linear_regression

Formula of linear regression

theory

Linear relation between y and x

$$y = c_0 + c_1 x + \varepsilon = f(x). \tag{1}$$

Intercept from Eqn (1) is

$$c_0 = \frac{\sum_{i=1}^{N} y_i \sum x_i^2 - \sum_{i=1}^{N} x_i \sum x_i y_i}{N \sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i \right)^2} \tag{2}$$

and the slope from Eqn (1) is

$$c_1 = \frac{N \sum_{i=1}^{N} x_i y_i - \sum_{i=1}^{N} x_i \sum y_i}{N \sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i \right)^2}. \tag{3}$$

Coefficient of determination is defined as

$$R^2 = 1 - \frac{SS_{\mathrm{res}}}{SS_{\mathrm{tot}}}, \tag{4}$$

which requires residual sum of squares

$$SS_{\mathrm{res}} = \sum_{i=1}^{N} (y_i - f_i)^2 = \sum_{i=1}^{N} \varepsilon_i^2 \tag{5}$$

total sum of squares

$$SS_{\mathrm{tot}} = \sum_{i=1}^{N} (y_i - \bar{y})^2, \tag{6}$$

and mean of y

$$\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i, \tag{7}$$

where $R^2 \in [0, 1]$.

There are also other formulations

$$SS_{xy} = \sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y}), \tag{8a}$$

$$SS_{xx} = \sum_{i=1}^{N} (x_i - \bar{x})^2, \tag{8b}$$

$$SS_{yy} = \sum_{i=1}^{N} (y_i - \bar{y})^2, \tag{8c}$$

$$r = \frac{SS_{xy}}{\sqrt{SS_{xx} SS_{yy}}},$$

which is know as Pearson's correlation coefficient.

derivation

How to derive Eqns (2) and (3) will be added later.

functions

$$\sum_{i=1}^{n} a_i b_i$$

In [1]:
```
def sum_product(a, b):
    N = min(len(a), len(b))
    s = 0
    for i in range(N):
        s += a[i]*b[i]
    return s
```

$$y = f(x, c) = c_0 + c_1 x, \quad c = \{c_0, c_1\}$$

In [2]:
```
def f(x, c):
    y = []
    for i in x:
        y.append(c[0] + c[1] * i)
    return y
```

$$\bar{a} = \frac{1}{N} \sum_{i=1}^{N} a_i$$

In [3]:
```
def avg(a):
    N = len(a)
    s = sum(a)
    abar = s / N
    return abar
```

$$y = f(x, c) = c_0 + c_1 x, \quad c = \{c_0, c_1\}$$

$$SS_{\text{res}} = \sum_{i=1}^{N} (y_i - f_i)^2 = \sum_{i=1}^{N} \varepsilon_i^2$$

In [4]:
```
def SSres(x, y, c):
    N = min(len(x), len(y))
    ymod = f(x, c)
    s = 0
    for i in range(N):
        s += (y[i] - ymod[i])**2
    return s
```

$$\bar{a} = \frac{1}{N} \sum_{i=1}^{N} a_i$$

$$SS_{ab} = \sum_{i=1}^{N} (a_i - \bar{a})(b_i - \bar{b})$$

In [5]:
```
def SSab(x, y):
    N = min(len(x), len(y))
    ax = avg(x)
    ay = avg(y)
    s = 0
    for i in range(N):
        s += (x[i] - ax) * (y[i] - ay)
    return s
```

