Title: Lecture - Numerical Methods, PHYS 777

Speakers: Erik Schnetter, Dustin Lang

Collection/Series: Numerical Methods (Core), PHYS 777-, January 6 - February 5, 2025

Subject: Other

Date: January 09, 2025 - 10:15 AM

URL: https://pirsa.org/25010055

Linear Algebra 2

- matrix types
- eigenvalues and singular values
- sparse matrices
- Krylov subspace methods
- example: the power method

```
[ ]:
```

• example: the power method

Matrix

[1]:
```
A = randn(4, 4)
```

[1]:
```
4×4 Matrix{Float64}:
 -0.1467     0.752702  -0.179842    0.357416
  1.03975   -0.591779   0.0324229  -1.52922
  0.539381   0.315459   0.112141    0.158221
 -0.965721  -0.357836  -0.901844   -0.812315
```

[2]:
```
B = (A + A') / 2
```

[2]:
```
4×4 Matrix{Float64}:
 -0.1467     0.896227   0.179769  -0.304152
  0.896227  -0.591779   0.173941  -0.943526
  0.179769   0.173941   0.112141  -0.371811
 -0.304152  -0.943526  -0.371811  -0.812315
```

[ ]:
```
using LinearAlgebra
```

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code — Notebook — Julia 1.11.2

```
  0.539381   0.315459   0.112141    0.158221
 -0.965721  -0.357836  -0.901844   -0.812315
```

```
[2]: B = (A + A') / 2
```

```
[2]: 4×4 Matrix{Float64}:
 -0.1467     0.896227   0.179769  -0.304152
  0.896227  -0.591779   0.173941  -0.943526
  0.179769   0.173941   0.112141  -0.371811
 -0.304152  -0.943526  -0.371811  -0.812315
```

```
[3]: using LinearAlgebra
```

```
[4]: Symmetric(A)
```

```
[4]: 4×4 Symmetric{Float64, Matrix{Float64}}:
 -0.1467     0.752702  -0.179842    0.357416
  0.752702  -0.591779   0.0324229  -1.52922
 -0.179842   0.0324229  0.112141    0.158221
  0.357416  -1.52922    0.158221   -0.812315
```

```
[5]: Diagonal(A)
```

```
[5]: 4×4 Diagonal{Float64, Vector{Float64}}:
 -0.1467    ⋅          ⋅          ⋅
   ⋅      -0.591779    ⋅          ⋅
```

Simple 0 3 Julia 1.11.2 | Idle — Mode: Edit — Ln 1, Col 1 — Untitled1.ipynb

```
 0.752702  -0.591779   0.0324229  -1.52922
-0.179842   0.0324229  0.112141    0.158221
 0.357416  -1.52922    0.158221   -0.812315
```

```
[5]: Diagonal(A)
```

```
[5]: 4×4 Diagonal{Float64, Vector{Float64}}:
 -0.1467    ⋅          ⋅          ⋅
   ⋅      -0.591779    ⋅          ⋅
   ⋅        ⋅        0.112141     ⋅
   ⋅        ⋅          ⋅        -0.812315
```

```
[7]: b = rand(4);
```

```
[8]: Diagonal(A) \ b
```

```
[8]: 4-element Vector{Float64}:
 -4.322734028681198
 -1.5608999672552852
  4.03997938392708
 -0.23387877201805332
```

```
[ ]:
```

Safari File Edit View History Bookmarks Develop Window Help

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code | Notebook | Julia 1.11.2

Eigenvalues and eigenvectors

[9]:
```julia
eigen(A)
```

[9]:
```
Eigen{ComplexF64, ComplexF64, Matrix{ComplexF64}, Vector{ComplexF64}}
values:
4-element Vector{ComplexF64}:
 -1.0679978879513952 - 0.5404951051122822im
 -1.0679978879513952 + 0.5404951051122822im
 -0.2056048046482068 + 0.0im
   0.902947518090681 + 0.0im
vectors:
4×4 Matrix{ComplexF64}:
   -0.502702+0.135156im  …  -0.491863+0.0im  -0.280389+0.0im
    0.760381-0.0im            0.39398+0.0im  -0.722097+0.0im
 -0.00801287-0.106154im      0.652982+0.0im  -0.377823+0.0im
   -0.105176+0.358399im     -0.420077+0.0im   0.507157+0.0im
```

