Predicting solid-state qubit material host

by

Oliver Lerstøl Hebnes

Thesis

for the degree of

Master of Science

Faculty of Mathematics and Natural Sciences
University of Oslo

May 1, 2021

Contents

Part I

Introduction

Chapter 1

Introduction

To sustain the digital world's increasing computational demand, alternatives to the classical computer must be explored. Quantum computers are commonly thought of as a futuristic device, but are increasingly manifested today as a possible solution. Unfortunately, there are substantial challenges associated with the modern quantum platforms simultaneously as the selection of quantum platforms are slim. The majority of discoveries of potential quantum platforms have so far happened by accident, and there is an urgent need for new and better materials that can escalate the effort for a sustainable future.

Conveniently, we are progressively recognizing the fourth science paradigm which constitutes of big words like *Big data* and *Data science*, which all comes together into making it possible to extract knowledge from data. In particular, we have during the recent years seen the rise of computational materials science databases due to successful ab-initio approaches alike *density functional theory*. This catalysator has enabled a new approach for materials discovery; instead of calculating properties based on composition and structure, we are now able to reverse the approach into selecting a key property and finding materials that maximise this goal.

Discovery of current deep defects and material hots by serendipitet. (ref. data-mining artikkl)

This it The introduction. Another coffee.

Trengs det egentlig undertitler her? Tenketenketenk

Part II

Theory

Chapter 2

Quantum technologies

This chapter will provide a brief overview of the current state-of-the-art in quantum technological advances. This will not only give us insights in how the technology is being used today, but also grant us the opportunity to discuss key concepts that are fundamental to understand for this thesis. Thereafter we will look into how materials are composed, and what kind of properties a material needs to exhibit to be an eligible host for quantum devices. Finally, we will giving a few specific examples of materials with promising point defects that have been comprehensively researched. Importantly, this will motivate the reasoning for finding new materials that might excel in areas where other materials falls short for utilization in quantum technology.

Quantum technology (QT) refers to practical applications and devices that utilize the principles of quantum physics as a foundation. Technologies in this spectrum are based on concepts such as *superposition*, *entanglement* and *coherence*, which are all closely related to one another.

A quantum superposition refers to that any two or more quantum eigenstates can be added together into another valid quantum state, such that every quantum state can be represented as a sum, or a superposition, of two or more distinct states. This is according to the wave-particle duality which states that every particle or another quantum entity may be described as either a particle or a wave. When measuring the state of a system residing in a superposition of eigenstates, however, the system falls back to one of the basis states that formed the superposition, destroying the original configuration.

Quantum entanglement refers to when a two- or many-particle state cannot be expressed independently of the state of the other particles, even when the particles are separated by a significant distance. As a result, the many-particle state is termed an entangled state [1].

Quantum coherence arises if two waves coherently interfere with each other and generate a superposition of the two states with a phase relation. Likewise, loss of coherence is known as *decoherence*.

Another concept that the reader should be familiar with is the famous Heisenberg uncertainty principle. It states that

$$\sigma_x \sigma_p \leqslant \frac{\hbar}{2}, \tag{2.1}$$

where σ_x is the standard deviation for the position and σ_p is the standard deviation in momentum. This means that we cannot accurately predict both the position and momentum of a particle at the same time. Thus, we often calculate the probability for a particle to be in a state which results in concepts such as an electron sky surrounding an atom core. However, remember that equation (2.1) is an inequality, which means that it is possible to create a state where neither the position nor the momentum is well defined.

2.1 Quantum computing

The start of the digital world's computational powers can be credited to Alan Turing. In 1937, Turing [2] published a paper where he described the *Turing machine*, which is regarded as the foundation of computation and computer science. It states that only the simplest form of calculus, such as boolean Algebra (1 for true and 0 for false), is actually computable. This required developing hardware that could handle classical logic operations, and was the basis of transistors that are either in the state ON or OFF depending on the electrical signal. Equipped with a circuit consisting of wires and transistors, commonly known as a computer, we could develop software to solve all kinds of possible applications.

Driven by the development of software, conventional computers have in accordance to Moore's law [3], doubled the amount of transistors on integrated circuit chips every two years as a result of smaller transistors. Furthermore, the clock frequency has enhanced with time, resulting in a doubling of computer performance every 18 months [4]. Alas, miniaturization cannot go on forever as transistors are mass-produced at 5 nm today and are expected to reach a critical limit of 3 nm in the following years [5].

To sustain the digital world's increasing computational demand, other alternatives than the conventional classical computer must be explored. This is where quantum computing comes into the picture. The term quantum computer is a device that exploits quantum properties to solve certain computational problems more efficiently than allowed by Boolean logic [6].

The idea is to pass information in the form of a quantum bit, or *qubit* for short. They are the building blocks of quantum computers, and as opposed to the conventional 0 or 1-bits that classical computers are based on, they can inhabit any superposition of the states 0 or 1. This is illustrated in figure 2.1.

The architecture of a gate-based quantum computer is dependent on a set of quantum logic gates that perform unitary transformations on sets of qubits [7, 8]. A different implementation of quantum computers exists is adiabatic quantum computer. This approach is not based on gates, but on defining the answer of a problem as the ground state of a complex network of interactions between qubits, and then controlling the interactions to adiabatically evolve the system to the ground state [9].

It has been demonstrated that exponentially complex problems can be reduced to polynomially complex problems for quantum computers [4]. For example, a quantum search algorithm found by Grover [10] offers a quadratic speed-up compared to classical algorithms, while Shor's quantum integer factorization algorithm [11] presents an exponential speed-up. Intriguingly, Google reported in 2019 that they ran a random number generator algorithm on a superconducting processor containing 53 qubits in 200 seconds, which would most likely take several times longer for a classical supercomputer to solve [12]. It is anticipated that quantum computers will excel in exceedingly complex problems, while many simpler tasks may not see any speed-up at all compared to the classical regime. Hence, quantum- and classical computers are envisioned to coexist to utilize the strength of each technology.

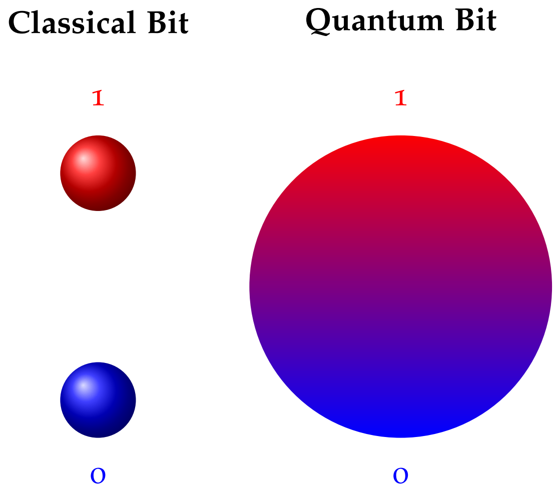


Figure 2.1: Conceptual illustration of the two-level classical bit, which are restricted to the boolean states 1 (true) or 0 (false), and the quantum bit that can be in any superposition of the states 0 or 1.

Quantum computing is a highly sought-after goal, but there are extensive challenges that needs to be adressed. Controlling a complex many-qubit system is difficult, since it is not always possible to establish interactions between qubits [7] and maintain entaglement over both time and distance. Additionally, decoherence and other quantum noise occurs as a result of the high volatility of quantum states, making quantum state manipulation prone to errors. The *quantum error correction* protocols and the theory of *threshold theorem* deals with this vulnerability, stating that noise most likely does not pose any fundamental barrier to the performance of large-scale computations [4].

2.1.1 Quantum computing requirements

As ever-promising the concepts of quantum technology are, the physical realizations are in the preliminary stage of development. Here we will concretize critical principles for a physical realisation of a quantum platform.

> "I always said that in some sense, these criteria are exactly the ones that you would teach to kindergarten children about computers, quantum or otherwise" DiVincenzo [13]

DiVincenzo formulated in the year of 2000 seven basic criteria for a physical qubit system with a logic-based architecture [7].

1. A scalable physical system with well characterized qubits
2. The ability to initialize the state of the qubits to a simple initial system
3. Have coherence times that are much longer than the gate operation time
4. Have a universal set of quantum gates
5. Have the ability to perform qubit-specific measurements

These five criteria must be met for a quantum platform to be considered a quantum computer.

2.2 Quantum communication

Quantum communication refers to the transfer of a state of one quantum system to another. Since information can be stored in qubits, we picture *flying qubits* that transfer information from one location to another [14]. The benefits of using flying qubits are in particular valued in quantum cryptography, since the quantum nature of qubits can be exploited to add extra layers of security [4].

Consider the example of encrypting a digitally transmitted conversation. It is difficult to avoid someone eavesdropping on a conversatio. However, the problem is diminished if the eavesdropper does not speak the language, keeping the information in the conversation safe. This is the original idea of encryption, such that the information has been encrypted into something incomprehensible for any eavesdropper. A common practice is to encrypt information and share a public key, which everyone can read, and a private key, only known for the sender and receiver of information. This should be sufficient to keep the information secure, given that the complexity of the private key is impenetrable.

Importantly, we live in a digital world where most of our actions are increasingly being stored as information, and we could imagine that the eavesdropper in the latter example stored the conversation. Even if the content of the conversation was encrypted, it still presents a challenge, since encrypted information stored today could be deciphered in ten or twenty years' time . Consequently, finding an encryption method that could make information either impossible to eavesdrop on or make the security unbreakable forever is very desirable. This is the ultimate goal of quantum cryptography [4].

Consider the example of information encoded into a qubit as a superposition of two quantum states. Now, if a wild eavesdropper would try to measure the information, the nature of quantum physics tells us that the original configuration would be destroyed and the receiver would be alerted of the eavesdropper. Furthermore, if the eavesdropper would try to make a copy of the message, the copying itself would be limited of the no-cloning theorem [15] which declare that quantum states cannot be copied.

A clever approach to ensure confidentiality is to send the encryption key before sending the actual encrypted information. If the key is received unperturbed, the key remains secret and can be safely employed. If it turns out perturbed, confidentiality is still intact since the key does not contain any information and can be discarded. This approach is termed the *quantum key distribution* (QKD) [15, 16]. It should be noted that this requires both the sender and receiver to have access to methods for sending, receiving and storing qubit states, such as a quantum computer. Additionally, the sender and receiver will need to initially exchange a common secret which is later expanded, making quantum key *expansion* a more exact term for QKD [4, 16].

Most applications and experiments use optical fibers for sending information via photons, with the distance regarded as the main limitation. This is because classical repeaters are unable to enhance quantum information because of the no-cloning theorem, making photon loss in optical fiber cables inevitable. Thus, quantum communication must reinvent the repeater concept, using hardware that preserves the quantum nature [17] and are compatible with wavelengths used in telecommunication. Nonetheless, secure QKD up to 400 km has recently been demonstrated using optical fibres in academic prototypes [18].

2.3 Quantum sensing

Measurements are part of our digital world today to a great extent. There would be no way to exchange goods, services or information without reliable and precise measurements [17]. Thus, improving the accuracy of sensors for all types of measurements is desirable. One potential method to improve measurement accuracy, resolution and sensitivity is utilizing quantum sensors. Quantum sensors exploit quantum properties to measure a physical quantity [19]. This is possible because quantum systems are highly susceptible to pertubations to its surroundings, and can be used to detect physical properties such as either temperature or an electrical or magnetic field [19].

For a quantum system to be able to function as a quantum sensors, a few criterias needs to be met. Firstly, the quantum system needs to have discrete and resolvable energy levels. The quantum system also needs to be controllably initialised into a state that can be identified and coherently manipulated by time-dependent fields. Lastly, the quantum system needs to be able to interact with the physical property one wants to measure through a coupling parameter [19].

It is also possible to also exploit quantum entanglement to improve the precision of a measurement. This gain of precision is used to reach what is called the Heisenberg-limit, which states that the precision scales as the number of particles N in an idealized quantum system [17, 19], while the best classical sensors scale with $\sqrt{N}$.

2.4 Available quantum platforms

Many different quantum platforms have been physically implemented, and this section will serve as a brief overview of the current status. For a more thorough review of qubit implementations, the reader is directed to Refs. [8, 17].

Superconducting circuits can be used in quantum computing, since electrons in superconducting materials can form Cooper pairs via an effective electron-electron attraction when the temperature is lower than a critical limit. Below the limit, electrons can move without resistance in the material [20]. Exploiting this intrinsic coherence, qubits can be made by forming microwave circuits based on loops of two superconducting elements separated by an insulator, also known as Josephson tunnel junctions [17]. Today, superconducting Josephson junctions are the most widely used quantum platform, but they requires very low temperature (mK) to function, making them costly to use. Additionally, the current devices experience a relatively short coherence time, causing challenges in scaling up.

Single photons is an eligible quantum platform that can be implemented as qubits with one-qubit gates being formed by rotations of the photon polarization. Its use in fiber optics are less prone to decoherence, but faces challenges since the more complex photon-photon entanglement and control of multi-qubits is strenuous [8].

By fixing the nuclear spin of solid-state systems, it is possible to implement a quantum platform that experience long spin coherence. This enables the manipulation of qubits that utilize electromagnetic fields, making one-qubit gates realizable.

The isolated atom platform is characterized by its well-defined atom isolation. Here, every qubit is based on energy levels of a trapped ion or atom. Quantum entanglement can be achieved through laser-induced spin coupling, however scaling up to large atom numbers induce problems in controlling large systems and cooling of the trapped atoms or ions.

A quantum dot (QD) can be imagined as an artifical atom which is confined in a solid-state host. As an example, a quantum dot can occur when a hole or an electron is trapped in the localized potential of a semiconductor's nanostructure. QDs exhibit similar coherence potential as the isolated atom platform, but without the drawback of confining and cooling of the given atom or ion [17]. Moreover, it is possible to limit decoherence due to nuclear spins by dynamic decoupling of nuclear spin noise and isotope purification [8].

A QD can normally be defined litographically using metallic gates, or as self-assembled QDs where a growth process creates the potential that traps electrons or holes. The difference between them is a question of controllability and temperature, since the metallic gates is primarily controlled electrically and operate at < 1 K, while self-assembly QDs are primarily controlled optically at ~ 4 K [8]. Despite requiring very low temperatures, QDs have the potential for fast voltage control and opticial initialization. As with trapped ions, electrostatically defined quantum dots experience a short-range exchange interaction, imposing a limitation for quantum computing and quantum error correction protocols. A potential solution could include photonic connections between quantum dots. Indeed, self-assembled quantum dots couple strongly to photons due to their large size in comparison to single atoms. However, the size and shapes of self-assembled quantum dots are decided randomly during the growth process, causing an unfavourable large range of optical absorption and emission energies [8].

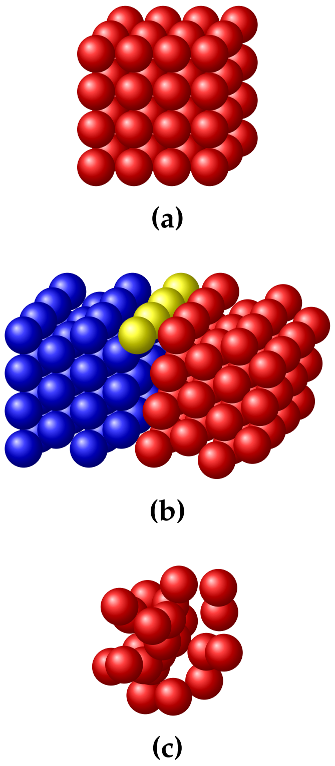
Lastly, we will turn towards point defects in bulk semiconductors as a physical implementation of a quantum platform. Point defects shares many of the attributes of quantum dots, such as discrete optical transitions and controllable coherent spin states. Depending on the semiconductor host and the defect system of interest, they may exhibit extended coherence times and greater optical homogeneity than other quantum dot systems. Before we dwell into the intricacies of point defect qubits as a building block for QT, we will provide the neccessary background for the crystal- and electronic structure of semiconductors.

2.5 Introduction to semiconductor physics

The interactions between atoms and the resulting characteristics of matter form the foundation of materials science. The applications of materials science are extensive, with examples such as a bottle of water or to a chair to sit in.

Solid materials, like plastic bottles, are formed by densely packed atoms. These atoms can randomly occur through the material without any long-range order or periodically ordered in small regions of the material, which would categorize the material as either an *amorphous solid* or *polycrystalline solid*. A third option is to have these atoms arranged with infinite periodicity, making the material a *crystalline solid* or more commonly named a *crystal*. The three options are visualised in figure 2.2. Hereon, we will focus on crystalline solids.

The periodicity in a crystal is defined in terms of a symmetric array of points in space called the *lattice*, which can be simplified as either a one-dimensional array, a two-dimensional matrix or a three dimensional vector space, depending on the material. At each lattice point we can add an atom to make an arrangement called a *basis*. The basis can be one atom or a cluster of atoms having the same spatial arrangement. Every crystal has periodically repeated building blocks called *cells* representing the entire crystal. The smallest cell possible is called a *primitive cell*, but such a cell only allows lattice points at its corners and it is often quite rigid to work with when the structure becomes complex. As a solution, we will consider the *unit cell*, which allows lattice points on face centers and body centers.


(a)

(b)

(c)

Figure 2.2: Schematic representation of different degrees of ordered structures, where (a) is a crystalline of a simple cubic lattice, (b) is a polycrystalline hexagonal lattice, and (c) is an amorphous lattice.

One example of a crystal structure is the perovskite structure. Compounds with this structure are characterized by having an ABX_3 stoichiometry whose symmetri belong to one of 15 space groups identified by Lufaso & Woodward [21], such as the cubic, orthorombic and tetragonal. For our purpose, we will be looking into when the X atom is oxygen, and refer to the oxygen-perovskite ABO_3. The A atom is nine- to 12-fold coordinated by oxygen, while the B atom is sixfold coordinated by oxygen, and the BO_6 octahedra are connected to the corners in all three directions as visualized in figure 2.3.

The motivation behind the research on perovskites is related to the large amount of available ABO_3 chemistries, where a significant portion of these take the perovskite structure. Perovskites have a broad specter of applications, ranging from high-temperature superconductors [22] and ionic conductors [23] to multiferroic materials [24]. Additionally, adding a perovskite-type compound to solar cells has reportedly resulted in higher performance efficiencies while being cheap to produce and simple to manufacture [25, 26]. However, this includes the use of hybrid organic-inorganic compounds and excludes the use of oxygen.

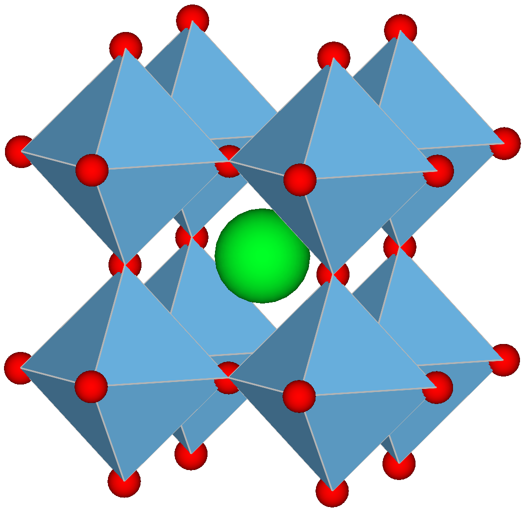


Figure 2.3: A crystal structure of $SrTiO_3$ which is a cubic perovskite. The red atoms are oxygen, whereas the green atom is strontium, and inside every corner-sharing BO_6 octahedral unit is a titanium atom.

Isolated atoms have distinct energy levels, where the Pauli exlusion principle [27] states for fermions that each energy level can at most accomodate two electrons of opposite spin. In a solid, the discrete energy levels of the isolated atom spread into continuous energy bands since the wavefunctions of the electrons in the neighboring atoms overlap. Hence, an electron is not neccessarily localized at a particular atom anymore. This is exemplified as every material has a unique band structure, similar to every human having their unique fingerprint.

Knowing which energy bands are occupied by electrons is the key in understanding the electrical properties of solids. The highest occupied electron band at 0 K is called the valence band (VB), while the lowest unoccupied electron band is called the conduction band (CB). The energy gap between the maximum VB and the minimum CB is known as the band gap, and its energy is denoted as E_g. If a material can be classified as a semiconductor depends on the band gap and the electrical conductivity. As an example, Silicon is commonly thought of as a semiconductor, and has a band gap of about 1.12 eV at 275 K [28].

To be able to accelerate electrons in a solid using an electrical field, they must be able to move into new energy states. At 0 K, the entire valence band of a semiconductor is full with electrons and there are no available states nearby, making it impossible for current to flow through the material. This can be solved by using either thermal or optical energy to excite electrons from the valence band to the conduction band, in order to *conduct* electricity. At a given temperatuer, some semiconductors will have electrons excited to the conduction band solely from thermal energy matching the energy band gap [29].

In some scenarios, thermal or optical energy is not sufficient for an excitation since the energy bands are also dependent on the crystal momentum. A difference in the momentum of the minimal-energy state in the conduction band and the maximum-energy state in the valence band results in an *indirect bandgap* as seen in figure 2.4a. If there is no difference at all, the material has a *direct bandgap*, which is visualized in figure 2.4b.

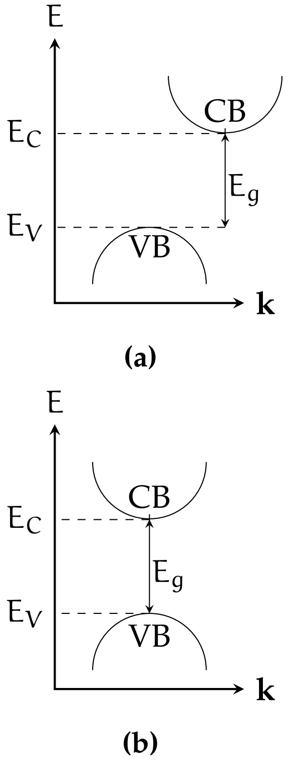


Figure 2.4: A schematic drawing of (a) an indirect- and (b) a direct bandgap.

Electrons in semiconductor materials can be described according to the Fermi-Dirac distribution

$$f(E) = \frac{1}{1 + e^{(E-E_F)/kT}},$$

where k is Boltzmann's constant, T is temperature, E is the energy and E_F is the Fermi level. The Fermi-Dirac distribution gives the probability that a state will be occupied by an electron, and at $T = 0$ K, every energy state lower than E_F is occupied by electrons while the opposite is true for energy states above E_F [29].

2.5.1 Point defects in semiconductors

In real life, a perfect crystal without any symmetry-breaking flaw does not exist. These flaws are known as defects and can occur up to three dimensions. An example one-dimensional defect is known as a *line defect*, while two dimensional defects can be *planar defects*, and in three dimensions we have *volume defects*. Lastly, defects can also occur in zero dimensions and are then termed *point defects*. Point defects normally occur as either vacancies, interstitially placed atoms inbetween lattice sites (called interstitials) or as substitution of another existing atom in the lattice.

Defects can greatly influence both the electronic and optical propertires of a material. A substitional defect may be an unintentionally introduced impurity or an antisite, but they can also be intentionally inserted, an approach normally known as *doping*. Doping can result in an excess of electrons or holes, making the semiconductor either an n- or p-type, respectively. Consequently, the semiconductor will have energy levels in the (forbidden) band gap that originates from the defects. If the energy levels introduced are closer than $\sim$ 0.2 eV to the band egdes, they are termed *shallow* defects.

Shallow defects can contribute with either excess electrons to the conduction band, or excess holes to the valence band. However, the induced charge carriers (electrons or holes) interact strongly with the band egdes, resulting in a delocalized wavefunction.

For the opposite case, if the energy levels rests closer to the middle of the semiconductor's gap, the introduced defects are known as *deep level* defects. Deep levels may be of intrinsic or extrinsic origin, and have highly localized electron wavefunctions. This might assure the isolation required for long coherence times, which is an appealing promise in quantum technological advances.

Deep levels can be unfortunate in semiconductors since they can interact with the charge carriers, potentially modifying the desired electronic or optical property of the material. Deep level defects can function as electron-hole recombination centers, or to trap charge carriers, yielding the commonly used name deep level *traps*. Both of the given situations results in a lower concentration of charge carriers, which showcase why deep levels normally are unwanted in semiconductor devices. However, deep level defects may be beneficially used in quantum technology.

2.5.2 Optical defect transitions

Optical transitions refers to excitation or de-excitation of charge carriers due to either emission or absorption of electromagnetic radiation.Figure 2.5 represents a configuration coordinate (CC) diagram of a defect transition. The y-axis is the energy E, while the x-axis is the configuration coordination Q, which represents the atomic displacement from the equilibrium position Q_{GS}. The lowest point in the lower parabola is known as the ground state (GS) configuration Q_{GS}, which is the most stable atomic position, while for the upper parabola it is known as the excited state configuration Q_{ES}. The dotted lines represent vibronic excitations to the energy of the ground state Q_{GS} for the lower parabola, while it represents Q_{ES} for the higher parabola.

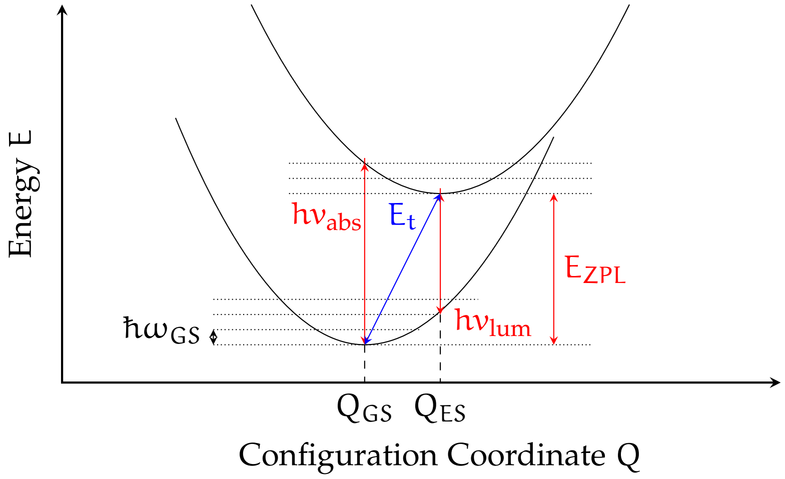


Figure 2.5: A schematic representation of a configuration coordination diagram based on Ref. [30].

The optical transitions in figure 2.5 are marked with red arrows. During slow transitions, such as during thermodynamic defect transitions, the original configuration have time to rearrange due to phonon vibrations. This is schematically drawn as the blue arrow, where the energy E_t equals the ionization energy or the position of the defect level. Optical transitions, on the other hand, are marked in red and occur in a short time range such that the original configuration does not change. They can appear in the exchange of charge carriers with the band egdes, and in a defect's internal excited state, with the latter scenario being most relevant for this thesis.

Consider a defect that rests in the ground state configuration Q_{GS}. Suddenly, it absorbs a photon with energy $h\nu_{abs}$ and occupies an excited vibronic state of the upper parabola after a vertical transition. Through lattice reconfigurations, the defect will move towards the bottom of the upper parabola, also known as Q_{ES}. Eventually, it will relax to the lower parabola by emitting a photon with energy $h\nu_{lum}$, also known as a zero-phonon line (ZPL) of energy E_{ZPL}. On the other hand, any transitions between vibronic excitation levels are phonon-related. How strong the electron-phonon interaction is can be quantified by the Huang-Rhys factor S [31]. If the two parabolas in figure 2.5 have the same configuration of Q, emission into the ZPL is enabled and $S \sim 0$. The stronger the coupling, the smaller amount of emission in the ZPL.

The optical properties of a host material can be greatly influenced by defects, in particular the ES to GS transition that can occur in a defect, as discussed for figure 2.5. If the defect were to fascilitate the emission of single photons with a detectable time inbetween together with a distinguishable ZPL, the defect would be referred to as a single photon source (SPS). The criteria for SPS are not met in many materials or defect systems, since charge-state transitions often comprise interactions with either the VB or the CB. Thus, most SPSs' GS and ES levels are situated within the band gap of a host material. Consequently, mostly wide-band gap semiconductors are used as host materials for SPSs.

2.6 Semiconductor candidates for quantum technology

The properties of point defects are promising in a quantum technological perspective. We have seen that point defects can fasciliate deep energy levels within the band gap of the semiconductor, and provide isolation in the solid-state matrix as a result from a high degree of localization of the defect orbitals. If the host material have a small spin-orbit coupling, it could provide long coherence times for a deep level trap in localized and high-spin states. Additionally, point defects have the potential to be single-photon sources, giving rise to sharp and distinguishable optical transitions, where a significant amount of the emission can be of the energy E_{ZPL}. This is in particular seen in wide-bandgap semiconductors, and combined with a weak electron-phonon interaction, can have the capacity to be fabricated as a high-fidelity SPS with a significant ZPL part.

In this section we will provide specific examples of a variety of promising candidates, and what properties they possess that makes them auspicious. Additionally, we will briefly mention what the challenges with the candidates are, and why it is important to explore other viable options.

2.6.1 Diamond - the benchmark material for QT

The most studied point defect system is the nitrogen-vacancy (NV^{-1}) in diamond. Figure 2.6 schematically shows the negative charge state of the electronic structure in diamond. Panel 2.6a shows the electronic states that correspond to the difference for an isolated atom and a lattice of atoms, as a superposition of sp^3 orbitals that generates valence and conduction bands. In panel 2.6b, a vacancy has been created by removing a carbon atom, and the four orbitals interact with each other resulting in two new states with a_1 and t_2 symmetry due to dangling bonds. Substituting a carbon atom with a nitrogen atom further splits the t_2-states into two new states. The states $a(1)$ and $e_{x'}e_y$ are of importance, as they are the GS and the ES of the qubit defects, respectively. Here, an optical spin-conserving transition can occur due to a laser light of correct wavelength [32], as exemplified from the discussion from the last section.

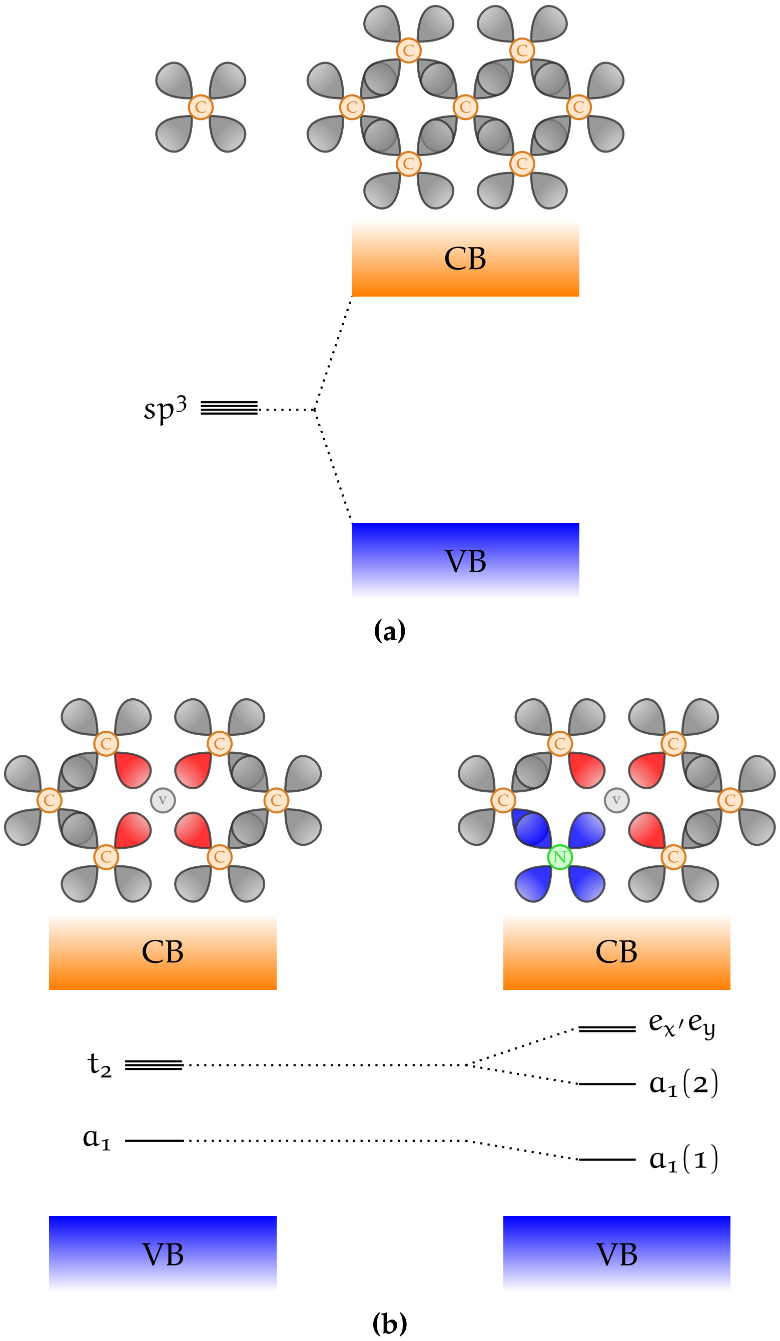


Figure 2.6: A schematic representation of the electronic structure of the NV^{-1} defect in a tetrahedrally coordinated semiconductor, exemplified by diamond. Figure adapted from Ref. [32].

The nitrogen-vacancy in diamond is a prominent single-photon source up to room temperatures. This involves initializing, manipulating and reading out of the qubit state using optical and electric excitations, and electric and magnetic fields [32]. The potential qubit system have promising applications in quantum- communication and computation, with a demonstrated entanglement between two NV center spins that are separated by 3 m [33]. Nevertheless, perhaps the most propitious application can be seen in quantum sensing as high-sensivity magnetometer with nanoscale resolution [34].

Unfortunately, the NV-center display several drawbacks that may limit the use in quantum communication and computation. In particular, the amount of emission into the zero-phonon line is only 4% at 6 K [35]. The emission of the qubit center is not completely compatible with current optical fiber technologies, since the emission is in the red wave-length specter. Additionally, fabricating materials of diamond is far from unchallenging and serves as a signficant incentive to find other promising qubit candidates.

2.6.2 Qubit material host requirements

Therefore, we turn to the search of other QT compatible hosts that offers similar capabilities, but that are more user-friendly. In particular, we need to search for new promising materials that can host a potential point defect. Weber *et al.* [6] proposed in 2010 four criteria that should be met for a solid-state semiconductor material hosting a qubit defect, whereas some of the criteria has already been discussed. An ideal crystalline host should have [6]

(H1) A wide-band gap to accomodate a deep center.

(H2) Small spin-orbit coupling in order to avoid unwanted spin flips in the defect bound states.

(H3) Availability as high-quality, bulk, or thin-film single crystals.

(H4) Constituent elements with naturally occuring isotopes of zero nuclear spin.

Table (2.1) lists several material host candidates that exhibit promising band gap capable of accommodating a deep level defect. For example, the spin-orbit splitting is an indication of the strength of the spin-orbit interaction, and is taken at the Γ point from the valence-band splitting. A smaller value may indicate less susceptibility to decoherence.

Criterion (H3) is important for scalability and further potential for a large-scale fabrication. The given candidate hosts provided in table (2.1) can all be grown as single crystals, but with varying quality and size.

Material	Band gap E_g (eV)	Spin-orbit splitting Δ_{so} (meV)	Stable spinless nuclear isotopes?
3C-SiC	2.39	10	Yes
4H-SiC	3.26 [36]	6.8	Yes
6H-SiC	3.02	7.1	Yes
AlN	6.13	19 [37]	No
GaN	3.44	17.0	No
AlP	2.45	50 [38]	No
GaP	2.27	80	No
AlAs	2.15	275	No
ZnO	3.44 [39]	-3.5	Yes
ZnS	3.72 [40]	64	Yes
ZnSe	2.82	420	Yes
ZnTe	2.25	970	Yes
CdS	2.48	67	Yes
C (Diamond)	5.5	6	Yes
Si	1.12	44	Yes

Table 2.1: Table taken from Gordon *et al.* [32] that lists a number of tetrahedrally coordinated hosts whose band gaps are larger than 2.0 (eV), and compares it to diamond and Si. All experimental values are from Ref. [28], except for where explicity cited otherwise.

Normally, nuclear spin is a major source of decoherence for all semiconductor-based quantum technologies. This would exclude the use of all elements in odd groups in the periodic table, since these elements exhibit nonzero nuclear spin. As a result, the spin-coherence time of a paramagnetic deep center [6] might increase. However, nuclear spin can also induce additional quantum degrees of freedom for applications in the proper configuration [41]. Therefore, criterion (H4) is not a strict requirement but is a general recommendation for reducing decoherence time.

Weber *et al.* [6] use criteria $(H1) - (H4)$ to specifically find analogies to the NV^{-1} center in other material systems, thus leaving the discussion of other criteria out, such as the choice of crystal system. The atomic configuration and crystal structure of a material strongly influences the properties of a defect, since a defect's orbital and spin structure is dependent on its spatial symmetry [41]. In particular, it is the point group that decides which multiplicity a given energy level should have [42]. A higher defect symmetry group generally facilitates degenerate states, which may give rise to high spin states according to Hund's rules [41, 43]. Inversion symmetry in the host crystal can also be beneficial, resulting in reduced inhomogenous broadening and spectral diffusion of optical transitions as a consequence of being generally insensitive to external electric fields [41].

2.6.3 Silicon carbide

Silicone carbide (SiC) is an emerging quantum platform that exists in a wide variety of polytypes, with 3C, 4H and 6H being the most prominent configurations. Several of the polytypes have been demonstrated to host SPEs with a slightly different emitter characteristic, which provides the opportunity to select the desired properties based on the variety of lattice configurations and point defects available [6, 45, 46]. While 3C has a cubic structure, 4H has a hexagonal structure with both hexagonal (h) and pseudo-cubic (k) lattice sites. 6H is also a hexagonal structure, but with the three orientations that are labelled h, k_1 and k_2. Importantly, SiC in the three varieties experience wide-band gaps, low spin orbit coupling and stable spinless nuclear isotopes [6, 28, 36], as seen from table 2.1. Furthermore, SiC benefits from mature fabrication on the wafer-scale, which checks the last of the four (H1-H4) QT host requirements, marking it as a suitable quantum material platform.

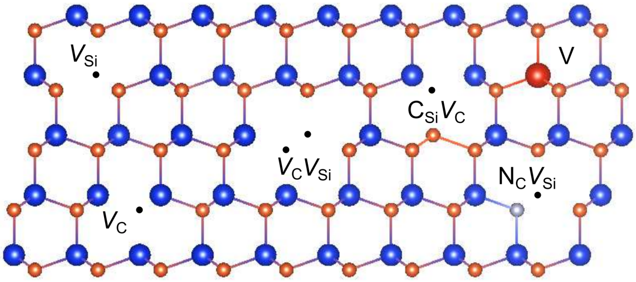


Figure 2.7: Schematic illustration of various point defects in 4H-SiC, where Si atoms are blue while C atoms are orange. The illustration includes the point defects Si vacancy (V_{Si}), C vacancy (V_C), divacancy ($V_{Si}V_C$), carbon antisite-vacancy pair ($C_{Si}V_C$), nitrogen-vacancy (N_CV_{Si}) and the vanadium impurity (V). Figure taken from Ref. [44].

The most studied emitters in SiC include the carbon antisite-vacancy pair $C_{Si}V_C$ that emits in the red, the silicon vacancy V_{Si} that emits in the near infra-red, and the divacancy ($V_{Si}V_C$) and the nitrogen-vacancy center (N_CV_{Si}) that both emit at near-telecom wavelengths. Thus, the two latter emitters could potentially ease the integration with optic fiber technologies as compared to e.g. the NV^-. Additionally, the four different point defects have all been identified as room-temperature SPEs with demonstrated coherent spin control [47]. Illustrations of several configurations of emitters in 4H-SiC are included in figure 2.7.

2.6.4 Alternative promising material hosts

Single photon emitters have been observed in other semiconductor materials, however most of the emitters are yet to be identified or are in an early stage of identification. Therefore, specific details about spin- or emission-related structure are yet to be implemented. In this section we will briefly mention recent promising materials for QT.

One immediate potential candidate is silicon, considering the favorable device fabrication processes that are available. It has demonstrated that phosphorous impurities at Si sites can store a quantum state for over 30 seconds, enabling their use in a potential Kane quantum computer [48]. Unfortunately, the P impurity lack any single photon source capabilities. Recently, however, the G-center arising from the carbon-interstitial carbon-substitutional (C_sC_i) complex was identified as an promising SPE candidate with single photon emissions at telecom wavelength [49].

Other materials that emits individual photons have been detected in other wide-band gap semiconductors, including ZnO, ZnS, GaAs, GaN and AlN [48, 50], although the defect centers responsible for most of the SPE lines have yet to be identified. Additionally, challenges due to the specific materials complicate the implementation of defects for QT. ZnO and ZnS experience a broad emission due to a large photon involvement. GaAs is promising since it has been demonstrated as a SPS, but demonstration of spin manipulation is still in an early phase[50]. GaN and AlN, on the other hand, are more susceptible to a more narrow emission, where room-temperature SPE has been demonstrated for both GaN [51] and wurtzite AlN films [52]. The defect levels for AlN films have been tentatively assigned to the nitrogen-vacancy and divacancy complexes, but they tend to occur too close to the band edges for any SPE [48, 53].

