SI 618
Exploratory Data Analysis

Principal components analysis

Exploratory factor analysis

Instructor: Dr. Chris Teplovs (cteplovs@umich.edu)

Lead Developer, Digital Innovation Greenhouse, Office of Academic Innovation

Adjunct Lecturer of Information, School of Information

GSI: SungJin Nam (sjnam@umich.edu)

Coming up

- Final project report due Friday, Dec 16, 1:00pm (late days can apply)
- Homework 5 (Factor analysis) is an optional bonus assignment, due Friday, Dec. 16, 1:00 pm
  - Up to +10% on course grade

Reminder:
This course has been shortened!

- Last day of class is DECEMBER 9, 2016
  - Per information from the SI Registrar
- Due date for project is still DECEMBER 16, 2016
- SLIDES ASSIGNMENT HAS BEEN ELIMINATED
- December 9 class will be brief intro to machine learning, review of 618, and teaching evaluations

SI 618 Data Exploration: Class Schedule

Date	Topic	Assignments Due
Week 1	Course introduction Basics of Programming with R	
Week 2	Basic analysis and visualization using ggplot2: qplot() Manipulating data frames using plyr	Homework 1
Week 3	Smoothing and Trend-finding. Building ggplot Layer by Layer	Homework 2
Week 4	Cluster analysis	Homework 3
Week 5	(Thanksgiving: no class!)	
Week 6	Factor Analysis Methods (PCA, EFA)	Homework 4
Week 7	Machine Learning, Review, Evaluations	

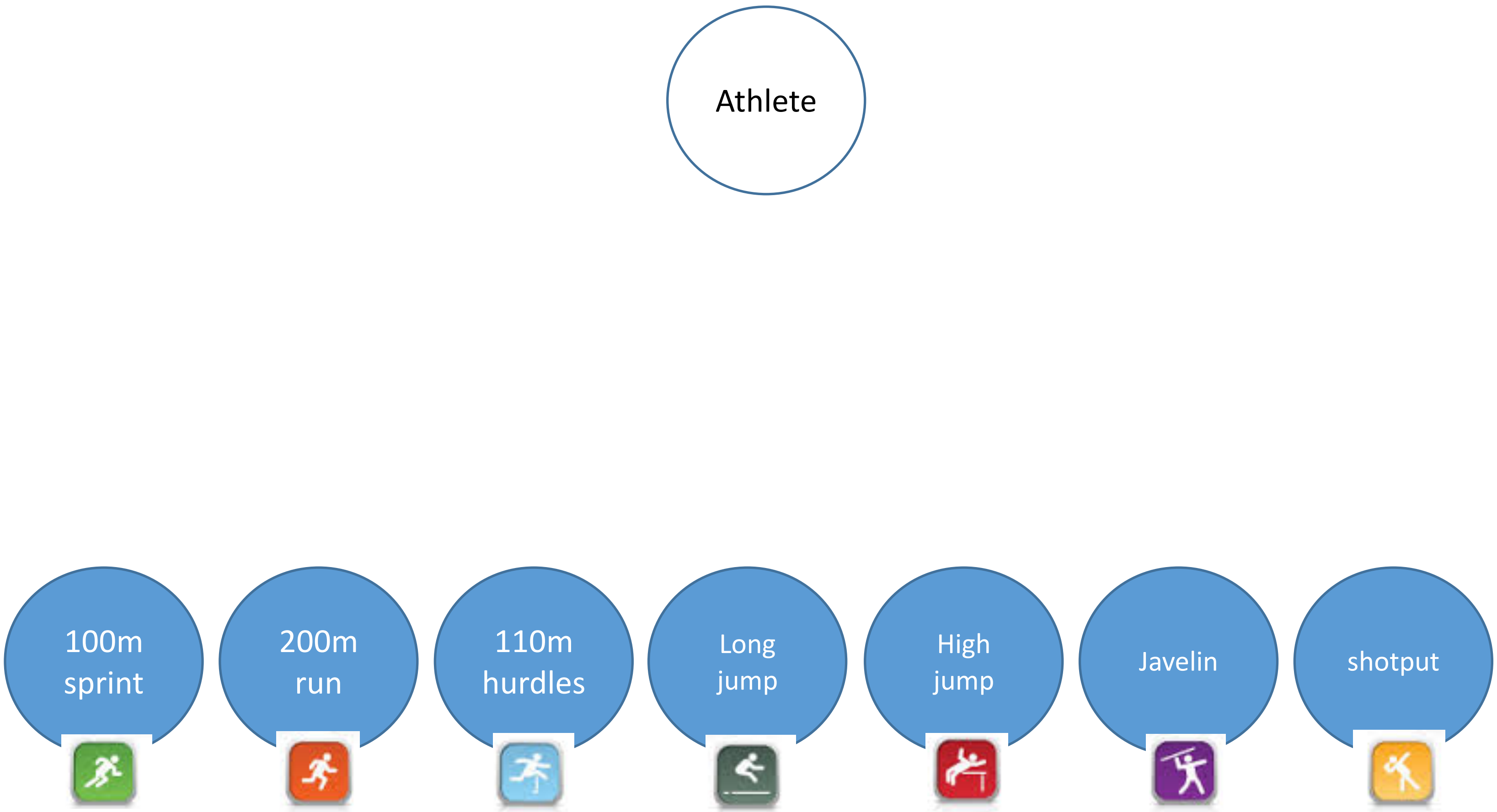
Class Schedule for Today

- Discussion about clustering
- Factor analysis
  - Overview
  - Principal components analysis (PCA)
  - Exploratory factor analysis (EFA)

Clustering…

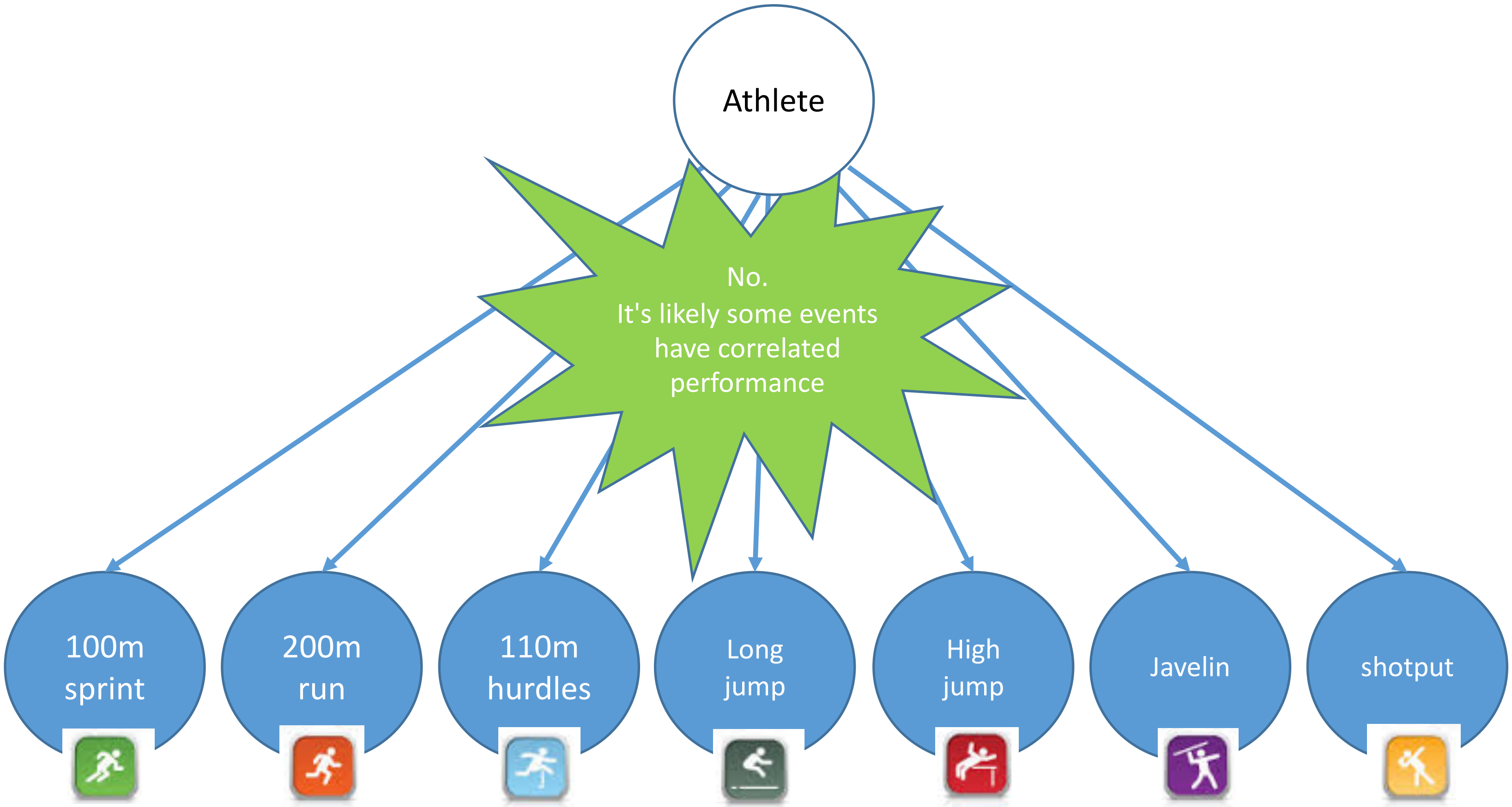
- What makes a good cluster?
- How do you know?
- Why is this important?

Consider an athlete competing in a set of events (e.g. heptathlon)



We observe their score in each event = variable

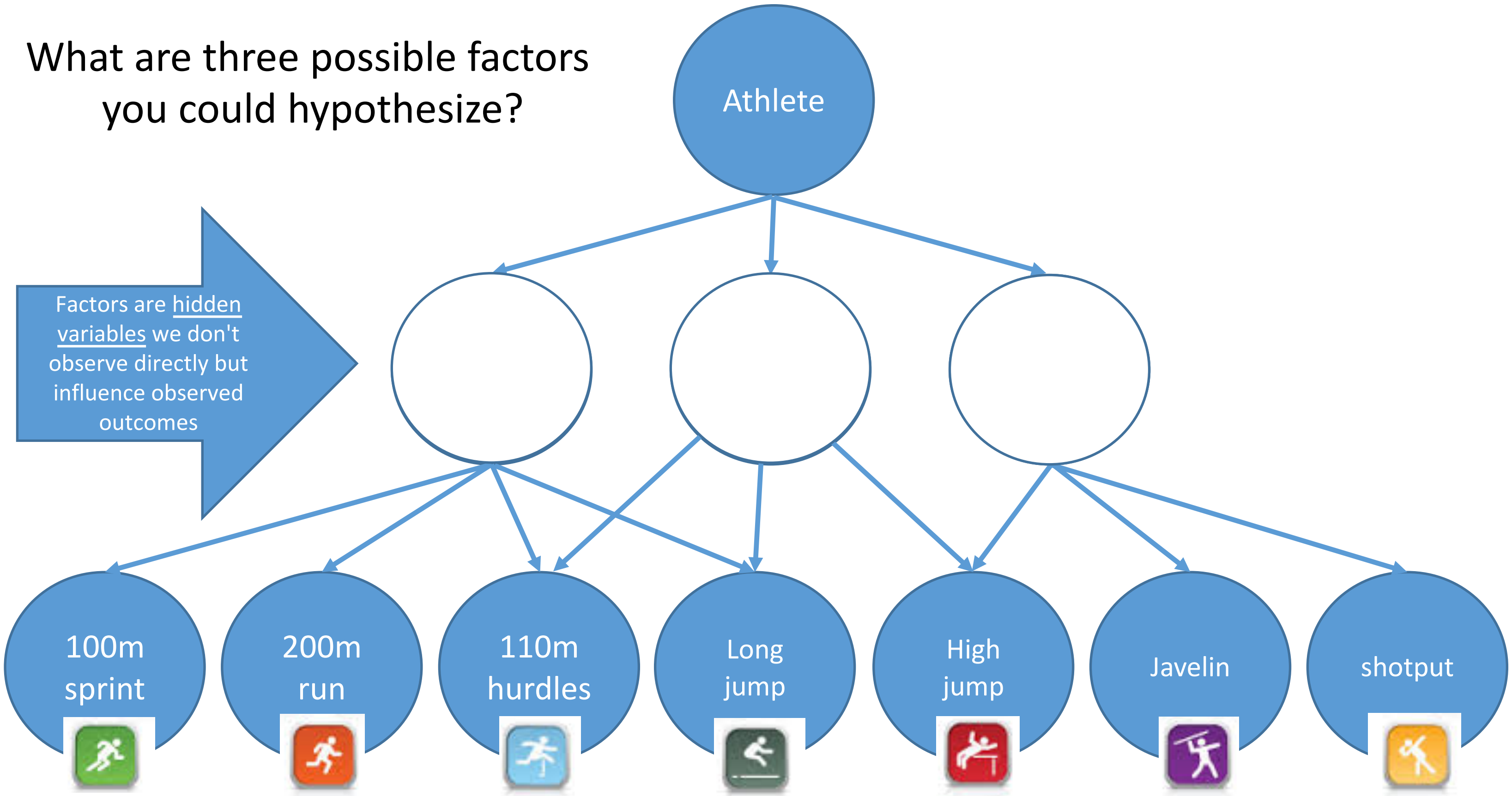
Given a particular athlete, is their score in one event completely unrelated to their score in other events?



Each event score = variable

We might be able to describe an athlete's abilities in terms of a smaller number of factors that strongly influence their scores in all events.

What are three possible factors you could hypothesize?



