The Gauss-Seidel method

We again employ the decomposition $A = D - L - U$

$$Ax = b$$

$$(D - L - U)x = b$$

$$(D - L)x = Ux + b$$

At this point, we place $x^{(k+1)}$ on the LHS and $x^{(k)}$ on the RHS

$$\boxed{(D - L)x^{(k+1)} = Ux^{(k)} + b} \qquad (1)$$

The benefit of the Gauss-Seidel method (1) over Jacobi is the improved convergence, which is guaranteed not only for diagonally dominant matrices, but also for symmetric and positive definite matrices

In terms of complexity, each iteration of (1) amounts to solving a lower-triangular system via forward substitution, i.e. incurs a cost $O(k)$, k = # of nonzero entries in A.

Once again, form (1) is useful for proofs, while the pseudocode version is given as:

(Without in-place update)

$\underline{x}^{(0)} \leftarrow$ initial guess

for $k = 0, 1, \ldots,$ <max>

  for $i = 1, 2, \ldots, n$

$$x_i^{(k+1)} \leftarrow \frac{1}{a_{ii}} \left(b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)} \right)$$

  end

  <check for convergence

end

(In-place)

$\underline{x} \leftarrow$ initial guess

for $k = 0, 1, \ldots,$ <max>

  for $i = 1, 2, \ldots, n$

$$x_i \leftarrow \frac{1}{a_{ii}} \left(b_i - \sum_{i \neq j} a_{ij} x_j \right) \quad (*)$$

  end

  <check for convergence>

end

Compare: Jacobi

$\underline{x} \leftarrow$ initial guess

for $k = 0, 1, \ldots,$ <max>

  for $i = 1, 2, \ldots, n$

$$x_i^{new} \leftarrow \frac{1}{a_{ii}} \left(b_i - \sum_{i \neq j} a_{ij} x_j \right) \quad (**)$$

  end

$\underline{x} \leftarrow x^{new}$ (***)

end

The real difference in performance is:

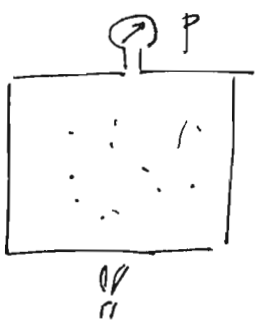
Gauss-Seidel is generally serial in nature (although parallel variants exist), while steps (**) & (***) are highly parallel.

Overdetermined systems

So far, we considered linear systems $Ax = b$ with the same number of equations & unknowns (i.e. $A \in \mathbb{R}^{n \times n}$).

In the case where $A \in \mathbb{R}^{m \times n}$, with $m > n$ (more equations), the existence of a true solution is not guaranteed; in this case we look for the "best possible" substitute for a solution. Before analyzing what that means, let's look at how such problems arise:

e.g. In an experiment, we measure the pressure of a gas in a closed container, as a function of the temperature



From physics:

$$pV = nR\,\frac{5}{9}\,(T + 459.67)$$

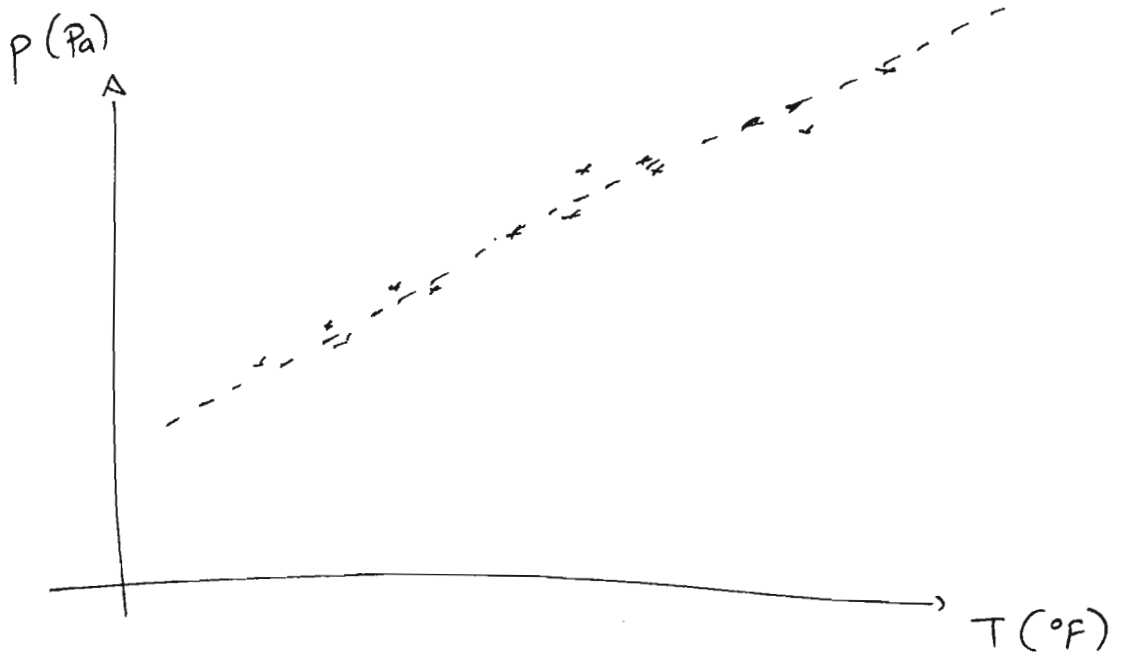
T: Fahrenheit.

