PHYS598 - Data Analysis - March 14, 2019

Reading the data in the structure-specific CSV file as an R “dataframe”

```r
DVH.CTV7000 <- read.csv("CTV7000.csv", header = TRUE, row.names = NULL)
DVH.CTV5940 <- read.csv("CTV5940.csv", header = TRUE, row.names = NULL)
DVH.Brainstem <- read.csv("Brainstem.csv", header = TRUE, row.names = NULL)
DVH.BrainstemMarg <- read.csv("BrainstemMarg.csv", header = TRUE, row.names =
NULL)
DVH.ParotidLT <- read.csv("ParotidLT.csv", header = TRUE, row.names = NULL)
DVH.ParotidRT <- read.csv("ParotidRT.csv", header = TRUE, row.names = NULL)
DVH.Pharynx <- read.csv("Pharynx.csv", header = TRUE, row.names = NULL)
DVH.SpinalCord <- read.csv("SpinalCord.csv", header = TRUE, row.names = NULL)
DVH.SpinalCordMarg <- read.csv("SpinalCordMarg.csv", header = TRUE, row.names
= NULL)
DVH.SubmandLT <- read.csv("SubmandLT.csv", header = TRUE, row.names = NULL)
DVH.SubmandRT <- read.csv("SubmandRT.csv", header = TRUE, row.names = NULL)
```

Most important structures with specific dose parameters

```r
CTV7000_D95 = DVH.CTV7000[,c(1,3,98)]
CTV7000_D99 = DVH.CTV7000[,c(1,3,102)]
CTV7000_D20 = DVH.CTV7000[,c(1,3,23)]
CTV5940_D99 = DVH.CTV5940[,c(1,3,102)]
CTV5940_D95 = DVH.CTV5940[,c(1,3,98)]
CTV5940_D20 = DVH.CTV5940[,c(1,3,23)]
Brainstem_D1 = DVH.Brainstem[,c(1,3,4)]
ParotidLT_D50 = DVH.ParotidLT[,c(1,3,53)]
ParotidRT_D50 = DVH.ParotidRT[,c(1,3,53)]
SpinalCord_D1 = DVH.SpinalCord[,c(1,3,4)]
```

Obtaining the row numbers with new patient

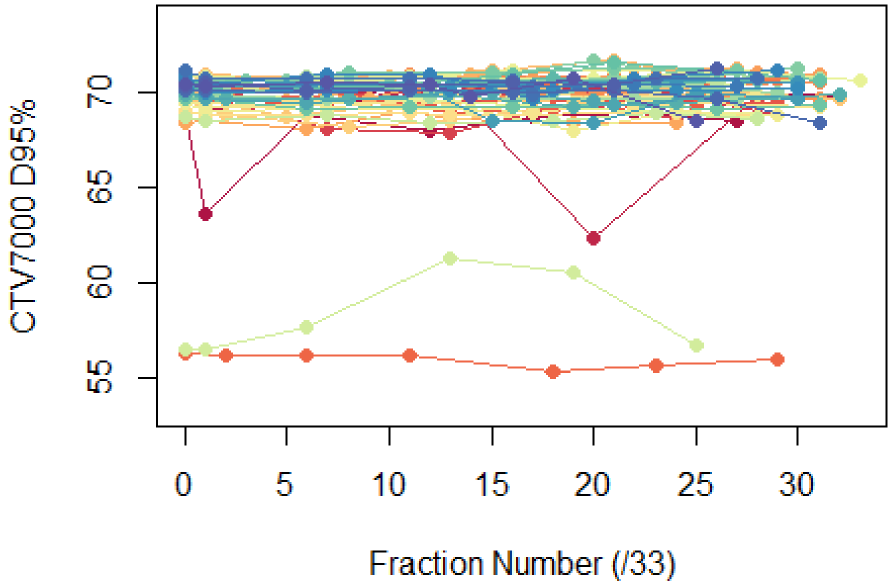
```r
#Deleting NA entries
CTV7000_D99 <- CTV7000_D99[complete.cases(CTV7000_D99),]
CTV5940_D99 <- CTV5940_D99[complete.cases(CTV5940_D99),]
Brainstem_D1 <- Brainstem_D1[complete.cases(Brainstem_D1),]
ParotidLT_D50 <- ParotidLT_D50[complete.cases(ParotidLT_D50),]
ParotidRT_D50 <- ParotidRT_D50[complete.cases(ParotidRT_D50),]
SpinalCord_D1 <- SpinalCord_D1[complete.cases(SpinalCord_D1),]

CTV7000_ptrows<- c(1);
for (i in 1:(length(CTV7000_D99[,1])-1))
{
  if  (!(CTV7000_D99[i,1] == CTV7000_D99[i+1,1]))
    {CTV7000_ptrows <- c(CTV7000_ptrows,i+1)}
}

CTV5940_ptrows<- c(1);
```

```r
for (i in 1:(length(CTV5940_D99[,1])-1))
{
  if  (!(CTV5940_D99[i,1] == CTV5940_D99[i+1,1]))
    {
    CTV5940_ptrows <- c(CTV5940_ptrows,i+1)}
}

Brainstem_ptrows<- c(1);

for (i in 1:(length(Brainstem_D1[,1])-1))
{
  if  (!(Brainstem_D1[i,1] == Brainstem_D1[i+1,1]))
    {
    Brainstem_ptrows <- c(Brainstem_ptrows,i+1)}
}

ParotidLT_ptrows<- c(1);

for (i in 1:(length(ParotidLT_D50[,1])-1))
{
  if  (!(ParotidLT_D50[i,1] == ParotidLT_D50[i+1,1]))
    {
    ParotidLT_ptrows <- c(ParotidLT_ptrows,i+1)}
}

ParotidRT_ptrows<- c(1);

for (i in 1:(length(ParotidRT_D50[,1])-1))
{
  if  (!(ParotidRT_D50[i,1] == ParotidRT_D50[i+1,1]))
    {
    ParotidRT_ptrows <- c(ParotidRT_ptrows,i+1)}
}

SpinalCord_ptrows<- c(1);

for (i in 1:(length(SpinalCord_D1[,1])-1))
{
  if  (!(SpinalCord_D1[i,1] == SpinalCord_D1[i+1,1]))
    {
    SpinalCord_ptrows <- c(SpinalCord_ptrows,i+1)}
}

# Max function for tables with NA
my.max <- function(x) ifelse( !all(is.na(x)), max(x, na.rm=T), NA)
my.min <- function(x) ifelse( !all(is.na(x)), min(x, na.rm=T), NA)

library("RColorBrewer")
```

```
## Warning: package 'RColorBrewer' was built under R version 3.5.2

#Initializing a color vector to help us color code our graphs for each
patient
color = brewer.pal(n = 10,name = "Spectral")
color = colorRampPalette(color)(60)
```

Plotting CTV7000 D95% in time

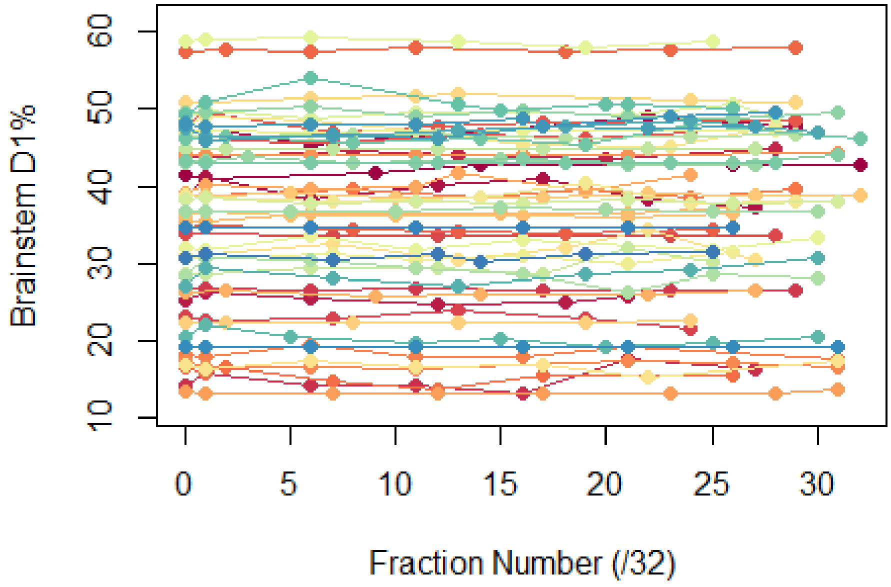
```
plot(NULL,xlab="Fraction Number (/33)", ylab="CTV7000 D95%",xlim =
c(0,33),ylim = c(my.min(CTV7000_D95$D95.Gy) - 2,my.max(CTV7000_D95$D95.Gy) +
2))

#Iterating through the patients and plotting trends in D95% with fraction
number
for (i in 1:(length(CTV7000_ptrows)-1) )
{
  rm(vec)
  ind1 = CTV7000_ptrows[i]
  ind2 = CTV7000_ptrows[i+1]-1
  vec <- data.frame(Fraction.Number = CTV7000_D95[c(ind1:ind2),2], D95.Gy =
CTV7000_D95[c(ind1:ind2),3])
  points(vec$Fraction.Number,vec$D95.Gy,pch = 19, col = color[i])
  lines(vec$Fraction.Number,vec$D95.Gy,pch = 19, col = color[i])
}

## Warning in rm(vec): object 'vec' not found
```



- more or less consistent

Brainstem D1%

```
plot(NULL,xlab="Fraction Number (/32)", ylab="Brainstem D1%",xlim =
c(0,32),ylim = c(my.min(Brainstem_D1$D1.Gy) - 2,my.max(Brainstem_D1$D1.Gy) +
2))

#Iterating through the patients and plotting trends with fraction number
for (i in 1:(length(Brainstem_ptrows)-1) )
{
  rm(vec)
  ind1 = Brainstem_ptrows[i]
  ind2 = Brainstem_ptrows[i+1]-1
  vec <- data.frame(Fraction.Number = Brainstem_D1[c(ind1:ind2),2], D1.Gy =
Brainstem_D1[c(ind1:ind2),3])
  points(vec$Fraction.Number,vec$D1.Gy,pch = 19, col = color[i])
   lines(vec$Fraction.Number,vec$D1.Gy,pch = 19, col = color[i])
}
```

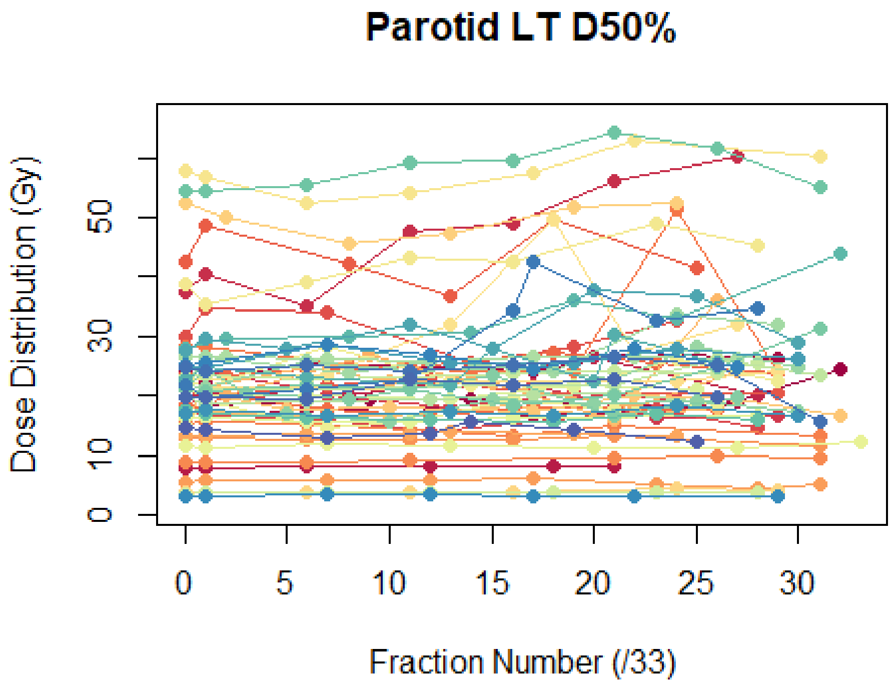


also consistent

Parotid LT D50%

```
plot(NULL,xlab="Fraction Number (/33)", ylab="Dose Distribution (Gy)", main =
"Parotid LT D50%",xlim = c(0,33),ylim = c(my.min(ParotidLT_D50$D50.Gy) -
2,my.max(ParotidLT_D50$D50.Gy) + 2))
```

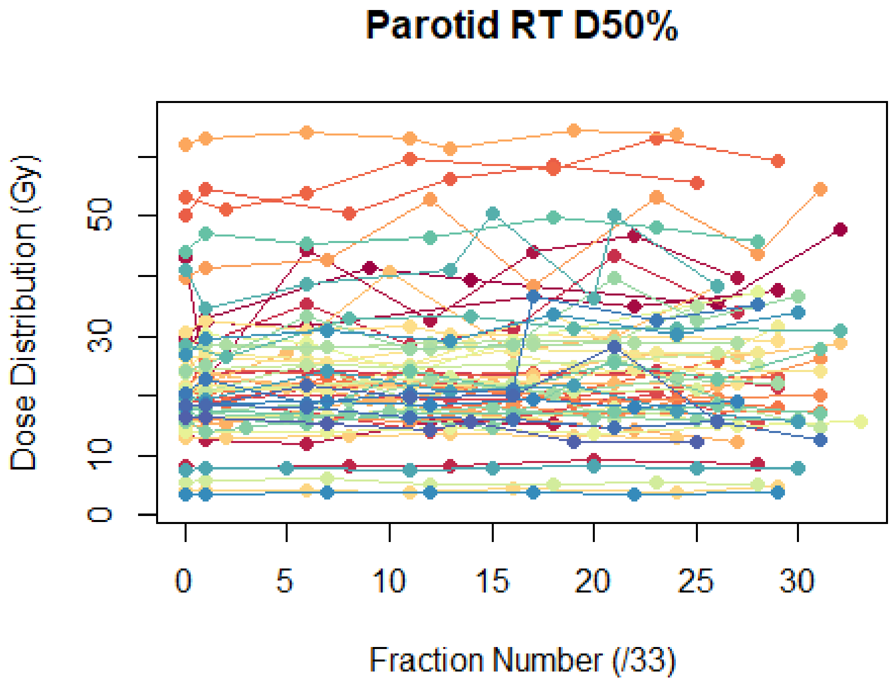
```r
#Iterating through the patients and plotting trends with fraction number
for (i in 1:(length(ParotidLT_ptrows)-1) )
{
  rm(vec)
  ind1 = ParotidLT_ptrows[i]
  ind2 = ParotidLT_ptrows[i+1]-1
  vec <- data.frame(Fraction.Number = ParotidLT_D50[c(ind1:ind2),2], D50.Gy =
ParotidLT_D50[c(ind1:ind2),3])
  points(vec$Fraction.Number,vec$D50.Gy,pch = 19, col = color[i])
   lines(vec$Fraction.Number,vec$D50.Gy,pch = 19, col = color[i])
}
```



Parotid RT D50%

```r
library("RColorBrewer")
#Initializing a color vector to help us color code our graphs for each
patient
color = brewer.pal(n = 10,name = "Spectral")
color = colorRampPalette(color)(60)

#R needs a generic "plot" call before "points" or "lines" can be superimposed

plot(NULL,xlab="Fraction Number (/33)", ylab="Dose Distribution (Gy)", main =
"Parotid RT D50%",xlim = c(0,33),ylim = c(my.min(ParotidRT_D50$D50.Gy) -
2,my.max(ParotidRT_D50$D50.Gy) + 2))
```

```
#Iterating through the patients and plotting trends with fraction number
for (i in 1:(length(ParotidRT_ptrows)-1) )
{
  rm(vec)
  ind1 = ParotidRT_ptrows[i]
  ind2 = ParotidRT_ptrows[i+1]-1
  vec <- data.frame(Fraction.Number = ParotidRT_D50[c(ind1:ind2),2], D50.Gy =
ParotidRT_D50[c(ind1:ind2),3])
  points(vec$Fraction.Number,vec$D50.Gy,pch = 19, col = color[i])
   lines(vec$Fraction.Number,vec$D50.Gy,pch = 19, col = color[i])
}
```



- both parotids are changing quite a bit in time.

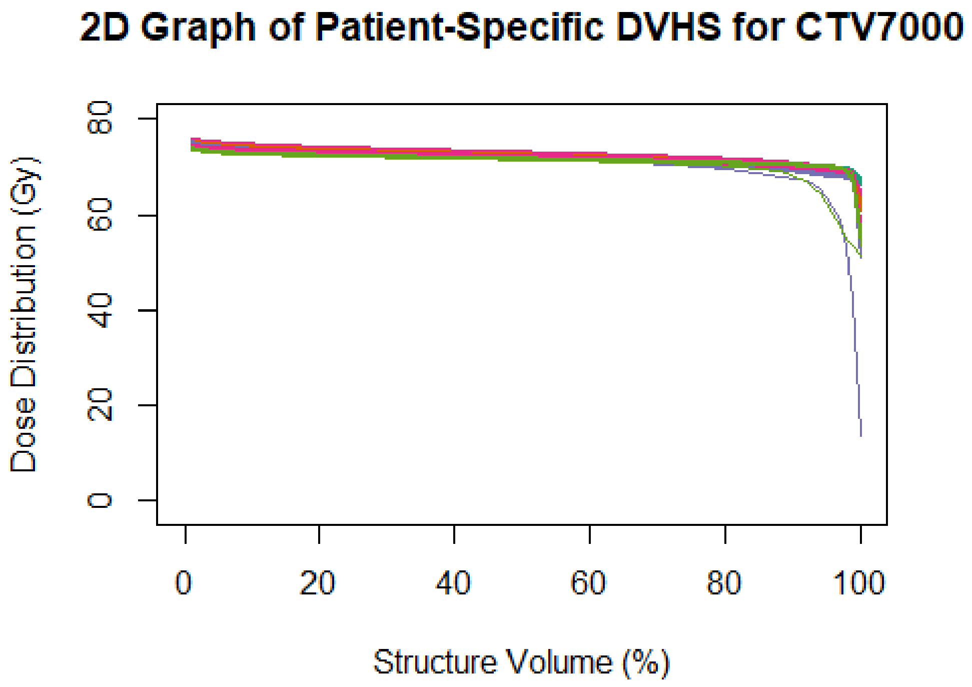
Plotting Full DVH plots in 2D

- to visualize the spread, and what changes look like beyond just conventional Dxx%

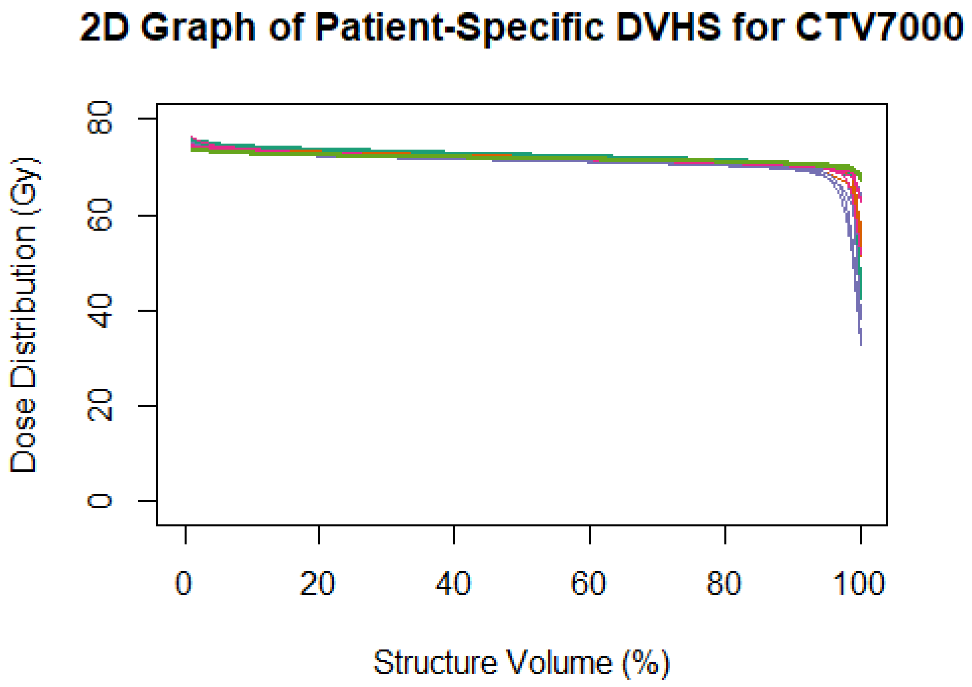
```
library("RColorBrewer")
#Initializing a color vector to help us color code our graphs for each
patient
color = brewer.pal(n = 5,name = "Dark2")
color = colorRampPalette(color)(5)

DVHplot.ParotidRT <- function(x,y)
{
  plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",xlim =
c(0,100), ylim = c(my.min(DVH.ParotidRT[,c(4:103)]) -
2,my.max(DVH.ParotidRT[,c(4:103)]) + 2), main="2D Graph of Patient-Specific
DVHS for Parotid RT")
  for (i in x:y )
{
  for (j in 0:(ParotidRT_ptrows[i+1]-ParotidRT_ptrows[i] - 1) )
  {
    ind1 = ParotidRT_ptrows[i] + j
    lines(seq(1,100,1),DVH.ParotidRT[ind1,c(4:103)],pch = 19, col = color[i-
x+1])
  }
}
}

DVHplot.ParotidLT <- function(x,y)
{
  plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",xlim =
c(0,100), ylim = c(my.min(DVH.ParotidLT[,c(4:103)]) -
2,my.max(DVH.ParotidLT[,c(4:103)]) + 2), main="2D Graph of Patient-Specific
DVHS for Parotid LT")
  for (i in x:y )
{
  for (j in 0:(ParotidLT_ptrows[i+1]-ParotidLT_ptrows[i] - 1) )
  {
    ind1 = ParotidLT_ptrows[i] + j
    lines(seq(1,100,1),DVH.ParotidLT[ind1,c(4:103)],pch = 19, col = color[i-
x+1])
  }
}
}
DVHplot.CTV7000 <- function(x,y)
{
  plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",xlim =
c(0,100), ylim = c(my.min(DVH.CTV7000[,c(4:103)]) -
2,my.max(DVH.CTV7000[,c(4:103)]) + 2), main="2D Graph of Patient-Specific
DVHS for CTV7000")
  for (i in x:y )
{
  for (j in 0:(CTV7000_ptrows[i+1]-CTV7000_ptrows[i] - 1) )
  {
    ind1 = CTV7000_ptrows[i] + j
    lines(seq(1,100,1),DVH.CTV7000[ind1,c(4:103)],pch = 19, col = color[i-
x+1])
  }
}
}
```

Looking at CTV7000 again:

