cAMP signaling and differentiation state in BRAF-mutant melanoma

Caitlin Jagla

October 2021 Rotation Project

Introduction

Melanocyte differentiation depends on cAMP: MC1R -> Gαs -> AC -> cAMP -> PKA -> CREB -> MITF.

Workflow

- Data access
  - Tsoi et al. GREIN database
  - CCLE gene counts: DepMap portal
  - both resources provide length-scaled gene counts processed in similar pipelines
- Data integration
  - Join gene counts tables
  - Combat-Seq batch correction: "negative binomial regression model that retains the integer nature of count data in RNA-seq studies, making the batch-adjusted data compatible with common differential expression software packages that require integer counts"
- Integrated data processing
  - DESeq2 variance stabilizing transformation: "log-space transformation to decouple per-gene variance from mean expression level (alternative/preferred vs. TPM)"
  - Gene-wise z-scores across all samples
- Dimensionality reduction
  - PCA for gene-wise z-scores across all samples
- Linear regression
  - mean z-score for designated genesets: Are mean z-scores for CREB family transcription factor targets correlated to mean z-scores for differentiation state signatures?
- Expression levels of individual cAMP/CREB pathway components across differentiation states (per-gene ANOVA)

Tidy data & annotations

1. Tidy length-scaled gene expression counts data from Tsoi et al. (2018) and CCLE (downloaded from GREIN and DepMap, respectively)
2. Tidy manually generated annotation tables and gene lists (downloaded from Harmonizome, BioPlanet, and GO)
3. Aggregate annotations into combined tables

```
source("scripts/1_tidy_data.R")
source("scripts/2_tidy_annotations.R")
source("scripts/3_aggregate_annotations.R")
```

Integrate gene expression counts data and get z-scores

1. Batch correct for data source using CombatSeq
2. Use DESeq2 Variance Stabilizing Transformation (log-space transformation that decouples per-gene variance from mean expression level; a preferred alternative to TPM)
3. Calculate gene-wise z-scores on VST values
4. Calculate mean z-scores for gene lists

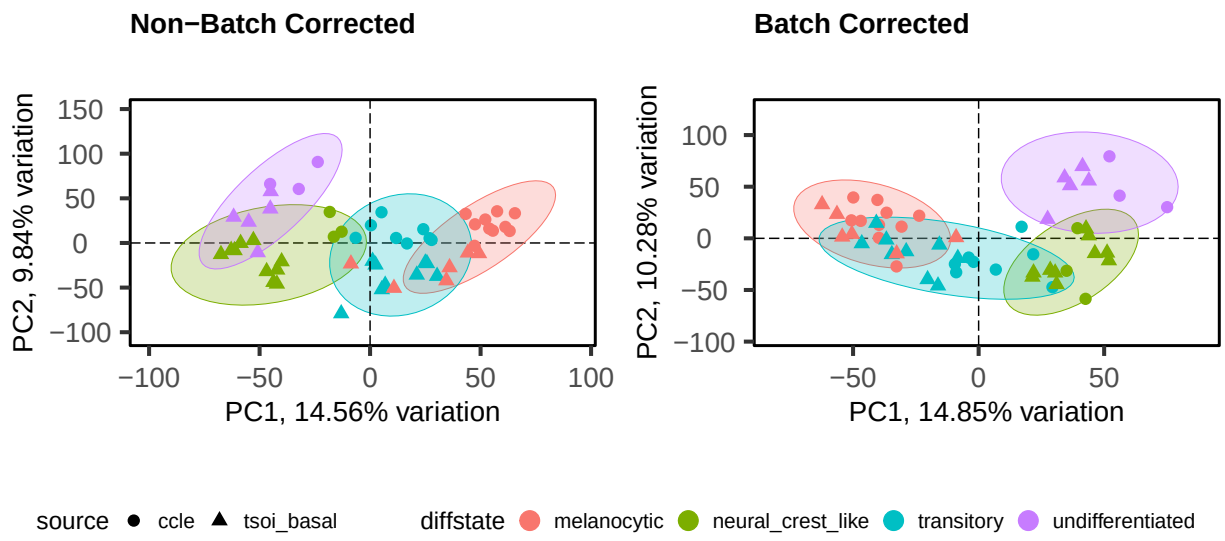
```
source("scripts/4_get_integrated_vst_zscores.R")
source("scripts/5_get_mean_zscores.R")
```

Analysis

1. Is batch correction required?

Cell lines separate in PC1 vs PC2 space regardless of batch correction, but the clusters are tighter and there is more integration between data sources when batch correction is applied.

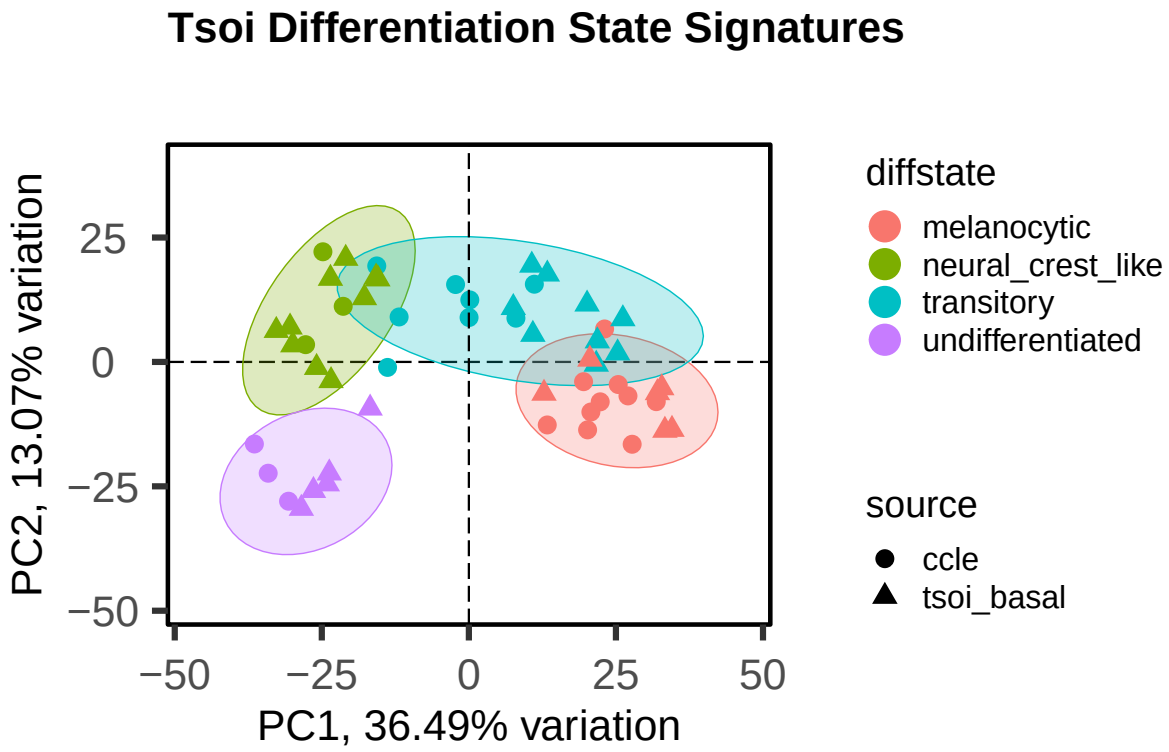
```
pca <- list()
pca$noncorr <- pca(data$vst_noncorr, metadata = metadata, removeVar=0.1)
pca$corr <- pca(data$vst_corr, metadata = metadata, removeVar=0.1)

plots <- list()
plots$biplot_noncorr_PC1PC2 <- biplot_diff(pca$noncorr, x="PC1", y="PC2") +
  ggtitle("Non-Batch Corrected")
plots$biplot_corr_PC1PC2 <- biplot_diff(pca$corr, x="PC1", y="PC2") +
  ggtitle("Batch Corrected")

ggarrange(plots$biplot_noncorr_PC1PC2, plots$biplot_corr_PC1PC2,
          ncol = 2, common.legend=TRUE, legend="bottom")
```