Recent advances in material growth have enabled the use of hole spin-based semiconductors, such as SiGe quantum wells due to their low disorder and large intrinsic spin-orbit coupling strength [54]. Promising materials can also emerge from placing an impurity next to a vacancy. Cation vacancies in possible structures tend to be negatively charged, thus the impurities should act as donors. Therefore, the self-activation center in ZnSe can be a promising defect [6], but is still in the early stage of development.

Two-dimensional materials such as hexagonal boron nitride (h-BN), MoS_2, WSe_2 and WS_2 are also of interest as quantum platforms [55, 56]. The structure of h-BN exists in single- or multilayers, and it has been demonstrated a broad range of stable room-temperature single-photon emitters [57, 58]. In WSe_2, $MoSe_2$ and WS_2, there has been experimentally discovered optical excitation of defects, while also electrical excitation of defects for WS_2 [56]. However, secure identification for the source of the emission is yet to be established [56, 59, 60].

2.6.5 Associated challenges with material host discovery

The idea of finding new potential host candidates to utilise point defects in QT is challenging. Recall, we have made four criteria that deals with the requirements; (H1) band gaps, (H2) spin-orbit coupling, (H3) availability and (H4) spin-zero isotopes, but more criterias may be needed and the excisting ones refined.What we do know is that there are major advantages if materials exhibit properties such as isolation in the lattice and weak electron-photon interaction, however, the process to provide any quantity of measurements are through approximations and material-specific properties. These approximations does not neccessarily capture quantum properties well.

Furthermore, the identified candidates constitues an immensely selective group of only a handful potential hosts which have been discovered by serendipitet. As an example, most known potential hosts are elemental (unary) or binary compounds. This is probably due to the increasing complexity dealing with an additional level of interactions in the lattice. Therefore, there are reasons to believe that many potential hosts are yet to be discovered, which serves as a motivation for studies involving exploratory research for new candidates.

Chapter 3

The new paradigm of novel materials discovery

The discovery of novel materials enables the development of technological advances that are neccessary to overcome challenges faced in the society, and is a principal ingredient in defining who we are and what we have become. We have witnessed the material epochs starting from the bronze age, iron age and up to the era of modern silicon technologies, and we find novel materials transformed into other industries as well [61, 62].

However, the modern times have radically changed the methods of discovering novel materials. In the last decades, we have observed the generation of huge amount of theoretical and experimental data, commonly known as *Big data*. In the fields of computational material science, this inception is mainly enabled due to the success of *the density functional theory* (DFT). Conversely, to keep up with the pace of data generation, a new field named *Data science* combines the interdisciplinary fields of mathematics, statistics, computer science and programming to solve the challenge of extracting knowledge from unfeasibly big and complex data [63, 64]. This is considered the fourth paradigm of science, and is visualized together with the previous paradigms of science in figure 3.1.

This chapter aims to provide the neccessary understanding of the new research paradigm in the context of computational materials science and novel materials discovery. Starting from the beginning, we will be looking into information gained by *ab-initio* calculations, which means "from first principles". Since DFT is not a theory one simply understands, we will begin with an initial discussion of why it is difficult to calculate interactions between particles, followed up by a review of key approximations and methods regarding the theory. However, even if density functional theory solves some problems, it also introduce new challenges, which will be thoroughly discussed.

Thereafter, we try to provide a logical sequence into the emergence of

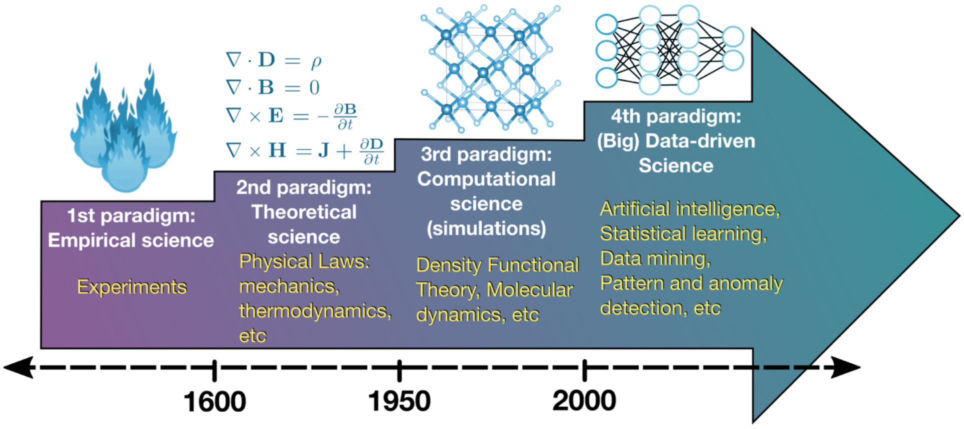

$\nabla \cdot \mathbf{D} = \rho$
$\nabla \cdot \mathbf{B} = 0$
$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$
$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}$
1st paradigm: Empirical science
Experiments
2nd paradigm: Theoretical science
Physical Laws: mechanics, thermodynamics, etc
3rd paradigm: Computational science (simulations)
Density Functional Theory, Molecular dynamics, etc
4th paradigm: (Big) Data-driven Science
Artificial intelligence, Statistical learning, Data mining, Pattern and anomaly detection, etc
1600
1950
2000

Figure 3.1: The four science paradigms: empirical, theoretical, computational, and data-driven. Figure taken from Ref. [64], which was originally adapted from Ref. [63].

high-throughput (HT) methods and tools neccessary to handle the resulting information. Finally, we review a state-of-the-art approach of novel materials discovery enabled by the new paradigm of big data and data science.

3.1 Introduction to density functional theory

Initially, the method of the calculations can be regarded as a black box where one provides the structure of a material as input, which the black box in return feed us the outcome in terms of interesting constants or a different structure. The black box is based on a successful theory called density functional theory (DFT), which is an approach for predicting physical properties of solid-state systems. In this and the next section we provide the neccessary knowledge to understand what is happening inside the black box, and the quality of the output.

To fully understanding what challenges the density functional theory solves, we will need to investigate how we can calculate the forces acting inside a crystal. Since these forces are happening on a microscopic scale, we will need to utilize the theory of quantum mechanics.

3.1.1 The Schrödinger equation

In principle, we can describe all physical phenomenas of a system with the wavefunction $\Psi(\mathbf{r}, t)$ and the Hamiltonian $\hat{H}(\mathbf{r}, t)$, where $\mathbf{r}$ is the spatial position and t is the time. Unfortunately, analytical solutions for the the time-dependent Schrödinger equation,

$$i\hbar\frac{\partial}{\partial t}\Psi(\mathbf{r},t) = \hat{H}(\mathbf{r},t)\Psi(\mathbf{r},t), \tag{3.1}$$

are extremely rare. More conveniently, we can generate a general wavefunction by a summation of eigenfunctions,

$$\Psi(\mathbf{r},t) = \sum_{\kappa} c_{\kappa}\psi_{\kappa}(\mathbf{r},t), \tag{3.2}$$

where c_{κ} is a constant and ψ_{κ} is the κ-th eigenfunction. A general wavefunction does not neccessarily describe stationary states, and consequently does not have distrinct energies but is rather represented statistically from the expectation value

$$E = \sum_{\kappa} |c_{\kappa}| E_{\kappa}. \tag{3.3}$$

Solving the Schrödinger equation for a general wavefunction is rather troublesome, but luckily we can use the eigenfunctions instead, transforming equation 3.1 into the time-independent Schrödinger equation for eigenfunctions

$$\hat{H}\psi_{\kappa}(\mathbf{r}) = E_{\kappa}\psi_{k}(\mathbf{r}), \tag{3.4}$$

where E_{κ} is the eigenvalue of the κ-th eigenstate $\psi_{\kappa}(\mathbf{r})$. The eigenfunctions have distinct energies, and the state with the lowest energy is called the ground state. They have the attribute that they are orthogonal and normalized with respect to

$$\left\langle \psi_{\kappa}(\mathbf{r}) | \psi_{\kappa'}(\mathbf{r}) \right\rangle = \delta_{\kappa\kappa'}. \tag{3.5}$$

The symmetry of an eigenfunction depends on the symmetry of the potential $V_{ext}(\mathbf{r})$ and the boundary conditions [65].

3.1.2 The many-particle Schrödinger equation

As we extend the theory to include many-particle systems, we will gradually explain and add the different contributions that make up the many-body Hamiltonian. During this process, we will neglect any external potential applied to the system.

If we place a simple electron with mass m_e in its own system, it will be in possession of kinetic energy. Instead of just one electron, we can place N_e electrons, and they will together have the total kinetic energy

$$T_e = -\sum_{j=1}^{N_e} \frac{\hbar^2\nabla_j}{2m_e}. \tag{3.6}$$

All the electrons are negatively charged, causing repulsive Coulomb interactions between each electron, totalling to

$$U_{ee} = \sum_{j=1}^{N_e} \sum_{j'<j} \frac{q^2}{|r_j - r_{j'}|}. \tag{3.7}$$

The summation voids counting each interaction more than once. Simultaneously, we can place N_n nuclei with mass m_n in the same system, accumulating the kinetic energy

$$T_n = -\sum_{a=1}^{N_n} \frac{\hbar^2 \nabla_a}{2m_n}. \tag{3.8}$$

As in the example with electrons, the nuclei are also experiencing repulsive interactions between every single nucleus, adding up the total interactions as

$$U_{nn} = \sum_{a=1}^{N_n} \sum_{a'<a} \frac{q^2 Z_a Z_{a'}}{|R_a - R_{a'}|}. \tag{3.9}$$

where Z_a is the atom number of nuclei number a. The system now contains N_e electrons and N_n nuclei, thus we need to include the attractive interactions between the them,

$$U_{en} = -\sum_{j=1}^{N_e} \sum_{a=1}^{N_n} \frac{q^2 Z_a}{|r_j - R_a|}. \tag{3.10}$$

Together, these equations comprise the time-independent many-particle Hamiltonian

$$\begin{aligned} \hat{H} = &-\sum_{j=1}^{N_e} \frac{\hbar^2 \nabla_j}{2m_e} - \sum_{a=1}^{N_n} \frac{\hbar^2 \nabla_a}{2m_n} + \sum_{j=1}^{N_e} \sum_{j'<j} \frac{q^2}{|r_j - r_{j'}|} \\ &+ \sum_{a=1}^{N_n} \sum_{a'<a} \frac{q^2 Z_a Z_{a'}}{|R_a - R_{a'}|} - \sum_{j=1}^{N_e} \sum_{a=1}^{N_n} \frac{q^2 Z_a}{|r_j - R_a|}. \end{aligned} \tag{3.11}$$

A few problems arise when trying to solve the many-particle Schrödinger equation. Firstly, the amount of atoms in a crystal is very, very massive. As an example, we can numerically try to calculate the equation 3.7 for a 1mm^3 silicon-crystal that contains $7 \cdot 10^{20}$ electrons. For this particular problem, we will pretend to use the current fastest supercomputer Fugaku [66] that can calculate 514 TFlops, and we will assume that we need 2000 Flops to calculate each term inside the sum [65], and we need to calculate it $N_e \cdot N_e/2$ times for the (tiny) crystal. The entire electron-electron interaction calculation would take $2.46 \cdot 10^{19}$ years to finish for a tiny crystal. Thus, the large amount of particles translates into a challenging numerical problem.

Secondly, the many-particle Hamiltonian contains operators that has to be applied to single-particle wavefunctions, and we have no prior knowledge of how Ψ depends on the single-particle wavefunctions ψ_κ.

3.1.3 The Born-Oppenheimer approximation

The many-particle eigenfunction describes the wavefunction of all the electrons and nuclei and we denote it as Ψ_κ^{en} for electrons (e) and nuclei (n), respectively. The Born-oppenheimer approximation assumes that nuclei, of substantially larger mass than electrons, can be treated as fixed point charges. According to this assumption, we can separate the eigenfunction into an electronic part and a nuclear part,

$$\Psi_\kappa^{en}(\mathbf{r}, \mathbf{R}) \approx \Psi_\kappa(\mathbf{r}, \mathbf{R})\Theta_\kappa(\mathbf{R}), \tag{3.12}$$

where the electronic part is dependent on the nuclei. This is in accordance with the assumption above, since electrons can respond instantaneously to a new position of the much slower nucleus, but this is not true for the opposite scenario. From here, one can obtain the electronic and nuclear eigenfunction, with the derivation shown in A.1.

3.1.4 The Hartree and Hartree-Fock approximation

The next question in line is to find a wavefunction $\Psi(\mathbf{r}, \mathbf{R})$ that depends on all of the electrons in the system. The Hartree [65, 67] approximation to this is to assume that electrons can be described independently, suggesting the *ansatz* for a two-electron wavefunction

$$\Psi_\kappa(\mathbf{r}_1, \mathbf{r}_2) = A \cdot \psi_1(\mathbf{r}_1)\psi_2(\mathbf{r}_2), \tag{3.13}$$

where A is a normalization constant. This approximation simplifies the many-particle Shrödinger equation a lot, but comes with the downside that the particles are distinguishable and do not obey the Pauli exclusion principle for fermions.

The Hartree-fock approach, however, overcame this challenge and presented an anti-symmetric wavefunction that made the electrons indistinguishable [1]:

$$\Psi_\kappa(\mathbf{r}_1, \mathbf{r}_2) = \frac{1}{\sqrt{2}}\Big(\psi_1(\mathbf{r}_1)\psi_2(\mathbf{r}_2) - \psi_1(\mathbf{r}_2)\psi_2(\mathbf{r}_1)\Big). \tag{3.14}$$

For systems containing more than one particles, the factor $1/\sqrt{2}$ becomes the Slater determinant and is used to normalize the wave function.

3.1.5 The variational principle

So far, we have tried to make the time-independent Schrödinger equation easier with the use of an *ansatz*, but we do not neccessarily have an adequate guess for the eigenfunctions and the ansatz can only give a rough estimate in most scenarios. Another approach, namely the *variational principle*, states that the energy of any trial wavefunction is always an upper bound to the exact ground state energy by definition E_0.

$$E_0 = \langle\psi_0| H |\psi_0\rangle \leqslant \langle\psi| H |\psi\rangle = E \tag{3.15}$$

This enables a minimization of energy in terms of wavefunction parameters. A more thorough walk-through of the variational principle is included in appendix A.2.

3.2 The density functional theory

Hitherto we have tried to solve the Schrödinger equation to get a ground state wave function, and from there we can obtain ground state properties. One fundamental problem that exists when trying to solve the many-electron Schrödinger equation is that the wavefunction is a complicated function that depends on $3N_e$ variables[1].

Hohenberg and Kohn [68] showed in 1964 that the ground-state density $n_0(r) = |\Psi_0(r)|$ determines a general external potential, which includes U_{en}, up to an additive constant, and thus also the Hamiltonian [69]. From another point of view, the theory states that all physical ground-state properties of the many-electron system are unique functionals of the density [65]. A consequence of this is that the number of variables is reduced from $3N_e$ to 3, significantly reducing the computational efforts.

However, the scheme is not without limitations, as the density functional theory (DFT) can only be used to find all the ground-state physical properties if the exact functional of the electron density is known. And 57 years after Hohenberg and Kohn published their paper, the exact functional still remains unknown.

We will start this chapter with a brief mention of the Hohenberg-Kohn theorems and its implications, before we delve further into the Kohn-Sham equation.

[1] not including spin

3.2.1 The Hohenberg-Kohn theorems

Theorem 1. *For any system of interacting particles in an external potential* V_{ext}*, the density is uniquely determined.*

The theorem can be proved by utilising the variational principle for two different external potentials with the same ground state density. The proof is included in appendix A.3.1.

Theorem 2. *There exists a variational principle for the energy density functional such that, if* n *is not the electron density of the ground state, then* $E\left[n_0\right] < E\left[n\right]$.

From theorem 1, we know that the external potential is uniquely determined by the density, which in turn uniquely determines the ground state wavefunction. Therefore, all other observables of the system are uniquely determined and we can express the energy as function of the density,

$$E[n] = \overbrace{T[n] + U_{ee}[n]}^{F[n]} + U_{en}[n]. \tag{3.16}$$

where $F[n]$ is an universel functional known as the Hohenberg-Kohn functional. The proof for theorem 2 is found in appendix A.3.2.

3.2.2 The Kohn-Sham equation

So far, we have tried to make the challenging Schrödinger equation less challenging by simplifying it, with the last attempt containing the Hohenberg-Kohn's theorems where the theory states that the total ground-state energy can, in principle, be determined exactly once we have found the ground-state density.

In 1965, Kohn and Sham [70] reformulated the Hohenberg-Kohn theorems by generating the exact ground-state density $n_0(r)$ using a Hartree-like total wavefunction

$$\Psi(\mathbf{r}_1, \mathbf{r}_2, .., \mathbf{r}_{N_e}) = \psi_1^{KS}(\mathbf{r}_2)\psi_2^{KS}(\mathbf{r}_2)...\psi_{N_e}^{KS}(\mathbf{r}_{N_e}), \tag{3.17}$$

where $\psi_j^{KS}(r_j)$ are some auxiliary independent single-particle wavefunctions. However, the Kohn-Sham wavefunctions cannot be the correct single-particle wavefunctions since our ansatz implies an exact density

$$n(\mathbf{r}) = \sum_{j=1}^{N_e} |\psi_j^{KS}(\mathbf{r})|^2. \tag{3.18}$$

Recalling that equation 3.16 describes the total energy as a functional of the density,

$$E[n] = T[n] + U_{ee}[n] + U_{en}[n], \tag{3.19}$$

we try to modify it to include the kinetic energy $T_s[n]$ and the interaction energy $U_s[n]$ of the auxiliary wavefunction, and the denotation s for single-particle wavefunctions.

$$\begin{aligned} E[n] &= T[n] + U_{ee}[n] + U_{en}[n] + \big(T_s[n] - T_s[n]\big) + \big(U_s[n] - U_s[n]\big) \\ &= T_s[n] + U_s[n] + U_{en}[n] + \underbrace{\big(T[n] - T_s[n]\big) + \big(U_{ee}[n] - U_s[n]\big)}_{E_{xc}[n]} \end{aligned}$$

Here we have our first encounter with the *exchange-correlation energy*

$$E_{xc}[n] = \Delta T + \Delta U = \big(T[n] - T_s[n]\big) + \big(U_{ee}[n] - U_s[n]\big)\,, \tag{3.20}$$

which contains the complex many-electron interaction. For non-interacting system, $E_{xc}[n]$ is conveniently zero, but in interacting systems it most likely is a complex expression. However, one can consider it as our mission to find good approximations to this term, as the better approximations, the closer we get to the exact expression.

The exact total energy functional can now be expressed as

$$\begin{aligned} E[n] = &\overbrace{\sum_j \int \psi_j^{KS*} \frac{-\hbar^2\nabla^2}{2m} \psi_j^{KS} d\mathbf{r}}^{T_s[n]} + \overbrace{\frac{1}{2}\int\int q^2 \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}d\mathbf{r}'}^{U_s[n]} \\ &+ \underbrace{\int V_{en}(\mathbf{r})n(\mathbf{r})d\mathbf{r}}_{U_{en}[n]} + \underbrace{\big(T[n] - T_s[n]\big) + \big(U_{ee}[n] - U_s[n]\big)}_{E_{xc[n]}}. \end{aligned} \tag{3.21}$$

given that the exchange-correlation functional is described correctly. By utilizing the variational principle, we can now formulate a set of Kohn-Sham single-electron equations,

$$\left\{-\frac{\hbar^2}{2m_e}\nabla_s^2 + V_H(\mathbf{r}) + V_{j\alpha}(\mathbf{r}) + V_{xc}(\mathbf{r})\right\}\psi_s^{KS}(\mathbf{r}) = \epsilon_s^{KS}\psi_s^{KS}(\mathbf{r}) \tag{3.22}$$

where $V_{xc}(\mathbf{r}) = \partial E_{xc}[n]/\partial n(\mathbf{r})$ and $V_H(\mathbf{r}) = \int q^2 \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}'$ is the Hartree potential describing the electron-electron interaction. It is worth to notice that $V_H(\mathbf{r})$ allows an electron to interacts with itself, resulting in a self-interaction contribution, however this will be taken care of in V_{xc}.

Finally, we can define the total energy of the system according to Kohn-Sham theory as

$$E[n] = \sum_j \epsilon_j^{KS} - \frac{1}{2} \int\int q^2 \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}d\mathbf{r}' + E_{xc}[n] - \int V_{xc}(\mathbf{r})n(\mathbf{r})d\mathbf{r}. \quad (3.23)$$

If V_{xc} is exact, and $E[n]$ gives the true total energy, we still do not know if the energy eigenvalues ϵ_s^{KS} are the true single-electron eigenvalues. However, there exists one exception, which is that the highest occupied eigenvalue of a finite system has to be exact if the density is exact.

The only task that is left for us now is to find the exact expression for $E_{xc}[n]$ as a functional of the density $n(r)$. With that expression, we would be able to calculate the total energies of any materialUnfortunately, the exchange-correlation potential is unknown for most systems.

It is possible to solve the Kohn-Sham equations by applying a self-consistent field method. This is a computational scheme, and for further details one can consult the appendix A.4.

3.2.3 The exchange-correlation energy

There is one scenario for which we can derive the exact expression of the exchange-correlation functional, namely the *homogeneous electron gas* (HEG). However, this has a natural cause, since by definition $n(\mathbf{r})$ is constant for this situation. Given that it is the variations of electron density that are the foundation of material properties, the usefulness of HEG is limited. The *local density approximation* (LDA) is an approximation based on this approach, where the local density is the only variable used to define the exchange-correlation functional. Specifically, we can set the exchange-correlation potential at each position to be the known exchange-correlation potential from homogeneous electron gas at the electron density observed at that position [70]:

$$V_{xc}(\mathbf{r}) = V_{xc}^{\text{electron gas}} \left[n(\mathbf{r})\right]. \quad (3.24)$$

This is the simplest and most known approximation to the exchange-correlation functional, and accordingly it has a few drawbacks. One of them is the incomplete cancellation of the self-interaction term, which leads to a repulsion that may cause artifical repulsion between electrons, and hence increased electron delocalization [71]. In addition, LDA has proven challenging to use when studying atoms and molecules because of their rapidly varying electron densities, however, the LDA is seen as succesful for bulk materials because of the slowly varying electron density [67]. Considering the relatively low computational cost and relatively high accuracy, the LDA overall makes a good model for estimation of the exchange-correlation functional for bulk-materials.

In the light of the merits of the LDA, an extensive search for new approximations was launched. The *generalized gradient approximation* (GGA) is an extension of the LDA, which includes the gradient of the density

$$V_{\mathrm{xc}}^{\mathrm{GGA}}(\mathbf{r}) = V_{\mathrm{xc}}\left[n(\mathbf{r}), \nabla n(\mathbf{r})\right]. \tag{3.25}$$

The GGA is a good approximation for the cases where the electron density varies slowly, but faces difficulties in many materials with rapidly varying gradients in the density, causing the GGA to fail. Thus, the annotation *generalized* in GGA is set to include the different approaches to deal with this challenge. Two of the most commonly implemented GGA functionals are the non-empirical approaches Perdew-Wang 91 (PW91) [72] and Perdew-Burke-Ernzerhof (PBE) [73].

Both LDA and GGA are commonly known to severely underestimate the band gaps of semiconductor materials, in addition to incorrectly predicting charge localizations originating from narrow bands or associated with local lattice distortions around defects [74]. The latter limitation is thought to be due to self-interaction in the Hartree potential in equation 3.22.

To improve the accuracy of excited state properties estimations, other methods have been developed. Tran and Blaha (TB) [75] adapted the exchange potential suggested by Becke-Roussel (BR) [76] that leads to band gaps close to experimental values while still using a cheap semilocal method [77]. The modified TB-mBJ version introduces a parameter relative to the BJ version,

$$V_{\mathrm{xc}}^{\mathrm{TB-mBJ}}(\mathbf{r}) = c \cdot V_{\mathrm{xc}}^{\mathrm{BR}}(\mathbf{r}) + (3c-2)\frac{1}{\pi}\sqrt{\frac{5}{6}}\sqrt{\frac{t(\mathbf{r})}{\rho(\mathbf{r})}}, \tag{3.26}$$

where ρ is the electron density, t is the kinetic-energy density, $V_{\mathrm{xc}}^{\mathrm{BR}}$ is the Becke-Roussel exchange potential and c is the parameter that changes weights of the BJ potential and is fit to experimental data.

Hybrid functionals intermix exact Hartree-Fock exchange with exchange and correlation from functionals based on the LDA or GGA. Hartree-Fock theory completely ignore correlation effects, but account for self-interaction and treats exchange as exact. Since LDA/GGA and Hartree-Fock supplement each other, they can be used as a combination for hybrid-functionals resulting in some cancellation of the self-interaction error. Becke [78] introduced a 50% Hartree-Fock exact exchange and 50% LDA energy functional, while Perdew *et al.* [79] altered it to $25\% - 75\%$ and favoring PBE-GGA instead of LDA.

The inclusion of Hartree Fock exchange improves the description of localized states, but requires significantly more computational power for large systems. Another method called the *GW* approximation includes screening of the exchange interaction [80], but has a computational price that does not neccessarily defend its use. Thus, the real challenge is to reduce the computational effort while still producing satisfactory results. Heyd *et al.* [81] suggested to separate the non-local Hartree-Fock exchange into a short- and long-range portion, incorporating the exact exchange in the short-range contribution. The separation is controlled by an adjustable parameter ω, which was empirically optimised for molecules to $\omega = 0.15$ and solids to $\omega = 0.11$ and are known as the HSE03 and HSE06 (Heyd-Scuseria-Ernzerhof), respectively [82]. The functionals are expressed as

$$E_{xc}^{HSE} = aE_{x}^{HF,SR}(\omega) + (1-a)E_{x}^{PBE,SR}(\omega) + E_{x}^{PBE,LR}(\omega) + E_{c}^{PBE} \tag{3.27}$$

where $a = 1/4$ is the Hartree-Fock mixing constant and the abbreviations SR and LR stands for short range and long range, respectively.

Hence, hybrid-functionals are *semi-empirical* functionals that rely on experimental data for accurate results. They give accurate results for several properties, such as energetics, bandgaps and lattice parameters, and can fine-tune parameters fitted to experimental data for even higher accuracy.

Furthermore, the computational effort required for the hybrid-functionals are significantly larger than for non-empirical functionals such as LDA or GGA. Krukau *et al.* [82] reported a substantial increase in computational cost when reducing the parameter ω from 0.20 to 0.11 for 25 solids, and going lower than 0.11 demanded too much to actually defend its use.

Unfortunately, an area where both GGA and hybrid functionals are reportedly inadequate is in calculating the dispersion interactions [83]. Many implementations have been developed to deal with this vulnerability, with one of them being the non-local van der Waals density functional (vdW-DF) [84]

$$E_{xc} = E_{x}^{GGA} + E_{c}^{LDA} + E_{c}^{nl}, \tag{3.28}$$

where E_{x}^{GGA} is GGA exchange energy [83].

3.2.4 Limitations of the DFT

If we had known the exact exchange-correlation functional, the density functional theory would yield the exact total energy. Alas, that is not the case and we are bound to use approximations in forms of functionals. What is common for all approximations is that they are specifically designed to solve one given optimization, therefore it is not neccessarily one functional that is consider superior in all fields of interest. One could consider that the hybrid functionals due to a high accuracy overall should be dominant, but that is only if one have the computational capacity required. The accuracy of calculations is dependent on which functional being used, and normally a higher accuracy means the use of a more complex and computationally demanding functional.

Nonetheless, density functional theory is considered a very successful approach and Walter Kohn was awarded the Nobel Price in chemistry in 1998 for his development of the density-functional theory [85]. It has matured into the undisputed choice of method for electronic structure calculations [64]. It is especially regarded as successful in contexts where DFT can make important contributions to scientific questions where it would be essentially impossible to determine through experiments [67].

3.3 High-throughput information storage

Fuelled by the widespread application of DFT in material science, we observe an increase of computational capacity due to advances in simulation methods, computational science and technologies [64]. The time required for calculations has been reduced substantially, and enables more time on simulation setup and analysis. Ultimately, this has led to a new type of workflow called high-throughput [86], where one can automate input creation and perform several (up to millions) simulations in parallel instead of performing many manually-prepared simulations. The HT engines are required to be fast and accurate, otherwise the purpose of its existence is lost. We will in this thesis mainly focus on high throughput in the context of first principles DFT calculations.

Normally, the implementation of HT-DFT methods is done in three steps. In the first step we perform thermodynamic or electronic structure calculations for a large number of materials. For the second step, we store the information gathered in a systematic database, normally known as a material repository. For the third and final step, we characterize and select novel materials or analyse and extract new physical insight [64]. In general, the two first steps are performed on high-performance computers (HPC), and are defining for the third step due to the vast amount of interesting properties of materials. Importantly, one needs to consider the ultimate goal in step three initially before deciding upon materials and properties to calculate.

One of the largest government funding projects is the 2011 launched Material Genome Initiative (MGI) [87] which seeks to accelerate the discovery, design, development and deployment of novel materials through the creation of a materials innovation infrastructure. As a result, we have seen an intensive implementation of material repositories that fascilitate sharing and distribution of step 1 and 2, with the ultimate goal of storing calculated properties of all feasible structures of materials [88]. Examples include AFLOW [89–91], Materials Project [92], OQMD [93, 94] and JARVIS-DFT [95]. On the other hand, we find a less diverse selection of experimental databases, with perhaps the most known being the Inorganic Crystal Structure Database (ICSD) [96].

In this section, we describe a few of the most widely known high-throughput databases, codes and tools, with an emphasis of what the particular speciality of each database is.

3.3.1 Materials project

Materials project [92] is an open source project that offers a variety of properties of over one hundred thousand of inorganic crystalline materials. It is known as the initiator of materials genomics and has as its mission to accelerate the discovery of new technological materials, with an emphasis on batteries and electrodes, through advanced scientific computic and innovative design.

Every compound has an initial relaxation of cell and lattice parameters performed using a 1000k-point mesh to ensure that all properties calculated are representative of the idealized unit cell for each respective crystal structure. The functional GGA is used to calculate band structures, while for transition metals it is applied +U correction to correct for correlation effects in d- and f-orbital systems that are not addressed by GGA calculations [97]. The thermodynamic stability for each phase with respect to decomposition, is also calculated. This is denoted as E Above Hull, with a value of zero is defined as the most stable phase at a given composition, while larger positive values indicate increased instability.

Each material contains multiple computations for different purposes, resulting in different 'tasks'. The reason behind this is that each computation has a purpose, such as to calculate the band structure or energy. Therefore, it is possible to receive several tasks for one material which results in more features per material.

3.3.2 AFLOW

The AFLOW[89–91] repository is an automatic software framework for the calculations of a wide range of inorganic material properties. They utilise the GGA-PBE functional within VASP with projector-augmented wavefunction (PAW) potentials to relax twice and optimize the ICSD-sourced structur. They are using a $3000 - 6000$ k-point mesh, indicating a more computationally expensive calculation compared to the Materials Project. Next, the band structure is calculated with an even higher k-point density, in addition to the +U correction term for most occupied d- and f-orbital systems, resulting in a standard band gap [98]. Furthermore, they apply a standard fit gathered from a study of DFT-computed versus experimentally measured band gap widths to the initial calculated value, obtaining a fitted band gap [99].

3.3.3 Open Quantum Materials Database

The Open Quantum Materials Database (OQDM) [93, 94] is a free and available database of DFT-calculations. It has included thermodynamic and structural properties of more than 600.000 materials, including all unique entries in the Inorganic Crystal Structure Database (ICSD) consisting of less than 34 atoms [100].

The DFT calculations are performed with the VASP software whereas the electron exchange and correlation are described with the GGA-PBE, while using the PAW potentials. They relax a structure using $4000-8000$ k-point mesh, indicating an even increasing computational expensive calculation than AFLOW again. Several element-specific settings are included such as using the $+U$ extension for various transition metals, lanthanides and actinides. In addition, any calculation containing 3d or actinide elements are spin-polarized with a ferromagnetic alignment of spins to capture possible magnetism. However, the authors note that this approach does not capture complex magnetic, such as antiferromagnetism, which has been found to result in substantial errors for the formation energy [101].

3.3.4 JARVIS

Joint Automated Repository for Various Integrated Simulations (JARVIS) [95] - DFT is an open database based based on the VASP software to perform a variety of material property calculations. It consists of roughly 40.000 3D and 1.000 2D materials using the vdW-DF-OptB88 van der Waals functional, which was originally designed to improve the approximation of properties of two-dimensional van der Waals materials, but has also shown to be effective for bulk materials [102, 103]. The functional has shown accurate predictions for lattice-parameters and energetics for both vDW and non-vdW bonded materials [104].

Structures included in the data set are originally taken from the materials project, and then re-optimized using the OPT-functional. Finally, the combination of the OPT and modified Becke-Johnson (mBJ) functionals are used to obtain a representative band gap of each structure, since both have shown unprecedented accuracy in the calculation of band gap compared to any other DFT-based calculation methods [105].

The JARVIS-DFT database is part of a bigger platform that includes JARVIS-FF, which is the evaluation of classical forcefield with respect to DFT-data, and JARVIS-ML, which consists of 25 machine learning to predict properties of materials. In addition, JARVIS-DFT also includes a data set of 1D-nanowire and 0D-molecular materials, yet not publically distributed.

3.4 Materials informatics

Despite a computationally demanding step 1 and a sophisticated database architecture defined in step 2, the third step remains more important in discovery of novel materials. In the third step, we apply constraints to the database in order to filter or select the best candidates according to the desired attributes [64], and is referred to as data *mining* and data *screening*. To achieve the most insight, we normally apply the constraints in a sequential manner to identify any potential underlying trend. The materials that satisfy the first constraint moves on to the next round, while the rest are eliminated from further consideration. The constraints can be applied either by the understanding of properties, human intuition or through a close interaction with a machine learning (ML) algorithm, where the latter will be extensively studied in the next chapter.

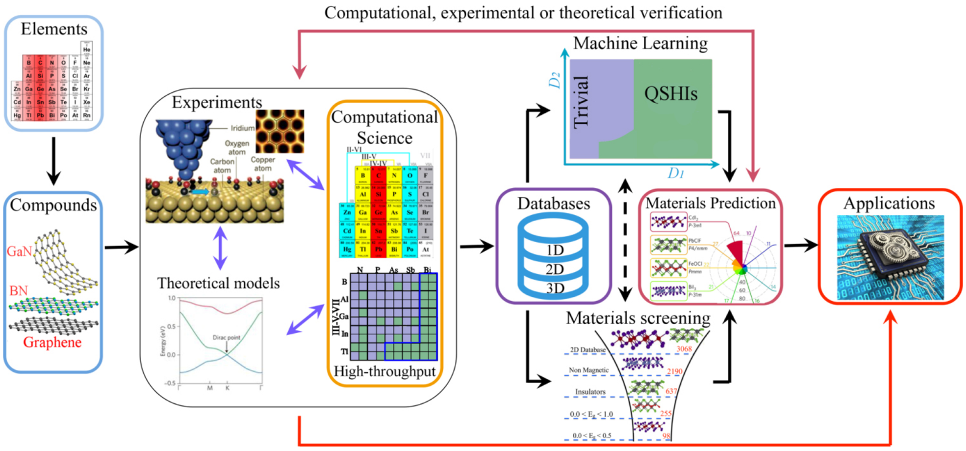


Figure 3.2: Schematic representation of the workflow of novel materials discovery. Figure taken from Ref. [64], which was originally adapted from Refs. [106–108].

Together, the three steps resembles figure 3.2. From building compounds based on elements, calculating theoretical, computational and experimental properties, storing the information in databases and applying material screening and machine learning, to finally receiving a material prediction. If the material prediction is verified iteratively by many independent sources, the time to market for new technologies based on a new material takes approximately 20 years [109].

Importantly, the data driven paradigm enables a new approach for novel material discovery. The traditional approach, namely the *direct approach*, rely on the calculation of properties given the structure and composition of a material, such that the search for eligible candidates exhibiting the target property is performed tediously case by case. In other words, what is the property of a given material. However, the *inverse approach* is of integral importance in this work: given a desired property, what material can present it [64]?

The application of machine learning and data-driven techniques to material science have developed into a new field named materials informatics [110]. Alex Szalay, director of the US National Virtual Observatory project, described informatics for astronomy in 2003 as the following.

> "Science was originally empirical, like Leonardo making wonderful drawings of nature. Next came the theorists who tried to write down equations that explained observed behaviors, like Kepler or Einstein. Then, when we got to complex enough systems like the clustering of a million galaxies, there came the computer simulations – the computational branch of science. Now, we are getting into the data exploration part of science, which is kind of a little bit of them all" Alex Szalay [111]

The formulation is true also for material informatics, where the scope is to discover relations between known standard features and materials properties through a combination of *a bit of everything*.

3.4.1 Materials informatics software packages

In practice, several software packages exists for the purpose of generating, describing, visualizing, calculating or predicting properties of materials.

The Atomic Simulation Environment (ASE) is an environment in the Python programming language that includes several tools and modules for setting up, modifying and analyze atomistic simulations [112]. It is in particular used together with the Computational Materials Repository (CMR) [113].

Another commonly used module is the Python Materials Genomics (pymatgen) [114]. This is a well-documented open module with both introductory and advanced use case examples written in Jupyter Notebook for easy reproducibility, and is integrated with the Materials Project RESTful API.

An increasingly popular library is matminer [115], which is an open-source toolkit for material analysis written in Python. Matminer is powered by a group known as *Hacking Materials Research Group*[2]. Matminer provides modules to extract data information from a wide variety of databases.

[2] Project's Github site: https://github.com/hackingmaterials.

Additionally, they provide the tools to construct possibly thousands of features from calculations based on a materials composition, structure and DFT-calculations, and have modules for visualization and automatic machine learning.

AFLOW-ML [116] is an API that uses machine learning to predict thermomechanical and electronic properties based on the chemical composition and atomic structure alone, which they denote as *fragment descriptors*. They start with applying a classification model to predict if a compound is either a metal or an insulator, where the latter is confirmed with an additional regression model to predict the band gap width. To be able to predict properties on an independent data set, they utilise a fivefold cross validation process for each model. They report a 93% prediction success rate of their initial binary classification model, whereas the majority of the wrongful predictions are narrow-gap semiconductors. The authors does not compare their predicted band gap to experimental values, but it is found that 93% of the machine-learning-derived values are within 25% of the DFT +U-calculated band gap width [100].

Chapter 4

Machine learning

The enourmous amount of data generated in the digital world today is beyond comprehension. In 2019, more than 500 hours of video was uploaded to Youtube every second, totalling to over 82 years of content every day[1]. In addition, more than 1.5 billion web sites exists[2].

However, an increasing amount of data comes hand in hand with an increasing demand of knowledge about the data. If we are unable to extract information from the data, the data serves no intention and exists as an excess. Therefore, we need methods to process and automate data analysis, which is what the promises of *machine learning* covers. Machine learning can reveal patterns in data with ease where a human would face difficulties, and use this information to predict or generate new data. Many tools in machine learning is based on probability theory, which can be applied to problems involving uncertainty. Thus, machine learning is also commonly named as *statistical learning* [117].

There are mainly two types of machine learning, either *supervised* or *unsupervised* learning. In unsupervised learning we are given inputs $\mathcal{D} = \{x_i\}_{i=1}^N$, where x_i is a training input that has D-dimensions that describes each entry, where each dimension is known as a *feature* or a *descriptor*. The features could be examplified as height or weight, or it could be something complex that has no practical meaning (at least not to humans). Since there are no features describing what an entry is, it is up to the tools of machine learning to find patterns in the data, and is the essence of unspervised learning. In the supervised approach, on the other hand, the model tries to learn a mapping from inputs x to outputs y, given a labeled set of pairs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$. The set $\mathcal{D}$ is known as the training set, and N is the number of entries. The flexibility of the shape of a feature is also shared with the output. It can in principle

[1] Source: https://www.youtube.com/intl/no/about/press extracted 15.02.2021

[2] Source: https://www.statista.com/chart/19058/how-many-websites-are-there extracted 29.03.2021

be anything, but it is mostly assumed that the output is either *categorical* or *nominal* restricted by a finite set $y_i \in \{1, ..., \mathcal{C}\}$. The problem is defined as *classification* if the output is categorical, or *regression* if the output is real-valued [117].