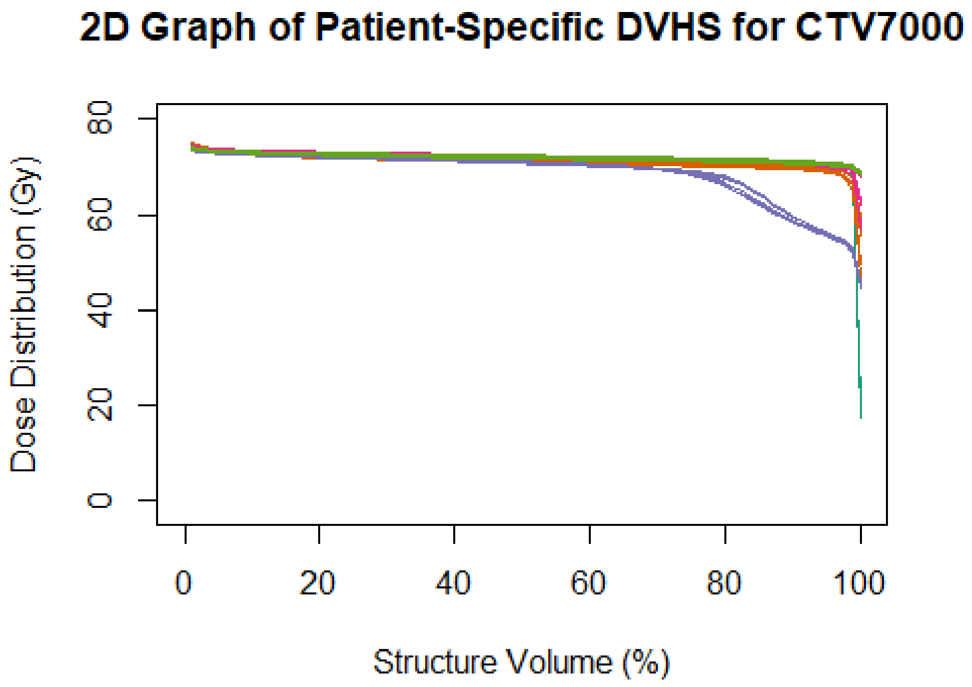
```
DVHplot.CTV7000(1,5)
```



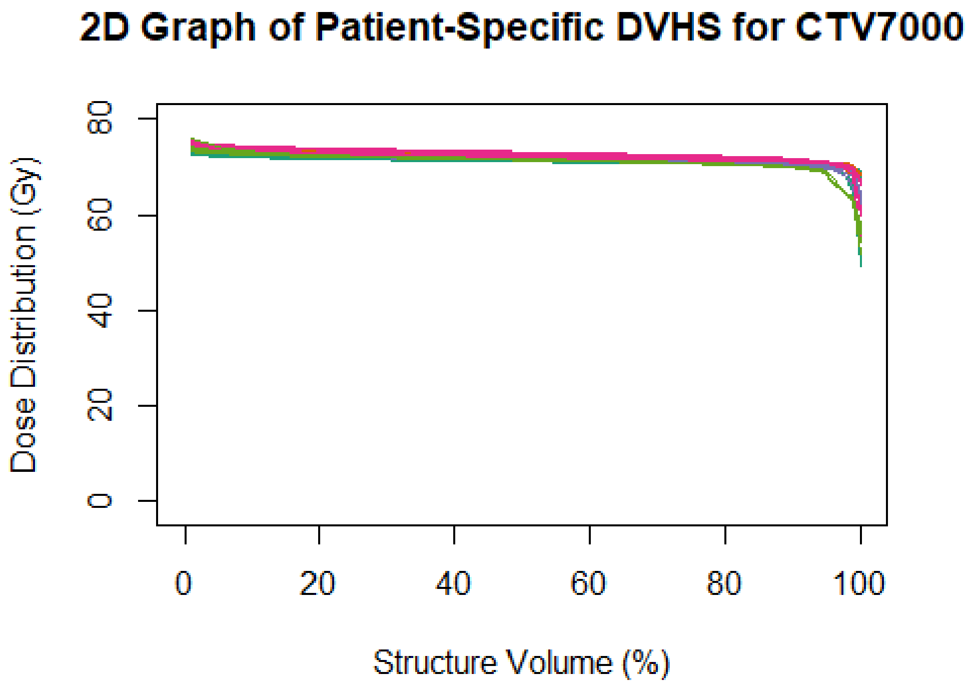
```
DVHplot.CTV7000(6,10)
```



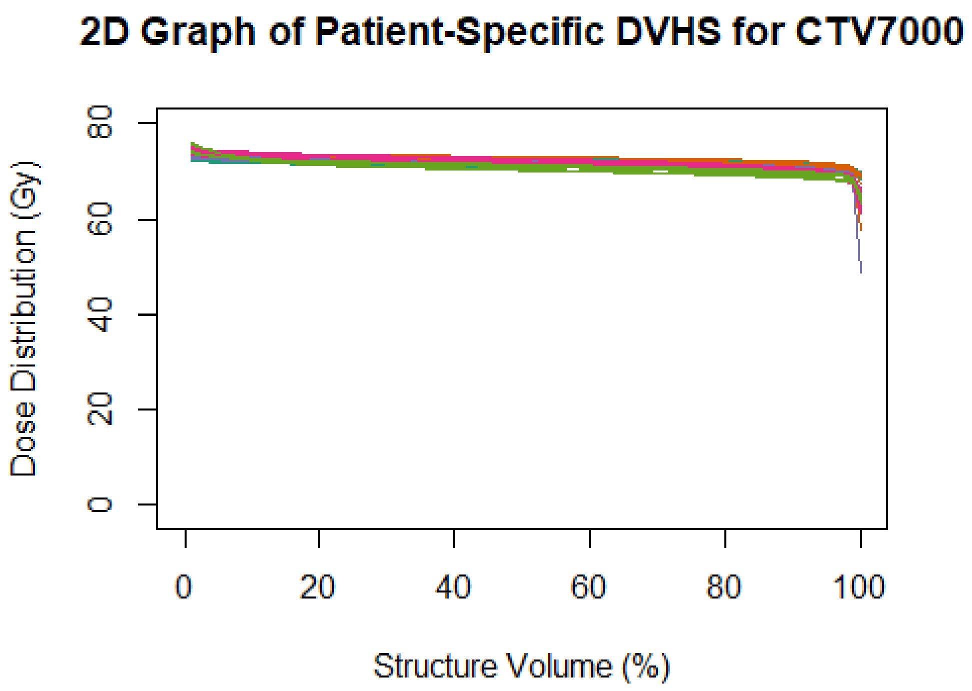
```
DVHplot.CTV7000(11,15)
```



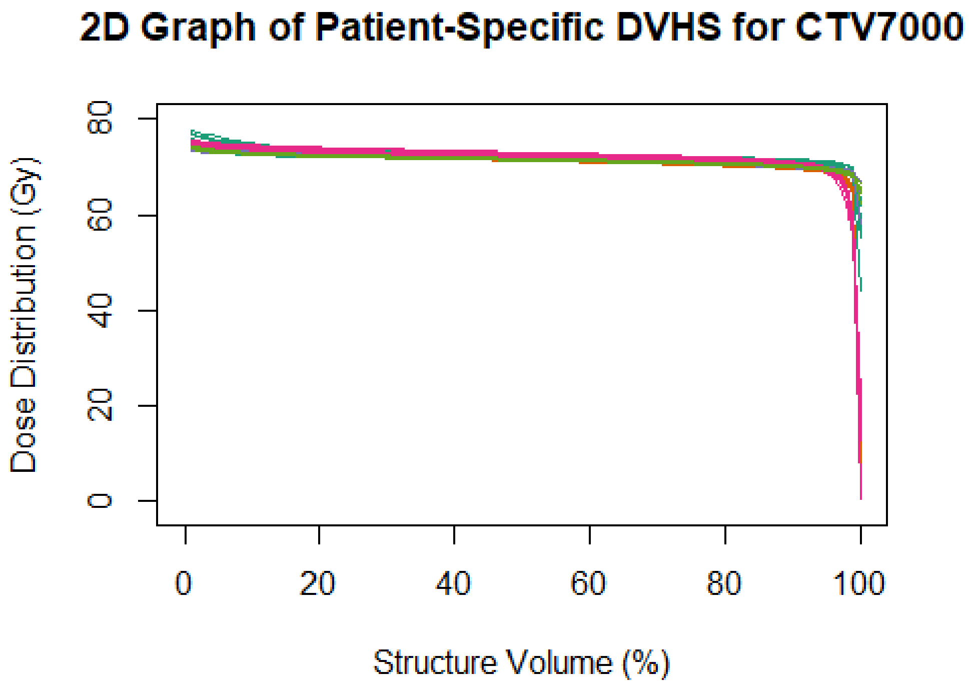
```
DVHplot.CTV7000(16,20)
```



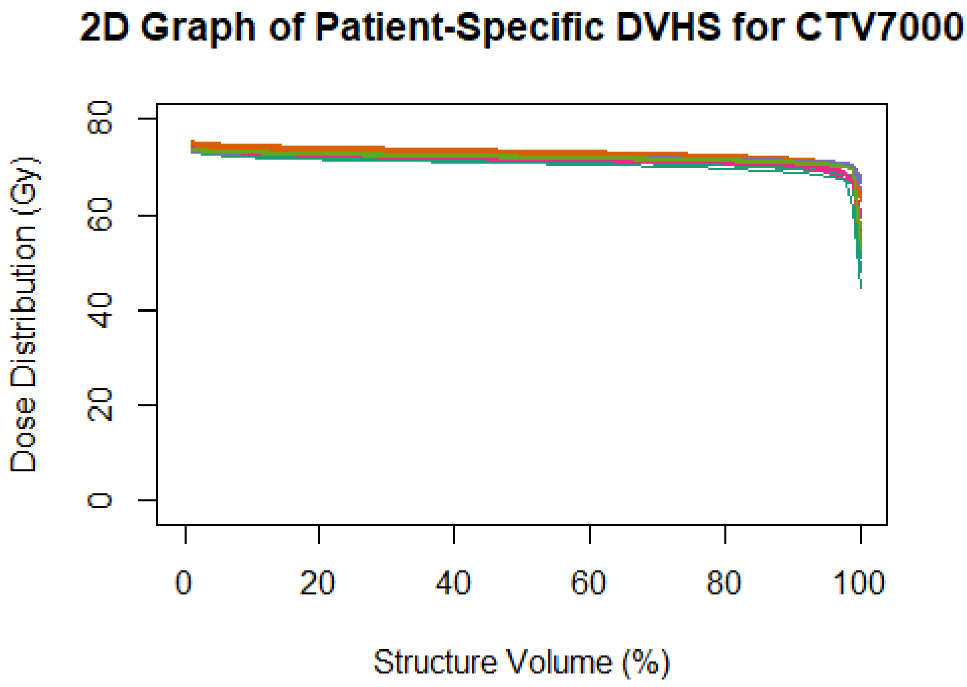
```
DVHplot.CTV7000(21,25)
```



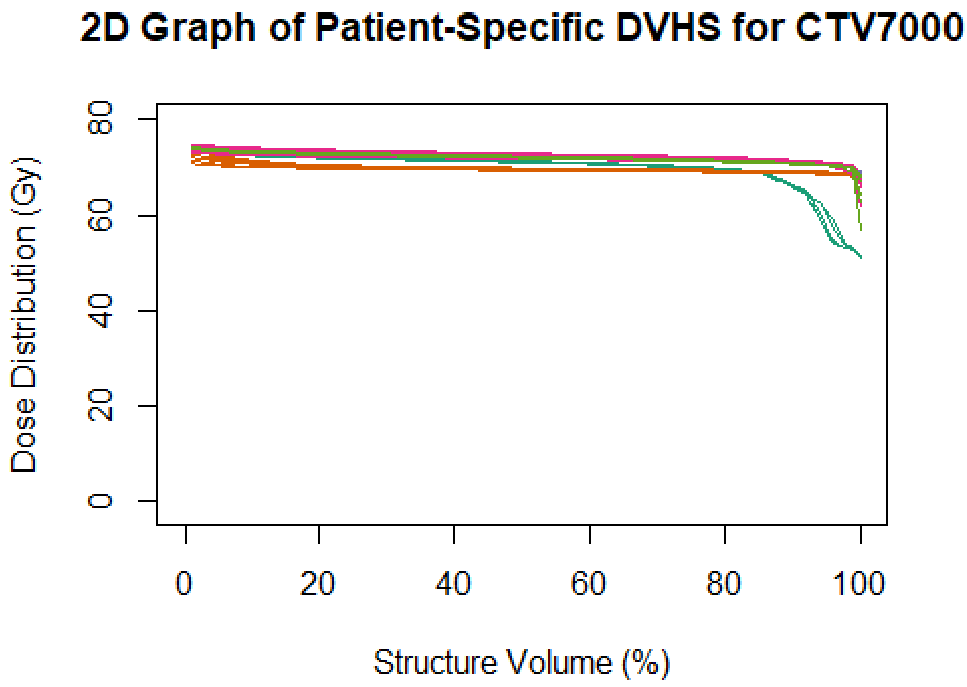
```
DVHplot.CTV7000(26,30)
```



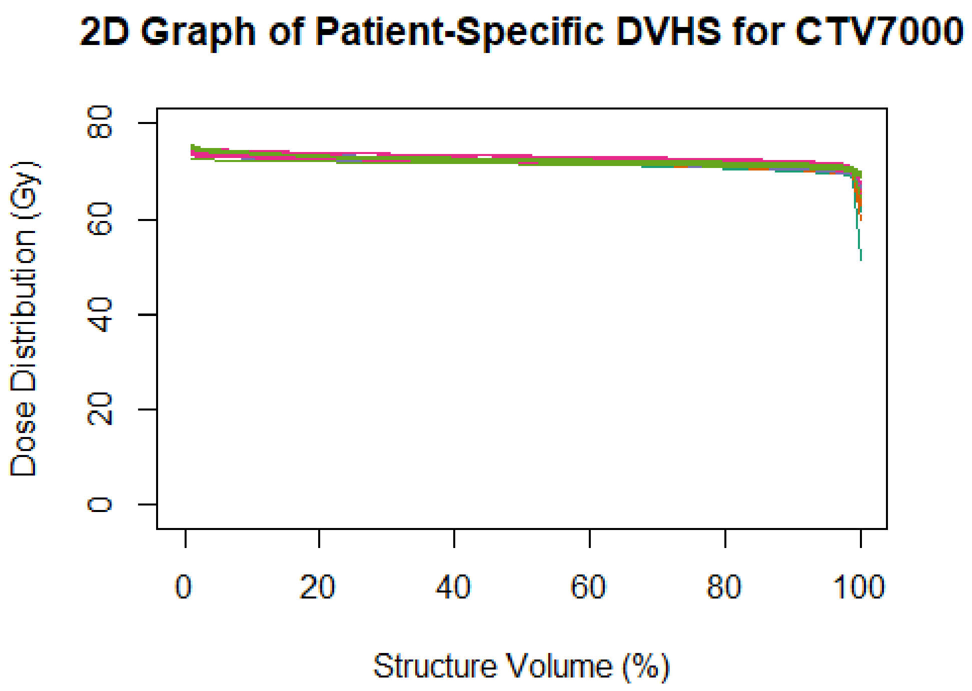
```
DVHplot.CTV7000(31,35)
```



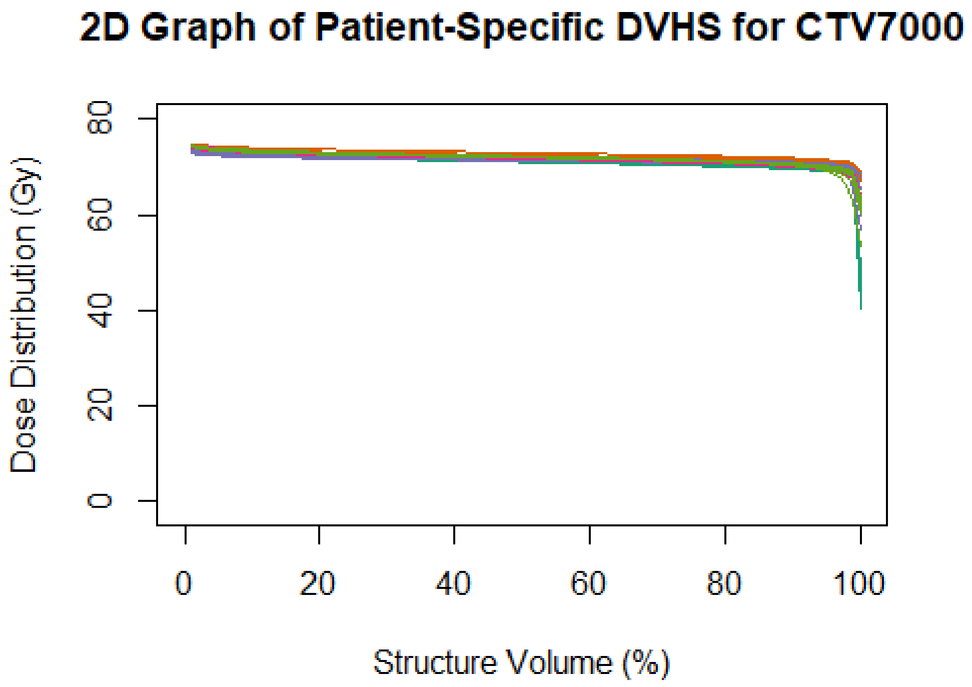
```
DVHplot.CTV7000(36,40)
```



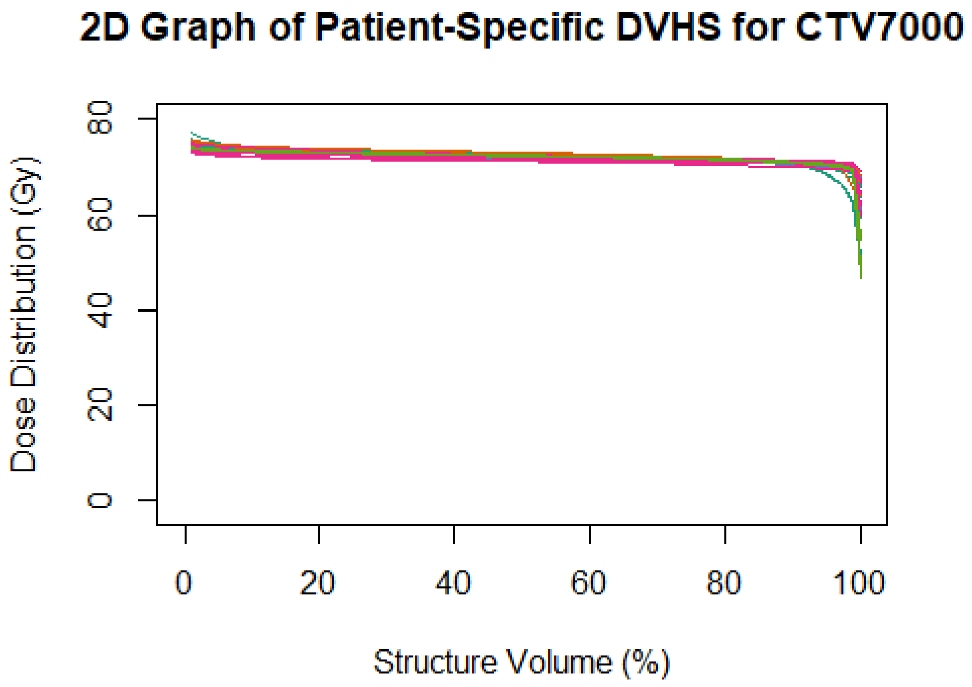
```
DVHplot.CTV7000(41,45)
```



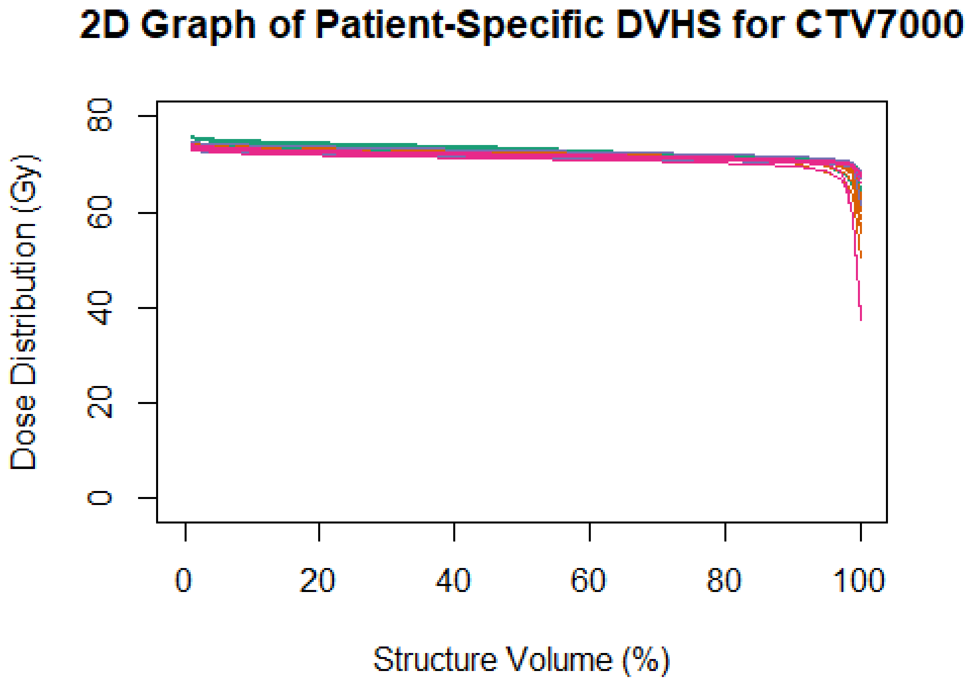
```
DVHplot.CTV7000(46,50)
```



```
DVHplot.CTV7000(51,55)
```

