Chapter 1
Exploratory data analysis

Xavier Gendre

M2 SE

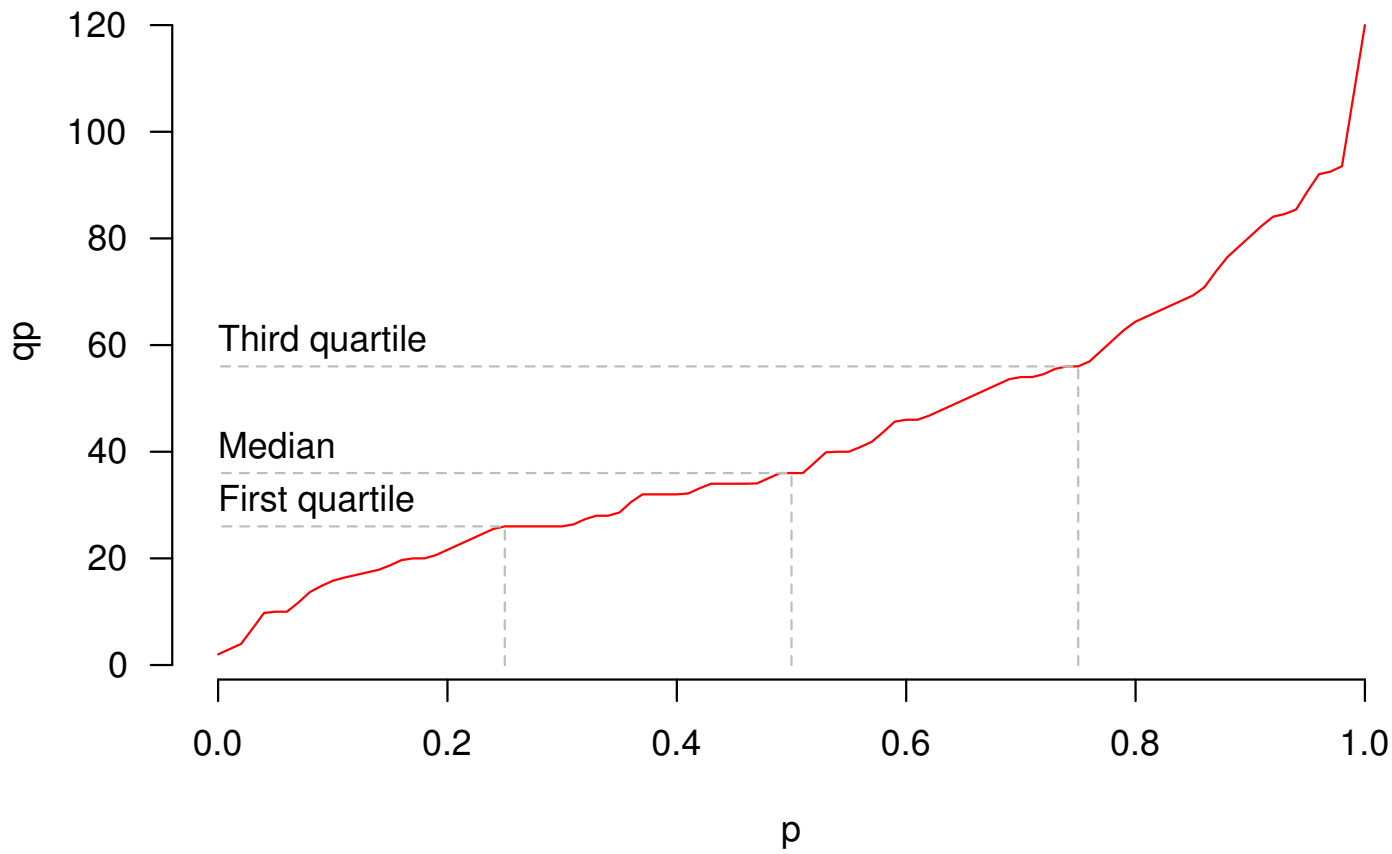
```
cars$dist
```

```
##  [1]   2  10   4  22  16  10  18  26  34  17  28  14  20  24  28  26  34
## [18]  34  46  26  36  60  80  20  26  54  32  40  32  40  50  42  56  76
## [35]  84  36  46  68  32  48  52  56  64  66  54  70  92  93 120  85
```

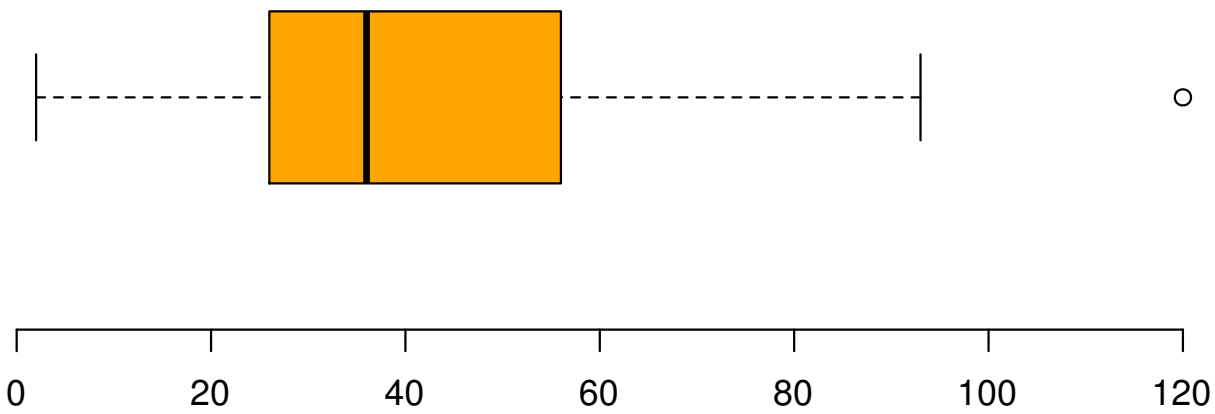
```
summary(cars$dist)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2      26      36      43      56     120
```

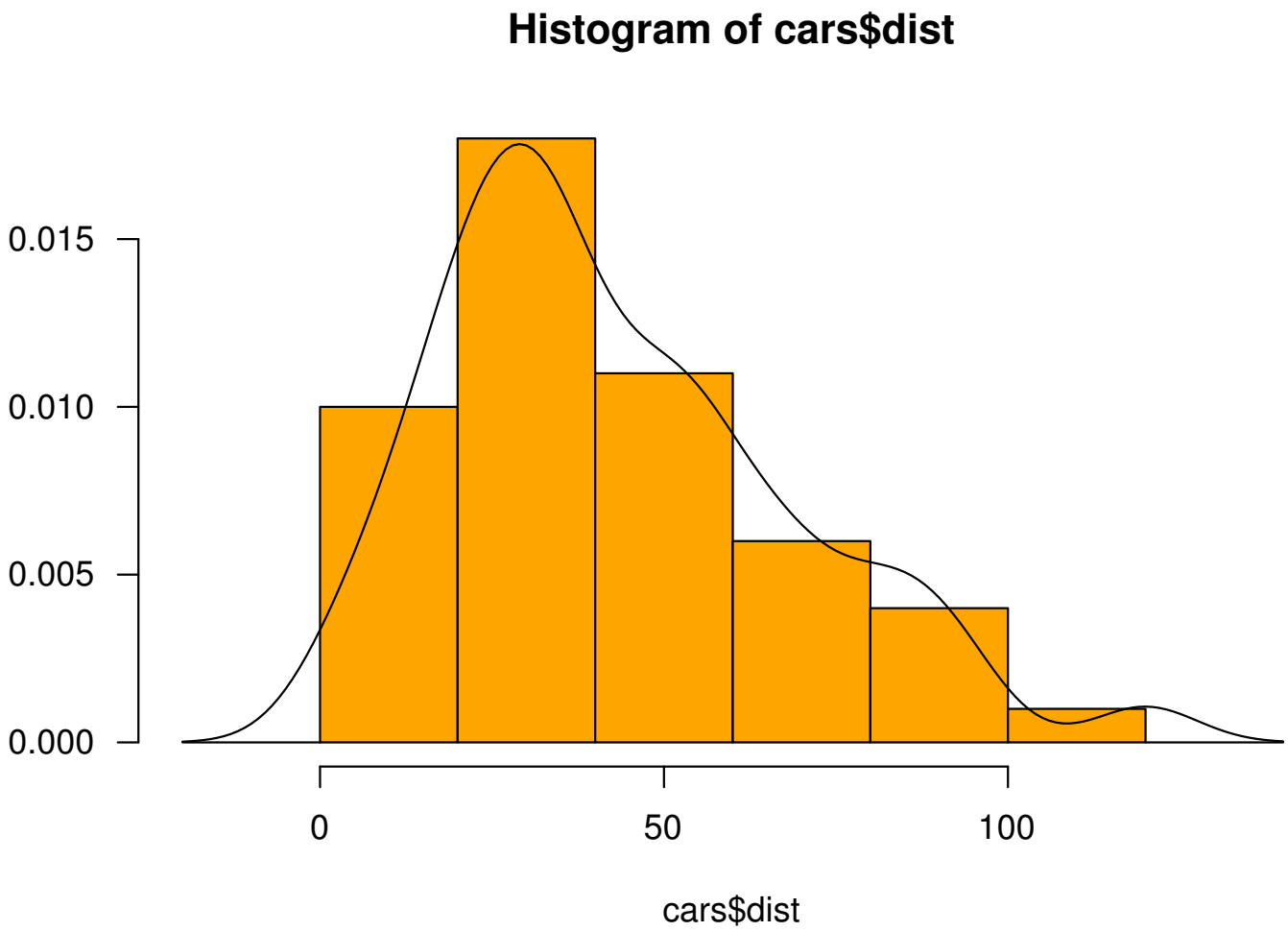
```
p <- (0:100)/100
qp <- quantile(cars$dist, probs = p)
plot(p, qp, type = "l", col = "red")
```



