```
library(vegan)

##Read in data files and sample info files for GCP and Colombia
(samples in rows)
gcp <- read.table (file = "GCP_data.txt", header=TRUE,
row.names=1)
gcp.info <- read.table (file = "GCP_sample_info.txt",
header=TRUE, row.names=1)

col <- read.table (file = "Colombian_data.txt", header=TRUE,
row.names=1)
col.info <- read.table (file = "Colombian_sample_info.txt",
header=TRUE, row.names=1)

##Dataset summaries - number of specimens, samples and taxa
sum(gcp)
dim(gcp)
sum(col)
dim(col)

##SAMPLE DATA FOR TABLE 7.1
results <- data.frame(rowSums(gcp), specnumber(gcp),
t(rarefy(gcp, sample = 50, se = TRUE)), t(rarefy(gcp, sample = 300,
se = TRUE)), t(estimateR(gcp)[,2:3]), diversity(gcp, index =
"invsimpson")/specnumber(gcp))
colnames(results) <- c("N", "S", "R50", "R50 s.e.","R300", "R300
s.e.", "Chao1", "Chao1 s.e.", "E1/D")
write.table(results, "Table7.1.txt", sep = "\t", row.names = T,
col.names = T)

##INDIVIDUAL-BASED ANALYSES (RAREFACTION, ESTIMATION AND
EVENNESS)
##GCP
gcp.rare.ind.50 <- rarefy(gcp, 50, se = TRUE)
gcp.rare.ind.300 <- rarefy(gcp, 300, se = TRUE)
gcp.spec.num <- specnumber(gcp)
gcp.ind.num <- rowSums(gcp)
gcp.sim <- diversity(gcp, "simpson")
gcp.invsim <- diversity(gcp, "invsimpson")
gcp.est.ind <- estimateR(gcp)
gcp.results.ind <- rbind(gcp.ind.num, gcp.spec.num,
gcp.rare.ind.50, gcp.rare.ind.300, gcp.est.ind, gcp.sim,
gcp.invsim)
gcp.results.ind <- t(gcp.results.ind)
colnames(gcp.results.ind) <- c("N", "S", "R^50", "R^50.se",
"R^300", "R^300.se", "S", "Chao1", "Chao1.se", "ACE", "ACE.se",
"1-D", "1/D")

##Colombia
col.rare.ind.50 <- rarefy(col, 50, se = TRUE)
col.rare.ind.300 <- rarefy(col, 300, se = TRUE)
col.spec.num <- specnumber(col)
col.ind.num <- rowSums(col)
col.sim <- diversity(col, "simpson")
col.invsim <- diversity(col, "invsimpson")
col.est.ind <- estimateR(col)
col.results.ind <- rbind(col.ind.num, col.spec.num,
col.rare.ind.50, col.rare.ind.300, col.est.ind, col.sim,
col.invsim)
col.results.ind <- t(col.results.ind)
colnames(col.results.ind) <- c("N", "S", "R^50", "R^50.se",
"R^300", "R^300.se", "S", "Chao1", "Chao1.se", "ACE", "ACE.se",
"1-D", "1/D")

##Plotting of results (in composite figures)
results.total <- rbind(col.results, gcp.results)
info.total <- rbind(col.info, gcp.info)
sampling.bins <- c("GCP.MP", "COL.MP", "GCP.LP", "COL.LP",
"GCP.EE", "COL.EE", "GCP.ME", "COL.ME")

##Three boxplots stacked one on top of the other - rare50, chao1,
sim.even.

