

enrichR的使用方法讲解

转录组可视化包(enrichR)

Release date: 2021/07/13

Version: V0.0.2.0

enrichR

安装依赖包和本包

```
if(!require(ggplot2))install.packages("ggplot2") if(!require(tidyverse))install.packages("tidyverse")
if(!require(ggprism))remotes::install_github("csdaw/ggprism")
if(!require(devtools))install.packages("devtools")
if(!require(ggvenn))devtools::install_github("yanlinlin82/ggvenn")
if(!require(UpSetR))install.packages("UpSetR")
if(!require(clusterProfiler))install.packages("clusterProfiler")
if(!require(ggord))install_github('fawda123/ggord')
if(!require(grid))remotes::install_github("thomasp85/grid")
if(!require(ggnewscale))devtools::install_github("eliocamp/ggnewscale")
```

下载压缩包后，本地安装

```
install.packages("enrichR_0.0.2.0.tar.gz", repos = NULL, type = "source")
```

直接从github安装

```
install.packages("devtools")
```

```
devtools::install_github("chaimol/enrichR")
```

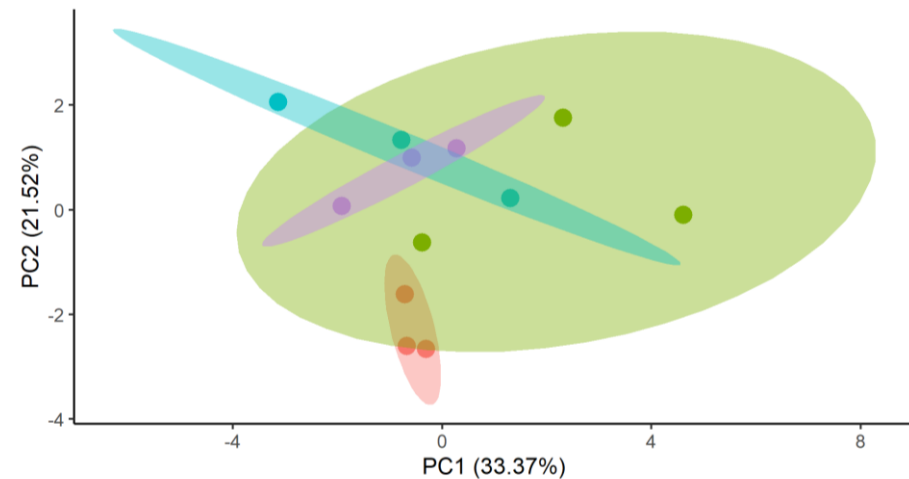
enrichR

PCA可视化

```
read_counts <-  
read.delim("all_raw_counts.txt",sep="\t",header =  
TRUE,row.names = 1)
```

```
group_level <- substr(colnames(read_counts),1,2)
```

```
get_pca(read_counts, group_level,groupname="WTvsMT")
```

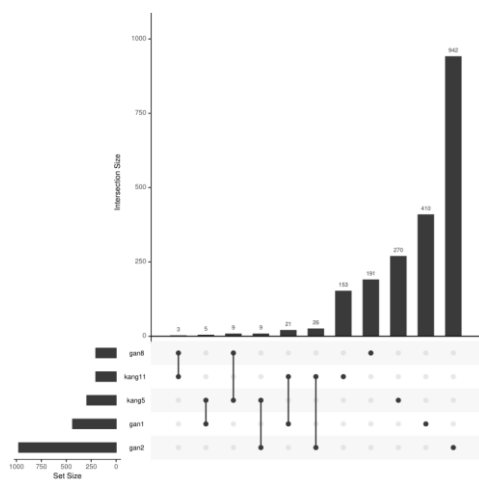
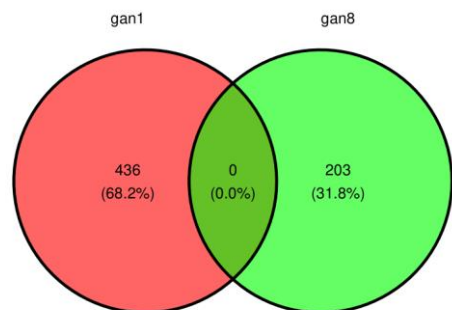
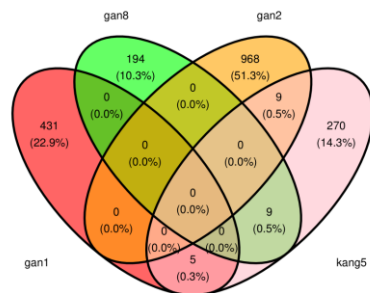
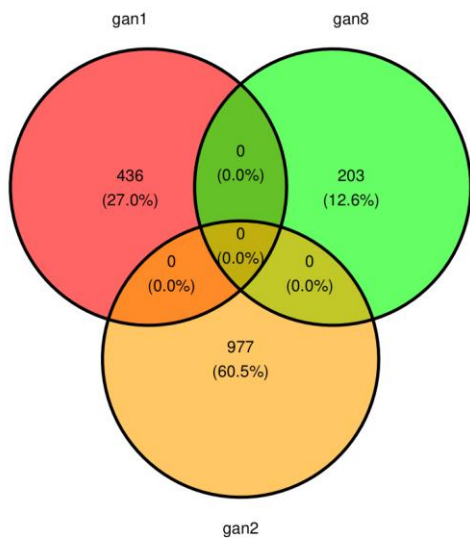