Each event score = variable

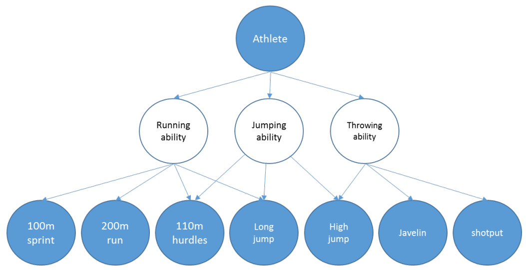
What is factor analysis?

- A set of useful and important tools for exploring data with multiple variables.
- The goal of factor analysis methods is to "explain" many observed variables in terms of a much smaller number of unobserved variables (factors).
- This is a form of <u>dimensionality reduction</u>
  - Compress/reduce huge # of variables to essential subset
  - Create interpretable models of observed phenomena in terms of relatively independent factors
  - Create a summary representation of an object
    - E.g. topic vector for a document
  - Address sparsity problems by finding groupings of similar data
    - E.g. find groups of users
  - Find representative samples from a much larger set

Factors and factor loadings

- Factor variables:

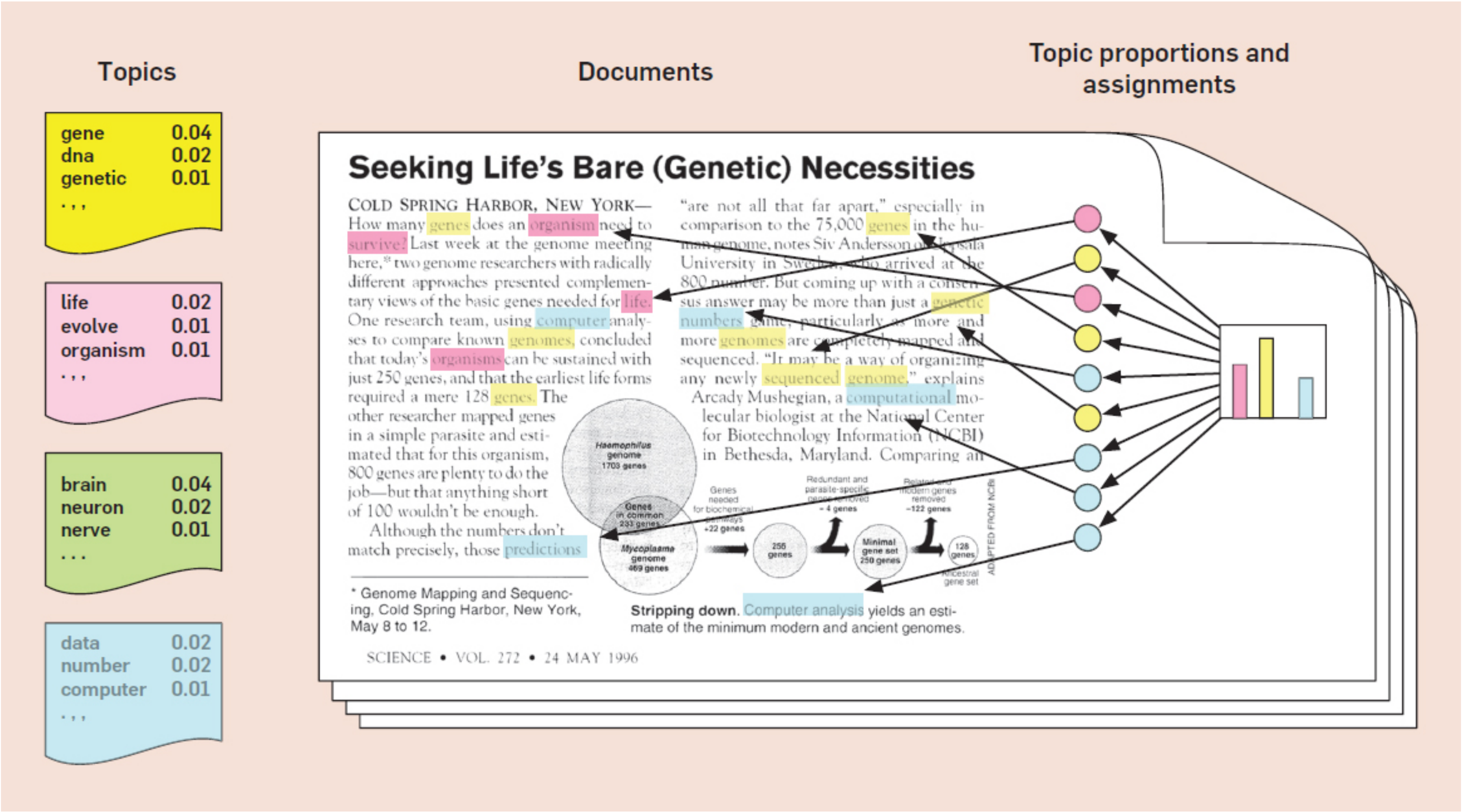
  F_{RUN} : Running ability (-1 to 1)
  F_{JUMP} : Jumping ability (-1 to 1)
  F_{THROW} : Throwing ability (-1 to 1)



- Goal: Rewrite our data in terms of a linear combination of the factor variables
- The factor <u>loadings</u> are the <u>weights</u> that relate the *p* observed variables to the *k* factors in a linear model

  $\text{Score}(A_E) = w_{RUN}F_{RUN}(A_E) + w_{JUMP}F_{JUMP}(A_E) + w_{THROW}F_{THROW}(A_E)$

Probabilistic topic modeling is a form of factor analysis: each topic is a factor.



Source: D. Blei, Probabilistic topic models. Commun. ACM 55, 4 (April 2012) 77-84.

Characterizing Microblogs with Topic Models [Ramage, Dumais, Liebling ICWSM 2010]

Examples of topic models (textual factors) extracted from Yelp and Amazon reviews

NOTE!
Factor interpretations come from human inspection (authors)

Be[illegible]				
pale ales	lambics	[illegible]	[illegible]	[illegible]
ipa	funk	chocolate	pump	eat
pine	brett	coffee	nutm	yellow
grapefruit	saison	black	co	straw
citrus	vinegar	dark	cinnamon	pilsner
ipas	raspberry	roasted	pie	summer
piney	lambic	stout	cheap	pale
citrusy	barnyard	bourbon	bud	lager
floral	funky	tan	water	banana
hoppy	tart	porter	macro	coriander
dipa	raspberries	vanilla	adjunct	pils

Musical instruments (Amazon)				
drums	strings	wind	microphones	software
cartridge	guitar	reeds	mic	software
sticks	violin	harmonica	microphone	interface
strings	strap	cream	stand	midi
snare	neck	reed	mics	windows
stylus	capo	harp	wireless	drivers
cymbals	tune	fog	microphones	inputs
mute	guitars	mouthpiece	condenser	usb
heads	picks	bruce	battery	computer
these	bridge	harmonicas	filter	mp3
daddario	tuner	harps	stands	program

Video games (Amazon)				
fantasy	nintendo	windows	ea/sports	accessories
fantasy	mario	sims	drm	cable
rpg	ds	flight	ea	controller
battle	nintendo	windows	spore	cables
tomb	psp	xp	creature	ps3
raider	wii	install	nba	batteries
final	gamecube	expansion	football	sonic
battles	memory	program	nhl	headset
starcraft	wrestling	software	basketball	wireless
characters	metroid	mac	madden	controllers
ff	smackdown	sim	hockey	component

Clothing (Amazon)				
bags	winter	formal	pants	bras
backpack	vest	scarf	pants	bra
bag	jacket	cards	jeans	bras
jansport	fleece	shirt	pair	support
costume	warm	shirts	dickies	cup
books	columbia	suit	these	cups
hat	coat	silk	levis	underwire
laptop	sweatshirt	wallet	waist	supportive
bags	russell	belt	pairs	breasts
backpacks	gloves	leather	socks	sports
halloween	sweater	tie	they	breast

Yelp Phoenix									
theaters	spas	mexican	vietnamese	snacks	italian	medical	donuts	coffee	seafood
theater	massage	mexican	pho	cupcakes	pizza	dr	donuts	coffee	sushi
movie	spa	salsa	vietnamese	cupcake	crust	stadium	donut	starbucks	dish
harkins	yoga	tacos	yogurt	hotel	pizzas	dentist	museum	books	restaurant
theaters	classes	chicken	brisket	resort	italian	doctor	target	latte	rolls
theatre	pedicure	burrito	beer	rooms	bianco	insurance	subs	bowling	server
movies	trail	beans	peaks	dog	pizzeria	doctors	sub	lux	shrimp
dance	studio	taco	mojo	dogs	wings	dental	dunkin	library	dishes
popcorn	gym	burger	shoes	frosting	pasta	appointment	frys	espresso	menu
tickets	hike	carne	froyo	bagel	mozzarella	exam	tour	stores	waiter
flight	nails	food	zoo	bagels	pepperoni	prescription	bike	gelato	crab

Table 4: Top ten words from each of $K = 5$ topics from five of our datasets (and with $K = 10$ from Yelp). Each column is labeled with an 'interpretation' of that topic. Note we display all the topics (and not only the 'interpretable' ones). All topics are clean and easily interpretable.

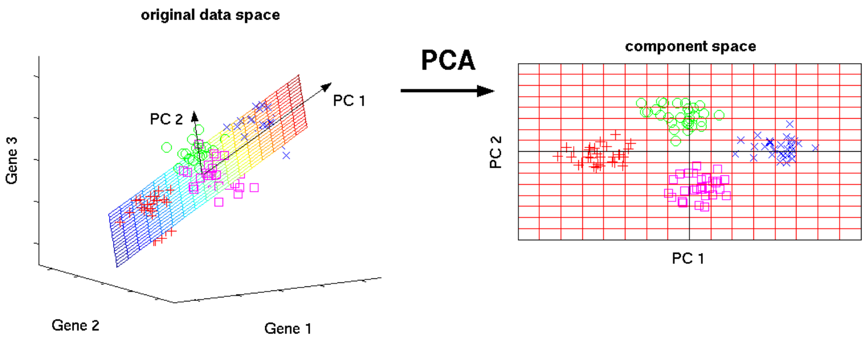
Source: http://i.stanford.edu/~julian/pdfs/recsys13.pdf

What is factor analysis?

- There are two main tools for factor analysis:
  - Principal Components Analysis (PCA)
    - PCA projects data to lower dimensions. PCA extracts all variance.
  - Exploratory Factor Analysis (EFA)
    - EFA seeks to find a small number of unobserved underlying variables that might explain the common variance (not all variance) in the data.

Intuitive view of Principal Components Analysis (PCA)

- Imagine data set as k-dimensional cloud
- We can project the cloud onto a 2-d surface
- PCA finds the most "informative" projection in terms of characterizing variation in the data
- Another view:
  - Fit k-dimensional ellipsoid to the cloud
    - Each axis represents a principal component
    - Axis is "small" = variance along that axis is small
    - Omitting that axis only loses "small" amount of information

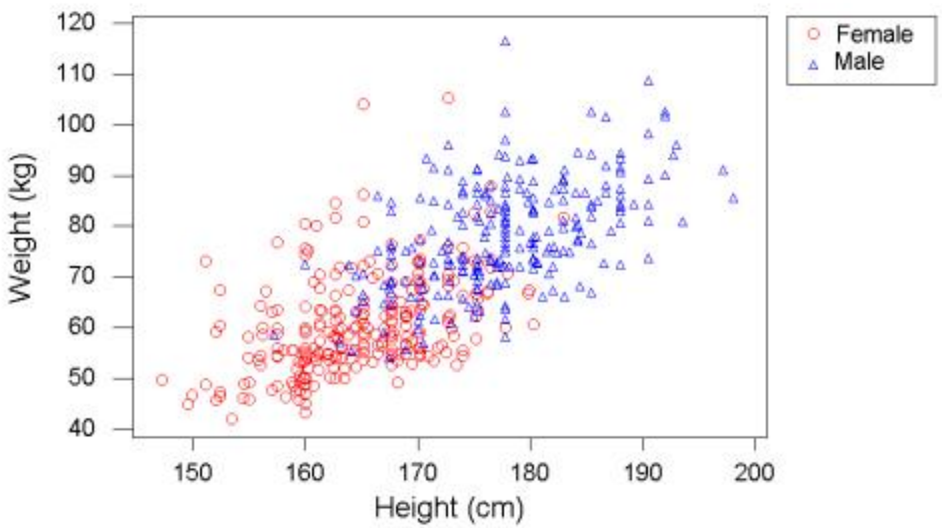


Principal Components Analysis (PCA): Why is it used?

- Identify combinations of variables that explain most of the variation in the data
- Compress high-dimensional datasets to a few dimensions
- Filter noise from data (as a result of the approximation)

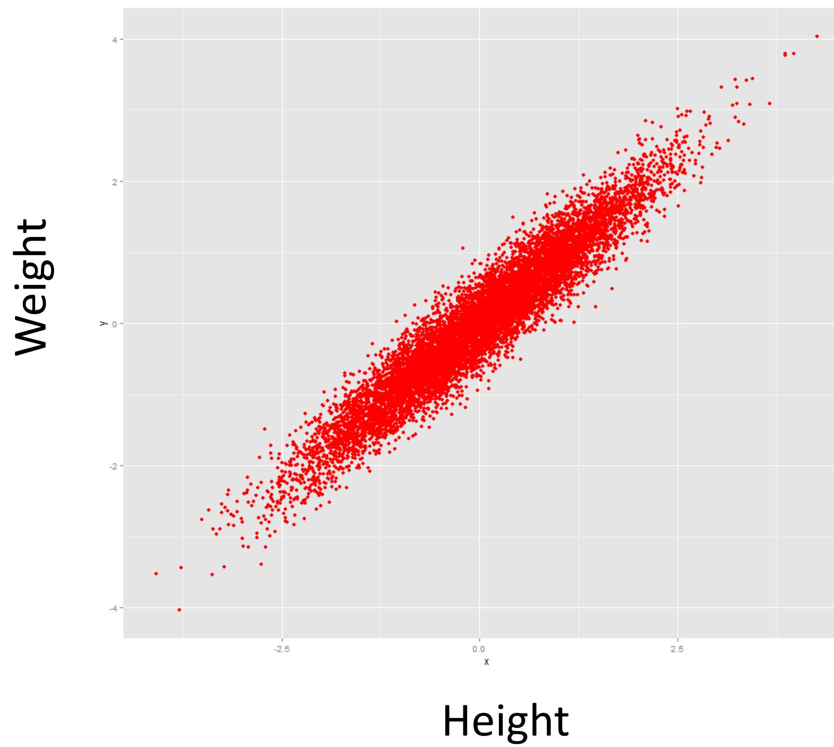
Human weight and height have a 2-d distribution that's not quite Gaussian but for example purposes we'll assume they are.

