Decathlon PCA Exercise

Eli Cook

Source files & load and clean data

```
setwd(sourcedir)
source("PCAplots.R")
```

```
##
## Attaching package: 'ggpubr'

## The following object is masked from 'package:plyr':
##
##     mutate
```

```
library(ade4)

# load olympic dataset with decathlon data
data(olympic)

# create a data frame with the decathlon data
decathlon.data <- olympic$tab
decathlon.data$score <- olympic$score

# change French event names to English
names(decathlon.data)[names(decathlon.data) == "long"] <- "long_jump"
names(decathlon.data)[names(decathlon.data) == "poid"] <- "shot_put"
names(decathlon.data)[names(decathlon.data) == "haut"] <- "high_jump"
names(decathlon.data)[names(decathlon.data) == "110"] <- "110_hurdles"
names(decathlon.data)[names(decathlon.data) == "disq"] <- "discus"
names(decathlon.data)[names(decathlon.data) == "perc"] <- "pole_vault"
names(decathlon.data)[names(decathlon.data) == "jave"] <- "javelin"
```

1. Describe this dataset. What are the variables and what are the different observations? What relationships would you expect from this dataset?

The variables are different events in the decathlon, and the observations are each athlete performing for one meet. I would expect certain events to score similarly, for instance all the throwing events would be higher or lower.

2. Now look at your data using scatter plot matrices.

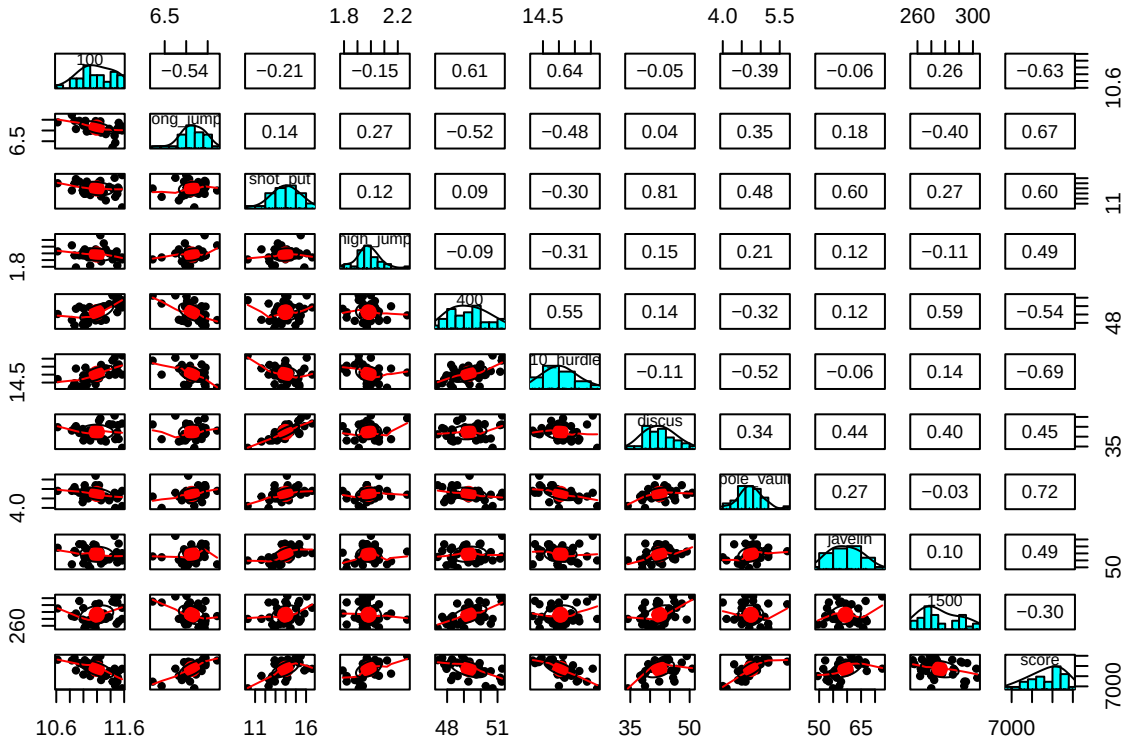
```
library(psych)
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:scales':
##
##     alpha, rescale

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
pairs.panels(decathlon.data)
```



