Cross-Validation in Cluster-Expansions

Exploiting the Hat-Matrix-Formalism

BACHELORARBEIT

zur Erlangung des akademischen Grades
Bachelor of Science
(B. Sc.)
im Fach Physik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät I
Institut für Physik
Humboldt-Universität zu Berlin

von
Herr Axel Felix Hübner
geboren am 13.11.1995 in Berlin

Betreuung:

1. *Prof. Dr. Dr. h.c. Claudia Draxl*
2. *Prof. Dr. Igor Sokolov*

eingereicht am: *05.07.2016*

Abstract

This thesis addresses the cross-validation score, CV, as statistical quantity in the context of the cluster expansion technique used in alloy theory. An analytical formula for the computation of *leave-many-out* CV is derived. The numerical stability and performance of the formula is investigated, analytically and by computer experiments. Furthermore, a strict relation between the CV and the noise in the data is outlined. For all this, the singular value decomposition as a solving technique for linear least-square problems is used, and the benefits and costs of this method are briefly discussed.

Keywords:
Cross-Validation, Cluster-Expansion, Hat-Matrix

Contents

Chapter 1

Introduction

The calculation of formation energies of alloys from first principles is a computationally very demanding task. It is practically impossible to calculate the partition function and other properties in a reasonable amount of time. To deal with this, one has to resort to the building of models that allow for a quick calculation of the energy. An example for such procedure is provided by the cluster expansion technique [16]. In this technique a model is fitted to a small set of *ab-initio* data, the so-called training set. One element of the training set is called a data point and the expectation of the model for the energy of formation is called a 'prediction'.

An essential part of the model building consists in determining how accurate the predictions are. This thesis addresses the model validation technique of cross-validation which provides a way to assess the predictive performance of a model [17, 22]. In this technique, a number of data points is excluded from the training set. With this reduced set a new model is built, and the predictive performance of the new model for the excluded data points is analyzed. If the excluded set contains only one point this is called *leave-one-out* cross-validation (LOOCV), otherwise *leave-many-out* cross-validation (LMOCV). This procedure, repeated for a number of sets, yields the cross-validation score (CV). If a new regression is made for each of the N sets, this is called 'direct' approach and scales in computation like N times the matrix inversion for $m \times m$ matrices, where m is the number of model parameters.

An analytical formula [27, 28] for the LOOCV,

$$\mathrm{CV}^2 = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{P_i - \widehat{P}_i}{1 - h_{ii}} \right)^2 ,$$

avoids frequent matrix inversions and can thus be faster than the direct computation. Here the $\widehat{P}_i$'s are the predictions for the data points P_i using the whole training set, while h_{ii} is the i^{th} diagonal entry of the models projection operator, also known as hat-matrix.

This formula presents numerical instabilities in actual implementations. The goal of this thesis is to understand, from an analytical point of view, the reasons for such instabilities, devise ways to circumvent them, and to extend this analytical formula for the LMOCV. Finally, the relation between the CV and the noise in the data is investigated.

To find the reasons of these instabilities and extensible methods for the calculation of the CV, the hat-matrix plays a central role. This matrix is a projector, that carries information about the configurations and relates data and expectation for a linear model [12, 20]. In that sense, the hat-matrix carries important information about the model design and can help to understand its influence on the CV. This turns out to be a central part of this thesis, leading to the development of an analytic *leave-many-out* CV formula and a strict relation to the calculation noise. To my best knowledge, both these achievements have not been published before.

This thesis is organized as follows: In the first chapter, a brief mathematical introduction is shown followed by an investigation on efficient ways to calculate the cross-validation score, only relying on linear algebra. Afterwards, the reasons for the instabilities are addressed using analytical methods, and a relation of the CV to the noise in the data is derived. The second chapter begins with considerations for the computation of the hat-matrix. Then the analytical expectations of the relation of the CV to the noise and the instabilities are reviewed in numerical experiments. Finally, the runtime and numerical stability of an improved algorithm, using the previous considerations about the LMOCV and the instabilities, are discussed and compared to the direct approach.

Chapter 2

Cross-Validation

In this chapter, first, the cluster expansion technique will be introduced (Sec. 2.1). Then, in Sec. 2.2, the mathematical background will be treated. In Sec. 2.3 the derivation of an analytic leave-many-out cross-validation formula will be shown, and the existence of the cross-validation score will be discussed. At the end, in Sec. 2.4, a relation between the cross-validation and noise in the data will be outlined.

