```
#R code for analyses and graphs in Jardine et al.
#The impact of oxidation on spore and pollen chemistry
#Journal of Micropalaeontology

#The code assumes that the data tables S1-S6 are saved directly from the
#Excel sheets as tab delimited text files (.txt)

lyco.spectra <- read.table (file = "Table S1.txt",
                            header=TRUE, row.names=1)
lyco.peaks <- read.table (file = "Table S2.txt",
                            header=TRUE, row.names=1)
lyco.info <- read.table (file = "Table S3.txt",
                             header=TRUE, row.names=1)

#Transpose spectra to put samples in rows and wavenumbers in columns
lyco.spectra <- t(lyco.spectra)

#Each sample standardised to mean = 0, variance = 1
lyco.spectra.mean <- apply(lyco.spectra, 1, mean)
lyco.spectra.minusMean <- apply(lyco.spectra, 2, function(x)
  x - lyco.spectra.mean)

lyco.spectra.minusMean.standev <- apply(lyco.spectra.minusMean,
                                        1, sd)
lyco.spectra.standardised <- apply(lyco.spectra.minusMean, 2,
                                   function(x)
                                     x / lyco.spectra.minusMean.standev)

#Test - check mean of each sample is 0 and variance 1
apply(lyco.spectra.standardised, 1, mean)
apply(lyco.spectra.standardised, 1, var)

plot(lyco.spectra[1,], type = "l")
plot(lyco.spectra.minusMean[1,], type = "l")
plot(lyco.spectra.standardised[1,], type = "l")

#Figure 1
wavenumbers <- colnames(lyco.spectra)
plot(wavenumbers, lyco.spectra[1,],
     type = "l", xlim = c(4000, 450), ylim = c(0, 0.2), lwd = 2,
     las = 1, ylab = "Absorbance", xlab = "Wavenumber (cm-1)",
     main = "Untreated")

#DIFFERENCE PLOTS
#Use averages for each treatment:
lyco.spectra.standardised.split <-
split(as.data.frame(lyco.spectra.standardised),
                                          lyco.info$Code)
lyco.spectra.standardised.splitMeans <- lapply(lyco.spectra.standardised.split,
colMeans)
lyco.spectra.standardised.codeMeans <- matrix(unlist

(lyco.spectra.standardised.splitMeans),
                                                nrow = 36, byrow = T)
rownames(lyco.spectra.standardised.codeMeans) = sort(unique(lyco.info$Code))
```

```
colnames(lyco.spectra.standardised.codeMeans) = colnames(lyco.spectra)
plot(lyco.spectra.standardised.codeMeans[1,], type = "l")

#Reorder to match original order of treatments:
lyco.codeMeans <- rbind(lyco.spectra.standardised.codeMeans
                        [match(unique(lyco.info$Code),

row.names(lyco.spectra.standardised.codeMeans)),])

#Figure 2
par(mfrow = c(1, 4))
#Acetolysis cold
plot(wavenumbers, seq(from = 1, to = 50, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 50, 10),
     yaxt = "n",
     las = 1, ylab = "Absorbance", xlab = expression(paste("Wavenumber ", (cm^-
1))),
     main = "A. Acetolysis, cold")

abline(h = 5, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[3,]-lyco.codeMeans[1,])+5,
      type = "l", lwd = 2)

abline(h = 10, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[4,]-lyco.codeMeans[1,])+10,
      type = "l", lwd = 2)

abline(h = 15, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[5,]-lyco.codeMeans[1,])+15,
      type = "l", lwd = 2)

abline(h = 20, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[6,]-lyco.codeMeans[1,])+20,
      type = "l", lwd = 2)

abline(h = 25, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[7,]-lyco.codeMeans[1,])+25,
      type = "l", lwd = 2)

abline(h = 30, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[8,]-lyco.codeMeans[1,])+30,
      type = "l", lwd = 2)

abline(h = 35, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[9,]-lyco.codeMeans[1,])+35,
      type = "l", lwd = 2)

abline(h = 40, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[10,]-lyco.codeMeans[1,])+40,
      type = "l", lwd = 2)

abline(h = 45, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[11,]-lyco.codeMeans[1,])+45,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46), sample.names[2:10], pos = 4)
```

```
#Acetolysis hot
plot(wavenumbers, seq(from = 1, to = 50, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 50, 10),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "B. Acetolysis, hot")

abline(h = 5, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[12,]-lyco.codeMeans[1,])+5,
      type = "l", lwd = 2)

abline(h = 10, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[13,]-lyco.codeMeans[1,])+10,
      type = "l", lwd = 2)