3. Which features are strongly correlated? Which are most predictive of score?

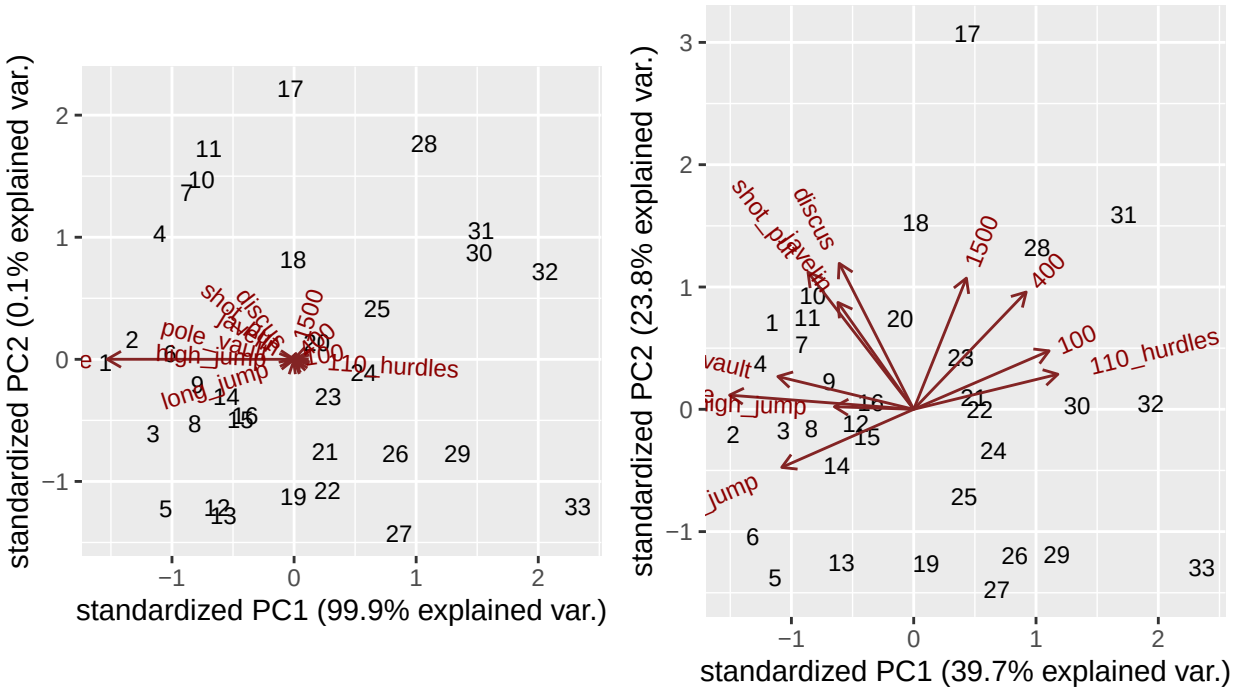
Discus and shotput are highly correlated, and the pole vault and the score are highly correlated.

4. Now create principal components with both the covariance and correlation matrices.

```
pca.cov = princomp(decathlon.data)
pca.cor = princomp(decathlon.data, cor = T)
```

5. Create a biplot of your data for both the covariance and correlation matrices

```
cov_biplot <- ggbiplot(pca.cov, labels=row(decathlon.data)[,1])
cor_biplot <- ggbiplot(pca.cor, labels = row(decathlon.data)[,1])
ggarrange(cov_biplot, cor_biplot, ncol = 2, nrow = 1)
```



6. What do you notice about the biplots with the 2 methods?

With the covariance biplot, almost all of the vairance is explained by the score, but with the correlation biplot there is much less variance explained by score.

