Evolution of Channel Islands Population: A Cranial Measurements Study Final Report

Denise Ferrari, Irina Kukuyeva

July 1, 2009

Contents

1 Background Information **2**

2 Overview of the Data **2**

3 Methodology **2**
3.1 Homogeneity Analysis (Homals) on all the Observations 2
3.2 Principal Component Analysis (PCA) on the Non-missing Data 3

4 Comparison of Homals to Classifying by Mean **3**

5 Summary **3**

6 Figures and Tables **4**

A R Code **10**
A.1 Preliminary Data Analysis 10
A.2 Homogeneity Analysis 11
A.3 Principal Component Analysis 12
A.4 Visualization and Final Touches 13
A.5 Generating the Cross-Table 16

1 Background Information

There has been population replacement during the prehistoric times on the Channel Islands. Before the Spanish came, there were two different Indian populations: one in the North and one in the South, confirmed by cranial measurements and Mitochondrial DNA (MDNA). The Northern population (closer to Santa Barbara) is the Chumash Indians; the Southern one is the Uto-aztecan. The Uto-aztecans spread to Central America and pushed out the Chumash. (It is yet unknown around what time this happened.)

The client has craniometric data for 128 skulls [1]. Using MDNA, the client has 7 reliable skull classifications (from 3 different islands), belonging to different groups. With this information, he would like to be able to classify the other individuals, based on their cranial measurements, taking their locations into account.

2 Overview of the Data

Originally, the data set had 128 observations and 56 variables composed of cranial and mandibular measurements and contains missing observations. After discussing with the client, 29 variables were selected, most of which correspond to cranial measurements.

For completeness, we present the variables analyzed below:

```
[1] "class"                "CranialLength"         "CranialBreadth"
[4] "BizygomaticBreadth"   "BasionBregmaHeight"    "BasionNasion"
[7] "BasionSubnasion"      "PalateWidth"           "PalateLength"
[10] "CranialBaseWidth"     "UpperFacialHeight"     "MinimumFrontalBreadth"
[13] "UpperFacialBreadth"   "NasalHeight"           "NasalWidth"
[16] "OrbitWidth"           "OrbitHeight"           "FullOrbitWidth"
[19] "InterorbitalWidth"    "FrontalChord"          "ParietalChord"
[22] "OccipitalChord"       "ForamenLength"         "ForamenWidth"
[25] "LeftMastoidLength"    "LeftMastoidWidth"      "RightMastoidLength"
[28] "RightMastoidWidth"    "CranialCircumference"
```

3 Methodology

Because of the sparsity of the data, we employed a dimension reduction technique called Principal Component Analysis (PCA). PCA creates linear combinations of the original variables in such a way that all the components taken together explain 100% of the variation in the dataset. Usually the leading PCs explain most of the dataset's variation and are extracted for interpretation. (For more information, please see a tutorial on PCA emailed on April 30, 2009.)

We have employed PCA in different ways:

1. applying the dimension reduction technique on the non-missing observations, which created two composite variables to analyze (more details in Section 3.1);

2. transforming the values of the variables to quantiles and applying the dimension reduction technique on the resulting data, which created two different composite variables to analyze (more details in Section 3.2).

3.1 Homogeneity Analysis (Homals) on all the Observations

For this analysis, we first transformed the values for each of the variables to be in 5 quantiles: 0-20, 20-40, 40-60, 60-80, 80-100. Then we performed Homogeneity Analysis via the package `homals` in `R,` allowing missing

observations in the analysis and a rank constraint of order 1 in two dimensions. These constraints are equivalent to performing PCA on the transformed data.
The resulting loadings are shown in Table 2, where the columns tell us how to combine the original variables to produce two new and different variables that we can proceed to analyze. Notice that these loadings differ from those in Table 1. More knowledge of the field is needed before we can interpret these variables. The relationship between the loadings is shown in Figure 1 (a). We can see that the loadings seem to comprise of two groups: highly and slightly correlated variables.

The resulting two variables (components) extracted for the analysis are shown in Figure 2 (a). Figure 3 (a) shows how to classify the data into two groups using a line as a separator. The inventory names of the skulls in Group 1 are (dark grey in the figure):

```
 [1]  7012  7022  7024  7025  7029  7044  7045  7053  7057  7059  7067  7073  7112 10248
[15] 10249 10250 10251 10252 22646 22650 22651 22652
```

Furthermore, we performed a Hotelling's T^2 test, a parametric multivariate test to determine if the means of the two resulting groups were different [2, 5].

> *Null hypothesis:* means of the two groups are the same.
> *Alternative hypothesis:* means of the two groups are different.

The resulting $p - value = 1 >>> 0.05$ when compared to a χ^2_2 (because the groups were not normally distributed). Therefore, we do not reject the null hypothesis. *Please note:* the assumptions behind this test are violated: the loadings are correlated with each other and the covariance matrices for the groups were unequal.

3.2 Principal Component Analysis (PCA) on the Non-missing Data

We performed PCA on the non-missing standardized observations (for each variable, the mean was zero and a standard deviation of one) to see if we can separate Group 2 into two groups. The resulting loadings are shown in Table 1, where the columns tell us how to combine the original variables to produce two new variables that we can proceed to analyze. The relationship between the loadings is shown in Figure 1(b). We can see that some are more correlated than others.

The resulting two variables (components) extracted for the analysis are shown in Figure 2 (b). The figure shows that if we were to use a line to separate the DNA-classified skulls (solid circles in the figure), we would be misclassifying at least one individual.

4 Comparison of Homals to Classifying by Mean

The client previously classified the long-headed and wide-headed individuals based on how their measurements compared to the average Cephalic Index. Figure 3 (b) builds on (a) by adding the classifications of the individuals performed by the client. The figure shows a high degree of misclassification. Using Figure 3 (b), we assigned each non-DNA-tested observation to a group closer to the seven DNA-tested observations. The results are summarized in Table 3. The numbers represent the number of observations in each of the seven groups.

5 Summary

We recommend performing homogeneity analysis for the quantiles of the variables in the data set. The analysis showed that we can identify two groups. Please see the figures and tables for more information.

