Conjugate Gradients (CG)

A $n \times n$

A symmetric positive definite

Since A is spd, it gives a norm

$\|\vec{x}\|_A = (\vec{x}^T A \vec{y})^{1/2}$ "A-norm"

CG has the following property:
In each iteration k, it finds $x_k \in \mathcal{K}_k$
that minimizes the A-norm of the
error e_k. I.e., $\|e_k\|_A = \min\limits_{x_k \in \mathcal{K}_k}$

Solution x^* $\quad Ax^* = b$

$$\min_{x_k \in \mathcal{K}_k} \quad (x^* - x_k)^T A (x^* - x_k)$$

$$\phi(x) = \frac{1}{2} x^T A x - b^T x + c$$

$$\delta\phi = \frac{1}{2} \delta x^T A x + \frac{1}{2} x^T A \delta x - b^T \delta x$$

$$\nabla\phi = Ax - b$$

$$\begin{aligned}
\phi((x-x^*) + x^*) &= \frac{1}{2}(x-x^*)^T A (x-x^*) \\
&\quad + \frac{1}{2} x^{*T} A x^* + (x-x^*)^T A x^* \\
&\quad - b^T(x-x^*) - b^T x^* + c \\
\langle Ax^* = b \rangle \quad &= \frac{1}{2} e^T A e + \frac{1}{2} b^T x^* + \cancel{e^T b - b^T e} - b^T x^* + c \\
\phi(x^* + e) \quad &= \frac{1}{2} e^T A e - \frac{1}{2} b^T x^* + c \\
&= \frac{1}{2} e^T A e + \text{constant}
\end{aligned}$$

```
$x_0$ , $r_0 = b - Ax_0$ , $s_0 = r_0$

for $k = 0, 1, 2, \ldots$
    $\alpha_k = ?$
    $x_{k+1} = x_k + \alpha_k s_k$
    $r_{k+1} = r_k - \alpha_k A s_k$
    $s_{k+1} = ?$
end
```

C.G.

