌ها که گلبول‌های قرمز، پلاکت‌ها یا سایر گلبول‌های سفید بجز لنفوسیت مانند گرانولوسیت‌ها و مونوسیت‌ها را می‌سازند است.در ادامه میخواهیم با تحلیل این داده ها به ارتباط بین تغییرات بیان ژن در افراد مبتلا به این بیماری و مقایسه ی آن افراد سالم بپردازیم و در نهایت ژن هایی که بیانشان کم یا زیاد شده را مشخص کنیم و در نهایت در مورد این ژن ها و عملکرد آنها اطلاعاتی بدست آوریم.

۲ آماده سازی و کنترل کیفیت داده ها

ابتدا نیاز داریم چند کتابخانه ی زبان آر که به آن ها نیاز داریم را به ابتدای کد خود اضافه کنیم :

```
1. library(Biobase)
2. library(GEOquery)
3. library(limma)
4. library(pheatmap)
5. library(ggplot2)
6. library(plyr)
7. library(reshape2)
8. library(gplots)
```

۳ خواندن داده از سایت و دسته بندی مجموعه دادگان

برای اینکه به داده ها را از سایت مورد نظر بخوانیم از کتابخانه ی GEOquery استفاده کنیم. دیتاست داده شده GSE48558 است و این دیتاست را در یک متغیر و پلتفرم که GPL6244 میباشد را در متغیر دیگر ذخیره میکنیم تا در صورت لزوم دیتاست یا پلتفرم را بتوانیم تغییر دهیم.

```
1. series <-"GSE48558"
2. platform <- "GPL6244"
```

در ادامه باید داده ها را ذخیره کنیم برای این منظور از کد زیر استفاده میکنیم :

```
gset <- getGEO(series, GSEMatrix =TRUE, AnnotGPL=TRUE, destdir =
"Data/")
if (length(gset) > 1){idx <- grep(platform, attr(gset, "names"))
} else {idx <- 1 }
gset <- gset[[idx]]
```

با توجه به موارد خواسته شده باید داده را به دو دسته تقسیم کنیم به این منظور در مجموعه دادگان ، داده هایی که Phenotype آنها Normal است را گروه Normal و نمونه هایی که name source آن ها patient AMLاست را گروه Test در نظر میگیریم و از کد زیر استفاده میکنیم :

```
gset <- gset[,which(gset$`phenotype:ch1` == "Normal" |
gset$source_name_ch1 == "AML Patient")]
CreatLabel <- function(x) {
if (gset$source_name_ch1[x] == "AML Patient") {
return("Test")}
else {return("Normal")}}
```

۴ کنترل کیفیت داده

به این منظور نیاز است میانه داده ها و چارک های اول و سوم را چک کنیم اگر نزدیک به یکدیگر بودند داده ها نیاز به نرمالایز شدن ندارند اما اگر تفاوت زیادی داشتند باید داده ها را نرمالایز کنیم. در ابتدا ماتریس بیان را تشکیل می دهیم و مینیمم و ماکسیسم را در خروجی چاپ کنیم :

```
ex <-exprs(gset)
min_ex <- min(ex)
max_ex <- max(ex)
print(c(min_ex, max_ex))
```

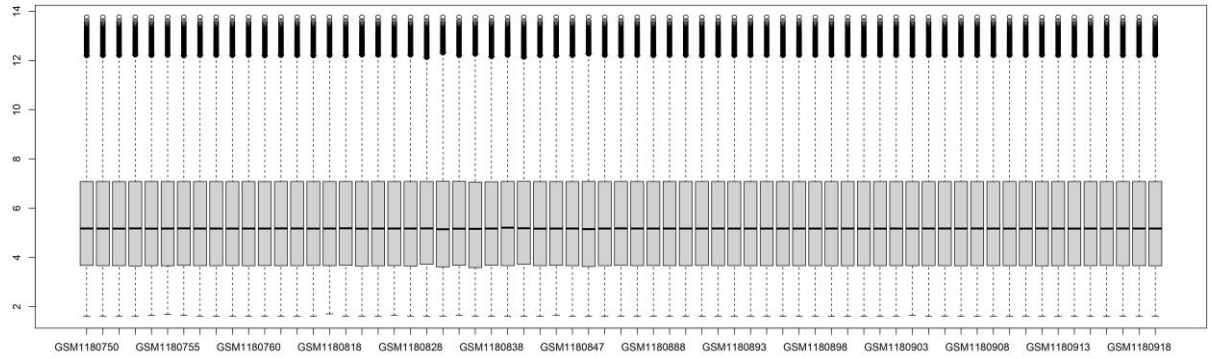
خروجی به این صورت میباشد. از این خروجی به این نتیجه میرسیم که در مقیاس لگاریتم هستند :

```
[1]   1.611473179  13.76153622
```

