Preprocessing and Subtype prediction

- preprocess obtained gene expression data from the gene expression omnibus database

Preprocessing

- Infos: https://www.frontiersin.org/articles/10.3389/fonc.2017.00135/full
- read data from the current directory
- uses the `affy` package from `bioconductor` to process affymetrix oligonucleotide arrays
- display the used annotation resource: `hg133plus2`

```
# set working directory
setwd("~/Desktop/projectmodul-ma-lutz/colectoral_data/test_dataset")
# read files
Data <- ReadAffy()
# log2 transform
#Data <- log(Data)
slotNames(Data)
```

```
##  [1] "cdfName"           "nrow"              "ncol"
##  [4] "assayData"         "phenoData"         "featureData"
##  [7] "experimentData"    "annotation"        "protocolData"
## [10] ".__classVersion__"
```

```
#sampleNames(Data)
annotation(Data)
```

```
## [1] "hgu133plus2"
```

- RMA-probe summary: https://academic.oup.com/biostatistics/article/4/2/249/245074
- Robust multichip average
- creates an expression matrix from the given affymetrix data with:
  - background correction $PM_{ijk} = bg_{ijk} + s_{ijk}$
    - s: signal for probe j of probe set k on array i
    - bg: background caused by optical noise + non-specific binding
  - quantile normalization
    - make n vectors have same distribution by projectin n-dim. quantile plot onto the diagonal
  - log2 transform
  - summarize
    - $Y_{ijk} = \mu_{ik} + \alpha_{jk} + \epsilon_{ijk}$
  - linear model for normalization to obtain expression measure for each probe set
- other methods: PLIER, MAS5...

```
# perform rma probe summary, quantile normalization
eset <- rma(Data, normalize=TRUE)
```

```
## Warning: replacing previous import 'AnnotationDbi::tail' by 'utils::tail' when
## loading 'hgu133plus2cdf'
```

```
## Warning: replacing previous import 'AnnotationDbi::head' by 'utils::head' when
## loading 'hgu133plus2cdf'
```

```
##
```

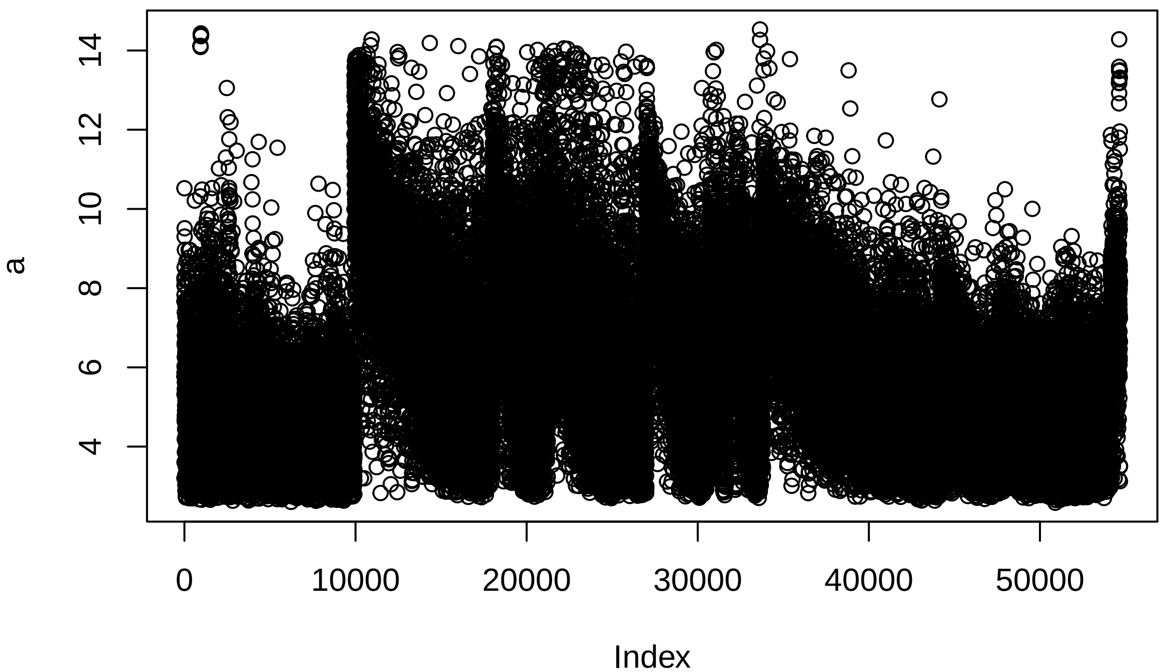
```
## Background correcting
## Normalizing
## Calculating Expression
```

```
rma_eset <- exprs(eset)
#exprs(eset)
#pData(eset)
pdat <- pData(eset)
dim(eset); class(eset)
```

```
## Features  Samples
##    54675      217
```

```
## [1] "ExpressionSet"
## attr(,"package")
## [1] "Biobase"
```

```
#plot(rowMeans(rma_eset), pch=".",  main="Scatter plot")
a <- rowMeans(rma_eset)
plot(a)
```



