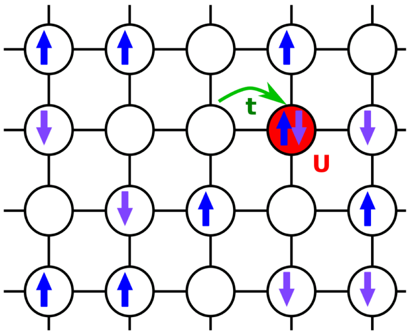
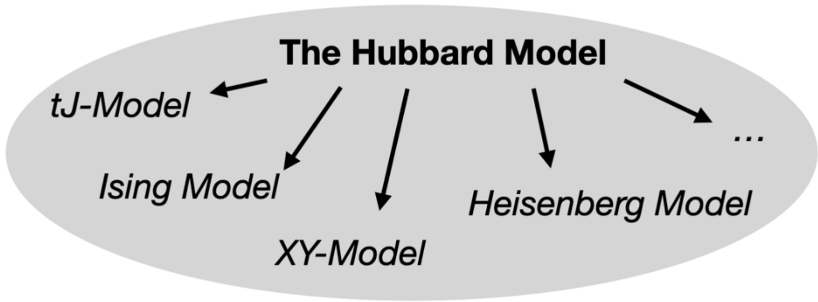
Master Project:

The 2D Hubbard Model meets Machine Learning

Join the frontier of condensed matter research and tackle the **“mother of all models”**. The Hubbard model in 2 and larger dimensions is an unsolved paradigm of modern condensed matter physics and believed to hold the key for understanding emergent phenomena like **high-temperature superconductivity, quantum magnetism,** and many other emergent phenomena.

Using our in-house developed code **SOLAX**, a powerful Python package leveraging Google's cutting-edge JAX library[1], you'll employ advanced machine learning techniques to simulate and analyze the model. If you're passionate **about quantum mechanics and machine learning**, this is the project for you!





The project begins by benchmarking our neural network in the **1D case**, where the problem is **analytically solvable** through the so-called Bethe Ansatz. This allows us to rigorously test and optimize the neural network. By moving to the 2D case, we will explore uncharted computational territory by treating clusters of increasing size to compute many-body wavefunctions. Although there are no exact solutions in the 2D case, we can compare results to other sophisticated quantum-field theoretical approximations.

Even the smallest incremental insights into the 2D Hubbard model can have profound implications. New numerical results may eventually **inspire new analytical approaches and approximations**, opening doors to a deeper understanding of one of the most critical models in condensed matter physics.

Skills you must have:

You have mastered basic Quantum Mechanics (and still like it) and you have *basic* knowledge in Python.

Skills you will learn during the project:

2nd quantization, Green function formalism, and Feynman diagrams.

Projects can start at any time.

If you want to learn more, get further reading material, or apply for the project: contact me!

Prof. Dr. Philipp Hansmann (philipp.hansmann@fau.de)

September 2024

---

[1] JAX is revolutionizing scientific computing by combining automatic differentiation and accelerated linear algebra on GPUs and TPUs, making it ideal for complex machine learning tasks in physics. (https://github.com/google/jax)