6 Figures and Tables

	Loading 1	Loading 2
CranialLength	-0.23	-0.02
CranialBreadth	-0.15	-0.03
BizygomaticBreadth	-0.26	0.05
BasionBregmaHeight	-0.22	0.06
BasionNasion	-0.22	-0.05
BasionSubnasion	-0.21	0.02
PalateWidth	-0.19	0.23
PalateLength	-0.18	0.08
CranialBaseWidth	-0.08	0.03
UpperFacialHeight	-0.21	0.03
MinimumFrontalBreadth	-0.18	-0.35
UpperFacialBreadth	-0.25	-0.19
NasalHeight	-0.22	0.02
NasalWidth	-0.15	-0.14
OrbitWidth	-0.17	-0.27
OrbitHeight	-0.09	-0.37
FullOrbitWidth	-0.24	-0.18
InterorbitalWidth	-0.18	-0.03
FrontalChord	-0.10	0.00
ParietalChord	-0.17	-0.13
OccipitalChord	-0.05	0.24
ForamenLength	-0.18	-0.14
ForamenWidth	-0.14	-0.10
LeftMastoidLength	-0.22	0.30
LeftMastoidWidth	-0.14	0.34
RightMastoidLength	-0.22	0.33
RightMastoidWidth	-0.19	0.30
CranialCircumference	-0.25	-0.06

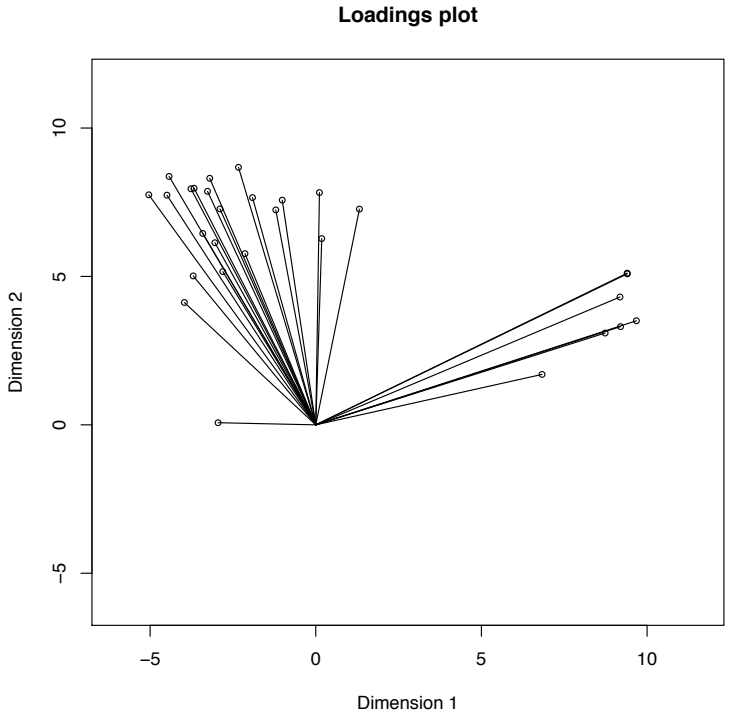
Table 1: Loadings for PCA on Non-missing Observations

	Loading 1	Loading 2
class	-2.9483125	0.0714071
CranialLength	-5.0403988	7.7508350
CranialBreadth	-1.0113568	7.5713081
BizygomaticBreadth	-4.4917898	7.7387638
BasionBregmaHeight	9.4022078	5.0876874
BasionNasion	9.3996105	5.1105118
BasionSubnasion	9.1825219	4.3072028
PalateWidth	-1.9089606	7.6558134
PalateLength	-1.2057782	7.2410688
CranialBaseWidth	-2.3329956	8.6762964
UpperFacialHeight	-3.2674658	7.8690705
MinimumFrontalBreadth	-3.6988744	5.0179314
UpperFacialBreadth	-3.6722889	7.9724740
NasalHeight	-3.2008101	8.3075006
NasalWidth	-3.0453323	6.1362109
OrbitWidth	-3.4101621	6.4483188
OrbitHeight	-3.9657588	4.1224319
FullOrbitWidth	-3.7664351	7.9537662
InterorbitalWidth	-2.8060616	5.1637223
FrontalChord	0.1109178	7.8265922
ParietalChord	6.8302386	1.6991778
OccipitalChord	9.1978481	3.3086938
ForamenLength	9.6775137	3.5073880
ForamenWidth	8.7339836	3.0926405
LeftMastoidLength	-2.8904859	7.2766770
LeftMastoidWidth	-2.1363709	5.7679342
RightMastoidLength	1.3183857	7.2697105
RightMastoidWidth	0.1811437	6.2752557
CranialCircumference	-4.4269759	8.3695806

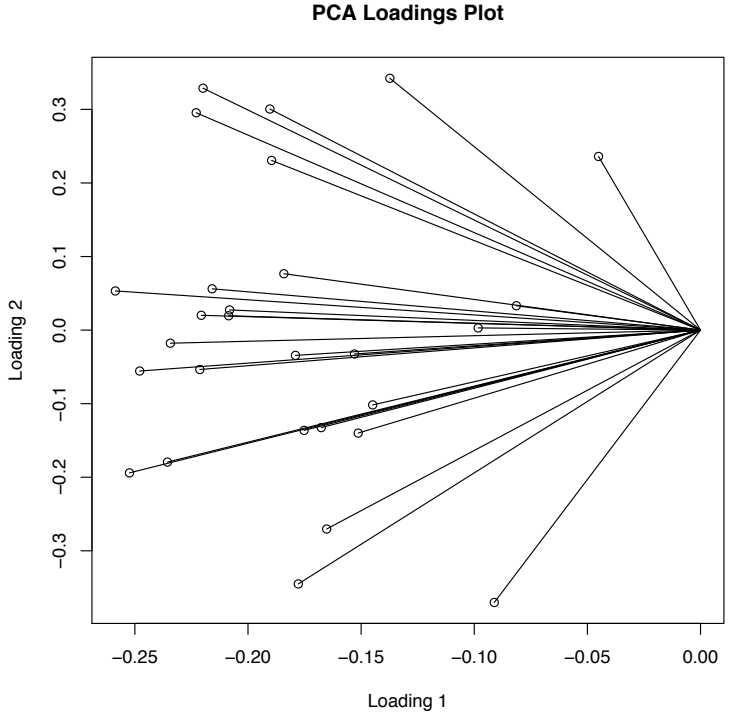
Table 2: Loadings for Homals on all Observations

Sylvere's Classification	A_{DNA}	B_{DNA}	C_{DNA}	Row Sums
San Miguel Wide-Headed	5	0	10	15
San Miguel Long-Headed	2	1	8	11
San Nicolas Wide-Headed	1	0	7	8
San Nicolas Long-Headed	3	4	3	10
Other	19	9	56	84
Column Sums	30	14	84	128