From here on out, use the correlation matrix.

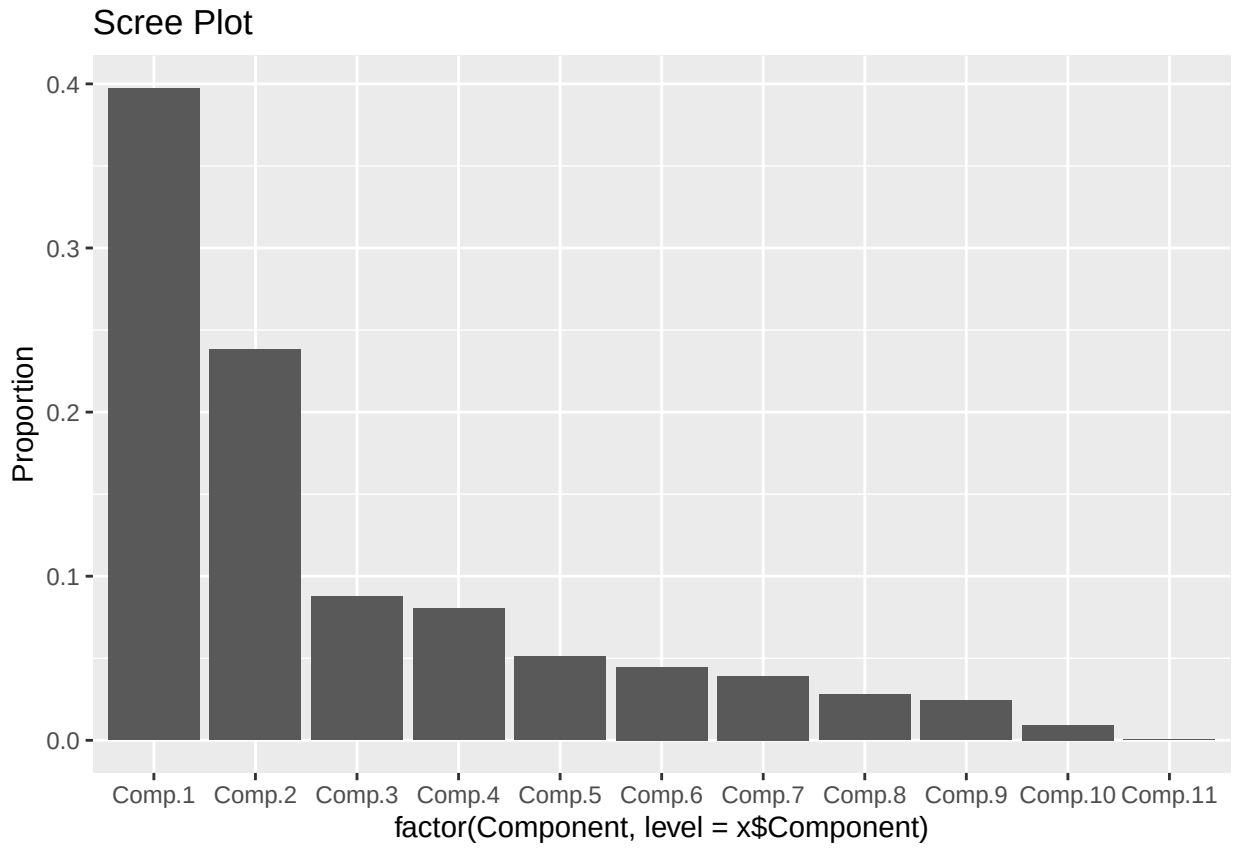
7. Describe the relationships between the different variables. What do they imply about which decathlon events require similar or different strengths? (Note: In some events you want high scores, and in some events you want low scores.)

Also the running events are more closely related, while the throwing events are clustered, and the jumping events are clustered too. The running events are across from the jumping events, which makes sense because the running events goal times are lower, and the jumping event goal distances are higher, so the opposite correlation means that the better runners (lower time) are better jumpers (further distances).

