Machine learning

- Unsupervised learning

- Supervised learning

Machine learning

- Unsupervised learning
  - dimension reduction, clustering
- Supervised learning

Machine learning

- Unsupervised learning
  - dimension reduction, clustering
- Supervised learning
  - classification, regression

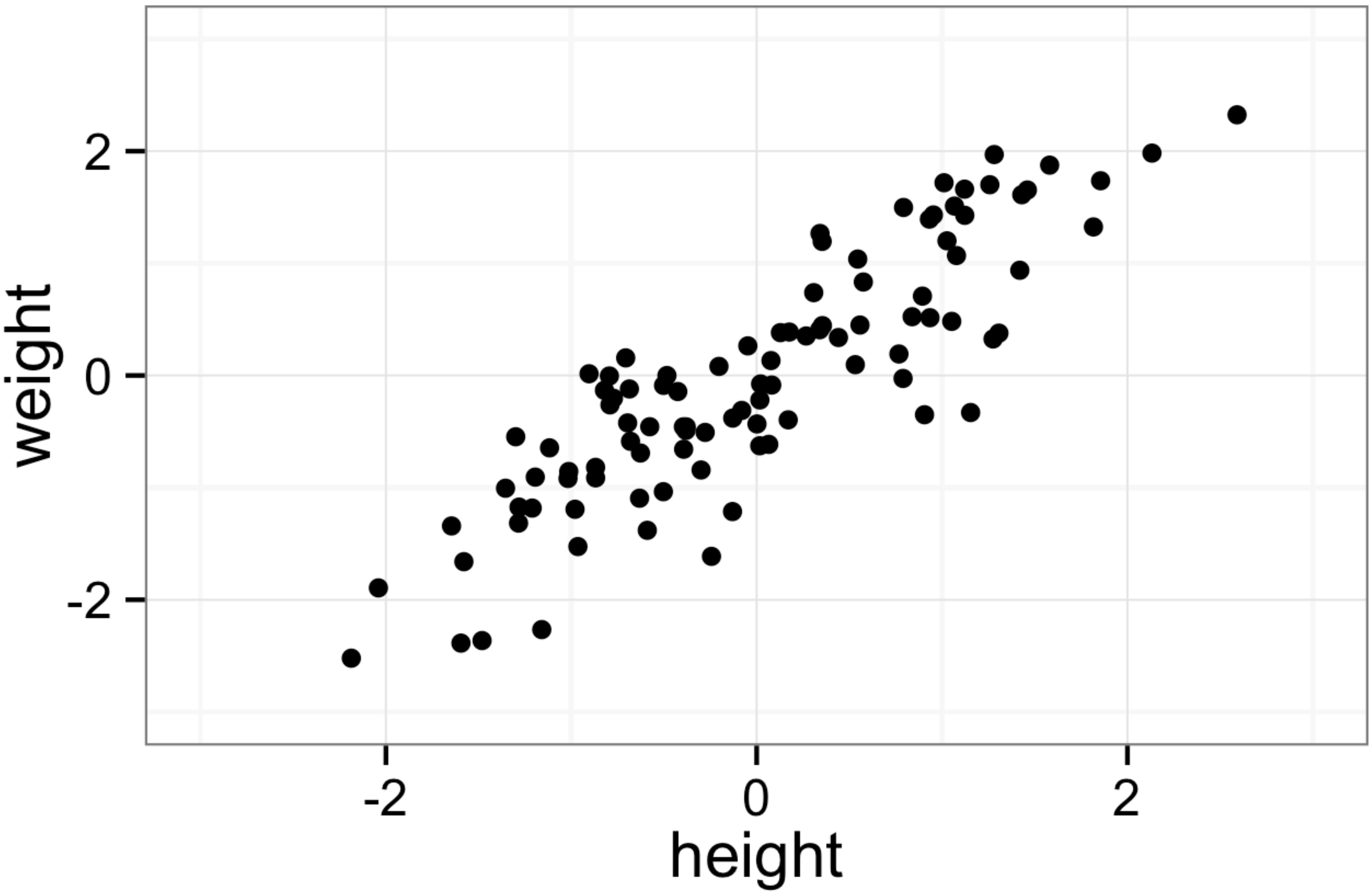
Machine learning

- Unsupervised learning
  - dimension reduction, clustering
- Supervised learning
  - classification, regression

