

Julia – Week 2 Day 2

Sophie van Genderen, Assoc Computational Specialist
Julia Giannini, Computational Specialist

Northwestern IT Research Computing and Data Services

December 4th 2025

Covered in Today's Class

- Review
- Machine Learning in Julia
- Machine Learning Packages
- Machine Learning Examples
- Questions?

Covered in Today's Class

- **Review**
- Machine Learning in Julia
- Machine Learning Packages
- Machine Learning Examples
- Questions?

Materials for this workshop

https://github.com/nuitrcs/Julia_workshop/tree/master/content_2025

The screenshot shows a GitHub repository interface for the 'Julia_workshop' repository. The repository path is 'nuitrcs / Julia_workshop'. The 'Code' tab is selected. On the left, the file tree shows a 'content_2025' folder expanded, containing subfolders for 'Day1_2025', 'Day2_2025', 'Day3_2025', and 'Day4_2025', along with files '.package_management_debuggi...', '.gitignore', and 'README.txt'. The main area displays the contents of the 'content_2025' folder. A commit from 'juliaGianniniRCDs' is shown: 'Merge branch 'master' of github.com:nuitrcs/Julia_workshop' (commit 746cf6d, 2 days ago). Below this, a table lists the files and their details:

Name	Last commit message	Last commit date
..		
Day1_2025	uploading slides from week 1 day 1	2 weeks ago
Day2_2025	adding slides for day2	2 weeks ago
Day3_2025	Merge branch 'master' of github.com:nuitrcs/Julia_workshop	2 days ago
Day4_2025	copied some stuff	last month
.package_management_debugging_notes.txt	instructions for configuring software environment and Julia kernel, s...	last month

Review – Julia kernel setup commands

1. From your laptop's terminal application, login to Quest

```
local-user@laptop ~ % ssh <NETID>@login.quest.northwestern.edu  
[<NETID>@quser43 ~] $
```

2. On Quest, navigate to our classroom allocation /projects directory

```
[<NETID>@quser43 ~] $ cd /projects/e33102  
[<NETID>@quser43 e33102] $
```

3. On Quest, if you have not already done so, create your own folder within /projects/e33102 and navigate into it

```
[<NETID>@quser43 e33102] $ mkdir <your_name_or_netid>  
[<NETID>@quser43 e33102] $ cd <your_name_or_netid>  
[<NETID>@quser43 <your_name_or_netid>] $
```

Review – Julia kernel setup commands cont.

4. On Quest, if you have not already done so, within /projects/e33102/<your_name_or_netid>/, make a new folder for a Julia project called my_ml_ds_project and navigate into it

```
[<NETID>@quser43 <your_name_or_netid>]$ mkdir my_ml_ds_project  
[<NETID>@quser43 <your_name_or_netid>]$ cd my_ml_ds_project  
[<NETID>@quser43 my_ml_ds_project]$
```

5. Copy our Project.toml and .ipynb files into this my_ml_ds_project directory

```
[<NETID>@quser43 my_ml_ds_project]$  
[<NETID>@quser43 my_ml_ds_project]$ cp /projects/e33102/ml_ds_kernel_example/Project.toml .  
[<NETID>@quser43 my_ml_ds_project]$ cp /projects/e33102/example-code/Julia_DS_Example.ipynb .  
[<NETID>@quser43 my_ml_ds_project]$ cp /projects/e33102/example-  
code/mnist_perceptron_training.ipynb .  
[<NETID>@quser43 my_ml_ds_project]$ ls  
Julia_DS_Example.ipynb  mnist_perceptron_training.ipynb  Project.toml
```

Review – Julia kernel setup commands cont.

- From the `my_ml_ds_project` directory, launch Julia with activating the Project `my_ml_ds_project`

```
[<NETID>@quser43 my_ml_ds_project]$ module load julia/1.11.4  
[<NETID>@quser43 my_ml_ds_project]$ julia --project=.  
julia >
```

- In Julia, if you have not done so already, instantiate the Julia Project defined by your copy of `/projects/e33102/<your_name_or_netid>/my_ml_ds_project/Project.toml`

```
julia > using Pkg  
julia > Pkg.status()  
julia > Pkg.instantiate()
```

Review – Julia kernel setup commands cont.

