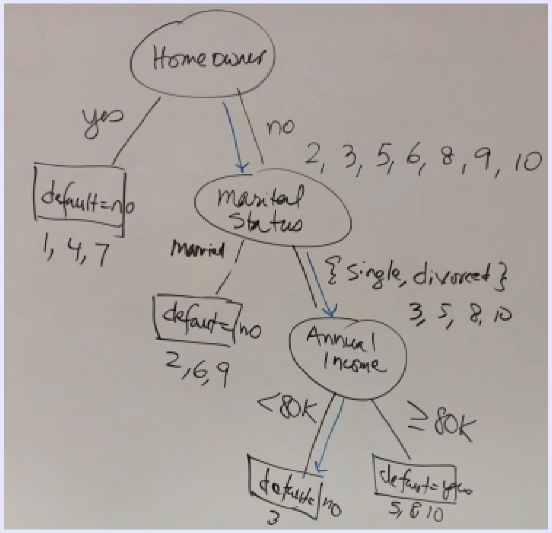
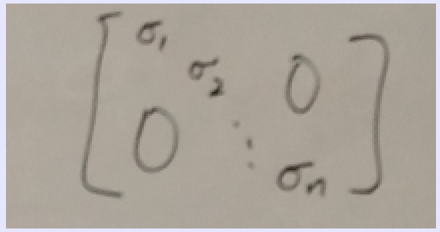
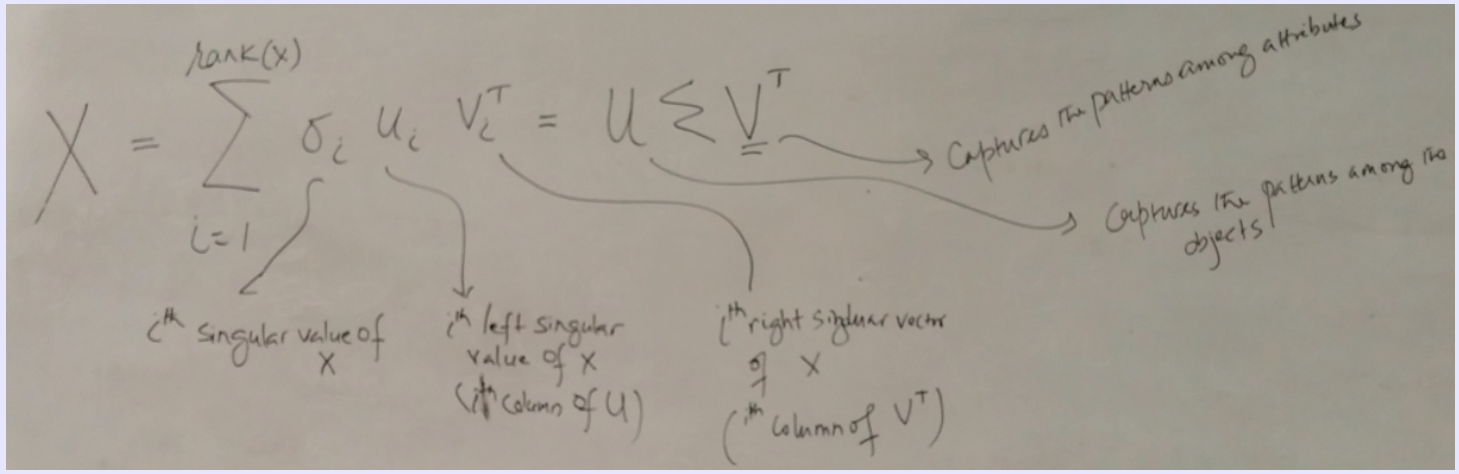
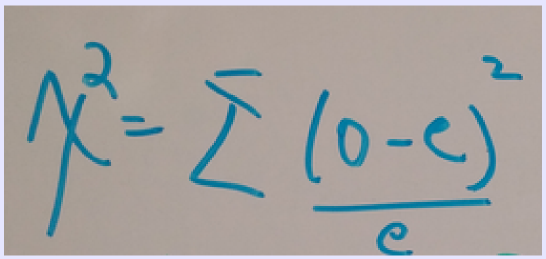
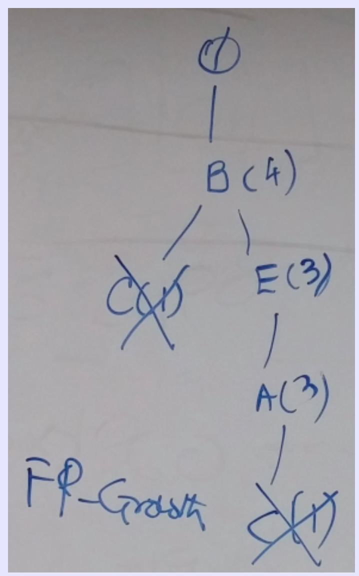
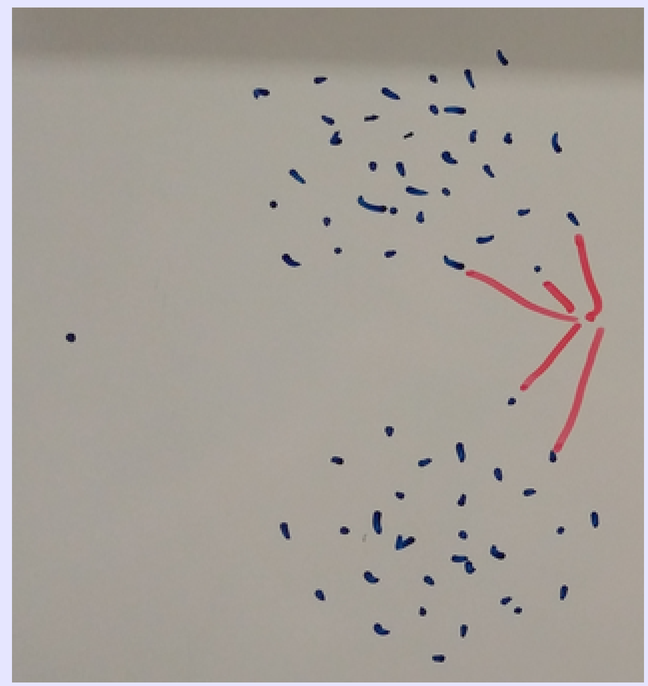