##Figure 7.1
##Upper plot
par(fig = c(0, 1, 11.1/16.8, 1))
par(mar = c(0, 5, 2, 2))
boxplot(results.total[results.total[,4] != 0,3] ~
info.total[results.total[,4] != 0,4], ylab = "R^50 richness",
boxwex = 0.6, xaxt = "n", las = 1, names = NULL, notch = T)

##Middle plot
par(fig = c(0, 1, 6.4/16.8, 11.1/16.8), new = T)
par(mar = c(0, 5, 0, 2))
boxplot(results.total[,8] ~ info.total[,4], ylab = "Chao1
richness", boxwex = 0.6, xaxt = "n", las = 1, names = NULL, notch
= T)

##Lower plot
par(fig = c(0, 1, 0, 6.4/16.8), new = T)
par(mar = c(3.3, 5, 0, 2))
boxplot(results.total[,13]/results.total[,2] ~ info.total[,4],
ylim = c(0,0.9), ylab = "Evenness: (1/D)/S", boxwex = 0.6, las =
1, names = sampling.bins, notch = T)

##SAMPLE-BASED RAREFACTION
##For GCP and Colombia within each time bin
gcp.MP.specaccum <- specaccum(gcp[gcp.info$Time.bin ==
"GCP.MP",])
gcp.LP.specaccum <- specaccum(gcp[gcp.info$Time.bin ==
"GCP.LP",])
gcp.EE.specaccum <- specaccum(gcp[gcp.info$Time.bin ==
"GCP.EE",])
gcp.ME.specaccum <- specaccum(gcp[gcp.info$Time.bin ==
"GCP.ME",])

gcp.MP.rare <- specaccum(gcp[gcp.info$Time.bin == "GCP.MP",],
method = "rarefaction")
gcp.LP.rare <- specaccum(gcp[gcp.info$Time.bin == "GCP.LP",],
method = "rarefaction")
gcp.EE.rare <- specaccum(gcp[gcp.info$Time.bin == "GCP.EE",],
method = "rarefaction")
gcp.ME.rare <- specaccum(gcp[gcp.info$Time.bin == "GCP.ME",],
method = "rarefaction")

col.MP.specaccum <- specaccum(col[col.info$Time.bin ==
"Col.MP",])
col.LP.specaccum <- specaccum(col[col.info$Time.bin ==
"Col.LP",])
col.EE.specaccum <- specaccum(col[col.info$Time.bin ==
"Col.EE",])
col.ME.specaccum <- specaccum(col[col.info$Time.bin ==
"Col.ME",])

col.MP.rare <- specaccum(col[col.info$Time.bin == "Col.MP",],
method = "rarefaction")
col.LP.rare <- specaccum(col[col.info$Time.bin == "Col.LP",],
method = "rarefaction")
col.EE.rare <- specaccum(col[col.info$Time.bin == "Col.EE",],
method = "rarefaction")
col.ME.rare <- specaccum(col[col.info$Time.bin == "Col.ME",],
method = "rarefaction")

##Figure 7.2A
plot(gcp.MP.rare$invidividuals, gcp.MP.specaccum$richness, type
= "l", xlim = c(0, 20000), ylim = c(0, 150), xlab = "", ylab =
"Within-Age richness", col = "plum")

lines(gcp.LP.rare$invidividuals, gcp.LP.specaccum$richness, col
= "green")
lines(gcp.LP.rare$invidividuals,
gcp.LP.specaccum$richness+1.96*gcp.LP.specaccum$sd, lty = 2, col
= "green")
lines(gcp.LP.rare$invidividuals,
gcp.LP.specaccum$richness-1.96*gcp.LP.specaccum$sd, lty = 2, col
= "green")

lines(gcp.EE.rare$invidividuals, gcp.EE.specaccum$richness, col
= "blue")
lines(gcp.EE.rare$invidividuals,
gcp.EE.specaccum$richness+1.96*gcp.EE.specaccum$sd, lty = 2, col
= "blue")
lines(gcp.EE.rare$invidividuals,
gcp.EE.specaccum$richness-1.96*gcp.EE.specaccum$sd, lty = 2, col
= "blue")

lines(gcp.ME.rare$invidividuals, gcp.ME.specaccum$richness, col
= "red")

##Figure 7.2B
plot(col.MP.rare$invidividuals, col.MP.specaccum$richness, type
= "l", xlim = c(0, 20000), ylim = c(0, 500), xlab = "Number of
specimens (based on mean number of specimens per sample)", ylab
= "Within-Age richness", col = "plum")
lines(col.MP.rare$invidividuals,
col.MP.specaccum$richness+1.96*col.MP.specaccum$sd, lty = 2, col
= "plum")
lines(col.MP.rare$invidividuals,
col.MP.specaccum$richness-1.96*col.MP.specaccum$sd, lty = 2, col
= "plum")

lines(col.LP.rare$invidividuals, col.LP.specaccum$richness, col
= "green")

lines(col.EE.rare$invidividuals, col.EE.specaccum$richness, col
= "blue")

lines(col.ME.rare$invidividuals, col.ME.specaccum$richness, col
= "red")
lines(col.MEE.rare$invidividuals,
col.ME.specaccum$richness+1.96*col.ME.specaccum$sd, lty = 2, col