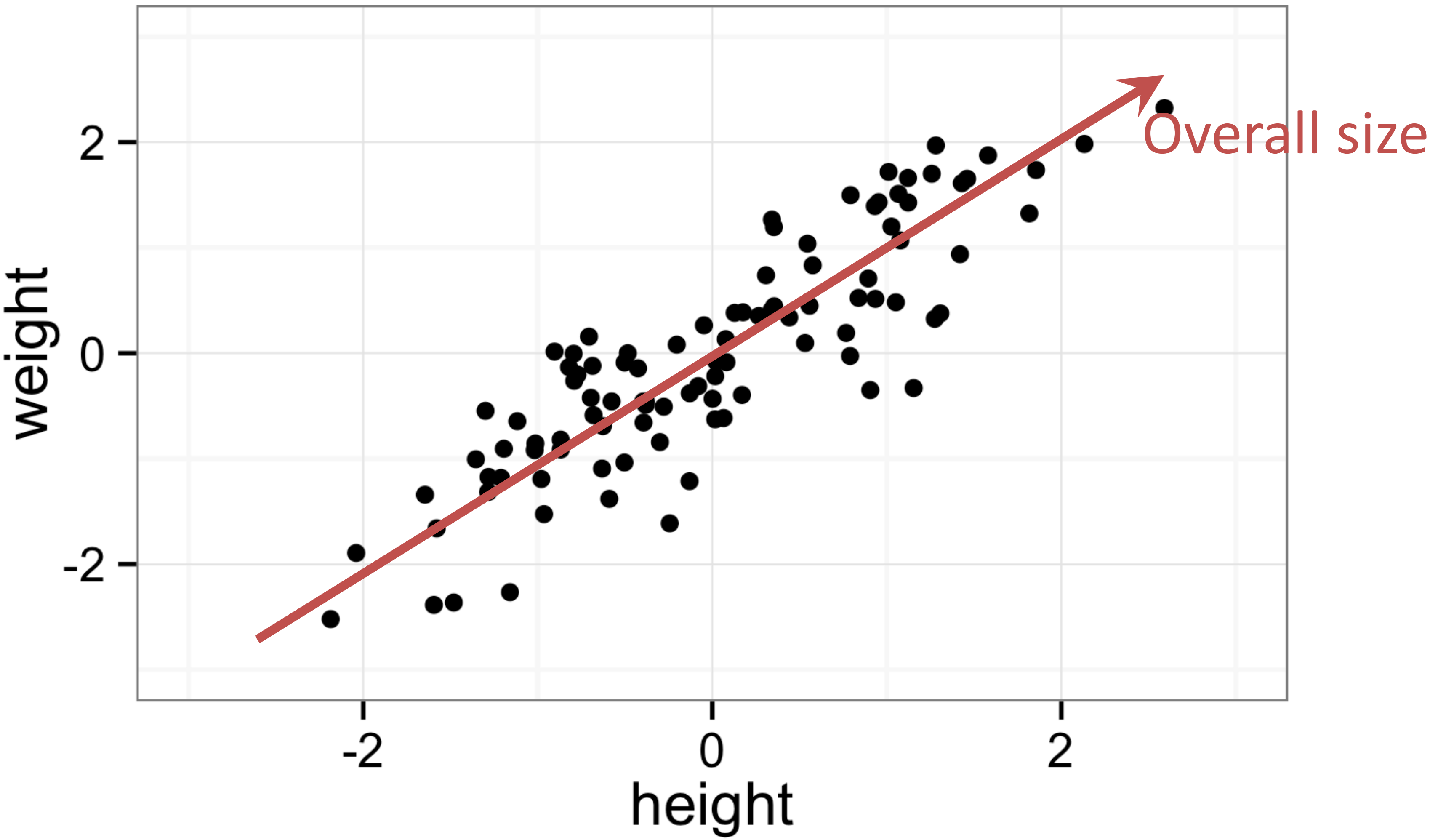
Principal Components Analysis (PCA)

- Dimension reduction
- Useful for exploratory data analysis of high-dimensional data sets.

