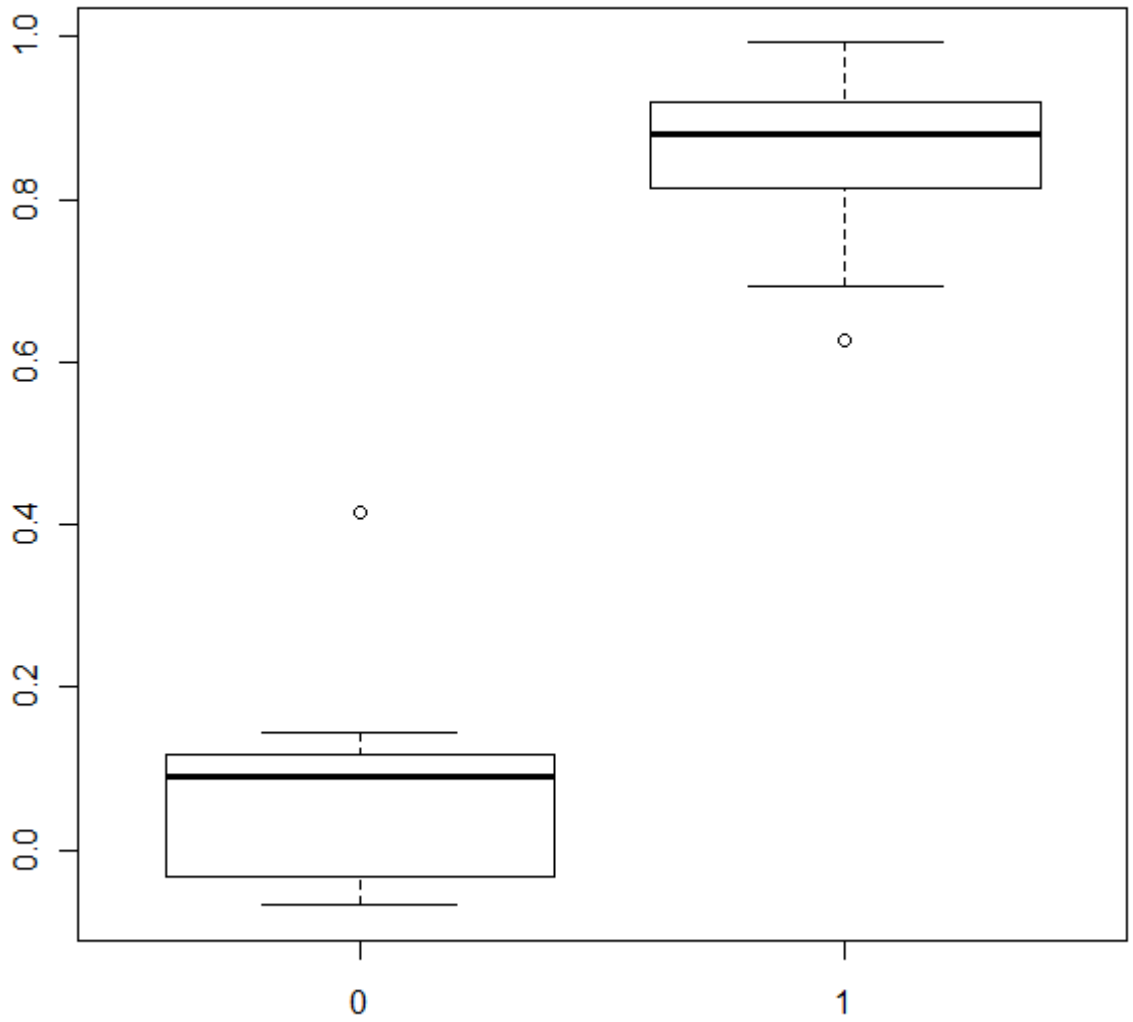
Preliminary analysis

1. SVM

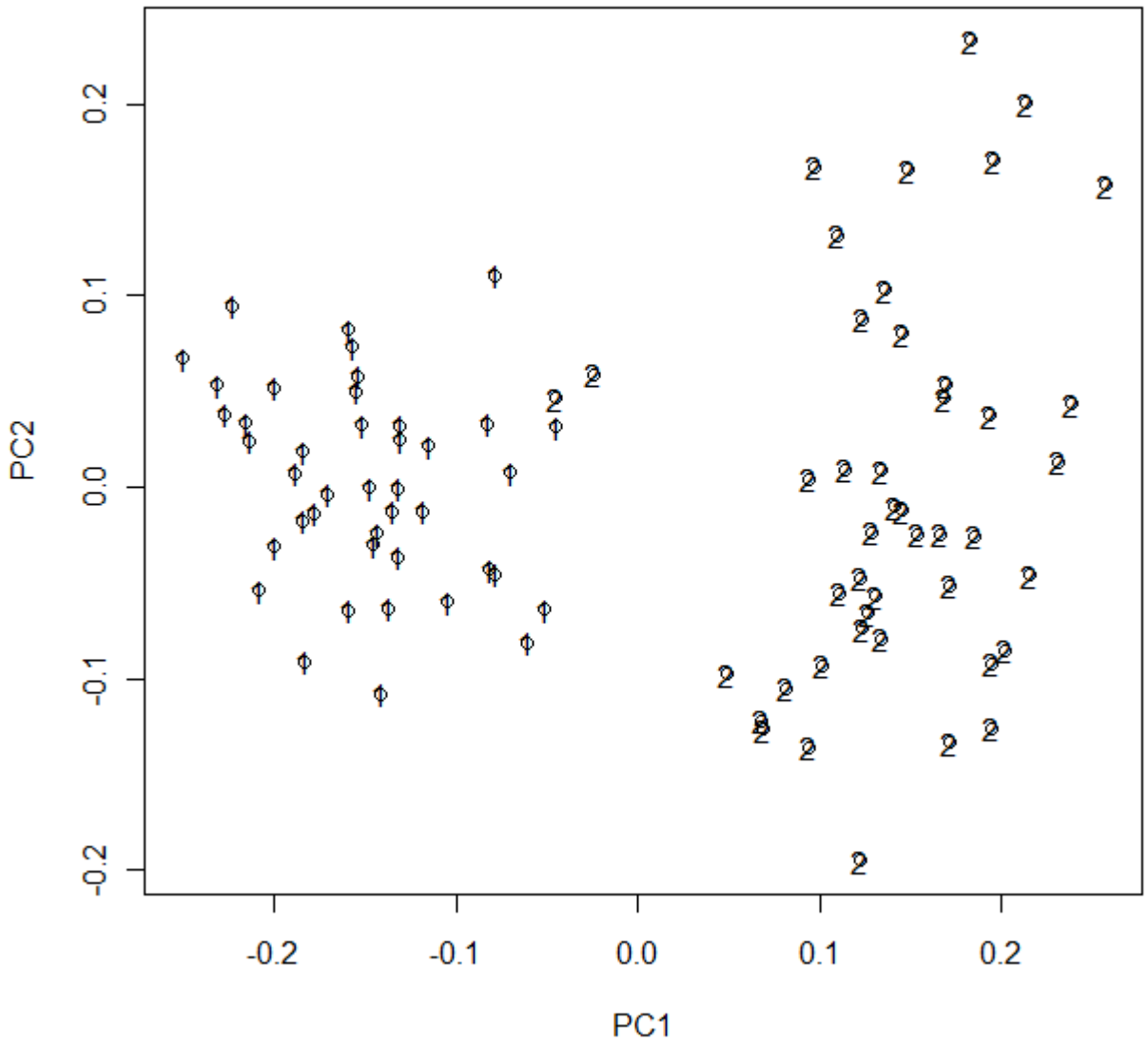
- x <- rbind(as.matrix(sam1[,-(1:2)]),as.matrix(sam2[,-1]),as.matrix(samw1[,-1]),as.matrix(samw2[,-1]))
- y<-c(rep(0,21),rep(0,21),rep(1,23),rep(1,22))
- library(e1071)
- index <- 1:nrow(x)
- test <- sample(index, trunc(length(index)/3))
- model <- svm(x[-test,],y[-test], method = "C-classification", kernel = "radial", cost = 10, gamma = 0.1)
- pred <- predict(model, x[test,])

- cor(pred,y[test])
- [1] 0.963056
- >
- > genotype <- factor(y[test])
- >
- > fit <- aov(pred ~ genotype)
- >
- > summary(fit)
- Df Sum Sq Mean Sq F value Pr(>F)
- genotype 1 4.3538 4.3538 345.29 < 2.2e-16 ***
- Residuals 27 0.3404 0.0126
- ---
- Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


1.0
0.8
0.6
0.4
0.2
0.0
