Calcolo dell'indice di condizionamento in MATLAB

Matrici di Hilbert
molto malcondizionate

$$H(n) = \begin{pmatrix} 1 & 1/2 & 1/3 \;\dots & 1/n \\ 1/2 & 1/3 & 1/4 \;\dots & 1/(n+1) \\ \vdots & \vdots & \vdots \;\dots & \vdots \\ 1/n & 1/(n+1) & \dots & 1/(2n-1) \end{pmatrix}$$

$\mu(H(3)) = 574$

$\mu(H(4)) = 28000$

$\mu(H(6)) = 1.5 \times 10^7$

$n \to \infty \quad \mu(H(N)) \to \infty$

matrici intrattabili numericamente

Matrice di Vandermonde

relativa al vettore $x = [x_1, x_2, x_3, \ldots, x_n]$

$$V(n) = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ x_1 & x_2 & x_3 & \ldots & x_n \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_1^{n-1} & x_2^{n-1} & x_3^{n-1} & \ldots & x_n^{n-1} \end{pmatrix}$$

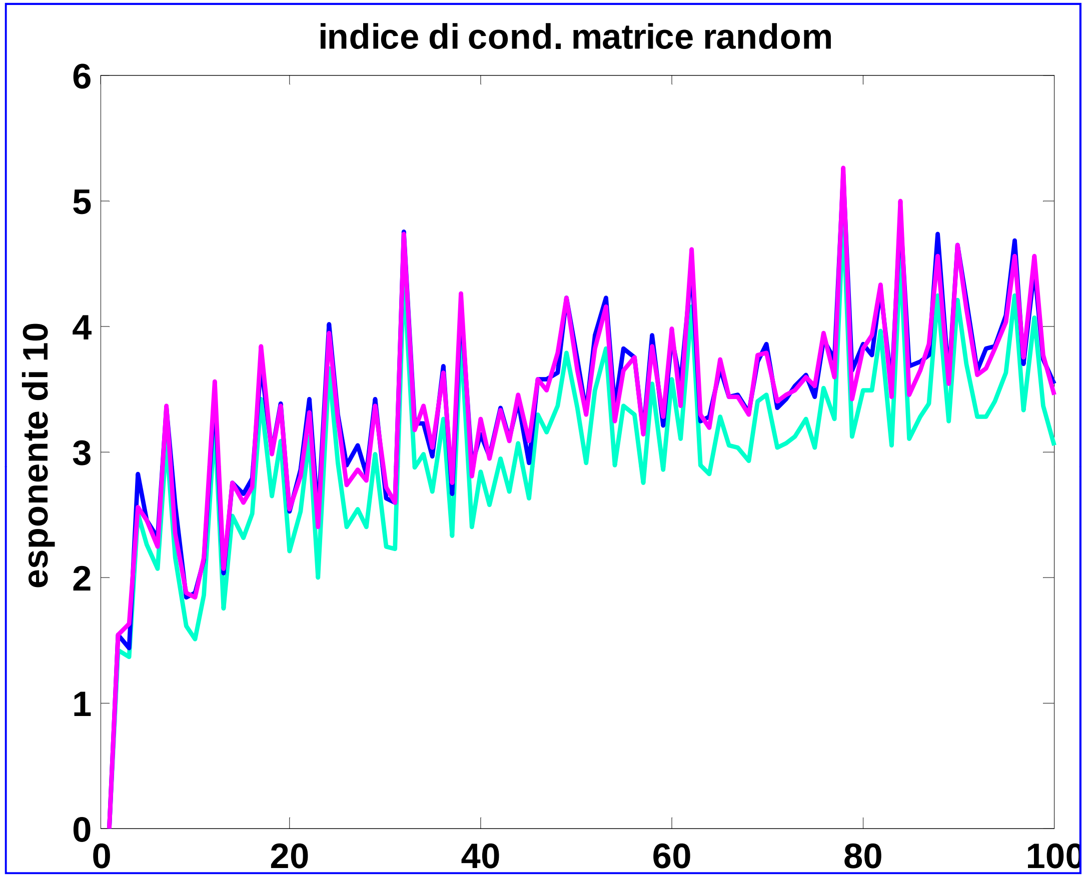
l'indice di condizionamento cresce almeno esponenzialmente con la dimensione *n*

