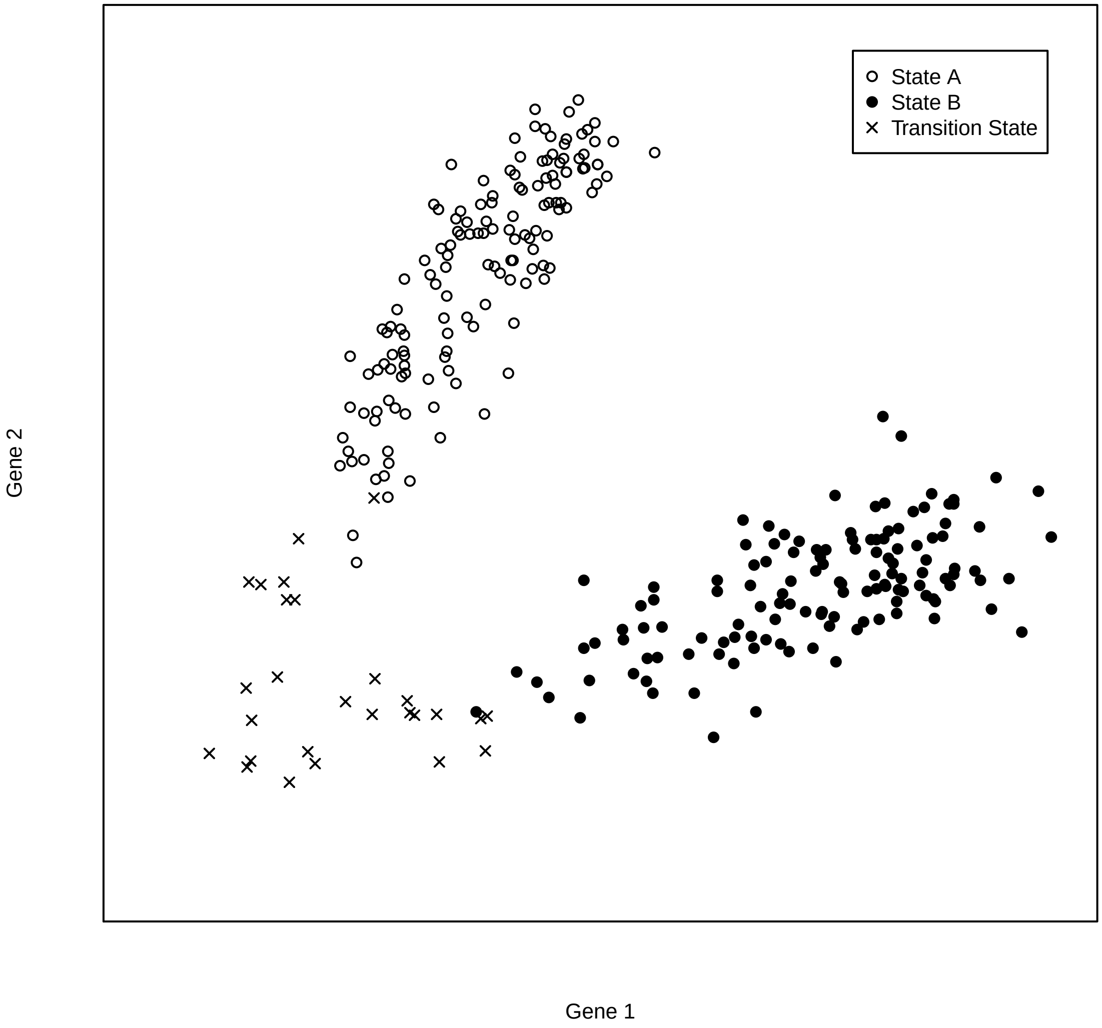
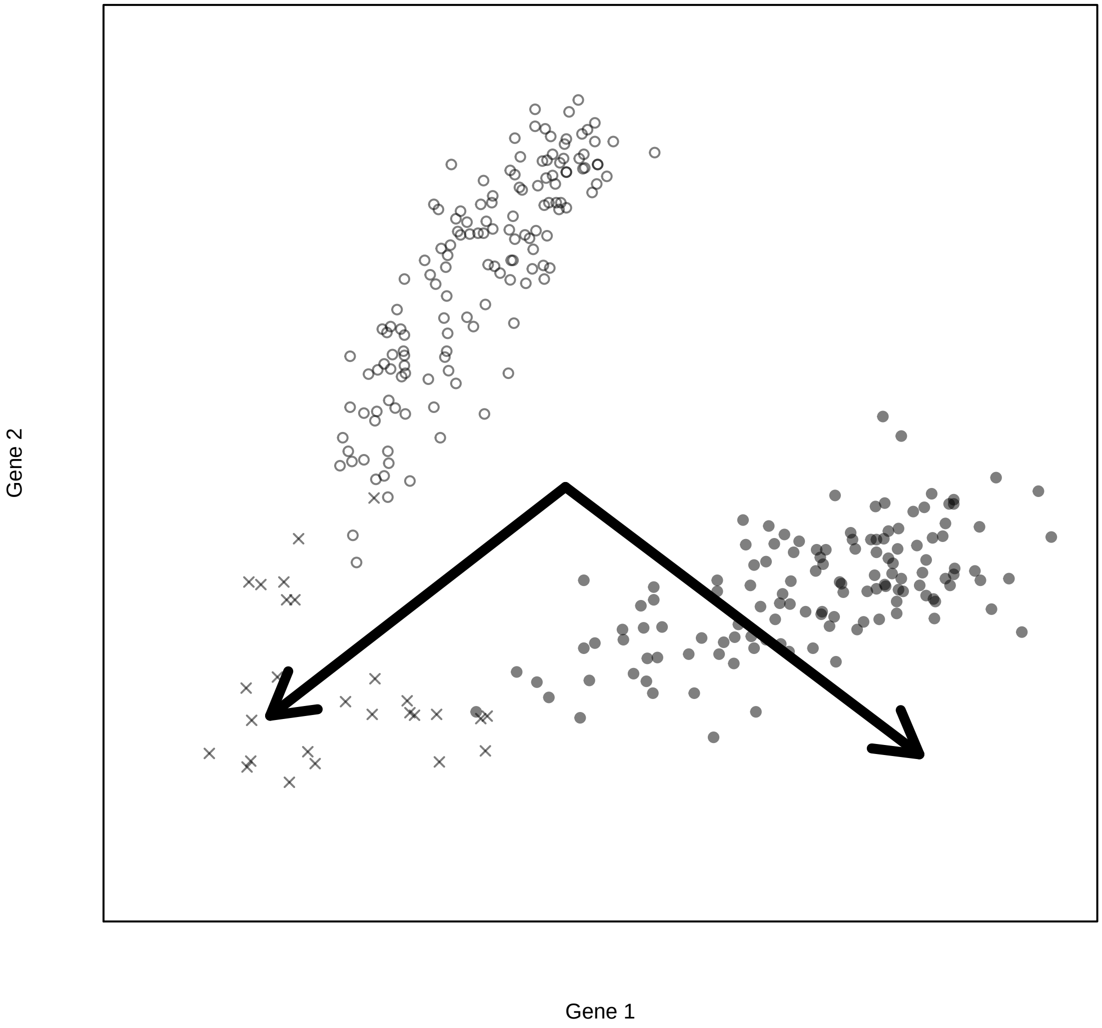
```
# xy = matrix(runif(10000), ncol = 2) xy = xy[xy[,1] < 0.1 | xy[,2] < 0.1,]
# xy = xy %*% matrix(c(1, 0.4, 0.4, 1), ncol = 2) xy = xy[xy[,1] <= 1 &
# xy[,2] <= 1,]

xyc = read.csv("synthetic_data.csv")
xy = xyc[, 1:2]

set.seed(1234)
subset = sample.int(nrow(xy), 300)

library(fastICA)
library(NMF)

## Loading required package:  methods
## Loading required package:  pkgmaker
## Loading required package:  registry
## Loading required package:  rngtools
## Loading required