

## Supplemental Data S3 for Boot et al., “In-depth characterization of the cisplatin mutational signature in human cell lines and in esophageal and liver tumors”

This analysis shows that reconstruction of observed mutational spectra from mutational signatures using pure mathematical optimization as the sole criterion leads to biologically implausible results, poor specificity, or both.

### Background

We compare detection of the cisplatin signature as shown in the main text Table 1 to reconstructions based solely on mathematical optimization. We optimize reconstructions using both:

- All COSMIC mutational signatures (<http://cancer.sanger.ac.uk/cosmic/signatures>)
- COSMIC mutational signatures previously observed in the given cancer type. It makes biological sense to restrict attention to mutational signatures previously observed in a specific tumor type – for this study either esophageal carcinoma (ESAD) or hepatocellular carcinoma (HCC).

We carry out these optimizations using the `fcnls` function in the R NMF package. This function does not carry out a full non-negative matrix factorization that both discovers signatures and determines the activity of each signature in tumor. Instead, given a set of signatures and observed spectra, `fcnls` returns a coefficient matrix (i.e. set of attributions) that minimizes the Frobenius norm between the reconstructed matrix and the observed spectra (page 39 in <https://cran.r-project.org/web/packages/NMF/NMF.pdf>).

### Methods / code

Read the full set of signatures.

```
all.sigs <-  
  read.delim('COSMIC_plus_cisplatin.tsv', sep = '\t')  
rownames(all.sigs) <-  
  paste0(all.sigs$Before, all.sigs$Ref, all.sigs$After, all.sigs$Var)  
standard.rownames <- rownames(all.sigs)  
all.sigs <- as.matrix(all.sigs[, -(1:4)])
```

Create sets of signatures previously observed in ESAD and HCC.

```
ESAD.sigs <-  
  as.matrix(  
    all.sigs[, c(paste('Signature', c(1,2,4:6,13,17), sep = '.'),  
                  'Cisplatin')])  
  
HCC.sigs <-  
  as.matrix(  
    all.sigs[, c(paste('Signature', c(1,4:6,12,16,17, 'AA',23,24),  
                      sep = '.'),  
                  'Cisplatin')])
```

Define a function to plot the signature assignments for those tumors with at least 5 percent of mutations attributed to cisplatin.

```
plot.stacked <- function(proportions, main) {  
  require(RColorBrewer)  
  palette <- brewer.pal(12, 'Paired')  
  palette <- rep(palette, 3)  
  to.plot <-
```

```

    proportions[,proportions['Cisplatin', ] > 0.05, drop=F]
to.plot2 <- to.plot[rowSums(to.plot) > 0, , drop=F]
palette <- rev(palette[1:nrow(to.plot2)])

par(mar=c(6,4,4,4))
bp <- barplot(
  to.plot2,
  main=paste0(main, '\nTumors with > 5% cisplatin'),
  xlim = c(1, 31), # Make this wide enough for the legend
  col = palette,
  border = T,
  las = 3,
  cex.main=0.8
)
legend(max(bp) + 1, 1, legend=rev(rownames(to.plot2)),
       ncol=2, fill=rev(palette),
       cex = 0.7, bty='n')
}

```

Define a function to read in one file of mutational spectra (path) and clean up the tumor IDs.

```

table.1.positives.by.msigact.and.DNS <-
  c('HK034',
    'RK028', 'RK056', 'RK074', 'RK205', 'RK241', 'RK256',
    'SA594320', 'SA594557', 'SA594775')

read.spectra <- function(path) {
  spectra <- read.delim(path, sep = '\t')

  # The next two lines ensure that the rows in 'spectra' are in the
  # standard order
  rownames(spectra) <-
    paste0(spectra$Before, spectra$Ref, spectra$After, spectra$Var)
  spectra <- spectra[standard.rownames,]
  spectra <- as.matrix(spectra[, -(1:4)])

  # Trim off non-informative parts of the ESAD tumor IDs at 8
  colnames(spectra) <- substr(colnames(spectra), 1, 8)

  prefix <-
    ifelse(colnames(spectra) %in% table.1.positives.by.msigact.and.DNS,
           'p', ' ')
  colnames(spectra) <- paste(prefix, colnames(spectra))
  spectra
}