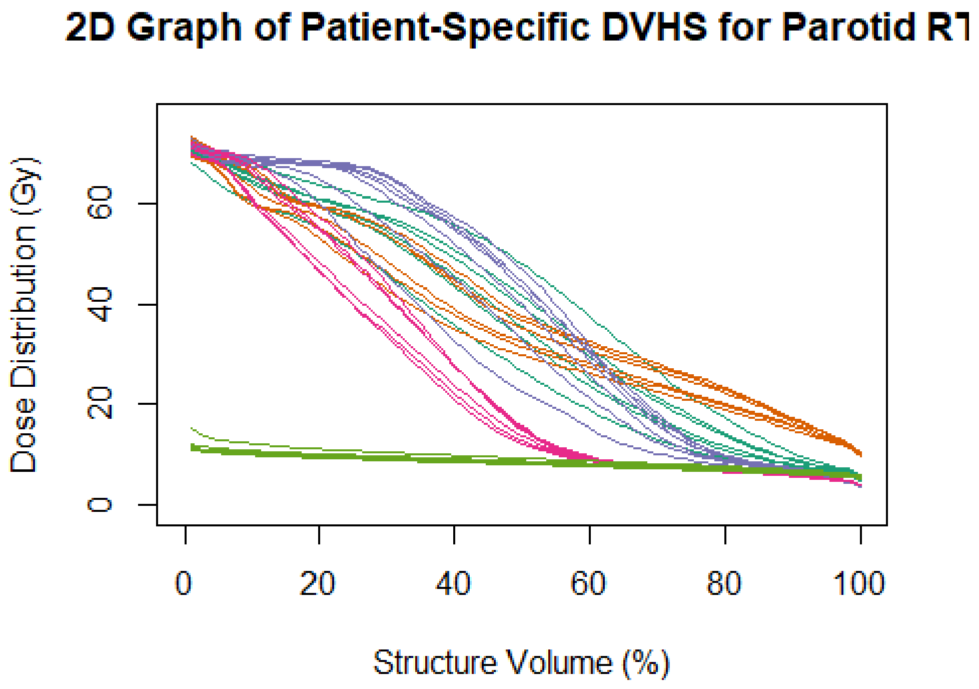


```
DVHplot.CTV7000(56,59)
```

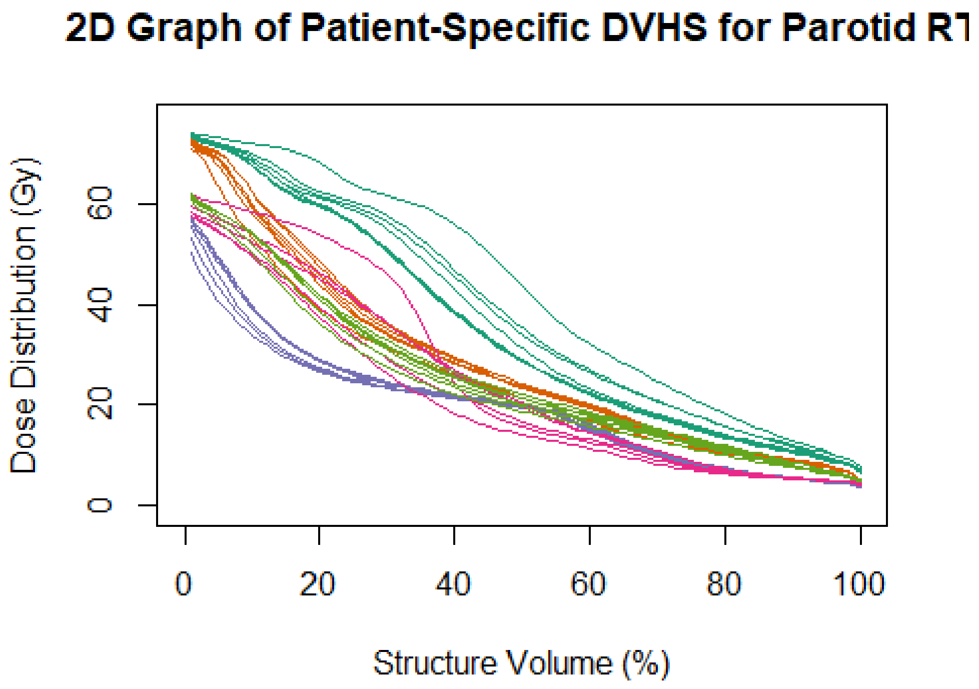


Parotid RT

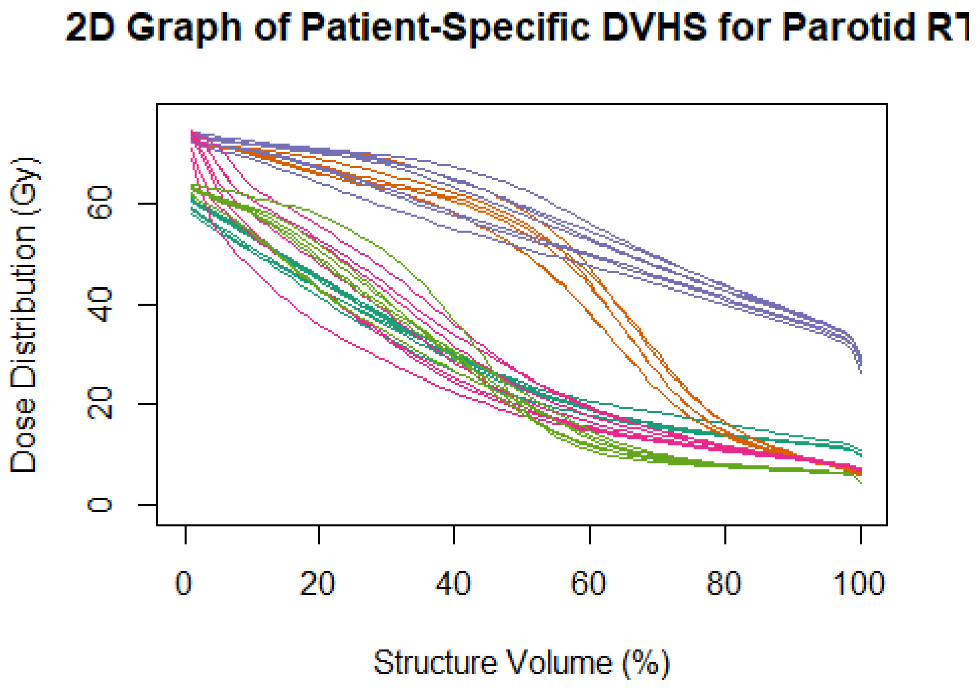
```
DVHplot.ParotidRT(1,5)
```



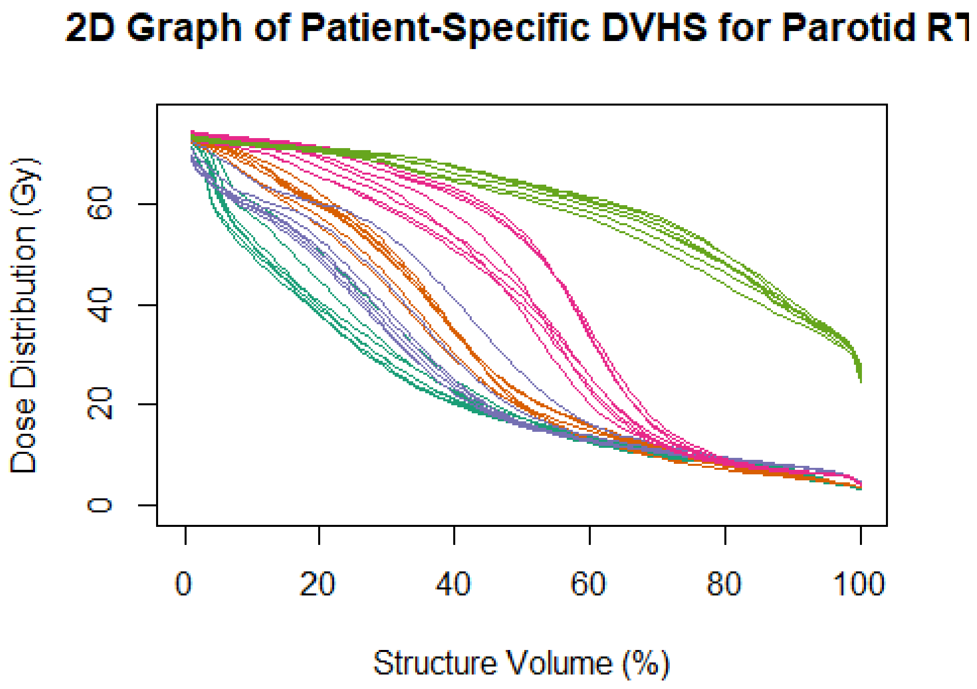
```
DVHplot.ParotidRT(6,10)
```



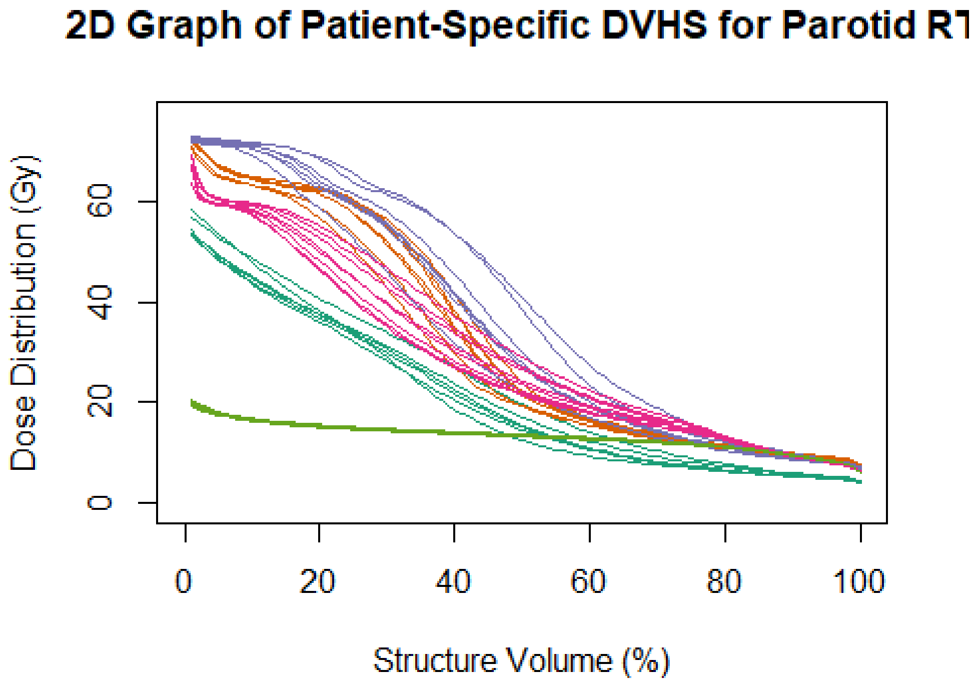
```
DVHplot.ParotidRT(11,15)
```



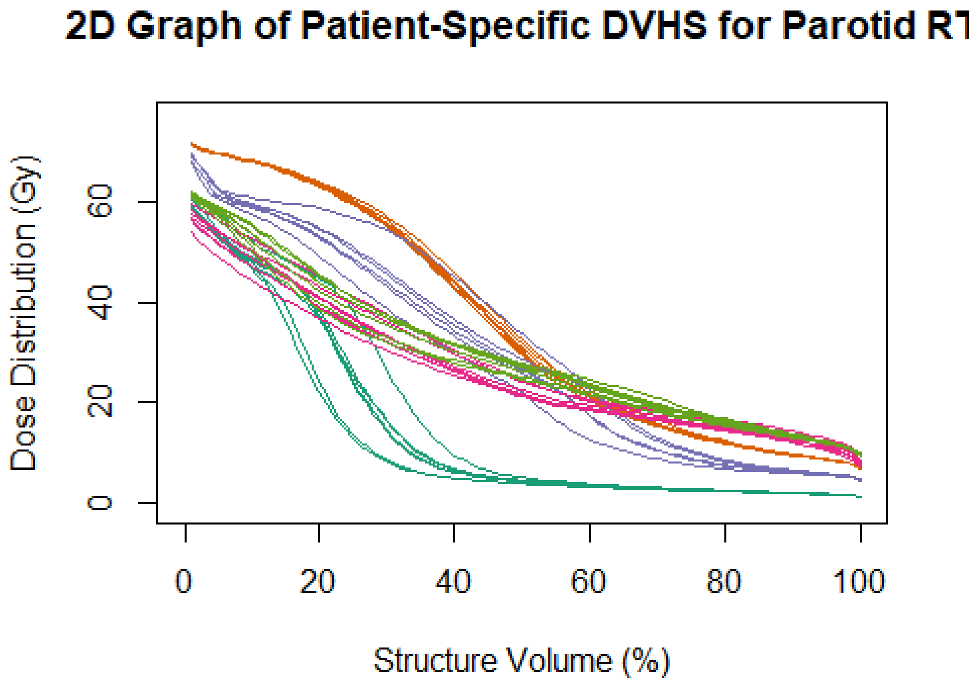
```
DVHplot.ParotidRT(16,20)
```



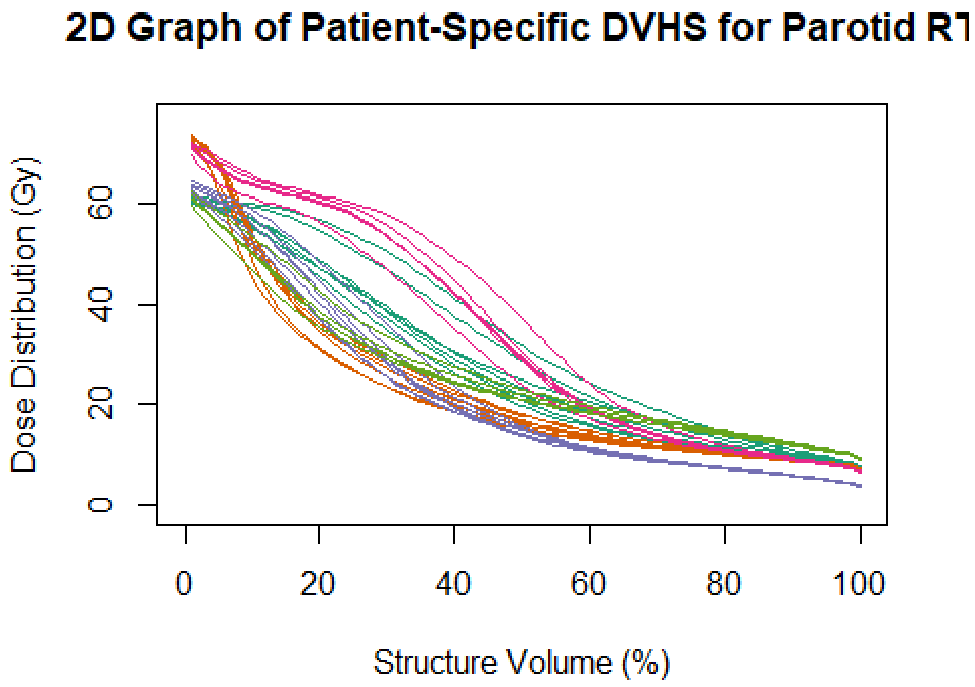
```
DVHplot.ParotidRT(21,25)
```



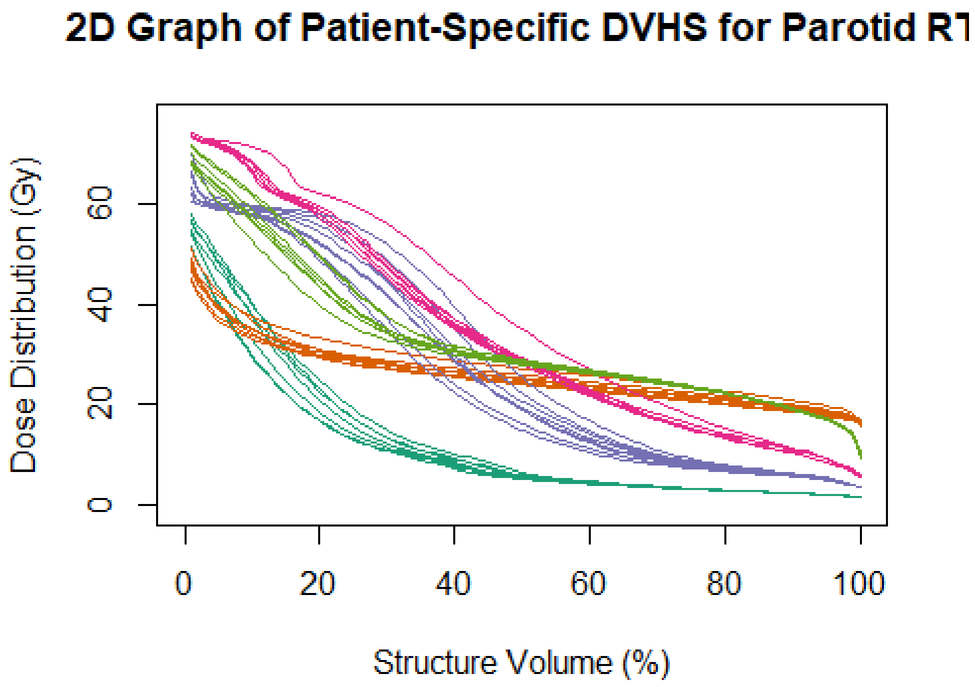
```
DVHplot.ParotidRT(26,30)
```



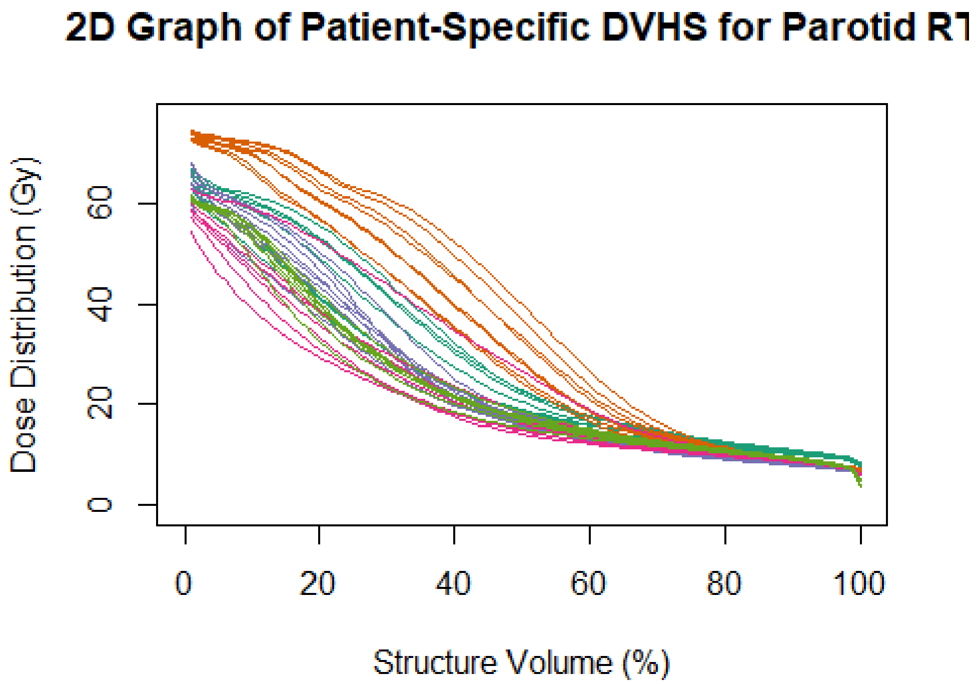
```r
DVHplot.ParotidRT(31,35)
```



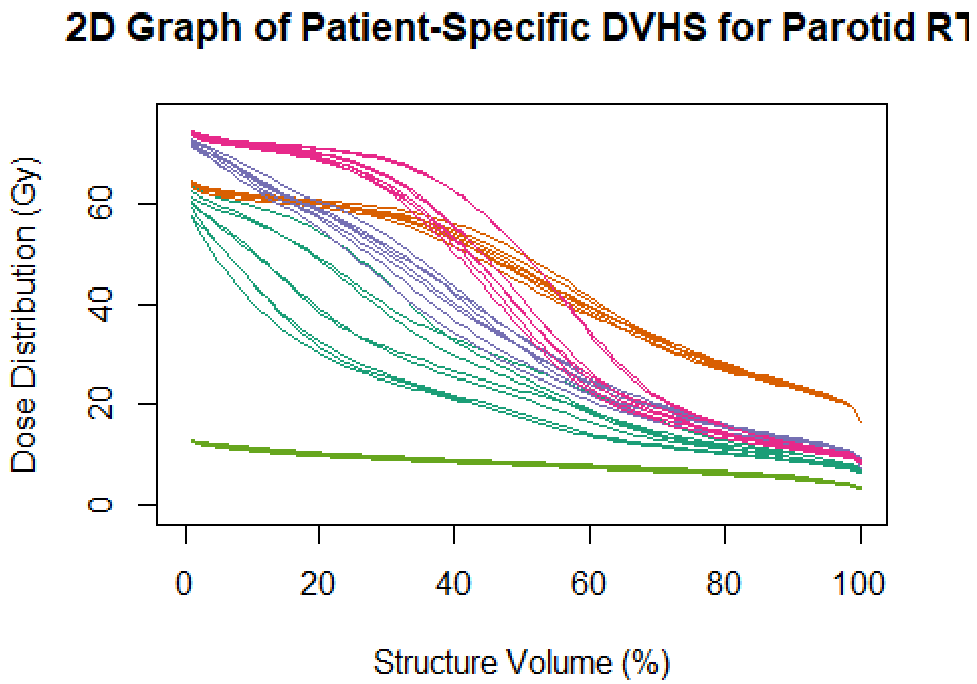
```r
DVHplot.ParotidRT(36,40)
```



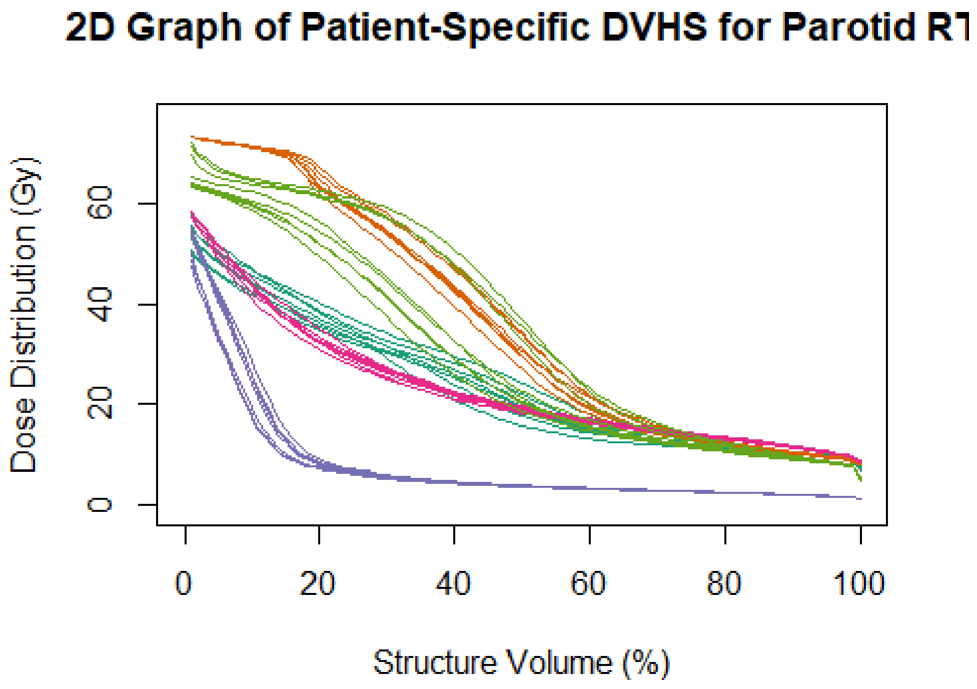
```
DVHplot.ParotidRT(41,45)
```



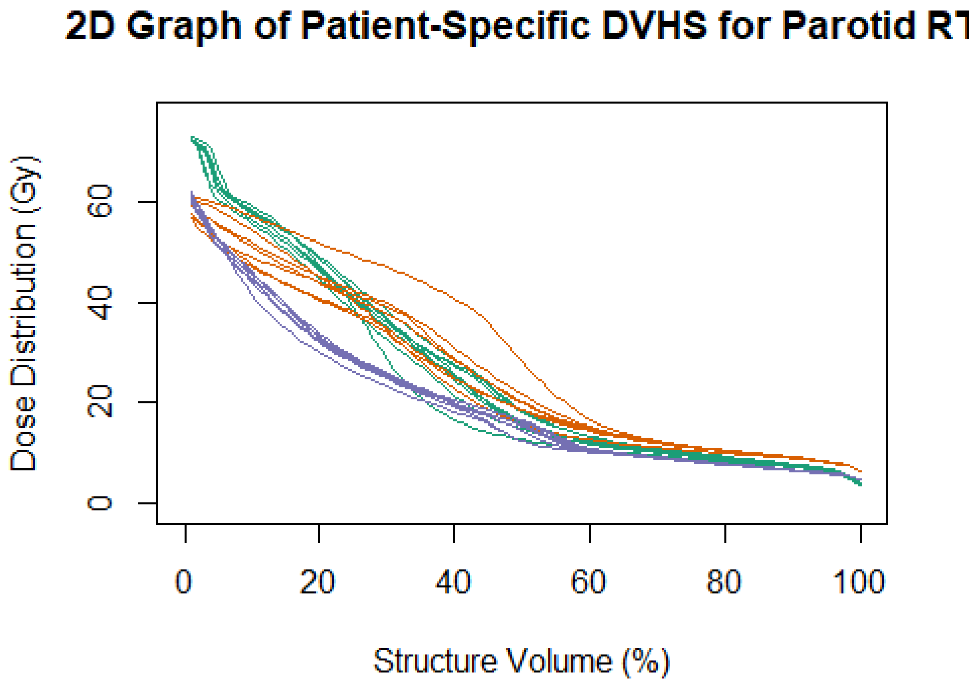
```
DVHplot.ParotidRT(46,50)
```



```r
DVHplot.ParotidRT(51,55)
```