CS 422: Data Mining
Vijay K. Gurbani, Ph.D.,
Illinois Institute of Technology

Principal Component Analysis (PCA)

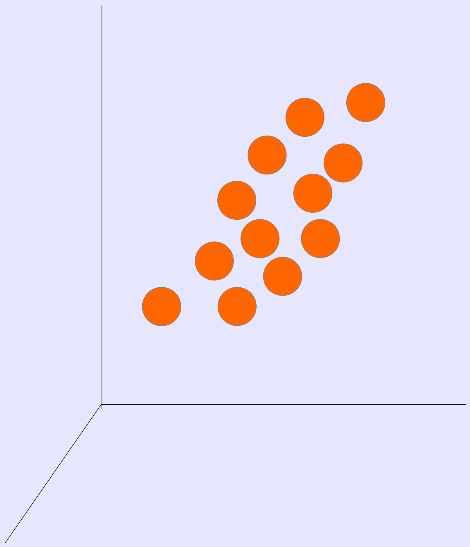
Principal Component Analysis (PCA)

- PCA is a dimensionality reduction technique.
- Big idea 1: Take a dataset in high dimension space and transform it so it can be represented in low dimension space, with minimal or no loss of information.
- Big idea 2: Extract *latent* information from the data.
- The transformation results in a smaller number of *principal components* that maximizes the variation of the original dataset, but in low dimension space.
- These principal components are linear combinations of the original variables, and become the new axes of the dataset in low dimension space.

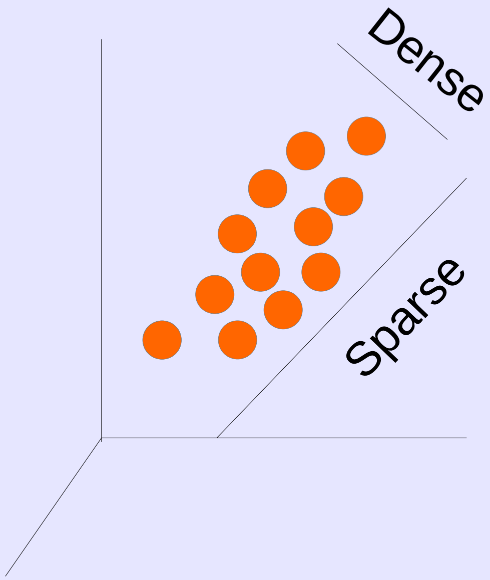
Principal Component Analysis (PCA)

- PCA has three goals:
  - Feature reduction: Reduce the number of features used to represent the data.
  - The reduced feature set should explain a large amount of information (or maximize variance).
  - Make visible the latent information in the data.
- Let’s explore what the first two means visually, and then we get into the (linear) algebra of it.

Principal Component Analysis (PCA)



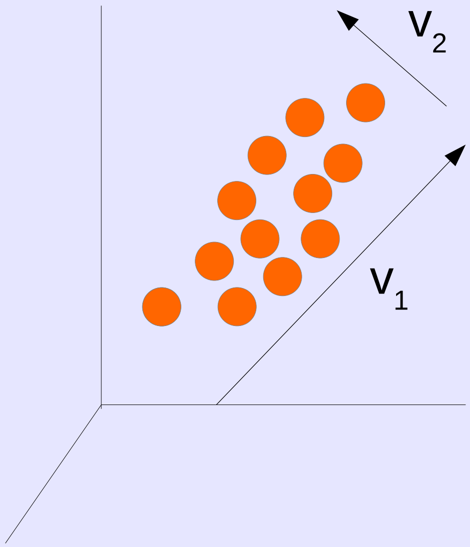
Principal Component Analysis (PCA)



PCA creates projections (principal components) in the direction that captures most of the variance.

- Sparser data has greater variance (spread out).
- Denser data has lesser variance (clustered together).