۴,۱ بررسی نرمال بودن داده

به این منظور نیاز به کد زیر داریم و در نهایت تصویر را بصورت pdf ذخیره کنیم :

```
pdf("Results/boxplot.pdf", width = 20)
boxplot(ex)
dev.off()
```



شکل 1 نمودار Box Plot

با توجه به توضیحات گفته شده و میانه ها و چارک ها داده ها نرمالایز هستند و نیاز به نرمال شدن ندارند. اما اگر داده ها نرمالایز نشده بودند باید از کد زیر استفاده کرد :

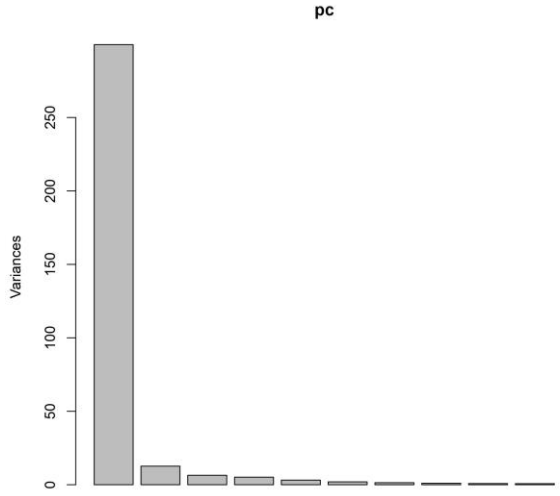
```
1. ex <- normalizeQuantiles(ex)
2. ex(gset) <- ex
```

۵ کاهش ابعاد

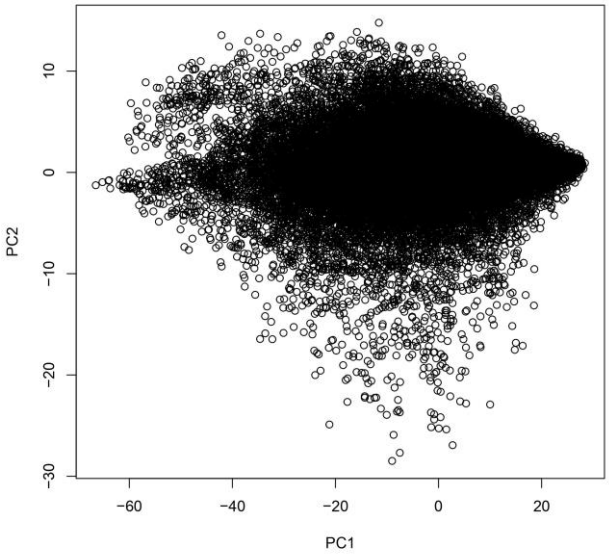
به این منظور از روش Principal Component Analysis استفاده میکنیم. تجزیه و تحلیل مؤلفه اصلی (PCA) تکنیکی برای کاهش ابعاد این مجموعه داده ها، افزایش قابلیت تفسیر و در عین حال به حداقل رساندن از دست دادن اطلاعات است. این کار را با ایجاد متغیرهای نامرتبط جدید انجام می دهد که متوالی واریانس را به حداکثر می رساند. به این منظور از این کد بهره میبریم و نتیجه را در پوشه ی مربوطه ذخیره میکنیم :

```
1. pc <- prcomp(ex)
2. pdf("Results/PC.pdf")
3. plot(pc)
4. plot(pc$x[,1:2])
5. dev.off()
```

خروجی به این صورت میباشد :



شکل2 نمودار میله ای تغییرات بیان ژن در مولفه های PCA



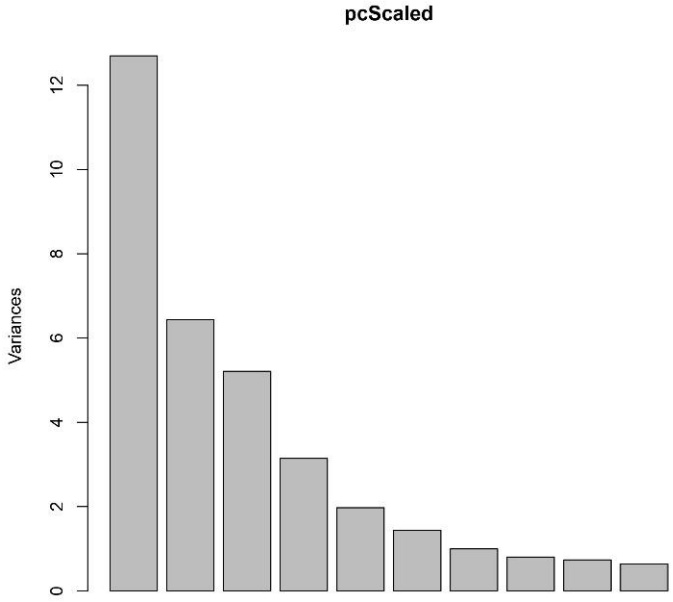
شکل3 نمودار نقطه ای داده ها با بیشترین اهمیت