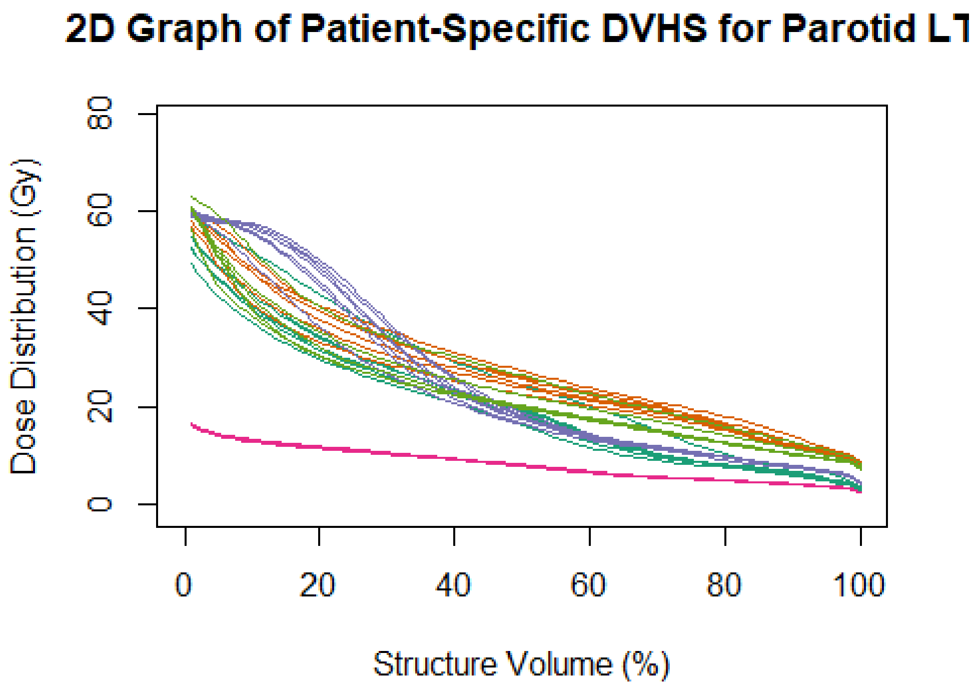


```r
DVHplot.ParotidRT(56,58)
```

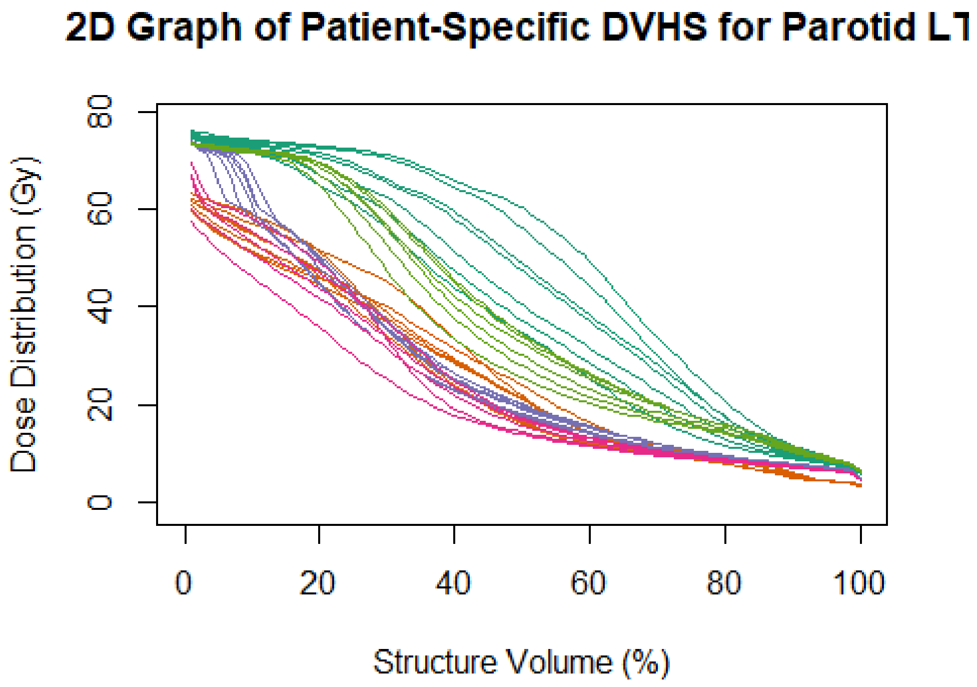


Parotid LT

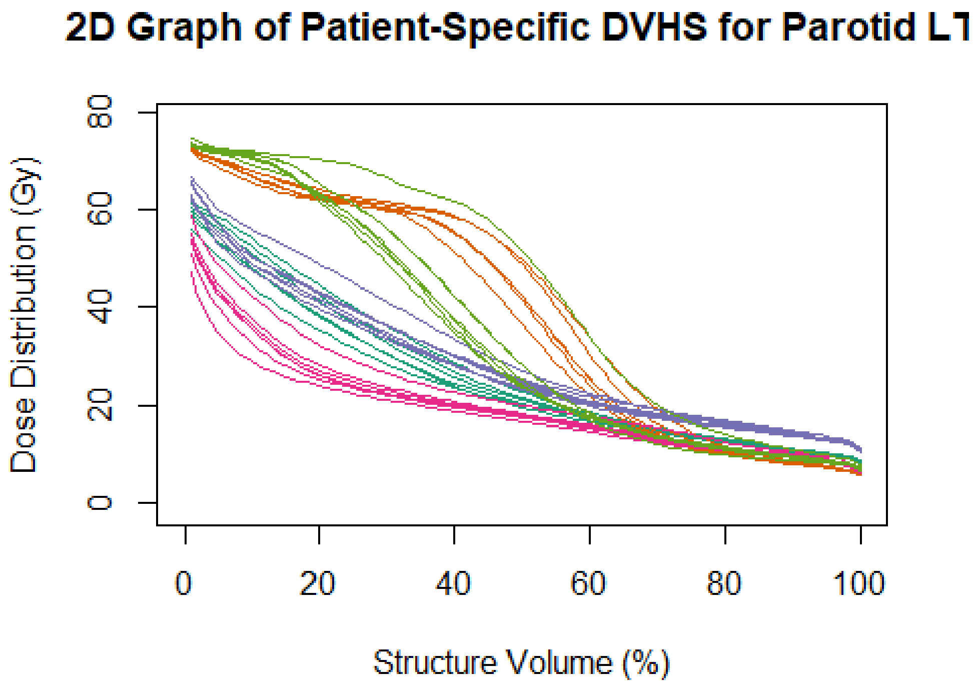
```
DVHplot.ParotidLT(1,5)
```



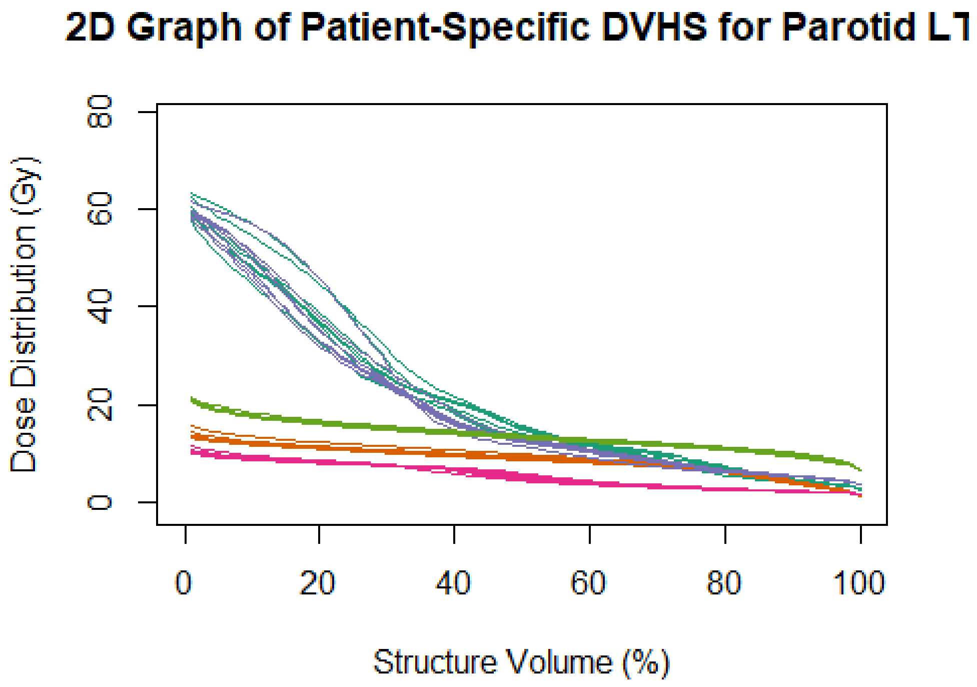
```
DVHplot.ParotidLT(6,10)
```



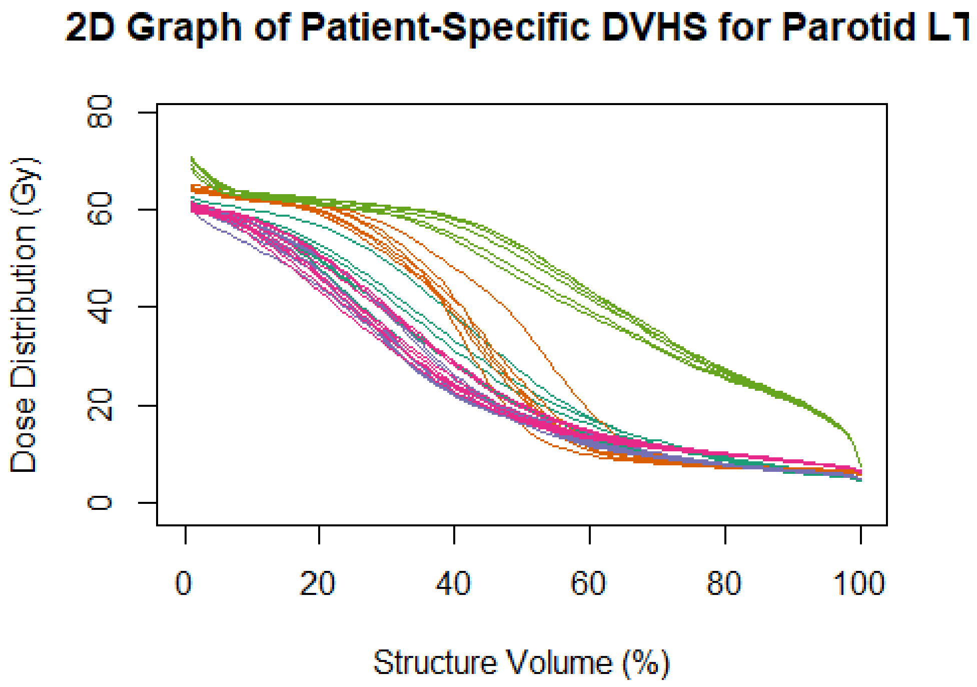
```
DVHplot.ParotidLT(11,15)
```



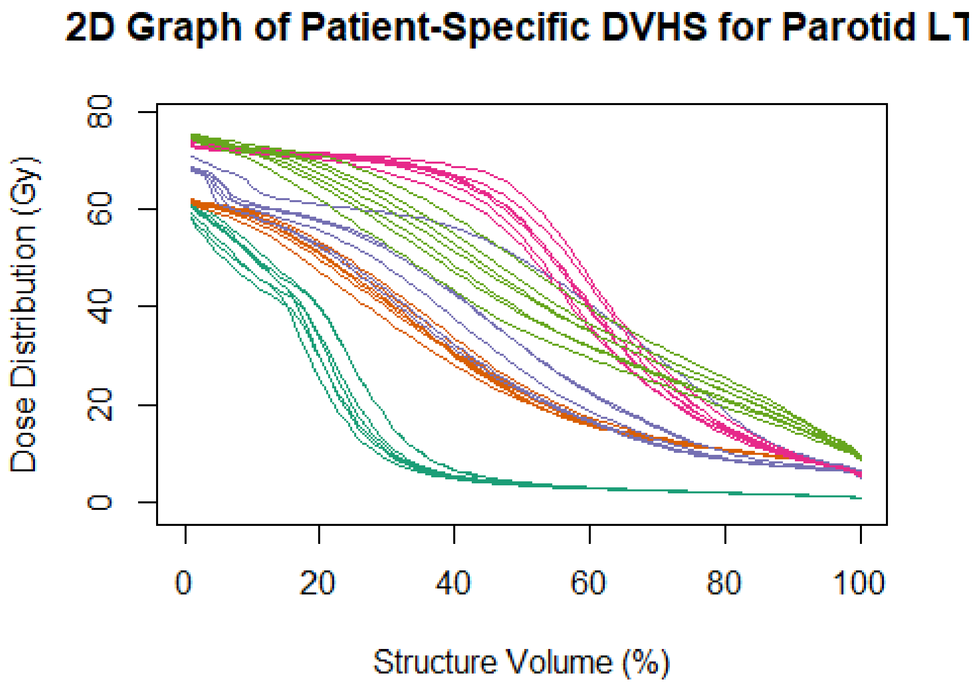
```
DVHplot.ParotidLT(16,20)
```



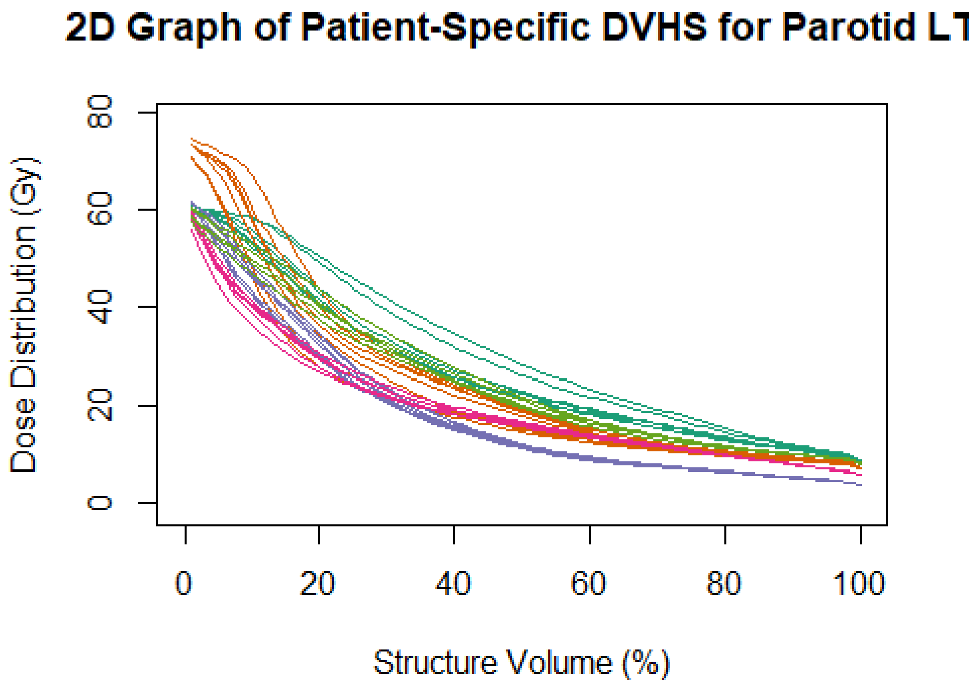
```
DVHplot.ParotidLT(21,25)
```



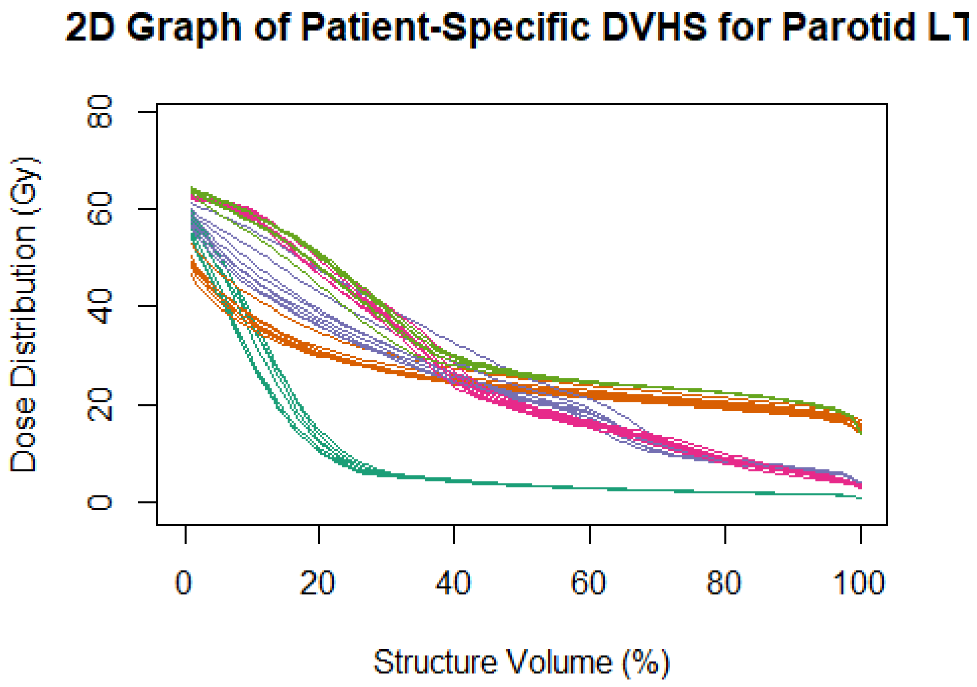
```
DVHplot.ParotidLT(26,30)
```



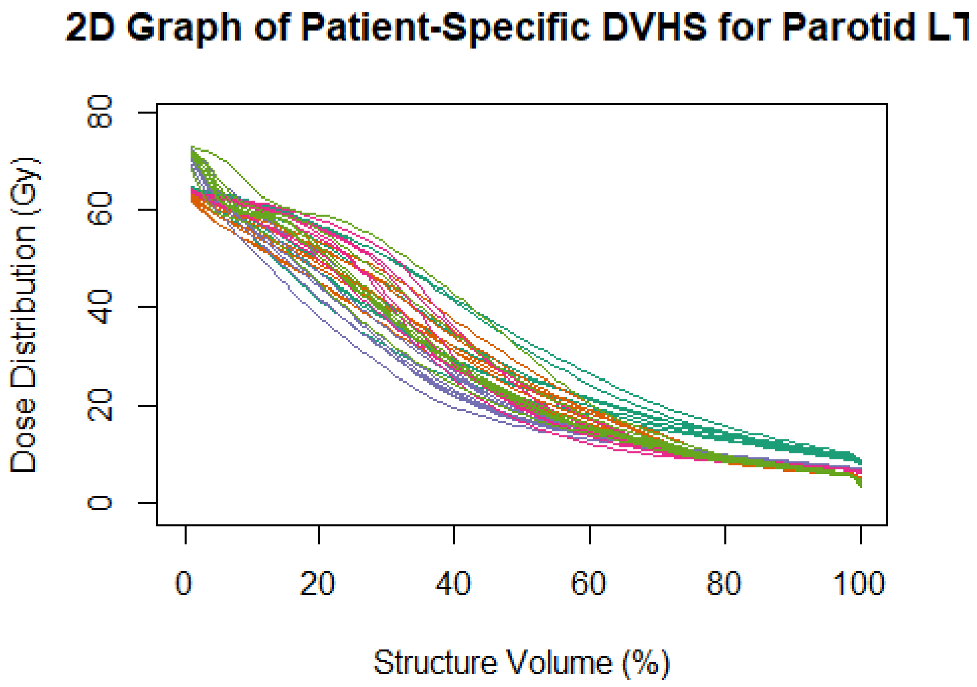
```
DVHplot.ParotidLT(31,35)
```



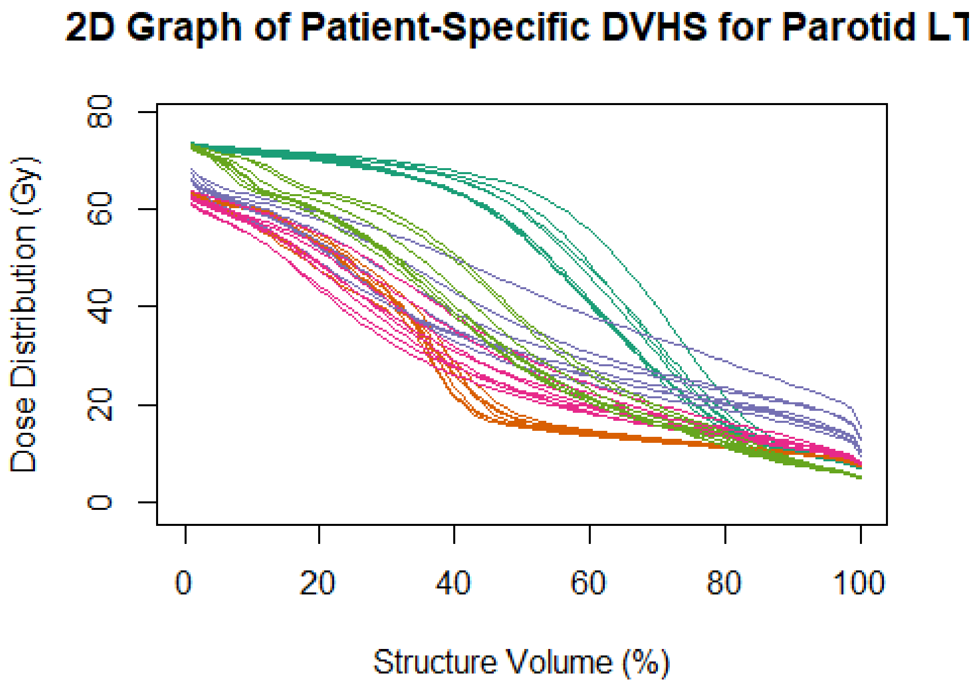
```
DVHplot.ParotidLT(36,40)
```



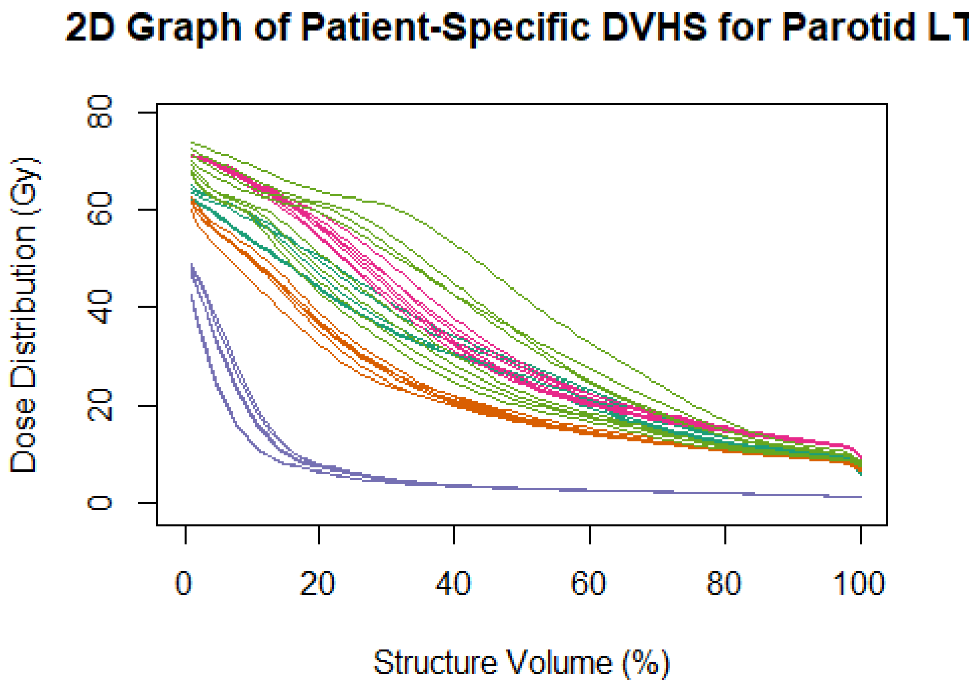
```
DVHplot.ParotidLT(41,45)
```



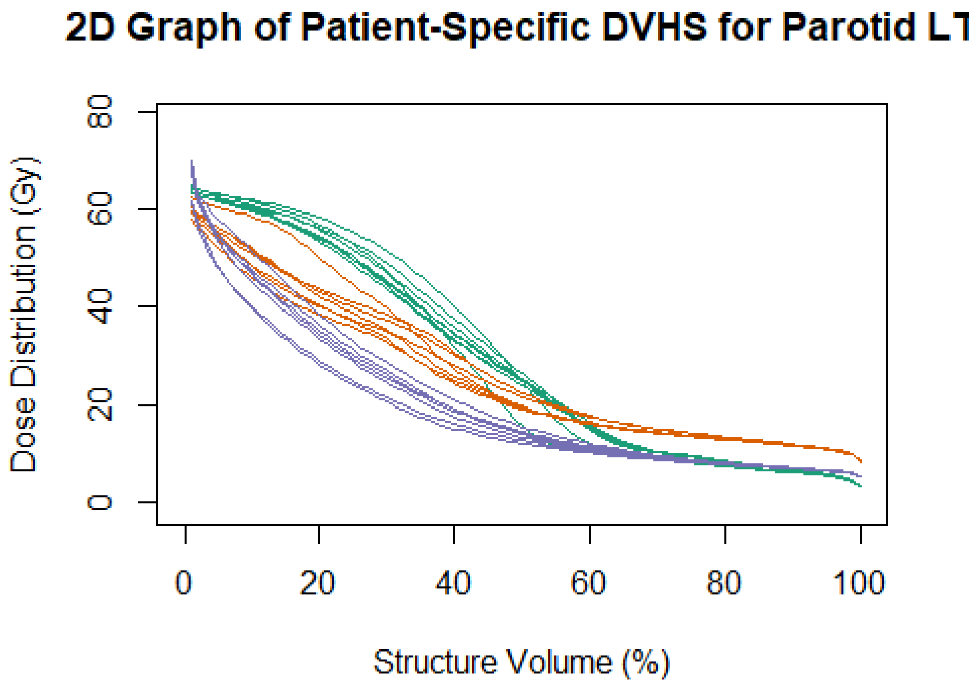
```
DVHplot.ParotidLT(46,50)
```



```
DVHplot.ParotidLT(51,55)
```



```
DVHplot.ParotidLT(56,58)
```



