

# ChangePoint change detection test

## 1 Second test using stable periods in changepoint

This is a second test for using the changepoint package and PELT algorithm. However changepoint works only on gap-free data, therefore this methodology is a bit more coarse than the other. Choose to aggregate to annual time series using a median composite. Loosing a lot of the variation within, but well...

### 1.0.1 Load data

```
path = "HackTimeSeries/"

# Load all files
ll = lapply( list.files(path, ".feather", full.names = T), function(x) read_feather(x) )

# Name them
names(ll) <- file_path_sans_ext( list.files(path, ".feather") )
```

Now aggregate to annual time series and linearly interpolate over gaps

```
band = "EVI2"
as.year <- function(x) as.integer(as.yearmon(x))

ll2 <- list()
for(i in names(ll)){
  sub <- as.data.frame( ll[[i]] )
  # Define time series
  zz <- suppressWarnings( zooreg(data = sub[,band], order.by = sub[, "date"] ) )

  # Aggregate to monthly data using a max value composite
  xx = suppressWarnings( aggregate(zz, as.year, function(x) median(x, na.rm = T)) )
  xx[which(is.infinite(xx))] <- NA

  xx <- na.approx(xx)
  ll2[[i]] <- xx
  rm(xx)
}
```

Now apply the changepoint PELT algorithm for mean shifts on the annual aggregations. This uses an information theory approach to detect stable periods using the PELT algorithm. As minimum segment length I specify exactly 2 years

```
stabPer <- function(x, cp){
  cp.l <- cpts(cp) # Location of changepoints
  # Return a list of subsets
  o <- list(x[1:cp.l[1]]) # The first one
  cp.l <- c(cp.l, length(x)) # Remove the first and add the last
  for(i in 1:(length(cp.l)-1)){
    o[[i+1]] <- x[cp.l[i]:cp.l[i+1]]
  }
  return(o)
}
```

```

# Do the loopie
res <- data.frame()
# Do the loopie
for( i in names(l12)){
  x = l12[[i]]
  # Change in mean/variance
  try( cp <- cpt.meanvar(x,penalty = "BIC", method = "PELT",minseglen = 2), silent= T)

  # If that did work and there are more breaks than one
  if(class(cp)!="try-error" & nseg(cp) > 1){
    # Number of changepoints - How many mean+variance shifts happened per pixel (if any)?
    nr_cp <- ncpts(cp)
    # Max length (months) of stable period (no mean/variance shift)
    nr_Mcp <- max(seg.len(cp))
    # What was its mean estimate ?
    lss <- stabPer(x,cp) # Get list of subsets
    nr_MAVG_cp <- mean( lss[[which.max(seg.len(cp))]] )
    # How long ago did it occur (if the entire pixel time series isn't stable).
    nr_Mcptimeago <- time(x)[ cpts(cp)[which.max(seg.len(cp))] ]
    # What was the greatest magnitude between shifts (difference in mean of stable periods)?
    lss_d <- diff(unlist(lapply(lss,mean))) # Subsequential subtraction
    nr_Mmag <- lss_d[which.max(abs(lss_d))]
    # When did this occur ?
    nr_Mmagtime <- time(x)[cpts(cp)[which.max(abs(lss_d))]]

    res <- rbind(res, data.frame( SSBS = i,nr_Mmag,nr_Mmagtime ))
  }
}