در شکل ۳ هر نقطه یک ژن میباشد که نشان میده در راستای PCA1 بیشترین تغییر در بین ژن ها وجود دارد. سپس متوجه میشویم که با توجه به این شکل یک سری ژن ها اصلا بیان نشدند و یک سری ژن ها هستند که

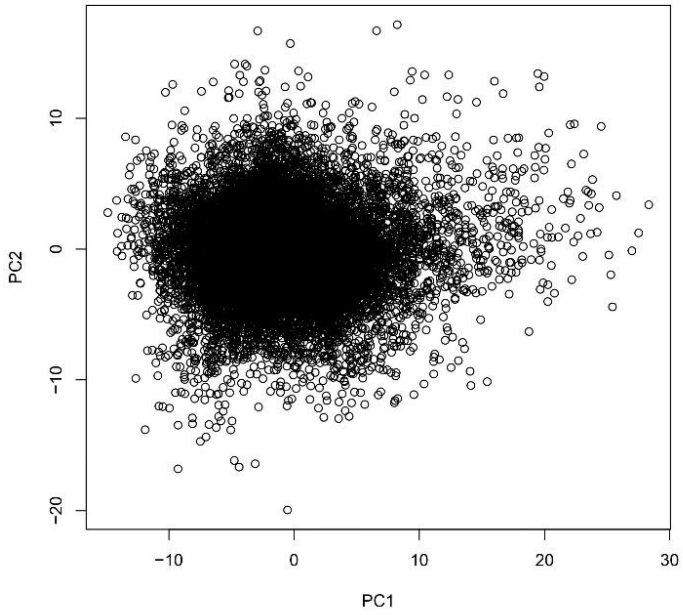
همیشه بیان شده اند که به این ژن ها Housekeeping میگوییم. حال باید میانگین بیان همه ی ژن ها را از خود ژن کم کنیم یا به بیان دیگر میانگین همه ی ژن ها را صفر کنیم یا به عبارتی Scale میکنیم. به این منظور نیاز به یک متغیر دیگر برای ذخیره ی ژن ها داریم تا داده ی اصلی هم از دست ندهیم. حال باید PCA را در تغییرات ژن ها اعمال کنیم. برای Scale کردن از کد زیر بهره میبریم :

```
1. ex.scaled <- t(scale(t(ex), scale = F))
2. pcScaled <- prcomp(ex.scaled)
3. pdf("Results/PC_Scaled.pdf")
4. plot(pcScaled)
5. plot(pcScaled$x[,1:2])
6. dev.off()
```

سپس مجدد خروجی را ذخیره میکنیم.



شکل 4 نمودار میله ای تغییرات بیان ژن در مولفه های PCA پس از Scale شدن



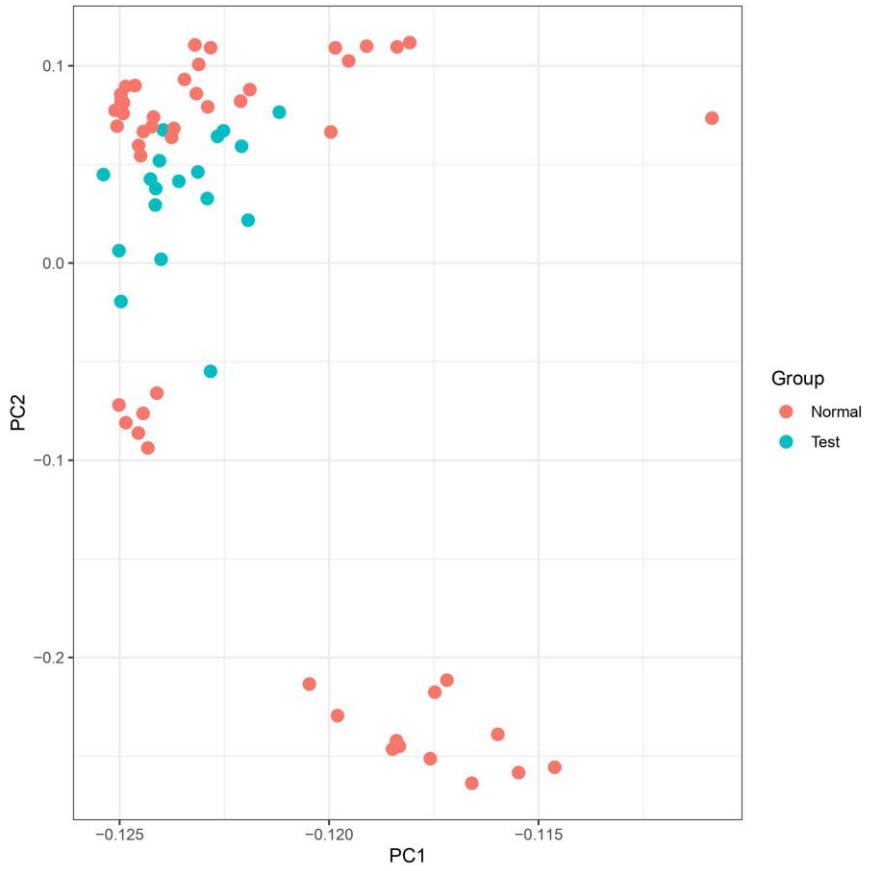
شکل5 نمودار نقطه ای داده ها با بیشترین اهمیت پس از Scale شدن