abline(h = 15, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[14,]-lyco.codeMeans[1,])+15,
      type = "l", lwd = 2)

abline(h = 20, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[15,]-lyco.codeMeans[1,])+20,
      type = "l", lwd = 2)

abline(h = 25, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[16,]-lyco.codeMeans[1,])+25,
      type = "l", lwd = 2)

abline(h = 30, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[17,]-lyco.codeMeans[1,])+30,
      type = "l", lwd = 2)

abline(h = 35, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[18,]-lyco.codeMeans[1,])+35,
      type = "l", lwd = 2)

abline(h = 40, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[19,]-lyco.codeMeans[1,])+40,
      type = "l", lwd = 2)

abline(h = 45, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[20,]-lyco.codeMeans[1,])+45,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46), sample.names[2:10], pos = 4)

#Nitric cold
plot(wavenumbers, seq(from = 1, to = 50, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 50, 10),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "C. Nitric, cold")

abline(h = 5, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[21,]-lyco.codeMeans[1,])+5,
      type = "l", lwd = 2)
```

```
abline(h = 10, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[22,]-lyco.codeMeans[1,])+10,
      type = "l", lwd = 2)

abline(h = 15, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[23,]-lyco.codeMeans[1,])+15,
      type = "l", lwd = 2)

abline(h = 20, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[24,]-lyco.codeMeans[1,])+20,
      type = "l", lwd = 2)

abline(h = 25, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[25,]-lyco.codeMeans[1,])+25,
      type = "l", lwd = 2)

abline(h = 30, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[26,]-lyco.codeMeans[1,])+30,
      type = "l", lwd = 2)

abline(h = 35, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[27,]-lyco.codeMeans[1,])+35,
      type = "l", lwd = 2)

abline(h = 40, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[28,]-lyco.codeMeans[1,])+40,
      type = "l", lwd = 2)

abline(h = 45, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[29,]-lyco.codeMeans[1,])+45,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46), sample.names[2:10], pos = 4)

#Nitric hot
plot(wavenumbers, seq(from = 1, to = 50, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 50, 10),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "D. Nitric, hot")

abline(h = 5, lty = 2, col = "grey40")
lines(wavenumbers, 2*(lyco.codeMeans[30,]-lyco.codeMeans[1,])+5,
      type = "l", lwd = 2)

abline(h = 10, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[31,]-lyco.codeMeans[1,])+10,
      type = "l", lwd = 2)

abline(h = 15, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[32,]-lyco.codeMeans[1,])+15,
      type = "l", lwd = 2)

abline(h = 20, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[33,]-lyco.codeMeans[1,])+20,
```

```
      type = "l", lwd = 2)

abline(h = 25, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[34,]-lyco.codeMeans[1,])+25,
      type = "l", lwd = 2)

abline(h = 30, lty = 2, col = "grey40")
lines(wavenumbers, (lyco.codeMeans[35,]-lyco.codeMeans[1,])+30,
      type = "l", lwd = 2)

abline(h = 35, lty = 2, col = "grey40")
lines(wavenumbers, 2*(lyco.codeMeans[36,]-lyco.codeMeans[1,])+35,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36), sample.names[2:8], pos = 4)

#PCA using prcomp function
lyco.pca <- prcomp(lyco.spectra.standardised)
#wavenumbers not scaled to var = 1, i.e. PCA carried on out covariance not
correlation matrix

#Importance of PCA axes
summary(lyco.pca)

barplot(summary(lyco.pca)[[6]][3,1:10],
        ylim = c(0, 1), las = 1, xlab = "PC axes (10 of 125)",
        ylab = "Cumulative proportion of variance",
        main = "")
abline(h = 0.9, lty = 2) #90% of variance explained with 4 axes (almost)

#Extract PCA scores for plotting
lyco.pca.scores <- as.data.frame(lyco.pca$x)

#To create point sizes related to time
lyco.time <- c(rep(1, 6), rep(2, 6),
               rep(3, 6), rep(5, 6), rep(10, 6), rep(30, 6),
               rep(60, 6), rep(120, 6), rep(240, 6))

#Figure 3A
plot(lyco.pca.scores[,1], lyco.pca.scores[,2],
     type = "n", xlab = "PCA 1 (48%)", ylab = "PCA 2 (23%)",
     xlim = c(-40, 40), ylim = c(-25, 25), las=1)
points(lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Cold",1],
       lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Cold",2],