Groups

- CK
- L1
- L2
- L3

```
> head(read_counts)
```

	CK1	CK2	CK3	L1_1	L1_2	L1_3	L2_1	L2_2	L2_3	L3_1	L3_2	L3_3
Sp01G000010	278	248	305	316	342	367	259	290	279	252	305	328
Sp01G000020	96	67	91	59	51	77	51	81	81	68	65	77
Sp01G000030	146	174	214	164	209	227	136	170	147	147	136	144
Sp01G000040	0	0	0	0	0	0	0	0	0	0	0	0
Sp01G000050	48	46	38	52	43	34	17	35	32	35	18	30
Sp01G000060	11	8	8	10	3	4	7	8	0	11	5	6



enrichR Venn图

```
library(enrichR)
```

```
venn <- read.csv("venn.csv",header = T)
```

###Venn图（Venn图只能输出2-4组的维恩图，大于4则只输出upset图）

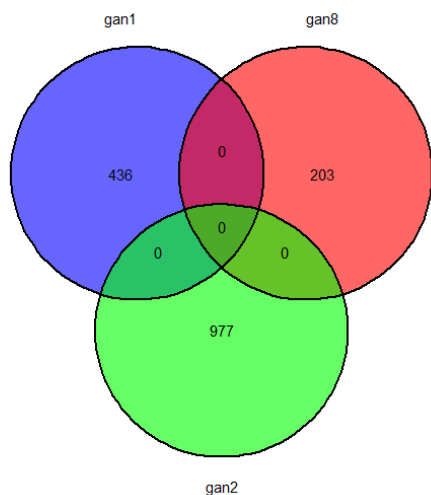
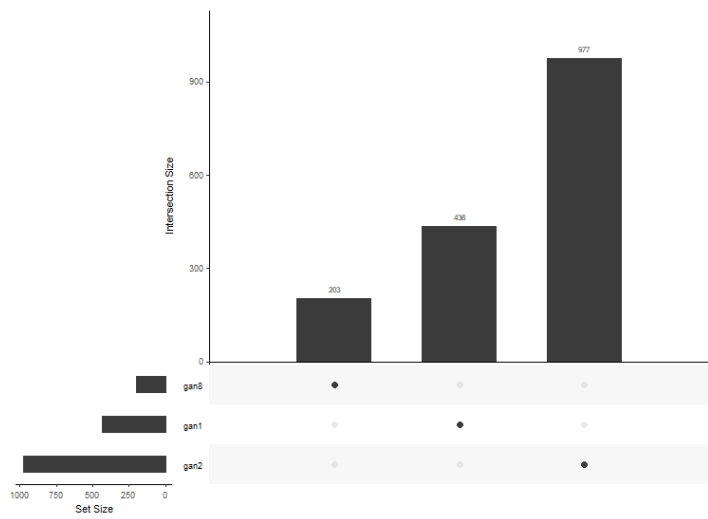
```
gene2venn(venn[1:3],"demo3")
```

```
gene2venn(venn[1:4],"demo4")
```

```
gene2venn(venn[1:2],"demo2")
```

```
> head(venn)
```