8. Create a screeplot.

```
ggscreeplot(pca.cor)
```

```
## $var
##     Component   Proportion
##        <char>        <num>
##  1:    Comp.1 0.3972695416
##  2:    Comp.2 0.2380642087
##  3:    Comp.3 0.0875544913
##  4:    Comp.4 0.0802439094
##  5:    Comp.5 0.0510331041
##  6:    Comp.6 0.0447553544
##  7:    Comp.7 0.0391986302
##  8:    Comp.8 0.0278923925
##  9:    Comp.9 0.0242681817
## 10:   Comp.10 0.0092872247
## 11:   Comp.11 0.0004329614
##
## $plot
```



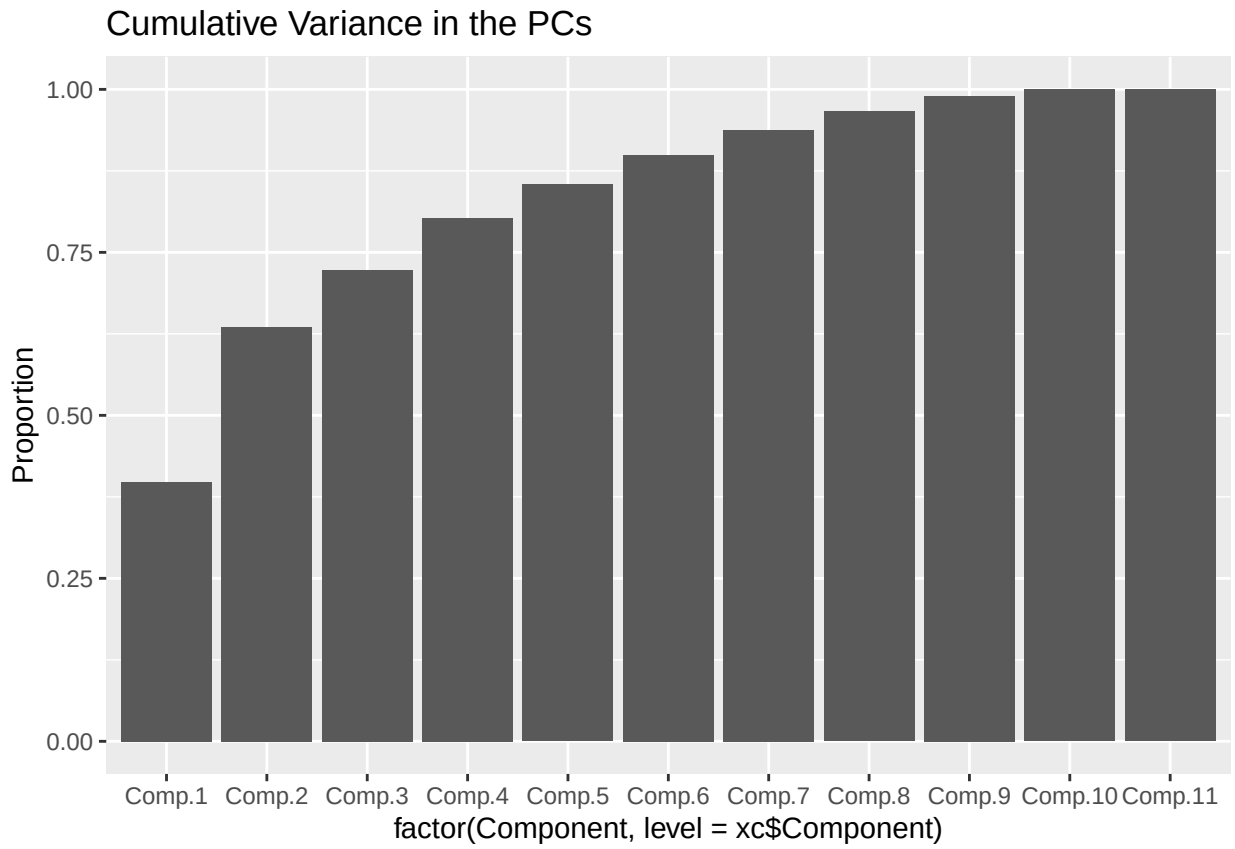
9. How many components do the graphs suggest is sufficient to explain most of the variation in the data? (Hint: consider the first method for choosing the number of PCs described in the slides.)