molto malcondizionate

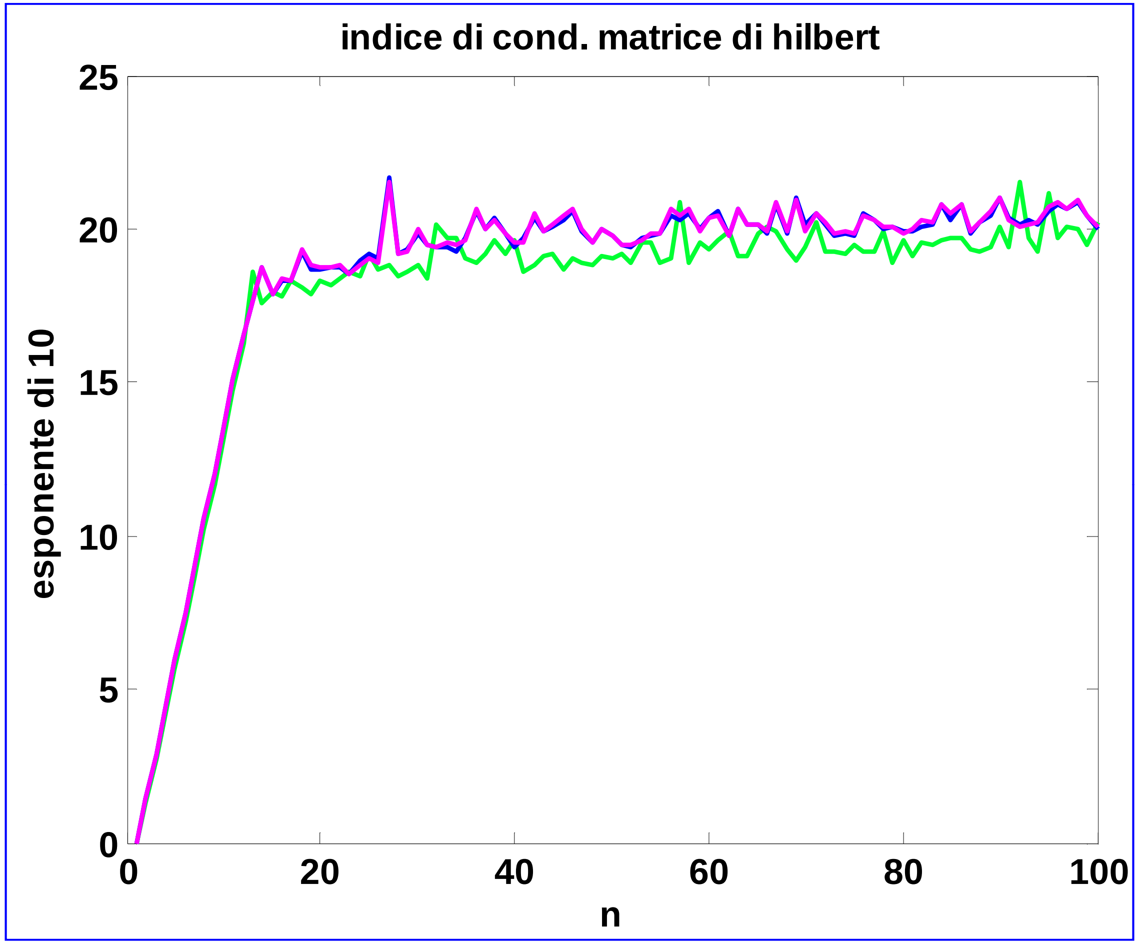
$$n \to \infty \qquad \mu(V(\underline{x})) \to \infty$$

matrici intrattabili numericamente

```
function plotcon(nmax)
%Calcola l'indice di condizionamento per le matrici
%RAND(N), con N=1:NMAX, confrontando su un grafico
%i risultati ottenuti in norma 1, norma 2, e norma inf
for n=1:nmax
a=rand(n);
con1(n) = cond(a,1);
con2(n) = cond(a); %norma 2
coninf(n) = cond(a,inf);
end
plot(log10(con2))
hold on
plot(log10(con1),'g')
plot(log10(coninf),'m')
```