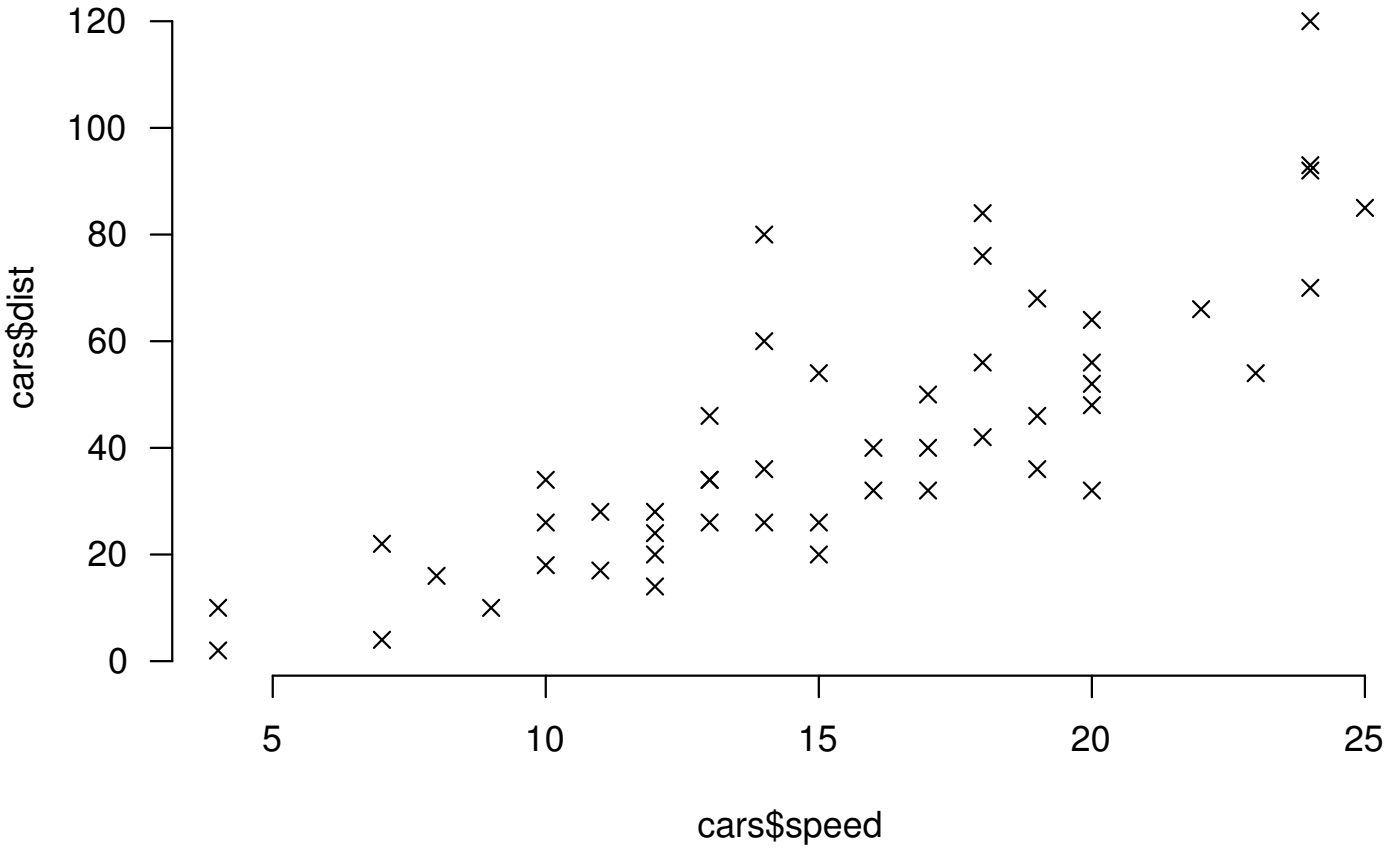
```
boxplot(cars$dist, horizontal = TRUE, col = "orange")
```



```
hist(cars$dist, breaks = 20 * (-1:7), freq = F, col = "orange", ylab = "")
```



```
plot(cars$speed, cars$dist, pch = 4)
```

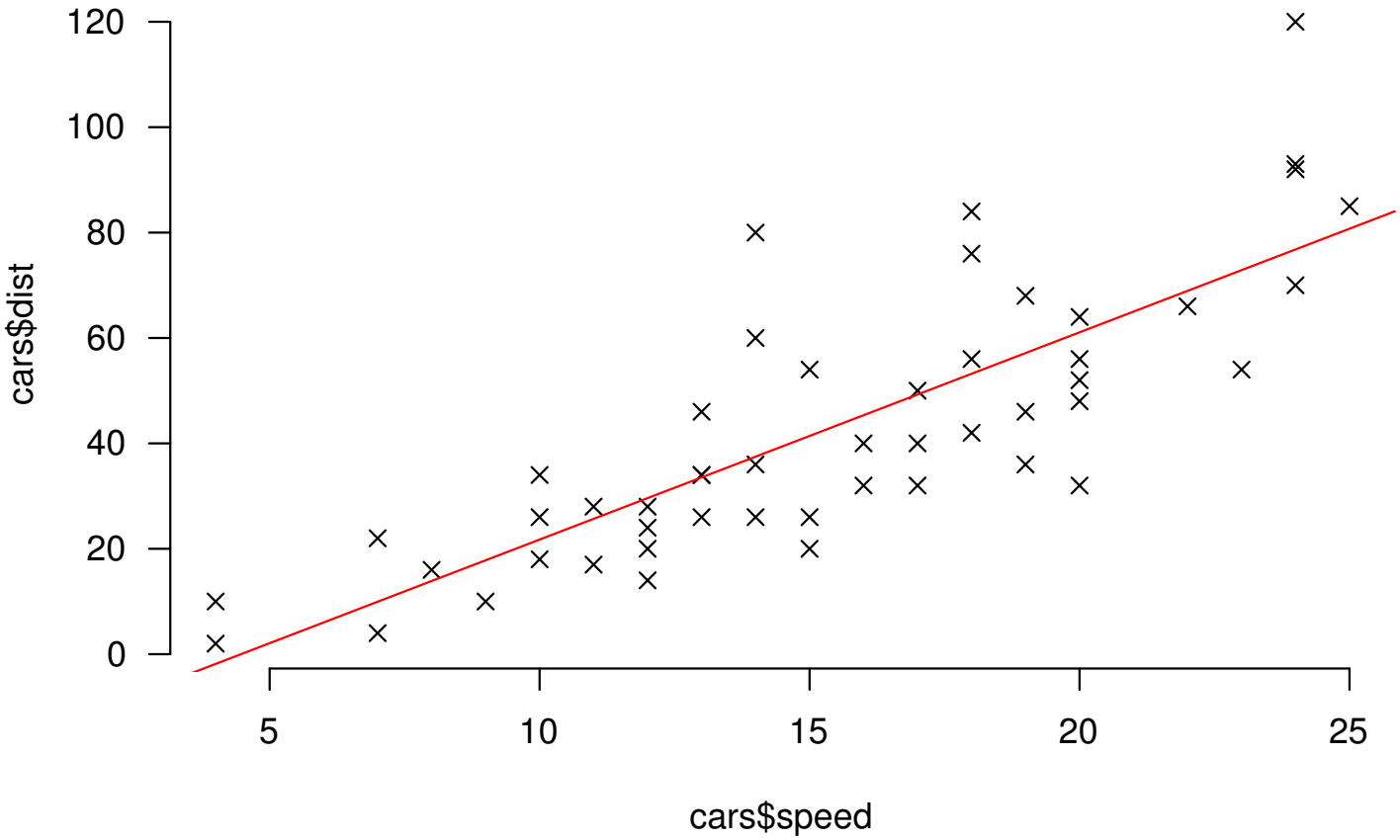


```
cor(cars$speed, cars$dist)

## [1] 0.8069

a <- cov(cars$speed, cars$dist)/var(cars$speed)
b <- mean(cars$dist) - a * mean(cars$speed)
```

```
abline(b, a, col = "red")
```



