STAT 545

THREE TASKS

1. EDA for Multivariate Structure
2. Summarizing / Estimating Univariate Distribution
3. Permutation version of Hypothesis Test

1. Forensic Glass Dataset

214 Glass fragments, of seven types

Nine measurements of physical characteristics on each fragment

Could the unknown type of future fragments be well estimated based on these measurements?

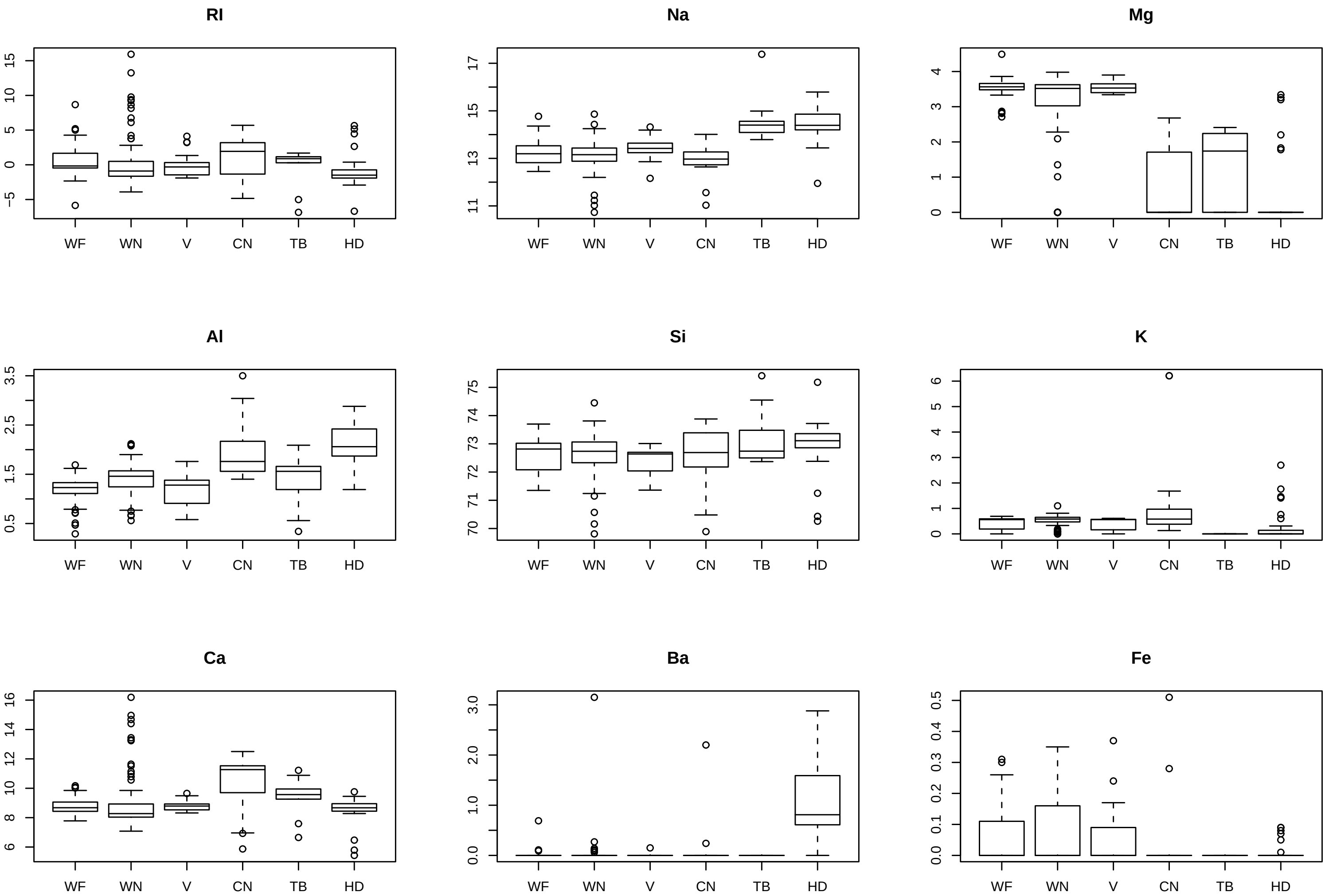
Relevant graphical summaries?

```
> names(fgl)
 [1] "RI"   "Na"   "Mg"   "Al"   "Si"   "K"
     "Ca"   "Ba"   "Fe"   "type"
> levels(fgl$type)
[1] "WinF"  "WinNF" "Veh"   "Con"   "Tabl"  "Head"
> levels(fgl$type) <- c("WF","WN","V","CN","TB","HD")
```

```
boxplot(RI~type, data=fgl)
boxplot(Na~type, data=fgl)
...
```