I would say that around 3 components is sufficient to explain most of the variation, because there is a kink in the graph around then.

10. Create a cumulative variance plot.

```
cumplot(pca.cor)
```

```
## $cumvar
##     Component Proportion
##        <char>      <num>
##  1:    Comp.1  0.3972695
##  2:    Comp.2  0.6353338
##  3:    Comp.3  0.7228882
##  4:    Comp.4  0.8031322
##  5:    Comp.5  0.8541653
##  6:    Comp.6  0.8989206
##  7:    Comp.7  0.9381192
##  8:    Comp.8  0.9660116
##  9:    Comp.9  0.9902798
## 10:   Comp.10  0.9995670
## 11:   Comp.11  1.0000000
##
## $plot
```




```
cumsumplot <- cumplot(pca.cor)

cumsumplot$cumvar
```

```
##     Component Proportion
##        <char>      <num>
##  1:    Comp.1  0.3972695
##  2:    Comp.2  0.6353338
##  3:    Comp.3  0.7228882
##  4:    Comp.4  0.8031322
##  5:    Comp.5  0.8541653
##  6:    Comp.6  0.8989206
##  7:    Comp.7  0.9381192
##  8:    Comp.8  0.9660116
##  9:    Comp.9  0.9902798
## 10:   Comp.10  0.9995670
## 11:   Comp.11  1.0000000
```

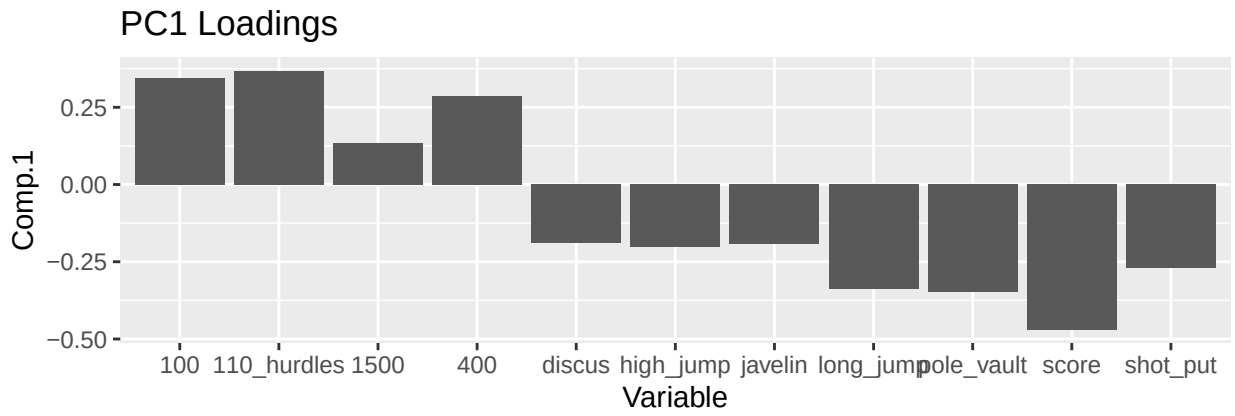
11. How many principal components do you need to explain at least 80% of the variance?