2.1 Cluster Expansion Technique

The cluster expansion is a technique to build models that can be used to predict the formation energies in crystalline alloys [28]. These models are adapted to a set of atomic arrangements, i.e. the training set, with energies calculated by first principles, for instance density functional theory (DFT) calculations [5].

A crystalline structure is defined by the (crystal) lattice, which is a periodic arrangement of groups of atoms. In an alloy, the atomic sites can be occupied, for example, by two different atomic species, labelled A and B. Every arrangement, or configuration, is defined by occupation variables σ_s for each crystal site, e.g. with values $\sigma_s = \pm 1$ depending on the sort of atom that can be found there. In the cluster expansion technique, atoms of type A are assigned -1 and atoms of type B (the substitutional species) +1. Thus an arbitrary configuration i can be assigned a vector $\sigma^{(i)} = (\sigma_1, \sigma_2, ...)$. Assigning a pseudo-spin variable σ_s to each crystal site the cluster expansion is closely related to the Ising model of a ferromagnet [21]. It can be extended to more than only binary compounds [9].

The cluster expansion models the properties of these configurations with smaller subsets of lattice sites, so called *clusters* α, see Fig. 2.1. A parameter $\mathfrak{J}_\alpha$ is assigned to each cluster in the model, describing the contribution of the cluster to the property of interest. In this case, the property is the energy, labelled E. If all possible clusters are included, every possible feature of the crystal can be represented [28].

In the cluster expansion the models are defined as [28]

$$\widehat{E}(\sigma^{(i)}) = \widehat{E}_i = \sum_\alpha m_\alpha \mathfrak{J}_\alpha \langle \prod_{s \in \beta(\alpha)} \sigma_s \rangle, \tag{2.1}$$

where $\widehat{E}_i$ is the predicted energy of a certain configuration $\sigma^{(i)}$. The sum is taken over the clusters α. The product of the occupation variables is taken over lattice sites

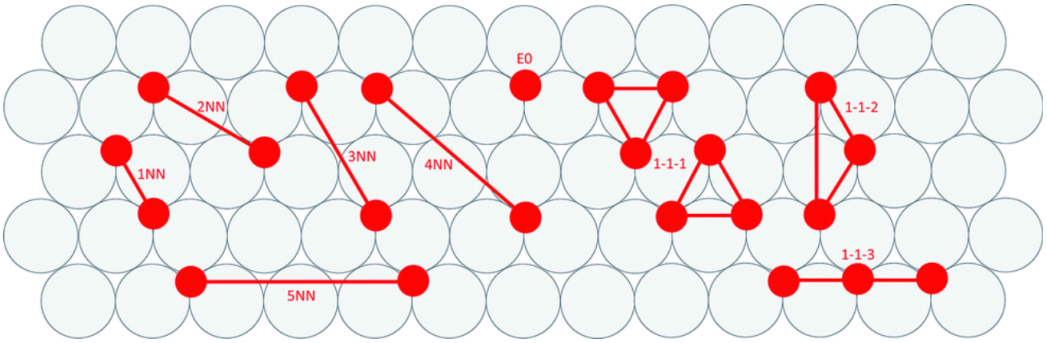

Figure 2.1: Visualization of two- and three-body clusters, that represent groups of atomic sites (red dots connected by red lines). Figure taken from Ref. [9].

in all clusters β which are symmetrically equivalent to the cluster α. The average taken over all clusters β is called the correlation between cluster α and configuration i,

$$X_{i\alpha} = \langle \prod_{s \in \beta(\alpha)} \sigma_s \rangle. \tag{2.2}$$

The correlations between all configurations i and all clusters α are contained in the correlation matrix X. The quantities $\mathfrak{J}_\alpha$ are the so-called *effective cluster interactions* (ECI's), and the m_α are their multiplicities, counting the number of symmetrically equivalent clusters. In this thesis, the parameters J mean the ECI's times their multiplicities[1]

$$J_\alpha = m_\alpha \mathfrak{J}_\alpha. \tag{2.3}$$

With this, Eq. (2.1) can be expressed as

$$\widehat{E}_i = \sum_\alpha X_{i\alpha} J_\alpha. \tag{2.4}$$

In this work, the cluster expansion code `CELL` has been used and applied to clathrate compounds, which are a class of complex alloys [18] of interest for thermoelectric applications.

2.2 Mathematical Background

This section begins with necessary definitions and a review of the role of the hat-matrix in least-square problems (Sec. 2.2.1), to treat the models given by the cluster expansion. Then the cross-validation score is introduced, and its meaning is discussed in (Sec. 2.2.2). Afterwards, in (Sec. 2.2.3), the analytic calculation of the ECI's is performed when a subset of the training set is excluded.