Source: http://www.amstat.org/publications/jse/v11n2/datasets.heinz.html

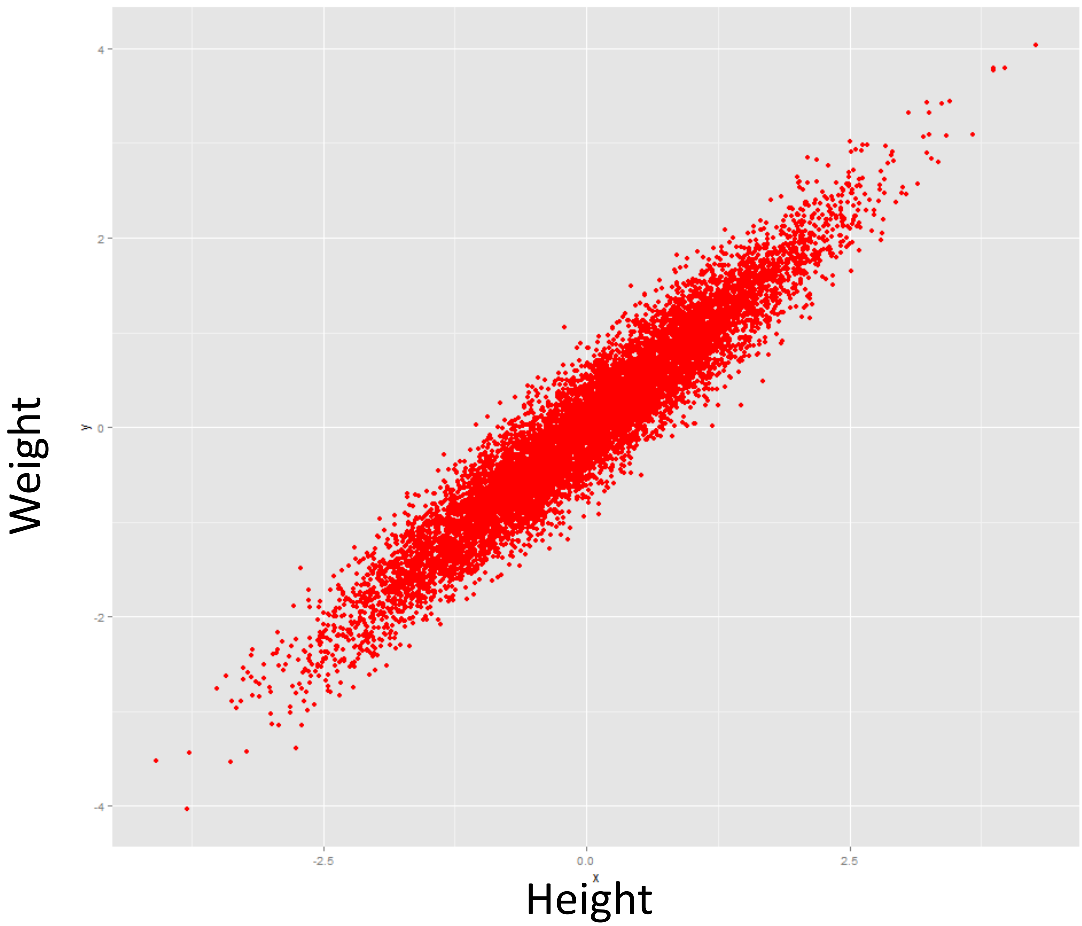


Towards Principal Component Analysis (PCA):
Here's a hypothetical 2-dimensional distribution of two variables e.g. human heights vs weights

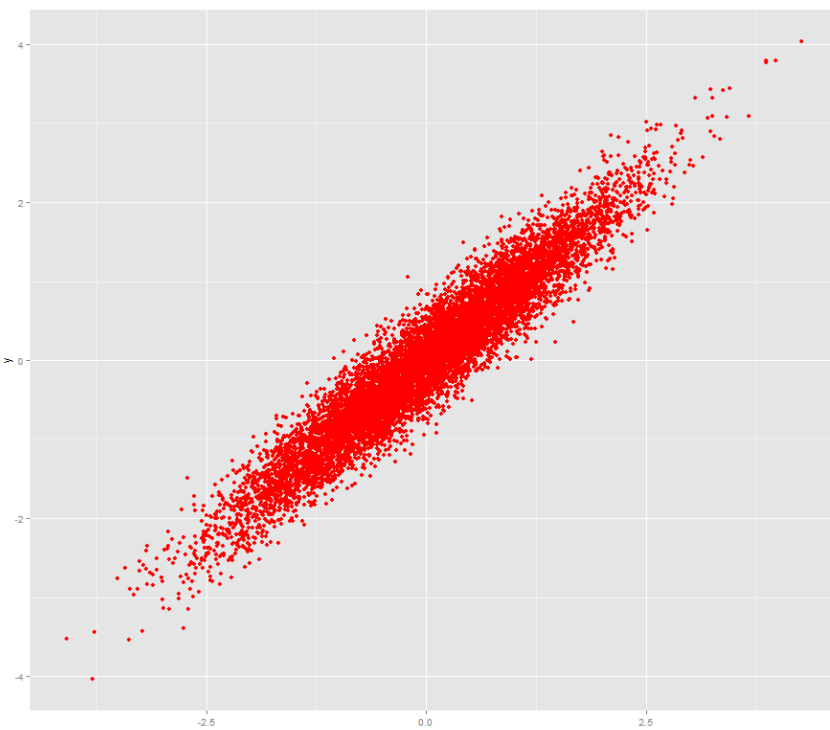
```
Sigma <- matrix(c(1.1,1,1,1),2,2)


# simulate from a Multivariate Normal
Distribution

toy = mvrnorm(n=10000, rep(0, 2), Sigma)

toy.df = as.data.frame(toy)

colnames(toy.df) = c('x', 'y')


qplot(x, y, data = toy.df, colour =
I("red"), size = I(2))
```



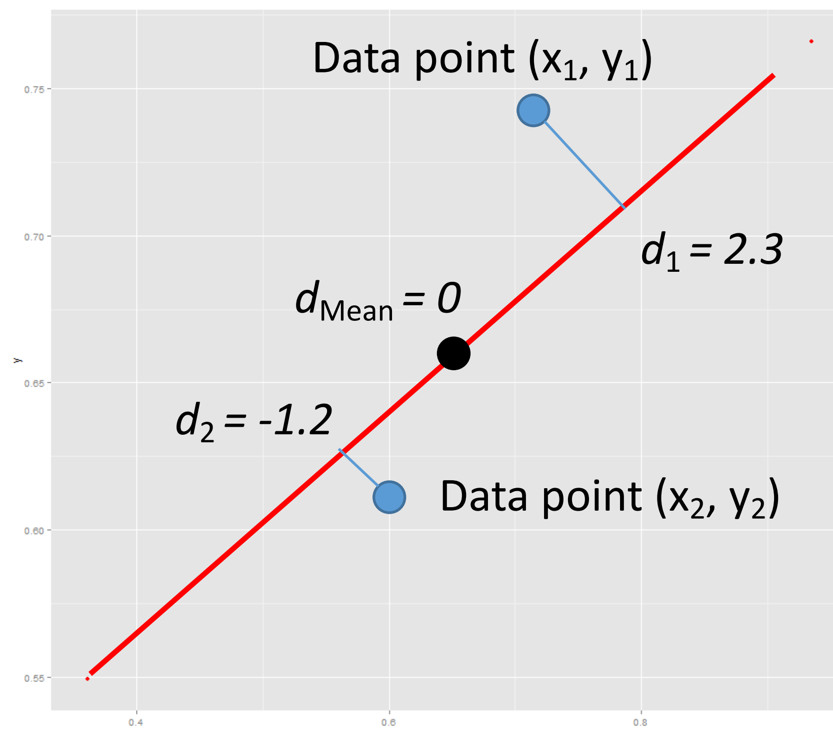
Each point is described with (x,y) coordinates. Suppose to save space you only had enough memory to store 1 number per point instead of 2. What number would you store to best approximate a point's position?



Idea: Collapse the cloud to 1 dimension, then approximate a point (x_i, y_i) by its projected position d_i onto the line.

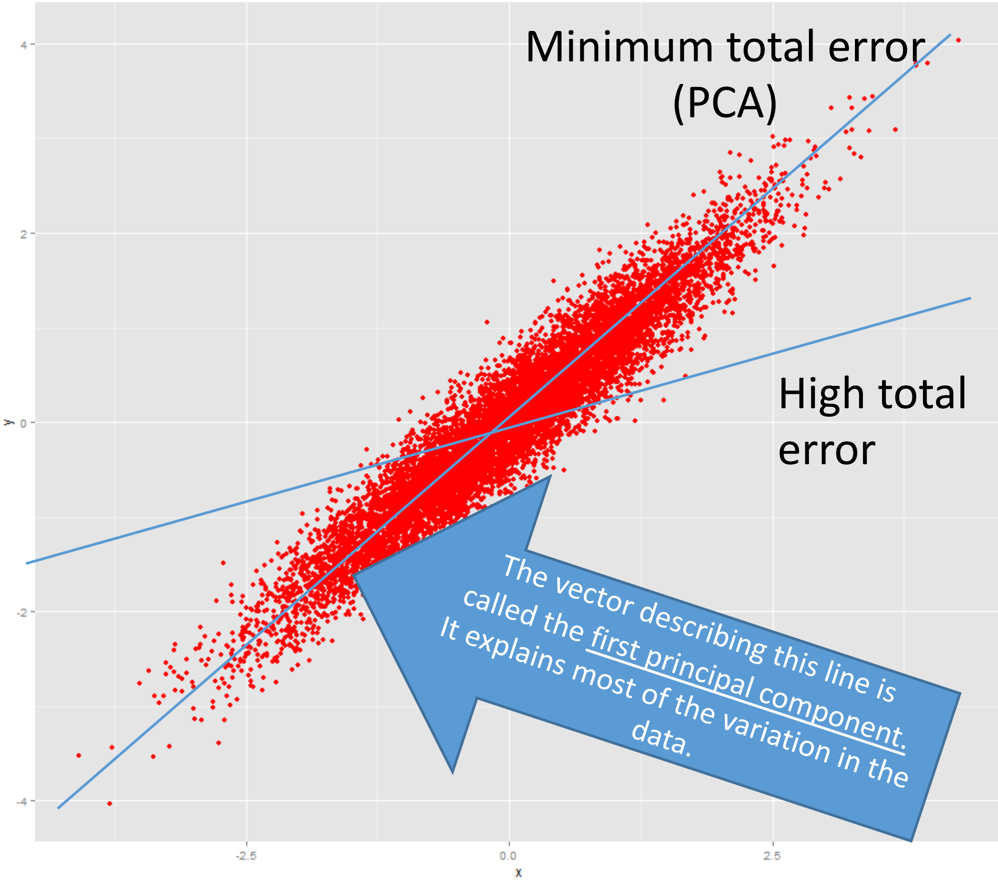


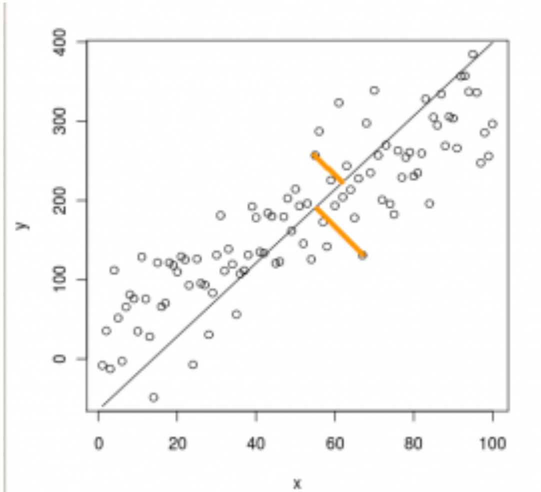
2-dimensional cloud



A 1-dimensional approximation to the cloud (using a linear model)

But how do we pick the 'best' linear approximation? This is what PCA does!

