Statistical Operations and Matrices (II)

Predictive Modeling & Statistical Learning

Gaston Sanchez

Geometry of the Data Matrix

Matrix Structure

Data

The analyzed data can be expressed in matrix format $\mathbf{X}$:

$$\underset{n \times p}{\mathbf{X}} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

- n objects in the rows
- p quantitative variables in the columns

Looking at Rows and Columns

Data Concerns

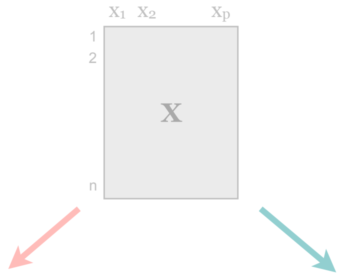
Two sides of the same coin

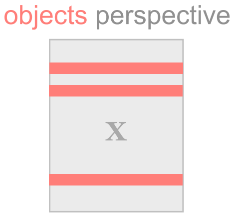
When the analyzed data can be expressed as a matrix with objects in rows, and variables in columns, we commonly care for two issues:

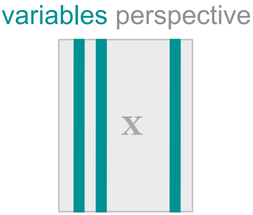
- Study the resemblance between objects
- Study the relationships among variables

Data Perspectives

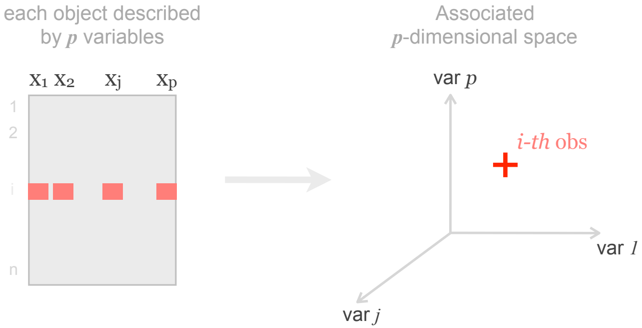
looking at a data matrix from two perspectives



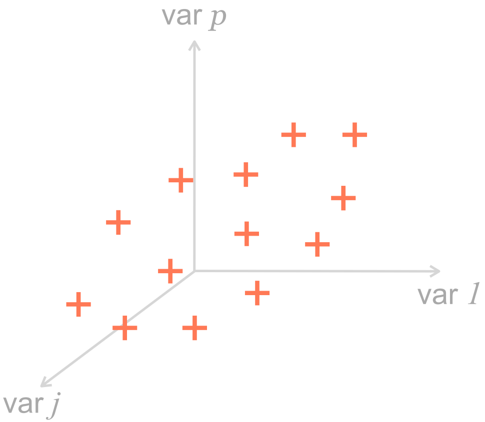




Objects Perspective

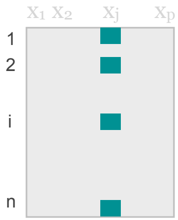


Objects as points in a p-dimensional space

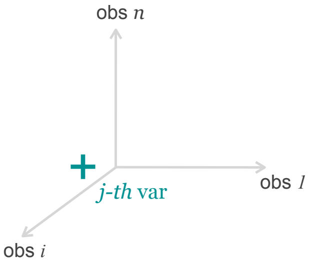


Variables Perspective

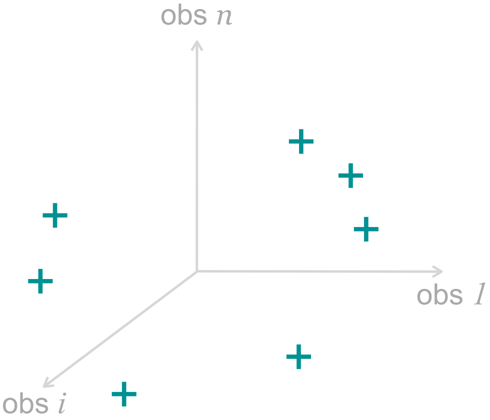
each variable described
by n observations