= "red")
lines(col.ME.rare$invidividuals,
col.ME.specaccum$richness-1.96*col.ME.specaccum$sd, lty = 2, col
= "red")

##For GCP and Colombia, by locality within each time bin
gcp.MP <- gcp[gcp.info$Time.bin == "GCP.MP",]
gcp.LP <- gcp[gcp.info$Time.bin == "GCP.LP",]
gcp.EE <- gcp[gcp.info$Time.bin == "GCP.EE",]
gcp.ME <- gcp[gcp.info$Time.bin == "GCP.ME",]

gcp.info.MP <- gcp.info[gcp.info$Time.bin == "GCP.MP",]
gcp.info.LP <- gcp.info[gcp.info$Time.bin == "GCP.LP",]
gcp.info.EE <- gcp.info[gcp.info$Time.bin == "GCP.EE",]
gcp.info.ME <- gcp.info[gcp.info$Time.bin == "GCP.ME",]

col.MP <- col[col.info$Time.bin == "Col.MP",]
col.LP <- col[col.info$Time.bin == "Col.LP",]
col.EE <- col[col.info$Time.bin == "Col.EE",]
col.ME <- col[col.info$Time.bin == "Col.ME",]

col.info.MP <- col.info[col.info$Time.bin == "Col.MP",]
col.info.LP <- col.info[col.info$Time.bin == "Col.LP",]
col.info.EE <- col.info[col.info$Time.bin == "Col.EE",]
col.info.ME <- col.info[col.info$Time.bin == "Col.ME",]

##Middle Paleocene
gcp.cores.MP <- unique(gcp.info.MP$Core)
gcp.rare.MP.exact <- vector("list", length =
length(gcp.cores.MP))
gcp.rare.MP.ind <- vector("list", length = length(gcp.cores.MP))
for (i in 1:length(gcp.cores.MP)) gcp.rare.MP.exact[[i]] <-
specaccum(gcp.MP[gcp.info.MP$Core == gcp.cores.MP[i],])
for (i in 1:length(gcp.cores.MP)) gcp.rare.MP.ind[[i]] <-
specaccum(gcp.MP[gcp.info.MP$Core == gcp.cores.MP[i],], method =
"rarefaction")

col.cores.MP <- unique(col.info.MP$Core)
col.rare.MP.exact <- vector("list", length =
length(col.cores.MP))
col.rare.MP.ind <- vector("list", length = length(col.cores.MP))
for (i in 1:length(col.cores.MP)) col.rare.MP.exact[[i]] <-
specaccum(col.MP[col.info.MP$Core == col.cores.MP[i],])
for (i in 1:length(col.cores.MP)) col.rare.MP.ind[[i]] <-
specaccum(col.MP[col.info.MP$Core == col.cores.MP[i],], method =
"rarefaction")

##Late Paleocene
gcp.cores.LP <- unique(gcp.info.LP$Core)
gcp.rare.LP.exact <- vector("list", length =
length(gcp.cores.LP))
gcp.rare.LP.ind <- vector("list", length = length(gcp.cores.LP))
for (i in 1:length(gcp.cores.LP)) gcp.rare.LP.exact[[i]] <-
specaccum(gcp.LP[gcp.info.LP$Core == gcp.cores.LP[i],])
for (i in 1:length(gcp.cores.LP)) gcp.rare.LP.ind[[i]] <-
specaccum(gcp.LP[gcp.info.LP$Core == gcp.cores.LP[i],], method =
"rarefaction")

col.cores.LP <- unique(col.info.LP$Core)
col.rare.LP.exact <- vector("list", length =
length(col.cores.LP))
col.rare.LP.ind <- vector("list", length = length(col.cores.LP))
for (i in 1:length(col.cores.LP)) col.rare.LP.exact[[i]] <-
specaccum(col.LP[col.info.LP$Core == col.cores.LP[i],])
for (i in 1:length(col.cores.LP)) col.rare.LP.ind[[i]] <-
specaccum(col.LP[col.info.LP$Core == col.cores.LP[i],], method =
"rarefaction")

##Early Eocene
gcp.cores.EE <- unique(gcp.info.EE$Core)
gcp.rare.EE.exact <- vector("list", length =
length(gcp.cores.EE))
gcp.rare.EE.ind <- vector("list", length = length(gcp.cores.EE))
for (i in 1:length(gcp.cores.EE)) gcp.rare.EE.exact[[i]] <-
```

```
specaccum(gcp.EE[gcp.info.EE$Core == gcp.cores.EE[i],])
for (i in 1:length(gcp.cores.EE)) gcp.rare.EE.ind[[i]] <-
specaccum(gcp.EE[gcp.info.EE$Core == gcp.cores.EE[i],], method =
"rarefaction")

col.cores.EE <- unique(col.info.EE$Core)
col.rare.EE.exact <- vector("list", length =
length(col.cores.EE))
col.rare.EE.ind <- vector("list", length = length(col.cores.EE))
for (i in 1:length(col.cores.EE)) col.rare.EE.exact[[i]] <-
specaccum(col.EE[col.info.EE$Core == col.cores.EE[i],])
for (i in 1:length(col.cores.EE)) col.rare.EE.ind[[i]] <-
specaccum(col.EE[col.info.EE$Core == col.cores.EE[i],], method =
"rarefaction")

##Middle Eocene