Principal Component Analysis (PCA)



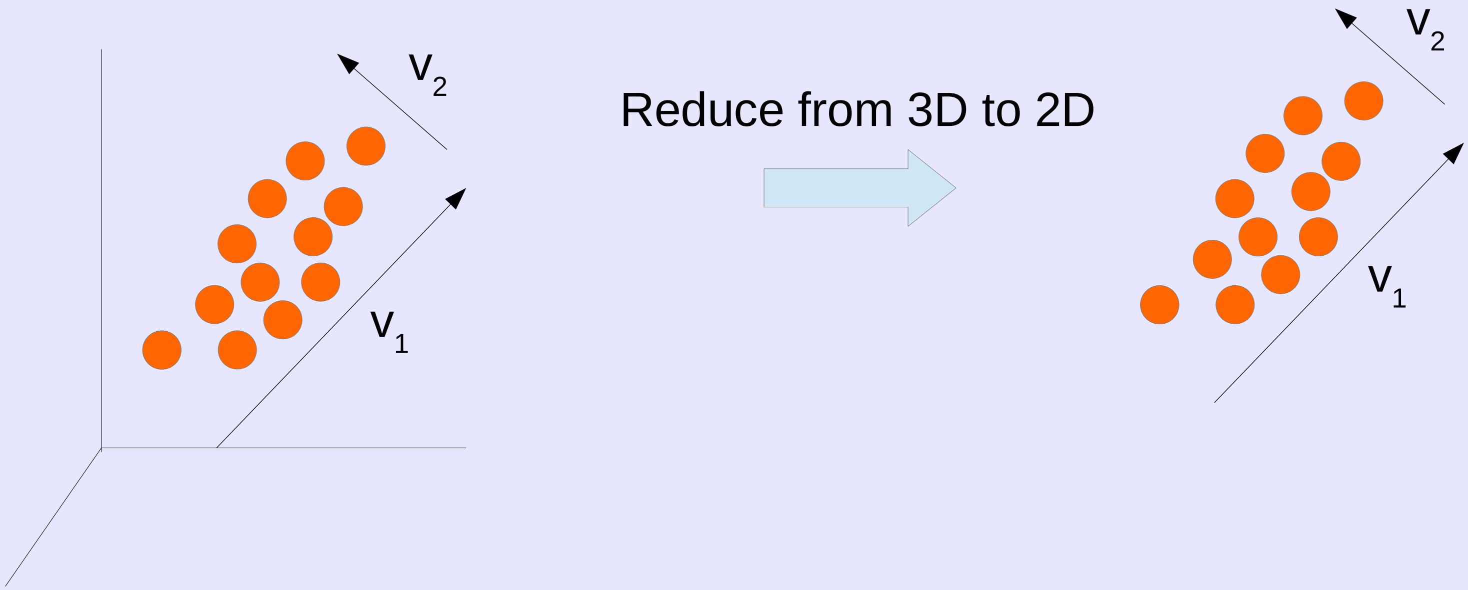
PCA creates projections (principal components) in the direction that captures most of the variance.

- Sparser data has greater variance (spread out).
- Denser data has lesser variance (clustered together).

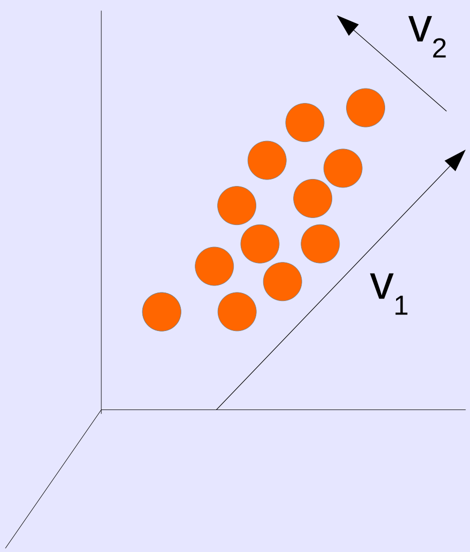
The projection v_1 maximizes the variance of the data. And the next projection, v_2, maximizes the remaining variance.

These projections are orthogonal to each other! And will always be so.

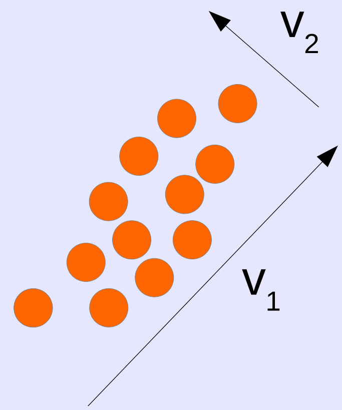
Principal Component Analysis (PCA)

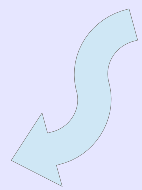


Principal Component Analysis (PCA)

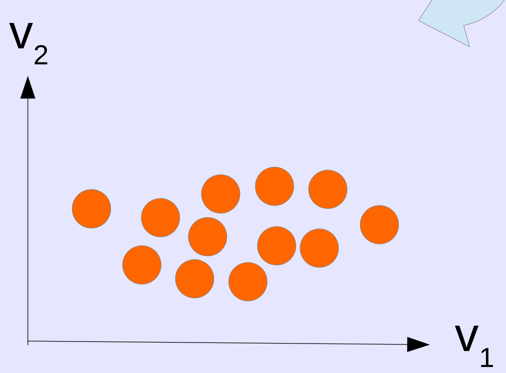


Reduce from 3D to 2D





And represent in new axes (rotated). These new axes are the reduced Features, v_1 and v_2 that explain most of the variance.