[ ]:

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 1, Col 1 Untitled1.ipynb 1

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

```
vectors:
4×4 Matrix{Float64}:
 -0.318948   -0.778606    0.423495   0.335704
  0.66376    -0.0440767  -0.159362   0.72944
 -0.0707617  -0.498884   -0.850378  -0.151539
  0.67282    -0.37808     0.268537  -0.576416
```

[11]:
```
eigen(Diagonal(A))
```

[11]:
```
Eigen{Float64, Float64, Matrix{Float64}, Vector{Float64}}
values:
4-element Vector{Float64}:
 -0.14669977917562027
 -0.5917788276417273
  0.11214059361184767
 -0.8123150492548141
vectors:
4×4 Matrix{Float64}:
 1.0  0.0  0.0  0.0
 0.0  1.0  0.0  0.0
 0.0  0.0  1.0  0.0
 0.0  0.0  0.0  1.0
```

[ ]:

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 1, Col 1 Untitled1.ipynb

Safari File Edit View History Bookmarks Develop Window Help

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code | Notebook | Julia 1.11.2

```
[12]: SVD{Float64, Float64, Matrix{Float64}, Vector{Float64}}
U factor:
4×4 Matrix{Float64}:
 -0.315989   -0.0200871  -0.846988  -0.427035
  0.903817    0.291698   -0.247296  -0.19202
 -0.0476697   0.349507   -0.41367    0.839315
  0.284593   -0.890142   -0.22435    0.276262
singular values:
4-element Vector{Float64}:
 2.0732352932594
 1.6520453159998763
 0.663046853827596
 0.1171258525318271
Vt factor:
4×4 Matrix{Float64}:
  0.330667  -0.429079  -0.0848297  -0.836274
  0.819825   0.145904   0.51756     0.196802
 -0.210151  -0.816535   0.452826    0.289923
  0.417593  -0.357594  -0.72103     0.421734
```

```
[ ]: C = randn(4, 3)
```

Simple 0 3 Julia 1.11.2 | Idle   Mode: Edit   Ln 1, Col 16   Untitled1.ipynb

Safari File Edit View History Bookmarks Develop Window Help

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code | Notebook | Julia 1.11.2

[13]:
```
SVD{Float64, Float64, Matrix{Float64}, Vector{Float64}}
U factor:
4×2 Matrix{Float64}:
 -0.0569784   0.769918
 -0.534023    0.368735
 -0.460166   -0.520824
  0.70698     0.00157886
singular values:
2-element Vector{Float64}:
 2.8169739017372732
 1.0154296867405945
Vt factor:
2×2 Matrix{Float64}:
 -0.926852  0.375428
  0.375428  0.926852
```

[18]:
```
sd.V * Diagonal(sd.S) * sd.Vt
```

[18]:
```
2×2 Matrix{Float64}:
  2.56305   -0.626876
 -0.626876   1.26935
```

[ ]:

Simple 0 3 Julia 1.11.2 | Idle Mode: Command Ln 1, Col 1 Untitled1.ipynb 1

```
2×2 Matrix{Float64}:
 -0.926852  0.375428
  0.375428  0.926852
```

[20]:
```
sd.U * Diagonal(sd.S) * sd.Vt - C
```

[20]:
```
4×2 Matrix{Float64}:
 -5.55112e-17   1.11022e-16
  0.0           2.77556e-17
  0.0           0.0
  0.0          -1.11022e-16
```

Sparse matrices

[ ]:

Safari File Edit View History Bookmarks Develop Window Help Thu Jan 9 10:45

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Markdown | Notebook | Julia 1.11.2

```
2×2 Matrix{Float64}:
 -0.926852  0.375428
  0.375428  0.926852
```

[20]:
```
sd.U * Diagonal(sd.S) * sd.Vt - C
```

[20]:
```
4×2 Matrix{Float64}:
 -5.55112e-17   1.11022e-16
  0.0           2.77556e-17
  0.0           0.0
  0.0          -1.11022e-16
```

Sparse matrices

```
Storage formats:
- coordinate format (all nonzero entries and their position)
- compressed sparse column

INefficient:
- find particular element
- add element
```

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 8, Col 1 Untitled1.ipynb 1

Sparse matrices

Storage formats:

- coordinate format (all nonzero entries and their position)
- compressed sparse column