test data 1

In [6]:

```
# define data
xobs = [1, 2, 3, 4, 5]
yobs = [3, 4, 5, 6, 7]
```

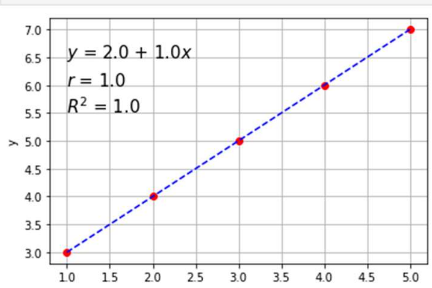
In [7]:

```
import math

N = len(xobs)

Sy = sum(yobs)
Sx = sum(xobs)
Sxx = sum_product(xobs, xobs)
Sxy = sum_product(xobs, yobs)

c0 = (Sy*Sxx - Sx*Sxy) / (N*Sxx - Sx*Sx)
c1 = (N*Sxy - Sx*Sy) / (N*Sxx - Sx*Sx)
c = [c0, c1]

r = SSab(xobs, yobs) / math.sqrt( SSab(xobs, xobs) * SSab(yobs, yobs) )
R2 = 1 - SSres(xobs, yobs, c) / SSab(yobs, yobs)

ymod = f(xobs, [c0, c1])

print("Data")
print("xobs =", xobs)
print("yobs =", yobs)
print()

print("Model")
print("c =", c)
print("ymod =", ymod)
print()

print("Pearson correlation coefficient")
print("r = ", r)
print("r2 = ", r*r)
print()

print("Coefficient of determination")
print("R2 = ", R2)
```

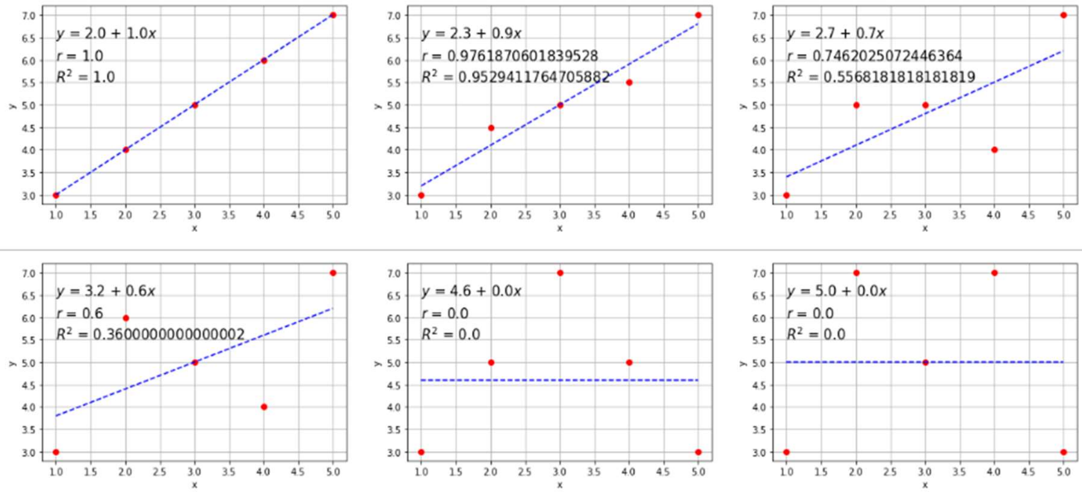
```
Data
xobs = [1, 2, 3, 4, 5]
yobs = [3, 4, 5, 6, 7]

Model
c = [2.0, 1.0]
ymod = [3.0, 4.0, 5.0, 6.0, 7.0]

Pearson correlation coefficient
r =  1.0
r2 =  1.0

Coefficient of determination
R2 =  1.0
```

In [8]:

```
import matplotlib.pyplot as plt

plt.grid()
plt.xlabel("x")
plt.ylabel("y")

plt.text(1, 6.5, f"$y$ = {c[0]} + {c[1]}$x$", fontsize=15)
plt.text(1, 6.0, f"$r$ = {r}", fontsize=15)
plt.text(1, 5.5, f"$R^2$ = {R2}", fontsize=15)

plt.plot(xobs, yobs, 'ro', xobs, ymod, 'b--')
plt.show()
```



comparison



y = 2.0 + 1.0x
r = 1.0
R² = 1.0
y = 2.3 + 0.9x
r = 0.9761870601839528
R² = 0.9529411764705882
y = 2.7 + 0.7x
r = 0.7462025072446364
R² = 0.5568181818181819
y = 3.2 + 0.6x
r = 0.6
R² = 0.3600000000000002
y = 4.6 + 0.0x
r = 0.0
R² = 0.0
y = 5.0 + 0.0x
r = 0.0
R² = 0.0
x
y