خروجی کد زیر نمودار نقطه ای نمونه ها بر حسب دو مولفه اصلی در PCA را بر اساس دسته بندی ای که کردیم نشان میدهد و چون به تمام PCAها نیاز نداریم و نهایت ۳ بعد مد نظرمان است از rotation[,1:3] استفاده میکنیم و خروجی را ذخیره میکنیم.

```
1. pcr <- data.frame(pc$r[,1:3] , Group = gr)
2. pdf("Results/PCA_samples.pdf")
3. ggplot(pcr, aes(PC1, PC2, color= Group)) + geom_point(size=3) +
   theme_bw()
4. dev.off()
```

از theme_bw() هم برای وضوح بیشتر نمودار استفاده میکنیم. در ادامه خروجی و توضیحاتی در مورد این نمودار قرار دارد.

شکل ۶ نشان میدهد که یک سری از داده های AML به خوبی از بقیه جدا شده اند و همچنین مشخص است که دسته ای از گروه Normal کاملا از گروه بیمار جدا هستند که در ادامه اگر نیاز بود می توانیم آن ها را کنار بگذاریم.



شکل6 نمودار میله ای با گروه بندی بر اساس دو مولفه اصلی PCA

۶ بررسی همبستگی بین نمونه ها

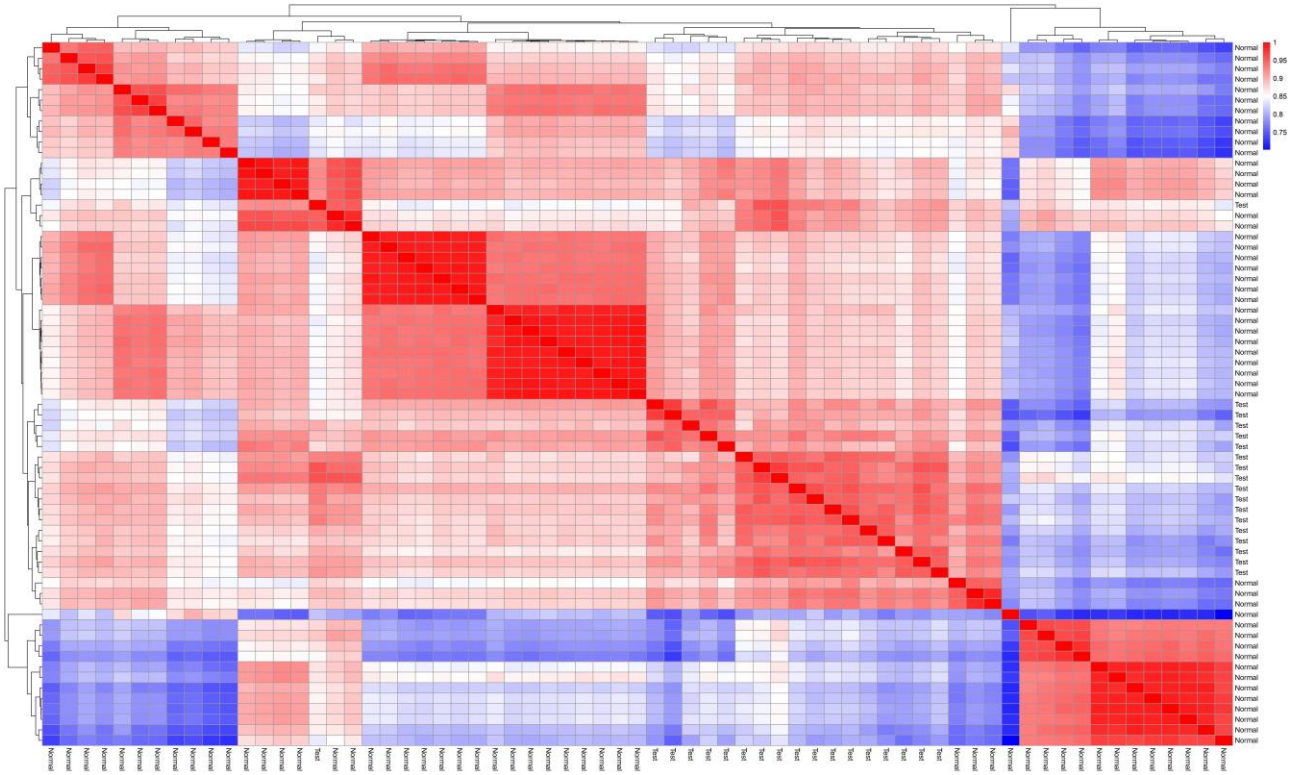
برای این منظور نیاز داریم تا نمودار Heatmap را رسم کنیم و برای رسم آن نیاز است از کد زیر استفاده کنیم:

```
1. pdf("Results/CorHeatmap.pdf", width = 25, height = 15)
2. pheatmap(cor(ex), labels_row =  gr , labels_col =  gr, color =
   bluered(256))
3. dev.off()
```

بوسیله دستور `color = bluered(256))` خروجی را به رنگ آبی و قرمز درآوردیم.