package:  cluster
## NMF - BioConductor layer [OK] | Shared memory capabilities [OK] | Cores 7/8

fit.pca = prcomp(xy, center = TRUE, scale = FALSE)

temp = replicate(1000, fastICA(xy, 2, method = "C"), simplify = FALSE)
temp2 = sapply(temp, function(x) shapiro.test(x$S)$statistic)
fit.ica = temp[[which.max(temp2)]]

fit.nmf = nmf(t(xy[subset, ]), rank = 2, nrun = 20, method = "snmf/r")
```

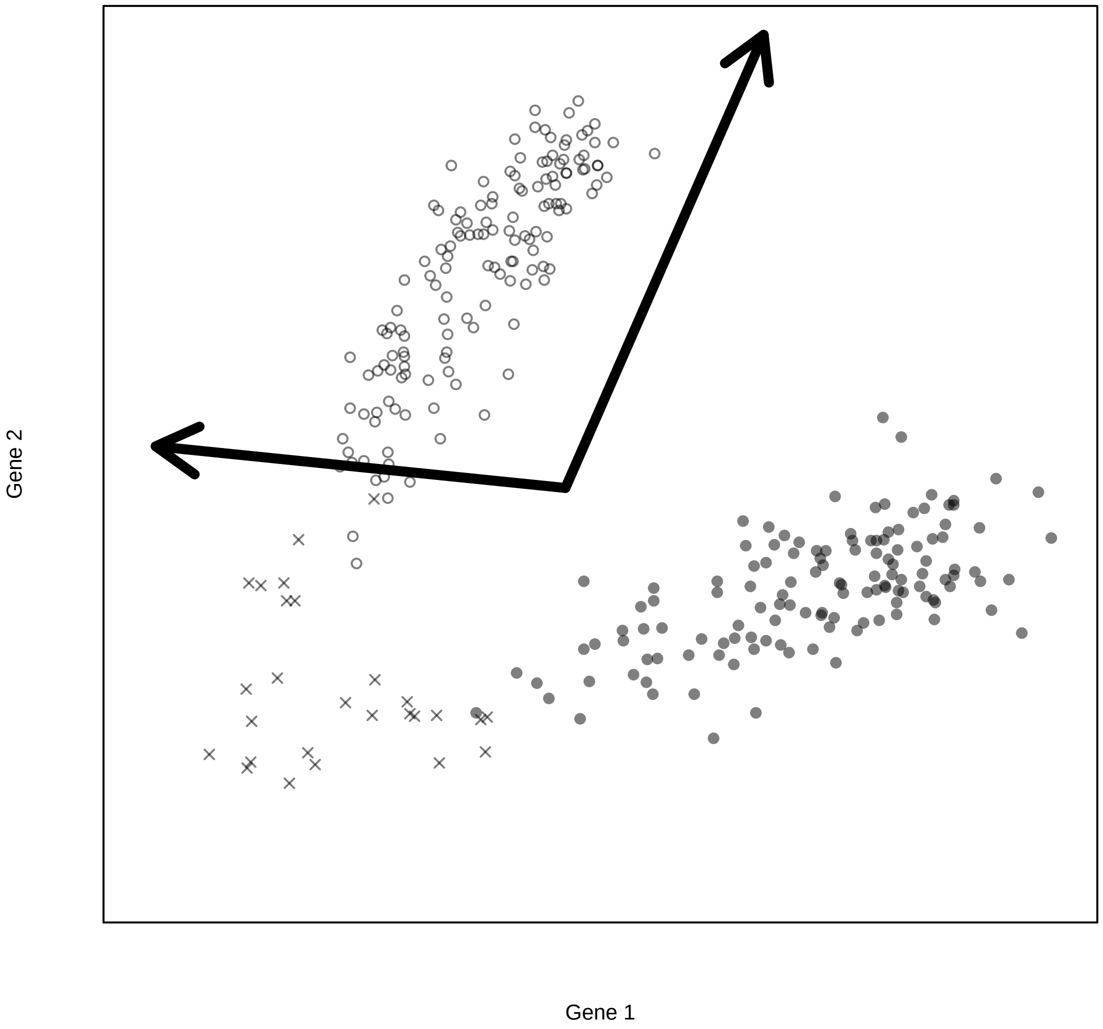
```
library(NMF)

## Loading required package:  methods
## Loading required package:  pkgmaker
## Loading required package:  