

Motivation



- Changing gears to dimensionality reduction, and in particular an unsupervised method for doing so.
- ► We will eventually bring this back around to supervised learning in the next lecture
- ► Note: dimensionality reduction is NOT (necessarily) the same as variable selection/feature reduction/etc. This will hopefully become clear while we progress...

Motivation



- ► Where we'll go with this on the application side...
- ► The heptathlon is a track and field competition with several (seven, specifically) running, throwing, and jumping events.
- ► The scoring system is...complex (we will outline it later). Can we use this particular form of dimensionality reduction to devise a 'simpler' scoring system?
- But first, let's get technical...

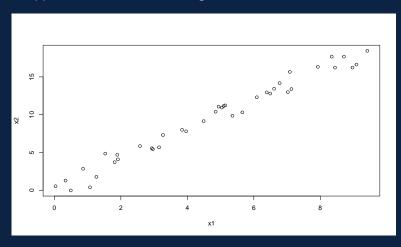
Basic Idea



- ▶ We have p predictors $X_1, X_2, ..., X_p$
- ▶ We will seek p 'new' variables, say Z_1, Z_2, \ldots, Z_p that
 - 1. are linear combinations of X_1, X_2, \ldots, X_p
 - 2. are uncorrelated (that is, $Cor(Z_j, Z_k) = 0$ for all $j \neq k$)
 - 3. provide the bulk of the variation (aka, information) in X_1, X_2, \ldots, X_p within the first few Z_j 's



► Suppose we have the following data





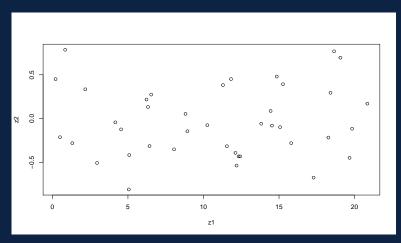
▶ We can note



- Now, suppose we create two new variables as linear combos of X_1 and X_2 , namely...
 - $ightharpoonup Z_1 = .45X_1 + .90X_2$
 - $ightharpoonup Z_2 = .90X_1 .45X_2$
- ▶ Note that with our current toolbox, this would seem to be a fairly random choice of coefficients for the linear combos...but let's see what the transformed data looks like...



ightharpoonup Scatterplot of Z_1 and Z_2





And further note

```
> cov(cbind(z1, z2))
           z1
                      z^2
z1 37.3862401 0.1354968
    0.1354968 0.1576611
z_2
```

```
> cor(cbind(z1, z2))
            7.1
                       7.2
z1 1.00000000 0.05580986
z2 0.05580986 1.00000000
```

Jeffrev L. Andrews Lecture 5 **DATA 572**



A square $p \times p$ matrix A is said to have an eigenvalue λ with corresponding eigenvector $\gamma \neq \vec{0}$ if

$$A\gamma = \lambda \gamma$$

▶ If A is symmetric, then A has p eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_p$ and p corresponding eigenvectors $\gamma_1, \gamma_2, \ldots, \gamma_p$



$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 2 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 2 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 2 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 2 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 2 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 3 & 3 & 3$$



▶ If A is $p \times p$ symmetric with eigenvalues, then we can write

$$A = P\Lambda P^T$$

where all matrices are $p \times p$.

► Further, note that

$$P = [\gamma_1 \quad \gamma_2 \quad \dots \quad \gamma_p]$$

▶ and Λ is a diagonal matrix with the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_p$ along the diag.



- ▶ Also, $PP^T = P^TP = I_p$, AKA the columns of P are orthonormal
- $lackbr{\blacktriangleright}$ AKA, $\gamma_j^{ au} {\gamma}_k = 0$ for all j
 eq k and $\gamma_j^{ au} {\gamma}_j = 1$



$$\begin{bmatrix} .38 & -.92 \\ .42 & .38 \end{bmatrix} \begin{bmatrix} 2.21 & 0.74 \\ 0 & 0.74 \end{bmatrix} \begin{bmatrix} .38 & .42 \\ -.92 & .38 \end{bmatrix} = \begin{bmatrix} .5 & .57 \\ 2 \end{bmatrix}$$



ightharpoonup A symmetric $p \times p$ matrix A is positive semi-definite (psd) if

$$\vec{c}^T A \vec{c} \ge 0 \quad \forall \quad \vec{c}$$

- ▶ If A is psd, then $\lambda_i \geq 0$ for all i.
- ► Note that covariance matrices are psd, and therefore have *p* non-negative eigenvalues.



