Simulated.RData是100次重复的最后一次！

有POI影响（一个变量，只是POI），与下面的等同：

——————————————————————————————————

POI<-simdata$pheno[,1]

CFs1<-simdata$pheno[,2]

CFs2<-simdata$pheno[,3]

mod0<-model.matrix(~1, data=data.frame(simdata$pheno))

mod<-model.matrix(~POI, data=data.frame(simdata$pheno))

——————————————————————————————————

> mod0<-model.matrix(~1, data=data.frame(tmp.m))

> mod<-model.matrix(~POI, data=data.frame(tmp.m))

> pValues = f.pvalue(simulate.data.m,mod,mod0) （simulate.data.m行为特征，列为样本）

> length(which(pValues<0.05))

[1] 56

> length(which(pValues[1:1800]<0.05))

[1] 52

> length(which(pValues[1801:2000]<0.05))

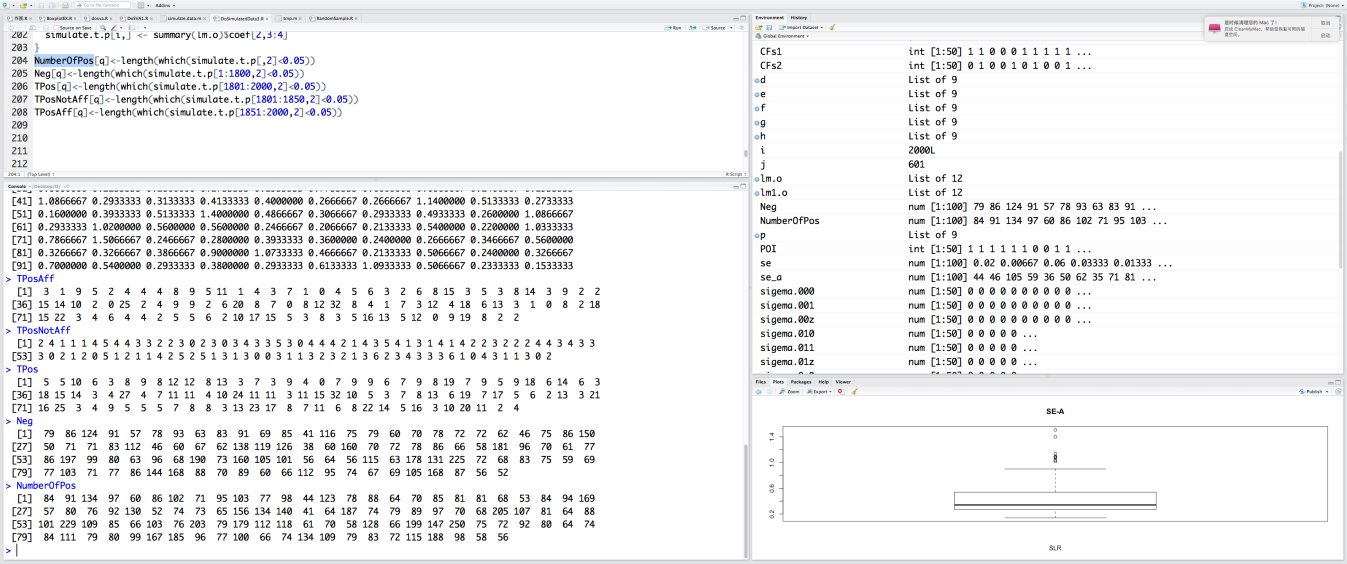
[1] 4

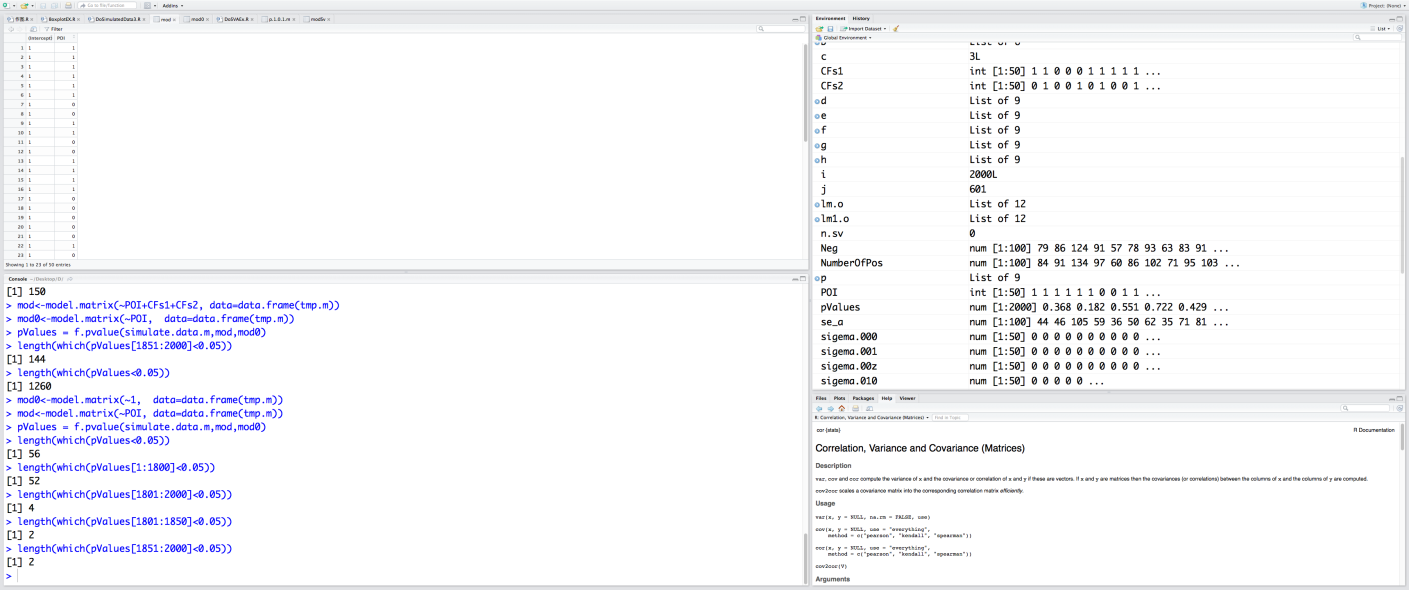
> length(which(pValues[1801:1850]<0.05))