Or automate:

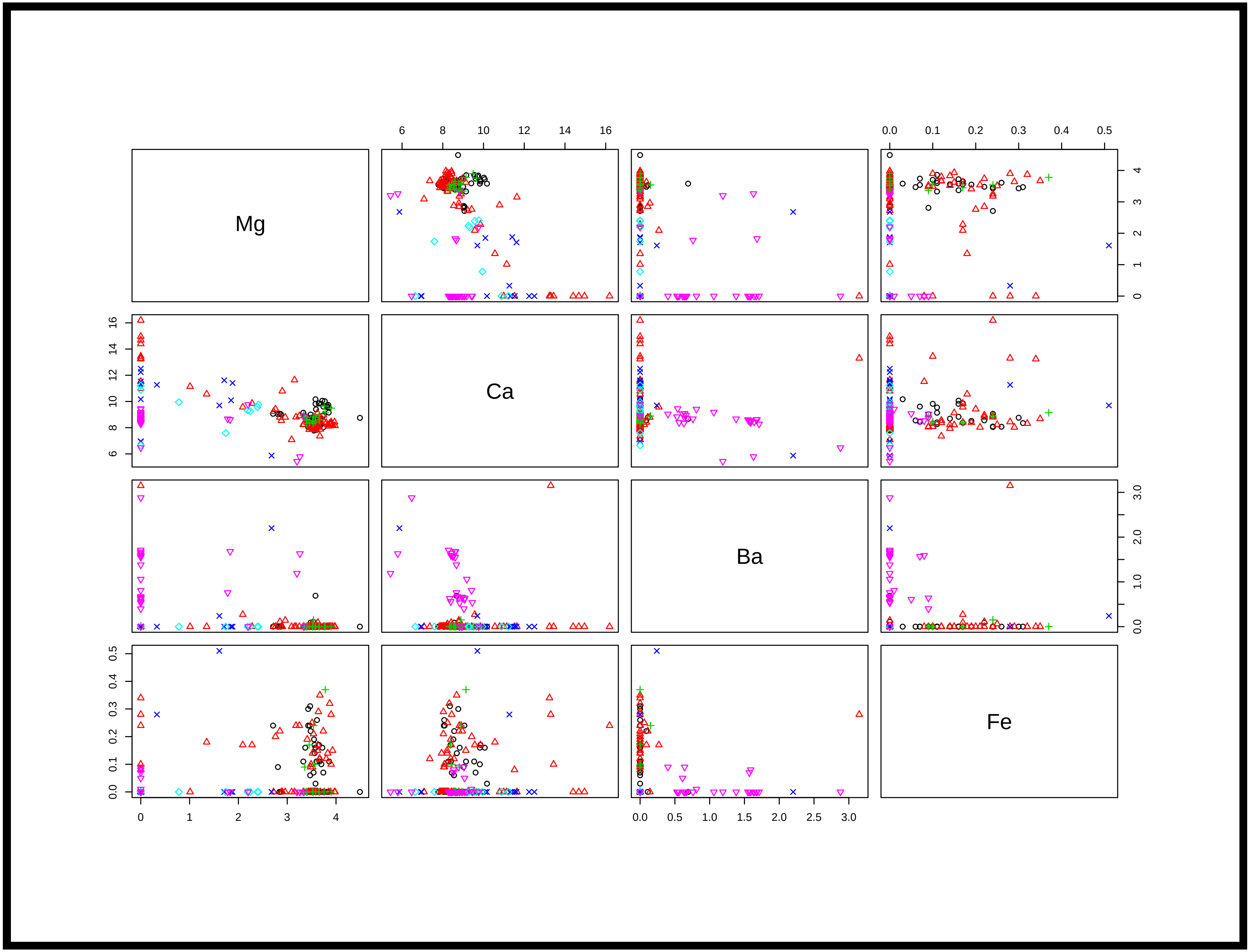
```
for (i in 1:9) {
  boxplot(fgl[[i]]~fgl$type, cex.lab=0.5)
  title(names(fgl)[i]) }
```