	gan1	gan8	gan2	kang5	kang2	kang11
1	MELO3C000147	MELO3C001026	MELO3C000019	MELO3C000053	MELO3C002102	MELO3C000200
2	MELO3C000258	MELO3C002014	MELO3C000030	MELO3C000062	MELO3C002107	MELO3C000273
3	MELO3C000268	MELO3C002032	MELO3C000298	MELO3C000087	MELO3C002128	MELO3C000515
4	MELO3C000398	MELO3C002033	MELO3C000376	MELO3C000115	MELO3C002243	MELO3C000527
5	MELO3C000512	MELO3C002183	MELO3C000378	MELO3C000138	MELO3C002264	MELO3C000682
6	MELO3C000538	MELO3C002253	MELO3C000599	MELO3C000296	MELO3C002441	MELO3C000836



enrichR

Venn进阶用法

#进阶用法(需要自行保存图片)

```
data2 <- get_matrix(venn[1:3])
```

```
get_venn(data2)
```

```
get_venn(data2,percentage = FALSE,fill_color =  
c("#FC8D62","#8DA0CB","#E78AC3")) #可以自定义颜色
```

```
get_venn(data2,percentage = FALSE,fill_color =  
c("blue","red","green"))#是否显示百分比
```

```
data2$geneID <- rownames(data2)
```

```
upset(data2,decreasing = c(FALSE,TRUE))#获取upset图
```

GO.info格式

```
GO:0016020 → membrane → MELO3C000003 → cellular_component
GO:0016021 → integral_component_of_membrane → MELO3C000003 → cellular_component
GO:0000166 → nucleotide_binding → MELO3C000009 → molecular_function
GO:0005524 → ATP_binding → MELO3C000009 → molecular_function
GO:0006810 → transport → MELO3C000009 → biological_process
GO:0016020 → membrane → MELO3C000009 → cellular_component
GO:0016021 → integral_component_of_membrane → MELO3C000009 → cellular_component
GO:0016887 → ATPase_activity → MELO3C000009 → molecular_function
```

KEGG.info格式

```
K05658 → ATP-binding_cassette_subfamily_B_(MDR/TAP),_member_1 → MELO3C000009
K02218 → casein_kinase_1 → MELO3C000093
K16297 → serine_carboxypeptidase-like_clade_II → MELO3C000105
K14488 → SAUR_family_protein → MELO3C000106
K02992 → small_subunit_ribosomal_protein_S7 → MELO3C000107
K14272 → glutamate--glyoxylate_aminotransferase → MELO3C000117
K09580 → protein_disulfide-isomerase_A1 → MELO3C000199
K15382 → solute_carrier_family_50_(sugar_transporter) → MELO3C000200
K04079 → molecular_chaperone_HtpG → MELO3C000208
K04079 → molecular_chaperone_HtpG → MELO3C000209
K00026 → malate_dehydrogenase → MELO3C000258
K04797 → prefoldin_alpha_subunit → MELO3C000273
K14272 → glutamate--glyoxylate_aminotransferase → MELO3C000280
K14272 → glutamate--glyoxylate_aminotransferase → MELO3C000284
```

TF.info格式

```
RAV → related_to_ABI3/VP1.2 → MELO3C012217 → AT1G68840.2
RAV → related_to_ABI3/VP1.1 → MELO3C003002 → AT1G13260.1
RAV → RAV_family_protein → MELO3C015667 → AT1G50680.1
RAV → RAV_family_protein → MELO3C029441 → AT1G50680.1
WRKY → WRKY_DNA-binding_protein_23 → MELO3C000030 → AT2G47260.1
LBD → LOB_domain-containing_protein_18 → MELO3C000068 → AT2G45420.1
LBD → LOB_domain-containing_protein_4 → MELO3C000076 → AT1G31320.1
YABBY → YABBY_family_protein → MELO3C000087 → AT2G45190.1
ZF-HD → homeobox_protein_30 → MELO3C000099 → AT5G15210.1
```

enrichR 富集分析 GO和KEGG,TF

#每一列都可以单独进行各种富集分析

#参数1是基因列，参数2是输出的文件前缀名，参数3是对应的数据库，需要自行下载对应物种的数据库

#GO,KEGG,TF数据格式参考 data目录里的格式

```
go1 <- GOenrich(venn$kang5,"kang5",GOfile = "data/GO.info")
```

```
kegg1 <- KEGGenrich(venn$gan8,"gan8","data/KEGG.info")
```

```
tf1 <- TFenrich(venn$kang2,"kang2","data/TF.info")
```

