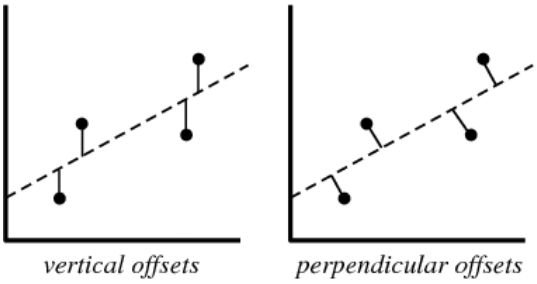
Example 48. Determine the line that "best" fits the data points $(2,1),(5,2),(7,3),(8,3)$.

Comment. As usual in practice, we are minimizing the (sum of squares of the) vertical offsets:



http://mathworld.wolfram.com/LeastSquaresFitting.html

Solution. We repeat precisely the computation from last class. This time, we let Sage do the work for us.

```
Sage] xx = vector([2,5,7,8]); yy = vector([1,2,3,3])
```

```
Sage] AT = matrix([[1,1,1,1], xx])
```

```
Sage] AT
```

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 5 & 7 & 8 \end{bmatrix}$$

```
Sage] A = AT.transpose()
```

```
Sage] (AT*A).solve_right(AT*yy)
```

$$\left(\frac{2}{7}, \frac{5}{14}\right)$$

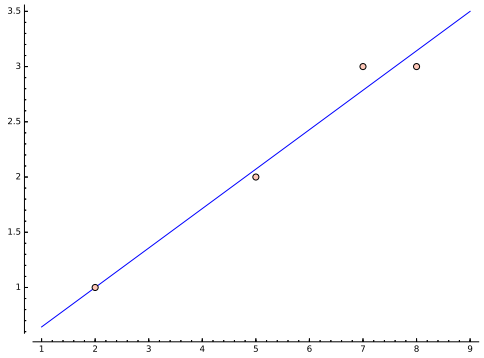
Hence, the least squares line is $y = \frac{2}{7} + \frac{5}{14}x$. Let's plot it again to marvel at the good fit!

```
Sage] points = zip(xx,yy)
```

```
Sage] points
```

$$[(2,1),(5,2),(7,3),(8,3)]$$

```
Sage] scatter_plot(points) + plot(2/7+5/14*x,1,9)
```



Alternative. Here is an alternative (more high-level) approach in Sage, which is using numerics and scipy. Note that this agrees with our symbolic answer.

```
Sage] var('a,b');
Sage] linear_model(x) = a+b*x
Sage] find_fit(points, linear_model)
```

$[a = 0.285714269997, b = 0.357142859662]$

```
Sage] n(2/7)
```

0.285714285714286

```
Sage] n(5/14)
```

0.357142857142857

How good is our fit? How well does the line $y = \frac{2}{7} + \frac{5}{14}x$ fit the data $(2,1),(5,2),(7,3),(8,3)$?

- **residual sum of squares**: $\mathrm{SS}_{\mathrm{res}} = \sum \underbrace{(y_i - (\beta_1 + \beta_2 x))}_{\text{error at } (x_i, y_i)}^2$ [This is what we are minimizing!]
- **total sum of squares**: $\mathrm{SS}_{\mathrm{tot}} = \sum (y_i - \bar{y})^2$ [$\bar{y} = \frac{1}{n}\sum y_i$ is the mean of the observed data]
- **coefficient of determination**: $R^2 = 1 - \frac{\mathrm{SS}_{\mathrm{res}}}{\mathrm{SS}_{\mathrm{tot}}}$

General rule: the closer R^2 is to 1, the better the regression line fits the data.

Here, $\bar{y} = 9/4$ and $R^2 = 1 - \frac{(1-(\frac{2}{7}+\frac{5}{14}2))^2 + (2-(\frac{2}{7}+\frac{5}{14}5))^2 + (3-(\frac{2}{7}+\frac{5}{14}7))^2 + (3-(\frac{2}{7}+\frac{5}{14}8))^2}{(1-\frac{9}{4})^2 + (2-\frac{9}{4})^2 + (3-\frac{9}{4})^2 + (3-\frac{9}{4})^2} = \frac{75}{77} \approx 0.974$.

