Wear Track Analysis

Gokce Mehmet AY

22 Mayıs 2015

Wear Track Analysis from Profilometer Data

Profilometer such as the one I use in lab Mitutoyo RJ210 output a nice report and raw data. One can get data, copy to a spreadsheet software such as MS Excel and can find relevant information from it, such as wear track area. However when tens of data needs to be analysed one needs a faster solution. A quick solution is R

Data

Mutitoya Profilometer Data has a long header and than 2 columns. These are Distance (mm) and Profile (µm). Importing these are fairly easy using read.csv. However in order to automate one has to import more than one data and plot and calculate. So another solution is needed.

Import data

Using list.files we can get list of CSV files.

```
filenames <- list.files(pattern="*.csv")
```

Then using lapply we'll import all those files to a list named All.

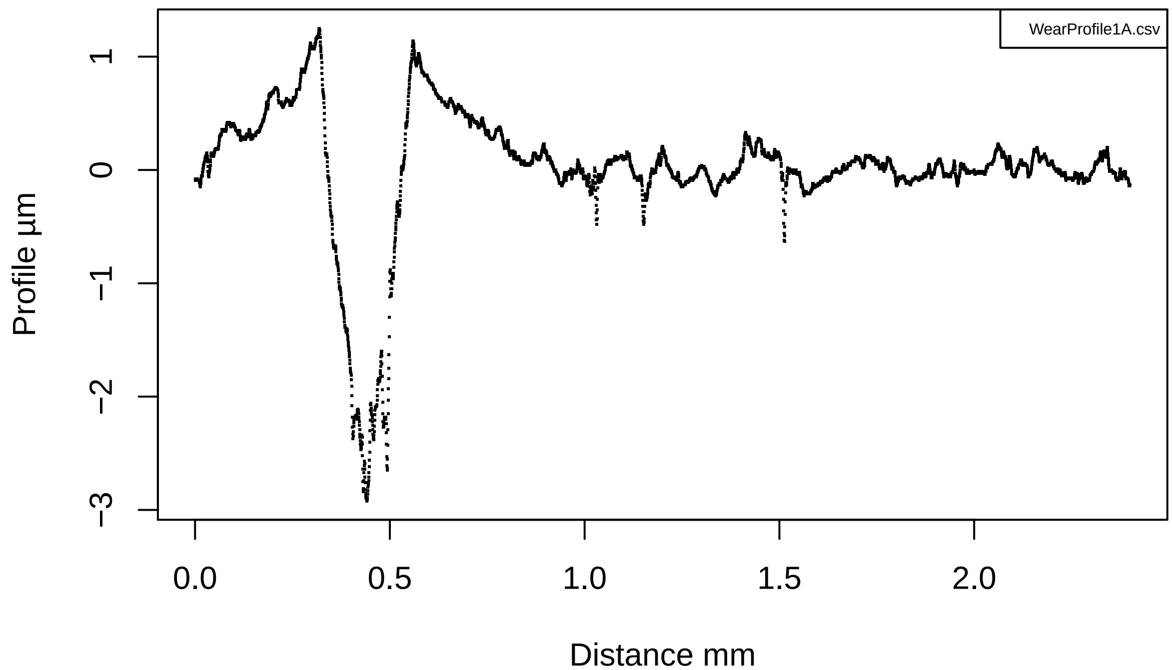
```
All <- lapply(filenames,function(i){
  read.csv(i, sep = ",",quote = "\"", dec = ",", fill = TRUE, comment.char = "")
})
```

Plots

Plot surface profile

Now we've got surface data, we can plot it.