2.2.1 Linear Least-Square Problems and the Hat-Matrix

In this thesis, the term 'data' refers to a function $E(\sigma^{(i)}) = E_i$ representing the calculated energy of the alloy for the configuration $\sigma^{(i)}$. The 'model' is a function $\widehat{E}(\sigma^{(i)}) = \widehat{E}_i$, depending on a vector J of ECI's. The value of $\widehat{E}_i$ for a configuration

[1] In the following, the terms 'ECI' and 'parameter' will be used interchangeably.

$\sigma^{(i)}$ is called the prediction of $\widehat{E}$ for $\sigma^{(i)}$. In a linear model[2], as used in Eq. (2.4) for the cluster expansion, the dependence of $\widehat{E}$ on the model parameters J can be expressed as

$$\widehat{E} = X\ J. \tag{2.5}$$

X is a rectangular matrix consisting of correlations $X_{i\alpha}$, where i indicates a configuration out of the training set $\mathscr{S}$ of size n, while α represents a feature of the model, i.e. a cluster. The number of features is m. $\widehat{E}$ is a column vector of dimension n with components $\widehat{E}_i$. J is a column vector of dimension m.

A criterion to select the parameters J to fit the model best is to minimize the mean square error (MSE) of the predictions for the training set:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(E_i - \widehat{E}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(E_i - X_{i\bullet}\ J)^2, \tag{2.6}$$

where the sum is taken over all configurations i in $\mathscr{S}$, and $X_{i\bullet}$ is a row vector that equals the i^{th} row in X. The optimization of the MSE with respect to the parameters yields (see Appendix A.1)

$$J = (X'\ X)^{-1}X'\ E.$$

Inserting this result in the definition of the MSE one arrives at [22]

$$\begin{aligned}\text{MSE} &= \frac{1}{n}\sum_{i=1}^{n}(E_i - X_{i\bullet}\ (X'\ X)^{-1}X'E)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}(E_i - (\widehat{H}E)_i)^2 \\ &= \frac{1}{n}\left|\left|(I - \widehat{H})E\right|\right|_2^2,\end{aligned} \tag{2.7}$$

where $||v||_q = (\sum_i |v_i|^q)^{1/q}$. Here the hat-matrix $\widehat{H}$ as presented in Refs. [12] and [20], is introduced. The rest of this section follows closely these references. The hat-matrix is given as

$$\widehat{H} = X(X'X)^{-1}X',$$

and is a square matrix of dimension $n \times n$. It projects the data vector E onto the nearest point in the m-dimensional subspace of the predictions. The predicted energies $\widehat{E}$ can be found when comparing Eqs. (2.6) and (2.7):

$$\widehat{E} = \widehat{H}\ E = X\ J. \tag{2.8}$$

The hat-matrix has its name from *putting the hat* on the values E. From this equation it can be seen that the entries of the hat-matrix, h_{ij}, measure the influence the data point i exerts on the prediction of point j in the fit [12]. The projection operator $\widehat{H}$ is symmetric and idempotent. Idempotency implies that all eigenvalues of $\widehat{H}$ are

[2]This is a model that is linear in its parameters.

either 0 or 1. Due to $\widehat{H} = \widehat{H}^2$ the entries of $\widehat{H}$ are given as [12, 20]:

$$\begin{aligned} h_{ii} &= \sum_{j=1}^{n} h_{ij}^2, \\ h_{ii}(1-h_{ii}) &= \sum_{j\neq i}^{n} h_{ij}^2, \\ \Rightarrow 0 \leq & h_{ii} \leq 1. \end{aligned} \tag{2.9}$$

In Eq. (2.9), especially the case $h_{ii} \to 1$ is interesting. It can be interpreted as follows: The prediction for this point in the training set becomes identical to the calculated value $\widehat{E}_i = E_i$. This happens because $h_{ij} \to 0, \forall j \neq i$. Thus, the predicted value of this point is independent of the remaining data. As an illustration of $h_{ii} \to 1$, one can think about a simple model with $m = 1$. In that case $\widehat{E}_i = J \cdot X_i$, where J is a scalar, and the entries of the hat-matrix are

$$h_{ij} = \frac{1}{\sum_k X_k^2} X_i X_j.$$

Thus, if one measuring point X_n lies far away from the rest of the points, which are near the origin, then $h_{nn} \approx 1$. Since $h_{in} \overset{i\neq n}{\approx} 0$, $\widehat{E}_n \approx E_n$ follows. This can be seen in Fig. 2.2: It shows a model where all data points are generated by $E(X) = X\ J + \varepsilon$, J is a scalar, and ε denotes gaussian noise with variance 1/5 and a mean of the distribution 0. Near the origin are ten points, with X-values generated by a uniform distribution in $[-1, 5]$. Far away is one point at $X_n = 35$. The model $\widehat{E}$ is indicated by a black line. The blue dashed line indicates the model that is obtained when one (blue) data point near the origin is changed ($\Delta y = -10$). The model, when the (red) point at X_n has a value $E_n^{(+)} = E(X_n) + 10$ is shown by the red dashed graph. It is obvious from the figure, that J is highly sensitive to the value E_n. This consideration will become important later when discussing the stability of the analytical formula for the cross-validation score.