Plotting Lag 1 Difference

- to visualize whether there is a systematic trend

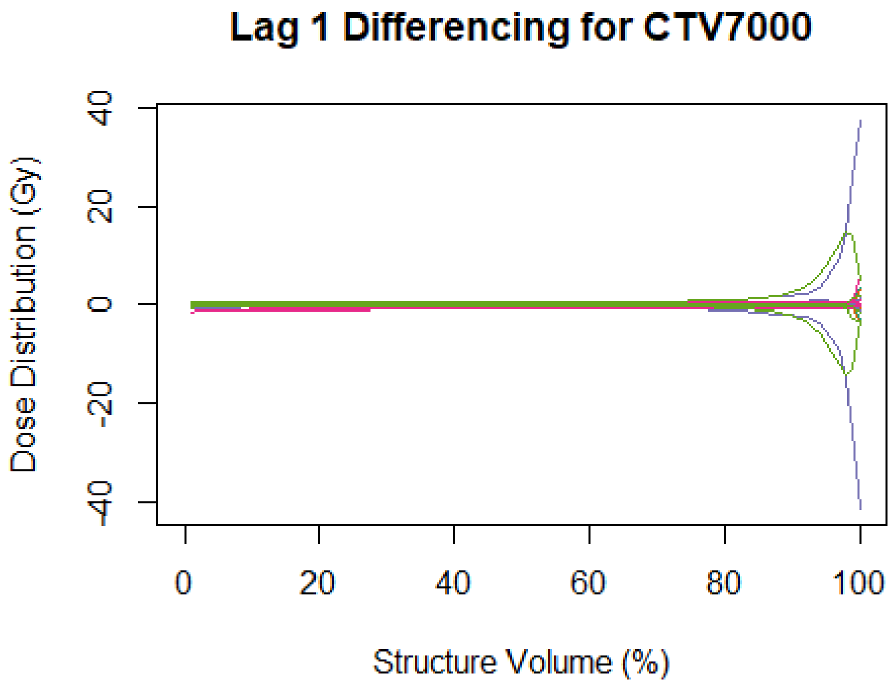
```
lagParotidRT <- function(x,y) {

maxlist <- c(0)
minlist <- c(0)
for (i in x:y) #first 5 patients
{
  for (j in 0:(ParotidRT_ptrows[i+1]-ParotidRT_ptrows[i] - 2) ) #time index
  {
    ind1 = ParotidRT_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.ParotidRT[ind1,c(4:103)]
    t2 = DVH.ParotidRT[ind2,c(4:103)]
    lagdiff = t1-t2
    maxlist <- c(maxlist,max(lagdiff))
    minlist <- c(minlist,min(lagdiff))
  }

}

plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",main =
"Lag 1 Differencing for Parotid RT",xlim = c(0,100), ylim =
c(min(minlist),max(maxlist)))
for (i in x:y)
```

```
{
  for (j in 0:(ParotidRT_ptrows[i+1]-ParotidRT_ptrows[i] - 2) ) #time index
  {
    ind1 = ParotidRT_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.ParotidRT[ind1,c(4:103)]
    t2 = DVH.ParotidRT[ind2,c(4:103)]
    lagdiff = t1-t2
    lines(seq(1,100,1),lagdiff, col = color[i-x+1])

      }
}

}

lagParotidLT <- function(x,y) {

maxlist <- c(0)
minlist <- c(0)
for (i in x:y) #first 5 patients
{
  for (j in 0:(ParotidLT_ptrows[i+1]-ParotidLT_ptrows[i] - 2) ) #time index
  {
    ind1 = ParotidLT_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.ParotidLT[ind1,c(4:103)]
    t2 = DVH.ParotidLT[ind2,c(4:103)]
    lagdiff = t1-t2
    maxlist <- c(maxlist,max(lagdiff))
    minlist <- c(minlist,min(lagdiff))
  }

}
#print(max(maxlist))
plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",main =
"Lag 1 Differencing for Parotid LT",xlim = c(0,100), ylim =
c(min(minlist),max(maxlist)))#,asp = 2)
for (i in x:y) #first 5 patients
{
  for (j in 0:(ParotidLT_ptrows[i+1]-ParotidLT_ptrows[i] - 2) ) #time index
  {
    ind1 = ParotidLT_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.ParotidLT[ind1,c(4:103)]
    t2 = DVH.ParotidLT[ind2,c(4:103)]
    lagdiff = t1-t2
    lines(seq(1,100,1),lagdiff, col = color[i-x+1])

      }
```

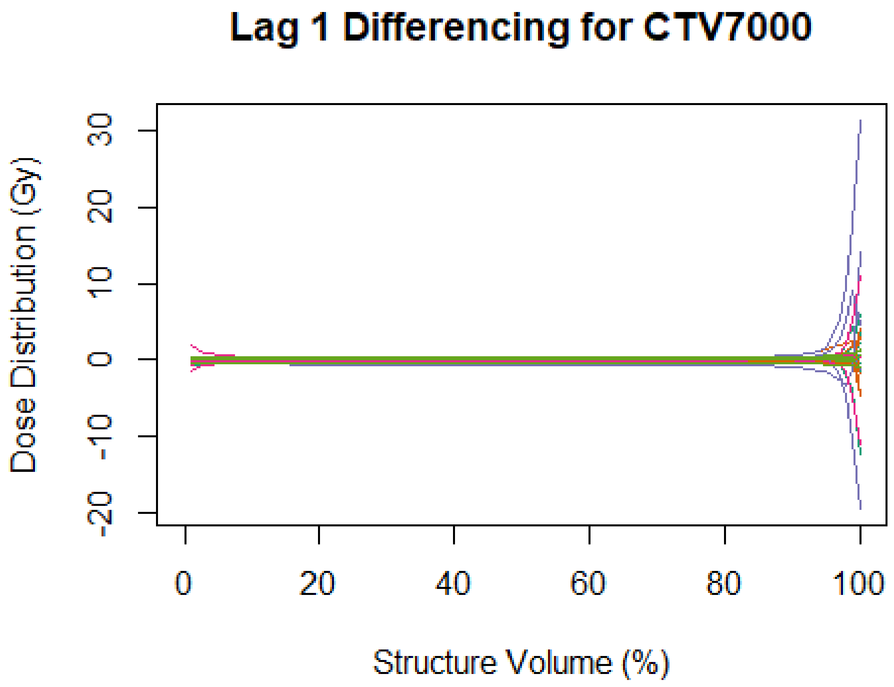
```
}


}

lagCTV7000 <- function(x,y) {

maxlist <- c(0)
minlist <- c(0)
for (i in x:y) #first 5 patients
{
  for (j in 0:(CTV7000_ptrows[i+1]-CTV7000_ptrows[i] - 2) ) #time index
  {
    ind1 = CTV7000_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.CTV7000[ind1,c(4:103)]
    t2 = DVH.CTV7000[ind2,c(4:103)]
    lagdiff = t1-t2
    maxlist <- c(maxlist,max(lagdiff))
    minlist <- c(minlist,min(lagdiff))
  }

}
#print(max(maxlist))
plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",main =
"Lag 1 Differencing for CTV7000",xlim = c(0,100), ylim =
c(min(minlist),max(maxlist)))#,asp = 2)
for (i in x:y) #first 5 patients
{
  for (j in 0:(CTV7000_ptrows[i+1]-CTV7000_ptrows[i] - 2) ) #time index
  {
    ind1 = CTV7000_ptrows[i] + j
    ind2 = ind1 + 1
    t1 = DVH.CTV7000[ind1,c(4:103)]
    t2 = DVH.CTV7000[ind2,c(4:103)]
    lagdiff = t1-t2
    lines(seq(1,100,1),lagdiff, col = color[i-x+1])

      }
}


}
```

Lag 1 for CTV7000

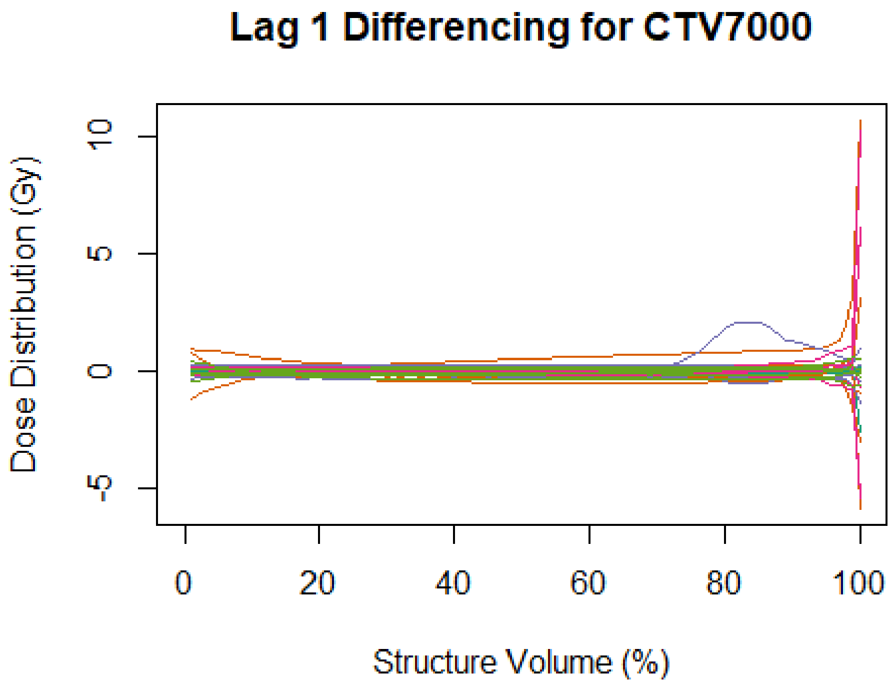
```
lagCTV7000(1,5)
```



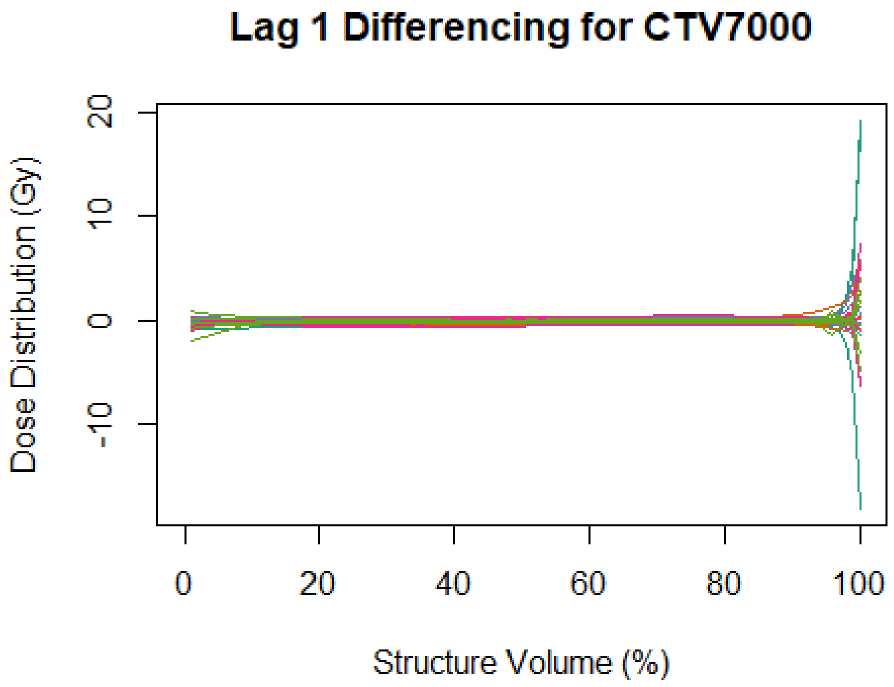
```
lagCTV7000(6,10)
```



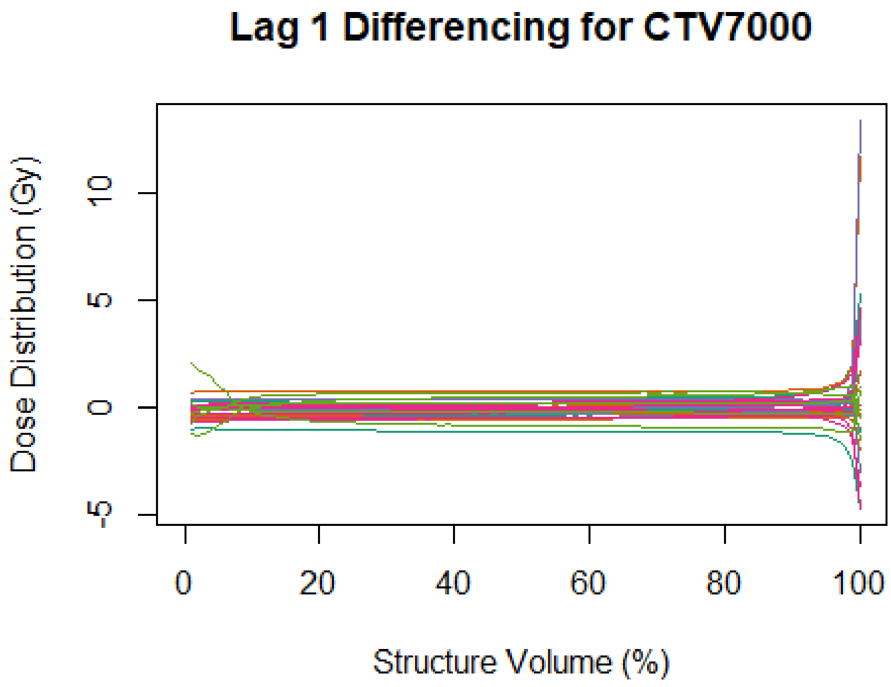
```
lagCTV7000(11,15)
```



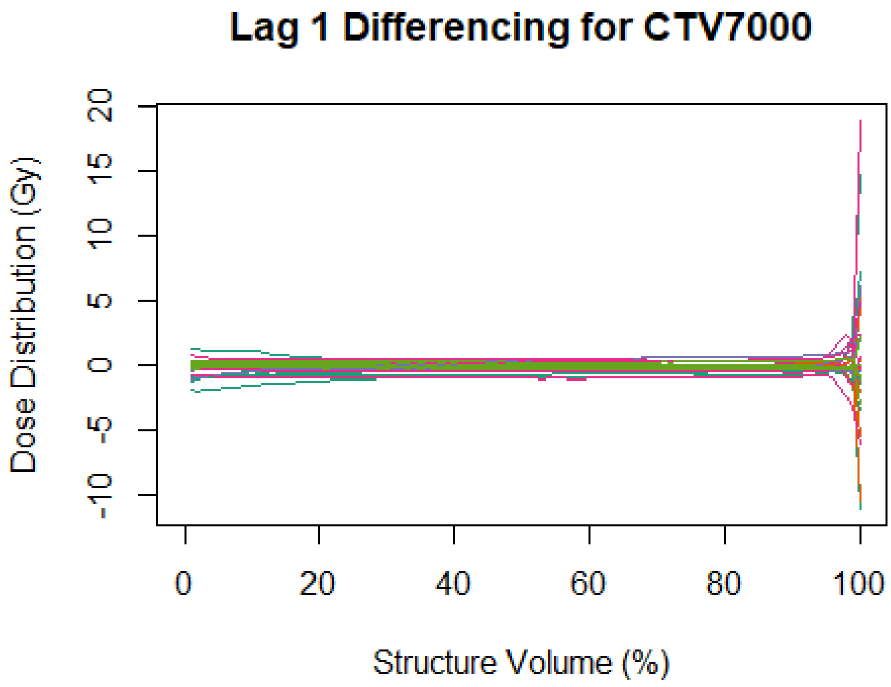
```
lagCTV7000(16,20)
```



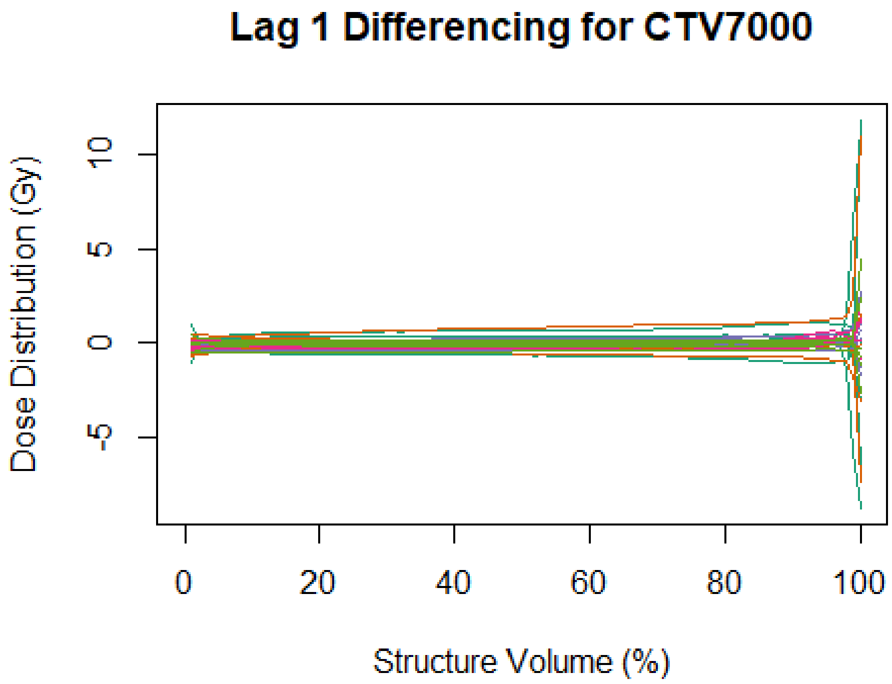
```
lagCTV7000(21,25)
```



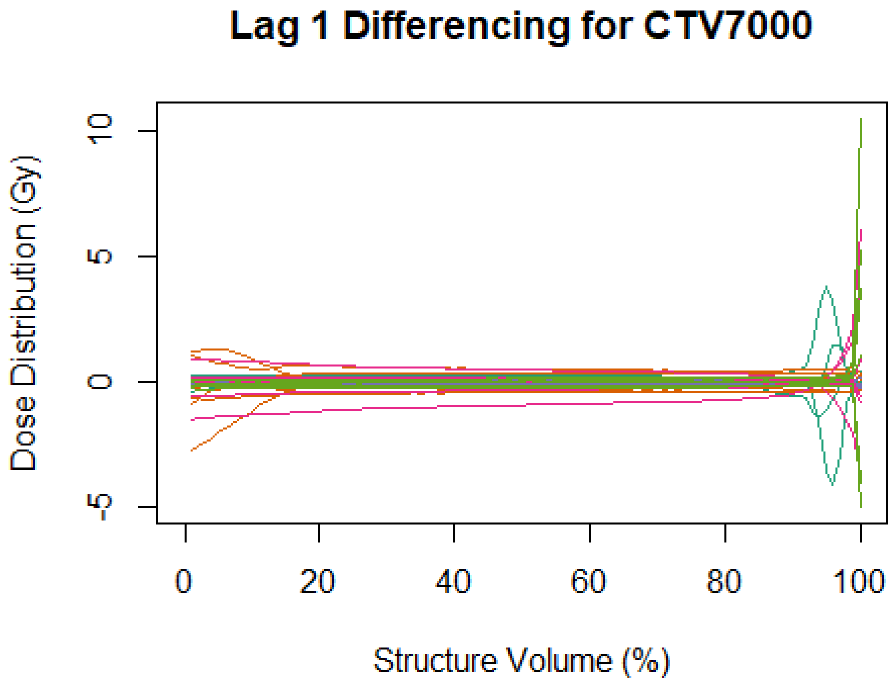
```
lagCTV7000(26,30)
```



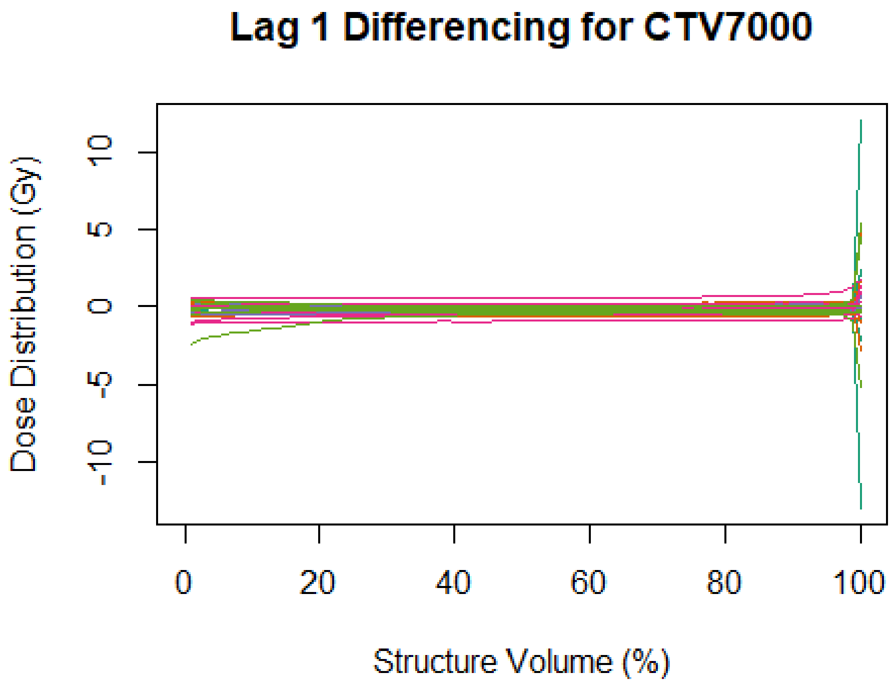
```
lagCTV7000(31,35)
```



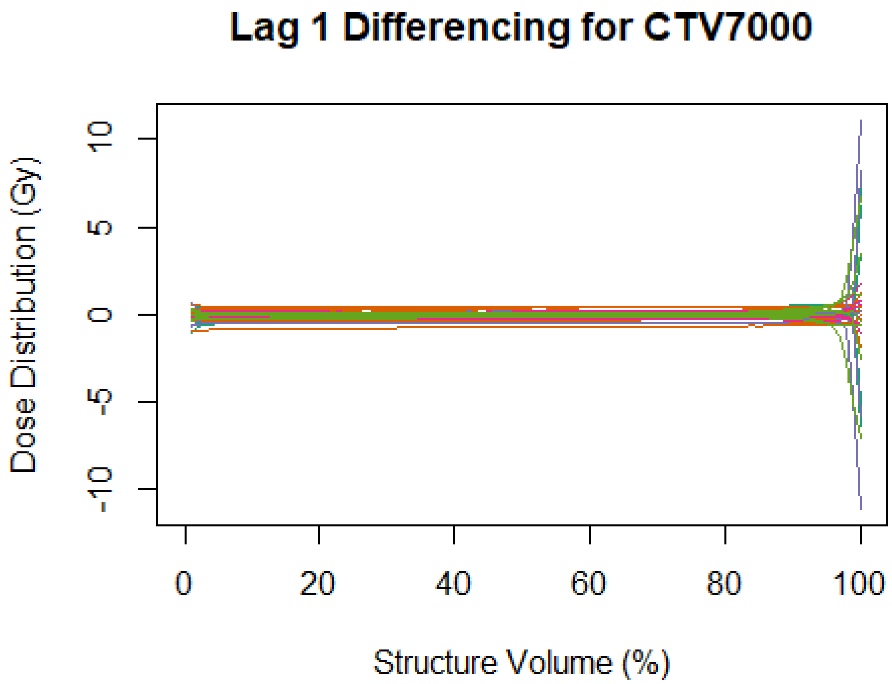
```
lagCTV7000(36,40)
```



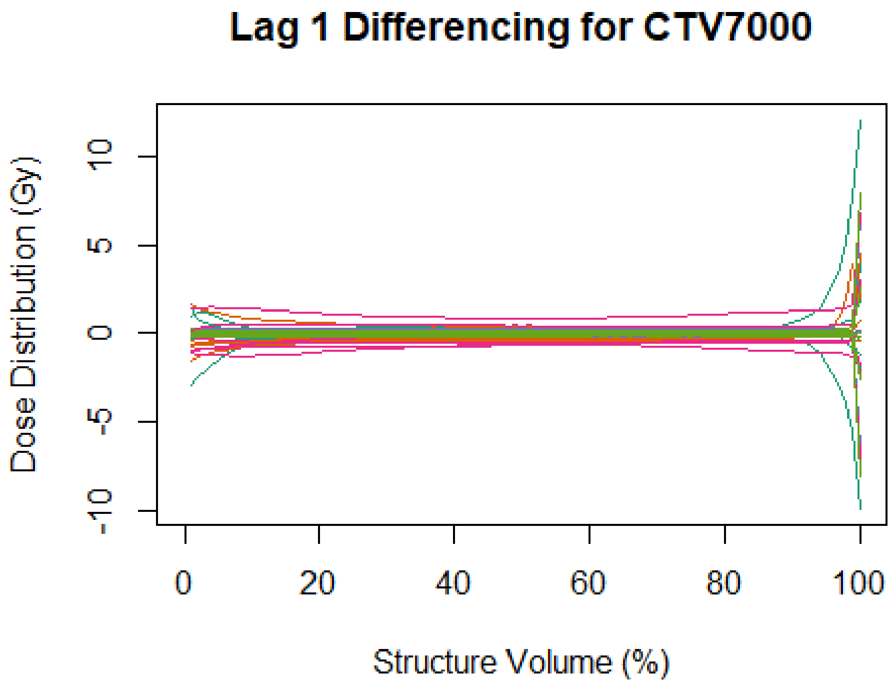
```
lagCTV7000(41,45)
```



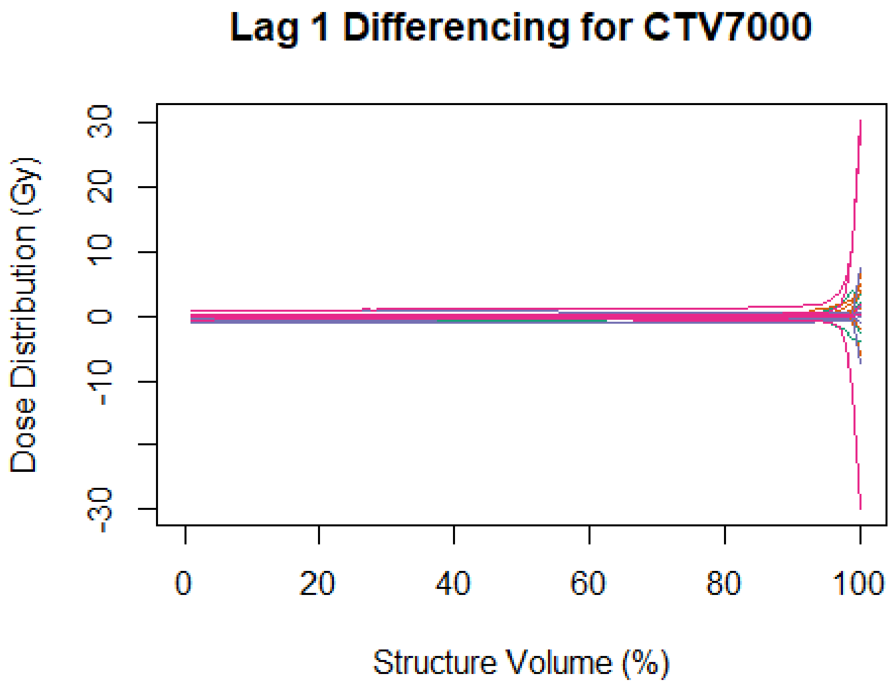
```r
lagCTV7000(46,50)
```



```r
lagCTV7000(51,55)
```