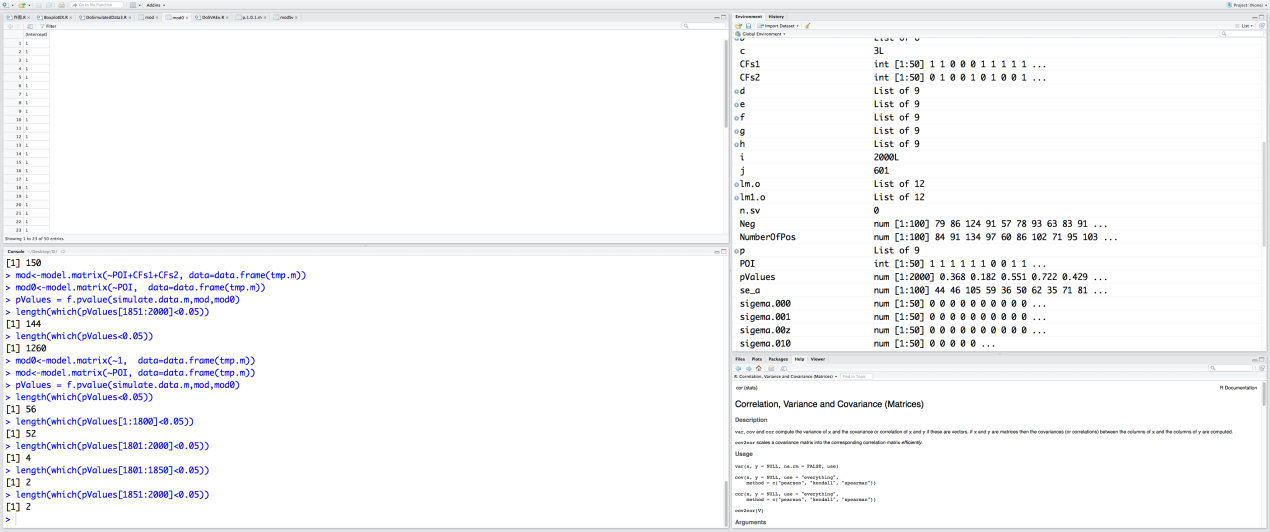
[1] 2

> length(which(pValues[1851:2000]<0.05))

[1] 2







与以下等同：

simulate.t.p <- matrix(NA,2000,2)

for (i in 1:2000) {

lm.o<-lm(simulate.data.m[i,]~POI) ####只有POI 与mode相同

simulate.t.p[i,] <- summary(lm.o)$coef[2,3:4]

}

#pValues = f.pvalue(simulate.data.m,mod,mod0)

NumberOfPos[q]<-length(which(simulate.t.p[,2]<0.05)) #length(which(pValues<0.05))

Neg[q]<-length(which(simulate.t.p[1:1800,2]<0.05)) #==length(which(pValues[1:1800]<0.05))

RealExprStu.R是针对学生给的

将GBMDODVA.RData，SVAGBM CANCER.RData全载入后再运行RealExprStu.R

将GBMDODVA.RData，SVAGBM CANCER.RData全载入后再运行RealExprStu.R

dodva，ESCA癌症相关.RData，SVAesca cancer.RData全载入后再运行RealExprStu.R



***sorteddvacorr[j,]<-sort(dvacorr[j,],decreasing = T,index.return = T)是对整个矩阵进行排序，按列！***

***Is.na()只能用于一个值，而非向量***

***sort(dvacorr[j,],decreasing = T,index.return = T)返回list！！！！！***

***E:\TCGA\_data文件下数据解释***

***XXXTCGA\_DNAm 是未做bmiq的数据（和beadchip是两码事），未含有beadchip差异，***

***Bmiq是比如全做很费钱，一部分是一种方法做，一部分是另外方法做，bmiq做校正***

***XXXTCGA\_DNAm做了bmiq变成TCGA\_DNAm\_bmiq下的数据（主要用的甲基化数据）***

***TCGA\_RNAseqV2dmclog2为基因差异表达数据，要+1及log2***

***0+1log2***

***bmiq校正芯片另外一种误差***

***PhenoTypesWB.lv[[selcat.idx[4]]为癌症0，1***

R内置函数kruskal.test()可以完成Kruskal-Wallis秩和检验，使用如下:

***> tmp.v <- rmt.pca$rotation[,1]***

***> k1<-kruskal.test(tmp.v ~ as.factor(PhenoTypesWB.lv[[selcat.idx[4]]]))$p.value;***

***> k1***

***[1] 0.3134927***

***> lm1<-summary(lm(tmp.v ~ PhenoTypesWB.lv[[selcat.idx[4]]]))$coeff[2,4];***

***> lm1***

***[1] 0.3100484***

***//根据因子类型使用！！***

***for(c in 1:topPCA){***

***tmp.v <- rmt.pca$rotation[,c];***

***for(f in 1:length(selcat.idx)){***

***if(factor.log[f]){***

***svdPVwb.m[c,f] <- kruskal.test(tmp.v ~ as.factor(PhenoTypesWB.lv[[selcat.idx[f]]]))$p.value;***

***}***

***else {***

***svdPVwb.m[c,f] <- summary(lm(tmp.v ~ PhenoTypesWB.lv[[selcat.idx[f]]]))$coeff[2,4];***

***}***

***}***

***print(c);***

***}***

***Heatmap1制作之***

**library(ggplot2)**

**library(reshape2)**

**library(scales)**

**library(plyr)**

**myPalette<-c("darkblue","blue","cyan","darkgreen","green","white","yellow","orange","purple","magenta","red");**

**melted\_sim <- melt(simdata$sim)**

**melted\_sim$Y1<-cut(melted\_sim$value,breaks = c(-Inf,-6:6,Inf),right = FALSE)**

**p <- ggplot(melted\_sim, aes(x=Var2, y=Var1)) + geom\_tile(aes(fill =Y1),colour = "transparent") + scale\_fill\_manual(values=myPalette)**

**p**

***Heatmap2制作***

**library(ggplot2)**

**library(reshape2)**

**melted\_sim <- melt(simdata$sim)**

**p <- ggplot(melted\_sim, aes(x=Var2, y=Var1)) + geom\_tile(aes(fill = value),colour = "transparent") + scale\_fill\_distiller(palette = "Greens")**

**“Spectral”**

***Heatmap制作3***

**library(ggplot2)**

**library(reshape2)**

**melted\_sim <- melt(simdata$sim)**

**colnames(melted\_sim)<-c("Genes","Arrays","value")**

**p <- ggplot(melted\_sim, aes(x=Arrays, y=Genes)) + geom\_tile(aes(fill = value),colour = "transparent") + scale\_fill\_gradientn(colours = c("blue", "white", "red"))+ theme\_bw()**

**p**

***Heatmap制作4***

**library(ggplot2)**

**library(reshape2)**

**library(scales)**

**library(plyr)**

**melted\_sim <- melt(simdata$sim)**

**melted\_sim <- ddply(melted\_sim,.(Var2),transform,rescale=rescale(value)) #，弄到0-1**

**colnames(melted\_sim)<-c("Genes","Arrays","value","rescale")**

**p <- ggplot(melted\_sim, aes(x=Arrays, y=Genes)) + geom\_tile(aes(fill = rescale),colour = "transparent") + scale\_fill\_gradientn(colours = c("green", "white", "darkred"))+ theme\_bw()**

**p**

***Heatmap5制作***

**library(ggplot2)**

**library(reshape2)**

**library(scales)**

**library(plyr)**

**#归一化**

**#simdataNorm<-(simdata$sim-rowMeans(simdata$sim))/(apply(simdata$sim,1,max)-apply(simdata$sim,1,min));**

**simdataNorm<-simdata$sim**

**melted\_sim <- melt(simdataNorm)**

**colnames(melted\_sim)<-c("Genes","Arrays","value")**

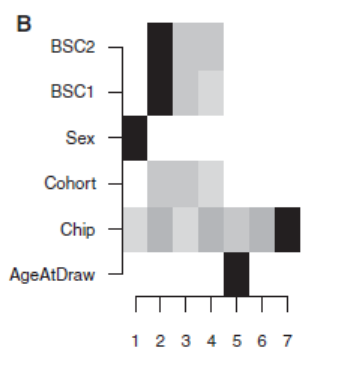
**p <- ggplot(melted\_sim, aes(x=Arrays, y=Genes)) + geom\_tile(aes(fill = value),colour = "transparent") + scale\_fill\_gradientn(colours = c("green", "white", "darkred"))+ theme\_bw()+**