Table 3: Cross-Table for Classification

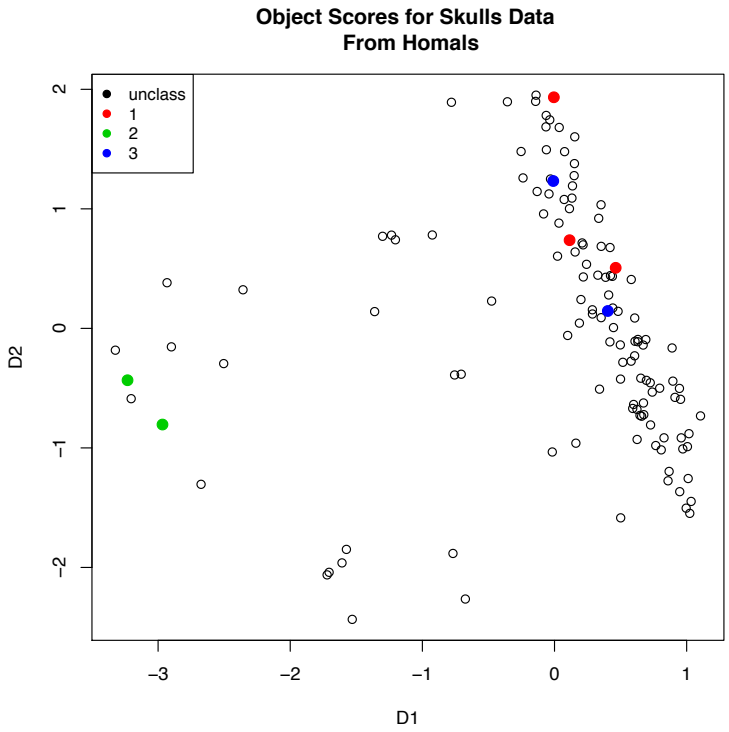


(a) Loadings from Homals

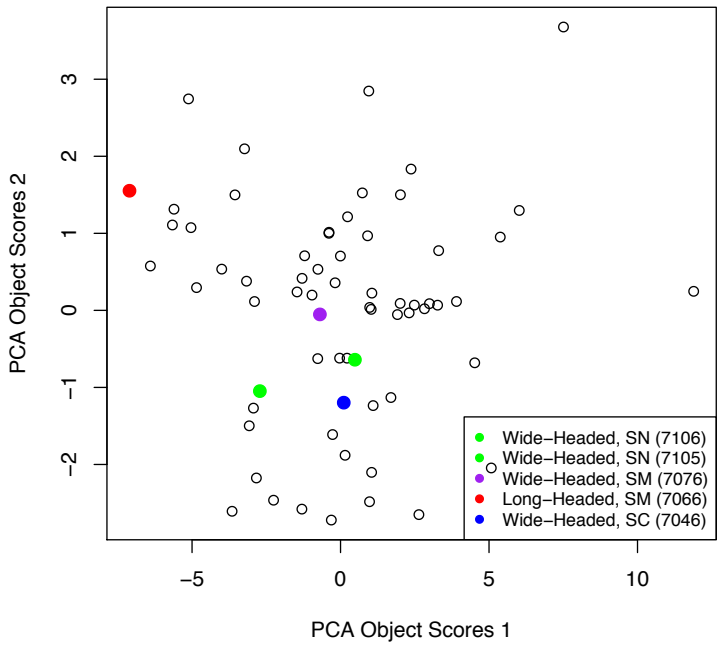


(b) Loadings from PCA

Figure 1: Loadings from Dimension Reduction Analysis

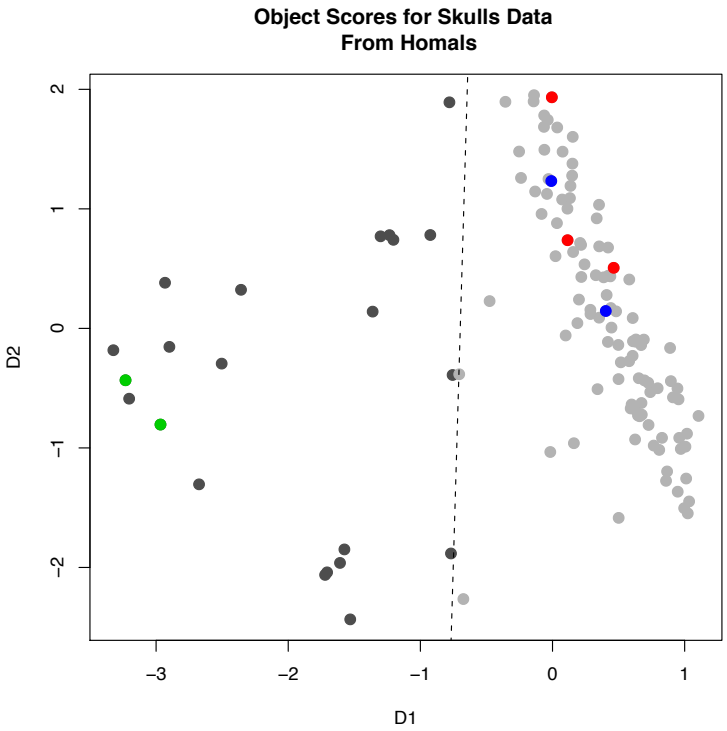


(a) Projections from Homals

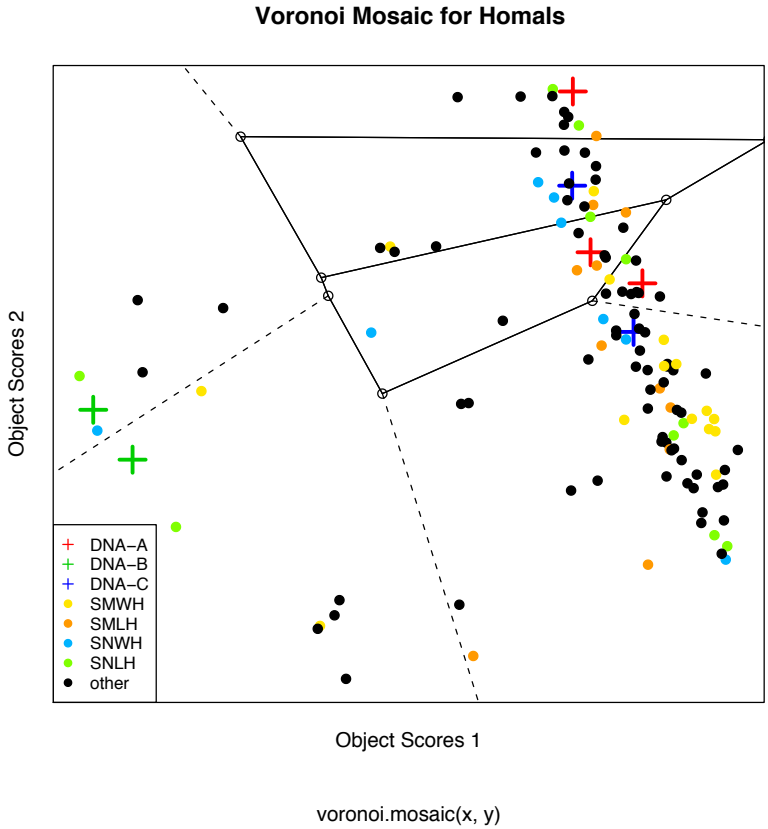


(b) Projections from PCA

Figure 2: Projections from Dimension Reduction Analysis



(a) Classification for Class into 2 Groups



(b) Voronoi Mosaic for Class Using Mean as Discriminant

Figure 3: Projections from Dimension Reduction Analysis

References

[1] S. Valentin. *Prehistoric Population Replacement on California's Channel Islands.* Personal Communication. May 1, 2009.