- ► Recall from beginning of these slides...
- ▶ We will seek p 'new' variables, say Z_1, Z_2, \ldots, Z_p that
 - 1. are linear combinations of X_1, X_2, \dots, X_p
 - 2. are uncorrelated (that is, $Cor(Z_j, Z_k) = 0$ for all $j \neq k$)
 - 3. provide the bulk of the variation (aka, information) in X_1, X_2, \ldots, X_p within the first few Z_j 's



- ▶ Suppose covariance matrix Σ has eigenvalues ordered such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$ with corresponding eigenvectors $\gamma_1, \gamma_2, \dots, \gamma_p$.
- It can be shown that γ_1 (aka, the eigenvector corresponding to the largest eigenvalue of Σ) provides coefficients such that $Var(\gamma_1^TX)$ is maximized subject to the constraint $\gamma_1^T\gamma_1=1$
- And furthermore, γ_2 maximizes $\text{Var}(\gamma_2^T\mathsf{X})$ subject to $\gamma_2^T\gamma_2=1$ AND $\gamma_2^T\gamma_1=0$
- ► Annnnd so on for the remaining eigenvectors...

- ▶ In summary, the eigendecomposition of Σ provides the solution for our desired properties for principal components. AKA, we can define $Z_j = \gamma_j X$.
- ► So the eigenvectors provide the coefficients for the linear combo, but the eigenvalues are interesting too!
- Note that the diagonal of Σ contains the variance of each variable. Summing that up, $\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_p^2$, provides a measure of 'total variance'
- It can be shown through matrix properties (namely trace) that $\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_p^2 = \lambda_1 + \lambda_2 + \cdots + \lambda_p$.
- ► So the total variance of *X* still exists in *PX*. AKA, there is no information loss in principal components (at least, to this point of our discussion...)

Jeffrey L. Andrews Lecture 5 DATA 572 PCA 18 / 43

Principal Components and Geometry



- ► A couple of geometric asides
- Not only is there no information loss, it is also true that distance between observations in the original data are preserved in the PCA-transformed space.
- ► Angles between vectors are also preserved.
- ▶ In fact, PCA is simply an orthogonal rotation about the origin.



- ▶ Furthermore, we can easily figure out the "proportion of variance explained" by any one component via $\frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j} = \frac{\text{Var}Z_i}{\text{"total variance"}}$
- ▶ Brings us to an interesting point...
- Suppose once we get to the k^{th} principal component, we see the percent of variance explained as quite small, say 0.001.
- ► Can we then toss out that principal component? Along with the remaining principal components (which by definition will have smaller λ)?



- ► NOTE: THIS is where the dimensionality reduction occurs in PCA
- ► Since we transform *p* variables (X) into *p* variables (Z), it is only when we toss out principal components that we reduce the dimensionality of the data.
- ▶ It is also the only point at which we experience a loss of information from the original data.
- ▶ But also note: even if we only keep one principal component (transforming from p-variate to univariate data) we don't actually remove any of our original measurements. All p original variables are needed to calculate $Z_1 = \gamma_{11}X_1 + \gamma_{12}X_2 + \cdots + \gamma_{1p}X_p$.

PCA and Scaling



- ▶ BIG BIG NOTE: PCA is NOT scale invariant
- ➤ As we'll see in an example, this has huge implications...notably, any variable with large variance (relative to the rest) will dominate the first principal component.
- ▶ In most cases, this is undesirable. Most commonly, you will need/want to scale your data to have mean 0, variance 1 (almost certainly when your measures are on vastly different scales).
- ► This amounts to performing an eigendecomposition on the correlation matrix rather than the covariance matrix.

PCA: How many components?