registry
## Loading required package:  rngtools
## Loading required package:  cluster
## NMF - BioConductor layer [OK] | Shared memory capabilities [OK] | Cores 7/8

library(RColorBrewer)
pal = brewer.pal(3, "Set2")[c(2, 3, 1)]
pal = sapply(pal, function(col) do.call(rgb, c(as.list(col2rgb(col)/255), alpha = 0.5)))
syms = c(19, 4, 21)
col = pal[xyc[, 3]]
pch = syms[xyc[, 3]]

plot(0 ~ 0, type = "n", xlab = "Gene 1", ylab = "Gene 2", xlim = c(0, 1), ylim = c(0,
    1), xaxt = "n", yaxt = "n")
# arrows( x0 = c(0, 0), y0 = c(0, 0), x1 = c(0.5, 1), y1 = c(1, 0.45), col =
# 'lightgrey', lwd = 5)
points(xy[subset, 1], xy[subset, 2], col = "black", pch = pch[subset])
legend("topright", legend = c("State A", "State B", "Transition State"), pch = syms[c(3,
    1, 2)], inset = 0.05)
```



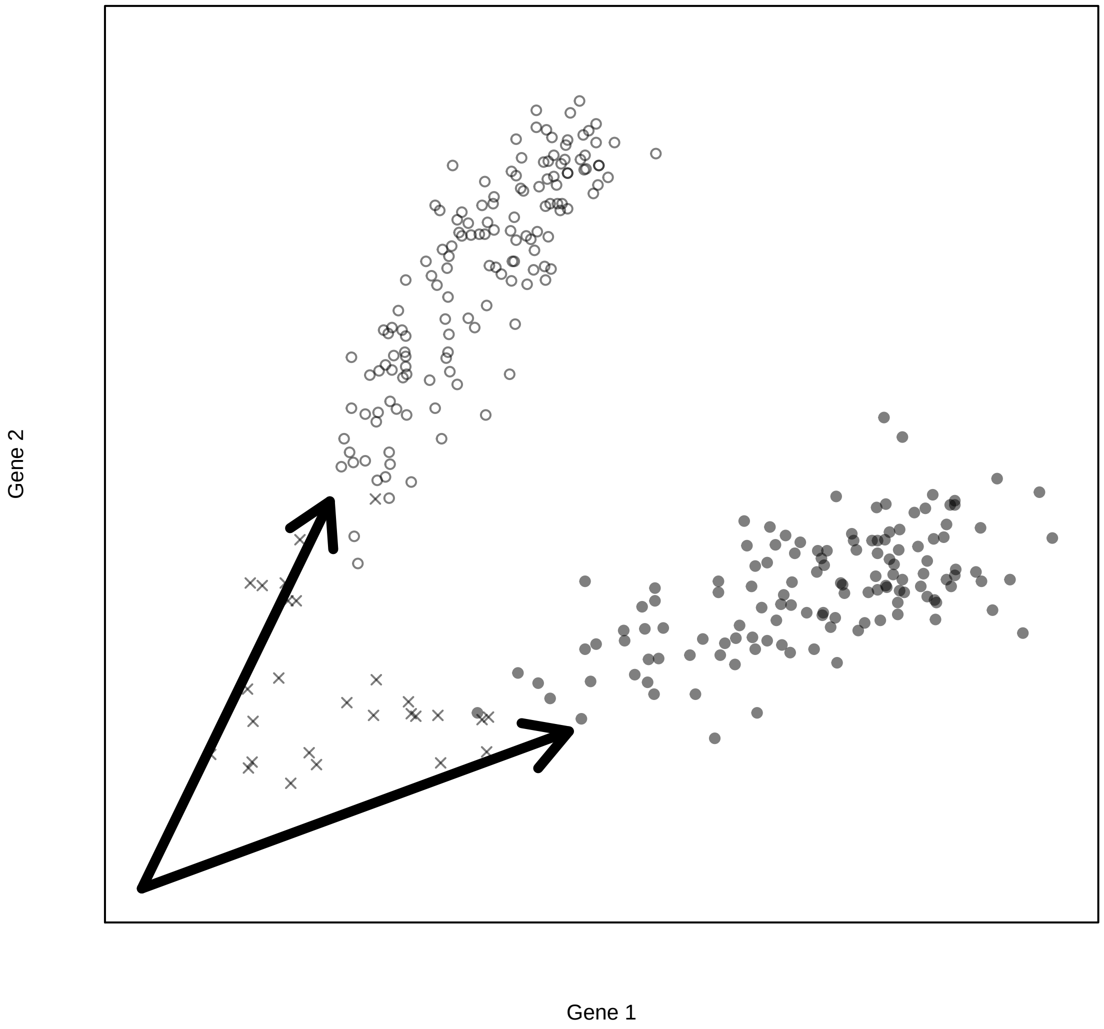
```
plot(xy[subset, 1], xy[subset, 2], col = rgb(0, 0, 0, 0.5), pch = pch[subset],
    xlab = "Gene 1", ylab = "Gene 2", xlim = c(0, 1), ylim = c(0, 1), xaxt = "n",
    yaxt = "n")
temp = fit.pca$rotation
temp = t(t(temp)/sqrt(colSums(temp^2))/2)
arrows(x0 = rep(mean(xy[, 1], 2)), y0 = rep(mean(xy[, 2], 2)), x1 = temp[1,