This is very close to 1 and indicates that we have a good fit. Let's see how to ask Sage to do this computation.

```
Sage] mean(yy)
```

$\frac{9}{4}$

```
Sage] SS_tot = sum((y-9/4)^2 for y in yy)
Sage] SS_tot
```

$\frac{11}{4}$

```
Sage] SS_res = sum((y-(2/7+5/14*x))^2 for x,y in zip(xx,yy))
Sage] SS_res
```

$\frac{1}{14}$

```
Sage] R_sq = 1 - SS_res/SS_tot
Sage] R_sq
```

$\frac{75}{77}$

```
Sage] R_sq.n()
```

0.974025974025974

Can you explain why we do not just $\mathrm{SS}_{\mathrm{res}}$ as a measure for how good our fit is? (What if there is more points?)

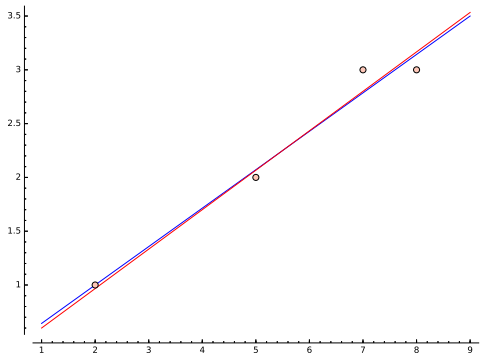
Comment. We get a (slightly) different "best fit" line if we change the role of x and y! Can you explain that?

```
Sage] xx = vector([2,5,7,8]); yy = vector([1,2,3,3])
Sage] AT = matrix([[1,1,1,1], yy])
Sage] A = AT.transpose()
Sage] (AT*A).solve_right(AT*xx)
```

$$\left(-\frac{7}{11}, \frac{30}{11}\right)$$

Note that $x = -\frac{7}{11} + \frac{30}{11}y$ is equivalent to $y = \frac{7}{30} + \frac{11}{30}x$.

```
Sage] scatter_plot(zip(xx,yy)) + plot(2/7+5/14*x,1,9) + plot(7/30+11/30*x,1,9,
      color='red')
```



The explanation is that (see pictures at the beginning of this class) we are minimzing vertical offsets in one case and horizontal offsets in the other case.

In linear regression, the relationship between a dependent variable and one or more explanatory variables is modeled. If y is the dependent variable, with x the explanatory variable, then it is natural to minimize the error we make in "predicting y through x" (vertical offsets).

4.3 Application: Fitting data to other curves

We can also fit the experimental data (x_i, y_i) using other curves.

Example 49. Set up a linear system to find values for the parameters a, b, c that result in the quadratic curve $y = a + bx + cx^2$ that best fits some given points $(x_1, y_1), (x_2, y_2), \ldots$

Solution. $y_i \approx a + bx_i + cx_i^2$ with parameters a, b, c.

The equations $y_i = a + bx_i + cx_i^2$ in matrix form:

$$\underbrace{\begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ \vdots & \vdots & \vdots \end{bmatrix}}_{\text{design matrix } A} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix}}_{\substack{\text{observation} \\ \text{vector } \boldsymbol{y}}}$$

Again, we determine values for a, b, c by computing a least squares solution to that system.

That is, we need to solve the system $A^T A \begin{bmatrix} a \\ b \\ c \end{bmatrix} = A^T \boldsymbol{y}$.

Example 50. Use Sage to find values for the parameters a, b, c that result in the quadratic curve $y = a + bx + cx^2$ that best fits the points $(0,1), (1,2), (2,3), (3,-4), (4,-7), (5,-12)$.