```

Define a function to reconstruct a matrix of spectra ('spectra') using one set of mutational signatures ('sigs'). The argument 'spec.name' is the name of set of spectra and 'sig.set' is the name of the set of mutational signatures, both used in plotting.

```

process.spectra <- function(spectra, sigs, spec.name, sig.set) {
  # NMF library needed for the fcnnls function
  require(NMF)

  s <- fcnnls(sigs, spectra)
}

```

```

# Get proportions of mutations due to each signature
ss <- apply(s$x, 2, function(v){v / sum(v)})

plot.stacked(ss,
             main=paste(spec.name, 'reconstructed with',
                        sig.set, 'signatures'))
}

```

Get the names of the three files containing the observed mutational spectra for one ESAD data set and two HCC data sets and read the spectra.

```

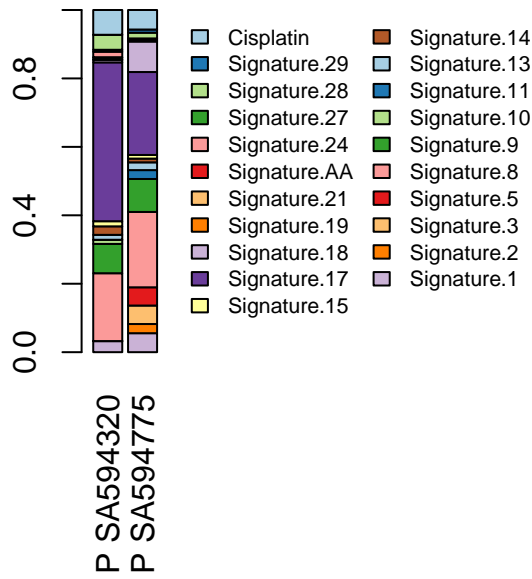
inputpaths <- Sys.glob('input catalogs/*')
all.spectra <- lapply(inputpaths, read.spectra)
names(all.spectra) <- c('ESAD', 'Fujimoto HCC', 'Kan HCC')

```

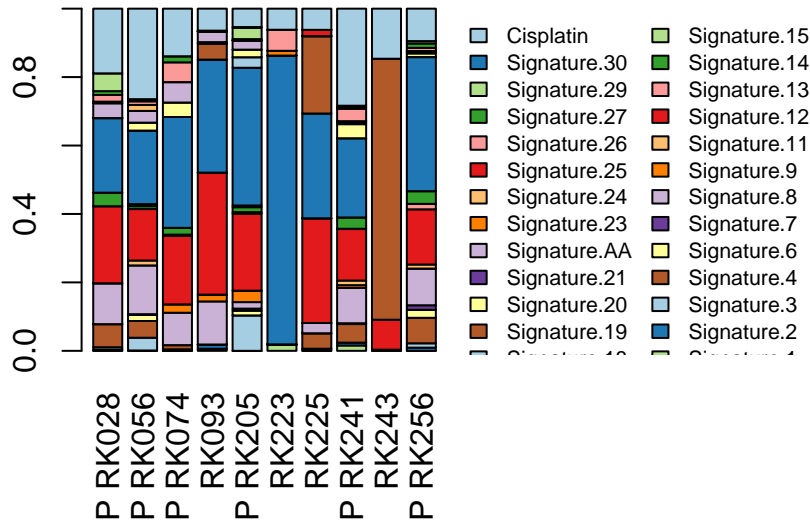
## Results using *ALL* mutational signatures

In the plots below, most tumors identified as cisplatin positive by both the mSigAct signature presence test and DNS signature analysis (main text Table 1) were identified by fcnns using all COSMIC signatures as input. Tumors previously identified as positive are marked with the prefix 'P'. The exception was ESAD SA594557, which was not identified. However, the reconstructions included numerous mutational-signature attributions that are biologically implausible. For example, there were small numbers of mutations attributed to DNA MMR deficiency processes, which are known to generate large numbers of single-base substitutions (COSMIC signatures 6, 14, 15, 20, 21, and 26). The presence of small numbers of mutations attributed to the ultraviolet-radiation signature COSMIC 7 (RK056, RK205, RK241, RK256) is also implausible. Four HCCs for which there is no DNS support for cisplatin exposure were identified: RK093, RK223, RK225, and RK243.

