A Machine Learning-Enabled Study of Superconductivity

Application of the XGBoost Algorithm

Rajeev Atla

John P. Stevens High School

July 24, 2020

Outline

Introduction

XGBoost

- eXtreme Gradient Boosting

Introduction

XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala

Introduction

XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]

Introduction
XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]
  - Won the Higgs Boson Machine Learning Challenge

Introduction

XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]
  - Won the Higgs Boson Machine Learning Challenge
- Ensemble learning [Friedman et al. 2017]

Introduction

XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]
    - Won the Higgs Boson Machine Learning Challenge
- Ensemble learning [Friedman et al. 2017]
    - Combination of homogenous weak learners

Introduction

XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]
  - Won the Higgs Boson Machine Learning Challenge
- Ensemble learning [Friedman et al. 2017]
  - Combination of homogenous weak learners
  - End result is a weighted sum of weak learners

$$\theta_f = \sum_j w_j \theta_j$$

Introduction
XGBoost

- eXtreme Gradient Boosting
- Package for **Python**, C++, Java, R, Julia, and Scala
- Was used to process data from Large Hadron Collider (LHC) [Chen 2015]
  - Won the Higgs Boson Machine Learning Challenge
- Ensemble learning [Friedman et al. 2017]
  - Combination of homogenous weak learners
  - End result is a weighted sum of weak learners

$$\theta_f = \sum_j w_j \theta_j$$

  - w_j are determined by backpropogation via gradient descent

Introduction

Ginzburg-Landau Theory

- For a homogenous superconductor, the Ginzburg-Landau equation is

$$\alpha\phi + \beta|\phi|^2\phi = 0$$

Introduction

Ginzburg-Landau Theory

- For a homogenous superconductor, the Ginzburg-Landau equation is

$$\alpha\phi + \beta|\phi|^2\phi = 0$$

- The nontrivial solution for $T < T_c$ is

$$|\phi|^2 = -\frac{\alpha}{\beta}(T - T_c)$$

Introduction

Ginzburg-Landau Theory

- For a homogenous superconductor, the Ginzburg-Landau equation is

$$\alpha\phi + \beta|\phi|^2\phi = 0$$

- The nontrivial solution for $T < T_c$ is

$$|\phi|^2 = -\frac{\alpha}{\beta}\left(T - T_c\right)$$

- The characteristic length scale ξ is called the Ginzburg-Landau coherence length

$$\xi = \sqrt{\frac{\hbar^2}{2m^*|\alpha|}}$$

Introduction

Types of Superconductors

- Two types - Type 1 and Type 2

Introduction

Types of Superconductors

- Two types - Type 1 and Type 2
- Notation

Introduction

Types of Superconductors

- Two types - Type 1 and Type 2
- Notation
  - $H_c(T)$ is critical field as a function of temperature

Introduction

Types of Superconductors

- Two types - Type 1 and Type 2
- Notation
  - $H_c(T)$ is critical field as a function of temperature
  - T_c is critical temperature

Introduction

Types of Superconductors

- Two types - Type 1 and Type 2
- Notation
  - $H_c(T)$ is critical field as a function of temperature
  - T_c is critical temperature

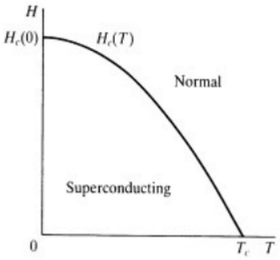

Figure: $H - T$ **phase diagram for a Type 1 superconductor [Tinkham]**

Introduction

Type 2 Superconductors

Introduction

Type 2 Superconductors

- Ginzburg-Landau parameter $\kappa > \frac{1}{\sqrt{2}}$

Introduction

Type 2 Superconductors

- Ginzburg-Landau parameter $\kappa > \frac{1}{\sqrt{2}}$
  - Definition: $\kappa = \frac{\lambda}{\xi} = \frac{e\hbar}{m_e c}\sqrt{\frac{\beta}{2\pi}}$
  - Surface energy is negative

Introduction

Type 2 Superconductors

- Ginzburg-Landau parameter $\kappa > \frac{1}{\sqrt{2}}$
  - Definition: $\kappa = \frac{\lambda}{\xi} = \frac{e\hbar}{m_e c}\sqrt{\frac{\beta}{2\pi}}$
  - Surface energy is negative
- $H_{c2} = H_{c1}\kappa\sqrt{2}$

