Lecture 28

More large-scale optimization

- Solving systems with the Hessian
- The conjugate gradient method
- Separable / stochastic optimization
- Alternating optimization

Projects?

Tolerances for ip method

10^{-10} to 10^{-2}

are possible

Large Scale optimization

↳ The key issue is what to do about Hessian! (Curvature info)

$\overset{H_k}{B_k}$ - too big to store as dense Matrix.

LM QN: Store

Sequence of updates to $T_k \overset{\text{ie}}{\sim} B_k^{-1}$

But, maybe you

lucky and you

is

- Sparse
- Structured

So you can store

How to compute

- Structure factor
- Sparse Direct M
- Iterative Methods

get
Hessian

d!

O_k ?

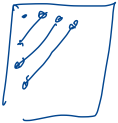
zn

ethod

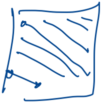
→ if H_k = penta-diagonal Matrix

if H_k = toeplitz / Hankel

Hankel



toeplitz



Sparse + Low-rank

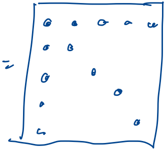
→ Sparse matrices are those w/ Enough zeros to take advantage of!

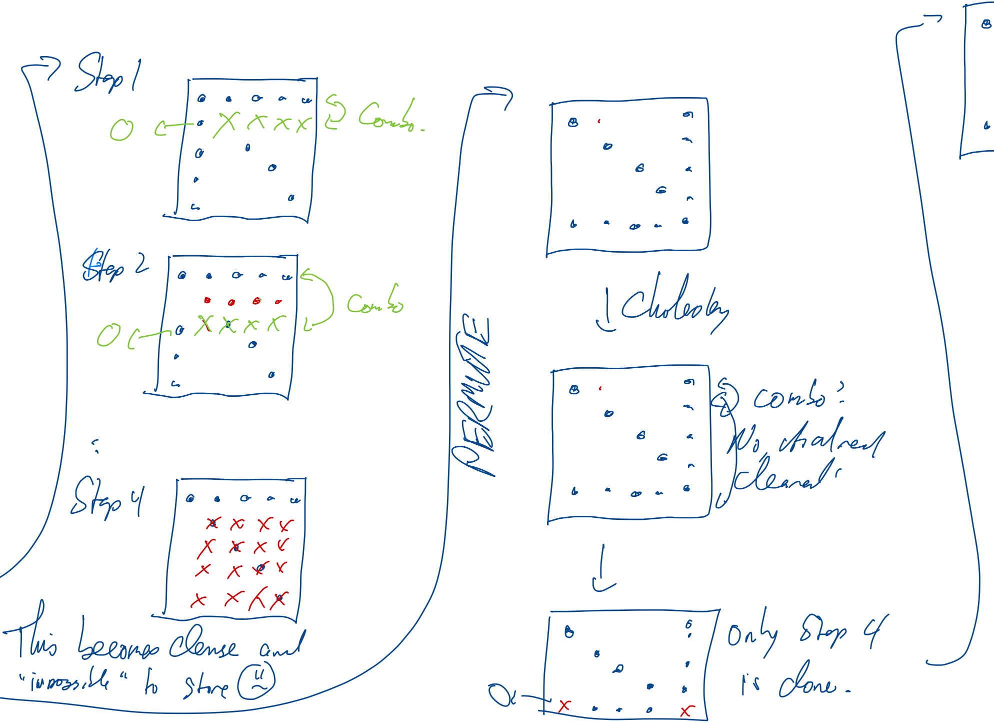
$O(n^2)$ - little-o non-zeros.

Cholesky factorization of Sparse Matrix

let's consider

$H_K =$

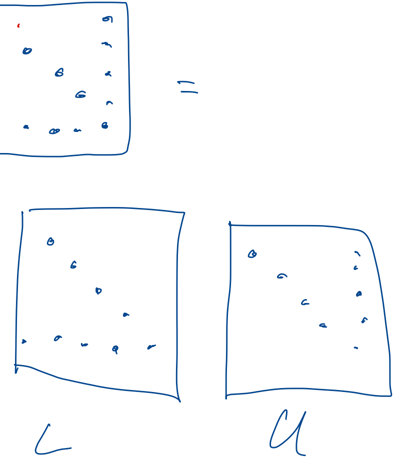




This becomes dense and "impossible" to store ☹

combo? No, already clean!

Only step 4 is done.



or $L \; L^T$ for Cholesky.

w/ full sparsity!

Suppose we can compute

$H_k \cdot v$

efficiently. Then you can use iterative methods

$\downarrow$

it suffices to get a vector $\bar{p}_k$ w/

$$\|H_k \bar{p}_k - g_k\|_2 \le \eta \|g_k\|$$

$0 < \eta < 1$

$\rightarrow C$

G can do this!

Conjugate Gradient
for linear Systems.

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

if $A \succ 0$, $A = A^T$, then

$$\text{argmin } f(x) = A^{-1}\underline{b}$$

$$\iff$$

$$g(x) = Ax - b = 0.$$

In CG, at each step,
we find p_k s.t. it is
A-Conjugate to all
existing search directions.

$$\underline{p}_k^T A \, \underline{p}_j = 0$$

if $k \neq j$ $\iff$

A-conjugate.

Note, residual is

$$\underline{r}_k = A\underline{x}_k - \underline{b} = g_k$$

Alg:

```
$x_0 = 0$
$r_0 = Ax_0 - b$
$p_0 = -r_0$
while $\|r_k\| \geq \tau$   (or $\|r_k\| \geq \tau \|b\|$)
    $\alpha_k$ = opt. line search
      ↳ $\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k}$
    $x_{k+1} = x_k + \alpha_k p_k$.
    $r_{k+1} = r_k + \alpha_k A p_k$
    $\beta_{k+1} = r_{k+1}^T r_{k+1} / r_k^T r_k$
    $p_{k+1} = -r_k + \beta_{k+1} p_k$ ← $k \leftarrow k+1$.
```

Theorem / Lemma

$$p_{k+1}^T A p_k = 0$$

$$\vdots$$

$$p_{k+1}^T A p_{k-1} = 0$$

$$p_{k+1}^T A p_{k-2} = 0$$

$$\vdots$$

$$p_{k+1}^T A p_0 = 0.$$

Also:

$$p_{k+1}^T r_k = 0$$

$$\vdots$$

$$p_{k+1}^T r_0 = 0.$$

for Opt.

$\beta_k \to g(x_k)$

$\alpha_k \to$ Strong Wolfe line search.

$\Rightarrow$ Fletcher-Reeves CG.

---

alt:

$$\beta_{k+1} = \frac{g_{k+1}^T(g_{k+1} - g_k)}{\|g_k\|^2}$$

$\to$ Polak-Ribiere.

Convergence

- Every n iterations, set $\beta_k = 0$.

$\Rightarrow$ Every n steps, do a step of grad. descent.