Grades data

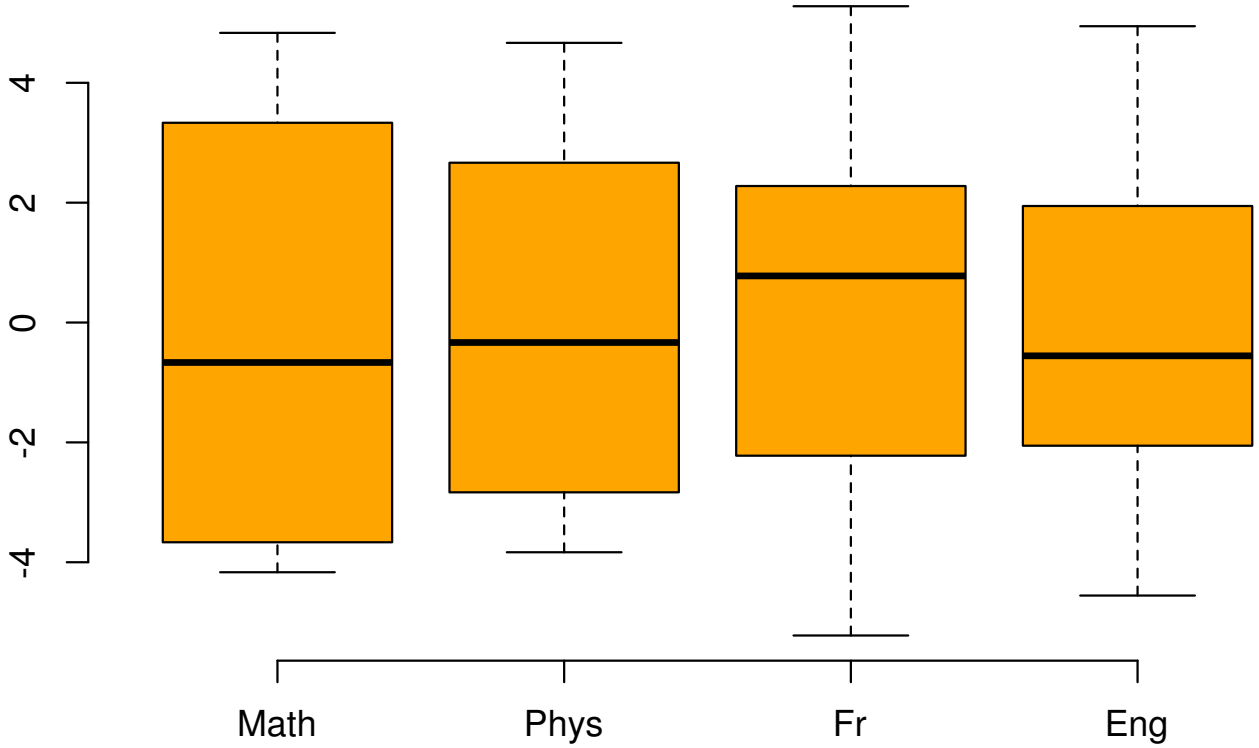
```
X
```

```
##        Math Phys   Fr  Eng
## Benny   6.0  6.0  5.0  5.5
## Bobby   8.0  8.0  8.0  8.0
## Brandy  6.0  7.0 11.0  9.5
## Coby   14.5 14.5 15.5 15.0
## Daisy  14.0 14.0 12.0 12.5
## Emily  11.0 10.0  5.5  7.0
## Judy    5.5  7.0 14.0 11.5
## Marty  13.0 12.5  8.5  9.5
## Sandy   9.0  9.5 12.5 12.0
```

```
Xbar <- scale(X, scale = F)
Sigma <- (t(Xbar) %*% Xbar)/9
```

```
##        Math  Phys     Fr   Eng
## Math 11.389 9.917  2.657 4.824
## Phys  9.917 8.944  4.120 5.481
## Fr    2.657 4.120 12.062 9.293
## Eng   4.824 5.481  9.293 7.914
```

```
boxplot(Xbar, col = "orange")
```



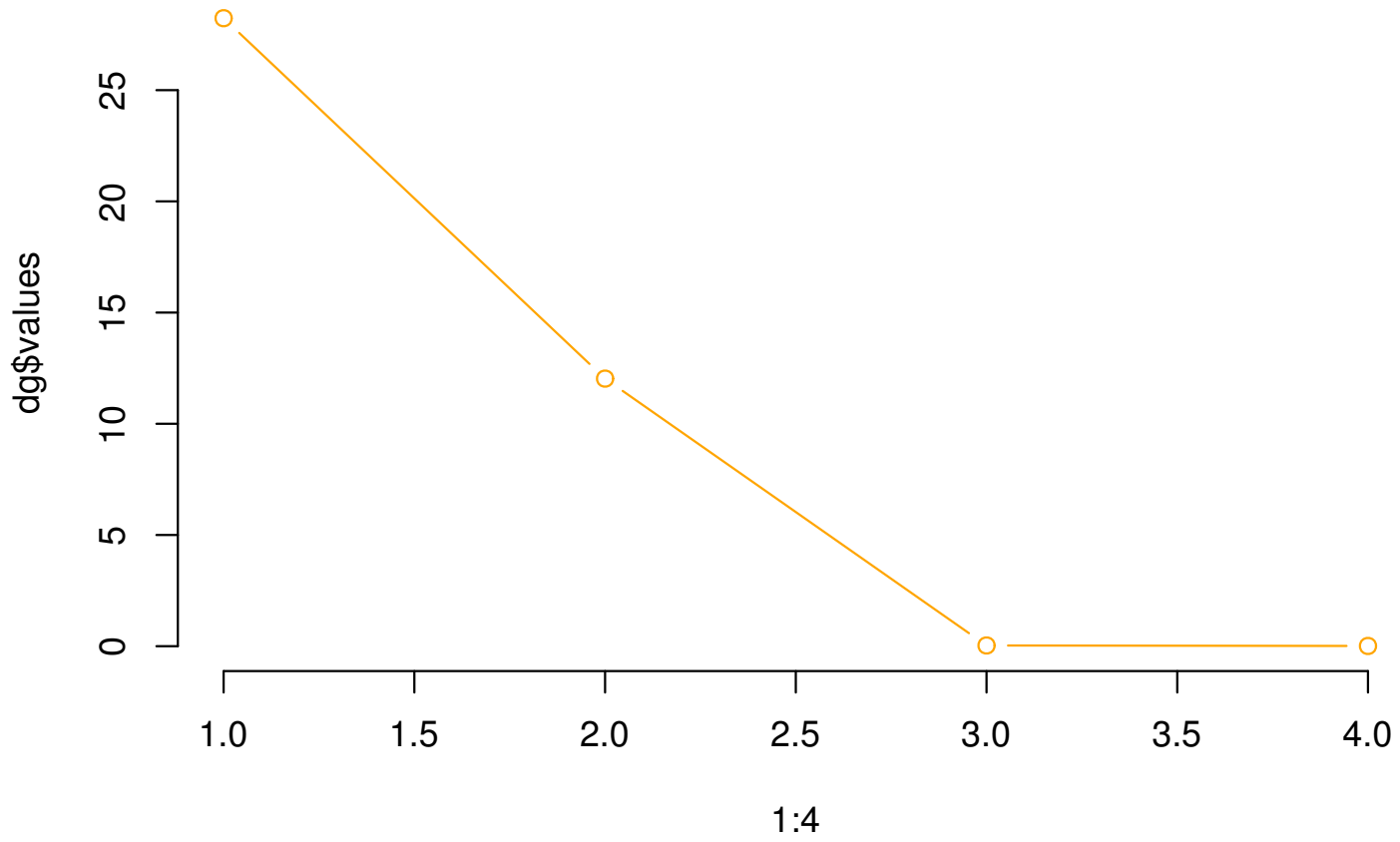
```
dg <- eigen(Sigma)
dg$values
```

```
## [1] 28.23487 12.03055  0.03263  0.01059
```

```
cumsum(dg$values)/sum(dg$values)
```

```
## [1] 0.7005 0.9989 0.9997 1.0000
```

```
plot(1:4, dg$values, type = "b", col = "orange")
```



Skyrim bows data

```
X
```

	Weight	Value	Damage	Speed
Long Bow	5	30	6	1.0000
Hunting Bow	7	50	7	0.9375
Orcish Bow	9	150	10	0.8125
Nord Hero Bow	7	200	11	0.8750
Dwarven Bow	10	270	12	0.7500
Elven Bow	12	470	13	0.6875
Glass Bow	14	820	15	0.6250
Ebony Bow	16	1440	17	0.5625
Daedric Bow	18	2500	19	0.5000
Dragonbone Bow	20	2725	20	0.7500
Crossbow	14	120	19	1.0000
Enhanced Crossbow	15	200	19	1.0000
Dwarven Crossbow	20	350	22	1.0000
Enhanced Dwarven Crossbow	21	550	22	1.0000