    ] + mean(xy[, 1]), y1 = temp[2, ] + mean(xy[, 2]), col = "black", lwd = 5)
```



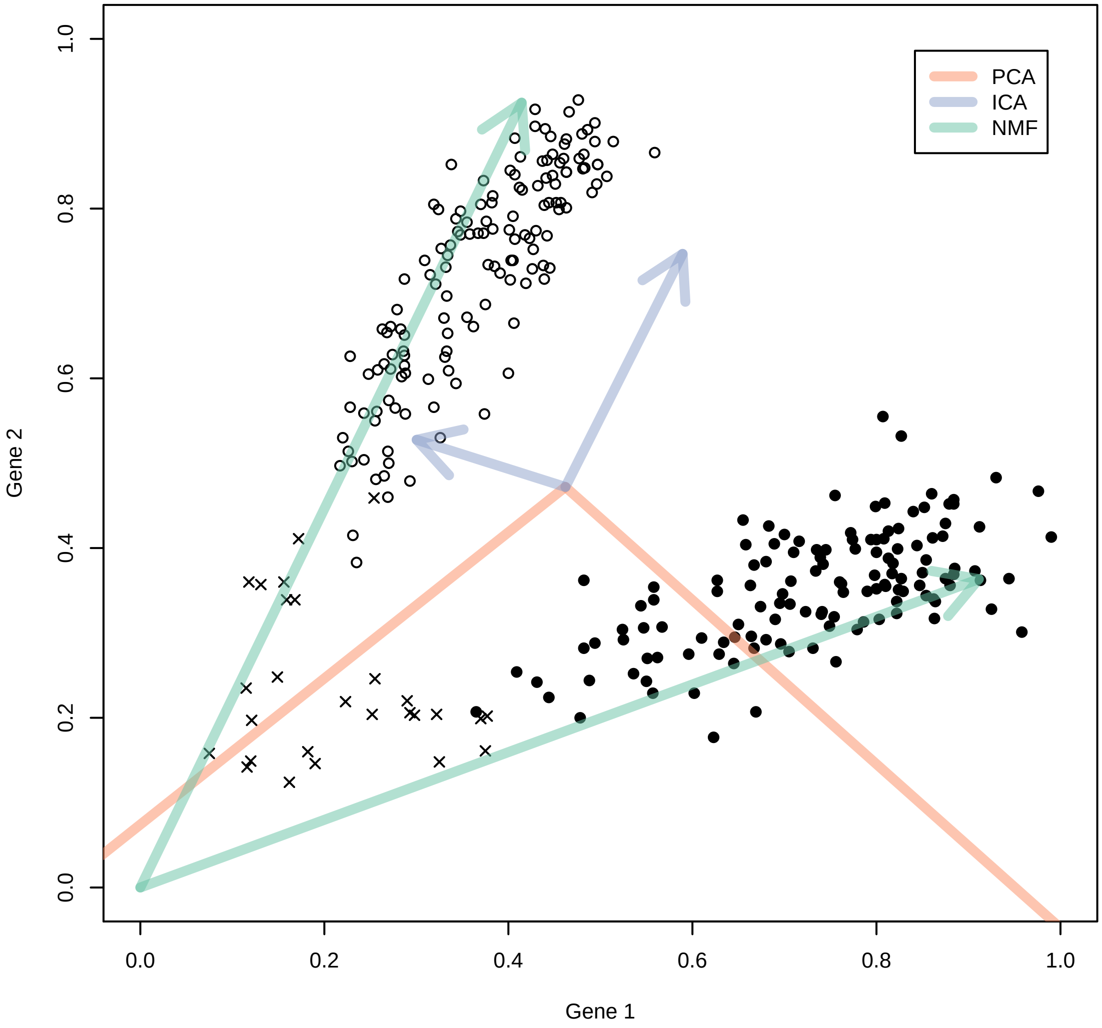
```
plot(xy[subset, 1], xy[subset, 2], col = rgb(0, 0, 0, 0.5), pch = pch[subset],
    xlab = "Gene 1", ylab = "Gene 2", xlim = c(0, 1), ylim = c(0, 1), xaxt = "n",
    yaxt = "n")
temp = fit.ica$A
temp = t(t(temp)/sqrt(colSums(temp^2))/2)
arrows(x0 = rep(mean(xy[, 1], 2)), y0 = rep(mean(xy[, 2], 2)), x1 = temp[1,
    ] + mean(xy[, 1]), y1 = temp[2, ] + mean(xy[, 2]), col = "black", lwd = 5)
```