$\boxed{\text{Step size}}$ α_k

$f(x_k + \alpha_k s_k)$ one-dim. minimization

$$\phi(\alpha_k) = f(x_k + \alpha_k s_k)$$

$$\frac{d\phi}{d\alpha}(\alpha_k) = \nabla f(x_k + \alpha_k s_k)^T s_k = 0$$

For C.G., $\nabla f(x) = b - Ax = r$

$$\Rightarrow \left(\frac{d\phi}{d\alpha}(\alpha_k) = 0\right) \Rightarrow$$

$$r_{k+1}^T s_k = \left[b - A(x_k + \alpha_k s_k)\right]^T s_k$$

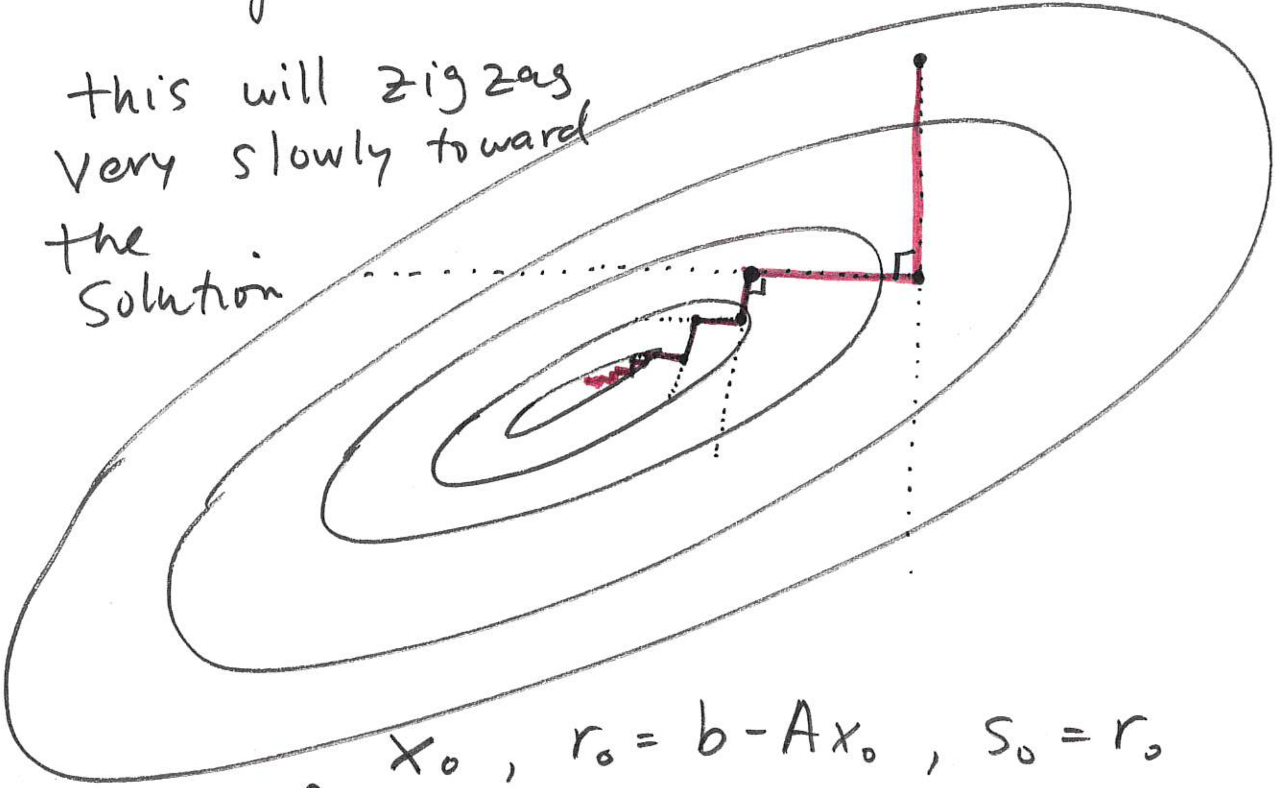
$$= b^T s_k - x_k^T A^T s_k - \alpha_k s_k^T A^T s_k = 0$$