RI
15
10
5
0
−5
WF
WN
V
CN
TB
HD
Na
17
15
13
11
WF
WN
V
CN
TB
HD
Mg
4
3
2
1
0
WF
WN
V
CN
TB
HD
Al
3.5
2.5
1.5
0.5
WF
WN
V
CN
TB
HD
Si
75
74
73
72
71
70
WF
WN
V
CN
TB
HD
K
6
5
4
3
2
1
0
WF
WN
V
CN
TB
HD
Ca
16
14
12
10
8
6
WF
WN
V
CN
TB
HD
Ba
3.0
2.0
1.0
0.0
WF
WN
V
CN
TB
HD
Fe
0.5
0.4
0.3
0.2
0.1
0.0
WF
WN
V
CN
TB
HD

Say we thought Mg, Ba, Fe, Ca were the most promising features for classification. One thing to try would be to look for pairwise structure...

```
> pairs(fgl[,c(3,7,8,9)],
        pch=as.numeric(fgl$type),
        col=as.numeric(fgl$type) )
```

Added value over univariate boxplots in this case???

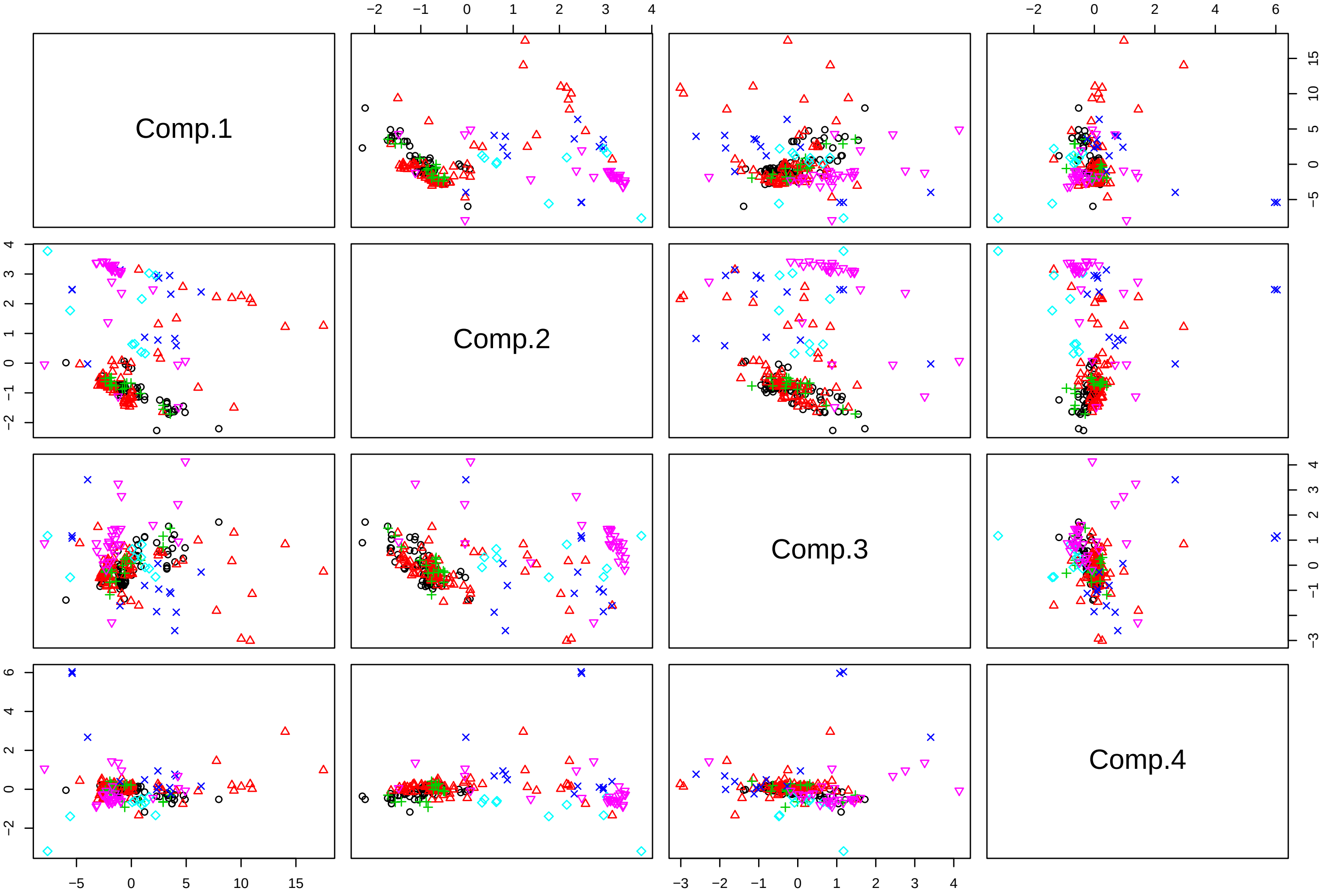

Mg
Ca
Ba
Fe
6
8
10
12
14
16
0.0
0.1
0.2
0.3
0.4
0.5
0
1
2
3
4
0.0
1.0
2.0
3.0
0
1
2
3
4
0.0
0.5
1.0
1.5
2.0
2.5
3.0

Maybe the leading principal components would be more promising for classification?

Why? Maximal variance might be ‘caused’ by type?

```
> tmp <- predict(princomp(fgl[,1:9]))

> pairs(tmp[,1:4],
        pch=as.numeric(fgl$type),
        col=as.numeric(fgl$type) )
```

Helpful? Might be time for formal classification techniques...


Comp.1
Comp.2
Comp.3
Comp.4
−2
−1
0
1
2
3
4
−2
0
2
4
6
15
10
5
0
−5
4
3
2
1
0
−1
−2
4
3
2
1
0
−1
−3
6
4
2
0
−2
−5
0
5
10
15
−3
−2
−1
0
1
2
3
4