4.1 Supervised learning

Supervised learning applied to classification has as goal to learn the target output $y \in \{1, .., \mathcal{C}\}$ from the inputs $\mathbf{x}$. The number of classes is $\mathcal{C}$, and depicts if the classification is *binary* ($\mathcal{C} = 2$), *multiclass* ($\mathcal{C} > 2$), or *multi-label* if the class labels are not mutually exclusive (examplified with the weather can be both sunny and cold at the same time). Normally, classification is used when the problem is formulated as a multiclass classification, and hereon we will adapt to the formulation as well [117].

In order to be able to learn from data, we will need to formulate a function approximate. Assume $y = f(x) + \epsilon$ for some unknown function f and a random error term ϵ with mean zero. We can then try to approximate f from a labeled training set, which we can use to make the predictions $\hat{y} = \hat{f}(\mathbf{x})$. With the estimated $\hat{f}$ we can make predictions on unlabeled data and achieve a *generalized model*. The estimated function $\hat{f}$ is often considered as a black box, since we are not neccessarily interested in the exact shape of the function but rather the predictions.

As simple as the idea behind supervised classification appears, a generalized model remains deeply dependent on the available data. Imagine a training set containing two entries. The first entry is a young and tall person labeled healthy. The other entry is an old and short person labeled sick. The pattern in this simple scenario is abundantly clear, but will face a challenge if it were to predict on a test set containing a person who is young and short. Therefore, it is desirable to compute the probability of an entry belonging to one class. The probability distribution is given by $p(y|\mathbf{x}, \mathcal{D})$, where the probability is conditional on the input vector (test set) x and the training set $\mathcal{D}$. If the output is probabilistic, we can compute the estimation to the true label as

$$\hat{y} = \hat{f}(\mathbf{x}) = \operatorname{argmax} f(x) p(y = 1|\mathbf{x}, \mathcal{D}), \tag{4.1}$$

which represents the most probable class label and is known as the *maximum a posteriori* estimate [117].

4.2 Evaluating accuracy of a model

It would be desirable to find one superior model that we could utilize on all types and sizes of datasets. Unfortunately, there is no algorithm that has this property, since one model might be recognized as best on one particular dataset, while others are far better on other datasets. This is known as the *no free lunch theorem* (Wolpert 1996 [118]). The same goes with evaluating the model - there is no metrics that stand alone as the best metric to evaluate a model. Choosing how to actually evaluate a model can be the most challenging part of a statistical learning procedure.

4.2.1 Bias-variance tradeoff

To illustrate a challenge in choosing the correct parameters, we give an example using the mean squared error (MSE) as a *cost function*, which we want to minimize in order to improve the accuracy of the model [117]. Assume that our data can be represented by

$$\mathbf{y} = f(\mathbf{x}) + \boldsymbol{\epsilon},$$

where $f(\mathbf{x})$ is an unknown function and $\boldsymbol{\epsilon}$ is normally distributed with a mean equal to zero and variance equal to σ^2. Furthermore, we also assume that the function $f(\mathbf{x})$ can be approximated to a model $\hat{\mathbf{y}}$, where the model is defined by a design matrix $\mathbf{X}$ and parameters $\boldsymbol{\beta}$,

$$\hat{\mathbf{y}} = \mathbf{X}\boldsymbol{\beta}.$$

The parameters $\boldsymbol{\beta}$ are in turn found by optimizing the mean squared error (MSE) via the cost function

$$C(\mathbf{X}, \boldsymbol{\beta}) = \frac{1}{n}\sum_{i=0}^{n-1}(y_i - \hat{y}_i)^2 = E\left[(\mathbf{y} - \hat{\mathbf{y}})^2\right].$$

The cost function can be rewritten as

$$\begin{aligned} E\left[(\mathbf{y} - \hat{\mathbf{y}})^2\right] &= \frac{1}{n}\sum_i \left(f_i - E\left[\hat{\mathbf{y}}\right]\right)^2 + \frac{1}{n}\sum_i \left(\hat{y}_i - E\left[\hat{\mathbf{y}}\right]\right)^2 + \sigma^2 \\ &= E\left[\left(\mathbf{f} - E\left[\hat{\mathbf{y}}\right]\right)^2\right] + Var\left(\hat{\mathbf{y}}\right) + \sigma_\epsilon^2 \end{aligned}$$

where $E[\mathbf{y}] = \mathbf{f}$, $E\left[\boldsymbol{\epsilon}\right] = \mathbf{0}$ and $Var\left(\mathbf{y}\right) = Var\left(\boldsymbol{\epsilon}\right) = \sigma_\epsilon^2$.

The first term on the right hand side is the squared bias, the amount by which the average of our estimate differs from the true mean, while the second term represents the variance of the chosen model. The last term is the variance of the error $\boldsymbol{\epsilon}$, also known as the irreducible error. In general, an estimated function $\hat{f}$ will never be a perfect estimate for f since we can not reduce the error introduced by $\boldsymbol{\epsilon}$. Therefore, any model will always be restricted to an upper bound of accuracy due to the irreducible error.

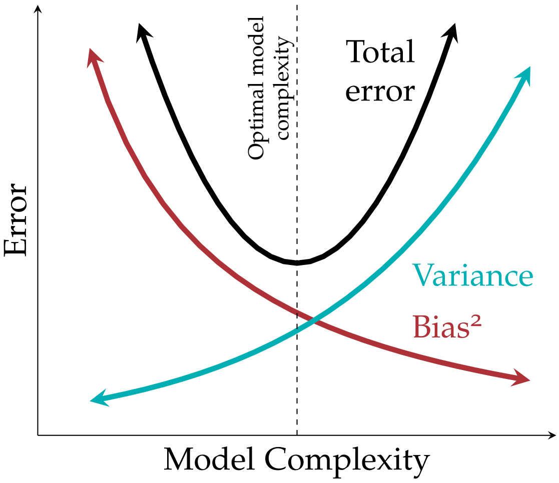


Figure 4.1: A schematic representation of the bias-variance tradeoff as a function of model complexity, adapted from Ref. [119]. The error associated with a model can be decomposed into variance and bias, where a compromise between the lowest bias and variance corresponds to the optimal model complexity.

A model with high variance will typically experience larger fluctuations around the true value, while a model with high bias corresponds to a larger error in the average of estimates. This is schematically visualized as function of model complexity in figure 4.1. If the model is not complex enough due to high bias and low variance, the algorithm can end up not learning the relevant relations between features and output. This is known as *underfitting* [117]. On the other hand, a complex model with low bias and high variance might find trends in random noise from the training data instead of the relevant features, resulting in *overfitting* [117]. An ideal model would be one that simultaneously achieves low variance and low bias Therefore, we have to do a trade-off between how much bias and variance we would like in the model.

4.2.2 Accuracy, precision and recall

Given a model that has dealt with the intricacy of increasing complexity, we would like to evaluate the model's output quality. For a binary supervised classification problem we can measure the accuracy by finding how many correct predictions have been made. Prediction accuracy can provide a fine initial analysis, but it has some significant drawbacks seen in unbalanced datasets. This can be easily explained with a dataset consisting of $99:1$ ratio of class, since just guessing the majority class will result in a very high 99% accuracy. Perhaps it is the 1% that is the most important class, thus the accuracy score severely lacks information for the model.

Therefore, we turn to other evaluation metrics such as a *a confusion matrix*. A confusion matrix is a method for measuring the performance of classifiers [117]. It is set up as a table with 4 different categories, where two of the categories are the predicted outcomes of the classifier and the two final categories are the true outcomes. An example of a confusion matrix for a binary classifier is shown in table 4.1.

For the binary confusion matrix there are two possible predicted out-

Table 4.1: A confusion matrix for a binary classifier. The entries true positive and true negative on the diagonal of the matrix are correctly predictions, while false positive and false negative are wrongly made predictions. P and N are the total number positive and negative predictions, respectively. Similarly, P' and N' are the number of true -positive or negative labels, respectively.

		Predicted label		
		1	0	**total**
Actual label	1	True Positive	False Negative	P'
	0	False Positive	True Negative	N'
	total	P	N	

comes, either positive or negative. This gives rise to some terminology.

- True Positive (TP): The classifier correctly predicts a positive event.
- True Negative (TN): The classifier correctly predicts a negative event.
- False Positive (FP): The classifier incorrectly predicts a positive event when the true event was negative.
- False Negative (FN): The classifier incorrectly predicts a negative event when the true event was positive.

From the confusion matrix one can then start estimating the performance of the model, by calculating different factors, such as [117]

- **Sensitivity**, also known as the true negative rate, is the ratio of the number of correct negative examples to the number classified as negative. It is defined as

$$\text{Sensitivity} = \frac{\mathsf{TN}}{\mathsf{TN} + \mathsf{FP}}. \tag{4.2}$$

- **Recall**, also known as the true positive rate, is the ratio of the number of correct positive examples to the number classified as positive. A high recall relates to a low false negative rate, and is defined as

$$\text{Recall} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}}. \tag{4.3}$$

- **Precision** is the ratio of correct positive examples to the number of actual positive examples. A high precision relates to a low false positive rate, and is defined as

$$\text{Precision} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}}. \tag{4.4}$$

Similar to the bias-variance tradeoff, it is common to compare the recall with the precision to identify the tradeoff for different thresholds. High scores for both reveals that a classifier returns accurate results combined with returning a majority of all positive results.

Sometimes a classifier can have drastically different values for the precision and recall. This leads to another estimator for the performance of a classifier, which is known as the F1-score. The F1-score is defined as the harmonic mean of precision and recall,

$$\text{F1-score} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}},$$

and can be used to find a good tradeoff between recall and precision. The highest value of F1-score is 1 and is considered an ideal classifier, while the lowest is 0.

However, the f1 score is insensitive to the number of negative predictions. Therefore, an adjustment of the normal accuracy is in place. The name of this metric is called the balanced accuracy, which equally weights how many true positive and true negative,

$$\text{Balanced accuracy} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{TN} + \text{FP}}\right),$$

which makes it particular handy for imbalanced datasets.

We have now only scratched the surface of potential evaluation metrics, and as a final note we would like to emphasis that it is up to the implementer which evaluation metric one should use.

4.2.3 Cross validation

When evaluating different parameters for models, commonly done in a grid-search scheme, there is an abundant risk of performing an overfit to the test set since we can tweak the parameters to a model so it can perform optimally. To solve this problem, we can exclude a part of the dataset as a validation set (in addition to a test set). Therefore, we can train a model on the training set, and evaluate the parameters on the validation set. After a lot of trial and error and the experiment seems successful, we can do one final evaluation on the test set.

Unfortunately, this reduces the number of samples that can be used for training drastically. A fix for this is to apply *cross validation* (CV) [117]. Cross validation is a technique used to evaluate predictive models by partitioning the original sample into a training set to train the model, and a test set to evaluate it.

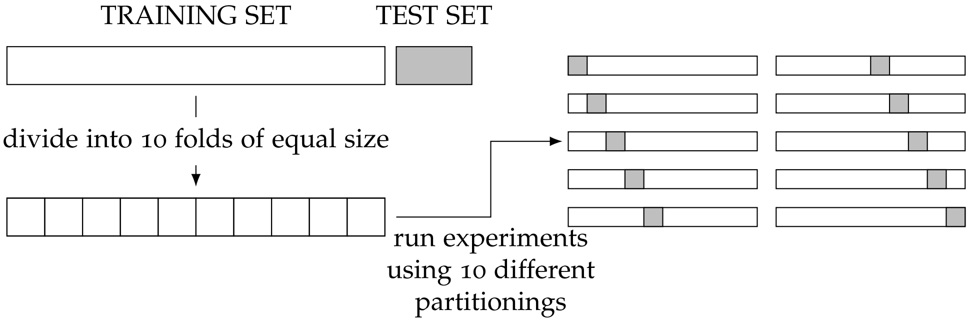

TRAINING SET
TEST SET
divide into 10 folds of equal size
run experiments using 10 different partitionings

Figure 4.2: A schematic representation of a 10-fold cross-validation scheme.

It is common to apply cross validation into fold, yielding the name of k-fold cross validation. In k-fold cross validation, the training set is partioned into k equal sized subsamples, as visualized in figure 4.2. Of the k samples, a single sample is used as validation set while the remaining k-1 samples are used as training data. The process is then repeated k-times, such that each of the k subsamples are used as a validation set exactly once. Therefore, all observations are used for both training and validation, and each observation is used for validation exactly once. The k results from the folds can then be averaged to produce an estimate. The subsamples are allowed to have an imbalanced dataset, so that each class is not neccessarily represented equally in each fold. Since supervised algorithms tend to weigh each instance equally, this may result in overrepresented classes being favored during training of the model. Even worse could be the result of a fold where one class is not represented at all, resulting in a model that do not learn how to predict a class at all.

To deal with the vulnerability of imbalanced datasets in CV, one can employ a stratified k-fold cross validation technique. Stratification is a process that seeks to ensure that each fold is representative of all classes (also named *strata* in this context) in the data, making each fold having approximately equal class-representation.

4.3 Logistic regression

Logistic regression, or *logit*, is considered a *soft* classification algorithm, which means that an output of the algorithm is considered to be categorical instead of numerical.

Assume we have a dataset with $\mathbf{x}_i = x_{i1}, x_{i2}, .., x_{ip}$ input data, where we have p predictors for each corresponding output data y_i. The outcomes y_i are discrete and can only take certain values or classes. In our case we have two classes with y_i either being equal to 0 or 1. Therefore, the probability that a datapoint belongs to either class can be given by the Sigmoid function,

$$p(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}.$$

Furthermore, we have the parameters $\boldsymbol{\beta} = \beta_1, \beta_2, ..., \beta_p$ of our fitting of the Sigmoid function, where the probabilities are defined as

$$p(y_i|\mathbf{x}_i\boldsymbol{\beta}) = \frac{e(\mathbf{x}_i\boldsymbol{\beta})}{1+e^{(\mathbf{x}_i\boldsymbol{\beta})}}.$$

The goal of logistic regression is then to correctly predict the category of a given dataset, which has different outcomes, by using an optimal parameter $\boldsymbol{\beta}$ that maximises the probability of seeing the observed data. How we find the parameters $\boldsymbol{\beta} = \beta_1, \beta_2, ..., \beta_p$ of the model, is to use the principle of *maximum likelihood estimation* (MLE),

$$P(\boldsymbol{\beta}) = \prod_{i=1}^{n} \left[p(y_i = 1|\mathbf{x}_i\boldsymbol{\beta})\right]^{y_i} \left[1 - p(y_i = 1|\mathbf{x}_i\boldsymbol{\beta})\right]^{1-y_i},$$

where we obtain the log-likelihood function, which is easier to work with, since the log-likelihood turns the exponentials into summations,

$$C(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i(\mathbf{x}_i\boldsymbol{\beta}) - \log\left(1 + \exp\{(\mathbf{x}_i\boldsymbol{\beta})\}\right)\right).$$

Finally, choose our cost function for the *cross entropy*, which is defined as the negative log-likelihood,

$$C(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \Big(y_i(\boldsymbol{x}_i\boldsymbol{\beta}) - \log\left(1 + \exp\big\{(\boldsymbol{x}_i\boldsymbol{\beta})\big\}\right) \Big)$$

To maximize the accuracy and precision of the logistic regression model, we need to find the optimal parameters $\boldsymbol{\beta}$ by minimising the cross entropy.

4.3.1 Stochastic gradient descent

One common numerical method for finding the minimum of a function is *stochastic gradient descent* (SGD). The fundamental idea of SGD comes from the observation that the cost function can be written as a sum over n data points $\{\boldsymbol{x}_i\}_{i=1}^{n}$,

$$C(\boldsymbol{\beta}) = \sum_{i=1}^{n} c_i(\boldsymbol{x}_i, \beta).$$

We can compute the gradient as

$$\nabla_\beta C(\boldsymbol{\beta}) = \sum_{i=1}^{n} \nabla_\beta c_i(\boldsymbol{x}_i, \beta)$$

Then, it is possible to introduce a randomness by only taking the gradient on a small interval of the data, called a minibatch. With n total data points, and M datapoints per minibatch, the number of minibatches is then $\frac{n}{M}$.

The idea is now to approximate the gradient by replacing the sum over all data points with a sum over the data points in one of the minibatches picked at random in each gradient descent step,

$$\nabla_\beta C(\boldsymbol{\beta}) = \sum_{i=1}^{n} \nabla_\beta c_i(\boldsymbol{x}_i, \beta) \rightarrow \sum_{i \in B_k}^{n} \nabla_\beta c_i(\boldsymbol{x}_i, \beta),$$

where B_k is the set of all minibatches, with $k = 1, .., \frac{n}{M}$. One step of gradient descent is then defined by

$$\beta_{j+1} = \beta_j - \gamma_j \sum_{i \in B_k}^{n} \nabla_\beta c_i(\boldsymbol{x}_i, \beta)$$

where k is picked at random with equal probability from $[1, \frac{n}{M}]$ and γ_j is the step length. An iteration over the number of minibathces $(\frac{n}{M})$ is commonly referred to as an epoch. Thus, it is typical to choose a number of epochs and for each epoch iterate over the number of minibatches.

4.4 Decision trees

Classification and regression trees (CART), also called decision trees, are one of the more basic supervised algorithms. They can be used for both regression and classification tasks, but we will for the relevancy of this work provide a special emphasis to classification trees.

The idea behind decision trees is to find the features that contain the most information regarding the target, and then split up the dataset along the values of these features. This feature selection enables the target values for the resulting underlying dataset to be as *pure* as possible, which means the dataset only contains one class [117]. The features that can reproduce the best target features are normally said to be the most informative features.

A decision tree can be divided into a *root node*, *interior nodes*, and the final *leaf nodes*, commonly known as *terminal nodes*. The nodes are connected by *branches*. The decision tree is able to learn an underlying structure of the training data and can, given some assumptions, make predictions on unseen observations. These predictions are based on the information stored in the leaf nodes in the tree.

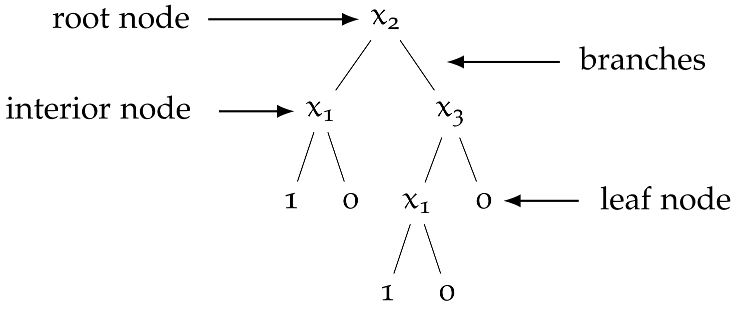


Figure 4.3: A schematic representation of a binary classification tree, which consists of three nodes that contain information of the features x_1, x_2 and x_3.

The process behind a decision tree can be seen as a top-down approach. First, we make a leaf provide the classification of a given instance. Then, a node specifies a test of some attribute of the instance, while a branch corresponds to a possible value of an attribute. Subsequently, the instance move down the tree branch corresponding to the value of the attribute. Then the steps can be repeated for a new subtree rooted at the new node.

A classification tree differs from a regression tree by the response of the prediction, since it produces a qualitative response rather than a quantitative one. The response is given by the most commonly occuring class of training observations specified by the attribute of the node. A schematic representation of a classification tree is visualised in figure 4.3.

4.4.1 Growing a classification tree

In growing a classification tree, a process called recursive binary splitting is applied. This involves two steps:

1. Split the set of possible values $(x_1, x_2, ..., x_p)$ into J distinct non-overlapping regions $R_1, R_2, ..., R_J$.
2. If an observation falls within the region R_J, we make the prediction given by the most commonly occuring class of training observations in R_J.

The computational aspect of recursively doing this for every possible combination of features does not defend its use, and therefore the common strategy is to use a top-down approach. Binary splitting begins at the top of the tree and consecutively splits the *predictor space*, which is a space that describes all possible combinations of the features in the dataset. This is indicated by two new branches further down the tree. It should be noted that the top-down approach is a greedy approach, since the best split is made at each step of the tree-growing process, instead of trying to pick a split that will lead to a better tree in a future step.

We can define a *probability density function* (PDF) p_{mk} that represents the number of observations k in a region R_m with N_m observations. This likelihood function can be represented in terms of observations of a class in region R_m as

$$p_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k), \tag{4.5}$$

where the *indicator* I function equals zero if we misclassify and one if we classify correctly. Therefore, we can define the splitting of the nodes by the misclassification error

$$m_e = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i \neq k) = 1 - p_{mk}. \tag{4.6}$$

However, other methods exists such as the Gini index

$$g = \sum_{k=1}^{K} p_{mk}(1 - p_{mk}) \tag{4.7}$$

and the information entropy

$$s = -\sum_{k=1}^{K} p_{mk} \log p_{mk}. \tag{4.8}$$

The two latter approaches are more sensitive to node purity than the misclassification error, i.e. only containing one class, and are in general preferred [117] for splitting of the nodes in a decision tree.

4.4.2 Classification algorithm

The CART algorithm splits the data set in two subsets using a single feature k and a threshold t_k. The pair of quantities (k, t_k) that constitute the purest subset using the Gini factor G results in the cost function

$$C(k, t_k) = \frac{m_{\mathrm{left}}}{m} G_{\mathrm{left}} + \frac{m_{\mathrm{right}}}{m} G_{\mathrm{right}}, \tag{4.9}$$

where G_{left} (G_{right}) measures the impurity of the left (right) subset and m_{left} (m_{right}) is the number of instances on the left (right) subset. The algorithm tries to minimize the cost function to find the pair (k, t_k) by splitting the training set in two, and then following the same logic for the next subsets. It will continue to do this recursively until it reaches the maximum depth hyperparameter, or if the next split does not reduce impurity.

4.4.3 Pruning a tree

A decision tree has the ability to turn into a very complex model, making it prone to overfitting. Pre-puning is a method that stops the growing of a tree if the decrease in error is not sufficient to justify an increasingly complex model by adding an extra subtree. However, this method should not be implemented for models with a large number of features, since features with small predictive powers might be extensively removed which might result in a tree without any splits at all [117]. Post-pruning, or just pruning, is the standard method which involves growing the tree to full size, and then prune the tree by cutting branches. To determine how much to prune it, we can use a cross-validated scheme to evaluate the amount of terminal nodes that has the lowest error.

4.4.4 Pros and cons of decision trees

Decision trees have several clear advantages compared to other algorithms. They are easy to understand and can be visualised effortlessly for small trees. The algorithm is completely invariant to scaling of the data since each feature is processed separately. Additionally, decision trees can handle both continuous and categorical data and can model interactions between different descriptive features.

As auspicious the advantages of decision trees seems, they are inevitably prone to overfitting and hence does not generalize well. Even with pre-pruning, post-pruning and setting a maximum depth of terminal nodes, the algorithm is still prone to overfit [119]. Another important issue concerns training on unbalanced datasets where one class occurs more frequently than other classes, since this will lead to biased trees because the algorithm will favor the more occuring class. Furthermore, small changes in the data may lead to a completely different tree. Many of these issues can be adressed by using ensemble methods such as either bagging, random forest, or boosting, and can result in a solid improvement of the predictive perfomance of trees.

4.5 Ensemble methods

By using a single decision tree, we often end up with an overfitted model that possess a high variance. Luckily, we can apply methods that aggregate different machine learning algorithms to reduce variance. If each of the algorithms get slightly different results, as they learn different parts of the data, we can combine the results into something that is better than any one algorithm alone. These approaches fall under the categorory of ensemble methods, and will be elaborated upon in this section.

4.5.1 Bagging

Bootstrap aggregation, or just *bagging*, is an ensemble method that involves averaging many estimates [117]. If we have M trained trees on different subsamples of the data, chosen randomly, we can compute the ensemble

$$f(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} f_b(\mathbf{x}), \tag{4.10}$$

where f_b is the b'th tree. Simply re-running the same algorithm on different subsamples can result in a small variance reduction compared to a single tree due to highly correlated predictors, which showcase the need for better approaches.

Random forests provide an improvement of normal bagged trees by choosing a random sample of m predictors as split candidates from the full set of p predictors. The split is restricted in choosing only one of the m predictors, which are normally chosen as either $m \approx \sqrt{p}$ or $m \approx \log p$. This means that at each split in a tree, the algorithm is restricted to a very small portion of the available predictors.

Algorithm 1: Random forest algorithm.

```
for For $b = 1 : B$ do
    Draw a bootstrap sample from the training data;
    Select a tree $T_b$ to grow based on the bootstrap data;
    while node size smaller than maximum node size do
        Select $m \leqslant p$ variables at random from $p$ predictors;
        Pick the best split point among the $m$ features using CART
          algorithm and create a new node;
        Split the node into daughter nodes;
    end
end
Output the ensemble of trees $\{T_b\}_{b=1}^{B}$ and make predictions
```

By introducing randomness into the model, we arrive at a suprisingly capable model that has a high predictive accuracy [120]. This can be examplified by supposing that there is one strong predictor in a dataset, together with several other fairly strong predictors. Most of the trees will use this strong predictor at the top split, which means that the bagged trees will look quite similar to each other and will have highly correlated predictions.

However, even with higher prediction accuracy, it comes as a compromise since we lose the easy ability of model interpretation. A single tree can be easy to understand, but interpretation of a huge jungle of trees does not neccessarily seem appealing for even an experienced data scientist. Furthermore, a random forest does not substantially reduce the variance as averaging many uncorrelated trees would do, as we will soon find out.

4.5.2 Boosting

Boosting is an ensemble method that fits an additive expansion in a set of elementary basis functions [117]. The basic idea is to combine several weak classifiers, that are only just better than a random guess, in order to create a good classifier. This can be done in an iterative approach were we apply a weak classifier to modify the data. For each iteration, we make sure to weigh the observations that are misclassified with a factor. The method is known as adaptive boosting, since the algorithm is able to adapt during the learning process.

In *forward stagewise additive modeling* we want to find an adaptive model

$$f_M(\mathbf{x}) = \sum_{m=1}^{M} \beta_m G_m(\mathbf{x}; \gamma_m), \tag{4.11}$$

where β_m are expansion parameters that will be determined in a minimization process, and $b(\boldsymbol{x};\gamma_m)$ are functions of the multivariable parameter $\boldsymbol{x}$ that are described by the parameters γ_m. We will in this example consider a binary classification problem with the outcomes $\gamma_i \in \{-1, 1\}$ where $i = 0, 1, 2, ..., n-1$ are the set of observables. The predictions are produced by the classification function $G(\boldsymbol{x})$. The error rate of the training sample is given as

$$\overline{\text{err}} = \frac{1}{n}\sum_{i=0}^{n-1} I(\hat{y}_i \neq G(\boldsymbol{x}_i)). \tag{4.12}$$

After defining a weak classifier, we can apply it iteratively to repeatedly modified versions of the data producing a sequence of different weak classifiers $G_m(\boldsymbol{x})$. The iterative procedure can be defined as

$$f_m(\boldsymbol{x}) = f_{m-1}(\boldsymbol{x}) + \beta_m G_m(\boldsymbol{x}), \tag{4.13}$$

where the function $f_M(\boldsymbol{x})$ will be expressed in terms of

$$G(\boldsymbol{x}) = \text{sign}\sum_{i=1}^{M} \alpha_m G_m(\boldsymbol{x}), \tag{4.14}$$

where α_m is the weight that descibes the contribution from the weak classifier $G_m(\boldsymbol{x})$. The main idea is that we do not go back and adjust earlier parameters, which is why this is called *forward* stagewise additive modeling.

We can demonstrate a binary classification example using the exponential cost function that leads to the *discrete AdaBoost* algorithm [121] at step m,

$$C(\boldsymbol{y}, \boldsymbol{f}) = \sum_{i=0}^{n-1} w_i^m \exp\big(-\hat{y}_i \beta G(\boldsymbol{x}_i)\big), \tag{4.15}$$

where $w_i^m = \exp\big(-\hat{y}_i f_{m-1}(\boldsymbol{x}_i)\big)$ is the weight of the corresponding observable i. We can optimize G for any $\beta > 0$ with

$$G_m(\boldsymbol{x}) = \text{sign}\sum_{i=0}^{n-1} w_i^m I(\hat{y}_i \neq G(\boldsymbol{x}_i)). \tag{4.16}$$

This is the classifier that minimize the weighted error rate in predicting y. Furthermore, we can rewrite the cost function to

$$C = (\exp(\beta) - \exp(-\beta)\sum_{i=0}^{n-1} w_i^m I(\hat{y}_i \neq G(x_i)) + \exp(-\beta)\sum_{i=0}^{n-1} w_i^m. \tag{4.17}$$

Substituting G_m into C and solving for β, we obtain

$$\beta_m = \frac{1}{2} \log \frac{1 - \overline{\mathrm{err}}}{\overline{\mathrm{err}}}, \tag{4.18}$$

with the error redefined as

$$\overline{\mathrm{err}} = \frac{1}{n} \frac{\sum_{i=0}^{n-1} w_i^m I(\hat{y}_i \neq G_m(\mathbf{x}_i))}{\sum_{i=0}^{n-1} w_i^m}. \tag{4.19}$$

Finally, this leads to an update of $f_m(\mathbf{x})$ as defined in equation 4.13 and the weights at the next iteration becomes

$$w_i^{m+1} = w_i^m \exp\big(-\hat{y}_i \beta_m G_m(\mathbf{x}_i)\big). \tag{4.20}$$

With the above definitions, we can define the discrete Adaboost algorithm in Algorithm 2.

Algorithm 2: Discrete Adaboost algorithm.

```
Initialize weights $w_i = 1/n$,    $i = 0, ..., n-1$, such that $\sum_{i=0}^{n-1} w_i = 1$;
for $m = 1 : M$ do
    Fit the classifier $f_m(\mathbf{x}) \in \{-1, 1\}$ using weights $w_i$ on the training
      data;
    Compute the error $\overline{\mathrm{err}} = \frac{1}{n} \frac{\sum_{i=0}^{n-1} w_i^m I(\hat{y}_i \neq G_m(\mathbf{x}_i))}{\sum_{i=0}^{n-1} w_i^m}$ ;
    Define a quantity $\alpha_m = \log\left[(1 - \overline{\mathrm{err}_m})/\overline{\mathrm{err}}_m\right]$ ;
    Set new weights to $w_i \leftarrow w_i \exp\big(\alpha_m I(y_i \neq G(\mathbf{x}_i))\big)$;
end
Compute the new classifier $G(\mathbf{x}) = \sum_{i=0}^{n-1} \alpha_m I(y_i \neq G(\mathbf{x}_i))$;
```

It is possible to apply different cost functions resulting in a variety of boosting algorithms. AdaBoost is an example with the cost function in equation 4.17. But instead of deriving new versions of boosting based on different cost functions, we can find one generic method. This approach is known as *gradient boosting* [122]. Initially, we want to minimize

$$\hat{\mathbf{f}} = \operatorname*{argmin}_{f} L(\mathbf{f}), \tag{4.21}$$

where $\mathbf{f} = \big(f(\mathbf{x}_1), ..., f(\mathbf{x}_N))\big)$ are the parameters of the models, and L is a chosen loss function.

This can be solved stagewise, using an approach named *gradient descent*. At step m, let $\mathbf{g}_m$ be the gradient evaluated at $f(\mathbf{x}_i) = f_{m-1}(\mathbf{x}_i)$:

$$\mathbf{g}_m(\mathbf{x}_i) = \left[\frac{\partial L(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}\right]_{f(\mathbf{x}_i) = f_{m-1}(\mathbf{x}_i)}. \tag{4.22}$$

Then we can update

$$\mathbf{f}_m = \mathbf{f}_{m-1} - \rho_m \mathbf{g}_m, \tag{4.23}$$

where ρ_m is the step length and can be found by approximating the real function

$$h_m(\mathbf{x}) = -\rho \mathbf{g}_m(\mathbf{x}). \tag{4.24}$$

So far, this only optimizes f at a fixed set of points, but we can modify it by fitting a weak classifier to approximate the negative gradient. Additionally, we add a step length parameter $0 < \nu < 1$ to perform partial updates, also known as *shrinking* [117]. The gradient boost algorithm is shown in Algorithm 3.

Algorithm 3: Gradient boost algorithm.

```
Initialize the estimate $f_0(\mathbf{x})$;
for $m = 1 : M$ do
    Compute the negative gradient vector $\mathbf{u}_m = -\partial C(\mathbf{y}, \mathbf{f})/\partial \mathbf{f}(\mathbf{x})$ at
      $\mathbf{f}(\mathbf{x}) = \mathbf{f}_{m-1}$;
    Fit the base learner to the negative gradient $h_m(\mathbf{u}_m, \mathbf{x})$;
    Update the estimate $f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \nu h_m(\mathbf{u}_m, \mathbf{x})$;
end
Output the final estimation $f_M(\mathbf{x}) = \sum_{m=1}^{M} \nu h_m(\mathbf{u}_m, \mathbf{x})$
```

4.6 Dimensionality reduction

Supervised learning introduces models that can be easy to understand, visualize and has well-defined tools and models. However, a dataset can be tedious to work with due to a large amount of descriptors. These descriptors may also be correlated, which means that no new information will be learned from a correlated feature and therefore could be disregarded. Furthermore, a large dataset pose a computational challenge and a reduction in descriptors could potentially reduce the computational time and effort required for any data analysis. Therefore, it would be beneficial to apply a method that finds correlated descriptors and reduce dimensionality of a dataset. This is the idea of *principal component analysis* (PCA).

4.6.1 Principal component analysis

Principal component analysis is an algorithm that tries to find a low-dimension representation of a dataset that contains as much of the variance in the data as possible [117, 123]. Each of the dimensions found by PCA are a linear combination of the features in the dataset, and are known as *principal components*.

We can write the design matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, with p features and n entries, in terms of its column vectors as

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \mathbf{x}_2 & \dots & \dots & \mathbf{x}_{p-1} \end{bmatrix}, \tag{4.25}$$

with a given vector

$$\mathbf{x}_i^T = \begin{bmatrix} x_{0,i} & x_{1,i} & x_{2,i} & \dots & \dots x_{n-1,i} \end{bmatrix}. \tag{4.26}$$

Then we can compute the *covariance matrix* of the design matrix $\mathbf{X}$, which is a measurement of the joint variability of the p features in $\mathbf{X}$. The covariance is defined as

$$\mathrm{cov}[\mathbf{v}, \mathbf{u}] = \frac{1}{n} \sum_{i=0}^{n-1} (v_i - \overline{v})(u_i - \overline{u}), \tag{4.27}$$

where $\mathbf{v}$ and $\mathbf{u}$ are two vectors with n elements each. The covariance matrix is defined by applying the covariance for every pairwise feauture, resulting in a $p \times p$ matrix. On the diagonal, the covariance of two equal features becomes the variance of one,

$$\mathrm{cov}[\mathbf{u}, \mathbf{u}] = \frac{1}{n} \sum_{i=0}^{n-1} (u_i - \overline{u})(u_i - \overline{u}) = \mathrm{var}[\mathbf{u}]. \tag{4.28}$$

The covariance can take values between $\pm\infty$, which give rise to computational issues due to loss of numerical precision. Therefore, we scale the covariance matrix by the variance. This is known as the correlation function

$$\mathrm{corr}[\mathbf{u}, \mathbf{v}] = \frac{\mathrm{cov}[\mathbf{u}, \mathbf{v}]}{\sqrt{\mathrm{var}[\mathbf{u}]\mathrm{var}[\mathbf{v}]}}. \tag{4.29}$$

Since all values are inbetween -1 and 1, we avoid any loss of numerical precision. The resulting covariance matrix $\mathbf{C} \in \mathbb{R}^{p \times p}$ becomes

$$\mathbf{C}[\mathbf{x}] = \begin{bmatrix} \mathrm{var}[\mathbf{x}_0] & \mathrm{cov}[\mathbf{x}_0, \mathbf{x}_1] & \mathrm{cov}[\mathbf{x}_0, \mathbf{x}_2] & \dots & \dots & \mathrm{cov}[\mathbf{x}_0, \mathbf{x}_{p-1}] \\ \mathrm{cov}[\mathbf{x}_1, \mathbf{x}_0] & \mathrm{var}[\mathbf{x}_1] & \mathrm{cov}[\mathbf{x}_1, \mathbf{x}_2] & \dots & \dots & \mathrm{cov}[\mathbf{x}_1, \mathbf{x}_{p-1}] \\ \mathrm{cov}[\mathbf{x}_2, \mathbf{x}_0] & \mathrm{cov}[\mathbf{x}_2, \mathbf{x}_1] & \mathrm{var}[\mathbf{x}_2] & \dots & \dots & \mathrm{cov}[\mathbf{x}_2, \mathbf{x}_{p-1}] \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \mathrm{cov}[\mathbf{x}_{p-1}, \mathbf{x}_0] & \mathrm{cov}[\mathbf{x}_{p-1}, \mathbf{x}_1] & \mathrm{cov}[\mathbf{x}_{p-1}, \mathbf{x}_2] & \dots & \dots & \mathrm{var}[\mathbf{x}_{p-1}] \end{bmatrix}, \tag{4.30}$$

for all vectors x_i where $i = 0, 1, \dots, p-1$. The correlation matrix $\mathbf{K}[x]$ becomes

$$\mathbf{K}[x] = \begin{bmatrix} 1 & \text{corr}[x_0, x_1] & \text{corr}[x_0, x_2] & \dots & \dots & \text{corr}[x_0, x_{p-1}] \\ \text{corr}[x_1, x_0] & 1 & \text{corr}[x_1, x_2] & \dots & \dots & \text{corr}[x_1, x_{p-1}] \\ \text{corr}[x_2, x_0] & \text{corr}[x_2, x_1] & 1 & \dots & \dots & \text{corr}[x_2, x_{p-1}] \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \text{corr}[x_{p-1}, x_0] & \text{corr}[x_{p-1}, x_1] & \text{corr}[x_{p-1}, x_2] & \dots & \dots & 1 \end{bmatrix}. \tag{4.31}$$

The covariance matrix can be rewritten as a function of the design matrix,

$$\mathbf{C}[x] = \mathbb{E}[\mathbf{X}\mathbf{X}^T] - \mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{X}^T], \tag{4.32}$$

where $\mathbb{E}[\mathbf{X}\mathbf{X}^T]$ is the expectation value, and assuming we have normalized the data such that $\mathbb{E}[X] = 0$, we can remove the last term.

Further on, we assume that we can do apply a number of orthogonal transformations by some orthogonal matrices $\mathbf{S} = [\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{p-1}] \in \mathbb{R}^{p \times p}$ with the column vectors $\mathbf{s}_i \in \mathbb{R}^p$. Additionally, we assume that there is a transformation

$$\mathbf{C}[y] = \mathbf{S}\mathbf{C}[x]\mathbf{S}^T = \mathbb{E}[\mathbf{S}\mathbf{X}\mathbf{X}^T\mathbf{S}^T], \tag{4.33}$$

such that the new matrix $\mathbf{C}[y]$ is diagonal with elements $[\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_{p-1}]$. By multiplying with $\mathbf{S}^T$, we arrive at the given eigenvalue number i of the covariance matrix that

$$\mathbf{s}_i^T \lambda_i = \mathbf{C}[x]\mathbf{s}_i^T. \tag{4.34}$$

Dimensions with large eigenvalue have a large variation and can therefore be used to find features with useful information since we multiply the eigenvalue with the eigenvectors. When the eigenvalues are small, it means that the eigenvectors shrink accordingly and there is a small variation in these specific features [124].

So far, we have been leading up to the classical PCA theorem. Assume that the data is represented as in equation 4.25 with $\mathbb{E}[X] = 0$, and assume that there exists an orthogonal transformation $\boldsymbol{W} \in \mathbb{R}^{p \times p}$. We can then define the reconstruction error

$$J(\boldsymbol{W}, \mathbf{Z}) = \frac{1}{n} \sum_i (x_i - \overline{x}_i)^2, \tag{4.35}$$

with $\overline{x}_i = \boldsymbol{W} z_i$, where z_i is a column vector with dimension $\mathbb{R}^n$ of the matrix $\mathbf{Z} \in \mathbb{R}^{p \times n}$.

The PCA theorem states that minimizing the above reconstruction error corresponds to setting $\boldsymbol{W} = \boldsymbol{S}$, which is the orthogonal matrix that diagonalizes the covariance matrix [117]. The optimal number of features that correspond to the encoding is given by the set of vectors $\boldsymbol{z}_i$ with at most l vectors. This is defined as the orthogonal projection of the data onto the columns spanned by the eigenvectors of the covariance matrix. Instead of using the covariance matrix, it is preferable to use the correlation matrix to avoid loss of numerical precision. Additionally, it is important to mention that the covariance matrix is sensitive to the standardization of variables, which is why one should always remember to center the data around before applying PCA. We recommend the reader to read Ref. [117] p. 387 for proof of the classical PCA theorem, as we will not elaborate any further. The algorithm for PCA is shown in Algorithm 4.