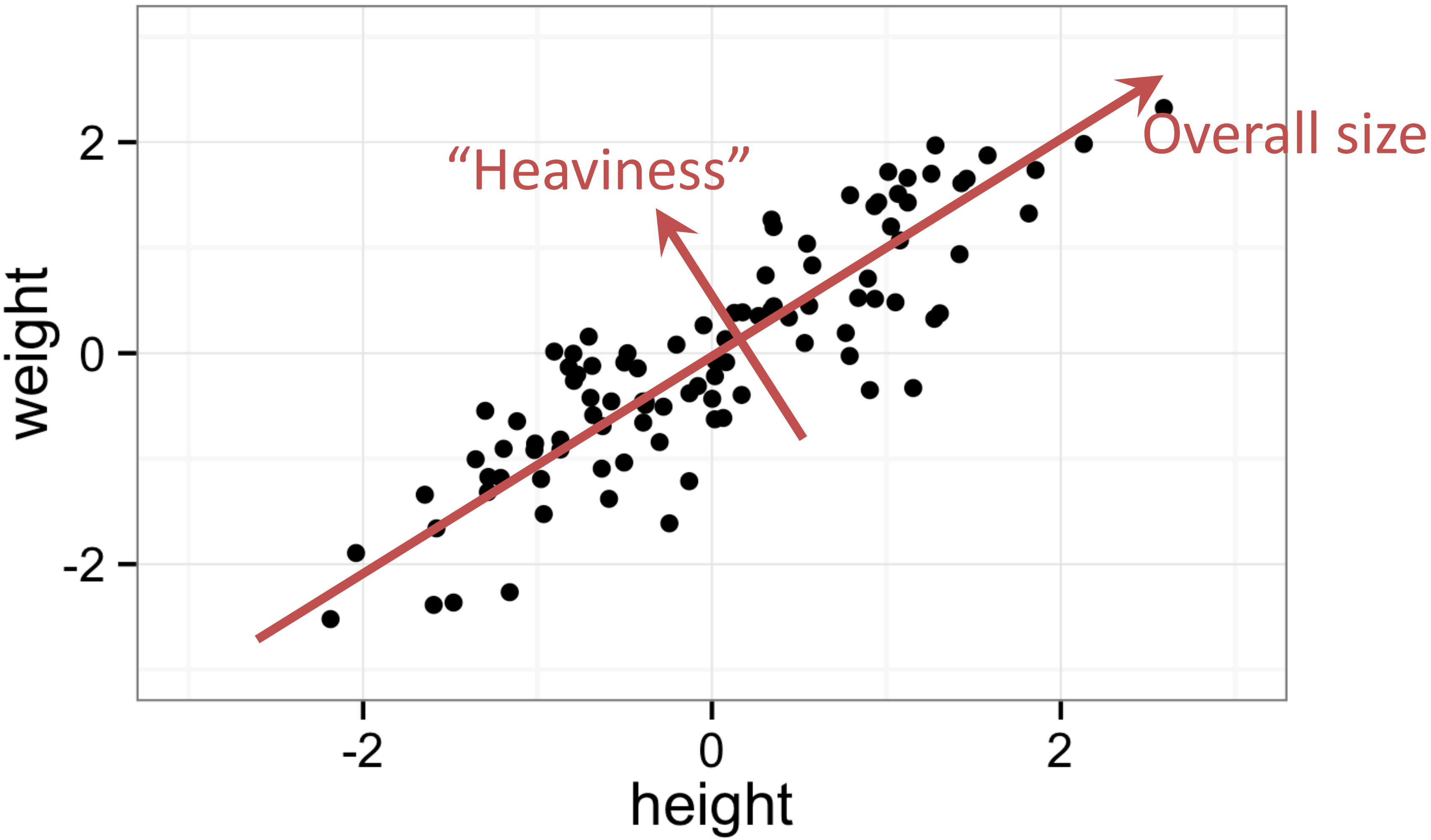
Example: Consider a data set of heights and weights of people



