CoralProject

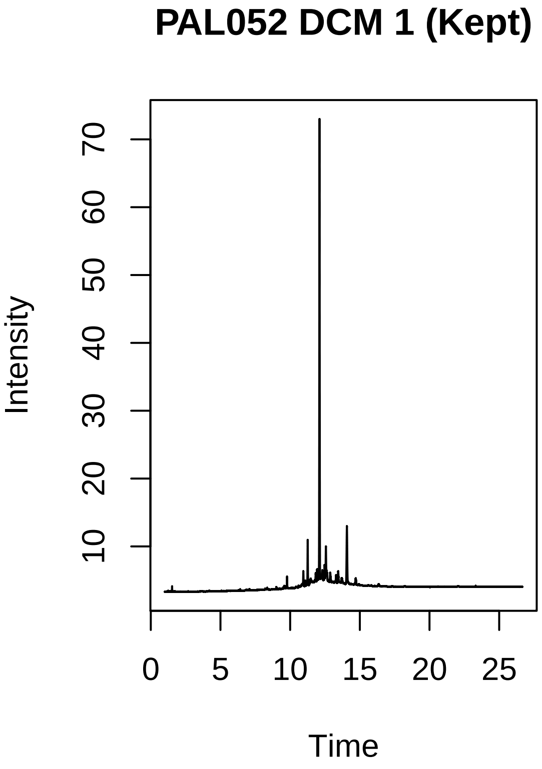
Jason Chari and Elizabeth Maloney

June 15, 2016

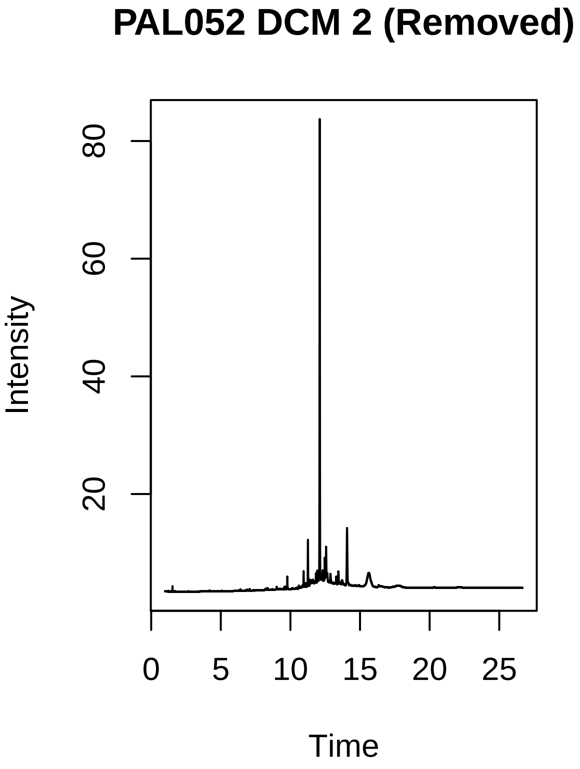
Data Entry

Data Notes:

- Readings occur at delta = 0.000333333333333 seconds apart
- Several of the Trocheliophorum samples were removed due to duplication or obvious corruption. See below for details:
  - Remark that Trocheliophorum is not the main focus of this research and these samples are used mainly as a reference for the S. Glaucum clades.
- Two data files found for sample PAL052 (DCM) - discarded 2nd sample (file PAL052D2.csv)
  - Incongruent with other DCM samples
  - Apparently identical to 2nd sample found for sample PAL250 (DCM)
  - Broad peak @ 15.8 looked less like rest of data.



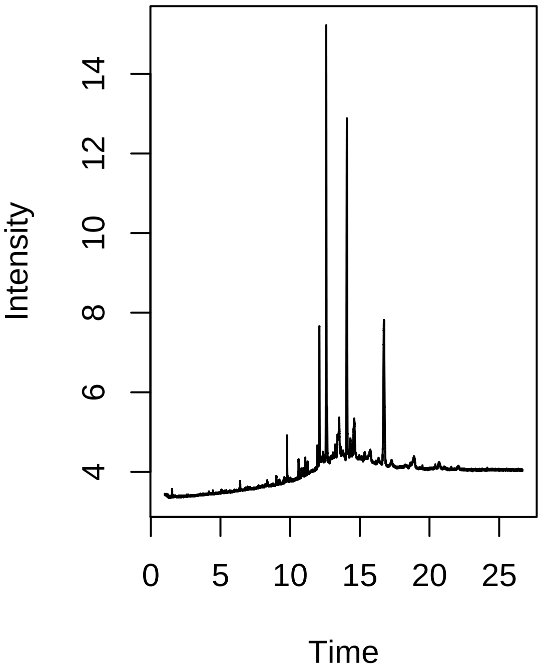
PAL052 DCM 1 (Kept)



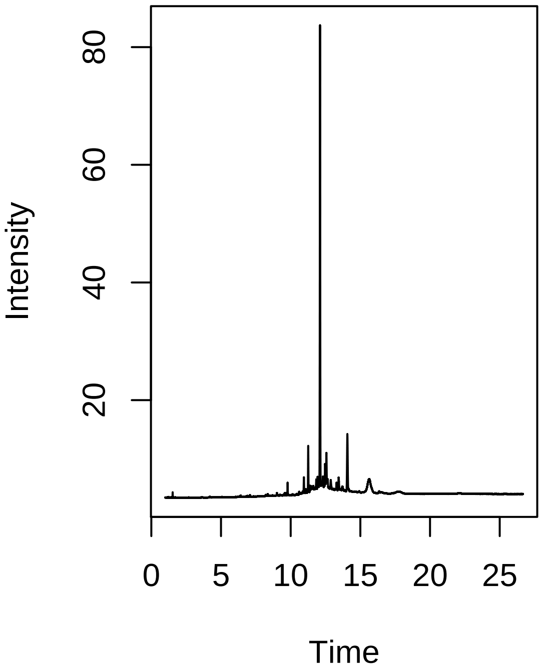
PAL052 DCM 2 (Removed)

- Two data files found for sample PAL250 (DCM) - discarded 2nd sample (file PAL250D2.csv)
    - Incongruent with other DCM samples
    - Apparently identical to 2nd sample found for sample PAL052 (DCM)

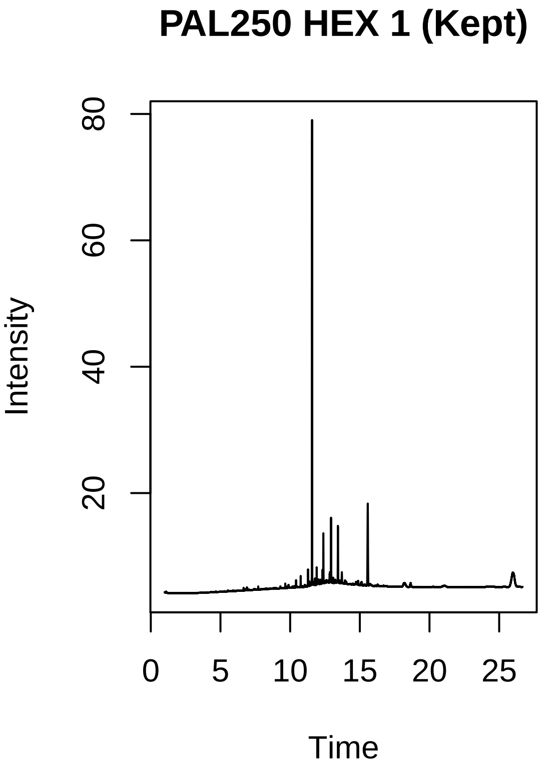
PAL250 DCM 1 (Kept)



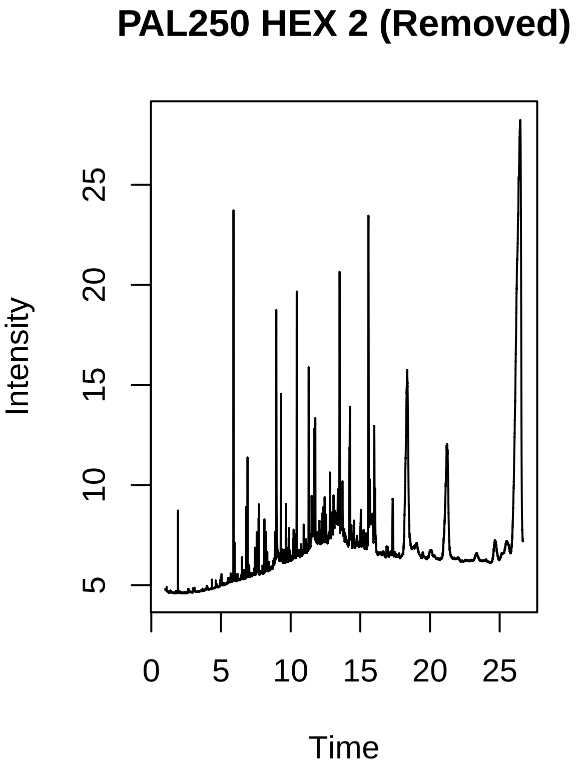
PAL250 DCM 2 (Removed)



- Two data files found for sample PAL250(HEX) - discarded 2nd sample (file PAL250H2.CSV)
  - Incongruent with other HEX Trocheliophorum samples
  - Choosing sample 2 is consistent with earlier choices where 2 data files found

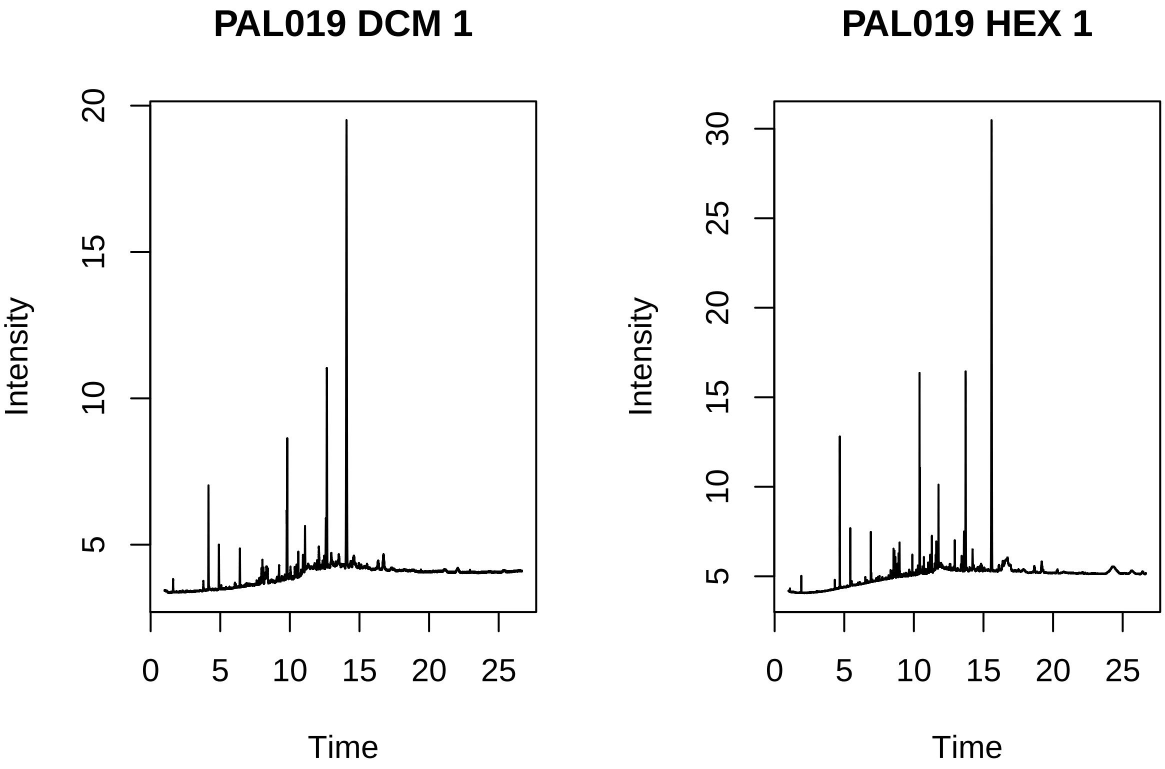


PAL250 HEX 1 (Kept)

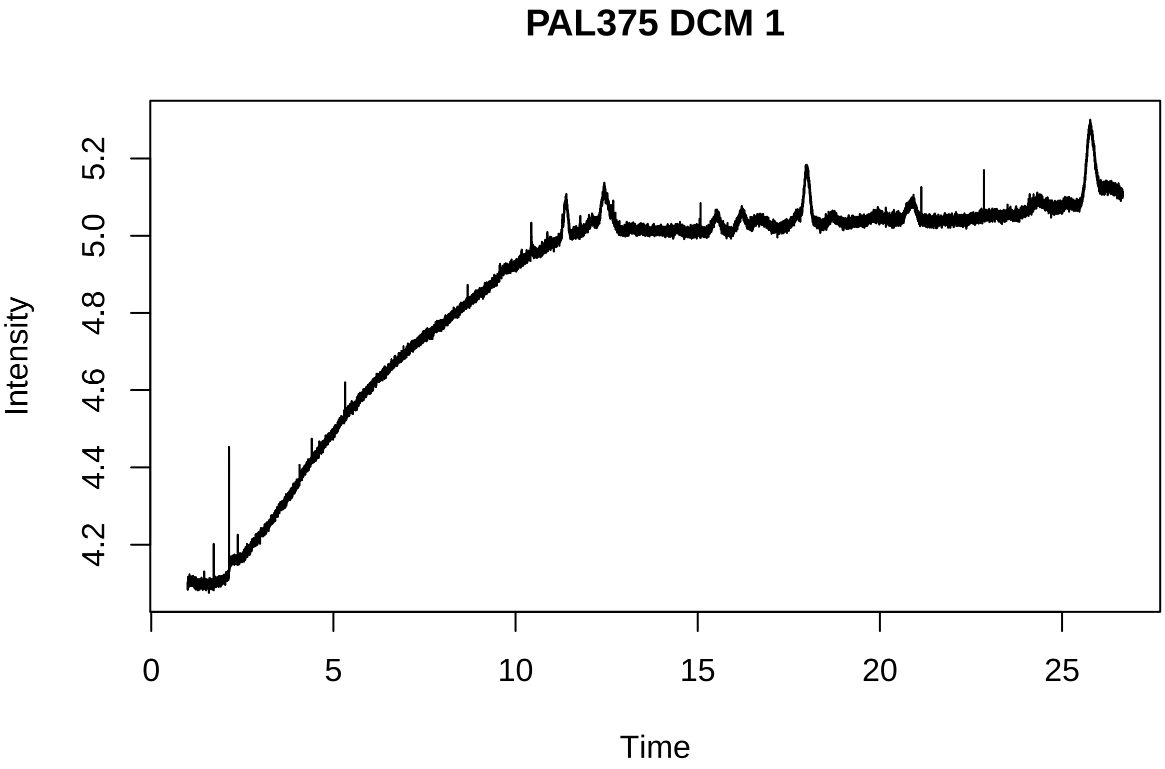


PAL250 HEX 2 (Removed)

- Trocheliophorum Comparison Plot: PAL019 DCM and HEX



- Chromatogram of sample PAL375 (DCM) indicates that nothing was injected and was discarded (file PAL375D1.csv)



- Some readings originally had 77,002 time points while others had 77,001
  - We omit the last time point and align based on the initial time which begin at approximately 1 minute
    - We observe that the standard deviation is greater among the final time points than among either the initial time points or the 77001th time points, and conclude that the samples start at approximately the same time, but some extend for an extra time point.

```
sd(initialDCMTimes) #standard deviation of initial time point for each sample
```

```
## [1] 9.001046e-05
```

```
sd(DCMTime77001) #standard deviation of 77001st time point for each sample
```

```
## [1] 9.001046e-05
```

```
sd(finalDCMTimes) #standard deviation of final time point for each sample
```

```
## [1] 0.000110811
```

Number of DCM ("medium polarity") Samples of Each Type:

Species	Count
S.glaucum Clade F	24
S.glaucum Clade D	10
Trocheliophorum	11
Total	45

Number of Hexane ("nonpolar") Samples of Each Type:

Species	Count
S.glaucum Clade F	18
S.glaucum Clade D	8
Trocheliophorum	7
Total	33

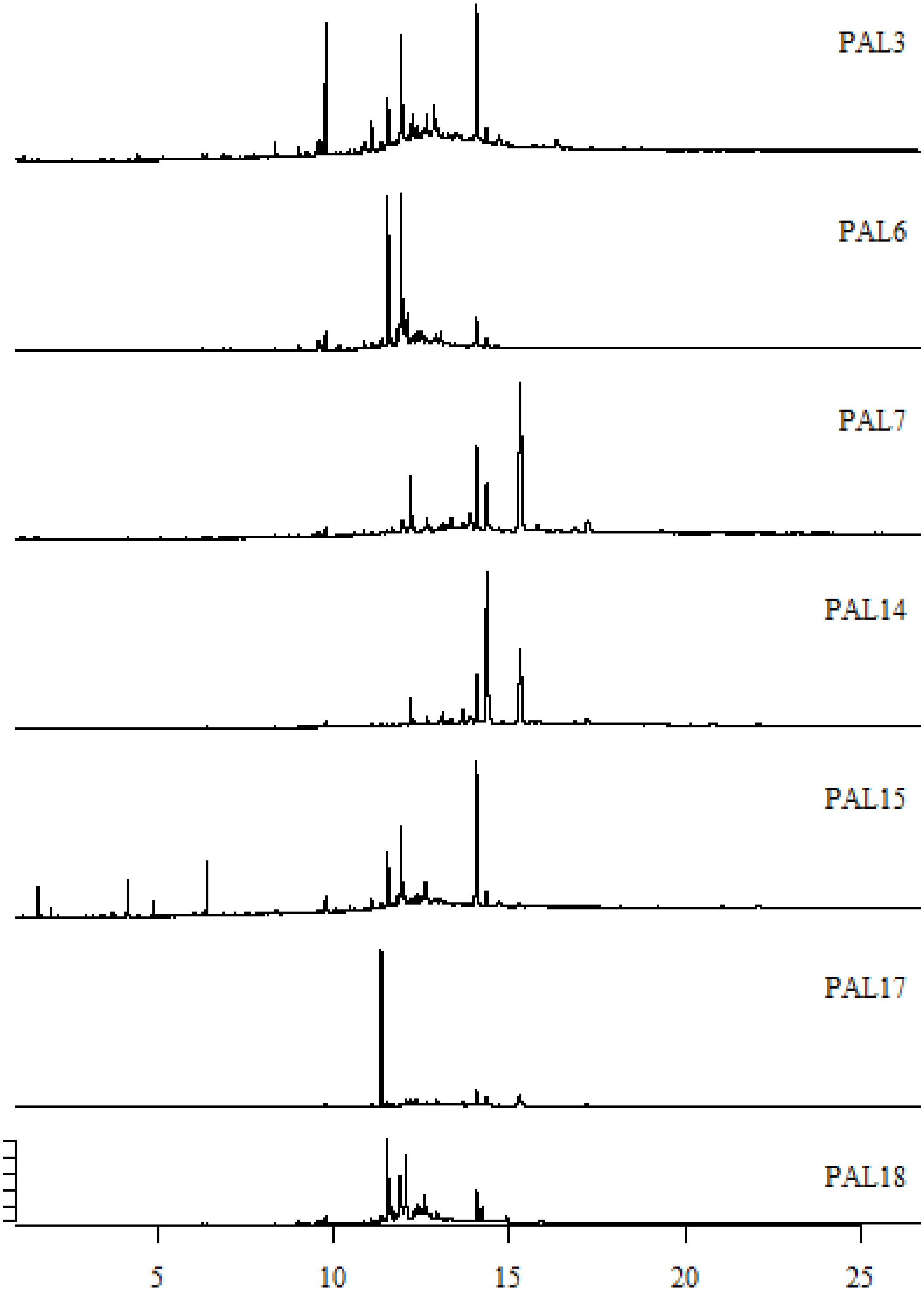
Table 3: Snapshot of All Time Points Dataframe (DCM)

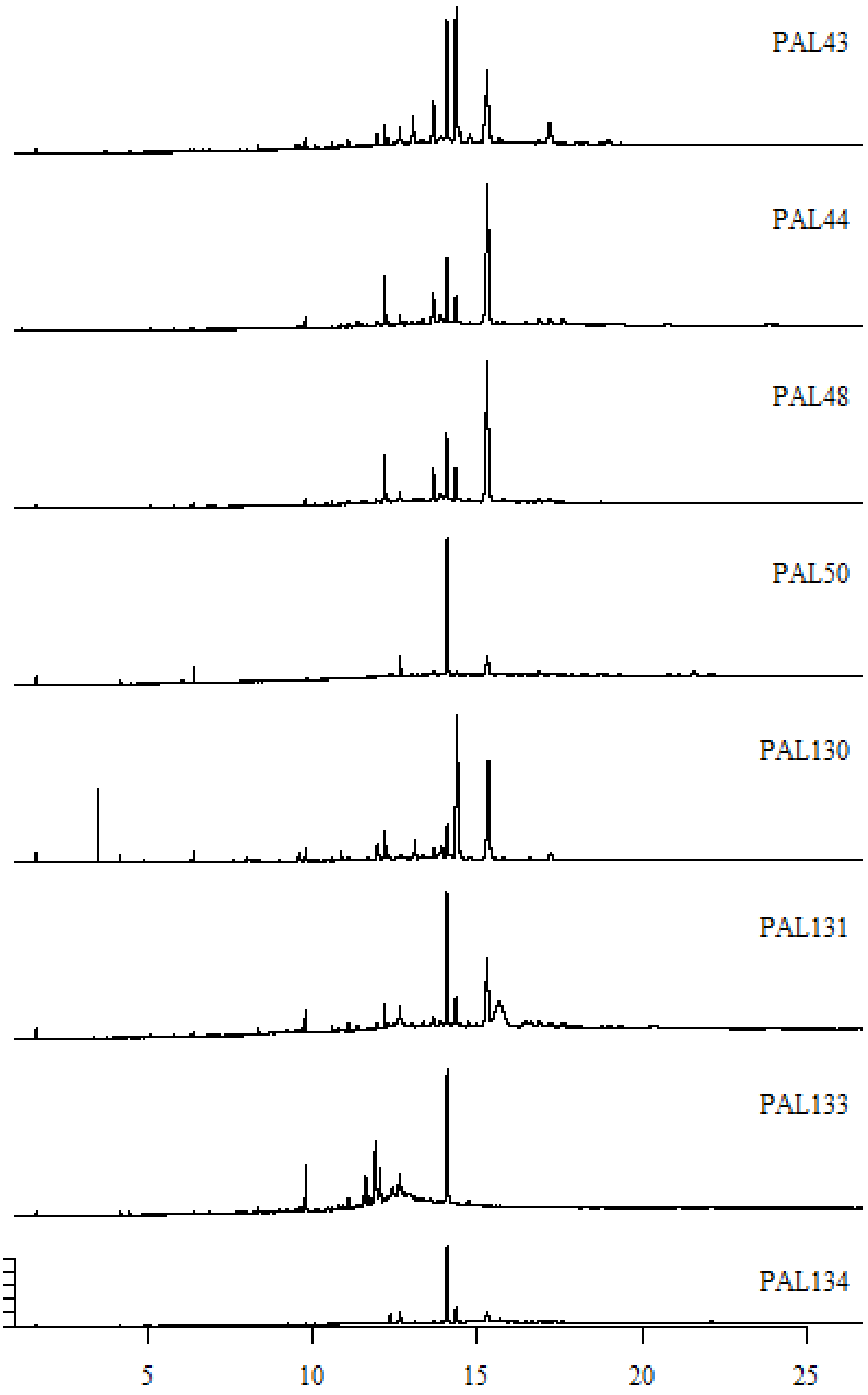
Sample	Greasiness	Clade	t1	t2
3	DCM	F	4.05859396167216	4.06250021187589
6	DCM	F	3.93515645523439	3.93190124673129
7	DCM	F	4.01484395939042	4.0098960424657
14	DCM	F	3.90481791198545	3.90065124510147
15	DCM	F	3.90716166210768	3.9088543705293

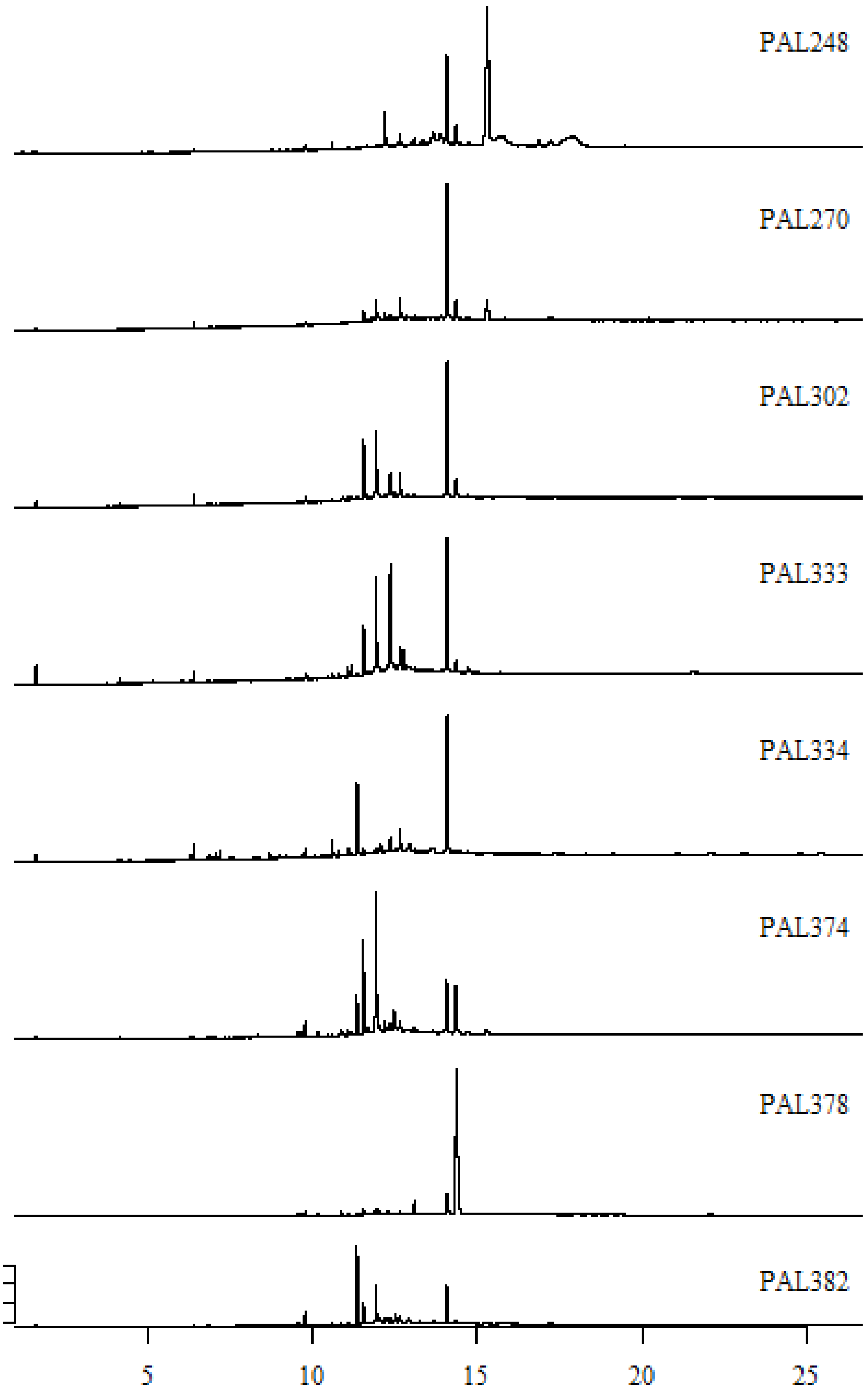
Table 4: Snapshot of All Time Points Dataframe (HEX)