هر مربع همبستگی بین متغیرها را در هر محور نشان می دهد. محدوده همبستگی از -۱ تا +۱ است. مقادیر نزدیک به صفر به این معنی است که هیچ روند خطی بین دو متغیر وجود ندارد. هر چه این همبستگی نزدیک به

۱ باشد، همبستگی مثبت‌تری دارند. یعنی با افزایش یکی، دیگری افزایش می یابد و هر چه به ۱ نزدیکتر باشد این رابطه قوی تر می شود. همبستگی نزدیک به -۱ مشابه است، اما به جای افزایش هر دو، یک متغیر با افزایش متغیر دیگر کاهش می یابد. قطرها همگی ۱/قرمز هستند زیرا هر متغیر با خودش مرتبط است (بنابراین همبستگی کامل است). برای بقیه، هر چه عدد بزرگتر و رنگ قرمز تر باشد، همبستگی بین دو متغیر بیشتر است و هر چه آبی تر شود کمتر.



شکل7 نمودار Heatmap به منظور نشان دادن همبستگی بین نمونه ها

۷ بررسی تمایز در بیان ژن ها در افراد سالم و بیمار

در این مرحله میخواهیم ژن هایی که در افراد بیماری بیانشان افزایش یا کاهش داشته را پیدا کنیم. برای اینکار از تکه کد های زیر استفاده میکنیم که با شماره سطرها قابل تشخیص هستند. ابتدا با توجه به شکل ۶ متوجه میشویم که یک سری داده های Normal کاملا از داده های Test جدا افتاده اند به این منظور این قسمت از داده ها را کنار میگذاریم و داده هایی که بیشتر به Test نزدیک است را لحاظ میکنیم. این داده ها CD34 نام دارند. حال برای سادگی کار به جز AML patient و CD34 که هر کدام را در یک دسته جدا قرار میدهیم بقیه را در یک دسته قرار میدهیم چون در نهایت Test و CD34 برای ما اهمیت دارند. به این صورت :

```
gr <- c(rep("Test", 13), rep("other", 86), "CD34", rep("other",
3), "CD34", rep("other", 3), "CD34", rep("other", 36),
rep("Test", 2), "other", rep("Test", 3), rep("other", 20))
```

حال که گروه ها مشخص شد باید کد زیر را اجرا میکنیم :

```
gr <- factor(gr)
gset$description <- gr
```

در خط اول از تابع factor برای تبدیل فرمت گروه هایمان (که بصورت تکست است) به فاکتور استفاده میکنیم.

تابع factor داده هایمان را به سه level یا گروهی که در مشخص کردیم تبدیل میکند. یعنی این کد :

```
gr <- factor(gr)
gr
```

این خروجی را می دهد :

```
levels : Test other CD34
```

حال به ادامه ی کد ها میپردازیم :

```
design <- model.matrix(~ description + 0, gset)
colnames(design) <- levels(gr)
```

متغیر design یک ماتریس است که به ازای هر گروه و هر نمونه یک سطر دارد و نشان میدهد که هر نمونه ای برای کدام یک از گروه ها میباشد. در ادامه ی کد ها از پکیج Limma استفاده می کنیم. ابتدا به وسیله تابع lmFit بر اساس گروه design یک مدل خطی به دیتا نسبت میدهیم. در خط دوم دیتاهای Test و CD34 را باهم مقایسه میکنیم.

```
fit <- lmFit(gset, design)
cont.matrix <- makeContrasts(Test - CD34, levels = design)
fit2 <- contrasts.fit(fit, cont.matrix)
fit2 <- eBayes(fit2, 0.01)
```

در ادامه شیب خط را بر اساس تفاوت Test - CD34 بدست می آوریم و در متغیر fit2 ذخیره میکنیم.

```
tT <- topTable(fit2, adjust="fdr", sort.by="B", number=Inf)
tT <- subset(tT,
select=c("ID","adj.P.Val","P.Value","t","B","logFC","Gene.symbol"
,"Gene.title"))
tT <- subset(tT , select = c("Gene.symbol" , "Gene.title",
"adj.P.Val"  , "logFC"))
```