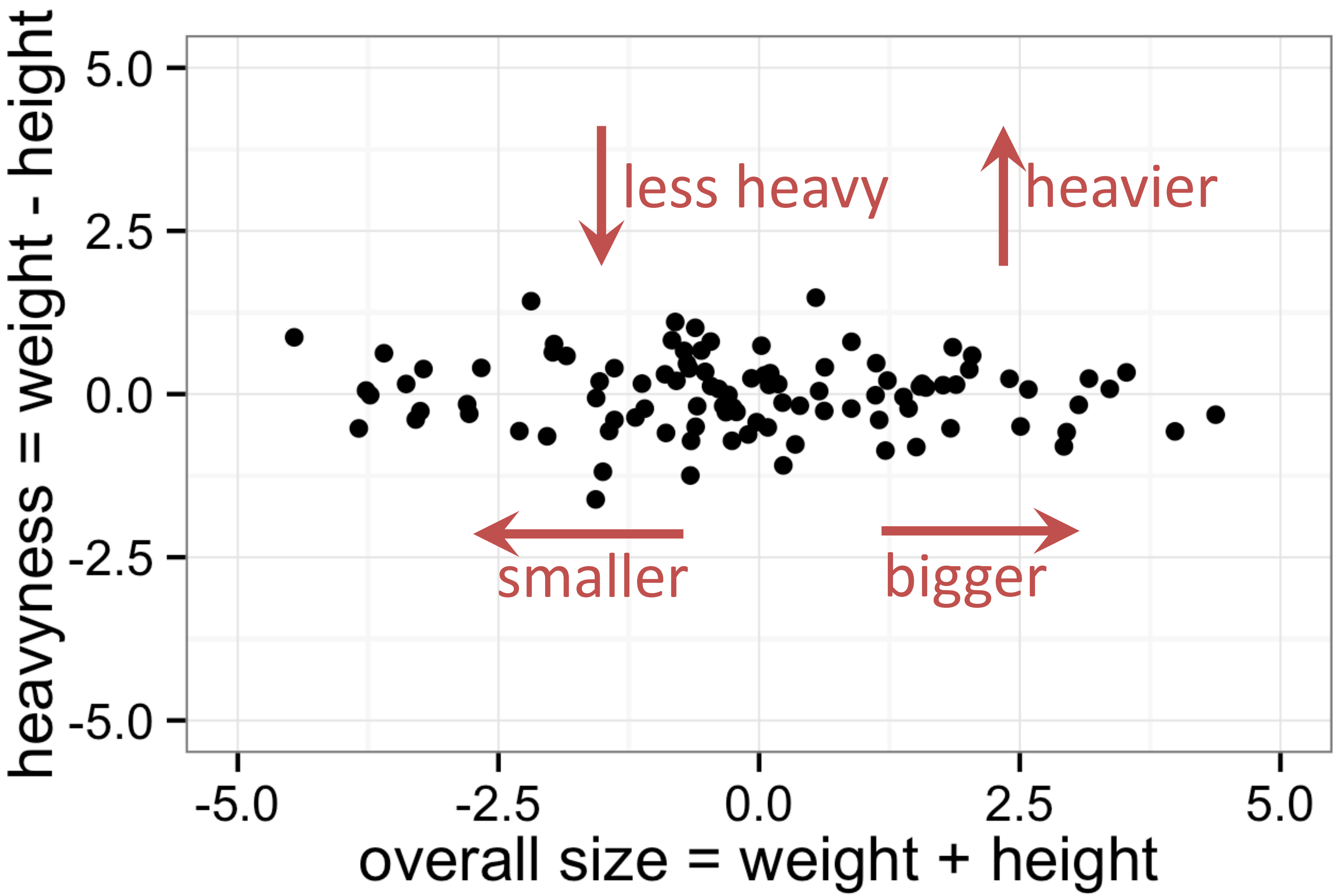
Example: Consider a data set of heights and weights of people



Example: Consider a data set of heights and weights of people



PCA on this data set reframes data in terms of overall size and heavyness



The math behind PCA

Variance of one variable:

$$\mathrm{Var}(X) = \frac{1}{n}\sum_{j}(\overline{x} - x_j)^2 = \sigma_X^2$$

Covariance of two variables:

$$\mathrm{Cov}(X,Y) = \frac{1}{n}\sum_{j}(\overline{x} - x_j)(\overline{y} - y_j) = \sigma_{XY}^2$$

The math behind PCA

Covariance matrix of n variables $X_1 \ldots X_n$:

$$\mathbf{C} = \begin{pmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1n}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1}^2 & \sigma_{n2}^2 & \cdots & \sigma_{nn}^2 \end{pmatrix}$$

The math behind PCA

PCA diagonalizes the covariance matrix **C**:

$$\mathbf{C} = \mathbf{U}\mathbf{D}\mathbf{U}^T$$

$$= \mathbf{U} \begin{pmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^2 \end{pmatrix} \mathbf{U}^T$$

The math behind PCA

PCA diagonalizes the covariance matrix **C**:

$$\mathbf{C} = \mathbf{U}\mathbf{D}\mathbf{U}^T$$

rotation matrix

$$= \mathbf{U}\begin{pmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^2 \end{pmatrix}\mathbf{U}^T$$

The math behind PCA

PCA diagonalizes the covariance matrix **C**:

$$\mathbf{C} = \mathbf{U}\mathbf{D}\mathbf{U}^T$$

diagonal matrix

$$= \mathbf{U}\begin{pmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^2 \end{pmatrix}\mathbf{U}^T$$

The math behind PCA

PCA diagonalizes the covariance matrix **C**:

$$\mathbf{C} = \mathbf{U}\,\mathbf{D}\,\mathbf{U}^T$$

$$= \mathbf{U}\begin{pmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^2 \end{pmatrix}\mathbf{U}^T$$

eigenvalues (= variance explained
by each component)