**guides(fill=FALSE)+theme(axis.text=element\_text(size=13))**

**p**

# DoPCAWB1.R

**首先作每个分量与各个表型的lm，代码在DoPCAWB1.R,结果PCAsummaryBZz.pdf（P值做了log10(svdPVwb.m)）**



**找最相关的颜色越深**

**然后每个分量特征值平方做为大变化，即：**

**pdf("FracVarPCA-ZZ.pdf",width=4,height=3);**

**par(mar=c(5,4,2,1));**

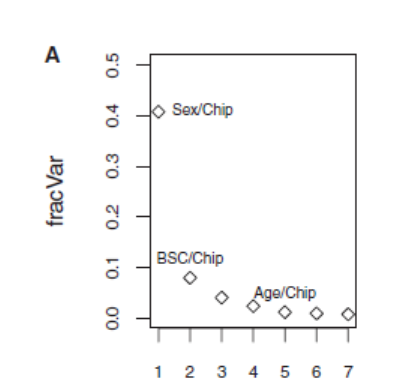
**tmp.v <- svd.o$d[1:topPCA];**

**f.v <- tmp.v^2/sum(svd.o$d^2);**

**plot(f.v,ylab="fVAR",xlab="PC",pch=23,type="b",lwd=2,col="red");**

**dev.off();**

**做出：**



**最后把相关最大的表型表上去**

**Chip、BSC于是被选出做CF！！**

# Lm.o各项GBM为例

**PhenoTypesList<-PhenoTypesTCGA\_GBM.lv**

**POI<-PhenoTypesTCGA\_GBM.lv$Cancer**

**CF<-PhenoTypesTCGA\_GBM.lv$age**

**lm.o<-lm(dvobj$dv ~ CF)**

**rsquared<-c()**

**fstatval<-c()**

**for(i in 1:dim(dvobj$dv)[2])**

**{**

**rsquared[i]<-summary(lm.o)[[i]][[8]]**

**fstatval[i]<-summary(lm.o)[[i]]$f[1]**

**}**

**sortedDVA<-sort(rsquared,decreasing = T,index.return = T)**

**> summary(lm.o)[[1]][1]**

**$call**

**lm(formula = Y1 ~ CF)**

**> summary(lm.o)[[1]][[1]]**

**lm(formula = Y1 ~ CF)**

**> summary(lm.o)[[2]][[1]]**

**lm(formula = Y2 ~ CF)**

**> summary(lm.o)[[1]][[2]]**

**Y1 ~ CF**

**attr(,"variables")**

**list(dvobj$dv, CF)**

**attr(,"factors")**

**CF**

**dvobj$dv 0**

**CF 1**

**attr(,"term.labels")**

**[1] "CF"**

**attr(,"order")**

**[1] 1**

**attr(,"intercept")**

**[1] 1**

**attr(,"response")**

**[1] 1**

**attr(,".Environment")**

**<environment: R\_GlobalEnv>**

**attr(,"predvars")**

**list(dvobj$dv, CF)**

**attr(,"dataClasses")**

**dvobj$dv CF**

**"nmatrix.14" "numeric"**

**summary(lm.o)[[1]][[3]] 为Residuals:残差向量**

**> (summary(lm.o)[[1]][[4]])**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) -0.0586410065 0.0144963824 -4.045217 9.164726e-05**