i.e $p = \alpha T + \beta$

$$\left(\alpha = \frac{5nR}{9V},\ \beta = \frac{5nR \cdot 459{,}67}{9V}\right)$$

What are α & β?

experimentally:



The measurements <u>should ideally</u> lie on a straight line $y = c_1 x + c_0$, but do not, due to measurement error. If we have n measurement pairs $(x_1, y_1), \ldots, (x_n, y_n)$ we would have wanted:

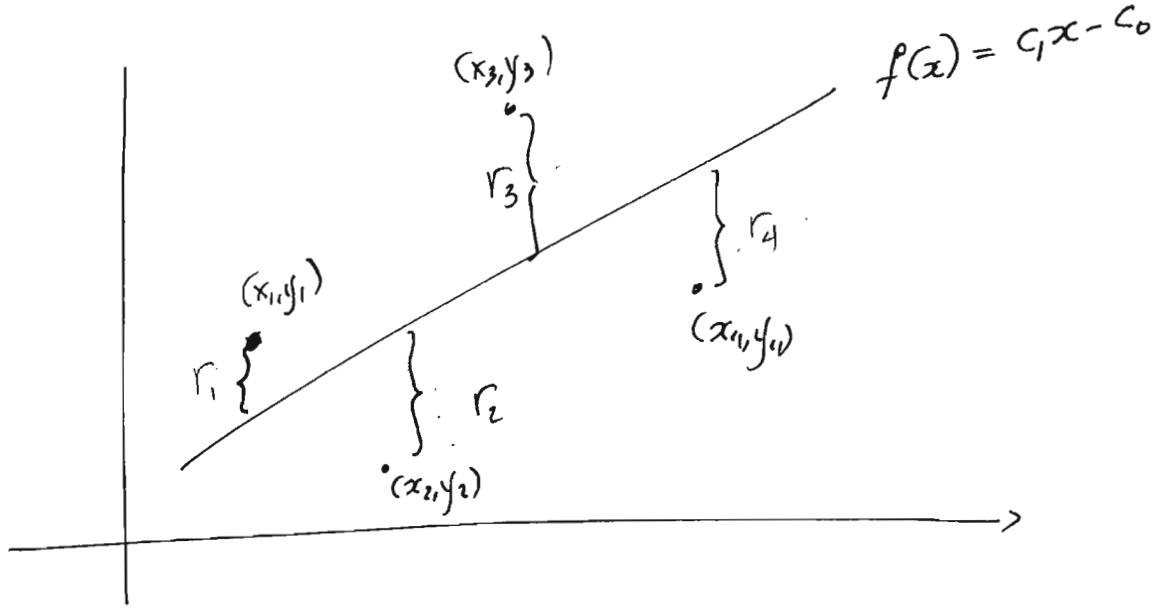
$$\left.\begin{aligned} y_1 &= c_1 x_1 + c_0 \\ y_2 &= c_1 x_2 + c_0 \\ &\vdots \\ y_n &= c_1 x_n + c_0 \end{aligned}\right\} \Rightarrow \underbrace{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \\ x_n & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} c_1 \\ c_0 \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}}_{b}$$

Here $Ax = b$ is a rectangular system.

