Static_Redox_Graphs

Julian Stanley

September 24, 2018

3 parameters –> OxD

```
FOx <- function(I, RR, ROX, delta) {
      return (
        (I - RR)/((I - RR) + (delta*(ROX - I)))
      )
}
```

Plot: OxD as a function of measured intensity, with a constant maximal and minimal intensity

Assume that $R_Red = 0.667$ and $R_Ox = 5.207$. Vary δ by factors of 2.

```
require(RColorBrewer)
```

```
## Loading required package: RColorBrewer
```

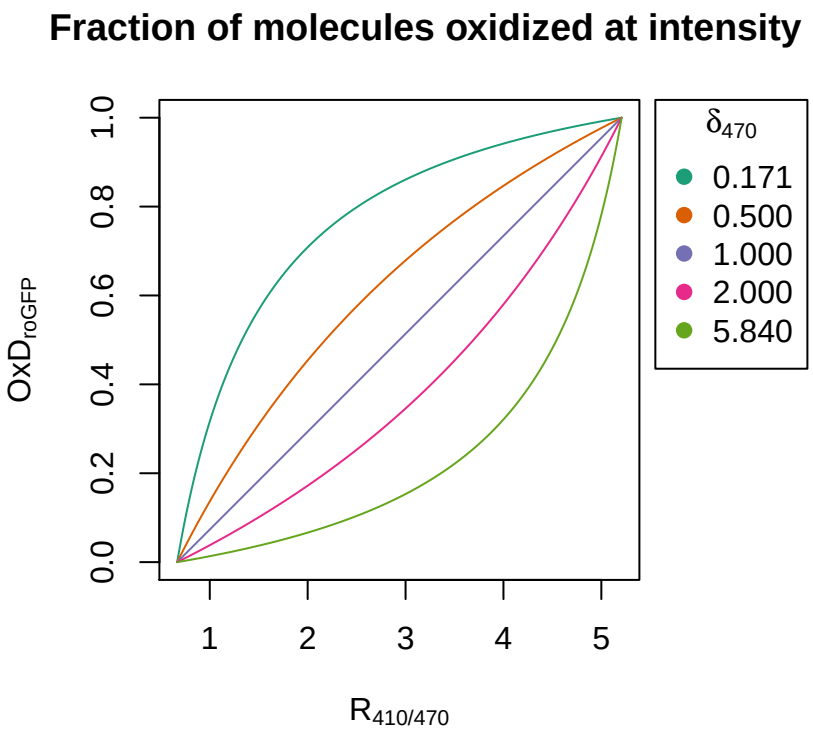
```
## Warning: package 'RColorBrewer' was built under R version 3.4.1
```

```
colors <- brewer.pal(n = 16, 'Dark2')
```

```
## Warning in brewer.pal(n = 16, "Dark2"): n too large, allowed maximum for palette Dark2 is 8
## Returning the palette you asked for with that many colors
```

```
maxRed <- 0.667
maxOx <- 5.207
delta_values <- c(0.171, 0.5, 1.0, 2.0, 5.84)

x <- seq(maxRed, maxOx, by = 0.001)
magX <- length(x)

yOx = FOx(x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(delta_values[1], each = magX))

par(mar=c(5, 5, 5, 8), pty = 's', bg = NA)
plot(x, yOx,
        type = 'l', main = "
     Fraction of molecules oxidized at intensity",
        ylab = expression('OxD'['roGFP']), xlab = expression('R'['410/470']),
     col = colors[1])



for (i in 2:length(delta_values)) {
  yOx_i <- FOx(x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(delta_values[i], each = magX))
```

```
  points(x, yOx_i, col = colors[i], type = 'l')

}

options(digits = 4)
legend("topright", title = expression(delta['470']), inset=c(-0.35,0), xpd=TRUE, as.character(format(del
```




```
dev.copy(pdf,'plot1.pdf', compress = FALSE)

## pdf
##   3
```

