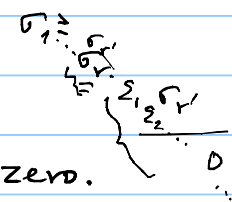
Note Title

Numerical rank & rank-revealing factorizations

Let $A \in \mathbb{R}^{m\times n}$ & suppose $r = \operatorname{rank} A \le \min\{m,n\}$.

If $A = U\Sigma V^T$, then $A = \sum_{k=1}^{r} \sigma_k u_k v_k^T$.

In practice one may not get the computed singular values equal to zero, so a choice of $\hat{r}$ needs to be made so that $\sigma_1, \ldots, \sigma_{\hat{r}}$ are non-zero & σ_i $(i > \hat{r})$ are stipulated to be zero.



Then $\hat{r}$ is called the "numerical rank" of A.
Typically $\hat{r} \le r$.

There are 2 popular ways of determining $\hat{r}$:

1. Using SVD - very reliable, but very sensitive and sometimes computationally expensive.

2. QR with column-pivoting.

[There are additional general rank-revealing factorizations of the form $AZ = QR$, where Z is orthogonal (Ref. Sections 5.4.5 - 5.4.7 of Golub-Loan.]

(I) Numerical rank using SVD (Section 5.4.1 of GL).

Suppose we have, in exact arithmetic, $A = U\Sigma V^T$,

$$\text{So} \quad A = \sum_{k=1}^{r} \sigma_k u_k v_k^T, \quad r = \operatorname{rank} A.$$

Now suppose the computed matrices are $\hat{U}$, $\hat{\Sigma}$ & $\hat{V}$

so that $\hat{U}^T A \hat{V} = \hat{\Sigma} = \text{diag}(\hat{\sigma}_1, \ldots, \hat{\sigma}_n)$

where $\hat{\sigma}_1 \geq \hat{\sigma}_2 \cdots \geq \hat{\sigma}_n \geq 0$

Unless remarkable cancellation occurs, none of the computed singular values will be zero (because of fl. pt. arithmetic)

At this point there are 2 possibilities:

Stick to the strict defn. & treat A as having full rank

i.e. $A \approx \sum_{k=1}^{n} \hat{\sigma}_k \hat{u}_k \hat{v}_k^T$

(But this way you will end up working with every matrix as if it is a full rank matrix, which is not a good idea)

Relax the defn. of rank & set the "small" computed singular values to zero; let $\hat{r}$ = numerical rank

$$A \approx \sum_{k=1}^{\hat{r}} \hat{\sigma}_k \hat{u}_k \hat{v}_k^T$$

$\downarrow$

What do we mean by "small"? This amounts to choosing a tolerance $\delta > 0$ & declaring A to have numerical rank $\hat{r}$ if the computed singular values satisfy

$$\hat{\sigma}_1 \geq \cdots \geq \hat{\sigma}_{\hat{r}} > \delta > \underbrace{\hat{\sigma}_{\hat{r}+1} \geq \cdots \geq \hat{\sigma}_n}_{\text{set these to zero.}}$$

Note: this computation is sensitive to changes in A & the choice of δ.

(II) QR with column pivoting.

This is a modification of the Householder QR procedure.

First: why is column pivoting required?

Consider the foll. situation that can occur if A is not full rank:

$$A = [a_1 \mid a_2 \mid a_3 \mid a_4]_{m\times 4} = \underbrace{[q_1 \mid q_2 \mid q_3 \mid q_4]}_{Q} \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{R.}$$

- rank $R = 3 \neq$ # of non-zero entries on the diagonal of R.

$$= \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ \vdots & & & \\ q_{m1} & & & q_{m4} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- range $A \not\subseteq$ span$\{q_1, q_2, q_3\}$ or span$\{q_1, q_2, q_4\}$ or span$\{q_1, q_3, q_4\}$ or span$\{q_2, q_3, q_4\}$.

$$= \left[\, q_1 \mid q_1 \mid q_1 + q_2 \mid q_1 + q_2 + q_3 + q_4 \,\right]$$

In this case, the QR factorization offers no insight into rank(A), or range(A) or null(A).

In exact arithmetic, the modified algorithm produces the factorization

$$Q^T A P = \begin{array}{c} \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right]_{m\times n} \begin{array}{l} r \\ m-r \end{array} \\ \begin{array}{cc} r & n-r \end{array} \end{array}$$

where $r = \text{rank } A$, Q is orthogonal, R_{11} is upper Δ^r & non-singular & P is a permutation.