$n \times 2$ $\quad 2 \times 1$ $\quad n \times 1$

We cannot hope to find a true solution to this system. Instead let's try to find an "approximate" solution, such that $Ax \approx b$.

Let's look at the residual of this "interpolation"



the residual of the approximation at each data point is

$$r_i = y_i - f(x_i) = y_i - c_1 x_i - x_0$$

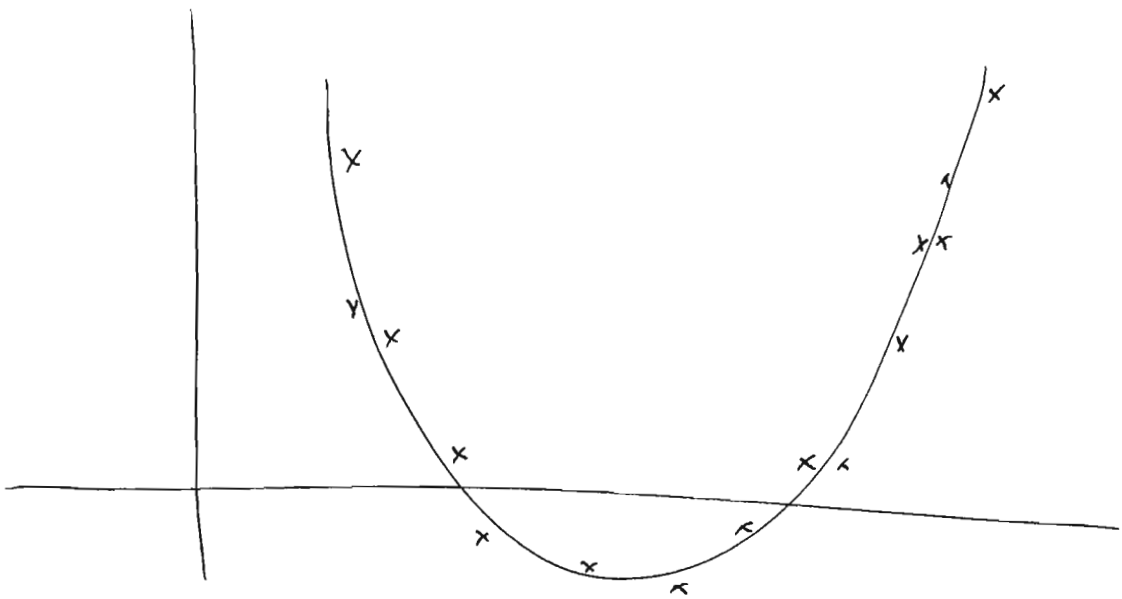
If we write the vector of all residuals:

$$\underline{r} = \begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix} = \begin{bmatrix} y_1 - c_1 x_1 - c_0 \\ y_2 - c_1 x_2 - c_0 \\ \vdots \\ y_n - c_1 x_n - c_0 \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} - \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \\ x_n & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_0 \end{bmatrix} = \underline{b} - A\underline{x}$$

Although we can't find an $\underline{x}$ s.t. $A\underline{x} = \underline{b}$ (thus $\underline{r} = 0$) we can at least try to make $\underline{r}$ small.

Another example

Find the best parabola $f(x) = c_2x^2 + c_1x + c_0$ that fits measurements $(x_1, y_1), \ldots, (x_n, y_n)$



We would like

$$\left.\begin{array}{l} f(x_1) \approx y_1 \\ f(x_2) \approx y_2 \\ \vdots \\ f(x_n) \approx y_n \end{array}\right\} \quad \left.\begin{array}{l} c_2x_1^2 + c_1x_1 + c_0 \approx y_1 \\ c_2x_2^2 + c_1x_2 + c_0 \approx y_2 \\ \vdots \\ c_2x_n^2 + c_1x_n + c_0 \approx y_n \end{array}\right\} \quad \underbrace{\begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ & \vdots & \\ x_n^2 & x_n & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} c_2 \\ c_1 \\ c_0 \end{bmatrix}}_{x} \approx \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}}_{b}$$

Once again we would like to make $\underline{r} = \underline{b} - A\underline{x}$ as small as possible.