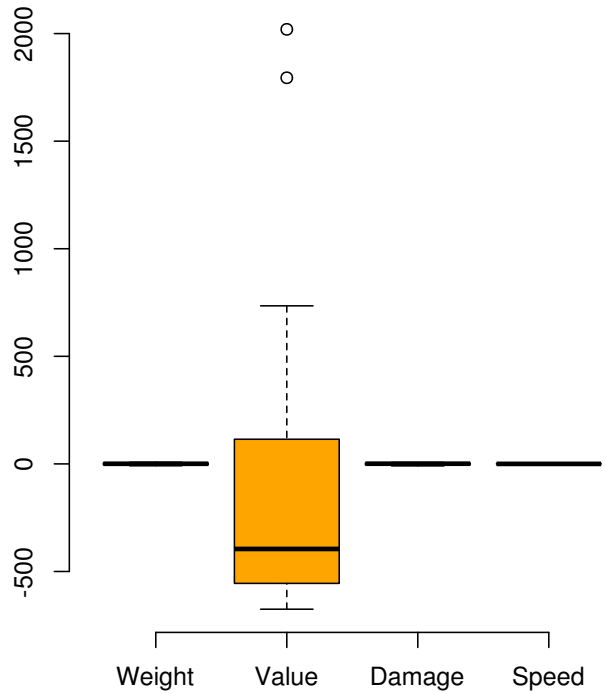
```r
Xbar <- scale(X, scale = F)
Sigma <- (t(Xbar) %*% Xbar)/nrow(X)
```

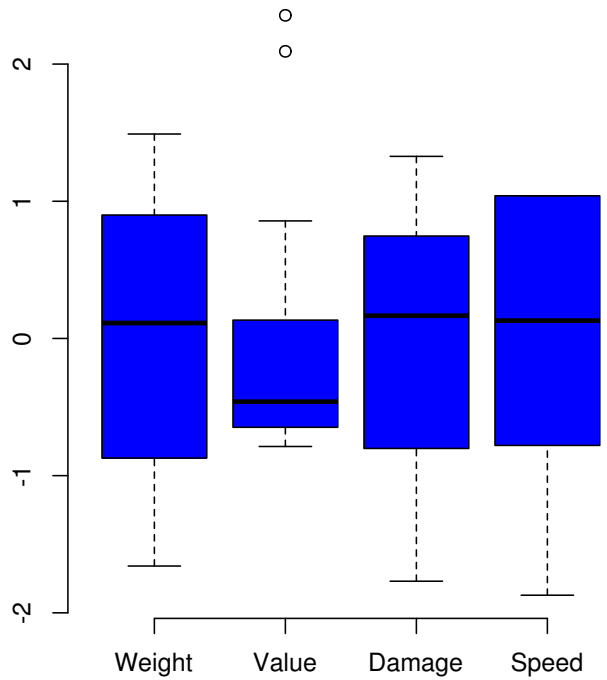
```
##           Weight     Value     Damage     Speed
## Weight   25.8163   2588.06   25.15306  -0.13776
## Value  2588.0612 735380.23 2050.30612 -98.28444
## Damage   25.1531   2050.31   26.69388  -0.03253
## Speed    -0.1378    -98.28   -0.03253   0.02950
```

```r
M <- diag(1/diag(Sigma))
```

```
##          Weight    Value  Damage Speed
## Weight 0.03874 0.00e+00 0.00000   0.0
## Value  0.00000 1.36e-06 0.00000   0.0
## Damage 0.00000 0.00e+00 0.03746   0.0
## Speed  0.00000 0.00e+00 0.00000  33.9
```

```
boxplot(Xbar, col = "orange")
boxplot(Xbar %*% sqrt(M), col = "blue")
```





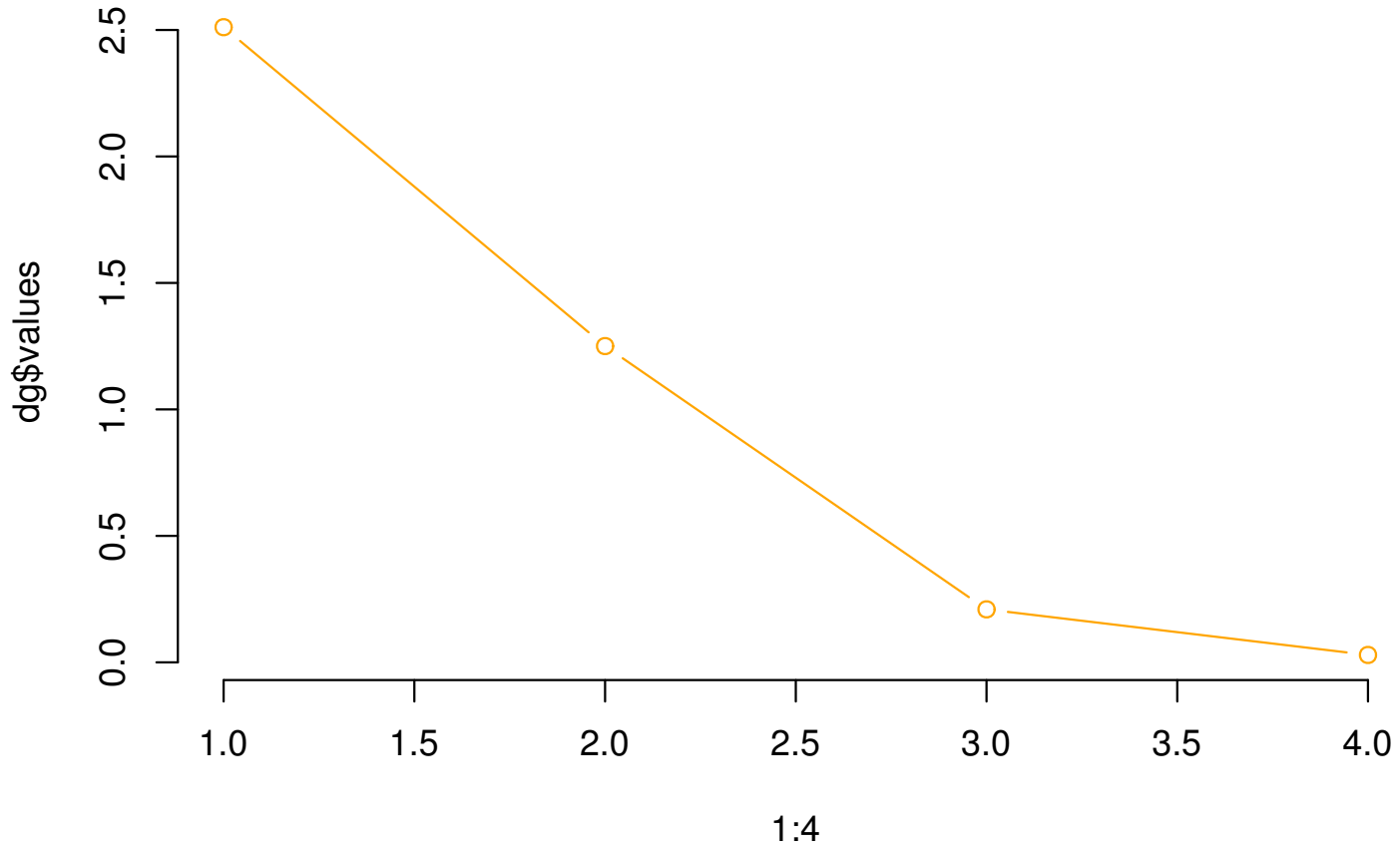
```
Mhalf <- diag(1/sqrt(diag(Sigma)))
dg <- eigen(Mhalf %*% Sigma %*% Mhalf)
dg$vectors <- diag(sqrt(diag(Sigma))) %*% dg$vectors
dg$values
```

```
## [1] 2.51105 1.25027 0.20949 0.02919
```

```
cumsum(dg$values)/sum(dg$values)
```

```
## [1] 0.6278 0.9403 0.9927 1.0000
```

```
plot(1:4, dg$values, type = "b", col = "orange")
```