Associated
n-dimensional space



Variables as points in a n-dimensional space



Raw Data

Raw Data Matrix

The analyzed data can be expressed in matrix format $\mathbf{X}$:

$$\underset{n\times p}{\mathbf{X}} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

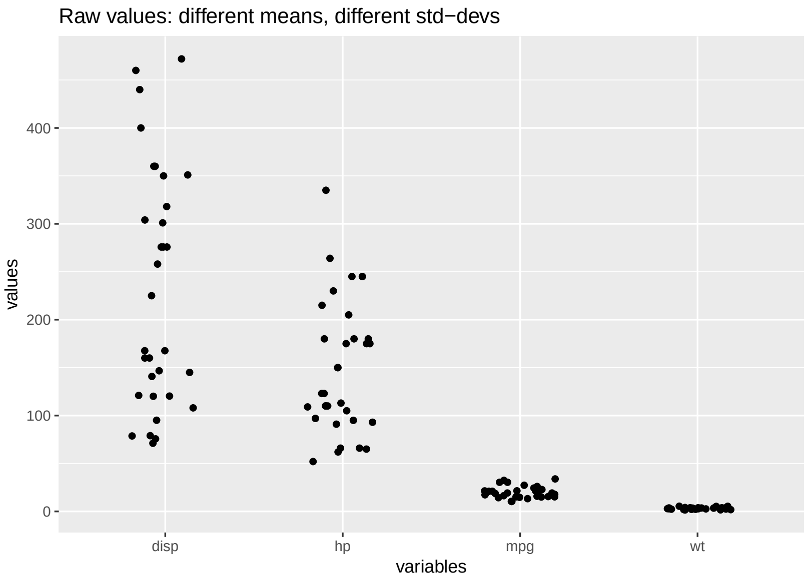
- n objects in the rows
- p quantitative variables in the columns

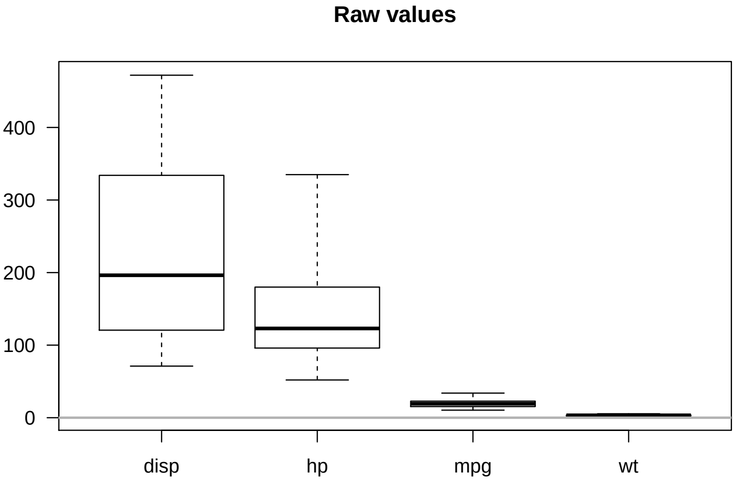
Data set `mtcars`

First 10 rows:

```
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D         24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230          22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
```

Let's use variables: `mpg`, `disp`, `hp`, and `wt`.


Raw values: different means, different std−devs
values
400
300
200
100
0
disp
hp
mpg
wt
variables


Raw values
400
300
200
100
0
disp
hp
mpg
wt

Centering Data Matrix

Mean-Centered Data Matrix

A common operation consists of **centering** the data, which involves mean-centering the variables so that they all have mean zero.