0
1

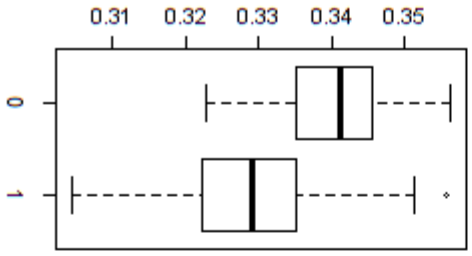
2. PCA

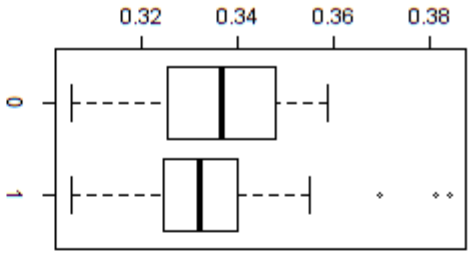
- > pcr <- prcomp(x,retx=TRUE)
- >
- > pcr$sdev/sum(pcr$sdev)
- [1] 0.317940915 0.166180882 0.138797489 0.116073979 0.034263914 0.027226814 0.023482727 0.020218250 0.018225094
- [10] 0.014768548 0.014264367 0.011578719 0.010347882 0.009162070 0.008715303 0.008537651 0.007413513 0.006573714
- [19] 0.005825174 0.005393407 0.004867561 0.004445146 0.004293583 0.003990926 0.003620615 0.003320557 0.003095090
- [28] 0.002705488 0.002510866 0.002159757
- >
- > features <- pcr$x
- >
- > fit <- aov(features[,1] ~ factor(y))
- >
- > summary(fit)
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 1.81030 1.81030 563.21 < 2.2e-16 ***
- Residuals 85 0.27321 0.00321
- ---
- Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