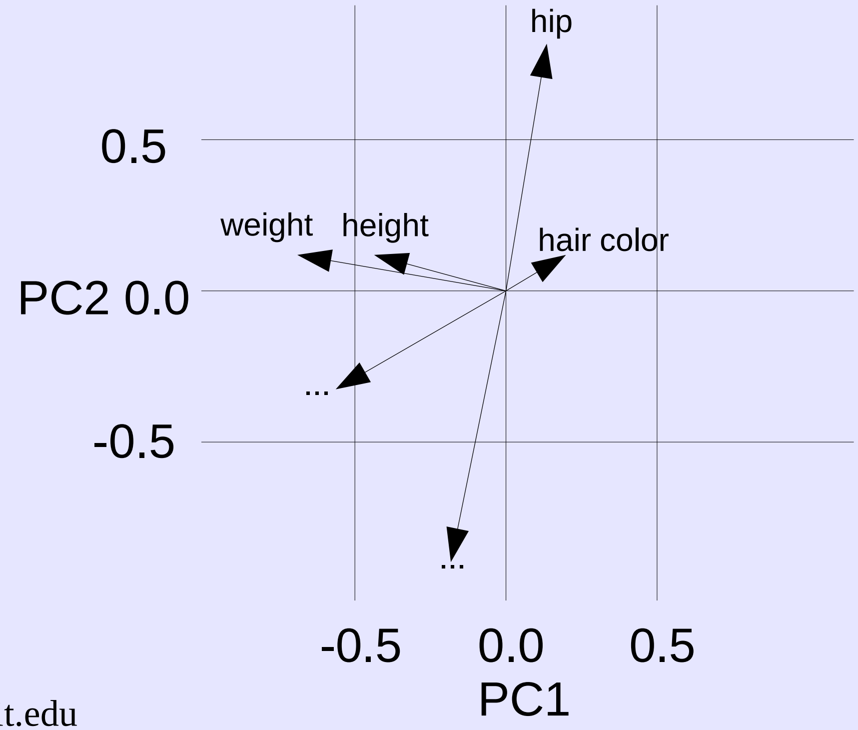


Principal Component Analysis (PCA)

- Motivating example
  - Dataset of multiple physical traits of people
    - Height, weight, arm length, leg length, hair color, waist circumference, hip circumference, chest circumference, …
  - Principal components could conceivably be:
    - Size
    - Gender

Principal Component Analysis (PCA)

- Motivating example
  - Dataset of multiple physical traits of people
    - Height, weight, arm length, leg length, hair color, waist circumference, hip circumference, chest circumference, …
    - Principal components:
      - Size
      - Gender



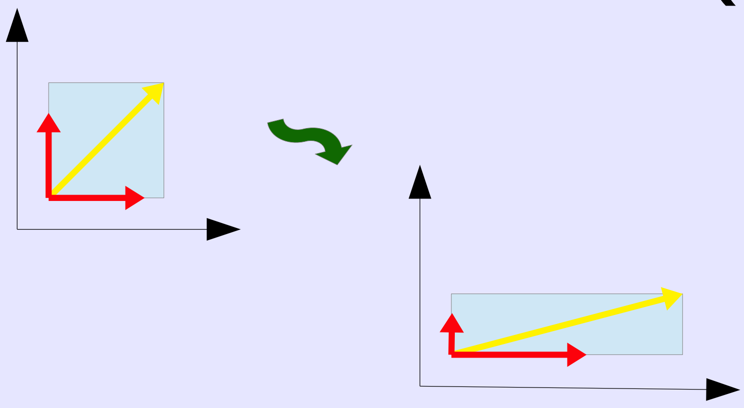
Principal Component Analysis (PCA)

- The mathematics: Eigenvalues and eignevectors.
- The eigenvalues and eigenvectors of a *mxn* matrix are the scalar values **λ** and vectors **x**, respectively, that are solutions to:

$$\mathbf{Ax = \lambda x}$$

- Eigenvectors are vectors that remain un-changed when multiplied by A, except for a change in magnitude.

Principal Component Analysis (PCA)



An eigenvector (red) is a vector whose direction remains unchanged when a linear transform is applied to it. Other vectors (yellow) change directions, and are not eigenvectors.

The transformation is a simple scaling with factor 2 in the x-axis and ½ in y-axis:

$$A = \begin{bmatrix} 2 & 0 \\ 0 & 0.5 \end{bmatrix}.$$

In general, the eigenvector $\vec{v}$ of a matrix A is the vector for which the following holds:

$$A\vec{v} = \lambda\vec{v},$$

where λ is a scalar value called the eigenvalue.

Principal Component Analysis (PCA)

- A square matrix A of rank N (N linearly independent columns) can be factorized as:
  - $A = X \Sigma X^{-1}$
- where X is a NxN matrix whose i^{th} column is the **eigenvector** x_i of A,

  Σ is a diagonal matrix whose diagonal elements are the corresponding **eigenvalues**, i.e., $\Sigma_{ii} = \lambda_i$.
- (See slides in backup section for manual eigen-decomposition of a matrix into its constituents.)

Principal Component Analysis (PCA)

- Our matrix/dataset (A) gets decomposed into:
  - Eigenvectors
  - Eigenvalues
- In R, the command for PCA is: prcomp()
  - Should we standardize A ($\mu = 0$, $\sigma = 1$)?

