11. Statistics II

- Covariance
- Correlation
- Covariance matrix/correlation matrix
- Eigenvalues/vectors of covariance matrix

Reading data from csv files

- Download the file 'cars.csv' from our course page and type as follows to load data
  - CSV=Comma Separated Value

```
>> data=csvread('cars.csv');
```

  - This file* contains 7 types of numeric data for 406 cars (e.g., MPG(Miles Per Gallon), Horsepower, etc.)
  - csvread can only read numeric data correctly

```
>> data =
 Columns 1 through 8:
   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00
   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00   0.0000e+00
   0.0000e+00   1.8000e+01   8.0000e+00   3.0700e+02   1.3000e+02   3.5040e+03   1.2000e+01   7.0000e+01
   0.0000e+00   1.5000e+01   8.0000e+00   3.5000e+02   1.6500e+02   3.6930e+03   1.1500e+01   7.0000e+01
   0.0000e+00   1.8000e+01   8.0000e+00   3.1800e+02   1.5000e+02   3.4360e+03   1.1000e+01   7.0000e+01
   0.0000e+00   1.6000e+01   8.0000e+00   3.0400e+02   1.5000e+02   3.4330e+03   1.2000e+01   7.0000e+01
   0.0000e+00   1.7000e+01   8.0000e+00   3.0200e+02   1.4000e+02   3.4490e+03   1.0500e+01   7.0000e+01
```

Car	MPG	Cylinders	Displacement	Horsepower	Weight	Acceleration	Model	Origin
STRING	DOUBLE	INT	DOUBLE	DOUBLE	DOUBLE	DOUBLE	INT	CAT
Chevrolet Chevelle Malibu	18	8	307	130	3504	12	70	US
Buick Skylark 320	15	8	350	165	3693	11.5	70	US
Plymouth Satellite	18	8	318	150	3436	11	70	US
AMC Rebel SST	16	8	304	150	3433	12	70	US
Ford Torino	17	8	302	140	3449	10.5	70	US

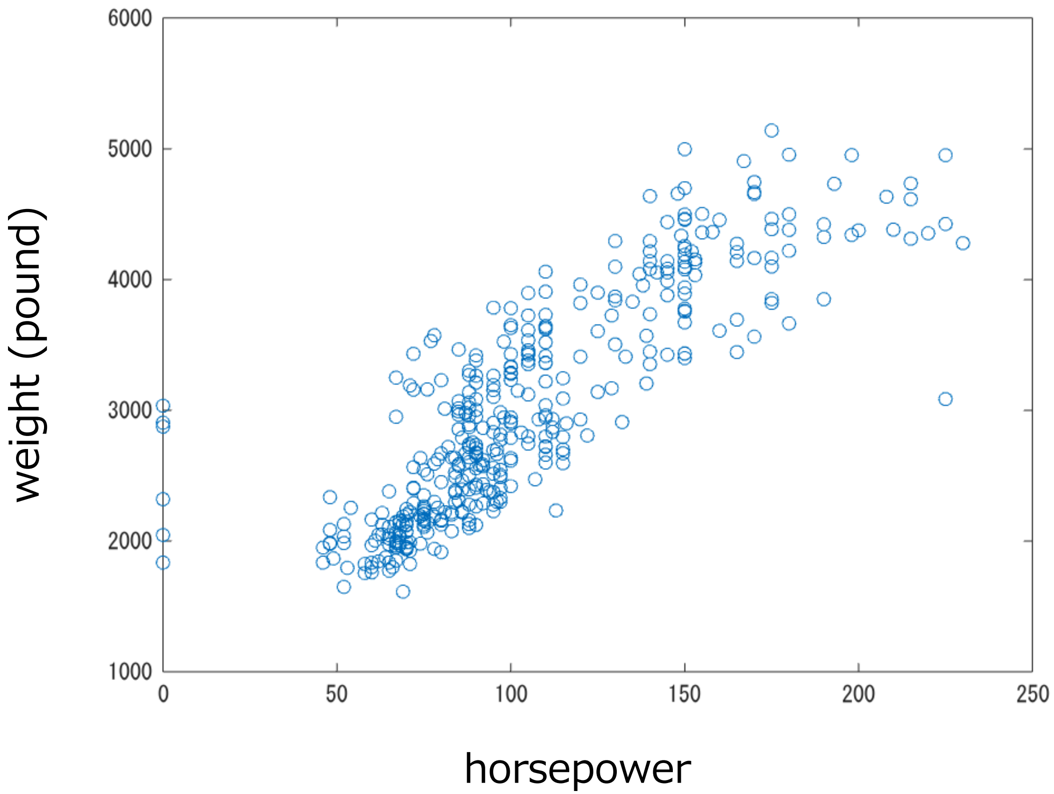
(pounds) (seconds for 0–60 mph (0–97 km/h))

