第七次实验作业

7 JUNE 2021

> 实验背景
>
> > GO 富集分析（Gene Ontology Enrichment Analysis）是对于分析出差异表达的基因作功能注释的常用研究方法，而癌细胞在面临不同压力条件时与正常体细胞或其他癌种也可能存在一定差异。故我们可以利用现有的研究数据，将同癌种的显著差异基因与其他癌种或正常体细胞相比较，以期揭露特定癌种细胞与其他类型细胞的质的区别。
>
> 实验目的
>
> > 通过多组实验数据的整合与宏分析，发现特定癌种细胞中较为敏感或易变的功能通路
>
> 运行环境（见下）

```
platform          x86_64-pc-linux-gnu
arch              x86_64
os                linux-gnu
system            x86_64, linux-gnu
R version         4.1.0 (2021-05-18)
nickname          Camp Pontanezen
radian version    0.5.11
python version    3.7.3
xonsh version     0.9.27
```

癌细胞表达量宏分析

实验方法

下载 EBI 上公开的研究数据（共 210 份研究），筛选与癌症条件相关研究，挑选出同癌种各显著差异基因，进行 GO 富集分析。

实验流程

数据下载

由于 EBI 未提供批量下载的选项，因此使用 `Python` 并行尝试下载各项目页面下文件链接。

```
import pandas as pd
import requests
import urllib.request
from bs4 import BeautifulSoup
import os
import multiprocessing


# temp.tsv 同样为 Python 生成，较为简单故不作展示，格式如下
# MainPage  DownloadPage    Title
# <proj_url>/Results    <proj_url>/Downloads    <proj_title>
df = pd.read_table("./temp.tsv")
ExpID = [link.split("/")[5] for link in df.iloc[:, 1]]
ExpTitle = [title for title in df.iloc[:, 2]]


def download(url, filename, savepath, alternate):
    blockSize = 1024 * 1024
    try:
        resp = urllib.request.urlopen(url)
        with open(f"{savepath}/{filename}", "wb") as f:
            while True:
                buffer = resp.read(blockSize)
                if not buffer:
                    break
                f.write(buffer)
    except Exception:
        if alternate:
            try:
                resp = urllib.request.urlopen(alternate)
                with open(f"{savepath}/{filename}", "wb") as f:
                    while True:
                        buffer = resp.read(blockSize)
                        if not buffer:
                            break
                        f.write(buffer)
            except Exception:
                print(f"{filename} download failed")
                return False
        else:
            return False
    return True


def downloadExperimentData(ID):
    if not os.path.isdir(f"data/{ID}"):
        os.mkdir(f"data/{ID}")
    status = {
        "ExpressionResult.tsv": True,
        "Analytics.tsv": True,
        "RawCounts-NormExpression.tsv": True,
        "Metadata.tsv": True,
    }
    urls = {
        "ExpressionResult.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/

        "Analytics.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/resourc

        "RawCounts-NormExpression.tsv": f"https://www.ebi.ac.uk/gxa/experiments-conte

        "Metadata.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/resource

    }
```

```
    alt = {
        "ExpressionResult.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/

        "Analytics.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/resource

        "RawCounts-NormExpression.tsv": f"https://www.ebi.ac.uk/gxa/experiments-conter

        "Metadata.tsv": f"https://www.ebi.ac.uk/gxa/experiments-content/{ID}/resources

    }
    processPool = []
    for fileName in status.keys():
        if not os.path.isfile(f"data/{ID}/{fileName}"):
            processPool.append(
                multiprocessing.Process(
                    target=download,
                    args=(urls[fileName], fileName, f"data/{ID}", alt[fileName]),
                )
            )
    for proc in processPool:
        proc.start()
    for proc in processPool:
        proc.join()
    return


if not os.path.isdir("data"):
    os.mkdir("data")
for ID, title in zip(ExpID, ExpTitle):
    print(f"https://www.ebi.ac.uk/gxa/experiments/{ID}/Downloads")
    print(f"{ID} started downloading")
    with open(f"data/{ID}/Info.log", "w") as f:
        f.write(title)
    downloadExperimentData(ID)

# # 除去以上手动下载方法，以下可输出所有链接
# # 可供 aria2c 等成熟的下载软件使用，对于较差的网络连接更为有效
# urlRoot = "https://www.ebi.ac.uk/gxa/experiments-content/"
# resources = [
#     "/resources/ExperimentDownloadSupplier.RnaSeqDifferential/tsv",
#     "/resources/DifferentialSecondaryDataFiles.RnaSeq/analytics",
#     "/resources/DifferentialSecondaryDataFiles.RnaSeq/raw-counts",
#     "/resources/ExperimentDesignFile.RnaSeq/experiment-design",
#     "/resources/ExperimentDownloadSupplier.Microarray/query-results",
#     "/resources/DifferentialSecondaryDataFiles.Microarray/analytics",
#     "/resources/DifferentialSecondaryDataFiles.Microarray/normalized-expressions",

#     "/resources/ExperimentDesignFile.Microarray/experiment-design",
# ]
# print(
#     "\n".join(
#         ["\n".join([urlRoot + ID + resource for resource in resources]) for ID in Ex

#     )
# )
```