gcp.cores.ME <- unique(gcp.info.ME$Core)
gcp.rare.ME.exact <- vector("list", length =
length(gcp.cores.ME))
gcp.rare.ME.ind <- vector("list", length = length(gcp.cores.ME))
for (i in 1:length(gcp.cores.ME)) gcp.rare.ME.exact[[i]] <-
specaccum(gcp.ME[gcp.info.ME$Core == gcp.cores.ME[i],])
for (i in 1:length(gcp.cores.ME)) gcp.rare.ME.ind[[i]] <-
specaccum(gcp.ME[gcp.info.ME$Core == gcp.cores.ME[i],], method =
"rarefaction")

col.cores.ME <- unique(col.info.ME$Core)
col.rare.ME.exact <- vector("list", length =
length(col.cores.ME))
col.rare.ME.ind <- vector("list", length = length(col.cores.ME))
for (i in 1:length(col.cores.ME)) col.rare.ME.exact[[i]] <-
specaccum(col.ME[col.info.ME$Core == col.cores.ME[i],])
for (i in 1:length(col.cores.ME)) col.rare.ME.ind[[i]] <-
specaccum(col.ME[col.info.ME$Core == col.cores.ME[i],], method =
"rarefaction")

##Figure 7.3
par(mfrow = c(2,2))
par(oma = c(3,2,0,0))
par(mar = c(2, 3, 2, 1))
plot(seq(0, 5000, 500), seq(0, 250, 25), type = "n", xlab = "",
ylab = "", main = "Middle Paleocene sites")
for (i in 1:length(col.cores.MP))
lines(col.rare.MP.ind[[i]]$invidividuals,
col.rare.MP.exact[[i]]$richness, col = "grey40", lty = 2)
for (i in 1:length(gcp.cores.MP))
lines(gcp.rare.MP.ind[[i]]$invidividuals,
gcp.rare.MP.exact[[i]]$richness)

par(mar = c(2, 3, 2, 1))
plot(seq(0, 12000, 1200), seq(0, 250, 25), type = "n", xlab = "",
ylab = "", main = "Late Paleocene sites")
for (i in 1:length(col.cores.LP))
lines(col.rare.LP.ind[[i]]$invidividuals,
col.rare.LP.exact[[i]]$richness, col = "grey40", lty = 2)
for (i in 1:length(gcp.cores.LP))
lines(gcp.rare.LP.ind[[i]]$invidividuals,
gcp.rare.LP.exact[[i]]$richness)

par(mar = c(2, 3, 2, 1))
plot(seq(0, 5000, 500), seq(0, 250, 25), type = "n", xlab = "",
ylab = "", main = "Early Eocene sites")
for (i in 1:length(col.cores.EE))
lines(col.rare.EE.ind[[i]]$invidividuals,
col.rare.EE.exact[[i]]$richness, col = "grey40", lty = 2)
for (i in 1:length(gcp.cores.EE))
lines(gcp.rare.EE.ind[[i]]$invidividuals,
gcp.rare.EE.exact[[i]]$richness)

par(mar = c(2, 3, 2, 1))
plot(seq(0, 5000, 500), seq(0, 250, 25), type = "n", xlab = "",
ylab = "", main = "Middle Eocene sites")
for (i in 1:length(col.cores.ME))
lines(col.rare.ME.ind[[i]]$invidividuals,
col.rare.ME.exact[[i]]$richness, col = "grey40", lty = 2)
for (i in 1:length(gcp.cores.ME))
lines(gcp.rare.ME.ind[[i]]$invidividuals,
gcp.rare.ME.exact[[i]]$richness)

title(xlab = "Number of individuals (based on mean number of
individuals per sample)", ylab =  "Species richness", outer = T,
line = 1)

##INCIDENCE-BASED RICHNESS ESTIMATION - CHAO2
##For GCP and Colombia within each time bin
gcp.est.MP <- poolaccum(gcp.MP)
gcp.est.LP <- poolaccum(gcp.LP)
gcp.est.EE <- poolaccum(gcp.EE)
gcp.est.ME <- poolaccum(gcp.ME)

gcp.chao.MP <- summary(gcp.est.MP, display = "chao")
gcp.chao.LP <- summary(gcp.est.LP, display = "chao")
gcp.chao.EE <- summary(gcp.est.EE, display = "chao")
gcp.chao.ME <- summary(gcp.est.ME, display = "chao")

col.est.MP <- poolaccum(col.MP)
col.est.LP <- poolaccum(col.LP)
col.est.EE <- poolaccum(col.EE)
col.est.ME <- poolaccum(col.ME)

col.chao.MP <- summary(col.est.MP, display = "chao")
col.chao.LP <- summary(col.est.LP, display = "chao")
col.chao.EE <- summary(col.est.EE, display = "chao")
col.chao.ME <- summary(col.est.ME, display = "chao")

##Figure 7.4A
plot(gcp.chao.MP[[1]][,1], gcp.chao.MP[[1]][,2], type = "l", xlim
= c(0, 60), ylim = c(0, 220), xlab = "", ylab = "Chao2 estimated
species richness", col = "plum")
lines(gcp.chao.MP[[1]][,1], gcp.chao.MP[[1]][,3], lty = 2, col =
"plum")
lines(gcp.chao.MP[[1]][,1], gcp.chao.MP[[1]][,4], lty = 2, col =