در حال حاضر ما ژن هایی که تفاوت معنی داری دارند را پیدا کرده ایم سپس سطح معنی داری را 0.05 در نظر میگیریم. حال میخواهیم ژن هایی با افزایش بیان را مشخص و ذخیره کنیم که برای این کار از کد زیر بهره میبریم:

```
aml.up <- subset(tT, logFC > 1 & adj.P.Val < 0.05)
aml.up.genes <-unique( as.character(strsplit2(
(aml.up$Gene.symbol),"///")))
write.table(aml.up.genes, file = "Results/Test_CD34_Up.txt",
row.names = F, col.names = F , quote = F)
```

و در آخر هم ژن های با کاهش بیان را مشخص و ذخیره میکنیم :

```
aml.down <- subset(tT, logFC < -1 & adj.P.Val < 0.05)
aml.down.genes <-unique( as.character(strsplit2(
(aml.down$Gene.symbol),"///")))
write.table(aml.down.genes, file = "Results/Test_CD34_down.txt",
row.names = F, col.names = F , quote = F)
```

فایل تکست ژن های با کاهش/افزایش بیان در فایل اصلی پروژه موجود می باشد.

۸ آنالیز Gene ontology و pathway ها

برای تحلیل این قسمت از Enrich بهره میبریم. با استفاده از TRANSFAC and JASPAR PWMs متوجه می شویم چه ژن هایی وجود دارند که ژن هایی که پیدا کردیم را کنترل میکنند و دلیل بالا یا پایین رفتن بیانشان شده اند. برای مثال PGR و STATB5 و PUR جز ژن هایی بودند که در در افزایش بیان برخی ژنها تاثیر داشتند. و همچنین EWSR1-FLI1 در کاهش بیان برخی دیگر از ژن ها. اما با توجه به APV این موارد مشخص است که Significant نیستند.

۸,۱ تحلیل Pathway ها

در بیوانفورماتیک، روش‌های تجزیه و تحلیل مسیر ممکن است برای شناسایی ژن‌ها/پروتئین‌های کلیدی در مسیری که قبلاً شناخته شده‌اند در رابطه با یک آزمایش خاص/شرایط پاتولوژیک یا ساختن مسیری جدید از پروتئین‌هایی که به عنوان عناصر اصلی آسیب‌دیده شناسایی شده‌اند، استفاده شود.

۸,۱,۱ تحلیل pathway ها برای ژن های با افزایش بیان

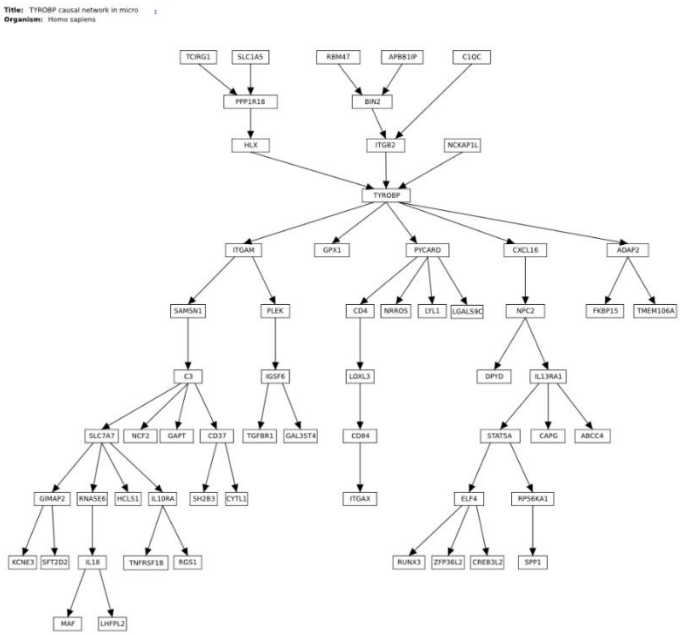
با توجه به خروجی که Wikipathways Human 2021 به ما داده است متوجه چند موارد زیر می شویم :