INefficient:

- find particular element
- add element

efficient:

- matrix-vector multiplication

[ ]:
```
using SparseArrays
```

INefficient:

- find particular element
- add element

efficient:

- matrix-vector multiplication

[21]:
```
using SparseArrays
```

[29]:
```
A = sprand(4, 4, 0.4)
```

[29]:
```
4×4 SparseMatrixCSC{Float64, Int64} with 7 stored entries:
 0.528813    ⋅          ⋅         0.0374223
  ⋅          ⋅         0.211107    ⋅
 0.0010273   ⋅         0.981117   0.597585
  ⋅         0.843683    ⋅          ⋅
```

[23]:
```
b = randn(4);
```

[24]:
```
A * b
```

[24]:
```
4-element Vector{Float64}:
```

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

```
re you have enough memory, please either convert your matrix to a dense matrix, e.g. by calling `Matrix` or if
A` can be factorized, use `\` on the dense identity matrix, e.g. `A \ Matrix{eltype(A)}(I, size(A)...)` restrict
ions of `\` on sparse lhs applies. Alternatively, `A\b` is generally preferable to `inv(A)*b`

Stacktrace:
 [1] error(s::String)
   @ Base ./error.jl:35
 [2] inv(A::SparseMatrixCSC{Float64, Int64})
   @ SparseArrays ~/.julia/juliaup/julia-1.11.2+0.aarch64.apple.darwin14/share/julia/stdlib/v1.11/SparseArrays/s
rc/linalg.jl:1777
 [3] top-level scope
   @ In[32]:1
```

[34]:
```
A \ b
```

[34]:
```
4-element Vector{Float64}:
 -0.24449743338913096
  0.020911463199273925
  5.686656843732336
 -8.584924764127988
```

[ ]:

Simple 0 3 Julia 1.11.2 | Idle Mode: Command Ln 1, Col 1 Untitled1.ipynb

Safari File Edit View History Bookmarks Develop Window Help Thu Jan 9 10:51

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code | Notebook | Julia 1.11.2

```
rc/linalg.jl:1777
 [3] top-level scope
   @ In[32]:1
```

[34]:
```
A \ b
```

[34]:
```
4-element Vector{Float64}:
 -0.24449743338913096
  0.020911463199273925
  5.686656843732336
 -8.584924764127988
```

Iterative methods

[ ]:

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 1, Col 1 Untitled1.ipynb 1

[34]: 4-element Vector{Float64}:
```
 -0.24449743338913096
  0.020911463199273925
  5.686656843732336
 -8.584924764127988
```

Iterative methods

[ ]:
```
# Power method
function power_method(A)
    x = randn(size(A, 1))
    for iter in 1:20
        x = A *
    end
    return v
end
```

```
[34]: A \ b
```

```
[34]: 4-element Vector{Float64}:
 -0.24449743338913096
  0.020911463199273925
  5.686656843732336
 -8.584924764127988
```

Iterative methods

```julia
[ ]: # Power method
function power_method(A)
    x = randn(size(A, 1))
    for iter in 1:20
        xnew = A * x
        eval = xnew[1] / x[1]
        err = norm(xnew - eval * x)
        x = xnew
    end
    return x
end
```

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

[34]:
```
4-element Vector{Float64}:
 -0.24449743338913096
  0.020911463199273925
  5.686656843732336
 -8.584924764127988
```

Iterative methods

[ ]:
```julia
# Power method
function power_method(A)
    x = randn(size(A, 1))
    for iter in 1:20
        xnew = A * x
        eval = xnew[1] / x[1]
        err = norm(xnew - eval * x)
        println("iter: $iter   eval: $eval   error: $err
        x = xnew
    end
    return x
end
```

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 8, Col 60 Untitled1.ipynb 1

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

```
            x = xnew
        end
        return x
    end
```

[35]: power_method (generic function with 1 method)