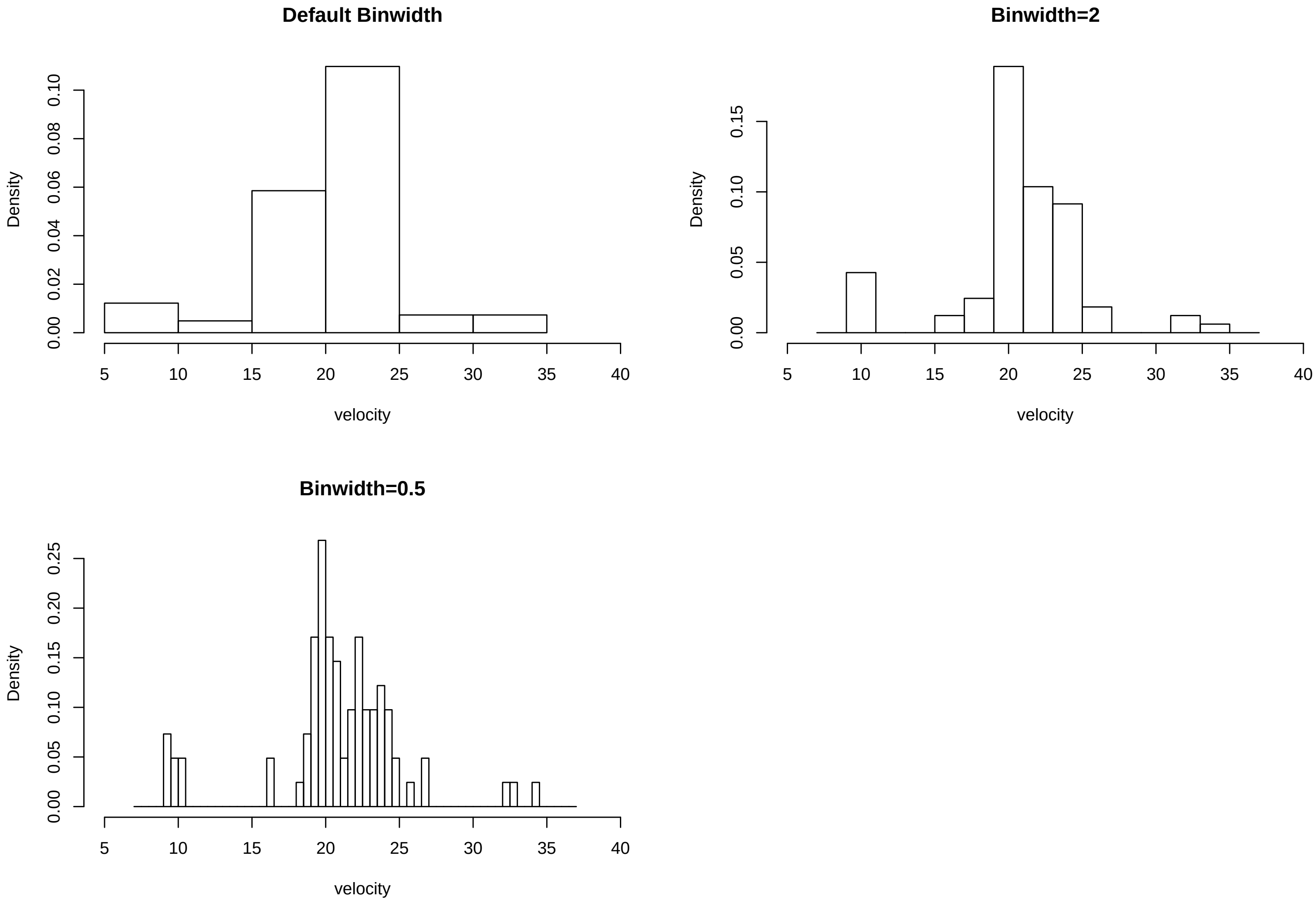
1. Kernel Density Estimation

galaxies dataset, velocities at which $n = 82$ galaxies in the Corona Borealis region are moving away from our galaxy

Distribution of these velocities? Claim that unimodal versus bimodal is scientifically relevant!

Histograms - ‘boxy’ and boring! - except for choice of binwidth

What about smooth summary or density estimate.


Default Binwidth
Density
0.00
0.02
0.04
0.06
0.08
0.10
5
10
15
20
25
30
35
40
velocity
Binwidth=2
Density
0.00
0.05
0.10
0.15
5
10
15
20
25
30
35
40
velocity
Binwidth=0.5
Density
0.00
0.05
0.10
0.15
0.20
0.25
5
10
15
20
25
30
35
40
velocity

Kernel Estimator

$X_1, \ldots, X_n$ *iid*, unknown density $f()$

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{b} k\left(\frac{x - x_i}{b}\right)$$

for some density function (kernel) $k()$, and some bandwith $b > 0$.

Intuition???
What happens as b changes?