Principal Component Analysis (PCA)

- Our matrix/dataset (A) gets decomposed into:
  - Eigenvectors
  - Eigenvalues
- In R, the command for PCA is: prcomp()
  - Should we standardize A ($\mu = 0$, $\sigma = 1$)?
  - Yes: prcomp(A, scale.=T)

Principal Component Analysis (PCA)

- USArrests dataset

```
> data("USArrests")
> head(USArrests)
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Alaska       10.0     263       48 44.5
Arizona       8.1     294       80 31.0
Arkansas      8.8     190       50 19.5
California    9.0     276       91 40.6
Colorado      7.9     204       78 38.7
```

USArrests datasets contains statistics, in arrests per 100,000 residents for assault murder and rape in all 50 states in 1973. Also provided is the percentage of population in each state living in an urban center.

- Perform PCA on it

```
> options(digits=3)
> p <- prcomp(USArrests, scale. = T)
> p
Standard deviations (1, .., p=4):
[1] 1.575 0.995 0.597 0.416

Rotation (n x k) = (4 x 4):
            PC1    PC2    PC3    PC4
Murder   -0.536  0.418 -0.341  0.649
Assault  -0.583  0.188 -0.268 -0.743
UrbanPop -0.278 -0.873 -0.378  0.134
Rape     -0.543 -0.167  0.818  0.089
```

- OK, so where are the eigenvalues? the eigenvectors? the data in rotated space?

Principal Component Analysis (PCA)

- The object returned from prcomp(A, ...) has five fields:
  - sdev: Square root of the eigen-values, ordered from largest eigenvalue to the smallest.
  - rotation: Matrix whose columns contain the eigenvectors. (Also called principal loadings.)
  - center: Mean of the columns of A.
  - scale: Std. dev of the columns of A.
  - x: Data from A in rotated space. (Also called principal component scores)

```
> options(digits=3)
> p <- prcomp(USArrests, scale. = T)
> p
Standard deviations (1, .., p=4):
[1] 1.575 0.995 0.597 0.416

Rotation (n x k) = (4 x 4):
            PC1    PC2    PC3    PC4
Murder   -0.536  0.418 -0.341  0.649
Assault  -0.583  0.188 -0.268 -0.743
UrbanPop -0.278 -0.873 -0.378  0.134
Rape     -0.543 -0.167  0.818  0.089
```

Principal Component Analysis (PCA)

- What is the rotation matrix telling us?

```
> p$rotation <- -p$rotation
> p$rotation
               PC1        PC2        PC3         PC4
Murder   0.5358995 -0.4181809  0.3412327 -0.64922780
Assault  0.5831836 -0.1879856  0.2681484  0.74340748
UrbanPop 0.2781909  0.8728062  0.3780158 -0.13387773
Rape     0.5434321  0.1673186 -0.8177779 -0.08902432
```

- And its impact on the data in rotated space?

```
> head(USArrests)
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Alaska       10.0     263       48 44.5
Arizona       8.1     294       80 31.0
Arkansas      8.8     190       50 19.5
California    9.0     276       91 40.6
Colorado      7.9     204       78 38.7
```

Original data

```
> p$x <- -p$x
> head(p$x)
                  PC1        PC2         PC3          PC4
Alabama     0.9756604 -1.1220012  0.43980366 -0.154696581
Alaska      1.9305379 -1.0624269 -2.01950027  0.434175454
Arizona     1.7454429  0.7384595 -0.05423025  0.826264240
Arkansas   -0.1399989 -1.1085423 -0.11342217  0.180973554
California  2.4986128  1.5274267 -0.59254100  0.338559240
Colorado    1.4993407  0.9776297 -1.08400162 -0.001450164
```

Data in rotated space (will come back to this)

Principal Component Analysis (PCA)

- How is the data in rotated space computed?
  - Dot product.
- Given two vectors, the dot product is the sum of the products of the individual vector elements.

Mathematical definition of dot product of vectors x and y $= \sum_{i=1}^{n} x_i y_i$

In linear algebra, if x and y are vectors, dot product $= x^T y$

- In R:

```
> x <- c(1, 9, 8, 3)
> y <- c(0, 1, 2, 4)
> x %*% y
     [,1]
[1,]   37
```

Principal Component Analysis (PCA)

- How is the data in rotated space computed?
  - Dot product

```
> head(scale(USArrests, center=T, scale=T))
           Murder Assault UrbanPop     Rape
Alabama    1.2426   0.783   -0.521 -0.00342
Alaska     0.5079   1.107   -1.212  2.48420
Arizona    0.0716   1.479    0.999  1.04288
Arkansas   0.2323   0.231   -1.074 -0.18492
California 0.2783   1.263    1.759  2.06782
Colorado   0.0257   0.399    0.861  1.86497
```

Scaled dataset

```
> p$x <- -p$x
> head(p$x)
                  PC1        PC2         PC3          PC4
Alabama     0.9756604 -1.1220012  0.43980366 -0.154696581
Alaska      1.9305379 -1.0624269 -2.01950027  0.434175454
Arizona     1.7454429  0.7384595 -0.05423025  0.826264240
Arkansas   -0.1399989 -1.1085423 -0.11342217  0.180973554
California  2.4986128  1.5274267 -0.59254100  0.338559240
Colorado    1.4993407  0.9776297 -1.08400162 -0.001450164
```