Solution. We first input the points:

```
Sage] xx = vector([0..5])
```

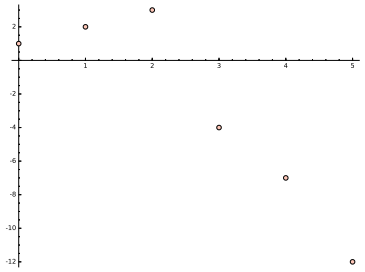
```
Sage] yy = vector([1,2,3,-4,-7,-12])
```

```
Sage] points = zip(xx,yy)
```

```
Sage] points
```

$$[(0,1), (1,2), (2,3), (3,-4), (4,-7), (5,-12)]$$

```
Sage] scatter_plot(points)
```



Option 1: using `find_fit`. This works exactly as in the line fitting case and results in a numerical answer:

```
Sage] var('a,b,c');
```

```
Sage] quadratic_model(x) = a+b*x+c*x^2
```

```
Sage] find_fit(points, quadratic_model)
```

$$[a = 1.5, b = 1.27857142857, c = (-0.821428571433)]$$

Option 2: doing the linear algebra. We set up the system described in the previous example.

```
Sage] AT = matrix([[1,1,1,1,1,1], xx, [t^2 for t in xx]])
```

```
Sage] A = AT.transpose()
```

```
Sage] abc = (AT*A).solve_right(AT*yy)
```

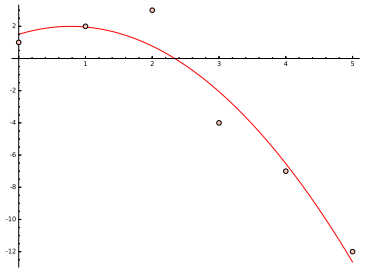
```
Sage] abc
```

$$\left(\frac{3}{2}, \frac{179}{140}, -\frac{23}{28}\right)$$

```
Sage] abc.n()
```

$$(1.50000000000000,\ 1.27857142857143,\ -0.821428571428571)$$

```
Sage] scatter_plot(points) + plot(abc[0]+abc[1]*x+abc[2]*x^2,0,5,color='red')
```

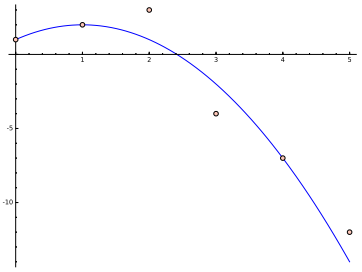


Comment. The initial points were randomly generated as follows. Can you figure out what's happening?

```
Sage] f(x) = 1+2*x-x^2
Sage] xx = vector([0..5])
Sage] yy = vector([f(t)+randint(-2,2) for t in xx])
Sage] points = zip(xx,yy)
Sage] points
```

$[(0,1),(1,2),(2,3),(3,-4),(4,-7),(5,-12)]$

```
Sage] scatter_plot(points) + plot(f,0,5)
```



Example 51. (homework) Find values for the parameters a, b, c such that $y = a + bx + cx^2$ best fits the points $(0,2), (1,3), (3,1), (4,1)$. If working by hand, just set up the system.

Solution. (final answer only) The system to be solved is $\begin{bmatrix} 4 & 8 & 26 \\ 8 & 26 & 92 \\ 26 & 92 & 338 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 7 \\ 10 \\ 28 \end{bmatrix}$.

4.4 Application: multiple linear regression

> *In statistics,* ***linear regression*** *is an approach for modeling the relationship between a scalar dependent variable and one or more explanatory variables.*
>
> *The case of one explanatory variable is called simple linear regression.*
>
> *For more than one explanatory variable, the process is called multiple linear regression.*
>
> *http://en.wikipedia.org/wiki/Linear_regression*

The experimental data might be of the form (x_i, y_i, z_i), where now the dependent variable z_i depends on two explanatory variables x_i, y_i (instead of just x_i).

Example 52. (homework) Set up a linear system to find values for the parameters a, b, c such that $z = a + bx + cy$ best fits some given points $(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots$

Solution. The equations $a + bx_i + cy_i = z_i$ translate into the system:

$$\underbrace{\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \\ \vdots & \vdots & \vdots \end{bmatrix}}_{\text{design matrix } A} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \underbrace{\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \end{bmatrix}}_{\substack{\text{observation} \\ \text{vector } \boldsymbol{z}}}$$

Of course, this is usually inconsistent. To find the best possible a, b, c we compute a least squares solution by solving $A^T A \begin{bmatrix} a \\ b \\ c \end{bmatrix} = A^T \boldsymbol{z}$.