"plum")

lines(gcp.chao.LP[[1]][,1], gcp.chao.LP[[1]][,2], col = "green")
lines(gcp.chao.LP[[1]][,1], gcp.chao.LP[[1]][,3], lty = 2, col =
"green")
lines(gcp.chao.LP[[1]][,1], gcp.chao.LP[[1]][,4], lty = 2, col =
"green")

lines(gcp.chao.EE[[1]][,1], gcp.chao.EE[[1]][,2], col = "blue")
lines(gcp.chao.EE[[1]][,1], gcp.chao.EE[[1]][,3], lty = 2, col =
"blue")
lines(gcp.chao.EE[[1]][,1], gcp.chao.EE[[1]][,4], lty = 2, col =
"blue")

lines(gcp.chao.ME[[1]][,1], gcp.chao.ME[[1]][,2], col = "red")
lines(gcp.chao.ME[[1]][,1], gcp.chao.ME[[1]][,3], lty = 2, col =
"red")
lines(gcp.chao.ME[[1]][,1], gcp.chao.ME[[1]][,4], lty = 2, col =
"red")

##Figure 7.4B
plot(col.chao.MP[[1]][,1], col.chao.MP[[1]][,2], type = "l", xlim
= c(0, 120), ylim = c(0, 900), xlab = "Number of samples", ylab
= "Chao2 estimated species richness", col = "plum")
lines(col.chao.MP[[1]][,1], col.chao.MP[[1]][,3], lty = 2, col =
"plum")
lines(col.chao.MP[[1]][,1], col.chao.MP[[1]][,4], lty = 2, col =
"plum")

lines(col.chao.LP[[1]][,1], col.chao.LP[[1]][,2], col = "green")
lines(col.chao.LP[[1]][,1], col.chao.LP[[1]][,3], lty = 2, col =
"green")
lines(col.chao.LP[[1]][,1], col.chao.LP[[1]][,4], lty = 2, col =
"green")

lines(col.chao.EE[[1]][,1], col.chao.EE[[1]][,2], col = "blue")
lines(col.chao.EE[[1]][,1], col.chao.EE[[1]][,3], lty = 2, col =
"blue")
lines(col.chao.EE[[1]][,1], col.chao.EE[[1]][,4], lty = 2, col =
"blue")
```

```
lines(col.chao.ME[[1]][,1], col.chao.ME[[1]][,2], col = "red")
lines(col.chao.ME[[1]][,1], col.chao.ME[[1]][,3], lty = 2, col =
"red")
lines(col.chao.ME[[1]][,1], col.chao.ME[[1]][,4], lty = 2, col =
"red")

##For GCP and Colombia, by locality within each time bin
##Middle Paleocene
gcp.est.MP.core <- vector("list", length = length(gcp.cores.MP))
for (i in 1:length(gcp.cores.MP)) gcp.est.MP.core[[i]] <-
poolaccum(gcp.MP[gcp.info.MP$Core == gcp.cores.MP[i],], minsize
= 1)
col.est.MP.core <- vector("list", length = length(col.cores.MP))
for (i in 1:length(col.cores.MP)) col.est.MP.core[[i]] <-
poolaccum(col.MP[col.info.MP$Core == col.cores.MP[i],], minsize
= 1)

##Late Paleocene
gcp.est.LP.core <- vector("list", length = length(gcp.cores.LP))
for (i in 1:length(gcp.cores.LP)) gcp.est.LP.core[[i]] <-
poolaccum(gcp.LP[gcp.info.LP$Core == gcp.cores.LP[i],], minsize
= 1)
col.est.LP.core <- vector("list", length = length(col.cores.LP))
for (i in 1:length(col.cores.LP)) col.est.LP.core[[i]] <-
poolaccum(col.LP[col.info.LP$Core == col.cores.LP[i],], minsize
= 1)

##Early Eocene - NB have to remove Colombian cores with only one
sample
gcp.est.EE.core <- vector("list", length = length(gcp.cores.EE))
for (i in 1:length(gcp.cores.EE)) gcp.est.EE.core[[i]] <-
poolaccum(gcp.EE[gcp.info.EE$Core == gcp.cores.EE[i],], minsize
= 1)

col.cores.EE <- col.cores.EE[-c(3, 4, 7)]
col.est.EE.core <- vector("list", length = length(col.cores.EE))
for (i in 1:length(col.cores.EE)) col.est.EE.core[[i]] <-
poolaccum(col.EE[col.info.EE$Core == col.cores.EE[i],], minsize
= 1)

##Middle Eocene - NB have to remove Colombian cores with only one
sample
gcp.est.ME.core <- vector("list", length = length(gcp.cores.ME))
for (i in 1:length(gcp.cores.ME)) gcp.est.ME.core[[i]] <-
poolaccum(gcp.ME[gcp.info.ME$Core == gcp.cores.ME[i],], minsize
= 1)

col.cores.ME <- col.cores.ME[-2]
col.est.ME.core <- vector("list", length = length(col.cores.ME))
for (i in 1:length(col.cores.ME)) col.est.ME.core[[i]] <-
poolaccum(col.ME[col.info.ME$Core == col.cores.ME[i],], minsize
= 1)

##Figure 7.5
par(mfrow = c(2,2))
par(oma = c(3,2,3,0))
par(mar = c(2, 3, 2, 1))

plot(seq(0, 40, 4), seq(0, 500, 50), type = "n", xlab = "", ylab