2. Do cell lines cluster by differentiation state when subsetted to include only differentiation state signatures?

As expected, subsetting the gene expression matrix to contain only 514 genes present in the Tsoi *et al.*-defined differentiation state signatures leads to distinct clustering based on previously designated differentiation state.

```
data$goi_subsets <- list()
data$goi_subsets$tsoi_sigs <- data$vst_corr |>
  dplyr::filter(row.names(data$vst_corr) %in% anno$tsoi_subtype_signatures$gene_ensembl)

pca$goi_subsets <- list()
pca$goi_subsets$tsoi_sigs <- pca(data$goi_subsets$tsoi_sigs, metadata = metadata)

plots$goi_subsets <- list()
plots$goi_subsets$tsoi_sigs_PC1PC2 <- biplot_diff(pca$goi_subsets$tsoi_sigs, x="PC1", y="PC2") +
  ggtitle("Tsoi Differentiation State Signatures")

plots$goi_subsets$tsoi_sigs_PC1PC2
```

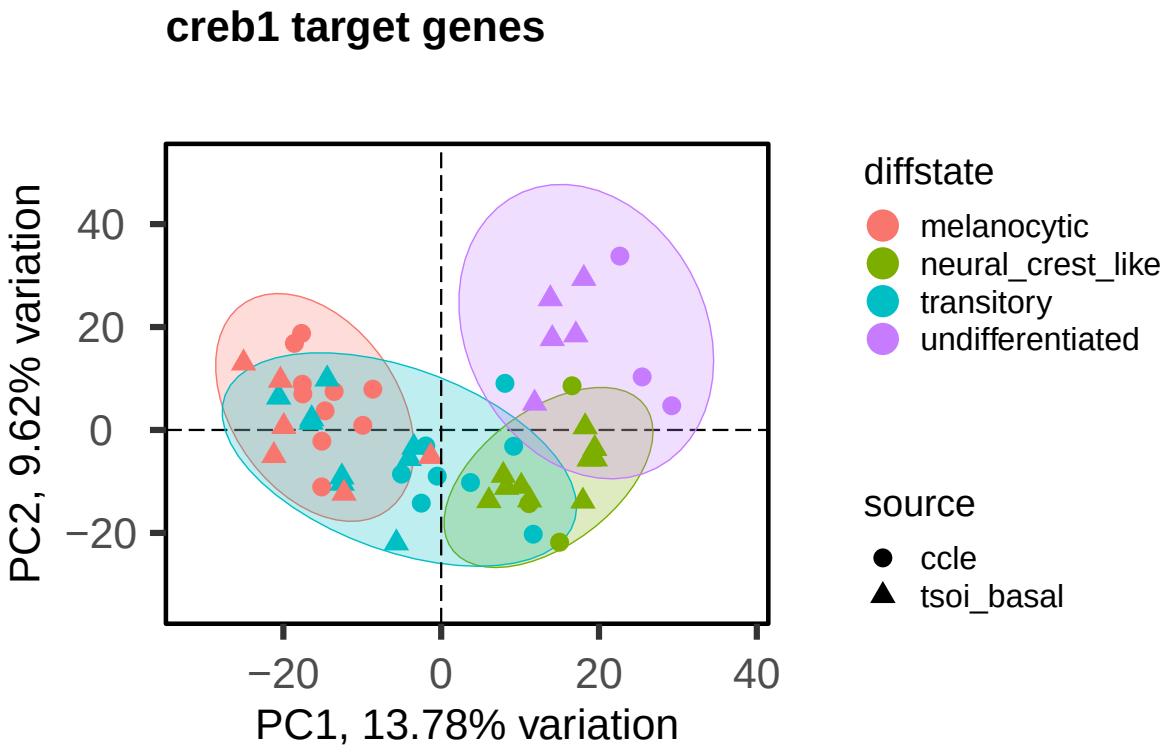


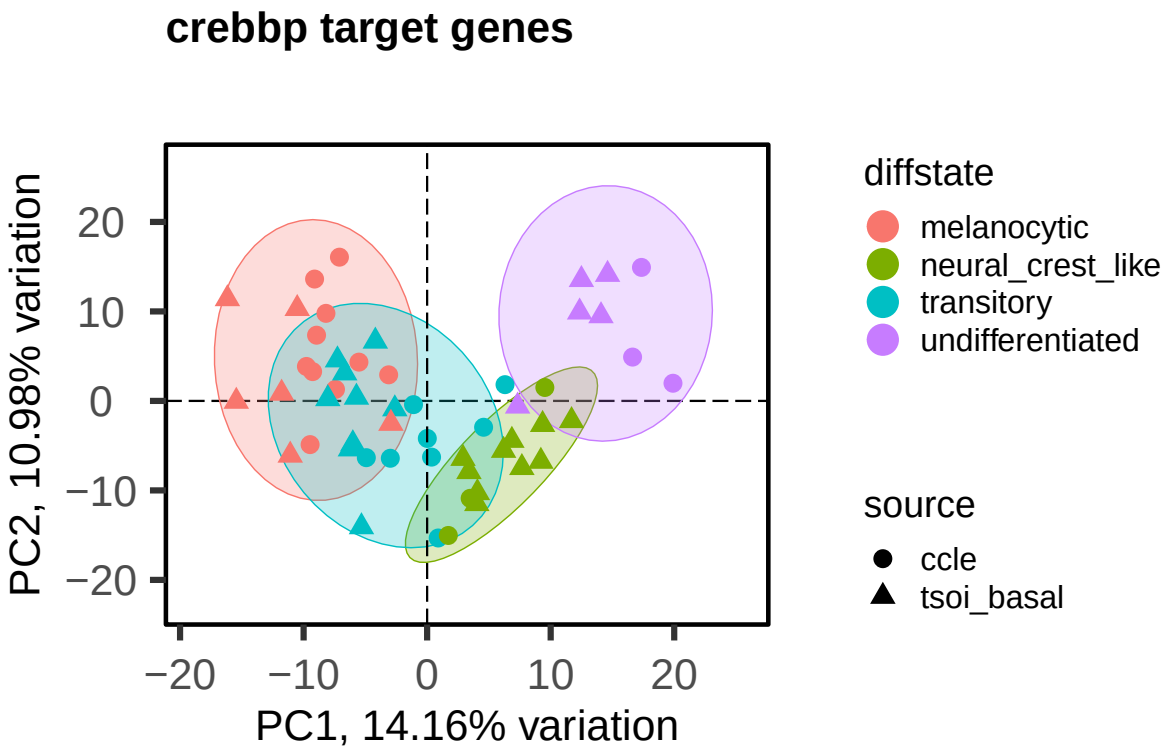
3. Do cell lines cluster by differentiation state when subsetted to include only transcription factor target genes?