- ► So how do we choose how many principal components to keep?
- ▶ There are several common options, we'll discuss three:
 - 1. Cumulative proportion/percent of variance
 - Keep number of components such that, say, 90% (or 95%, or 80%, etc) of the variance from original data is retained
 - 2. Kaiser criterion
 - ▶ Keep all $\lambda_j \geq \bar{\lambda}$ where $\bar{\lambda} = \frac{\sum_{j=1}^{p} \lambda_j}{p}$. Note this is further simplified if the data is scaled (mean 0, variance 1) since $\bar{\lambda} = \frac{p}{p} = 1$.
 - 3. Scree plot
 - Plot the (monotonically decreasing) eigenvalues, look for an 'elbow', or plateauing

Phew...



► And finally, an example...

PCA on Heptathlon Data



```
> matrix(rownames(heptathlon)[1:6], ncol=1)
    \lceil .1 \rceil
[1,] "Joyner-Kersee (USA)"
[2,] "John (GDR)"
[3.] "Behmer (GDR)"
[4,] "Sablovskaite (URS)"
[5,] "Choubenkova (URS)"
[6,] "Schulz (GDR)"
> print(heptathlon[1:6,], row.names=FALSE)
hurdles highjump shot run200m longjump javelin run800m score
  12.69
        1.86 15.80
                       22.56
                                 7.27 45.66 128.51
                                                     7291
  12.85 1.80 16.23 23.65 6.71 42.56 126.12 6897
  13.20 1.83 14.20
                       23.10
                                 6.68 44.54 124.20 6858
  13.61 1.80 15.23
                       23.92 6.25 42.78 132.24 6540
  13.51 1.74 14.76
                       23.93 6.32 47.46 127.90 6540
  13.75
           1.83 13.50
                       24.65
                                 6.33
                                      42.82 125.79 6411
```

Heptathlon Scoring¹



- ► Some notes on heptathlon scoring it's not simple .
- ► The heptathlon scoring system was devised by Dr. Karl Ulbrich, a Viennese mathematician.
- ► There is designated "standard" performance (for example, approximately 1.82 m for the high jump) scores 1000 points.
- ► Each event also has a minimum recordable performance level (e.g. 0.75 m for the high jump), corresponding to zero points.
- ► Then...

Jeffrey L. Andrews Lecture 5 DATA 572 PCA 26 / 43

¹https://en.wikipedia.org/wiki/Heptathlon

Heptathlon Scoring¹



Event	а	b	С	
200 metres	4.99087	42.5	1.81	
800 metres	0.11193	254	1.88	
100 metres hurdles	9.23076	26.7	1.835	
High jump	1.84523	75.0	1.348	
Long jump	0.188807	210	1.41	
Shot put	56.0211	1.50	1.05	
Javelin throw	15.9803	3.80	1.04	

► Running events (200m, 800m, 100m hurdles)

$$P = a(b - T)^c$$

► Jumping events (high, long)

$$P = a(M - b)^c$$

► Throwing events (shotput, javelin)

$$P = a(D - b)^c$$

Jeffrey L. Andrews Lecture 5 DATA 572 PCA 27 / 43

¹https://en.wikipedia.org/wiki/Heptathlon

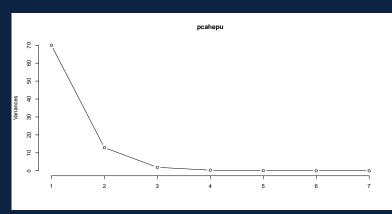
Scoring in General



- ► As a general concept, fairly combining scores from several sporting disciplines seems tricky
- ▶ But in effect, we want to find a scoring system that best separates the participants
- ► In more statistical lingo, we want to find a single variable (made of the original measures) which will provide the bulk of the variation present in the data
- ► In other words, PCA can suggest a different (simpler?) scoring system! We remove the score variable and work with the remaining...