参数1：基因列表

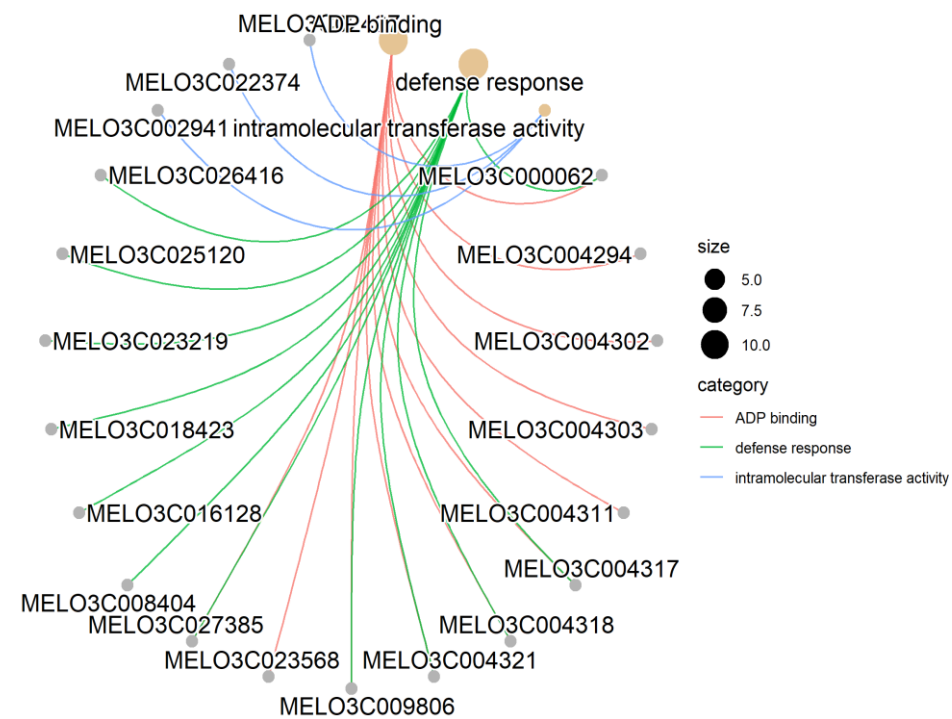
参数2：输出文件名前缀

参数3：富集种类的数据库文件(要求必须使用tab分隔符)