       col = "grey60", lwd = 2, cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Hot",1],
       lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Hot",2],
       pch = 21, bg = "grey60", cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Cold",1],
       lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Cold",2],
```

```
       col = "grey30", lwd = 2, cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Hot",1],
       lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Hot",2],
       pch = 21, bg = "grey30", cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Acetone",1],
       lyco.pca.scores[lyco.info$Agent == "Acetone",2],
       pch = 3, lwd = 1.5, cex = 1.5)
points(lyco.pca.scores[lyco.info$Agent == "Untreated",1],
       lyco.pca.scores[lyco.info$Agent == "Untreated",2],
       pch = 4, lwd = 1.5, cex = 1.5)

#Figure 3B
plot(lyco.pca.scores[,3], lyco.pca.scores[,4],
     type = "n", xlab = "PCA 3 (12%)", ylab = "PCA 4 (6%)",
     xlim = c(-40, 20), ylim = c(-20, 20), las = 1)
points(lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Cold",3],
       lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Cold",4],
       col = "grey60", lwd = 2, cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Hot",3],
       lyco.pca.scores[lyco.info$Agent == "Acetolysis"
                       & lyco.info$Temperature == "Hot",4],
       pch = 21, bg = "grey60", cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Cold",3],
       lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Cold",4],
       col = "grey30", lwd = 2, cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Hot",3],
       lyco.pca.scores[lyco.info$Agent == "Nitric"
                       & lyco.info$Temperature == "Hot",4],
       pch = 21, bg = "grey30", cex = log(lyco.time)+1)
points(lyco.pca.scores[lyco.info$Agent == "Acetone",3],
       lyco.pca.scores[lyco.info$Agent == "Acetone",4],
       pch = 3, lwd = 1.5, cex = 1.5)
points(lyco.pca.scores[lyco.info$Agent == "Untreated",3],
       lyco.pca.scores[lyco.info$Agent == "Untreated",4],
       pch = 4, lwd = 1.5, cex = 1.5)

#PEAK HEIGHT RATIOS
#Calculate ratios
lyco.peak.ratios <- cbind(lyco.peaks[,5]/lyco.peaks[,1],
                          lyco.peaks[,2]/lyco.peaks[,4],
                          lyco.peaks[,3]/lyco.peaks[,4],
                          lyco.peaks[,2]/lyco.peaks[,3],
                          lyco.peaks[,2]/lyco.peaks[,5],
                          lyco.peaks[,5], lyco.peaks[,1])
colnames(lyco.peak.ratios) <- c("1510/3300", "2925/1600", "1710/1600",
                                "2925/1710", "2925/1510", "1510", "3300")
```

```
#Split up into four datasets
lyco.acet.cold <- rbind(lyco.peak.ratios[1:12,],
                        lyco.peak.ratios[13:66,])
lyco.acet.cold <- cbind(lyco.acet.cold,
                        c(rep(1, 6), rep(2, 6),
                          rep(3, 6), rep(4, 6), rep(5, 6), rep(6, 6),
                          rep(7, 6), rep(8, 6), rep(9, 6), rep(10, 6),
                          rep(11, 6)))
colnames(lyco.acet.cold)[8] <- "Plotting.number"

lyco.acet.hot <- rbind(lyco.peak.ratios[1:12,],
                       lyco.peak.ratios[67:120,])
lyco.acet.hot <- cbind(lyco.acet.hot,
                       c(rep(1, 6), rep(2, 6),
                         rep(3, 6), rep(4, 6), rep(5, 6), rep(6, 6),
                         rep(7, 6), rep(8, 6), rep(9, 6), rep(10, 6),
                         rep(11, 6)))
colnames(lyco.acet.hot)[8] <- "Plotting.number"

lyco.nitric.cold <- rbind(lyco.peak.ratios[1:12,],
                          lyco.peak.ratios[121:174,])
lyco.nitric.cold <- cbind(lyco.nitric.cold,
                          c(rep(1, 6), rep(2, 6),
                            rep(3, 6), rep(4, 6), rep(5, 6), rep(6, 6),
                            rep(7, 6), rep(8, 6), rep(9, 6), rep(10, 6),
                            rep(11, 6)))
colnames(lyco.nitric.cold)[8] <- "Plotting.number"

lyco.nitric.hot <- rbind(lyco.peak.ratios[1:12,],
                         lyco.peak.ratios[175:215,])
lyco.nitric.hot <- cbind(lyco.nitric.hot,
                         c(rep(1, 6), rep(2, 6),
                           rep(3, 6), rep(4, 6), rep(5, 6), rep(6, 6),
                           rep(7, 6), rep(8, 6), rep(9, 5)))
colnames(lyco.nitric.hot)[8] <- "Plotting.number"

times <- c("Untreated", "Acetone", "1", "2", "3", "5",
           "10", "30", "60", "120", "240")

#Plot UV-B peak: aromaticB/OH
#Figure 4A