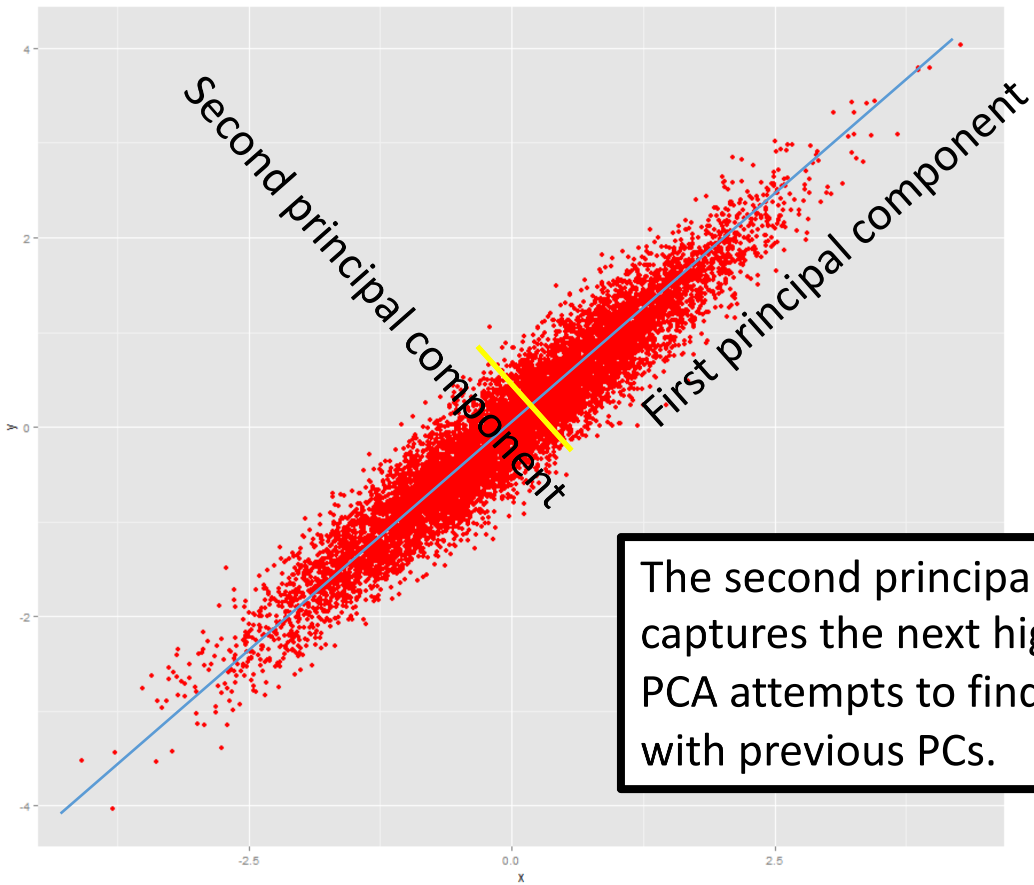


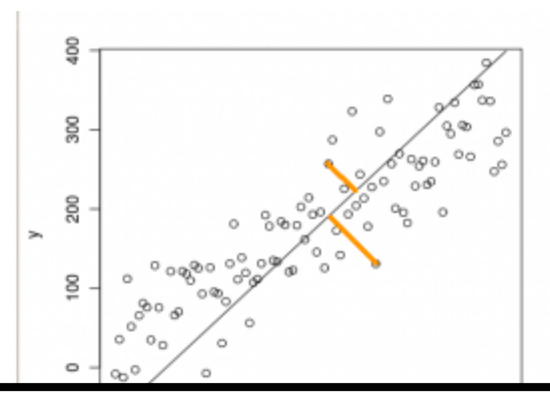


PCA finds the unique model line that minimizes error orthogonal (perpendicular) to the model line.

Source: http://www.cerebralmastication.com/2010/09/principal-component-analysis-pca-vs-ordinary-least-squares-ols-a-visual-explination/

We can repeat this process to improve the approximation.
This will give us the second principal component.





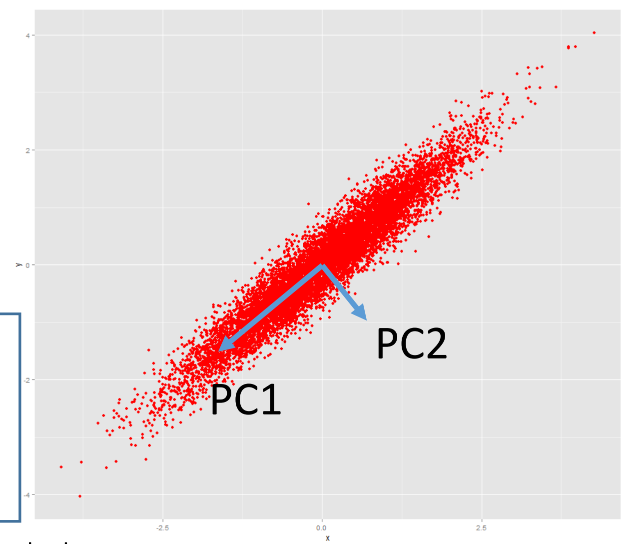
The second principal component (yellow)
captures the next highest orthogonal direction of variance.
PCA attempts to find the next PC that is uncorrelated
with previous PCs.

The principal component vectors have an origin that is the mean (centroid) of the cloud

Source: http://www.cerebralmastication.com/2010/09/principal-component-analysis-pca-vs-ordinary-least-squares-ols-a-visual-explination/

Let's apply PCA to our toy dataset: The R function is **prcomp**

```
> toy.pr = prcomp(toy.df,scale=TRUE)
> toy.pr
Standard deviations:
[1] 1.3978833 0.2142951

Rotation:
          PC1        PC2
x  -0.7071068  0.7071068
y  -0.7071068 -0.7071068
```



* Another R function princomp() does PCA using a different internal method.
Also, prcomp() can handle when # variables > # of observations, princomp can't.
So there's no advantage to using princomp()

Scaling is important for PCA

- Like k-means clustering, PCA is sensitive to the scaling of the variables. Scaling = TRUE invokes prcomp scaling.
- Variables in different units *must* be scaled (e.g. temperature vs mass).
- Variables in the same units but having wildly different variances *should* be scaled.

Two typical pre-processing steps for PCA:

1. Mean subtraction (a.k.a. "mean centering") (Default: center = TRUE)
   - Needed before performing PCA to ensure that the first principal component describes the direction of maximum variance. If mean subtraction is not performed, the first principal component might instead correspond more or less to the mean of the data. A mean of zero is needed for finding a basis that minimizes the mean square error of the approximation of the data.[8]
2. Variable scaling (Default: scale = FALSE)
   - Set scale = TRUE
   - Divide each variable by its standard deviation.

Heptathlon dataset
Scores in 7 events: Seoul 1988 Olympics

```
> heptathlon
```

	hurdles	highjump	shot	run200m	longjump	javelin	run800m	score
Joyner-Kersee (USA)	12.69	1.86	15.80	22.56	7.27	45.66	128.51	7291
John (GDR)	12.85	1.80	16.23	23.65	6.71	42.56	126.12	6897
Behmer (GDR)	13.20	1.83	14.20	23.10	6.68	44.54	124.20	6858
Sablovskaite (URS)	13.61	1.80	15.23	23.92	6.25	42.78	132.24	6540
Choubenkova (URS)	13.51	1.74	14.76	23.93	6.32	47.46	127.90	6540
Schulz (GDR)	13.75	1.83	13.50	24.65	6.33	42.82	125.79	6411
Fleming (AUS)	13.38	1.80	12.88	23.59	6.37	40.28	132.54	6351
Greiner (USA)	13.55	1.80	14.13	24.48	6.47	38.00	133.65	6297
Lajbnerova (CZE)	13.63	1.83	14.28	24.86	6.11	42.20	136.05	6252
Bouraga (URS)	13.25	1.77	12.62	23.59	6.28	39.06	134.74	6252
Wijnsma (HOL)	13.75	1.86	13.01	25.03	6.34	37.86	131.49	6205
Dimitrova (BUL)	13.24	1.80	12.88	23.59	6.37	40.28	132.54	6171
Scheider (SWI)	13.85	1.86	11.58	24.87	6.05	47.50	134.93	6137
Braun (FRG)	13.71	1.83	13.16	24.78	6.12	44.58	142.82	6109

25 observations (athletes), 8 variables

Step 1: Transform the data

Some results are measured in seconds (lower numbers better), others in scores, or metres (higher numbers better).

```
> heptathlon
                     hurdles highjump  shot run200m longjump javelin run800m score
Joyner-Kersee (USA)    12.69     1.86 15.80   22.56     7.27   45.66  128.51  7291
John (GDR)             12.85     1.80 16.23   23.65     6.71   42.56  126.12  6897
Behmer (GDR)           13.20     1.83 14.20   23.10     6.68   44.54  124.20  6858
Sablovskaite (URS)     13.61     1.80 15.23   23.92     6.25   42.78  132.24  6540
Choubenkova (URS)      13.51     1.74 14.76   23.93     6.32   47.46  127.90  6540
```

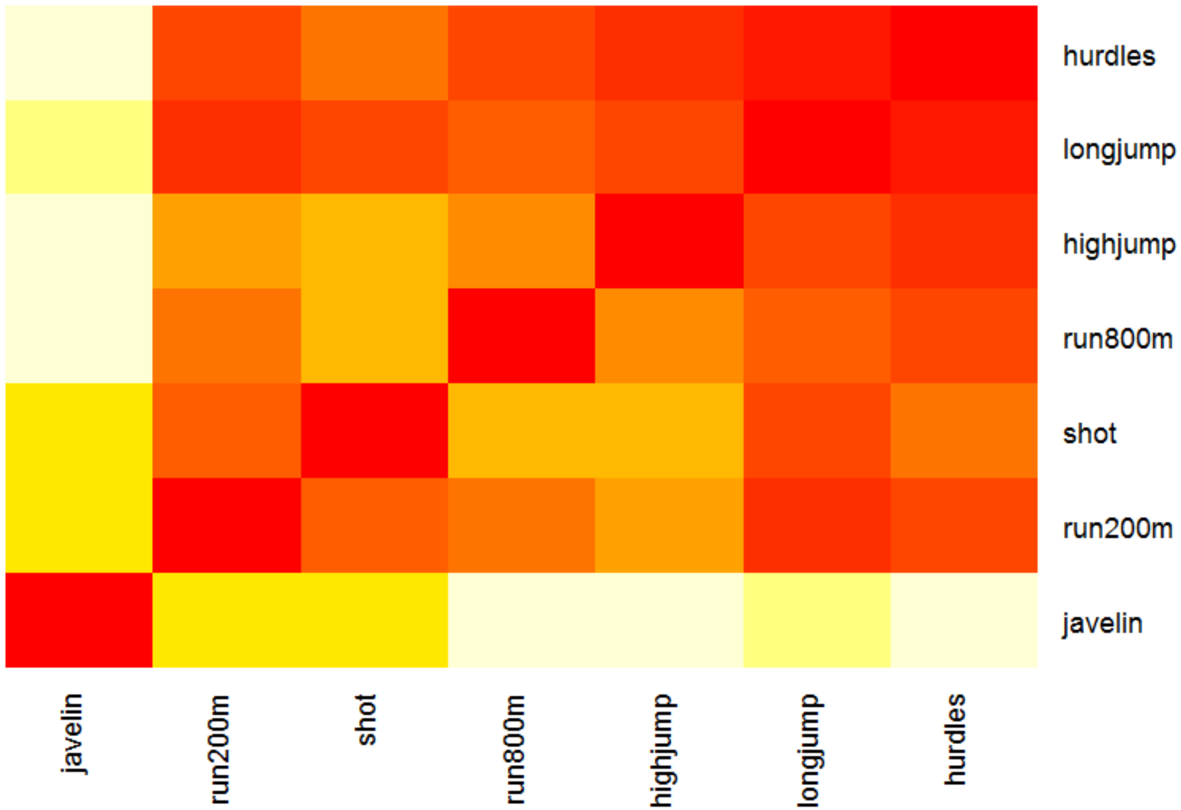
```
heptathlon$hurdles = max(heptathlon$hurdles) - heptathlon$hurdles
heptathlon$run200m = max(heptathlon$run200m) - heptathlon$run200m
heptathlon$run800m = max(heptathlon$run800m) - heptathlon$run800m
```

```
> heptathlon
                     hurdles highjump  shot run200m longjump javelin run800m score
Joyner-Kersee (USA)     3.73     1.86 15.80    4.05     7.27   45.66   34.92  7291
John (GDR)              3.57     1.80 16.23    2.96     6.71   42.56   37.31  6897
Behmer (GDR)            3.22     1.83 14.20    3.51     6.68   44.54   39.23  6858
Sablovskaite (URS)      2.81     1.80 15.23    2.69     6.25   42.78   31.19  6540
Choubenkova (URS)       2.91     1.74 14.76    2.68     6.32   47.46   35.53  6540
```

Step 2: How is performance in one event correlated with other events?

```
> round(cor(heptathlon[,c(1:7)]),2)
         hurdles highjump shot run200m longjump javelin run800m
hurdles     1.00     0.81 0.65    0.77     0.91    0.01    0.78
highjump    0.81     1.00 0.44    0.49     0.78    0.00    0.59
shot        0.65     0.44 1.00    0.68     0.74    0.27    0.42
run200m     0.77     0.49 0.68    1.00     0.82    0.33    0.62
longjump    0.91     0.78 0.74    0.82     1.00    0.07    0.70
javelin     0.01     0.00 0.27    0.33     0.07    1.00   -0.02
run800m     0.78     0.59 0.42    0.62     0.70   -0.02    1.00
```

Can you see any structure in the correlation matrix? Are there groups of related events (variables)?