* The file copied from https://perso.telecom-paristech.fr/eagan/class/igr204/datasets

Covariance/correlation of two variables

- Covariance = a measure of linear relation between variables, or a linear measure of dependency of two variables
- Correlation = extent to which two variables have a linear relationship with each other
- Draw a *scatter plot* of the horsepower and weight of each of 406 cars

```
>> plot(data(3:408,5),data(3:408,6),'o')
```



- A linear relationship is observed
- We ignore several invalid points, which are on the 'x=0' axis

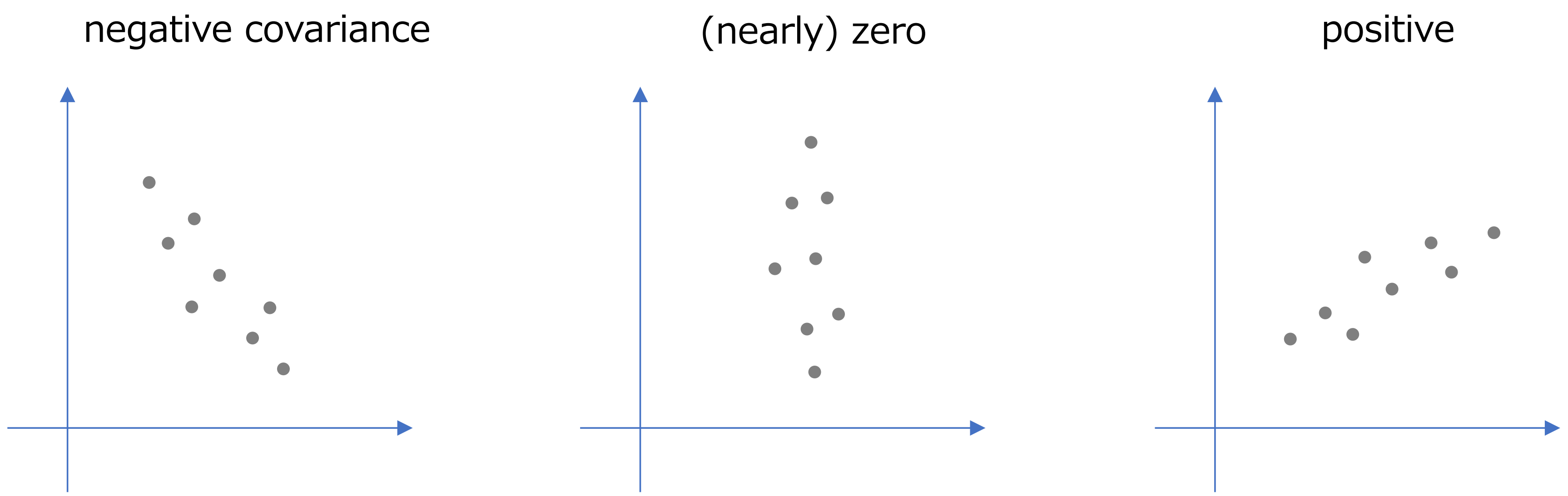
Covariance of two variables

- Definition (Covariance):

$$\mathrm{cov}(X, Y) = E[(X - E(X))(Y - E(Y))]$$

or $$\mathrm{cov}(\mathbf{x}, \mathbf{y}) = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$$

- Properties



- If two variables are identical, covariance is merely variance

$$\mathrm{cov}(X, X) = E[(X - E(X))^2] = \mathrm{var}(X) = \sigma^2(X)$$

Correlation coefficient (or simply called *correlation*)

- Definition (also known as *Pearson's correlation coefficient*):
  - Can be thought of as *normalized* covariance

$$r(X,Y) = \frac{\mathrm{cov}(X,Y)}{\sigma(X)\sigma(Y)}$$

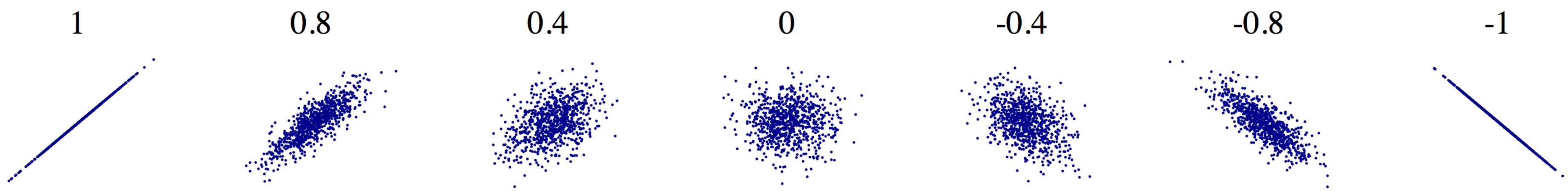
standard deviation:

$$\sigma(X) = \sqrt{\mathrm{var}(X)} \quad \sigma(Y) = \sqrt{\mathrm{var}(Y)}$$

- `corr` calculates correlation coefficient

```
>> corr(data(3:408,5),data(3:408,6))
ans = 0.84081
```

- Has a value in the range [-1,1]
  - Positive and negative; 0 means there is no correlation



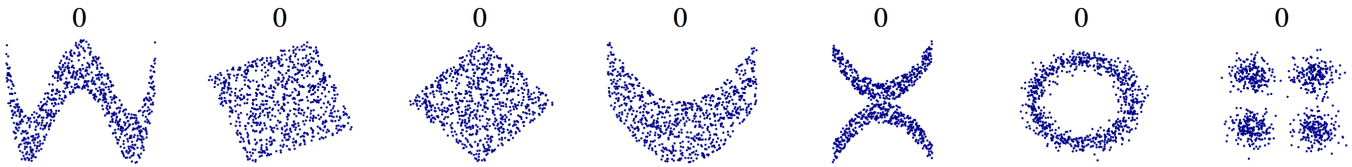
Remarks on correlation

- Correlation does not mean *causality*
  - There can be correlation between two variables even if there is no causal relationship between them
  - E.g., Nobel laureates and chocolate consumption

- *Dependence* is sometimes synonymous with *correlation*, but it is rigorously defined by *probabilistic independence*:
  - Two events A and B are mutually *independent* if and only if

$$\mathrm{P}(A \cap B) = \mathrm{P}(A)\mathrm{P}(B) \Leftrightarrow \mathrm{P}(B) = \mathrm{P}(B \mid A)$$

- Correlation captures only a linear relationship, not a nonlinear one
  - All the point data below have zero correlation!



Covariance matrix/correlation matrix

- There are seven variables in the 'car.csv' data
- We can calculate correlation/covariance between any two (including self) of the seven variables, which creates a 7x7 matrix, called correlation/covariance matrices

- Suppose a Nx7 matrix storing the data

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top$$

- Covariance matrix of the data is defined as

$$\mathrm{cov}(\mathbf{X}) = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^\top = \frac{1}{N-1}\tilde{\mathbf{X}}^\top\tilde{\mathbf{X}}$$

where m is the mean vector of x and $\tilde{\mathbf{X}} = [\mathbf{x}_1 - \mathbf{m}, \ldots, \mathbf{x}_N - \mathbf{m}]^\top$

- Correlation matrix can be defined similarly

Covariance matrix/correlation matrix

- `cov` and `corr` gives these matrices from X as below
  - *Check which pair of variables correlates and to what extent it is*

```
>> X=data(3:408,2:8);
>> size(X)
ans =

   406    7

>> cov(X)
ans =

   7.0590e+01  -1.0581e+01  -6.7374e+02  -2.4739e+02  -5.6042e+03   9.9981e+00   1.8464e+01
  -1.0581e+01   2.9315e+00   1.7098e+02   5.7130e+01   1.2983e+03  -2.5077e+00  -2.3155e+00
  -6.7374e+02   1.7098e+02   1.1009e+04   3.7148e+03   8.2869e+04  -1.6412e+02  -1.5014e+02
  -2.4739e+02   5.7130e+01   3.7148e+03   1.6419e+03   2.8858e+04  -7.7476e+01  -6.3788e+01
  -5.6042e+03   1.2983e+03   8.2869e+04   2.8858e+04   7.1742e+05  -1.0212e+03  -1.0014e+03
   9.9981e+00  -2.5077e+00  -1.6412e+02  -7.7476e+01  -1.0212e+03   7.8588e+00   3.1737e+00
   1.8464e+01  -2.3155e+00  -1.5014e+02  -6.3788e+01  -1.0014e+03   3.1737e+00   1.4053e+01

>> corr(X)
ans =

   1.00000  -0.73556  -0.76428  -0.72667  -0.78751   0.42449   0.58623
  -0.73556   1.00000   0.95179   0.82347   0.89522  -0.52245  -0.36076
  -0.76428   0.95179   1.00000   0.87376   0.93247  -0.55798  -0.38171
  -0.72667   0.82347   0.87376   1.00000   0.84081  -0.68205  -0.41993
  -0.78751   0.89522   0.93247   0.84081   1.00000  -0.43009  -0.31539
   0.42449  -0.52245  -0.55798  -0.68205  -0.43009   1.00000   0.30199
   0.58623  -0.36076  -0.38171  -0.41993  -0.31539   0.30199   1.00000
```