boxplot(lyco.acet.cold[c(1:12),1] ~ lyco.acet.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste(Aromatic^b, "/OH")),
        main = "Acetolysis cold", las = 1)
boxplot(lyco.acet.cold[c(13:66),1] ~ lyco.acet.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Figure 4B
boxplot(lyco.acet.hot[c(1:12),1] ~ lyco.acet.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
```

```
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste(Aromatic^b, "/OH")),
        main = "Acetolysis hot", las = 1)
boxplot(lyco.acet.hot[c(13:66),1] ~ lyco.acet.hot[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Figure 4C
boxplot(lyco.nitric.cold[c(1:12),1] ~ lyco.nitric.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste(Aromatic^b, "/OH")),
        main = "Nitric cold", las = 1)
boxplot(lyco.nitric.cold[c(13:66),1] ~ lyco.nitric.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Figure 4D
boxplot(lyco.nitric.hot[c(1:12),1] ~ lyco.nitric.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste(Aromatic^b, "/OH")),
        main = "Nitric hot", las = 1)
boxplot(lyco.nitric.hot[c(13:53),1] ~ lyco.nitric.hot[13:53,8],
        at = 3:9, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:9), labels = times[1:9], tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Figure 5 plots
#Plot Aliphatic/aromaticA
boxplot(lyco.acet.cold[c(1:12),2] ~ lyco.acet.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^a)),
        main = "Acetolysis cold", las = 1)
boxplot(lyco.acet.cold[c(13:66),2] ~ lyco.acet.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.acet.hot[c(1:12),2] ~ lyco.acet.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^a)),
        main = "Acetolysis hot", las = 1)
boxplot(lyco.acet.hot[c(13:66),2] ~ lyco.acet.hot[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
```

```
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.cold[c(1:12),2] ~ lyco.nitric.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^a)),
        main = "Nitric cold", las = 1)
boxplot(lyco.nitric.cold[c(13:66),2] ~ lyco.nitric.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.hot[c(1:12),2] ~ lyco.nitric.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^a)),
        main = "Nitric hot", las = 1)
boxplot(lyco.nitric.hot[c(13:53),2] ~ lyco.nitric.hot[13:53,8],
        at = 3:9, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:9), labels = times[1:9], tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Plot Aliphatic/aromaticB
boxplot(lyco.acet.cold[c(1:12),5] ~ lyco.acet.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,6), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^b)),
        main = "Acetolysis cold", las = 1)
boxplot(lyco.acet.cold[c(13:66),5] ~ lyco.acet.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.acet.hot[c(1:12),5] ~ lyco.acet.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,6), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^b)),
        main = "Acetolysis hot", las = 1)
boxplot(lyco.acet.hot[c(13:66),5] ~ lyco.acet.hot[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.cold[c(1:12),5] ~ lyco.nitric.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,6), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^b)),
        main = "Nitric cold", las = 1)
boxplot(lyco.nitric.cold[c(13:66),5] ~ lyco.nitric.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
```

```
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.hot[c(1:12),5] ~ lyco.nitric.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,6), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Aliphatic/", aromatic^b)),
        main = "Nitric hot", las = 1)
boxplot(lyco.nitric.hot[c(13:53),5] ~ lyco.nitric.hot[13:53,8],
        at = 3:9, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:9), labels = times[1:9], tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Plot Aliphatic/carboxyl
boxplot(lyco.acet.cold[c(1:12),4] ~ lyco.acet.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,3), xlim = c(0, 11.5),
        xaxt = "n", ylab = "Aliphatic/carboxyl",
        main = "Acetolysis cold", las = 1)
boxplot(lyco.acet.cold[c(13:66),4] ~ lyco.acet.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.acet.hot[c(1:12),4] ~ lyco.acet.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,3), xlim = c(0, 11.5),
        xaxt = "n", ylab = "Aliphatic/carboxyl",
        main = "Acetolysis hot", las = 1)
boxplot(lyco.acet.hot[c(13:66),4] ~ lyco.acet.hot[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.cold[c(1:12),4] ~ lyco.nitric.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,3), xlim = c(0, 11.5),