Algorithm 4: Principal component analysis algorithm.

```
Set up the design matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ with p features and n entries;
Center the data by subtracting the mean value for each column;
Compute the covariance matrix $\mathbb{E}[\overline{\mathbf{X}\mathbf{X}}^T]$;
Find the eigenpairs of $\mathbf{C}$ with eigenvalues $[\lambda_0, \lambda_1, \dots, \lambda_{p-1}]$ and
  eigenvectors $[\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{p-1}]$;
Order the eigenvalues, and therefore also the eigenvectors, in
  descending order. Keep only those $l$ eigenvalues larger than a selected
  threshold value.
```

Instead of choosing an arbitrary number of dimensions to reduce down to, it is common to choose the number of dimensions that accumulate a sufficient amount of variance. However, it remains a subjective analysis in how many principal components one should include as it will depend on both the specific application and specific data set. If it is impossible to give a motivation for reducing a large dataset to just two or three principal components, there might still be a reason for why to apply PCA to a dataset. PCA can be applied as a preprocessing method to reduce the dimensionality of a dataset, and therefore might drastically improve the efficiency of further supervised learning approaches.

4.7 Practical challenges associated with machine learning

So far, we have covered substantially researched topics such as dimensionality reduction, supervised algorithms and metric evaluation. However, there exists parts of machine learning that does not neccessarily get as much attention, but yet are crucial for the objective of machine learning. In this section, we will briefly mention both known and unknown challenges that are part of building a machine learning model.

The initial phase consists of gathering information systematically. This could be perhaps the most time-consuming part of the entire process, motivated by questions such as how much data is neccessary. The answer to this question is as vague as the question itself, since there is no lower or upper bound but rather a general recommendation that the more data the better. Additionally, we should have an hypothesis that we are collecting descriptors of something that can explain the objective of the entire machine learning process. Indeed, the promises of machine learning are limited to data containing good descriptors. For a supervised learning algorithm, it is neccessary to have one descriptor for the training data that contains information about what should be learned.

Thereafter follows an analysis of the data quality, often called *pre-processing*. This includes identifying outliers and finding out what to do about any potential missing value in the data. Normally, solutions such as removing outliers and filling any missing value with either the mean, median or zero are applied. The data is also required to be transformed into continuous or categorical values. For the latter case we can carry out a one-hot encoding to ensure that any algorithm does not assume one category being more important to another due to a larger number. Furthermore, it could be neccessary to scale the data and reduce dimensionality, with the motivation discussed in section 4.6.1.

If the algorithm has not been chosen yet, this is the time to do so. A clever first-hand approach is to apply a simple algorithm that is not computational demanding to see how it performs on a subset of the data. If the performance is satisfactory, any implementation of a more sophisticated algorithm could be redundant.

Next, the search for optimal hyperparameters while maintaining a generalized model can pose a challenge, but is achiveable. It is popular to apply a cross validation during this process with different evaluation metrics, as discussed in section 4.2.

Eventually, with a chosen algorithm and its optimal hyperparameters, we train the algorithm on the entire preprocessed training data and then perform predictions on unseen data. To avoid any bias in the predictions, it is crucial that predictions on unseen data is done only once. The reason for this is that we do not want to optimalize any model for the actual test data, since this would reduce the generalization and increase the bias.

Part III

Methodology and implementation

Chapter 5

Information flow

The information stream of this project can be regarded as many modular parts connected together in logical pieces, and is strongly influenced by the process that defines a *minimum viable product* (MVP) through iterative development. An MVP is commonly known (in the bussiness world) as a new product that enables the most learning out of the minimum effort possible. This method allows a product to be iteratively evolved by consistent feedback and development, which in return enables cooperation between cross-disciplinary fields.

Furthermore, by having several modules serving as the fundament of the project, it is possible to achieve a long-lasting and robust product that is simple to maintain yet straightforward to develop. Bugs can be tackled through a documented code simultaneously as visible future improvements can be adressed. Therefore, the product is not regarded as completed in any terms, but rather ready for a first release after iteratively finding the mimimum viable product.

The main project of this work can be found on the Github repository *predicting-solid-state-qubit-candidates* [125]. In this chapter we will look into the details and thoughts behind the extraction of data, constructing features, data preparation, data mining and eventually fabricating a generalized model that can predict unseen data with confidence.

5.1 Extraction and featurization of data

The initial step for gathering and building features can be visualised through the flowchart in figure 5.1. Initially, we start by extracting all entries in the Materials Project that matches a specific query. Thereafter, we apply Matminer's featurization tools to make thousands of features of the data. In a parallel step, entries that are deemed similar to the entries from the initial Materials Project query are extracted from AFLOW, AFLOW-ML, JARVIS-DFT, OQMD and Citrine Informatics. Finally, we combine the steps together as interim data that is ready for further analysis.

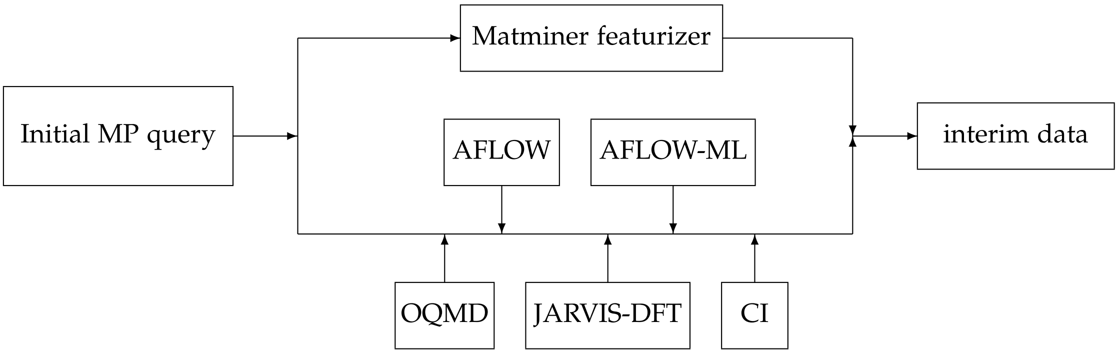


Figure 5.1: The data flow of the main project, starting from an initial MP-query, and ending with a featurized dataset with entries from several other databases. The matminer featurizer step is further visualized in-depth in figure 5.2.

The initial query has the requirement that all entries has to be derived from an experimental ICSD entry, and is reasoned by that we can identify equivalent entries in other databases. Furthermore, all entries in the Materials Project needs to have a band gap larger than 0.1eV. Recall that Materials Project applies the functional GGA in estimating the band gap, which is known to severely underestimate the given electronic property. Therefore, we have chosen a low value to not rule out any potential candidates but high enough to leave out all materials that can be considered metallic. Thus, out of a total of 139.367 entries in Materials Project, our initial requirement is satisfied by 25.352 of the entries.

From figure 5.1 we notice that by using many databases we do not add additional entries that exist in some databases but is not to be found in Materials Project. This is by design since it preserves the versatility of choosing a database to work with. Therefore, one can completely ignore steps such as the initial query of Materials Project or the featurization process, and rather focus on e.g. all the 400.000 entries existing in OQMD. The examples that follows will illustrate the ease of extracting data from several different databases, and can serve as the starting point for other research projects in computational material science.

5.1.1 API and HTTP requests

To extract information from a database it is convenient to interact through an *API* (Application Programming Interface), which defines important variables such as the kind of requests to be made, how to make them and the data format for transmission. Importantly, this permits communication between different software medias. An API is entirely customizable, and can be made to extend existing functionality or tailormade for specific user-demanding modules.

The APIs that will be encountered is handled by the use of *HTTP* (Hypertext Transfer Protocol), which in its simplest form is a protocol that allows the fetching of resources. The protocol is client-server based, such as the client is requesting information and the server is responding to the request.

The most common HTTP-methods are GET, POST and HEAD, which are used to either retrieve, send, or get information about data, respectively. The latter request is usually done before a GET-method for requests considering large amount of data, since this can be a significant variable for the client's bandwith and load time. Following a request, the server normally responds with one of the status codes in table 5.1.

Table 5.1: Numeric status code for response. The leftmost digit decide the type of response, while the two follow-up digits depends on the implemented API.

Status code	Description
2xx	OK - request was successful
3xx	Resource was redirected
4xx	Request failed due to either unsuccessful authentication or client error.
5xx	Request failed due to server error.

A RESTful (Representational State Transfer) allows users to communicate with a server via a HTTP using a REST Architectural Style [126]. This enables the utilisation of Uniform Resource Identifiers (URI), where each object is represented as a unique resource and can be requested in a uniform manner. Importantly, this allows the use of both URIs and HTTP methods in an API, such that an object is represented by an unique URI whereas a HTTP-method can act on the object. This action will then return either the result of the action, or structured data that represents the object.

To provide a Python example, we can check the response by doing a GET request at the database Materials Project RESTful API in code listing 5.1. We use the preamble to version 2 of Materials Project, and add an API-check and an API-key. The response is shown in code listing 5.2. From the output, it is possible to tell that the supplied API-key is not valid, however, the request is valid.

```python
import requests
preamble = "https://www.materialsproject.org/rest/v2/"
```

```
url = preamble + "api_check"
params = {"API_KEY":"unique_api_key"}
response = requests.get(url=url, params=params)
print(response.json())
```

Listing 5.1: Practical example of getting a response from Materials Project database.

```
{"valid_response": True,
"response":
  {"api_key_valid": False,
  "details":"API_KEY is not a valid key.",
  "version":
    {"db": "2020_09_08",
    "pymatgen": "2020.8.13",
    "rest": ""2.0"}
  }
}
```

Listing 5.2: Practical example of response from Materials Project request based on 5.1. The request was done 28. january 2020.

5.1.2 Practical data extraction with Python-examples

For this section, we will show practical examples of how to extract data that might fulfill the criteria for a material to host a qubit candidate given in the theory part. We will begin with the database of Materials Project, and then search for entries in other databases that match entries from MP. This process is reproducable as a jupyter notebook[1] and the databases in question are the ones refered to in the previous section.

Instead of building multiple HTTP-methods from scratch, we will here take a look at the easiest method at obtaining data from each database. The range of data in a database can consist of data from a few entries up to an unlimited amount of entries with even further optional parameters, and has limitless use in applications. However, the amount of data in a database is irrelevant if the data is inaccessible. Therefore, we provide a toolbox in how to extract information in the easiest way possible. This includes looking into the APIs that supports data-extraction and that are recommended by each respective database.

Every data extraction class is based on an abstract parent class. The advantages of using a base parent class are many, such as improving the readability during code reviews, reducing the main barrier for understanding the underlying structure of a project and utilising reusable components. Yet, the main

[1] add and insert DOI for JN 01-generateDataset-notebook.ipynb

advantage of using a base parent class is the fact that it can effortlessly be extended for further implementations since it provides a code skeleton.

Materials Project

The most up-to-date version of Materials Project can be extracted using the python package pymatgen, which is integrated with Materials Project REST API. Other retrievel tools that is dependent on pymatgen includes matminer, with the added functionality of returning a pandas dataframe. Copies of Materials Project exist in many databases, but the latest added entries are not guaranteed to be included in them.

Entries in Materials Project are characterized using more than 60 features[2], some features being irrelevant for some materials while fundamental for others. The data is divided into three different branches, where the first can be described as basic properties of materials including over 30 features, while the second branch describes experimental thermochemical information. The last branch yields information about a particular calculation, in particular information that's relevant for running a DFT script.

To extract information from the database, we will be utilising the module pymatgen. This query supports MongoDB query and projection operators[3], resulting in an almost instant query.

1. Register for an account[4], and generate a secret API-key.

2. Set the required critera.

3. Set the wanted properties.

4. Apply the query.

The code nippet in code listing 5.3 resembles steps $2-4$, and is filtered as the inital query.

```
from src.data.get_data_MP import data_MP

MAPI_KEY = ‘‘very_secret_key_here’’
MP = data_MP(API_KEY=MAPI_KEY)
df = MP.get_dataframe()
```

[2] All features can be viewed in the documentation of the project: https://github.com/materialsproject/mapidoc/master/materials

[3] https://docs.mongodb.com/manual/reference/operator/query/

[4] https://materialsproject.org

Listing 5.3: Practical example of extracting information from Materials Project using pymatgen, resulting in a Pandas DataFrame named entries that contains the properties given after performing a filter on the database. The criteria is given as a JSON, and supports MongoDB operators.

Citrine Informatics

Citrine Informatics is a framework consisting of both HT-DFT calculations and experimental data, which means that the spectrum of stored information varies broadly. We will access research through open access for institutional and educational purposes. Information in Citrine can be stored using a scheme that is broken down into two sections, with private properties for each entry in addition to common fields that are the same for all entries.

In this example, we will gather experimental data using the module matminer. The following steps are required to extract information from Citrine Informatics.

1. Register for an account[5], and generate a secret API-key.
2. Set the required critera.
3. Set the wanted properties and common fields.
4. Apply the query.

The code listed in code listing 5.4 gives an easy example to steps $2-4$ with experimental data as filter, which results in an almost instant query.

```
from src.data.get_data_Citrine import data_Citrine

CAPI_KEY = ''very_secret_key_here''
citrine = data_Citrine(API_KEY=CAPI_KEY)
df = citrine.get_dataframe()
```

Listing 5.4: Practical example of extracting information from Citrine Informatics using matminer, resulting in a Pandas DataFrame named experimental_entries that contains the properties given after performing a filter on the database. The criteria is given as a JSON.

[5] https://citrination.com

AFLOW

The query from AFLOW API [89] supports lazy formatting, which means that the query is just a search and does not return values but rather an object. This object is then used in the query when asking for values. For every object it is neccessary to request the desired property, consequently making the query process significantly more time-demanding than similar queries using APIs such as pymatgen or matminer for Citrine Informatics. Hence, the accessibility is strictly limited to either searching for single compounds or if the user possess sufficient time.

Matminer's data retrievel tool for AFLOW is currently an ongoing issue [127], thus we present in code listing 5.5 a function that extracts information from AFLOW and returns a Pandas DataFrame. In contrast to Materials Project and Citrine Informatics, AFLOW does not require an API-key for a query, which reduces the amount of steps to obtain data. The class searches for an stored AFLOW-data, and initialises a MP-query with the initial criteria if not successful. The resulting query will then be used as input to AFLOW.

```python
from src.data.get_data_AFLOW import data_AFLOW

AFLOW = data_AFLOW()
df = AFLOW.get_dataframe()
```

Listing 5.5: Practical example of extracting information from AFLOW. The function can extract all information in AFLOW for a given list of compounds, however, it is a slow method and requires consistent internet connection.

Restricted by the available API, the resulting query of 25212 entries in Materials Project took place during the period from january to february 2021 and took in total 23 days. Unfortunately, less than 0.02% of the entries screened from Materials Project was present in AFLOW.

AFLOW-ML

In this part, we will be using a machine learning algorithm named AFLOW-ML Property Labeled Material Fragments (PLMF) [116] to predict the band gap of structures. This algorithm is compatible with a POSCAR of a compound, which can be generated by the CIF (Crystallographic Information File) that describes a crystal's generic structure. It is possible to download a structure as a poscar by using Materials Project front-end API, but is a cumbersome process to do so individually if the task includes many structures. Extracting the feature of POSCAR is yet to be implemented in the RESful API of pymatgen, thus we demonstrate the versatility of pymatgen with a workaround.

We begin with extracting the desired compounds formula, their Materials Project IDs (MPIDs) for identification, and their respectful structure in CIF-format from Materials Project. In an iterative process, each CIF-structure is parsed to a pymatgen structure, where pymatgen can read and convert the structure to a POSCAR stored as a Python dictionary. Finally, we can use the POSCAR as input to AFLOW-ML, which will return the predicted band gap of the structure. This iteratively process parsing and converting, but is an undemanding process. The function that handles this is presented in code listing 5.6. Similar to AFLOW-query, this code listing is dependent on MP-data and will apply for a query if the data is not present.

A significant portion of the process is tied up to obtaining the input-file for AFLOW-ML, and fewer structures will result in an easier process. Nevertheless, we present the following steps in order to receive data from AFLOW-ML.

1. Download AFLOWmlAPI[6].
2. Getting POSCAR from MP.
   (a) Apply the query from Materials Project with "CIF", "material_id" and "full_formula" as properties.
   (b) Insert resulting DataFrame into function defined in code listing 5.6.
3. Insert POSCAR to AFLOW-ML.

```
from src.data.get_data_AFLOWML import data_AFLOWML

AFLOWML = data_AFLOWML()
df = AFLOWML.get_dataframe()
```

Listing 5.6: Practical example of extracting information from AFLOW-ML. The function will convert a CIF-file (from e.g. Materials Project) to a POSCAR, and will use it as input to AFLOW-ML. In return, one will get the structure's predicted band gap. It should be noted that this requires the AFLOW-ML library in the same directory.

The resulting ab-initio calculations used an average of 57s/compound, which in total sums up to 16.6 days. In contrast to AFLOW, 100% of the entries was present due to the fact that it is not based on a database but rather a model.

[6] http://aflow.org/src/aflow-ml/ to the same directory as code listing 5.6

OQMD

To extract information from the OQMD, the easiest way was through the interface of Matminer. The difficulty of extraction are mostly regarded to column which are not assigned to a type, however, this is taken care of in the extraction class visualized in code listing 5.7.

```
from src.data.get_data_OQMD import data_OQMD

OQMD = data_OQMD()
df = OQMD.get_dataframe()
```

Listing 5.7: Practical example of extracting information from OQMD through Matminer.

The query is done almost instantly, resulting in a DataFrame containing over 400.000 entries, where 40% of the entries are matching an entry of the initial MP query.

JARVIS-DFT

The newest version of the JARVIS-DFT dataset can be obtained by requesting an account at the official webpage, but with the drawback that an administrator has to either accept or deny the request. Thus, the accessibility of the database is dependent on if there is an active administrator paying attention to the requests, which is a limitation experienced during this work. Another approach is to download the database through matminer, however with the limitation of not neccessarily having the latest version of the database. A third approach is to download a version of JARVIS-DFT that have been made available for requests the 30.04.2020 at http://figshare.com by Choudhary *et al.* [95]. The author provides tools for extraction, yet not compatible with the latest version of Python (3.8) at the time writing (12.03.2021). Therefore, we provide a tool to extract this data through the use of our base class.

```
from src.data.get_data_JARVIS import data_JARVIS

JARVIS = data_JARVIS()
df = JARVIS.get_dataframe()
```

Listing 5.8: Practical example of extracting information from JARVIS-DFT. For this example, we exclude all metals by removing all non-measured band gaps.

We observe that there is no advanced search filter when loading the database from matminer. The author of matminer regards this as the user's task, and is indeed easily done through the use of the python library Pandas.

The resulting screening of 25212 entries from Materials Project was done almost instantly, and it was found 11% and 17.8% similar entries for the TBMBJ and OptB88 functionals with MP, respectively. Moreover, JARVIS-DFT contains information about spin-orbit splitting, but only 0.12% of the calculations was found as a match with the initial MP query.

5.2 Matminer featurization

Before applying any machine learning algorithm, raw data needs to be transformed into a numerical representation that reflects the relationship between the input and output data. This transformation is known as generating descriptors or features, however, we will in this work adapt the name *featurization*. The open source library of Matminer provides many tools to featurize existing features extracted from Materials Project. In this section we will describe how to extract the features from an initial Materials Project query result (see subsection. 5.1.2), and the resulting features. It is beyond the scope of this work to go in-depth of each feature since the resulting dataset contains a quantity of more than 4500 features, but we will here take the liberty to serve a brief overview of the features and refer to each respective citation for more information. The respective table with information regarding 39 distinct matminer featurizers is situated in the Appendix, table B.1.

The motivation behind the choice of featurizers is that we do not precisely know which features that describes a good potential host. A few potential candidates were briefly mentioned in section 2.6.4, while most candidates are probably yet do be discovered. If we had precise knowledge of what to look for, then there is a good chance that the list of hosts would be longer. Therefore, we strive to collect an achievable quantity of descriptors with the hope of getting wiser in terms of describing a potential material host.

To apply matminer's featurization tools, we extend an existing implementation by Breuck *et al.* [128] called the Materials Optimal Descriptor Network (MODNet). The author Breuck *et al.* specifies that MODNet is a supervised machine learning framework for learning material properties based on either composition or crystal structure. To provide the training data for their model, MODNet featurizes (through matminer) structures either from Materials Project or in the form of a structure object made by pymatgen. Their current implementation provides featurization for compositions, structures and sites. However, matminer also provides featurization tools for density of states (DOS) and band structures, therefore we modify MODNet and extend it to fascilitate such featurizations.

One immediate limitation of our extension is that Matminer's tools is dependent on a pymatgen DOS- and bandstructure object. These objects contains information up to 10MB, and becomes a challenge when dealing with data containing several thousand such objects. This is solved by the required features for matminer's featurization for a subsample of the data, followed by a featurization process of the same subsample. When the feaurization is done, we store the new features and throw away the pymatgen features. This is done iteratively for the entire data set. Thus, a compromise between applying several queries and storing information has been done. The scheme can be visualised as the flow chart seen in figure 5.2.

In the extended version of the featurization process, we eliminate all columns that does not have any entries with physical meaning. This is beneficial for several reasons, such as to reduce memory allocated and to preprocess the data. If there are entries existing with both physical and non-physical for the same column, we replace the non-physical meanings with -1 for recognition in a later step. Additionally, we convert columns that are categorical or lacks a numerical representation into a categorical portrayal. Thus, we strive to limit the neccessary steps for further processing of data into a machine learning algorithm. Nevertheless, the featurization process results in 4876 descriptors.

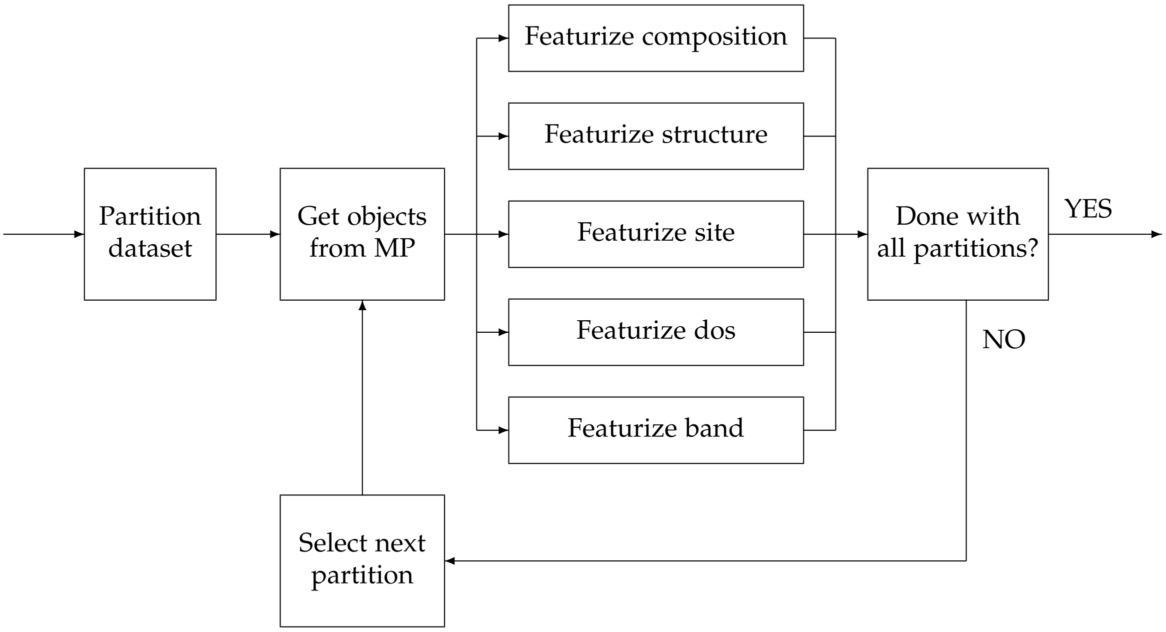


Figure 5.2: The process of the matminer featurizer step as seen in figure 5.1. To limit the memory and computational usage, the data is partioned into smaller subsets where the respective pymatgen objects are obtained through a query to be used in the following featurization steps. This is iteratively done until all the data has been featurized.

Even if the first version of Matminer was released in 2016, many issues concerning daily operational use are still present. During the featurization process in this work, we manually identified 14 (TODO: Update number) erroneous entries that are summarized in the Appendix, table B.2, which were excluded from the dataset. These entries were part of the reason why the featurization process is a time-consuming process, as there is currently no implementation in Matminer that can potential pick up and catch erroneous entries in Materials Project. The process of manually catching such an entry was identified by featurization of single entries causing one of two problems. The first problem could be that an entry could be causing a memory leak which leads to an exceedingly large memory allocation, or it could be that the featurization process needed days to calculate oxidation states for a structure.

5.3 Data mining

After selecting entries based on an initial query from Materials Project followed by a thorough featurization process using Matminer, we face a challenge in terms of defining a training set that we can train data on. This is not only challenging due to the lack of known candidates, but also due to the intricacy of defining materials as bad candidates. Therefore, in this section we describe three different approaches of finding a training set consisting of (1) good candidates and (0) bad candidates.

5.3.1 First approach; the Ferrenti approach

The first approach on defining a training set is based on the criteria from the paper "Identifying candidate hosts for quantum defects via data mining" of Ferrenti *et al.* [100], therefore we will name this approach *the Ferrenti approach*. They suggest a data mining process consisting of four stages by systematically evaluating the suitability of host materials from Materials Project. This procedure is referred to as *data mining*, and we will initially begin with looking at labelling good candidates.

Labelling good candidates

The first stage consists of the following steps to include materials that

Stage 1

- contains elements with a $>$ 50% natural abundance of zero spin isotopes.
- crystallize in nonpolar space groups.
- is present in the ICSD database.
- is calculated nonmagnetic.

The restriction of materials to only contain elements with at least 50% nuclear spin-free isotopes might help with reducing decoherence for all semiconductor-based quantum technologies, as discussed in section 2.6.2. The limit is chosen due to that elemental species with at least 50% nuclear spin free isotopes could likely be isotopically enriched to higher concentrations [100], which has been accomplished for carbon [129, 130] and silicon [131]. In particular, the restriction excludes the use of 53 elements from any species. Any magnetic noise or any presence of electric dipole moment could also potentially increase decoherence of defects. Therefore, we try to reduce any decoherence by restricting materials to possess highly symmetric structures which are nonmagnetic.

Stage two consists applying additional filtering due to practical reasons. This includes removing all materials containing radioactive or toxic elements, as well as removing noble gases because none exists as solids under standard conditions. Rare-earth metals were also excluded due to the difficulty of obtaining pure materials that are sufficiently free of nuclear spin. Lastly, we remove entries that occur mostly in very complex cluster structures (Ru, Os) or are not present in any identified phases (Fe, Ni). Therefore, the additional filter constitutes of obtaining materials that

Stage 2

- does not include Th, U, Cd or Hg.
- does not include any noble gases or rare-earth elements.
- does not include Ru, Os, Fe, Ni

Stage three consists of setting a lower band gap limit similar to that of silicon, but due to severe underestimation of bandgaps by PBE-GGA we set this restriction lower since we do not want to exclude any potential host candidates. The materials are required to have

Stage 3

- a bandgap larger than $0.5eV$ as calculated by MP PBE-GGA.

Finally, the last stage consists of identifying the thermodynamic stability of each compound. A large energy above hull (E Above Hull) is an indication of an unstable compound and would likely cause decomposition, therefore the last filter requires the materials to have

Stage 4

- a calculated E Above Hull $< 0.2eV/\text{atom}$.

The quantity of entries through the different stages have been visualized in table 5.2. The table compares our and their implementation of the same screen procedure with different results. In particular, we see that the remaining materials that have survived four stages of filtering are twice as many. This could be due to the date of extracting since it differs with 13 months, since over 14.000 new entries were added to Materials Project. However, another reason could be due to that they have done additional manual screening. Unfortunately, precise information of which entries that were excluded from the manual filtering were not included in neither the article or the supplementary information [100]. Yet, after doing a data mining procedure we have found 1046 potential candidates that exhibit promising features.

Table 5.2: A table that compares two different implementations of the same screen procedure. Ferrenti *et al.* extracted information March of 2020, while we did the extraction during April of 2021. The adjusted difference is given as our reported entries divided on their reported entries.

Stage	Good candidates based on Ferrenti *et al.* [100]	Good candidates based on approach 1	Adjusted difference
Total entries in Materials Project [92, 132]	125.223	139.367	11%
Stage 1	3363	4347	29%
Stage 2	1993	2226	12%
Stage 3	920	1181	28%
Stage 4	541	1046	93%

Labelling bad candidates

We have now defined good candidates, and turn our attention to defining bad candidates. This is perhaps the difficult part, since we do not know exactly which properties or combination of features a material needs to exhibit for it to be excluded from any use in quantum technology. Therefore, we try to find the opposite criteria of the four stages that defined good candidates.

If we were to turn around all criteria defined in the four stages above (except for energy above hull), it would result in only 52 entries which would make the combined data set very imbalanced. Instead, we try to provide a more general process that includes a larger variety of entries, which could potentially increase the predictor space for bad candidates. The screening procedure for stage 1 requires bad candidates to

Stage 1

- crystallize in polar space groups.
- be present in the ICSD database.
- be calculated as magnetic.

Stage 2

- have a bandgap larger than 0.1eV as calculated by MP PBE-GGA.

We include only ICSD entries and a lower band gap limit for consistency, since our data does not contain entries outside of these limits. The number of entries after stage 1 is 1520, while stage 2 reduces the entries to 684.

5.3.2 Second approach; the augmented Ferrenti approach

In the second approach we try to make adjustment of the first approach to improve the dataset. This approach is therefore named *the augmented Ferrenti Approach*.

Labelling good candidates

The first approach included unphysical criteria such as removing elements that are either radioactive, toxic, elements not occuring under standard conditions, or rare-earth elements that are difficult to obtain. In this approach, we remove those constraints since these are not criteria that neccessarily deem a material as either good or bad for QT, and it is eventually up to experimentalists for evaluation of such practicalities. Therefore, we remove stage 2.

We will in this approach not consider if a material is stable or not, since this is eventually up to experimentalists to evaluate. Additionally, we will include a few interesting elements that showed promising properties as discussed in section 2.6.4, and was originally excluded due to lack of spin zero isotopes. Thus, the second approach consists of the following steps to include materials that

Stage 1

- contains elements with a $> 50\%$ natural abundance of zero spin isotopes except Al, P, Ga, As, B and N.
- crystallize in nonpolar space groups.
- is present in the ICSD database.
- is calculated nonmagnetic.

Stage 2

- have a bandgap larger than 1.5eV as calculated by MP PBE-GGA.

Stage 3

- have a calculated E Above Hull $<$ 0.2eV/atom.

Since we have removed restrictions, we can also infer a stronger one for the band gap. Therefore, we can be considerable more certain if a band gap can a accomodate deep defect due to an increasing amount of entries when removing restrictions.

Labelling bad candidates

For bad candidates, we implement the same strategy as defined for bad candidates in approach 1. The resulting table for both good and bad candidates is found in table 5.3. The table reveals a considerable imbalanced dataset with up to 75% being good candidates, while only 25% of the training data are labelled as bad candidates. However, the training set is 78% larger than in approach 1.

Table 5.3: A table showing the number of entries through the data mining process for good candidates in approach 2 and bad candidates in approach 1 and 2.

Stage	Good candidates approach 2	Bad candidates approach 1 and 2	Ratio
Total entries in Materials Project [92, 132]	139.367	139.367	-
Stage 1	7433	1520	83%/17%
Stage 2	2373	684	78%/22%
Stage 3	2141	–	75%/25%

5.3.3 Third approach; the insightful approach

The third approach is vastly different than the two first approaches in terms of labelling, therefore it is named *the insightful approach*.

Recall, in section 2.6.4 we discussed alternative promising material host candidates. The third approach for finding good candidates is to search our current data for any materials that overlap with known good candidates. Due to a concern of having a too small dataset, we will include materials that are promising and have shown suitable properties to accomodate deep defects that potentially can exhibit quantum effects.

Labelling good candidates

Stage 1

- matches the formulas SiC [6, 28, 36, 45, 46], BN [55, 56], MoS_2[56], WSe_2[56], WS_2[56], GaN [51], GaAs [50], AlN [6, 52], ZnS [48], ZnSe [6], ZnO [48], AlP[6], GaP[6], AlAs[6], ZnTe[6], CdS[6], SiGe [54], C [32, 34, 35] or Si [48, 49].

- is present in the ICSD database.

Stage 2

- have a bandgap larger than 0.5eV as calculated by MP PBE-GGA.

Stage 3

- Manual screening of correct structures.

After stage 1, it was found 202 matching formulas which included 12 entries that had a bandgap lower than 0.5eV. These entries were structures that was reported as unstable in terms of energy above hull calculations, and would decompose into entries that were already present in the data after stage 1 with bandgap substantially larger than 0.5. We choose to exclude these entries with an additional band gap restriction due to the fact that the bandgap is not large enough to accomodate a deep defect. Therefore, these entries were instead labelled as bad entries.

Entries matching the formula C, SiC, BN, MoS_2, WSe_2 and WS_2 were manually screened to see if the entries have a matching structure to the respective candidates discussed in section 2.6.1 and 2.6.3 and 2.6.4, respectively. For C, we admit only three-dimensional diamond-like structures, as explicitly stated in the column tags at Materials Project. Two-dimensional graphite-like structures are labelled as bad candidates, while complex structures (eg. C_{28}, C_{48}, C_{60}) were moved to the test set. For SiC, we admit only entries with the polytypes 3C, 4H and 6H, while moving structures similar to 2H to the test set. Concerning BN, MoS_2, WSe_2 and WS_2, we only admit two-dimensional structures.

The materials AlP, GaP, AlAs, ZnTe and CdS were manually screened for tetrahedrally coordinated structures, and have been included since Weber *et al.* [6] has identified them as potential promising candidates due to acceptable properties defined in requirements (H1-H4) in section 2.6.2. We note that only tetrahedrally coordinated structures of the given formulas were present after the bandgap restriction of 0.5eV.

Since the number of elements in the good candidates are not containing more than two elements, we decide to remove the feature that explains how many elements due to that we do not want the model to discriminate based on this feature. After three stages, a total of 172 entries were labelled as good candidates.

Labelling bad entries

Since the training data that constitutes good candidates are few, we choose to add 400 random entries from the dataset of bad candidates used in approach

1 and 2, in addition to the entries stated above. We only add a subsample for increasing the potential dimensional space for predictions of candidates while avoiding having a too inbalanced dataset. Thus, the total amount of bad candidates accumulates to 418 entries.

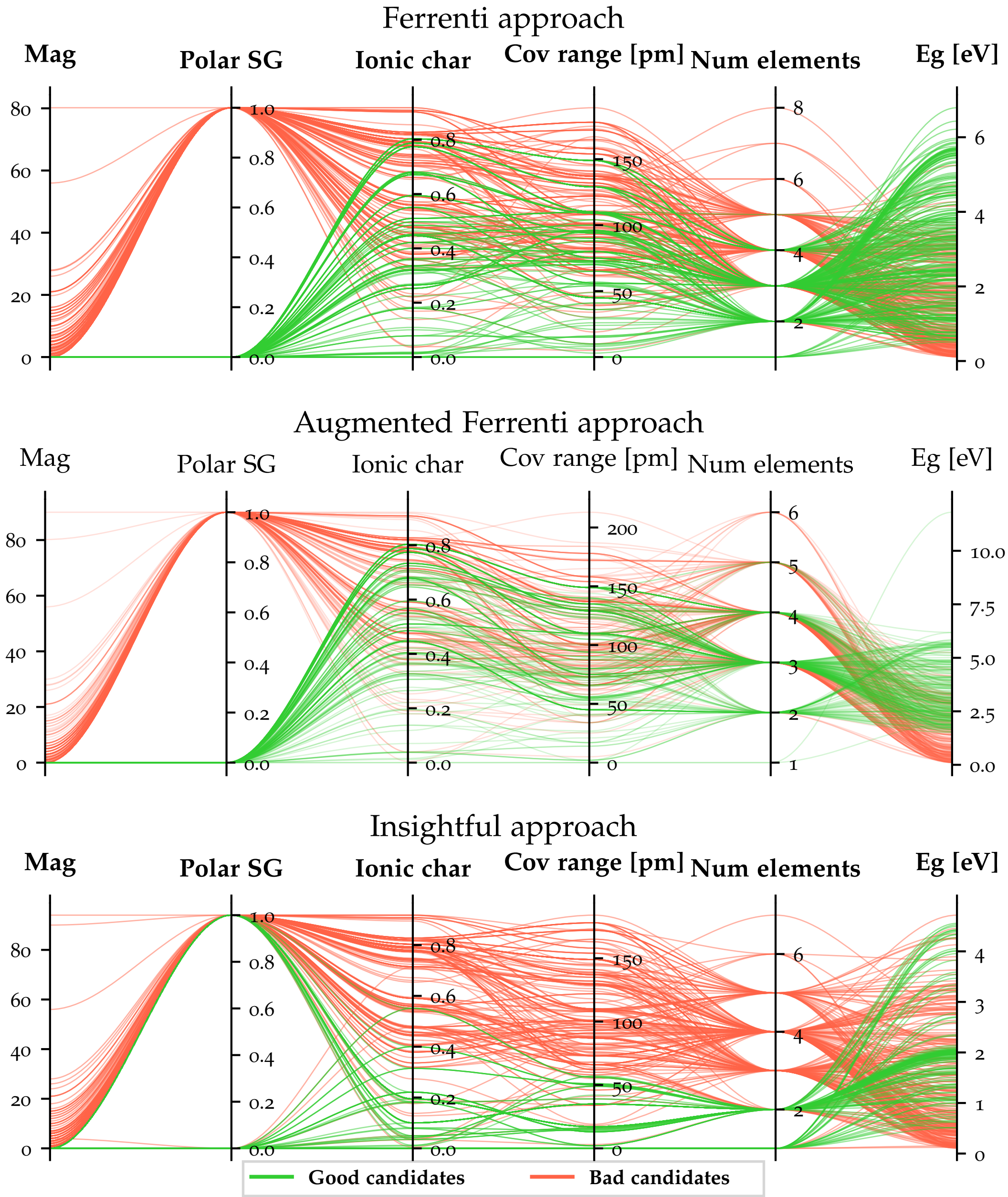


Figure 5.3: Parallel coordinate plots for the different approaches. To limit the data cluttering, we have random collected up to 250 entries for each class and made the lines transparent. For the insightful approach, we have used all 172 good candidates.

5.3.4 Comparison of the approaches

The three approaches provide special emphasis on each their goal. The Ferrenti approach is dependent on choosing only elements with zero spin isotopes together with practical filters, while the augmented Ferrenti approach allows a larger variety of elements and removes the practical reasons for excluding elements. Thus, the first approach targets a more narrow prediction space than the second approach does, and we would expect that the second approach will lead to more predicted candidates compared to approach one. However, perhaps the most restricted approach is the insightful approach. Since we only include known candidates, they should share the same properties and therefore provide a very narrow prediction space.

Unfortunately, the downside of including all the known candidates in one approach is that it becomes increasingly challenging to evaluate the approach or the resulting model. For the two first approaches, we can see if some of the known candidates are present in the predictions, while this is not possible for the latter approach.

We provide a visualization of each approach's training data as a parallel coordinate plot for a few selected features in 5.3. Parallel coordinate schemes [133, 134] represents a multi-dimensional data tuple as one polyline crossing parallel axis. The selected features are found on the x-axis, while the y-axis show the value of the data present. Thus, parallel coordinate plots can turn complex many dimensional data into a compact two-dimensional representation. However, due to data cluttering and that one entry can potentially reserve a large visual area of the figure, the utilization becomes limited when facing large datasets [135]. Therefore, we have chosen to plot a random sample of each class with an upper limit of 250 per class with transparent lines.

The Ferrenti approach and the augmented Ferrenti approach share similarities, such as only having polar space groups present and having an equal amount of upper limit for both ionic character and covalent range. Additionally, they share that entries constitute of up to five different elements. Interestingly, we can see that even if the augmented Ferrenti approach is less restricted, it appears that the entries map over the same dimension based on 5.3.

The biggest difference is seen for the insightful approach. The chosen entries do not posess any magnetization, even if there are both polar and nonpolar space groups present. The range of covalent radius and maximum ionic character is significantly lower than the two other approaches.