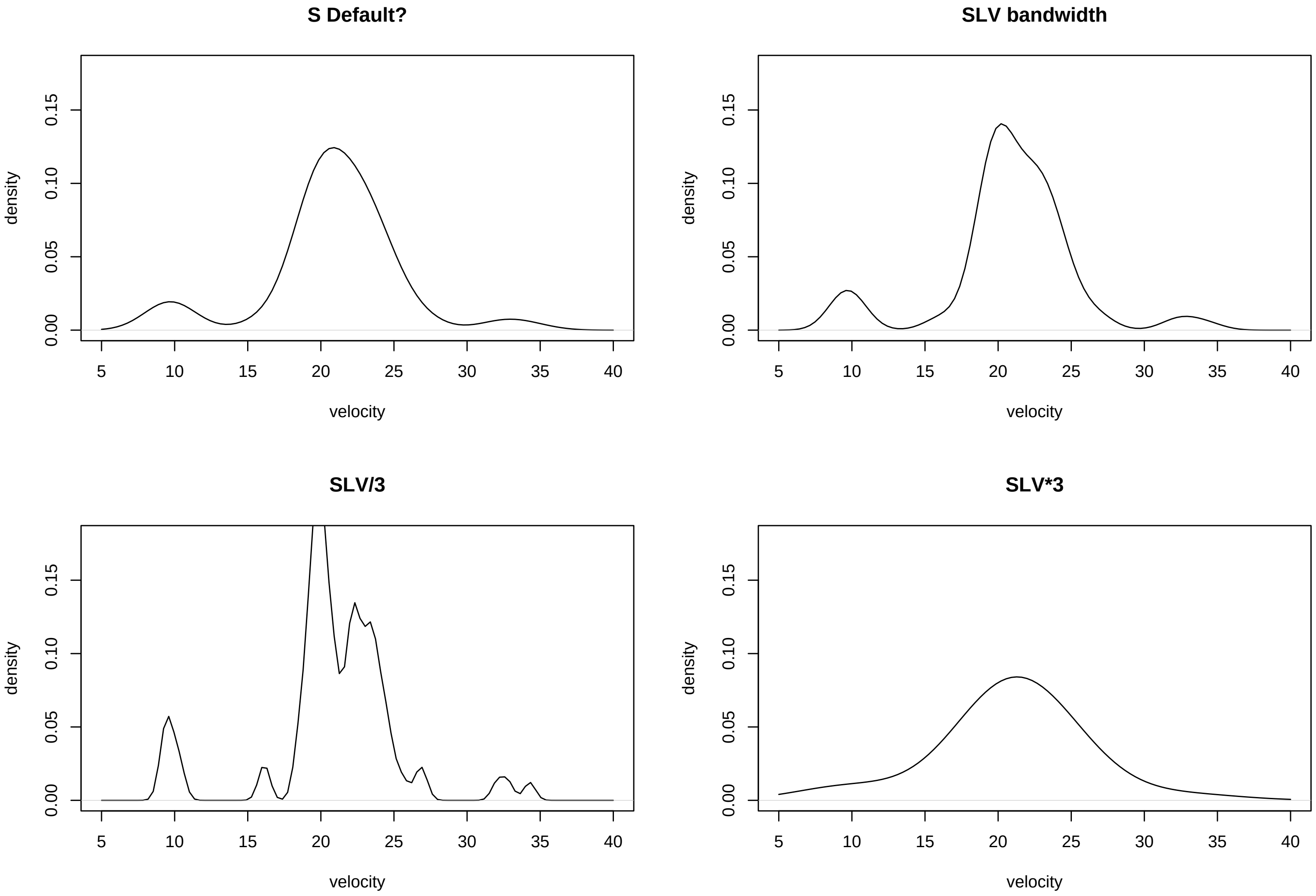
Choose b by ‘eye’? Automated choice?

S-Plus default:

$$b \quad = \quad \frac{\text{range}(x)}{2(1+\log_2 n)}$$

Often cited Silverman (1986) suggestion:

$$b \quad = \quad 1.06 \times \min\left\{ SD(x), \frac{IQR(x)}{1.34} \right\} \times n^{-1/5}$$



S Default?