Mean-Centered Data Matrix

The mean-centered (a.k.a. column centered) matrix $\mathbf{X_C}$:

$$\underset{n\times p}{\mathbf{X_C}} = \begin{bmatrix} x_{11} - \bar{x}_1 & x_{12} - \bar{x}_2 & \cdots & x_{1p} - \bar{x}_p \\ x_{21} - \bar{x}_1 & x_{22} - \bar{x}_2 & \cdots & x_{2p} - \bar{x}_p \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} - \bar{x}_1 & x_{n2} - \bar{x}_2 & \cdots & x_{np} - \bar{x}_p \end{bmatrix}$$

where $\bar{x}_j$ is the mean of the j-th variable $(j = 1, \ldots, p)$

Mean-Centered Data Matrix

Using matrix notation, the centering operation is expressed as:

$$\mathbf{X_C} = (\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T})\mathbf{X}$$

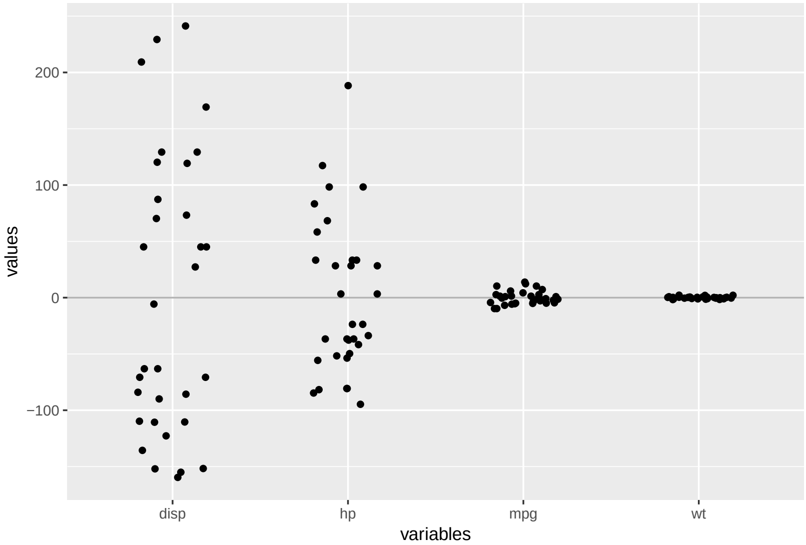
- $\mathbf{I}$ is the $n \times n$ identity matrix
- $\mathbf{1}$ is an $n \times 1$ vector of ones

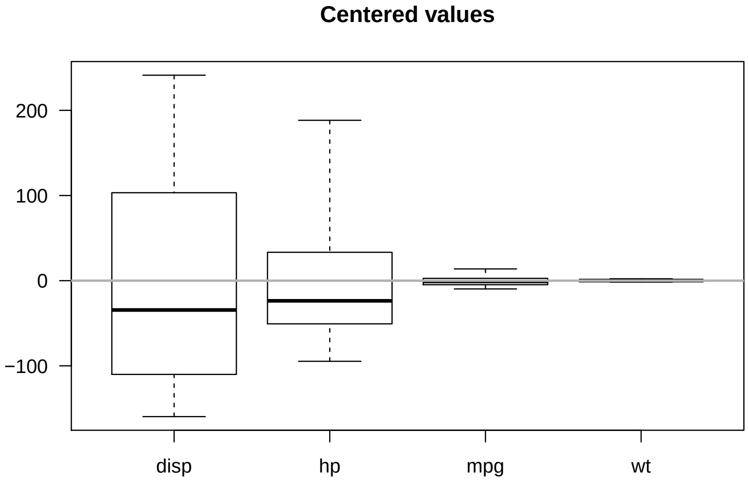
$\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T}$ is sometimes called the *centering* operator

Centering Effects

What does mean-centering do to the cloud of points?

Centered: mean = 0, different std−devs




Centered values
200
100
0
−100
disp
hp
mpg
wt

Centering Matrices in R

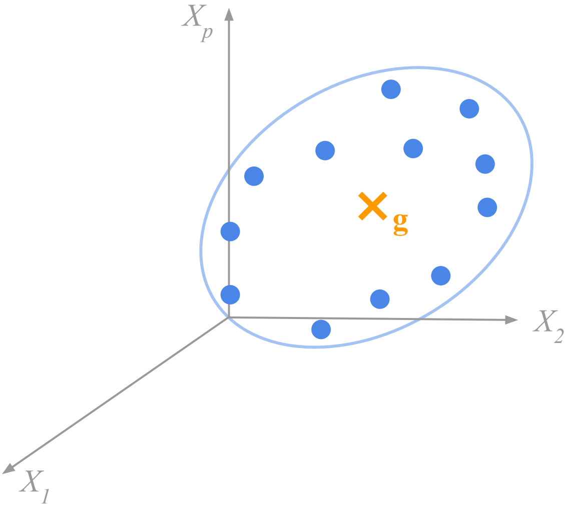
Centering with `scale()`

```
X_centered <- scale(X, center = TRUE, scale = FALSE)
```