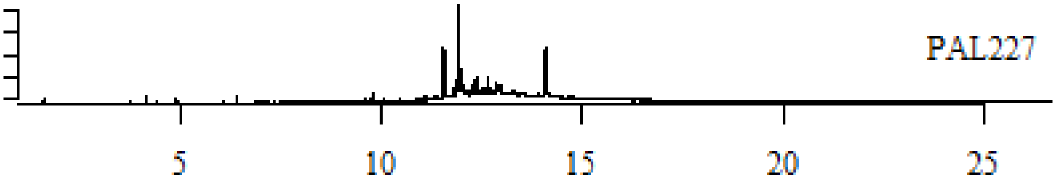
Sample	Greasiness	Clade	t1	t2
1	HEX	F	4.82070337641926	4.81888045965752
3	HEX	F	4.79622420847591	4.7972658751969
7	HEX	F	4.80494816726423	4.79661483349628
14	HEX	F	4.82669296006497	4.82916691852734
15	HEX	F	4.73854191380087	4.73411483023665

Gas Chromatograms - Clade F (DCM)

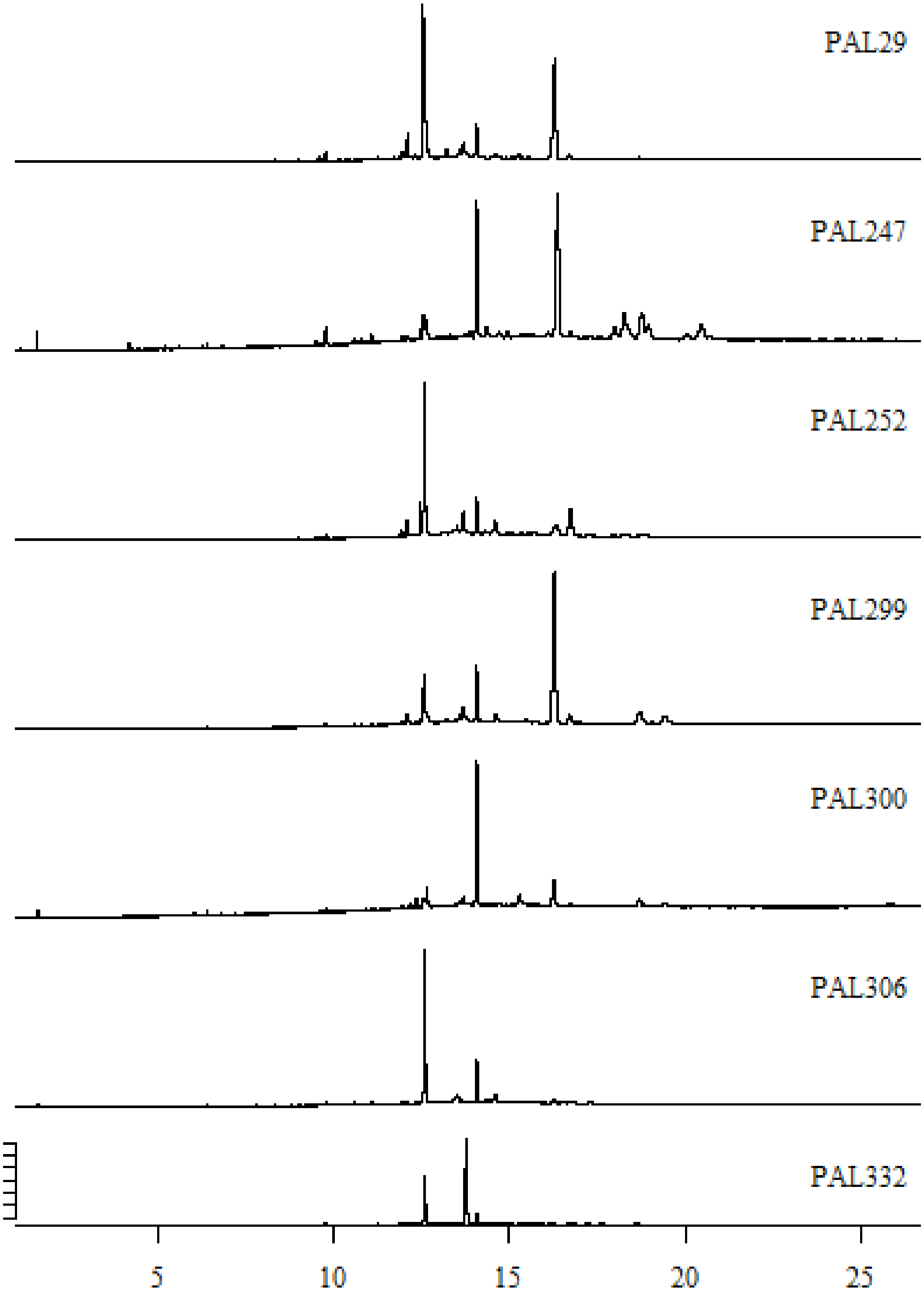


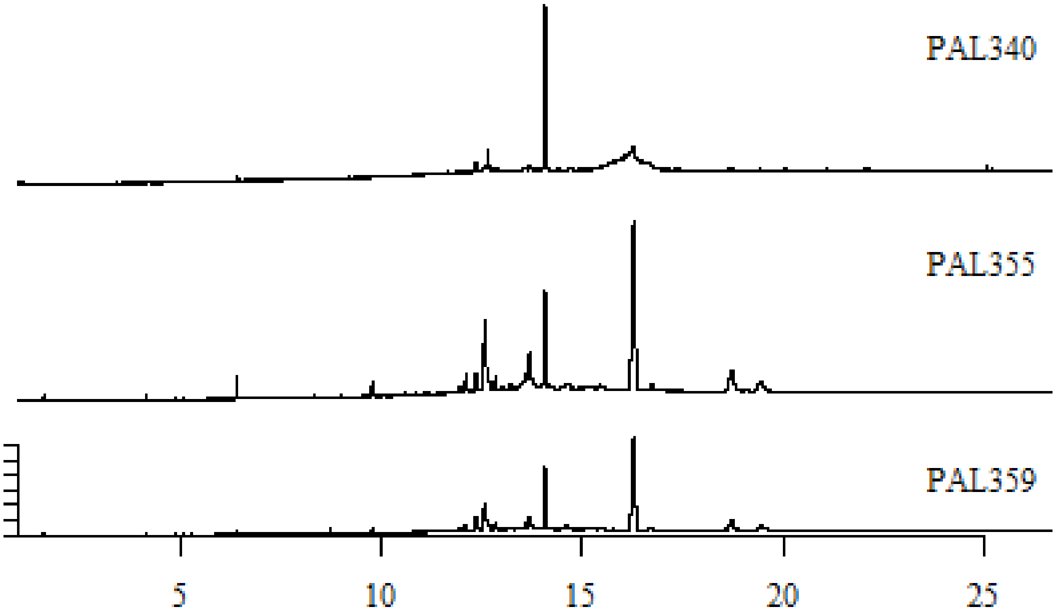

PAL43
PAL44
PAL48
PAL50
PAL130
PAL131
PAL133
PAL134
5
10
15
20
25


PAL248
PAL270
PAL302
PAL333
PAL334
PAL374
PAL378
PAL382
5
10
15
20
25

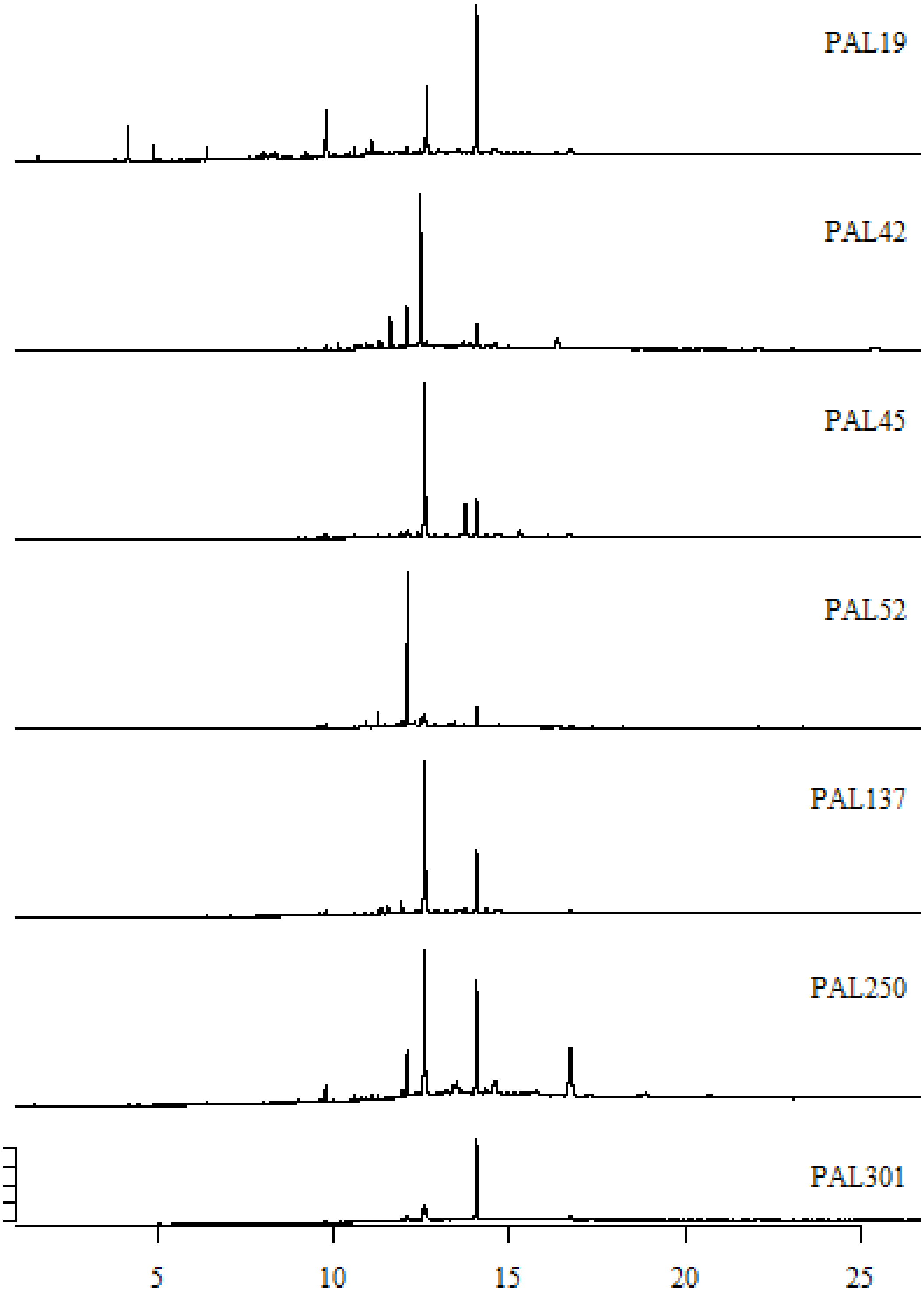

PAL227
5
10
15
20
25

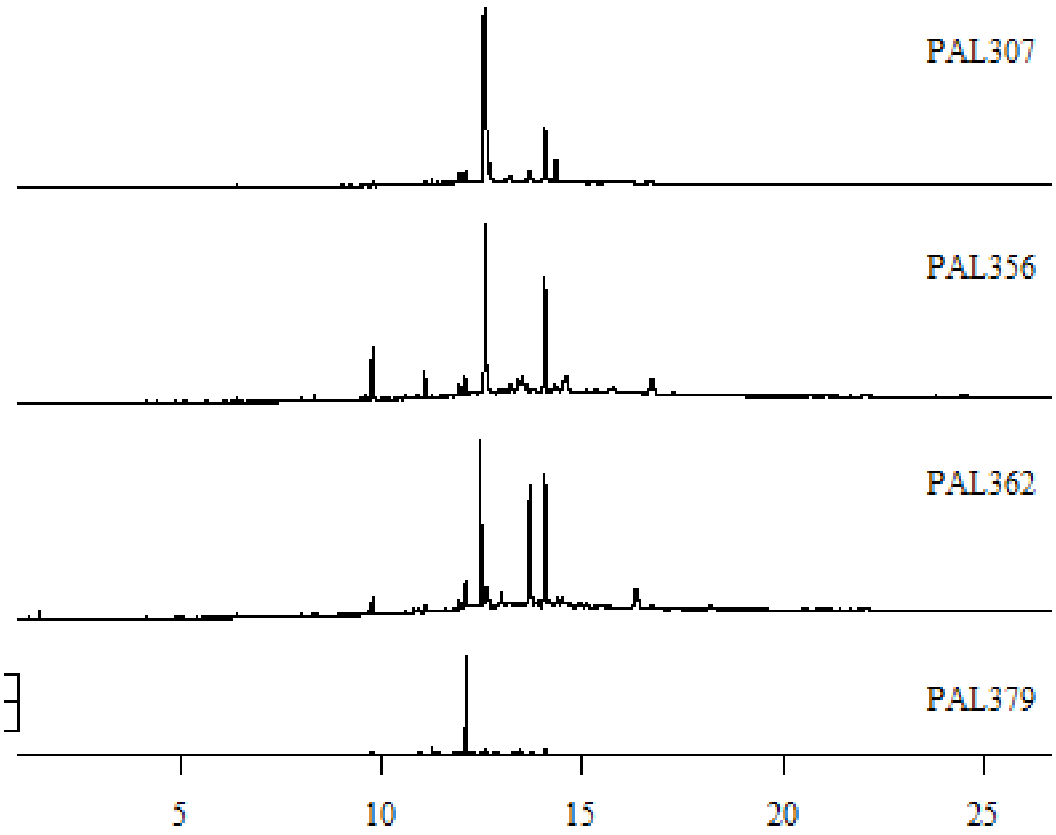
Gas Chromatograms - Clade D (DCM)



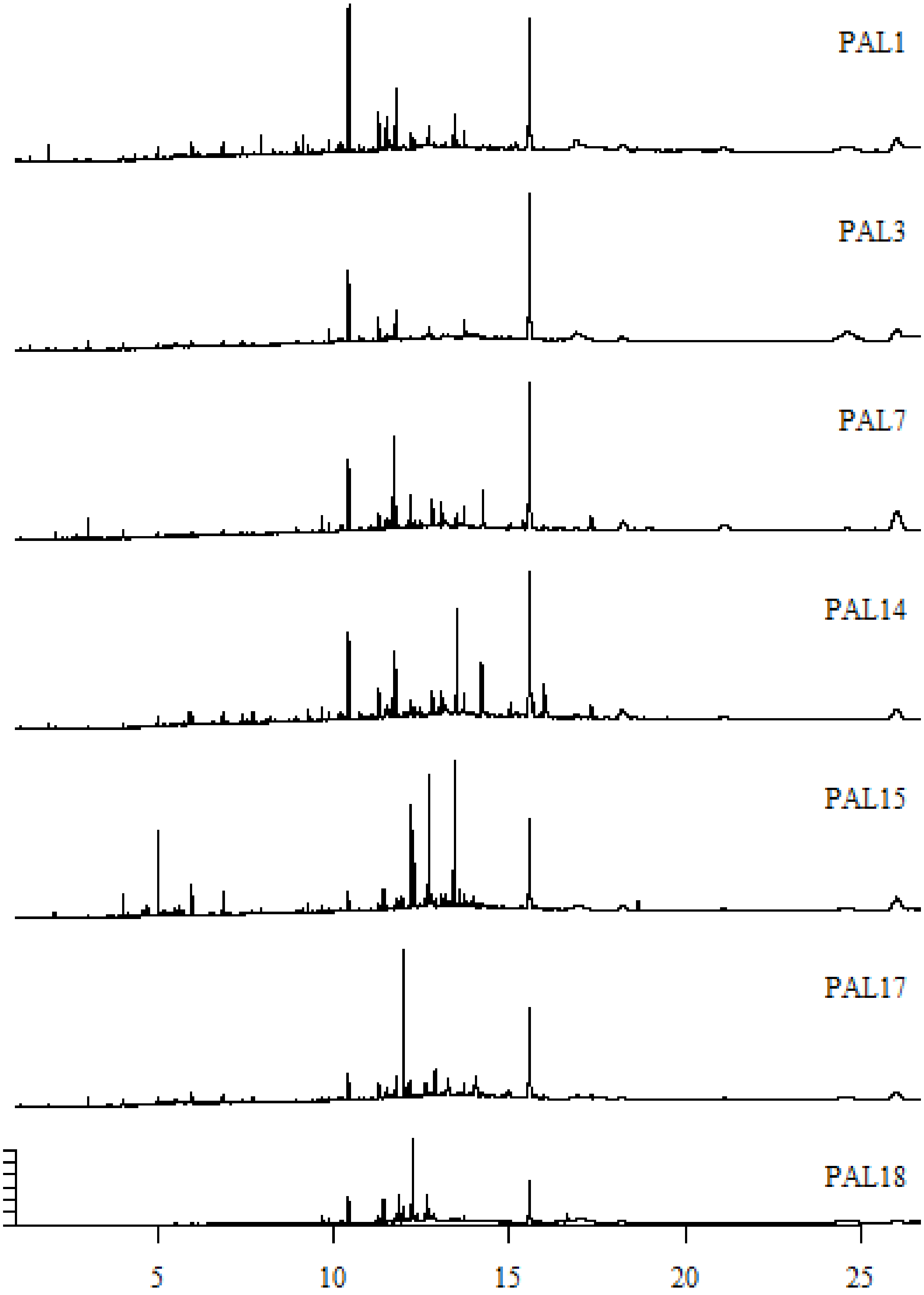

PAL340
PAL355
PAL359
5
10
15
20
25

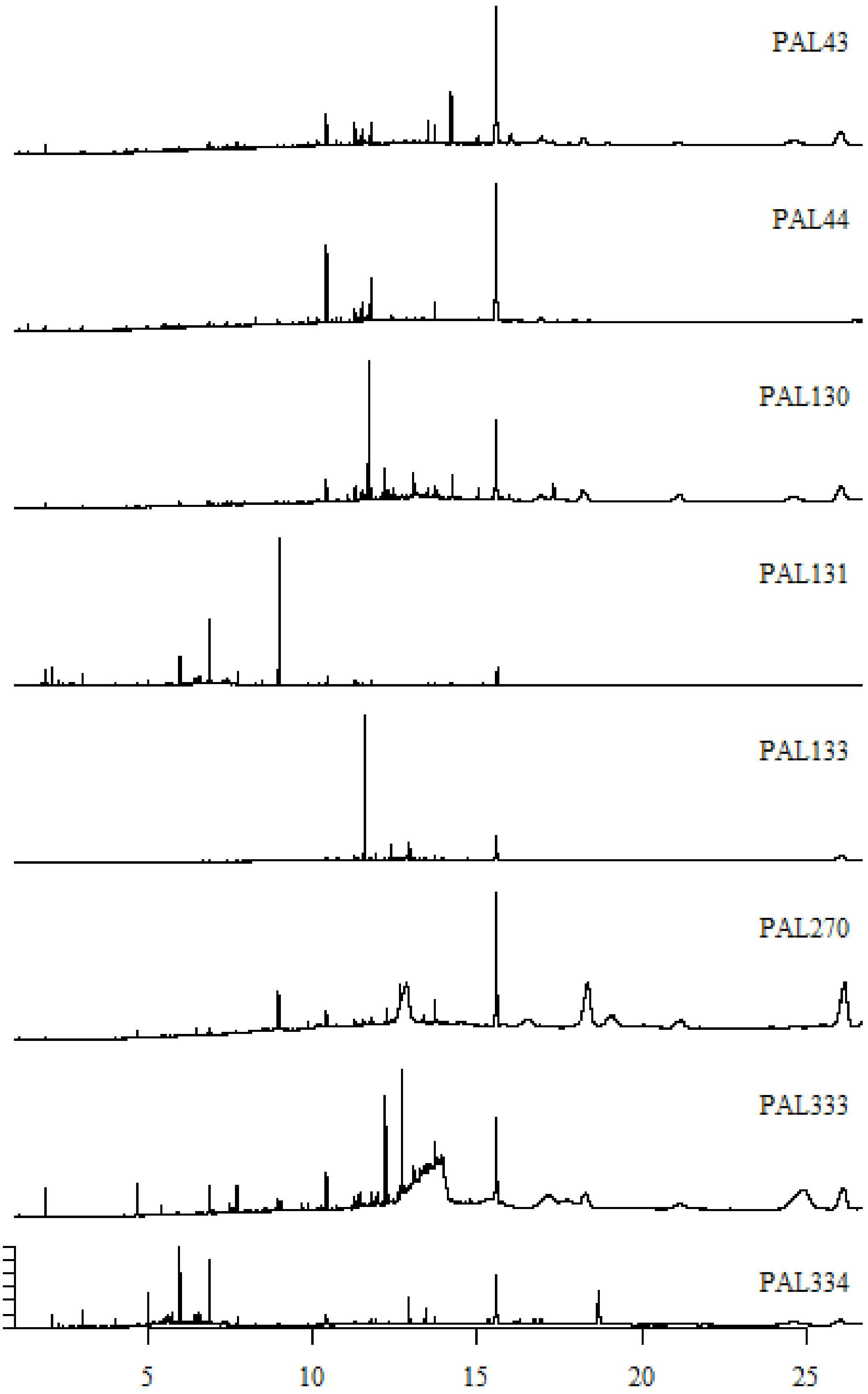
Gas Chromatograms - Trocheliophorum (DCM)

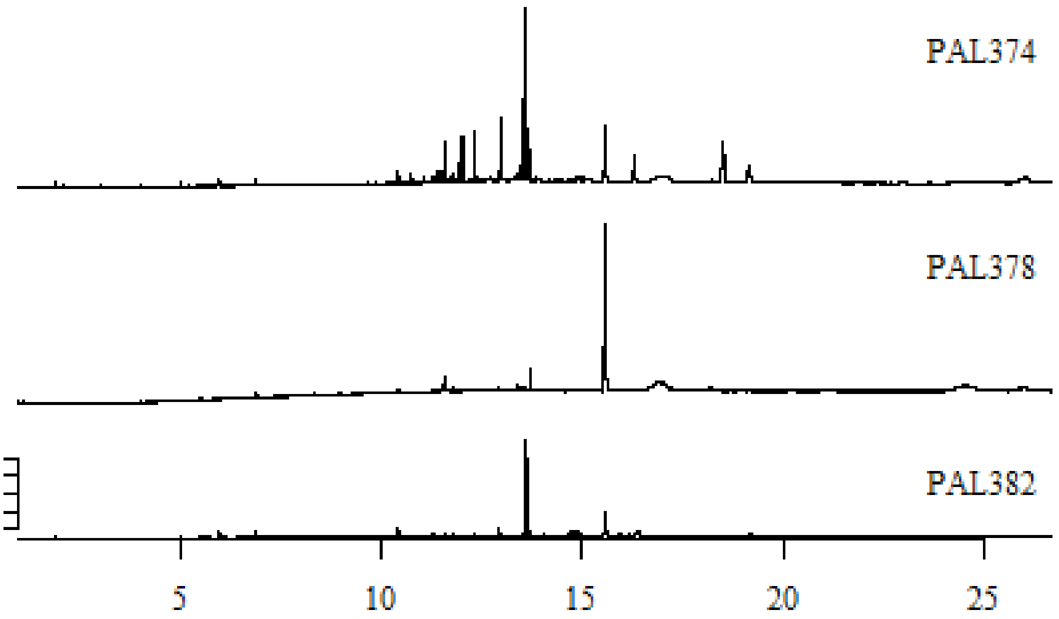



PAL307
PAL356
PAL362
PAL379
5
10
15
20
25

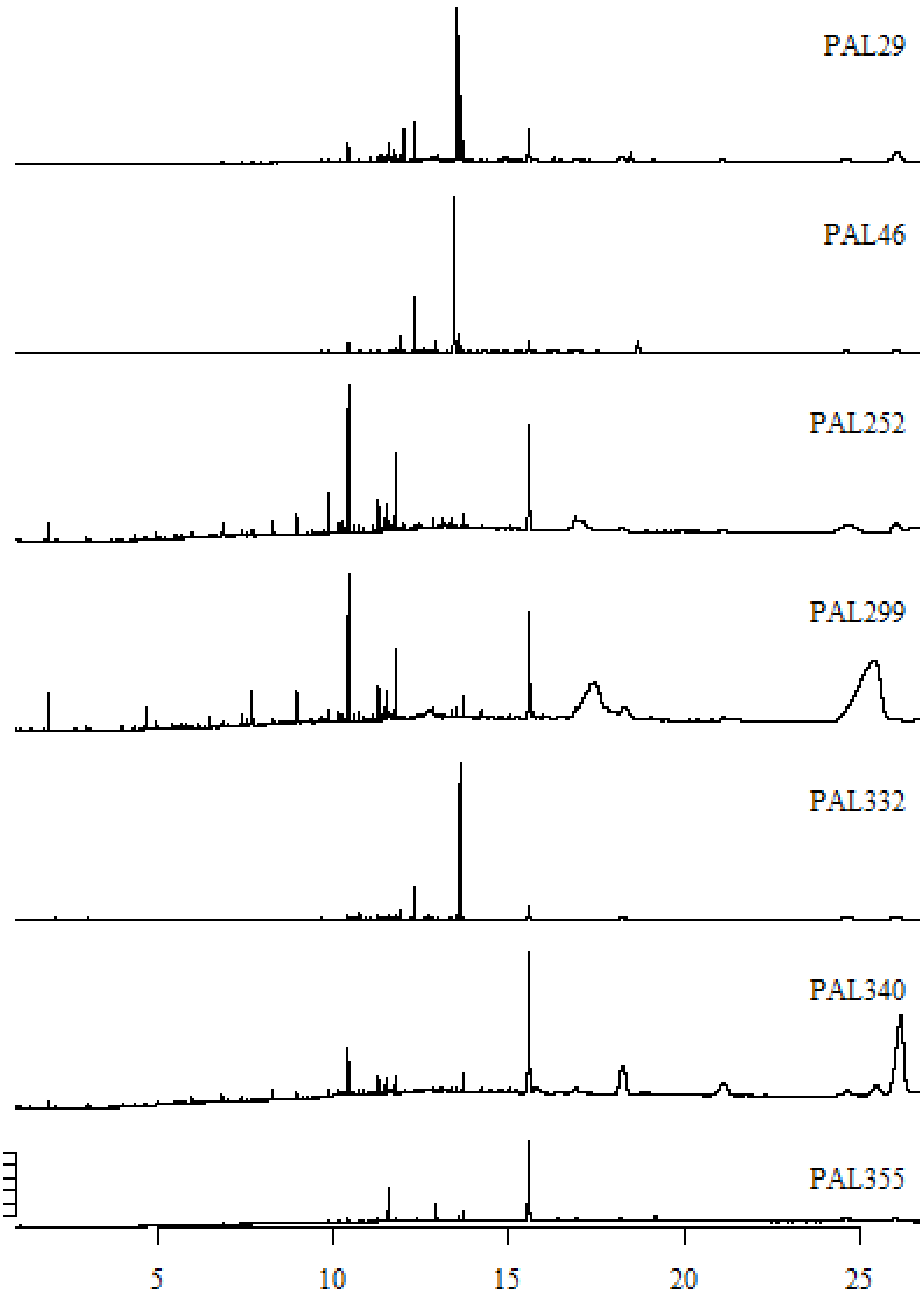
Gas Chromatograms - Clade F (HEX)