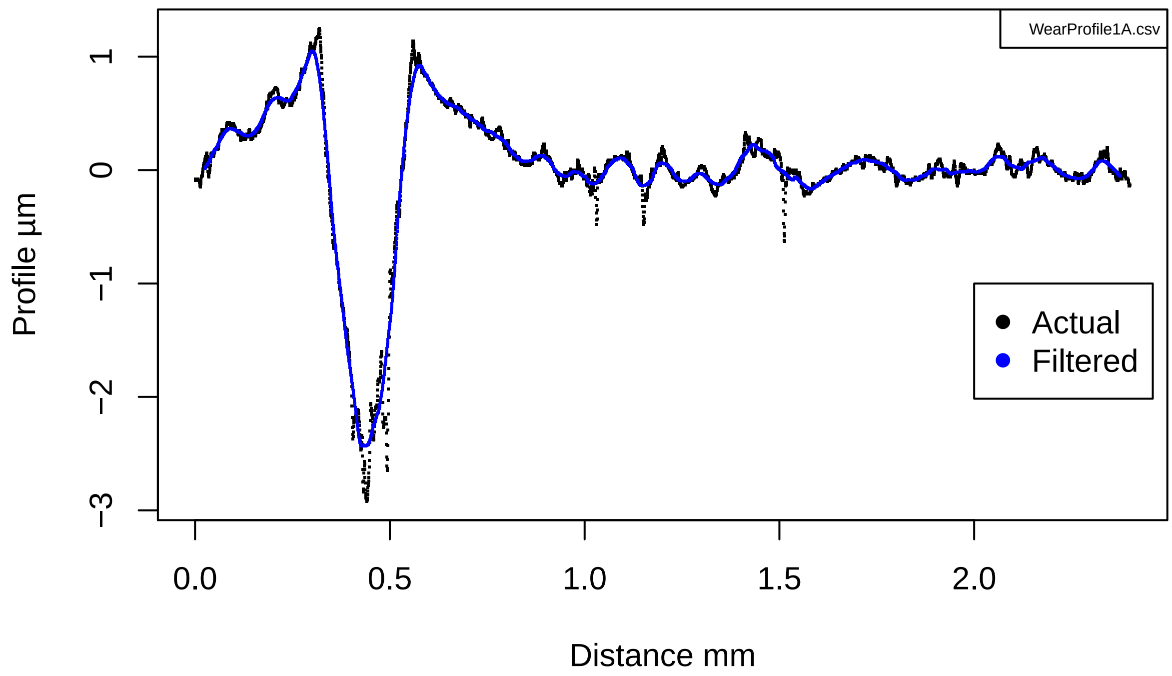
```
EndV<-length(All[[i]]$Profile)-1        # Set End point
plot(All[[i]]$Distance, All[[i]]$Profile,pch='.',xlab='Distance mm',ylab='Profile µm') # Plot Full Data
legend("topright", legend=filenames[i],cex=.5)     # Add filename legend
```



Determine wear track

As you can see there is a definite wear track with a minimum at around -3µm. However in order to find peaks we need to filter data. Moving average is a robust method for filtering.

```
EndV<-length(All[[i]]$Profile)-1        # Set End point
plot(All[[i]]$Distance, All[[i]]$Profile,pch='.',xlab='Distance mm',ylab='Profile µm') # Plot Full Data
legend("topright", legend=filenames[i],cex=.5)     # Add filename legend
#Apply filter to data in order to eliminate noise Filter is moving average with 100 elements
fltN <- 100 #lag
flt1 <- rep(1/fltN,fltN)    # Filter
ProfileFiltered<- filter(All[[i]]$Profile,flt1,sides=1) #filter profile
points(filter(All[[i]]$Distance,flt1,sides=1),filter(All[[i]]$Profile,flt1,sides=1),col='blue',pch='.')
legend(2,-1,legend = c("Actual","Filtered"), col = c("black", "blue"),pch = 16)
```



Next step is finding left and right peak values for wear track. In order to find those first we'll find minimum value of profile which is assumed to be the bottom of wear track. Than we'll find right maximum and left maximum.

```
MinF <- min(ProfileFiltered,na.rm=TRUE)    #Find filtered min
PMinF <- match(MinF,ProfileFiltered)       # Get position of filtered min


# Get left maximum from minimum
MaxF1 <-max(ProfileFiltered[0:PMinF],na.rm=TRUE)  # Find max value on filtered
PMaxF1 <- match(MaxF1,ProfileFiltered)  #Find position of maximum value in filtered vector


Max1<-max(All[[i]]$Profile[0:PMinF],na.rm=TRUE)       #Find maximum value on actual
PMax1<- match(Max1,All[[i]]$Profile)                # Get positon of maximum value on actual



# Getting maximum value near filterd maximum value
# Check for PMaxF1-PMax1 if it's 50 points near use that value for Maximum
while (All[[i]]$Distance[PMaxF1]-All[[i]]$Distance[PMax1]>0.03) {
  PMax1<-PMax1+1;
  Max1 <-max(All[[i]]$Profile[PMax1:PMinF],na.rm=TRUE);
  PMax1<- match(Max1,All[[i]]$Profile[PMax1:PMinF])+PMax1}

# Get right maximum from minimum
```