(avoid batch effect)

```
# remove batch effect(?)
#strain <- c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,
#design <- model.matrix(~factor(strain))
#colnames(design) <- c("CC", "N")
#design
#dim(eset)
#dim(design)

#fit <- lmFit(eset, design)
#ebayes <- eBayes(fit)
#names(ebayes)
#topTable(ebayes, coef=2)
```

- take the annotation resource (here `hgu133plus2`)
- obtain geneIDs and symbols for the corresponding ID_REFs for the gene expression data
- map gene IDs and symbols to the processed data
- combine the data and annotation in a new dataframe
- write output to `csv` file

```
# annotations
annotation(eset)
```

```
## [1] "hgu133plus2"
```

```
#rownames(rma_eset)
genenames <- rownames(rma_eset)
#geneID <- getEG(genenames, "hgu133plus2.db")
geneID <- getEG(genenames, annotation(eset))
geneSymbols <- getSYMBOL(genenames, annotation(eset))

geneID[1:5]
```

```
## 1007_s_at   1053_at    117_at    121_at 1255_g_at
##     "780"    "5982"    "3310"    "7849"    "2978"
```

```
geneSymbols[1:5]
```

```
## 1007_s_at   1053_at    117_at    121_at 1255_g_at
##    "DDR1"    "RFC2"   "HSPA6"    "PAX8"  "GUCA1A"
```

```
# print gene expressions
test <- data.frame(geneSymbols, rma_eset)
res <- data.frame(geneID, test)
res[1:1,]
```

```
##           geneID geneSymbols GSM215051.CEL GSM215052.CEL GSM215053.CEL
## 1007_s_at    780        DDR1      10.75657      10.60066      10.57203
##           GSM215054.CEL GSM215055.CEL GSM215056.CEL GSM215057.CEL GSM215058.CEL
## 1007_s_at      10.51407      10.58489      10.60942      10.65138      10.67384
##           GSM215059.CEL GSM215060.CEL GSM215061.CEL GSM215062.CEL GSM215063.CEL
## 1007_s_at      10.45536      10.62666      10.18406      10.48099       10.4012
##           GSM215064.CEL GSM215065.CEL GSM215066.CEL GSM215067.CEL GSM215068.CEL
## 1007_s_at      10.46146      10.36702      10.49067      10.63043      10.63176
##           GSM215069.CEL GSM215070.CEL GSM215071.CEL GSM215072.CEL GSM215073.CEL
## 1007_s_at      10.57755      10.60577      10.52968       10.3724      10.48029
##           GSM215074.CEL GSM215075.CEL GSM215076.CEL GSM215077.CEL GSM215078.CEL
## 1007_s_at      10.53463      10.64089      10.32638      10.44051      10.50026
##           GSM215079.CEL GSM215080.CEL GSM215081.CEL GSM215082.CEL GSM215083.CEL
## 1007_s_at      10.72077      10.58267      10.49973      10.73748      10.68709
##           GSM215084.CEL GSM215085.CEL GSM215086.CEL GSM215087.CEL GSM215088.CEL
## 1007_s_at      10.62089      10.36807      10.47619      10.30593      10.46062
##           GSM215089.CEL GSM215090.CEL GSM215091.CEL GSM215092.CEL GSM215093.CEL
## 1007_s_at      10.67671       10.3886      10.38559      10.49644      10.80594
##           GSM215094.CEL GSM215095.CEL GSM215096.CEL GSM215097.CEL GSM215098.CEL
## 1007_s_at      10.22355      10.39405      10.52828      10.23417        10.139
##           GSM215099.CEL GSM215100.CEL GSM215101.CEL GSM215102.CEL GSM215103.CEL
## 1007_s_at      10.31639       10.3038      10.54842      10.91867      10.44248
##           GSM215104.CEL GSM215105.CEL GSM215106.CEL GSM215107.CEL GSM215108.CEL
## 1007_s_at      10.45666      10.36262      10.31795      10.06892      10.54805
##           GSM215109.CEL GSM215110.CEL GSM215111.CEL GSM215112.CEL GSM215113.CEL
```