[2] H. Oja and H. Rundles. *Multivariate Nonparametric Tests.* Statistical Science. 2004, Vol. 19, No. 4, 598-605.

[3] A. Gifi. Nonlinear Multivariate Analysis 1981.

[4] G. and J. de Leeuw. *The Gifi System Of Descriptive Multivariate Analysis* Statistical Science. 1998. http://www.stat.ucla.edu/papers/preprints/

[5] B. Everitt. An R and S-plus companion to multivariate analysis. 2005.

A R Code

A.1 Preliminary Data Analysis

```
#######################################
###  Read-in the data:
#######################################
rm(list=ls())
data.orig<-read.csv("GoodCopy.csv", h=T, sep=",")
# data includes 128 observations

# Column 2 corresponds to the skulls classes
data<-data.orig[, c(2:3,5:33)]
attach(data)

# Made data changes per Sylvere:
CranialBaseWidth[87]<-12
FrontalChord[40]<-106.5
InterorbitalWidth[70]<-NA
detach()
attach(data)

# to refer to indiv with inventory name:
row.names(data)<-data.orig[, 1]

summary(data)
save.image("data.RData")

#######################################
### Discritizing the data:
#######################################

# Use quantiles, every 20th quantile is one level of a factor:
# 5 groups:
data.quant<-apply(data[, -c(1,2)], 2, function(x) quantile(x, seq(0,1,length.out=6), na.rm=T))
data.quant

# making into factor:
classif<-matrix(0, nrow=nrow(data), ncol=(ncol(data)-2))
colnames(classif) <- colnames(data[,-c(1,2)])
for(i in 3:ncol(data)){
                classif[data[, i]<=data.quant[2, (i-2)], (i-2)]<-1
                classif[data[, i]> data.quant[2, (i-2)] & data[, i]<=data.quant[3, (i-2)], (i-2)]<-2
                classif[data[, i]> data.quant[3, (i-2)] & data[, i]<=data.quant[4, (i-2)], (i-2)]<-3
                classif[data[, i]> data.quant[4, (i-2)] & data[, i]<=data.quant[5, (i-2)], (i-2)]<-4
                classif[data[, i]> data.quant[5, (i-2)] & data[, i]<=data.quant[6, (i-2)], (i-2)]<-5
        }

# missing data coded as NA
#classif[which(classif==0)]<-NA
# to recode back to zeros:
#classif[is.na(classif)]<-0

classif<-apply(classif, 2, as.integer)

# take out Cephalic Index=l/w
classif<-classif[, -3]
# include class variable to the classif data.frame
classif <- data.frame(class=data.orig$Class, classif)
# Missing classes coded as zero:
classif$class[is.na(classif$class)] <- 0
```

A.2 Homogeneity Analysis

```
#####################################
### Homogeneity Analysis
#####################################

library(homals)
#res<-homals(classif, ndim=2)
#pdf("spanplot.pdf", onefile=T)
#plot(res, plot.type="spanplot", plot.dim=c(1,2))
#dev.off()

#pdf("vorplot.pdf", onefile=T)
#plot(res, plot.type="vorplot")
#dev.off()

# extract the group classified by DNA evidence:
# 7105(WH), 7106(WH): C =3
# 10250(LH), 10252(LH): B = 2
# 7066 (LH), 7046(WH), 7076(WH): A = 1

# IDs 7105,7106,10250,10252,7066,7046,7076
# classified as "C", "C", "B","B","A","A","A"

id <- c(7105,7106,10250,10252,7066,7046,7076)
dna <-subset(classif, rownames(data) %in% id)
# data should be in data frame format
dna <- data.frame(dna)

# Applying homals to the dna class. obs.
# breaks with higher dimensions
res.dna<-homals(dna, level="ordinal",rank= 1, ndim=1, verbose=TRUE, active=c(FALSE, rep(TRUE,28)))
plot(unlist(res.dna$loadings))
#plot(res.dna$objscores, col="white")
plot(res.dna$objscores, col=dna$class)

# Applying homals to the entire data set
res.all <- homals(classif, level="ordinal",rank= 1, ndim=2, verbose=TRUE, active=c(FALSE, rep(TRUE,28)))
```

A.3 Principal Component Analysis

```
#######################################
### Principal Component Analysis
#######################################
# To perform PCA on the data
# read-in the data:
rm(list=ls())
data.orig<-read.csv("GoodCopy.csv", h=T, sep=",")
# data includes 128 observations

# Drop columns that we will not be using for PCA
data.pca<-data.orig[, -c(1:4,7, 34:57)]
detach()
attach(data.pca)

# Made data changes per Sylvere:
CranialBaseWidth[87]<-12
FrontalChord[40]<-106.5
InterorbitalWidth[70]<-NA
detach()
attach(data.pca)

# to refer to indiv with inventory name:
row.names(data.pca)<-data.orig[, 1]

summary(data.pca)
save.image("data.pca.RData")

# Diagnostic check
boxplot(data.pca)
# potentially unequal variances

bartlett.test(data.pca)
# reject the null that the variances are the same
#=> use the correlation matrix for PCA analysis

### Perform PCA
# create the COR matrix ourselves
data.cor<-cor(data.pca, use="pairwise.complete.obs")

eigen.out<-eigen(data.cor)
eva<-eigen.out$values
# we want the square roots of these:
eva.sqrt<-sqrt(eva)
propVar<-eva/sum(eva)

pdf("screePlot.pdf")
plot(propVar[1:10], type="b", ylab="Proportion of Variance")
dev.off()

## extract 2 components (component 3 contributes only 5%)
## 1/28=0.0357

# loadings:
evect<-eigen.out$vectors[, c(1,2)]

dat.scores<-data.cor%*%evect

# plot the object scores
plot(dat.scores[, 1], dat.scores[, 2])
# Results not comparable to Voronoi plot, b/c we used the correlation matrix above

# To repeat the analysis for complete data:
na.omit(data.pca, use="pairwise.complete.obs")->data.clean

# To center the data:
library(QuantPsyc)
Make.Z(data.clean)->data.centr

eigen2.out<-svd(data.centr)
eva2<-eigen2.out$d
# we want the square roots of these:
eva2.sqrt<-sqrt(eva2)
propVar2<-eva2/sum(eva2)

pdf("screePlot2.pdf")
plot(propVar2[1:10], type="b", ylab="Proportion of Variance")
dev.off()

# propVar2:
# [1] 0.155640363 0.064710300 0.059680046
# extract 2 components

# loadings:
evect2<-eigen2.out$v[, c(1,2)]

data.scores2<-data.centr%*%evect2

index <- rownames(data.clean)
class.ind <- subset(data.orig, rownames(data.pca)%in%index, Class)
class.ind[is.na(class.ind)] <- 0
class.ind <- class.ind+1

# labeling the points
na.omit(cbind(class=(data.orig[, 2]+1), data.pca), use="pairwise.complete.obs")->data.clean2
# plot the object scores
pdf("scoresPlot.pdf")
plot(data.scores2[, 1], data.scores2[, 2], col=as.vector(as.matrix(class.ind)), pch=19)
dev.off()
```