Rotated dataset

```
> p$rotation <- -p$rotation
> p$rotation
               PC1        PC2        PC3         PC4
Murder   0.5358995 -0.4181809  0.3412327 -0.64922780
Assault  0.5831836 -0.1879856  0.2681484  0.74340748
UrbanPop 0.2781909  0.8728062  0.3780158 -0.13387773
Rape     0.5434321  0.1673186 -0.8177779 -0.08902432
```

Rotation matrix (Eigenvectors)

Principal Component Analysis (PCA)

- How is the data in rotated space computed?
  - Dot product

```
> head(scale(USArrests, center=T, scale=T))
           Murder Assault UrbanPop     Rape
Alabama    1.2426   0.783   -0.521 -0.00342
Alaska     0.5079   1.107   -1.212  2.48420
Arizona    0.0716   1.479    0.999  1.04288
Arkansas   0.2323   0.231   -1.074 -0.18492
California 0.2783   1.263    1.759  2.06782
Colorado   0.0257   0.399    0.861  1.86497
```

Scaled dataset

```
> p$x <- -p$x
> head(p$x)
                 PC1        PC2         PC3          PC4
Alabama    0.9756604 -1.1220012  0.43980366 -0.154696581
Alaska     1.9305379 -1.0624269 -2.01950027  0.434175454
Arizona    1.7454429  0.7384595 -0.05423025  0.826264240
Arkansas  -0.1399989 -1.1085423 -0.11342217  0.180973554
California 2.4986128  1.5274267 -0.59254100  0.338559240
Colorado   1.4993407  0.9776297 -1.08400162 -0.001450164
```

Rotated dataset

```
> p$rotation <- -p$rotation
> p$rotation
               PC1        PC2        PC3         PC4
Murder   0.5358995 -0.4181809  0.3412327 -0.64922780
Assault  0.5831836 -0.1879856  0.2681484  0.74340748
UrbanPop 0.2781909  0.8728062  0.3780158 -0.13387773
Rape     0.5434321  0.1673186 -0.8177779 -0.08902432
```

Rotation matrix (Eigenvectors)

Dot Product
(in R): data.scaled[1,] %*% p$rotation[, 1]

Principal Component Analysis (PCA)

```
> summary(USArrests)
     Murder          Assault         UrbanPop          Rape      
 Min.   : 0.800   Min.   : 45.0   Min.   :32.00   Min.   : 7.30  
 1st Qu.: 4.075   1st Qu.:109.0   1st Qu.:54.50   1st Qu.:15.07  
 Median : 7.250   Median :159.0   Median :66.00   Median :20.10  
 Mean   : 7.788   Mean   :170.8   Mean   :65.54   Mean   :21.23  
 3rd Qu.:11.250   3rd Qu.:249.0   3rd Qu.:77.75   3rd Qu.:26.18  
 Max.   :17.400   Max.   :337.0   Max.   :91.00   Max.   :46.00  
```

```
> USArrests[c(9,10,24,29,34,45), ]
              Murder Assault UrbanPop Rape
Florida         15.4     335       80 31.9
Georgia         17.4     211       60 25.8
Mississippi     16.1     259       44 17.1
New Hampshire    2.1      57       56  9.5
North Dakota     0.8      45       44  7.3
Vermont          2.2      48       32 11.2
```

Raw data: FL, GA, MS high in murder, assault, and rape.
NH, ND, VT low in murder, assault, and rape.

```
> p$rotation <- -p$rotation
> p$rotation
               PC1        PC2        PC3         PC4
Murder   0.5358995 -0.4181809  0.3412327 -0.64922780
Assault  0.5831836 -0.1879856  0.2681484  0.74340748
UrbanPop 0.2781909  0.8728062  0.3780158 -0.13387773
Rape     0.5434321  0.1673186 -0.8177779 -0.08902432
```

```
> pca$x <- -pca$x
> pca$x[c(9,10,24), ]
               PC1      PC2    PC3      PC4
Florida     2.9828 -0.03883 0.5710  0.09532
Georgia     1.6228 -1.26609 0.3390 -1.06597
Mississippi 0.9865 -2.36974 0.7334 -0.21334
>
> pca$x[c(29,34,45), ]
                 PC1     PC2      PC3    PC4
New Hampshire -2.360  0.0179 -0.03648 0.0328
North Dakota  -2.962 -0.5931 -0.29825 0.2514
Vermont       -2.773 -1.3882 -0.83281 0.1434
```

In rotated space,
- FL, GA, MS are positively correlated to PC1 (which explains murder, assault, rape).
- NH, ND, VT are negatively correlated to PC1.