Or also like this:

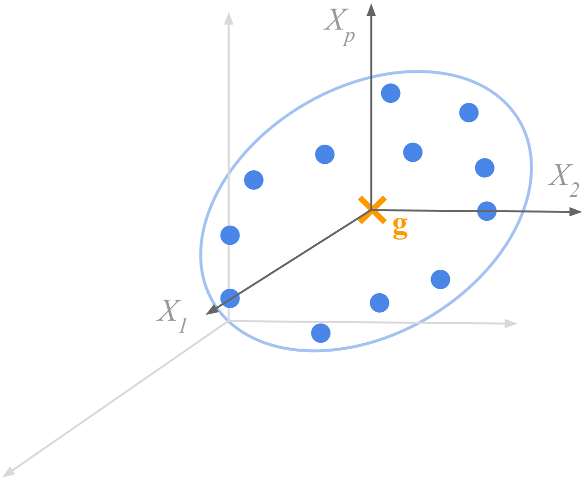
```
centroid <- colMeans(X)
X_centered <- sweep(X, 2, centroid, FUN = "-")
```

Cloud of individuals



Raw (i.e. non-centered) variables

Cloud of individuals



Centered variables

Scaled Data Matrix

Scaled or Normalized Data Matrix

The scaled or *Normalized* matrix $\mathbf{X_N}$:

$$\underset{n\times p}{\mathbf{X_N}} = \begin{bmatrix} a_1x_{11} & a_2x_{12} & \cdots & a_px_{1p} \\ a_1x_{21} & a_2x_{22} & \cdots & a_px_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_1x_{n1} & a_2x_{n2} & \cdots & a_px_{np} \end{bmatrix}$$

where a_j is a scaling factor for the j-th column

Some Scaling Options

Probably the most common scaling option is to divide by the standard deviation:

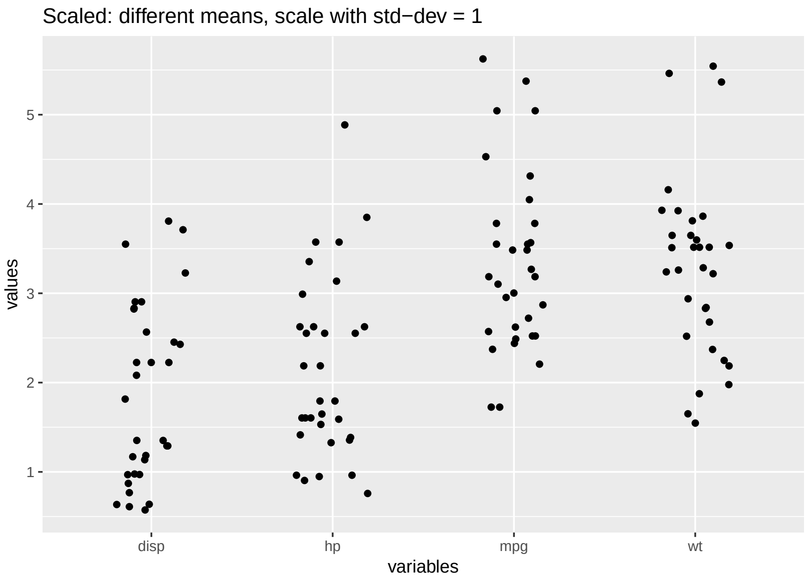
$$a_j = \frac{1}{sd_j} = 1/\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}$$