A.4 Visualization and Final Touches

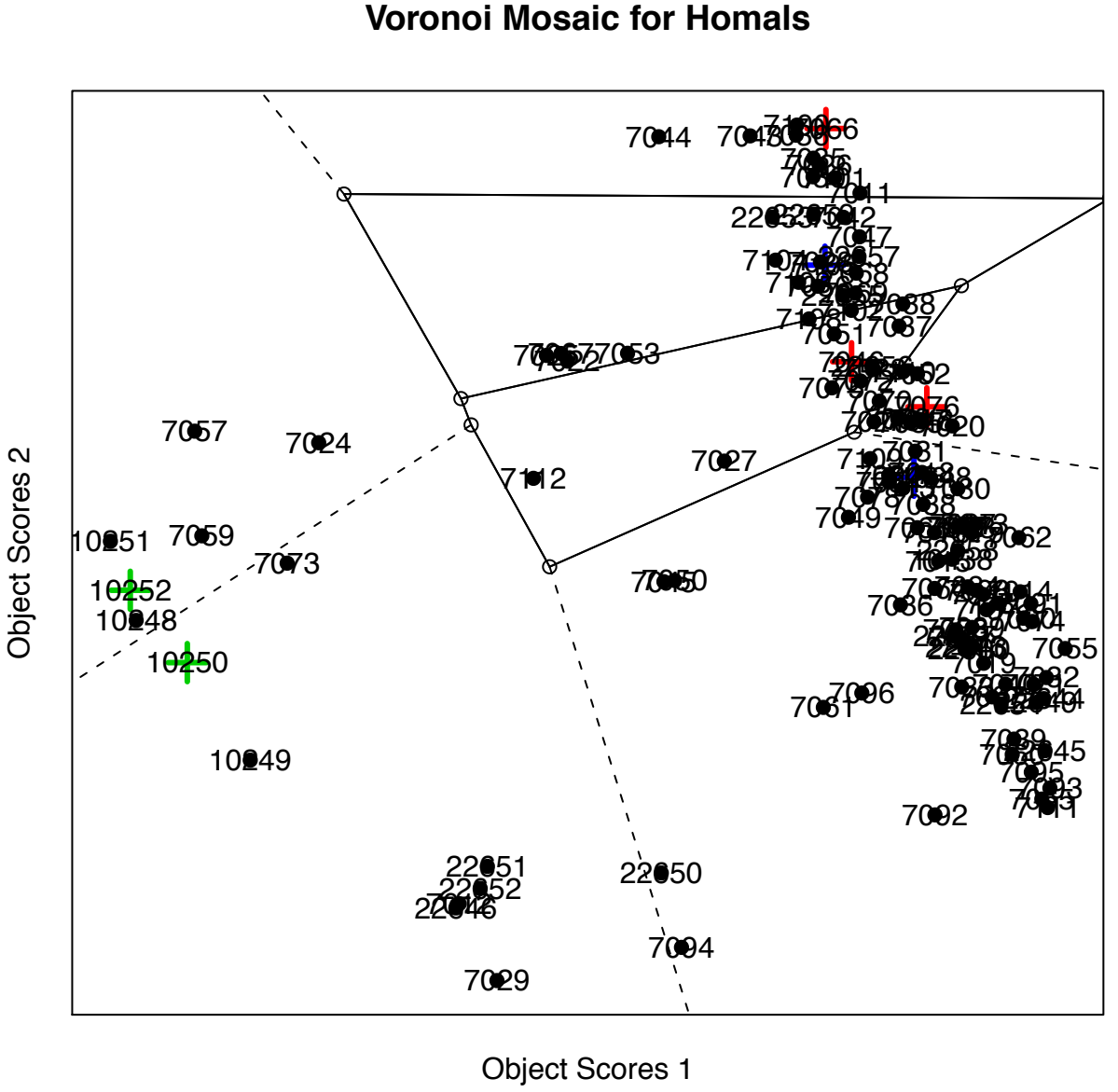
```
#####################################
###Visualization
#####################################

# creating different colors for different classes
newcol <- classif$class+1
# Plot object scores
plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data")
legend("topleft", c("unclass", "1", "2", "3"), col=1:4, lty=rep(-1,4), pch=rep(1,4), cex=.7)
#
plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data (D1>0)", xlim=c(0,1.2))
legend("topright", c("unclass", "1", "2", "3"), col=1:4, lty=rep(-1,4), pch=rep(1,4), cex=.7)
# Plot voronoi plot
pdf("vorplot.pdf")
plot(res.all, plot.type="vorplot", col=newcol, cex=.7)
dev.off()
# Plot starplot
pdf("starplot.pdf")
plot(res.all, plot.type="starplot", pch=newcol, cex=.7)
dev.off()
# Plot hull plot
pdf("hullplot.pdf")
plot(res.all, plot.type="hullplot", cex=.7)
dev.off()
# Load plot
pdf("loadplot.pdf")
plot(res.all, plot.type="loadplot", cex=.7)
dev.off()
# Discrimination measure plot
pdf("dmplot.pdf")
plot(res.all, plot.type="dmplot", cex=.7)
dev.off()

# Applying 3D homals to the entire data set
res.3D <- homals(classif, level="ordinal",rank= 1, ndim=3, verbose=TRUE, active=c(FALSE, rep(TRUE,28)))

# Plot object scores
plot3d(res.3D, col=newcol,plot.type="objplot")

#####################################
### Goal: Modify the existing plots to make them easier to read
#####################################
load("homals.data.RData")
# contains results from the the homals for DNA and whole data set
# classifications are in NEWCOL

### Part I: Re-graphing the homals result

pdf("ObjScoresHomals.pdf")
# Plot object scores
plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data \n From Homals")
points(res.all$objscores[which(newcol!=1), 2]~res.all$objscores[which(newcol!=1), 1], cex=1.3,
pch=19, col=newcol[which(newcol!=1)])
legend("topleft", c("unclass", "1", "2", "3"), col=1:4, lty=rep(-1,4), pch=rep(19,4), cex=0.9)
dev.off()

### Part II: Re-graphing the PCA result
load("data.pca.RData")

# color-coding
pdf("ObjScoresPCA.pdf", width=6, height=6)
plot(data.scores2[, 2]~ data.scores2[, 1], col=as.vector(as.matrix(class.ind)), xlab="PCA Object Scores 1",
ylab="PCA Object Scores 2")
# 7106:
points(data.scores2[5, 2]~ data.scores2[5, 1], col="green", pch=19, cex=1.3)
# 7105:
points(data.scores2[6, 2]~ data.scores2[6, 1], col="green", pch=19, cex=1.3)
# 7076:
points(data.scores2[25, 2]~ data.scores2[25, 1], col="purple", pch=19, cex=1.3)
# 7066:
points(data.scores2[30, 2]~ data.scores2[30, 1], col="red", pch=19, cex=1.3)
# 7046:
points(data.scores2[44, 2]~ data.scores2[44, 1], col="blue", pch=19, cex=1.3)
legend("bottomright", c("Wide-Headed, SN (7106)", "Wide-Headed, SN (7105)", "Wide-Headed, SM (7076)",
 "Long-Headed, SM (7066)", "Wide-Headed, SC (7046)"), col=c("green", "green", "purple", "red", "blue"), pch=rep(19,5), cex=0.8)
dev.off()

### Part III: Loadings
# For Homals, see LOADPLOT, from 04/23/09 R Code
# Copied into current directory

# For PCA
# Loadings are in EVECT2
pdf("LoadingsPCA.pdf")
plot(evect2[1,2]~evect2[1,1], xlim=c(min(evect2[, 1]),0), ylim=c(min(evect2[, 2]),max(evect2[, 2])), xlab="Loading 1",
ylab="Loading 2", main="PCA Loadings Plot")
segments(evect2[1,1],evect2[1,2],0,0)

for(i in 2:28){
        points(evect2[i,2]~evect2[i,1])
        segments(evect2[i,1],evect2[i,2],0,0)
                }
dev.off()

## Seems as though PCA Loading 1 is more similar to Homals (-Loading 2)

### Part IV: Voroni Plots
```