数据过滤

为了比较癌种细胞的差异表达，我们实际上需要研究中同时包含正常体细胞作比对才可进行有效的统计检验。然而实际上，同时包含了正常体细胞的研究少之又少，大多研究为癌细胞在不同压力条件下或 miRNA 处理下的基因差异表达，因此我们只能分析不同癌种细胞间，各通路的易变性是否有显著差异。

```
import os
import pandas as pd
import numpy as np

allExpDirList = [
    os.path.join("data", dirName)
    for dirName in os.listdir("data")
    if os.path.isdir(os.path.join("data", dirName))
]
# 总项目数
len(allExpDirList)
# 210
expDirList = allExpDirList
totalCancerCount = set()

for dirPath in expDirList:
    fileList = [
        os.path.join(dirPath, fileName) for fileName in sorted(os.listdir(dirPath))

    ]
    designFilePath = [filePath for filePath in fileList if "experiment" in filePath]

    design = pd.read_table(designFilePath[0])
    if design.filter(regex="[^m]\[disease\]").shape[1] == 0:
        continue
    totalCancerCount = totalCancerCount.union(
        set(design.filter(regex="[^m]\[disease\]").iloc[:, 0])
    )

# 涉及到的癌种
print(len(totalCancerCount))
# 124
```

尽管此处结果展示有 124 种癌种，实际上包含了部分癌种的不同称呼，以及部分癌种的不同时期。此处应当需要将同属一个癌种的数据进行整合，但由于时间原因，故此部分未作人工校对。

在数据筛选的部分，我们首先需要确定筛选方法。参考相关推荐做法[1][2][3][4][5]，最终选择采用直接同一校正后 p-value，并结合变化倍率的绝对值筛选单个项目内显著差异的基因[6]，同一癌种的差异基因采用直接取并集的方法。

```
expDirList = allExpDirList
# 显著性阈值
pValueThreshold = 0.01
# 变化倍率阈值
logFoldChangeThreshold = 1
cancerCount = {}
filteredOut = []
cancerSpecificSignificantGenes = {}
cancerTypeAnnotation = {}

for dirPath in expDirList:
    fileList = [
        os.path.join(dirPath, fileName) for fileName in sorted(os.listdir(dirPath))

    ]
    designFilePath = [filePath for filePath in fileList if "experiment" in filePath]

    design = pd.read_table(designFilePath[0])
    analyticsFilePath = [filePath for filePath in fileList if "analytic" in filePath]

    analytics = pd.read_table(analyticsFilePath[0]).fillna(1)
    analytics = analytics.set_index("Gene Name")
    # 研究所用样本无癌细胞，则排除
    if design.filter(regex="[^m]\[disease\]").shape[1] == 0:
        filteredOut.append(dirPath)
```

```
            continue
        significantGenes = set(
            analytics.iloc[
                np.all(
                    [
                        np.any(
                            analytics.filter(regex=r"p-value$")
                            < pValueThreshold / len(analytics.index),
                            axis=1,
                        ),
                        np.any(
                            np.abs(analytics.filter(regex=r"log2foldchange"))
                            > logFoldChangeThreshold,
                            axis=1,
                        ),
                    ],
                    axis=0,
                ),
                :,
            ].index
        )
        # 研究所得无显著差异基因，则排除
        if len(significantGenes) == 0:
            filteredOut.append(dirPath)
            continue
        # 将差异表达基因合并到对应的癌种
        for cancerType in design.filter(regex="[^m]\[disease\]").iloc[:, 0]:
            cancerCount[cancerType] = cancerCount.get(cancerType, 0) + 1
        cancerTypeAnnotation[os.path.basename(dirPath)] = design.filter(
            regex=r"Sample Characteristic\[disease\]"
        )
        for cancerType in design.filter(regex="[^m]\[disease\]").iloc[:, 0]:
            cancerSpecificSignificantGenes[cancerType] = cancerSpecificSignificantGenes.ge

                cancerType, set()
            ).union(significantGenes)

    # 条件筛选后涉及到的癌种
    print(len(cancerCount))
    # 104

    # 将不符合要求的项目移出路径列表
    for each in filteredOut:
        expDirList.remove(each)
```