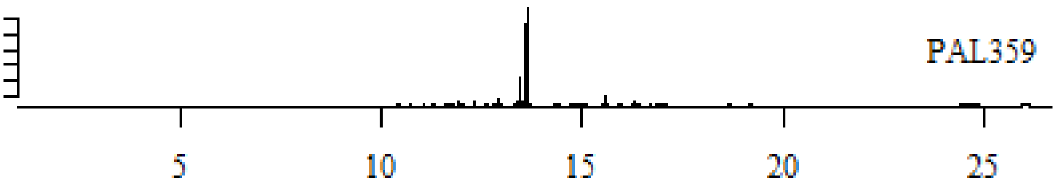



PAL43
PAL44
PAL130
PAL131
PAL133
PAL270
PAL333
PAL334
5
10
15
20
25


PAL374
PAL378
PAL382
5
10
15
20
25

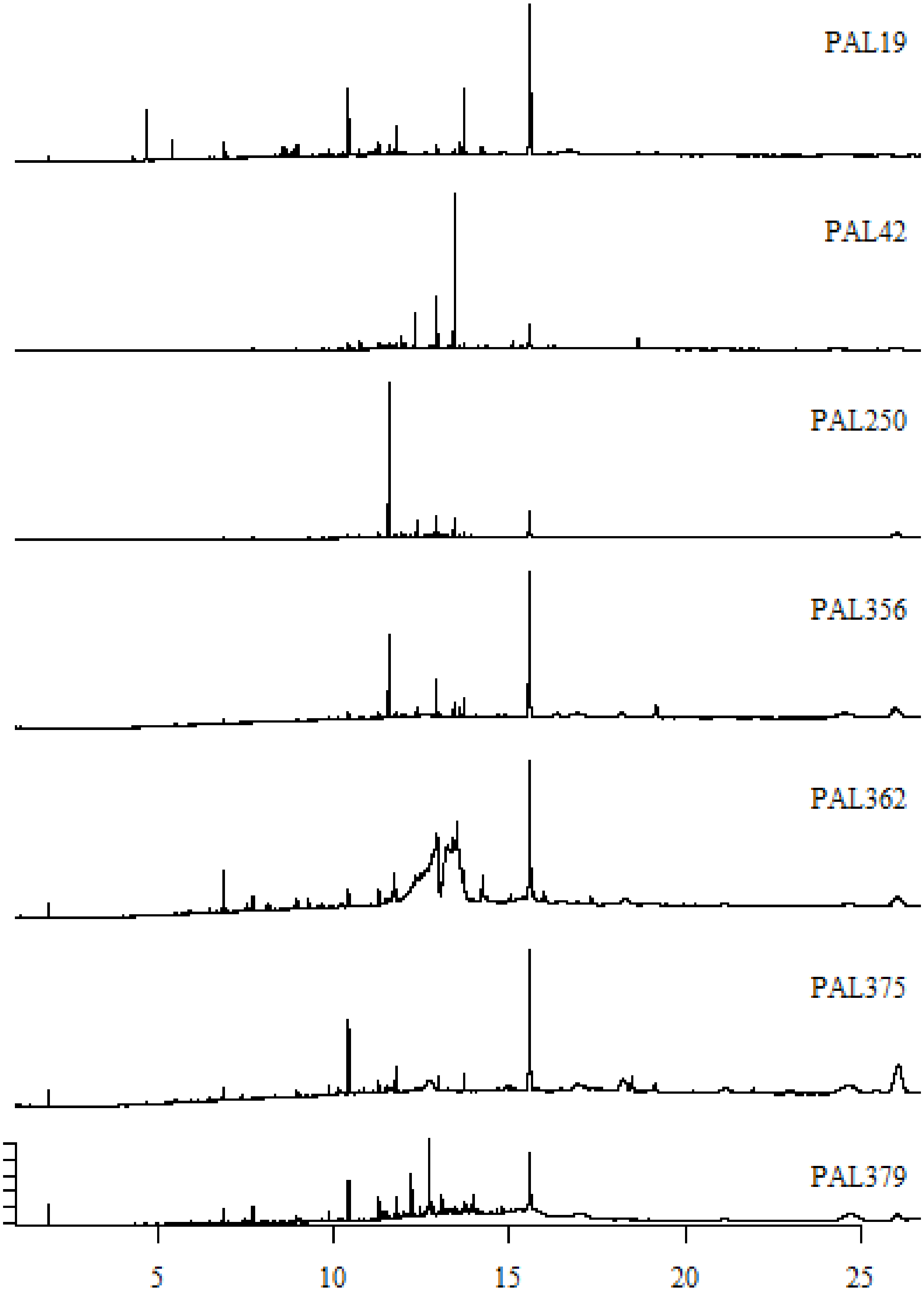
Gas Chromatograms - Clade D (HEX)





PAL359
5
10
15
20
25

Gas Chromatograms - Trocheliophorum (HEX)



Truncated Normalized Trapezoidal Areas

Note:

- We omit the first 15000 time points (about 5 minutes) where there is little activity
- We omit the last 15002 time points (about 5 minutes) where there is little activity

```r
#computes the trapezoidal area of each time interval in the given vector using the delta as the height
trap <- function(intens, delta) {
  intens1 <- rep(intens)
  intens1 = intens1[-1]
  intens = intens[1:length(intens)-1]
  trapArea = (intens + intens1)*delta/2
  return(c(as.numeric(trapArea), 0))
}

#normalizes the vector input so that the total area adds to 1
norm <- function(trapArea){
  tot = sum(trapArea)
  return(trapArea/tot)
}

#We omit the first and last 15000 time observations (5 minutes) because there is little observed activi
TRUNC_BEGIN = 15003
TRUNC_END = 62002

#Process as dataframe - CHANGE TO AS.NUMERIC
dfDCM_area = dfAllT_DCM[,TRUNC_BEGIN:TRUNC_END]
for(i in 1:nrow(dfDCM_area)){
  dfDCM_area[i, ] = norm(trap(as.numeric(dfDCM_area[i, ]), DELTA))
}
rownames(dfDCM_area) = dfAllT_DCM[,1]

dfHEX_area = dfAllT_HEX[,TRUNC_BEGIN:TRUNC_END]
for(i in 1:nrow(dfHEX_area)){
  dfHEX_area[i, ] = norm(trap(as.numeric(dfHEX_area[i, ]), DELTA))
}
rownames(dfHEX_area) = dfAllT_HEX[,1]
```

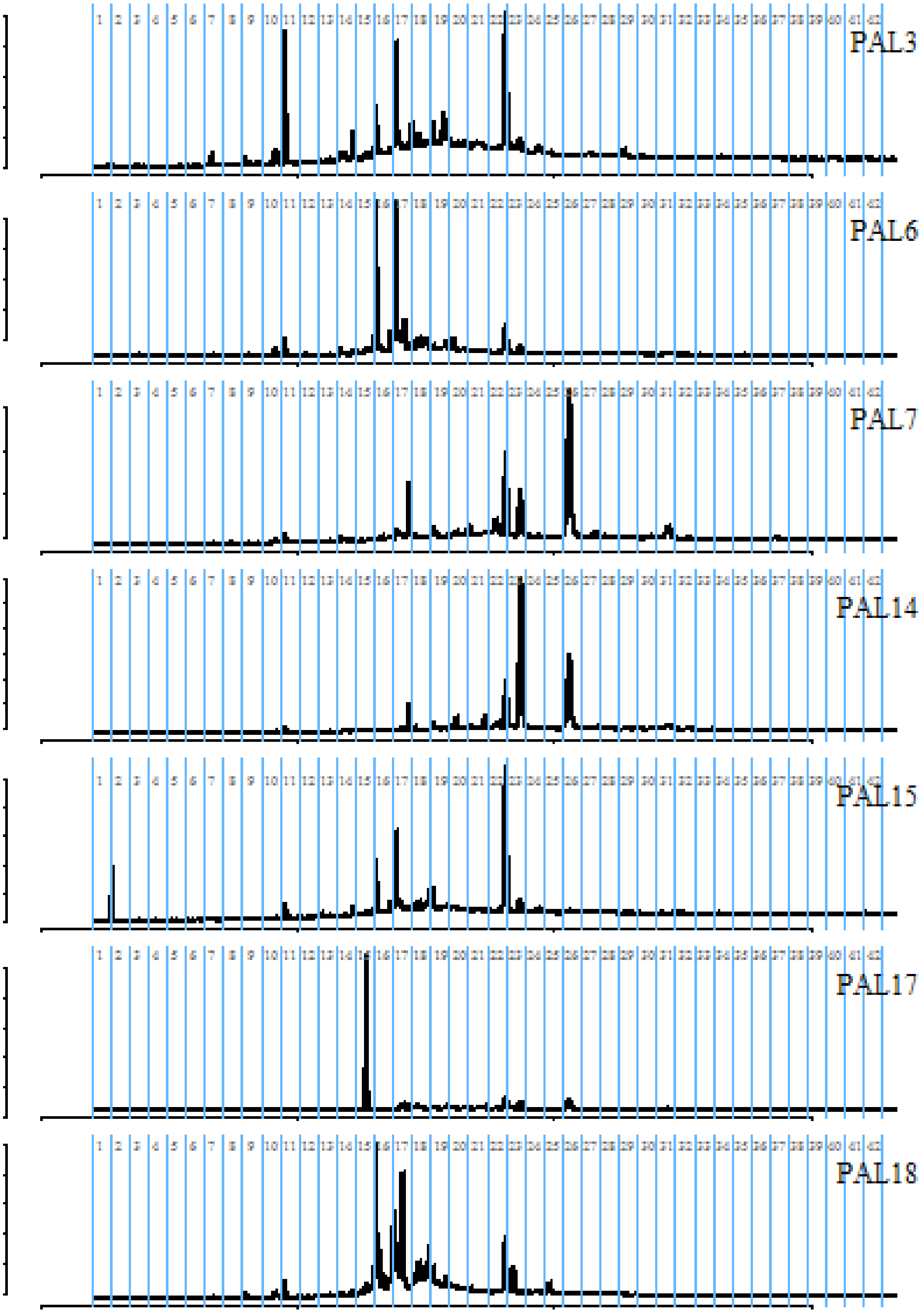
Binned Areas

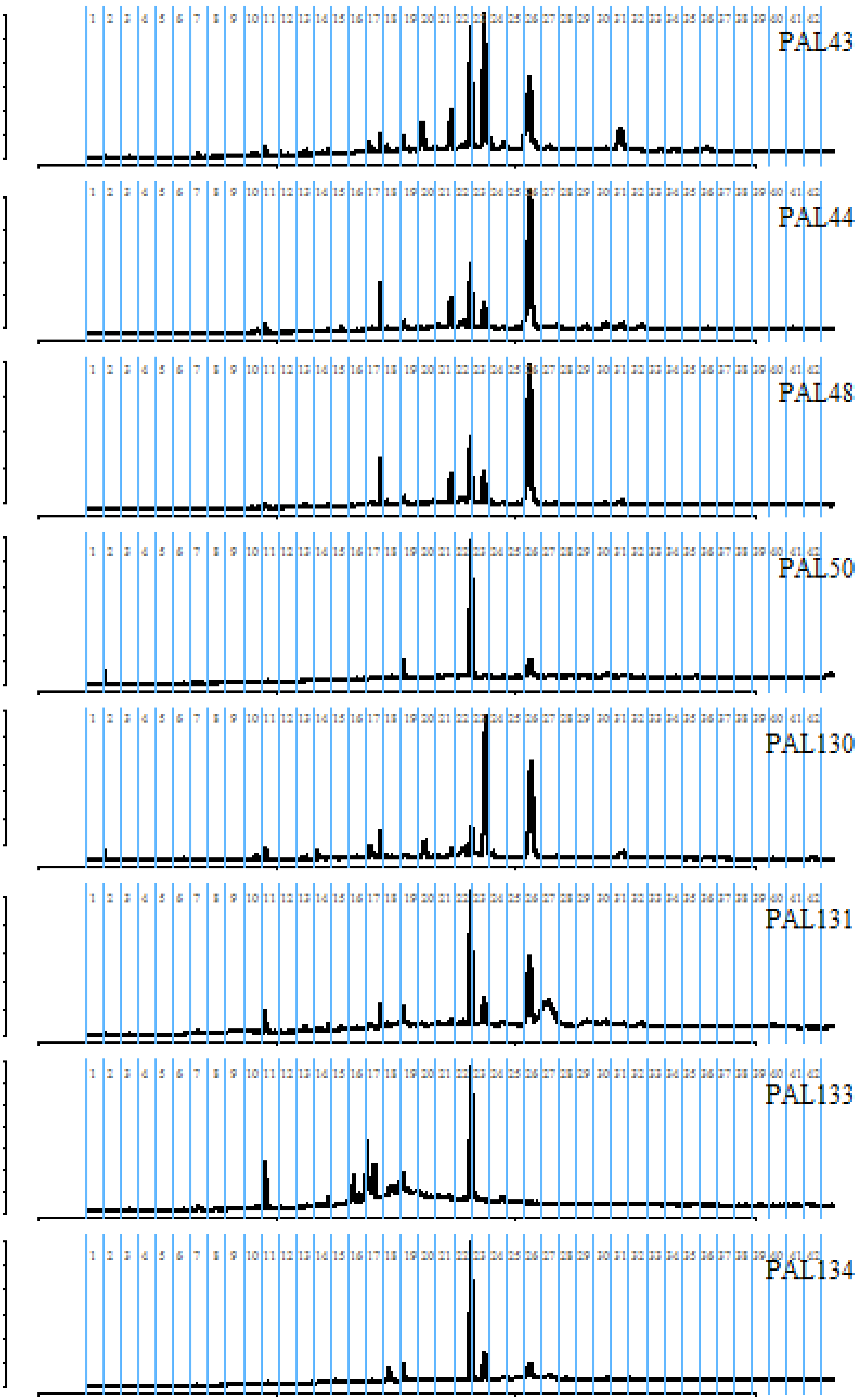
```r
#returns a binned matrix whose bins are the sums of the areas df for each bin interval
#each bin has as many time intervals as specified by binWidth
#any remainder from ncol(df)/binwidth will not be included in the binned matrix
binner <- function(df, binWidth, delta) {
  rowz = c()
  for(j in 1:nrow(df)) {
    col = c()
    for(i in seq(1, ncol(df)-binWidth, binWidth)){
      end = min(ncol(df),i + binWidth - 1)
      col = c(col, sum(as.numeric(df[j,i:end])))
    }
    rowz = rbind(rowz, col)
  }
  return(rowz)
}

binned_matrix_DCM = binner(dfDCM_area, 1100, DELTA)
bins = 1:ncol(binned_matrix_DCM)
for (i in 1:ncol(binned_matrix_DCM)){
  bins[i] = paste("Bin", as.character(i))
}
dimnames(binned_matrix_DCM) = list(dfAllT_DCM[,1], bins)

binned_matrix_HEX = binner(dfHEX_area, 1100, DELTA)
dimnames(binned_matrix_HEX) = list(dfAllT_HEX[,1], bins)
```

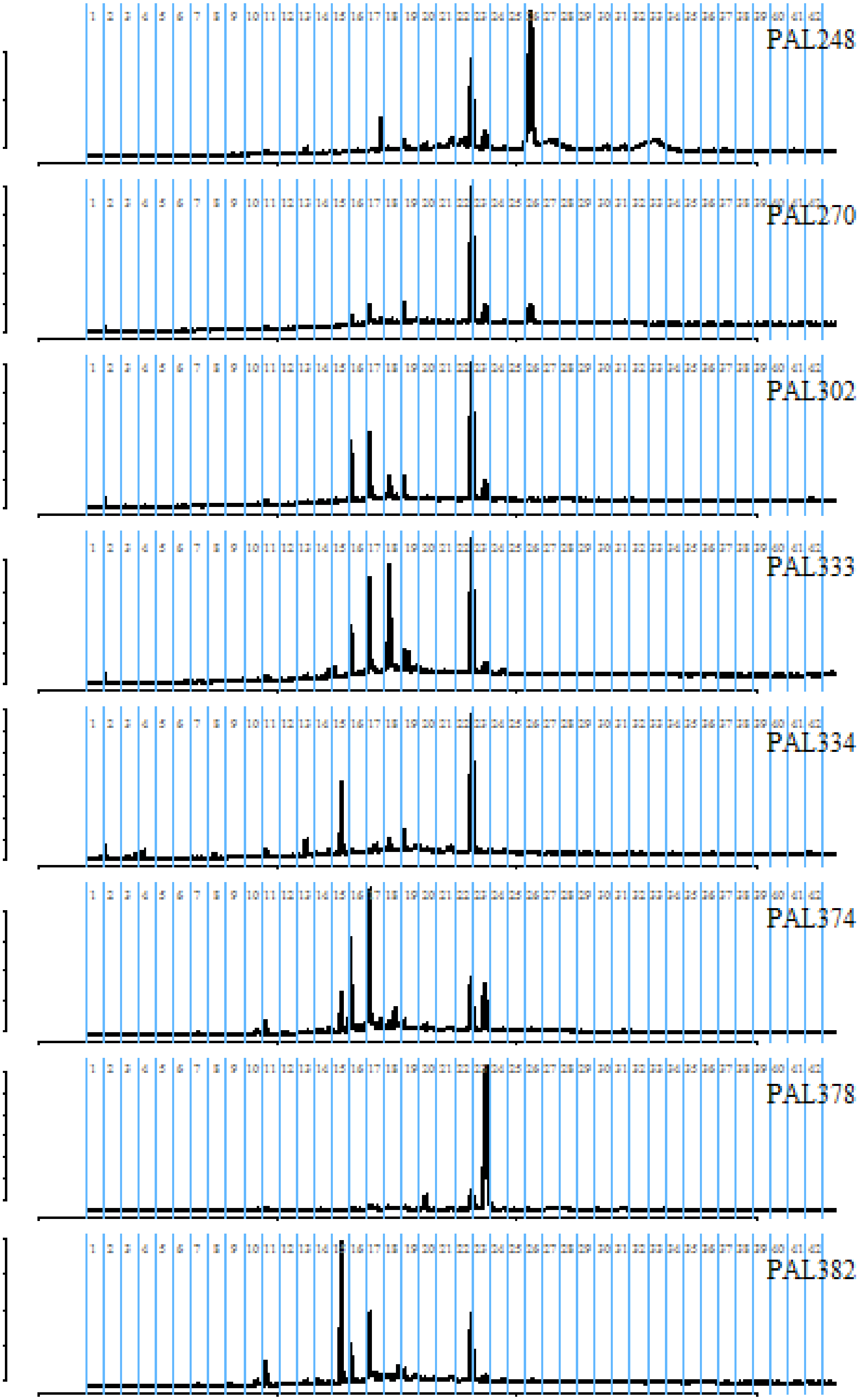
	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5
3	0.0196983	0.0197666	0.0200023	0.0198696	0.0201119
6	0.0178640	0.0179466	0.0188559	0.0181555	0.0182886
7	0.0194413	0.0195503	0.0196366	0.0196392	0.0197458
14	0.0190575	0.0191769	0.0192920	0.0192632	0.0194491
15	0.0204703	0.0218515	0.0206287	0.0207084	0.0209207

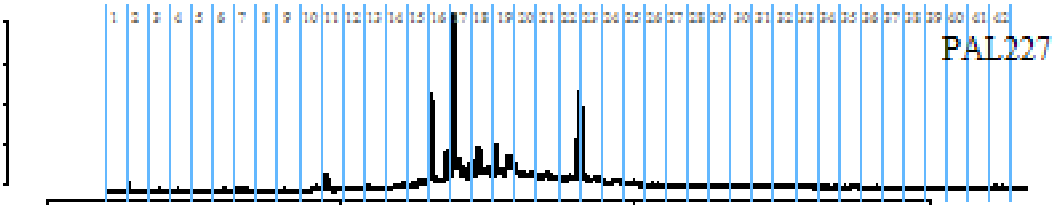
	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5
1	0.0204115	0.0207935	0.0211689	0.0210199	0.0209994
3	0.0204800	0.0207290	0.0209415	0.0209799	0.0211188
7	0.0202705	0.0204378	0.0207545	0.0207985	0.0209747
14	0.0199825	0.0202935	0.0207790	0.0206028	0.0209530
15	0.0196398	0.0197934	0.0205295	0.0200836	0.0204511

Gas Chromatograms (Truncated and Showing Bins) - Clade F (DCM)

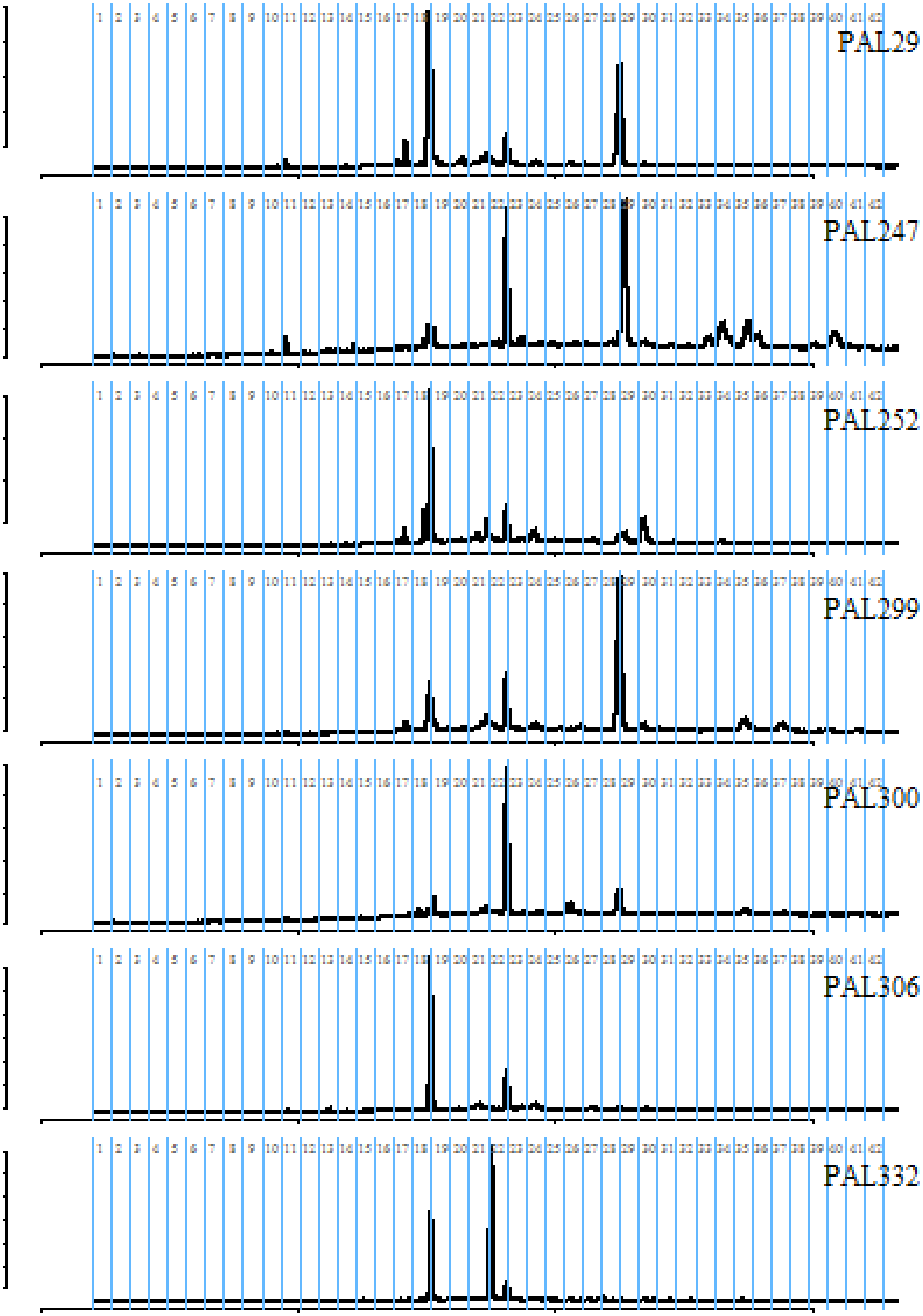


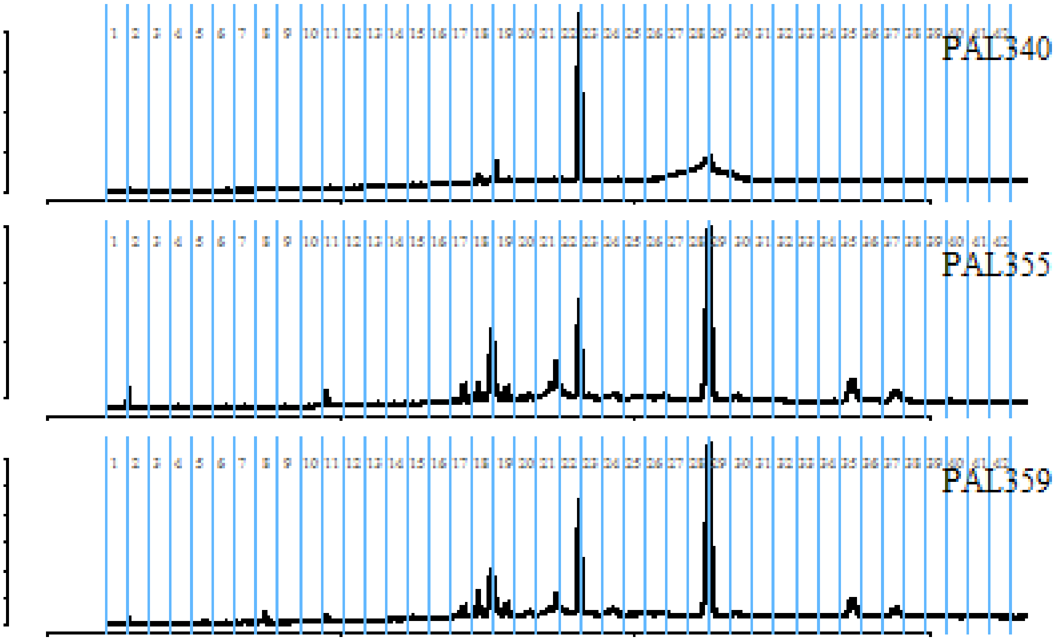

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL43
PAL44
PAL48
PAL50
PAL130
PAL131
PAL133
PAL134