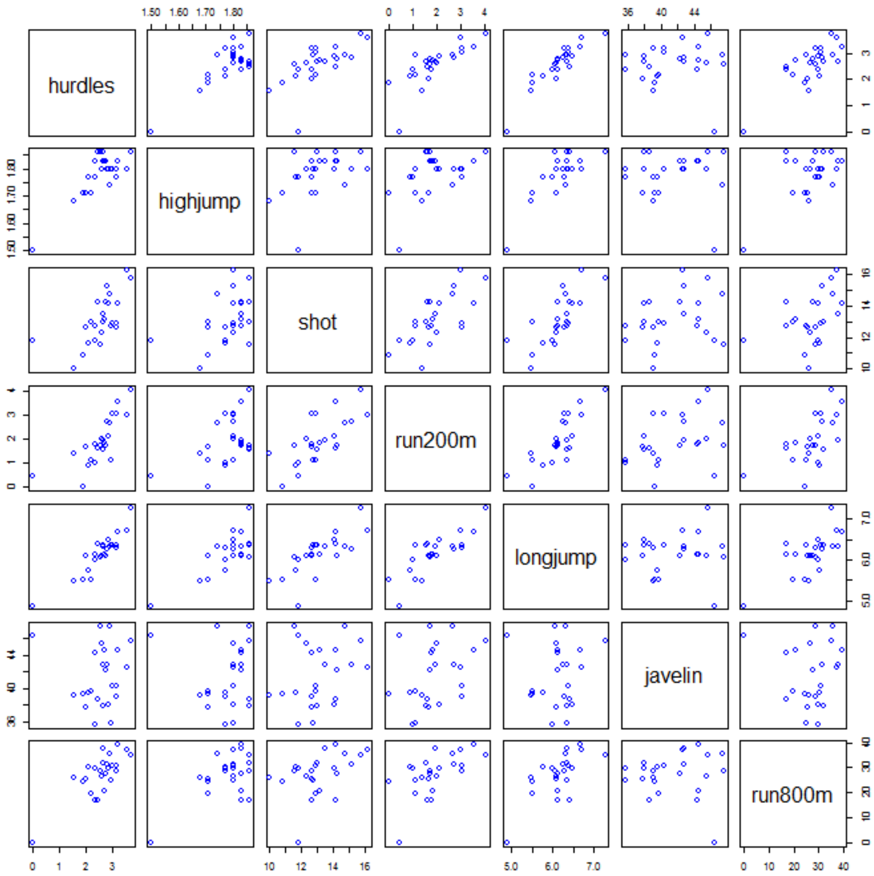
Dark = high correlation
Light = low correlation

```
heatmap.2(round(1- abs(cor(heptathlon[,c(1:7)])),2), trace = "none", dendrogram="none", margins = c(10,10))
```

Could we find a small set of core athletic abilities (factors) that mostly explain the structure of correlated results for a large set of event scores (variables)?

- Is there a "running factor" that would lead to good results across multiple running events?
- A "jumping factor" ?
- A "throwing factor" (arm strength) ?
- Endurance?
- Seven variables is not a large number
    - PCA comes into its own in larger data sets

Scatterplots confirm what we observed in the correlation matrix




```
pairs(heptathlon[,c(1:7)],col="blue")
```

Observations?

1. Clear linear relationships between hurdles, high jump, shot put, 200m, and long jump.

2. Javelin and, to some extent, 800m results are less correlated with the other events.

Running PCA

- Note that the seven events have very different variances. Standard deviation for the 800m is 8.29 (sec) whereas for the high jump it's only 0.078 (m).
- If we work with unscaled scores, the 800m results will have a disproportionate effect.
- Thus we will tell the PCA function to scale all results to have a variance of 1.0.

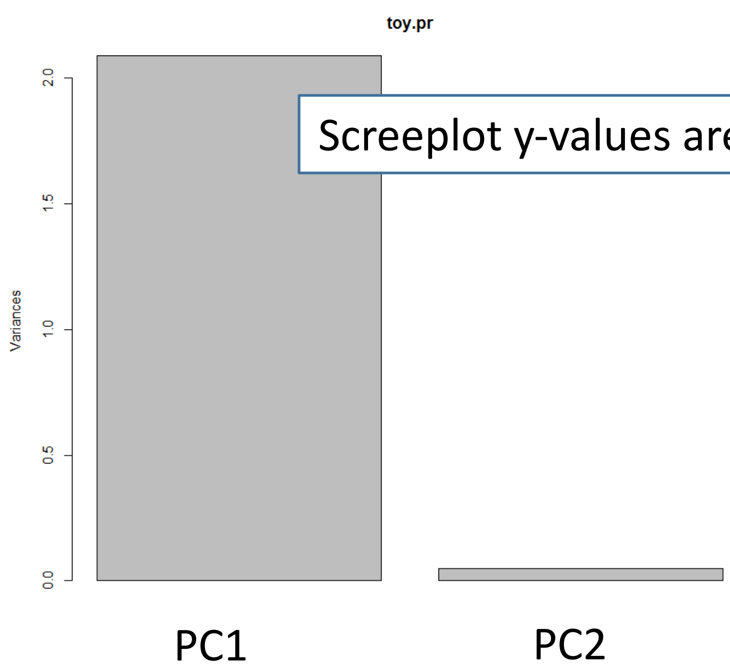
```
hepPCA = prcomp(heptathlon[,c(1:7)], scale=TRUE)
```

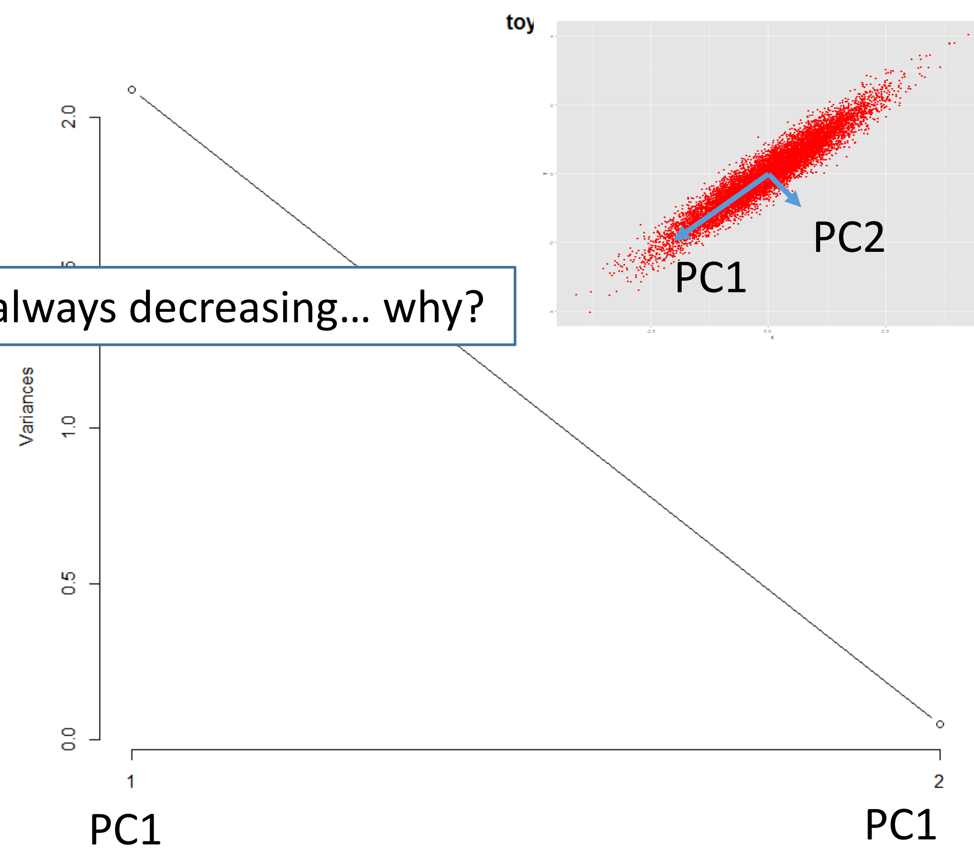
Source: http://users.ecs.soton.ac.uk/jn2/teaching/pca.pdf

Summary of PCA results for heptathlon data

```
> print(hepPCA)
Standard deviations:
[1] 2.1119364 1.0928497 0.7218131 0.6761411 0.4952441 0.2701029 0.2213617

Rotation:
                PC1         PC2         PC3         PC4         PC5         PC6         PC7
hurdles  -0.4528710  0.15792058 -0.04514996  0.02653873 -0.09494792 -0.78334101  0.38024707
highjump -0.3771992  0.24807386 -0.36777902  0.67999172  0.01879888  0.09939981 -0.43393114
shot     -0.3630725 -0.28940743  0.67618919  0.12431725  0.51165201 -0.05085983 -0.21762491
run200m  -0.4078950 -0.26038545  0.08359211 -0.36106580 -0.64983404  0.02495639 -0.45338483
longjump -0.4562318  0.05587394  0.13931653  0.11129249 -0.18429810  0.59020972  0.61206388
javelin  -0.0754090 -0.84169212 -0.47156016  0.12079924  0.13510669 -0.02724076  0.17294667
run800m  -0.3749594  0.22448984 -0.39585671 -0.60341130  0.50432116  0.15555520 -0.09830963

> summary(hepPCA)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6    PC7
Standard deviation     2.1119 1.0928 0.72181 0.67614 0.49524 0.27010 0.2214
Proportion of Variance 0.6372 0.1706 0.07443 0.06531 0.03504 0.01042 0.0070
Cumulative Proportion  0.6372 0.8078 0.88223 0.94754 0.98258 0.99300 1.0000
```

Screeplots show how much variance is explained by each principal component





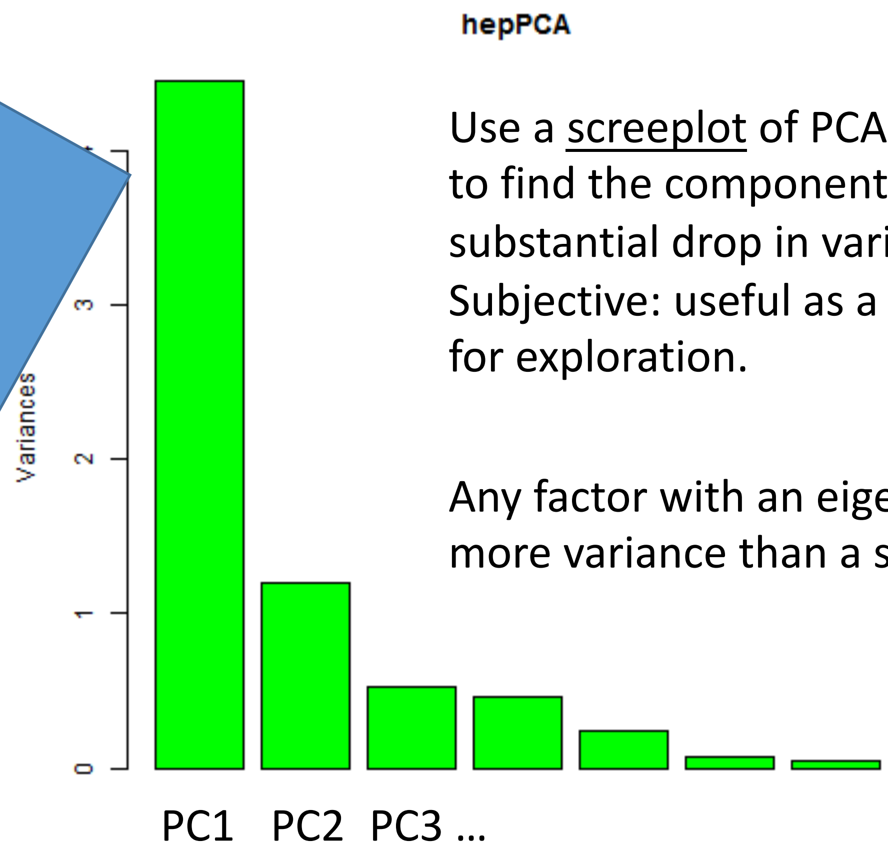
Screeplot y-values are always decreasing… why?

`screeplot(toy.pr)`

`screeplot(toy.pr, type = "lines")`

How to pick the "right" number of factors?

Suggests the first PC captures most of the interesting variation in the data.
Implies that a single factor (skill?) may "explain" performance across multiple events



Use a <u>screeplot</u> of PCA components to find the component giving the last substantial drop in variance.
Subjective: useful as a starting point for exploration.

Any factor with an eigenvalue ≥1 explains more variance than a single observed variable.

Let's test that idea by projecting each athlete's data point (their event scores) onto the first principal component vector to get a single score.

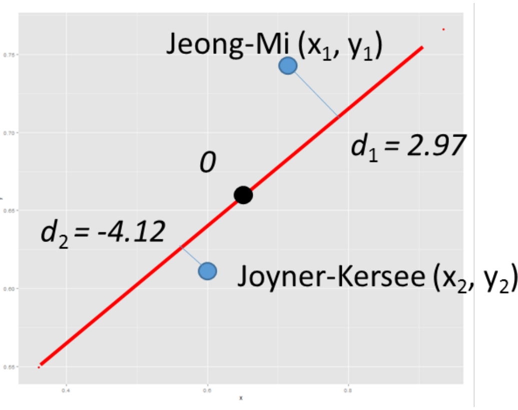
```
> hepPCA$rotation[,1]
   hurdles   highjump       shot    run200m   longjump    javelin     run800m
-0.4528710 -0.3771992 -0.3630725 -0.4078950 -0.4562318 -0.0754090 -0.3749594
> c1 = predict(hepPCA)[,1]
> c1
Joyner-Kersee (USA)          John (GDR)         Behmer (GDR)  Sablovskaite (URS)   Choubenkova (URS)
       -4.121447626        -2.882185935        -2.649633766        -1.343351210        -1.359025
      Fleming (AUS)        Greiner (USA)     Lajbnerova (CZE)        Bouraga (URS)        Wijnsma (HOL
       -1.100385639        -0.923173639        -0.530250689        -0.759819024        -0.556268302
      Scheider (SWI)         Braun (FRG)   Ruotsalainen (FIN)        Yuping (CHN)         Hagger
        0.015461226         0.003774223         0.090747709        -0.137225440         0.17112
       Mulliner (GB)     Hautenauve (BEL)         Kytola (FIN)       Geremias (BRA)       Hui-Ing (TAI)
        1.125481833         1.085697646         1.447055499         2.014029620          2.88029862
        Launa (PNG)
        6.270021972
```

The PCA-based score captures the actual score ranking well.