        xaxt = "n", ylab = "Aliphatic/carboxyl",
        main = "Nitric cold", las = 1)
boxplot(lyco.nitric.cold[c(13:66),4] ~ lyco.nitric.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.hot[c(1:12),4] ~ lyco.nitric.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,3), xlim = c(0, 11.5),
        xaxt = "n", ylab = "Aliphatic/carboxyl",
        main = "Nitric hot", las = 1)
boxplot(lyco.nitric.hot[c(13:53),4] ~ lyco.nitric.hot[13:53,8],
        at = 3:9, boxwex = 0.8,
```

```
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:9), labels = times[1:9], tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#Plot Carboxyl/aromaticA
boxplot(lyco.acet.cold[c(1:12),3] ~ lyco.acet.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Carboxyl/", aromatic^a)),
        main = "Acetolysis cold", las = 1)
boxplot(lyco.acet.cold[c(13:66),3] ~ lyco.acet.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.acet.hot[c(1:12),3] ~ lyco.acet.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Carboxyl/", aromatic^a)),
        main = "Acetolysis hot", las = 1)
boxplot(lyco.acet.hot[c(13:66),3] ~ lyco.acet.hot[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.cold[c(1:12),3] ~ lyco.nitric.cold[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Carboxyl/", aromatic^a)),
        main = "Nitric cold", las = 1)
boxplot(lyco.nitric.cold[c(13:66),3] ~ lyco.nitric.cold[13:66,8],
        at = 3:11, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:11), labels = times, tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

boxplot(lyco.nitric.hot[c(1:12),3] ~ lyco.nitric.hot[1:12,8],
        at = 1:2 - 0.5, boxwex = 0.8,
        ylim = c(0,5), xlim = c(0, 11.5),
        xaxt = "n", ylab = expression(paste("Carboxyl/", aromatic^a)),
        main = "Nitric hot", las = 1)
boxplot(lyco.nitric.hot[c(13:53),3] ~ lyco.nitric.hot[13:53,8],
        at = 3:9, boxwex = 0.8,
        xaxt = "n", yaxt = "n", add = T)
axis(1, at = c(1:2 - 0.5, 3:9), labels = times[1:9], tick = TRUE)
abline(v = 2.25, lty = 2, col = "grey40")

#ANGIOSPERM POLLEN
angio.spectra <- read.table (file = "Table S4.txt",
                             header=TRUE, row.names=1)
angio.peaks <- read.table (file = "Table S5.txt",
                           header=TRUE, row.names=1)
angio.info <- read.table (file = "Table S6.txt",
```

```
                          header=TRUE, row.names=1)

#Transpose spectra to put samples in rows and wavenumbers in columns
angio.spectra <- t(angio.spectra)

#Each sample standardised to mean = 0, variance = 1
angio.spectra.mean <- apply(angio.spectra, 1, mean)
angio.spectra.minusMean <- apply(angio.spectra, 2, function(x)
  x - angio.spectra.mean)

angio.spectra.minusMean.standev <- apply(angio.spectra.minusMean,
                                         1, sd)
angio.spectra.standardised <- apply(angio.spectra.minusMean, 2,
                                    function(x)
                                      x / angio.spectra.minusMean.standev)

#Difference plots
#Figure 6
plot(wavenumbers, seq(from = 1, to = 45, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 45, 9),
     yaxt = "n",
     las = 1, ylab = "Absorbance, relative to untreated sample",
     xlab = "Wavenumber",
     main = "Angiosperm difference plots")

abline(h = 5, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[1,]-
angio.spectra.standardised[2,])+5,
      type = "l", lwd = 2)

abline(h = 10, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[3,]-
angio.spectra.standardised[4,])+10,
      type = "l", lwd = 2)

abline(h = 15, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[5,]-
angio.spectra.standardised[6,])+15,
      type = "l", lwd = 2)

abline(h = 20, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[7,]-
angio.spectra.standardised[8,])+20,
      type = "l", lwd = 2)

abline(h = 25, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[9,]-
angio.spectra.standardised[10,])+25,
      type = "l", lwd = 2)

abline(h = 30, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[11,]-
angio.spectra.standardised[12,])+30,
      type = "l", lwd = 2)
```

```
abline(h = 35, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[13,]-
angio.spectra.standardised[14,])+35,
      type = "l", lwd = 2)

abline(h = 40, lty = 2, col = "grey40")
lines(wavenumbers, (angio.spectra.standardised[15,]-
angio.spectra.standardised[16,])+40,