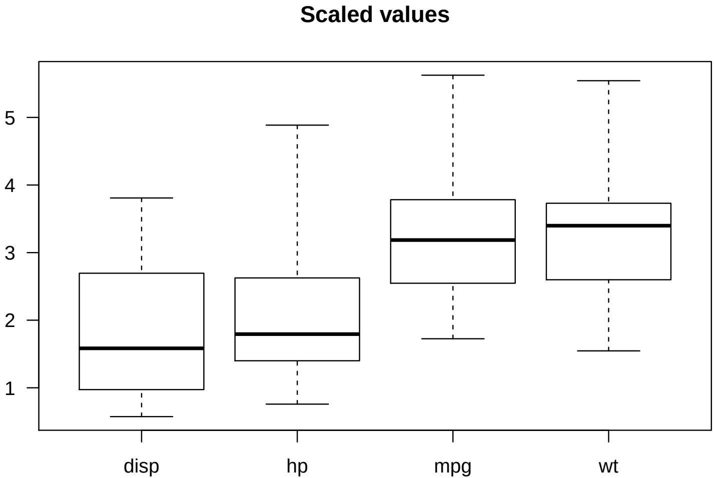
Scaling Matrices in R

Scaling with standard deviation

```
stdevs <- apply(X, 2, sd)

X_scaled <- scale(X, center = FALSE, scale = stdevs)
```


Scaled: different means, scale with std−dev = 1
values
5
4
3
2
1
disp
hp
mpg
wt
variables


Scaled values
5
4
3
2
1
disp
hp
mpg
wt

Some Scaling Options

Other typical scaling options are based on L_p-norms:

$$L_p\text{-norm} = \left(\sum_{i=1}^{n} |x_{ij}|^p \right)^{1/p}$$

The most common L_p-norms are:

- L_1-norm: $\sum_{i=1}^{n} |x_{ij}|$
- L_2-norm: $\sqrt{\sum_{i=1}^{n} (x_{ij})^2}$
- L_∞-norm: $max\{|x_{i1}|, \ldots, |x_{ip}|\}$

Some Scaling Options

Using L_p-norms, the scaling factors a_j are:

- L_1-norm: $a_j = 1/\sum_{i=1}^{n} |x_{ij}|$
- L_2-norm: $a_j = 1/\sqrt{\sum_{i=1}^{n} (x_{ij})^2}$
- L_∞-norm: $a_j = 1/max\{|x_{i1}|, \ldots, |x_{ip}|\}$
- L_p-norm: $a_j = 1/\left(\sum_{i=1}^{n} |x_{ij}|^p\right)^{1/p}$

Scaled or Normalized Data Matrix

The scaling factors a_j can be put in a diagonal matrix $\mathbf{D_a}$

$$\underset{p\times p}{\mathbf{D_a}} = \begin{bmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_p \end{bmatrix}$$

then the scaled or normalized data matrix is given by:

$$\mathbf{X_N} = \mathbf{X}\mathbf{D_a}$$

Normalizing Effects

What does normalizing (i.e. scaling) do to the cloud of points?

Scaling Matrices in R

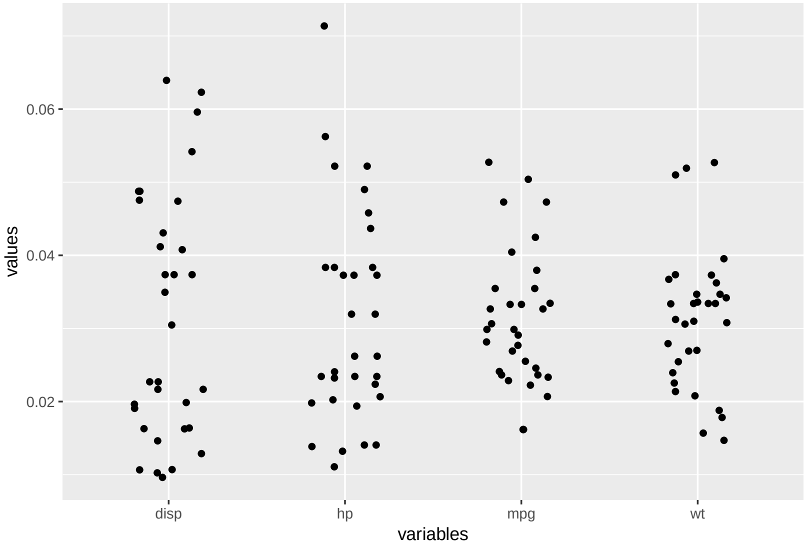
Scaling with L_1-norm:

$$\sum_{i=1}^{n} |x_{ij}|$$

```r
# L-1 norm
one_norms <- apply(X, 2, function(u) sum(abs(u)))

X_scaled <- scale(X, center = FALSE, scale = one_norms)
```