GO.info 需要4列

KEGG.info 需要3列

TF.info 需要3列或4列（只使用前三列）



富集分析结果

输出的文件

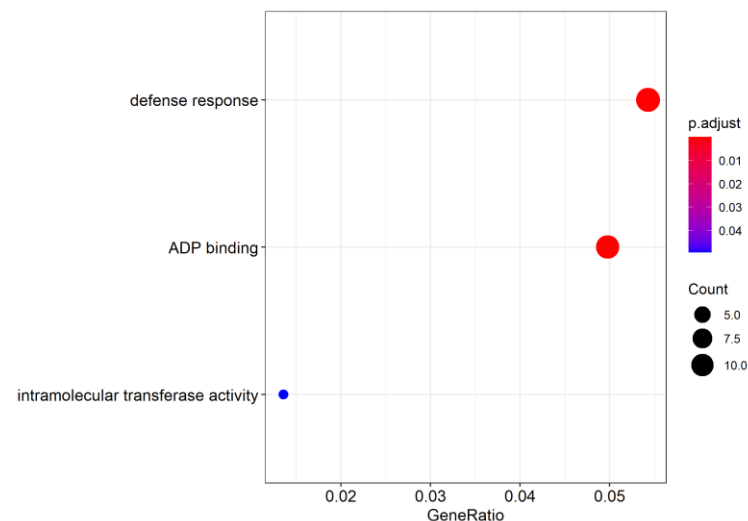
kang5ALL_GO.csv

kang5ALL.GO.tiff

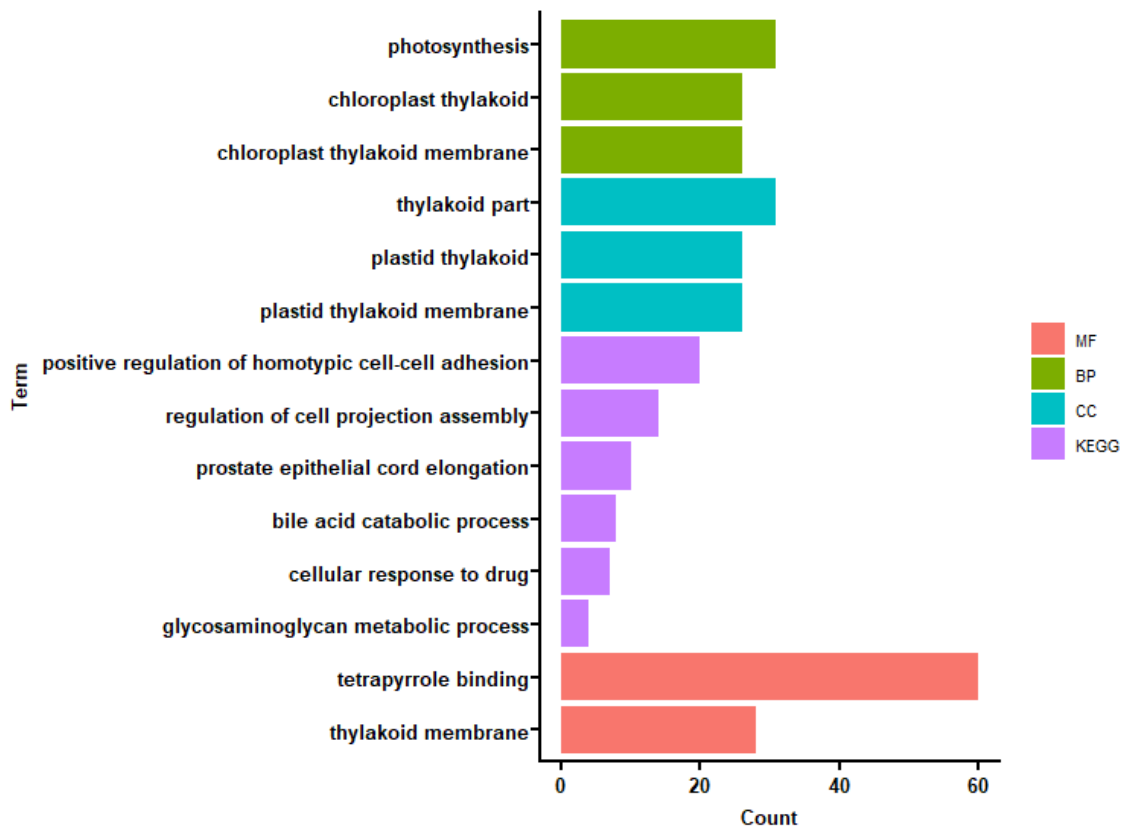
kang5ALL.GO.pdf

kang5ALL.GO.circle.tiff

kang5ALL.GO.circle.pdf



ID	Description	GeneRatio	BgRatio	pvalue	p.adjust	qvalue	geneID	Count
GO:0043531	ADP binding	11/221	127/22274	5.81E-08	1.03E-05	9.54E-06	MELO3C000062/MELO3C000062	11
GO:0006952	defense response	12/221	200/22274	8.03E-07	7.15E-05	6.59E-05	MELO3C000062/MELO3C000062	12
GO:0016866	intramolecular transferase activity	3/221	19/22274	0.000830277	0.049263121	0.045446759	MELO3C002941/MELO3C002941	3



```
> head(dat1, 30)
  ID Description count type
1 GO:0046906 tetrapyrrole binding 60 MF
2 GO:0015979 photosynthesis 31 BP
3 GO:0009535 chloroplast thylakoid membrane 26 BP
4 GO:0055035 plastid thylakoid membrane 26 CC
5 GO:0044436 thylakoid part 31 CC
6 GO:0042651 thylakoid membrane 28 MF
7 GO:0009534 chloroplast thylakoid 26 BP
8 GO:0031976 plastid thylakoid 26 CC
9 KO:0034112 positive regulation of homotypic cell-cell adhesion 20 KEGG
10 KO:0060491 regulation of cell projection assembly 14 KEGG
11 KO:0060523 prostate epithelial cord elongation 10 KEGG
12 KO:0030573 bile acid catabolic process 8 KEGG
13 KO:0035690 cellular response to drug 7 KEGG
14 KO:0030203 glycosaminoglycan metabolic process 4 KEGG
```