```
## 1007_s_at      10.67632      10.46141       10.5409      10.44495      10.58702
##           GSM215114.CEL GSM588828.CEL GSM588829.CEL GSM588830.CEL GSM588831.CEL
## 1007_s_at      10.57124      10.43125      10.56259      11.00055      10.65032
##           GSM588832.CEL GSM588833.CEL GSM588834.CEL GSM588835.CEL GSM588836.CEL
## 1007_s_at        10.735      10.25507      10.56694      10.39355        11.257
##           GSM588837.CEL GSM588838.CEL GSM588839.CEL GSM588840.CEL GSM588841.CEL
## 1007_s_at      10.57418      10.09049      10.23345      10.87454      9.758848
##           GSM588842.CEL GSM588843.CEL GSM588844.CEL GSM588845.CEL GSM588846.CEL
## 1007_s_at       10.0687      9.569561      9.515976      10.58101      10.73398
##           GSM588847.CEL GSM588848.CEL GSM588849.CEL GSM588850.CEL GSM588851.CEL
## 1007_s_at      9.184561       10.7892       9.37044      10.17294      10.03143
##           GSM588852.CEL GSM588853.CEL GSM588854.CEL GSM588855.CEL GSM588856.CEL
## 1007_s_at      10.21121      10.26351      9.878277      10.46445      11.12495
##           GSM588857.CEL GSM588858.CEL GSM588859.CEL GSM588860.CEL GSM588861.CEL
## 1007_s_at       10.3513      9.613717      10.72523       10.0986      10.02294
##           GSM588862.CEL GSM588863.CEL GSM588864.CEL GSM588865.CEL GSM588866.CEL
## 1007_s_at         10.14      10.10239      9.916989      10.40317      10.05967
##           GSM588867.CEL GSM588868.CEL GSM588869.CEL GSM588870.CEL GSM588871.CEL
## 1007_s_at      9.185089       9.67575      10.06458      9.769896      9.655567
##           GSM588872.CEL GSM588873.CEL GSM588874.CEL GSM588875.CEL GSM588876.CEL
## 1007_s_at      9.311869      9.910383      10.04611      9.932999      10.03239
##           GSM588877.CEL GSM588878.CEL GSM588879.CEL GSM588880.CEL GSM588881.CEL
## 1007_s_at      10.50595      10.14236      9.699015       10.1566      9.824583
##           GSM588882.CEL GSM588883.CEL GSM588884.CEL GSM588885.CEL GSM588886.CEL
## 1007_s_at      10.18439      10.06308      10.37541      10.70229      10.41674
##           GSM916687_AH1.CEL GSM916688_AH10.CEL GSM916689_AH11.CEL
## 1007_s_at          10.82064           10.9147           11.02612
##           GSM916690_AH12.CEL GSM916691_AH13.CEL GSM916692_AH2.CEL
## 1007_s_at           10.55015           10.01362           11.16186
##           GSM916693_AH3.CEL GSM916694_AH4.CEL GSM916695_AH5.CEL
## 1007_s_at           10.6225           10.83443           10.91832
##           GSM916696_AH6.CEL GSM916697_AH7.CEL GSM916698_AH8.CEL
## 1007_s_at          10.23453           10.83837           10.87016
##           GSM916699_AH9.CEL GSM916700_AL1.CEL GSM916701_AL10.CEL
## 1007_s_at          10.80817           10.86599           10.85325
##           GSM916702_AL11.CEL GSM916703_AL12.CEL GSM916704_AL13.CEL
## 1007_s_at           10.75046           10.76432           10.91951
##           GSM916705_AL14.CEL GSM916706_AL15.CEL GSM916707_AL16.CEL
## 1007_s_at           10.87427           10.78523           10.76982
##           GSM916708_AL2.CEL GSM916709_AL3.CEL GSM916710_AL4.CEL
## 1007_s_at           11.0175           10.63092           10.84956
##           GSM916711_AL5.CEL GSM916712_AL6.CEL GSM916713_AL7.CEL
## 1007_s_at           10.7119           