[36]:
```
power_method(Symmetric(A))
```

```
iter: 1   eval: 0.5376217411385715   error: 1.8128783144813911   evec: [0.6664959090425118, -1.5245795156397541, -2.129756352606922, 0.15689137694522995]
iter: 2   eval: 0.3984988102326889   error: 2.379179312779193   evec: [0.35832269108117015, -0.4496059180382801 5, -2.3176337773117783, -1.2477685658510431]
iter: 3   eval: 0.16935426747113205   error: 3.856952123414704   evec: [0.14279116607522163, -0.489268108461939 9, -3.11443306173871, -1.371573848793906]
iter: 4   eval: -2.3430469650667956   error: 16.135704835471078   evec: [0.02418229333201792, -0.657477806875801 2, -3.9785437170571236, -1.8557948169639875]
iter: 5   eval: 2.09849114734882   error: 4.624213675315964   evec: [-0.0566602489999396, -0.8398973892827472, -5.151210961861169, -2.3766129781298875]
iter: 6   eval: 1.4983245554670876   error: 1.527034617023125   evec: [-0.11890103093295307, -1.087455347031378 4, -6.65147813417231, -3.0804066164053725]
iter: 7   eval: 1.364690712448646   error: 0.6932471152949234   evec: [-0.17815233431719535, -1.404171857107279 7, -8.596253791961582, -3.9792729271165985]
iter: 8   eval: 1.3205427002622294   error: 0.3493400856294579   evec: [-0.2431228360437227, -1.814727103921573 8, -11.108316726763675, -5.143658955440418]
iter: 9   eval: 1.3036227429081217   error: 0.18075592090693401   evec: [-0.3210540864045889, -2.345040517749003
```

Simple 0 3 Julia 1.11.2 | Idle Mode: Edit Ln 1, Col 1 Untitled1.ipynb 1

Safari File Edit View History Bookmarks Develop Window Help

localhost

Home | Untitled1.ipynb (3) - JupyterLab

File Edit View Run Kernel Tabs Settings Help

Julia Plotting.ipynb | Untitled.ipynb | Untitled1.ipynb

Code | Notebook | Julia 1.11.2

```
iter: 27   eval: 1.2923249011275015   error: 1.2748764705974097e-8   evec[1]: -0.020368689526318196
iter: 28   eval: 1.2923249086464332   error: 5.228269628257325e-9   evec[1]: -0.020368689325423173
iter: 29   eval: 1.2923249117299491   error: 2.1441126900110033e-9   evec[1]: -0.020368689243036097
iter: 30   eval: 1.2923249129944985   error: 8.79300531737066e-10   evec[1]: -0.020368689209249172
iter: 31   eval: 1.29232491351309   error: 3.6060123606967693e-10   evec[1]: -0.020368689195393155
iter: 32   eval: 1.2923249137257642   error: 1.4788260107270414e-10   evec[1]: -0.0203686891897108
iter: 33   eval: 1.2923249138129822   error: 6.064648386446695e-11   evec[1]: -0.020368689187380 47
iter: 34   eval: 1.29232491384875   error: 2.487111566137318e-11   evec[1]: -0.020368689186424805
iter: 35   eval: 1.2923249138634183   error: 1.0199738220863255e-11   evec[1]: -0.020368689186032882
iter: 36   eval: 1.2923249138694342   error: 4.182872503452636e-12   evec[1]: -0.020368689185872157
iter: 37   eval: 1.292324913871901   error: 1.715526367640199e-12   evec[1]: -0.020368689185806244
iter: 38   eval: 1.2923249138729125   error: 7.038089015993518e-13   evec[1]: -0.02036868918577921
iter: 39   eval: 1.2923249138733277   error: 2.886052473254436e-13   evec[1]: -0.020368689185768122
iter: 40   eval: 1.2923249138734978   error: 1.184196667743558e-13   evec[1]: -0.020368689185763577
```

[40]:
```
4-element Vector{Float64}:
 -0.02036868918576171
 -0.14660000339437243
 -0.8974362345658389
 -0.41557401313594233
```

[ ]:

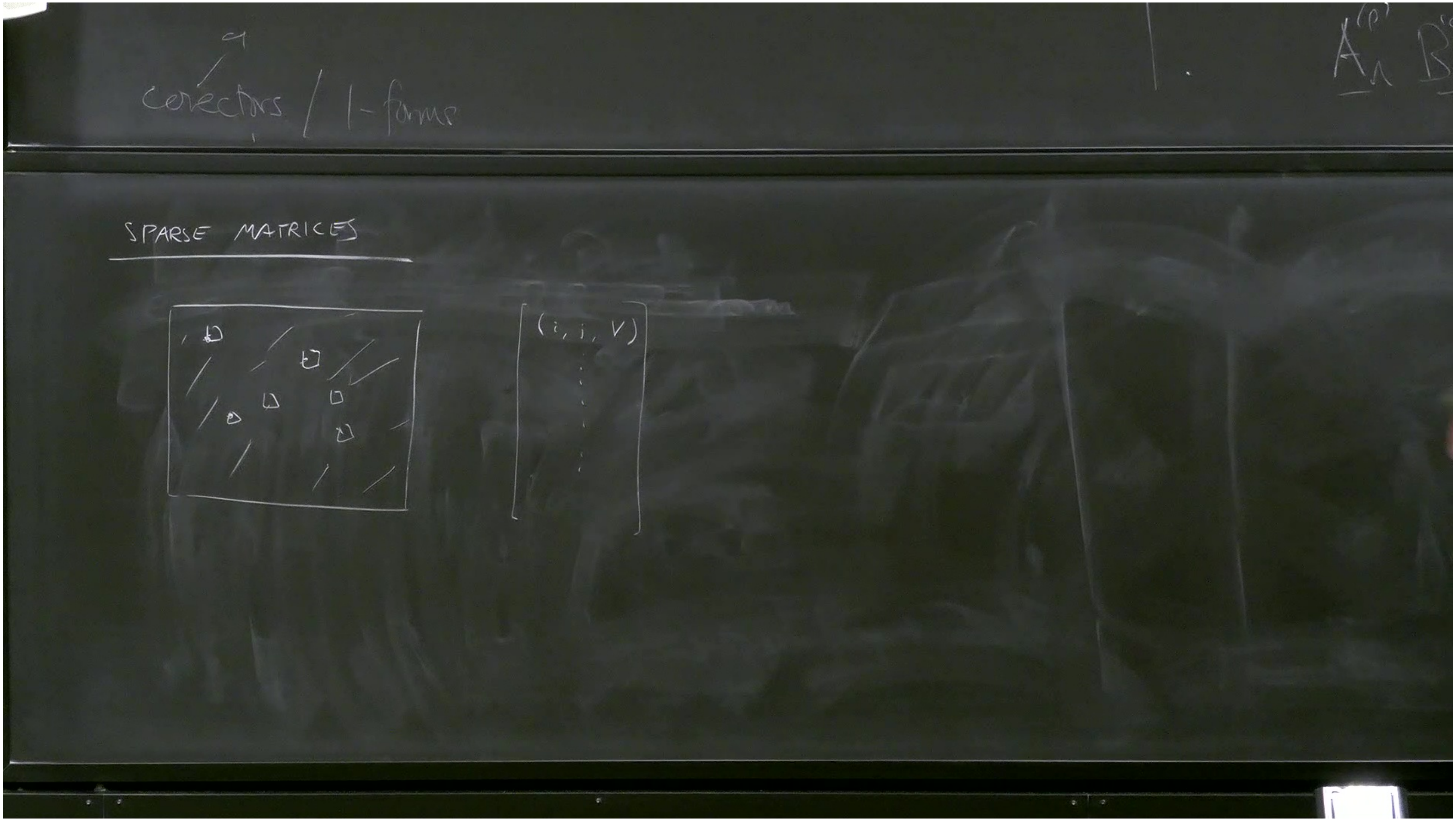
Simple 0 3 Julia 1.11.2 | Idle Mode: Command Ln 1, Col 1 Untitled1.ipynb 1