$A = [a_1 | \cdots | a_n]$

If $AP = [a_{c_1} | \cdots | a_{c_n}]$, $Q = [q_1 | \cdots | q_m]$,

then $$a_{c_k} = \sum_{i=1}^{\min\{r,k\}} r_{ik} q_i \in \operatorname{span}\{q_1, \ldots, q_r\} \quad \forall\, 1 \le k \le n.$$

which implies that $\operatorname{range}(A) = \operatorname{span}\{q_1, \ldots, q_r\}$.

Steps of the algorithm :

Step 1: Compute $\|a_1\|_2, \ldots, \|a_n\|_2$, let p be the smallest index for which
$$\|a_p\|_2 = \max\{\|a_1\|_2, \ldots, \|a_n\|_2\}.$$

Step 2 : Exchange columns 1 & p (i.e. post-multiply by a perm. matrix P_1)
Then apply a suitable Householder matrix to annihilate the subdiagonal entries of column 1.

Thus $A \xmapsto[P_1]{\text{apply}} AP_1 \xmapsto[\text{transf. } Q_1]{\text{apply Householder}} Q_1 A P_1 = A_1$

$\underset{m\times m}{Q_1}\ \underset{m\times n}{A}\ \underset{n\times n}{P_1}$

$$= \underset{m\times n}{\left[\begin{array}{c|ccc} * & * & \cdots & * \\ \hline 0 & * & & * \\ \vdots & \vdots & & \\ 0 & * & & * \end{array}\right]}$$

Now repeat the above steps for the right-lower submatrix of A_1.

$\vdots$

Step k : We have reached

$$A_{k-1} = Q_{k-1} \cdots Q_1 A P_1 \cdots P_{k-1} = \left[\begin{array}{c|c} R_{11}^{(k-1)} & R_{12}^{(k-1)} \\ \hline 0 & R_{22}^{(k-1)} \end{array}\right]$$

$(k-1)\times(k-1)$ upper Δ^r & non-singular.

Quick description:

$$\underbrace{\left[\, a_1 \,\middle|\, \cdots \,\middle|\, a_n \,\right]}_{A} \xrightarrow{\;A P_1\;} \left[\, \overset{\text{exchange}}{a_p \,\middle|\, \cdots \,\middle|\, a_1} \,\middle|\, \cdots \,\middle|\, a_n \,\right]$$

$$\downarrow \; Q_1(AP_1) = A_1$$

$$\left[\begin{array}{c|ccc} \|a_p\|_2 & * & \cdots & * \\ \hline 0 & & & \\ \vdots & & * & \\ 0 & & & \end{array}\right]$$

$$\downarrow \; Q_2 A_1 P_2$$

$$Q_2 = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & \underset{\sim}{Q_2'} \end{array}\right] \qquad P_2 = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & P_2' \end{array}\right]$$

$$\left[\begin{array}{c|cccc} \|a_p\|_2 & * & * & \cdots & * \\ \hline 0 & \|a_{p'}\|_2 & * & \cdots & * \\ \vdots & 0 & & & \\ \vdots & \vdots & & \square & \\ 0 & 0 & & & \end{array}\right]$$

$$\downarrow \; \text{Continuing this way}$$

$$R = \begin{array}{c} \left[\begin{array}{cccc|c} \|a_p\|_2 & * & \cdots & * & R_{12} \\ 0 & \|a_{p'}\|_2 & & & * \\ \vdots & 0 & \ddots & \|a_{p_r}\|_2 & \\ \hline \vdots & \vdots & & \vdots & R_{22} \\ 0 & 0 & & 0 & \underset{\sim}{*} \end{array}\right] \begin{array}{l} r \\ \\ \\ m-r \end{array} \\ \hspace{-2em} r \hspace{8em} n-r \end{array}$$

rank $A = r$ $\Rightarrow$ the last $(n-r)$ columns of R must be contained in the span of the first r columns, but these have their last $(m-r)$ components as zero. Thus $R_{22} = 0$ & we're done

($\because$ this implies rank $R = r =$ rank A)

In principle, QR with column pivoting reveals the rank of A.

But in the presence of floating point errors, it is unlikely that we will get all entries of R_{22} equal to zero. So it is reasonable to terminate the reduction & declare A to have rank k if

$R_{22}^{(k)}$ is suitably small.

A typical terminating criterion is of the form

$$\|R_{22}^{(k)}\|_2 \leq \varepsilon \cdot \|A\|_2$$

for some machine-dependent/user-defined parameter ε.

<u>Caution</u>: In theory, it does not follow that $\|R_{22}^{(k)}\|_2$ is small if rank $A = k$

(See page 279, G-L for an example)

However, in practice, it is almost always true that $\|R_{22}^{(k)}\|_2$ is small if rank $A = k$.