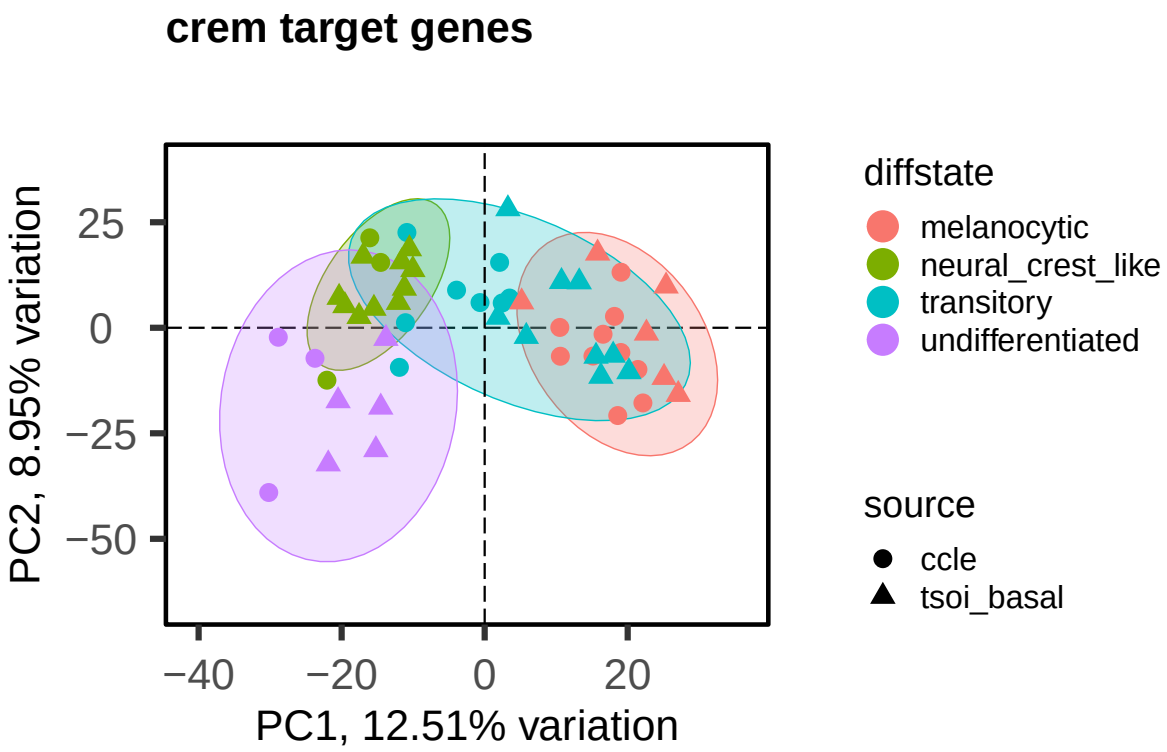
Subsetting the gene expression matrix to contain only genes targeted by transcription factors of interest leads to distinct clustering based on previously designated differentiation state, for the majority of transcription factors tested.

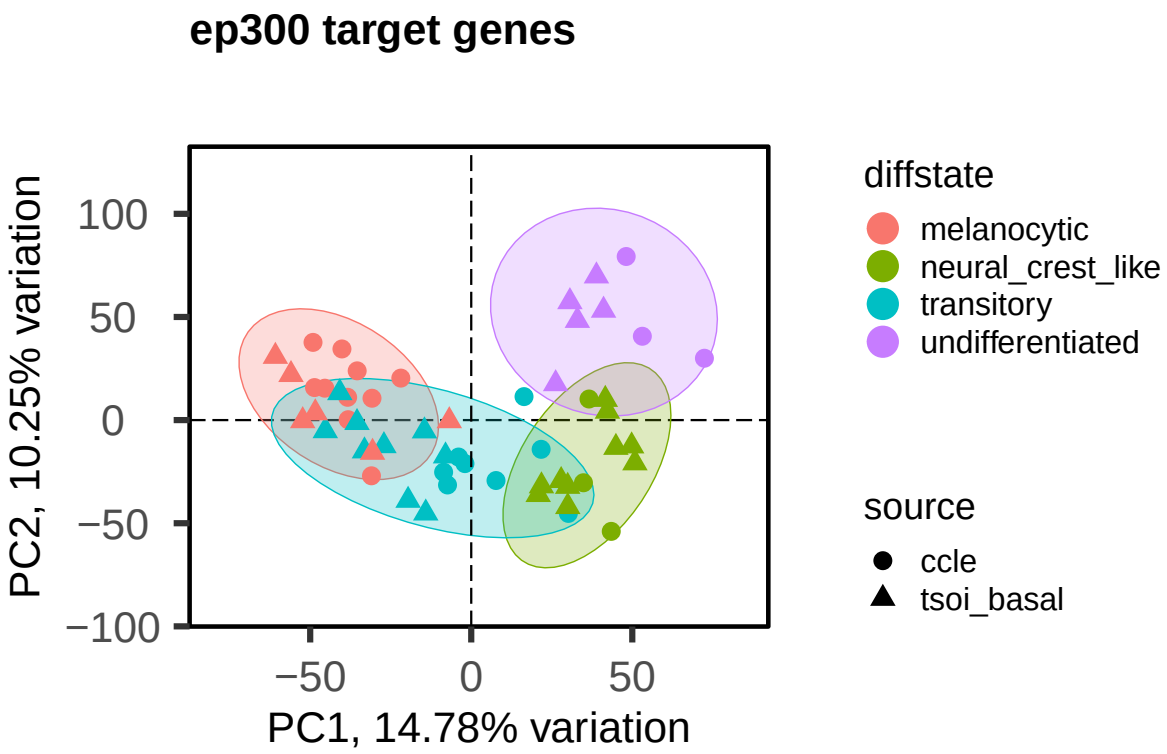
```
data$goi_subsets$tf_targets <- list()
pca$goi_subsets$tf_targets <- list()
plots$goi_subsets$tf_targets <- list()

for (tf in 1:length(anno$tf_targets)){
  gene_list <- anno[["tf_targets"]][[tf]][["gene_ensembl"]]
  tf_name <- names(anno[["tf_targets"]])[tf]

  data[["goi_subsets"]][["tf_targets"]][[tf_name]] <- data$vst_corr |>
  dplyr::filter(row.names(data$vst_corr) %in% gene_list)

  pca[["goi_subsets"]][["tf_targets"]][[tf_name]] <- pca(data[["goi_subsets"]][["tf_targets"]][[tf_name

  plots[["goi_subsets"]][["tf_targets"]][[tf_name]] <- biplot_diff(pca[["goi_subsets"]][["tf_targets"]]
  ggtitle(paste0(tf_name, " target genes"))
}

rm(gene_list, tf_name, tf)
```