Grades data

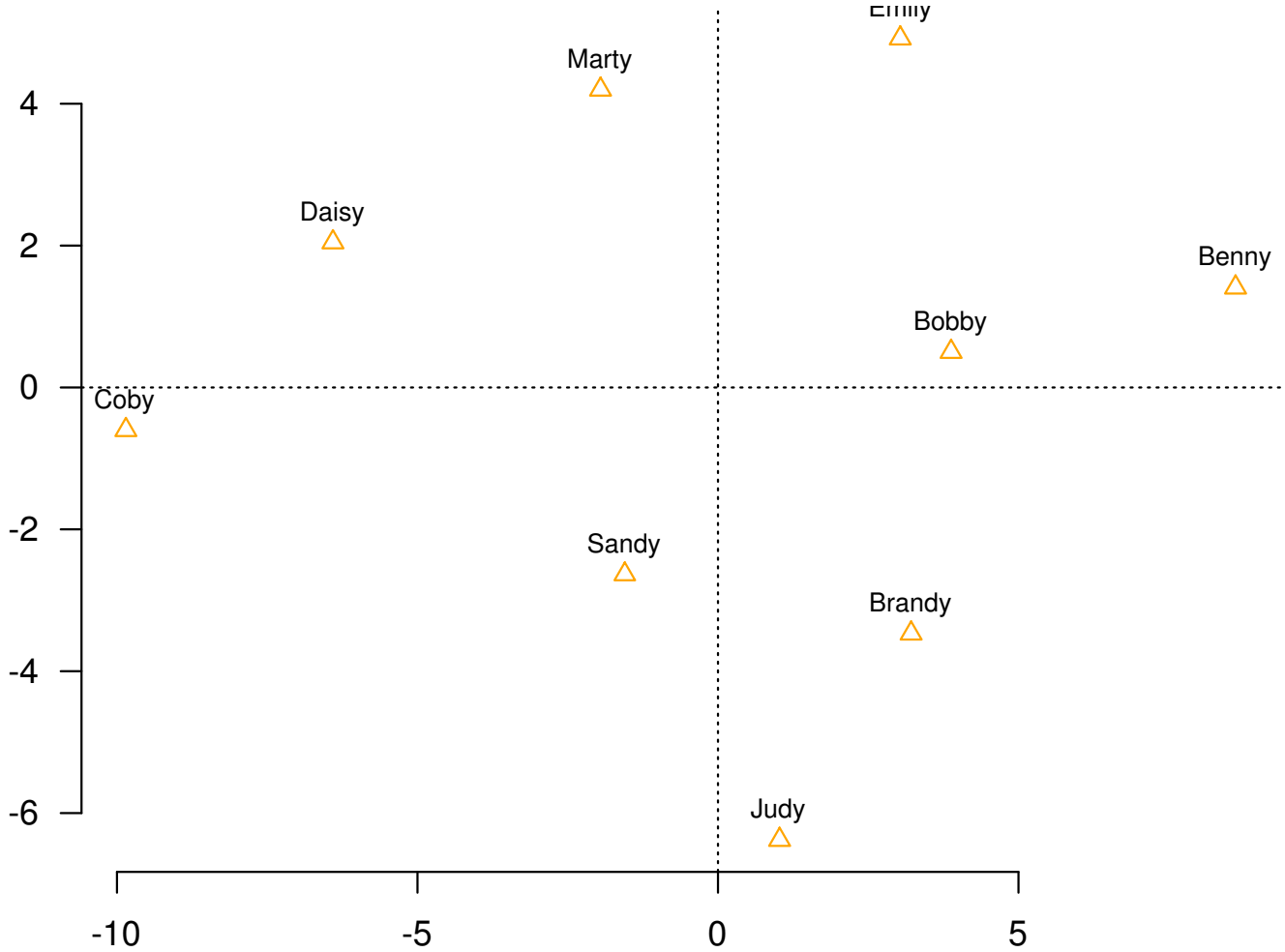
```
C <- Xbar %*% dg$vectors
cos2 <- rowSums(C[, 1:2]^2)/rowSums(C^2)

##  Benny  Bobby Brandy   Coby  Daisy  Emily   Judy  Marty  Sandy
## 0.9999 0.9997 0.9986 0.9998 0.9991 0.9993 0.9993 0.9980 0.9808

rbind(C[, 1]^2/(nrow(X) * dg$values[1]), C[, 2]^2/(nrow(X) * dg$values[2]))

##         Benny   Bobby  Brandy     Coby   Daisy  Emily     Judy   Marty
## [1,] 0.29187 0.05921 0.04063 0.381947 0.16152 0.0362 0.004138 0.01502
## [2,] 0.01835 0.00233 0.11110 0.003319 0.03868 0.2237 0.375596 0.16289
##         Sandy
## [1,] 0.00946
## [2,] 0.06407
```

```
plot(C[, 1:2], pch = 2, cex = cos2, col = "orange", xlab = "", ylab = "")
```

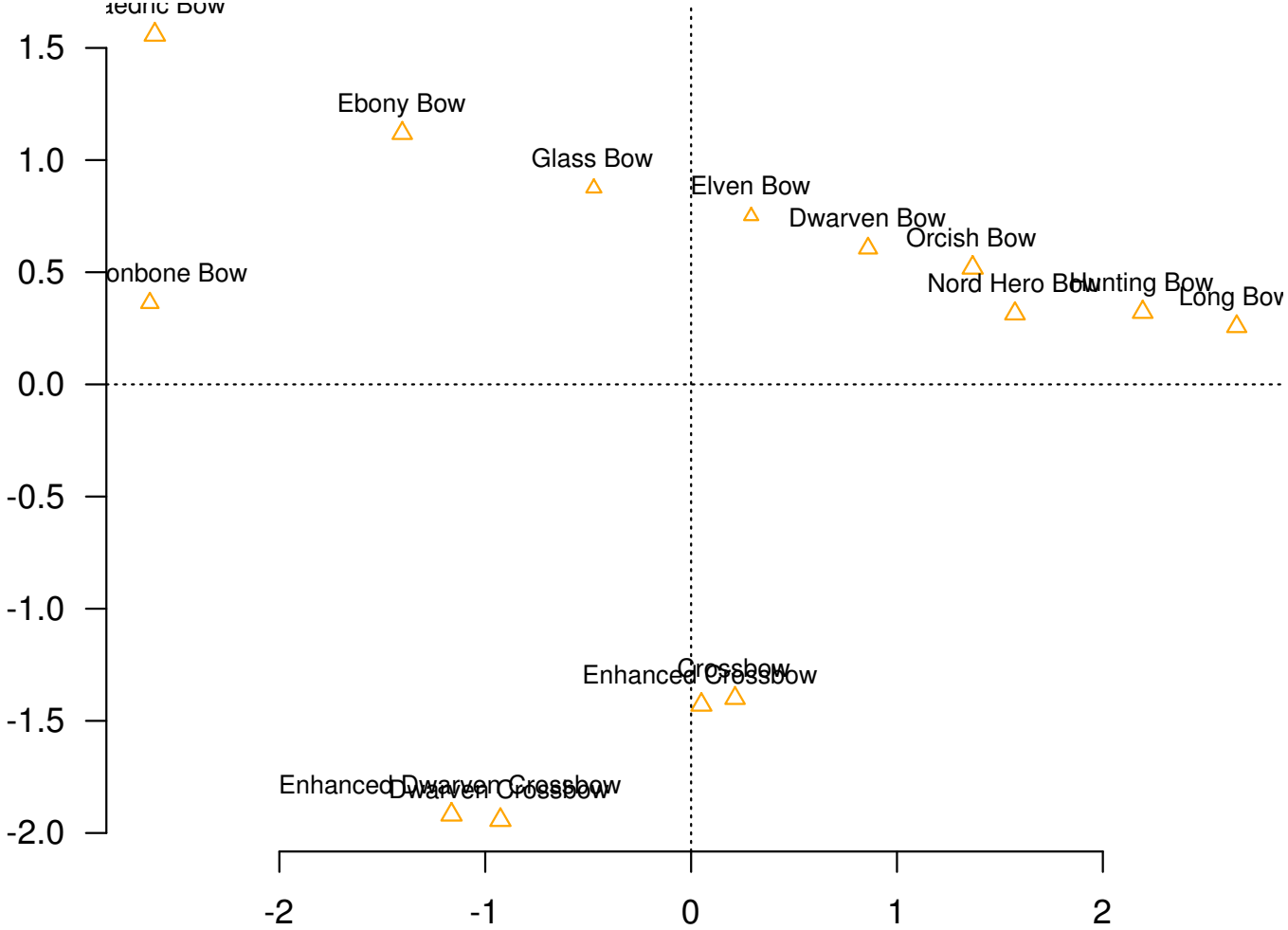


Skyrim bows data

```
C <- Xbar %*% M %*% dg$vectors
cos2 <- rowSums(C[, 1:2]^2)/rowSums(C^2)
```

```
##                  Long Bow                Hunting Bow
##                    0.9349                     0.9590
##                 Orcish Bow              Nord Hero Bow
##                    0.9853                     0.9575
##               Dwarven Bow                 Elven Bow
##                    0.8813                     0.6941
##                 Glass Bow                 Ebony Bow
##                    0.7384                     0.9497
##               Daedric Bow             Dragonbone Bow
##                    0.9963                     0.8511
##                 Crossbow          Enhanced Crossbow
##                    0.9467                     0.9817
##          Dwarven Crossbow Enhanced Dwarven Crossbow
##                    0.9880                     0.9879
```

```
plot(C[, 1:2], pch = 2, cex = cos2, col = "orange", xlab = "", ylab = "")
```