This places every athlete somewhere on the PC1 line.

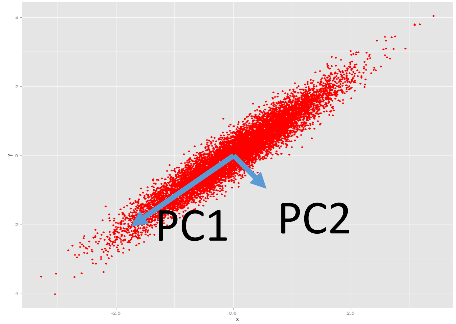
Compare to ranking based on original data:

	hurdles	highjump	shot	run200m	longjump	javelin	run800m	score
Joyner-Kersee (USA)	3.73	1.86	15.80	4.05	7.27	45.66	34.92	7291
John (GDR)	3.57	1.80	16.23	2.96	6.71	42.56	37.31	6897
Behmer (GDR)	3.22	1.83	14.20	3.51	6.68	44.54	39.23	6858
Sablovskaite (URS)	2.81	1.80	15.23	2.69	6.25	42.78	31.19	6540
Choubenkova (URS)	2.91	1.74	14.76	2.68	6.32	47.46	35.53	6540
Schulz (GDR)	2.67	1.83	13.50	1.96	6.33	42.82	37.64	6411
Fleming (AUS)	3.04	1.80	12.88	3.02	6.37	40.28	30.89	6351
Greiner (USA)	2.87	1.80	14.13	2.13	6.47	38.00	29.78	6297
Lajbnerova (CZE)	2.79	1.83	14.28	1.75	6.11	42.20	27.38	6252
Bouraga (URS)	3.17	1.77	12.62	3.02	6.28	39.06	28.69	6252
Wijnsma (HOL)	2.67	1.86	13.01	1.58	6.34	37.86	31.94	6205
Dimitrova (BUL)	3.18	1.80	12.88	3.02	6.37	40.28	30.89	6171
Scheider (SWI)	2.57	1.86	11.58	1.74	6.05	47.50	28.50	6137
Braun (FRG)	2.71	1.83	13.16	1.83	6.12	44.58	20.61	6109



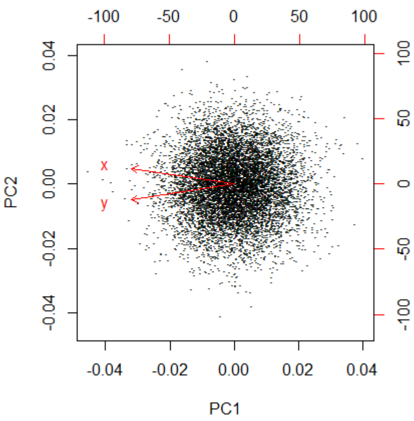
Visualizing PCA results with biplots

- A biplot shows:
  - The data points (top/right scale)
  - The variables (bottom/left scale)



Original plot

- Data are plotted as "seen" from the PC axes
- Variables are plotted by their PC loadings
  - The <u>angle</u> formed by the vectors for any two variables reflects their actual pairwise correlation



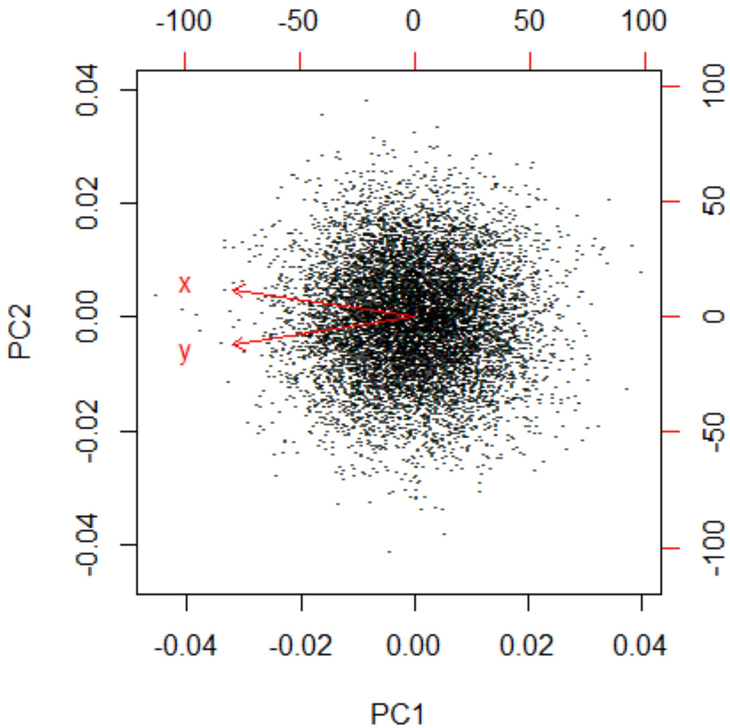
Biplot

Biplot: combines data scatterplot with variable plot (vectors)

Points close =
Observations with
similar component projections

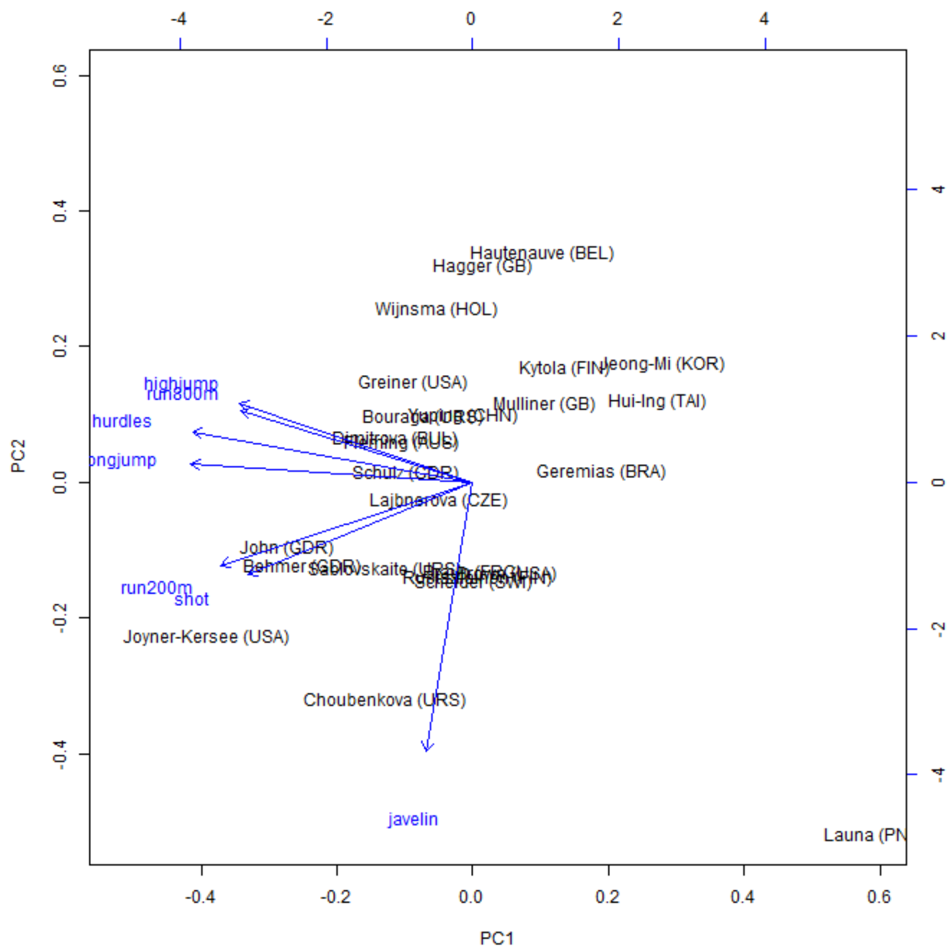
Vectors close =
Variables that are correlated

Observations whose points project furthest in the direction of a variable have the most of whatever the variable measures.




```
biplot(toy.pr, xlabs=rep(".",nrow(toy.df)))
```

Biplot of heptathlon PCA components

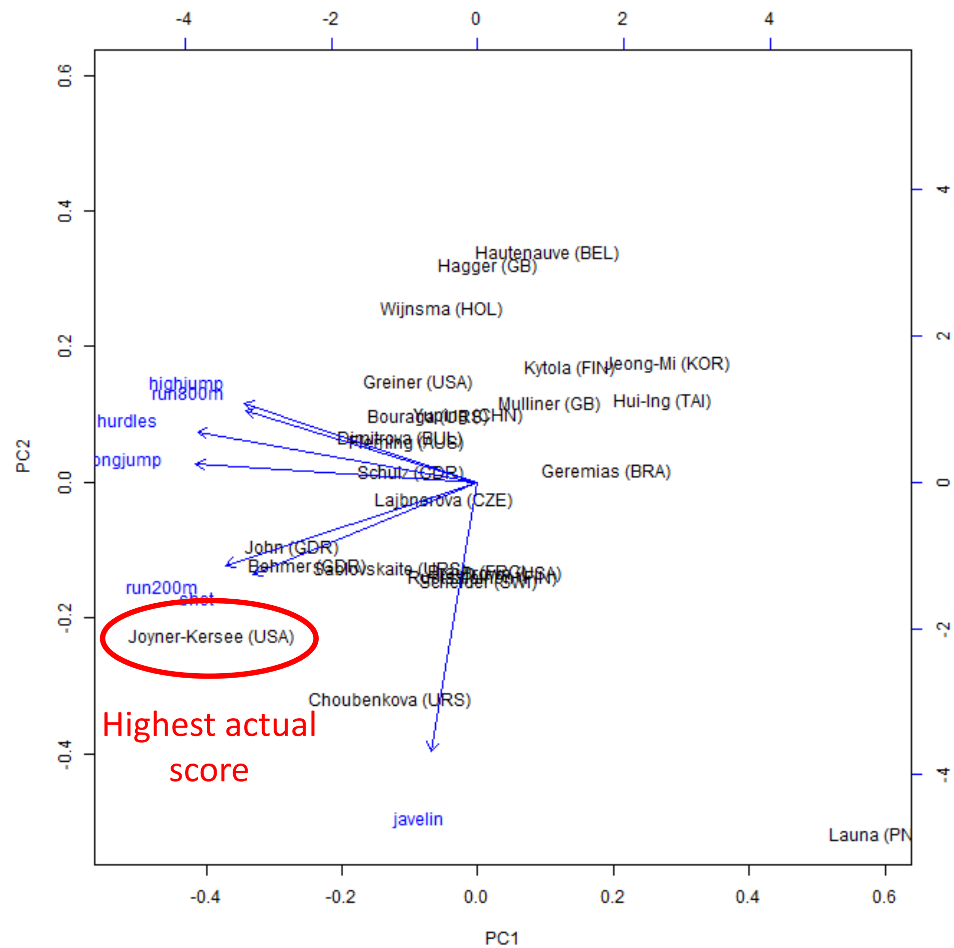


<u>Remember</u>:
For any pair of variables, the angle between their biplot vectors reflects their actual correlation

Athletes good at one event are likely to be good at similar events

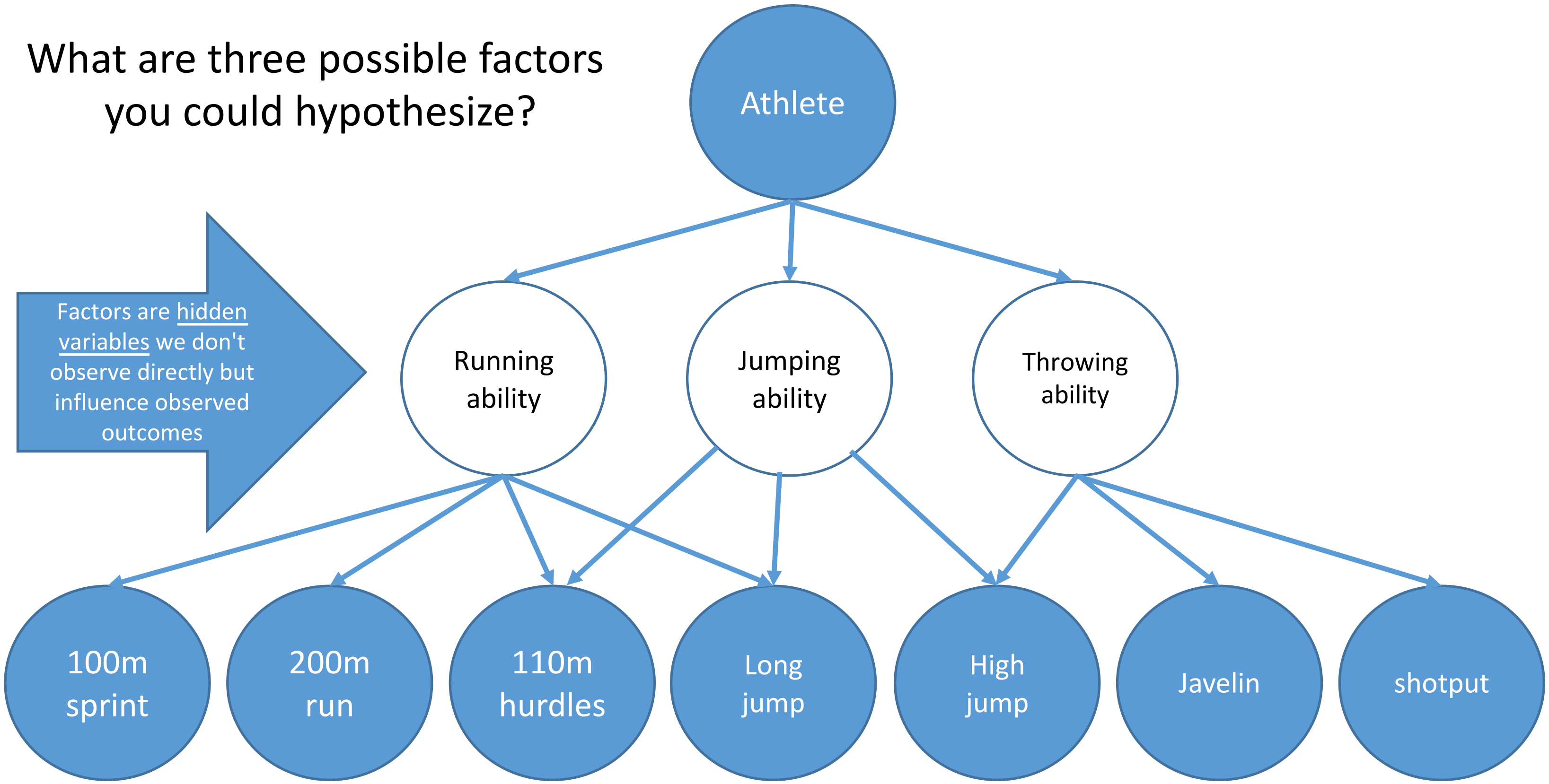
Interpreting heptathlon PCA results

- Knowing only an athlete's score on component 1 (along the first line) we can recover a pretty good guess about their results, i.e. location in 7-d space
- If we also know component 2 our guess would be even more accurate.
- Having all seven components, we could completely reconstruct their scores
  - But there isn't much point in this since we had the scores already!