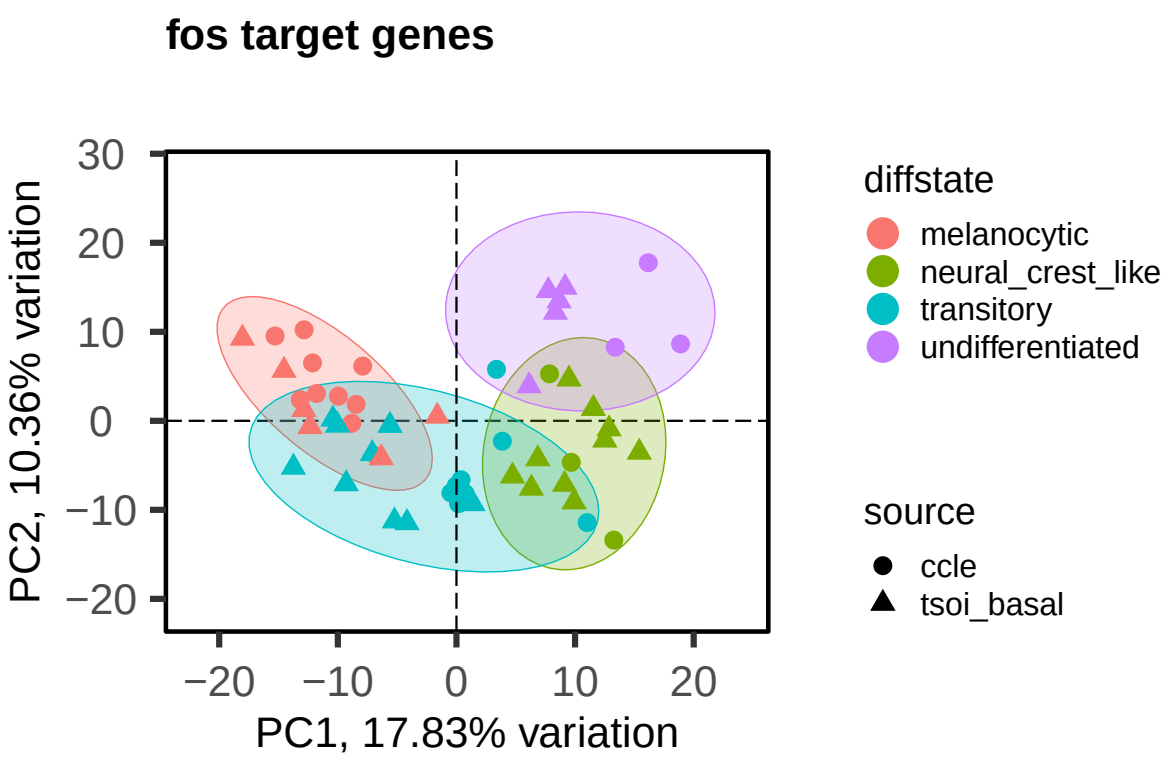
```
plots[["goi_subsets"]][["tf_targets"]]
```