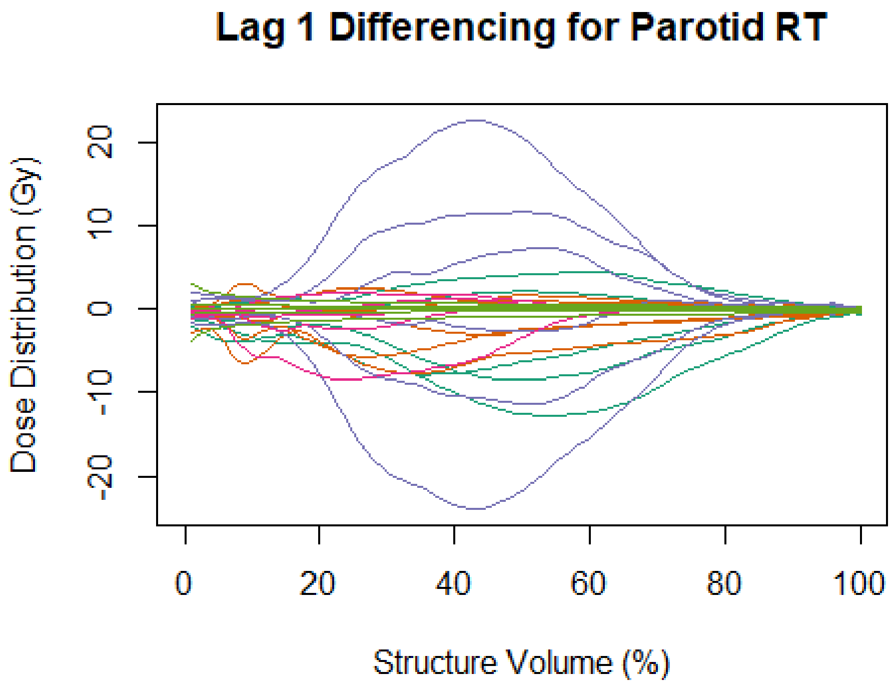


```
lagCTV7000(56,59)
```



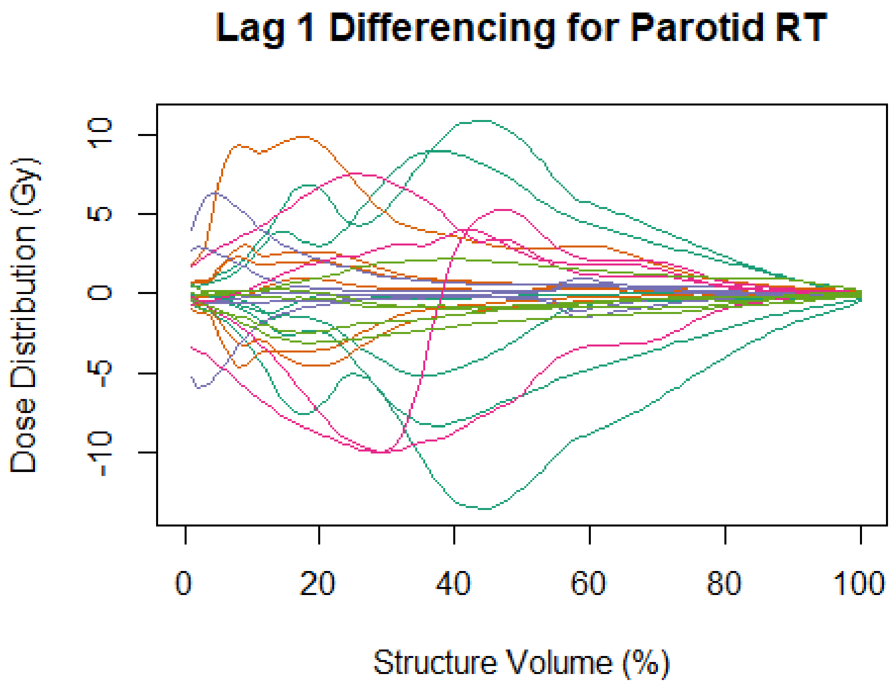
Lag 1 for Parotid RT

```
lagParotidRT(1,5)
```



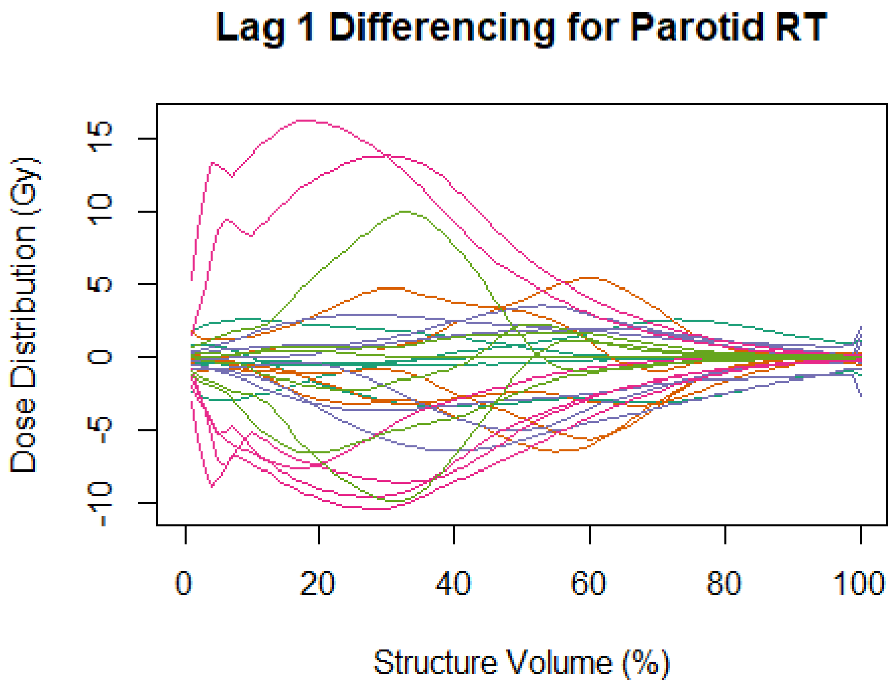
Lag 1 Differencing for Parotid RT

```
lagParotidRT(6,10)
```