To visualize the complexity of the training sets, we have found the two largest eigenvalues of the covariance matrix of the initial data from Materials Project, and transformed the training sets according to the corresponding two eigenvectors. The resulting scatter plots is found in figure 5.4. In green squares, we find the good candidates for each approach, while the labelled

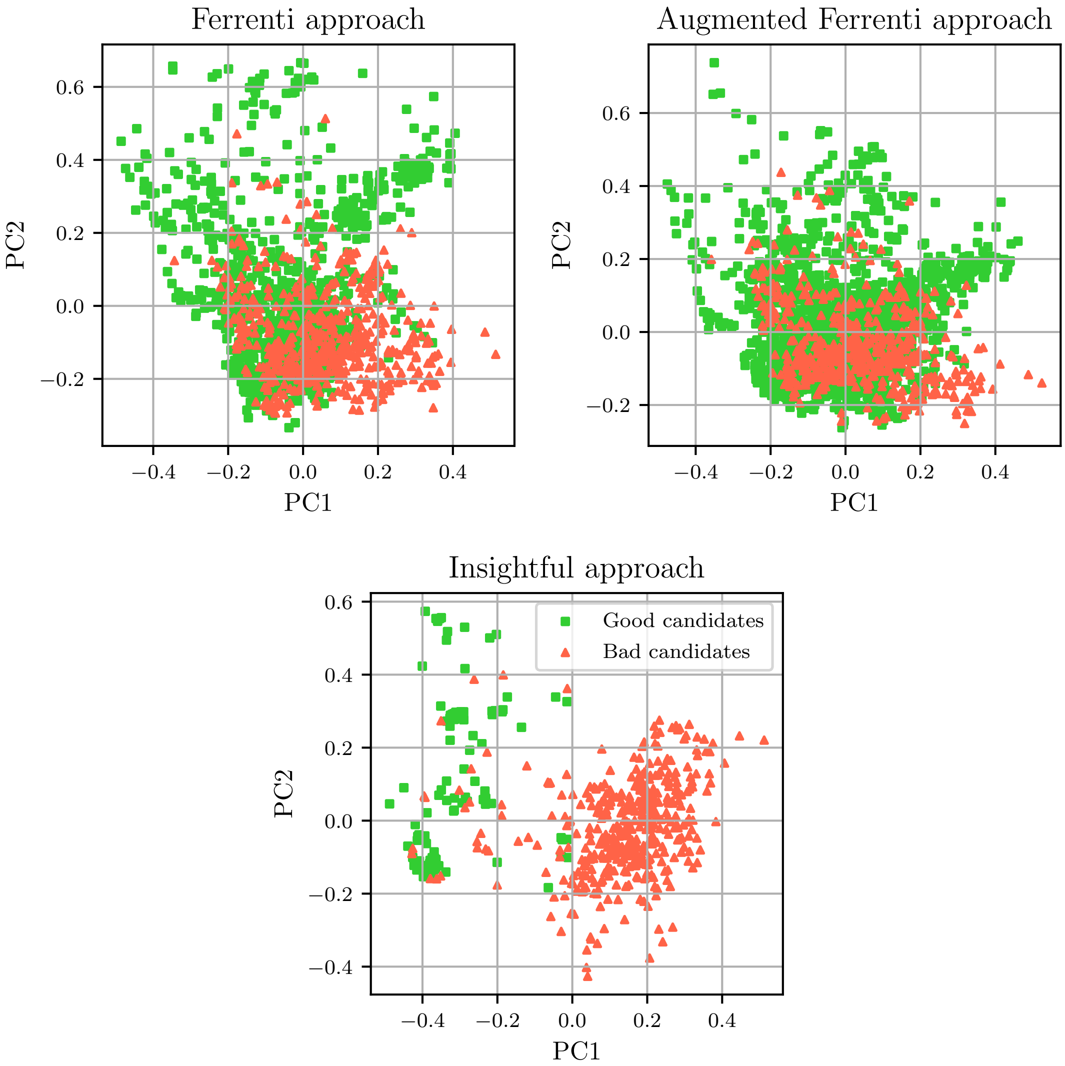


Figure 5.4: Two-dimensional scatter plots for the three different approaches. We have found the two eigenvectors corresponding to the two largest eigenvalues of the covariance-matrix, that is the two most important principal components, of the initial data from the Materials Project query. Then, we have transformed the three training sets resulting from the three approches and visualized it as a scatter plot for visualization purposes.

bad candidates are dressed in red triangles. Due to the simplicity of reducing the number of features down to 2 features, both good and bad candidates for the Ferrenti approach are overlapping which could be challenging for any model that would try to learn a clear-cut boundary. However, for the insightful approach, we can already start to see a trend where the upper left part of the figure is dominated by good candidates. Therefore, we can expect that the two Ferrenti approaches would need either supplementary dimensions for further distinguishment, or could be in trouble of finding a generalized model.

5.4 Model selection

After building a dataset through extraction, featurization and labelling, we turn our attention towards training a model. The flowchart is visualized in figure 5.5. After gathering and featurization, we achieve the interim data that goes through a final data preparation step to become preprocessed data. Then, we perform a data mining step using three different approaches as discussed in the last section. For each of the three approaches, we train and predict in the step called supervised learning. In the summary, we compare the different approaches and results.

In the data preparation step, we assess the quality of the data. Due to the large dimension of 25000 × 4500, we can afford to be picky and therefore we assume that there is a large amount of non-physical values present, accordingly we fill all the missing values with zero and remove all columns with more than 70% containing only zeros. This value was chosen since all categorical features have at least 30% of an respective class present in the column, and a majority of the removed columns contained between 90% and 100% only zeros. This reduces the dimensionality substantially to only 679 features. It should be noted that other methods of data preparation resulted in equivalent preprocessed data due to the large amount of missing values in the data.

Four different supervised models has been selected for each of the three approaches defined in the previous section, resulting in a total of 12 unique models. As discussed in section 4.2, models are unique and does not neccessarily perform optimal on all kinds of data. Therefore, the four models have been selected as a function of increasing complexity and ranges from the simplistic logistic regression and decision trees and up to random forest and gradient boost. We utilize the implementation of Scikit-learn for all models [136].

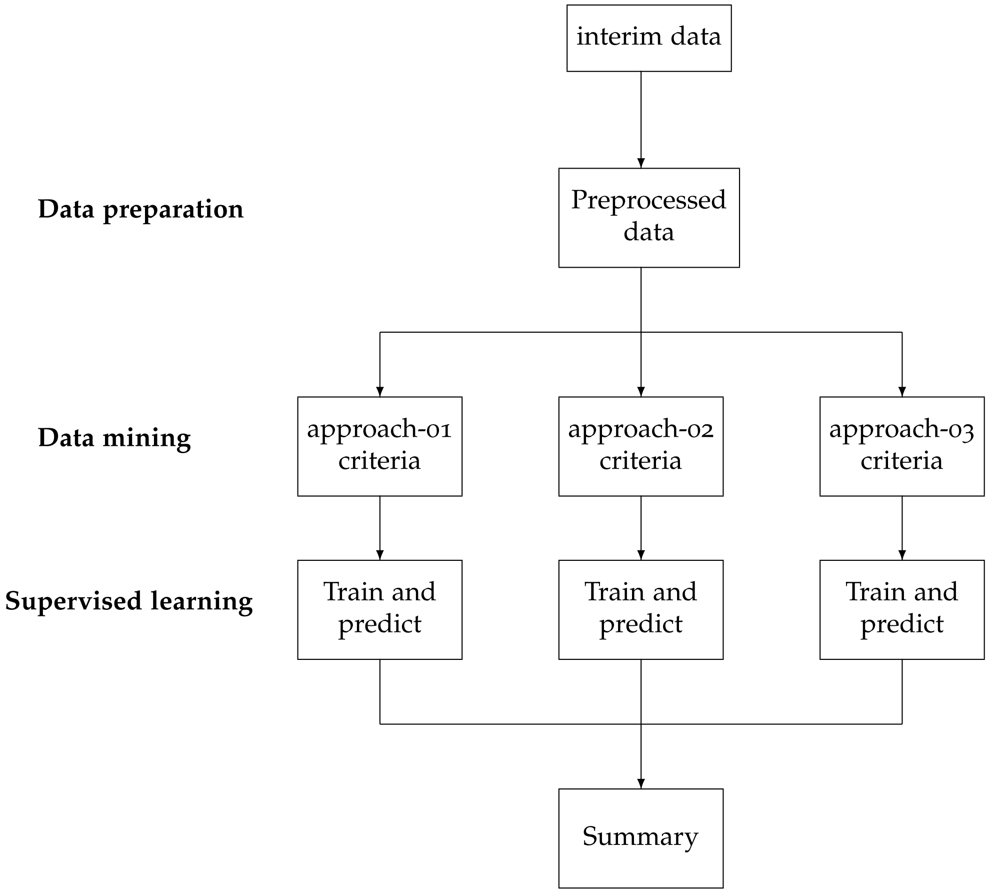


Figure 5.5: A continuity of the flowchart in figure 5.1 that visualise the steps of data preparation, the three approaches in the data mining step and the subsequent supervised learning. Finally, a summary will be provided. From a hierarchic perspective, we find the steps leading up to the preprocessed data as the top level, while each of the approaches are found one level down.

Due to the fact that the current dimension of the entire dataset is still large, we apply the dimensionality reduction technique PCA to the dataset. This is benefical for several reasons, such as finding correlated features and reducing dimensionality. Additionally, it opens up for a visualized interpretation if we were to choose 3 or less principal components.

The optimal parameters are then searched for with the use of Scikit-learn's gridsearch and imblearn's pipeline [136, 137]. Imblearn's pipeline enable the use of resampling methods, in contrast to Scikit-learn's pipeline, but does not differ in any other way. In the pipeline, we provide a standardscaler that scales the data such that every feature will have a mean of 0 and a standard deviation of 1 [136]. Thereafter follows the dimensionality reduction and a supervised learning algorithm. It should be noted that the number of optimal principal components, up to an upper limit of accumulated explained variance of 95%, are also searched for in the grid-search scheme.

Part IV

Results and discussion

Chapter 6

Validation

A thorough testing procedure is important to find out if the code is working as intentionally. The procedure might reveal the presence or absence of bugs, and as a project grows, it can give an indication if a new implementation breaks the original project. Therefore, we present a test-case scenario to test if a few machine learning algorithms are able predict the correct label. It is the same algorithms that will be used in the following chapters, and it will provide us the opportunity to understand how the algorithm works and to draw parallells between the separate works. The entire work of the validation process can be found in the Github project *predicting-ABO3-structures* [138].

The validation process is a reproduction of Ref. [139]. To be able to draw any parallell to their work, we use the exact same dataset in the beginning phase. It should be noted that even if the computational aspects of the validation is closely related to Ref. [139], the work eventually diverges in terms of focus. In their work they include a stability analysis using convex hull analysis in DFT calculations from OQMD, however, we will in this work not decide whether a compound is considered stable or not in an atomic configuration, but rather focus on the predictive aspects of the task. Herein, we will refer to the word "cubics" for perovskites in the cubic structure, "noncubics" for perovskites in a structure other than cubic, and "nonperovskites" for all other cases.

6.1 The ABO3 dataset

The data used in the validation process is offered as supplimentary data from Ref. [139]. They provide the entire training data with both features and labels, but only provide the entries (compounds) of the test data. Therefore, it is neccessary to obtain the features for the test set ourself without knowledge if the resulting test set is identical with Ref. [139].

The training dataset in question contains 390 experimentally reported

ABO_3 compounds. All compounds are charged balanced, and for every compound there is a feature explaining which structure the compound takes, either being a cubic perovskite, perovskite, or not a perovskite at all. Off the 390 compounds, there are 254 perovskites and 136 non-perovskites. Of the 254 perovskites, 232 takes a non-cubic perovskite structure while only 22 takes the cubic perovskite structure. Consequently, this will be visualized by two columns named Perovskite, which represents if a compound is either perovskite (1) or not perovskite (-1), and Cubic, which represents if a compound is cubic perovskite (1), non-cubic perovskite (-1), or not perovskite(0).

The original training dataset consists of 41 unique A atoms and 55 unique B atoms. To generate the test set, we implement all different combinations that are eligible with a total of (VI) oxidation number for the $A + B$ atoms. The resulting test data contains 625 entries and is considerable larger than the training data.

6.1.1 Features

There are in total 9 features we can train a model on. Many of the features are based on the Shannon ionic radii [140], which are estimates of an element's ionic hard-sphere radii extracted from experiment. They are dimensionless numbers, and are frequently used in studies involving perovskite structures of materials since they can be a measurement of the ionic misift of the B atom. This can be used to find the deviation of the structure from an ideal cubic geometry. The octahedral factor for an ABO_3 solid is known as

$$O = \frac{r_b}{r_O}, \tag{6.1}$$

where r_b and r_O are the Shannon radii for the B-atom and oxygen ($r_O = 1.4$Å), respectively. If the octahedral factor is $O = 0.435$, it corresponds to a hard-sphere closed-packed arrangement where B and O ions are touching, while a six-fold coordination appear to require $0.414 < O < 0.732$ according to empirical studies [141]. O, r_A and r_b are represented as features in our data set. We can also compute the Goldschmidt tolerance factor [142], which is defined as

$$t = \frac{r_A + r_O}{\sqrt{2}(r_A + r_O)}. \tag{6.2}$$

The tolerance factor favors the following structures in the interval:

- $t > 1$: Hexagonal nonperovskite.
- $0.9 < t < 1.0$: Cubic perovskite.

- $0.75 < t < 0.9$: Orthorombic perovskite.
- $t < 0.75$: Not a perovskite.

If the tolerance factor is exactly $t = 1$, the structure is known as perfectly cubic and is free for any structural alterations.

Furthermore, the Shannon radii r_A and r_B can be directly correlated with the structure. Perovskites require $r_A > r_B$, and that A-atoms are in a 12-fold coordinated site if $r_A > 0.9\text{Å}$. A-atoms also occur in a sixfold coordinated site if $r_A < 0.8\text{Å}$ and $r_B > 0.7\text{Å}$.

From bond valence theory we can find the valence of an ion to be the sum of valences, that is

$$V_i = \sum_i \nu_{ij} \tag{6.3}$$

$$= \sum_i \frac{\exp(d_o - d_{ij})}{b}, \tag{6.4}$$

where d_{ij} is the bond length while d_o and b are parameters from experimental data. The bond length can be found from 6.4 given the general value $b = 1.4\text{Å}$ and d_o, that can be found from Zhang *et al.* database [141]. The valence of an ion is associated with its neighboring ions and the chemical bonds, and therefore the band length d_{AO} and d_{BO} are included in the data set.

The two last features originates from the Mendeleev numbers of Villars *et al.* [143] for the A- and B atom, MA and MB, respectively. The given values positions the elements in structurally similar groups. This means that he groups the elements in the following interval.

- s-block $\in \{1, 10\}$.
- $Sc = 11$.
- $Y = 12$.
- f-block $\in \{13, 42\}$.
- d-block $\in \{43, 66\}$.
- p-block $\in \{67, 10\}$.

The dataset and its features have been visualized in the parallel coordinate [133] figure 6.1, and reveals several trends already. We can observe that an entry's A atom should preferably have a small Mendeley number (MA) and a large bond length d_{AO}. Yet, perhaps the most clear trend is the tolerance factor that should be around 1. A parallel coordinate plot can easily show

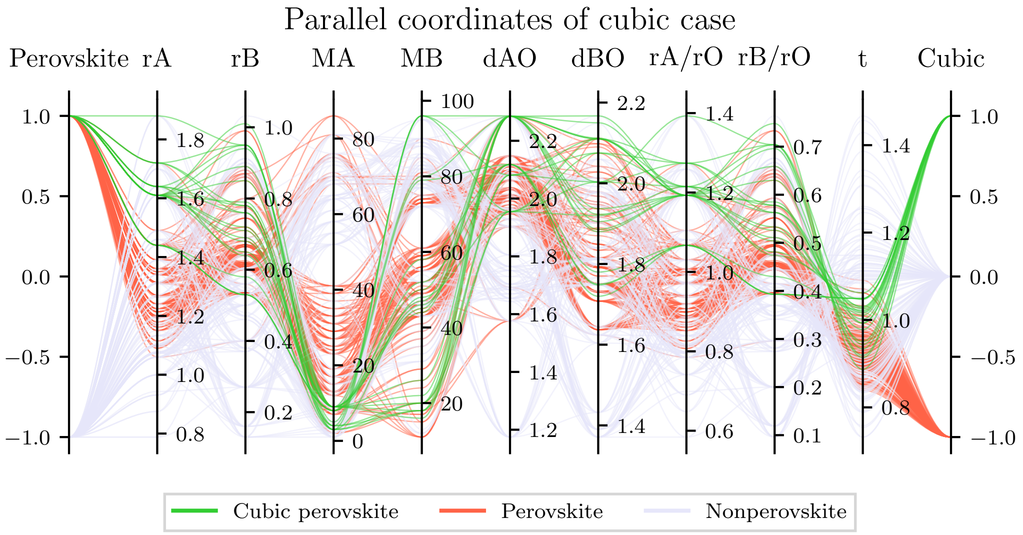


Figure 6.1: A parallel coordinate plot of the perovskite dataset, where the color is given by the cubic label of an entry. A cubic perovskite is labelled as 1, while only a perovskite as -1, or not a perovskite as 0.

trends, but becomes harder to interpret for many features and entries with a growing amount of overlapping. The trend for t values becomes harder to interpret when comparing with the distribution of entries for t-values in figure 6.2. From the distribution we learn that there is an overlap of perovskites or not for tolerance factor values in the interval 0.8 to 1.0, but the label perovskite is in general preferred. Additionally, we see that the interval $q1, q3$ for the label (1) completely overlaps with the corresponding interval for non-perovskites (-1), with very few entries outside of the intervals. This is presumably due to easy labelling for entries that rest outside of the intervals, but the exclusion of entries could potentially alter any model due to not enough entries.

6.2 Implementation

The machine learning classifiers that we will utilize are logistic regression, random forest and gradient boost. The implementation is optimized for adding new algorithms from libraries such as sklearn [136] or imblearn [137] with only few lines of code. This is in particular visualized through the implementation of the current algorithms in code listing 6.2, since a special emphasis on reuse and simplicity of code is in focus of this project.

```
InsertAlgorithms = [
  LogisticRegression(),
```

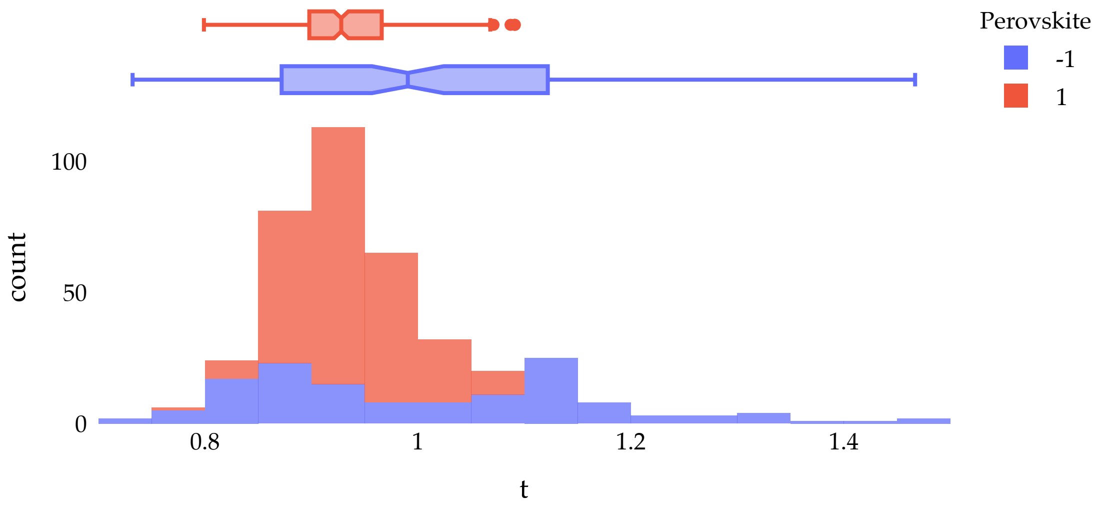


Figure 6.2: The t-distribution of entries in the dataset for perovskite (1) or not (-1). The upper part for perovskite (1) displays minimum value at 0.80, q1 at 0.90, median at 0.93, q3 at 0.97 and max at 1.10. For the non-perovskites (-1), the minimimum is at 0.73, q1 at 0.87, median at 0.99, q3 at 1.12 and max at 1.47.

```
    DecisionTreeClassifier(),
    RandomForestClassifier(),
    GradientBoostClassifier()
]
InsertAbbreviations = [
    "LOG", "DT" ,"RF", "GB"
]
InsertPrettyNames = [
    "Logistic regression",
    "Decision tree",
    "Random forest",
    "Gradient boost"
]
```

The predictions are divided into two parts; perovskite predictions and cubic perovskite predictions. We apply the standard scaler of sklearn [136] to the training data, followed up by a search of optimal hyperparameters using a 5x5-stratified cross-validation. This ensures that the percentage of perovskites (cubic perovskites) or not are the same in every subsample in a cross validation as it is in the entire dataset. This is not neccessarily important for the perovskites predictions due to 65/35% of perovskites or not, but becomes significant for the cubic case where the ratio of cubic perovskites or not are 91/9%.

6.3 Results and discussion

Utilising four different classifiers on two different tasks, starting with prediction of perovskite and then prediction of the predicted perovskites into cubic perovskite or only perovskites, yields in total eight different models. We search for optimal parameters for each of the two tasks. Decision tree, random forest and gradient boost share the range of maximum depth starting from 1 and up to 8, while we optimise logistic regression for regularization parameters in the range of 10^{-3} to 10^{5}.

6.3.1 Technical details on ML classifiers

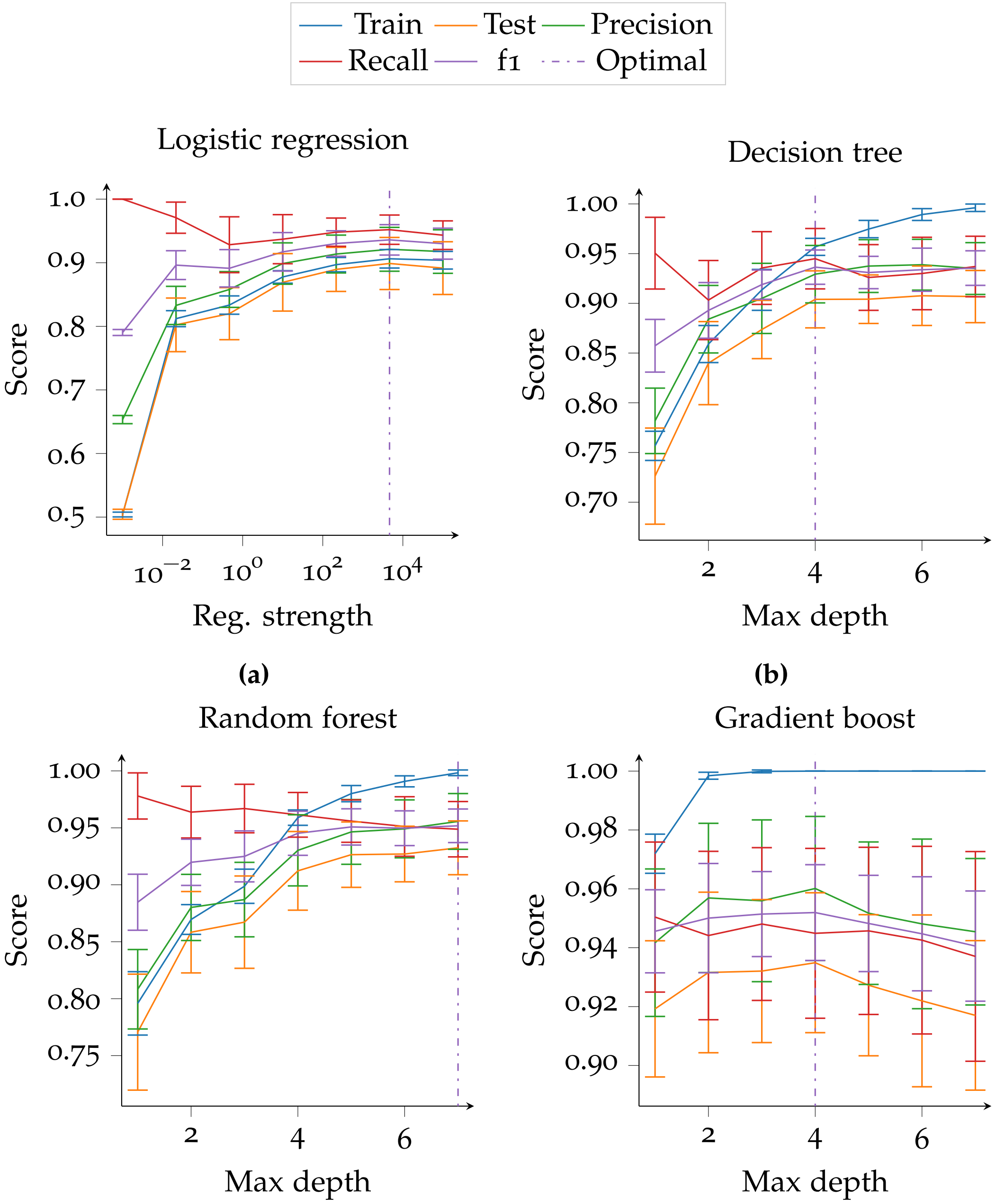
Perovskite case

We first consider the ML classification of known ABO_3 into perovskite or nonperovskites. A search for optimal hyperparameters using scikit-learn's grid search scheme [136] reveals the following table with optimal parameters 6.1. We find that all classifiers have for all scores at least 90% accuracy, with gradient boost performing sligthly better than the rest.

Table 6.1: Table with corresponding best estimators during a grid search scheme for predicting perovskites or not. The test score is here referred to as a balanced accuracy score, and we list all standard deviations in paranthesis.

Model	Mean test	Mean precision	Mean recall	Mean f1
LOG	0.90(0.041)	0.92(0.034)	0.95(0.023)	0.94(0.024)
DT	0.90(0.029)	0.93(0.029)	0.95(0.033)	0.94(0.017)
RF	0.93(0.023)	0.96(0.025)	0.95(0.024)	0.95(0.015)
GB	0.94(0.025)	0.96(0.025)	0.94(0.036)	0.95(0.019)

The parameter search is visualized in figure 6.3 for all four models. For logistic regression, we find that by increasing the regulariation the model becomes more general due to a better compromise between precision and recall. For the decision tree model, we find that the optimal maximum number of depth should be 4. It is clear that the training accuracy increases for larger depth, yet the other test evaluation metrics does not improve, causing the model do be prone for overfitting. Random forest, on the other hand, experience an improvement in scores for all metrics with increasing depth, except for the recall. We find a high recall for all models with an underfitting model due to a imbalanced dataset with a larger amount of perovskites than nonperovskites, and recall is the metric for evaluating if perovskites are correctly predicted. Random forest experience a good compromise between recall and precision with maximum depth at 7. Lastly, we find the optimal depth of gradient boost as 4, whereas larger values tend towards overfitting.



(a)

(b)

(c)

(d)

Figure 6.3: Four figures displaying hyperparameter search for predicting perovskites or nonperovskites. The best estimator is visualized for all hyperparameters as a function of (a, b and c) max depth or (d) regularization strength during a grid search with a 5x5 stratified cross validation. The dotted lines marks the optimal hyperparameter-combination, while the error bars visualizes the standard deviation.

A total of 25 classification attempts were done, and we choose the cubic training dataset based on perovskites that the models were able to predict correctly atleast 50% of the time. None of the perovskites were excluded for Random forest and gradient boost due to high correct prediction rate, but 11 perovskites were wrongly predicted as nonperovskites by the logistic regression, while the number was 4 for the decision tree model. Importantly, all models were able to predict the cubic perovskites as perovskites, which could potentially alter the further prediction due to a small amount of cubics.

Cubic perovskite case

Then, we consider the ML classification of known perovskites into cubic perovskites and noncubic perovskites. Due to a severaly imbalanced dataset with one cubic perovskite for every ten perovskites, we randomly pick perovskites to include in the training set such that the class balance becomes more or less equal to 50 : 50 ratio. Thus, we leave out a large part of the data but it is found helpful to reduce the variation of the evaluation metrics. Specifically, this means that the logistic regression is training on a dataset containing 22 cubics and 20 noncubics, while the three remaining models train on 22 cubics and 21 noncubics.

The optimal combination of hyperparameters during a 5×5 stratified cross validation grid search result in table 6.2. We recognize an increase in standard deviation for the metrics as every prediction counts higher due to a small dataset. Interestingly, we find the decision tree model as the best performing model with 0.98 f1 score, while logistic regression also perform well with 0.96. Random forest and gradient boost experience a f1 score of 0.93 and 0.94, respectively. The relevant hyperparameters found was for logistic regression the regularization term of 0.46 and number of iterations at 200. For decision tree, the optimal maximum depth was set as 1 with increasing deviations for increasing values. Therefore, we believe that the model has potentially set a decision boundary that nearly all entries follow. Random forest and gradient boost performed optimally for the maximum depth of 3 and 4, respectively.

We observe that all models experience high scores, but due to the small dataset we need to bear in mind that if we were to add a single datapoint, which would be predicted falsely, could potentially alter the scores up to 5%. Therefore, we cannot accurately determine if the models perform but rather find a general trend due to the data present.

Table 6.2: Table with corresponding best estimators during a grid search scheme for predicting cubic perovskites or only perovskites. The test score is here referred to as a balanced accuracy score, and we list all standard deviations in paranthesis.

Model	Mean test	Mean precision	Mean recall	Mean f1
LOG	0.96(0.073)	0.96(0.080)	0.96(0.123)	0.96(0.085)
DT	0.97(0.046)	0.96(0.078)	1.00(0.000)	0.98(0.043)
RF	0.92(0.065)	0.89(0.146)	0.97(0.071)	0.93(0.061)
GB	0.94(0.083)	0.92(0.104)	0.97(0.071)	0.94(0.074)

6.3.2 Predictions of new compounds

With the optimal hyperparameters found from the 5 × 5 cross validation, we train the four models on the entire dataset with the labels indicating a perovskite or a nonperovskite. Then we use the each respective cubic training dataset, of varying size due to perovskite misclassifications, to train four new models that will predict if a predicted perovskite will belong to the cubic perovskite class or non at all based on an equal class distributed training set.

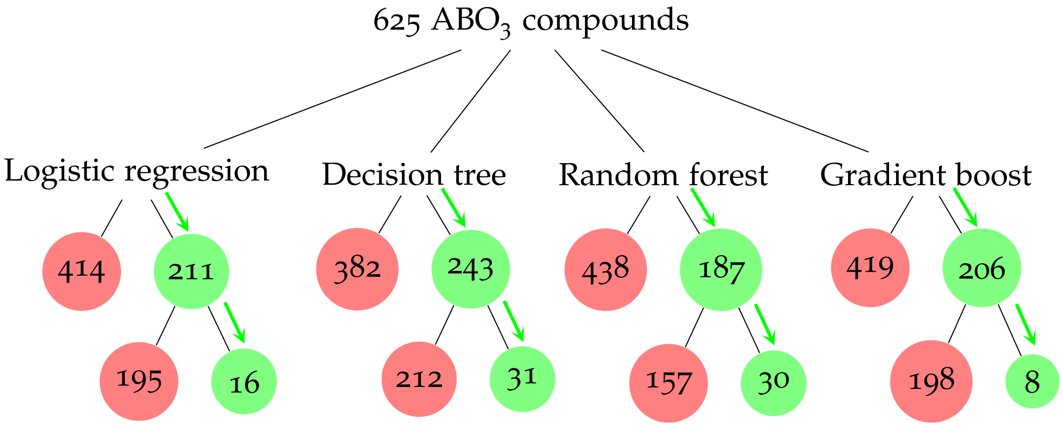


Figure 6.4: A figure visualizing the top-down workflow of predicting new ABO_3 cubic perovskites for all models. First layer of predictions display the number of predicted perovskites in green, while nonperovskites in red. Thereafter follows the prediction of perovskites into cubic perovskites in green, while only perovskites as red.

For the predictions, we input the test set consisting of 625 unlabelled possible perovskites. The workflow is visualized as a top-down approach in figure 6.4, where we start with the input of test set. Thus, every model has the same input. For the first branch of each model, we find the green and red circle indicating the number of predicted perovskites or nonperovskite, respectively. Then, we use the predicted perovskites to predict if they belong to the cubic perovskite class (green) or not (red). To simplify the top-down approach, we have added arrows to indicate the direction of predictions.

For the case of prediction perovskites or nonperovskites, we find that logistic regression, decision tree, random forest and gradient boost predict 211, 243, 187 and 206 as perovskites, respectively. Interestingly, we find decision tree as the one admitting most entries into the perovskite category, but about 60 of the entries predicted as perovskites are done so with the precision of a coin-flip, that is about 50%. The majority of these entries are also based on coin-flips for the other models. We observe that all models agree on classifying 141 of the initial 625 entries as perovskites.

We then turn to the prediction of ABO_3 compounds into cubic perovskites based on the model's predicted perovskites, which is the second level of branches in figure 6.4. We find that most of the perovskites are not predicted in the cubic structure, with decision tree and random forest predicting the most cubic perovskites with 31 and 30, respectively. Importantly, they agree on 29 of the predictions, where 17 of the entries have Pb as A-atom while the rest include K, Rb, Cs, Ba and Sr as A-atom. By including gradient boost in the comparison, we find 7 out of 8 cubic perovskites as predicted by gradient boost also in the 29 cubic perovskites predicted by decision tree and random forest. These predicted cubic perovskites are $PbIrO_3$, $PbRuO_3$, $RbBiO_3$, $BaVO_3$, $PbCoO_3$, $PbCrO_3$ and $PbNiO_3$. Logistic regression, on the other hand, predicts cubic perovskites with the A-atom being one of the alkali metals K, Rb, Cs or the alkaline metal Sr, but disagrees with gradient boost of the choice of B atom since no clear trends have been observed for neither. We believe the randomness of the B-atom originates from when we balanced the training sets, where we removed over 70% of the training set, and consequently removed important distinctions of information.

Thus, we experience similar results as Ref. [139] but with one important distinguishment; none of the models seems to be able to verify their suggestion of any Tl as an eligible A-atom. All models, however, agree on one entry as a cubic perovskite with 1 in probability, which is $RbBiO_3$.

6.4 Conclusion

In this validation chapter we implemented four algorithms, namely logistic regression, decision tree, random forest and gradient boost to predict if experimental data of ABO_3 solids take the perovskite or cubic perovskite, perovskite or nonperovskite structure. Based on this list, we optimalize the models using Scikit-learn's [136] gridsearch scheme during 5×5 stratified cross validations. We approached the task in two steps; (1) predict perovskites or nonperovskites and consecutively (2) predict cubic perovskites out of the predicted perovskites. For the second step, we balanced the training set of perovskites and cubic perovskites until it was approximately 1 : 1 ratio of each by randomly selecting the majority class. Thus, we achieved consistent results but with the loss of data points.

Even if we experienced discrepency in the models and statistical fluctuations of the data, we note that the models were able to independently agree on that 141 ABO_3 compounds were predicted as perovskites. Additionally, all models agreed on one cubic perovskite out of the 141 perovskites, which was $RbBiOO_3$. However, we found a tendency for all models to prefer the alkali metals K, Rb, Cs or the alkaline metals Ba and Cr for the A-atom in ABO_3 compounds that take the cubic perovskite structure.

We note that this work was a reproduction of Ref. [139], where we initially started with the identical dataset given in the supplementary table. However, we are unable to verify if we are using the same test data due to unavailability of their data and features used for the predictions. We have followed their discussion in regards of generating the test set, and we have made our approach freely distributed under the MIT license on the Github repository [138]. We achieve similar results, but with one important distinguishment which is that we do not find the Tl atom as an eligible A-atom for the cubic perovskite structure of ABO_3 compounds. On a final note, we believe that the verification of a compound's final structure is up to experimentalists to confirm.

Chapter 7

Optimalization

This chapter is named optimalization due to its contents; here we will account for the choices we compose to optimize the four machine learning algorithms for each of the three approaches. Initially, that involves finding what information is stored within the databases and the compromise of gathering the information, which further evolves into finding optimal hyperparameters for each approach.

The first step of this work was to find data from Materials Project, involving entries that are associated with an ICSD structure and have a PBE-GGA calculated band gap of minimum 0.1eV. Out of 126.335 existing entries in Materials Project, 48.644 (39%) were found to have an associated ICSD-structure, while 65.783 (52%) materials had a calculated band gap of at least 0.1eV. It was found that 25271 (20%) materials have the band gap minimum and an associated ICSD-structure. It should be noted that these numbers are based on data extraction in December of 2020, while the extraction from other databases and featurization related to this work was done in the time period of December 2020 to March 2021. In February of 2021, over 30.000 new materials were added in a large update[1]. These new entries are not included in this work, and therefore the number of entries in our Materials Project are based on the latest release in 2020, which is named V2020.09.08.

Two visualizations of two different distributions of the data is found in figure 7.1 and 7.2. The first figure visualize the distribution of oxid types as a function of compound type, and reveal that the majority of compounds are either binary, ternary, quaternary or quinary, hence the majority of the materials are oxide. This is important to know considering our labelling approaches, in particular the insightful approach where we handpicked good entries. Only a single oxid (ZnO) was deemed a potentially good candidate, which will be interesting to compare towards the different models and approaches.

[1] https://matsci.org/t/materials-project-database-release-log/1609/16 23.04.2021

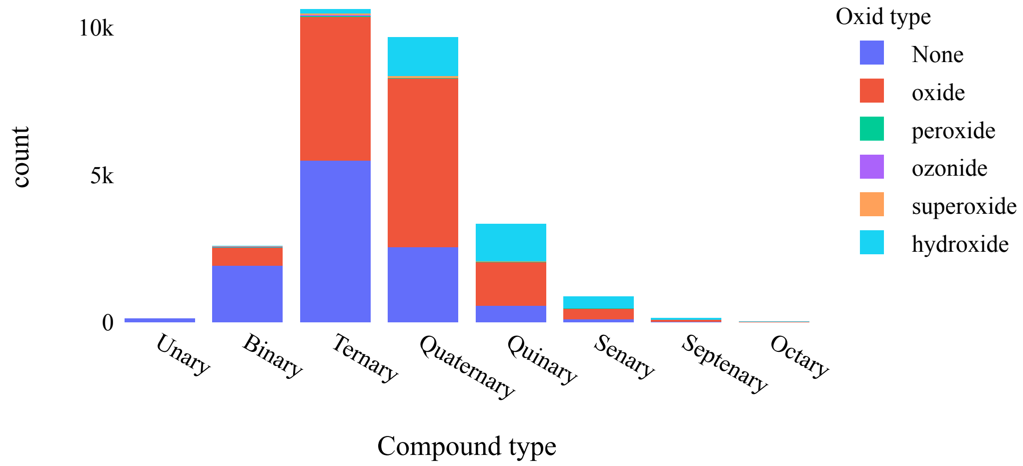


Figure 7.1: Distribution of oxid types as a function of number of elements in compounds in the data. The majority of the entries are found as oxides, while the second most frequent type is not an oxid.

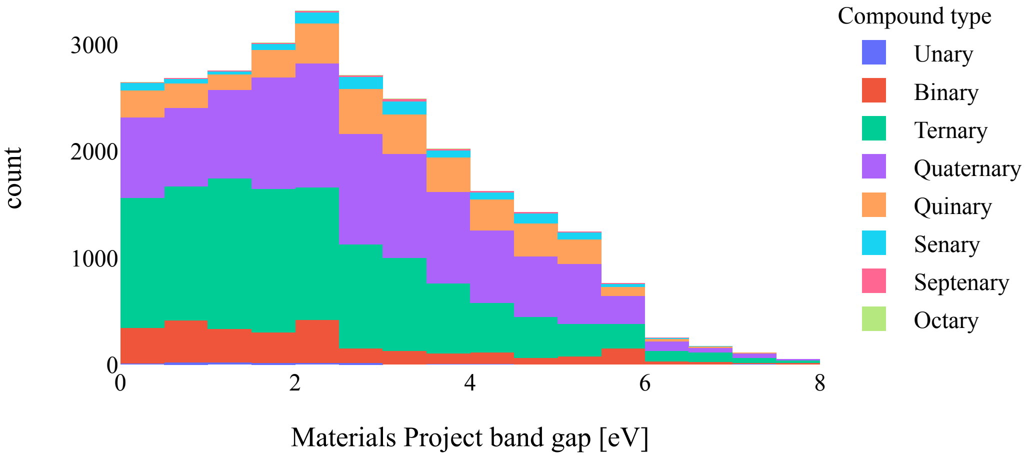


Figure 7.2: Distribution of band gaps as function of compound type in the data. The majority of compounds are ternary and quaternary, while the simpler compounds are few.

The second figure visualize the compound type as function of band gap, as calculated by Materials Project. Most of the materials existing in the data has a band gap lower than 2.3eV, where ternary compounds are most prominent. For larger values, we observe that quaternary compounds becomes dominant for larger values.