Scaled: different means, scaled with L1−norm



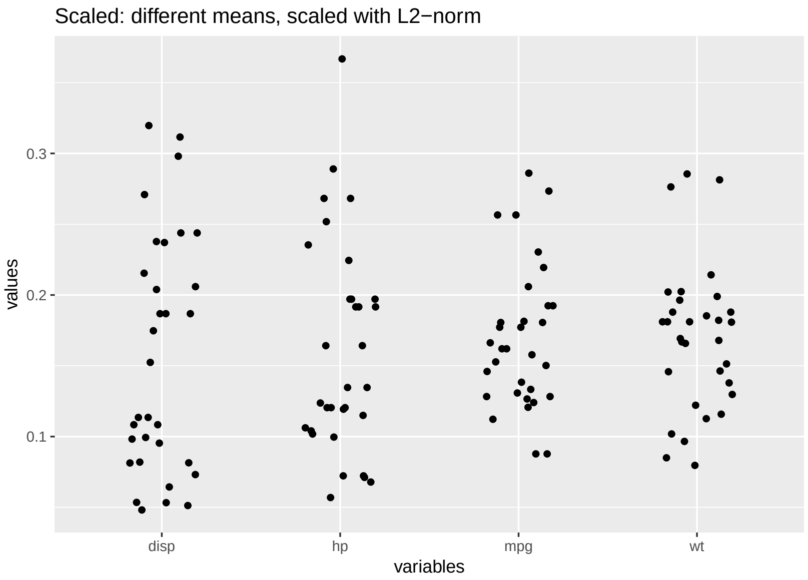
Scaling in R examples

Scaling with L_2-norm

$$\sqrt{\sum_{i=1}^{n}(x_{ij})^2}$$

```
# L-2 norm
two_norms <- apply(X, 2, function(u) sqrt(sum(u*u)))

X_scaled <- scale(X, center = FALSE, scale = two_norms)
```


Scaled: different means, scaled with L2−norm
values
0.3
0.2
0.1
disp
hp
mpg
wt
variables

Scaling Matrices in R

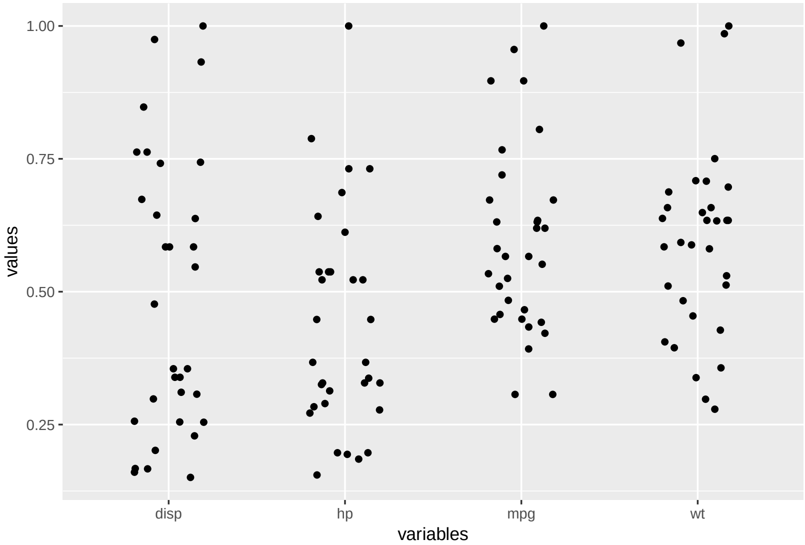
Scaling with L_∞-norm

$$max\{|x_{i1}|, \dots, |x_{ip}|\}$$

```r
# L-inf norm
inf_norms <- apply(X, 2, function(u) max(abs(u)))

X_scaled <- scale(X, center = FALSE, scale = inf_norms)
```

