• A simple demonstration of ill-conditioning using MATLAB.

```
>> A = [1 2; 2 4.0001]

A =

    1.0000    2.0000
    2.0000    4.0001
```

← a 2x2 matrix $\underline{\underline{A}}$ that is almost linearly dependent (2nd row ≈ 2 * 1st row)

```
>> b = [3; 6.0001]

b =

    3.0000
    6.0001
```

← consider a vector $\underline{b}$ and the linear equation $\underline{\underline{A}}\,\underline{c} = \underline{b}$

```
>> inv(A)*b

ans =

     1
     1
```

← solve for $\underline{c} = \underline{\underline{A}}^{-1}\underline{b}$ and we get a reasonable result.

```
>> eig(A)

ans =

    0.0000
    5.0001
```

← check the eigenvalues. One is zero (to the number of digits we printed).

```
>> format long
>> eig(A)
```

← print more digits.

```
ans =

   0.000019999680004
   5.000080000319996
```

← now we see a very small eigenvalue ⇒ this signals trouble.

```
>> A_perturbed = [1.0001 2; 2 4.0001]
```

← a slight change (perturbation) of $\underline{\underline{A}}$

```
A_perturbed =

   1.000100000000000   2.000000000000000
   2.000000000000000   4.000100000000000

>> eig(A_perturbed)

ans =

   0.000100000000000
   5.000100000000000
```

← still has a small eigenvalue, although changed.

```
>> inv(A_perturbed)*b

ans =

   0.199996000079409
   1.399992000160637
```

← but now $\underline{c}' = \underline{\underline{A}}^{-1}_{\text{perturbed}}\,\underline{b}$ is radically different from $\underline{\underline{A}}^{-1}\underline{b}$. This is a disaster for numerical calculations.

```
>> b_perturbed = [2.9999; 6.0001]

b_perturbed =

   2.999900000000000
   6.000100000000000

>> inv(A)*b_perturbed

ans =

  -3.000100000019302
   3.000000000014552

>> inv(A_perturbed)*b_perturbed

ans =

  -0.600007999841182
   1.799984000321274

>>
```

try perturbing $\underline{b}$

the other calculations of $\underline{c}$ go all over the place!

Clearly this is a problem we need to fix!
[more later...]