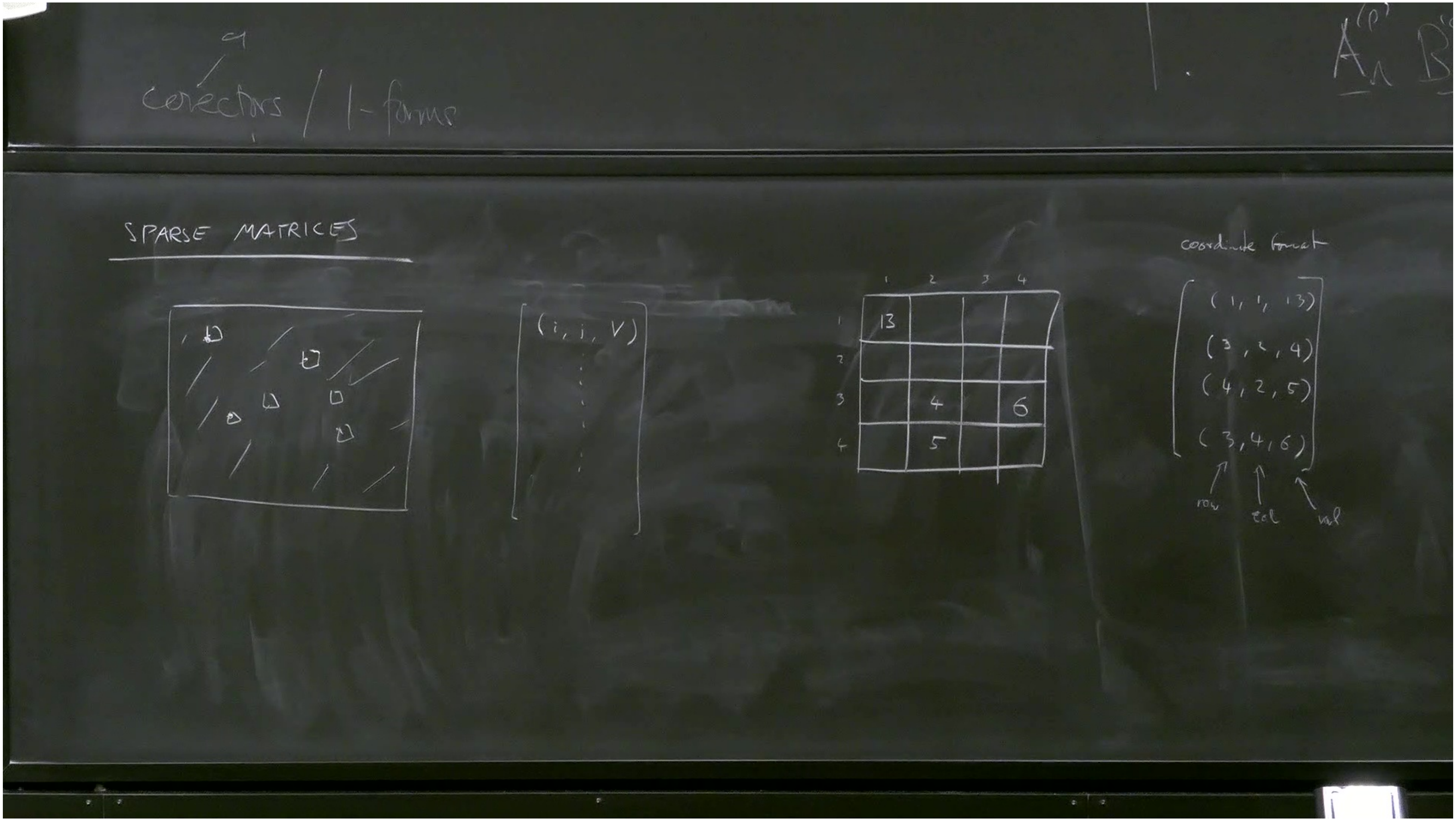
```julia
# Power method
function power_method(A)
    x = randn(size(A, 1))
    for iter in 1:40
        xnew = A * x
        eval = xnew[1] / x[1]
        err = norm(xnew - eval * x)
        println("iter: $iter   eval: $eval   error: $err   evec[1]: $(x[1])")
        x = xnew
        x = x / norm(x)
    end
    return x
end
```