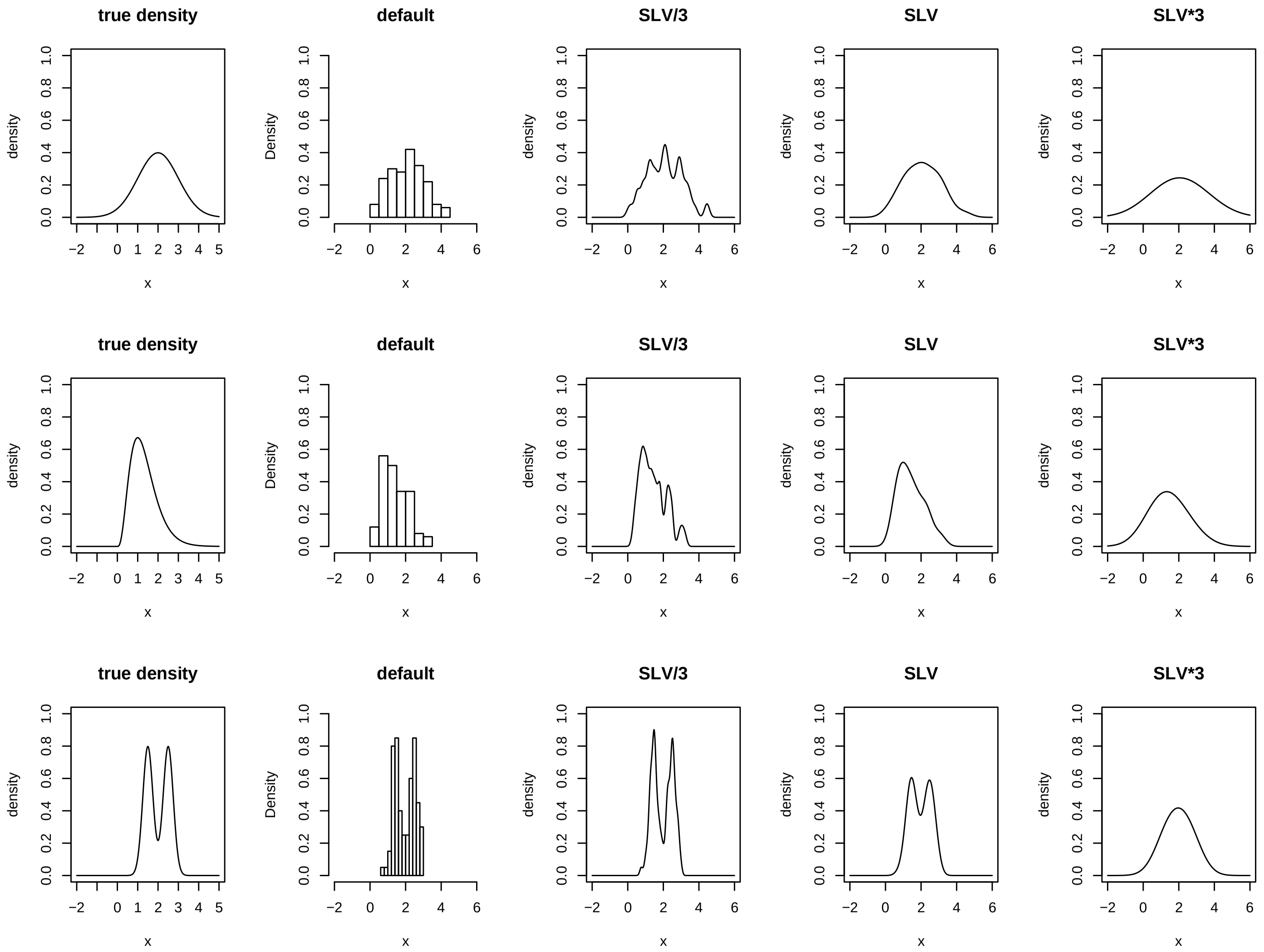
SLV bandwidth

SLV/3

SLV*3

Small simulation experiment to check out Silverman's proposal, e.g.

```
n <-100
x <- rgamma(n,shape=4)/3
sd <- sqrt(var(x))
iqr <- sum(quantile(x,c(.25,.75))*c(-1,1))
bw <- 1.06 * min(sd , iqr/1.34) * (n^(-1/5))
opt <- density(x, n=300, from=-2, to=6, bw=bw)
plot(opt$x,opt$y,type="l",xlab="x",ylab="density")
```