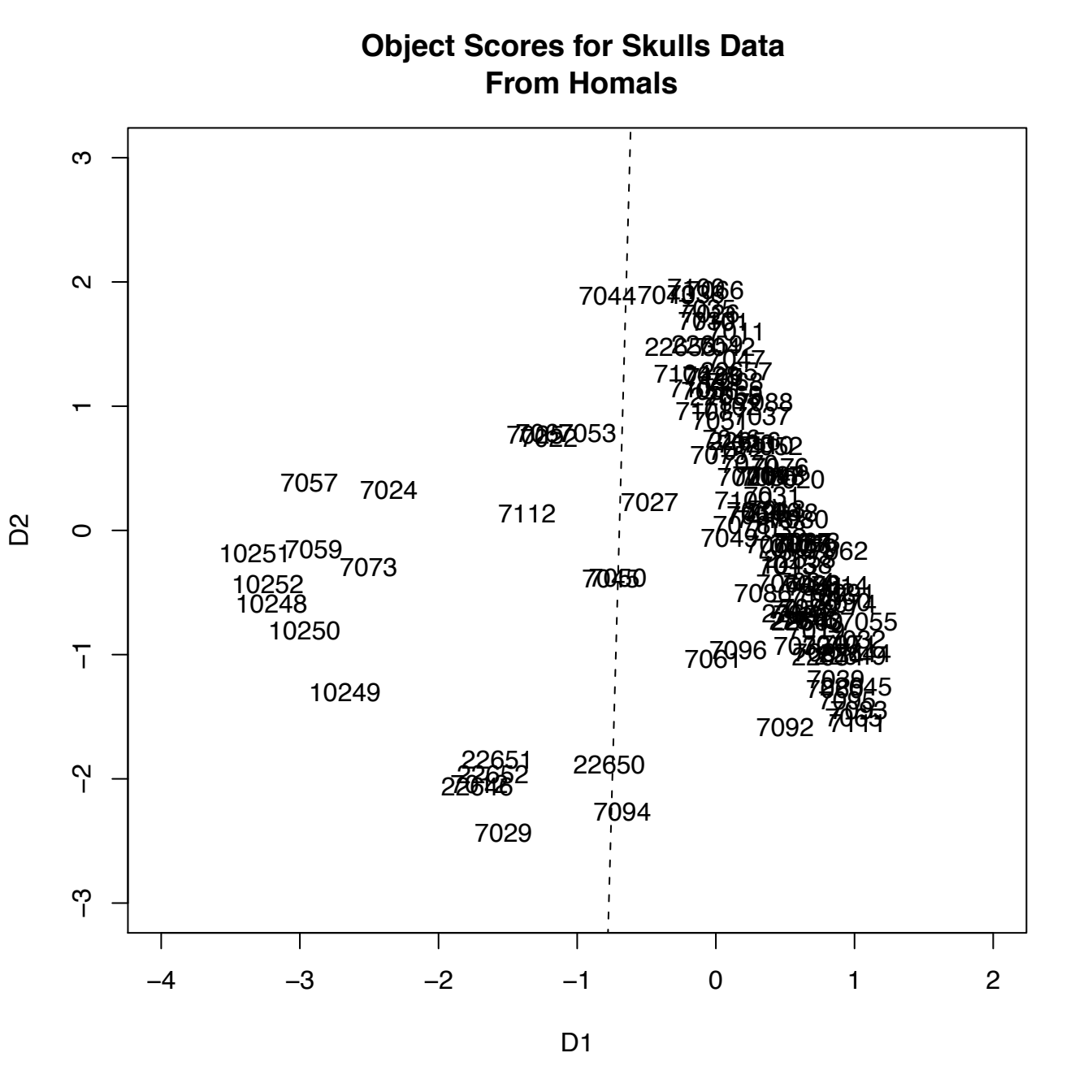
```
# from: http://zoonek2.free.fr/UNIX/48_R/16_Miscellaneous.txt

## Section 1: Homals
# look at the DNA data (from "homals.data.RData")
x<-res.all$objscores[which(newcol!=1), 1]
y<-res.all$objscores[which(newcol!=1), 2]
x.rest<-res.all$objscores[which(newcol==1), 1]
y.rest<-res.all$objscores[which(newcol==1), 2]

library(tripack)
pdf("VoronoiHomals.pdf")
plot(voronoi.mosaic(x, y), xlim=c(min(res.all$objscores[, 1]), max(res.all$objscores[, 1])),
ylim=c(min(res.all$objscores[, 2]), max(res.all$objscores[, 2])),
main="Voronoi Mosaic for Homals")
mtext("Object Scores 1", side=1, line=1)
mtext("Object Scores 2", side=2, line=1)
points(x, y, pch=3, cex=2, lwd=3, col=newcol[which(newcol!=1)])
box()
# add the points
points(x.rest, y.rest, cex=1, col=newcol[which(newcol==1)], pch=rep(19, length(newcol[which(newcol==1)])))
dev.off()

## Section 2: PCA
# look at the DNA data (from "data.pca.RData")
x<-data.scores2[which(class.ind!=1), 1]
y<-data.scores2[which(class.ind!=1), 2]
x.rest<-data.scores2[which(class.ind==1), 1]
y.rest<-data.scores2[which(class.ind==1), 2]

# classification in CLASS.IND
class.ind<-as.numeric(as.matrix(class.ind))
library(tripack)

pdf("VoronoiPCA.pdf")
plot(voronoi.mosaic(x, y), xlim=c(min(data.scores2[, 1]), max(data.scores2[, 1])), ylim=c(min(data.scores2[, 2]),
max(data.scores2[, 2])), main="Voronoi Mosaic for PCA")
plot(voronoi.mosaic(x, y))
mtext("Object Scores 1", side=1, line=1)
mtext("Object Scores 2", side=2, line=1)
points(x, y, pch=3, cex=2, lwd=3, col=class.ind[which(class.ind!=1)])
box()
# add the points
points(x.rest, y.rest, cex=1, col=class.ind[which(class.ind==1)], pch=rep(19, length(class.ind[which(class.ind==1)])))
dev.off()

########################################
### Goal 1: Identify which observations are in Group 1 and 2
### Goal 2: Perform a Non-parametric test for the eqaulity of means between the two groups
########################################
load("data.RData")
load("data.pca.RData")
load("homals.data.RData")
library(homals)

### Use the locator and identify functions to determine the Inventory names of the skulls
# from the two groups for class from Voronoi plot

plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data \n From Homals")

# index<-identify(res.all$objscores[, 1], res.all$objscores[, 2])
# 10  15  16  17  18  19  32  45  54  67  74  76  78  82 100 101 102 106 118 119 121
grp1<-c(10,  15,  16,  17,  18,  19,  32,  45,  54,  67,  74,  76,  78,  82, 100, 101, 102, 106, 118, 119, 121, 114)
grp2<-setdiff(1:nrow(classif), grp1)

# Highlight where the two groups are:
plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data \n From Homals")
points(res.all$objscores[grp1, 2]~res.all$objscores[grp1, 1], cex=1.3, pch=19, col=grey(0.3))
points(res.all$objscores[grp2, 2]~res.all$objscores[grp2, 1], cex=1.3, pch=19, col=grey(0.7))
points(res.all$objscores[which(newcol!=1), 2]~res.all$objscores[which(newcol!=1), 1], cex=1.3, pch=19,
col=newcol[which(newcol!=1)])

##### To find equation for the Voronoi line separator:
### Changing the size of the labels for the Voronoi plot for class
plot(res.all, plot.type="vorplot", col=newcol, cex=ifelse(newcol==1, 1,2), var.subset=1)
a<-locator(n = 2, type = "l", col="red")
m=( a$y[2]- a$y[1])/( a$x[2]- a$x[1]); m        # slope
b=a$y[1]-m*a$x[1]; b                    # intercept

### To add the line to the plot:
pdf("ClassifData.pdf")
plot(res.all$objscores, col=newcol, main="Object Scores for Skulls Data \n From Homals")

points(res.all$objscores[grp1, 2]~res.all$objscores[grp1, 1], cex=1.3, pch=19, col=grey(0.3))
points(res.all$objscores[grp2, 2]~res.all$objscores[grp2, 1], cex=1.3, pch=19, col=grey(0.7))
points(res.all$objscores[which(newcol!=1), 2]~res.all$objscores[which(newcol!=1), 1], cex=1.3, pch=19,
col=newcol[which(newcol!=1)])
abline(a=b, b=m, col="black", lty=2)
dev.off()

### Performing Hypothesis test
# Step 1: Covariance Matrix for the groups
s1<-cov(res.all$objscores[grp1,])
s2<-cov(res.all$objscores[grp2,])

n1=length(grp1)
n2=length(grp2)

s_pooled<-((n1-1)*s1+(n2-1)*s2)/(n1+n2-2)

m1<-apply(res.all$objscores[grp1,], 2, mean)
```