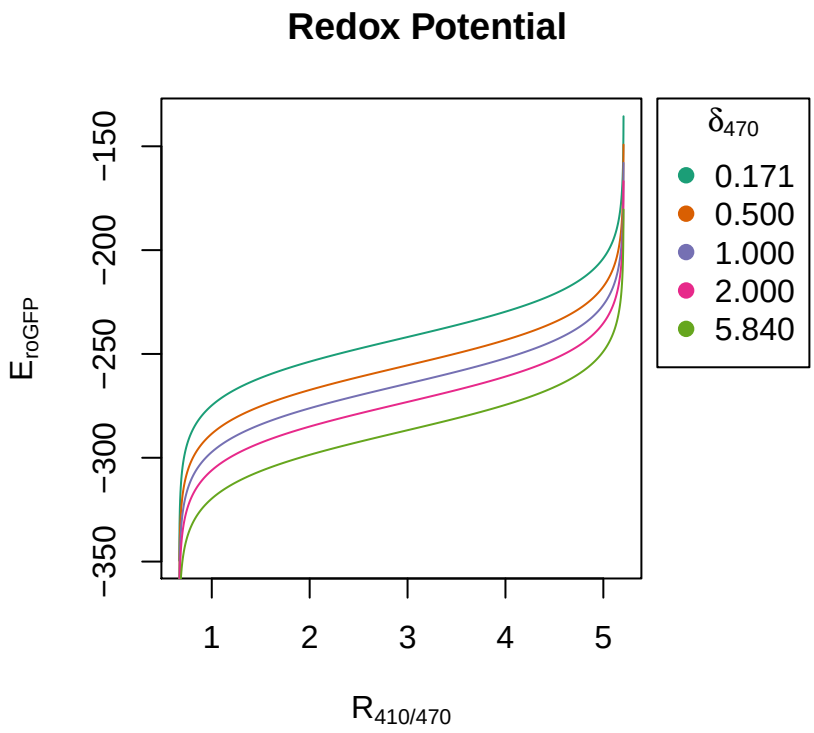
4 parameters –> E

At fixed E = -265, vary delta

```
# Define the Nernst function
    FE <- function(e0, I, RR, ROX, delta) {
      return(e0 - 12.71 * log((1-FOx(I, RR, ROX, delta))/FOx(I, RR, ROX, delta)))
    }
```

```
yE = FE(-265, x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(delta_values[1], each = magX))

par(mar=c(5, 5, 5, 8), pty = 's', bg = NA)
```

```
plot(x, yE,
        type = 'l', main = "
     Redox Potential",
        ylab = expression('E'['roGFP']), xlab = expression('R'['410/470']),
     col = colors[1])



for (i in 2:length(delta_values)) {
  yE_i <- FE(-265, x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(delta_values[i], each = magX))
  points(x, yE_i, col = colors[i], type = 'l')

}

options(digits = 4)
legend("topright", title = expression(delta['470']), inset=c(-0.35,0), xpd=TRUE, as.character(format(de
```

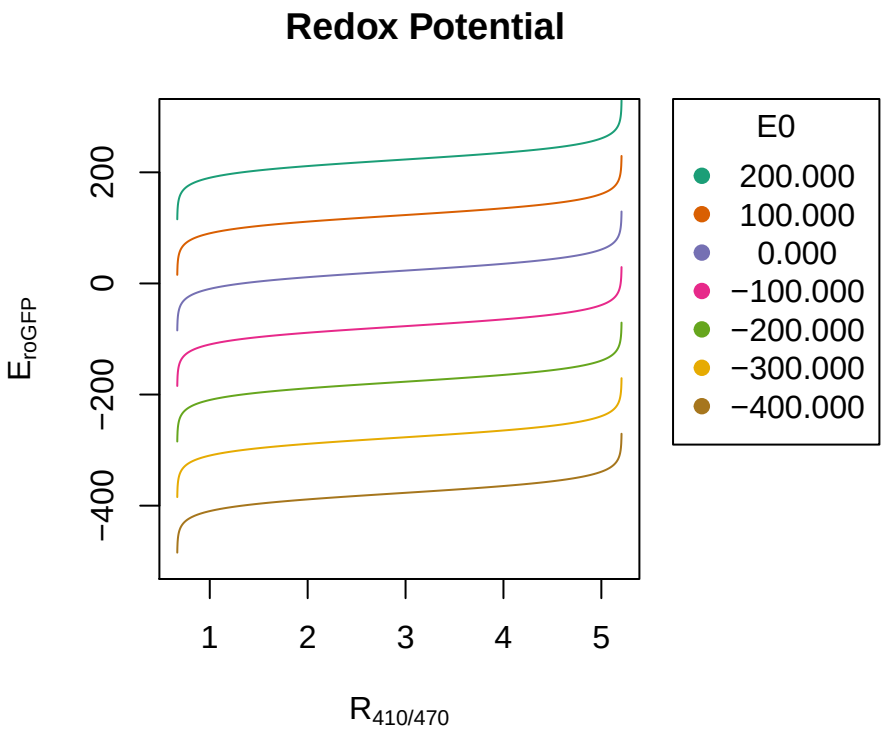



```
dev.copy(pdf,'plot2.pdf', compress = FALSE)
```

```
## pdf
##   4
```

** At fixed $\delta = 0.171$, vary E0 **

```
E0_values <- c(200, 100,0,  -100, -200, -300, -400)

yE = FE(E0_values[1], x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(0.171, each = magX))

par(mar=c(5, 5, 5, 8), pty = 's', bg = NA)

plot(x, yE,
        type = 'l', main = "
     Redox Potential",
        ylab = expression('E'['roGFP']), xlab = expression('R'['410/470']),
     col = colors[1], ylim = c(-500, 300))



for (i in 2:length(E0_values)) {
  yE_i <- FE(E0_values[i], x, rep(maxRed, each = magX),
        rep(maxOx, each = magX),
        rep(0.171, each = magX))
  points(x, yE_i, col = colors[i], type = 'l')

}

options(digits = 4)
legend("topright", title = 'E0', inset=c(-0.5,0), xpd=TRUE, as.character(format(E0_values, nsmall = 3))
```




```
dev.copy(pdf,'plot2.pdf', compress = FALSE)
```

```
## pdf
##   5
```