```

Now assess the match between the “validation” columns. So merge back with the original

```

# Assemble full
df <- do.call("rbind",l1)
# Construct names by replacing space with underscores
df$mergeSSBS <- str_replace_all(df$SSBS,pattern = " ",replacement = "_")

# make troubles
res$SSBS <- as.character(res$SSBS)

# Subset to columns of interest
sub <- df %>% dplyr::mutate(SampleStart = year( Sample_start_earliest ) ) %>%
  dplyr::select(SSBS,mergeSSBS,SampleStart,Hansenlossyear, YearsOfConversion) %>% distinct_() %>%
  # Merge
  merge.data.frame(.,res,by.x = "mergeSSBS",by.y = "SSBS")

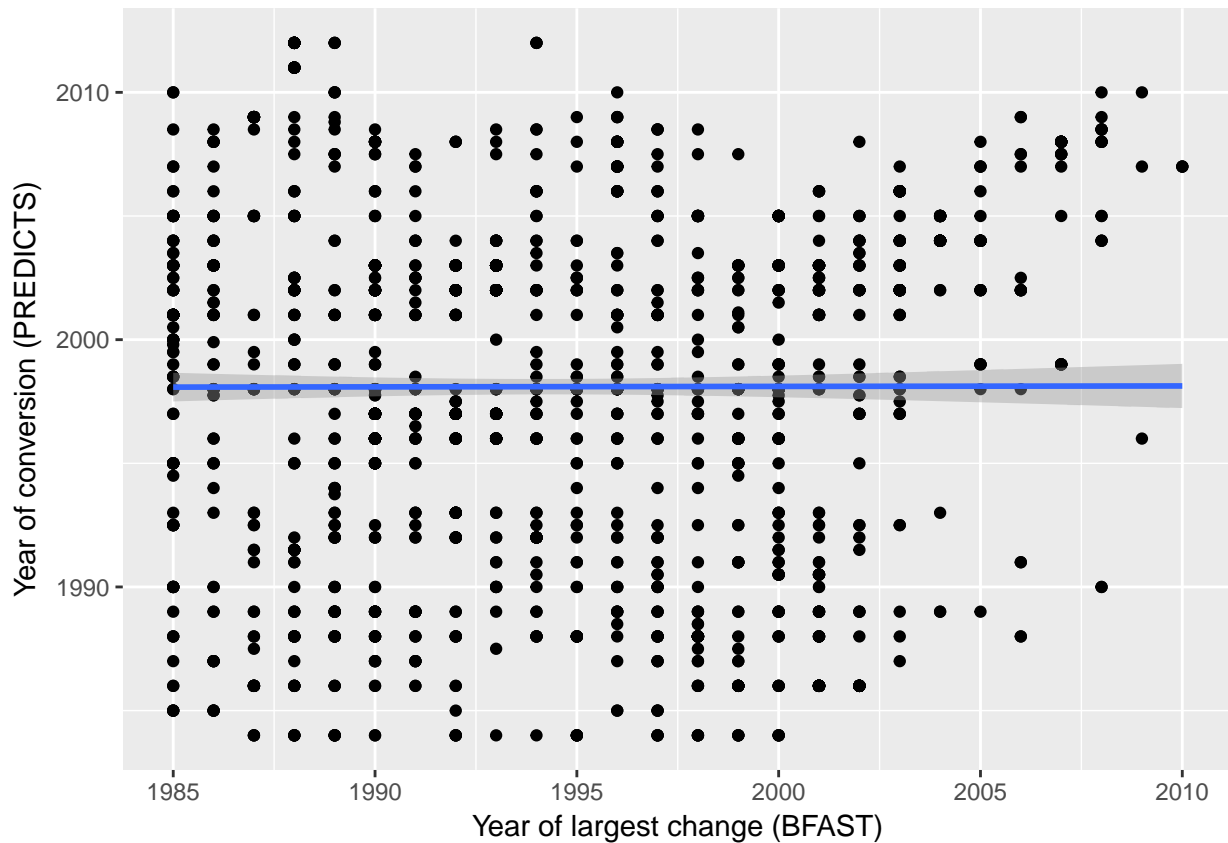
```

Now assess the match (if any)

```

qplot(sub$nr_Mmagtime,sub$YearsOfConversion,xlab = "Year of largest change (BFAST)",ylab = "Year of conversion")
  geom_smooth(method="lm")

```



```
cor.test(sub$nr_Mmagtime,sub$YearsOfConversion)
```

```
##
## Pearson's product-moment correlation
##
## data: sub$nr_Mmagtime and sub$YearsOfConversion
## t = 0.069327, df = 1885, p-value = 0.9447
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.04353087 0.04671794
## sample estimates:
## cor
## 0.001596783
```

```
summary(lm(sub$nr_Mmagtime~sub$YearsOfConversion))
```

```
##
## Call:
## lm(formula = sub$nr_Mmagtime ~ sub$YearsOfConversion)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -9.231 -4.221 -1.220  4.788 15.773
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.991e+03  3.920e+01  50.804  <2e-16 ***
## sub$YearsOfConversion 1.360e-03  1.962e-02   0.069   0.945
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.878 on 1885 degrees of freedom
## Multiple R-squared:  2.55e-06,    Adjusted R-squared:  -0.000528
## F-statistic: 0.004806 on 1 and 1885 DF,  p-value: 0.9447
```

Answer: **NO** Repeat: **Don't replicate**