**CF 0.0005600044 0.0002363235 2.369652 1.936051e-02**

**> (summary(lm.o)[[1]][[5]])**

**(Intercept) CF**

**FALSE FALSE**

**> summary(lm.o)[[1]][[6]]**

**[1] 0.03370738**

**Residual standard error: 0.03371 on 123 degrees of freedom**

**> summary(lm.o)[[1]][[7]]**

**[1] 2 123 2**

**123 degrees of freedom**

**> summary(lm.o)[[1]][[8]]**

**[1] 0.04365929**

**Multiple R-squared: 0.04366, 为abs(cor(dvobj$dv, CF)) 的平方**

**> summary(lm.o)[[1]][[9]]**

**[1] 0.03588416**

**Adjusted R-squared: 0.03588**

**> summary(lm.o)[[1]][[10]]**

**value numdf dendf**

**5.615251 1.000000 123.000000**

**> summary(lm.o)[[1]][[10]][1]**

**value**

**5.615251**

**> summary(lm.o)[[1]][[10]][[1]]**

**[1] 5.615251**

**F-statistic: 5.615 on 1 and 123 DF,**

**> summary(lm.o)[[1]][[11]]**

**(Intercept) CF**

**(Intercept) 0.184956351 -2.949273e-03**

**CF -0.002949273 4.915454e-05**

**> summary(lm.o)[[1]][[4]][2,4]**

**[1] 0.01936051**

**为P值为以下黄色部分sort(rsquared,decreasing = FALSE,index.return = T) 从小到大排！**

**Pr(>|t|) 共有特征总数个P**

**WB\_pvalue.v <- isva.o$spv;**

**pdf("pvalue\_corrected\_hist2.pdf", width=22,height=11);**

**par(mfrow=c(1,2));**

**#par(mar=c(4,4,3,1));**

**#hist(BC\_pvalue.v);**

**hist(WB\_pvalue.v);**

**dev.off();**

**把cancer作为数量因子会导致P值变大**

**RealDoPCA.R为图1做法，仅使用PCA，SVD做出的结果**

**x1<-kruskal.test(tmp.v ~ as.factor(PhenoTypes.lv[[selcat.idx[f]]])) 对于标量**

**>x1[[1]] x1$statistic**

**Kruskal-Wallis chi-squared**

**0.02764977**

**> x1[[2]] x1$statistic**

**df**

**1**

**> x1[[3]] x1$p.value**

**[1] 0.8679349**

**> x1[[4]]**

**[1] "Kruskal-Wallis rank sum test"**

**> x1$method**

**[1] "Kruskal-Wallis rank sum test"**

**> x1[[5]]**

**[1] "tmp.v by as.factor(PhenoTypes.lv[[selcat.idx[f]]])"**

**> x1$data.name**

**[1] "tmp.v by as.factor(PhenoTypes.lv[[selcat.idx[f]]])"**

# Kruskal-Wallis单因子方差分析

**Source: http://www.r-bloggers.com/kruskal-wallis-one-way-analysis-of-variance/**

**如果你在进行多个群组之间比较时，因为群组不满足正态分布而不能使用ANOVA多比较，那么你可以使用Kruskal-Wallis检验。该检验类似于前面两个样本的Wilcox检验。**

**假设你想看看以下4个数值集合的均值是否统计相似：**

**Group A: 1, 5, 8, 17, 16**

**Group B: 2, 16, 5, 7, 4**

**Group C: 1, 1, 3, 7, 9**

**Group D: 2, 15, 2, 9, 7**

**为使用Kruskal-Wallis的检验，只要简单地输入数据，然后将它们再组织成一个list：**

**a = c(1, 5, 8, 17, 16)**

**b = c(2, 16, 5, 7, 4)**

**c = c(1, 1, 3, 7, 9)**

**d = c(2, 15, 2, 9, 7)**

**dati = list(g1=a, g2=b, g3=c, g4=d)**

**现在我们直接使用kruskal.test()函数：**

**kruskal.test(dati)**

**Kruskal-Wallis rank sum test**

**data: dati**

**Kruskal-Wallis chi-squared = 1.9217, df = 3, p-value = 0.5888**

**p-value大于0.05；并且检验统计的值1.9217也比chi-square的查表值低：**

**qchisq(0.950, 3)**

**[1] 7.814728**

**因此结论就是我们接受null hypothesis H0，即4个群组的均值统计相等。