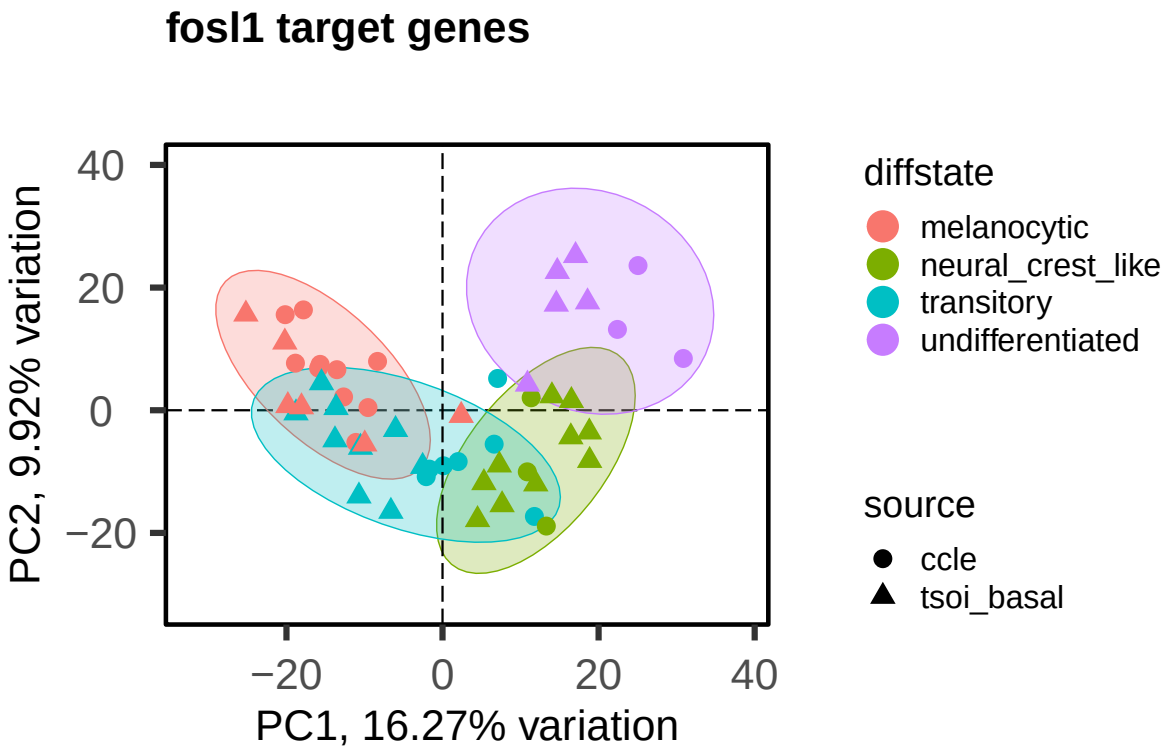


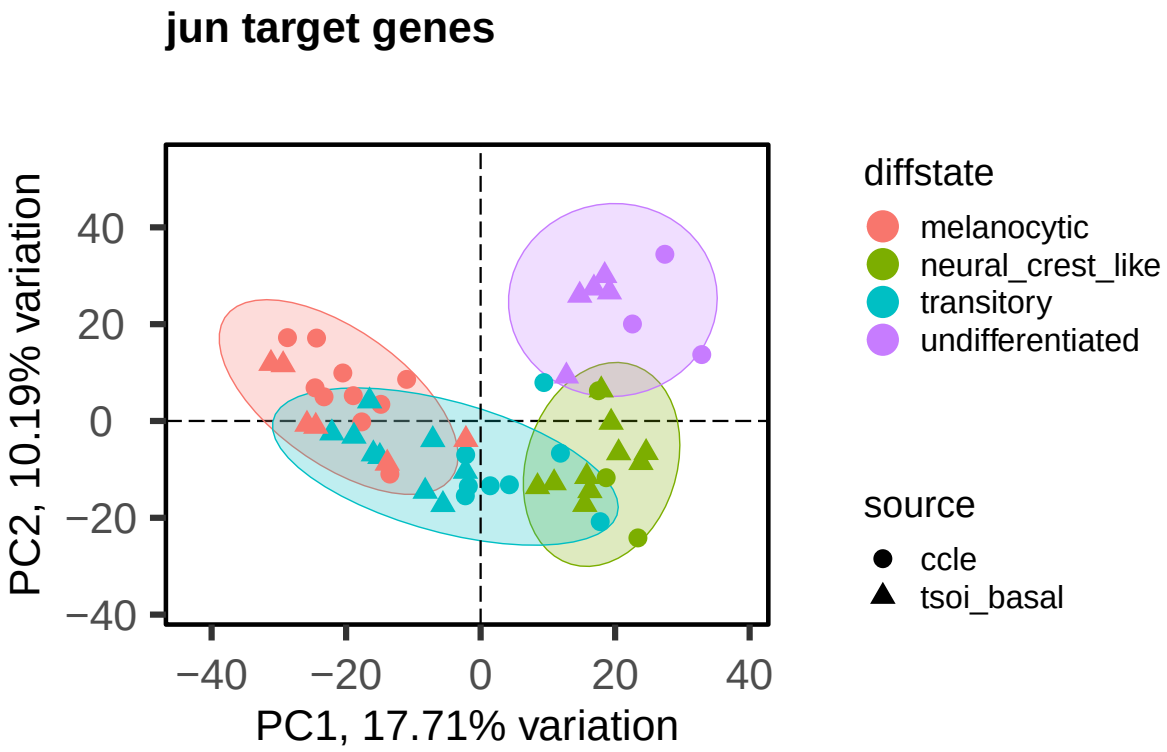

crebbp target genes
PC2, 10.98% variation
20
10
0
−10
−20
−20
−10
0
10
20
PC1, 14.16% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

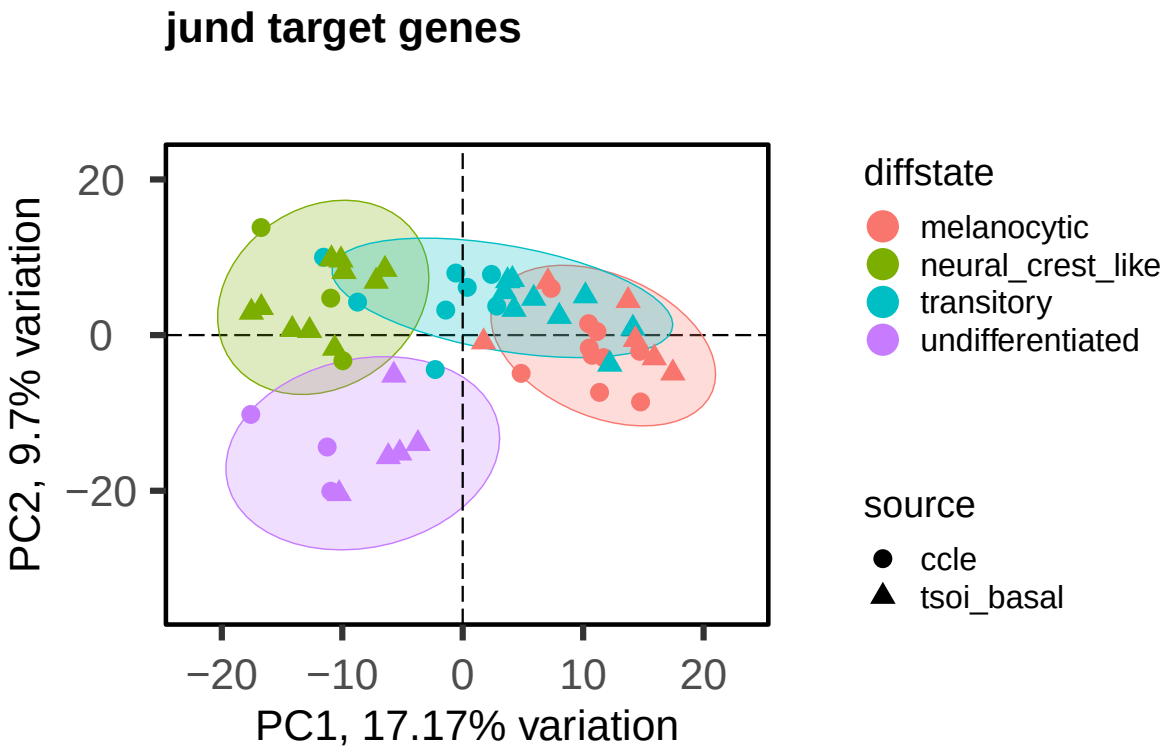

crem target genes
PC2, 8.95% variation
25
0
−25
−50
−40
−20
0
20
PC1, 12.51% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

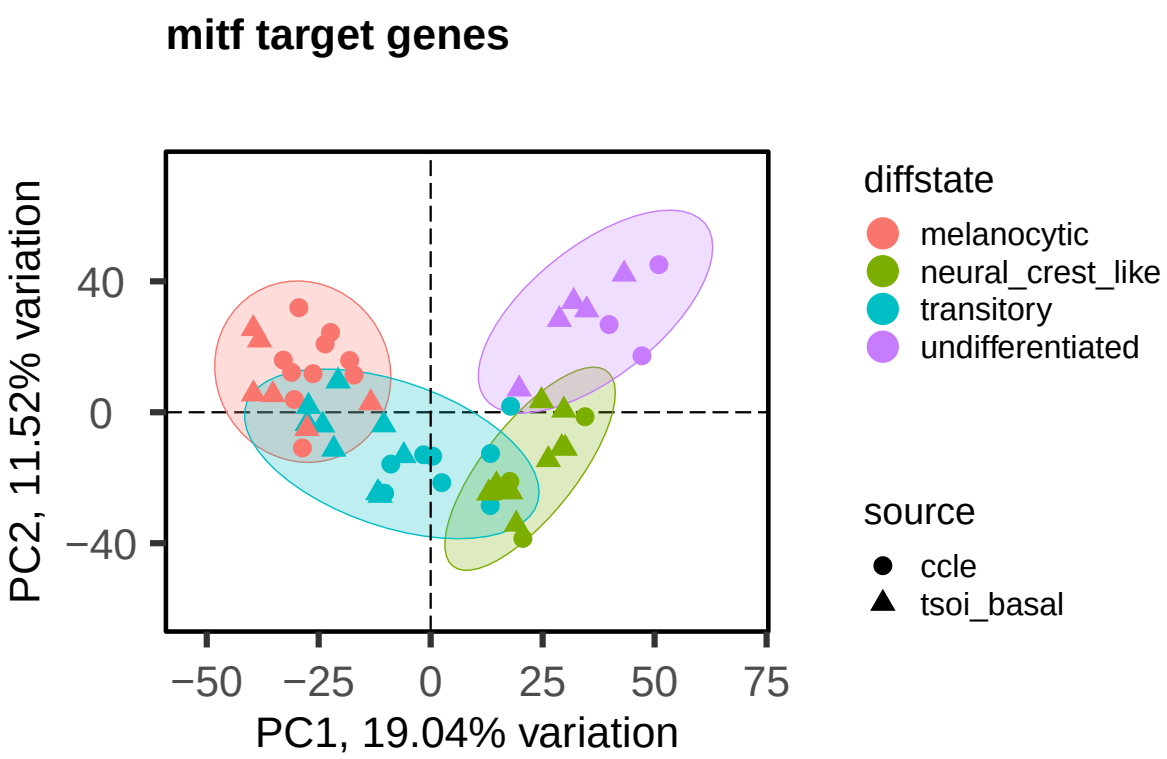

ep300 target genes
PC2, 10.25% variation
100
50
0
−50
−100
−50
0
50
PC1, 14.78% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