= "", main = "Middle Paleocene sites")
for (i in 1:length(col.cores.MP))
lines(summary(col.est.MP.core[[i]], display = "chao")[[1]][,1],
summary(col.est.MP.core[[i]], display = "chao")[[1]][,2], col =
"grey40", lty = 2)
for (i in 1:length(gcp.cores.MP))
lines(summary(gcp.est.MP.core[[i]], display = "chao")[[1]][,1],
summary(gcp.est.MP.core[[i]], display = "chao")[[1]][,2])

plot(seq(0, 40, 4), seq(0, 500, 50), type = "n", xlab = "", ylab
= "", main = "Late Paleocene sites")
for (i in 1:length(col.cores.LP))
lines(summary(col.est.LP.core[[i]], display = "chao")[[1]][,1],
summary(col.est.LP.core[[i]], display = "chao")[[1]][,2], col =
"grey40", lty = 2)
for (i in 1:length(gcp.cores.LP))
lines(summary(gcp.est.LP.core[[i]], display = "chao")[[1]][,1],
summary(gcp.est.LP.core[[i]], display = "chao")[[1]][,2])

plot(seq(0, 40, 4), seq(0, 500, 50), type = "n", xlab = "", ylab
= "", main = "Early Eocene sites")
for (i in 1:length(col.cores.EE))
lines(summary(col.est.EE.core[[i]], display = "chao")[[1]][,1],
summary(col.est.EE.core[[i]], display = "chao")[[1]][,2], col =
"grey40", lty = 2)
for (i in 1:length(gcp.cores.EE))
lines(summary(gcp.est.EE.core[[i]], display = "chao")[[1]][,1],
summary(gcp.est.EE.core[[i]], display = "chao")[[1]][,2])

plot(seq(0, 40, 4), seq(0, 500, 50), type = "n", xlab = "", ylab
= "", main = "Middle Eocene sites")
for (i in 1:length(col.cores.ME))
lines(summary(col.est.ME.core[[i]], display = "chao")[[1]][,1],
summary(col.est.ME.core[[i]], display = "chao")[[1]][,2], col =
"grey40", lty = 2)
for (i in 1:length(gcp.cores.ME))
lines(summary(gcp.est.ME.core[[i]], display = "chao")[[1]][,1],
summary(gcp.est.ME.core[[i]], display = "chao")[[1]][,2])

title(xlab = "Number of samples", ylab =  "Chao2 estimated species
richness", outer = T, line = 1)

##NMDS
```

```
gcp.nmds <- metaMDS(gcp[,specnumber(gcp, MARGIN = 2) > 1], trymax
= 50)

##Figure 7.6
##Plot with formation names and convex hulls
plot(gcp.nmds, display=c("sites"), choices = c(1,2), type="n",
shrink=FALSE, xlab = "NMDS axis 1", ylab = "NMDS axis 2")
points(gcp.nmds, display=c("sites"), choices = c(1,2), type="p",
col = "grey40", shrink=FALSE)
ordihull(gcp.nmds, gcp.info$Formation, display = "sites", draw =
c("polygon"), label = TRUE)

##TURNOVER
##Enter data - GCP, binned into formations with formation uniques
removed
GCP.fmns <- read.table (file = "Dataset_fmns.txt", header=TRUE,
row.names=1)
fmns.details <- read.table (file = "Formations_details.txt",
header=TRUE, row.names=1)

##Set up list of rarefied matrices and remove unique/zero entry
taxa
GCP.fmns.subs <- vector("list", 1000)
for (i in 1:1000) GCP.fmns.subs[[i]] <- rrarefy(GCP.fmns, 290)
GCP.fmns.subs.nouniques <- lapply(GCP.fmns.subs, function(x)
x[,colSums(x > 0) > 1])

##Set up list of range-through matrices, and combine rowSums into
one 8 x 1000 cell matrix
rangethrough <- vector("list", 1000)

for (i in 1:1000) rangethrough[[i]] <- matrix(0,nrow =
nrow(GCP.fmns.subs.nouniques[[i]]), ncol =
ncol(GCP.fmns.subs.nouniques[[i]]), dimnames =
list(rownames(GCP.fmns.subs.nouniques[[i]]),
colnames(GCP.fmns.subs.nouniques[[i]])))

for (i in 1:length(rangethrough)) {
    for (j in 1:ncol(rangethrough[[i]])) rangethrough[[i]]
[min(which(GCP.fmns.subs.nouniques [[i]] [,j] != 0)):max(which
(GCP.fmns.subs.nouniques [[i]] [,j] !=0)), j] <- 1
}

standing.richness <- matrix(0,nrow = 8, ncol = 1000, dimnames =
list(rownames(GCP.fmns), NULL))
for (i in 1:1000) standing.richness [,i] <-
rowSums(rangethrough[[i]])

##Set-up list of taxon FADs and LADs
taxon.ranges <- vector("list", 1000)
turnover <- function(x) fmns.details[range(which(x != 0)),2]
```

```