2.2.2 Cross-Validation Score: Definition and Meaning

The cross-validation score (CV) is a quantity that estimates the predictive power of a model [15, 17, 22, 26]. It is defined as follows

$$\mathrm{CV}^2 = \frac{1}{N} \sum_{\mathscr{E}} \frac{1}{\aleph(\mathscr{E})} \sum_{i \in \mathscr{E}} \left(E_i - \widehat{E}_i^{(\mathscr{E})} \right)^2 . \tag{2.10}$$

Here $\widehat{E}_i^{(\mathscr{E})}$ is the predicted energy for a configuration i when a set of configurations $\mathscr{E}$, which includes i, has been removed from the training set. The sums are taken over all points in the set $\mathscr{E}$ and over N sets $\mathscr{E}$ with a certain cardinality $\aleph$, respectively. The case when $\aleph = 1$ is leave-one-out cross-validation, LOOCV. Here the procedure is repeated for all points in the training set $N = n$:

$$\mathrm{CV}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(E_i - \widehat{E}_i^{(i)} \right)^2 .$$

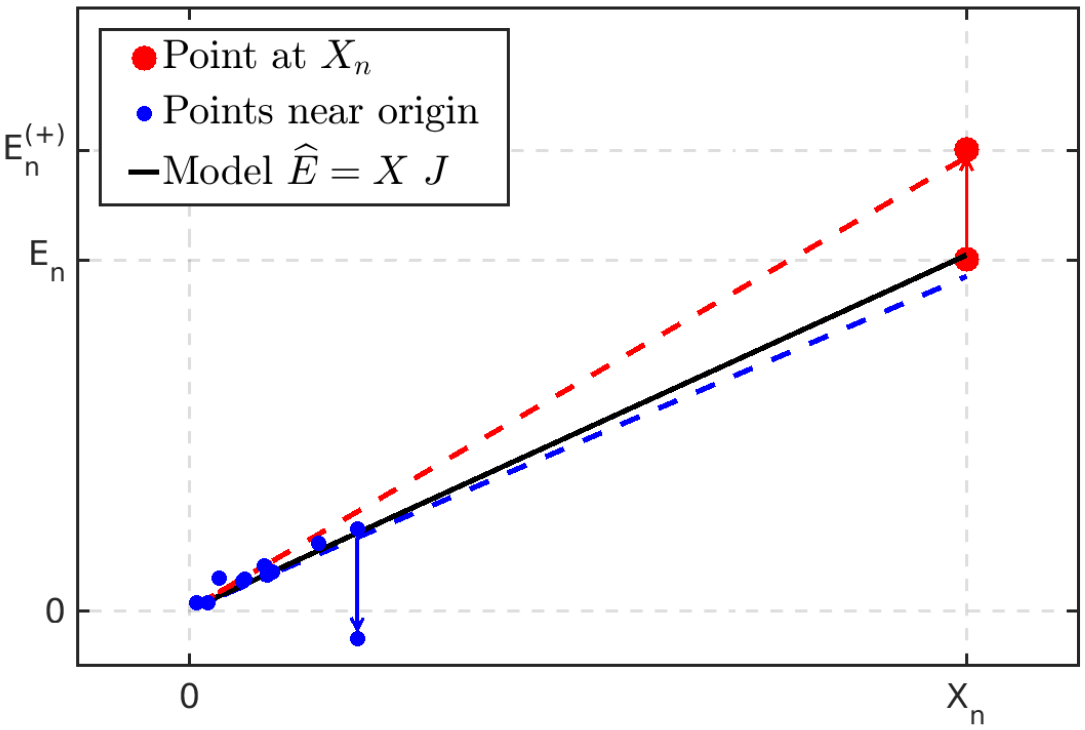

Figure 2.2: Energy E versus correlation X. Illustration of the case $h_{nn} \to 1$. The black solid line indicates the model $\widehat{E}$ for the unchanged data points. The dashed lines indicate the model when a data point is moved as indicated by the arrows.