- TYROBP causal network : Adj.p-value = 2.596e-10

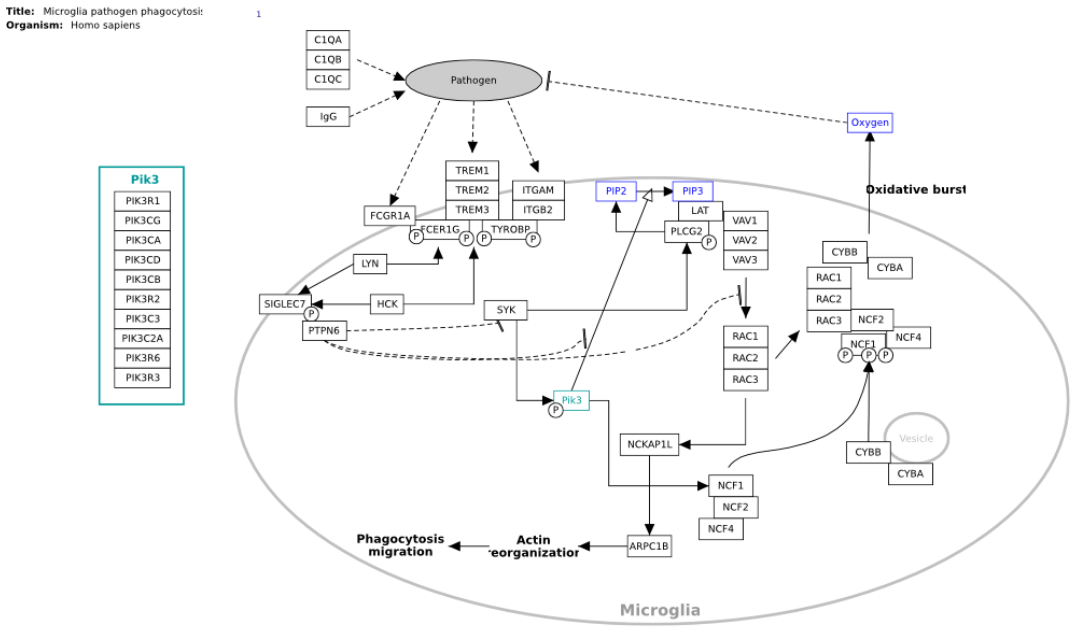
- Microglia Pathogen Phagocytosis : Adj.p-value = 7.656e-7

در ادامه تصویر این pathway ها آورده شده است.

جدول تمامی این بخش ها در فایل گزارش کار قرار دارد.



شکل 8 : TYROBP causal network

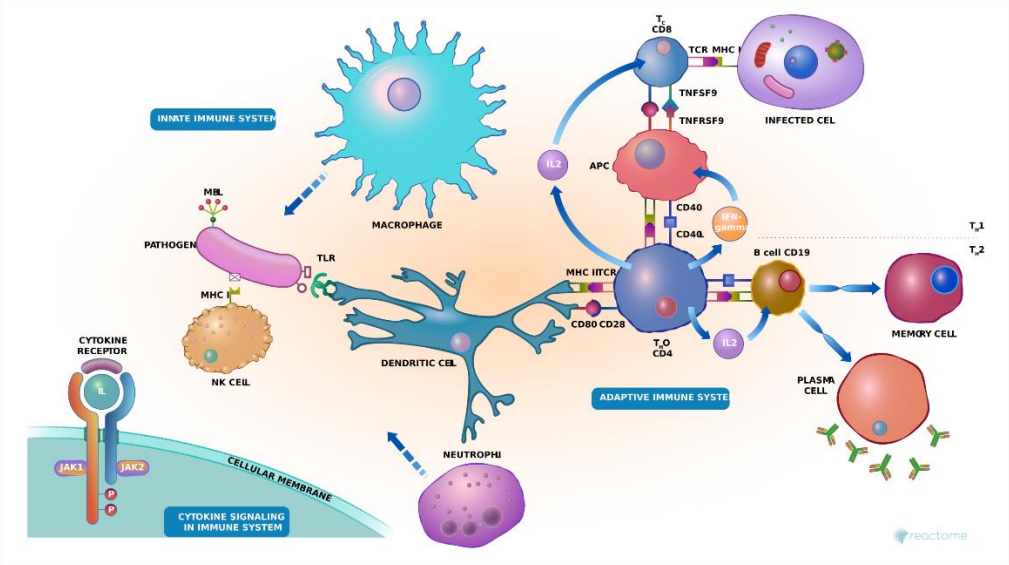


شکل 9 : Microglia Pathogen Phagocytosis

میتوانیم Kegg یا Reactome را هم بررسی کنیم.

Reactome 2016 :

- Immune System Homo sapiens R-HSA-168256 : Adj.p-value = 1.455e-13



شکل 10 : Immune System Homo sapiens R-HSA-168256

جدول این بخش در فایل گزارش کار قرار دارد.