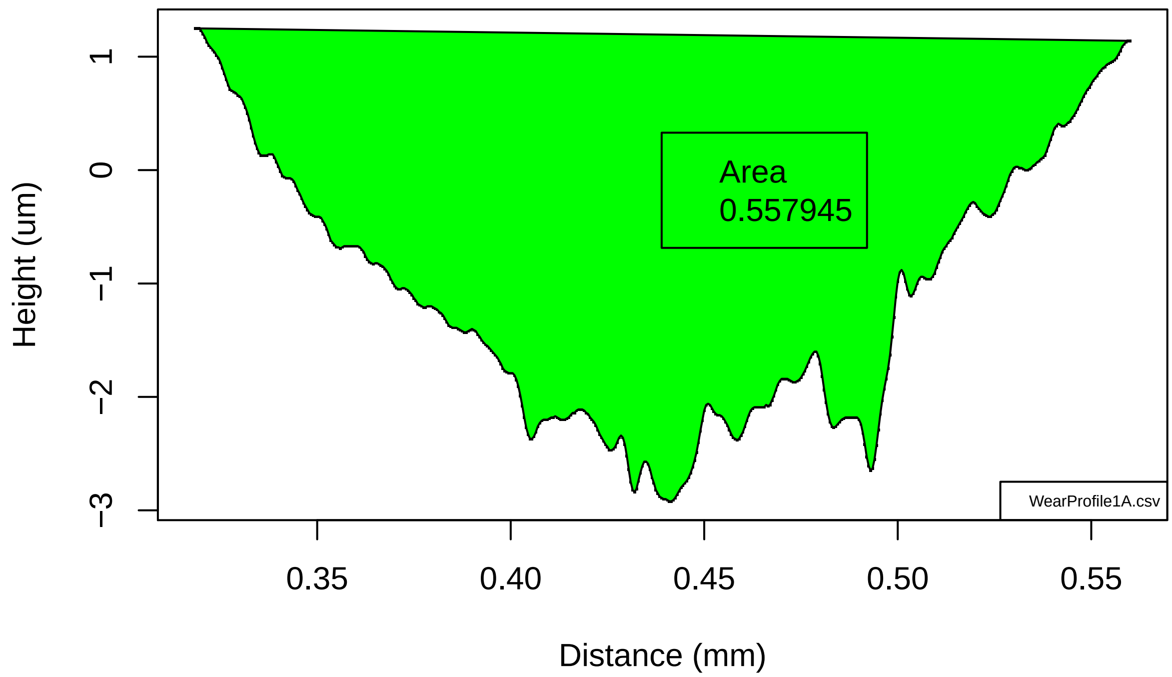
```
MaxF2 <-max(ProfileFiltered[PMinF:EndV],na.rm=TRUE)  # Find max value on filtered
PMaxF2 <- match(MaxF2,ProfileFiltered[PMinF:EndV])+PMinF  #Find position of maximum value in filtered v


Max2<-max(All[[i]]$Profile[PMinF:EndV],na.rm=TRUE)      #Find maximum value on actual
PMax2<- match(Max2,All[[i]]$Profile[PMinF:EndV])+PMinF                # Get positon of maximum value on


# Getting maximum value near filterd maximum value
# Check for PMaxF1-PMax1 if it's 50 points near use that value for Maximum
while (All[[i]]$Distance[PMax2]-All[[i]]$Distance[PMaxF2]>0.03) {
  PMax2<-PMax2+1;
  Max2 <-max(All[[i]]$Profile[PMinF:PMax2],na.rm=TRUE);
  PMax2<- match(Max2,All[[i]]$Profile[PMinF:PMax2])+PMinF

}
```

As wear track is found now its time to plot it and calculating its area. First we'll calculate minimum value of wear track. Than we'll calculate area under curve and that minimum.

```
Min<-min(All[[i]]$Profile[PMax1:PMax2])      #Find minimum value of curve
AUCc <- sum(diff(All[[i]]$Distance[PMax1:PMax2])*rollmean(All[[i]]$Profile[PMax1:PMax2]-Min,2))      #Ca
```

Next we'll define a line between two peaks. Calculate area under and substract area under curve from that value.

```
xL <- c(All[[i]]$Distance[PMax1],All[[i]]$Distance[PMax2])   # A vector with start and end values for x
yL <- c(All[[i]]$Profile[PMax1],All[[i]]$Profile[PMax2])     # A vector with start and end values for y
AUCl <- sum(diff(xL)*rollmean(yL-Min,2))                    #Calculate area under line
AUC <- AUCl-AUCc                                           #Calculate area between
```



If you put all this into a for loop, you can get wear area of all your samples in seconds.

You can get this script from GitHub gokceay You can reach me at www.gmay.me