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL248
PAL270
PAL302
PAL333
PAL334
PAL374
PAL378
PAL382

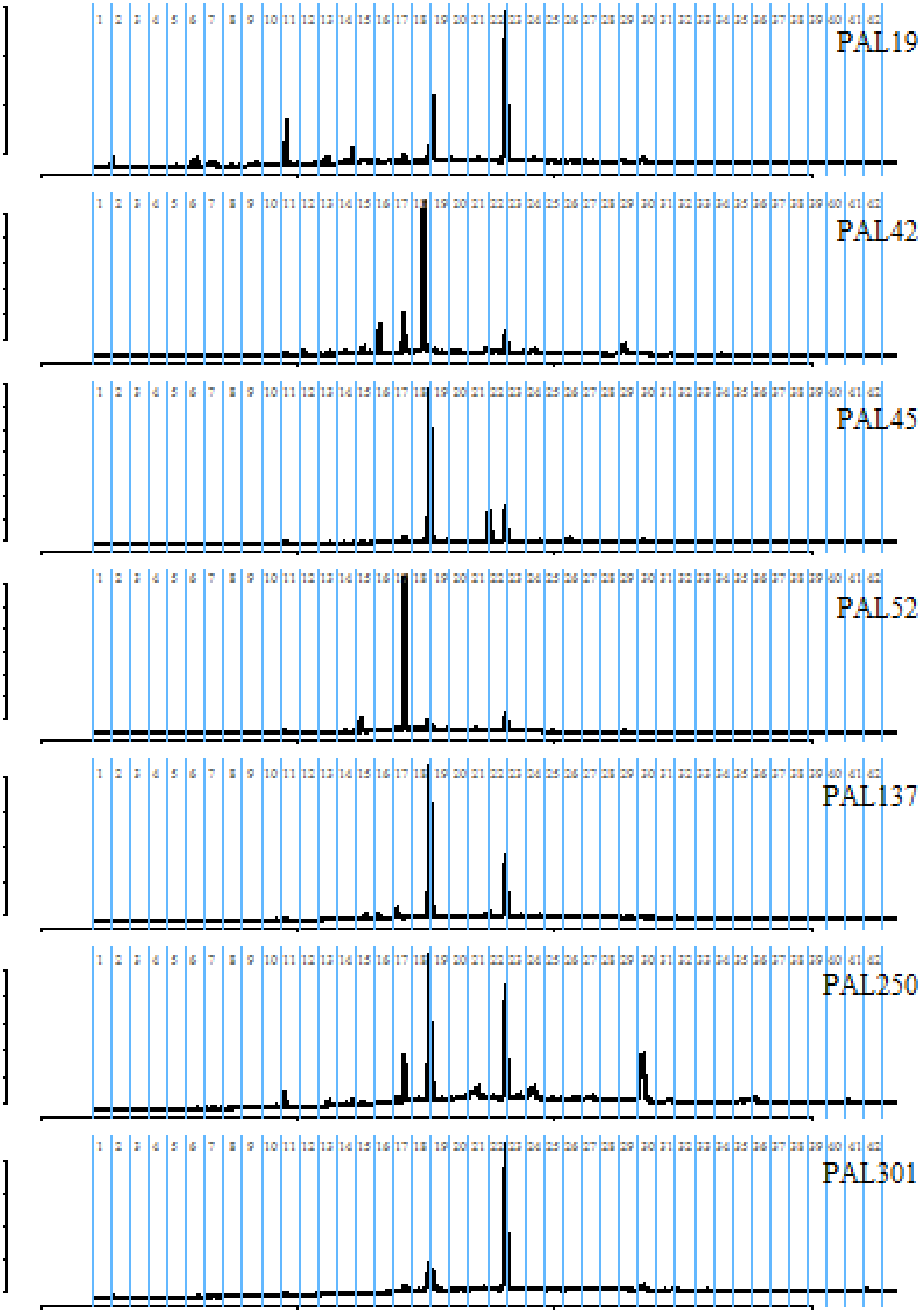

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL227

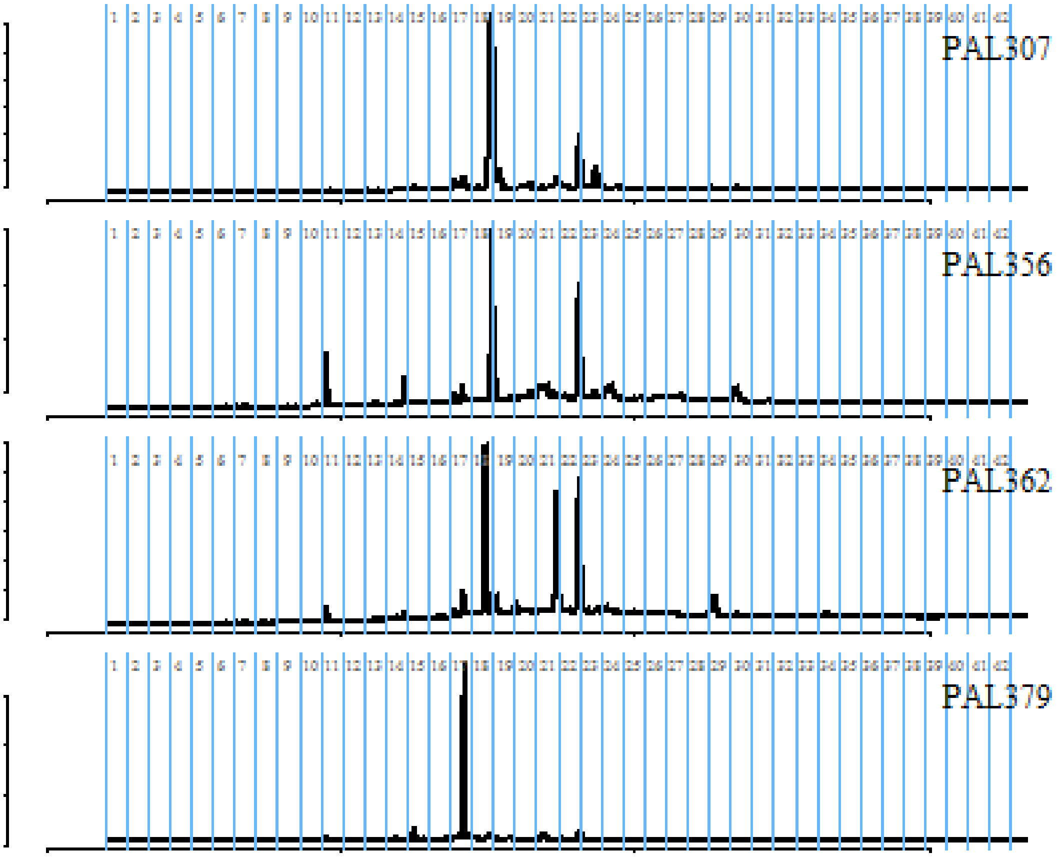
Gas Chromatograms (Truncated and Showing Bins) - Clade D (DCM)



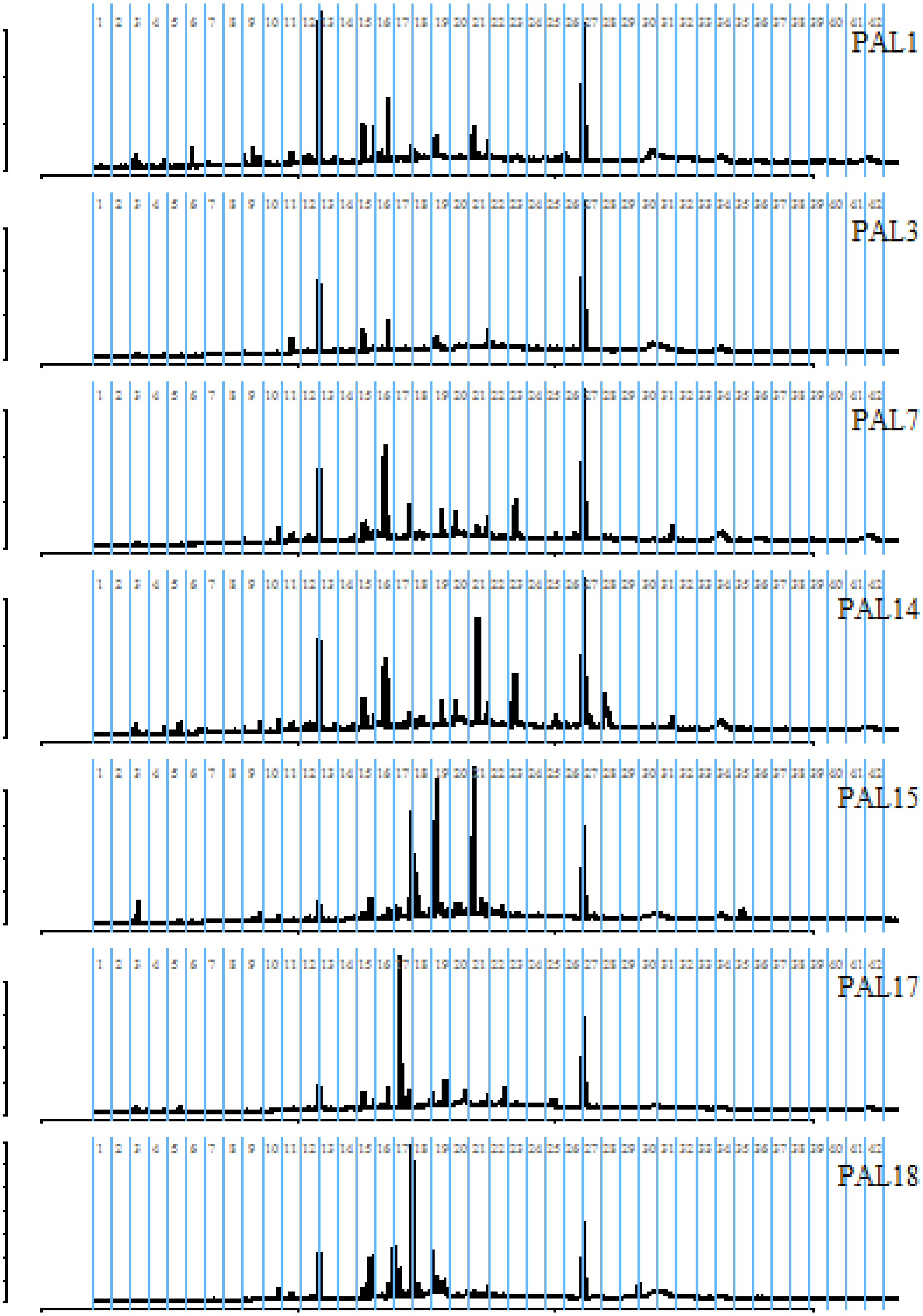

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL340
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL355
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL359

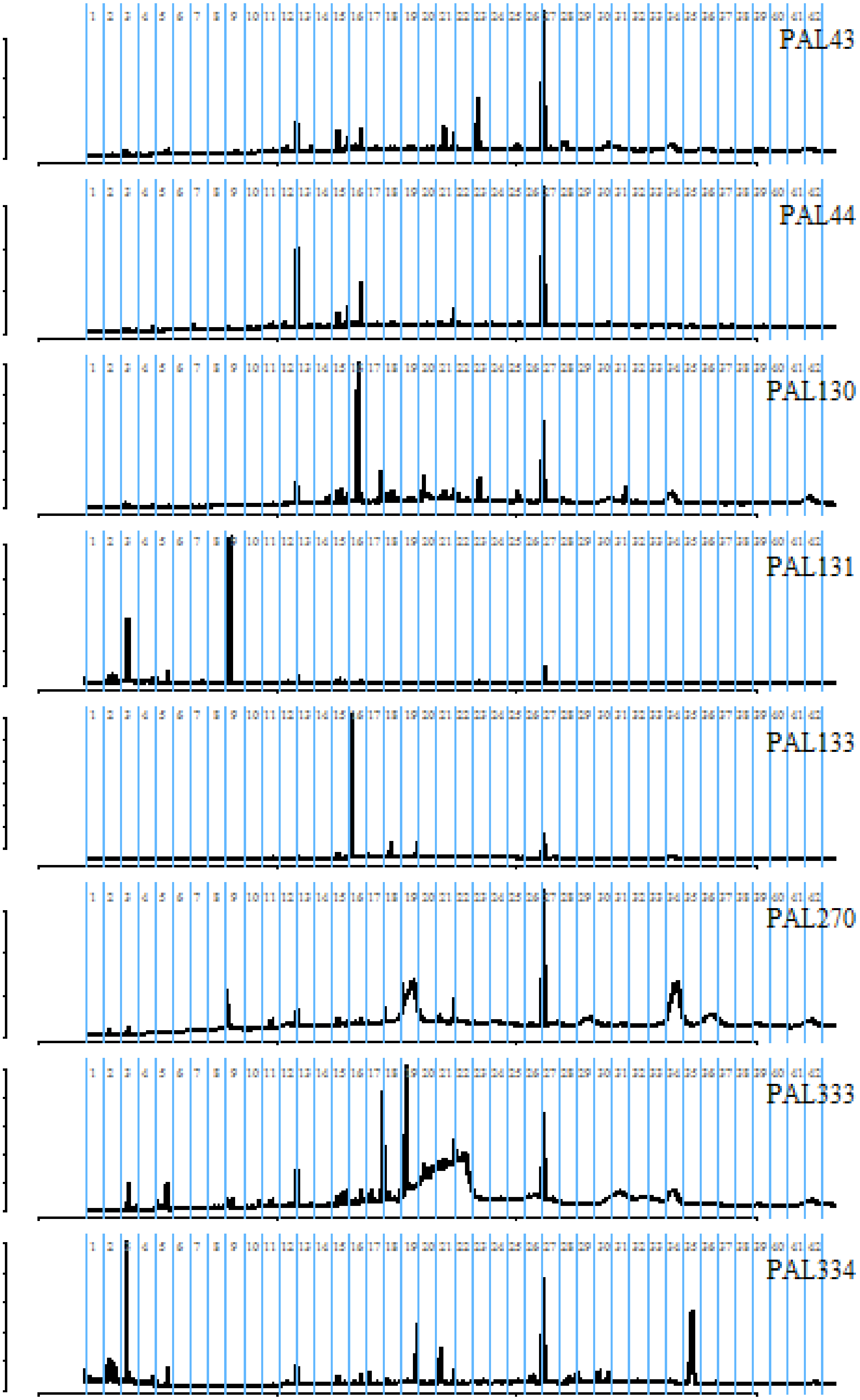
Gas Chromatograms (Truncated and Showing Bins) - Trocheliophorum (DCM)

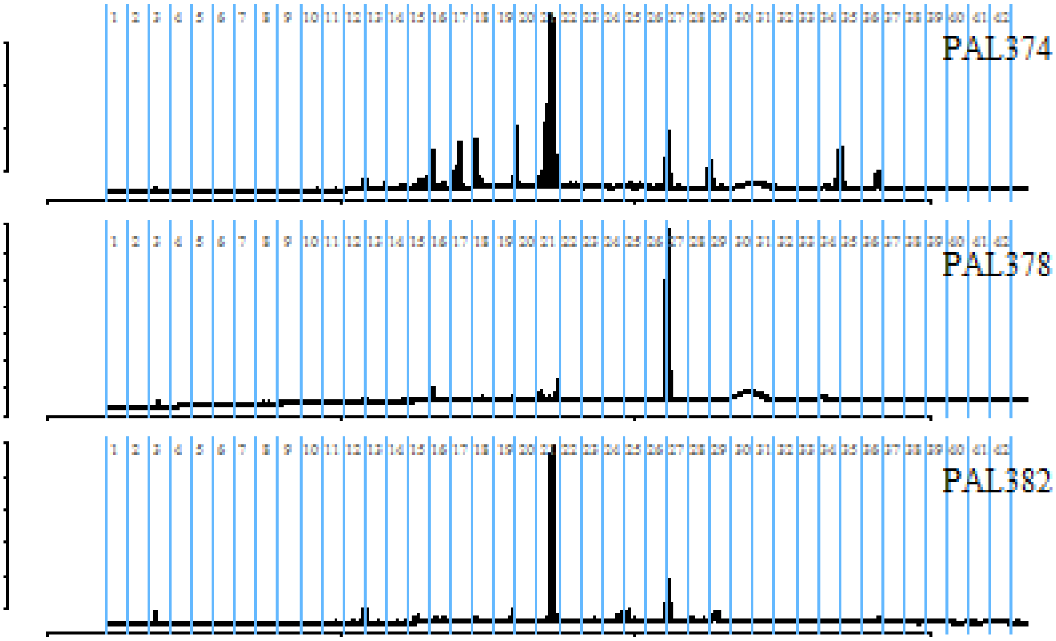



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL307
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL356
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL362
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL379

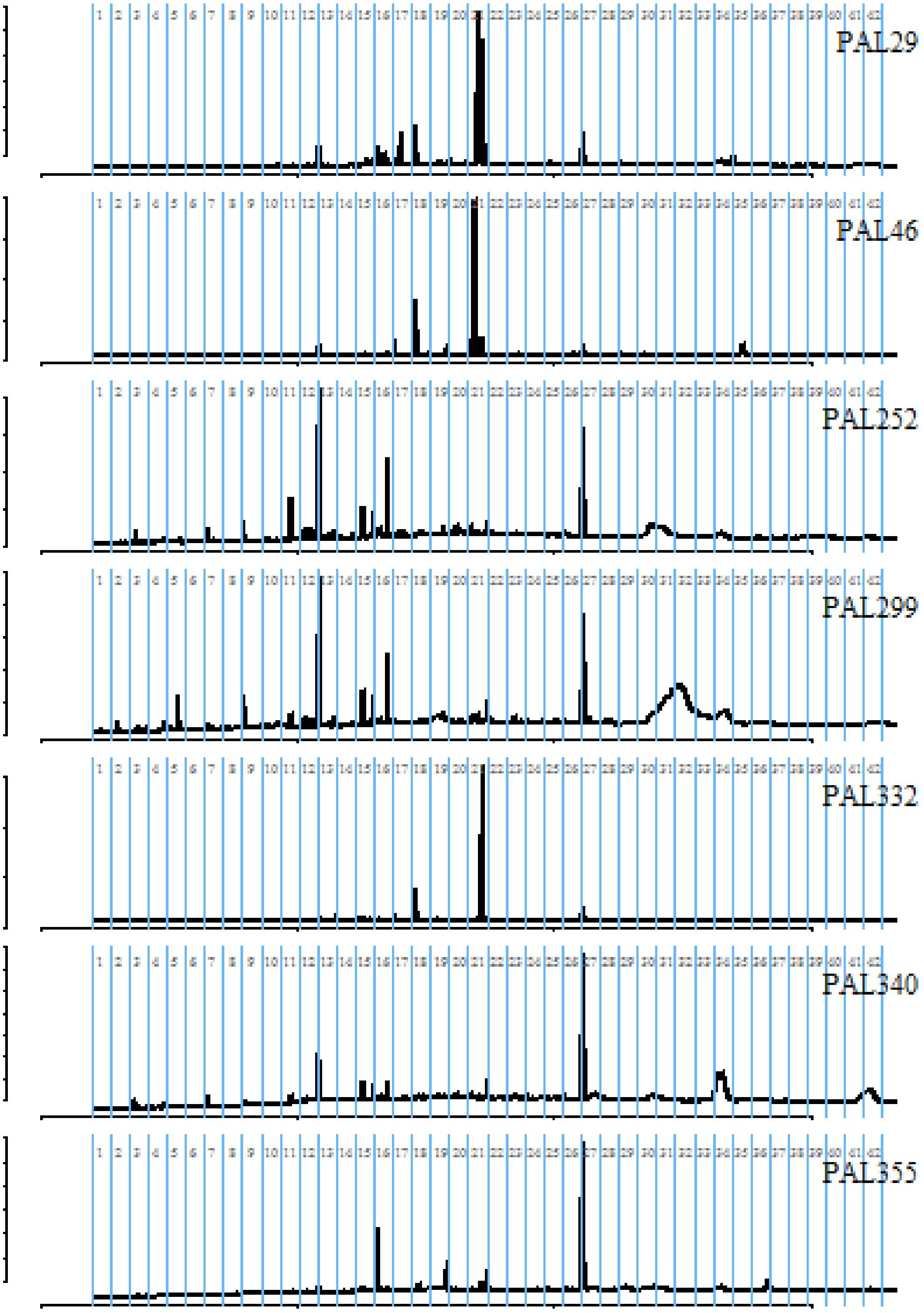
Gas Chromatograms (Truncated and Showing Bins) - Clade F (HEX)

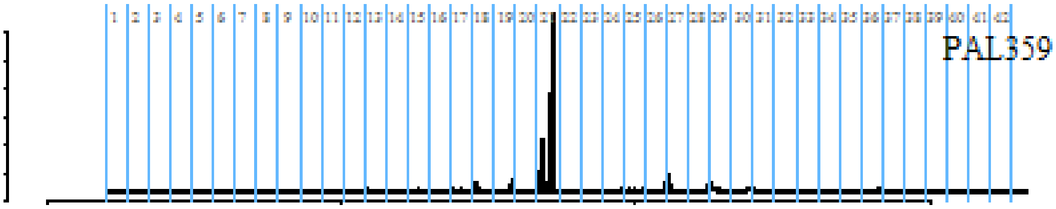



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL43
PAL44
PAL130
PAL131
PAL133
PAL270
PAL333
PAL334

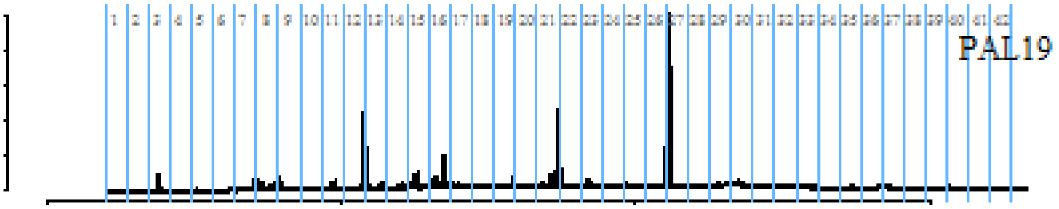

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL374
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL378
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL382

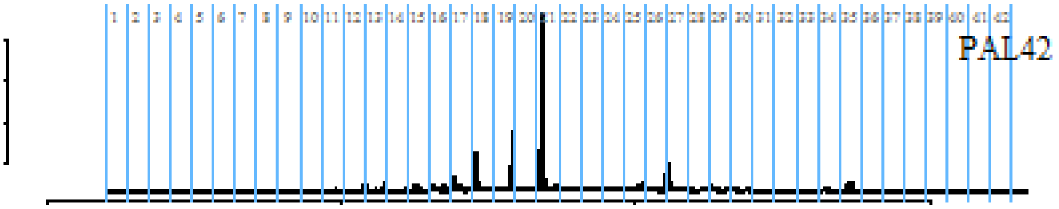
Gas Chromatograms (Truncated and Showing Bins) - Clade D (HEX)

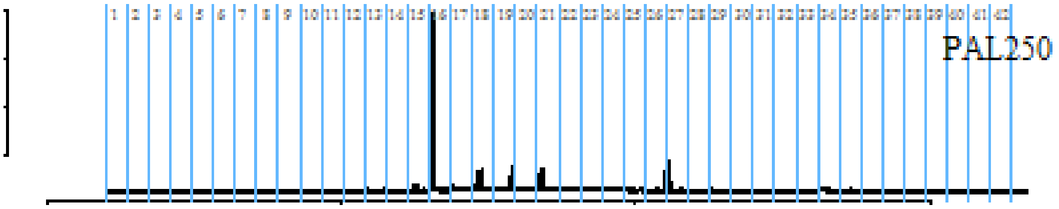


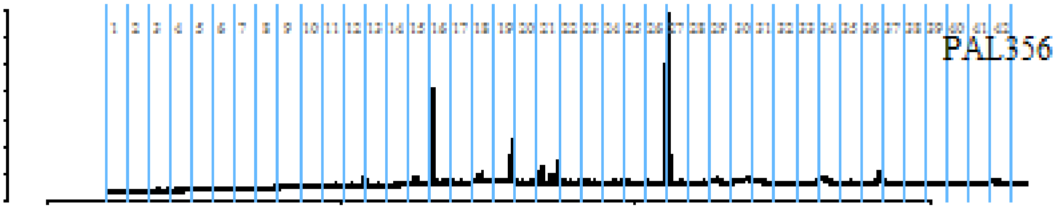

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
PAL359

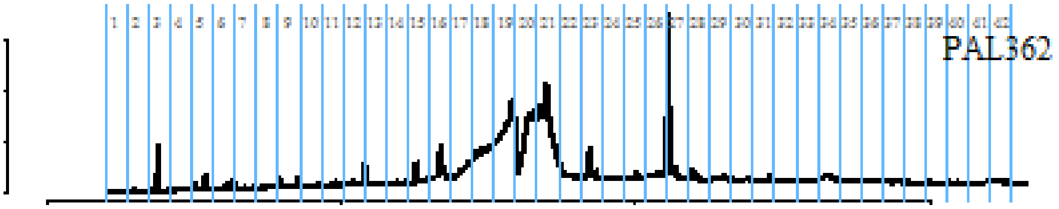
Gas Chromatograms (Truncated and Showing Bins) - Trocheliophorum (HEX)