**

**Kruskal-Wallis 检测是利用多个样本的秩和来推断各样本分别代表的总体的位置有无差别，最后按所取检验水准作出推断结论。**

**Kruskal-Wallis检验又叫克鲁斯卡沃利斯测试当样本含量超过两个，达到三个或三个以上时称为多样本比较，用Kruskal-Wallis检验。**

**是一个关于三组或更多数据的非参数性测试。关于Kruskal-Wallis检测的测试统计量为H。**

**Kruskal-Wallis检验所用的统计量，是把k（k>2）个独立的简单随机样本的观察值放在一起，排列秩序后算出。其假设可陈述为：**

**Ho：k个总体是同分布的**

**H1：k个总体的分布不完全相同**

**见：Kruskal-Wallis秩和检验及其应用.田兵.包头师范学院《阴山学刊》编辑部**

**Real0.R全！**每一个CF都进行相关 real1.R是去除race中NA的样本

**Real0.R与Real1.R 的区别在于：**

**Real0.R是POI<-PhenoTypesList$Cancer，表示Cancer事先知道，**

**Real1.R 去除表型NA样本，POI<-PhenoTypesList$age，表示Cancer事先不知道**

**PhenoTypesTCGA\_BRCA.lv 要进行5次改换成其他癌症**

RealExprCorrRData0.R **Real1.R Real0.R**有RData不需要临时算，去掉以下5行注释就要dva，sva临时算

#dvobj<-dva(data.m,POI)

#mod = model.matrix(~as.factor(Cancer), data=data.frame(PhenoTypesList))

#mod0 = model.matrix(~1,data=data.frame(PhenoTypesList))

#n.sv = num.sv(data.m,mod,method="leek")

#svobj = sva(data.m,mod,mod0,n.sv=n.sv)

**RealDoPCA.R算Teschendorff图1**

**Heatmapggplot2.R 算SVD 的heatmap**

**heatmapMarray.R 用marray算heatmap**

RealExprHist.R是运用本方法后histo

**数据都在ORI里面：BRCA.RData是POI为Cancer， BRCA1.RData是POI为Age**

**CF<-PhenoTypesList$Sex**

**dva1 <- data.frame(dvobj$dv[,realDVA$sortcorreIX[1,1]],CF)**

**dva2 <- data.frame(dvobj$dv[,realDVA$sortcorreIX[2,1]],CF)**

**sva1 <- data.frame(svobj$sv[,realSVA$sortcorreIX[1,1]],CF)**

**sva2 <- data.frame(svobj$sv[,realSVA$sortcorreIX[2,1]],CF)**

**替换1为sex，2为race，3为age，4为cancer**

**POI<-PhenoTypesList$age 是为了找更多的潜在因子CFs如：race，sex之类的，作为Cancer的胁变量 Age为主要表型与癌症相关**

**Table 1.** The performance of DVA and SVA on DNA methylation Datasets 1-3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset 1 BRCA | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 0.88 | 0.01 (2) | 0.94 | 0.68 (19) |
| Race | 0.95 | 0.07 (2) | 0.89 | 0.33 (3) |
| Age | 0.19 | 0.06 (2) | 0.93 | 0.19 (10) |
| Dataset 2 COAD | SVA-P | SVA-R | DVA-P | DVA-R |
| Race | 0.74 | 0.86 (4) | 0.88 | 0.86 (1) |
| Sex | 0.34 | 0.25 (2) | 0.95 | 0.79 (9) |
| Age | 0.79 | 0.14 (3) | 0.98 | 0.13 (8) |
| Dataset 3ESCA | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | - | - |  | 0.63 (1) |
| Age | - | - |  | 0.21 (1) |
| Race | - | - |  | 0.11 (4) |

**2018.0808 real0.R**

**PhenoTypesList<-list(Cancer=PhenoTypesTCGA\_SKCM.lv$Cancer[pkyr.idx],Sex=PhenoTypesTCGA\_SKCM.lv$Sex[pkyr.idx],age= PhenoTypesTCGA\_SKCM.lv$age[pkyr.idx], Race=PhenoTypesTCGA\_SKCM.lv$Race[pkyr.idx])**