The math behind PCA

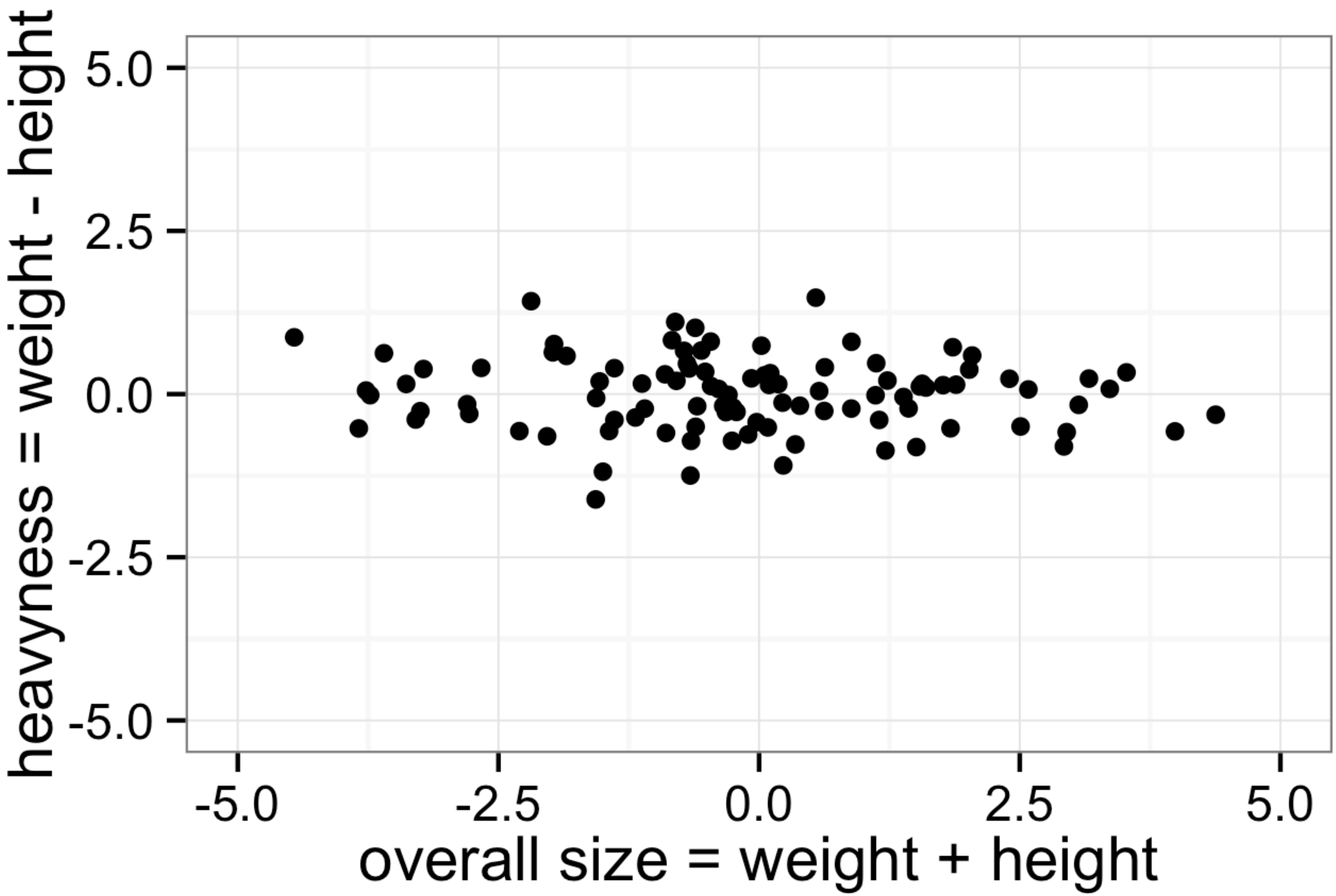
PCA diagonalizes the covariance matrix **C**:

$$\mathbf{C} = \mathbf{U}\,\mathbf{D}\,\mathbf{U}^T$$

$$= \mathbf{U}\begin{pmatrix} \lambda_1^2 & 0 & \cdots & 0 \\ 0 & \lambda_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^2 \end{pmatrix}\mathbf{U}^T$$

covariance between components
is zero (they are uncorrelated)

In our earlier example, overall size and heaviness are uncorrelated



Doing a PCA in R

```
iris %>%
    select(-Species) %>%    # remove Species column
    scale() %>%             # scale to zero mean
                            # and unit variance
    prcomp() ->             # do PCA
    pca                     # store result
                            # in variable “pca”
```