为了方便 `Python` 和 `R` 之间数据衔接，故采用 json 格式来存储二次筛选后的结果

```python
import json

resultSignificantGenes = [
    {
        "CancerName": cancerName,
        "CaseCount": cancerCount[cancerName],
        "GenesCount": len(sigGene),
        "SigDiffGenes": list(sigGene),
    }
    for cancerName, sigGene in cancerSpecificSignificantGenes.items()
    # 此处额外的筛选条件为同癌种样本数大于 10 且显著差异基因数也大于 10
    if cancerCount[cancerName] > 10 and len(sigGene) > 10
]

# 得到的数据存于 genes.json 文件种
with open("genes.json", "w") as f:
    json.dump(resultSignificantGenes, f, indent = 4)
    # json 格式必需加此换行，否则 R 中无法正确读取
    f.write("\n")
```

GO 富集分析

由于所找到的相关宏分析结果较少，且时间有限，故此处仅选取其中一个癌种作 GO 富集分析。

```r
library(rjson)
library(clusterProfiler)
library(org.Hs.eg.db)

json_file <- "genes.json"
json_data <- fromJSON(paste(readLines(json_file), collapse = ""))

pValueCutOff <- 0.05
eGO <- enrichGO(
    gene = json_data[[1]]$SigDiffGenes,
    OrgDb = org.Hs.eg.db,
    keyType = "SYMBOL",
    ont = "BP",
    pAdjustMethod = "BH",
    pvalueCutoff = pValueCutOff,
    qvalueCutoff = 2 * pValueCutOff
)
GOplot <- plotGOgraph(eGO)
```

