

Oncoplot

T. Tavella

Figure S5 A. Oncoplot

```
table(mutations$patient, mutations$Biomarker)
```

```
##
##      ABL1 ASXL1 CBL  CEBPA DNMT3A FLT3 IDH1 IDH2 KRAS NF1 NPM1 NRAS PTPN11
##  UPN01    0    0  0      0      0    0    0    0    0    0    0    1    0
##  UPN02    0    0  0      0      0    1    0    0    0    0    0    0    1
##  UPN03    1    0  1      0      1    1    1    0    0    0    1    0    0
##  UPN04    0    0  0      0      0    1    0    0    0    0    1    0    0
##  UPN05    0    0  0      0      1    1    0    0    0    0    0    0    0
##  UPN06    0    0  0      0      1    1    0    0    0    0    0    0    0
##  UPN07    0    1  0      0      0    1    0    0    0    0    0    0    0
##  UPN08    0    0  0      1      0    0    0    0    0    0    0    0    0
##  UPN09    0    0  0      0      0    0    0    0    0    0    0    1    0
##  UPN10    0    0  1      0      1    1    0    0    0    0    1    0    0
##  UPN11    0    0  0      0      0    0    0    0    0    0    0    1    0
##  UPN12    0    1  0      0      0    0    0    0    0    0    0    0    0
##  UPN14    0    0  0      0      0    1    0    0    0    0    0    0    0
##  UPN16    0    1  0      0      0    0    0    0    2    0    0    0    0
##  UPN17    0    0  0      0      0    0    0    1    0    1    0    0    0
##  UPN18    0    0  0      0      0    0    0    0    0    2    0    0    0
##  UPN19    0    0  0      0      0    0    0    0    0    0    0    0    0
##  UPN22    0    0  0      0      1    0    0    0    0    0    1    0    1
##  UPN23    0    0  0      2      0    2    0    0    0    0    1    0    0
##
##      RUNX1 STAG2 TET2 TP53 U2AF1 WT1
##  UPN01    0    0    0    0    1    0
##  UPN02    0    0    0    0    0    0
##  UPN03    0    0    0    0    0    0
##  UPN04    0    0    0    0    0    1
##  UPN05    0    0    1    0    0    0
##  UPN06    0    0    0    0    0    0
##  UPN07    0    0    0    0    0    0
##  UPN08    0    0    0    0    0    1
##  UPN09    0    0    0    0    0    0
##  UPN10    0    0    0    0    0    0
##  UPN11    0    0    0    0    0    0
##  UPN12    0    0    0    1    1    0
##  UPN14    0    0    0    0    0    0
##  UPN16    0    0    0    1    0    0
##  UPN17    2    1    0    0    0    0
##  UPN18    0    0    0    0    0    0
##  UPN19    0    0    1    0    0    0
##  UPN22    0    0    0    0    0    0
##  UPN23    0    0    0    0    0    0
```

```

mutations$Impact <- ifelse(mutations$Impact == '-', 'unknown', mutations$Impact )

slow <- c('UPN01', 'UPN02', 'UPN03', 'UPN04', 'UPN05', 'UPN12')
mutations$group <- ifelse(mutations$patient %in% slow, 'SenLow', 'SenHigh')

mat = dcast(mutations, Biomarker~patient)
rownames(mat)<- mat$Biomarker
mat <- mat[,-c(1)]

n <- 1:dim(mat)[[1]]
for (i in n){
  gene <- rownames(mat[i,])
  for (pat in colnames(mat)){
    p <- mat[gene,pat]
    if (p != 0) {
      print(i)
      print(p)
      print(gene)
      print(pat)
      val = gsub(' ', '_', as.character(paste(mutations[mutations$Biomarker == gene
& mutations$patient == pat, 'Impact', drop=TRUE],
                                             collapse = ';')))
      val = gsub('^-', 'unknown', val)
      val = gsub('In-Frame', 'In_Frame', val)
      print(val)
      mat[gene, pat] <- val
    }
  }
}

mat[mat == 0] <- ''

ordermat <- c("UPN01", "UPN02", "UPN03", "UPN04", "UPN05", "UPN12",
              "UPN06", "UPN07", "UPN08", "UPN09", "UPN10", "UPN11", "UPN14", "UPN1
6",
              "UPN17", "UPN18", "UPN19", "UPN22", "UPN23")
mat <- mat[,ordermat]

## Adjust code
for ( i in 1:dim(mat)[1]) {
  for (j in 1:dim(mat)[2]) {
    val = mat[i,j]
    if (grepl(';', val)) {
      # split condition
      print(mat[i,j])
      v <- str_split( val, ';')
    }
  }
}

```

```

    #if (v[[1]][2] !=
    if (val == 'Missense;Missense') {
      val <- "Missense;Multi_hit_missense"
      mat[i,j] <- val
    } else if (val == "Frameshift;Frameshift" ) {
      val <- "Frameshift;Multi_hit_frameshift"
      mat[i,j] <- val
    } else if (val == "In_Frame;In_Frame") {
      val <- 'In_Frame;Multi_hit_in_frame'
      mat[i,j] <- val
    }
    else if (val == "unknown;unknown") {
      val <- 'unknown;Multi_hit_unknown'
      mat[i,j] <- val
    }
  }
}
}