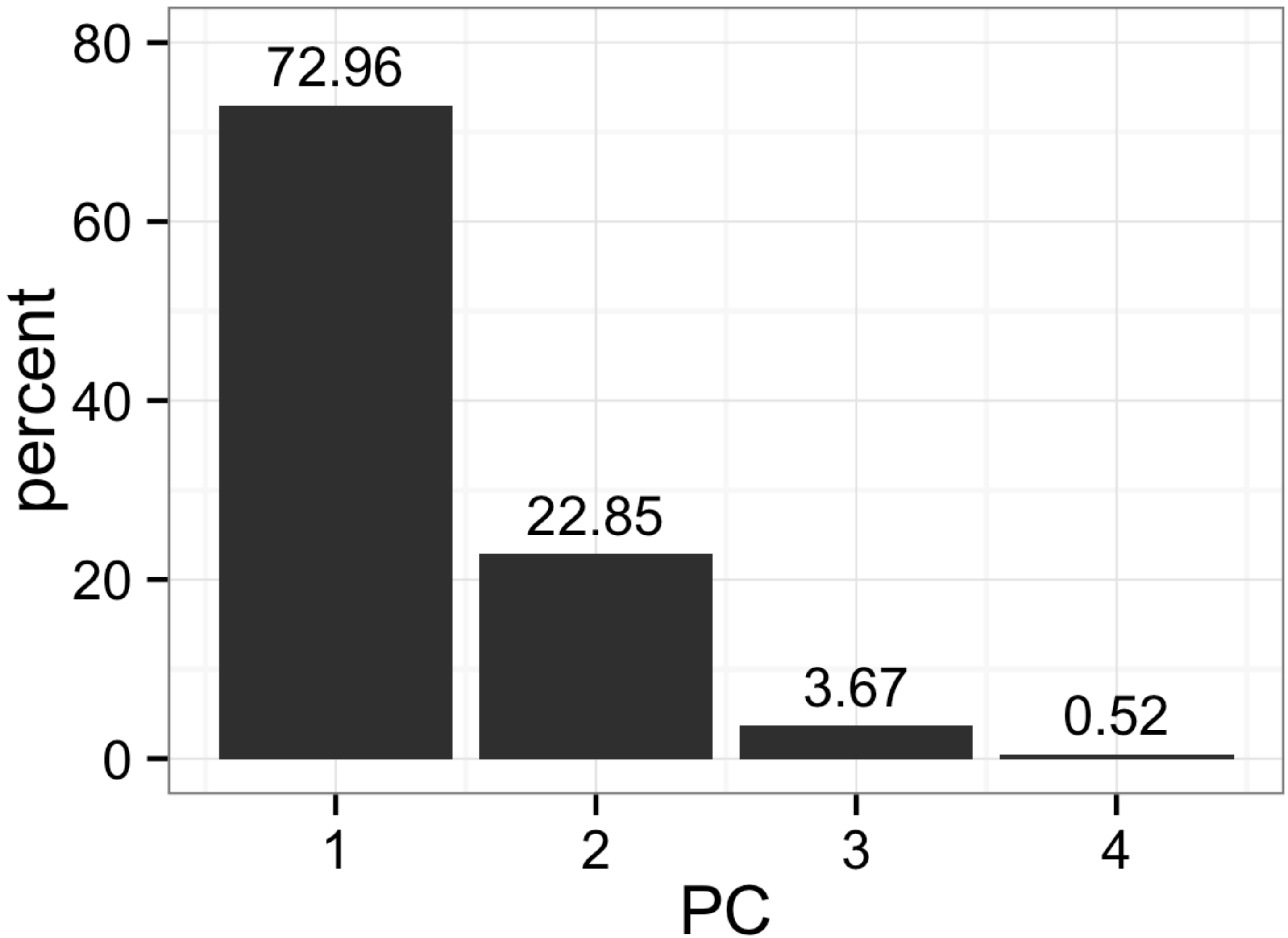
Doing a PCA in R

```
> pca
Standard deviations:
[1] 1.7083611 0.9560494 0.3830886 0.1439265

Rotation:
                    PC1         PC2        PC3        PC4
Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

Doing a PCA in R

```
> pca
Standard deviations:
[1] 1.7083611 0.9560494 0.3830886 0.1439265

Rotation:
                    PC1         PC2        PC3        PC4
Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