for (i in 1:1000) taxon.ranges [[i]] <-
t(apply(GCP.fmns.subs.nouniques [[i]], 2, turnover))

##Convert FAD/LAD matrices to tables, and combine into 8 x1000 cell
matrices
last.app.freq <- lapply(taxon.ranges, function (x)
as.data.frame(table(x[,1]), stringsAsFactors = FALSE))
last.appearances <- as.matrix(fmns.details[,2])
colnames(last.appearances) <- "mid.point"
for (i in 1:1000) last.appearances <- merge(last.appearances,
last.app.freq [[i]], by.x = "mid.point", by.y = "Var1", all=T)
last.appearances[is.na(last.appearances)] <- 0
last.appearances <-
last.appearances[rev(order(last.appearances$mid.point)),]
last.appearances <- last.appearances[,2:1001]
last.appearances.prop <- last.appearances/standing.richness

first.app.freq <- lapply(taxon.ranges, function (x)
as.data.frame(table(x[,2]), stringsAsFactors = FALSE))
first.appearances <- as.matrix(fmns.details[,2])
colnames(first.appearances) <- "mid.point"
for (i in 1:1000) first.appearances <- merge(first.appearances,
first.app.freq [[i]], by.x = "mid.point", by.y = "Var1", all=T)
first.appearances[is.na(first.appearances)] <- 0
first.appearances <-
first.appearances[rev(order(first.appearances$mid.point)),]
first.appearances <- first.appearances[,2:1001]
first.appearances.prop <- first.appearances/standing.richness

##Bring together results into summary table, and export
turnover.fmns.results <- matrix(0,nrow = 8, ncol = 15, dimnames
= list(rownames(GCP.fmns), c("Mean.standing.richness",
"M.s.r.+95%", "M.s.r.-95%", "Mean.no.LAD", "M.n.LAD+95%",
"M.n.LAD-95%", "Mean.no.FAD", "M.n.FAD+95%", "M.n.FAD-95%",
"Mean.prop.LAD", "M.p.LAD+95%", "M.p.LAD-95%", "Mean.prop.FAD",
"M.p.FAD+95%", "M.p.FAD-95%")))
quartiles.975 <- function(m) sort(m)[975]
quartiles.25 <- function(m) sort(m)[25]
turnover.fmns.results[,1] <- rowMeans(standing.richness)
turnover.fmns.results[,2] <- apply(standing.richness, 1,
quartiles.975)
turnover.fmns.results[,3] <- apply(standing.richness, 1,
quartiles.25)
turnover.fmns.results[,4] <- rowMeans(last.appearances)
turnover.fmns.results[,5] <- apply(last.appearances, 1,
quartiles.975)
turnover.fmns.results[,6] <- apply(last.appearances, 1,
quartiles.25)
turnover.fmns.results[,7] <- rowMeans(first.appearances)
turnover.fmns.results[,8] <- apply(first.appearances, 1,
quartiles.975)
```

```
turnover.fmns.results[,9] <- apply(first.appearances, 1,
quartiles.25)
turnover.fmns.results[,10] <- rowMeans(last.appearances.prop)
turnover.fmns.results[,11] <- apply(last.appearances.prop, 1,
quartiles.975)
turnover.fmns.results[,12] <- apply(last.appearances.prop, 1,
quartiles.25)
turnover.fmns.results[,13] <- rowMeans(first.appearances.prop)
turnover.fmns.results[,14] <- apply(first.appearances.prop, 1,
quartiles.975)
turnover.fmns.results[,15] <- apply(first.appearances.prop, 1,
quartiles.25)
write.table(turnover.fmns.results,
"Turnover_sub_fmns_results.txt", sep = "\t", row.names = T,
col.names = T)

##Setting up matrix of expected first and last appearances under
constant turnover
turnover.fmns.expected <- matrix(0,nrow = 8, ncol = 4, dimnames
= list(rownames(turnover.fmns.results), c("Expected.last",
"sd.last", "Expected.first", "sd.first")))
(expected.last <-
sum(turnover.fmns.results[2:7,4])/sum(turnover.fmns.results[2:
7, 1]))
(expected.first <-
sum(turnover.fmns.results[2:7,7])/sum(turnover.fmns.results[2:
7, 1]))
turnover.fmns.expected[,1] <- rep(expected.last,
nrow(turnover.fmns.expected))
turnover.fmns.expected[,3] <- rep(expected.first,
nrow(turnover.fmns.expected))
turnover.fmns.expected[,2] <-
sqrt((turnover.fmns.expected[,1]*(1-turnover.fmns.expected[,1]
))/turnover.fmns.results[,1])
turnover.fmns.expected[,4] <-