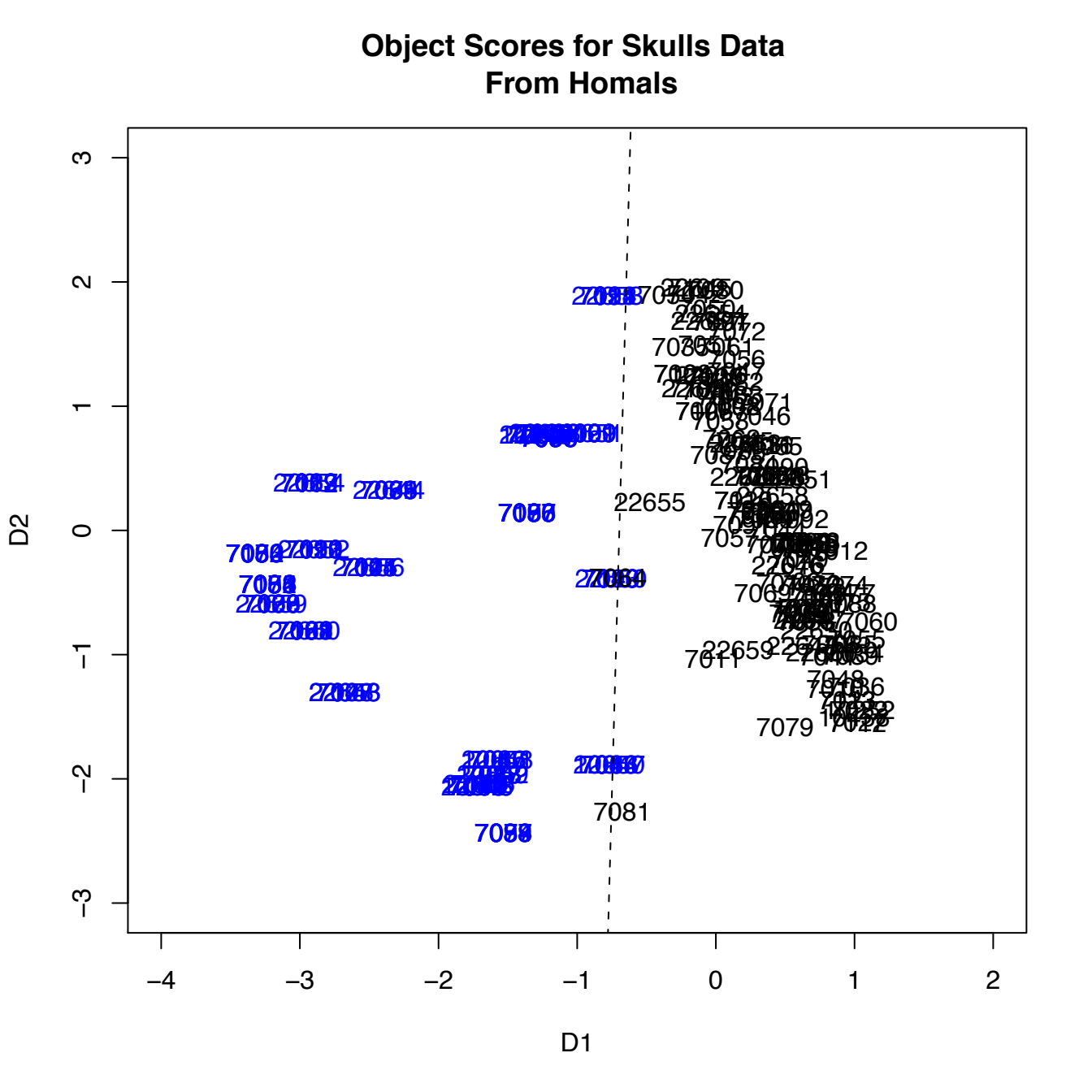
```
m2<-apply(res.all$objscores[grp2,], 2, mean)

dist<-(m1-m2)%*%solve(s_pooled)%*%(m1-m2)

T_2<-(n1*n2)*dist^2/(n1+n2)

# NOT multivariate normal
# mshapiro.test(t(res.all$objscores[grp1, 1:2])); mshapiro.test(t(res.all$objscores[grp2, 1:2]))

# compare to the chi-squared dist b/c each group is indiv non-normally dist.
pchisq(q=T_2, df=2) # =1 => reject the null hypothesis that the means are equivalent
```