      type = "l", lwd = 2)

sample.names <- c("A.trid", "B.font", "I.xant", "J.nigr", "K.scop", "P.trem",
                  "S.cera", "S.hale")
text(4000, c(7, 12, 17, 22, 27, 32, 37, 42), sample.names, pos = 4)

#Peak height ratios
#Calculate ratios
angio.peak.ratios <- cbind(angio.peaks[,5]/angio.peaks[,1],
                           angio.peaks[,2]/angio.peaks[,4],
                           angio.peaks[,3]/angio.peaks[,4],
                           angio.peaks[,2]/angio.peaks[,3],
                           angio.peaks[,2]/angio.peaks[,5],
                           angio.peaks[,5], angio.peaks[,1])
colnames(angio.peak.ratios) <- c("1510/3300", "2925/1600", "1710/1600",
                                 "2925/1710", "2925/1510", "1510", "3300")

#Figure 7 plots
#AromaticB/OH
plot(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Untreated",1],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",1],
     xlim = c(0,2), ylim = c(0,2), type = "n", las = 1,
     xlab = "Untreated", ylab = "Acetolysis, hot, 10 minutes", main =
"1510/3300")
abline(a = 0, b = 1, col = "grey40", lty = 2)
points(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Untreated",1],
       angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Acetolysis",1],
       cex = 1.5)
text(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Untreated",1],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",1],
     labels = taxon.names, pos = 4)
points(lyco.untreated.mean[1], lyco.acet.mean[1], pch = 15, cex = 1.5)
lines(c(lyco.untreated.mean[1] - lyco.untreated.stdev[1],
        lyco.untreated.mean[1] + lyco.untreated.stdev[1]),
      c(lyco.acet.mean[1], lyco.acet.mean[1]))
lines(c(lyco.untreated.mean[1], lyco.untreated.mean[1]),
      c(lyco.acet.mean[1] - lyco.acet.stdev[1],
        lyco.acet.mean[1] + lyco.acet.stdev[1]))

#Aliphatic/aromaticA
plot(angio.peak.ratios[angio.info$Group == "Angiosperm" &
```

```
                        angio.info$Agent == "Untreated",2],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Acetolysis",2],
     xlim = c(0,4), ylim = c(0,4), type = "n", las = 1,
     xlab = "Untreated", ylab = "Acetolysis, hot, 10 minutes", main =
"2925/1600")
abline(a = 0, b = 1, col = "grey40", lty = 2)
points(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                          angio.info$Agent == "Untreated",2],
       angio.peak.ratios[angio.info$Group == "Angiosperm" &
                          angio.info$Agent == "Acetolysis",2],
       cex = 1.5)
text(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Untreated",2],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Acetolysis",2],
     labels = taxon.names, pos = 4)
points(lyco.untreated.mean[2], lyco.acet.mean[2], pch = 15, cex = 1.5)
lines(c(lyco.untreated.mean[2] - lyco.untreated.stdev[2],
        lyco.untreated.mean[2] + lyco.untreated.stdev[2]),
      c(lyco.acet.mean[2], lyco.acet.mean[2]))
lines(c(lyco.untreated.mean[2], lyco.untreated.mean[2]),
      c(lyco.acet.mean[2] - lyco.acet.stdev[2],
        lyco.acet.mean[2] + lyco.acet.stdev[2]))

#Aliphatic/aromaticB
plot(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Untreated",5],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Acetolysis",5],
     xlim = c(0,4), ylim = c(0,4), type = "n", las = 1,
     xlab = "Untreated", ylab = "Acetolysis, hot, 10 minutes", main =
"2925/1510")
abline(a = 0, b = 1, col = "grey40", lty = 2)
points(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                          angio.info$Agent == "Untreated",5],
       angio.peak.ratios[angio.info$Group == "Angiosperm" &
                          angio.info$Agent == "Acetolysis",5],
       cex = 1.5)
text(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Untreated",5],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Acetolysis",5],
     labels = taxon.names, pos = 4)
points(lyco.untreated.mean[5], lyco.acet.mean[5], pch = 15, cex = 1.5)
lines(c(lyco.untreated.mean[5] - lyco.untreated.stdev[5],
        lyco.untreated.mean[5] + lyco.untreated.stdev[5]),
      c(lyco.acet.mean[5], lyco.acet.mean[5]))
lines(c(lyco.untreated.mean[5], lyco.untreated.mean[5]),
      c(lyco.acet.mean[5] - lyco.acet.stdev[5],
        lyco.acet.mean[5] + lyco.acet.stdev[5]))

#Aliphatic/carboxyl
plot(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                        angio.info$Agent == "Untreated",4],
```

```
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",4],
     xlim = c(0,8), ylim = c(0,8), type = "n", las = 1,