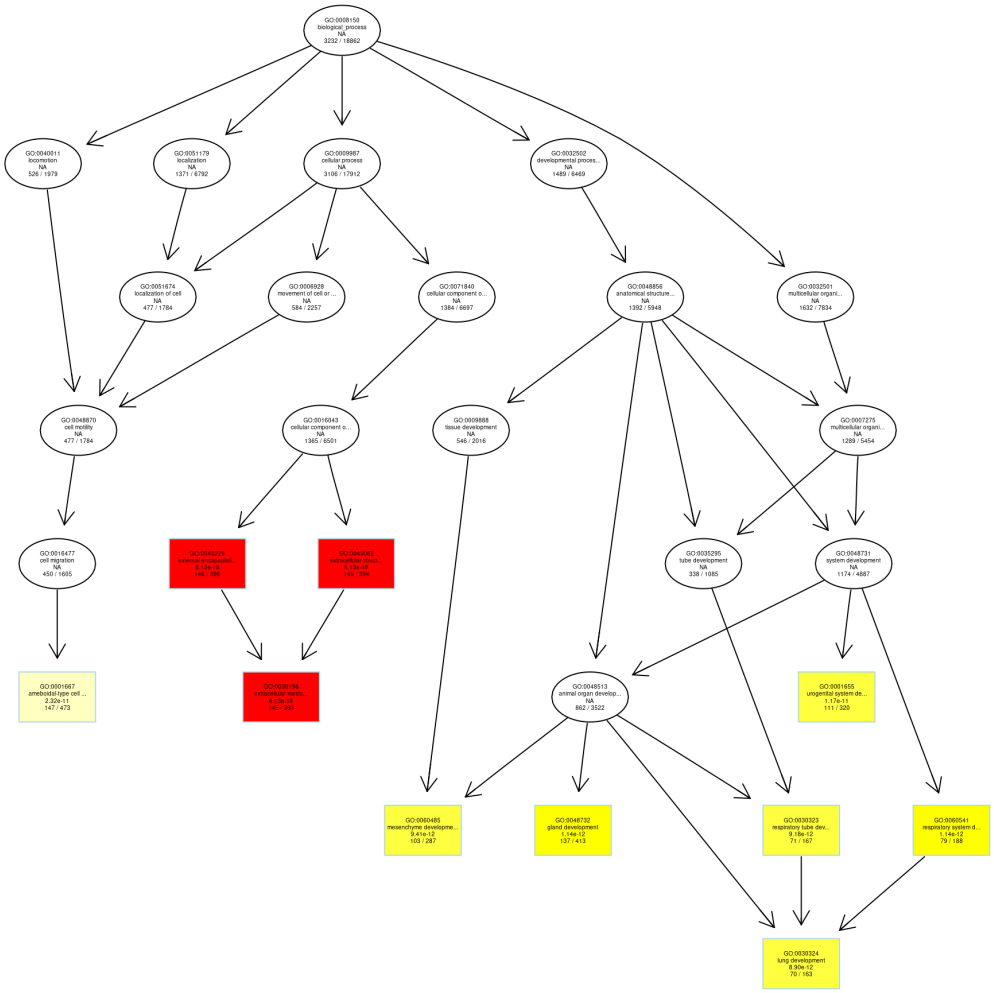

图 1: BREAST CANCER 癌种显著差异表达基因 GO 富集分析结果

由图 1结果可见，breast cancer细胞中最易变通路功能主要富集在胞外基质组织和结构形成、肺部及呼吸管道与发育、腺体发育、间充质发育、泌尿系统发育、变形型细胞迁移。胞外基质与结构的形成可能与癌细胞的迁移有关，而其他富集注释可能与乳腺癌细胞分化路径以及发育时形态位置有关。

结论

从富集分析结果可以看出，如此分析流程得出的结果可解释性较差，且可能与其他癌种注释结果类似。很大一部分原因可能来自于所用数据集中缺乏空白对照，无法分析癌细胞与正常细胞差异，而各类处理方法各异，不同的结果间重叠较小所导致，因此实际上在第一步质控时应为此类分析的关键，且很难避免人工仔细阅读实验设计乃至文章正文，因此如何改进此方面的自动化宏数据注释与整合会是宏分析效率与质量提升的关键。

参考资料

[1] SUSZYNSKA M, KLONOWSKA K, JASINSKA A J, 等. Large-scale meta-analysis of mutations identified in panels of breast/ovarian cancer-related genes Providing evidence of cancer predisposition genes[J/OL]. Gynecologic Oncology, 2019, 153(2): 452–462. https://doi.org/10.1016%2Fj.ygyno.2019.01.027. DOI:10.1016/j.ygyno.2019.01.027.

[2] LIU C, CHANG H, LI X-H, 等. Network Meta-Analysis on the Effects of DNA Damage Response-

Related Gene Mutations on Overall Survival of Breast Cancer Based on TCGA Database[J/OL]. Journal of Cellular Biochemistry, 2017, 118(12): 4728–4734. https://doi.org/10.1002%2Fjcb.26140. DOI:10.1002/jcb.26140.

[3] O'MARA T A, ZHAO M, SPURDLE A B. Meta-analysis of gene expression studies in endometrial cancer identifies gene expression profiles associated with aggressive disease and patient outcome[J/OL]. Scientific Reports, 2016, 6(1). https://doi.org/10.1038%2Fsrep36677. DOI:10.1038/srep36677.

[4] TSENG G C, GHOSH D, FEINGOLD E. Comprehensive literature review and statistical considerations for microarray meta-analysis[J/OL]. Nucleic Acids Research, 2012, 40(9): 3785–3799. https://doi.org/10.1093%2Fnar%2Fgkr1265. DOI:10.1093/nar/gkr1265.

[5] KIM K-Y, KI D, JEONG H, 等. Novel and simple transformation algorithm for combining microarray data sets[J/OL]. BMC Bioinformatics, 2007, 8(1): 218. https://doi.org/10.1186%2F1471-2105-8-218. DOI:10.1186/1471-2105-8-218.

[6] JAFARI M, ANSARI-POUR N. Why, When and How to Adjust Your P Values?[J/OL]. Cell J (Yakhteh), 2018, 20(04). https://doi.org/10.22074/cellj.2019.5992. DOI:10.22074/cellj.2019.5992.