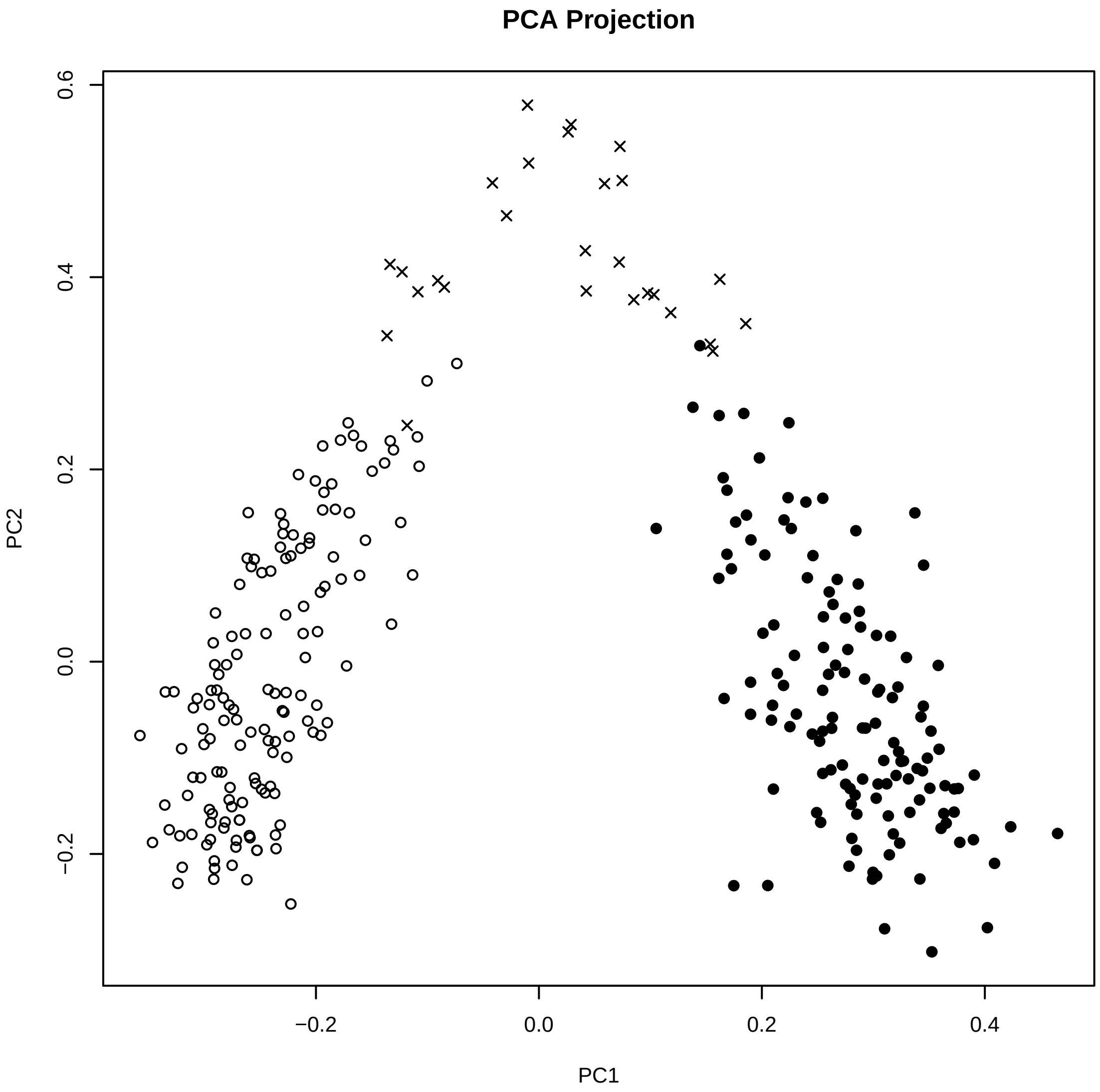
```
plot(xy[subset, 1], xy[subset, 2], col = rgb(0, 0, 0, 0.5), pch = pch[subset],
    xlab = "Gene 1", ylab = "Gene 2", xlim = c(0, 1), ylim = c(0, 1), xaxt = "n",
    yaxt = "n")
temp = basis(fit.nmf)
temp = t(t(temp)/sqrt(colSums(temp^2))/2)
arrows(x0 = c(0, 0), y0 = c(0, 0), x1 = temp[1, ], y1 = temp[2, ], col = "black",
    lwd = 5)
```



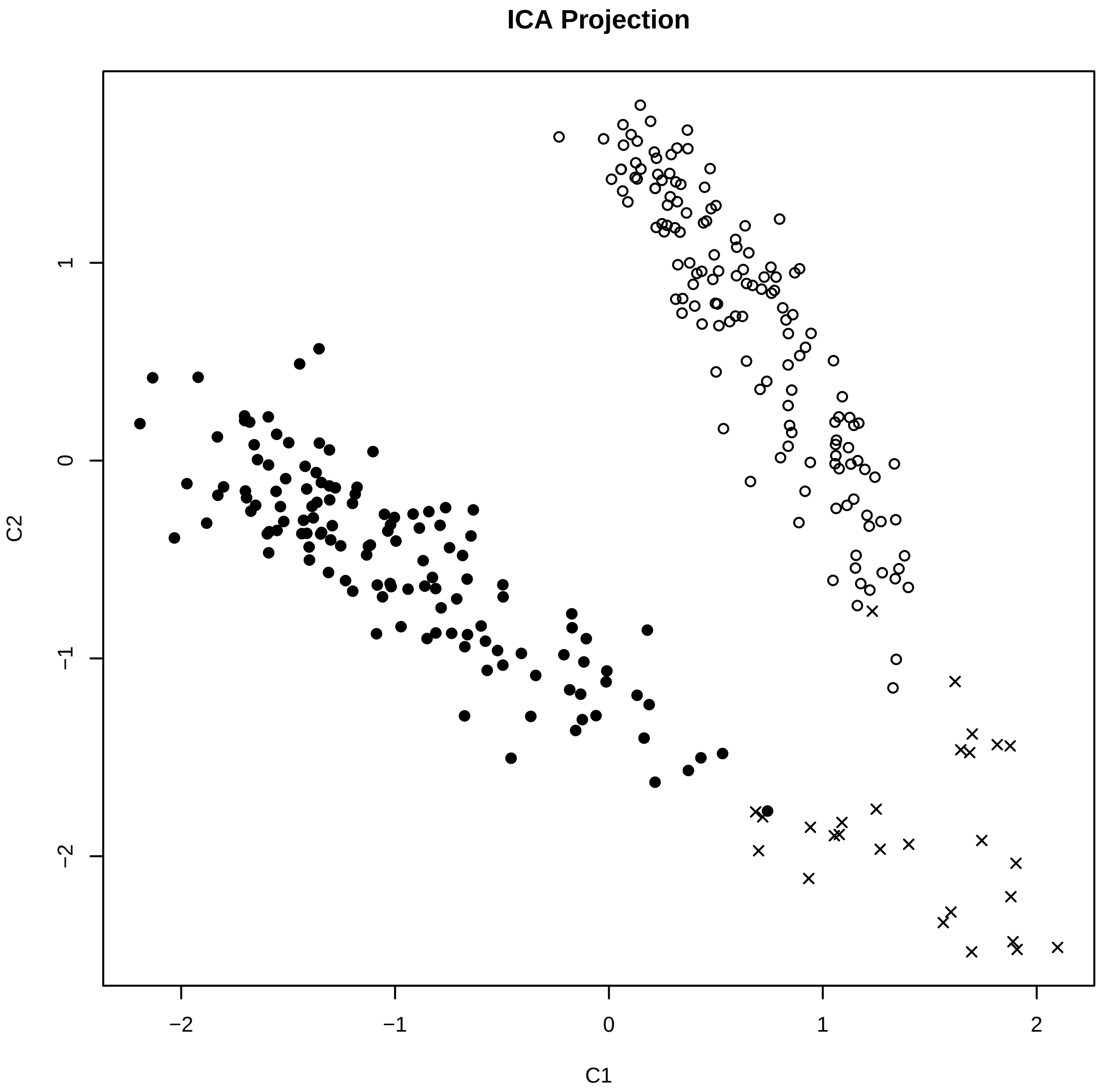
```
plot(xy[subset, 1], xy[subset, 2], col = "black", pch = pch[subset], xlab = "Gene 1",
    ylab = "Gene 2", xlim = c(0, 1), ylim = c(0, 1))