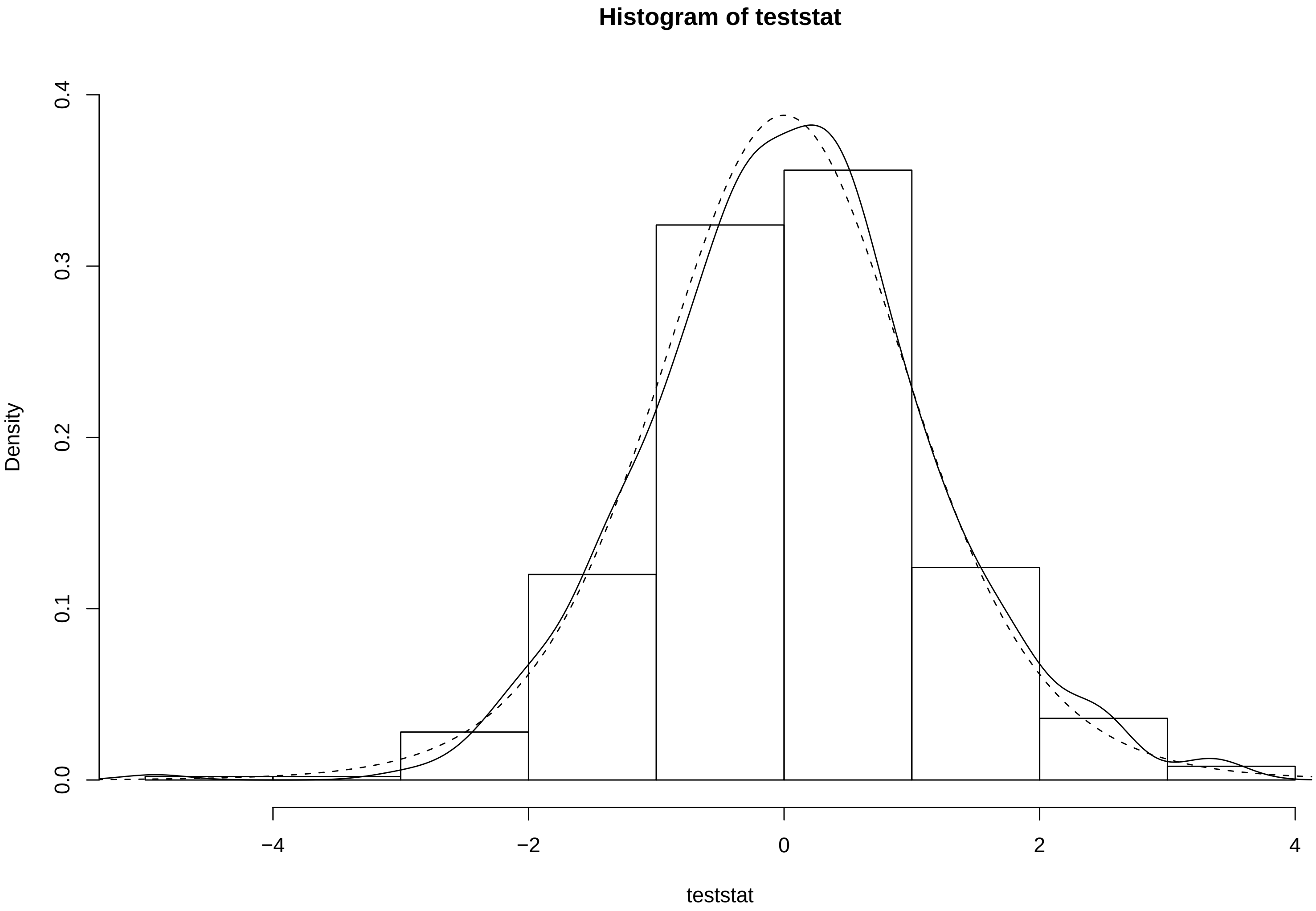

true density
default
SLV/3
SLV
SLV*3
density
Density
x
0.0
0.2
0.4
0.6
0.8
1.0
−2
0
1
2
3
4
5
6

3. Permutation Test

```
> shoes
$A
 [1] 13.2  8.2 10.9 14.3 10.7  6.6  9.5 10.8  8.8 13.3
$B
 [1] 14.0  8.8 11.2 14.2 11.8  6.4  9.8 11.3  9.3 13.6

> cor(shoes$A, shoes$B)
[1] 0.9882255

> dif <- shoes$A-shoes$B
> mean(dif)/sqrt(var(dif)/10)
[1] -3.348877
> pt(-3.35, df=9)
[1] 0.004261772
```

Large-sample approx???

```
teststat <- rep(NA, 500)
for (i in 1:500) {
  perm <- sample(c(-1,1), size=10, replace=T)
  teststat[i] <- mean( abs(dif)*perm ) /
                 sqrt( var(abs(dif)*perm)/10 ) }

> mean(teststat<(-3.35))
[1] 0.002

hist(teststat, prob=T)
tmp <- density(teststat)
points(tmp$x, tmp$y, type="l")
points(tmp$x, dt(tmp$x, df=9), type="l", lty=2)
```


Histogram of teststat
Density
0.0
0.1
0.2
0.3
0.4
−4
−2
0
2
4
teststat