Exploratory factor analysis: recall our original goal

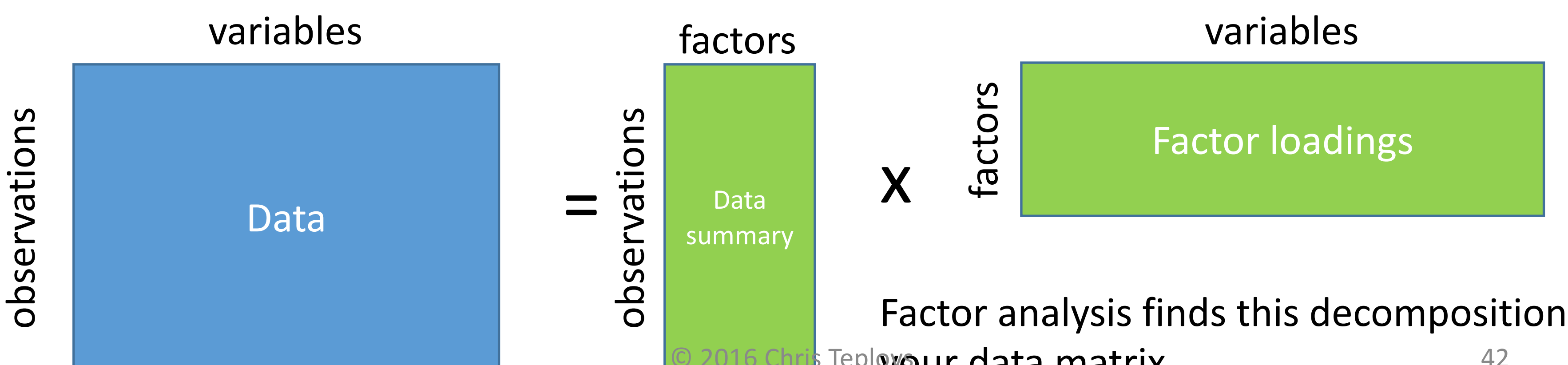
What are three possible factors you could hypothesize?



Each event score = variable

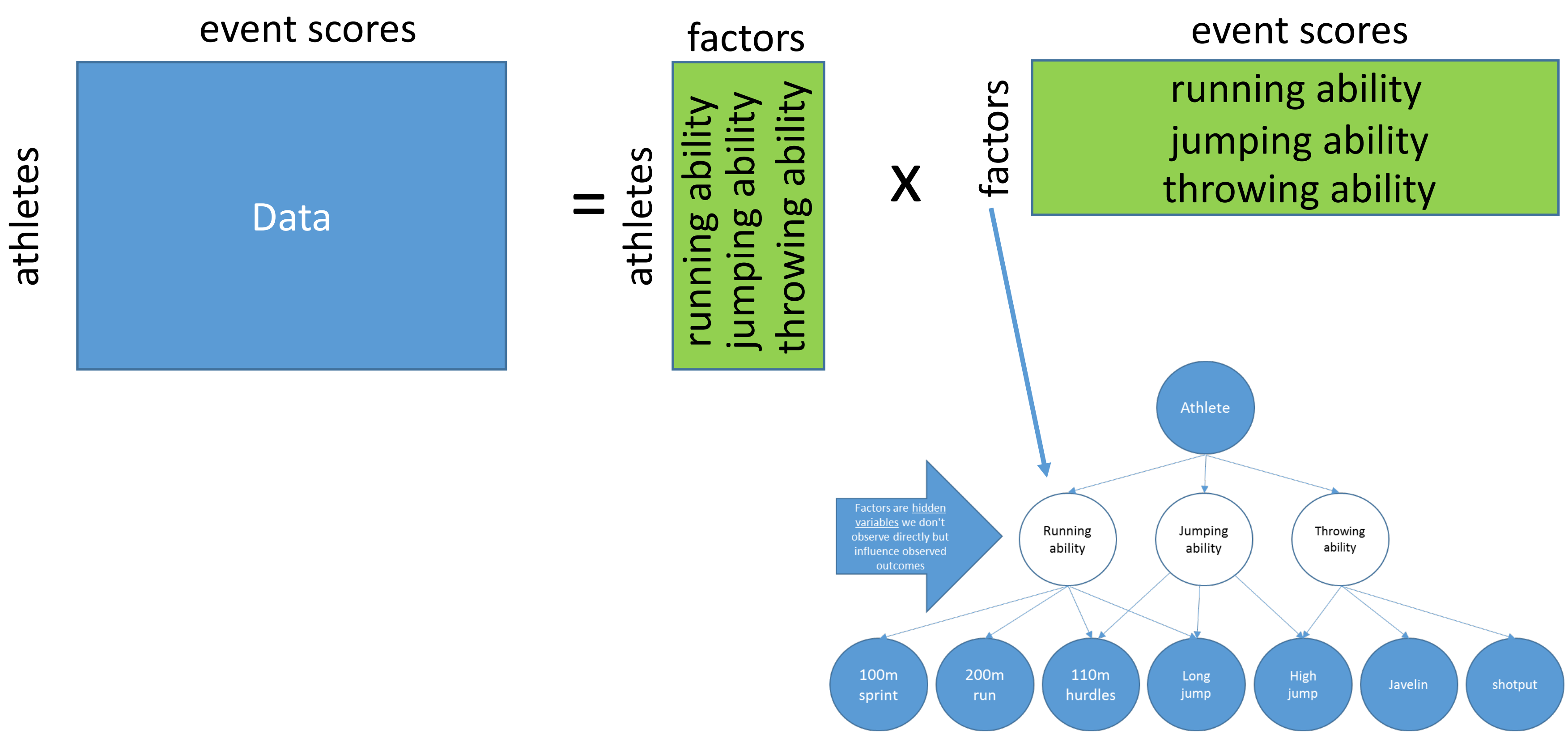
Exploratory factor analysis

- Finds *k* factors that "explain" the correlation structure in the observed variables
- Traditionally specify *k* in advance
- You can specify whether the factors can be correlated or uncorrelated
- More flexible and general than PCA



Factor analysis finds this decomposition of your data matrix

Factors in the hepthalon



EFA on smoking survey dataset

Data source:

http://en.wikiversity.org/wiki/Survey_research_and_design_in_psychology/Tutorials/Psychometrics/Exploratory_factor_analysis

Five observed variables are the responses to these 5 questions:

1. I think smoking is acceptable.
2. I don't care if people smoke around me.
3. I don't think people should smoke in restaurants
4. I think people should have the right to smoke.
5. I don't think people should smoke around food.

Each person's response to Q1-Q5 is on a scale of 1-100.

N=107 people responded to the survey.

```
smoke = spss.get("data_14_1.sav") # Hmisc package
```

Performing EFA in R using factanal

```
Factor Analysis

Description
Perform maximum-likelihood factor analysis on a covariance matrix or data matrix.

Usage
factanal(x, factors, data = NULL, covmat = NULL, n.obs = NA,
         subset, na.action, start = NULL,
         scores = c("none", "regression", "Bartlett"),
         rotation = "varimax", control = NULL, ...)
```

<u>Key Arguments:</u>

```
x
A formula or a numeric matrix or an object that can be coerced to a numeric matrix.

factors
The number of factors to be fitted.

rotation
character. "none" or the name of a function to be used to rotate the factors: it will be
called with first argument the loadings matrix, and should return a list with component
loadings giving the rotated loadings, or just the rotated loadings.
```

```
factanal(x = smoke, factors = 2, rotation = "varimax")
```

EFA: Output using orthogonal factors

```
Call:
factanal(x = smoke, factors = 2, rotation = "varimax")

Uniquenesses:
  qn1   qn2   qn3   qn4   qn5
0.025 0.197 0.389 0.010 0.042

Loadings:
    Factor1 Factor2
qn1  0.987
qn2  0.885  -0.138
qn3 -0.110   0.774
qn4  0.994
qn5          0.978

               Factor1 Factor2
SS loadings      2.760   1.577
Proportion Var   0.552   0.315
Cumulative Var   0.552   0.867

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 0.69 on 1 degree of
freedom.
The p-value is 0.405
```

Uniqueness is how much each variable is unlike other variables:

- Close to 1 = unique
- Close to zero = heavily correlated with other variables

Factor loadings show the correlation of the original variable with a factor.

- Shows importance of the variable to a factor

EFA: Allowing correlated (oblique) factors with "promax" rotation method

```
Call:
factanal(x = smoke, factors = 2, rotation = "varimax")

Uniquenesses:
  qn1   qn2   qn3   qn4   qn5 
0.025 0.197 0.389 0.010 0.042 

Loadings:
    Factor1 Factor2
qn1  0.987         
qn2  0.885  -0.138 
qn3 -0.110   0.774 
qn4  0.994         
qn5          0.978 

               Factor1 Factor2
SS loadings      2.760   1.577
Proportion Var   0.552   0.315
Cumulative Var   0.552   0.867

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 0.69 on 1 degree of
freedom.
The p-value is 0.405
```

```
Call:
factanal(x = smoke, factors = 2, rotation = "promax")

Uniquenesses:
  qn1   qn2   qn3   qn4   qn5 
0.025 0.197 0.389 0.010 0.042 

Loadings:
    Factor1 Factor2
qn1  0.993         
qn2  0.881         
qn3          0.775 
qn4  1.002         
qn5          0.986 

               Factor1 Factor2
SS loadings      2.770   1.581
Proportion Var   0.554   0.316
Cumulative Var   0.554   0.870

Factor Correlations:
        Factor1 Factor2
Factor1   1.000  -0.171
Factor2  -0.171   1.000

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 0.69 on 1 degree of
freedom.
The p-value is 0.405
```

EFA: Rotation parameter

```
factanal(x = smoke, factors = 2, rotation = "varimax")
```

- Rotation: optimization method used to find factors that make the pattern of loadings clearer
- varimax: orthogonal factors (uncorrelated)
- promax: oblique (correlated)
  - Note that with promax, you get a matrix of factor correlations
- Strategy: First try promax, look at factor correlations
  - Look at factor correlation matrix for correlations > 0.32
  - If many correlations exceed 0.32 there is 10% or more overlap in variance among factors and promax is justified
  - Otherwise use varimax

```
Factor Correlations:
        Factor1 Factor2
Factor1   1.000  -0.171
Factor2  -0.171   1.000
```

More information: http://jalt.org/test/PDF/Brown31.pdf

Results of EFA on smoking survey data. How well does a 2-factor model fit?

```
factanal(x = smoke, factors = 2, rotation = "varimax")

Uniquenesses:
  qn1   qn2   qn3   qn4   qn5
0.025 0.197 0.389 0.010 0.042

Loadings:
    Factor1 Factor2
qn1  0.987
qn2  0.885  -0.138
qn3 -0.110   0.774
qn4  0.994
qn5          0.978

               Factor1 Factor2
SS loadings      2.760   1.577
Proportion Var   0.552   0.315
Cumulative Var   0.552   0.867

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 0.69 on 1 degree of freedom.
The p-value is 0.405
```

- Significance level of p-value
  - Test of the null hypothesis that N common factors are sufficient to explain the intercorrelations among the variables
  - Increase the # of factors until non-significant result is obtained
  - p-values > 0.05 indicate good fit: hypothesis of perfect fit is not contradicted

