Cancer Personalized Drug Recommendation (CPDR)

Ruzhen Chen

2022-04-17

Table of Contents

Ruzhen Chen

Package overview

Due to cancer heterogeneity, only some patients can benefit from drug therapy. The personalized drug use is important for improving the treatment response rate of cancer patients. Patient transcriptome has recently demonstrated its value in guiding personalized drug use, and Connectivity map (CMAP) is a reliable computational approach for drug recommendation. However, there is still no personalized drug recommendation tool based on patient transcriptomic profiles and CMAP. To fill this gap, here we proposed such a feasible workflow and a user-friendly R package - Cancer Personalized Drug Recommendation (CPDR).

CPDR has three features. 1) It identifies the individual disease signature by using the patient subgroup with transcriptomic profiles similar to that of the input patient. 2) Transcriptomic profile purification is supported for the subgroup with high infiltration of non-cancerous cells. 3) It supports in silico drug efficacy assessment using drug sensitivity data of cancer cell lines.

Workflow

We demonstrated the pipeline of CPDR with the aid of a dataset from GEO (GSE164541), containing gene expression profiles of 5 patients with colorectal cancer.

1. Data download

1.1 Clinical patients' RNA-seq count data (GSE164541)

```
library(CPDR)
library(GEOquery)
library(org.Hs.eg.db)

getGEOSuppFiles('GSE164541', makeDirectory = TRUE, fetch_files = TRUE)
clinical <- read.csv("./GSE164541/GSE164541_ANT_count.csv.gz")
clinical <- inner_join(clinical,
                       bitr(clinical$ENSEMBL,
                            fromType = "ENSEMBL",
                            toType = "SYMBOL",
                            OrgDb = org.Hs.eg.db),by = "ENSEMBL")
clinical <- clinical[!duplicated(clinical$SYMBOL),]
row.names(clinical) <- clinical$SYMBOL
clinical_profile_set <- clinical[,grep("PT",colnames(clinical))]

head(clinical_profile_set)
```

Table1: Clinical patients' RNA-seq count data

	PT1	PT2	PT3	PT4	PT5
DDX11L2	0	0	0	0	0
DDX11L16	0	0	0	0	0
DDX11L1	0	0	0	0	0
DDX11L5	0	0	0	0	0
DDX11L17	0	0	0	0	0
WASH7P	45	47	14	22	3

1.2 Background data

```
download_db(pset = c('CCLE','PRISM'), # pharmacogenetic data
            tset = 'coadread',        # TCGA data. View(CPDR::TCGA_sets) for a
vailable cancer type
            nset = 'GTEX',            # GTEX data
            saveDir = '.',
            verbose = T)

# read downloaded TCGA dataset
exdir <- cBioPortalData::untarStudy("./CPDR_db/TCGA/
                                    25dc53d4417_coadread_tcga_pan_can_atlas_20
18.tar.gz")
coadread <- cBioPortalData::loadStudy(exdir)
```

2. Identification of individual disease signals

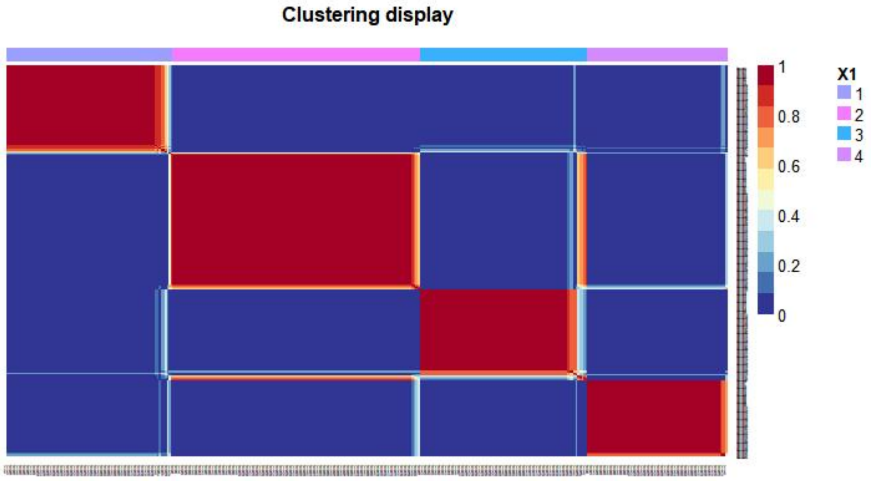
2.1 Preprocessing

```
pmat = select_db(Assay = coadread,
                 cmat = clinical_profile_set,
                 removeBatchEffect = TRUE,
                 OrgDb = org.Hs.eg.db,
                 minSampleSize = 10,
                 MSI_status = NULL,
                 driver_gene = NULL,
                 MUT_status = NULL,
                 CNA_status = NULL)
gc()
```