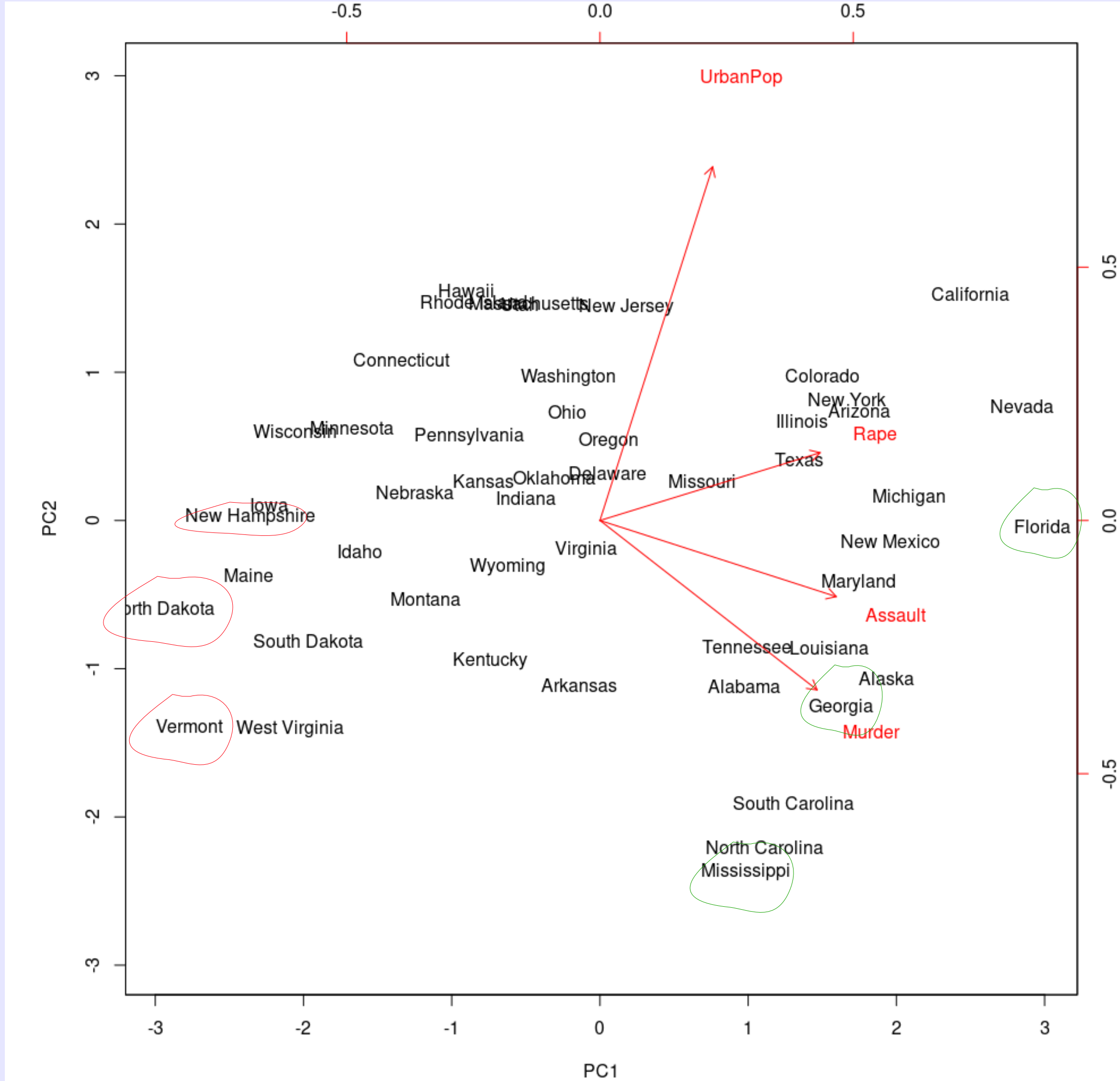
Principal Component Analysis (PCA)

- To plot the first two principal components:

```
> biplot(p, scale=0)
```

- To plot other principal components:

```
> biplot(pca, choices=c(1,3))
```



Principal Component Analysis (PCA)

- How many principal components do we need?
- As many that explain most of the variance, and adding any more to the model results in diminishing gains in variance.
- Key idea: What is the proportion of variance contributed by each principal component loading?

$$\text{TotalVariation} = \sum_{i=1}^{p} PC_i$$

Proportion of variance explained by first principal component loading $= \dfrac{PC_1}{TotalVariation}$,

Proportion of variance explained by second principal component loading $= \dfrac{PC_2}{TotalVariation}$ and so on...

Principal Component Analysis (PCA)

$$\text{TotalVariation} = \sum_{i=1}^{p} PC_i$$

Proportion of variance explained by first principal component loading $= \dfrac{PC_1}{TotalVariation}$,

Proportion of variance explained by second principal component loading $= \dfrac{PC_2}{TotalVariation}$ and so on...

The eigenvalues indicate variance being explained.
∑e$values = 4.0
The first eigenvalue explains 2.4802/4.0 = 0.62 of the variance.
The second explains 0.9898/4.0 = 0.247 of the variance, and so on…

Or, examine your PCA object in R:

```
> summary(pca)
Importance of components:
                        PC1   PC2    PC3    PC4
Standard deviation     1.57 0.995 0.5971 0.4164
Proportion of Variance 0.62 0.247 0.0891 0.0434
Cumulative Proportion  0.62 0.868 0.9566 1.0000
```