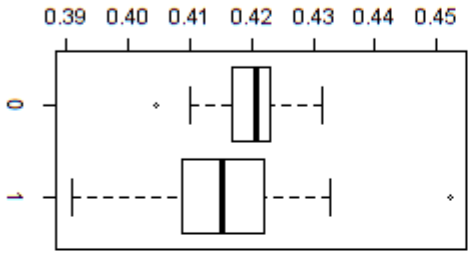

PC2
0.2
0.1
0.0
-0.1
-0.2
-0.2
-0.1
0.0
0.1
0.2
PC1

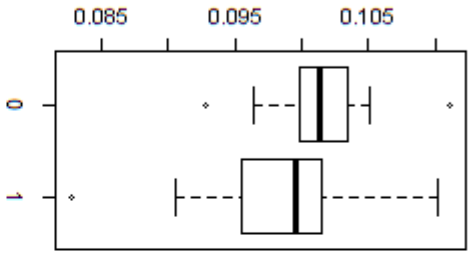
3. Polygon features

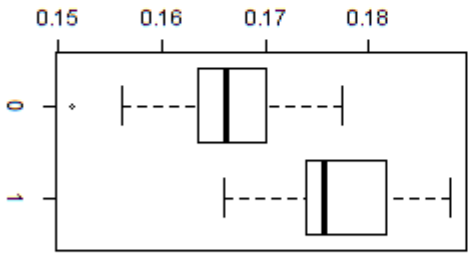
- polygon1<-c(1,2,7)
- polygon2<-c(2,6,12,7)
- polygon3<-c(6,8,13,12)
- polygon4<-c(8,9,10,14,13)
- polygon5<-c(5,11,10,9)
- polygon6<-c(10,11,15,14)
- polygon7<-c(4,15,11,5)
- Feature = polygon area / diameter

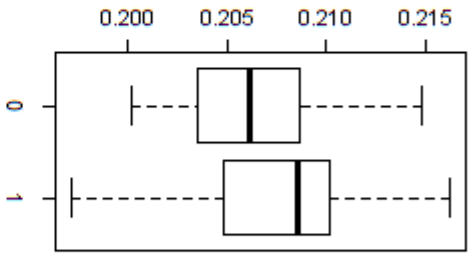

0.31
0.32
0.33
0.34
0.35
0
1

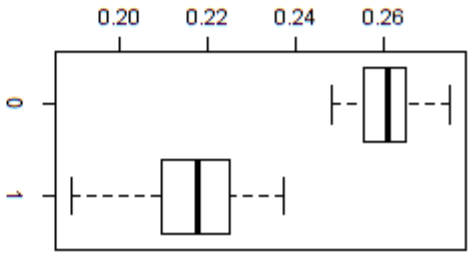

0.32
0.34
0.36
0.38
0
1


0.39
0.40
0.41
0.42
0.43
0.44
0.45
0
1


0.085
0.095
0.105
0
1


0.15
0.16
0.17
0.18
0
1


0.200
0.205
0.210
0.215
0
1


0.20
0.22
0.24
0.26
0
1

- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.0000709 7.0923e-05 0.3131 0.5772
- Residuals 85 0.0192527 2.2650e-04
-
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.00016613 0.00016613 10.697 0.001551 **
- Residuals 85 0.00132007 0.00001553
- ---
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.00004944 4.9437e-05 3.1319 0.08036 .
- Residuals 85 0.00134171 1.5785e-05
- ---
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.039912 0.039912 535.74 < 2.2e-16 ***
- Residuals 85 0.006332 0.000074
- ---
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.0030376 0.00303755 35.446 5.714e-08 ***
- Residuals 85 0.0072841 0.00008569
- ---
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.0004225 0.00042249 5.4852 0.02152 *
- Residuals 85 0.0065470 0.00007702
- ---
- Df Sum Sq Mean Sq F value Pr(>F)
- factor(y) 1 0.0024442 0.00244416 82.836 3.365e-14 ***
- Residuals 85 0.0025080 0.00002951

Questions

- Data is already aligned and registered (Does the data contain variation due to rotation)?
- Variation among individuals (or rotations) is not vary large compared to variation caused by genotype?

Idea: Supervised nonlinear dimension reduction

- Assume Y = shape extracted from image (already aligned and registered).
- Want to find features F(Y) in Y such that X is dependent of Y through and only trough Y.
- This is called sufficient dimension reduction, and is solved when X is univariate, and F(X) is linear.
- What can we do? Develop new machinge learning methods when F(x) is nonlinear.

What if Y = pixel level image?

- More challenging. Must use supervised dimension reduction.
- Anova:
- Y = genotype + rotation + artifacts + individual
- Are rotation, artifacts, individual variations independent of genotype?
- What are the size of these variations?
- Use pairwise distance to remove rotational variation.
- User defined rotation invariant features.