2.2 Subtyping

```
result = get_NMF(mat = pmat$mat,
                 method = 'MAD',
                 clusterNum = 4,
                 seed = 3211232,
                 nrun = 10,
                 doPlot = T)
```

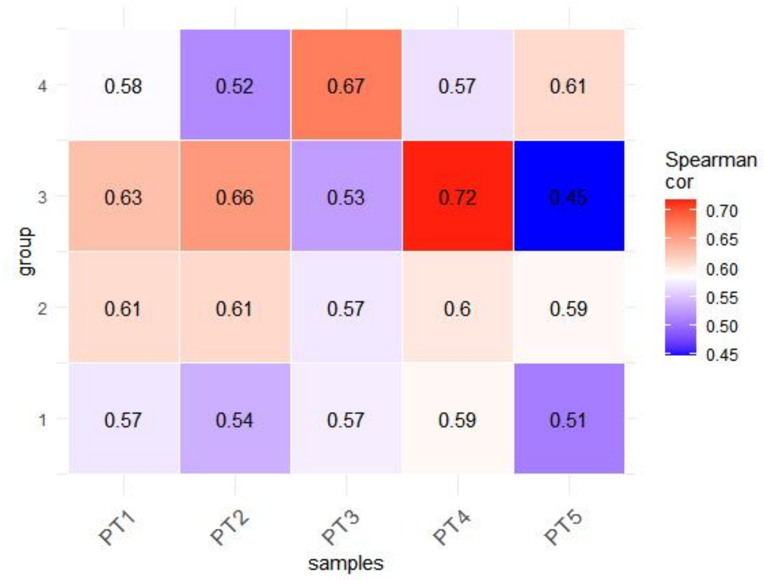
Figure1: NMF subtyping result



2.3 Identification of subgroups

```
subgroup = get_subgroup(cmat = pmat$cmat,
                        mat = pmat$mat,
                        subtype = result,
                        k = 10,
                        biopsy = "COLON",
                        adjacent = F,
                        doplot = T,
                        db.path = '.',
                        OrgDB = org.Hs.eg.db)
```

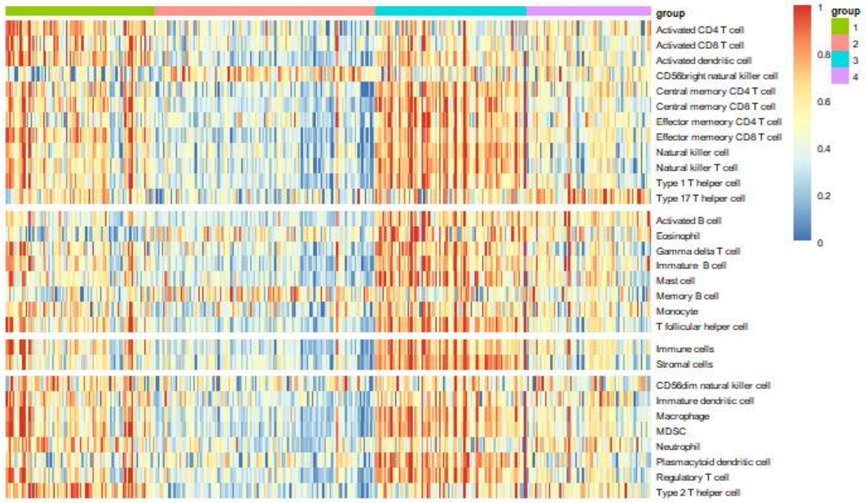
Figure2: The heatmap of Spearman correlation coefficients between query samples and subtypes