sqrt((turnover.fmns.expected[,3]*(1-turnover.fmns.expected[,3]
))/turnover.fmns.results[,1])
write.table(turnover.fmns.expected,
"Turnover_sub_fmns_expected.txt", sep = "\t", row.names = T,
col.names = T)
```

```
##Figure 7.7
plot(turnover.fmns.results[2:7,13], fmns.details[2:7,1], type =
"n", main = "First appearances", xlab = "Proportional first
appearances (prop)", ylab = "", bty= "n")
lines(turnover.fmns.expected[2:7,3], fmns.details[2:7,2], col =
"red")
lines(turnover.fmns.expected[2:7,3]+(turnover.fmns.expected[2:
7,4]*1.96), fmns.details[2:7,2], col = "red", lty = 2)
lines(turnover.fmns.expected[2:7,3]-(turnover.fmns.expected[2:
7,4]*1.96), fmns.details[2:7,2], col = "red", lty = 2)
```

```
lines(turnover.fmns.results[2:7,13], fmns.details[2:7,2], col =
"blue")

plot(turnover.fmns.results[2:7,10], fmns.details[2:7,1], type =
"n", main = "Last appearances", xlab = "Proportional last
appearances", ylab = "", bty= "n")
lines(turnover.fmns.expected[2:7,1], fmns.details[2:7,2], col =
"red")
lines(turnover.fmns.expected[2:7,1]+(turnover.fmns.expected[2:
7,2]*1.96), fmns.details[2:7,2], col = "red", lty = 2)
lines(turnover.fmns.expected[2:7,1]-(turnover.fmns.expected[2:
7,2]*1.96), fmns.details[2:7,2], col = "red", lty = 2)
lines(turnover.fmns.results[2:7,10], fmns.details[2:7,2], col =
"blue")

##SPECIATION VERSIS IMMIGRATION
##Enter data - first appearance codes for each taxon
first.app <- read.table (file = "First_appearances.txt",
header=TRUE, row.names=1)
immigrants.all <- table(first.app[,1])
immigrants.all <- immigrants.all[c(3, 1, 4, 5, 6, 2, 7, 9)]

##Without sample uniques
gcp.vetted <- gcp[,specnumber(gcp, MARGIN = 2) > 1]
gcp.nosamplesuniques <- cbind(1:150, colnames(gcp.vetted))
first.app <- cbind(first.app, rownames(first.app))
colnames(first.app) <- c("code", "taxon")
colnames(gcp.nosamplesuniques) <- c("ID", "taxon")
gcp.nosamplesuniques <- merge(gcp.nosamplesuniques, first.app)
immigrants.nosampleuniques <- table(gcp.nosamplesuniques[,3])
immigrants.nosampleuniques <- immigrants.nosampleuniques[c(3, 1,
4, 5, 6, 2, 7, 9)]

##Without formation uniques
 gcp.fmns.vetted <- GCP.fmns[,specnumber(GCP.fmns, MARGIN = 2) >
1]
gcp.nofmnuniques <- cbind(1:126, colnames(gcp.fmns.vetted))
colnames(gcp.nofmnuniques) <- c("ID", "taxon")
gcp.nofmnuniques <- merge(gcp.nofmnuniques, first.app)
immigrants.nofmnuniques <- table(gcp.nofmnuniques[,3])
immigrants.nofmnuniques <- immigrants.nofmnuniques[c(3, 1, 4, 5,
6, 2, 7, 9)]

##Figure 7.8
par(mfrow = c(3,1))
pie(immigrants.all, main = "A) All taxa (n = 189)", init.angle =
90, radius = 1)
pie(immigrants.nosampleuniques, main = "B) No sample uniques (n
= 146)", init.angle = 90, radius = 1)
pie(immigrants.nofmnuniques, main = "C) N0 formation uniques (n
= 126)", init.angle = 90, radius = 1)
```

```
##Tabulate by formation and produce bar chart
fmns.details <- cbind(rownames(GCP.fmns), 1:8)
turnover <- function(x) fmns.details[range(which(x != 0)),1]
taxon.ranges <- t(apply(GCP.fmns, 2, turnover))
write.table(taxon.ranges, "Taxon_ranges.txt", sep = "\t",
row.names = T, col.names = T) ##range data for all taxa

##First appearances with formation uniques removed, and bar plot
taxon.ranges.nouniques <- t(apply(gcp.fmns.vetted, 2, turnover))
taxon.ranges.nouniques <- cbind(taxon.ranges.nouniques,
rownames(taxon.ranges.nouniques))
fa <- taxon.ranges.nouniques[,2:3]
colnames(fa) <- c("datum", "taxon")
fa <- merge(fa, first.app)
immigrants.fmn <- table(fa[,2], fa[,3])
immigrants.fmn <- immigrants.fmn[c(4, 2, 3, 6, 1, 5),c(3, 1, 4,
5, 6, 2, 7, 9)]

##Figure 7.9
barplot(t(immigrants.fmn), col=rainbow(9), ylim = c(0, 60),
legend.text = TRUE)
```