Scaled: different means, scaled with Lmax−norm



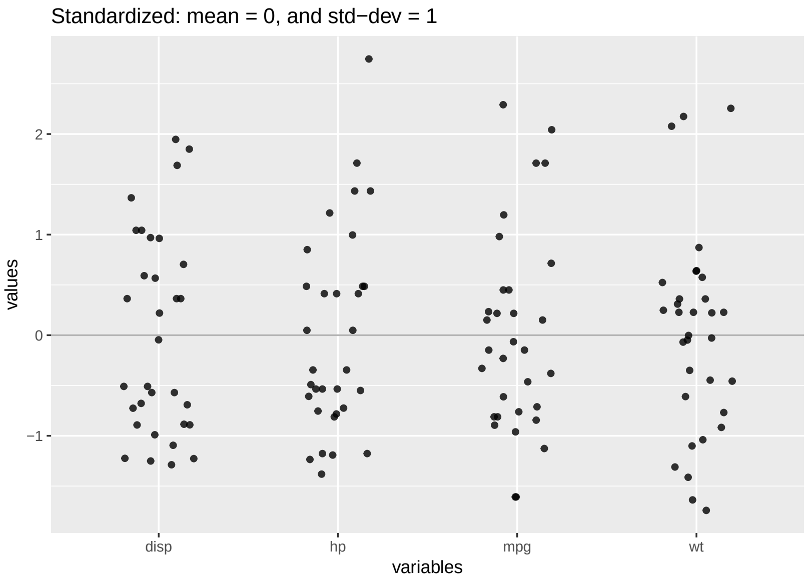
Standardized Data Matrix

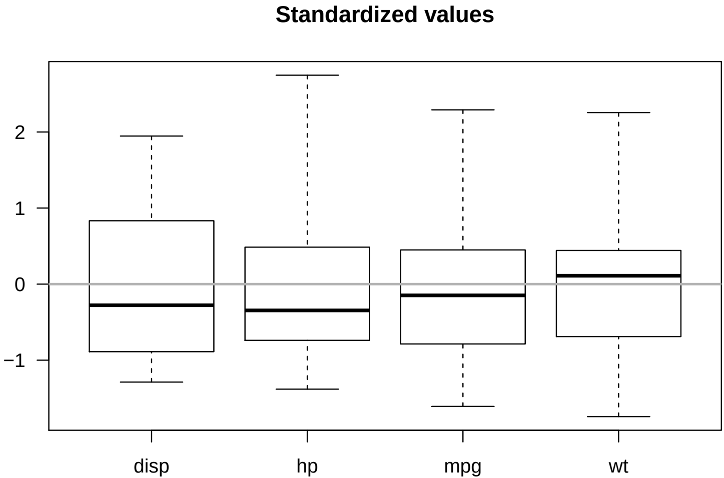
Standardized Data Matrix

The standardized matrix $\mathbf{X_S}$ is the mean-centered and scaled (by the standard deviation) matrix:

$$\underset{n \times p}{\mathbf{X_S}} = \begin{bmatrix} \frac{x_{11} - \bar{x}_1}{sd_1} & \frac{x_{12} - \bar{x}_2}{sd_2} & \cdots & \frac{x_{1p} - \bar{x}_p}{sd_p} \\ \frac{x_{21} - \bar{x}_1}{sd_1} & \frac{x_{22} - \bar{x}_2}{sd_2} & \cdots & \frac{x_{2p} - \bar{x}_p}{sd_p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{x_{n1} - \bar{x}_1}{sd_1} & \frac{x_{n2} - \bar{x}_2}{sd_2} & \cdots & \frac{x_{np} - \bar{x}_p}{sd_p} \end{bmatrix}$$

- $\bar{x}_j$ is the mean of the j-th variable
- sd_j is the standard deviation of the j-th variable