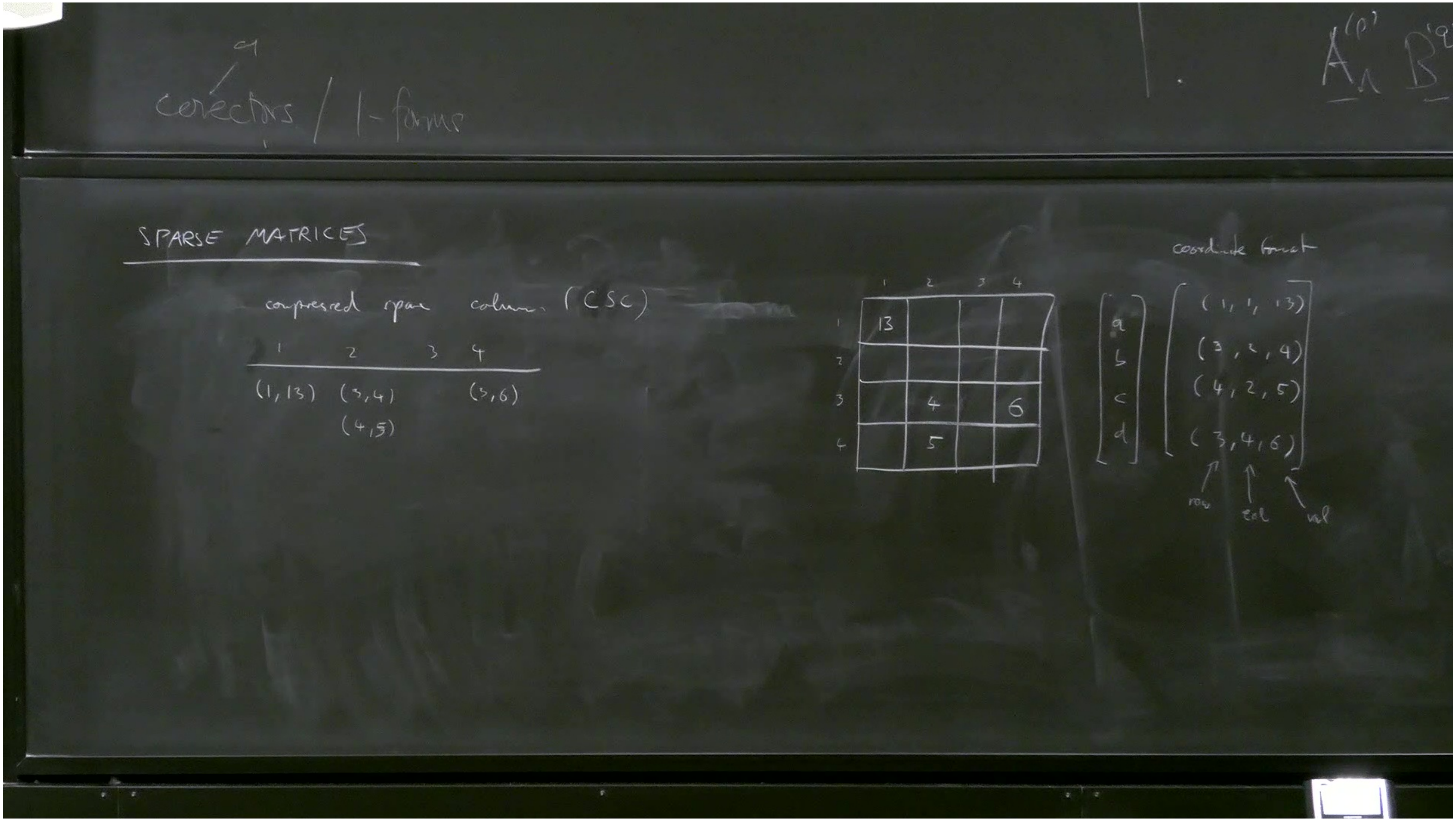
[39]: power_method (generic function with 1 method)

```julia
power_method(Symmetric(A))
```