Grades data

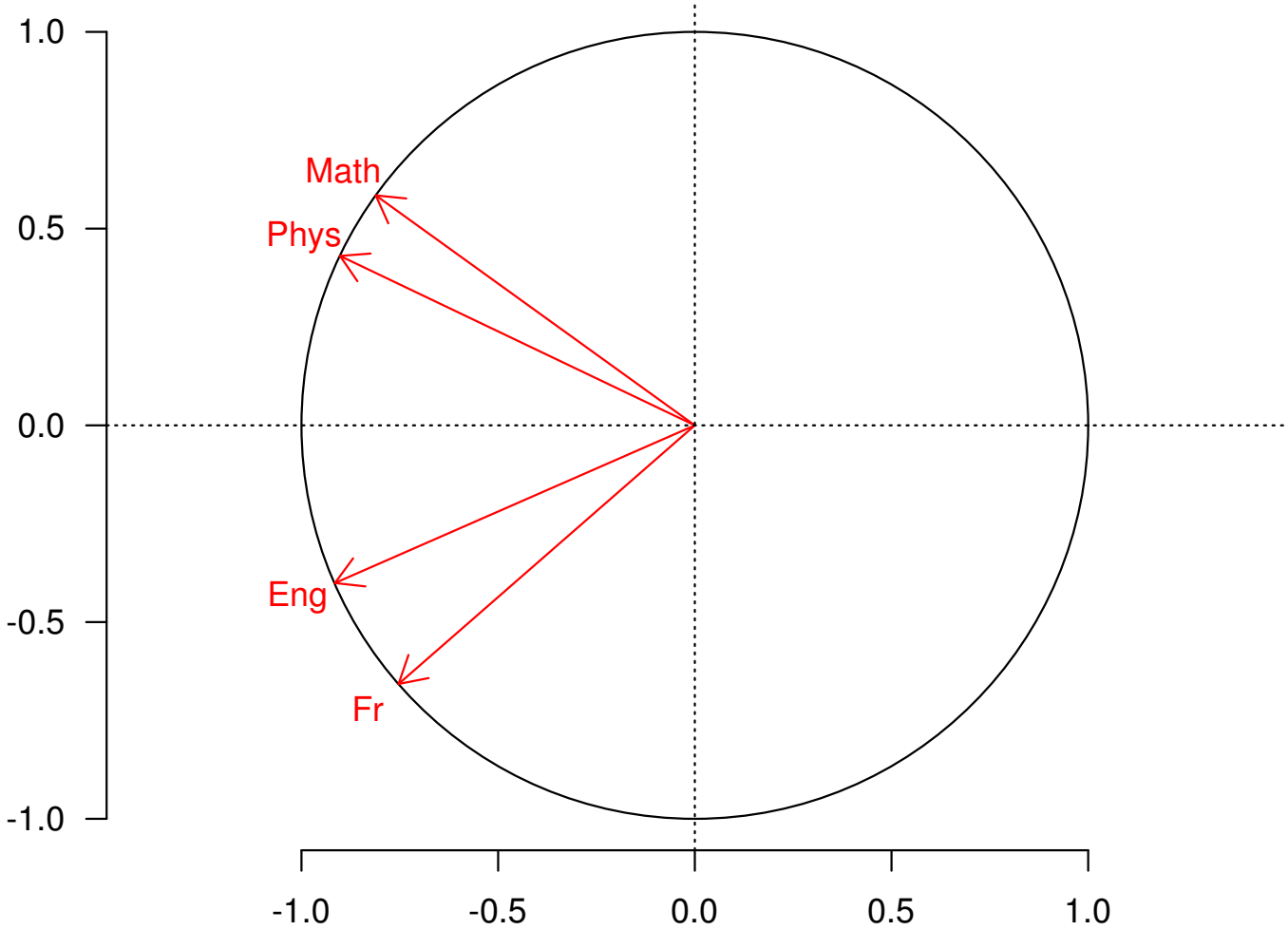
```
Rho <- diag(1/sqrt(diag(Sigma))) %*% dg$vectors %*% diag(sqrt(dg$values))
Rho[, 1:2]
```

```
##          [,1]    [,2]
## Math -0.8112  0.5845
## Phys -0.9019  0.4306
## Fr   -0.7532 -0.6573
## Eng  -0.9149 -0.4007
```

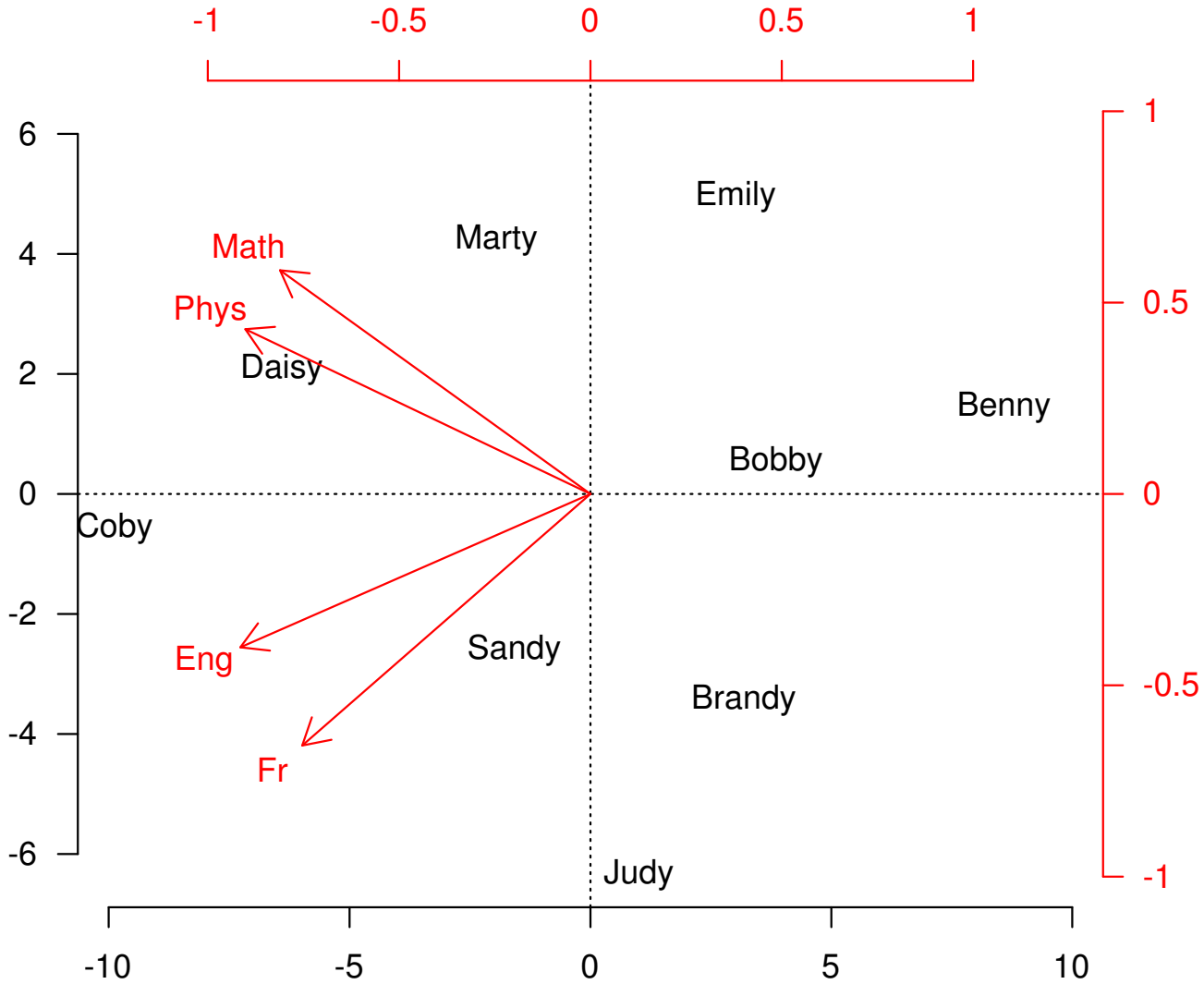
```
rowSums(Rho[, 1:2]^2)
```

```
##   Math   Phys     Fr    Eng
## 0.9996 0.9988 0.9993 0.9976
```

```
DrawCorCircle(Rho)
```



```
DrawBiplot(C, Rho)
```