Introduction

Type 2 Superconductors

- Ginzburg-Landau parameter $\kappa > \frac{1}{\sqrt{2}}$
  - Definition: $\kappa = \frac{\lambda}{\xi} = \frac{e\hbar}{m_e c}\sqrt{\frac{\beta}{2\pi}}$
  - Surface energy is negative
- $H_{c2} = H_{c1}\kappa\sqrt{2}$
- In type 1, $H_{c2} = H_{c1}$

Introduction

Type 2 Superconductors

- Ginzburg-Landau parameter $\kappa > \frac{1}{\sqrt{2}}$
  - Definition: $\kappa = \frac{\lambda}{\xi} = \frac{e\hbar}{m_e c}\sqrt{\frac{\beta}{2\pi}}$
  - Surface energy is negative
- $H_{c2} = H_{c1}\kappa\sqrt{2}$
- In type 1, $H_{c2} = H_{c1}$

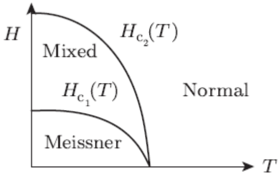

Figure: $H - T$ **phase diagram for a Type 2 superconductor [Girvin and Yang 2019]**

Introduction

Type 2 Superconductors: Abrikosov Lattice Vortices

- For $H_{c1} < H < H_{c2}$ in a Type 2 Superconductor, Abrikosov vortices appear in the material

- For $H_{c1} < H < H_{c2}$ in a Type 2 Superconductor, Abrikosov vortices appear in the material
- These are flux vortices that are quantized, with

$$\Phi = \frac{nhc}{2e}, \quad n \in \mathbb{Z}$$

- For $H_{c1} < H < H_{c2}$ in a Type 2 Superconductor, Abrikosov vortices appear in the material
- These are flux vortices that are quantized, with

$$\Phi = \frac{nhc}{2e}, \quad n \in \mathbb{Z}$$

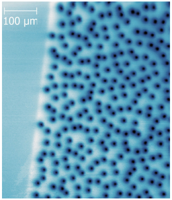

Figure: **Abrikosov vortices in YBCO - created by Wells et al. 2015 using scanning SQUID microscopy**

Data

- Taken from UCI (University of California, Irvine) Machine Learning Repository

Data

- Taken from UCI (University of California, Irvine) Machine Learning Repository
- 21,263 examples with 81 features

Data

- Taken from UCI (University of California, Irvine) Machine Learning Repository
- 21,263 examples with 81 features
- Model was only trained with 11 features to prevent overfitting

Data

- Taken from UCI (University of California, Irvine) Machine Learning Repository
- 21,263 examples with 81 features
- Model was only trained with 11 features to prevent overfitting

Figure: **UCI Machine Learning Repository**

Methods

Methods

- XGBoost library

Methods

- XGBoost library
  - XGBClassifier class

Discussion

Rajeev Atla

Discussion

- Confusion matrix made using matplotlib library

Discussion

- Confusion matrix made using matplotlib library

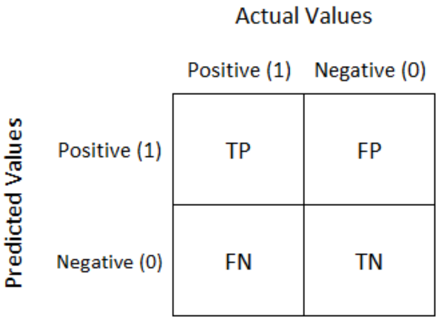

Figure: **Example Confusion Matrix**

Conclusion

Conclusion

- Python training files, these slides, the dataset, etc. can be found at `https://github.com/RajeevAtla/Graphene-Research/`

Conclusion

- Python training files, these slides, the dataset, etc. can be found at `https://github.com/RajeevAtla/Graphene-Research/`
- Easiest way to access is using git

References

Acknowledgements

I would like to thank Leo Lo and Dr. Serena McCalla for their mentorship through the iResearch Institute.

iResearch
INSTITUTE

I would also like to acknowledge my parents for their constant support.