10.92433           10.55689
##           GSM916714_AL8.CEL GSM916715_AL9.CEL GSM916716_CRC_AB1.CEL
## 1007_s_at          10.53669           10.59517              10.29029
##           GSM916717_CRC_AB10.CEL GSM916718_CRC_AB11.CEL GSM916719_CRC_AB12.CEL
## 1007_s_at               10.87103               10.67986               10.32701
##           GSM916720_CRC_AB13.CEL GSM916721_CRC_AB14.CEL GSM916722_CRC_AB2.CEL
## 1007_s_at               10.53969               11.14551              11.02385
##           GSM916723_CRC_AB3.CEL GSM916724_CRC_AB4.CEL GSM916725_CRC_AB5.CEL
## 1007_s_at               10.7733              10.51572              10.99957
##           GSM916726_CRC_AB6.CEL GSM916727_CRC_AB7.CEL GSM916728_CRC_AB8.CEL
## 1007_s_at              10.99551              10.60398              11.04548
##           GSM916729_CRC_AB9.CEL GSM916730_CRC_CD1.CEL GSM916731_CRC_CD10.CEL
```

```
## 1007_s_at              10.36806              10.08843               10.42594
##           GSM916732_CRC_CD11.CEL GSM916733_CRC_CD12.CEL GSM916734_CRC_CD13.CEL
## 1007_s_at               10.83213               10.60986               11.31725
##           GSM916735_CRC_CD2.CEL GSM916736_CRC_CD3.CEL GSM916737_CRC_CD4.CEL
## 1007_s_at                10.419              11.20603              10.25741
##           GSM916738_CRC_CD5.CEL GSM916739_CRC_CD6.CEL GSM916740_CRC_CD7.CEL
## 1007_s_at              11.00935              9.716989              10.56804
##           GSM916741_CRC_CD8.CEL GSM916742_CRC_CD9.CEL GSM916743_N1.CEL
## 1007_s_at              10.23886              10.22127         10.39755
##           GSM916744_N10.CEL GSM916745_N11.CEL GSM916746_N12.CEL
## 1007_s_at          10.83739          10.45972          10.79046
##           GSM916747_N13.CEL GSM916748_N14.CEL GSM916749_N15.CEL
## 1007_s_at          10.55358          10.61105          10.86217
##           GSM916750_N16.CEL GSM916751_N17.CEL GSM916752_N18.CEL
## 1007_s_at          10.75767          11.11278          10.87982
##           GSM916753_N19.CEL GSM916754_N2.CEL GSM916755_N20.CEL
## 1007_s_at          10.89926         10.96967          10.75145
##           GSM916756_N21.CEL GSM916757_N22.CEL GSM916758_N23.CEL
## 1007_s_at          11.07075          10.98186          11.13593
##           GSM916759_N24.CEL GSM916760_N25.CEL GSM916761_N26.CEL
## 1007_s_at          10.84017          11.03783           10.7659
##           GSM916762_N27.CEL GSM916763_N28.CEL GSM916764_N29.CEL
## 1007_s_at          10.70128          10.98234           11.0447
##           GSM916765_N3.CEL GSM916766_N30.CEL GSM916767_N31.CEL
## 1007_s_at         10.79589          10.82276          10.73887
##           GSM916768_N32.CEL GSM916769_N33.CEL GSM916770_N34.CEL
## 1007_s_at          10.81389          11.11984          10.91676
##           GSM916771_N35.CEL GSM916772_N36.CEL GSM916773_N37.CEL
## 1007_s_at          10.94162          10.93549          10.96265
##           GSM916774_N38.CEL GSM916775_N4.CEL GSM916776_N5.CEL GSM916777_N6.CEL
## 1007_s_at          10.85035         10.89259         10.39198         11.06788
##           GSM916778_N7.CEL GSM916779_N8.CEL GSM916780_N9.CEL
## 1007_s_at         10.56867         10.61503         10.86667
```

```
# output to file
write.csv(res, file="test_preprocessed.csv")
```