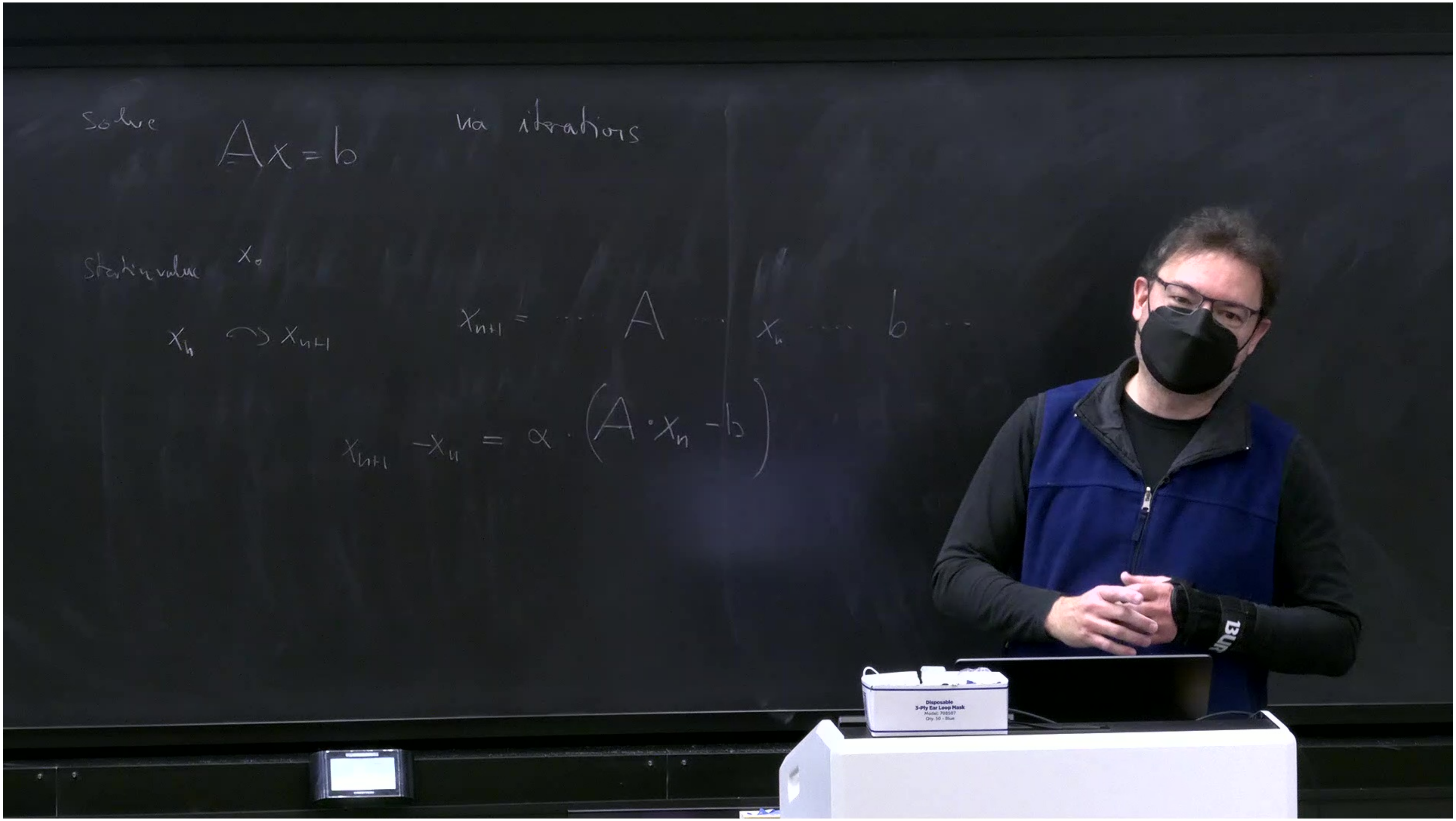
```
iter: 1   eval: 0.6729973992920767   error: 0.5587388394396237   evec[1]: -0.14892672035307536
iter: 2   eval: 0.49309898120705126   error: 0.6895634509682687   evec[1]: -0.5512634677517412
iter: 3   eval: 0.5824102660434007   error: 0.6642297945619118   evec[1]: -0.49249866352583704
iter: 4   eval: 0.5664484603581822   error: 0.7042162122557142   evec[1]: -0.28523494999524296
iter: 5   eval: 0.6571810481319954   error: 0.6318841513411477   evec[1]: -0.12999045077901897
iter: 6   eval: 0.7592179726058195   error: 0.5327083646511265   evec[1]: -0.06648516180827782
```


covectors / 1-forms
SPARSE MATRICES
(i, j, V)


covectors / 1-forms
SPARSE MATRICES
(i, j, V)
13
4
6
5
coordinate format
(1, 1, 13)
(3, 2, 4)
(4, 2, 5)
(3, 4, 6)
row
col
val


covectors / 1-forms
SPARSE MATRICES
compressed sparse column (CSC)
1 2 3 4
(1,13) (3,4) (3,6)
(4,5)
coordinate format
13
4 6
5
a
b
c
d
(1, 1, 13)
(3, 2, 4)
(4, 2, 5)
(3, 4, 6)
row
col
val


POWER METHOD
KRYLOV


Solve Ax = b via iterations
Starting value x0
xn → xn+1
xn+1 = ... A ... xn ... b ...
xn+1 − xn = α · (A · xn − b)