۸,۱,۲ تحلیل pathway ها برای ژن های با کاهش بیان

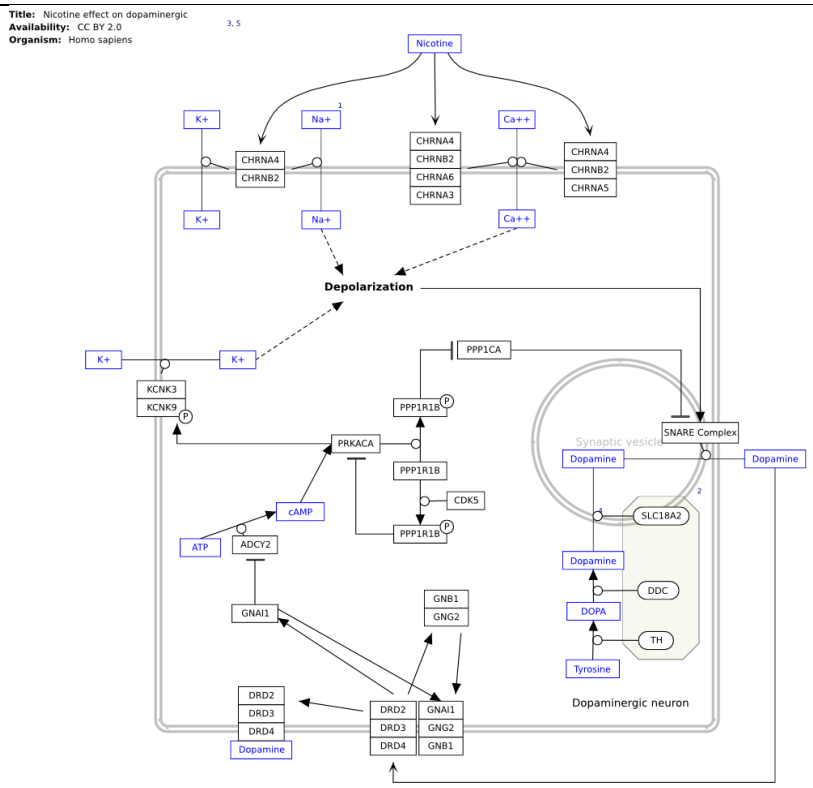
APV هایی که در بخش Wikipathways Human 2021 قرار داشتند چندان مناسب نبودند چون مقدارشان بالا بود. در KEGG و Reactome هم به همین شکل. اما در بخش ARCHS4 KINASES COEXP متوجه وجود یک pathway با APV مناسب شدیم :

- PRKG2 human : Adj.p-value = 0.004122

در بخش BioPlanet :

- Nicotine activity on dopaminergic neurons : p-value = 0.008081

جدول تمامی این بخش ها در فایل گزارش کار قرار دارد.



شکل 11 : Nicotine activity on dopaminergic neurons

۸,۲ بررسی و تحلیل Gene Ontology

هستی شناسی ژن (GO) یک ابتکار بیوانفورماتیک بزرگ برای یکسان کردن نمایش ژن و ویژگی های محصول ژن در همه گونه ها است. ... در حالی که نامگذاری ژن بر ژن و محصولات ژنی تمرکز دارد، هستی شناسی ژن بر عملکرد ژن ها و محصولات ژنی تمرکز دارد.

۸,۲,۱ بررسی و تحلیل GO در ژن های با بیان بیشتر در بیماران

در زمینه ی Biological Process، موارد زیر قابل توجه هستند:

- neutrophil activation involved in immune response : Adj.p-value = 3.627e-15
- neutrophil mediated immunity : Adj.p-value = 8.699e-15
- neutrophil degranulation : Adj.p-value = 7.479e-15

در زمینه ی Molecular Function، موارد زیر قابل توجه هستند:

- amyloid-beta binding : Adj.p-value = 0.001651
- aryl sulfotransferase activity : Adj.p-value = 0.003494

- protein kinase binding : Adj.p-value = 0.004194

در زمینه ی Cellular Component، موارد زیر قابل توجه هستند:

- secretory granule membrane : Adj.p-value = 1.457e-9
- tertiary granule : Adj.p-value = 1.457e-9
- specific granule : Adj.p-value = 4.653e-9

جدول تمامی این بخش ها در فایل گزارش کار قرار دارد.

۸,۲,۲ بررسی و تحلیل GO در ژن های با بیان کمتر در بیماران

در زمینه ی Biological Process، مورد زیر قابل توجه است اما مابقی موارد APV مناسبی نداشتند:

- activation of GTPase activity (GO:0090630) : Adj.p-value = 0.0002731

در زمینه ی Molecular Function، ، مورد زیر قابل توجه است اما مابقی موارد APV مناسبی نداشتند:

- GTPase activator activity (GO:0005096) : Adj.p-value = 0.04220

در زمینه ی Cellular Component، موردی با APV مناسب و پایین وجود نداشت.

جدول تمامی این بخش ها در فایل گزارش کار قرار دارد.

۱۰ نتایج نهایی

هدف از انجام این پروژه تحلیل داده های ریز آرایه ی بیماری لوکی حاد مغزاستخوان بود در ابتدا کیفیت داده ها را بررسی کردیم و همبستگی بین داده ها را مشخص کردیم و با نمودار نقطه ای گروه ها متوجه شدیم تعدادی از داده های دسته ی نرمال شباهتی به گروه تست نداشتند و آن ها را در دسته ای جدا گانه قرار دادیم و ژن های با افزایش یا کاهش بیان در افراد بیمار را با مقایسه ی داده های آن با دسته ی CD34 بدست آوردیم. در ادامه متوجه شدیم ژن هایی که به عنوان ژن های با افزایش بیان پیدا کرده ایم تا حد خوبی دارای APV خوب و معنی داری بودند که به نسبت این موضوع در مورد ژن های با کاهش بیان مشاهده نشد و اکثرا APV بالایی داشتند.
- فایل گزارش شامل سه بخش میباشد که کدها و نتایج و نمودار ها و داده ی اصلی و گزارش کار در آن قرار گرفته است.