A.5 Generating the Cross-Table

```
### Computing dist b/w points:
centroids<-rbind(res.all$objscores[which(newcol!=1),])
other.obs<-rbind(res.all$objscores[setdiff(1:128,which(newcol!=1)),])

# Calculating the distance of every obs. to the centroids (dna tested obs)
D.mat <- matrix(NA, ncol=7, nrow=121)
for(i in 1:7){
        C.mat <- matrix(rep(centroids[i,], 121), ncol=2,byrow=T)
        D.mat[,i] <- sqrt(rowSums((C.mat-other.obs)^2))
        }
# Indicator matrix with the closest centroid to a given point
D.min <- apply(D.mat,1, which.min)

# Table indexing centroid number to DNA class
centroids.copy <- centroids; rownames(centroids.copy) <- 1:7
centroids.table <- data.frame(ctr=1:7, dna=LETTERS[newcol-1])

# Assigning DNA classes to all other obs based on the closest centroid
group.code <- rep(NA,128)
group.code[which(newcol!=1)] <- LETTERS[newcol-1]
group.dna <- rep(NA,128)
group.dna[which(!is.na(group.code))]<- 1:7
group.dna[which(is.na(group.code))]<- D.min
group.code2 <- group.code
group.code2[is.na(group.code)] <- LETTERS[centroids.table$dna[group.dna[is.na(group.code)]]]

### Group classification based on geographical location (per Sylvere)
smwh<-c(7085, 7081, 7077, 7069, 7067, 7092, 7090, 7088, 7084,7082, 7078, 7076, 7072, 7070, 7068)
smlh<-c(7083, 7079, 7075, 7071, 7094, 7091, 7086, 7080, 7074,7073, 7066)
snwh<-c(7103, 7108, 7106, 7105, 7104, 7111, 7110, 7113)
snlh<-c(7100, 7102, 7101, 7107, 7112, 7109, 10252, 10251, 10250,10248)

# create vectors indexing group to the position of the obs in the data
v.smwh <- which(rownames(data)%in%smwh)
v.smlh <- which(rownames(data)%in%smlh)
v.snwh <- which(rownames(data)%in%snwh)
v.snlh <- which(rownames(data)%in%snlh)
v.other <- which(rownames(data)%in%setdiff(rownames(data), c(smwh,smlh,snwh,snlh)))

final.table <- data.frame(A=rep(NA, 5), B=rep(NA,5), C=rep(NA,5), row.names=c("smwh", "smlh","snwh","snlh", "other"))
#rownames(final.table) <- c("smwh", "smlh","snwh","snlh", "other")
final.table[1,] <- c(length(which(group.code2[v.smwh]=="A")),
length(which(group.code2[v.smwh]=="B")), length(which(group.code2[v.smwh]=="C")))
final.table[2,] <- c(length(which(group.code2[v.smlh]=="A")),
length(which(group.code2[v.smlh]=="B")), length(which(group.code2[v.smlh]=="C")))
final.table[3,] <- c(length(which(group.code2[v.snwh]=="A")),
length(which(group.code2[v.snwh]=="B")), length(which(group.code2[v.snwh]=="C")))
final.table[4,] <- c(length(which(group.code2[v.snlh]=="A")),
length(which(group.code2[v.snlh]=="B")), length(which(group.code2[v.snlh]=="C")))
final.table[5,] <- c(length(which(group.code2[v.other]=="A")),
length(which(group.code2[ v.other]=="B")), length(which(group.code2[v.other]=="C")))

Col.Sums <- colSums(final.table)
final.table <- rbind(final.table, Col.Sums=Col.Sums)
Row.Sums <- rowSums(final.table)
final.table <- cbind(final.table, Row.Sums=Row.Sums)

library(xtable)
xtable(final.table)
```


Voronoi Mosaic for Homals
Object Scores 2
Object Scores 1
7044
7057
7024
7112
7027
10251
7059
7073
10252
10248
10250
10249
7049
7062
7050
7086
7055
7096
7061
7092
22051
22050
22052
7094
7029
voronoi.mosaic(x, y)


Object Scores for Skulls Data
From Homals
D1
D2
−4
−3
−2
−1
0
1
2
3
7057
7024
7112
7027
10251
7059
7073
10252
10248
10250
10249
7096
7061
7092
22651
22650
7094
7029


Object Scores for Skulls Data
From Homals
D1
D2
−4
−3
−2
−1
0
1
2
3
22655
7081
7079
7060
7011
7048