7.1 Comparing functionals for band gaps

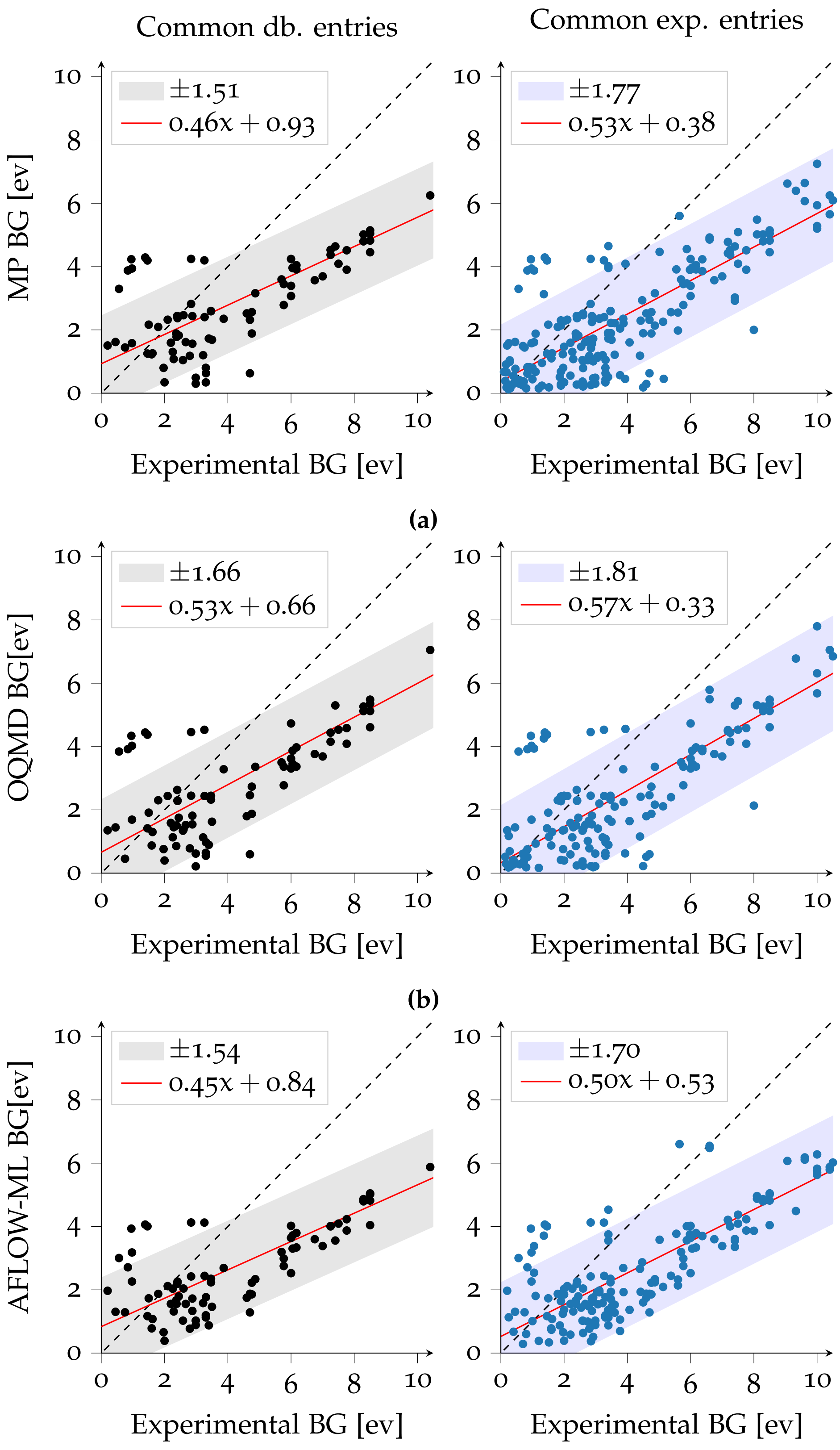
Since the true size of a band gap can not be accurately determined by ab-initio calculations, we provide information regarding five different methods to obtain band gaps as visualized in figure 7.3. We have extracted experimental band gaps from Citrine Informatics that match the entries made by the initial MP query, involving entries that are associated with an ICSD structure that have a PBE-GGA calculated band gaps of minimum 0.1. All the band gaps to the left are found common with all databases through screening of correct structure, space group and ICSD-ID, while the figures to the right are only compared to the experimental database of Citrine Informatics. We found it helpful with the ICSD-tag to find similarities due to databases often have different norms and data-structures of descriptors, which proves challenging for comparison of stored calculations. If we were to exclude ICSD-tags, it would result in a much larger foundation to find similar entries, however, we found that the determination of similar entries would experience a large deviation when it comes to structures. By including an ICSD-tag, we reduce the basis of comparison but find more than 98% identical space group for entries in each database compared to Materials Project.

It was found that a very small portion of the data extracted from AFLOW was associated with an ICSD-tag, only 5 similar entries to the other databases, and therefore we have excluded the database from further consideration.

In the figures of 7.3, we observe each entry marked as black or blue dots. The dotted lines visualize the optimal ratio of estimated band gap to experimental values, while the red lines shows an linear least square fit to the data with the scrabbled area being the 95% confidence interval. The data that constitute the left figures are based on 82 similar entries, while the right figures constitute of more entries depending on the respective database. The data restriction was due to a small experimental database.

Initally, we wanted to include the right figures in the attempt of reducing the confidence interval with increasing the data points, but instead we find that the uncertainty of the confidence interval increase for all ab-initio calculations. This is due to the fact that the majority of the new entries are found for low band gap values, where the mismatch between experimental and calculated values are the largest. The discrepency seems to be largest for values under 5eV, where entries are either calculated to have a very large band gap where the experimental values report a very low band gap, which is also true the opposite case. One reason for this is that we have no information regarding the experiment where the band gap was determined. The information we from the experimental database is only considered the chemical formula of a compound, whereas the structure or ICSD-entry remains unknown. However, the same data of experimental values have been considered through other articles [100, 115].

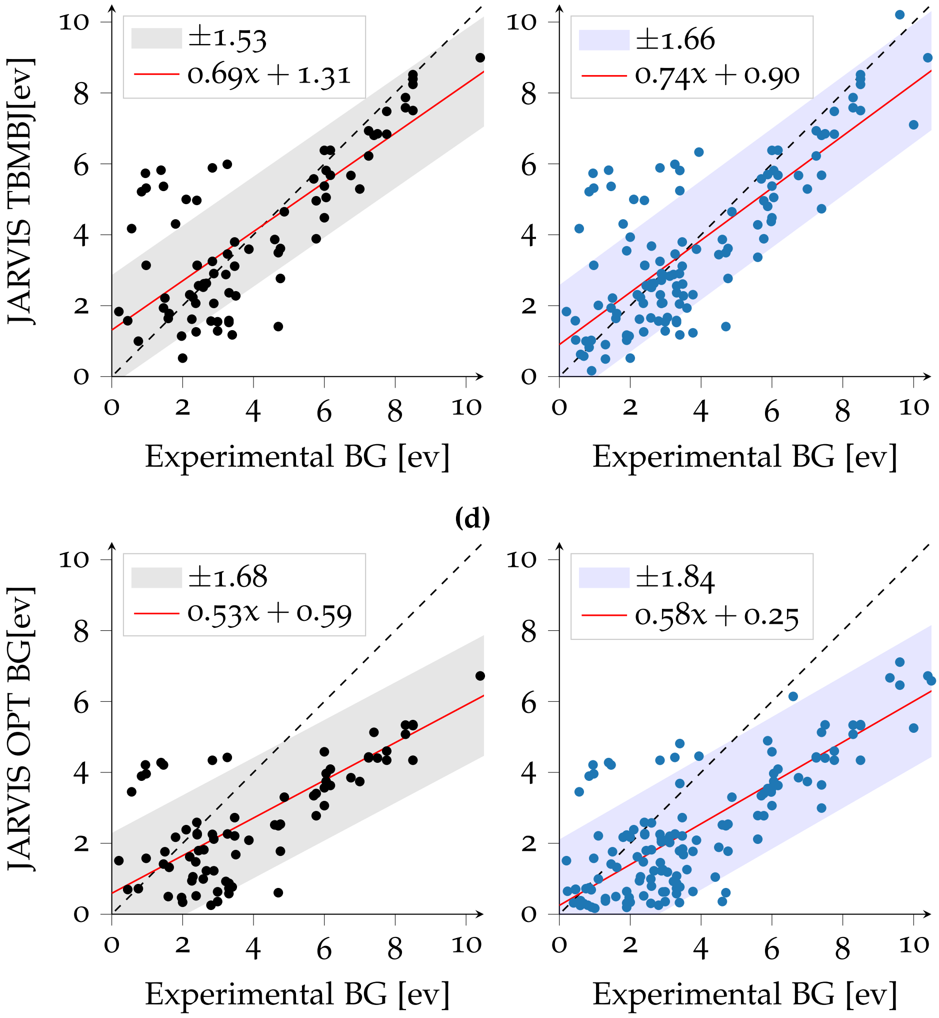
Therefore, we find that the functional applied for Materials Project are found to underestimate the band gap with $30-60\%$ while OQMD underestimates the band gap by $25-55\%$. AFLOW-ML also severily underestimates the band gap by $30-60\%$, but additionally have problems to accurately predict if a material is a metal or not. Many materials with both experimental and ab-initio calculations that showed a band gap of more than 1eV was predicted as metals by AFLOW-ML. JARVIS-DFT, on the other hand, was found to underestimate the band gap by $20-60\%$ for the OptB88 and $0-30\%$ for TB-mBJ functionals.


Common db. entries
Common exp. entries
MP BG [ev]
±1.51
0.46x + 0.93
±1.77
0.53x + 0.38
OQMD BG[ev]
±1.66
0.53x + 0.66
±1.81
0.57x + 0.33
AFLOW-ML BG[ev]
±1.54
0.45x + 0.84
±1.70
0.50x + 0.53
Experimental BG [ev]
0
2
4
6
8
10

(a)

(b)

(c)



(d)

(e)

Figure 7.3: Comparison of reported experimental band gaps to those calculated by (a) Materials Project, (b) Open Quantum Materials Database, (c) AFLOW-ML, (d) JARVIS-DFT (TB-mBJ) and (e) JARVIS-DFT (OptB88). The figures to the left show reported band gaps that have been found to be common through all databases, while the figures to the right are only common with experimentally reported values from Citrine Informatics. All entries have been extracted in the period of january to march of 2021.

7.2 Technical details on ML classifiers

In the evaluation of the approaches, we apply a 5 × 5 stratified cross-validation when iterating through the hyperparameter combinations. We acknowledge that the three approaches experience imbalanced datasets, but from the Validation chapter we find that by adjusting the class balance helped with reducing the variance due to a very small dataset and a class ratio of 1 : 9. In this section, we find all three approaches to have substantially larger datasets than the cubic perovskite dataset, thus we choose to not apply any technique for balancing the classes. Instead, we apply four different algorithms to compare them up to each other, and use four different evaluation metrics to estimate how the classifiers are performing.

For random forest, gradient boost and decision tree, we found that by adjusting the parameters responded to severe overfitting except for the default values defined by Scikit-learn. The only parameter that we found could potentially improve the evaluation metric f1 was maximum number of depth for the trees grown, which we adjusted between 1 and 8. For logistic regression, we choose to adjust the regulariation strength with seven logaritmical adjusted values 10^{-3} to 10^{5}, and use either 200 or 400 iterations to reach convergence.

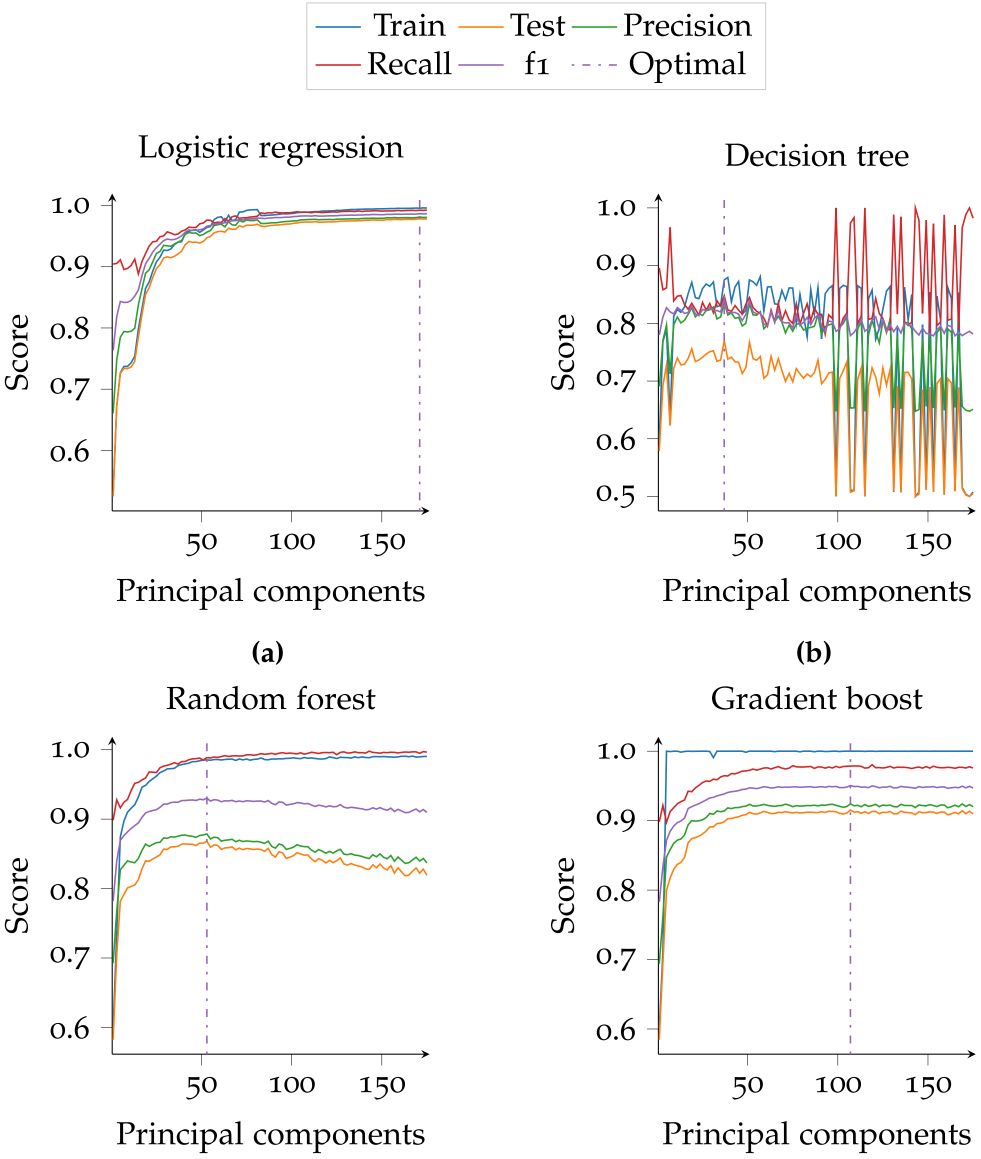
When searching for the optimal number of principal components, we iterated over every odd number of principal components from 1 to the upper restricted number which defines an accumulated variance of 95% from the principal component analysis. Due to the large number of principal components, we end up fitting 25 folds for each of 1232 parameter combinations, totalling up to 30800 individual models, just for logistic regression for one approach. This serves as an additional motivator to keep the models simple, and accordingly shows how easy an initial complex step might evolve into an unfeasible amount of information. Therefore, we will not make an extensive analysis for every model, but emphasis important distinctions between the general models and provide background for principal choices made. However, it should be noted that a larger automated analysis is distributed through the MIT license at the Github repository *predicting-solid-state-qubit-candidates* [125].

7.2.1 The Ferrenti approach

We visualize the grid search for the optimal number of principal components in figure 7.4, where we present the mean accuracy on the training set, and the balanced accuracy, precision, recall and f1 score on the test set as a function of principal components used in the models. For each principal component, we visualize the optimal combination of hyperparameters based on the f1-score in the model. Common to all models is the improvement of scores up to around 50 principal components, where random forest and the decision tree slowly starts to overfit for larger values. For decision trees, we observe a large fluctuation for principal components larger than 100. The f1-score is not varying as much as the other metrics due to an increasing number of positive predictions. This means that the accuracy of positive predictions are dominating the overall accuracy measurement, and we would expect a large amount of training data being predicted as positive candidates for those combinations. However, we see that the fluctuations are smaller in size for the optimal number of principal components. Similar to the decision tree is the random forest model, which also show sign of overfitting for larger values of principal components. The recall score is unaltered for increasing principal components, but consequently we find the precision declining due to a large amount of predicted false positives. However, as a result of an ensemble of decision trees, it show smaller signs of overfitting than the indications seen by the decision tree algorithm.

Gradient boost, on the other hand, experience minor changes for larger number of principal components, where the optimal number of components marked could be 50 principal components less without any remarks to the models metrics. We find that by using only a few principal components will make it reach almost 100% training accuracy, but does not show any clear sign on overfitting. Similarly, logistic regression show signs of almost a perfect classifier, with high scores for all metrics.

In table 7.1, we find the precise measurements for each evaluation metric for the optimal number of principal components, which is visualized as dotted lines in figure 7.4. The relevant hyperparameter for logistic regression were the maximum iterations, which were set at 400, and the regulariation term, which was found optimal at 0.46, and max iterations at 400. For random forest and decision trees, we find the maximum depth of 7, while gradient boost was found to overfit for deeper depths, as visualized in figure 7.5 and thus we found an optimal compromise at 4. We find that the best performing model is logistic regression, but is dependent on a large amount of principal components. Random forest and gradient boost perform comparably, with and f1 score of 0.93 and 0.95, respectively. However, it seems that only logistic regression is able to improve for additional principal components after the first 100.



(a)

(b)

(c)

(d)

Figure 7.4: Four figures displaying hyperparameter search for the Ferrenti approach. The best estimator is visualized for all hyperparameters as a function of principal components during a grid search with a 5 × 5 stratified cross validation, and the dotted lines marks the optimal hyperparameter-combination. Train stands for normal training accuracy, while test is the balanced accuracy on the test set. Precision, recall and f1 scores are based on the test set. The number of principal components that explain the 95% accumulated variance is 144, while the optimal model is found using the f1 score.

Table 7.1: A table of the optimal number of principal components and the respective scores (standard deviation) for the ferrenti approach, as visualized in the dash-dotted line in figure 7.4.

Model	PC	Mean test	Mean precision	Mean recall	mean f1
LOG	171	0.98(0.012)	0.98(0.011)	0.99(0.007)	0.99(0.007)
DT	37	0.77(0.034)	0.84(0.034)	0.85(0.044)	0.84(0.022)
RF	53	0.87(0.027)	0.88(0.022)	0.98(0.010)	0.93(0.014)
GB	107	0.92(0.016)	0.92(0.015)	0.98(0.010)	0.95(0.009)

In figure 7.6, we visualize how the models interpret the principal components that are sorted in descending order by the explained variance, found through a 5/times5 stratified cross validation. We find that we need to involve 144 principal components to reach the 95% accumulated explained variance. We have visualized the first 25 since this captures the most important information, and we note that most of the important features are within the first five principal components.

For logistic regression, we have visualized the mean fitted coefficients and the standard variation in 7.6a. Large positive or negative coefficients can be considered increasingly important, where positive (negative) coefficients will contribute to make positive (negative) predictions. The three next figures, namely the decision tree, random forest and gradient boost we find the mean impurity based feature importance, along with the standard deviation. We observe that the single most important feature for all models is the fifth principal component. Interestingly, if we select the highest values in this eigenvector, we find that the corresponding features originates from the DFT bandgap of elemental solid among elements in the composition as calculated by OQMD.

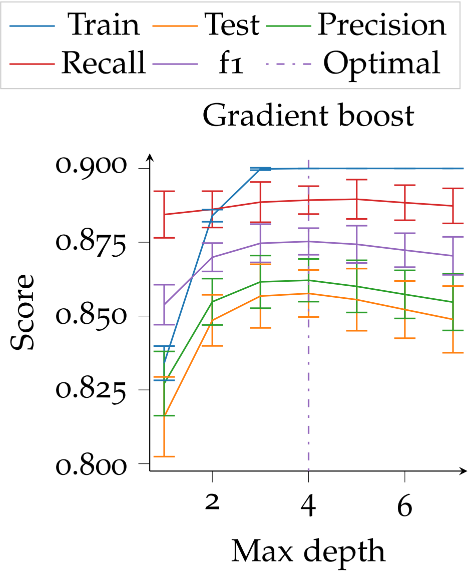
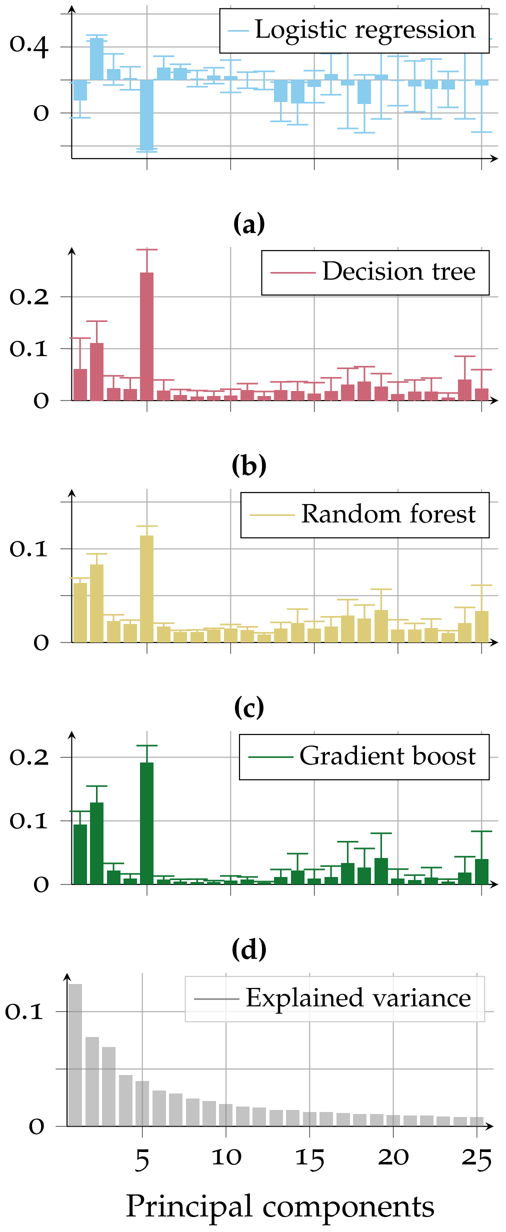


Figure 7.5: Parameter search for the Ferrenti approach regarding maximum depth for gradient boost for several metrics, where the error bars visualize the standard deviation.

After the first ten principal components, we see that the models adapt the other principal components with varying degree. Logistic regressions's coefficients experience large fluctuations, but the three remaining models find the first and second principal component important. In order of importance, we observe that the second component's largest values corresponds to the electronegativity, ionic property and covalence radius among the elements in the composition. The aggregations are either calculated as minimum, mean, standard deviation or maximum. While the first principal component has by far the largest explained variance, it does not provide any specific information of which features it represent. We observe that a variety of features is represented, such as the rows that a composition in the periodic table represents, structural packing efficiency and atomic weights of the components, but we are unable to confirm the prominent features due to small variations.

We note that looking at feature importance can be regarded as misleading for data involving correlated features, but we consider the analysis safe due to the projection of the original data to orthogonal vectors, known as principal components, which results in uncorrelated features.



(a)

(b)

(c)

(d)

(e)

Figure 7.6: Five figures visualizing different parameters for the (e) 25 most principal components ranked in descending order by the explained variance for the Ferrenti approach. The parameters are (a) the logistic regression coefficients, (b) decision tree feature importance, (c) random forest feature importance, (d) gradient boost feature importance and (e) explained variance that is retained by choosing each of the eigenvectors.

7.2.2 The augmented Ferrenti approach

For the augmented Ferrenti approach, we find the parameter grid search for principal components visualized in figure 7.7. All models experience an almost perfect recall score for 1 principal component due to the largely imbalanced dataset with 2141 good and 684 bad candidates, which is a ratio of 75 : 25%. This is a result due to the models being able to correctly label many good candidates compared to the amount of labelling them as bad candidates. On the other hand, we find a small precision for 1 component since the model predicts many materials, both actually labelled good and bad, as good candidates, and the latter case is in particularly large. This trend is revealed when looking at the balanced accuracy score. For all figures, it remains the lowest score of the evaluation metrics largely due to the inaccuracy of true negatives for the cross validations. Therefore, one can argue that we should use the balanced accuracy score for evaluation and not the f1 score, but the choice is independent on evaluation metric since the optimal f1 score is also the optimal balanced accuracy score for all figures.

Overall, the search for optimal hyperparameters in figure 7.7 for the augmented Ferrenti approach bear resemblance to the figure 7.4 for the Ferrenti approach. Logistic regression performs optimally for many principal component, and is the only model that continues to improve with an increasing number of components. The decision trees model experience a large fluctuation of scores, where the number of false positives is dominating the balanced accuracy score. Random forest experience less fluctuations compared to the decision tree as a consequence of the ensemble decision trees, while gradient boost does not improve after around 100 principal components.

Table 7.2: A table of the optimal number of principal components and the respective scores (standard deviation), as visualized in the dash-dotted line in figure 7.7.

Model	PC	Mean test	Mean precision	Mean recall	mean f1
LOG	175	0.98(0.008)	0.99(0.004)	0.99(0.004)	0.99(0.003)
DT	25	0.69(0.034)	0.86(0.015)	0.93(0.021)	0.90(0.008)
RF	25	0.70(0.028)	0.86(0.011)	1.00(0.003)	0.93(0.006)
GB	93	0.85(0.025)	0.93(0.011)	0.99(0.004)	0.96(0.007)

The optimal hyperparameters are summarized in table 7.2. We find that the logistic regression model with 175 principal components peform more or less as a perfect classifier with overall high scores. The decision tree and random forest models have similar balanced accuracy score with 0.69 and 0.70, respectively, due to challenges associated in predicting true negative

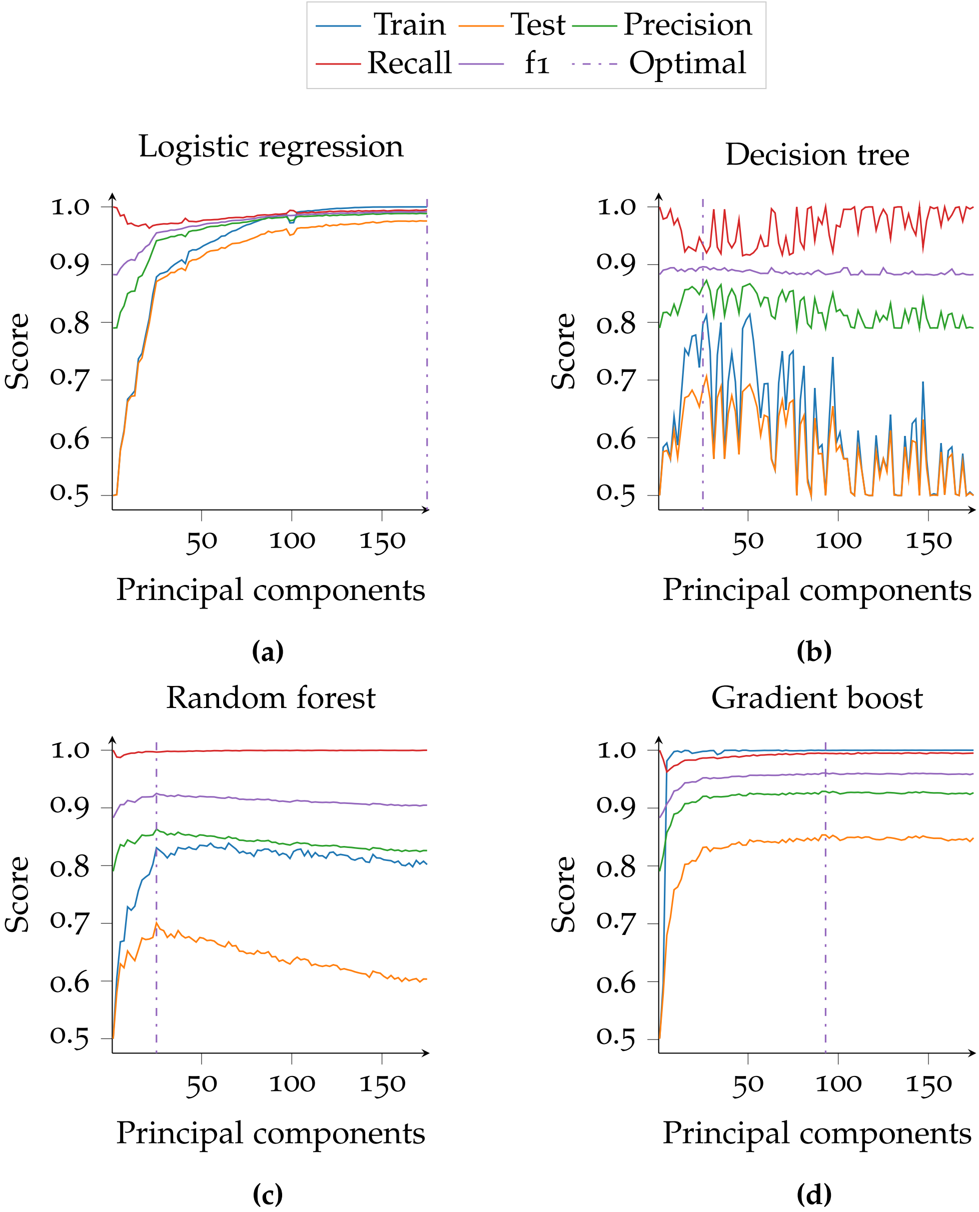


Figure 7.7: Four figures displaying hyperparameter search for the augmented Ferrenti approach. The best estimator is visualized for all hyperparameters as a function of principal components during a grid search with a 5×5 stratified cross validation, and the dotted lines marks the optimal hyperparameter-combination. Train stands for normal training accuracy, while test is the balanced accuracy on the test set. Precision, recall and f1 scores are based on the test set. The number of principal components that explain the 95% accumulated variance is 159, while the optimal model is found using the f1 score.

labels for 25 principal components. Lastly, we find gradient boost perform optimally at 93 principal components with a balanced accuracy score of 0.85.

The relevant hyperparameters of logistic regression were the regularization strength, which was set as 0.46, as visualized in figure 7.8, and we set maximum iterations at 400. Smaller regularization values resulted in worse scores, while increasing values did not noteworthy alter the results. The decision tree and random forest found an optimal maximum depth of 7, where smaller values resulted in low precision but high recall. Therefore, the choice was made to fasciliate a compromise between precision and recall. For gradient boost, we find the optimal maximum depth as 4 due to a decline in overall metrics for increasing depth except for training accuracy, which could potentially result in overfitting.

The interpretation of feature importance of for the Ferrenti approach is substantially more difficult than in the Ferrenti approach. We find for logistic regression and decision trees that no feature is different than any other in the cross validation due to a large variation of models. However, we find that random forest and gradient boost experience the fifth principal component as important. Similar to the Ferrenti approach, the corresponding features with highest value for the first principal component originates the DFT bandgap of elemental sold among elements in the composition.

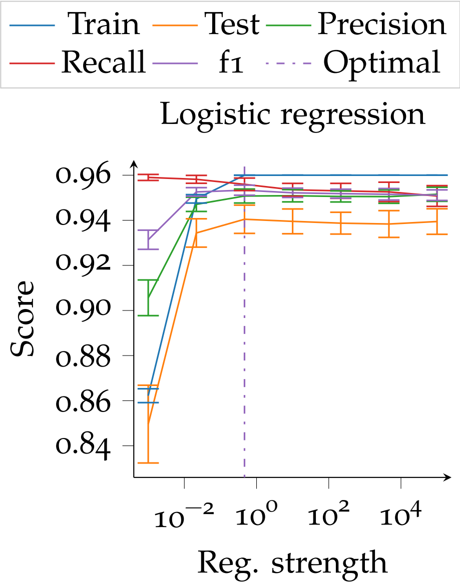


Figure 7.8: Parameter search for the augmented Ferrenti approach regarding regularization parameter for logistic regression for several metrics, where the error bars visualize the standard deviation.

7.2.3 The insightful approach

Lastly, we turn to the insightful approach, which involves 418 bad and 172 good candidates in the imbalanced training set. However, in contrast to the two other datasets, the majority of the entries are labelled as bad candidates.

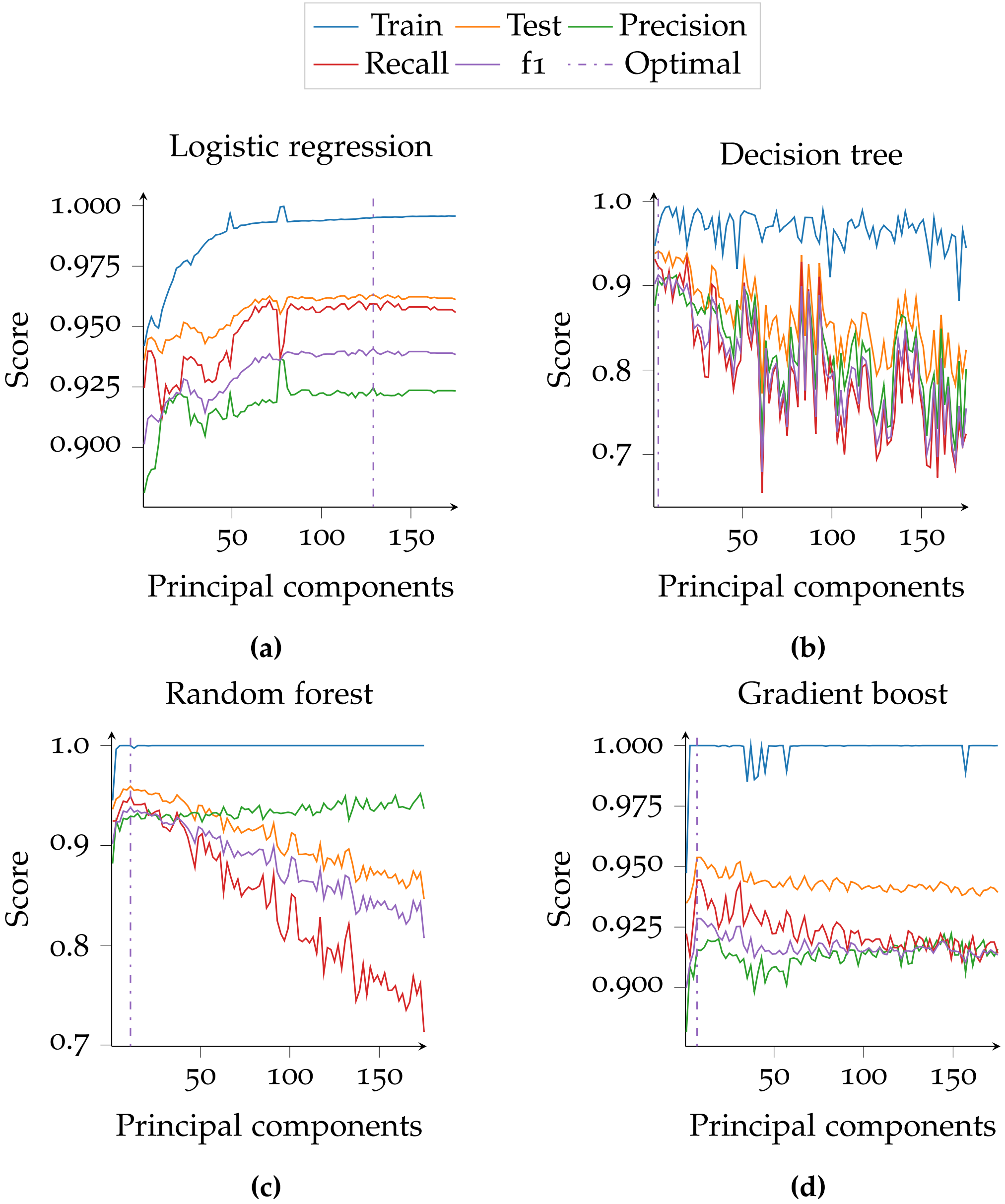
The grid search for the optimal number of principal components is visualized in figure 7.9. Interestingly, we find that all models experience high scores for just a few principal component, where 1 principal component earns at least 0.875 in score for all evaluation metrics. This information was also revealed for an earlier 2D-visualization of a scatter plot showing the two most important principal components in figure 5.4, and consequently can make the models find the optimal decision boundary more easily.

Logistic regression experience improvement of all scores for increasing number of principal components, yet only up 5% in scores compared to the 1D-representation of 1 principal components. Thus, one can argue if the increase in performance is worth it considering a one-dimensional representation with just a few percentage loss of performance. However, with multiple principal components, we find the largest increase in precision, which is a sign that the one-dimensional representation tend to wrongly predicts candidates as good when they are in fact bad. The decision tree and the random forest models exhibit best performance for just a few principal components, and experience considerable overfitting for larger values. Gradient boost, in contrast to the two other approaches, also experience best performance for a few principal components.

The optimal hyperparameters are summarized in table 7.3, where all models exhibit high evalution metrics. Importantly, we find the difference in number of principal component as most prominent, where logistic regression finds an optimum at 129 with the f1 score of 0.94. The decision tree model use only 3 principal components to achieve a f1 score of 0.91, while random forest needs 11 principal components to gain a f1 score of 0.94. Lastly, gradient boost performs optimally at 7 principal components with a mean f1 score of 0.93. The relevant hyperparameters was the regularization term for logistic regression, which was set as 0.021, and the maximum number of iterations as 200. The decision tree used an maximum depth of 4, where larger values increased the training accuracy but not any other metric. Random forest was set with maximum depth of 7, and gradient boost was given 4.

Table 7.3: A table of the optimal number of principal components and the respective scores (standard deviation), as visualized in the dash-dotted line in figure 7.9.

Model	PC	Mean test	Mean precision	Mean recall	mean f1
LOG	129	0.96(0.018)	0.93(0.041)	0.96(0.036)	0.94(0.025)
DT	3	0.94(0.025)	0.91(0.048)	0.92(0.050)	0.91(0.032)
RF	11	0.96(0.019)	0.93(0.039)	0.95(0.040)	0.94(0.024)
GB	7	0.95(0.023)	0.92(0.044)	0.94(0.047)	0.93(0.030)



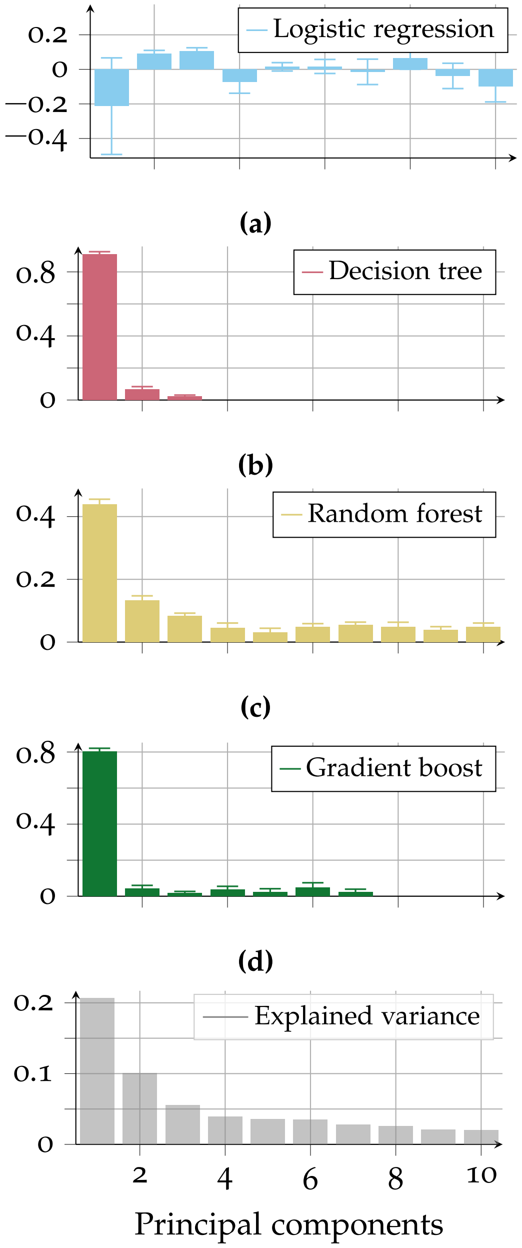
(a) **(b)** **(c)** **(d)**

Figure 7.9: Four figures displaying hyperparameter search for the insightful approach. The best estimator is visualized for all hyperparameters as a function of principal components during a grid search with a 5 × 5 stratified cross validation, and the dotted lines marks the optimal hyperparameter-combination. Train stands for normal training accuracy, while test is the balanced accuracy on the test set. Precision, recall and f1 scores are based on the test set. The number of principal components that explain the 95% accumulated variance is 103, while the optimal model is found using the f1 score.