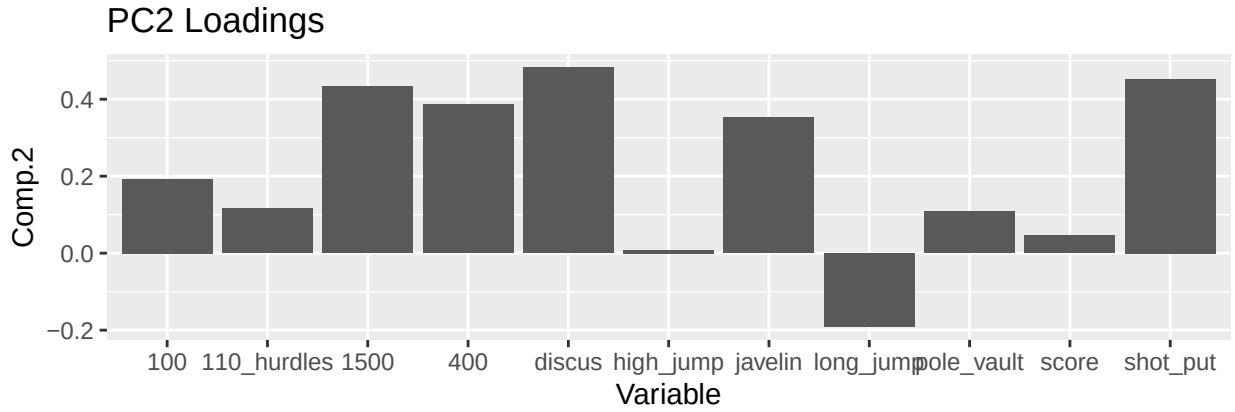
We need 4 pricinple components to explain at least 80% of the variance