enrichR barplot

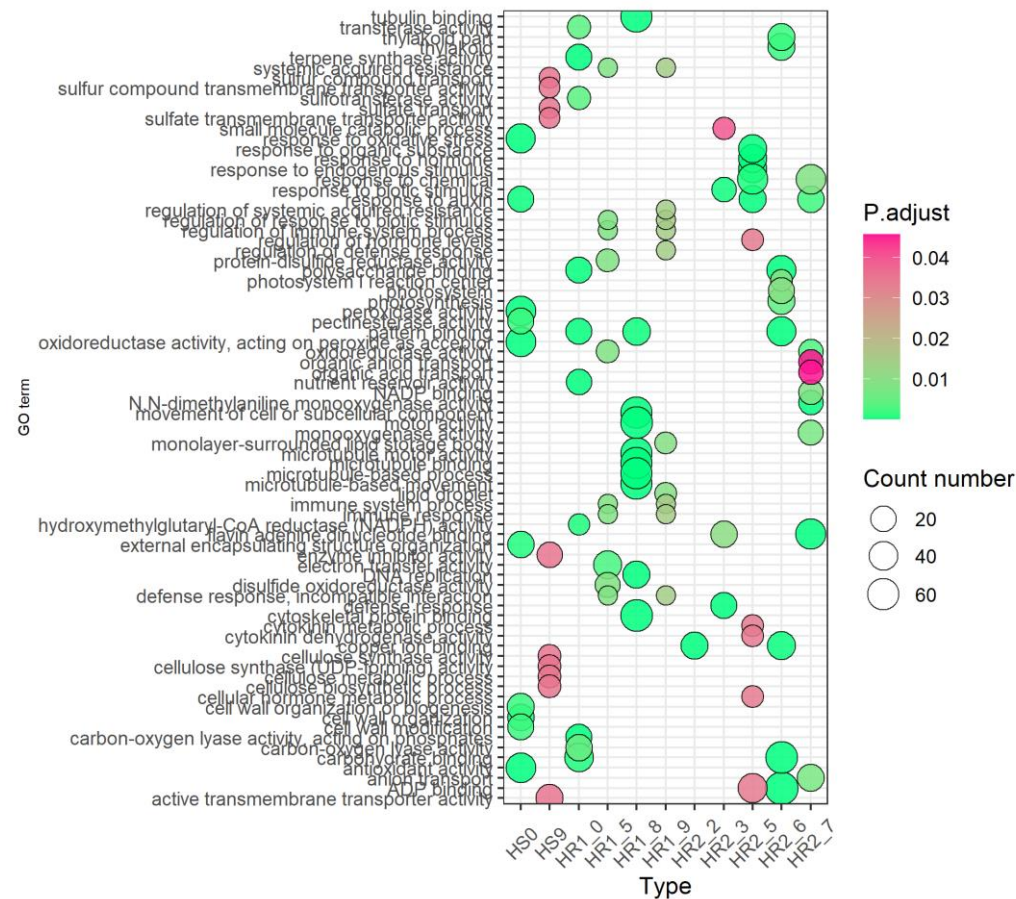
#barplot绘制多分组的柱状图（示例数据是:dat1）

```
barplot(dat1)
```

```
barplot(dat1,term_num =
6,fill_level=c("MF","BP","CC","KEGG","TF"),term
_width=55)
```

```
barplot(dat1,term_num =
8,fill_level=c("GO","KEGG","TF"),term_width=55)
```

```
barplot(dat1,term_num =
6,fill_level=c("MF","BP","CC"))
```

dotplot: 气泡图

```
load("data/data_GO_all.Rdata")
```

```
dotplot(data_GO_all,xlabels=xlabels,output="All_GO")
```

```
dotplot(data_GO_all,xlabels=xlabels,output="All_GO",range=c(3,5))
```

参数1: 数据框, 需要包含列名
"type","Description","adjust","Count"

参数2: 向量, x轴的标签

参数3: 字符串, 输出文件的前缀

参数4: 数字向量, 用于控制气泡的缩放大小比例