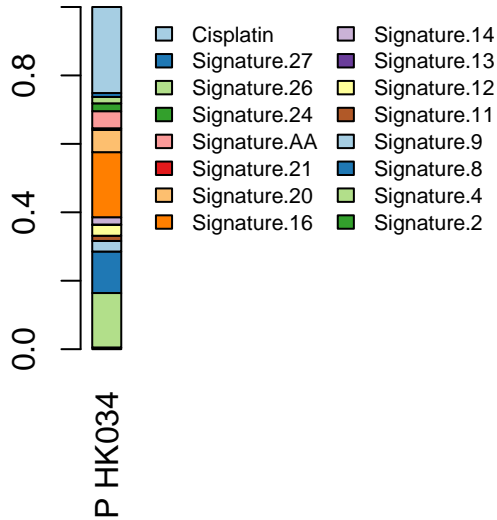
**ESAD reconstructed with all signatures  
Tumors with > 5% cisplatin**



**Fujimoto HCC reconstructed with all signatures  
Tumors with > 5% cisplatin**



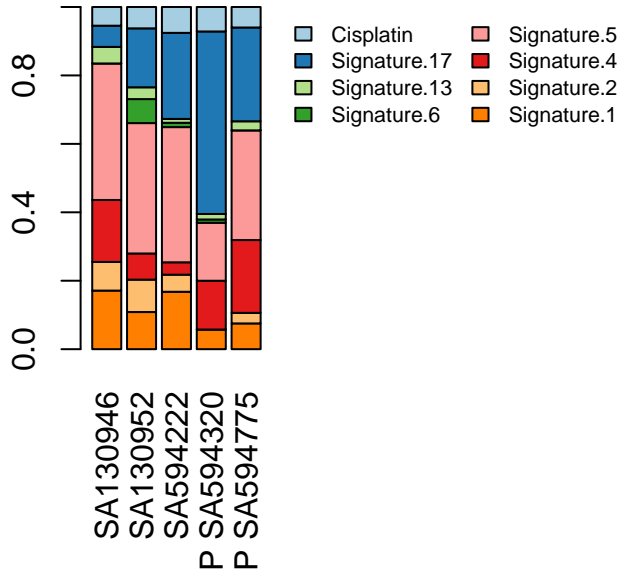
**Kan HCC reconstructed with all signatures  
Tumors with > 5% cisplatin**



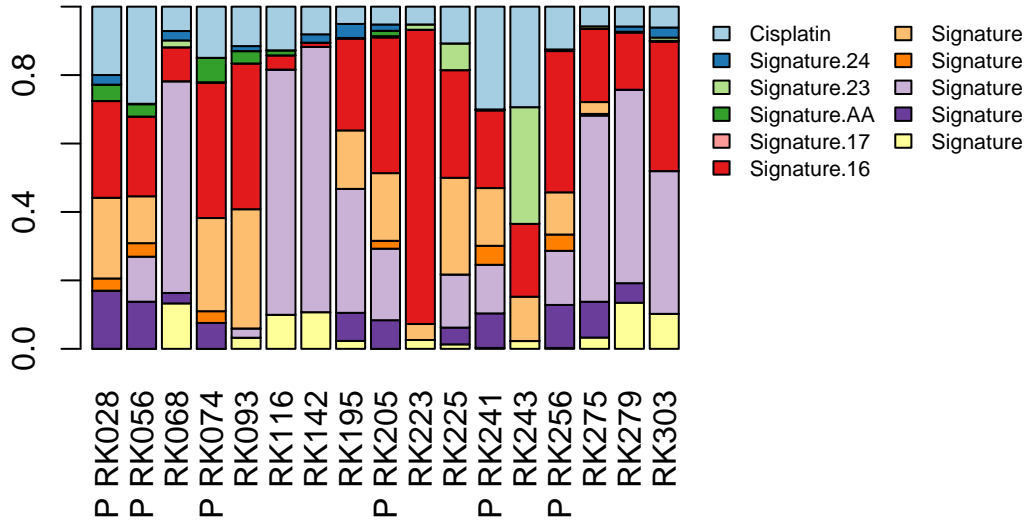
**Results using only mutational signatures previously observed in ESAD or HCC**