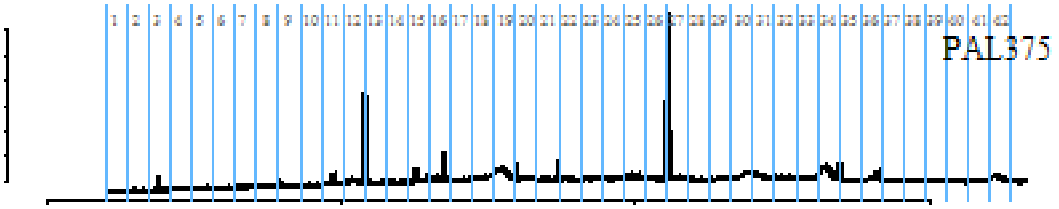


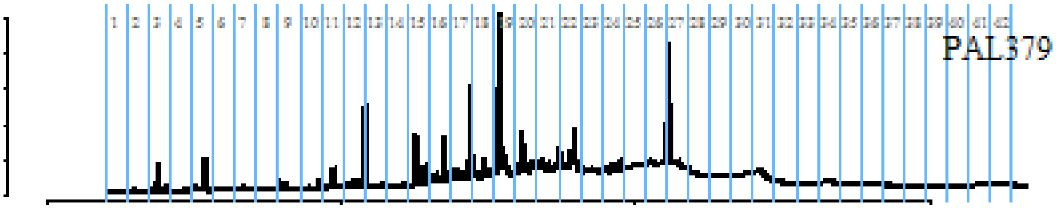




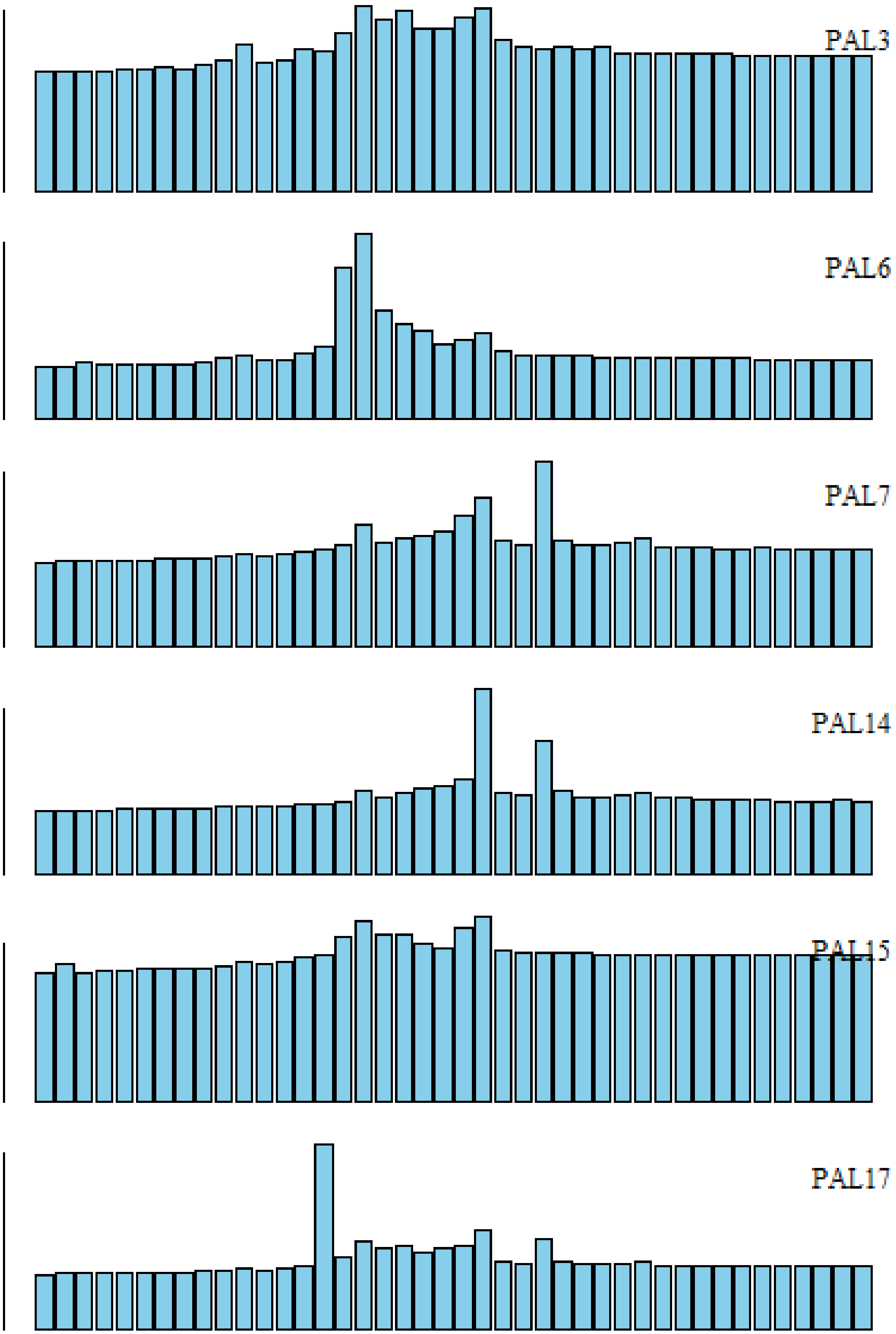


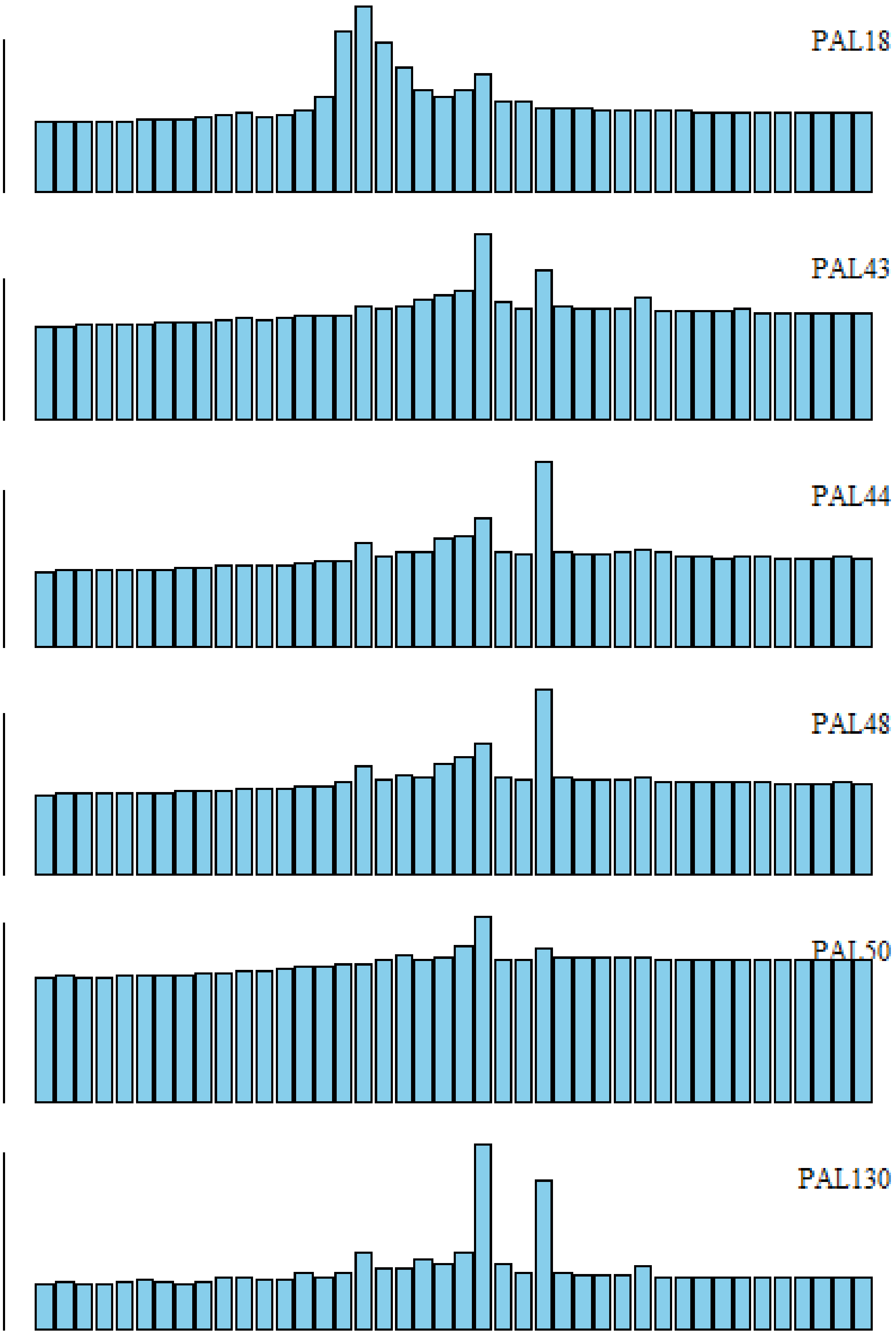


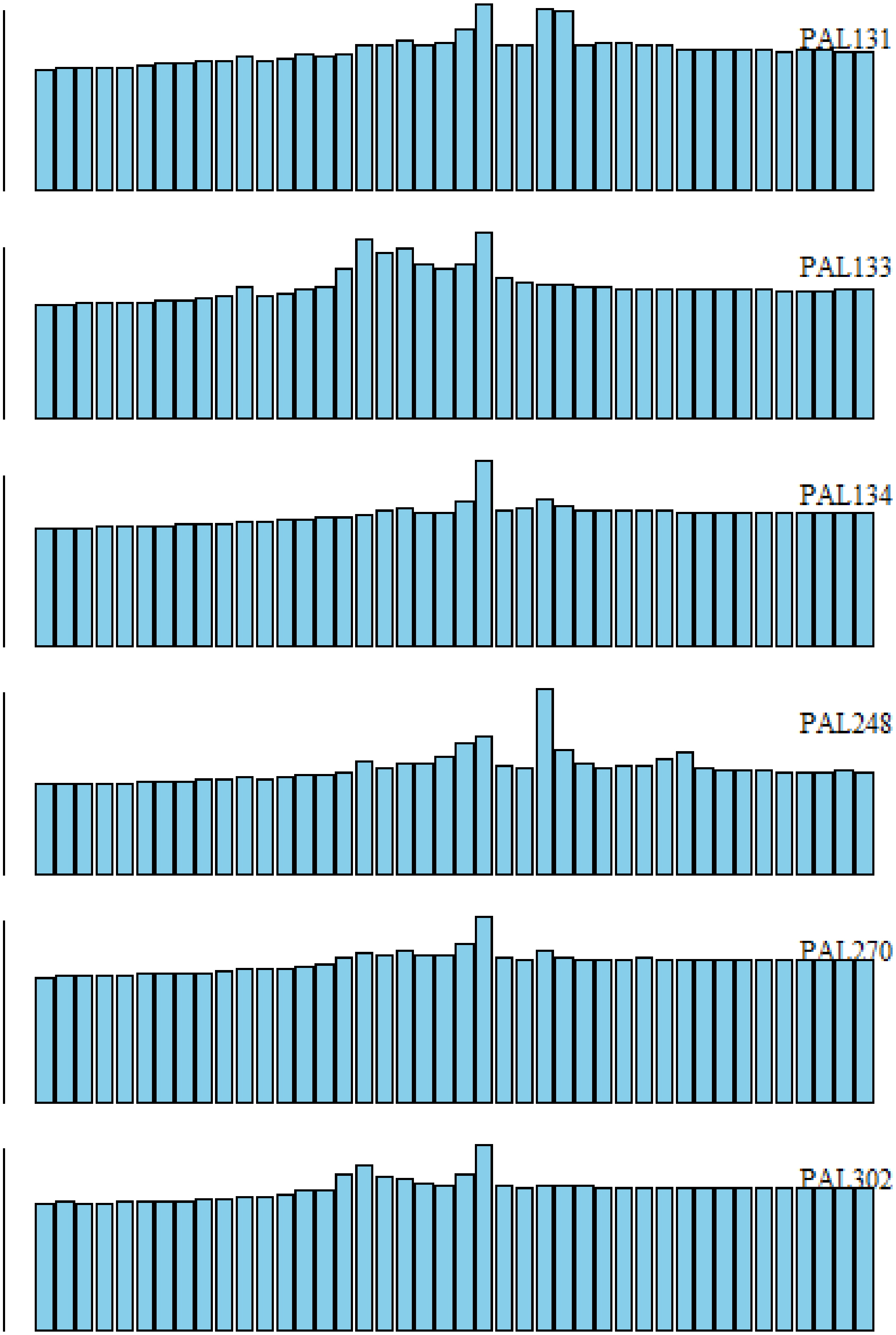


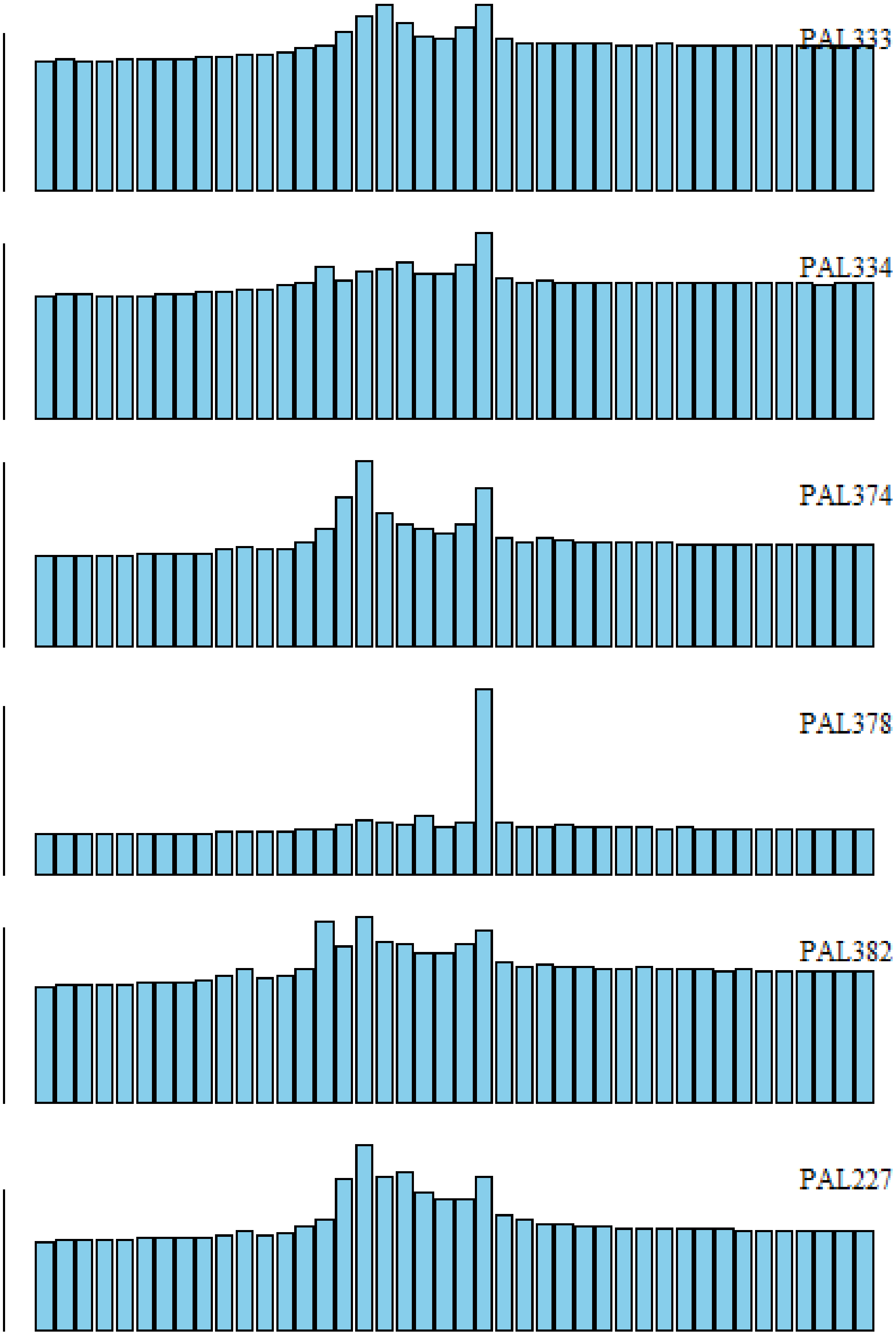


Gas Binned Barplots - Clade F (DCM)

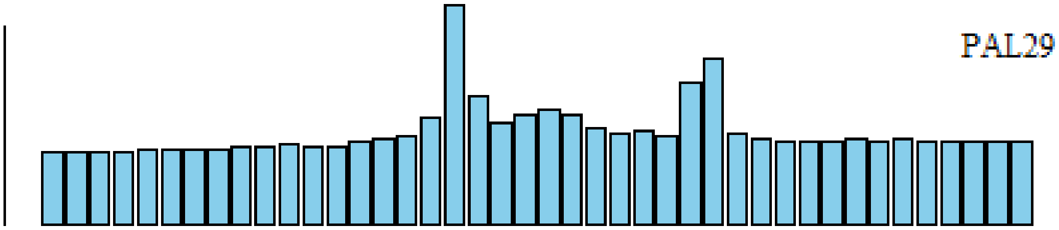


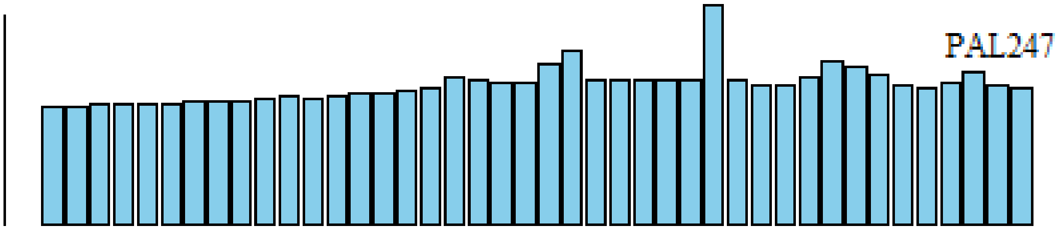

PAL18
PAL43
PAL44
PAL48
PAL50
PAL130

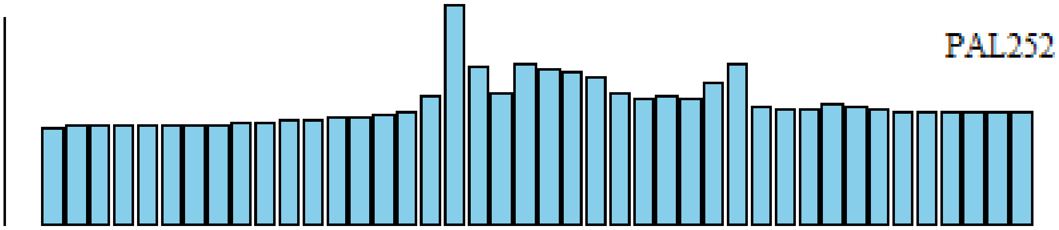

PAL131
PAL133
PAL134
PAL248
PAL270
PAL302

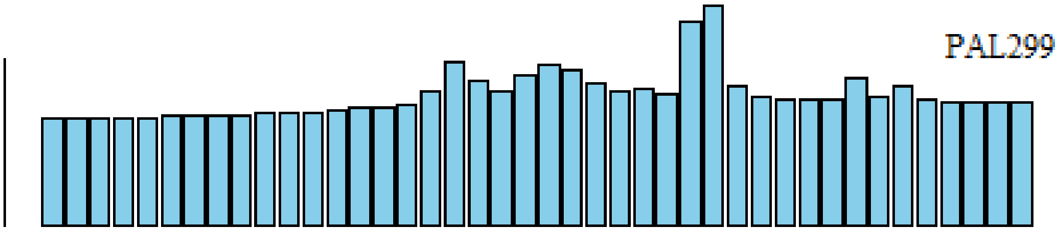

PAL333
PAL334
PAL374
PAL378
PAL382
PAL227

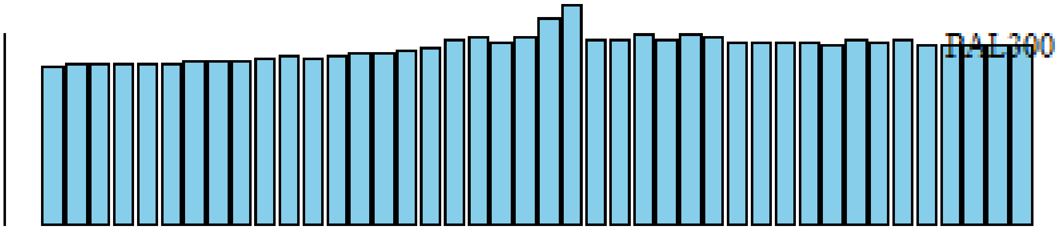
Gas Binned Barplots - Clade D (DCM)

