Master Class:
Large Scale Data Analysis and Visualization

Daniel Armyr
Application Engineer
Parsing complex text data
Using disk to work with huge data
Writing clean code
Visualization in 3d
Parsing complex text data

Using disk to work with huge data

Writing clean code

Visualization in 3d
Regular Expressions

- Wikipedia: “A sequence of characters that forms a search pattern for pattern matching with strings, or "find and replace"-like operations”

- A very powerful, but not very beginner-friendly, way of working with text data.

- Want more?
Parsing complex text data

Using disk to work with huge data

Writing clean code

Visualization in 3d
Memory mapped files

- Use the **virtual memory** system of the operating system to treat a file like a normal variable in MATLAB.
'Format', { ... 'int16', [2 2], 'model'; ... 'uint32', [1 1], 'serialno'; ... 'single', [1 3], 'expenses'});
Memory mapped files

- <Header>
- \( A_1 \) - [1000 1]
- \( B_1 \) - [1000 1]
- \( C_1 \) - [1000 1]
- \( A_2 \) - [1000 1]
- \( B_2 \) - [1000 1]
- \( C_2 \) - [1000 1]
- ...
Memory mapped files

- Use the **virtual memory** system of the operating system to treat a **file** like a normal **variable** in MATLAB

- Want more?
  - http://www.mathworks.se/help/matlab/memory-mapping.html
Parsing complex text data
Using disk to work with huge data

Writing clean code

Visualization 3D data
Writing clean code

- Runs faster
- Easier to debug
- Easier to change
- Easier to read
- Can be converted to C-code*

Want more?

* Using MATLAB Coder
Parsing complex text data
Using disk to work with huge data
Writing clean code
Visualization 3D data
Visualization 3D data
Plotting scalar data

Plotting vector data

A unifying example
\[ x, y, z = \text{meshgrid}([-1 \ 0 \ 1]); \]
\[ v = x .* \exp(-x.\text{^2} - y.\text{^2} - z.\text{^2}); \]
\[ \text{scatter3}(x(:,), y(:,), z(:,), [], v(:)); \]
[x, y, z] = meshgrid(-3:0.25:3);
v = x .* exp(-x.^2 - y.^2 - z.^2);
scatter3(x(:), y(:), z(:), [], v(:));
slice(x, y, z, v, [], [], 0)
slice(x, y, z, v, [-1.5, 1.5], 0, 0)
[xs,ys] = meshgrid(-3:0.1:3);
zs = sin(xs) - cos(ys);
slice(x, y, z, v, xs, ys, zs);
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
isosurface(x, y, z, v, 1e-5);
fv = isosurface(x,y,z,v,1e-5);
h = patch(fv);
view(3); % shortcut for view(-37.5, 30);
set(h, 'FaceColor', [0.5 1 0.5])
set(h, 'EdgeColor', 'none')
camlight;
camproj perspective
daspect([1 1 1]);
alpha(0.75);
fv2 = isosurface(x, y, z, v, 1e-3);
patch(fv2, 'FaceColor', [1 0.5 0], 'EdgeColor', 'none');
Plotting scalar data

Plotting vector data

A unifying example
[x, y, z] = meshgrid([-1 0 1]);
u = x + cos(4*x) + 3; v = sin(4*x) - sin(2*y); w = -z;
quiver3(x, y, z, u, v, w);
\[
[x, y, z] = \text{meshgrid}(-1.5:0.1:1.5)
\]
\[
u = x + \cos(4x) + 3; \quad v = \sin(4x) - \sin(2y); \quad w = -z;
\]
\[
quiver3(x, y, z, u, v, w);
\]
Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[sx, sy, sz] = meshgrid(-1:1);
streamline(x, y, z, u, v, w, sx, sy, sz);
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
streamslice(x,y,z,u,v,w, [-1.5 0 1.5], [], []);
streamslicex,y,z,u,v,w, 1.5, 1.5, -1.5;
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
[Parsing complex text data] [Using disk to work with huge data] [Writing clean code] [Visualization in 3d]
\[
[sx, sy, sz] = \text{meshgrid}(-1.5, -1:1, -1:1);
\text{streamribbon}(x, y, z, u, v, w, sx, sy, sz);
\]
Plotting low-dimensional data
Plotting scalar data
Plotting vector data

A unifying example
Visualizing 3D Data

- Just because you can, doesn’t mean that you should.

- But please do try this at home.

- Want more?
Parsing complex text data
Using disk to work with huge data
Writing clean code
Visualization in 3d