2.4 Estimation of non-cancerous infiltration

```
es = get_estimateScore(mat = pmat$mat, subtype=result, doplot = T)
```

Figure3: The non-cancerous cell infiltration heatmap of colorectal cancer subtypes



2.5 Purification of subgroups

```
# subgroup simple version
subgroup_simple = lapply(subgroup, function(x){
  lapply(x, function(y){
    gene = intersect(row.names(y),row.names(octad.db::lincs_signatures))
    return(y[gene,])
  })
})
```

```
purify_data = get_puretumor(subgroup = subgroup_simple)
head(subgroup_simple[[1]]$case)
```

Table2: Subgroup of PT1

	TCGA-WS-AB45-01	TCGA-F4-6704-01	TCGA-AZ-4323-01	TCGA-AY-6196-01	TCGA-AZ-6607-01	TCGA-EI-7004-01	TCGA-G4-6302-01	TCGA-A6-6651-01	TCGA-AF-2687-01	TCGA-AF-2690-01
TSPAN6	1348	2626	1559	1151	2211	1589	1746	7967	9139	6969
SCYL3	728	875	713	1033	384	791	481	631	969	743
BAD	1317	3698	1095	1148	1674	1544	1278	1535	1023	1732
LAP3	6746	5744	4743	7648	2617	4296	5296	6653	6979	8694
SNX11	1200	1220	1162	884	1043	1484	888	1280	1302	1301
CASP10	1028	2176	1032	1587	878	1662	1596	871	887	1268

2.6 Differential expression analysis

```
signature = get_diff(data = purify_data,
                     DE_method = 'limma',
                     normalize_samples = TRUE,
                     threshold_log2foldchange = 2,
                     threshold_pval = 1,
                     threshold_adjpval = 0.05)
head(signature[[1]])[,1:7]
```

Table3: Inidividual disease signature of PT1

log2FoldChange	AveExpr	t	pvalue	padj	identifier	ensembl
-7.338973	6.909496	-9.480442	0e+00	3.00e-07	GSTM2	ENSG00000213366
-4.772934	8.663945	-6.606637	5e-07	3.24e-05	CEBPD	ENSG00000221869
-4.759664	13.018551	-9.010200	0e+00	6.00e-07	MYL9	ENSG00000101335
-4.700296	5.481505	-9.399755	0e+00	3.00e-07	HOXA5	ENSG00000106004
-4.254804	12.538385	-7.867757	0e+00	3.70e-06	CSRP1	ENSG00000159176
-4.215757	8.172159	-9.360803	0e+00	3.00e-07	SLC25A4	ENSG00000151729

3. Screening of candidate agents by reversing signals

```
sRGES = lapply(signature, function(x, LINCS){
  return(get_reverse_score(dz_signature = x,
                           max_gene_size=500,
                           permutations = 10000,
                           LINCS_data = LINCS))
  },LINCS = NULL)

gc()
head(sRGES[[1]])
```

Table4: Candidate agents for PT1

pert_iname	mean	min	max	n	median	sd	sRGES
BRD-K91899208	-0.7450238	-0.7450238	-0.7450238	1	-0.7450238	NA	-0.7450238
BRD-K83757561	-0.7007165	-0.7007165	-0.7007165	1	-0.7007165	NA	-0.7007165
BRD-K12230293	-0.6912223	-0.6912223	-0.6912223	1	-0.6912223	NA	-0.6912223
BRD-K20347745	-0.6892153	-0.6892153	-0.6892153	1	-0.6892153	NA	-0.6892153
BRD-K31593084	-0.6856379	-0.6856379	-0.6856379	1	-0.6856379	NA	-0.6856379
BRD-K97217923	-0.6448751	-0.6448751	-0.6448751	1	-0.6448751	NA	-0.6448751

