# Krylov Methods!

## A short introduction

**Conjugate Gradient**

→ for symmetric systems with positive eigenvalues

**MINRES**

→ symmetric systems with nonzero eigenvalues

**GMRES**

→ works with any invertible A

→ developed by Yousuf Saad, local celebrity at U. Minnesota.

**BICG-STAB**

→ works with any invertible A

→ All of these methods start from an initial guess $x_0$ for $Ax = b$.

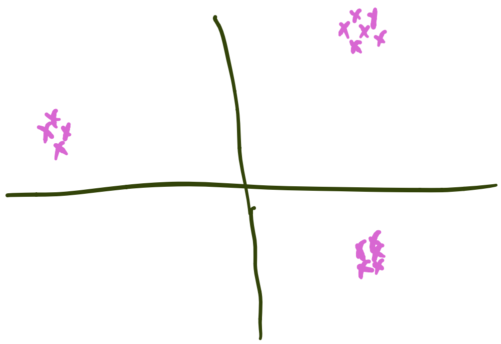
→ At each step, the new guess is the best within the <u>Krylov subspace</u>

$A \in \mathbb{R}^{n\times n}$

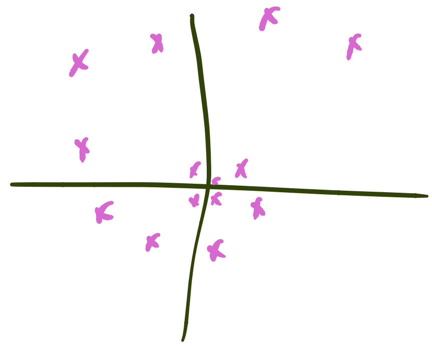
$m < n$ $$K_m = \text{span}\{x_0, Ax_0, A^2x_0, \dots, A^m x_0\}.$$

→ All you need is a function that accepts a vector $v$ and returns $Av$ (you don't need to store the entries of $A$ if you have some other way of computing $Av$ from $v$).

→ These methods generally work quickly if the eigenvalues of $A$ are clustered together and away from $0$, in the complex plane:



Good eigenvalue distribution



Bad eigenvalue distribution

$\longrightarrow$ <u>Preconditioning</u> is important:

suppose $P$ is vaguely like $A$, but $P^{-1}$ is easy; then solve

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

and $P^{-1}A \approx I$ will have nicely clumped eigenvalues and fast convergence properties.

Today we'll look more at CG (conjugate gradients).

Main idea is to view $Ax=b$ as an optimization problem:

If $f(x) = \frac{1}{2}x^T A x - b^T x$

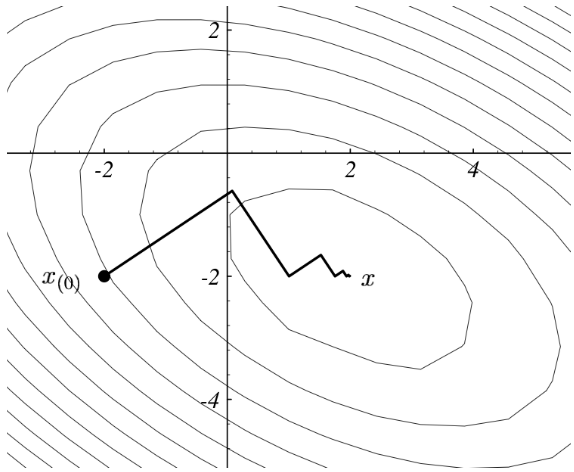
Then $f$ is a quadratic, opening upward;

minimum is when $\nabla f = 0$

or $Ax = b$.

$$\boxed{\begin{array}{c} \dfrac{\partial f}{\partial x_1} = 0 \\ \dfrac{\partial f}{\partial x_2} = 0 \ \ldots \end{array}}$$

One idea is "gradient descent," On contour plot of $f(x)$:




"Always go downhill"

The CG idea is a clever variation on this. Idea:

If $\vec{x}_{n+1} = \vec{x}_n + \vec{c}_n$,

$\vec{c}_n$ is the <u>$n^{th}$ step</u>

It would be nice to have

$\vec{c}_2^{\,T} \vec{c}_1 = 0$ , $\vec{c}_3^{\,T} c_2 = 0 = \vec{c}_3^{\,T} c_1$

and so on.

No one knows how to do this

but we can arrange for

$c_2^T A c_1 = 0$

Vocab: $c_2$, $c_1$ are A-conjugate directions.

$c_3^T A c_2 = 0 = c_3^T A c_1$

The algorithm

$\mathbf{x}_0 =$ initial guess
$\mathbf{d}_0 = \mathbf{b} - A\mathbf{x}_0$ (initial direction)
$\mathbf{r}_0 = \mathbf{d}_0$ (initial residual)

**for** $k = 0, 1, 2, 3, \ldots, n-1$

if ($\mathbf{r}_k = 0$) **stop**

$\alpha_k = (\mathbf{r}_k^T \mathbf{r}_k)/(\mathbf{d}_k^T A\mathbf{d}_k)$ new step length Crux of CG!

$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ take step

$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{d}_k$ new residual

$\beta_k = (\mathbf{r}_{k+1}^T \mathbf{r}_{k+1})/(\mathbf{r}_k^T \mathbf{r}_k)$

$\mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$ new search direction reminder,

this needs symmetric

and positive-definite $A$.