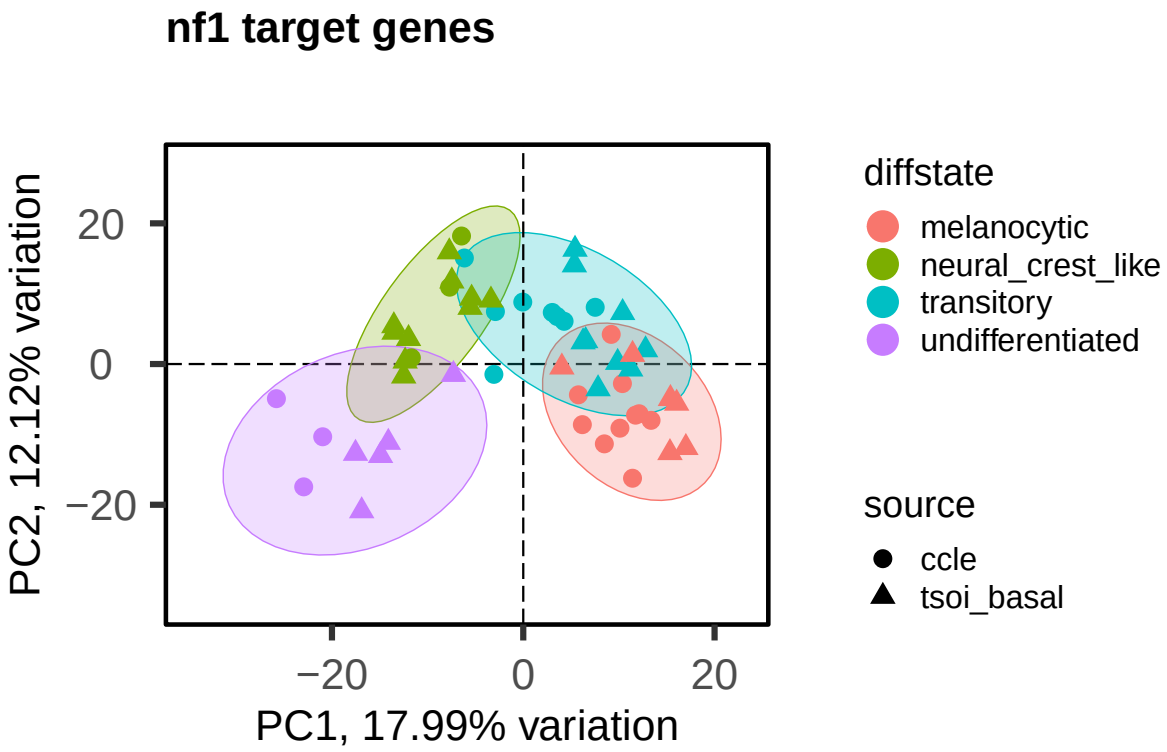

fos target genes
PC2, 10.36% variation
PC1, 17.83% variation
30
20
10
0
−10
−20
−20
−10
0
10
20
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

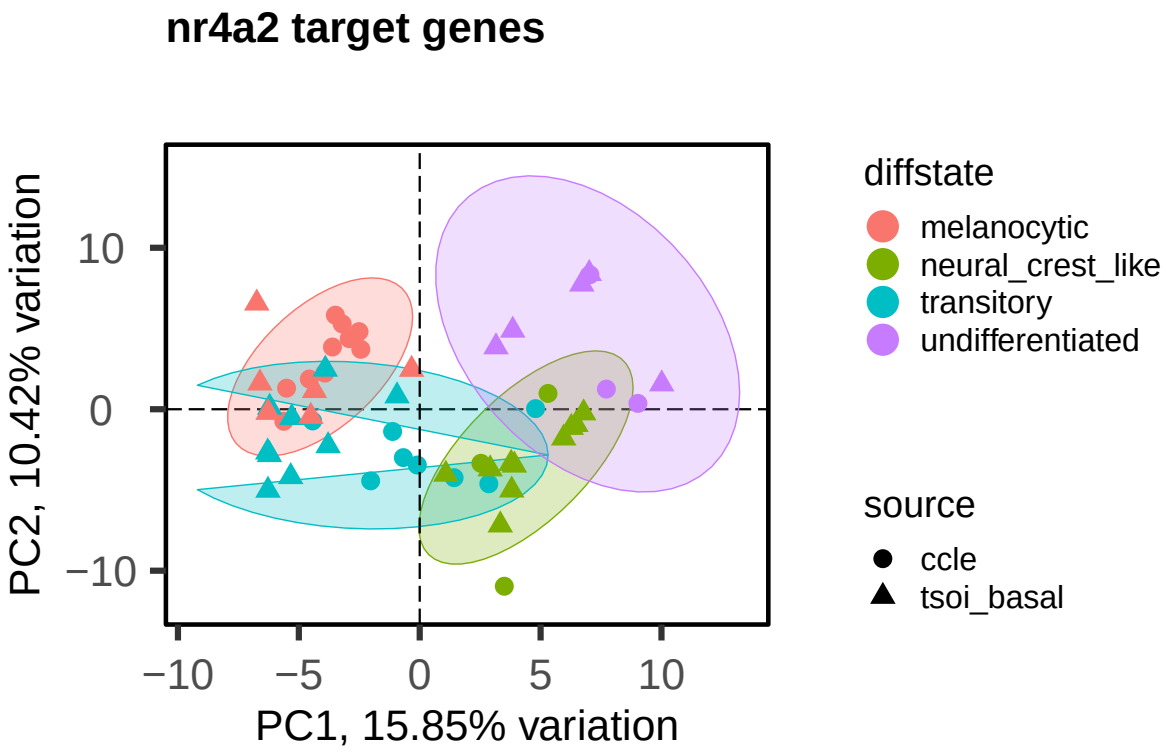

fosl1 target genes
40
20
0
−20
PC2, 9.92% variation
−20
0
20
40
PC1, 16.27% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