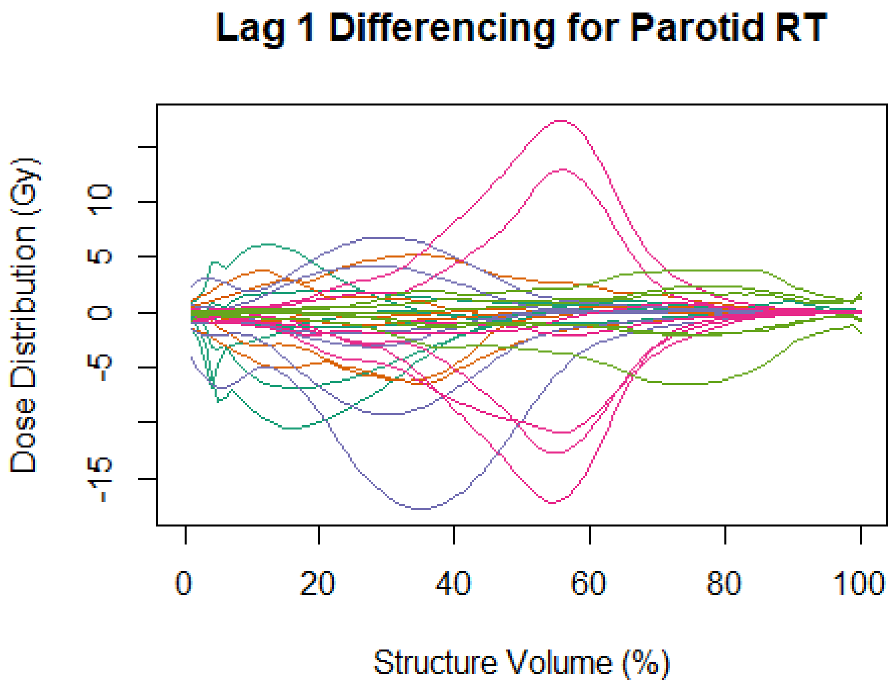


Lag 1 Differencing for Parotid RT

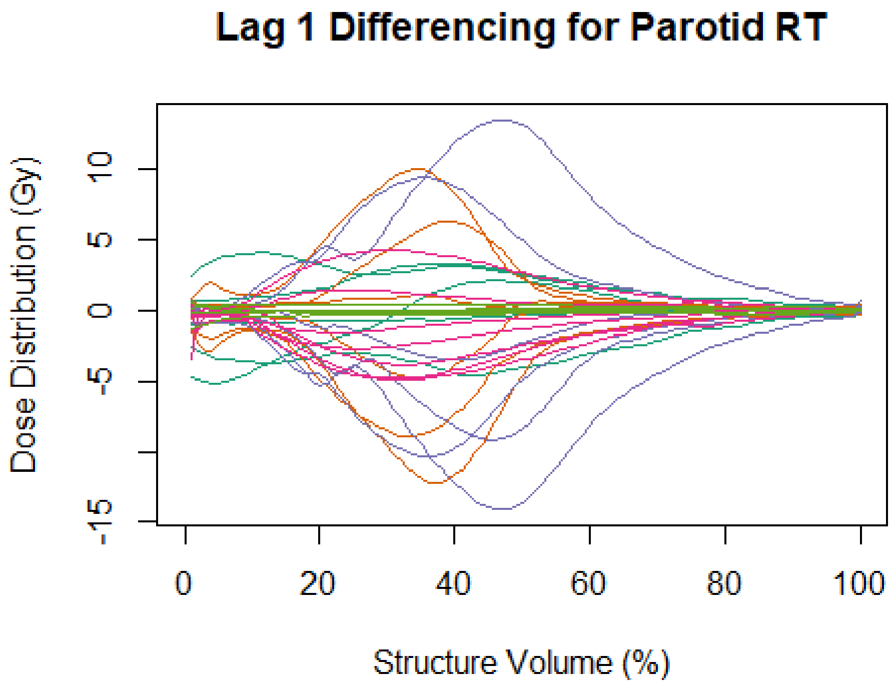
```
lagParotidRT(11,15)
```



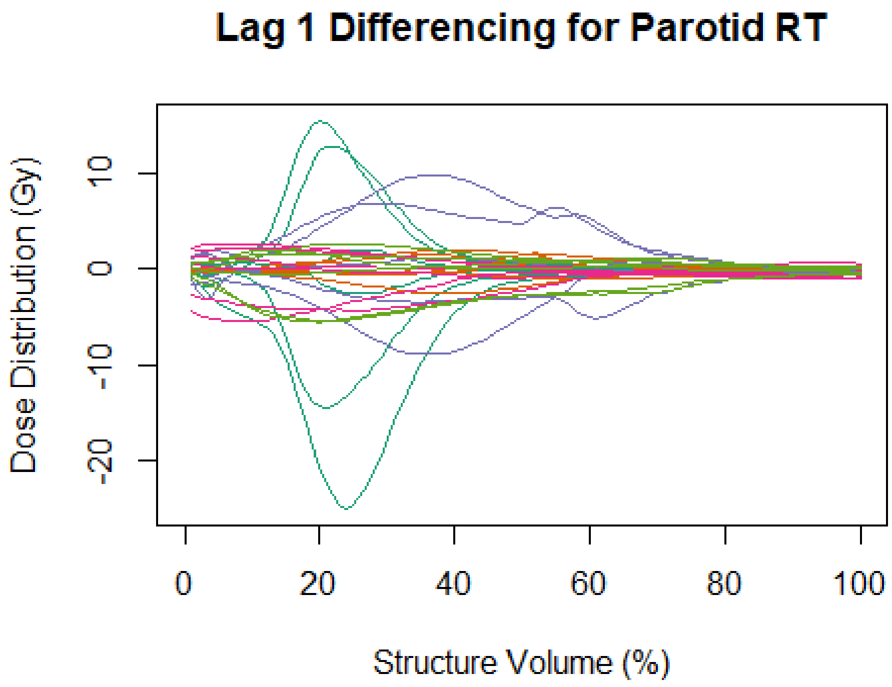
```
lagParotidRT(16,20)
```



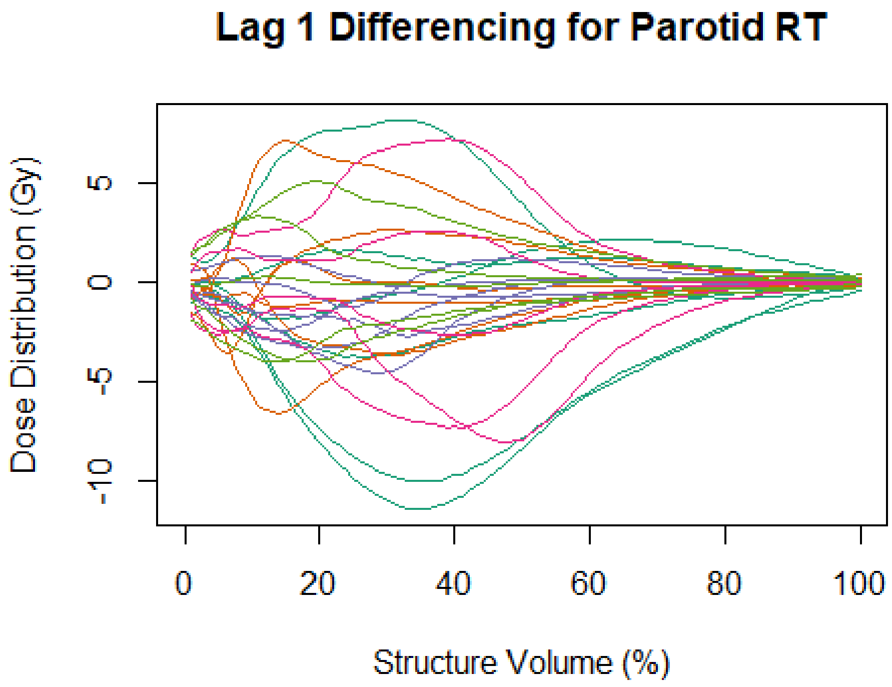
```
lagParotidRT(21,25)
```



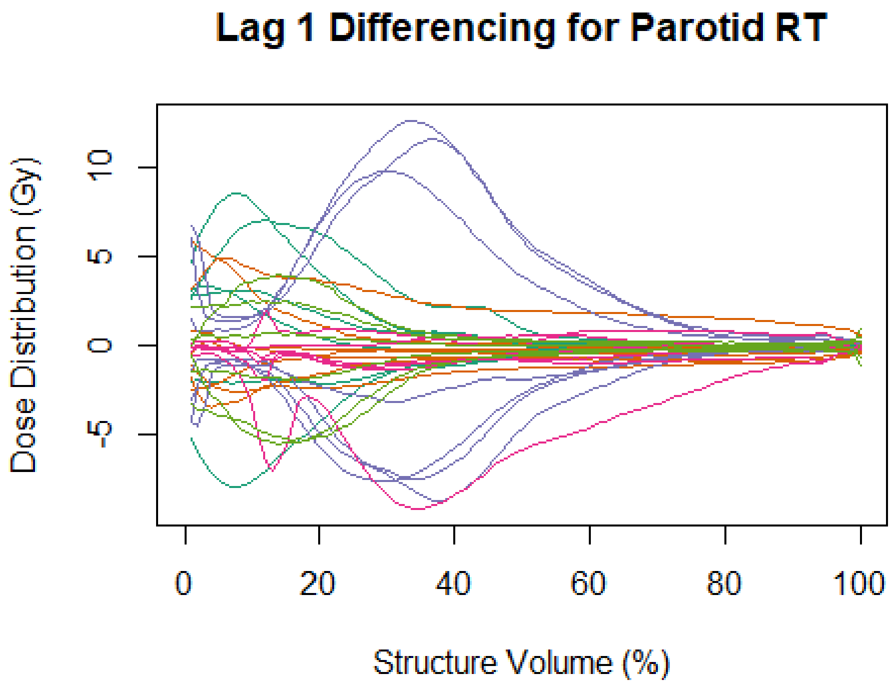
```
lagParotidRT(26,30)
```



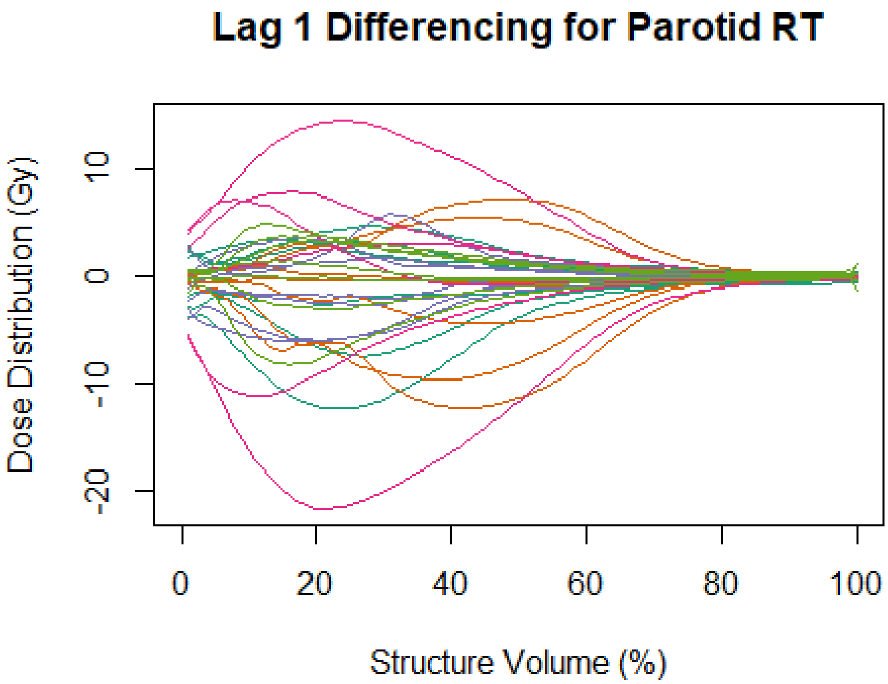
```
lagParotidRT(31,35)
```



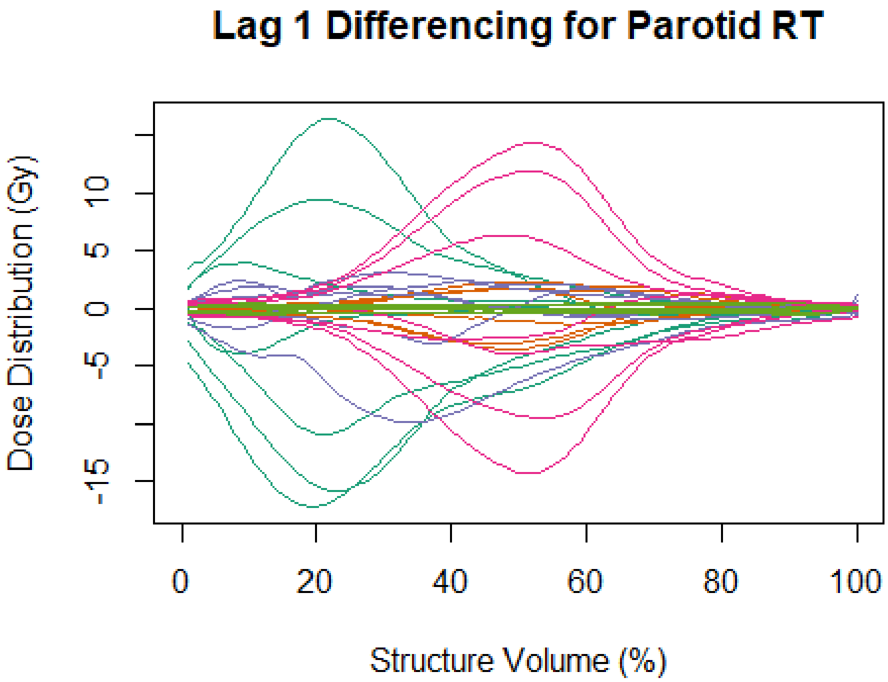
```
lagParotidRT(36,40)
```



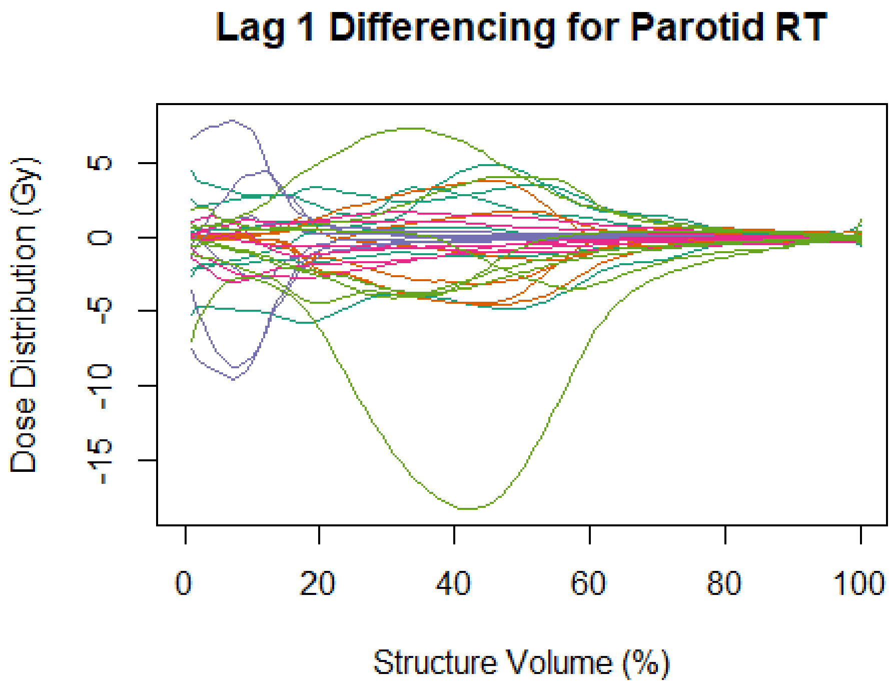
```
lagParotidRT(41,45)
```



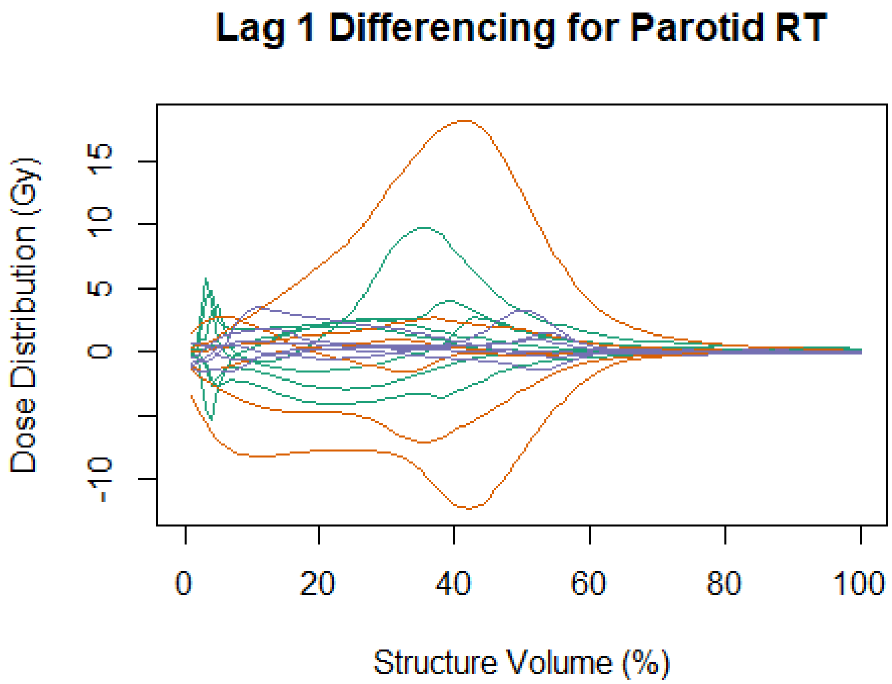
```
lagParotidRT(46,50)
```



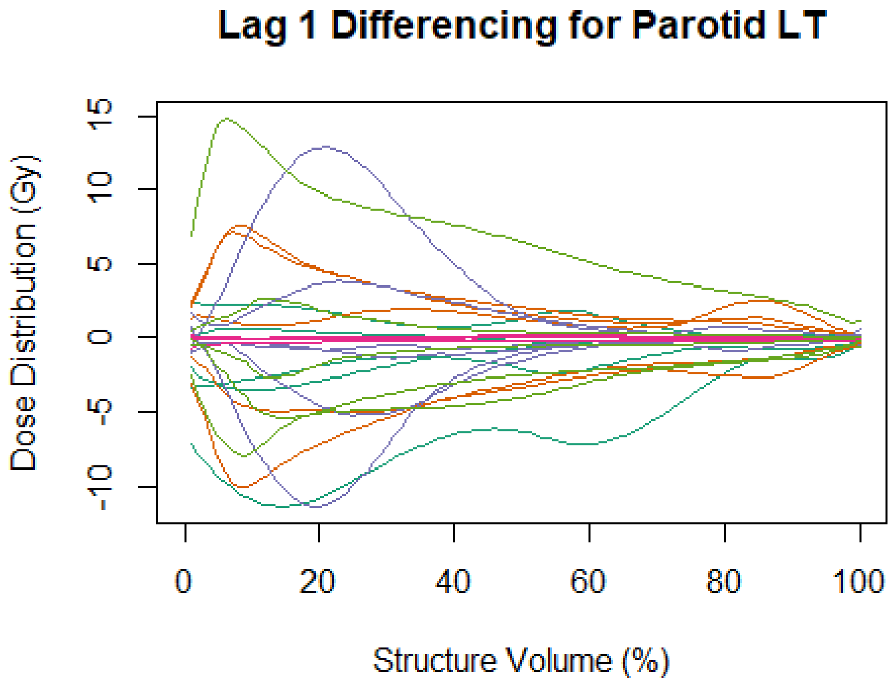
```
lagParotidRT(51,55)
```



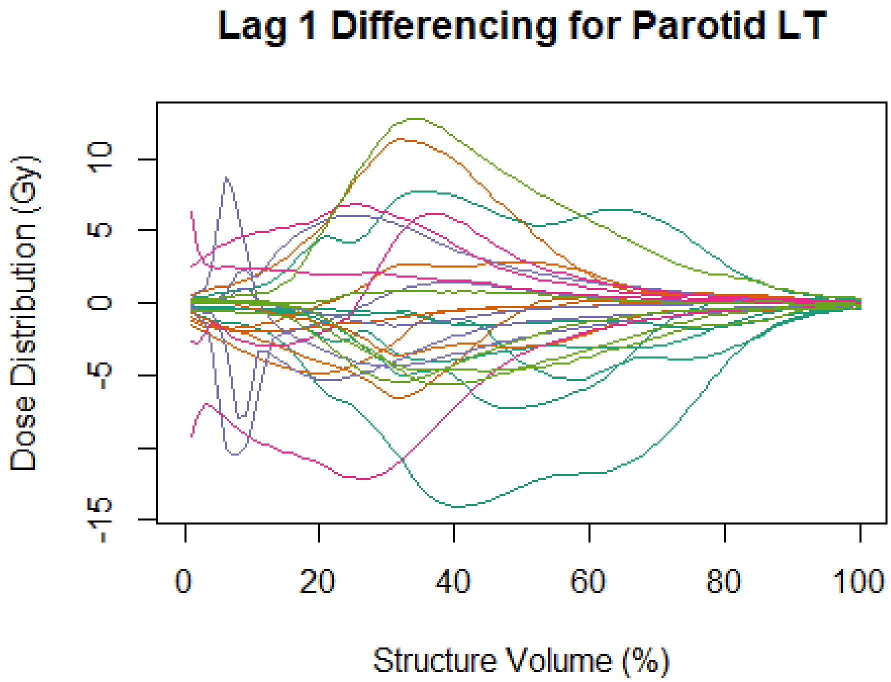
```
lagParotidRT(56,58)
```



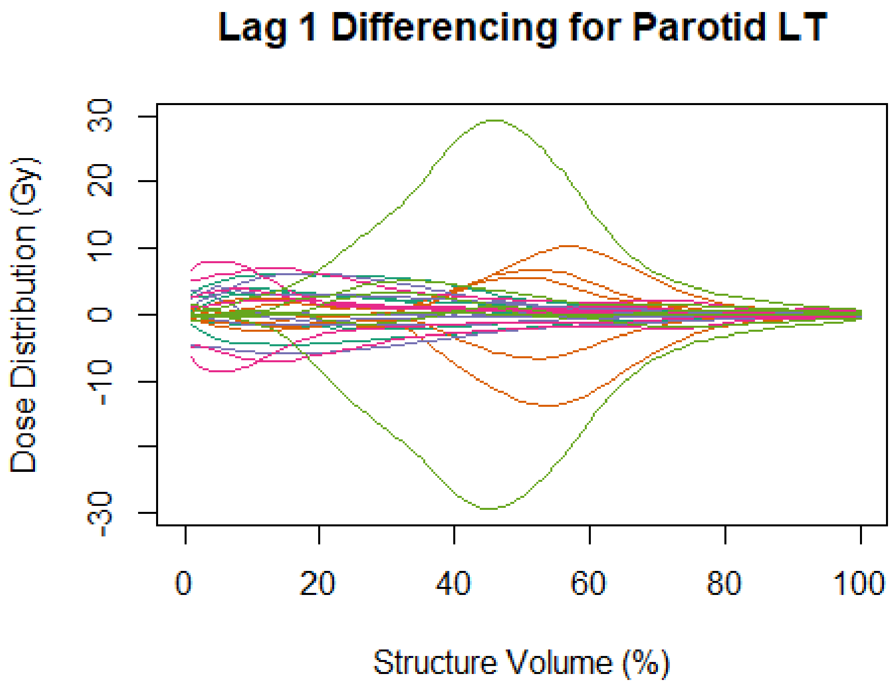
```
lagParotidLT(1,5)
```



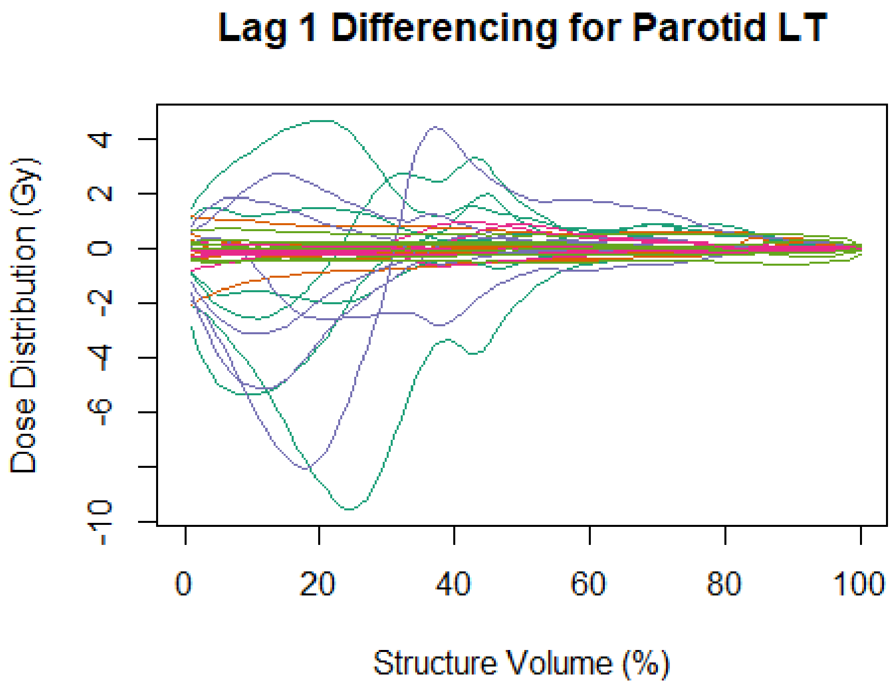
```
lagParotidLT(6,10)
```



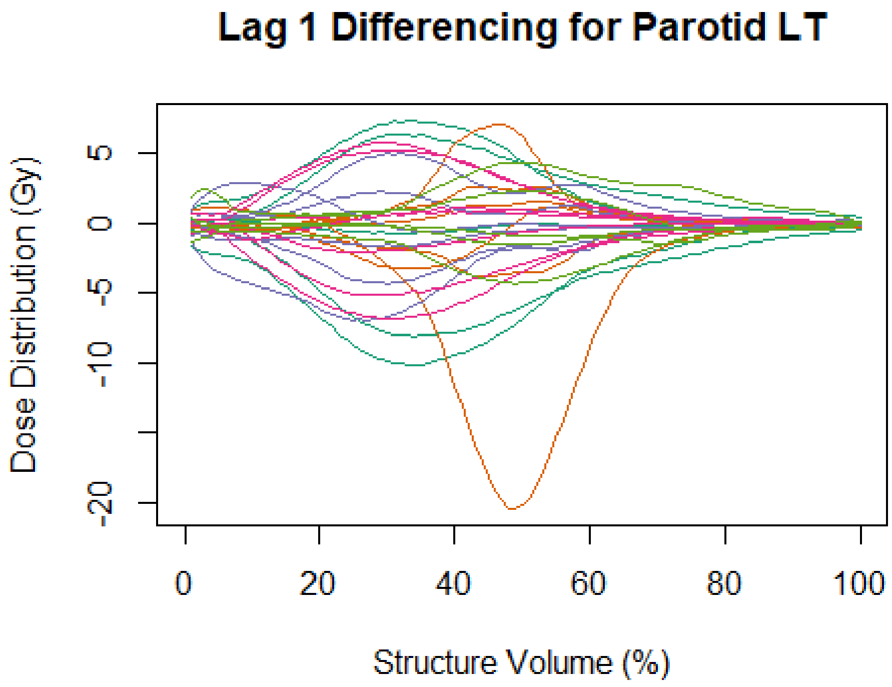
```
lagParotidLT(11,15)
```



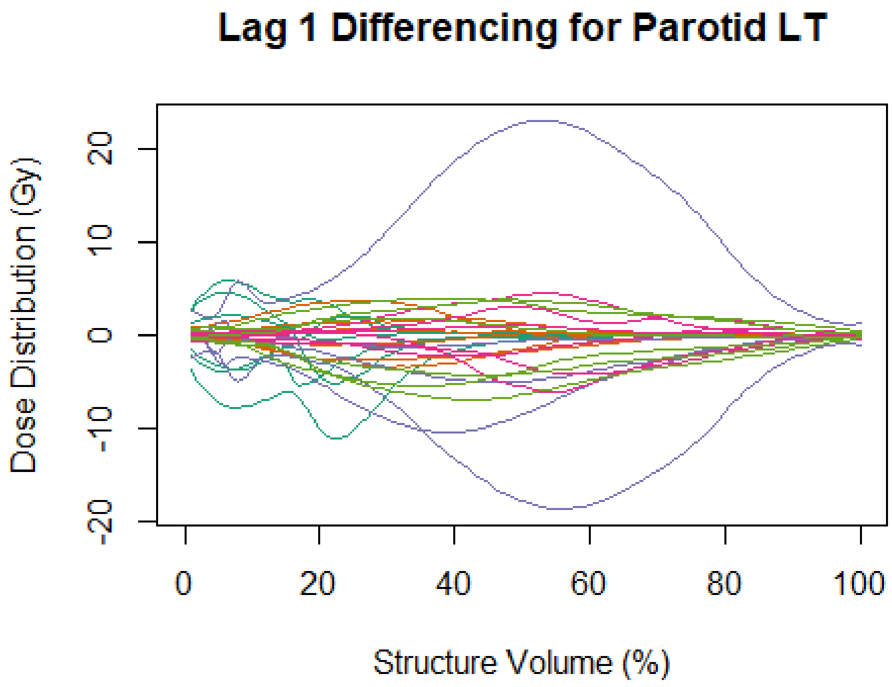
```
lagParotidLT(16,20)
```



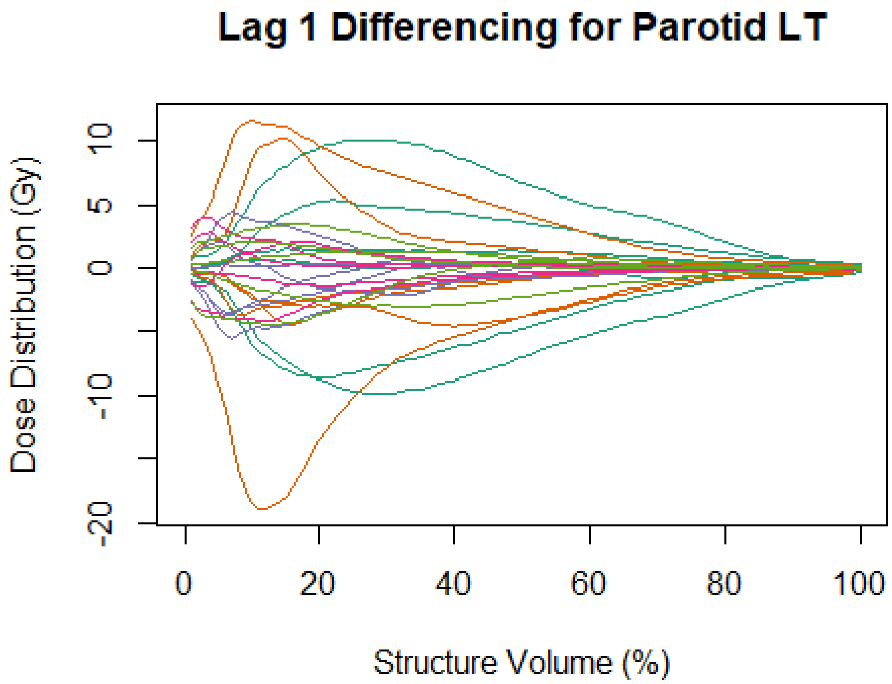
```
lagParotidLT(21,25)
```



```
lagParotidLT(26,30)
```

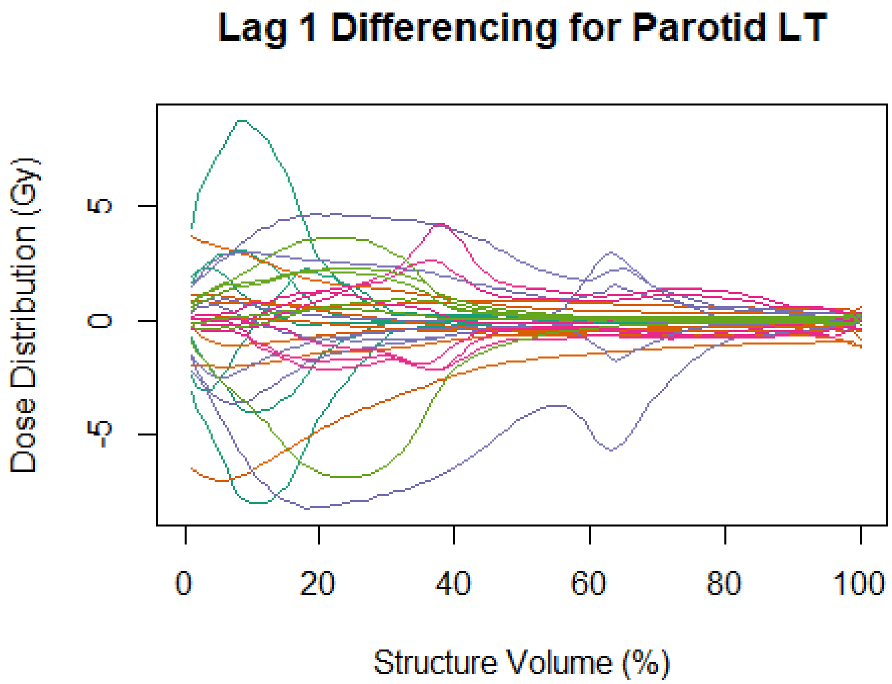


```
lagParotidLT(31,35)
```



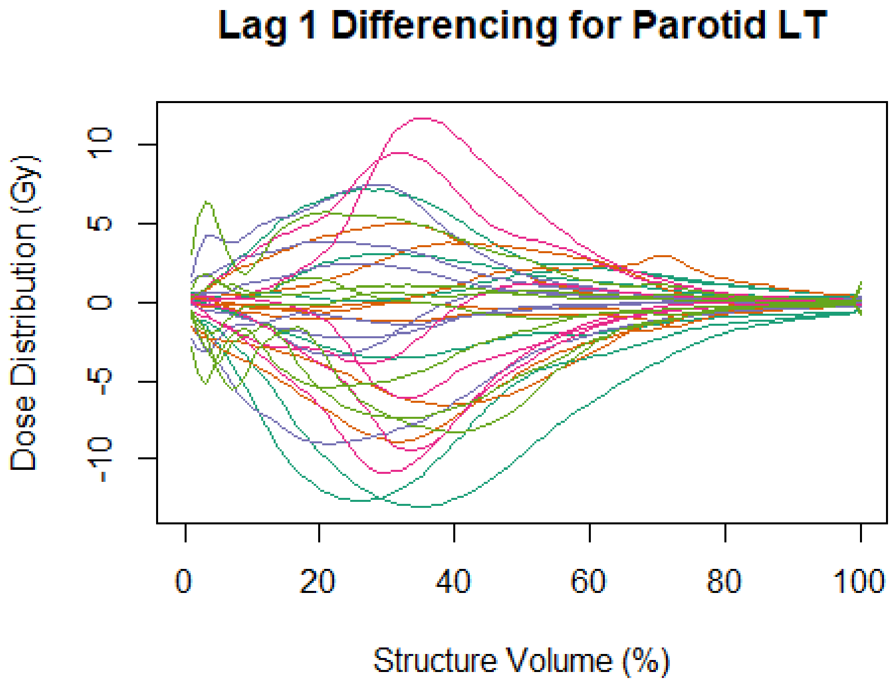
Lag 1 Differencing for Parotid LT

```
lagParotidLT(36,40)
```

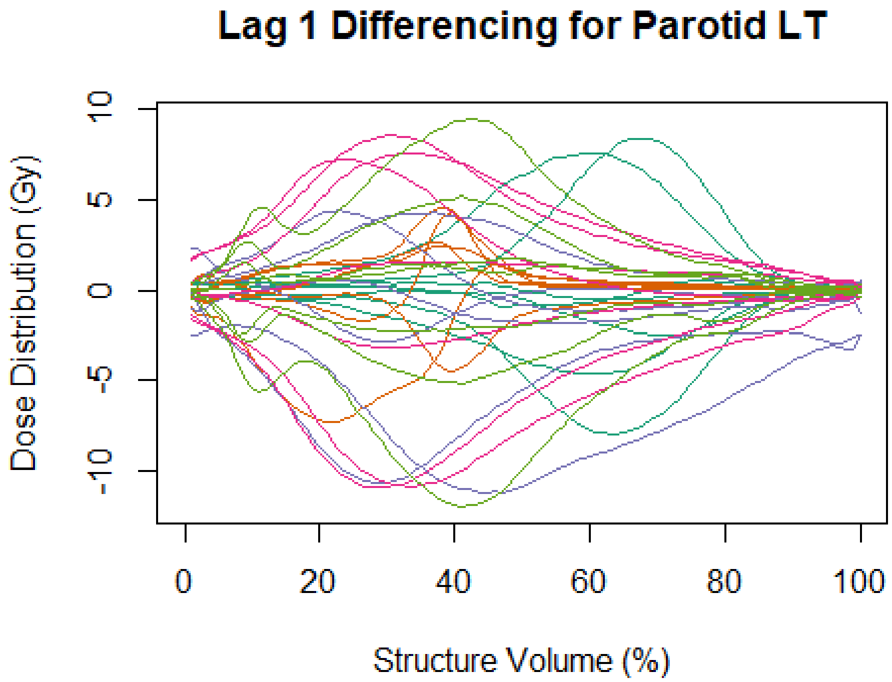


Lag 1 Differencing for Parotid LT

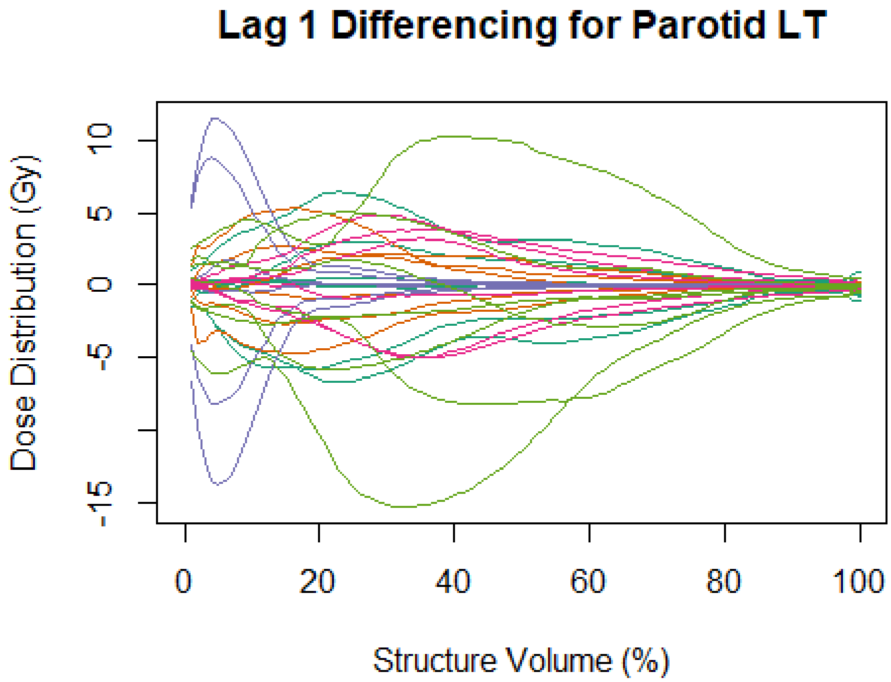
```
lagParotidLT(41,45)
```



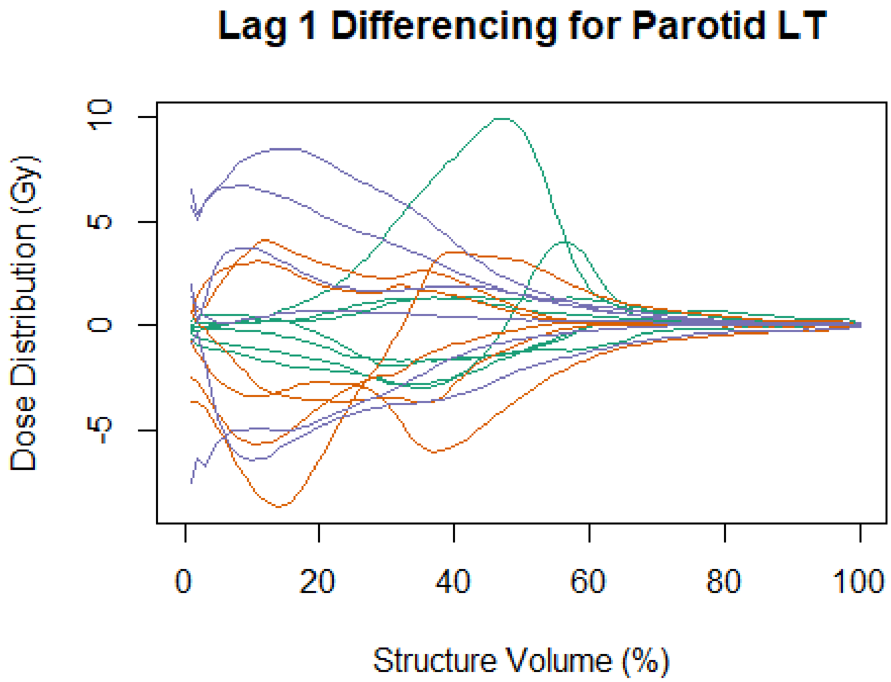
```
lagParotidLT(46,50)
```



```
lagParotidLT(51,55)
```