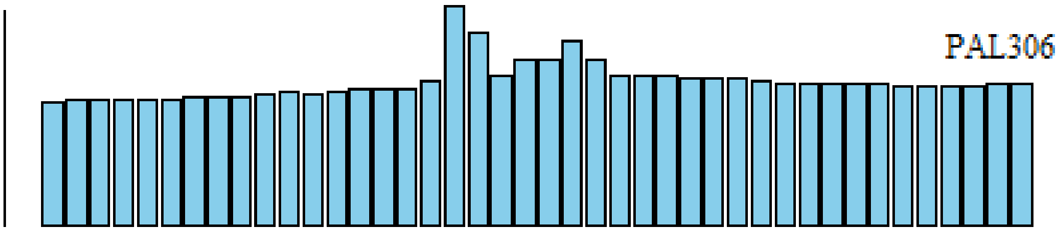


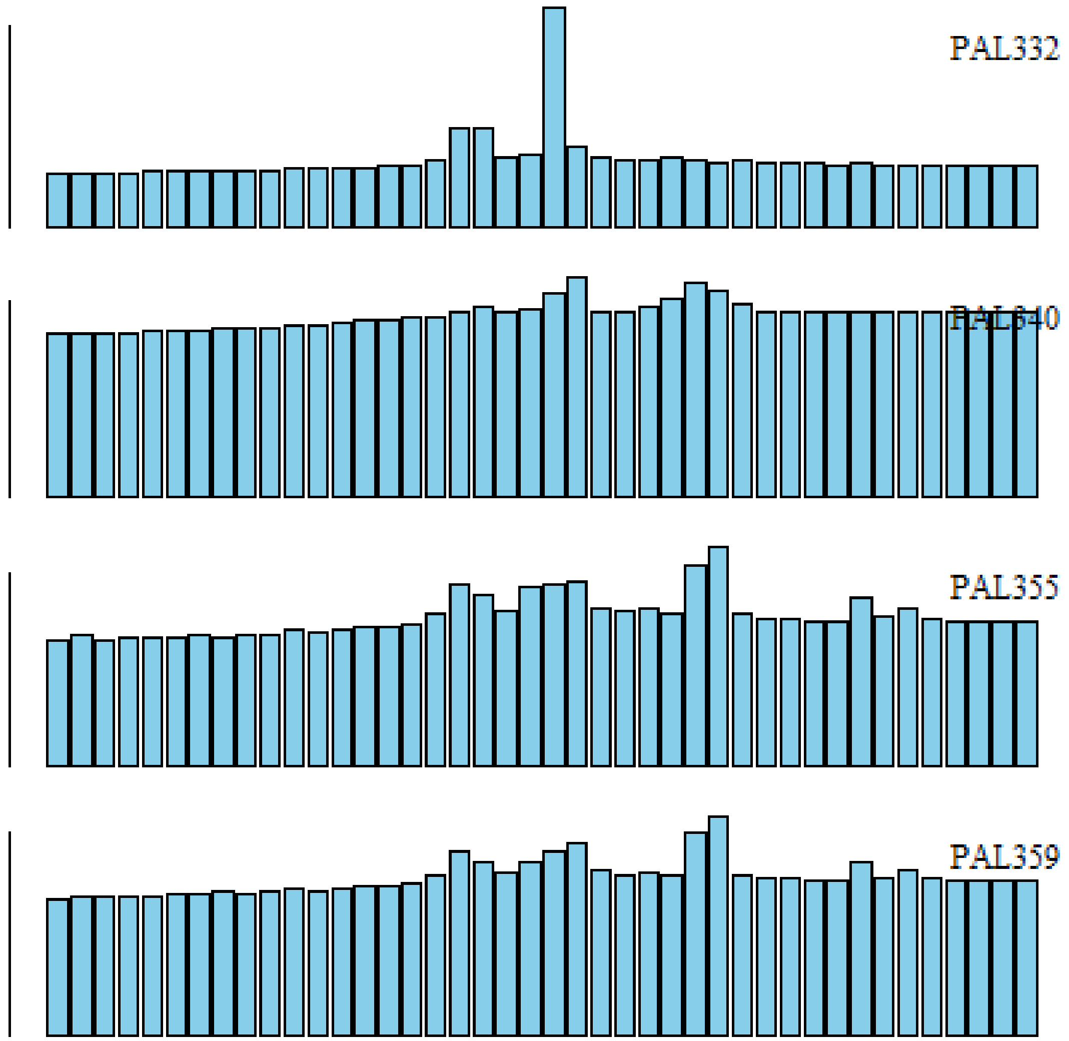




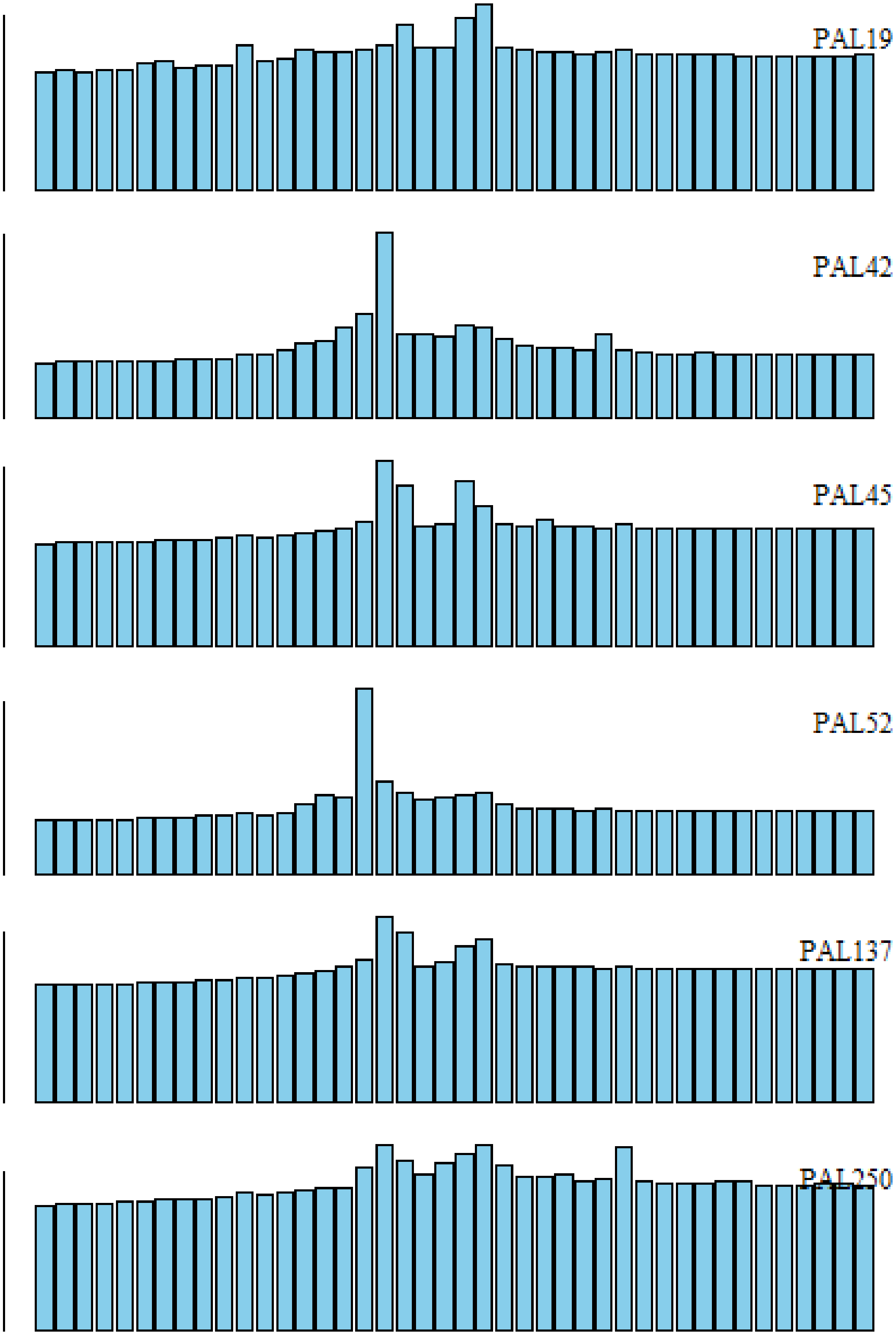


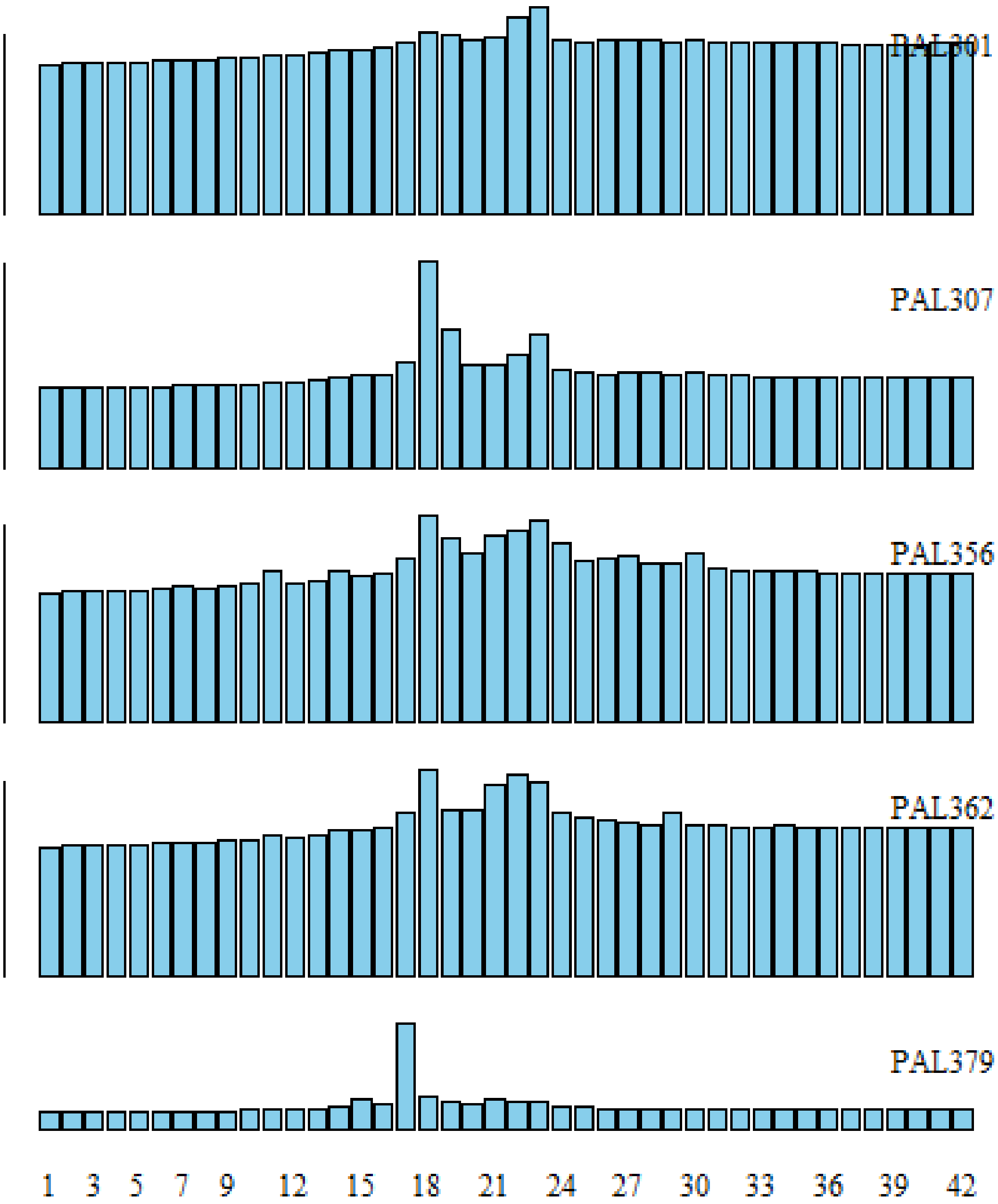




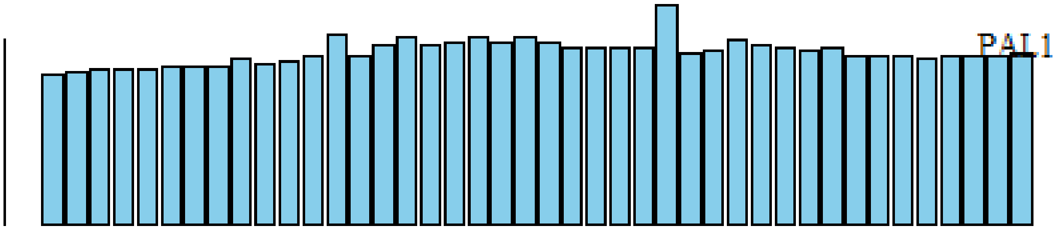

PAL332
PAL340
PAL355
PAL359

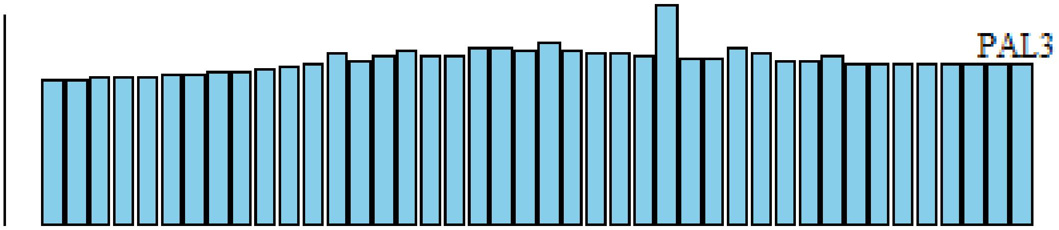
Gas Binned Barplots - Trocheliophorum (DCM)

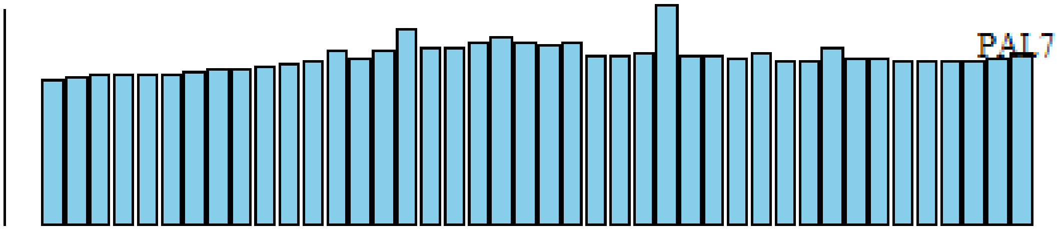


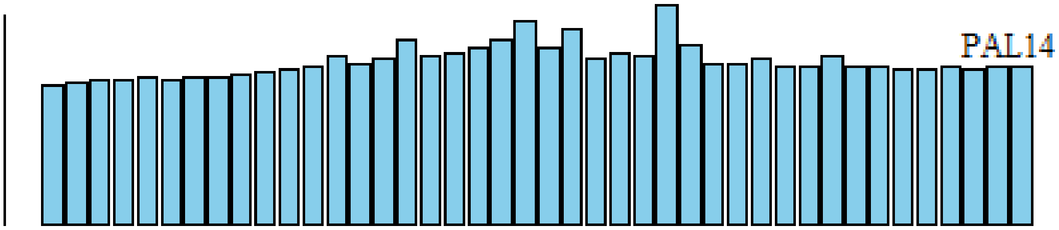

PAL301
PAL307
PAL356
PAL362
PAL379
1
3
5
7
9
12
15
18
21
24
27
30
33
36
39
42

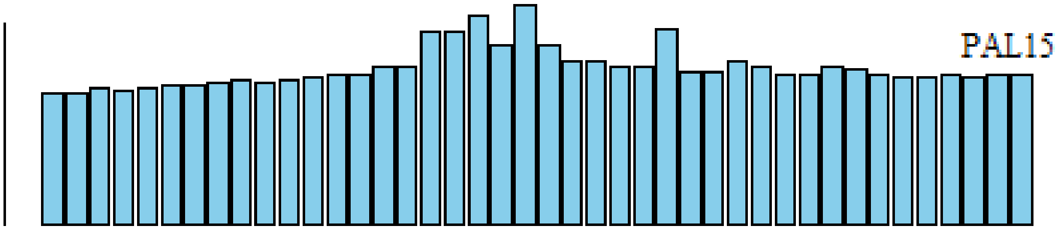
Gas Binned Barplots - Clade F (HEX)

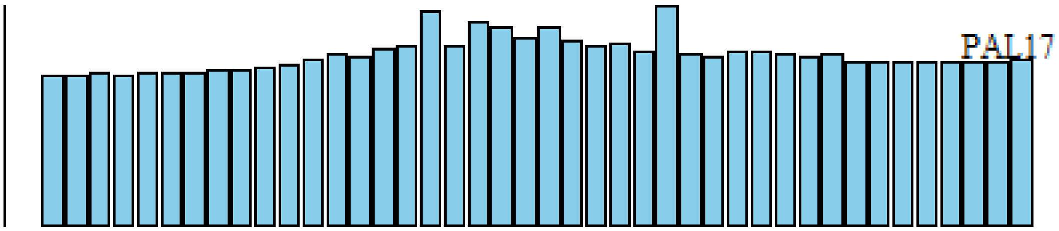


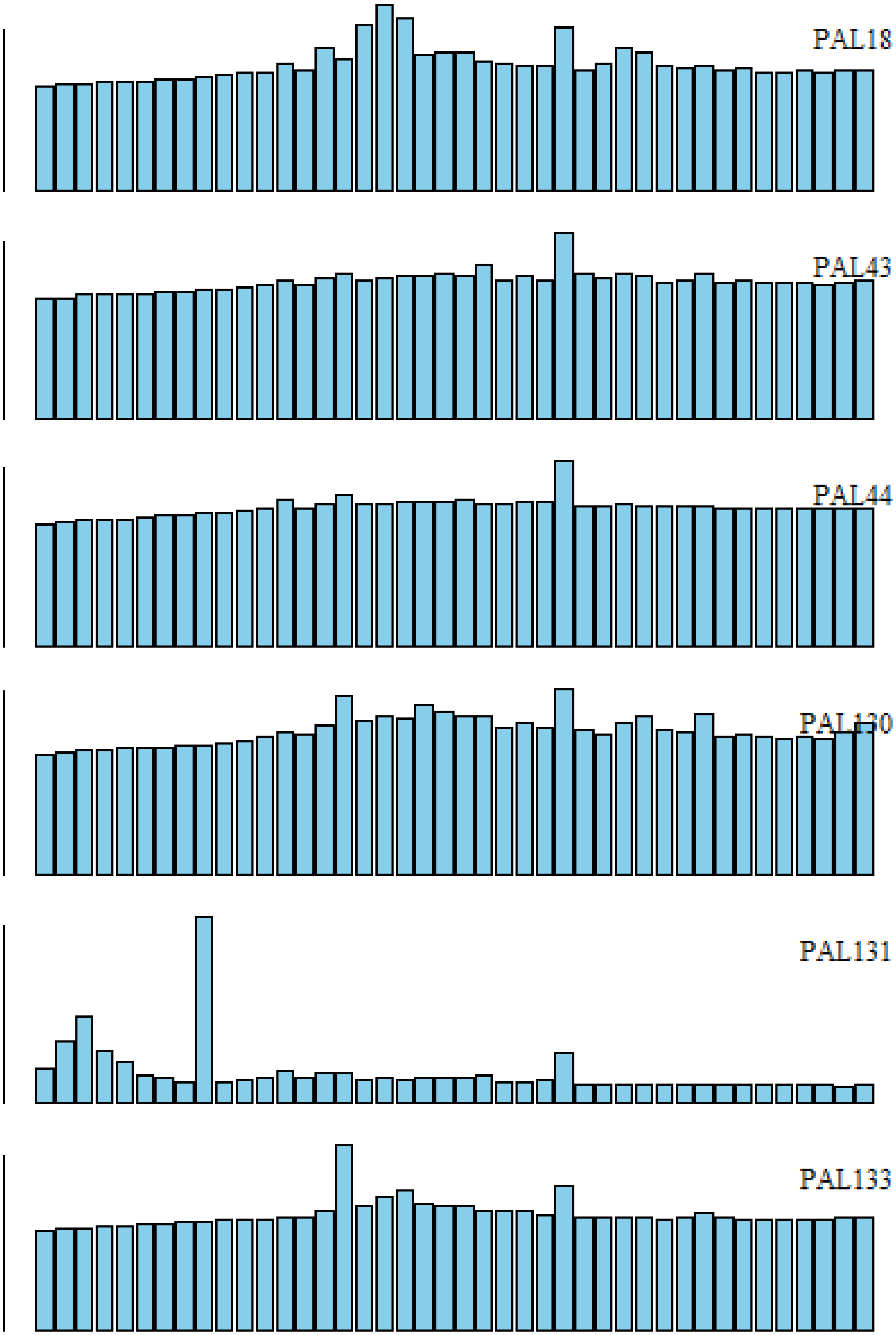


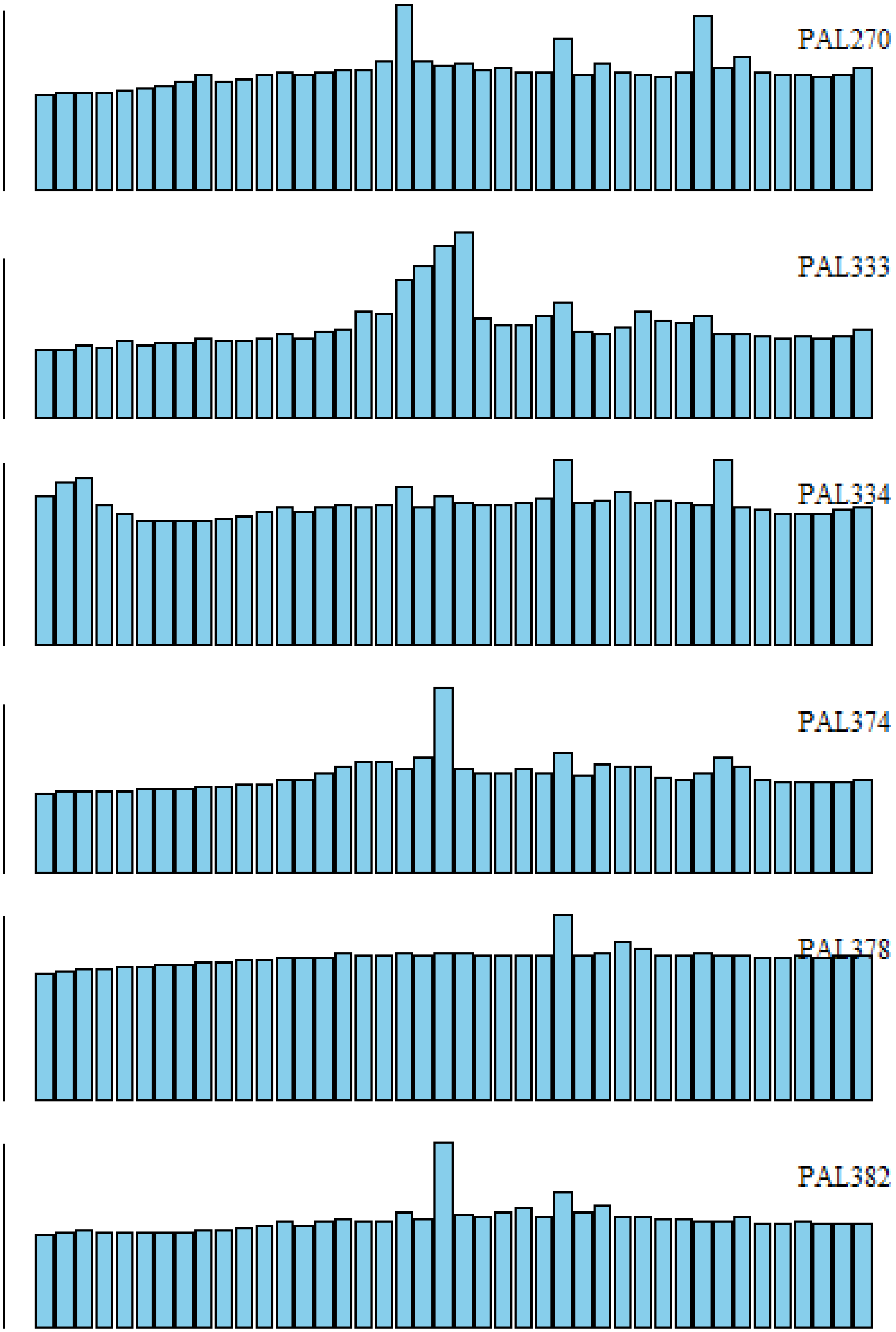




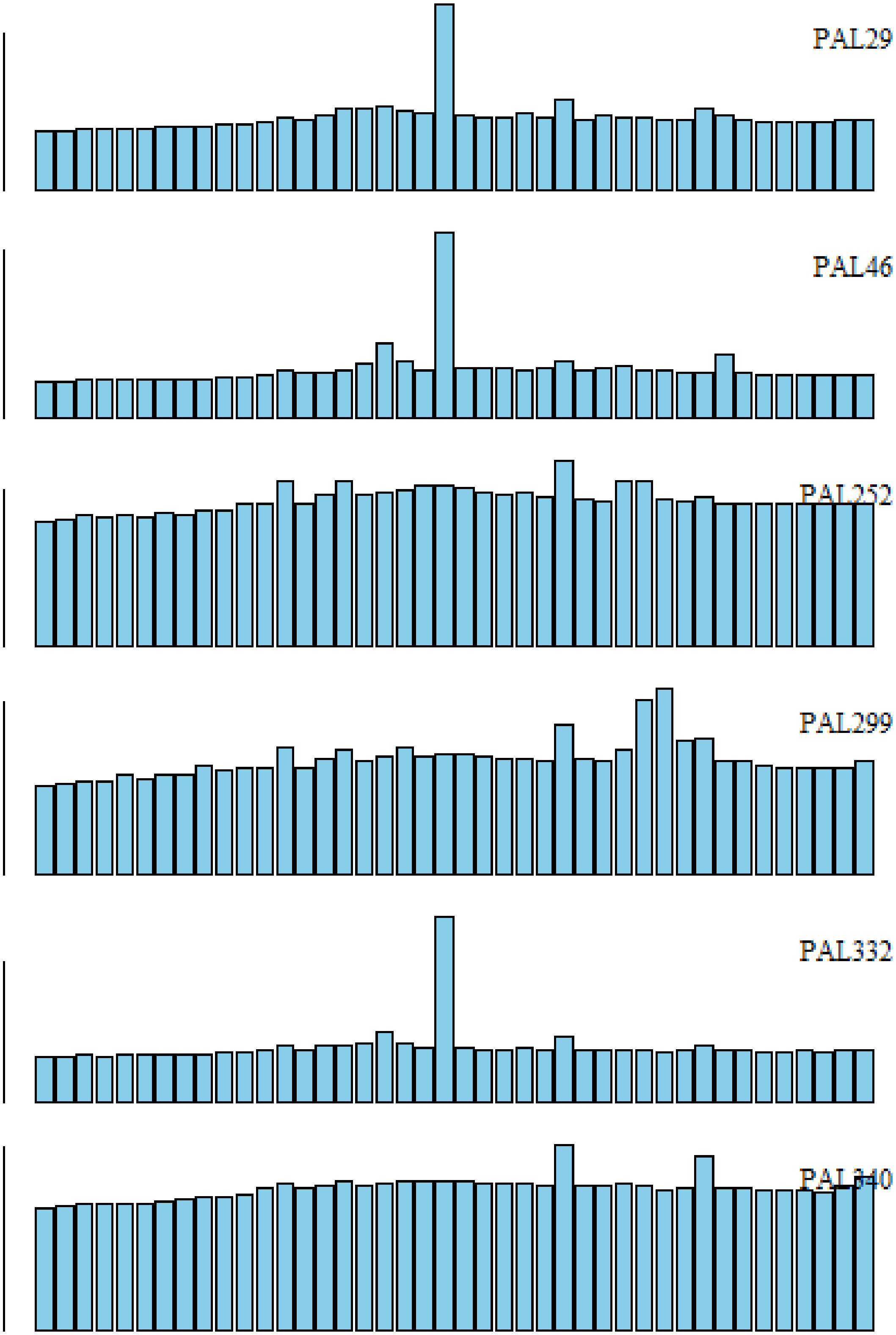


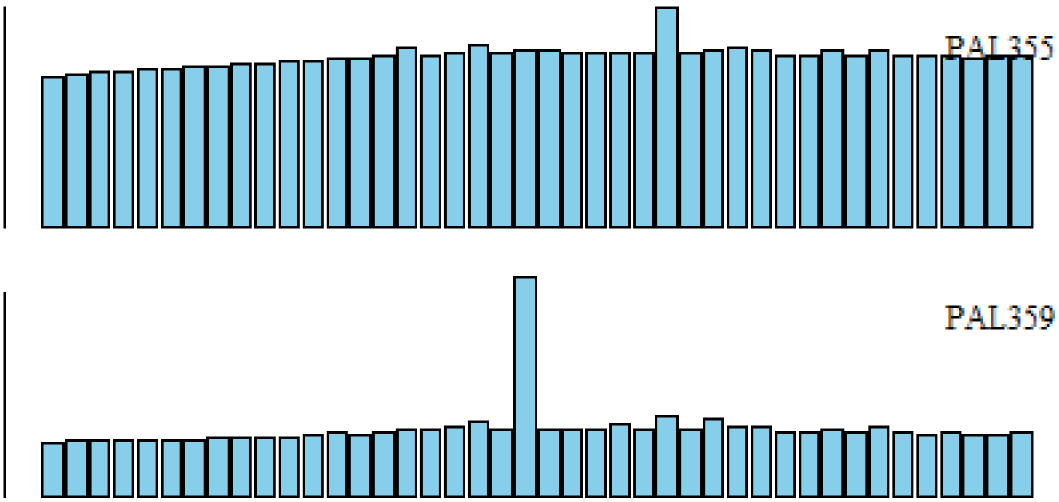



PAL18
PAL43
PAL44
PAL130
PAL131
PAL133

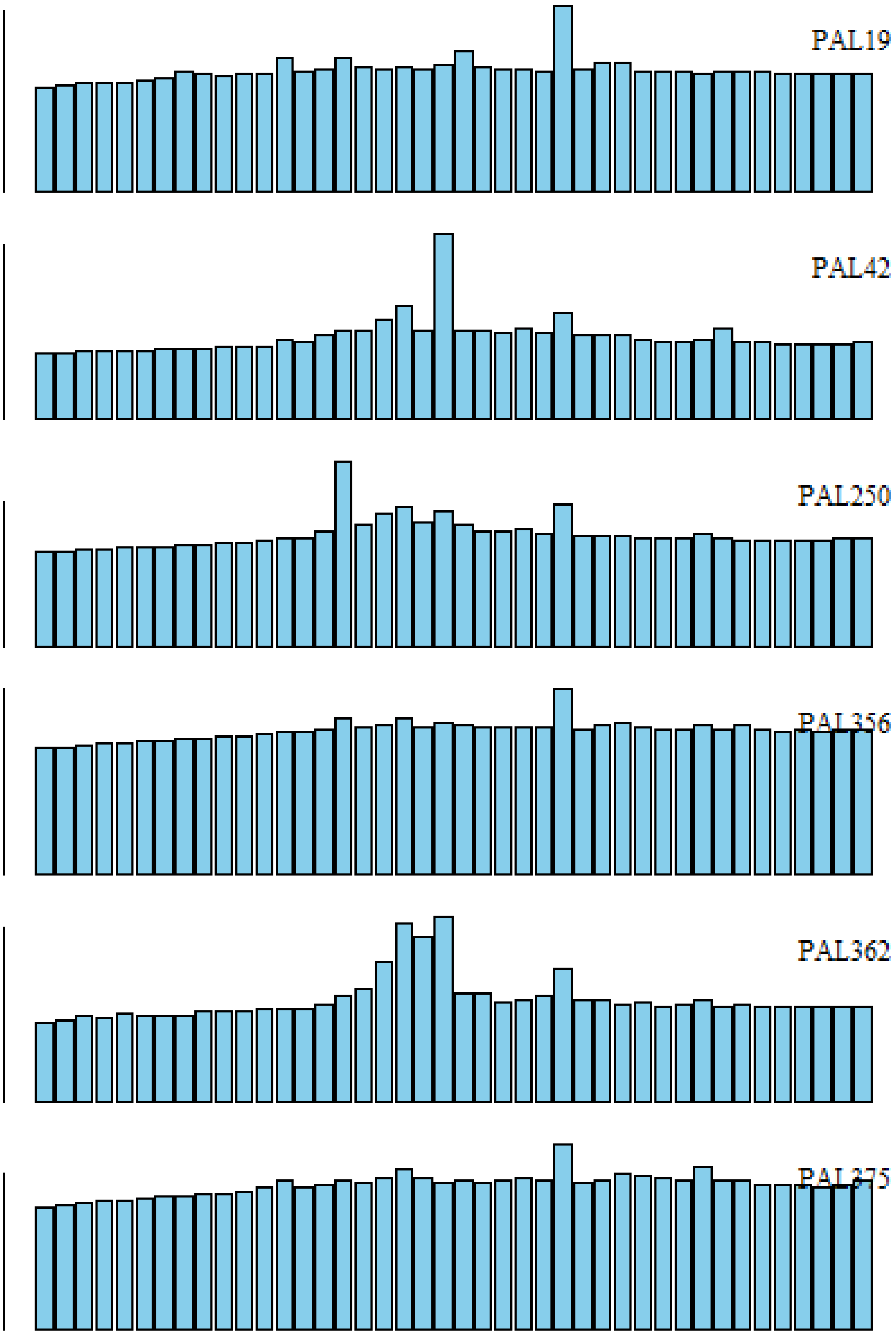

PAL270
PAL333
PAL334
PAL374
PAL378
PAL382

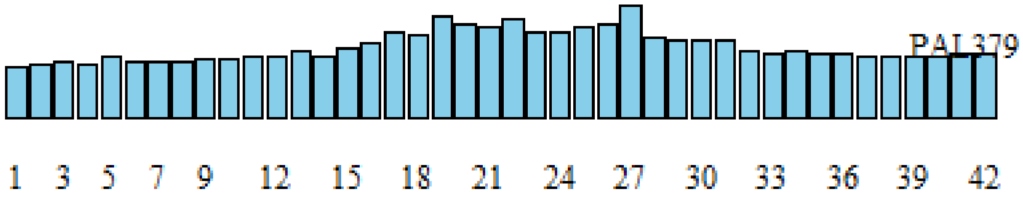
Gas Binned Barplots - Clade D (HEX)