12. Plot the loadings in the first 2 PCS.

```
loadplot.cor <- loadingsplot(pca.cor)
loadplot.cor
```

```
## $loadings
##        Variable     Comp.1       Comp.2      Comp.3      Comp.4        Comp.5
##          <char>      <num>        <num>       <num>       <num>         <num>
##  1:         100  0.3460636  0.192923628  0.31492797  0.06878888  0.4456563466
##  2:   long_jump -0.3373220 -0.191685942  0.15211029  0.19846521 -0.4065491285
##  3:    shot_put -0.2701354  0.452323529 -0.11540092  0.10680097  0.0003999519
##  4:   high_jump -0.2027541  0.008517787  0.76980490 -0.50885656 -0.0235860295
##  5:         400  0.2874165  0.387101249  0.19546693 -0.09795057 -0.1479048952
##  6: 110_hurdles  0.3684487  0.115769372  0.21187052  0.37742516  0.1003678039
##  7:      discus -0.1903842  0.481851078 -0.07043605 -0.02767192 -0.0127580630
##  8:  pole_vault -0.3472486  0.108407587 -0.18039913 -0.13391284  0.6863121003
##  9:     javelin -0.1937305  0.352748439  0.23652830  0.55509335 -0.1357207377
## 10:        1500  0.1346569  0.433087312 -0.28166804 -0.45135559 -0.3285180614
## 11:       score -0.4706560  0.046788607  0.12602317  0.05427017  0.0779160411
##          Comp.6      Comp.7       Comp.8       Comp.9     Comp.10    Comp.11
##           <num>       <num>        <num>        <num>       <num>      <num>
##  1:  0.01589632  0.25761105  0.663258581  0.108242546  0.10391563  0.1070342
##  2: -0.10213962  0.75278911  0.141693426 -0.046021107  0.06301135 -0.1355464
##  3:  0.22193043 -0.10030758  0.074276366 -0.422143929  0.65996141 -0.1382753
##  4:  0.06291675 -0.12432923 -0.153392071  0.102450312  0.13107035 -0.1900169
##  5: -0.31750177  0.13007380 -0.148502404 -0.651040299 -0.34473985  0.1240496
##  6:  0.20872024  0.27207383 -0.639839802  0.207127424  0.25251111  0.1406540
##  7:  0.60736272  0.15656258  0.010675200  0.167463296 -0.52554991 -0.1670979
##  8: -0.38280339  0.28387284 -0.275171111  0.018010045 -0.05429301 -0.2001298
##  9: -0.44263849 -0.33935655  0.060114387  0.306482634 -0.12059764 -0.1798939
## 10: -0.28087826  0.17112064  0.005216954  0.456516463  0.23212532  0.1811864
## 11:  0.03850092 -0.03068542 -0.006551930 -0.001265576 -0.04815391  0.8640651
##
## $plot
```






```
loadplot.cor$loadings
```

```
##        Variable     Comp.1       Comp.2      Comp.3      Comp.4        Comp.5
##          <char>      <num>        <num>       <num>       <num>         <num>
##  1:         100  0.3460636  0.192923628  0.31492797  0.06878888  0.4456563466
##  2:   long_jump -0.3373220 -0.191685942  0.15211029  0.19846521 -0.4065491285
##  3:    shot_put -0.2701354  0.452323529 -0.11540092  0.10680097  0.0003999519
##  4:   high_jump -0.2027541  0.008517787  0.76980490 -0.50885656 -0.0235860295
##  5:         400  0.2874165  0.387101249  0.19546693 -0.09795057 -0.1479048952
##  6: 110_hurdles  0.3684487  0.115769372  0.21187052  0.37742516  0.1003678039
##  7:      discus -0.1903842  0.481851078 -0.07043605 -0.02767192 -0.0127580630
##  8:  pole_vault -0.3472486  0.108407587 -0.18039913 -0.13391284  0.6863121003
##  9:     javelin -0.1937305  0.352748439  0.23652830  0.55509335 -0.1357207377
## 10:        1500  0.1346569  0.433087312 -0.28166804 -0.45135559 -0.3285180614
## 11:       score -0.4706560  0.046788607  0.12602317  0.05427017  0.0779160411
##          Comp.6      Comp.7       Comp.8       Comp.9      Comp.10     Comp.11
##           <num>       <num>        <num>        <num>       <num>       <num>
##  1:  0.01589632  0.25761105  0.663258581  0.108242546  0.10391563  0.1070342
##  2: -0.10213962  0.75278911  0.141693426 -0.046021107  0.06301135 -0.1355464
##  3:  0.22193043 -0.10030758  0.074276366 -0.422143929  0.65996141 -0.1382753
##  4:  0.06291675 -0.12432923 -0.153392071  0.102450312  0.13107035 -0.1900169
##  5: -0.31750177  0.13007380 -0.148502404 -0.651040299 -0.34473985  0.1240496
##  6:  0.20872024  0.27207383 -0.639839802  0.207127424  0.25251111  0.1406540
##  7:  0.60736272  0.15656258  0.010675200  0.167463296 -0.52554991 -0.1670979
##  8: -0.38280339  0.28387284 -0.275171111  0.018010045 -0.05429301 -0.2001298
```

```
##  9: -0.44263849 -0.33935655  0.060114387  0.306482634 -0.12059764 -0.1798939
## 10: -0.28087826  0.17112064  0.005216954  0.456516463  0.23212532  0.1811864
## 11:  0.03850092 -0.03068542 -0.006551930 -0.001265576 -0.04815391  0.8640651
```

13. Which 5 variables explain most of the variability in the first PC and how are they related to each other?

Score, 110 hurdles, pole_vault, 100, and long_jump. They are related to each other because they generally require good 100 meter speed. For the 100 and 110 hurdles this is straight forward to explain, and for the pole_vaule and long_jump, atheletes need to be able to run quickly towards their jump. The score relates to everything because the higher (or lower in the case of running time) something is the higher the score is.

14. Which 5 variables explain most of the variability in the second PC and how are they related to each other?

For the second PC, the 1500, discus, shotput, 400 and javalin were the most important. In this list there are three throwing events, and the two longest distance running events.

15. Based on all of the analyses above, which events do you think are most important to an athlete's decathlon score?

I think that the 100, long jupm, pole vault and maybe the hurdles are the most important to an athlete's score, as they are the most important in the first PCA.