**第一列Cancer ，第二列Sex，age，race ，与原始PhenoTypesTCGA\_SKCM.lv不一样！**

**————————————————————————————————————————2018年8月8日改real0.R为real1.R更正了这个问题,另存为了real2.R**

**> corrDVA<-realDVA$sortcorre**

**> View(corrDVA)**

**> corrDVA[1,]**

**Sex Race Age Cancer**

**[1] 0.85075485 0.09908821 0.25775275 0.28758518**

**> corrDVAIX<-realDVA$sortcorreIX**

**> corrDVAIX[1,]**

**[1] 10 8 9 15**

**P不能用sort的 由corrDVAIX找corrDVAIX、dvaFstat[c,f]**

**dvapv<-realDVA$pv**

**dvapv[10,] [1] 4.427070e-128**

**dvapv[9,] 2.619922e-08**

**dvapv[8,]**

**0.035001813**

**> corrSVA<-realSVA$sortcorre**

**> View(corrSVA)**

**> corrSVA[1,]**

**[1] 0.09851819 0.01459026 0.13169701 0.01787345**

**> corrSVAIX<-realSVA$sortcorreIX**

**> corrSVAIX[1,]**

**[1] 2 2 2 2**

**svapv<-realSVA$pv**

**svapv[2,]**

**[1] 0.036068933 0.756793147 0.004992677 0.704395228**

**Table 1.** The performance of DVA and SVA on DNA methylation Datasets 1-3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset 1 (STAD) | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 0.361 | 0.05 (1) | 1e-167 | 0.94 (9) |
| Race | 0.018 | 0.12 (1) | 2e-10 | 0.33 (5) |
| Age | 1e-07 | 0.27 (2) | 1e-05 | 0.23 (7) |
| Dataset 2 (SKCM) | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 0.036 | 0.10 (2) | 4e-128 | 0.85 (10) |
| Age | 0.005 | 0.13 (2) | 3e-08 | 0.26 (9) |
| Race | 0.757 | 0.01 (2) | 0.035 | 0.10 (8) |
| Dataset 3 (COAD) | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 1e-83 | 0.86 (4) | 1e-81 | 0.86 (13) |
| Age | 0.017 | 0.14 (3) | 0.006 | 0.16 (7) |
| Race | 0.254 | 0.07 (5) | 0.140 | 0.09 (8) |
| Dataset 4 (READ) | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | − | − | 3e-40 | 0.91 (2) |
| Age | − | − | 0.044 | 0.20 (10) |

**DNAm文件夹是中间结果存放文件夹**

**ESCASeqV2.RData两者相当**

**LIHCSeqV2.RData DVA>>SVA, Sex**

**STADSeqV2.RData DVA>>SVA, Sex**

**COADSeqV2.RData DVA>>SVA, Sex**

**SKCM SeqV2.RData DVA>>SVA, Sex**

**LUAD SeqV2.RData DVA>>SVA, Sex**

**LSCC SeqV2.RData DVA>>SVA, Sex 非常好**

**KIRC\_RNAseqV2 非常好**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset 8LUAD | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 3e-50 | 0.62 (5) | 0 | 0.98 (16) |
| Age | 5e-03 | 0.13 (38) | 3e-04 | 0.17 (2) |
| Race | 8e-05 | 0.18 (20) | 4e-02 | 0.09 (12) |
| Dataset 5KIRC | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 3e-44 | 0.54 (9) | 1e-224 | 0.91 (11) \* |
| Age | 2e-05 | 0.18 (9) | 8e-04 | 0.14 (11) |
| Race | 1e-07 | 0.22 (17) | 1e-03 | 0.13 (11) |
| Dataset 6STAD | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 1e-12 | 0.35 (10) | 5e-77 | 0.77 (4) |
| Age | 7e-04 | 0.17 (19) | 2e-03 | 0.16 (10) |
| Race | 7e-06 | 0.23 (27) | 1e-02 | 0.13 (13) |
| Dataset 7LIHC | SVA-P | SVA-R | DVA-P | DVA-R |
| Sex | 3e-28 | 0.52 (10) | 3e-105 | 0.84 (9) |
| Age | 4e-08 | 0.27 (15) | 1e-07 | 0.26 (4) |
| Race | 4e-09 | 0.29 (15) | 2e-05 | 0.21 (9) |