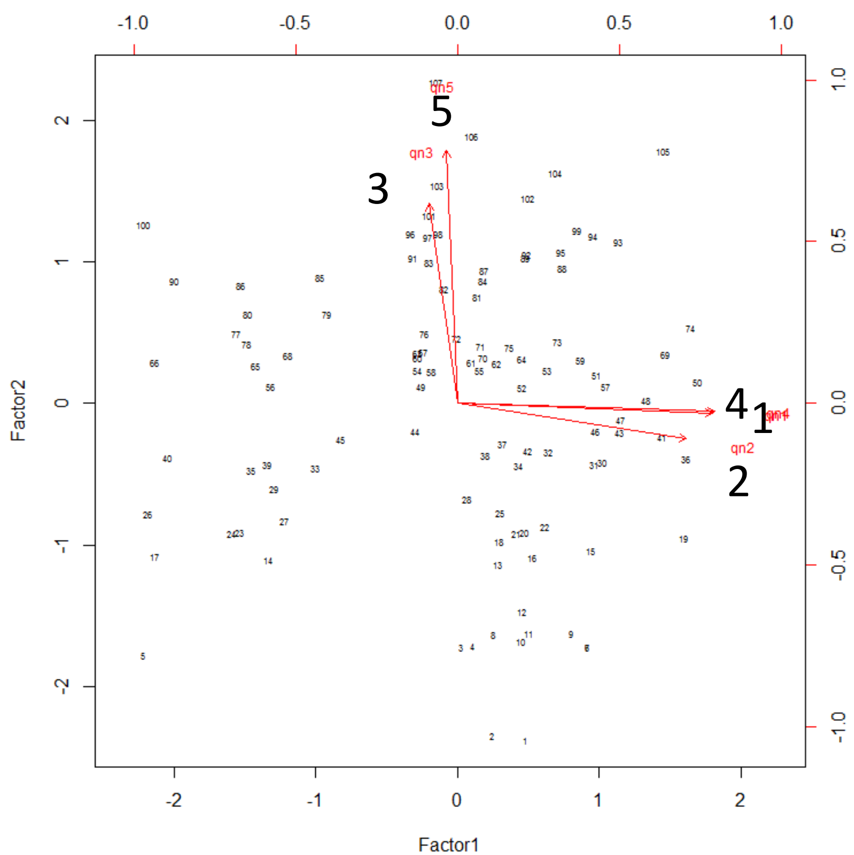
What about a 1-factor model?

```
factanal(x = smoke, factors = 1, rotation = "varimax")

Uniquenesses:
  qn1   qn2   qn3   qn4   qn5 
0.024 0.208 0.981 0.012 0.993 

Loadings:
    Factor1
qn1  0.988 
qn2  0.890 
qn3 -0.139 
qn4  0.994 
qn5        

               Factor1
SS loadings      2.782
Proportion Var   0.556

Test of the hypothesis that 1 factor is sufficient.
The chi square statistic is 95.08 on 5 degrees of freedom.
The p-value is 5.75e-19
```

- Significance level of p-value
  - Test of the null hypothesis that N common factors are sufficient to explain the intercorrelations among the variables
  - Increase the # of factors until non-significant result is obtained
  - p-values > 0.05 indicate good fit: hypothesis of perfect fit is not contradicted

Thus the 2-factor model is more likely to be a perfect fit, while the 1-factor model is highly unlikely to be a perfect fit

Biplot for a two-factor EFA analysis of smoking survey responses



\# 1 I think smoking is acceptable.
\# 2 I don't care if people smoke around me.
\# 3 I don't think people should smoke in restaurants
\# 4 I think people should have the right to smoke.
\# 5 I don't think people should smoke around food.

```
Loadings:
    Factor1 Factor2
qn1  0.987
qn2  0.885  -0.138
qn3 -0.110   0.774
qn4  0.994
qn5          0.978
```

```
biplot(smoke.fac$scores, smoke.fac$loadings, cex=c(0.5,0.75))
```

How should we interpret and name the factors?

- Examine the variables that load heavily on the factor
- Try do decide what model is common to these variables.
- Name the factor after that construct.

- How would you interpret the two smoking survey factors?
  - Factor 1: Pro-smoking
  - Factor 2: Anti-smoking

# 1	I think smoking is acceptable.
# 2	I don't care if people smoke around me.
# 3	I don't think people should smoke in restaurants
# 4	I think people should have the right to smoke.
# 5	I don't think people should smoke around food.

```
Loadings:
    Factor1 Factor2
qn1  0.987
qn2  0.885  -0.138
qn3 -0.110   0.774
qn4  0.994
qn5          0.978
```

When have we found factors with "simple" structure?

Thurstone (1947) listed five criteria:

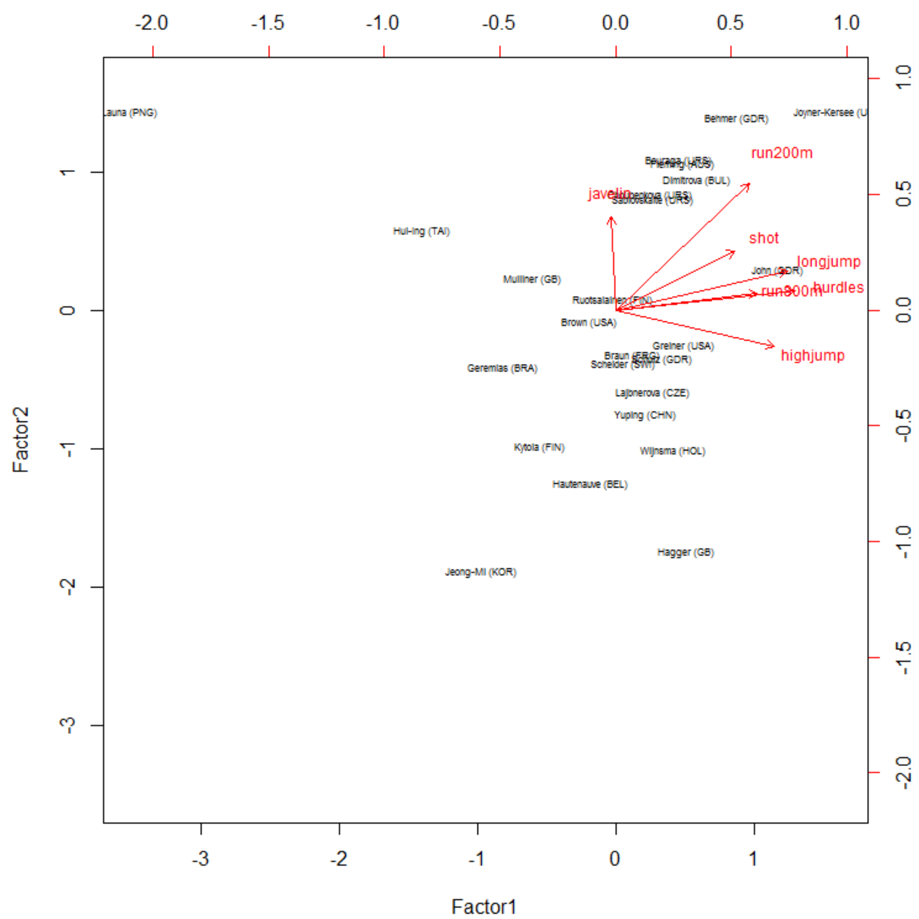
1. Each variable should produce at least one zero loading on some factor. ✓
2. Each factor should have at least as many zero loadings as there are factors. ~
3. Each pair of factors should have variables with significant loadings on one and zero loadings on the other. ✓
4. Each pair of factors should have a large proportion of zero loadings on both factors (if > 3 factors total). N/A
5. Each pair of factors should have only a few complex variables. ~

Smoking survey

```
Loadings:
    Factor1 Factor2
qn1  0.987
qn2  0.885  -0.138
qn3 -0.110   0.774
qn4  0.994
qn5          0.978
```

Source: **Thurstone**, L.L. (1947), Multiple Factor Analysis, U of C Press.

A two-factor EFA analysis of heptathlon results



How can we interpret heptathlon factors? Are they "simple"?

```
Call:

factanal(x = heptathlon[, c(1:7)], factors = 2, rotation = "varimax")

Uniquenesses:
 hurdles highjump     shot  run200m longjump  javelin  run800m 
   0.052    0.224    0.484    0.005    0.094    0.743    0.406 

Loadings:
         Factor1 Factor2
hurdles   0.968   0.104 
highjump  0.859  -0.196 
shot      0.644   0.318 
run200m   0.725   0.685 
longjump  0.928   0.210 
javelin           0.507 
run800m   0.765         

               Factor1 Factor2
SS loadings      4.064   0.928
Proportion Var   0.581   0.133
Cumulative Var   0.581   0.713

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 13.07 on 8 degrees of freedom.
The p-value is 0.11 
```

<u>Factor interpretations</u>

Factor 1: Running and Jumping?

- Hurdles, long jump, high jump

Factor 2: Running and throwing?

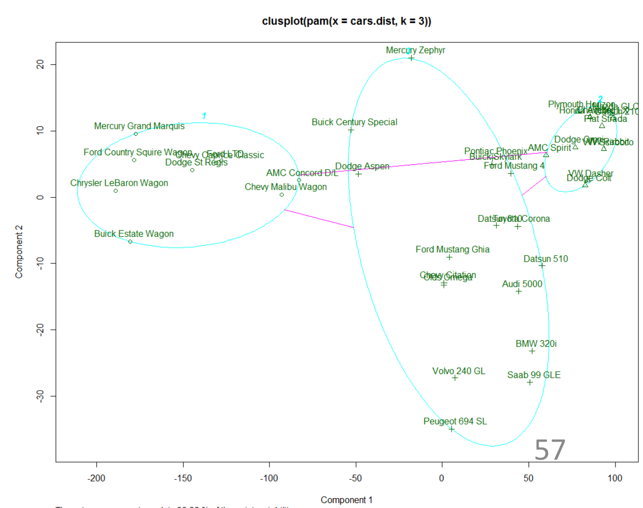
- Javelin

Comparing PCA and EFA

- Both are variable reduction techniques:
  - Explain lots of observed variables in terms of a few hidden variables
- But they solve different optimization problems
- PCA
  - Reduces data using a small number of principal components that account for most of the variance
  - Typically used when variables are highly correlated
  - Good as a fast, initial look at likely number of factors
  - Sensitive to scaling
  - Solution is usually a means to an end, e.g. compress the data
- EFA
  - More general and flexible than PCA in finding interesting structure
  - Identifies the number and nature of likely latent variables (factors)
  - Factors may be correlated or have specific structure
  - Not sensitive to scaling (if maximum likelihood method)
  - Solution is of interest in itself
- Considering k+1 components does not change first k PCA.. But it may change solution to EFA factors.

Dimensionality reduction methods can optimize for different objectives

- Clustering
  - Maximize likelihood of the observed data
  - Maximize cluster 'quality'
  - Cluster = factor: many similarities w/ factor analysis
- Topic modeling
  - Maximize likelihood of observed data (under constraints)
- MDS: multi-dimensional scaling
  - Minimize distance distortion



Bonus homework 5: Factor analysis

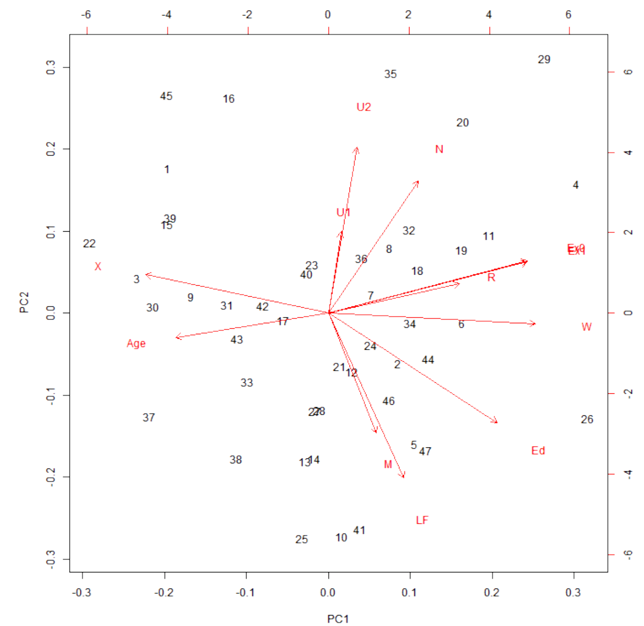
- Part 1: U.S. crime dataset (PCA) for 47 states
  - Variables include:
    - R: Crime rate: # of offenses reported to police per million population
    - Ed: Mean # of years of schooling x 10 for persons of age 25 or older
    - Ex0: 1960 per capita expenditure on police by state and local government
    - Ex1: 1959 per capita expenditure on police by state and local government

  <u>Use PCA</u> to answer questions about the relationships between these variables.

- Part 2: Managerial survey dataset (EFA)

  11 observed Variables:
  - I show confidence in my staff
  - I let my staff know they are doing well
  - I give feedback to staff on how well they are working
  - I would personally compliment staff if they did outstanding work
  - I believe in setting goals and achieving them
  - I achieve the things I want to get done in a day
  - I never try to put off until tomorrow what I can finish today
  - I plan the use of my time well
  - I remain clear headed when too many demands are made upon me
  - I rarely overlook important factors when plans are made
  - I handle complex problems efficiently

  <u>Use EFA</u>: Are there underlying fundamental skills that "produce" these 11 skills?



Crime variables PCA biplot

What you should know

- The basic concepts behind PCA and EFA
- How PCA & EFA are similar and different
- How to prepare data for factor analysis
- How to apply factor analysis in R and interpret the results
- How to generate and interpret screeplots and biplots
- Familiarity with heuristics for choosing and interpreting factors

Break time

Further references

- Exploratory factor analysis
  - http://www.let.rug.nl/~nerbonne/teach/rema-stats-meth-seminar/Factor-Analysis-Kootstra-04.PDF
- Principal components analysis
  - http://www.amazon.com/Principal-Component-Analysis-I-T-Jolliffe/dp/0387954422

EFA: The number of observed variables limits the number of factors

- If you set *k* too high, you will get this error:
- e.g. "*k* factors are too many for 7 variables"
- It means you don't have enough data to fit a model with the # of parameters implied by *k* factors
  - E.g. 2-factor model
    - 3 loadings on each of 2 factors = 6 parameters
    - 3 residual variances (1 per variable)
    - 2-factor model is underidentified with only 3 variables