Skyrim bows data

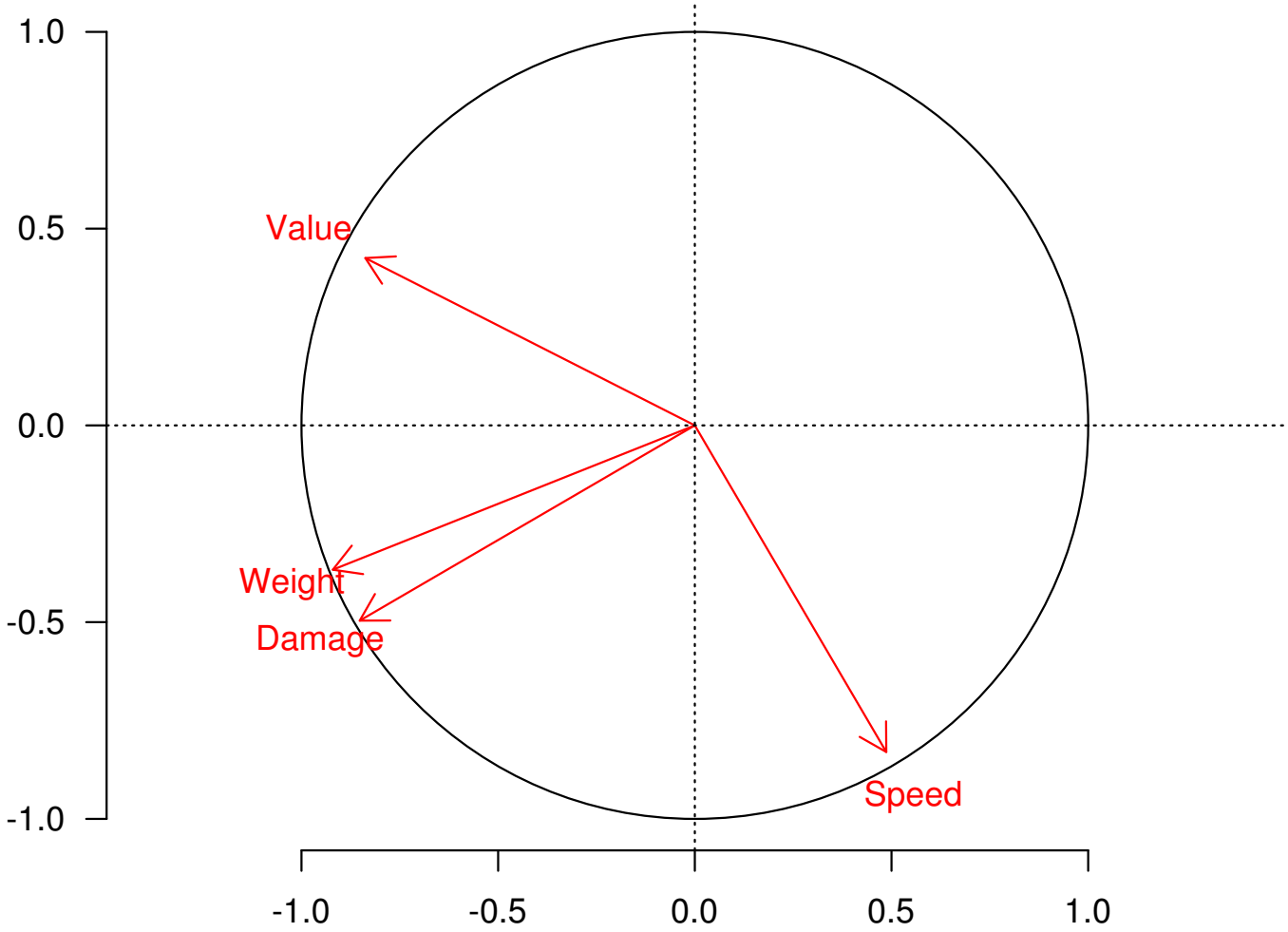
```
Rho <- diag(1/sqrt(diag(Sigma))) %*% dg$vectors %*% diag(sqrt(dg$values))
rownames(Rho) <- colnames(X)
Rho[, 1:2]
```

```
##             [,1]    [,2]
## Weight -0.9203 -0.3667
## Value   -0.8376  0.4252
## Damage -0.8519 -0.4961
## Speed    0.4867 -0.8299
```

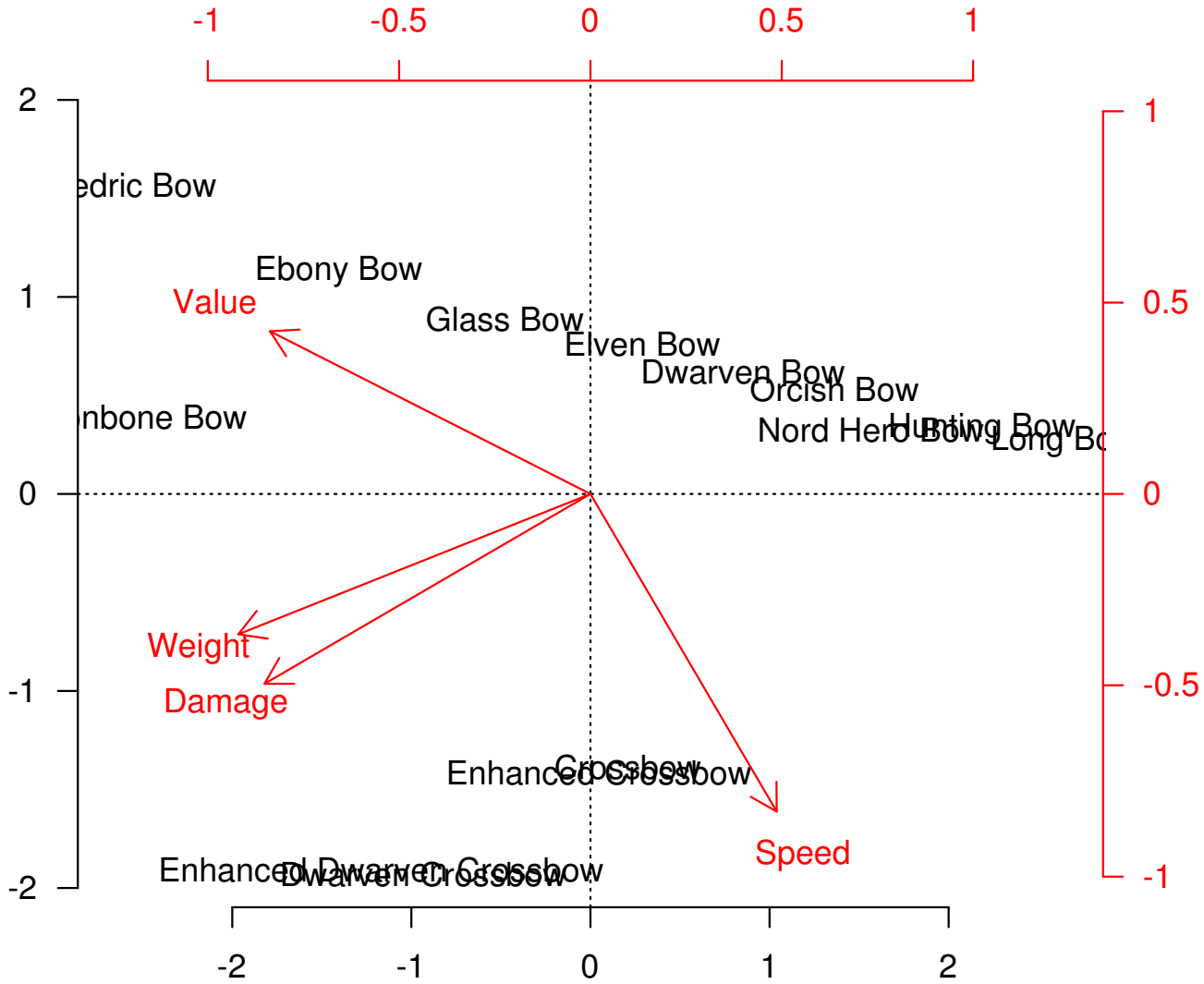
```
rowSums(Rho[, 1:2]^2)
```

```
## Weight  Value Damage  Speed
## 0.9814 0.8824 0.9718 0.9257
```

```
DrawCorCircle(Rho)
```



```
DrawBiplot(C, Rho)
```



Bourdieu data

	DR	SCE	LET	SC	MD	PH	PD	IUT	Total
EAG	80	36	134	99	65	28	11	58	511
SAG	6	2	15	6	4	1	1	4	39
PT	168	74	312	137	208	53	21	62	1035
PLCS	470	191	806	400	876	164	45	79	3031
CM	236	99	493	264	281	56	36	87	1552
EMP	145	52	281	133	135	30	20	54	850
OUV	16	6	27	11	8	2	2	8	80
OTH	305	115	624	247	301	47	42	90	1771
Total	1426	575	2692	1297	1878	381	178	442	8869

```
D1 <- diag(1/rowSums(T))
P1 <- D1 %*% T
D2 <- diag(1/colSums(T))
P2 <- D2 %*% t(T)
```

PCA of line profiles:

```
M1half <- diag(sqrt(n/colSums(T)))
M1halfInv <- diag(sqrt(colSums(T)/n))
dg <- eigen(M1half %*% t(P1) %*% t(P2) %*% M1halfInv)
dg$values <- dg$values[2:8]  # Avoid the trivial eigenvlue
dg$vectors <- (M1halfInv %*% dg$vectors)[, 2:8]  # Idem
C1 <- P1 %*% (n * D2) %*% dg$vectors
```

Transition formula:

```
C2 <- P2 %*% C1 %*% diag(1/sqrt(dg$values))
```

```r
# Representation quality of the x's values
rowSums(C1[, 1:2]^2)/rowSums(C1^2)
```

```
##    EAG    SAG     PT   PLCS     CM    EMP    OUV    OTH
## 0.9988 0.9030 0.3171 0.9995 0.7107 0.9779 0.8648 0.9915
```