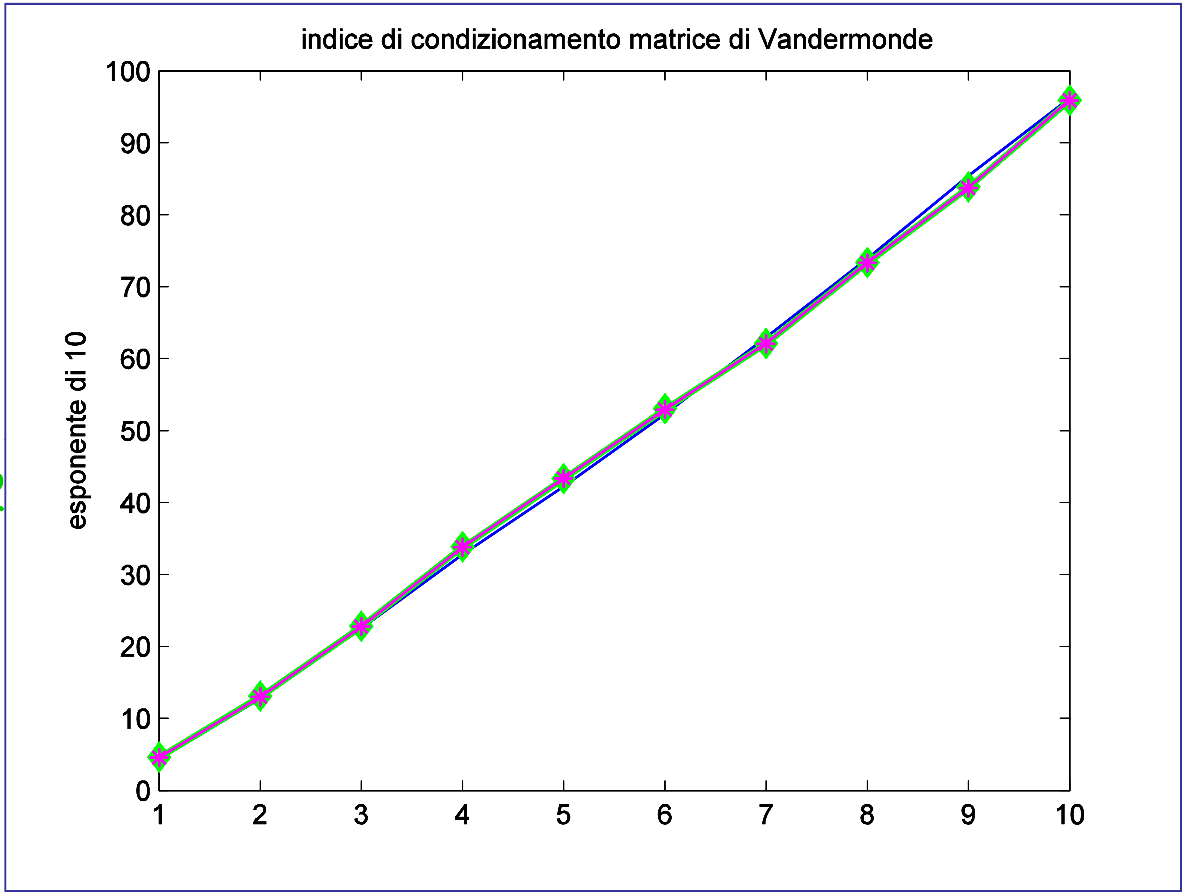
```
function plotconh(nmax)
%Calcola l'indice di condizionamento per le matrici
%HILB(N), con N=1:NMAX, confrontando su un grafico
%i risultati ottenuti in norma 1, norma 2, e norma inf
for n=1:nmax
a=hilb(n);
con1(n) = cond(a,1);
con2(n) = cond(a); %norma 2
coninf(n) = cond(a,inf);
end
plot(log10(con2))
hold on
plot(log10(con1),'g')
plot(log10(coninf),'m')
```



```
nmax=10;

function plotconV(nmax)
%Calcola l'indice di
%condizionamento per le
%matrici di Vandermonde,
%vand(N), con N=1:NMAX,
%confrontando su un
%grafico i risultati ottenuti
% in norma 1, norma 2 e
% norma inf
for n=1:nmax
a =vander(1:.5:3*n);
con1(n) = cond(a,1);
con2(n) = cond(a); %norma 2
coninf(n) = cond(a,inf);
end
plot(log10(con2))
hold on
plot(log10(con1),'g-d')
plot(log10(coninf),'m-*')
```

con1(1)	con1(2)	
ans =	ans =	
3.9795e+004	1.1252e+013	...



```
»H=hilb(10);
A=rand(10);
x=ones(10,1);

>>bh=H*x; b=A*x;
» cond(H)
ans =
 1.6025e+013
   %Perdita di 13 cifre significative
>> cond(A)
ans =
  1.6863e+02
    %Perdita di 2 cifre significative
>> xch=H\bh;  xc=A\b;
% errore relativo
>> norm(x-xch)/norm(x)
ans =
  1.9426e-004
» norm(x-xc) /norm(x)
ans =
 2.2928e-015
```

```
>> a=hilb(12);cond(a)
ans =
  1.7945e+016
>> x=ones(12,1);b=a*x;
>> xc=a\b;
Warning: Matrix is close to singular
or badly scaled.
        Results may be inaccurate.
RCOND = 2.220525e-017
   err=norm(x-xc) /norm(x)
   err =
      0.5819
```

matrice molto malcondizionata

Intrattabile!

RCOND - LAPACK reciprocal condition estimator ...
… If X is badly conditioned, RCOND(X) is near EPS.

Osservazione:

La sensibilità della soluzione di Ax=b alle perturbazioni sui dati A e b può essere misuata attraverso l'indice di condizionamento in norma-2:

Norma spettrale (indotta da $\|\cdot\|_2$ vettoriale)

$$\|A\|_2 = \left(\max_i \lambda_i^*\right)^{\frac{1}{2}}, \quad \lambda_i^*$$

Autovalori di $A^* A$, $A^* = A^T$ se A è reale

$$\mu_2(A) = \|A\|_2 \|A^+\|_2 = \sqrt{\frac{\lambda^*_{\max}}{\lambda^*_{\min}}} = \frac{\sigma_1}{\sigma_{rank(A)}}$$

```
in matlab:
>> help norm
>> help cond
```

```
cond(A)=cond(A,2)
cond(A,1)   in norma 1
cond(A,inf) in norma infinito
```