     xlab = "Untreated", ylab = "Acetolysis, hot, 10 minutes", main =
"2925/1710")
abline(a = 0, b = 1, col = "grey40", lty = 2)
points(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Untreated",4],
       angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Acetolysis",4],
       cex = 1.5)
text(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Untreated",4],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",4],
     labels = taxon.names, pos = 4)
points(lyco.untreated.mean[4], lyco.acet.mean[4], pch = 15, cex = 1.5)
lines(c(lyco.untreated.mean[4] - lyco.untreated.stdev[4],
        lyco.untreated.mean[4] + lyco.untreated.stdev[4]),
      c(lyco.acet.mean[4], lyco.acet.mean[4]))
lines(c(lyco.untreated.mean[4], lyco.untreated.mean[4]),
      c(lyco.acet.mean[4] - lyco.acet.stdev[4],
        lyco.acet.mean[4] + lyco.acet.stdev[4]))

#Carboxyl/aromaticA
plot(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Untreated",3],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",3],
     xlim = c(0,4), ylim = c(0,4), type = "n", las = 1,
     xlab = "Untreated", ylab = "Acetolysis, hot, 10 minutes", main =
"1710/1600")
abline(a = 0, b = 1, col = "grey40", lty = 2)
points(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Untreated",3],
       angio.peak.ratios[angio.info$Group == "Angiosperm" &
                           angio.info$Agent == "Acetolysis",3],
       cex = 1.5)
text(angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Untreated",3],
     angio.peak.ratios[angio.info$Group == "Angiosperm" &
                         angio.info$Agent == "Acetolysis",3],
     labels = taxon.names, pos = 4)
points(lyco.untreated.mean[3], lyco.acet.mean[3], pch = 15, cex = 1.5)
lines(c(lyco.untreated.mean[3] - lyco.untreated.stdev[3],
        lyco.untreated.mean[3] + lyco.untreated.stdev[3]),
      c(lyco.acet.mean[3], lyco.acet.mean[3]))
lines(c(lyco.untreated.mean[3], lyco.untreated.mean[3]),
      c(lyco.acet.mean[3] - lyco.acet.stdev[3],
        lyco.acet.mean[3] + lyco.acet.stdev[3]))

#PCA
angio.pca <- prcomp(angio.spectra.standardised)
summary(angio.pca)
barplot(summary(angio.pca)[[6]][3,1:10],
```

```
        ylim = c(0, 1), las = 1, xlab = "PC axes (10 of 28)",
        ylab = "Cumulative proportion of variance",
        main = "")
abline(h = 0.9, lty = 2)

#Extract PCA sample scores for plotting
angio.pca.scores <- as.data.frame(angio.pca$x)

#Figure 8
plot(angio.pca.scores[,1], angio.pca.scores[,2],
     type = "n", xlab = "PCA 1", ylab = "PCA 2",
     las=1, ylim = c(-30, 30), xlim = c(-50, 50))
points(angio.pca.scores[angio.info$Agent == "Untreated"
                        & angio.info$Group == "Angiosperm",1],
       angio.pca.scores[angio.info$Agent == "Untreated"
                        & angio.info$Group == "Angiosperm",2],
       cex = 1.5, lwd = 2)
points(angio.pca.scores[angio.info$Agent == "Untreated"
                        & angio.info$Group == "Lycopodium",1],
       angio.pca.scores[angio.info$Agent == "Untreated"
                        & angio.info$Group == "Lycopodium",2],
       pch = 2, cex = 1.5, lwd = 2)
points(angio.pca.scores[angio.info$Agent == "Acetolysis"
                        & angio.info$Group == "Angiosperm",1],
       angio.pca.scores[angio.info$Agent == "Acetolysis"
                        & angio.info$Group == "Angiosperm",2],
       pch = 19, cex = 1.5)
points(angio.pca.scores[angio.info$Agent == "Acetolysis"
                        & angio.info$Group == "Lycopodium",1],
       angio.pca.scores[angio.info$Agent == "Acetolysis"
                        & angio.info$Group == "Lycopodium",2],
       pch = 17, cex = 1.5)

text(angio.pca.scores[angio.info$Agent == "Untreated"
                      & angio.info$Group == "Angiosperm",1],
     angio.pca.scores[angio.info$Agent == "Untreated"
                      & angio.info$Group == "Angiosperm",2],
     labels = taxon.names, pos = 4)
text(angio.pca.scores[angio.info$Agent == "Acetolysis"
                      & angio.info$Group == "Angiosperm",1],
     angio.pca.scores[angio.info$Agent == "Acetolysis"
                      & angio.info$Group == "Angiosperm",2],
     labels = taxon.names, pos = 4)

#SUPPLEMENTARY FIGURES
#Figure S1
sample.names <- c("Untreated", "1 min", "2 mins", "3 mins", "5 mins", "10
mins",
                  "30 mins", "60 mins", "120 mins", "240 mins")
par(mfrow = c(1, 4))
#Acetolysis cold (untreated at base)