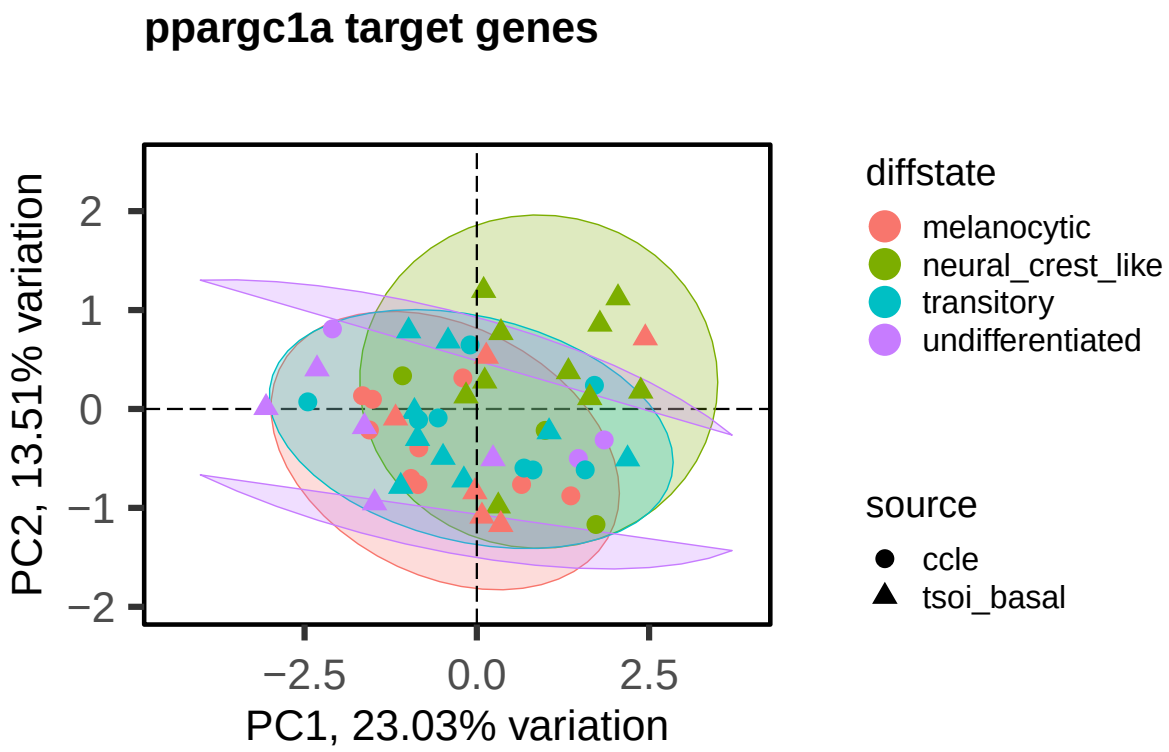

jun target genes
PC2, 10.19% variation
PC1, 17.71% variation
40
20
0
−20
−40
−40
−20
0
20
40
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal


jund target genes
20
0
−20
PC2, 9.7% variation
−20
−10
0
10
20
PC1, 17.17% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal


mitf target genes
PC2, 11.52% variation
40
0
−40
−50
−25
0
25
50
75
PC1, 19.04% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal


nf1 target genes
PC2, 12.12% variation
20
0
−20
−20
0
20
PC1, 17.99% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal


nr4a2 target genes
PC2, 10.42% variation
10
0
−10
−10
−5
0
5
10
PC1, 15.85% variation
diffstate
melanocytic
neural_crest_like
transitory
undifferentiated
source
ccle
tsoi_basal

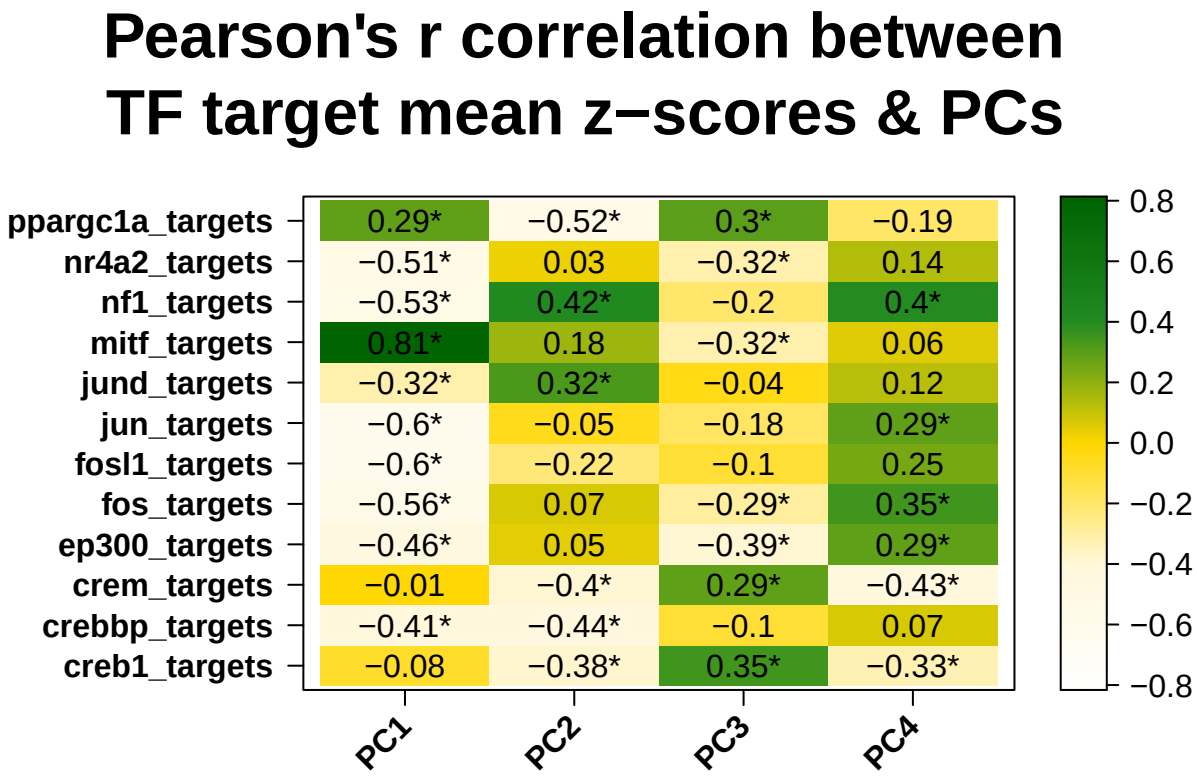


3. Do expression levels of transcription factor targets correlate with principal components that cluster cell lines into previously assigned differentiation states?