$$= (b - Ax_k)^T s_k - \alpha_k s_k^T A^T s_k = 0$$

$$\Rightarrow \boxed{\alpha_k = \frac{r_k^T s_k}{s_k^T A s_k}}$$

Steepest Descent Method

or gradient descent



this will zigzag very slowly toward the solution

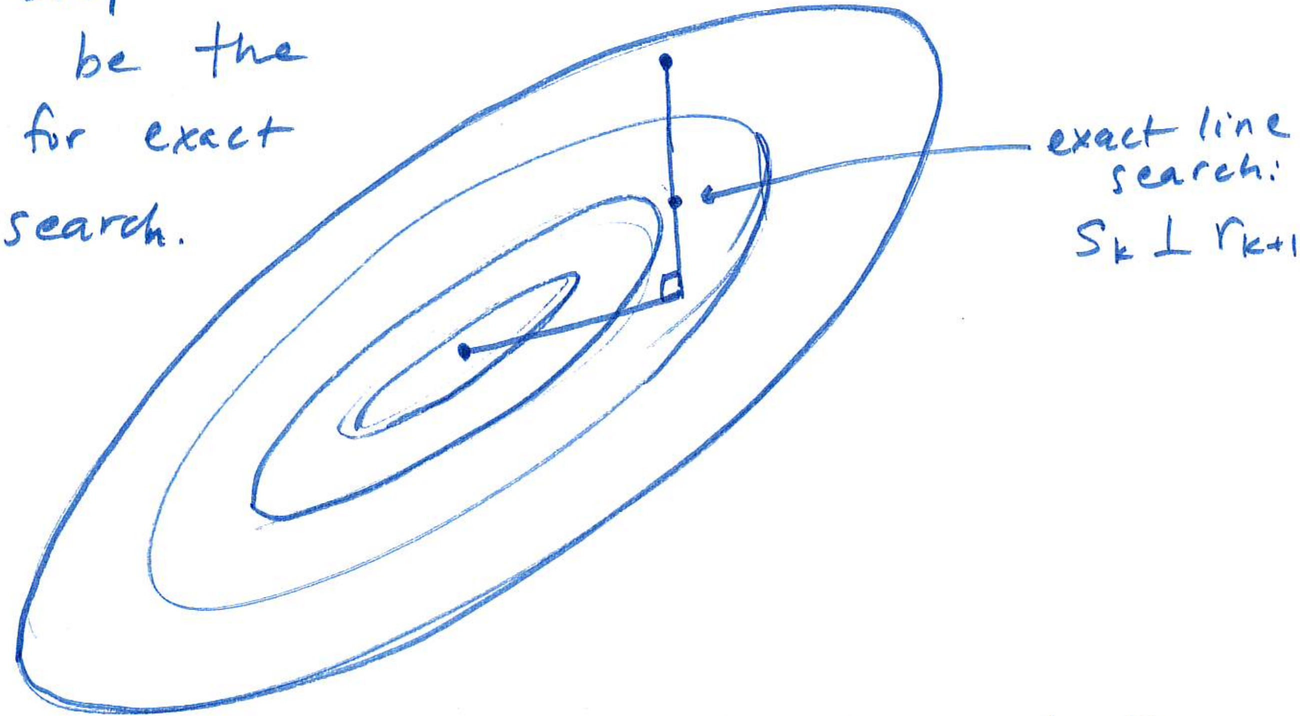
```
$x_0$ , $r_0 = b - Ax_0$ , $s_0 = r_0$
for $k = 0, 1, 2, \ldots$
    $\alpha_k = \frac{r_k^T s_k}{s_k^T A s_k}$
    $x_{k+1} = x_k + \alpha_k s_k$
    $s_{k+1} = s_k - \alpha_k A s_k$
end
```

Progress

$$\frac{\phi(x_k) - \phi(x^*)}{\phi(x_{k-1}) - \phi(x^*)} \leq 1 - \frac{1}{\text{Cond} A}$$

What if we try s_k mutually orthogonal?

If we tried to do n steps with orthogonal directions, as illustrated, the step size we would need would not be the one for exact linesearch.



Mathematically, we want to start at x_0 and reach x^* in n steps along s_k:

$$x^* = x_0 + \sum_{k=0}^{n-1} \alpha_k s_k$$

Let $e_0 = x_0 - x^*$ & $e_k = x_k - x^*$

Find α_k:

$$e_0 = x_0 - x^* = -\sum_{k=0}^{n-1} \alpha_k s_k$$

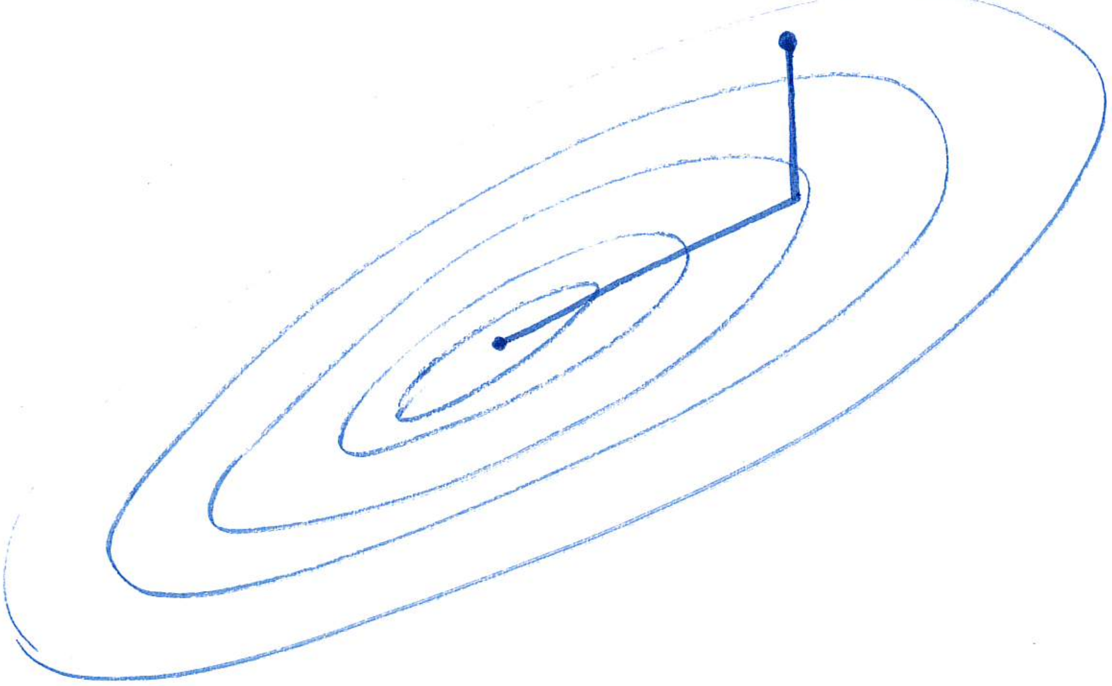
$$s_j^T e_0 = -\sum_{k=0}^{n-1} \alpha_k s_j^T s_k = \langle \text{ assume } s_j^T s_k = 0, j \neq k)$$