Standardized: mean = 0, and std−dev = 1
2
1
0
−1
values
disp
hp
mpg
wt
variables


Standardized values
2
1
0
−1
disp
hp
mpg
wt

Standardized Data Matrix

When the scaling factors a_j are the standard deviations sd_j, the scaling matrix $\mathbf{D}_{\frac{1}{sd}}$ is:

$$\underset{p\times p}{\mathbf{D}_{\frac{1}{sd}}} = \begin{bmatrix} \frac{1}{sd_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{sd_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{sd_p} \end{bmatrix}$$

then the standardized data matrix $\mathbf{X_S}$

$$\mathbf{X_S} = \mathbf{X_C}\mathbf{D}_{\frac{1}{sd}} = (\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathsf{T}})\mathbf{X}\mathbf{D}_{\frac{1}{sd}}$$

Standardizing Matrices in R

Standardizing with `scale()`

```
X_std <- scale(X, center = TRUE, scale = TRUE)

# equivalent to
X_std <- scale(X)
```

Objects and their weights

Weights of Objects

- We can assume that each object is associated to a **weight**
- Think of a weight as the "importance" of an observation
- Usually, we assume equal weights $1/n$ (i.e. equal importance)
- If we assume that objects come from a random sample, then the n objects have the same chance $1/n$ of being selected
- Sometimes, however, it is convenient to assume that each object has a general weight $w_i > 0$, such that $\sum_{i=1}^{n} w_i = 1$

Weights of Objects

We can consider a diagonal matrix of object weights $\mathbf{D}$:

$$\underset{n\times p}{\mathbf{D}} = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_n \end{bmatrix}$$

In the more common case that all weights are equal, we have $\mathbf{D} = \frac{1}{n}\mathbf{I}$

Weights of Objects

The vector $\mathbf{g}$ containing the means $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_p$ of all variables can be written as:

$$\mathbf{g} = \mathbf{X}^\mathsf{T}\mathbf{D}\mathbf{1_n}$$

where $\mathbf{1_n}$ is an $n \times 1$ vector of ones.

The vector $\mathbf{g}$ is also known as the **centroid** of the objects.

Centered Data Matrix

Using $\mathbf{D}$ and $\mathbf{g}$ we can write an expression to get a centered data matrix $\tilde{\mathbf{X}}$

$$\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{1}\mathbf{g}^{\mathsf{T}} = (\mathbf{I} - \mathbf{1}\mathbf{1}^{\mathsf{T}}\mathbf{D})\mathbf{X}$$

Cross-Products

Data Matrix Products

There are two fundamental matrix products that play a crucial role when the data is in an $n \times p$ matrix X with objects in rows, and variables in columns (assume $n > p$):

$$\mathbf{X}^\mathsf{T}\mathbf{X} \quad \& \quad \mathbf{X}\mathbf{X}^\mathsf{T}$$

Minor Product Moment

$$\mathbf{X}^\mathsf{T}\mathbf{X}$$

- a.k.a. “minor product moment”
  (because is of size $p \times p$, assuming $n > p$)
- sum-of-squares and cross-products (SSCP) of columns
- made of inner products of the columns of $\mathbf{X}$
- *association* matrix for the variables

Major Product Moment

$$\mathbf{X}\mathbf{X}^\mathsf{T}$$

- a.k.a. "major product moment"
  (because is of size $n \times n$, assuming $n > p$)
- sum-of-squares and cross-products of rows
- made of inner products of the rows of $\mathbf{X}$
- association matrix for the objects

Covariance Matrix

If $\mathbf{X}$ is mean-centered, then

$$\frac{1}{n}\mathbf{X}^\intercal\mathbf{X} \qquad \text{and} \qquad \frac{1}{n-1}\mathbf{X}^\intercal\mathbf{X}$$

are the covariance matrices (population and sample flavors)

Correlation Matrix

If $\mathbf{X}$ is standardized, then

$$\frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X} \qquad \text{and} \qquad \frac{1}{n-1}\mathbf{X}^\mathsf{T}\mathbf{X}$$

are the correlation matrices (population and sample flavors)