- > pcahepu <- prcomp(heptathlon[,-8])</pre>
- > plot(pcahepu, type="lines")





► "rotation" are the eigenvectors, aka coefficients of the linear combo, aka component "loadings"

```
> pcahepu$rotation[,1:3]
                               PC2
                 PC1
                                           PC3
hurdles
         0.069508692 -0.0094891417
                                    0.22180829
highjump -0.005569781 0.0005647147 -0.01451405
shot
        -0.077906090 0.1359282330 -0.88374045
run 200m
         0.072967545 -0.1012004268 0.31005700
longjump -0.040369299 0.0148845034 -0.18494319
javelin
        0.006685584 0.9852954510 0.16021268
run800m
         0.990994208 0.0127652701 -0.11655815
```

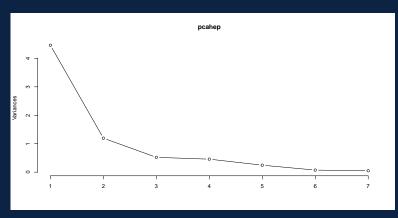
► What do you notice?



► What do you notice?



- > pcahep <- prcomp(heptathlon[,-8], scale.=TRUE)</pre>
- > plot(pcahep, type="lines")





► What do you notice?



```
> summary(pcahep)
Importance of comp
```

Importance of components:

```
PC1 PC2 PC3 PC4 PC5 PC6 PC6 Standard deviation 2.1119 1.0928 0.72181 0.67614 0.49524 0.27010 0.221 Proportion of Variance 0.6372 0.1706 0.07443 0.06531 0.03504 0.01042 0.007 Cumulative Proportion 0.6372 0.8078 0.88223 0.94754 0.98258 0.99300 1.000
```

► Scree plot suggests probably 2, most criterion would probably look at 2. 3. or 4



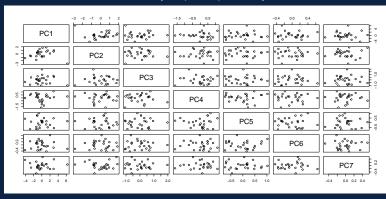
Also contained in the pca object as "x" are what's commonly referred to as 'scores'. AKA, the transformed observations!

> head(pcahep\$x)

	PC1	PC2	PC3	PC4	PC
Joyner-Kersee (USA)	-4.121448	-1.24240435	-0.3699131	-0.02300174	0.426006
John (GDR)	-2.882186	-0.52372600	-0.8974147	0.47545176	-0.703065
Behmer (GDR)	-2.649634	-0.67876243	0.4591767	0.67962860	0.105525
Sablovskaite (URS)	-1.343351	-0.69228324	-0.5952704	0.14067052	-0.453928
Choubenkova (URS)	-1.359026	-1.75316563	0.1507013	0.83595001	-0.687194
Schulz (GDR)	-1.043847	0.07940725	0.6745305	0.20557253	-0.737933



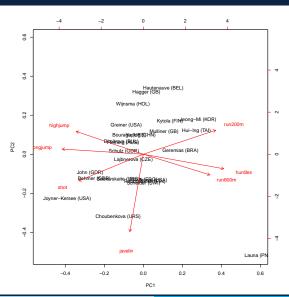
► Which we could visualize by a pairs plot, say



Or perhaps better in a bivariate form...



> biplot(pcahep)



Back to the task...