In the plots below, all but one of the tumors identified as cisplatin positive by both the mSigAct signature presence test and DNS signature analysis (main text Table 1) were also identified by fcnnls using the COSMIC signatures that were previously observed in the given tumor type (ESAD or HCC). Tumors previously identified as positive are marked with the prefix 'P'. ESAD SA594557 was not identified. The reconstructions still include one mutational signature, COSMIC 6, that for these tumors does not seem to be biologically well supported: COSMIC 6 is associated with DNA mismatch repair deficiency and is associated with large numbers of mutations, which are not seen in the tumors to which it is attributed here. Specificity of this analysis was poor: it found the single-base mutation signature of cisplatin in multiple ESADs and HCCs in which the combined mSigAct and DNS analysis provided no evidence for this exposure.

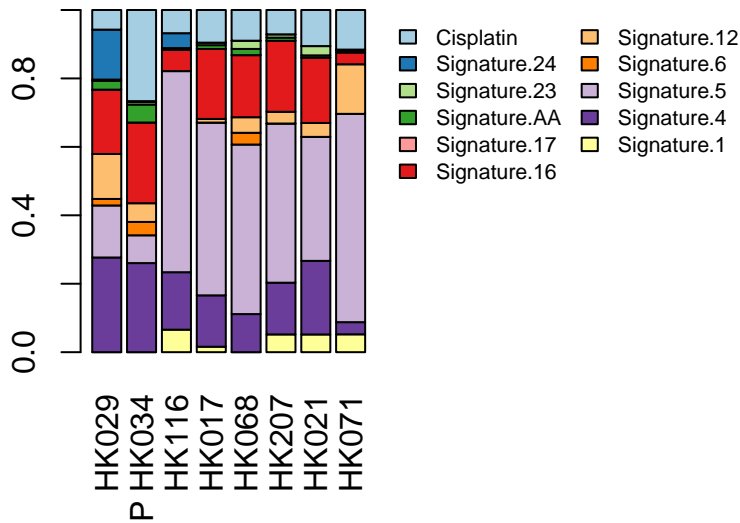
**ESAD reconstructed with ESAD signatures**  
**Tumors with > 5% cisplatin**



**Fujimoto HCC reconstructed with HCC signatures**  
**Tumors with > 5% cisplatin**



### Kan HCC reconstructed with HCC signatures Tumors with > 5% cisplatin



Attribution of the cisplatin signature without sparsity leads to many false positives

Define a function to return IDs of tumors with more than 0 mutations attributed to cisplatin.

```

cisplatin.gt.0 <- function(spectra) {
  # NMF library needed for the fcnnls function
  require(NMF)

  s <- fcnnls(all.sigs, spectra) # all.sigs is defined globally
  cis.pos <- s$x['Cisplatin', ] > 0
  colnames(s$x)[cis.pos, drop=F]
}

```

The results appear below. Tumor IDs prefixed by P were identified as cisplatin positive by both the mSigAct signature presence test and DNS signature analysis.

```
cisplatin.gt.0(all.spectra$ESAD)
```

```
## [1] " SA130952" " SA528868" " SA594893" "P SA594320" "P SA594557"
## [6] "P SA594775" " SA594906" " SA594988" " SA594992"
```

```
cisplatin.gt.0(all.spectra$`Fujimoto HCC`)
```

```
## [1] "P RK028" " RK029" " RK042" "P RK056" " RK067" " RK072" "P RK074"
## [8] " RK093" " RK101" " RK103" " RK105" " RK116" " RK128" " RK140"
## [15] " RK188" "P RK205" " RK213" " RK223" " RK225" " RK227" "P RK241"
## [22] " RK243" " RK244" "P RK256" " RK275" " RK338"
```

```
cisplatin.gt.0(all.spectra$`Kan HCC`)
```

```
## [1] " HK029" "P HK034" " HK035" " HK017" " HK068" " HK172" " HK203"
## [8] " HK021" " HK266" " HK276" " HK038" " HK071"
```