- significantly correlates with PC1, which divides cell lines by differentiation state:
  - positive correlation: MITF
  - negative correlation: NR4A2, NF1, JUND, JUN, FOSL1, FOS, EP300, CREBBP
- significantly correlates with PC2, which divides cell lines by differentiation state:
  - positive correlation: NF1, JUND
  - negative correlation: PPARGC1A, CREM, CREBBP, CREB1

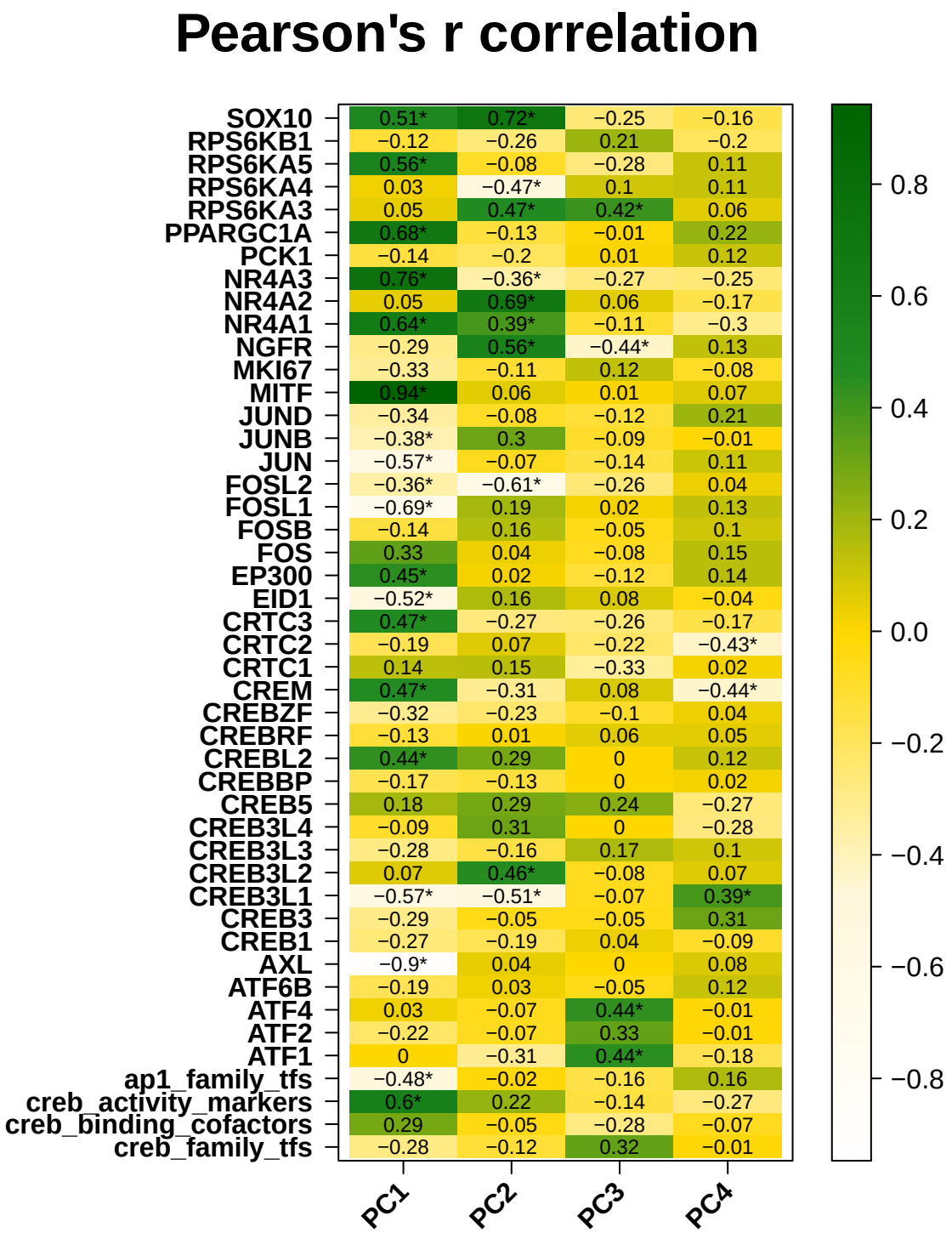
```
data$mean_z <- read_csv(file.path("output", dataset_name, "mean_zscores", "all_genesets_z.csv")) |>
  rename("...1"="sample")

pca$goi_subsets$tsoi_sigs$metadata <- left_join(pca$goi_subsets$tsoi_sigs$metadata, data$mean_z, by="sa

eigen <- list()
eigen$corr_tftargets <- eigencorplot(pca$goi_subsets$tsoi_sigs,
                                     metavars = pca$goi_subsets$tsoi_sigs$metadata |> dplyr::select(conta
                                     components = getComponents(pca$goi_subsets$tsoi_sigs, 1:4),
                                     col = c('white', 'cornsilk1', 'gold', 'forestgreen', 'darkgreen'),
                                     cexCorval = 1,
                                     fontCorval = 1,
                                     posLab = 'bottomleft',
```

```
                                    rotLabX = 45,
                                    scale = TRUE,
                                    main = "Pearson's r correlation between\nTF target mean z-scores & P
                                    corFUN = 'pearson',
                                    corUSE = 'pairwise.complete.obs',
                                    #corMultipleTestCorrection = 'BH',
                                    signifSymbols = c('*', ''),
                                    signifCutpoints = c(0, 0.05, 1))
eigen$corr_tftargets
```




```
eigen$corr_tfs <- eigencorplot(pca$goi_subsets$tsoi_sigs,
                                    metavars = pca$goi_subsets$tsoi_sigs$metadata |>
                                               dplyr::select(!contains(c("targets","diffstate","signatu
                                               colnames(),
                                    components = getComponents(pca$goi_subsets$tsoi_sigs, 1:4),
                                    col = c('white', 'cornsilk1', 'gold', 'forestgreen', 'darkgreen'),
                                    cexCorval = 0.75,
                                    fontCorval = 1,
                                    posLab = 'bottomleft',
                                    rotLabX = 45,
                                    scale = TRUE,
                                    main = "Pearson's r correlation",
                                    corFUN = 'pearson',
                                    corUSE = 'pairwise.complete.obs',
                                    corMultipleTestCorrection = 'BH',
```

```
                                    signifSymbols = c('*', ''),
                                    signifCutpoints = c(0, 0.05, 1))
eigen$corr_tfs
```



4. Is mean expression level of transcription factor targets correlated with expression level of differentiation state signatures?

```
lr <- list()
lr$data <- pca$goi_subsets$tsoi_sigs$metadata |>
  dplyr::select(c(2,3,8,9,11,13,15,16:195)) |>
  column_to_rownames("cell_line")

lr$pearson <- rcorr(as.matrix(lr$data[,3:186]))


lr$plot_list <- c("creb_family_tfs", "creb_binding_cofactors", "creb_activity_markers", "ap1_family_tfs
                  "AXL", "NGFR", "SOX10", "MITF", #basic melanoma diff state markers
                  "NR4A1", "NR4A2", "NR4A3", # marker of creb activity
                  #"ATF1", "ATF4", "CREB1", "CREB3", "CREB5", "CREB3L1", "CREB3L2", "CREB3L3", "CREB3L4
                  #"CRTC2", "CRTC3", "EID1", "EP300", "PPARGC1A", # creb binding cofactors
                  #"FOS", "FOSL1", "FOSL2", "JUN", "JUNB", # ap1 family tfs
                  "creb1_targets", "crebbp_targets", "crem_targets", "ep300_targets", "fos_targets", "f
                  "mitf_targets", "nf1_targets", "nr4a2_targets", "ppargc1a_targets" #tf targets mean z
                  )

lr$plot_r <- lr$pearson$r[row.names(lr$pearson$r) %in% lr$plot_list, 1:4]
lr$plot_r <- lr$plot_r[order(match(rownames(lr$plot_r), lr$plot_list)), ]

lr$plot <- ggcorrplot(lr$plot_r) + ggtitle("Pearson's r for transcription factor targets vs.\ndifferenti

lr$plot
```