plot(wavenumbers, seq(from = 1, to = 55, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 55, 11),
     yaxt = "n",
```

```
     las = 1, ylab = "Absorbance", xlab = expression(paste("Wavenumber ", (cm^-
1))),
     main = "Acetolysis, cold")
lines(wavenumbers, lyco.codeMeans[1,]+5,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[3,]+10,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[4,]+15,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[5,]+20,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[6,]+25,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[7,]+30,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[8,]+35,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[9,]+40,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[10,]+45,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[11,]+50,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46, 51), sample.names, pos = 4)

#Acetolysis hot (untreated at base)
plot(wavenumbers, seq(from = 1, to = 55, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 55, 11),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "Acetolysis, hot")
lines(wavenumbers, lyco.codeMeans[1,]+5,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[12,]+10,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[13,]+15,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[14,]+20,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[15,]+25,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[16,]+30,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[17,]+35,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[18,]+40,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[19,]+45,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[20,]+50,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46, 51), sample.names, pos = 4)

#Nitric cold (untreated at base)
plot(wavenumbers, seq(from = 1, to = 55, length.out = length(wavenumbers)),
```

```
     type = "n", xlim = c(4000, 450), yaxp = c(0, 55, 11),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "Nitric, cold")
lines(wavenumbers, lyco.codeMeans[1,]+5,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[21,]+10,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[22,]+15,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[23,]+20,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[24,]+25,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[25,]+30,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[26,]+35,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[27,]+40,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[28,]+45,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[29,]+50,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41, 46, 51), sample.names, pos = 4)

#Nitric hot (untreated at base)
plot(wavenumbers, seq(from = 1, to = 55, length.out = length(wavenumbers)),
     type = "n", xlim = c(4000, 450), yaxp = c(0, 55, 11),
     yaxt = "n",
     las = 1, ylab = "", xlab = expression(paste("Wavenumber ", (cm^-1))),
     main = "Nitric, hot")
lines(wavenumbers, lyco.codeMeans[1,]+5,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[30,]+10,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[31,]+15,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[32,]+20,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[33,]+25,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[34,]+30,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[35,]+35,
      type = "l", lwd = 2)
lines(wavenumbers, lyco.codeMeans[36,]+40,
      type = "l", lwd = 2)
text(4000, c(6, 11, 16, 21, 26, 31, 36, 41), sample.names[1:8], pos = 4)

#Figure S2
par(mfrow = c(4,1))
plot(wavenumbers, lyco.pca$rotation[,1],
     type = "n", xlim = c(4000, 450),
     ylab = "PCA1 loadings",  xlab = "", las = 1)
```

```
abline(h = 0, lty = 2, col = "grey60")
lines(wavenumbers, lyco.pca$rotation[,1])

plot(wavenumbers, lyco.pca$rotation[,2],
     type = "n", xlim = c(4000, 450),
     ylab = "PCA2 loadings",  xlab = "", las = 1)
abline(h = 0, lty = 2, col = "grey60")
lines(wavenumbers, lyco.pca$rotation[,2])

plot(wavenumbers, lyco.pca$rotation[,3],
     type = "n", xlim = c(4000, 450),
     ylab = "PCA3 loadings",  xlab = "", las = 1)
abline(h = 0, lty = 2, col = "grey60")
lines(wavenumbers, lyco.pca$rotation[,3])

plot(wavenumbers, lyco.pca$rotation[,4],
     type = "n", xlim = c(4000, 450),
     ylab = "PCA4 loadings",  xlab = "Wavenumber (cm-1)", las = 1)
abline(h = 0, lty = 2, col = "grey60")
lines(wavenumbers, lyco.pca$rotation[,4])
```