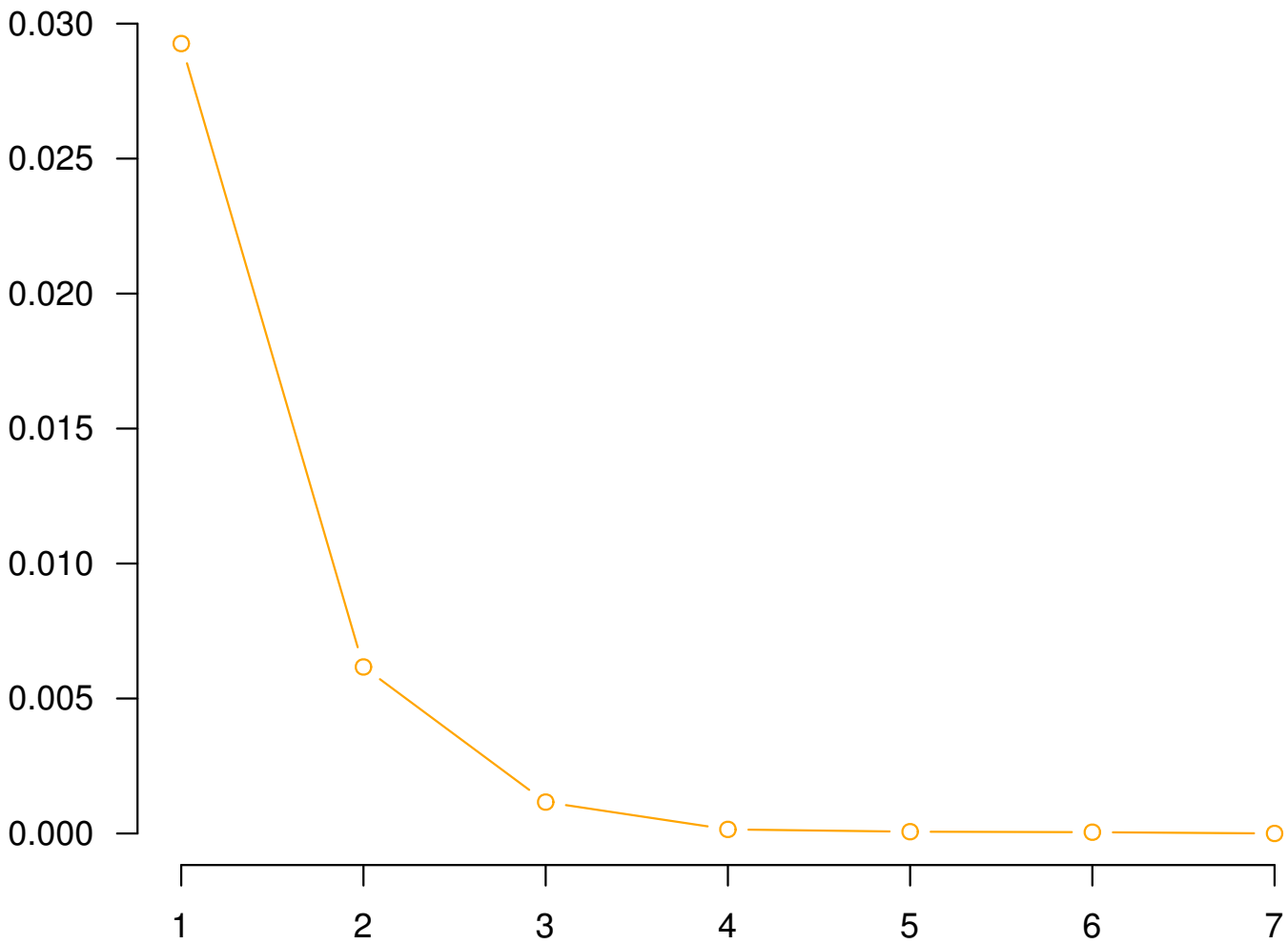
```r
# Representation quality of the y's values
rowSums(C2[, 1:2]^2)/rowSums(C2^2)
```

```
##     DR    SCE    LET     SC     MD     PH     PD    IUT
## 0.5036 0.2708 0.9927 0.6678 0.9991 0.9605 0.9764 0.9890
```

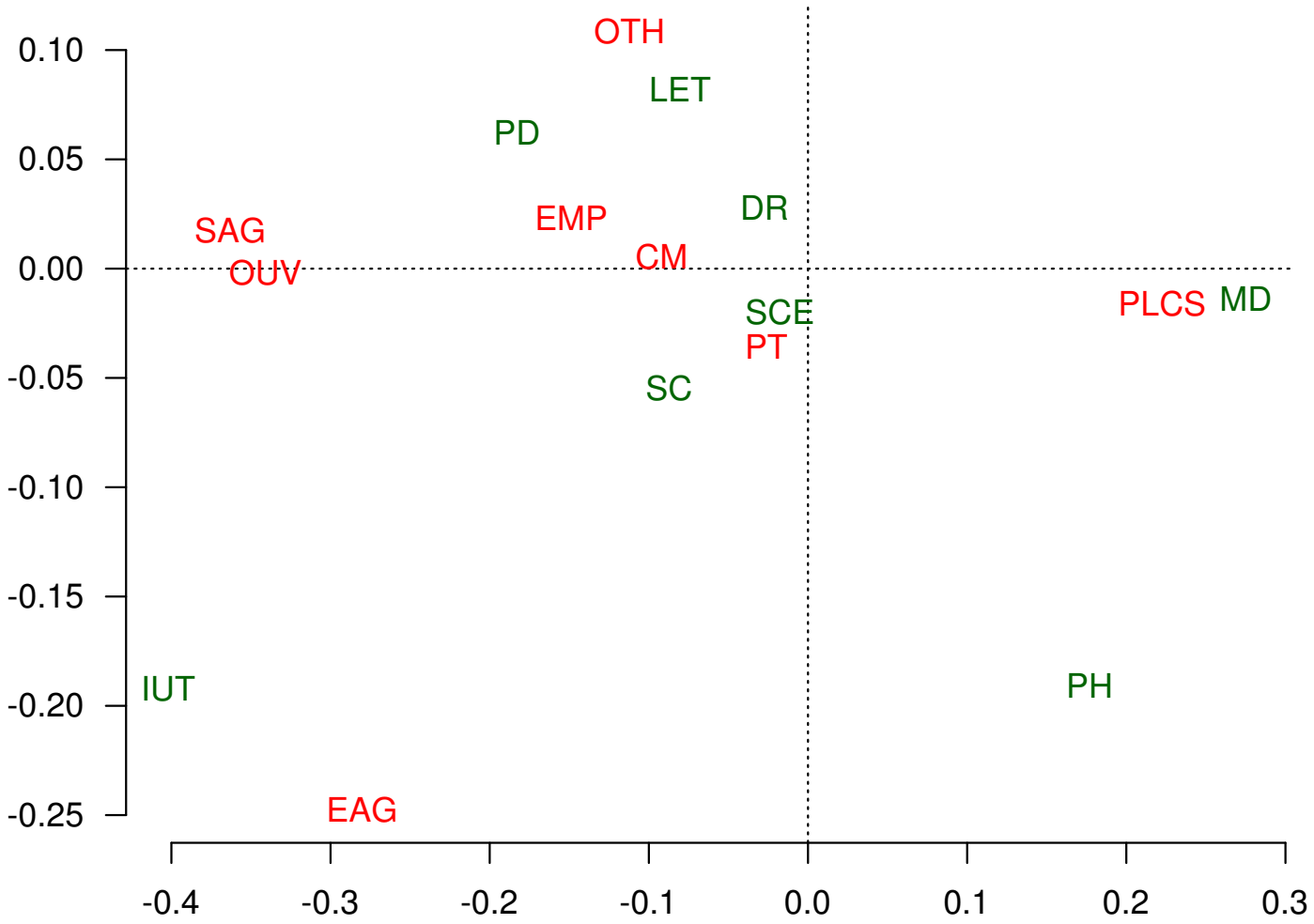
```r
cumsum(dg$values)/sum(dg$values)
```

```
## [1] 0.7939 0.9613 0.9928 0.9969 0.9987 1.0000 1.0000
```

```
plot(dg$values, type = "b", col = "orange", xlab = "", ylab = "")
```



```
DrawCA(C1, C2)
```


OTH
LET
PD
EMP
DR
SAG
OUV
CM
SCE
PT
PLCS
MD
SC
IUT
PH
EAG
0.10
0.05
0.00
-0.05
-0.10
-0.15
-0.20
-0.25
-0.4
-0.3
-0.2
-0.1
0.0
0.1
0.2
0.3

Bibliography

- *The Elements of Statistical Learning : Data Mining, Inference and Prediction*, J. Friedman, T. Hastie and R. Tibshirani (2009)
- *Principal Component Analysis*, I.T. Jolliffe (2002)
- *Wiki Stat*, `http://wikistat.fr/`