PAL355
PAL359

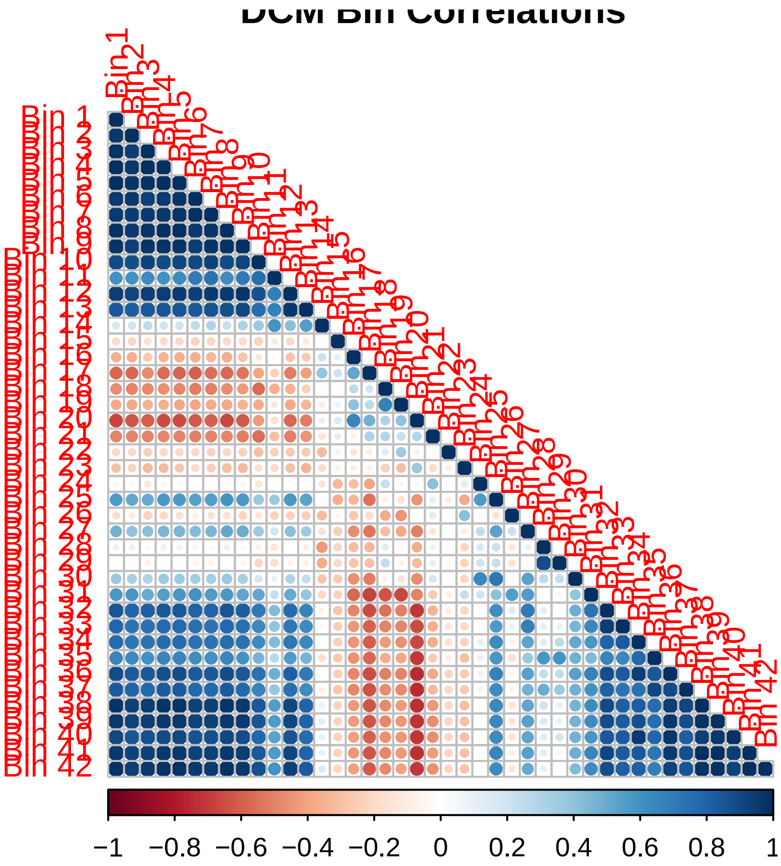
Gas Binned Barplots - Trocheliophorum (HEX)

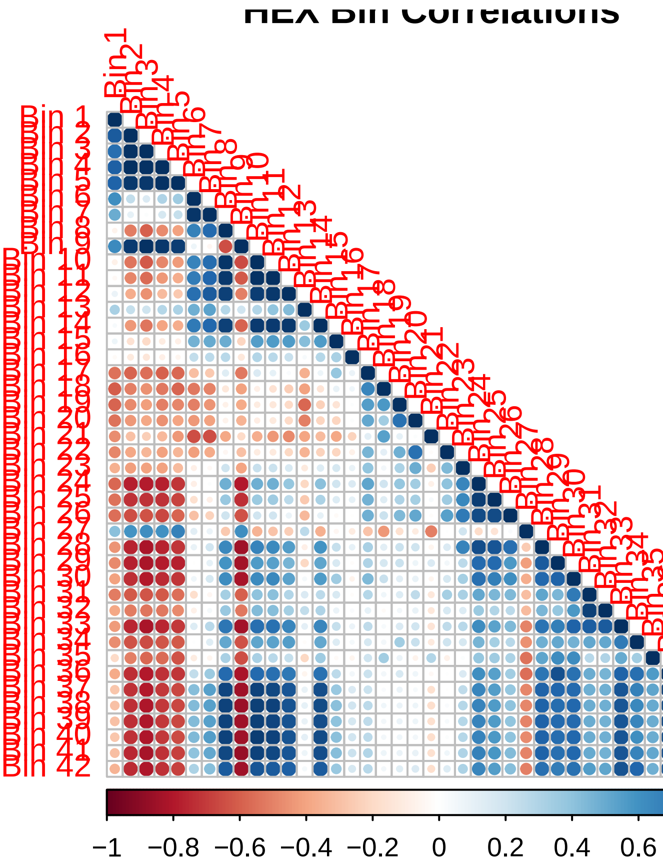



PAI 379
1
3
5
7
9
12
15
18
21
24
27
30
33
36
39
42

Principal Components Analysis (PCA)

As a descriptive measure, we examine the correlation between the bins we use as our explanatory variables.

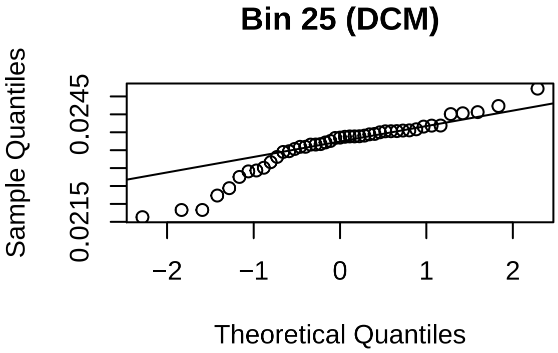


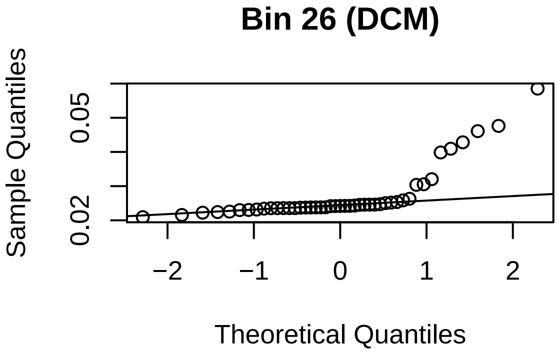


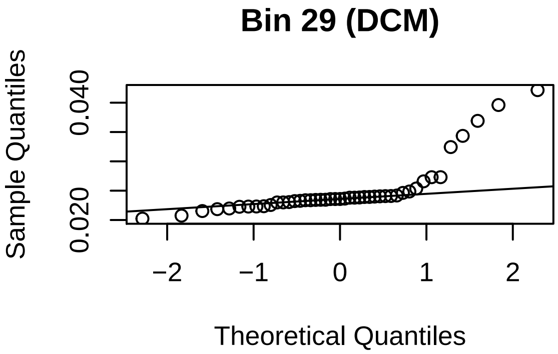
Because we see that there are fairly high correlations between bins in several regions of both correlation plots, we have reason to believe that PCA might be appropriate.

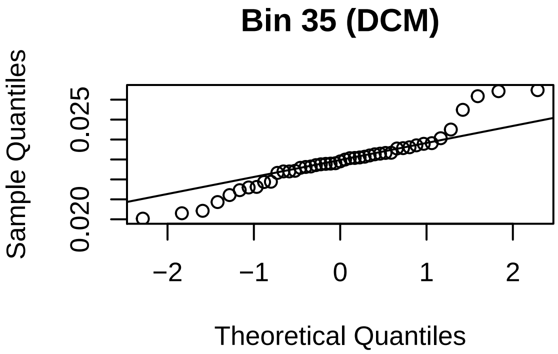
We proceed by validating the assumptions of PCA (as outlined in the Princeton Tutorial on Principal Component Analysis listed below).

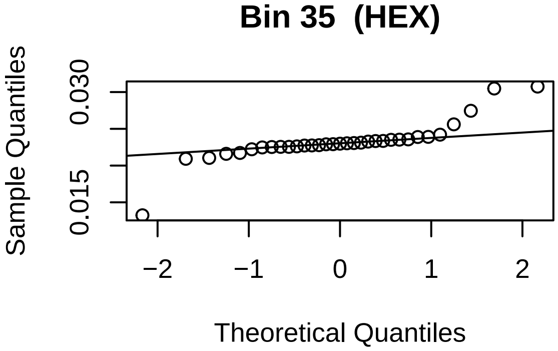
Normal Distribution of Explanatory Variables / QQ plots indicate that the explanatory variables are not normally distributed, which violates one PCA assumption. We show some of these non-normal QQ plots below as examples. PCA tends to be fairly robust to such assumptions, so we proceed cautiously with the analysis.

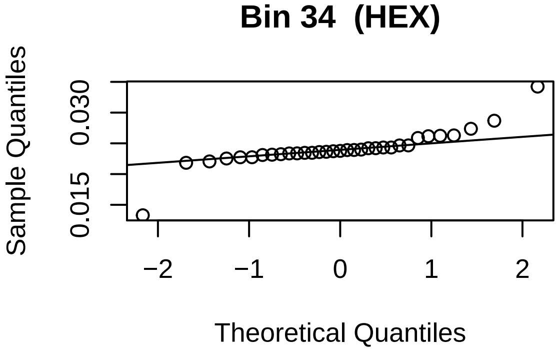


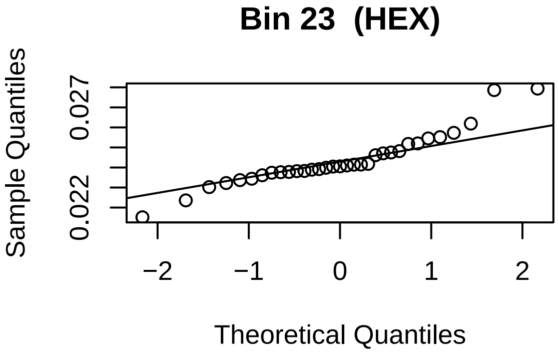


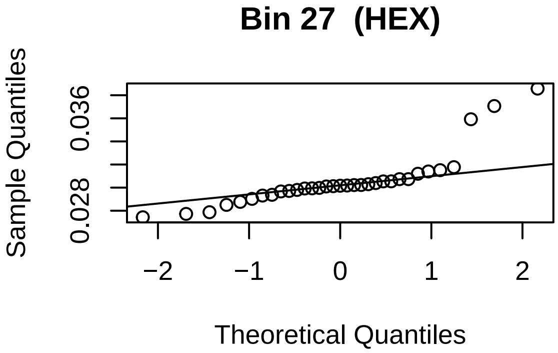












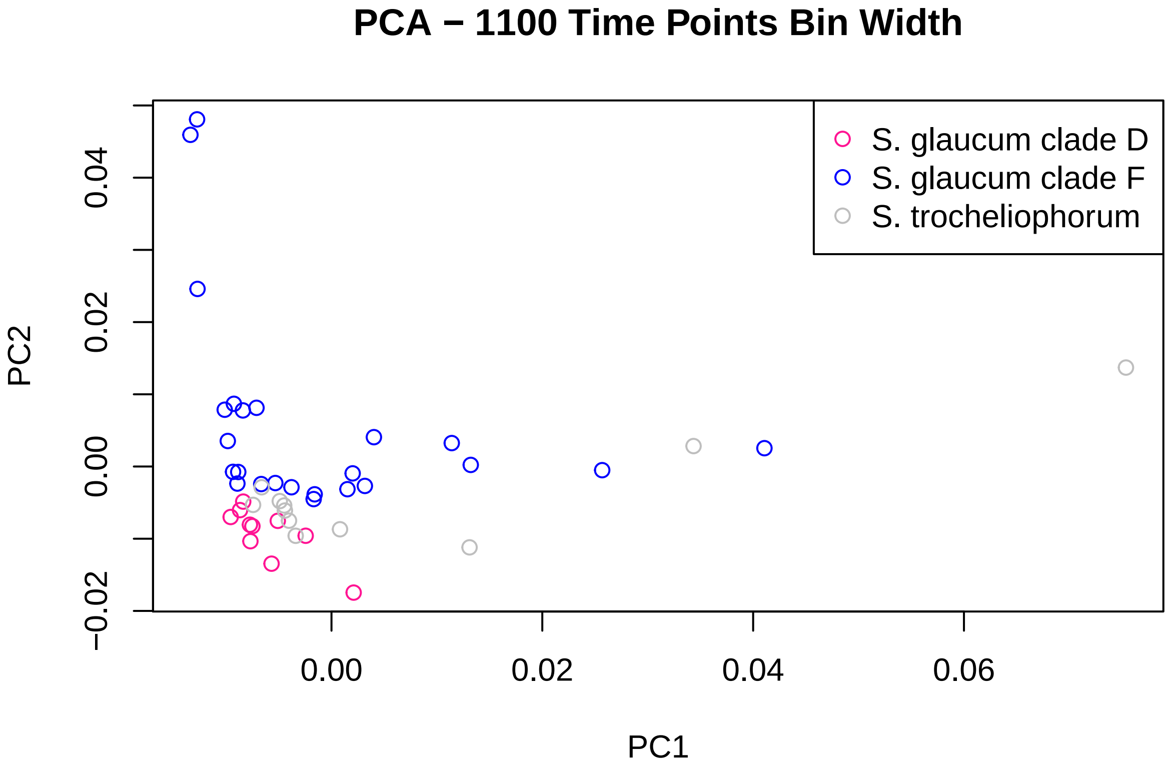
LINEARITY ASSUMPTION?

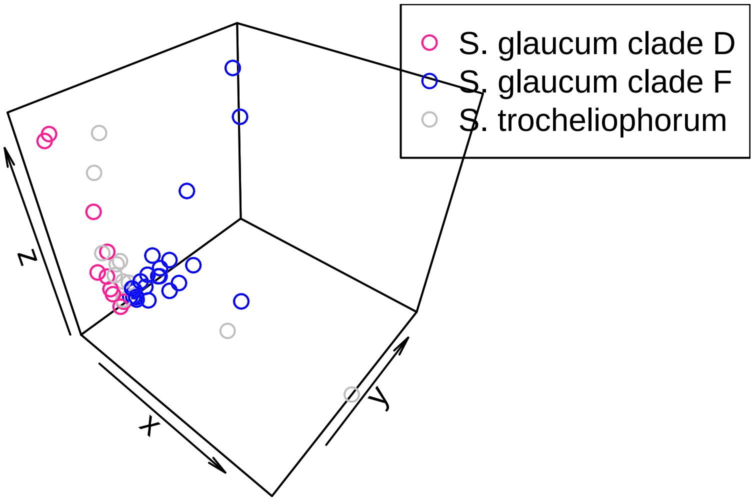
LARGE VARIANCES HAVE IMPORTANT DYNAMICS (SNR)?

ORTHOGONALITY OF PRINCIPAL COMPONENTS?

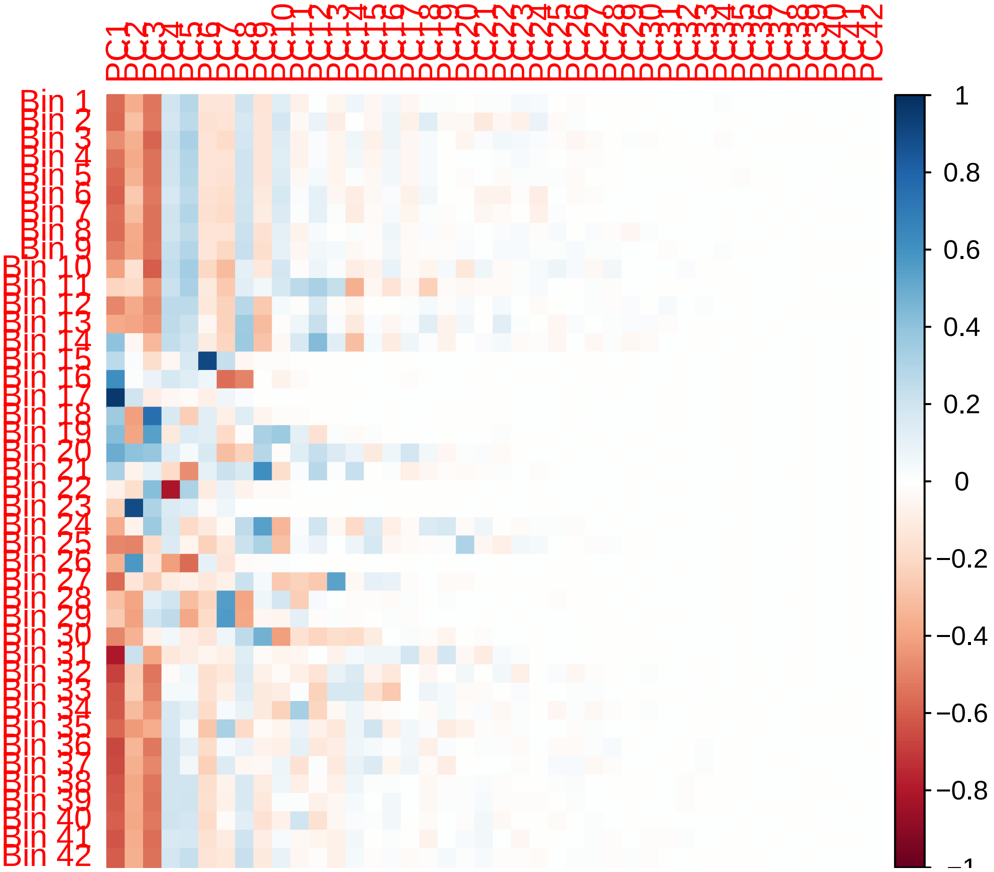
We center the explanatory variables, but do not scale them, since all explanatory variables are measured in the same units.

```r
#DCM:
pca.result_DCM <- prcomp(binned_matrix_DCM, center = TRUE, scale = FALSE)
#str(pca.result)
PC_DCM = pca.result_DCM$x

F = as.numeric(which(dfAllT_DCM[,3]=="F"))
D = as.numeric(which(dfAllT_DCM[,3]=="D"))
T = as.numeric(which(dfAllT_DCM[,3]=="T"))
```





Note the nice separation between Clade F and Clade D given by the PCA plots, even using only the first two principal components.



	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Bin 9	Bin 10	Bin 11	Bin 12
PC1	-0.565	-0.577	-0.463	-0.543	-0.573	-0.599	-0.551	-0.562	-0.506	-0.409	-0.218	-0.489
PC2	-0.363	-0.293	-0.354	-0.379	-0.346	-0.263	-0.302	-0.371	-0.384	-0.150	-0.191	-0.371
PC3	-0.536	-0.524	-0.583	-0.541	-0.542	-0.528	-0.550	-0.547	-0.538	-0.605	-0.442	-0.477

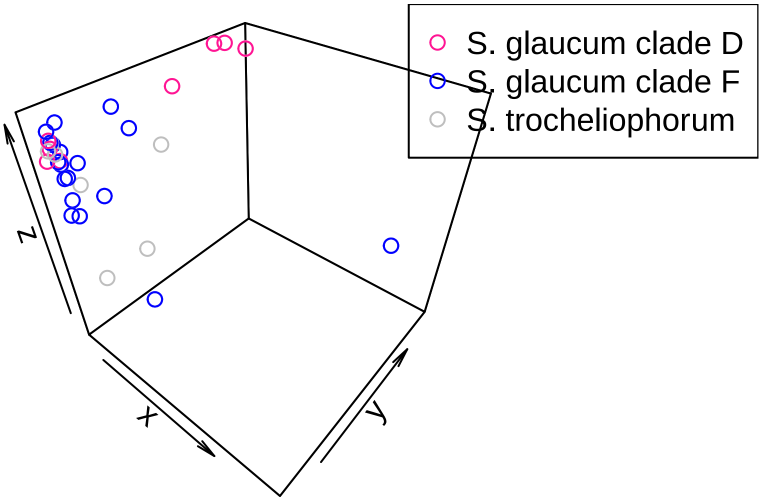
	Bin 13	Bin 14	Bin 15	Bin 16	Bin 17	Bin 18	Bin 19	Bin 20	Bin 21	Bin 22	Bin 23	Bin 24
PC1	-0.374	0.402	0.263	0.618	0.969	0.355	0.427	0.496	0.322	-0.073	-0.238	-0.366
PC2	-0.393	-0.048	0.027	0.012	0.197	-0.413	-0.397	0.406	-0.063	-0.173	0.898	-0.066
PC3	-0.441	-0.325	-0.173	0.080	-0.092	0.758	0.543	0.381	0.108	0.427	0.302	0.361

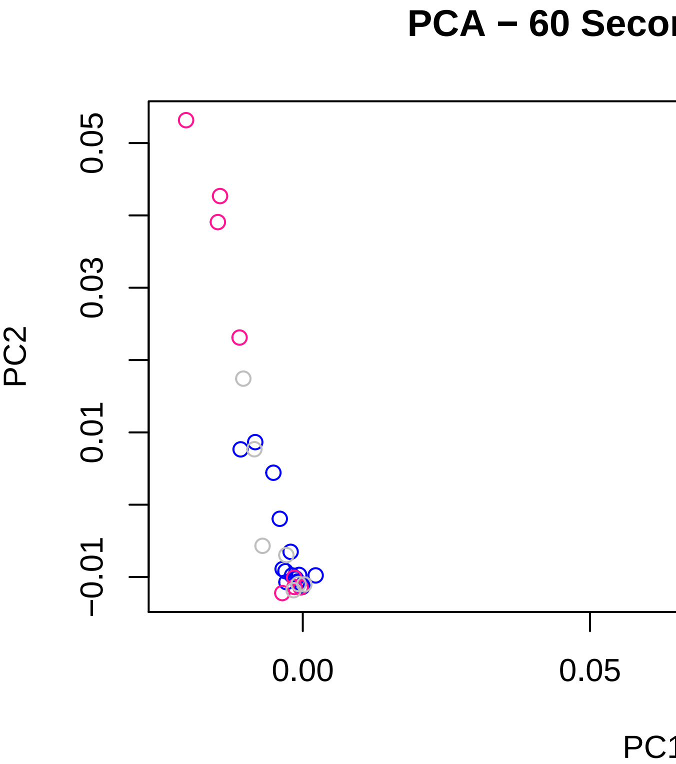
	Bin 25	Bin 26	Bin 27	Bin 28	Bin 29	Bin 30	Bin 31	Bin 32	Bin 33	Bin 34	Bin 35	Bin 36
PC1	-0.486	-0.345	-0.566	-0.299	-0.250	-0.485	-0.802	-0.687	-0.624	-0.614	-0.571	-0.667
PC2	-0.497	0.578	-0.138	-0.394	-0.410	-0.343	0.217	-0.245	-0.238	-0.318	-0.425	-0.337
PC3	-0.185	-0.136	-0.246	0.128	0.195	-0.077	-0.381	-0.534	-0.503	-0.441	-0.363	-0.527

	Bin 37	Bin 38	Bin 39	Bin 40	Bin 41	Bin 42
PC1	-0.650	-0.620	-0.616	-0.593	-0.623	-0.606
PC2	-0.360	-0.388	-0.376	-0.381	-0.360	-0.356
PC3	-0.481	-0.540	-0.550	-0.521	-0.557	-0.541

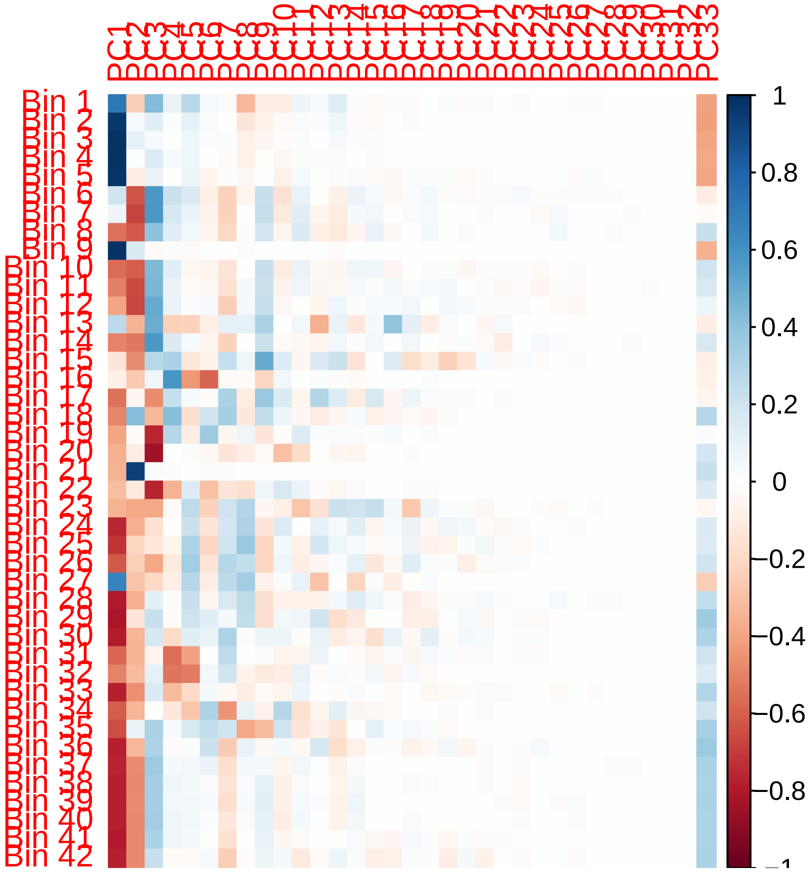
We see high (in absolute value) correlations between the first principal component and bins 31 and 17. Note that in the binned barplots clade D exhibits a greater area in bin 31 as compared to surrounding areas whereas no such jump occurs in clade F. Similarly, clade F has greater area in bin 17.

```
#HEX:
pca.result_HEX <- prcomp(binned_matrix_HEX, center = TRUE, scale = FALSE)
#str(pca.result_HEX)
PC_HEX = pca.result_HEX$x

F = as.numeric(which(dfAllT_HEX[,3]=="F"))
D = as.numeric(which(dfAllT_HEX[,3]=="D"))
T = as.numeric(which(dfAllT_HEX[,3]=="T"))
```





We do not see the separation we would hope to using the Hexane samples. Note that we are dealing with a fairly small sample size (just 7 clade D samples and 18 clade F samples), and PCA may be sensitive to any outliers.