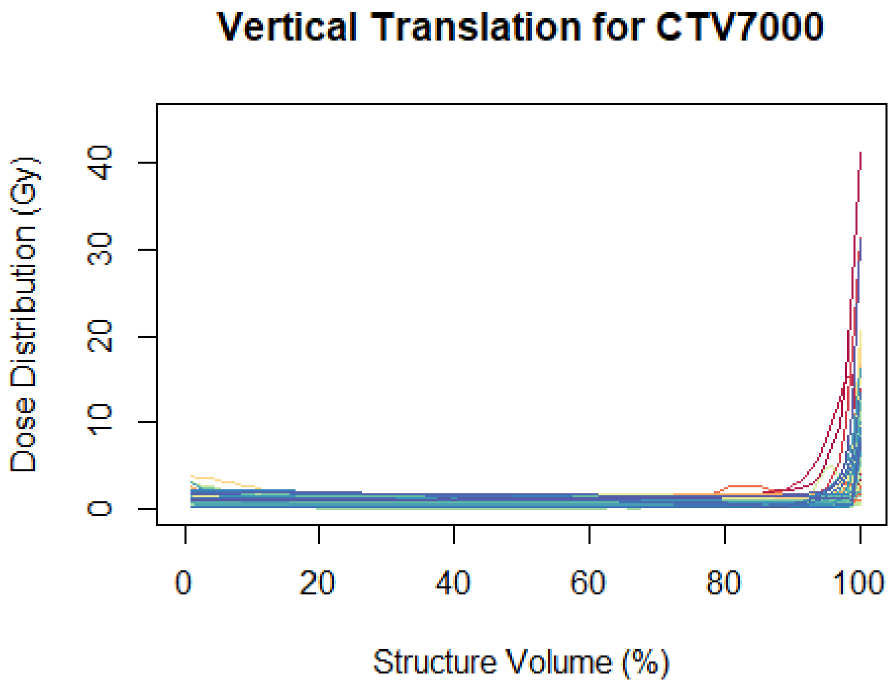
```
lagParotidLT(56,58)
```



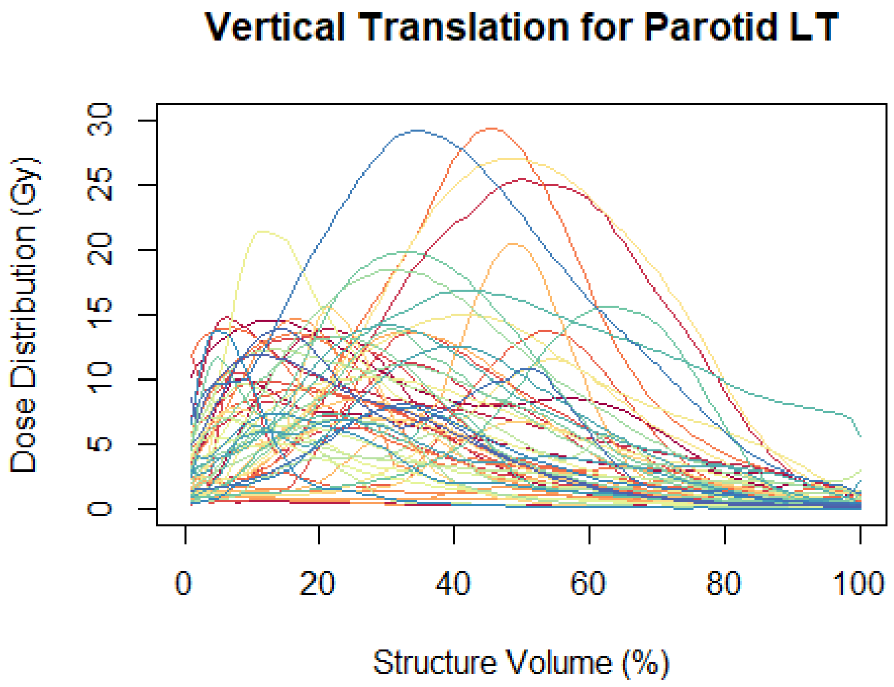
Vertical Translation

- to visualize random effects - random noise?

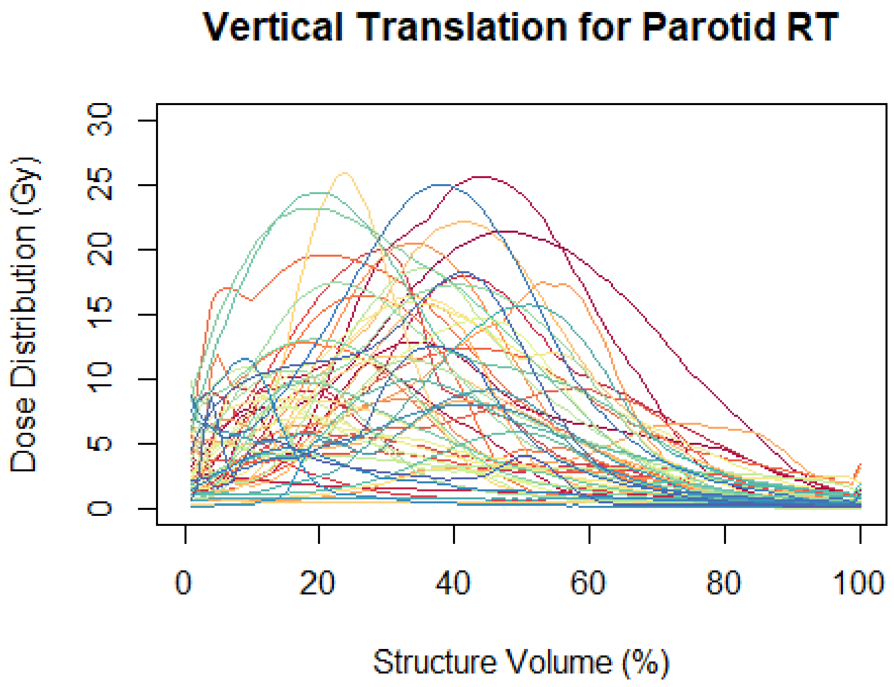
```
library("RColorBrewer")
#Initializing a color vector to help us color code our graphs for each
patient
color = brewer.pal(n = 10,name = "Spectral")
color = colorRampPalette(color)(60)

verticaltranslation.ParotidRT <- function(x,y)
{
plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)", main =
"Vertical Translation for Parotid RT" ,xlim = c(0,100), ylim = c(0,30))#,asp
= 2)

for (i in x:y)
{
  vtrans <- c()
  for (j in 4:103)
  {
    ind1 = ParotidRT_ptrows[i]
    ind2 = ParotidRT_ptrows[i+1] - 1
    mindose = min(DVH.ParotidRT[ind1:ind2,j])
    maxdose = max(DVH.ParotidRT[ind1:ind2,j])
    trans = maxdose-mindose
    vtrans <- c(vtrans,trans)

  }
  lines(seq(1,100,1),vtrans, col = color[i-x+1])
}
}

verticaltranslation.ParotidLT <- function(x,y)
{
plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)", main =
"Vertical Translation for Parotid LT" ,xlim = c(0,100), ylim = c(0,30))#,asp
= 2)

for (i in x:y)
{
  vtrans <- c()
  for (j in 4:103)
  {

    ind1 = ParotidLT_ptrows[i]
    ind2 = ParotidLT_ptrows[i+1] - 1
    mindose = min(DVH.ParotidLT[ind1:ind2,j])
    maxdose = max(DVH.ParotidLT[ind1:ind2,j])
    trans = maxdose-mindose
    vtrans <- c(vtrans,trans)
  }
  lines(seq(1,100,1),vtrans, col = color[i-x+1])
}
}

verticaltranslation.CTV7000 <- function(x,y)
{
plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)", main =
"Vertical Translation for CTV7000" ,xlim = c(0,100), ylim = c(0,45))#,asp =
2)

for (i in x:y)
{
  vtrans <- c()
  for (j in 4:103)
  {
    #mindose = min(DVH.CTV7000[CTV7000_ptrows[i]:CTV7000_ptrows[i+1]],j)
    #dvol = DVH.CTV7000[CTV7000_ptrows[i]:CTV7000_ptrows[i+1],j]
    ind1 = CTV7000_ptrows[i]
    ind2 = CTV7000_ptrows[i+1] - 1
    #print(ind2)
    mindose = min(DVH.CTV7000[ind1:ind2,j])
    maxdose = max(DVH.CTV7000[ind1:ind2,j])
    trans = maxdose-mindose
    vtrans <- c(vtrans,trans)

  }
  lines(seq(1,100,1),vtrans, col = color[i-x+1])
}
}

verticaltranslation.CTV7000(1,59)
```



```
verticaltranslation.ParotidLT(1,58)
```



```
verticaltranslation.ParotidRT(1,58)
```

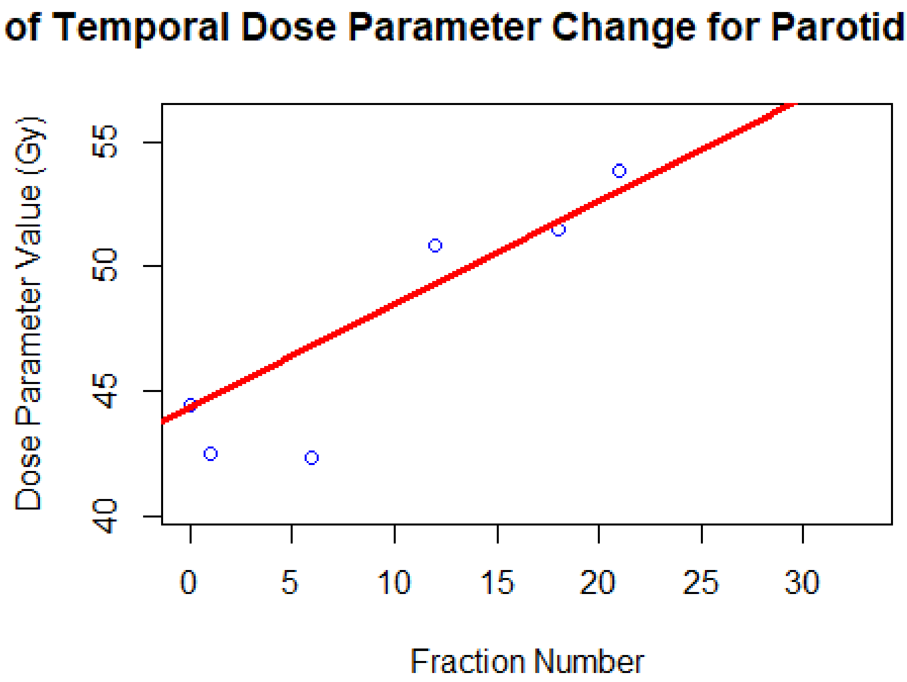


Taking the slope and standard error of each dose parameter with resepct with time

Sample function that looks at each dose parameter separately and fits a line corresponding to the change in that parameter with respect to time. Return Values: 1. Maximum slope of a line of best fit when considering all dose parameters - adaptive radiaton therapy related, greatest systematic trend 2. Index of the maximum slope 3. Maximum standard error of a line of best fit when considering all dose considering all dose parameters - daily set up realted, greatest random noise 4. Index of the maximum standard error

Parotid RT

```
featuresParotidRT <- function(i)
{

vecSlope = rep(NA,100)
vecSlopeSE = rep(NA,100)

numEl = ParotidRT_ptrows[i+1] - ParotidRT_ptrows[i]
vecInd = c(ParotidRT_ptrows[i]:(ParotidRT_ptrows[i+1] - 1))

for (j in 1:100)
  {
  vecDoseParam = DVH.ParotidRT[vecInd,3+j]
  vecFraction = DVH.ParotidRT[vecInd,3]
  # Weighting the first observation (CTsim) more heavily to ensure line of
best fit passess through it
  vecWeights = rep(1,numEl)
  vecWeights[1] <- 100
  lm.fit = lm(vecDoseParam ~ vecFraction, weights = vecWeights)
  #lm.fit = lm(vecDoseParam ~ vecFraction)

  #plot(NULL,xlab="Structure Volume (%)", ylab="Dose Distribution (Gy)",xlim
= c(0,33), ylim = c(my.min(DVH.ParotidRT[vecInd,c(3+j)]) -
2,my.max(DVH.ParotidRT[vecInd,c(3+j)]) + 2), main="2D Graph of Temporal Dose
Parameter Change for Parotid RT")
  #points(vecFraction, vecDoseParam, col = "blue")
  #abline(lm.fit,lwd = 3, col = "red")

  #vecIntercept[j] = coef(summary(lm.fit))[1,1]
  vecSlope[j] = coef(summary(lm.fit))[2,1]
  vecSlopeSE[j] = coef(summary(lm.fit))[2,2]
}

maxSlope = max(vecSlope)
maxSlopeInd = which.max(vecSlope)
maxSlopeSE = max(vecSlopeSE)
maxSlopeSEInd = which.max(vecSlopeSE)

plot(NULL,xlab="Fraction Number", ylab="Dose Parameter Value (Gy)",xlim =
c(0,33), ylim = c(my.min(DVH.ParotidRT[vecInd,maxSlopeInd+3]) -
2,my.max(DVH.ParotidRT[vecInd,maxSlopeInd+3]) + 2), main="2D Graph of
Temporal Dose Parameter Change for Parotid RT (at max slope)")
points(vecFraction, DVH.ParotidRT[vecInd,maxSlopeInd+3], col = "blue")
lmMax.fit = lm(DVH.ParotidRT[vecInd,maxSlopeInd+3] ~ vecFraction, weights =
vecWeights)
abline(lmMax.fit,lwd = 3, col = "red")

returnVec = c(maxSlope, maxSlopeInd, maxSlopeSE, maxSlopeSEInd)

}

library("openxlsx")

## Warning: package 'openxlsx' was built under R version 3.5.2

numRows = length(ParotidRT_ptrows) - 1
ptMat_toExport = matrix(nrow = numRows, ncol = 4)
for (i in 1:numRows){
  ptMat_toExport[i,] = featuresParotidRT(i)
}
```



of Temporal Dose Parameter Change for Parotid RT

```
write.xlsx(ptMat_toExport,file = "exportTest_ParotidRT.xlsx")
```

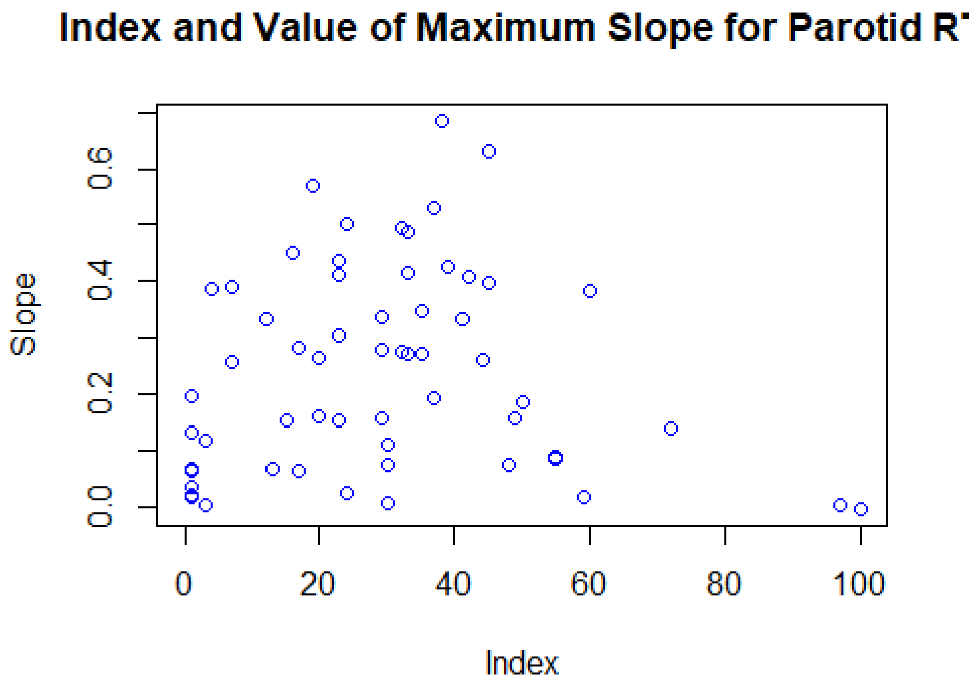
Using the data extracted and ML methods:

```
library(tree)

## Warning: package 'tree' was built under R version 3.5.2

rm(list = ls())

dataTypes = c("integer","integer","factor","integer","factor",
              "factor","factor","factor","factor","factor",
              "factor","factor","numeric","numeric","factor",
              "factor","numeric","numeric","numeric","numeric",
              "numeric","numeric","numeric","numeric","numeric",
              "numeric","numeric","numeric","integer","numeric",
              "integer")

Data_ParotidRT.Full <- read.csv("AnonymizedData_forML.csv", header = TRUE,
                    colClasses = dataTypes, row.names = NULL)

# Min and Max function for tables with NA
my.max <- function(x) ifelse( !all(is.na(x)), max(x, na.rm=T), NA)
my.min <- function(x) ifelse( !all(is.na(x)), min(x, na.rm=T), NA)
```

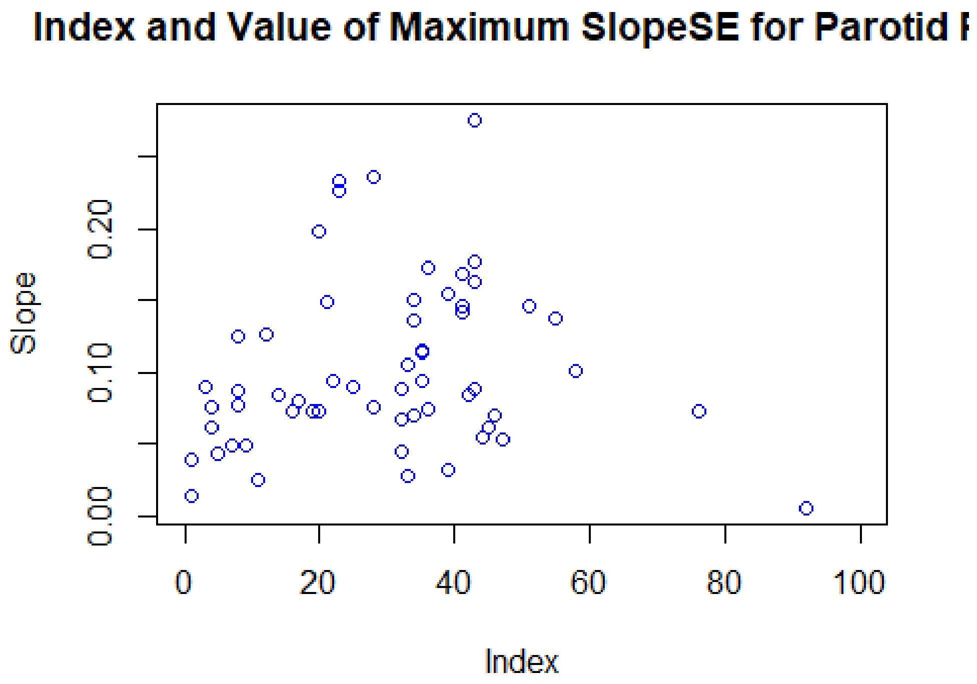
```
#Domain-based Feature Selection
Data_ParotidRT.Select =
Data_ParotidRT.Full[c(1:34,36:60),c(5:8,13:14,23,28:31)]
```

Clustering

```
plot(NULL,xlab="Index", ylab="Slope",xlim = c(0,100), ylim =
c(my.min(Data_ParotidRT.Select[,8]), my.max(Data_ParotidRT.Select[,8])),
main="Index and Value of Maximum Slope for Parotid RT")
points(Data_ParotidRT.Select[,9], Data_ParotidRT.Select[,8], col = "blue")
```



```
plot(NULL,xlab="Index", ylab="Slope",xlim = c(0,100), ylim =
c(my.min(Data_ParotidRT.Select[,10]), my.max(Data_ParotidRT.Select[,10])),
main="Index and Value of Maximum SlopeSE for Parotid RT")
points(Data_ParotidRT.Select[,11], Data_ParotidRT.Select[,10], col = "blue")
```

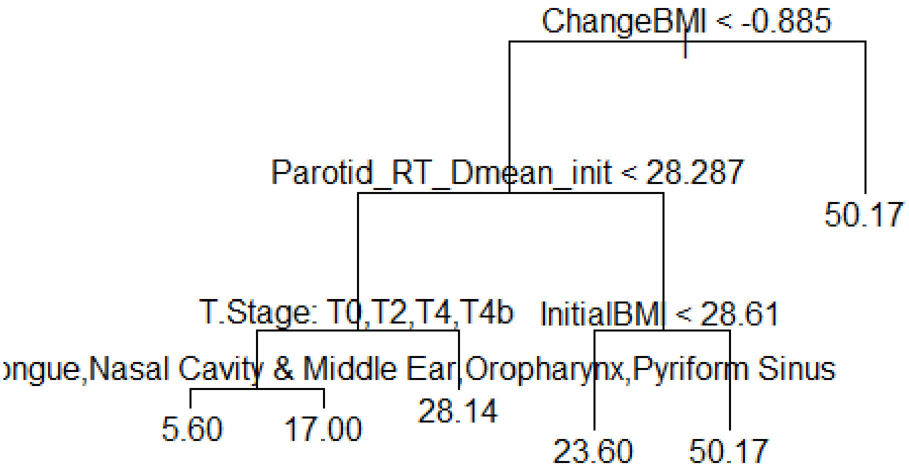


```
##All Data
Predictors = Data_ParotidRT.Select[,c(1:7)]
Response = Data_ParotidRT.Select[,c(9)]

set.seed(2)
train = sample(1:nrow(Predictors),35)
Predictors.train = Predictors[train,]
Predictors.test = Predictors[-train,]
Response.train = Response[train]
Response.test = Response[-train]
```

Simple Classification Tree

```
result.tree = tree(Response.train~.,Predictors.train)
summary(result.tree)

##
## Regression tree:
## tree(formula = Response.train ~ ., data = Predictors.train)
## Variables actually used in tree construction:
## [1] "ChangeBMI"             "Parotid_RT_Dmean_init" "T.Stage"
## [4] "Cancer.Site"           "InitialBMI"
## Number of terminal nodes:  6
## Residual mean deviance:  289 = 8381 / 29
## Distribution of residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -21.170  -7.667   0.000   0.000   6.200  49.830
```

```
plot(result.tree)
text(result.tree,pretty=0)
```



ChangeBMI < -0.885
Parotid_RT_Dmean_init < 28.287
50.17
T.Stage: T0,T2,T4,T4b
InitialBMI < 28.61
ongue,Nasal Cavity & Middle Ear,Oropharynx,Pyriform Sinus
5.60
17.00
28.14
23.60
50.17