Manual eigen-decomposition

```
> data("USArrests")
>
> X <- scale(USArrests)
>
> head(X)
               Murder   Assault   UrbanPop         Rape
Alabama    1.24256408 0.7828393 -0.5209066 -0.003416473
Alaska     0.50786248 1.1068225 -1.2117642  2.484202941
Arizona    0.07163341 1.4788032  0.9989801  1.042878388
Arkansas   0.23234938 0.2308680 -1.0735927 -0.184916602
California 0.27826823 1.2628144  1.7589234  2.067820292
Colorado   0.02571456 0.3988593  0.8608085  1.864967207
>
> cov(X)
             Murder   Assault   UrbanPop      Rape
Murder   1.00000000 0.8018733 0.06957262 0.5635788
Assault  0.80187331 1.0000000 0.25887170 0.6652412
UrbanPop 0.06957262 0.2588717 1.00000000 0.4113412
Rape     0.56357883 0.6652412 0.41134124 1.0000000
>
> e <- eigen(cov(X))
> row.names(e$vectors) <- c("Murder", "Assault", "UrbanPop", "Rape")
> colnames(e$vectors) <- c("PC1", "PC2", "PC3", "PC4")
> e
eigen() decomposition
$values
[1] 2.4802416 0.9897652 0.3565632 0.1734301

$vectors
                PC1        PC2        PC3         PC4
Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
Rape     -0.5434321 -0.1673186  0.8177779  0.08902432

>
> phi <- -e$vectors
> phi
               PC1        PC2        PC3         PC4
Murder   0.5358995 -0.4181809  0.3412327 -0.64922780
Assault  0.5831836 -0.1879856  0.2681484  0.74340748
UrbanPop 0.2781909  0.8728062  0.3780158 -0.13387773
Rape     0.5434321  0.1673186 -0.8177779 -0.08902432
```

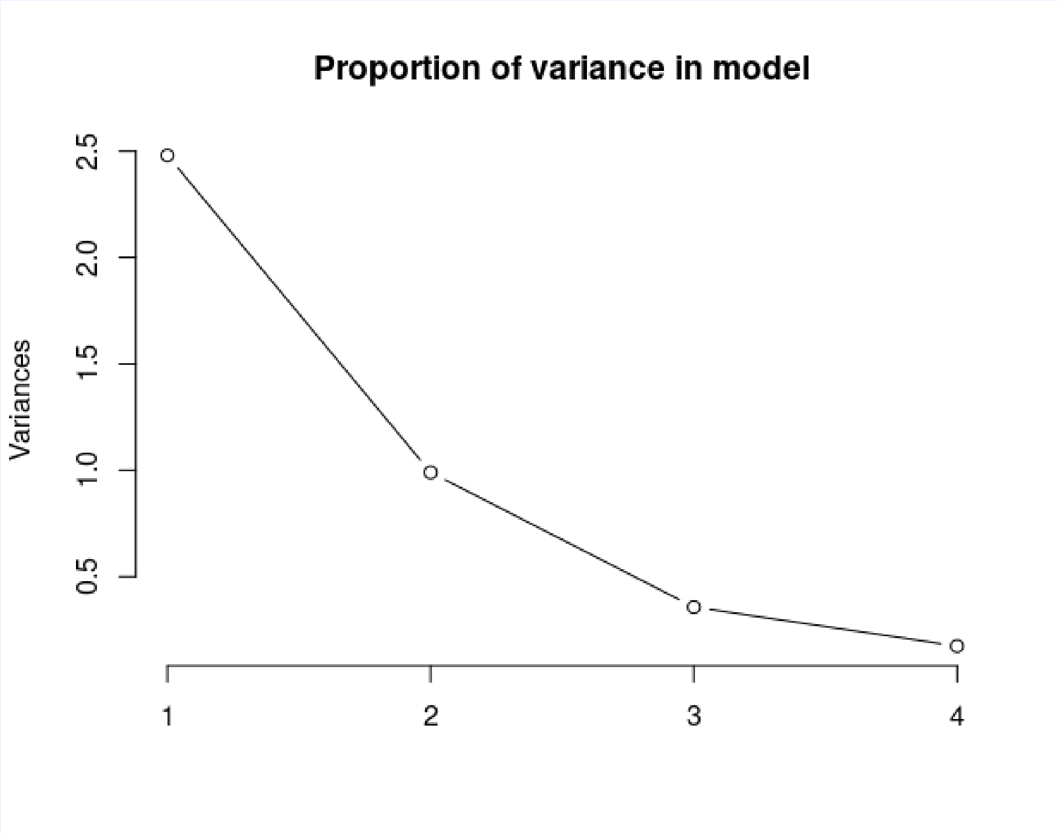
Principal Component Analysis (PCA)

Or, examine your PCA object in R:

```
> summary(pca)
Importance of components:
                         PC1   PC2    PC3    PC4
Standard deviation     1.57 0.995 0.5971 0.4164
Proportion of Variance 0.62 0.247 0.0891 0.0434
Cumulative Proportion  0.62 0.868 0.9566 1.0000
```

You can also create a “Scree” plot to visually show the proprtion of variance in the model:

```
> screeplot(pca, type='l', main="Proportion of variance in model")
```



Principal Component Analysis (PCA)

- PCA as dimensionality reduction technique:
  - You can use the principal component loadings that explain the highest proportion of variance as new attributes.
  - Because each principal component *score* is a linear combination of original observation and the principal component loading, you may end up with a smaller number of “attributes” that explain a lot of variance in the data.
    - In USArrests, the first two principal components explain about 87% of the variance, reducing the attributes from 4 to 2.
  - You will have to remember to transform **all** your observations (in sample, out of sample) from their natural representation to principal component scores before attempting to use them in any model.