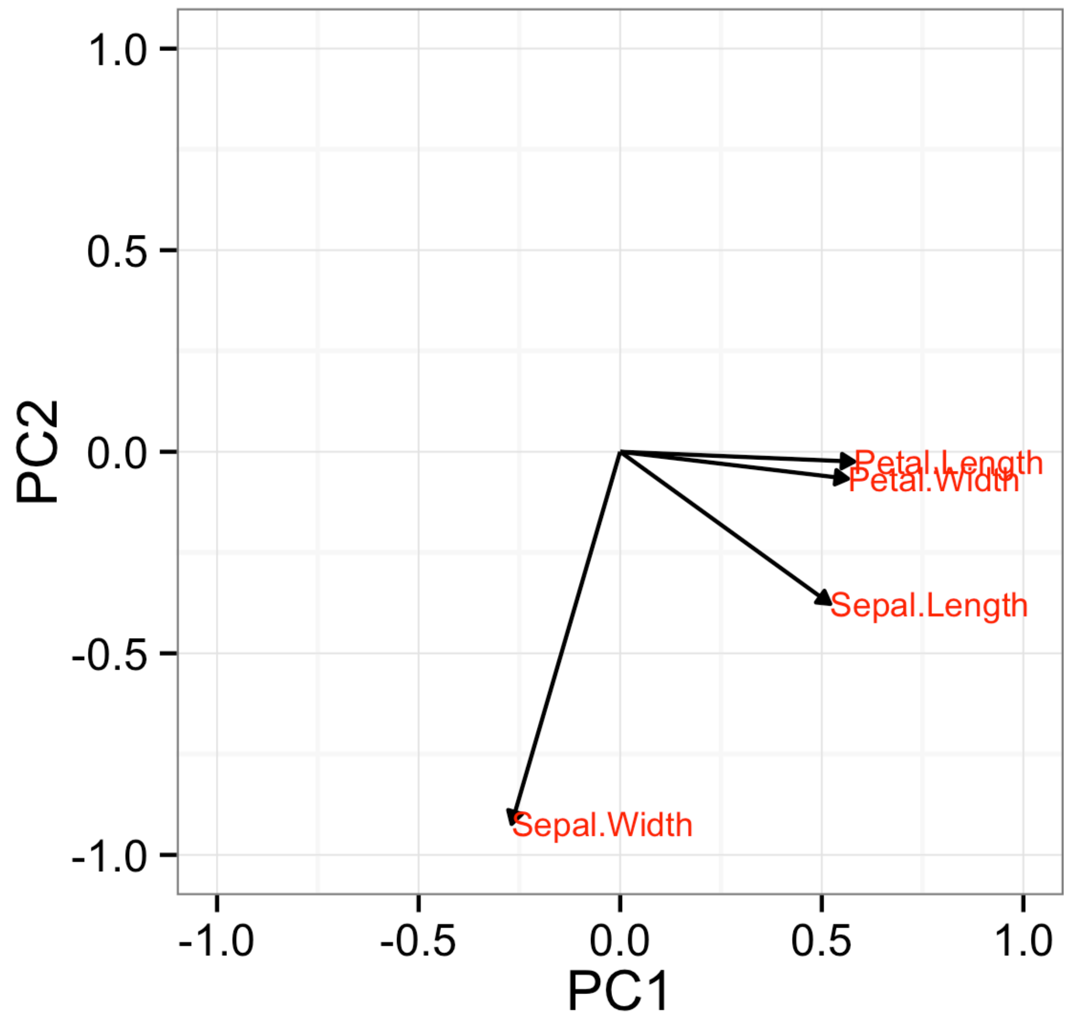
Squares of the std. devs represent the % variance explained by each PC