$$= -\alpha_j s_j^T s_j$$

$$\Rightarrow \boxed{\alpha_j = \frac{s_j^T e_0}{s_j^T s_j}} = \frac{s_j^T e_k}{s_j^T s_j}$$

Problem: we don't know e_0 or e_k.

What about A-orthogonal directions s_k ?



$$e_0 = -\sum_{k=0}^{n-1} \alpha_k s_k$$

$$s_j^T A e_0 = -\sum_{k=0}^{n-1} \alpha_k s_j^T A s_k$$

$$= -\alpha_j s_j^T A s_j$$

$$\Rightarrow \quad \alpha_j = -\frac{s_j^T A e_0}{s_j^T A s_j} = \frac{-s_j^T A e_j}{s_j^T A s_j} = \frac{s_j^T r_j}{s_j^T A s_j}$$

this is a quantity we know, and it is the one given by exact line search along s_j

But what are the search directions s_k?

A-orthogonal search directions will allow us to terminate in n steps:

$$e_0 = -\sum_{i=0}^{n-1} \alpha_i s_i$$

$$s_k^T A e_0 = -\sum_{i=0}^{n-1} \alpha_i s_k^T A s_i = -\alpha_k s_k^T A s_k$$

$$\Rightarrow \quad \alpha_k = -\frac{s_k^T A e_0}{s_k^T A s_k} = \frac{-s_k^T A e_k}{s_k^T A s_k} = \frac{s_k^T r_k}{s_k^T A s_k}$$

$$x_0 + e_0 = x \quad , \quad e_0 = x - x_0$$

$$x_0 + \sum_{k=0}^{n-1} \alpha_k s_k = x$$

Create the search directions through Gram-Schmidt A-orthogonalization

$u_0, \dots, u_{n-1}$

$$s_i = u_i + \sum_{j=0}^{i-1} \beta_{ij} s_j$$

$$0 \le k < i, \; s_k^T A s_i = s_k^T A u_i + \sum_{j=0}^{i-1} \beta_{ij} s_k^T A s_j = 0$$

$$\Rightarrow \quad \beta_{ik} = \frac{-s_k^T A u_i}{s_k^T A s_k}$$

need to keep all old $\vec{s}_i$ in memory, & orthogonalize against all.

$$S_k = u_k + \sum_{j=0}^{k-1} \beta_{kj} \vec{s}_j$$

Gram-Schmidt A-orthogonalization:

$$\boxed{i < k} \Rightarrow$$

$$0 = S_i^T A S_k = s_i^T A u_k + \sum_{j=0}^{k-1} \beta_{kj}\, s_i^T A s_j$$

$$= s_i^T A u_k + \beta_{ki}\, s_i^T A s_i$$

$$\Rightarrow \quad \beta_{ki} = \frac{-s_i^T A u_k}{s_i^T A s_i}$$

CG: use Krylov subspaces. use residuals instead of arbitrary u_k, and A-orthogonalize residuals:

$$\beta_{ki} = \frac{-s_i^T A r_k}{s_i^T A s_i} = -\frac{r_k^T (A s_i)}{s_i^T A s_i}$$

$$= \left\langle A s_i = \frac{1}{\alpha_i}(r_i - r_{i+1}) \right\rangle$$

$$= \frac{r_k^T (r_{i+1} - r_i)}{\alpha_i\, s_i^T A s_i} = \frac{r_k^T r_{i+1} - \cancel{r_k^T r_i}}{\alpha_i\, s_i^T A s_i} \xrightarrow{} 0 \; \langle \text{residuals orthogonal} \rangle \; ① \quad = \frac{r_k^T r_{i+1}}{r_i^T r_i} \; ②$$

$$= \begin{cases} \dfrac{r_k^T r_k}{r_{k-1}^T r_{k-1}} & i = k-1 \\ 0 & \text{otherwise} \end{cases}$$

For ① + ② see next page

There is only one non-zero $\beta_{ki} \triangleq \beta_k$

① residuals orthogonal

$$r_k^T s_j = 0 \, , \quad j < k$$

$$r_k^T s_j = -(Ae_k)^T s_j$$

$$= -s_j^T A \left[- \sum_{i=k}^{n-1} \alpha_i s_i \right] = 0$$

$$s_j = r_j + \sum_{i=0}^{j-1} \beta_{ji} s_i$$

So

$$r_k^T r_j = r_k^T \left[s_j - \sum_{i=1}^{j-1} \beta_{ji} s_i \right] = 0$$

② $\alpha_i s_i^T A s_i = r_i^T r_i$

$$r_{i+1} = r_i - \alpha_i A s_i \Rightarrow \alpha_i A s_i = r_i - r_{i+1}$$

$$\alpha_i s_i^T A s_i = s_i^T \left[r_i - r_{i+1} \right]$$

$$= \left(r_i + \sum_{j=0}^{i-1} \beta_{ij} s_j \right)^T \left(r_i - r_{i+1} \right) = r_i^T r_i$$

Conjugate Gradients for $Sx = b$

applies to s.p.d. matrices S

- in theory gives the exact result in n steps
- in practice, gives good result in much fewer than n steps.

- residuals are orthogonal $r_k^T r_j = 0$
- search directions are A-orthogonal $s_k^T A s_j = 0$

C.G.

```
$x_0$ = initial guess
$r_0 = b - Ax_0$
$s_0 = r_0$
for $k = 0, 1, 2, \ldots$
    $\alpha_k = r_k^T r_k / s_k^T A s_k$
    $x_{k+1} = x_k + \alpha_k s_k$
    $r_{k+1} = r_k - \alpha_k A s_k$
    $\beta_{k+1} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
    $s_{k+1} = r_{k+1} + \beta_{k+1} s_k$
end
```

Convergence of CG

$$\| x - x_k \|_A \le 2 \, \| x - x_0 \|_A \left(\frac{\sqrt{\lambda_{max}} - \sqrt{\lambda_{min}}}{\sqrt{\lambda_{max}} + \sqrt{\lambda_{min}}} \right)^k$$

$$K_2(A) = \frac{\lambda_{max}}{\lambda_{min}}$$

$$\| x - x_k \|_A \le 2 \, \| x - x_0 \|_A \left(\frac{\sqrt{K} - 1}{\sqrt{K} + 1} \right)^k$$

Preconditioning $Ax = b$

Find P close to A and solve

$$P^{-1}Ax = P^{-1}b$$

with the idea that solver converges more quickly on $P^{-1}A$ than on A.

P^{-1} should be reasonable to computed. (Not actually computed, but represents solve $P^{-1}c \Rightarrow y \mid Py = c$)

Frequent choices of P:

1. main diagonal of A (Jacobi)
2. triangular part of A (Gauss-Seidel)
3. $P = L_0 U_0$ incomplete LU, avoid fill-in
4. P = same diff matrix but on coarser grid (multigrid)