	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Bin 9	Bin 10	Bin 11	Bin 12
PC1	0.718	0.969	0.987	0.982	0.978	0.196	0.079	-0.544	0.986	-0.561	-0.505	-0.397
PC2	-0.241	0.040	0.103	0.008	-0.108	-0.626	-0.676	-0.615	0.156	-0.602	-0.660	-0.664
PC3	0.432	0.128	0.037	0.145	0.095	0.578	0.575	0.404	-0.030	0.445	0.450	0.510

	Bin 13	Bin 14	Bin 15	Bin 16	Bin 17	Bin 18	Bin 19	Bin 20	Bin 21	Bin 22	Bin 23	Bin 24
PC1	0.261	-0.494	-0.137	-0.086	-0.540	-0.487	-0.398	-0.353	-0.341	-0.303	-0.346	-0.753
PC2	-0.342	-0.526	-0.464	-0.269	-0.046	0.430	-0.010	-0.095	0.939	-0.116	-0.382	-0.351
PC3	0.491	0.574	0.272	0.061	-0.466	-0.321	-0.760	-0.843	0.024	-0.763	-0.389	-0.158

	Bin 25	Bin 26	Bin 27	Bin 28	Bin 29	Bin 30	Bin 31	Bin 32	Bin 33	Bin 34	Bin 35	Bin 36
PC1	-0.711	-0.614	0.663	-0.794	-0.818	-0.780	-0.571	-0.497	-0.779	-0.602	-0.647	-0.771
PC2	-0.216	-0.235	-0.288	-0.356	-0.136	-0.339	-0.341	-0.318	-0.450	-0.339	0.083	-0.329
PC3	-0.129	-0.385	-0.206	0.118	0.222	0.179	-0.053	0.104	0.157	-0.001	0.319	0.305

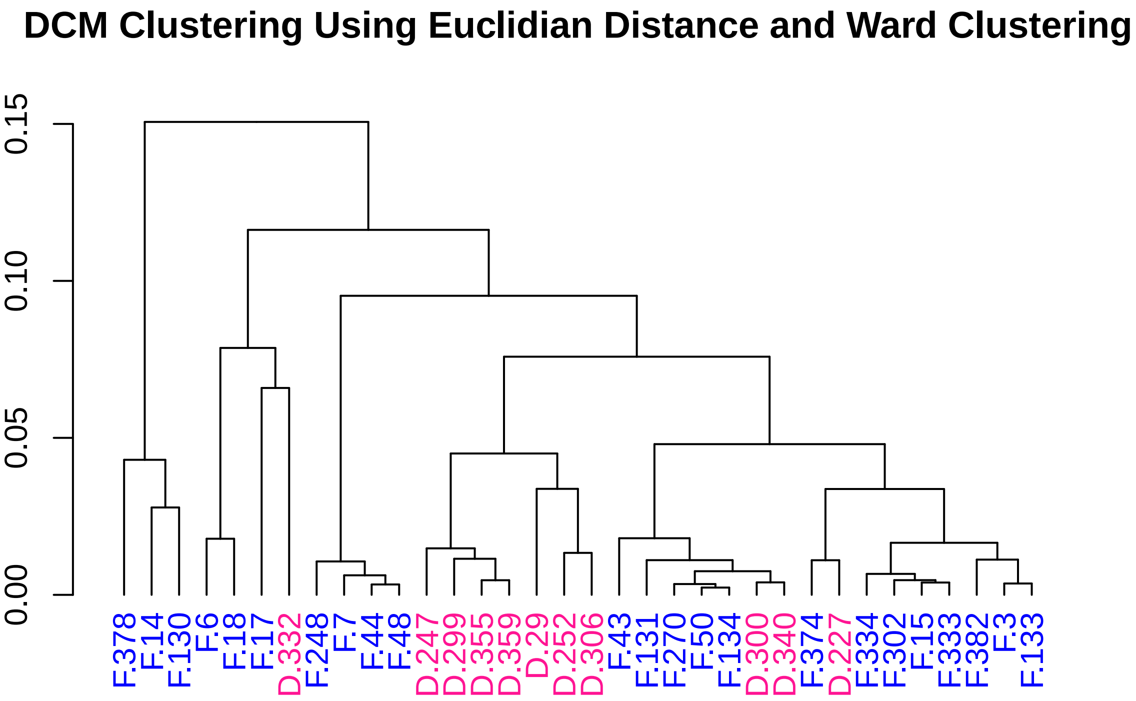
	Bin 37	Bin 38	Bin 39	Bin 40	Bin 41	Bin 42
PC1	-0.770	-0.777	-0.776	-0.772	-0.793	-0.773
PC2	-0.465	-0.479	-0.470	-0.478	-0.473	-0.479
PC3	0.359	0.336	0.333	0.349	0.312	0.228

We see that many bins correlate highly with our first principal component.

Cluster Analysis - Hierarchial Clustering

After trying all possible measures of distance and clustering techniques we see the neatest clusters from using euclidian distance with ward (ward.D minimizing within-cluster standard deviation) clustering for DCM and median clustering for HEX.

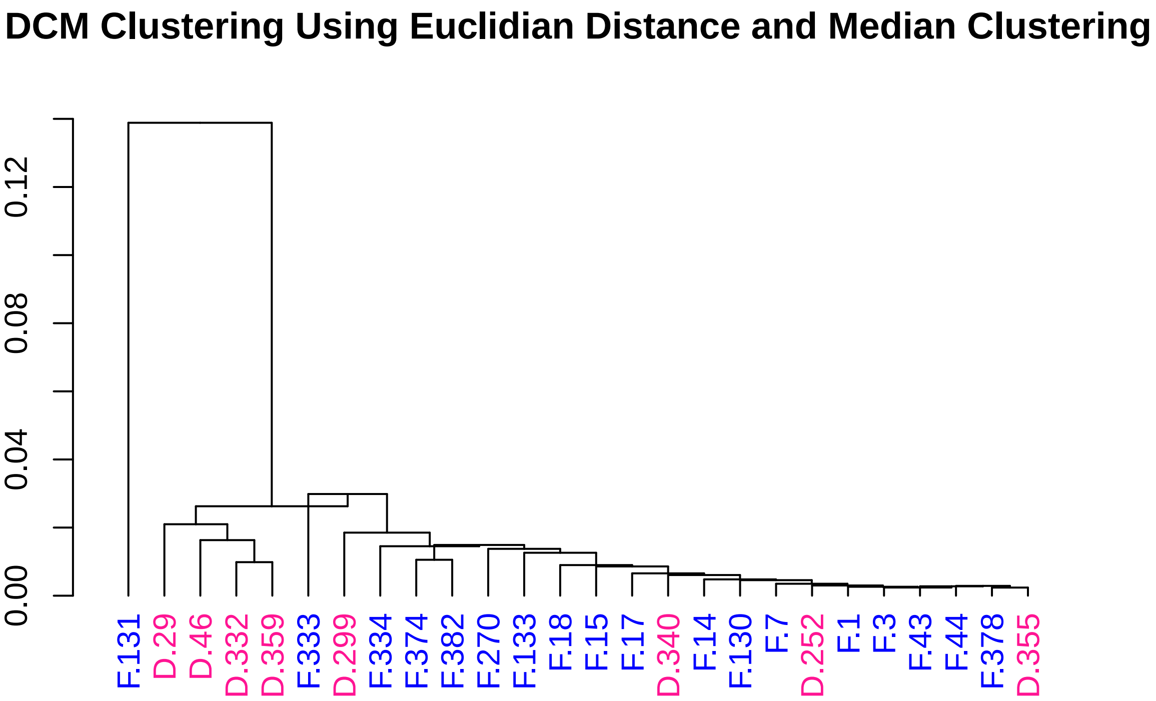
```r
#incorporate clade and sample info to name matrix for cluster
samplesDCM = rownames(binned_matrix_DCM[1:34,])
cladesDCM = c(rep("F", 23), rep("D", 11))
for(j in 1:length(cladesDCM)){
  samplesDCM[j] = paste(cladesDCM[j], ".", samplesDCM[j],sep="")
}
colorCodes <- c(rep("blue", 23), rep("deeppink", 11))

clust_DCM = binned_matrix_DCM[1:34,]
rownames(clust_DCM) = samplesDCM

hcDCM = hclust(dist(clust_DCM, method="euclidian"), method="ward.D")
colLab=c()
for(i in 1:length(hcDCM$order)){
      colLab[i] = colorCodes[hcDCM$order[i]]
    }
dend <- as.dendrogram(hcDCM)
labels_colors(dend) <- colLab
plot(dend, main="DCM Clustering Using Euclidian Distance and Ward Clustering")
```



```r
#incorporate clade and sample info to name matrix for cluster
samplesHEX = rownames(binned_matrix_HEX[1:26,])
cladesHEX = c(rep("F", 18), rep("D", 8))
hcHEX = hclust(dist(clust_HEX, method="euclidian"), method="median")
colLab=c()
for(j in 1:length(cladesHEX)){
  samplesHEX[j] = paste(cladesHEX[j], ".", samplesHEX[j],sep="")
}
colorCodes <- c(rep("blue", 18), rep("deeppink", 8))
```

Note: All but four of our clade D samples are clustered into one group. Samples 300, 400, 227, and 332 remain outside this cluster.

```
clust_HEX = binned_matrix_HEX[1:26,]
rownames(clust_HEX) = samplesHEX

for(i in 1:length(hcHEX$order)){
  colLab[i] = colorCodes[hcHEX$order[i]]
}
dend <- as.dendrogram(hcHEX)
labels_colors(dend) <- colLab
plot(dend, main="DCM Clustering Using Euclidian Distance and Median Clustering")
```



Note: Half of our clade D samples are clustered into one group. Samples 299, 340, 252, and 355 remain outside this cluster.

Linear Discriminant Analysis

```
library(MASS)
#leave-one-out cross validation
lda_cv_DCM = lda(as.factor(dfAllT_DCM[,3])~binned_matrix_DCM, CV=TRUE)
tab_lda_DCM = table(dfAllT_DCM[,3], lda_cv_DCM$class)
percents_lda_DCM = rbind(tab_lda_DCM[1, ]/sum(tab_lda_DCM[1, ]), tab_lda_DCM[2, ]/sum(tab_lda_DCM[2, ])

dimnames(tab_lda_DCM) = list(c("Actual D", "Actual F", "Actual T"), c("Predicted D", "Predicted F", "Pr
dimnames(percents_lda_DCM) = list(c("Actual D", "Actual F", "Actual T"), c("Predicted D", "Predicted F"

kable(tab_lda_DCM)
```

	Predicted D	Predicted F	Predicted T
Actual D	4	4	2
Actual F	10	8	6
Actual T	5	4	2

```
kable(percents_lda_DCM, digits=3, caption="Proportion Actual vs. Predicted (DCM)")
```

Table 16: Proportion Actual vs. Predicted (DCM)

	Predicted D	Predicted F	Predicted T
Actual D	0.400	0.400	0.200
Actual F	0.417	0.333	0.250
Actual T	0.455	0.364	0.182

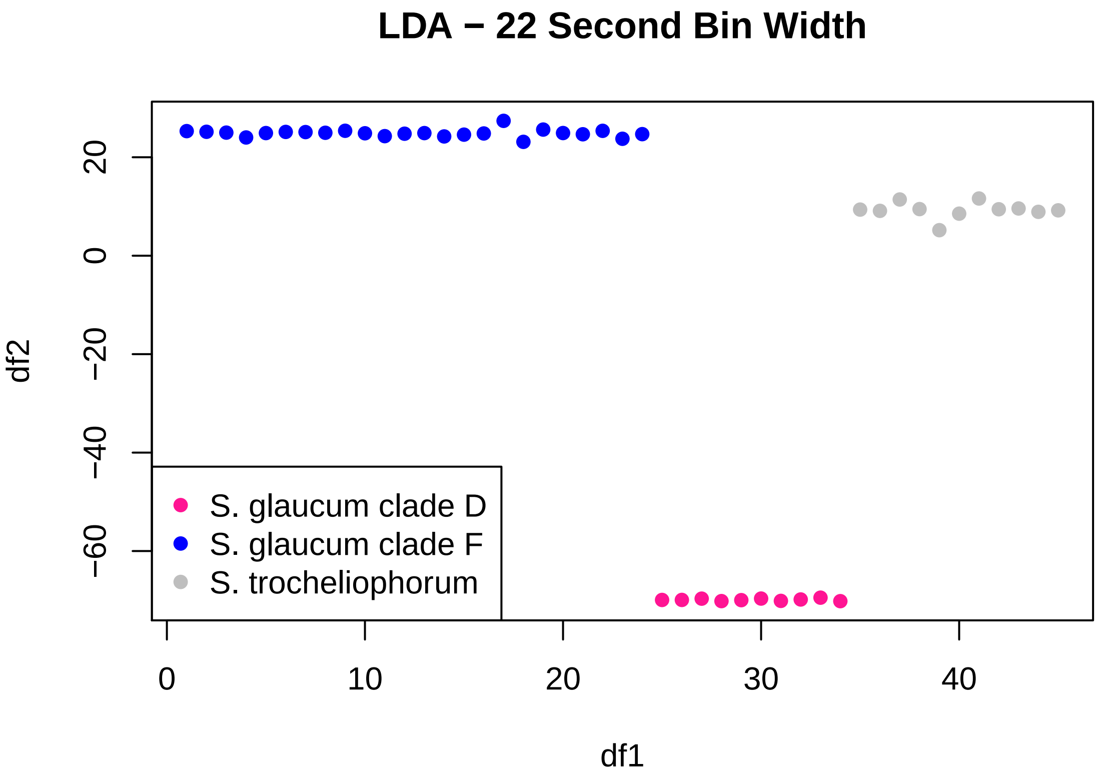
```
lda_info_DCM = lda(as.factor(dfAllT_DCM[,3])~binned_matrix_DCM, CV=FALSE)
kable(lda_info_DCM$scaling, caption="Linear Discriminant Coefficient Matrix (DCM)") #Linear discriminan
```

Table 17: Linear Discriminant Coefficient Matrix (DCM)

	LD1	LD2
binned_matrix_DCMBin 1	338118.906	99315.756
binned_matrix_DCMBin 2	96655.254	5530.078
binned_matrix_DCMBin 3	-72713.468	59890.333
binned_matrix_DCMBin 4	236080.257	-178498.620
binned_matrix_DCMBin 5	482453.371	111271.978
binned_matrix_DCMBin 6	-219227.149	67359.270
binned_matrix_DCMBin 7	521795.491	3425.716
binned_matrix_DCMBin 8	-80731.650	-5514.701
binned_matrix_DCMBin 9	-431862.906	-11138.483
binned_matrix_DCMBin 10	137386.052	3874.717
binned_matrix_DCMBin 11	151492.639	15350.165
binned_matrix_DCMBin 12	-244636.977	-37334.246
binned_matrix_DCMBin 13	287545.425	68582.048
binned_matrix_DCMBin 14	183810.527	27977.576
binned_matrix_DCMBin 15	111354.091	19125.226
binned_matrix_DCMBin 16	120745.589	21271.901
binned_matrix_DCMBin 17	112217.127	19769.655
binned_matrix_DCMBin 18	113006.088	19118.009
binned_matrix_DCMBin 19	121886.086	22997.873
binned_matrix_DCMBin 20	106851.873	18856.893
binned_matrix_DCMBin 21	121444.857	25417.576
binned_matrix_DCMBin 22	118508.534	20647.335
binned_matrix_DCMBin 23	118647.488	20629.286
binned_matrix_DCMBin 24	77475.378	13110.520
binned_matrix_DCMBin 25	132693.260	17858.832
binned_matrix_DCMBin 26	112178.226	18262.945
binned_matrix_DCMBin 27	95435.078	23755.599
binned_matrix_DCMBin 28	1963.333	13014.272
binned_matrix_DCMBin 29	196099.069	26650.563

	LD1	LD2
binned_matrix_DCMBin 30	134250.497	19320.962
binned_matrix_DCMBin 31	172515.856	30249.240
binned_matrix_DCMBin 32	-134798.351	-16661.560
binned_matrix_DCMBin 33	328472.789	40961.901
binned_matrix_DCMBin 34	-192965.938	-12229.748
binned_matrix_DCMBin 35	137487.639	3310.844
binned_matrix_DCMBin 36	-158692.709	21979.107
binned_matrix_DCMBin 37	199771.470	11087.496
binned_matrix_DCMBin 38	-564240.744	166839.648
binned_matrix_DCMBin 39	1641449.597	-8193.577
binned_matrix_DCMBin 40	-316033.010	37553.790
binned_matrix_DCMBin 41	744709.209	94179.233
binned_matrix_DCMBin 42	-171821.531	-56255.773

```
lda_info_DCM_p = predict(lda_info_DCM,as.data.frame(binned_matrix_DCM))
lda_info_DCM_pclass = predict(lda_info_DCM)$class
par(mar=c(5,4,4,4))
v = as.numeric(lda_info_DCM_pclass)
v[which(v==1)] =  "deeppink"   #D
v[which(v==2)] =  "blue"       #F
v[which(v==3)] =  "grey"       #Trochi

plot(lda_info_DCM_p$x[,1],pch=16, col = v,main = "LDA - 22 Second Bin Width",xlab="df1",ylab="df2") #Pl
legend("bottomleft",legend=c("S. glaucum clade D","S. glaucum clade F","S. trocheliophorum"),col=c("dee
```



```
clade.manova = manova(binned_matrix_DCM~as.factor(dfAllT_DCM[,3]))
clade.wilks = summary(clade.manova,test="Wilks"); clade.wilks
```

```
##                             Df      Wilks approx F num Df den Df Pr(>F)
## as.factor(dfAllT_DCM[, 3])  2 1.2226e-05   6.7857     84      2 0.1368
## Residuals                   42
```

```
#HEX:
lda_info_HEX = lda(as.factor(dfAllT_HEX[,3])~binned_matrix_HEX)
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

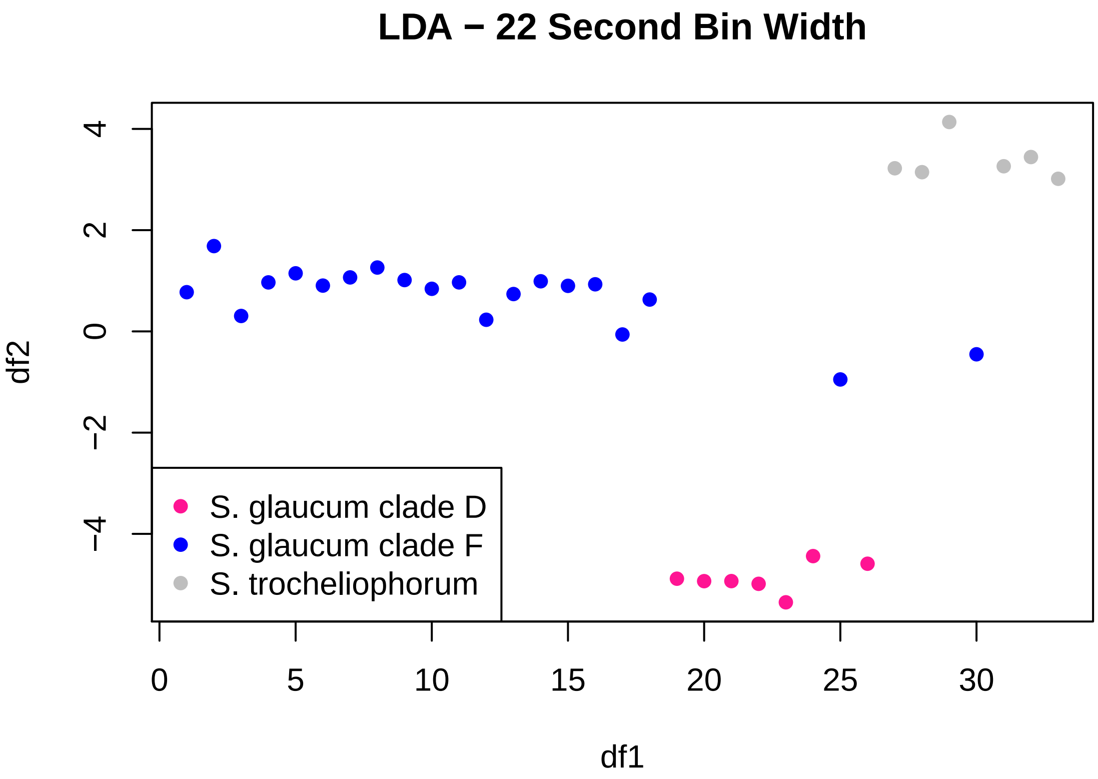
```
kable(lda_info_HEX$scaling, caption="Linear Discriminant Coefficient Matrix (HEX)") #Linear discriminan
```

Table 18: Linear Discriminant Coefficient Matrix (HEX)

	LD1	LD2
binned_matrix_HEXBin 1	878.58956	-2850.11451
binned_matrix_HEXBin 2	-544.73505	100.65267
binned_matrix_HEXBin 3	262.15380	-514.32789
binned_matrix_HEXBin 4	-1884.74018	2113.71820
binned_matrix_HEXBin 5	-6612.27209	3283.66203
binned_matrix_HEXBin 6	8150.38149	-10885.79773
binned_matrix_HEXBin 7	-7379.39458	-5485.38239
binned_matrix_HEXBin 8	-2464.13884	6323.52835
binned_matrix_HEXBin 9	412.67344	51.48666
binned_matrix_HEXBin 10	9376.19699	-11763.08871
binned_matrix_HEXBin 11	-11986.10247	3469.48282
binned_matrix_HEXBin 12	-4139.41912	7209.05362
binned_matrix_HEXBin 13	1177.64944	-1367.37561
binned_matrix_HEXBin 14	8270.64740	106.26470
binned_matrix_HEXBin 15	-3943.49654	1445.88414
binned_matrix_HEXBin 16	58.76898	325.23743
binned_matrix_HEXBin 17	-178.67222	-24.53087
binned_matrix_HEXBin 18	-156.31105	-331.68834
binned_matrix_HEXBin 19	507.30709	500.98613
binned_matrix_HEXBin 20	-966.56438	-886.60766
binned_matrix_HEXBin 21	-544.26046	-118.44455
binned_matrix_HEXBin 22	-378.48907	584.65789
binned_matrix_HEXBin 23	-127.19201	-1489.33043
binned_matrix_HEXBin 24	-2228.94739	-6037.59722
binned_matrix_HEXBin 25	2481.50779	384.31560
binned_matrix_HEXBin 26	-1400.02597	2725.24396
binned_matrix_HEXBin 27	1258.75780	-612.24328
binned_matrix_HEXBin 28	-928.98269	2279.24874
binned_matrix_HEXBin 29	-901.53727	-1588.40625
binned_matrix_HEXBin 30	69.16192	55.41972
binned_matrix_HEXBin 31	1113.93555	-765.21391
binned_matrix_HEXBin 32	-3820.22082	2403.95344
binned_matrix_HEXBin 33	6926.36922	-6237.22837
binned_matrix_HEXBin 34	-466.79591	-2093.77441

	LD1	LD2
binned_matrix_HEXBin 35	-481.05453	140.79634
binned_matrix_HEXBin 36	-38.04730	309.60827
binned_matrix_HEXBin 37	-5503.62153	5577.96482
binned_matrix_HEXBin 38	3819.18605	-2602.26042
binned_matrix_HEXBin 39	-107.95483	7125.10427
binned_matrix_HEXBin 40	5074.25980	-1458.73884
binned_matrix_HEXBin 41	-5406.74199	4402.59496
binned_matrix_HEXBin 42	-1056.89405	-152.73704

```
lda_info_p = predict(lda_info_HEX,as.data.frame(binned_matrix_HEX))
lda_info_pclass = predict(lda_info_HEX)$class
par(mar=c(5,4,4,4))
v = as.numeric(lda_info_pclass)
v[which(v==1)] =  "deeppink"   #D
v[which(v==2)] =  12           #F
v[which(v==3)] =  "grey"       #Trochi

plot(lda_info_p$x[,1],pch=16, col = v,main = "LDA - 22 Second Bin Width",xlab="df1",ylab="df2") #Plot t
legend("bottomleft",legend=c("S. glaucum clade D","S. glaucum clade F","S. trocheliophorum"),col=c("dee
```



```
clade.manova.HEX = manova(binned_matrix_HEX~as.factor(dfAllT_HEX[,3]))
#clade.wilks = summary(clade.manova.HEX,test="Wilks"); clade.wilks
```

The Wilk's Lambda statistic corresponds to a p-value of ~.09, suggesting that the clades account for a large (but only marginally significant) proportion of the variance in binned area.

“‘

```r
binned_matrix_DCM = binner(dfDCM_area, 2000, DELTA)
bins = 1:ncol(binned_matrix_DCM)
for (i in 1:ncol(binned_matrix_DCM)){
  bins[i] = paste("Bin", as.character(i))
}
dimnames(binned_matrix_DCM) = list(dfAllT_DCM[,1], bins)

binned_matrix_HEX = binner(dfHEX_area, 2000, DELTA)
dimnames(binned_matrix_HEX) = list(dfAllT_HEX[,1], bins)
```

Documentation of Resources

Packages Used for Statistical Analysis:

- stats
  - standard deviation
  - pca
  - qqplots
  - correlation
- MASS
  - lda

```
citation("stats")
citation("MASS")
```

Other Packages Used:

- knitr
  - clean table output in markdown file
- corrplot
  - correlation matrix plots
- plot3d
  - 3d PCA plot
- dendextend
  - dendogram with color

```
citation("knitr")
citation("base")
citation("corrplot")
citation("plot3D")
citation("dendextend")
```

PCA Sources:

- http://setosa.io/ev/principal-component-analysis/
  - Provides nice visualization of dimension reduction
- https://www.unt.edu/rss/class/mike/6810/Principal%20Components%20Analysis.pdf
  - Describes difference between PCA and Factor Analysis in powerpoint
- https://www.cs.princeton.edu/picasso/mats/PCA-Tutorial-Intuition_jp.pdf
  - Provides background on PCA
  - Lists assumptions and limitations of PCA
- http://www.floppybunny.org/robin/web/virtualclassroom/stats/statistics2/pca1.pdf

HC Sources:

- http://www.r-tutor.com/gpu-computing/clustering/hierarchical-cluster-analysis
  - R tutorial for hierarchial clustering analysis
- http://stats.stackexchange.com/questions/109949/what-algorithm-does-ward-d-in-hclust-implement-if-it-is-not-wards-cr
  - Describes differences between ward.D and ward.D2 (user response)
- https://en.wikipedia.org/wiki/Ward%27s_method
  - Ward's method
  - Ward's minimizes within cluster variance
- http://www.cs.princeton.edu/courses/archive/spr08/cos424/slides/clustering-2.pdf
  - Nice description of clustering

LDA Sources:

- http://maths-people.anu.edu.au/~johnm/courses/dm/math3346/2008/pdf/r-exercisesVI.pdf
  -