Doing a PCA in R

```
> pca
Standard deviations:
[1] 1.7083611 0.9560494 0.3830886 0.1439265

Rotation:
                    PC1         PC2        PC3        PC4
Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

The rotation matrix tells us which variables contribute to which PCs



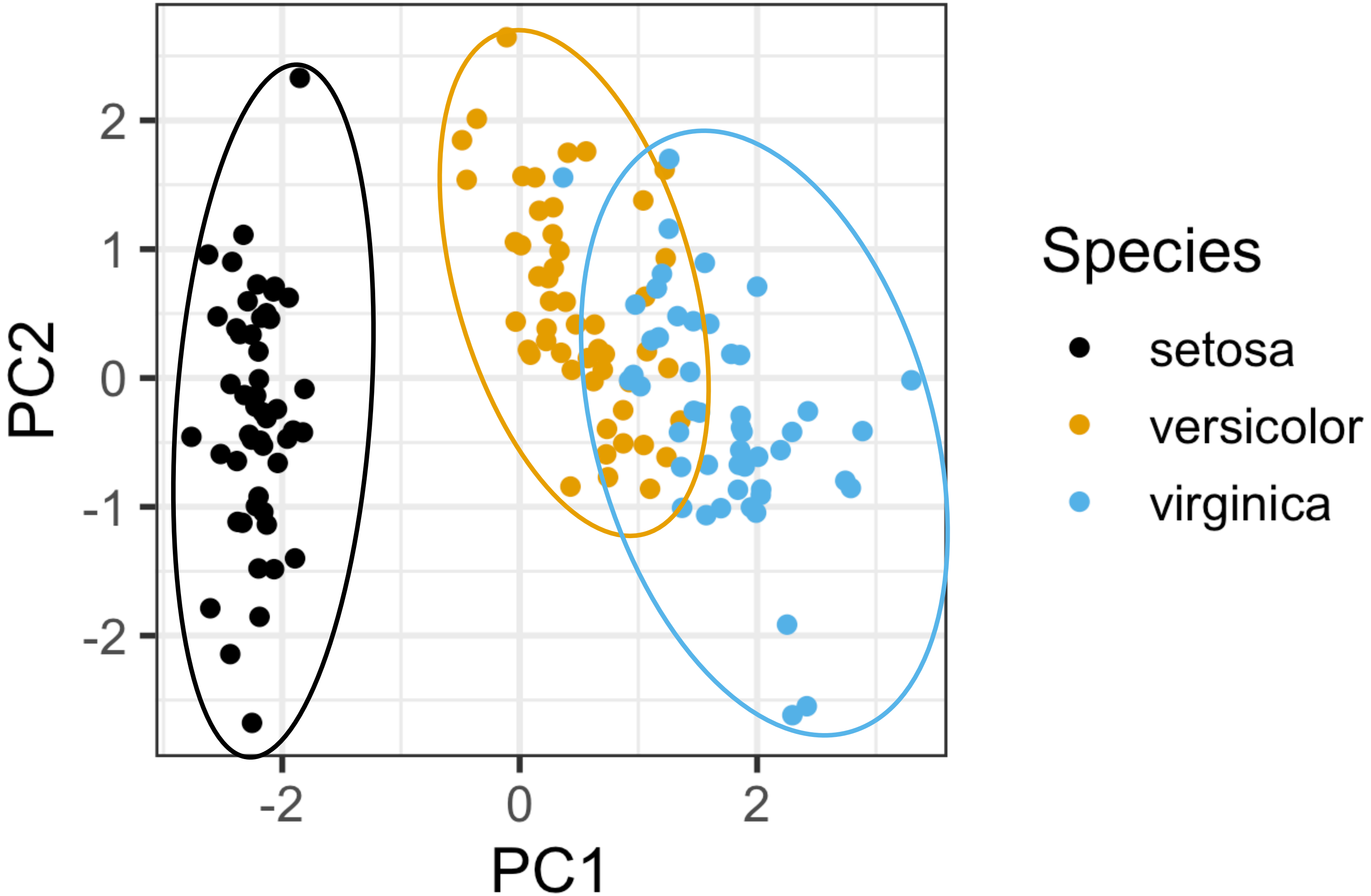
We can also recover each original observation expressed in PC coordinates

```
> pca$x
```

We can also recover each original observation expressed in PC coordinates

```
> pca$x
             PC1          PC2          PC3          PC4
  [1,] -2.25714118 -0.478423832  0.127279624  0.024087508
  [2,] -2.07401302  0.671882687  0.233825517  0.102662845
  [3,] -2.35633511  0.340766425 -0.044053900  0.028282305
  [4,] -2.29170679  0.595399863 -0.090985297 -0.065735340
  [5,] -2.38186270 -0.644675659 -0.015685647 -0.035802870
  [6,] -2.06870061 -1.484205297 -0.026878250  0.006586116
  [7,] -2.43586845 -0.047485118 -0.334350297 -0.036652767
  [8,] -2.22539189 -0.222403002  0.088399352 -0.024529919
  [9,] -2.32684533  1.111603700 -0.144592465 -0.026769540
 [10,] -2.17703491  0.467447569  0.252918268 -0.039766068
 [11,] -2.15907699 -1.040205867  0.267784001  0.016675503
 [12,] -2.31836413 -0.132633999 -0.093446191 -0.133037725
 [13,] -2.21104370  0.726243183  0.230140246  0.002416941
```

Plot of iris plants in PC coordinates reveals differences among species



These differences are much harder to see in the original variables