The insightful approach differs in many aspects from the Ferrenti or augmented Ferrenti approach. Firstly, we find that the number of principal components neccessary to obtain 95% variance is reduced to 103 components, which is 41 and 56 less than the Ferrenti or augmented Ferrenti approach, respectively. Thus, the variance of the training set is found to be described with fewer principal components, indicating a simpler model.

Secondly, we find that the first principal component is by far the most important feature for all models, as visualized in figure 7.10. This is part of the reason why we experience a large accuracy for only a single feature, seen in figure 7.9. The first principal component's corresponding features are challenging to explain due to small variations of values. However, it differs when it comes which top features it describes, which includes bond orientational parameters, coordination numbers and radial distribution function of a compound's crystal system.

Thirdly, the insightful approach differs in how much explained variation is retained by the first component, which is 20% while it is 14% for the Ferrenti approach and 11% for the augmented Ferrenti approach. We find the difference striking considering the approaches share the same ultimate goal, but where the training set apparantely constitutes of large variations.



(a)

(b)

(c)

(d)

(e)

Figure 7.10: Five figures visualizing different parameters for the (e) 10 most principal components ranked in descending order by the explained variance for the insightful approach. The parameters are (a) the logistic regression coefficients, (b) decision tree feature importance, (c) random forest feature importance, (d) gradient boost feature importance and (e) explained variance that is retained by including each of the eigenvectors.

Chapter 8

Predictions

Using the four algorithms, optimized at each of the three approaches, and applying them to the case of predicting materials as good qubit material hosts or not yields 12 sets of results. In this chapter we present sets of representative results for each approach. Because of their length, we provide comprehensive tables of the machine learning classifications of the test sets and the training sets in Ref. [125].

8.1 The Ferrenti approach

We first consider the machine learning classification of the test set based on the Ferrenti approach.

Out of the known good candidates defined for the insightful approach, we find many of them in the Ferrenti training set. Carbon in diamond-like structures is present, but we also find two-dimensional carbon in graphite-like structure labelled as good. All structures of Si is defined as good candidates, together with one entry of SiC. Of other potentially good entries, we find ZnS, ZnSe, ZnO and ZnTe present.

The number of predicted candidates are labelled in figure 8.1. Logistic regression finds a total of 12380 good candidates, while decision tree is the most conservative with 11315. Random forest has the most optimistic estimate with 14278, while gradient boost finds 11835 good candidates. The models seems to agree on 6804 good candidates, however, many of the materials are predicted with the probability of similar proportions to a coin-flip. This is examplified if we were to raise the minimum bar of a prediction to 0.7, which would make the models only agree on 3000 good candidates. We have included a histogram displaying the distribution of probabilites on the test set in figure 8.2. In particular, we find that almost all of random forest's predictions are based on a large uncertainty. This behaviour is explained by the nature of random forest, since random forest base the predictions on an

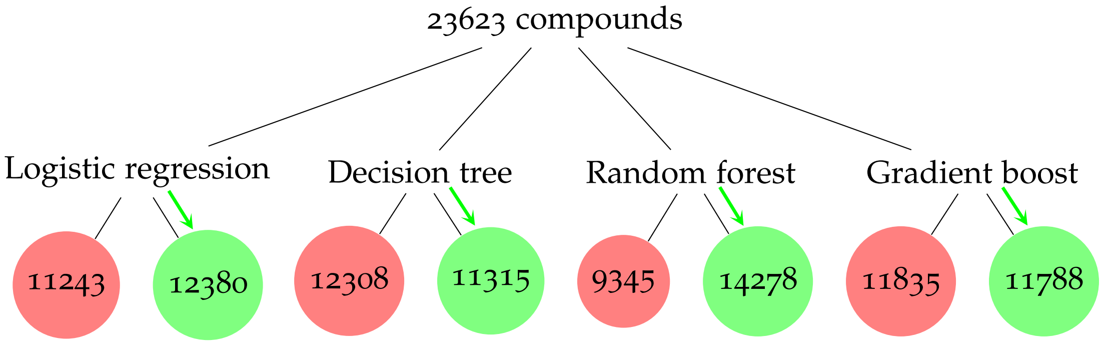


Figure 8.1: A figure visualizing the predictions of potential qubit candidates for all models for the Ferrenti approach. The green nodes display the number of predicted good candidates, while the bad candidates are marked red.

average of predictions in the ensemble of trees. Variance in the underlying trees will bias predictions close to either 0 or 1 [144]. Thus, all trees needs to agree on either one or zero for a resulting probability close to either one or zero.

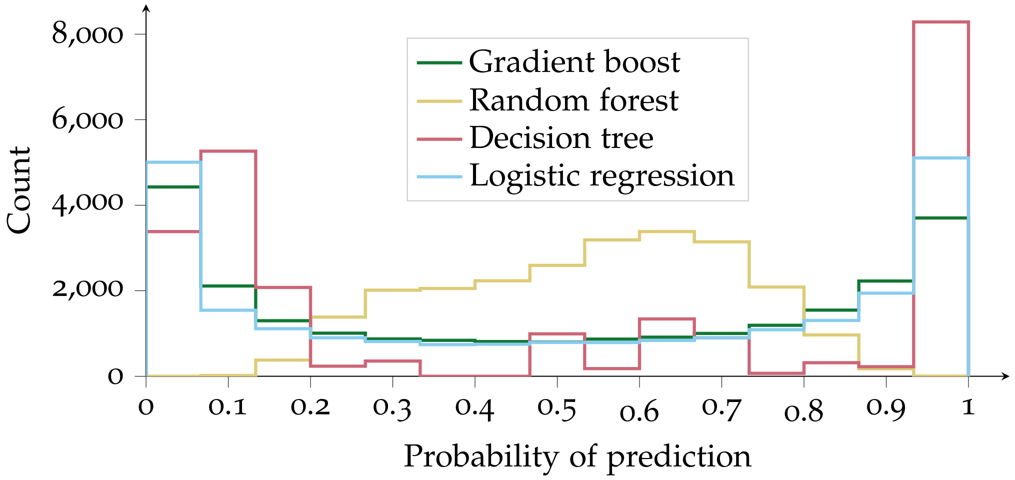


Figure 8.2: A histogram displaying the distribution of probabilites for all models based on the Ferrenti approach. If the probability is higher (lower) than 0.5, we label the material as a good (bad) candidate.

For the known materials that were present in the test set, we find that all models admit almost all materials with a chemical formula matching the known candidates. The one exception is the decision tree model, which labels C60 Fullerene, cubic Si, cubic GaAs, AlP, GaP, AlaAs and ZnTe as bad candidates. This is unfortunate, since this allows materials with unfortunate structures to be labelled as a good candidate by all models. Consequently, the models does not recognize the strict band gap restriction which makes it challenging to fascilitate deep defects. This is visualized in the parallel coordinate plot in figure 8.3, where the probability for being labelled a good candidate for 250 random entries with band gap less than 5eV is displayed. Ideally, we would expect that the models would have probabilities lower than 0.5 for all models when the band gap is lower than 0.5eV, which would be expected behaviour based on the training set, but this is not the case. We find that many entries with band gap lower than 0.5, marked as strong red lines in the parallel histogram, are present as both good and bad candidates for all models. Therefore, it seems the algorithms have found a trend in the data set that does not neccessarily favor deep defects.

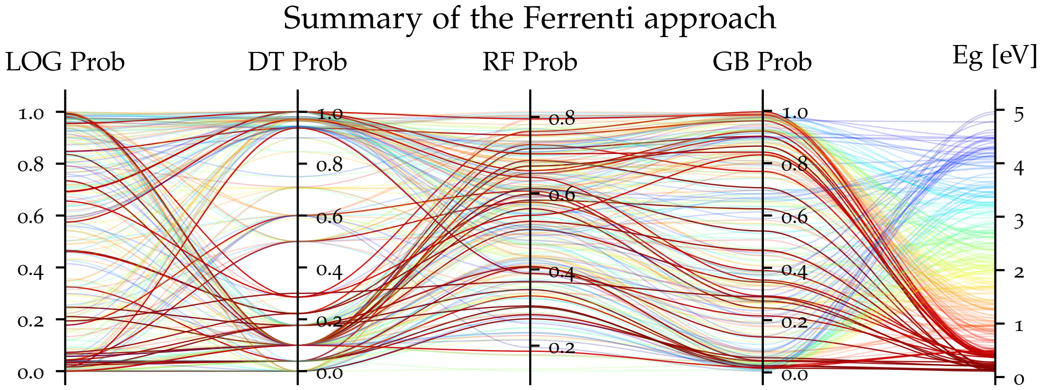


Figure 8.3: A parallel coordinate plot of 250 random entries in the test set with MP-calculated band gap less than 5eV. The figure is based on the Ferrenti approach.

8.2 The augmented Ferrenti approach

Then we turn towards the perhaps more liberal augmented Ferrenti approach with the result visualized in figure 8.4, where we find the most predicted candidates with 14993, 14407, 15351 and 13788 for logistic regression, decision tree, random forest and gradient boost, respectively. The probability distribution of the predictions are visualized in figure 8.5. Three of the models, that is gradient boost, decision tree and logistic regression, are very confident in their labelling of good candidates and base their predictions on close to 100% probability. Random forest, on the other hand, experience the same variance as in the Ferrenti approach. We observe a peak between 0.75 and 0.8, indicating a larger number of positive predictions. Due to the easier restrictions compared to the Ferrenti approach, we find the large amount of 9227 entries that the four models agree on.

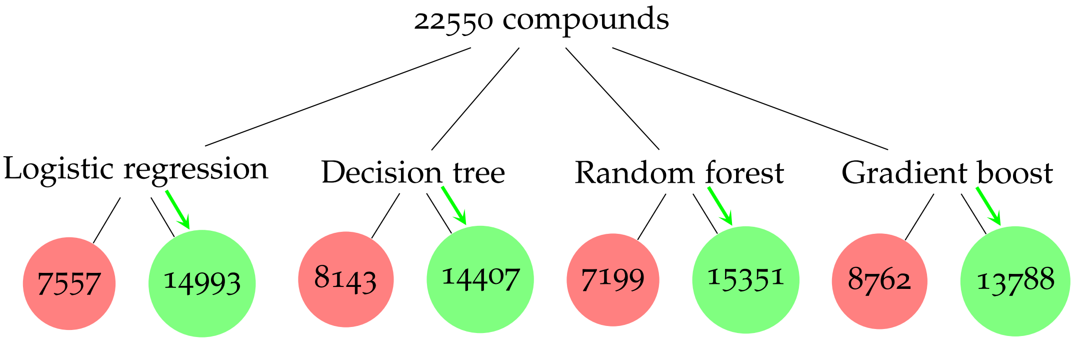


Figure 8.4: A figure visualizing the predictions of potential qubit candidates for all models for the augmented Ferrenti approach. The green nodes display the number of predicted good candidates, while the bad candidates are marked red.

In the training set, we find a single entry of SiC, Si, GaN, ZnS, GaP, AlAs and AlP, carbon in both diamond- and graphite-like structure andAlN in three different structures. Interestingly, a larger variety of the known candidates are present compared to the Ferrenti approach, but due to the larger band gap restriction we find fewer of each known chemical formula present in the training set.

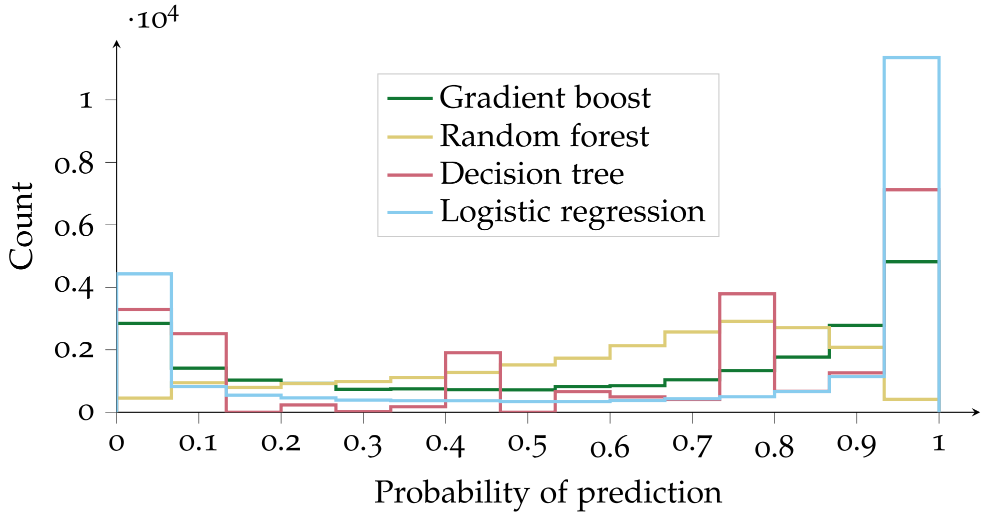


Figure 8.5: A histogram displaying the distribution of probabilites for all models based on the augmented Ferrenti approach. If the probability is higher (lower) than 0.5, we label the material as a good (bad) candidate.

The summary of the test set reveals that all of the unlabelled known good candidates are, in fact, predicted as good candidates. Logistic regression predicts a single exception, as it labels almost all structures present of ZnTe as bad candidates. Unfortunately, due to the large number of good candidates, it also reveals unqualified predictions. All models confidently predict NaCl as a good candidate, which is in fact bad due to the phonon-interactions within the lattice that would substantially increase decoherence. Additionally, we find that this approach also predicts materials with band gap lower than 0.5eV as good candidates.

8.3 The insightful approach

Finally, we turn to the insightful approach, with the results displayed in figure 8.6. The four models predicts radically fewer good candidates compared to the two latter approaches, where only 493, 442, 321 and 629 materials are predicted good by logistic regression, decision tree, random forest and gradient boost, respectively. The large majority of the bad candidates are predicted with large probability by all models. However, we find also in this approach the presence of good candidates with band gap lower than 0.5eV. All the models agree on only 105 good candidates, which reduces to 85 by imposing the bandgap restriction.

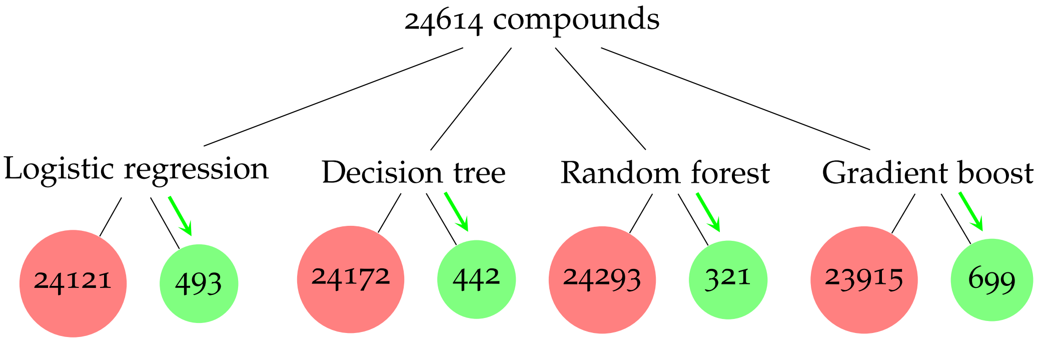


Figure 8.6: A figure visualizing the predictions of potential qubit candidates for all models for the insightful approach. The green nodes display the number of predicted good candidates, while the bad candidates are marked red.

Initially, we begin with looking at all materials that are predicted by all models with more than 80% probability, which are $ZnGeP_2$, He, BC_2N, N_2 and RuC. None of these entries have a band gap lower than 0.5eV, but we see that the list includes noble gases. The noble gases are described in the data with no ionic character, no electronegativity, low covalent radius, large band gaps and simple structures. Furthermore, they are missing entries on most of the descriptors and we do not have a feature describing any physics of noble gases. We therefore believe the noble gases can be regarded as outliers, and are therefore not offered additional consideration.

$ZnGeP_2$ (mp-4524) is a tetrahedrally coordinated material, chalcopyrite-like structure, with reported MP calculated indirect band gap of 1.2 [145] and experimentally reported as 1.99eV[146]. It crystallize in a non-polar space group, possess no magnetic moment, have strong covalent bondings and has been reported as an excellent mid-IR transparent crystal material which is suitable for nonlinear optical applications [145]. Importantly, it is possible to integrate sources of photon quantum states based on nonlinear optics [147]. An eligible candidate indeed, but it remains unknown if the candidate can provide isolation and shelter to experimentally fascilitate a deep defect with quantum effects.

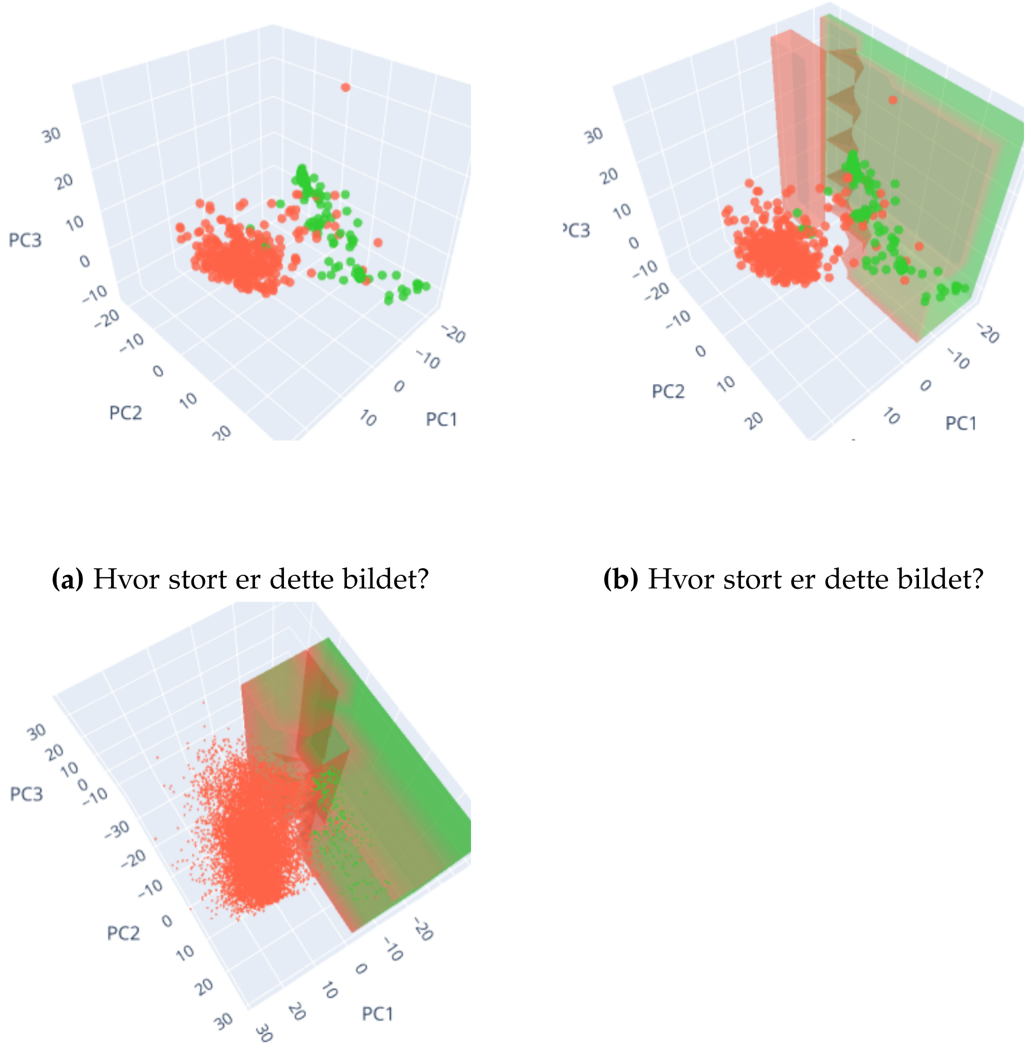
We also find two compositions with the same chemical formula, the orthorombic coordinated (mp-629458) and the chalcopyrite-like structured BC_2N (mp-1008523). The first structure is in a polar space group while the latter is not. The band gaps are in MP calculated as 1.85eV and 1.65eV, respectively. BC_2N is known as heterodiamond and is a super hard hybrid of diamond and BN. Both structures have, as expected, strong covalent character and have been studied for application as nanostructures [148], hydrogen storage [149] and superhard materials [150, 151] in ab-initio calculations. The structures are still in early development, but might show promising host qualities for use in quantum technology.

Lastly, we find RuC (mp-1009792) in the rock-salt cubic structure as a predicted candidate. It is a relatively new and unstudied composition, which is found unstable in terms of energy above hull / atom in MP, and have an MP calculate indirect band gap of 0.72eV [152].

Random forest, due to the average of trees, experience a smaller probability than the other models. If we leave the model out, we gain a list of 65 predicted good candidates for the three other models with probability of at least 80%. A few noteworthy compounds that are not yet mentioned include Ge, GeC, BP and InP. Ge in cubic structure (mp-1198022) share many similar properties with Si and C as well as sharing periodic column number. In fact, the first transistors was made in germanium to its appealing eletrical properties, but silicon took over as the material of choice for microelectronics due to the outstanding quality of silicone dioxide, which allowed the fabrication and integration of increasingly smaller transistors [153, 154]. Ge has the highest hole mobility of semiconductors at room temperature, and is therefore considered a key material when in the process of extending the chip performance in classical computers beyond the limits imposed by miniaturization [153].

-Write about GeC, BP, LnP.

We acknowledge that many compositions deemed as good candidates consists of either rare or dangerous elements. By utilizing an enormously large database as Materials Project, we have to account for their ultimate goal - to model all possible materials and their properties. Thus, unphysical descriptors needs to be manually applied during the postpartum analysis.



(a) Hvor stort er dette bildet?

(b) Hvor stort er dette bildet?

(c) Hvor stort er dette bildet?

8.4 Comparison of the approaches

Out of the three approaches, we find that the augmented approach is the least restricted approach and admits the most entries. The Ferrenti approach also admits a large amount of entries, and is considered to not be very different from the Augmented Ferrenti approach. The models in the two approaches are unable to reproduce the criteria that the approaches are based on, such as band gap restriction or polar space group. Of course, the materials that the two initial approaches label as good candidates are challenging to go through due to their extensive lengths, whereas the insightful approach predicts fewer good candidates and we are able to manually verify many of the compounds. However, we note that we found predicted good candidates with band gap lower than 0.5eV for the insightful approach as well, but to a smaller extent.

Of the 85 predicted good candidates with band gap lower than 0.5eV that the models in the insightful approach agreed upon, we find 54 of them also predicted as good by all models in the augmented Ferrenti approach. Similarly, 35 of them are also predicted as good by all the models in the Ferrenti approach.

The constructed dataset consists of compounds formed by all possible combinations of surfaces, interfaces, nanostructures, compositions and structures. We note that this complexity is not neccessarily reflected in the descriptors.

Part V

Concluding remarks

Chapter 9

Conclusion

In this present work, we performed an exploratory analysis for identifying new potential qubit material hosts candidates using machine learning.

In the process of becoming acquainted with the databases, we have developed tools for simple data extraction and processing for six high-throughput databases, including AFLOWlib, AFLOW-ML, Citrination, Materials Project, OQMD and JARVIS-DFT. We utilized the high-throughput code and tools of Matminer to extend a featurization process done by MODnet. Due to a small amount of similar entries in the databases, we apply the featurization procedure to a subsample of 25.000 materials in the Material Project.

Thereafter, we developed and implemented three approaches to define good and bad candidates, namely the Ferrenti approach, the augmented Ferrenti approach and the insightful approach. For each of the approaches, we applied the dimensionality reduction technique principal component and trained the machine learning algorithms logistic regression, decision tree, random forest and gradient boost. We find the Ferrenti approach and the augmented Ferrenti approach not being able to correctly predict properties that favour materials that can fascilitate any quantum effects, since the machine learning trends reveals unconsistent results of both good and bad candidates. We credit this result to the general and inaccurate criteria for the two approaches, in addition to the absent of any features describing potential quantum effets. However, the insightful approach delivers more consistent candidates and provides promising materials such as $ZnGeP_2$, Ge, GeC, BP, LnP, and many more after a manual screening in the post analysis.

Chapter 10

Future remarks

Future remarks Due to an explosive interest in the field of material informatics.

- future remarks – new features Matminer – better starting materials (e.g. all with included bandstructure/dos)

Part VI

Appendices

Appendix A

Extracting information from ab-initio calculations

A.1 The Born-Oppenheimer approximation

The many-particle eigenfunction describes the wavefunction of all the electrons and nuclei and we denote it as Ψ_κ^{en} for electrons (e) and nuclei (n), respectively. The Born-oppenheimer approximation assumes that nuclei, of substantially larger mass than electrons, can be treated as fixed point charges. According to this assumption, we can separate the eigenfunction into an electronic part and a nuclear part,

$$\Psi_\kappa^{en}(\mathbf{r},\mathbf{R}) \approx \Psi_\kappa(\mathbf{r},\mathbf{R})\Theta_\kappa(\mathbf{R}), \tag{A.1}$$

where the electronic part is dependent on the nuclei. This is in accordance with the assumption above, since electrons can respond instantaneously to a new position of the much slower nucleus, but this is not true for the opposite scenario. To our advantage, we already have knowledge of the terms in the many-particle Hamiltionian, and we can begin by separating the Hamiltionian into electronic and nuclear parts:

$$\hat{H}^{en} = \overbrace{T_e + U_{ee} + U_{en}}^{\hat{H}^e} + \overbrace{T_n + U_{nn}}^{\hat{H}^n}. \tag{A.2}$$

Starting from the Schrödinger equation, we can formulate separate expressions for the electronic and the nuclear Schrödinger equations.

$$H\hat{en}\Psi_{\kappa}^{en}(\mathbf{r},\mathbf{R}) = E_{\kappa}^{en}\Psi_{\kappa}^{en}(\mathbf{r},\mathbf{R}) \quad |\times \int \Psi^{*}(\mathbf{r},\mathbf{R})\,d\mathbf{r} \tag{A.3}$$

$$\int \Psi_{\kappa}^{*}(\mathbf{r},\mathbf{R})(\hat{H}^{e}+\hat{H}^{n})\Psi_{\kappa}(\mathbf{r},\mathbf{R})\Theta_{\kappa}(\mathbf{R})\,d\mathbf{r} = E_{\kappa}^{en}\underbrace{\int \Psi_{\kappa}^{*}(\mathbf{r},\mathbf{R})\Psi_{\kappa}(\mathbf{r},\mathbf{R})\,d\mathbf{r}}_{1}\,\Theta_{\kappa}(\mathbf{R}). \tag{A.4}$$

Since $\Theta_{\kappa}(\mathbf{R})$ is independent of the the spatial coordinates to electrons, we get E_{κ} as the total energy of the electrons in the state κ.

$$E_{\kappa}(\mathbf{R})\Theta_{k}(\mathbf{R}) + \int \Psi_{k}^{*}(\mathbf{r},\mathbf{R})H^{n}\Psi_{k}(\mathbf{r},\mathbf{R})\Theta_{k}(\mathbf{R})\,d\mathbf{r} = E_{k}^{en}\Theta_{k}(\mathbf{R}). \tag{A.5}$$

Now, the final integration term can be simplified by using the product rule, which results in

$$\Big(T_{n} + T_{n}^{'} + T_{n}^{''} + U_{nn} + E_{\kappa}(\mathbf{R})\Big)\Theta_{\kappa}(\mathbf{R}) = E_{\kappa}^{en}\Theta_{\kappa}(\mathbf{R}). \tag{A.6}$$

If we neglect T_{n}' and T_{n}'' to lower the computational efforts, we obtain the Born-Oppenheimer approximation with the electronic eigenfunction as

$$(T_{e} + U_{ee} + U_{en})\,\Psi_{\kappa}(\mathbf{r},\mathbf{R}) = E_{\kappa}(\mathbf{R})\Psi_{\kappa}(\mathbf{r},\mathbf{R}) \tag{A.7}$$

and the nuclear eigenfunction as

$$\big(T_{n} + U_{nn} + E_{\kappa}(\mathbf{R})\big)\,\Theta_{\kappa}(\mathbf{R}) = E_{\kappa}^{en}(\mathbf{R})\Theta_{\kappa}(\mathbf{r},\mathbf{R}). \tag{A.8}$$

How are they coupled, you might ask? The total energy in the electronic equation is a potential in the nuclear equation.

A.2 The variational principle

So far, we have tried to make the time-independent Schrödinger equation easier with the use of an *ansatz*, but we do not neccessarily have an adequate guess for the eigenfunctions and the ansatz can only give a rough estimate in most scenarios. Another approach, namely the *variational principle*, states that the energy of any trial wavefunction is always an upper bound to the exact ground state energy by definition E_0.

$$E_0 = \langle\psi_0|\,H\,|\psi_0\rangle \leqslant \langle\psi|\,H\,|\psi\rangle = E \tag{A.9}$$

The eigenfunctions of H form a complete set, which means any normalized Ψ can be expressed in terms of the eigenstates

$$\Psi = \sum_n c_n \psi_n, \quad \text{where} \quad H\psi_n = E_n \psi_n \tag{A.10}$$

for all $n = 1, 2, ...$ The expectation value for the energy can be calculated as

$$\begin{aligned} \langle\Psi| H |\Psi\rangle &= \left\langle \sum_n c_n\psi_n \middle| H \middle| \sum_{n'} c_{n'}\psi_{n'} \right\rangle \\ &= \sum_n \sum_{n'} c_n^* c_{n'} \langle\psi_n| H |\psi_{n'}\rangle \\ &= \sum_n \sum_{n'} c_n^* E_n c_{n'} \langle\psi_n|\psi_{n'}\rangle \end{aligned}$$

Here we assume that the eigenfunctions have been orthonormalized and we can utilize $\langle\psi_m|\psi_n\rangle = \delta_{mn}$, resulting in

$$\sum_n c_n^* c_n E_n = \sum_n |c_n|^2 E_n.$$

We have already stated that Ψ is normalized, thus $\sum_n |c_n|^2 = 1$, and the expectation value conveniently is bound to follow equation A.9. The quest to understand the variational principle can be summarized in a sentence - it is possible to tweak the wavefunction parameters to minimize the energy, or summed up in a mathematical phrase,

$$E_0 = \min_{\Psi\rightarrow\Psi_0} \langle\Psi| H |\Psi\rangle . \tag{A.11}$$

A.3 The Hohenberg-Kohn theorems

A.3.1 The Hohenberg-Kohn theorem 1

Proof. Assume that two external potentials $V_{ext}^{(1)}$ and $V_{ext}^{(2)}$, that differ by more than a constant, have the same ground state density $n_0(r)$. The two different potentials correspond to distinct Hamiltonians $\hat{H}_{ext}^{(1)}$ and $\hat{H}_{ext}^{(2)}$, which again give rise to distinct wavefunctions $\Psi_{ext}^{(1)}$ and $\Psi_{ext}^{(2)}$. Utilizing the variational principle, we find that no wavefunction can give an energy that is less than the energy of $\Psi_{ext}^{(1)}$ for $\hat{H}_{ext}^{(1)}$, that is

$$E^{(1)} = \left\langle \Psi^{(1)} \middle| \hat{H}^{(1)} \middle| \Psi^{(1)} \right\rangle < \left\langle \Psi^{(2)} \middle| \hat{H}^{(1)} \middle| \Psi^{(2)} \right\rangle \tag{A.12}$$

and

$$E^{(2)} = \left\langle \Psi^{(2)} \middle| \hat{H}^{(2)} \middle| \Psi^{(2)} \right\rangle < \left\langle \Psi^{(1)} \middle| \hat{H}^{(2)} \middle| \Psi^{(1)} \right\rangle . \tag{A.13}$$

Assuming that the ground state is not degenerate, the inequality strictly holds. Since we have identical ground state densities for the two Hamiltonian's, we can rewrite the expectation value for equation A.12 as

$$\begin{aligned}
E^{(1)} &= \left\langle \Psi^{(1)} \middle| \hat{H}^{(1)} \middle| \Psi^{(1)} \right\rangle \\
&= \left\langle \Psi^{(1)} \middle| T + U_{ee} + U^{(1)}_{ext} \middle| \Psi^{(1)} \right\rangle \\
&= \left\langle \Psi^{(1)} \middle| T + U_{ee} \middle| \Psi^{(1)} \right\rangle + \int \Psi^{*(1)}(\mathbf{r}) V^{(1)}_{ext} \Psi^{(1)}(\mathbf{r}) d\mathbf{r} \\
&= \left\langle \Psi^{(1)} \middle| T + U_{ee} \middle| \Psi^{(1)} \right\rangle + \int V^{(1)}_{ext} n(\mathbf{r}) d\mathbf{r} \\
&< \left\langle \Psi^{(2)} \middle| \hat{H}^{(1)} \middle| \Psi^{(2)} \right\rangle \\
&= \left\langle \Psi^{(2)} \middle| T + U_{ee} + U^{(1)}_{ext} + \overbrace{U^{(2)}_{ext} - U^{(2)}_{ext}}^{0} \middle| \Psi^{(2)} \right\rangle \\
&= \left\langle \Psi^{(2)} \middle| T + U_{ee} + U^{(2)}_{ext} \middle| \Psi^{(1)} \right\rangle + \int \left(V^{(1)}_{ext} - V^{(2)}_{ext} \right) n(\mathbf{r}) d\mathbf{r} \\
&= E^{(2)} + \int \left(V^{(1)}_{ext} - V^{(2)}_{ext} \right) n(\mathbf{r}) d\mathbf{r}.
\end{aligned}$$

Thus,

$$E^{(1)} = E^{(2)} + \int \left(V^{(1)}_{ext} - V^{(2)}_{ext} \right) n(\mathbf{r}) d\mathbf{r} \tag{A.14}$$

A similar procedure can be performed for $E^{(2)}$ in equation A.13, resulting in

$$E^{(2)} = E^{(1)} + \int \left(V^{(2)}_{ext} - V^{(1)}_{ext} \right) n(\mathbf{r}) d\mathbf{r}. \tag{A.15}$$

If we add these two equations together, we get

$$\begin{aligned}
E^{(1)} + E^{(2)} &< E^{(2)} + E^{(1)} + \int \left(V^{(1)}_{ext} - V^{(2)}_{ext} n(\mathbf{r}) d\mathbf{r} \right) \\
&\qquad + \int \left(V^{(2)}_{ext} - V^{(1)}_{ext} n(\mathbf{r}) d\mathbf{r} \right) \\
E^{(1)} + E^{(2)} &< E^{(2)} + E^{(1)},
\end{aligned} \tag{A.16}$$

which is a contradiction. Thus, the two external potentials cannot have the same ground-state density, and $V_{ext}(\mathbf{r})$ is determined uniquely (except for a constant) by $n(\mathbf{r})$. □

A.3.2 The Hohenberg-Kohn theorem 2

Proof. Since the external potential is uniquely determined by the density and since the potential in turn uniquely determines the ground state wavefunction (except in degenerate situations), all the other observables of the system are uniquely determined. Then the energy can be expressed as a functional of the density.

$$E[n] = \overbrace{T[n] + U_{ee}[n]}^{F[n]} + \overbrace{U_{en}[n]}^{\int V_{en} n(r) dr} \tag{A.17}$$

where $F[n]$ is a universal functional because the treatment of the kinetic and internal potential energies are the same for all systems, however, it is most commonly known as the Hohenberg-Kohn functional.

In the ground state, the energy is defined by the unique ground-state density $n_0(r)$,

$$E_0 = E[n_0] = \langle \Psi_0 | H | \Psi_0 \rangle . \tag{A.18}$$

From the variational principle, a different density $n(r)$ will give a higher energy

$$E_0 = E[n_0] = \langle \Psi_0 | H | \Psi_0 \rangle < \langle \Psi | H | \Psi \rangle = E[n] \tag{A.19}$$

Thus, the total energy is minimized for n_0, and so has to be the ground-state energy. □

A.4 Self-consistent field methods

So, the remaining question is, how do we solve the Kohn-Sham equation? First, we would need to define the Hartree potential, which can be found if we know the electron density. The electron density can be found from the single-electron wave-functions, however, these can only be found from solving the Kohn-Sham equation. This *circle of life* has to start somewhere, but where? The process can be defined as an iterative method, *a computational scheme*, as visualized in figure A.1.

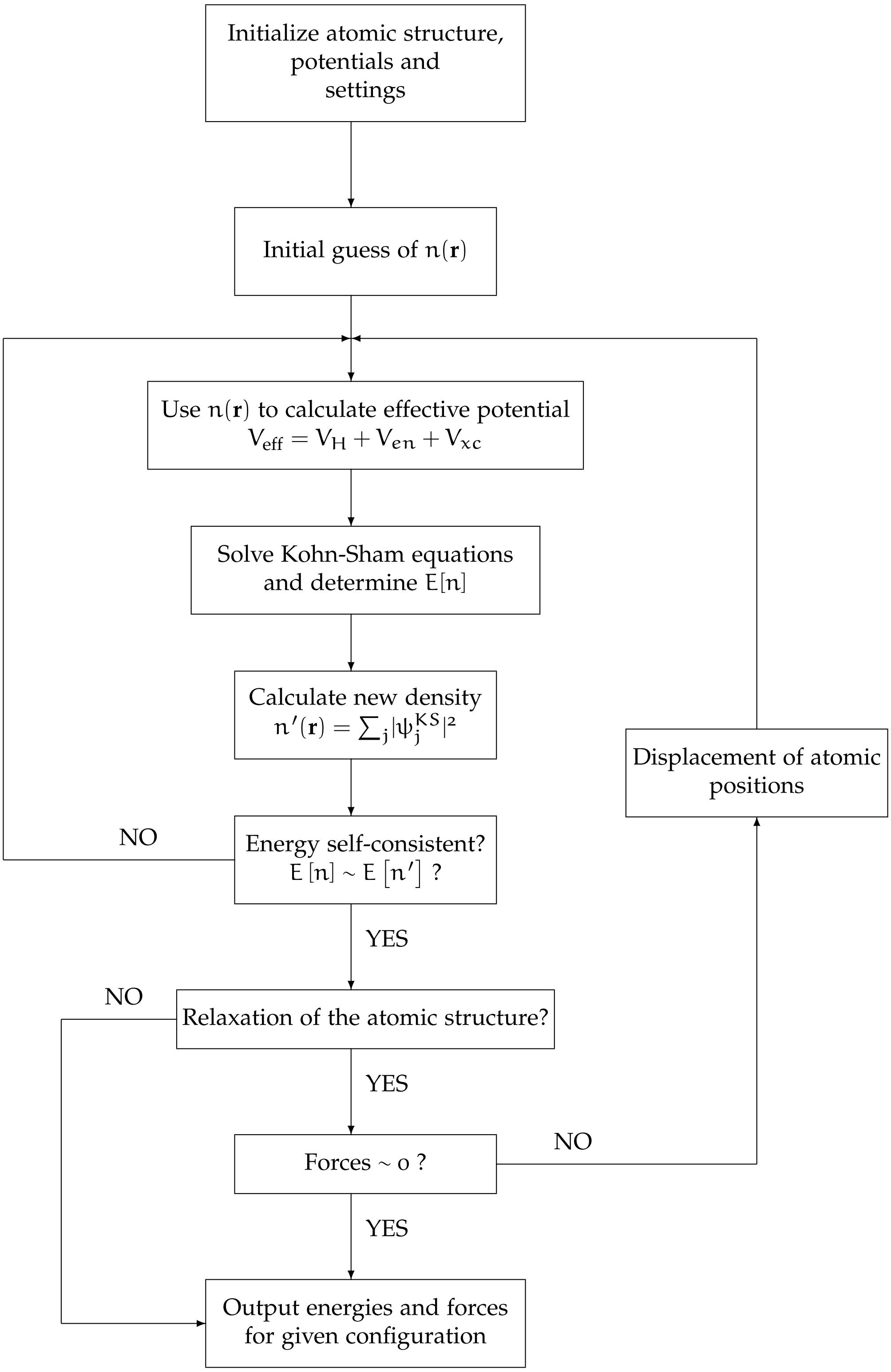

Initialize atomic structure, potentials and settings
Initial guess of $n(\mathbf{r})$
Use $n(\mathbf{r})$ to calculate effective potential $V_{\text{eff}} = V_H + V_{en} + V_{xc}$
Solve Kohn-Sham equations and determine $E[n]$
Calculate new density $n'(\mathbf{r}) = \sum_j |\psi_j^{KS}|^2$
Energy self-consistent? $E[n] \sim E[n']$?
NO
YES
Relaxation of the atomic structure?
NO
YES
Forces ~ 0 ?
NO
YES
Displacement of atomic positions
Output energies and forces for given configuration

Figure A.1: A flow chart of the self-consistent field method for DFT.

Appendix B

Featurizaton

B.1 Table of featurizers

Table B.1: This thesis' chosen 39 featurizers from matminer. Descriptions are either found from Ref. [115] or from the project's Github page.