```

```

## [1] "Frameshift;Frameshift"
## [1] "In_Frame;In_Frame"
## [1] "Missense;Missense"
## [1] "Frameshift;Splicing"
## [1] "Missense;Frameshift"

```

adjust function

```

for ( i in 1:dim(mat)[1]) {
  for (j in 1:dim(mat)[2]) {
    val = mat[i,j]
    if (grepl(';', val)) {
      # split condition
      print(mat[i,j])
      v <- str_split( val, ';')
      #if (v[[1]][2] !=
      if (val == 'Missense;Missense') {
        val <- "Missense;Multi_hit"
        mat[i,j] <- val
      } else if (val == "Frameshift;Frameshift" ) {
        val <- "Frameshift;Multi_hit"
        mat[i,j] <- val
      } else if (val == "In_Frame;In_Frame") {
        val <- 'In_Frame;Multi_hit'
        mat[i,j] <- val
      }
      else if (val == "unknown;unknown") {
        val <- 'unknown;Multi_hit'
        mat[i,j] <- val
      }
    }
  }
}
}
}

```

```

## [1] "Frameshift;Multi_hit_frameshift"
## [1] "In_Frame;Multi_hit_in_frame"
## [1] "Missense;Multi_hit_missense"
## [1] "Frameshift;Splicing"
## [1] "Missense;Frameshift"

col = c("Multi_hit_missense" = 'black',
        "Multi_hit_frameshift" = 'red',
        "Multi_hit_in_frame" = 'cyan1',
        "Multi_hit_unknown" = 'grey60',
        'unknown' = 'orange', 'Synonymous' = 'yellow',
        "Missense" = "dodgerblue3", "Frameshift" = "brown3", "In_Frame" = "magenta
4", "Splicing" = 'forestgreen', "Stop_Gain" = "darkorange")

alter_fun = list(
  background = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h-unit(2, "pt"),
              gp = gpar(fill = "#CCCCCC", col = NA))
  },
  # big blue
  Missense = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h-unit(2, "pt"),
              gp = gpar(fill = col["Missense"], col = NA))
  },
  # big red
  Frameshift = function(x, y, w, h) {
    grid.rect(x, y, w*0.33, h-unit(2, "pt"),
              gp = gpar(fill = col["Frameshift"], col = NA))
  },
  # small green
  In_Frame = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h*0.33,
              gp = gpar(fill = col["In_Frame"], col = NA))
  },
  Splicing = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h*0.33,
              gp = gpar(fill = col["Splicing"], col = NA))
  },
  Stop_Gain = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h*0.33,
              gp = gpar(fill = col["Stop_Gain"], col = NA))
  },
  Multi_hit_missense = function(x, y, w, h) {
    grid.rect(x, y, w*0.33, h-unit(2, "pt"),
              gp = gpar(fill = col["Multi_hit_missense"], col = NA))
  },
  Multi_hit_frameshift = function(x, y, w, h) {

```

```

    grid.rect(x, y, w-unit(2, "pt"), h*0.33,
              gp = gpar(fill = col["Multi_hit_frameshift"], col = NA))
  },
  Multi_hit_in_frame = function(x, y, w, h) {
    grid.rect(x, y, w*0.33, h-unit(2, "pt"),
              gp = gpar(fill = col["Multi_hit_in_frame"], col = NA))
  },
  #Multi_hit_unknown = function(x, y, w, h) {
  #   grid.rect(x, y, w*0.33, h-unit(2, "pt"),
  #             gp = gpar(fill = col["Multi_hit_unknown"], col = NA))
  #},
  unknown = function(x, y, w, h) {
    grid.points(x, y, pch = 16, size = unit(0.5, "char"),
               gp = gpar(col = col["unknown"]))
  },

  Synonymous = function(x, y, w, h) {
    grid.rect(x, y, w-unit(2, "pt"), h-unit(2, "pt"),
              gp = gpar(fill = col["Synonymous"], col = NA))
  }
)

d <- oncoPrint(mat, show_column_names = TRUE, column_order = colnames(mat),
               top_annotation = HeatmapAnnotation(
                 column_barplot = anno_oncoprint_barplot(show_fraction = FALSE),
                 border = TRUE #, height = unit(4, "cm"),
               ),
               right_annotation = rowAnnotation(
                 row_barplot = anno_oncoprint_barplot(show_fraction = FALSE),
                 border = TRUE
               ), column_names_rot = 45,
               row_names_gp = gpar(fontsize = 12), #fontfamily = "Arial", fontface
               = 'italic'),

               column_names_gp = gpar(fontsize = 12), #fontfamily = "Arial"),
               pct_gp = gpar(fontsize = 12), #fontfamily = "Arial"),
               #pct_side = "right", #row_names_side = "left",
               alter_fun = alter_fun, col = col, border = TRUE)

wdir_new= '/beegfs/scratch/ric.dimicco/ric.dimicco/Diego1330_Genewiz_AML_ARA_CDK/P
aper_Analysis_v9'
plot(d)

```