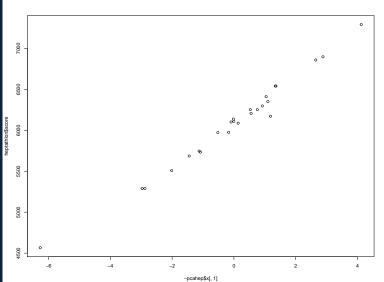


- How does the first variable function as a scoring system?
- Because of the sign, minimizing would be the goal. For PCA, the sign of the entire component is arbitrary. Opposite signs within a component are meaningful, however.
- ► Thus, we can multiply an entire component by -1 without changing the underlying mathematics.



```
> round(-pcahep$rotation[,1], 2)
hurdles highjump
                      shot run200m longjump javelin
                                                        run800m
   -0.45
             0.38
                      0.36
                               -0.41
                                         0.46
                                                  0.08
                                                           -0.37
> print(cbind(-sort(pcahep$x[,1]), rownames(heptathlon), heptathlon$score)
                                           [.2]
                                                                  [.3]
                     [,1]
Joyner-Kersee (USA)
                    "4.12144762636023"
                                           "Joyner-Kersee (USA)"
                                                                  "7291"
John (GDR)
                    "2.88218593484013"
                                           "John (GDR)"
                                                                  "6897"
Behmer (GDR)
                    "2.64963376599126"
                                           "Behmer (GDR)"
                                                                  "6858"
Choubenkova (URS)
                    "1.35902569554282"
                                           "Sablovskaite (URS)"
                                                                  "6540"
Sablovskaite (URS)
                    "1.34335120967757"
                                           "Choubenkova (URS)"
                                                                  "6540"
Dimitrova (BUL)
                                           "Schulz (GDR)"
                                                                  "6411"
                     "1.18645383210095"
Fleming (AUS)
                    "1.10038563857154"
                                           "Fleming (AUS)"
                                                                  "6351"
Schulz (GDR)
                    "1.04384747092169"
                                           "Greiner (USA)"
                                                                  "6297"
Greiner (USA)
                    "0.92317363886205"
                                           "Lajbnerova (CZE)"
                                                                  "6252"
Bouraga (URS)
                     "0.759819023916292"
                                            "Bouraga (URS)"
                                                                  "6252"
Wijnsma (HOL)
                                           "Wijnsma (HOL)"
                     "0.556268302151919"
                                                                  "6205"
Lajbnerova (CZE)
                     "0.530250688783237"
                                           "Dimitrova (BUL)"
                                                                  "6171"
Yuping (CHN)
                     "0.13722543980327"
                                           "Scheider (SWI)"
                                                                  "6137"
Braun (FRG)
                     "-0.0037742225569839"
                                           "Braun (FRG)"
                                                                  "6109"
```





Lecture 5 Jeffrey L. Andrews



Scheider (SWI)	"-0.015461226409337"	"Ruotsalainen (FIN)"	"6101"
Ruotsalainen (FIN)	"-0.0907477089383147"	"Yuping (CHN)"	"6087"
Hagger (GB)	"-0.171128651449238"	"Hagger (GB)"	"5975"
Brown (USA)	"-0.51925264574111"	"Brown (USA)"	"5972"
Hautenauve (BEL)	"-1.08569764619083"	"Mulliner (GB)"	"5746"
Mulliner (GB)	"-1.12548183277136"	"Hautenauve (BEL)"	"5734"
Kytola (FIN)	"-1.44705549915266"	"Kytola (FIN)"	"5686"
Geremias (BRA)	"-2.01402962042439"	"Geremias (BRA)"	"5508"
Hui-Ing (TAI)	"-2.88029863527855"	"Hui-Ing (TAI)"	"5290"
Jeong-Mi (KOR)	"-2.97011860698208"	"Jeong-Mi (KOR)"	"5289"
Launa (PNG)	"-6.27002197162809"	"Launa (PNG)"	"4566"



> head(hentathlon)

> nedd (nepudunion)							
	hurdles	highjump	shot	run200m	longjump	javelin	run800
Joyner-Kersee (USA)	12.69	1.86	15.80	22.56	7.27	45.66	128.5
John (GDR)	12.85	1.80	16.23	23.65	6.71	42.56	126.1
Behmer (GDR)	13.20	1.83	14.20	23.10	6.68	44.54	124.2
Sablovskaite (URS)	13.61	1.80	15.23	23.92	6.25	42.78	132.2
Choubenkova (URS)	13.51	1.74	14.76	23.93	6.32	47.46	127.9
Schulz (GDR)	13.75	1.83	13.50	24.65	6.33	42.82	125.7
> tail(heptathlon)							

	hurdles	highjump	shot	run200m	Longjump	javelin	run800m	S
Hautenauve (BEL)	14.04	1.77	11.81	25.61	5.99	35.68	133.90	
Kytola (FIN)	14.31	1.77	11.66	25.69	5.75	39.48	133.35	
Geremias (BRA)	14.23	1.71	12.95	25.50	5.50	39.64	144.02	
Hui-Ing (TAI)	14.85	1.68	10.00	25.23	5.47	39.14	137.30	
Jeong-Mi (KOR)	14.53	1.71	10.83	26.61	5.50	39.26	139.17	
Launa (PNG)	16.42	1.50	11.78	26.16	4.88	46.38	163.43	

Jeffrey L. Andrews Lecture 5 **DATA 572**