8. In Julia, if you have not done so already, create an IJulia kernel associated with my_ml_ds_project

```
julia > using Pkg  
julia > using Conda  
julia > using IJulia  
julia > installkernel("Julia 1.11.4 - my_ml_ds_project", "--  
project=projects/e33102/<your_name_or_netid>/my_ml_ds_project")
```

9. In your laptop's web browser, log into Quest OnDemand via <https://ondemand.quest.northwestern.edu/> using your Northwestern credentials. Start a Jupyter job with the following parameters. Then navigate to projects/e33102/<your_name_or_netid>/my_ml_ds_project and open our notebooks `Julia_DS_Example.ipynb` and `mnist_perceptron_training.ipynb`. In the notebooks, select your kernel `Julia 1.11.4 - my_ml_ds_project`

Quest OnDemand – Job Card Options

<input type="checkbox"/> Pre-Installed Kernel	<input type="checkbox"/> Request more than a single node (Optional)
ml-data-science-kernel-py311	<input type="checkbox"/> Total memory or RAM do you need in GB.
<input type="checkbox"/> SLURM Partition	20
short	<input type="checkbox"/> Wall Time (in number of hours)
<input type="checkbox"/> SLURM Account	2
e33102	<input type="checkbox"/> Use JupyterLab instead of Jupyter Notebook?
<input type="checkbox"/> Number of CPUs/cores/processors	<input type="checkbox"/> Jupyter root directory (Optional)
1	/projects/e33102

Julia

- Review
- **Machine Learning in Julia**
- Machine Learning Packages
- Machine Learning Examples
- Questions?

Machine Learning

Why Julia

- <https://www.datacamp.com/blog/introduction-machine-learning-julia>

Julia

- Review
- Machine Learning in Julia
- **Machine Learning Packages**
- Machine Learning Examples
- Questions?

Machine Learning packages

- **Notebook:** Pluto, IJulia, Jupyter
- **Package/environment management:** Pkg
- **Importing and handling data:** CSV, DataFrames
- **Plotting and output:** Plots, StatsPlots, LaTeXStrings, Measures, Makie
- **Statistics and Math:** Random, Statistics, LinearAlgebra, StatsBase, Distributions, HypothesisTests, KernelDensity, Lasso, Combinatorics, SpecialFunctions, Roots
- **Individual machine learning packages:**
 - Generalized linear models (e.g. linear regression, logistic regression): GLM
 - Deep Learning: Flux, Knet
 - Support vector machines: LIBSVM
 - Decision tree, random forest, AdaBoost: DecisionTree
 - K-nearest neighbors: NearestNeighbors
 - K-means clustering: Clustering
 - Principal component analysis: MultivariateStats

Julia

- Review
- Machine Learning in Julia
- Machine Learning Packages
- **Machine Learning Examples**
- Questions?

Machine Learning – Working Example

MNIST dataset:

- Julia package FluxML example models:
 - <https://github.com/FluxML/model-zoo/tree/master?tab=readme-ov-file#examples-in-the-model-zoo> > ‘Simple multi-layer perceptron’
 - Helpful youtube video about training on the MNIST dataset:
 - https://www.youtube.com/watch?v=aircAruvnKk&list=PLZHQBObOWTQDNU6R1_67000Dx_ZCJB-3pi

Machine Learning – Working Example

Work through the Jupyter Notebook

`mnist_perceptron_training.ipynb` in your own directory
`/projects/e33102/<your_name_or_netid>/my_ml_ds_project/` (see previous slides for set-up details)

Machine Learning – More Examples

- Time series data / Fourier analysis
 - <https://github.com/Marco-Congedo/FourierAnalysis.jl>
- Transformer language processing:
 - <https://juliapackages.com/p/transformers>
 - <https://developers.google.com/machine-learning/clustering/dnn-clustering/supervised-similarity>

Julia

- Review
- Machine Learning in Julia
- Machine Learning Packages
- Machine Learning Examples
- Questions?

Thank You!

Questions about Quest? Email us at:
quest-help@northwestern.edu