About the technique - fit\_curvature\_max: computes curvature via the second derivative of a smoothed spline and returns the maximum curvature position for decreasing curves; useful to choose a trade-off point where further reductions add little benefit.

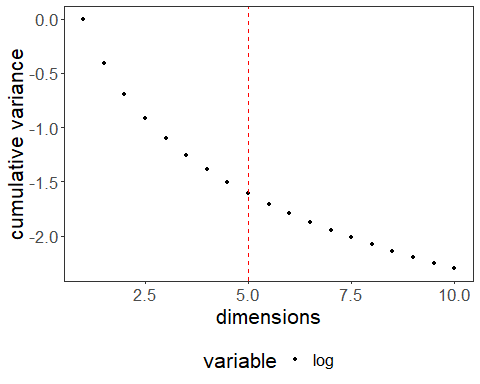
# installation   
#install.packages("daltoolbox")  
  
# loading DAL  
library(daltoolbox)   
  
# for ploting  
library(ggplot2)  
library(dplyr)

Synthetic example of a decreasing curve (-log) to illustrate the maximum curvature point.

# Maximum curvature  
# If the curve is decreasing, use maximum curvature analysis.   
# It brings a trade-off between having lower x values (with not so low y values) and having higher x values (not having to much decrease in y values).   
  
x <- seq(from=1,to=10,by=0.5)  
dat <- data.frame(x = x, value = -log(x), variable = as.factor("log"))  
myfit <- fit\_curvature\_max()  
res <- transform(myfit, dat$value)  
head(res)

## x y yfit  
## 1 9 -1.609438 9.224359e-08

grf <- plot\_scatter(dat, label\_x = "dimensions", label\_y = "cumulative variance", colors="black") +   
 theme(text = element\_text(size=16))  
plot(grf + geom\_vline(xintercept = dat$x[res$x], linetype="dashed", color = "red", size=0.5))



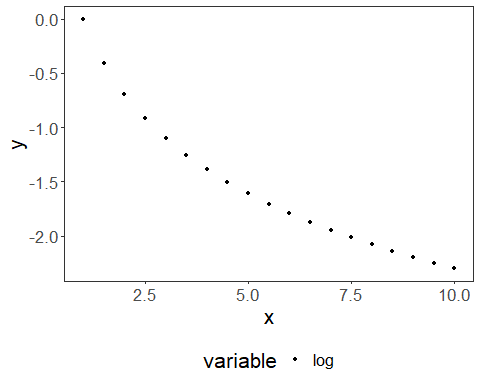
About the technique - fit\_curvature\_max: computes curvature via the second derivative of a smoothed spline over the sequence and returns the maximum curvature position (elbow) for decreasing or concave curves; useful to identify a knee where diminishing returns begin.

Environment setup.

# installation   
#install.packages("daltoolbox")  
  
# loading DAL  
library(daltoolbox)   
  
# for plotting  
library(ggplot2)  
library(dplyr)

Example curve and elbow detection.

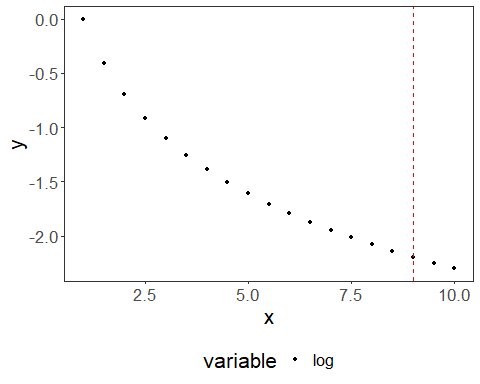
x <- seq(from=1,to=10,by=0.5)  
dat <- data.frame(x = x, value = -log(x), variable = "log")  
dat$variable <- as.factor(dat$variable)  
grf <- plot\_scatter(dat, label\_x = "x", label\_y = "y", colors="black") +   
 theme(text = element\_text(size=16))  
plot(grf)



myfit <- fit\_curvature\_max()  
res <- transform(myfit, dat$value)  
res

## x y yfit  
## 1 9 -1.609438 9.224359e-08

plot(grf + geom\_vline(xintercept = res$x, linetype="dashed", color = "red", size=0.5))



References - Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B. (2011). Finding a “Kneedle” in a Haystack: Detecting Knee Points in System Behavior.