




## M4: Metals, Magnets, and Miscellaneous Materials

Christina Colleen Lee<sup>1</sup>

<sup>1</sup> Independent Researcher

DOI: [10.21105/jose.00058](https://doi.org/10.21105/jose.00058)

### Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Submitted: 10 June 2019

Published: 04 July 2019

### License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

### Summary

M4 contains jupyter notebooks on condensed matter physics and topics helpful for a condensed matter physicist. This repository is an amalgamation of self-contained tutorials on a variety of subjects and at a variety of skill levels, unified under a loose theme, style, and delivery method.

While most notebooks fully finish one topic, some fit into themed series (Monte Carlo, Markov Chain, Ferromagnet, Phase Transitions). I split one topic because of length and difficulty (Exact Diagonalization of 1D Spin Chain).

In addition to being self-contained in terms of content, the notebooks are self-contained in terms of level and prerequisites. To account for this disparity, I divide the notebooks into the categories:

- General: Those with some background in programming, math, and physics should be able to follow.
- Prerequisites: I spell out classes in the beginning that one should have taken to understand the physics of the post. For example, to understand calculating the orbitals of a hydrogen-like atom, I assume you analytically solved it in a Quantum Mechanics course. Even if you haven't taken the classes, feel free to try still.
- Graduate: Specialty technique that assumes you have a solid grounding in the basics. I still try to not take any knowledge for granted, but not everyone tries to learn these things in the first place.
- Numerics: Without a specific physics goal. Sometimes I want to talk about something programming or numerical methods related.
- Programming: Something related to the language itself.

The content is both downloadable on GitHub and hosted on a static website, [albi3ro.github.io/M4](https://albi3ro.github.io/M4), for easier access.

I have recently updated almost all posts to Julia v1.0.1 and unified the plotting utility to `Plots.jl` with `gr()` as the backend. A few posts are being excluded from maintenance and deprecated due to language changes and personal preferences. For example, I wrote a post on Gadfly, a plotting utility, but I no longer use it, keep up to date with the package syntax, or think it's the best option for someone else.

### Statement of Need

Many computational techniques are presented either through fluffy words and descriptions or highly technical, fully fledged research implementations. Mere words are quite

different than the actual code. Trying to convert a description to code uncovers quite a few details that only matter during implementation. Research implementations often prioritize results over readability and can leave a beginner with gaps in their understanding of the system. I try to bridge this gap by coding bare-bones versions of algorithms and examining critical aspects of them.

While a formal course could cover these topics, someone only really learns them through usage or starts paying attention when they need the information for a project. That in mind, I write for the independent learner. A user could merely scan or read the notebooks, but I try to suggest different parameter combinations or extensions so users can play around with the base notebook on their own, to test its limits and peculiarities.

## Acknowledgments

Thanks to Lee James O’Riordan ((??)) for discussions and revisions.