4. Assessment of drug efficacy

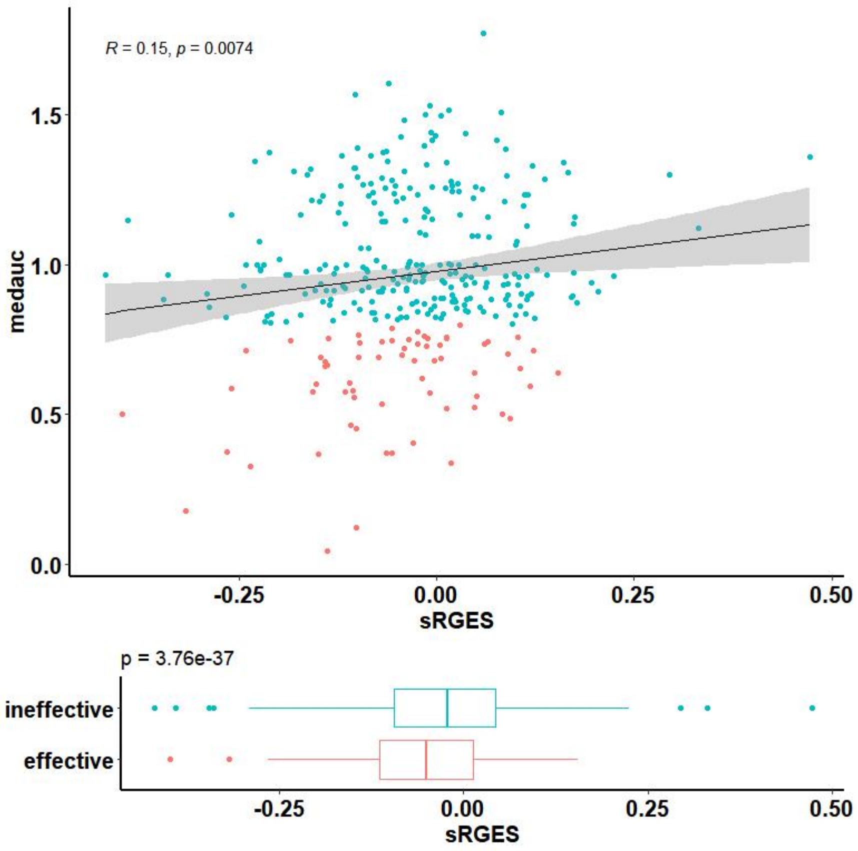
4.1 Calculation of inidividual-related cell lines

```
cell_info = get_cell(cmat = pmat$cmat,
                     db.path = './CPDR_db/Pharmacogenomic',
                     removeBatchEffect = FALSE,
                     orgDB = org.Hs.eg.db)
```

4.2 Evaluattion of drug effectiveness at cell-line level

```
cor = drugcorTest(mysRGES=sRGES, topline = topline, cell_info = cell_in
fo)
draw_cor_map(sRGES[[1]],topline[1], cell_info = cell_info)
```

Figure4: The predicted drugs efficacy test in PT1



Session information

Here is the output of sessionInfo on the system where this document was compiled:

```
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Chinese (Simplified)_China.936
## [2] LC_CTYPE=Chinese (Simplified)_China.936
## [3] LC_MONETARY=Chinese (Simplified)_China.936
## [4] LC_NUMERIC=C
## [5] LC_TIME=Chinese (Simplified)_China.936
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
```

```
##
## loaded via a namespace (and not attached):
##  [1] compiler_4.1.1  magrittr_2.0.1  fastmap_1.1.0   tools_4.1.1
##  [5] htmltools_0.5.2 yaml_2.2.1      stringi_1.7.4   rmarkdown_2.11
##  [9] highr_0.9       knitr_1.36      stringr_1.4.0   xfun_0.25
## [13] digest_0.6.27   rlang_0.4.11    evaluate_0.14
```