# legend('topright', legend = c('State A', 'State B', 'Transition State'),
# pch = syms[c(3, 1, 2)], inset = 0.05)
legend("topright", legend = c("PCA", "ICA", "NMF"), lty = "solid", lwd = 5,
    col = pal, inset = 0.05)
arrows(x0 = rep(mean(xy[, 1], 2)), y0 = rep(mean(xy[, 2], 2)), x1 = fit.pca$rotation[1,
    ] + mean(xy[, 1]), y1 = fit.pca$rotation[2, ] + mean(xy[, 2]), col = pal[1],
    lwd = 5)
arrows(x0 = rep(mean(xy[, 1], 2)), y0 = rep(mean(xy[, 2], 2)), x1 = fit.ica$A[1,
    ] + mean(xy[, 1]), y1 = fit.ica$A[2, ] + mean(xy[, 2]), col = pal[2], lwd = 5)
arrows(x0 = c(0, 0), y0 = c(0, 0), x1 = basis(fit.nmf)[1, ], y1 = basis(fit.nmf)[2,
    ], col = pal[3], lwd = 5)
```



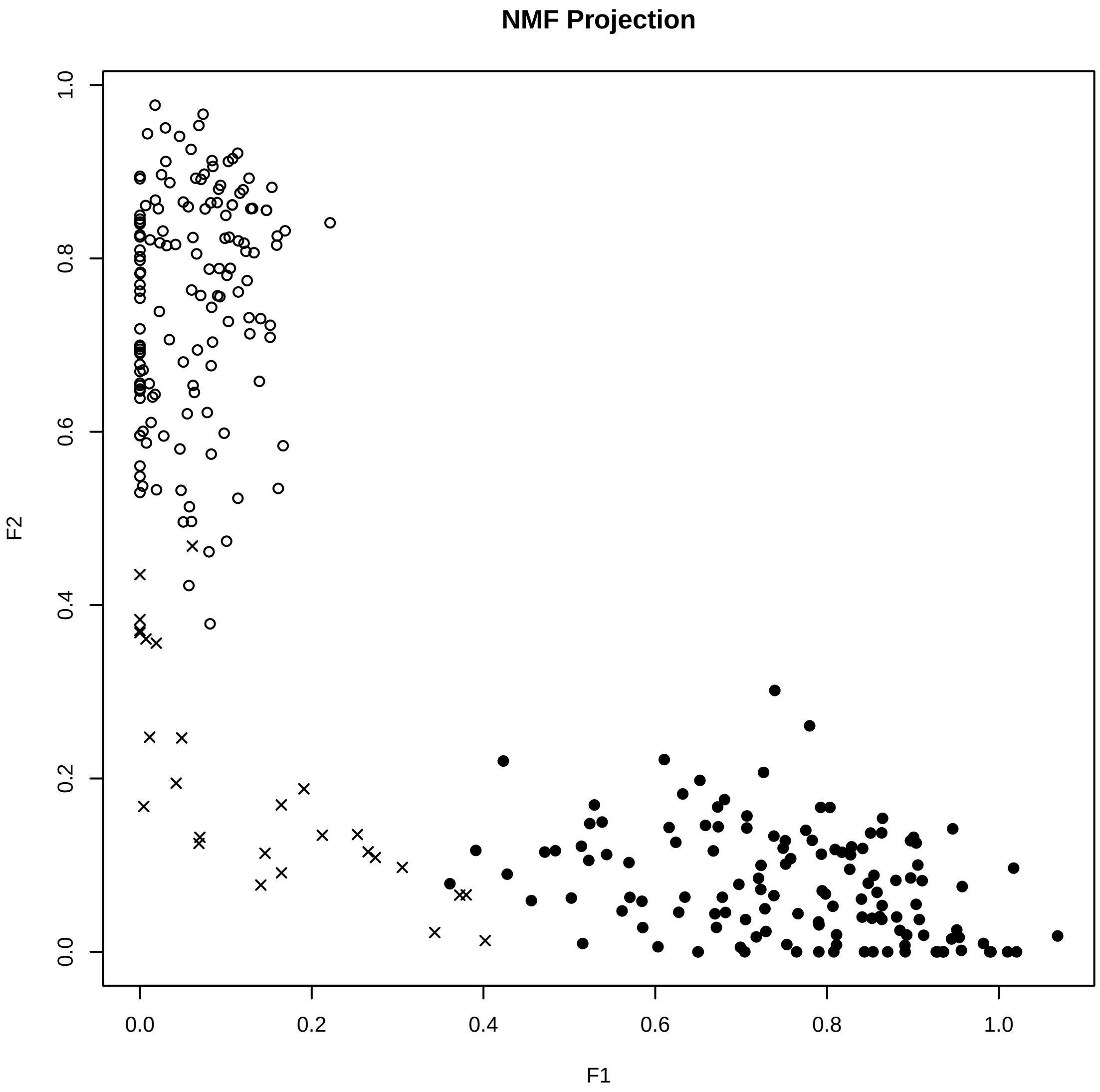
```
plot(fit.pca$x[subset, ], pch = pch[subset], xlab = "PC1", ylab = "PC2", main = "PCA Projection")
```



```
plot(fit.ica$S[subset, ], pch = pch[subset], xlab = "C1", ylab = "C2", main = "ICA Projection")
```



```
plot(t(coef(fit.nmf)), pch = pch[subset], xlab = "F1", ylab = "F2", main = "NMF Projection")
```


NMF Projection
F2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
F1