the previously computed horsepower-weight correlation here

MPG	Cylinders	Displacement	Horsepower	Weight	Acceleration	Model

Eigenvalues/vectors of a covariance matrix

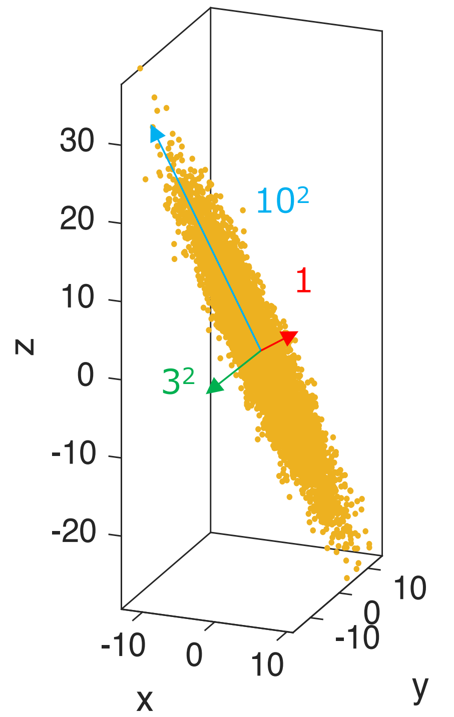
- Covariance matrices explain how data points distribute in the data space
- Eigenvectors of a covariance matrix explain in which directions data points spread in the space
- The eigenvalue associated with each eigenvector indicates the width of the spread in that direction

```
>> load('3d_ptdata')
>> size(X)
ans =
   10000         3

>> plot3(X(:,1),X(:,2),X(:,3),'.'); axis equal
>> [V,W]=eig(cov(X))
V =
   0.867213  -0.208827  -0.452032
   0.496518   0.431157   0.753375
   0.037571  -0.877778   0.477591

E =
Diagonal Matrix

     0.98420          0          0
           0    9.05693          0
           0          0  103.15470
```

Download 'e11files.zip' from the course page and extract '3d_ptdata' from it before typing these commands

Three orthogonal axes

Spread widhts (variances)



Exercises 11.1

(also known as principal component analysis)

- The last method of analyzing data based on eigenvalue/vectors of covariance matrices can be applied to any type of data; let's consider a set of images here
- First, download a set of face images from URL below and expand them in directory 'att_faces' in your working directory
  - http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
- Second, copy the script 'load_faces.m' in 'e11files.zip' downloaded from the course page
- Using this script, load 400 face images (92x112 pixels) to X, a 400x10304(=92x112) matrix, by typing

```
>> load_faces
```

- You can display, say, the 100th image, by typing

```
>> imshow(reshape(X(100,:),[112,92])/255)
```

'/255' may not be necessary depending on your system

`reshape` reshapes a vector into a matrix of 92x112, which is treated as an image

Continued to the next page…

Exercises 11.1

- Calculate the first 20 eigenvalues of the covariance matrix of X and plot them
  - Remark: In this example, each data point is a single image; it resides in 92x112=10304-dimensional space; there are 400 data points (=face images); thus, cov(X) is a 10304x10304 matrix and its computation is very, very time-consuming (don't do this)
  - Hint: Recall the relation between SVD and the eigenvalue problem; use SVD instead of `eig(cov(X))`; to be specific, type below

```
>> [U,W,V]=svds(X-ones(400,1)*mean(X),20);
```

`svds` calculates a specified number of largest singular values and related vectors

  - See that the first few singular values (square root of eigenvalues) are very large and the subsequent singular values are very small
  - Remark: This means that the data reside only in a *low-dimensional subspace* in the 10304-dim data space
- Calculate also the eigenvectors and then display them as images of 92x112 pixels
  - Hint: Eigenvectors have negative elements in general and thus some normalization of brightness necessary; you can display the first eigenvector as a 92x112 image by

```
>> svec=V(:,1);
>> imshow(reshape(svec,[112,92]),[min(svec),max(svec)])
```