How do we quantify $\underline{r}$ being small?

⇒ Using a norm!

We could ask that → $\|\underline{r}\|_1$, $\|\underline{r}\|_2$, or $\|\underline{r}\|_\infty$ be as small as possible

Any of these norms would be intuitive to consider for minimization (esp. 1- and ∞-norms are very intuitive).

However, we typically use the 2-norm for this purpose, because it's the easiest to work with in this problem!

Def The <u>least squares solution</u> of the overdetermined system $A\underline{x} \approx \underline{b}$ is the vector $\underline{x}$ that minimizes

$$\|\underline{r}\|_2 = \|\underline{b} - A\underline{x}\|_2$$

Define $Q(\underline{x}) = Q(x_1, x_2, \ldots, x_n)$

$= \| \underline{b} - A\underline{x} \|_2^2$ where $\underline{x} = (x_1, \ldots, x_n)$

and $A \in \mathbb{R}^{m \times n}$ $b \in \mathbb{R}^m$ $(m > n)$. The least squares solution is the set of values $x_1, \ldots, x_n$ that minimize $Q(x_1, \ldots, x_n)$!

$$Q(x_1, \ldots, x_n) = \| \underline{b} - A\underline{x} \|_2^2 = \| \underline{r} \|_2^2 = \sum_{i=1}^{m} r_i^2$$

$$\underline{r} = \underline{b} - A\underline{x} \Rightarrow r_i = b_i - [Ax]_i \Rightarrow r_i = b_i - \sum a_{ij} x_j$$

$$\Rightarrow Q(x_1, \ldots, x_n) = \sum_{i=1}^{m} \left(b_i - \sum_{j=1}^{n} a_{ij} x_j \right)^2.$$

If $x_1, \ldots, x_n$ are those that minimize Q, then:

$$\frac{\partial Q}{\partial x_1} = 0$$

$$\frac{\partial Q}{\partial x_2} \ldots = 0$$

$$\vdots$$

$$\frac{\partial Q}{\partial x_n} = 0.$$

} in order to guarantee a minimum

$$\frac{\partial Q}{\partial x_k} = \frac{\partial}{\partial x_k}\left(\sum_{i=1}^{m}\left(b_i - \sum_{j=1}^{n} a_{ij}x_j\right)^2\right)$$

$$= \sum_{i=1}^{m} \frac{\partial}{\partial x_k}\left(b_i - \sum_{j=1}^{n} a_{ij}x_j\right)^2$$

$$= \sum_{i=1}^{m} 2\underbrace{\left(b_i - \sum_{j=1}^{n} a_{ij}x_j\right)}_{r_i} \frac{\partial}{\partial x_n}\left(b_i - \sum_{j=1}^{n} a_{ij}x_j\right)$$

$$= \sum_{i=1}^{m} r_i a_{ik} = \sum_{i=1}^{m} [A^T]_{ki} r_i = [A^T r]_k$$

Thus:

$$\left.\begin{array}{l} \frac{\partial Q}{\partial x_1} = 0 \Rightarrow [A^T r]_1 = 0 \\ \vdots \\ \frac{\partial Q}{\partial x_n} = 0 \Rightarrow [A^T r]_n = 0 \end{array}\right\} \quad \boxed{A^T \underline{r} = \underline{0}}$$

Since $\underline{r} = \underline{b} - A\underline{x}$, we have:

$$0 = A^T r = A^T(b - Ax) = A^T b - A^T A x \Rightarrow$$

$$\Rightarrow \boxed{A^T A \underline{x} = A^T b}$$

The system above is called the normal equations system

it is a square system, that has as solution the least-squares approximation of $Ax \approx b$

$$\underbrace{A^T}_{n\times m}\underbrace{A}_{m\times n}\underbrace{x}_{n\times 1} = \underbrace{A^T}_{n\times m}\underbrace{b}_{m\times 1}$$

$$n\times n \quad n\times 1 \qquad n\times 1$$

The normal equations always have a solution (with the simple condition that the columns of A have to be linearly independent (usually true)).

Problem: The condition number of A^TA is the square of that of A (if A was square itself!).