Features	Description	Original reference
Composition features		
AtomicOrbitals	Highest occupied molecular orbital (HOMO) and lowest unoccupied molecular orbital (LUMO).	[155]
AtomicPacking-Efficiency	Packing efficiency.	[156]
BandCenter	Estimation of absolute position of band center using geometric mean of electronegativity.	[157]
ElementFraction	Fraction of each element in a composition.	-
ElementProperty	Statistics of various element properties.	[114, 158, 159]
IonProperty	Maximum and average ionic character.	[158]

Continued on next page

Table B.1 – continued from previous page

Features	Description	Original reference
Miedema	Formation enthalpies of intermetallic compounds, solid solutions, and amorphous phases using semi-empirical Miedema model.	[160]
Stoichiometry	L^p norm-based stoichiometric attributes.	[158]
TMetalFraction	Fraction of magnetic transition metals.	[159]
ValenceOrbital	Valence orbital attributes such as the mean number of electrons in each shell.	[158]
YangSolidSolution	Mixing thermochemistry and size mismatch terms.	[161]
Oxid composition features		
ElectronegativityDiff	Statistics on electronegativity difference between anions and cations.	[159]
OxidationStates	Statistics of oxidation states.	[159]
Structure features		
DensityFeatures	Calculate density, volume per atom and packing fraction.	-
GlobalSymmetryFeatures	Determines spacegroup number, crystal system (1-7) and inversion symmetry.	-
RadialDistributionFunction	Calculates the radial distribution function of a crystal system.	-
CoulombMatrix	Generate the Coulomb matrix, which is a representation of the nuclear coulombic interaction of the input structure.	[162]
PartialRadialDistributionFunction	Compute the partial radial distribution function of a crystal structure	[163]

Continued on next page

Table B.1 – continued from previous page

Features	Description	Original reference
SineCoulomb-Matrix	Computes a variant of the coulomb matrix developed for periodic crystals.	[164]
EwaldEnergy	Computes the energy from Coulombic interactions based on charge states of each site.	[165]
BondFractions	Compute the fraction of each bond in a structure, based on nearest neighbours.	[166]
Structural-Heterogeneity	Calculates the variance in bond lengths and atomic volumes in a structure.	[167]
MaximumPacking-Efficiency	Calculates the maximum packing efficiency of a structure.	[167]
ChemicalOrdering	Computes how much the ordering of species differs from random in a structure.	[167]
XRDPowder-Pattern	1D array representing normalized powder diffraction of a structure as calculated by pymatgen.	[114]
Site features		
AGNI-Fingerprints	Calculates the product integral of RDF and Gaussian window function	[168]
AverageBond-Angle	Determines the average bond angle of a specific site with its nearest neighbors using pymatgens implementation.	[169]
AverageBond-Length	Determines the average bond length between one specific site and all its nearest neighbors using pymatgens implementation.	[169]
BondOrientational-Paramater	Calculates the averages of spherical harmonics of local neighbors	[170, 171]

Continued on next page

Table B.1 – continued from previous page

Features	Description	Original reference
ChemEnvSite Fingerprint	Calculates the resemblance of given sites to ideal environment using pymatgens ChemEnv package.	[172, 173]
Coordination-Number	The number of first nearest neighbors of a site	[173]
CrystalNN-Fingerprint	A local order parameter fingerprint for periodic crystals.	-
GaussianSymm-Func	Calculates the gaussian radial and angular symmetry functions originally suggested for fitting machine learning potentials.	[174, 175]
GeneralizedRadial-Distribution-Function	Computes the general radial distribution function for a site	[170]
LocalProperty-Difference	Computes the difference in elemental properties between a site and its neighboring sites.	[167, 169]
OPSite-Fingerprint	Computes the local structure order parameters from a site's neighbor environment.	[173]
Voronoi-Fingerprint	Calculates the Voronoi tessellation-based features around a target site.	[176, 177]
Density of state features		
DOSFeaturizer	Computes top contributors to the density of states at the valence and conduction band edges. Thus includes chemical specie, orbital character, and orbital location information.	[178]
Band structure features		

Continued on next page

Table B.1 – continued from previous page

Features	Description	Original reference
BandFeaturizer	Converts a complex electronic band structure into quantities such as band gap and the norm of k point coordinates at which the conduction band minimum and valence band maximum occur.	-

B.2 Erroneous entries

MPID	Full formula	Reference
mp-555563	$PH_6C_2S_2NCl_2O_4$	[179]
mp-583476	$Nb_7S_2I_{19}$	[180]
mp-600205	$H_{10}C_5SeS_2N_3Cl$	-
mp-600217	$H_{80}C_{40}Se_8S_{16}Br_8N_{24}$	-
mp-1195290	$Ga_3Si_5P_{10}H_{36}C_{12}N_4Cl_{11}$	-
mp-1196358	$P_4H_{120}Pt_8C_{40}I_8N_4Cl_8$	-
mp-1196439	$Sn_8P_4H_{128}C_{44}N_{12}Cl_8O_4$	-
mp-1198652	$Te_4H_{72}C_{36}S_{24}N_{12}Cl_4$	-
mp-1198926	$Re_8H_{96}C_{24}S_{24}N_{48}Cl_{48}$	-
mp-1199490	$Mn_4H_{64}C_{16}S_{16}N_{32}Cl_8$	-
mp-1199686	$Mo_4P_{16}H_{152}C_{52}N_{16}Cl_{16}$	-
mp-1203403	$C_{121}S_2Cl_{20}$	-
mp-1204279	$Si_{16}Te_8H_{176}Pd_8C_{64}Cl_{16}$	-
mp-1204629	$P_{16}H_{216}C_{80}N_{32}Cl_8$	-

Table B.2: A table of manually identified entries from Materials Project that experience issues concerning Matminer's featurization tools. These were excluded from the dataset.

B.3 Optimalization

Table B.3: Table of scores versus principal components used in the model for the Ferrenti approach.

Model	PC	Mean (std) test score	Mean (std) f1-score
LOG	1	0.64(0.010)	0.76(0.007)
	2	0.74(0.020)	0.82(0.013)
	5	0.78(0.018)	0.84(0.013)
	10	0.79(0.019)	0.85(0.013)
	50	0.95(0.014)	0.96(0.010)
	176	0.98(0.010)	0.99(0.008)
Optimal	173	0.98(0.008)	0.99(0.006)
DT	1	0.67(0.015)	0.78(0.011)
	2	0.74(0.017)	0.83(0.010)
	5	0.77(0.024)	0.83(0.019)
	10	0.76(0.028)	0.82(0.024)
	50	0.77(0.029)	0.82(0.021)
	176	0.72(0.030)	0.79(0.023)
Optimal	37	0.79(0.024)	0.84(0.020)
RF	1	0.68(0.015)	0.78(0.010)
	2	0.75(0.019)	0.83(0.011)
	5	0.82(0.023)	0.87(0.016)
	10	0.84(0.025)	0.88(0.017)
	50	0.90(0.020)	0.93(0.014)
	176	0.87(0.016)	0.91(0.010)
Optimal	50	0.90(0.020)	0.93(0.014)
GB	1	0.68(0.015)	0.78(0.010)
	2	0.74(0.019)	0.83(0.011)
	5	0.83(0.020)	0.87(0.015)
	10	0.85(0.020)	0.89(0.015)
	50	0.93(0.017)	0.95(0.012)
	176	0.93(0.017)	0.95(0.012)
Optimal	104	0.93(0.014)	0.95(0.010)

Bibliography

1. Griffiths, D. *Introduction to quantum mechanics* ISBN: 9781107179868 (Cambridge University Press, Cambridge, 2017).
2. Turing, A. M. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London mathematical society* **2,** 230–265 (1937).
3. Moore, G. Cramming More Components Onto Integrated Circuits. *Proceedings of the IEEE* **86,** 82–85 (Jan. 1965).
4. Pavičić, M. *Quantum computation and quantum communication : theory and experiments* ISBN: 9786610743704 (Springer, New York, 2006).
5. Gwennap, L. Apple's 5 Nanometer Chip Is Another Signpost That Moore's Law Is Running Out. *Forbes.* <https://www.forbes.com/sites/linleygwennap/2020/10/12/apple-moores-law-is-running-out/> (Oct. 12, 2020).
6. Weber, J. R. *et al.* Quantum computing with defects. *Proceedings of the National Academy of Sciences* **107,** 8513–8518 (Apr. 2010).
7. DiVincenzo, D. P. The Physical Implementation of Quantum Computation. *Fortschritte der Physik* **48,** 771–783 (Sept. 2000).
8. Ladd, T. D. *et al.* Quantum computers. *Nature* **464,** 45–53 (Mar. 2010).
9. Mizel, A., Lidar, D. A. & Mitchell, M. Simple Proof of Equivalence between Adiabatic Quantum Computation and the Circuit Model. *Physical Review Letters* **99.** doi:10.1103/physrevlett.99.070502 (Aug. 2007).
10. Grover, L. K. A framework for fast quantum mechanical algorithms. arXiv: quant-ph/9711043v2 `[quant-ph]` (Nov. 20, 1997).
11. Shor, P. *Algorithms for quantum computation: discrete logarithms and factoring* in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE Comput. Soc. Press, 1994). doi:10.1109/sfcs.1994.365700.
12. Martinis, J. M. *et al. Quantum supremacy using a programmable superconducting processor* en. 2019. doi:10.5061/DRYAD.K6T1RJ8.
13. Georgescu, I. The DiVincenzo criteria 20 years on. *Nature Reviews Physics* **2,** 666–666 (Nov. 2020).

14. Griffiths, R. B. Nature and location of quantum information. *Physical Review A* **66.** doi:10.1103/physreva.66.012311 (July 2002).

15. Gisin, N., Ribordy, G., Tittel, W. & Zbinden, H. Quantum cryptography. *Reviews of Modern Physics* **74,** 145–195 (Mar. 2002).

16. Gisin, N. & Thew, R. Quantum communication. *Nature Photonics* **1,** 165–171 (Mar. 2007).

17. Acín, A. *et al.* The quantum technologies roadmap: a European community view. *New Journal of Physics* **20,** 080201 (Aug. 2018).

18. Boaron, A. *et al.* Secure Quantum Key Distribution over 421 km of Optical Fiber. *Physical Review Letters* **121.** doi:10.1103/physrevlett.121.190502 (Nov. 2018).

19. Degen, C. L., Reinhard, F. & Cappellaro, P. Quantum sensing. *Reviews of Modern Physics* **89.** doi:10.1103/revmodphys.89.035002 (July 2017).

20. Kristian Fossheim, A. S. *Superconductivity: Physics and Applications* 442 pp. ISBN: 0470844523. <https://www.ebook.de/de/product/3608091/kristian_fossheim_asle_sudboe_superconductivity_physics_and_applications.html> (WILEY, 2004).

21. Lufaso, M. W. & Woodward, P. M. Prediction of the crystal structures of perovskites using the software program SPuDS. *Acta Crystallographica Section B Structural Science* **57,** 725–738 (Nov. 2001).

22. Bednorz, J. G. & Müller, K. A. Perovskite-type oxides—The new approach to high-Tcsuperconductivity. *Reviews of Modern Physics* **60,** 585–600 (July 1988).

23. Boivin, J. C. & Mairesse, G. Recent Material Developments in Fast Oxide Ion Conductors. *Chemistry of Materials* **10,** 2870–2888 (Oct. 1998).

24. Cheong, S.-W. & Mostovoy, M. Multiferroics: a magnetic twist for ferroelectricity. *Nature Materials* **6,** 13–20 (Jan. 2007).

25. Ibn-Mohammed, T. *et al.* Perovskite solar cells: An integrated hybrid lifecycle assessment and review in comparison with other photovoltaic technologies. *Renewable and Sustainable Energy Reviews* **80,** 1321–1344 (Dec. 2017).

26. Chen, P.-Y. *et al.* Environmentally responsible fabrication of efficient perovskite solar cells from recycled car batteries. *Energy Environ. Sci.* **7,** 3659–3665 (2014).

27. Pauli, W. Über den Zusammenhang des Abschlusses der Elektronengruppen im Atom mit der Komplexstruktur der Spektren. *Zeitschrift für Physik* **31,** 765–783 (Feb. 1925).

28. Martienssen, W. *Springer handbook of condensed matter and materials data* ISBN: 9786610625949 (Springer, Heidelberg New York, 2005).

29. Ben Streetman, S. B. *Solid State Electronic Devices, Global Edition* 632 pp. ISBN: 1292060557. <https://www.ebook.de/de/product/30394493/ben_streetman_sanjay_banerjee_solid_state_electronic_devices_global_edition.html> (Pearson Education Limited, 2015).

30. Pelant, I. *Luminescence spectroscopy of semiconductors* ISBN: 0191738549 (Oxford University Press, Oxford, 2012).

31. Kun Huang, A. R. Theory of light absorption and non-radiative transitions in F -centres. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* **204,** 406–423 (Dec. 1950).

32. Gordon, L. *et al.* Quantum computing with defects. *MRS Bulletin* **38,** 802–807 (Oct. 2013).

33. Bernien, H. *et al.* Heralded entanglement between solid-state qubits separated by three metres. *Nature* **497,** 86–90 (Apr. 2013).

34. Taylor, J. M. *et al.* High-sensitivity diamond magnetometer with nanoscale resolution. *Nature Physics* **4,** 810–816 (Sept. 2008).

35. Barclay, P. E., Fu, K.-M. C., Santori, C., Faraon, A. & Beausoleil, R. G. Hybrid Nanocavity Resonant Enhancement of Color Center Emission in Diamond. *Physical Review X* **1.** doi:10.1103/physrevx.1.011007 (Sept. 2011).

36. Neudeck, P. G. Progress in silicon carbide semiconductor electronics technology. *Journal of Electronic Materials* **24,** 283–288 (Apr. 1995).

37. Silveira, E., Freitas, J. A., Glembocki, O. J., Slack, G. A. & Schowalter, L. J. Excitonic structure of bulk AlN from optical reflectivity and cathodoluminescence measurements. *Physical Review B* **71.** doi:10.1103/physrevb.71.041201 (Jan. 2005).

38. Lawaetz, P. Valence-Band Parameters in Cubic Semiconductors. *Physical Review B* **4,** 3460–3467 (Nov. 1971).

39. Beckers, L. *et al.* Structural and optical characterization of epitaxial waveguiding BaTiO3 thin films on MgO. *Journal of Applied Physics* **83,** 3305–3310 (Mar. 1998).

40. Kumbhojkar, N., Nikesh, V. V., Kshirsagar, A. & Mahamuni, S. Photophysical properties of ZnS nanoclusters. *Journal of Applied Physics* **88,** 6260–6264 (Dec. 2000).

41. Bassett, L. C., Alkauskas, A., Exarhos, A. L. & Fu, K.-M. C. Quantum defects by design. *Nanophotonics* **8,** 1867–1888 (Oct. 2019).

42. James, W. J. Theory of defects in solidsby A. M. Stoneham. *Acta Crystallographica Section A* **32,** 527–527 (May 1976).

43. Togan, E. *et al.* Quantum entanglement between an optical photon and a solid-state spin qubit. *Nature* **466,** 730–734 (Aug. 2010).

44. Bathen, M. E. *Point defects in silicon carbide for quantum technologies: Identification, tuning and control* PhD thesis (The Faculty of Mathematics and Natural Sciences, University of Oslo).

45. Son, N. T. *et al.* Developing silicon carbide for quantum spintronics. *Applied Physics Letters* **116,** 190501 (May 2020).

46. Falk, A. L. *et al.* Polytype control of spin qubits in silicon carbide. *Nature Communications* **4.** doi:10.1038/ncomms2854 (May 2013).

47. Widmann, M. *et al.* Coherent control of single spins in silicon carbide at room temperature. *Nature Materials* **14,** 164–168 (Dec. 2014).

48. Zhang, G., Cheng, Y., Chou, J.-P. & Gali, A. Material platforms for defect qubits and single-photon emitters. *Applied Physics Reviews* **7,** 031308 (Sept. 2020).

49. Redjem, W. *et al.* Single artificial atoms in silicon emitting at telecom wavelengths. *Nature Electronics* **3,** 738–743 (Nov. 2020).

50. Wang, J. *et al.* Gallium arsenide (GaAs) quantum photonic waveguide circuits. *Optics Communications* **327,** 49–55 (Sept. 2014).

51. Berhane, A. M. *et al.* Photophysics of GaN single-photon emitters in the visible spectral range. *Physical Review B* **97.** doi:10.1103/physrevb.97.165202 (Apr. 2018).

52. Xue, Y. *et al.* Single-Photon Emission from Point Defects in Aluminum Nitride Films. *The Journal of Physical Chemistry Letters* **11,** 2689–2694 (Mar. 2020).

53. Varley, J. B., Janotti, A. & de Walle, C. G. V. Defects in AlN as candidates for solid-state qubits. *Physical Review B* **93.** doi:10.1103/physrevb.93.161201 (Apr. 2016).

54. Hardy, W. J. *et al.* Single and double hole quantum dots in strained Ge/SiGe quantum wells. *Nanotechnology* **30,** 215202 (Mar. 2019).

55. Toth, M. & Aharonovich, I. Single Photon Sources in Atomically Thin Materials. *Annual Review of Physical Chemistry* **70,** 123–142 (June 2019).

56. Atatüre, M., Englund, D., Vamivakas, N., Lee, S.-Y. & Wrachtrup, J. Material platforms for spin-based photonic quantum technologies. *Nature Reviews Materials* **3,** 38–51 (Apr. 2018).

57. Tran, T. T. *et al.* Robust Multicolor Single Photon Emission from Point Defects in Hexagonal Boron Nitride. *ACS Nano* **10,** 7331–7338 (July 2016).

58. Tran, T. T., Bray, K., Ford, M. J., Toth, M. & Aharonovich, I. *Quantum Emission from Hexagonal Boron Nitride Monolayers* in *Conference on Lasers and Electro-Optics* (OSA, 2016). doi:10.1364/cleo_qels.2016.ftu4d.1.

59. Weston, L., Wickramaratne, D., Mackoit, M., Alkauskas, A. & de Walle, C. G. V. Native point defects and impurities in hexagonal boron nitride. *Physical Review B* **97.** doi:10.1103/physrevb.97.214104 (June 2018).

60. Abdi, M., Chou, J.-P., Gali, A. & Plenio, M. B. Color Centers in Hexagonal Boron Nitride Monolayers: A Group Theory and Ab Initio Analysis. *ACS Photonics* **5,** 1967–1976 (Apr. 2018).

61. Jain, A., Persson, K. A. & Ceder, G. Research Update: The materials genome initiative: Data sharing and the impact of collaborative ab initio databases. *APL Materials* **4,** 053102 (Mar. 2016).

62. Magee, C. L. Towards quantification of the role of materials innovation in overall technological development. *Complexity* **18,** 10–25 (June 2012).

63. Agrawal, A. & Choudhary, A. Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science. *APL Materials* **4,** 053208 (Apr. 2016).

64. Schleder, G. R., Padilha, A. C. M., Acosta, C. M., Costa, M. & Fazzio, A. From DFT to machine learning: recent approaches to materials science–a review. *Journal of Physics: Materials* **2,** 032001 (May 2019).

65. Persson, C. *Brief Introduction to the density functional theory* 2020.

66. Top500. *SUPERCOMPUTER FUGAKU* June 2020. <https://www.top500.org/system/179807/> (visited on 10/02/2020).

67. David Sholl, J. A. S. *Density Functional Theory: A Practical Introduction* 238 pp. ISBN: 0470373172. <https://www.ebook.de/de/product/7207845/david_sholl_janice_a_steckel_density_functional_theory_a_practical_introduction.html> (WILEY, 2009).

68. Hohenberg, P. & Kohn, W. Inhomogeneous Electron Gas. *Physical Review* **136,** B864–B871 (Nov. 1964).

69. Toulouse, J. *Introduction to density-functional theory* Sept. 2019. <http://www.lct.jussieu.fr/pagesperso/toulouse/enseignement/introduction_dft.pdf> (visited on 10/25/2020).

70. Kohn, W. & Sham, L. J. Self-Consistent Equations Including Exchange and Correlation Effects. *Physical Review* **140,** A1133–A1138 (Nov. 1965).

71. Allen, J. P. & Watson, G. W. Occupation matrix control of d- and f-electron localisations using DFT+U. *Phys. Chem. Chem. Phys.* **16,** 21016–21031 (2014).

72. Perdew, J. P. & Wang, Y. Accurate and simple analytic representation of the electron-gas correlation energy. *Physical Review B* **45,** 13244–13249 (June 1992).

73. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized Gradient Approximation Made Simple. *Physical Review Letters* **77,** 3865–3868 (Oct. 1996).

74. Freysoldt, C. *et al.* First-principles calculations for point defects in solids. *Reviews of Modern Physics* **86,** 253–305 (Mar. 2014).

75. Tran, F. & Blaha, P. Accurate Band Gaps of Semiconductors and Insulators with a Semilocal Exchange-Correlation Potential. *Physical Review Letters* **102.** doi:10.1103/physrevlett.102.226401 (June 2009).

76. Becke, A. D. & Johnson, E. R. A simple effective potential for exchange. *The Journal of Chemical Physics* **124,** 221101 (June 2006).

77. Koller, D., Tran, F. & Blaha, P. Merits and limits of the modified Becke-Johnson exchange potential. *Physical Review B* **83.** doi:10.1103/physrevb.83.195134 (May 2011).

78. Becke, A. D. A new mixing of Hartree–Fock and local density-functional theories. *The Journal of Chemical Physics* **98,** 1372–1377 (Jan. 1993).

79. Perdew, J. P., Ernzerhof, M. & Burke, K. Rationale for mixing exact exchange with density functional approximations. *The Journal of Chemical Physics* **105,** 9982–9985 (Dec. 1996).

80. Aryasetiawan, F. & Gunnarsson, O. TheGWmethod. *Reports on Progress in Physics* **61,** 237–312 (Mar. 1998).

81. Heyd, J., Scuseria, G. E. & Ernzerhof, M. Hybrid functionals based on a screened Coulomb potential. *The Journal of Chemical Physics* **118,** 8207–8215 (May 2003).

82. Krukau, A. V., Vydrov, O. A., Izmaylov, A. F. & Scuseria, G. E. Influence of the exchange screening parameter on the performance of screened hybrid functionals. *The Journal of Chemical Physics* **125,** 224106 (Dec. 2006).

83. Klimeš, J., Bowler, D. R. & Michaelides, A. Chemical accuracy for the van der Waals density functional. *Journal of Physics: Condensed Matter* **22,** 022201 (Dec. 2009).

84. Dion, M., Rydberg, H., Schröder, E., Langreth, D. C. & Lundqvist, B. I. Van der Waals Density Functional for General Geometries. *Physical Review Letters* **92.** doi:10.1103/physrevlett.92.246401 (June 2004).

85. Freitas, L. C. G. Prèmio Nobel de Quìmica em 1998: Walter Kohn e John A. Pople. *Quìmica Nova* **22,** 293–298 (Apr. 1999).

86. Yang, D. *et al.* Functionality-Directed Screening of Pb-Free Hybrid Organic–Inorganic Perovskites with Desired Intrinsic Photovoltaic Functionalities. *Chemistry of Materials* **29,** 524–538 (Jan. 2017).

87. Warren, J. A. The Materials Genome Initiative and artificial intelligence. *MRS Bulletin* **43,** 452–457 (June 2018).

88. *Machine Learning Meets Quantum Physics* (eds Schütt, K. T. *et al.*) doi:10.1007/978-3-030-40245-7 (Springer International Publishing, 2020).

89. Curtarolo, S. *et al.* AFLOWLIB.ORG: A distributed materials properties repository from high-throughput ab initio calculations. *Computational Materials Science* **58,** 227–235 (June 2012).

90. Curtarolo, S. *et al.* AFLOW: An automatic framework for high-throughput materials discovery. *Computational Materials Science* **58,** 218–226 (June 2012).

91. Calderon, C. E. *et al.* The AFLOW standard for high-throughput materials science calculations. *Computational Materials Science* **108,** 233–238 (Oct. 2015).

92. Jain, A. *et al.* Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials* **1,** 011002 (July 2013).

93. Saal, J. E., Kirklin, S., Aykol, M., Meredig, B. & Wolverton, C. Materials Design and Discovery with High-Throughput Density Functional Theory: The Open Quantum Materials Database (OQMD). *JOM* **65,** 1501–1509 (Sept. 2013).

94. Kirklin, S. *et al.* The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies. *npj Computational Materials* **1.** doi:10.1038/npjcompumats.2015.10 (Dec. 2015).

95. Choudhary, K. *et al.* JARVIS: An Integrated Infrastructure for Data-driven Materials Design. arXiv: `2007.01831v1 [cond-mat.mtrl-sci]` (July 3, 2020).

96. Allen, F., Bergerhoff & Sievers, R. *Crystallographic Databases* Chester, UK, 1987.

97. Wang, L., Maxisch, T. & Ceder, G. Oxidation energies of transition metal oxides within the GGA+U framework. *Physical Review B* **73.** doi:10.1103/physrevb.73.195107 (May 2006).

98. Setyawan, W. & Curtarolo, S. High-throughput electronic band structure calculations: Challenges and tools. *Computational Materials Science* **49,** 299–312 (Aug. 2010).

99. Setyawan, W., Gaume, R. M., Lam, S., Feigelson, R. S. & Curtarolo, S. High-Throughput Combinatorial Database of Electronic Band Structures for Inorganic Scintillator Materials. *ACS Combinatorial Science* **13,** 382–390 (June 2011).

100. Ferrenti, A. M., de Leon, N. P., Thompson, J. D. & Cava, R. J. Identifying candidate hosts for quantum defects via data mining. *npj Computational Materials* **6.** doi:10.1038/s41524-020-00391-7 (Aug. 2020).

101. Stevanović, V., Lany, S., Zhang, X. & Zunger, A. Correcting density functional theory for accurate predictions of compound enthalpies of formation: Fitted elemental-phase reference energies. *Physical Review B* **85.** doi:10.1103/physrevb.85.115104 (Mar. 2012).

102. Thonhauser, T. *et al.* Van der Waals density functional: Self-consistent potential and the nature of the van der Waals bond. *Physical Review B* **76.** doi:10.1103/physrevb.76.125112 (Sept. 2007).

103. Klimeš, J., Bowler, D. R. & Michaelides, A. Van der Waals density functionals applied to solids. *Physical Review B* **83.** doi:10.1103/physrevb.83.195131 (May 2011).

104. Choudhary, K., Cheon, G., Reed, E. & Tavazza, F. Elastic properties of bulk and low-dimensional materials using van der Waals density functional. *Physical Review B* **98.** doi:10.1103/physrevb.98.014107 (July 2018).

105. Choudhary, K. *et al.* Computational screening of high-performance optoelectronic materials using OptB88vdW and TB-mBJ formalisms. *Scientific Data* **5.** doi:10.1038/sdata.2018.82 (May 2018).

106. Mounet, N. *et al.* Two-dimensional materials from high-throughput computational exfoliation of experimentally known compounds. *Nature Nanotechnology* **13,** 246–252 (Feb. 2018).

107. Acosta, C. M. *et al.* Analysis of Topological Transitions in Two-dimensional Materials by Compressed Sensing. arXiv: 1805.10950v1 [cond-mat.mtrl-sci] (May 28, 2018).

108. Polini, M., Guinea, F., Lewenstein, M., Manoharan, H. C. & Pellegrini, V. Artificial honeycomb lattices for electrons, atoms and photons. *Nature Nanotechnology* **8,** 625–633 (Sept. 2013).

109. Eagar, T. *Technology Review* **98,** 42 (1995).

110. Rajan, K. Materials informatics. *Materials Today* **8,** 38–45 (Oct. 2005).

111. A., S. *quoted in New York Times* May 20, 2003. <www.nytimes.com/2003/05/20/science/space/20DWAR.html?ex=1054449062&ei=1&e>.

112. Larsen, A. H. *et al.* The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter* **29,** 273002 (June 2017).

113. Landis, D. D. *et al.* The Computational Materials Repository. *Computing in Science & Engineering* **14,** 51–57 (Nov. 2012).

114. Ong, S. P. *et al.* Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science* **68,** 314–319 (Feb. 2013).

115. Ward, L. *et al.* Matminer: An open source toolkit for materials data mining. *Computational Materials Science* **152,** 60–69 (Sept. 2018).

116. Isayev, O. *et al.* Universal fragment descriptors for predicting properties of inorganic crystals. *Nature Communications* **8.** doi:10.1038/ncomms15679 (June 2017).

117. Murphy, K. *Machine learning : a probabilistic perspective* ISBN: 9780262018029 (MIT Press, Cambridge, Mass, 2012).

118. Wolpert, D. & Macready, W. No Free Lunch Theorems for Search (Mar. 1996).

119. Guido, S. *Introduction to Machine Learning with Python* 400 pp. ISBN: 1449369413. <https://www.ebook.de/de/product/23308778/sarah_guido_introduction_to_machine_learning_with_python.html> (O'Reilly UK Ltd., 2016).

120. Caruana, R. & Niculescu-Mizil, A. *An empirical comparison of supervised learning algorithms* in *Proceedings of the 23rd international conference on Machine learning - ICML '06* (ACM Press, 2006). doi:10.1145/1143844.1143865.

121. Friedman, J., Hastie, T. & Tibshirani, R. Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). *The Annals of Statistics* **28,** 337–407 (Apr. 2000).

122. Friedman, J. H. Greedy function approximation: A gradient boosting machine. *Ann. Statist.* **29,** 1189–1232 (Oct. 2001).

123. James, G., Witten, D., Hastie, T. & Tibshirani, R. *An Introduction to Statistical Learning* ISBN: 1461471370. <https://www.ebook.de/de/product/20292548/gareth_james_daniela_witten_trevor_hastie_robert_tibshirani_an_introduction_to_statistical_learning.html> (Springer-Verlag GmbH, 2017).

124. Marsland, S. *Machine Learning* 457 pp. ISBN: 9781466583337 (Taylor & Francis Ltd., 2014).

125. Ohebbi. *ohebbi/predicting-solid-state-qubit-candidates: v0.1-beta* 2021. doi:10.5281/ZENODO.4633959.

126. Battle, R. & Benson, E. Bridging the semantic Web and Web 2.0 with Representational State Transfer (REST). *Journal of Web Semantics* **6,** 61–69 (Feb. 2008).

127. Rosenbrock, C. W. A Practical Python API for Querying AFLOWLIB. arXiv: 1710.00813v1 [cs.DB] (Sept. 28, 2017).

128. Breuck, P.-P. D., Evans, M. L. & Rignanese, G.-M. Robust model benchmarking and bias-imbalance in data-driven materials science: a case study on MODNet. arXiv: 2102.02263v1 [cond-mat.mtrl-sci] (Feb. 3, 2021).

129. Markham, M. *et al.* CVD diamond for spintronics. *Diamond and Related Materials* **20,** 134–139 (Feb. 2011).

130. Balasubramanian, G. *et al.* Ultralong spin coherence time in isotopically engineered diamond. *Nature Materials* **8,** 383–387 (Apr. 2009).

131. Tyryshkin, A. M. *et al.* Electron spin coherence exceeding seconds in high-purity silicon. *Nature Materials* **11,** 143–147 (Dec. 2011).

132. Ong, S. P. *et al.* The Materials Application Programming Interface (API): A simple, flexible and efficient API for materials data based on REpresentational State Transfer (REST) principles. *Computational Materials Science* **97,** 209–215 (Feb. 2015).

133. Inselberg, A. The plane with parallel coordinates. *The Visual Computer* **1,** 69–91 (Aug. 1985).

134. Inselberg, A. & Dimsdale, B. *Parallel coordinates: a tool for visualizing multi-dimensional geometry* in *Proceedings of the First IEEE Conference on Visualization: Visualization 90* (IEEE Comput. Soc. Press, 1990). doi:10.1109/visual.1990.146402.

135. Ericson, D., Johansson, J. & Cooper, M. *Visual Data Analysis using Tracked Statistical Measures within Parallel Coordinate Representations* in *Coordinated and Multiple Views in Exploratory Visualization (CMV'05)* (IEEE). doi:10.1109/cmv.2005.21.

136. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research (2011).* arXiv: 1201.0490v4 [cs.LG] (Jan. 2, 2012).

137. Lemaitre, G., Nogueira, F. & Aridas, C. K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. arXiv: 1609.06570v1 [cs.LG] (Sept. 21, 2016).

138. Ohebbi. *ohebbi/predicting-ABO3-structures: v0.1-alpha* 2021. doi:10.5281/ZENODO.4633968.

139. Balachandran, P. V. *et al.* Predictions of new ABO3 perovskite compounds by combining machine learning and density functional theory. *Physical Review Materials* **2.** doi:10.1103/physrevmaterials.2.043802 (Apr. 2018).

140. Shannon, R. D. Revised effective ionic radii and systematic studies of interatomic distances in halides and chalcogenides. *Acta Crystallographica Section A* **32,** 751–767 (Sept. 1976).

141. Zhang, H., Li, N., Li, K. & Xue, D. Structural stability and formability of ABO3-type perovskite compounds. *Acta Crystallographica Section B Structural Science* **63,** 812–818 (Nov. 2007).

142. Goldschmidt, V. M. Die Gesetze der Krystallochemie. *Die Naturwissenschaften* **14,** 477–485 (May 1926).

143. Villars, P., Cenzual, K., Daams, J., Chen, Y. & Iwata, S. Data-driven atomic environment prediction for binaries using the Mendeleev number. *Journal of Alloys and Compounds* **367,** 167–175 (Mar. 2004).

144. Niculescu-Mizil, A. & Caruana, R. *Predicting good probabilities with supervised learning* in *Proceedings of the 22nd international conference on Machine learning - ICML '05* (ACM Press, 2005). doi:10.1145/1102351.1102430.

145. Zhang, S. R., Xie, L. H., Ouyang, S. D., Chen, X. W. & Song, K. H. Electronic structure, chemical bonding and optical properties of the nonlinear optical crystal ZnGeP2by first-principles calculations. *Physica Scripta* **91,** 015801 (Nov. 2015).

146. Xing, G. C., Bachmann, K. J., Posthill, J. B. & Timmons, M. L. ZnGeP2: A Wide Bandgap Chalcopyrite Structure Semiconductor for Nonlinear Optical Applications. *MRS Proceedings* **162.** doi:10.1557/proc-162-615 (1989).

147. Caspani, L. *et al.* Integrated sources of photon quantum states based on nonlinear optics. *Light: Science & Applications* **6,** e17100–e17100 (June 2017).

148. Gao, Y. *et al.* Superhard sp2-sp3 hybridized BC2N: A 3D crystal with 1D and 2D alternate metallicity. *Journal of Applied Physics* **121,** 225103 (June 2017).

149. Cai, Y., Xiong, J., Liu, Y. & Xu, X. Electronic structure and chemical hydrogen storage of a porous sp3 tetragonal BC2N compound. *Journal of Alloys and Compounds* **724,** 229–233 (Nov. 2017).

150. Li, H., Xiao, X., Tie, J. & Lu, J. Electronic and magnetic properties of bare armchair BC2N nanoribbons. *Journal of Magnetism and Magnetic Materials* **426,** 641–645 (Mar. 2017).

151. Jiang, C.-L., Zeng, W., Liu, F.-S., Tang, B. & Liu, Q.-J. The shape type of bonds and the direction of phonons in orthorhombic BC2N from first-principles calculations. *Journal of Physics and Chemistry of Solids* **140,** 109349 (May 2020).

152. None Available. *Materials Data on RuC by Materials Project* en. 2020. doi:10.17188/1326297.

153. Scappucci, G. *et al.* The germanium quantum information route. *Nat Rev Mater (2020).* doi:10.1038/s41578-020-00262-z. arXiv: 2004.08133v1 [cond-mat.mes-hall] (Apr. 17, 2020).

154. Pillarisetty, R. Academic and industry research progress in germanium nanodevices. *Nature* **479,** 324–328 (Nov. 2011).

155. Kotochigova, S., Levine, Z. H., Shirley, E. L., Stiles, M. D. & Clark, C. W. Local-density-functional calculations of the energy of atoms. *Physical Review A* **55,** 191–199 (Jan. 1997).

156. Laws, K. J., Miracle, D. B. & Ferry, M. A predictive structural model for bulk metallic glasses. *Nature Communications* **6.** doi:10.1038/ncomms9123 (Sept. 2015).

157. Butler, M. A. & Ginley, D. S. Prediction of Flatband Potentials at Semiconductor-Electrolyte Interfaces from Atomic Electronegativities. *Journal of The Electrochemical Society* **125,** 228–232 (Feb. 1978).

158. Ward, L., Agrawal, A., Choudhary, A. & Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Computational Materials* **2.** doi:10.1038/npjcompumats.2016.28 (Aug. 2016).

159. Deml, A. M., O'Hayre, R., Wolverton, C. & Stevanović, V. Predicting density functional theory total energies and enthalpies of formation of metal-nonmetal compounds by linear regression. *Physical Review B* **93.** doi:10.1103/physrevb.93.085142 (Feb. 2016).

160. Weeber, A. W. Application of the Miedema model to formation enthalpies and crystallisation temperatures of amorphous alloys. *Journal of Physics F: Metal Physics* **17,** 809–813 (Apr. 1987).

161. Yang, X. & Zhang, Y. Prediction of high-entropy stabilized solid-solution in multi-component alloys. *Materials Chemistry and Physics* **132,** 233–238 (Feb. 2012).

162. Rupp, M., Tkatchenko, A., Müller, K.-R. & von Lilienfeld, O. A. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Physical Review Letters* **108.** doi:10.1103/physrevlett.108.058301 (Jan. 2012).

163. Schütt, K. T. *et al.* How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. *Physical Review B* **89.** doi:10.1103/physrevb.89.205118 (May 2014).

164. Faber, F., Lindmaa, A., von Lilienfeld, O. A. & Armiento, R. Crystal structure representations for machine learning models of formation energies. *International Journal of Quantum Chemistry* **115,** 1094–1101 (Apr. 2015).

165. Ewald, P. P. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annalen der Physik* **369,** 253–287 (1921).

166. Hansen, K. *et al.* Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space. *The Journal of Physical Chemistry Letters* **6,** 2326–2331 (June 2015).

167. Ward, L. *et al.* Including crystal structure attributes in machine learning models of formation energies via Voronoi tessellations. *Physical Review B* **96.** doi:10.1103/physrevb.96.024104 (July 2017).

168. Botu, V. & Ramprasad, R. Adaptive machine learning framework to accelerate ab initio molecular dynamics. *International Journal of Quantum Chemistry* **115,** 1074–1083 (Dec. 2014).

169. De Jong, M. *et al.* A Statistical Learning Framework for Materials Science: Application to Elastic Moduli of k-nary Inorganic Polycrystalline Compounds. *Scientific Reports* **6.** doi:10.1038/srep34256 (Oct. 2016).

170. Seko, A., Hayashi, H., Nakayama, K., Takahashi, A. & Tanaka, I. Representation of compounds for machine-learning prediction of physical properties. *Physical Review B* **95.** doi:10.1103/physrevb.95.144110 (Apr. 2017).

171. Steinhardt, P. J., Nelson, D. R. & Ronchetti, M. Bond-orientational order in liquids and glasses. *Physical Review B* **28,** 784–805 (July 1983).

172. Waroquiers, D. *et al.* Statistical Analysis of Coordination Environments in Oxides. *Chemistry of Materials* **29,** 8346–8360 (Sept. 2017).

173. Zimmermann, N. E. R., Horton, M. K., Jain, A. & Haranczyk, M. Assessing Local Structure Motifs Using Order Parameters for Motif Recognition, Interstitial Identification, and Diffusion Path Characterization. *Frontiers in Materials* **4.** doi:10.3389/fmats.2017.00034 (Nov. 2017).

174. Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics* **134,** 074106 (Feb. 2011).

175. Khorshidi, A. & Peterson, A. A. Amp: A modular approach to machine learning in atomistic simulations. *Computer Physics Communications* **207,** 310–324 (Oct. 2016).

176. Peng, H. L., Li, M. Z. & Wang, W. H. Structural Signature of Plastic Deformation in Metallic Glasses. *Physical Review Letters* **106.** doi:10.1103/physrevlett.106.135503 (Mar. 2011).

177. Wang, Q. & Jain, A. A transferable machine-learning framework linking interstice distribution and plastic heterogeneity in metallic glasses. *Nature Communications* **10.** doi:10.1038/s41467-019-13511-9 (Dec. 2019).

178. Dylla, M. T., Dunn, A., Anand, S., Jain, A. & Snyder, G. J. Machine Learning Chemical Guidelines for Engineering Electronic Structures in Half-Heusler Thermoelectric Materials. *Research* **2020,** 1–8 (Apr. 2020).

179. None Available. *Materials Data on PH6C2S2N(ClO2)2 by Materials Project* en. 2020. doi:10.17188/1268877.

180. None Available. *Materials Data on Nb7S2I19 by Materials Project* en. 2014. doi:10.17188/1277059.