CRC Subtype prediction

- create dataframe with corresponding gene symbols
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4636487/
- four consensus molecular subtypes (CMS) 1-4
  - CMS1: (MSI Immune)
  - CMS2: (Canonical)
  - CMS3: (Metabolic)
  - CMS4: (Mesenchymal)
- background: https://www.nature.com/articles/s41598-017-16747-x#Sec7
- cmscaller: https://github.com/peterawe/CMScaller
  - classification based on pre-defined cancer-cell intrinsic CMS templates
  - uses nearest template prediction (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2990751/)
    - nearest neighbor-based
    - requires gene signatures + dataset
    - define representative expression pattern of signature genes (template)
    - find nearest template to assign a prediction label to test sample

- compute predictioin confidence

```
par(mfrow=c(1,2))

# get input data (use preprocessed)
#data <- ReadAffy()
#rma_eset
# TODO: remove healthy datasets (they won't have any subtype...)

genenames <- rownames(rma_eset)
#geneID <- getEG(genenames, "hgu133plus2.db")
#geneID <- getEG(genenames, annotation(eset))
geneSymbols <- getSYMBOL(genenames, annotation(eset))
symbolNames <- as.vector(geneSymbols)
#symbolNames

# print gene expressions
emat <- data.frame(rma_eset)
row.names(emat) <- make.names(c(geneSymbols), unique=TRUE)
#test
#emat <- data.frame(geneID, test)


# classify subtypes with cmscaller, input data was already preprocessed
subtypes <- CMScaller(emat, RNAseq = FALSE, rowNames = "symbol", doPlot = FALSE)
```

```
## 36200/54675 rownames [NA.number] (no valid translation)

## 0/54675 rownames [id.number] (translation gives duplicates)

## 36200/54675 rownames(emat) failed to match to human gene identifiers

## Warning: verify that rownames(emat) are symbol

## cosine correlation distance

## 2/529 templates features not in emat, discarded

## 217 samples; 4 classes; 82-236 features/class

## serial processing; 1000 permutation(s)...

## predicted samples/class (FDR<0.05)

##
## CMS1 CMS2 CMS3 CMS4 <NA>
##   13   29   48   55   72

## 72/217 samples set to NA
```

```
#subtypes

# get/write output data
write.csv(subtypes, file = "test_subtype_pred.csv")
```