The case when the cardinality ℵ is larger than one is called leave-many-out cross-validation, LMOCV.

There are two ways to calculate the CV: One way, that I call the 'direct' approach, consists in building least-square fits for every subset $\mathscr{S} \setminus \mathscr{E}$ (here rest set) of the training set $\mathscr{S}$ excluding $\mathscr{E}$, yielding the predictions $\widehat{E}_i^{(\mathscr{E})}$. The other, analytic way, will be outlined later.

As it is impossible to show an m-dimensional illustration for alloys on paper, here, a similar one-dimensional problem is shown.[3] Say we have $n = 11$ data points E_i depending on just one variable σ. All data points are generated by a parabola $f(\sigma) = \frac{1}{4}(\sigma - \frac{1}{2})^2 - \frac{1}{4} + \varepsilon$. ε stands for gaussian noise with a mean of distribution equal to 0 and standard deviation $\frac{1}{\sqrt{10}}$. This can be seen in Fig. 2.3. Here, polynomials of 0^{th}, 2^{nd} and 9^{th} order, are the models, applied to each measuring point σ_i

$$\widehat{E}(\sigma^{(i)}) = \sum_\alpha J_\alpha f_\alpha(\sigma^{(i)}) = (X\ J)_i.$$

J_α are the parameters while f_α are the parameter functions for the model, which are chosen here to be $f_\alpha(\sigma) = \sigma^{\alpha-1}$. They are related to the correlations as

$$f_\alpha(\vec{\sigma}) = X_{\bullet\alpha}.$$

[3] This example is created by me, but supported by literature [12].

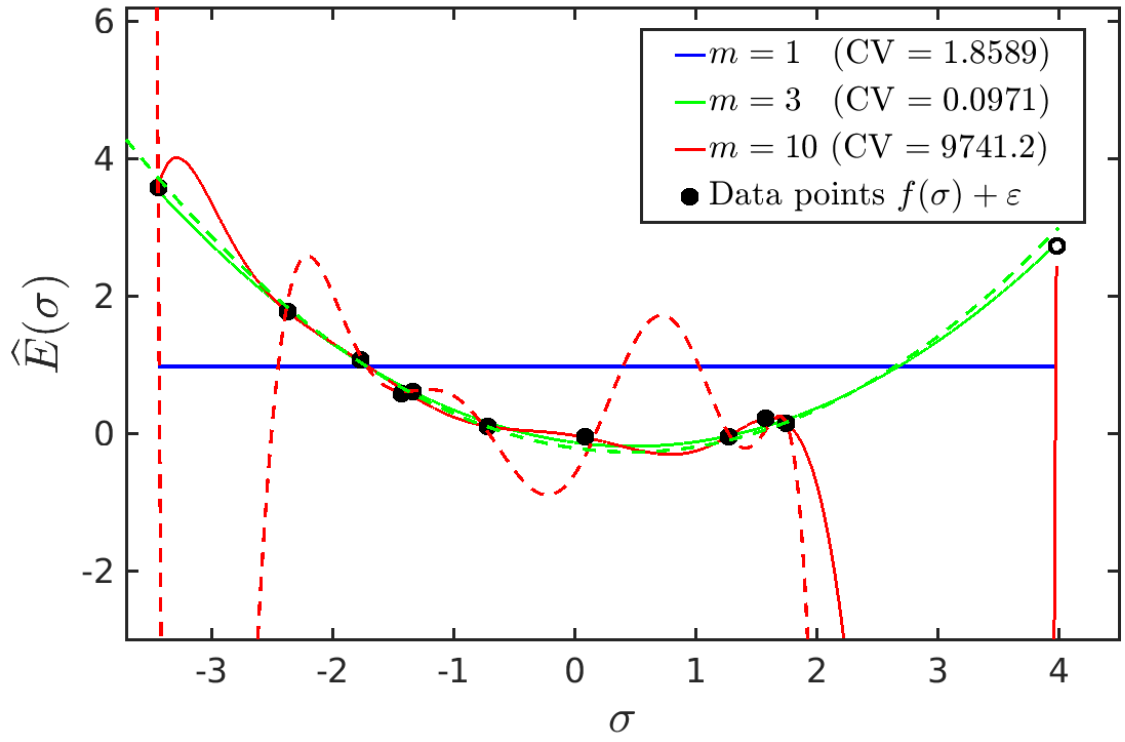

Figure 2.3: Predicted energy $\widehat{E}$ versus configuration σ for different numbers of parameters m. The model predictions for the outermost data point excluded are indicated by dashed lines.

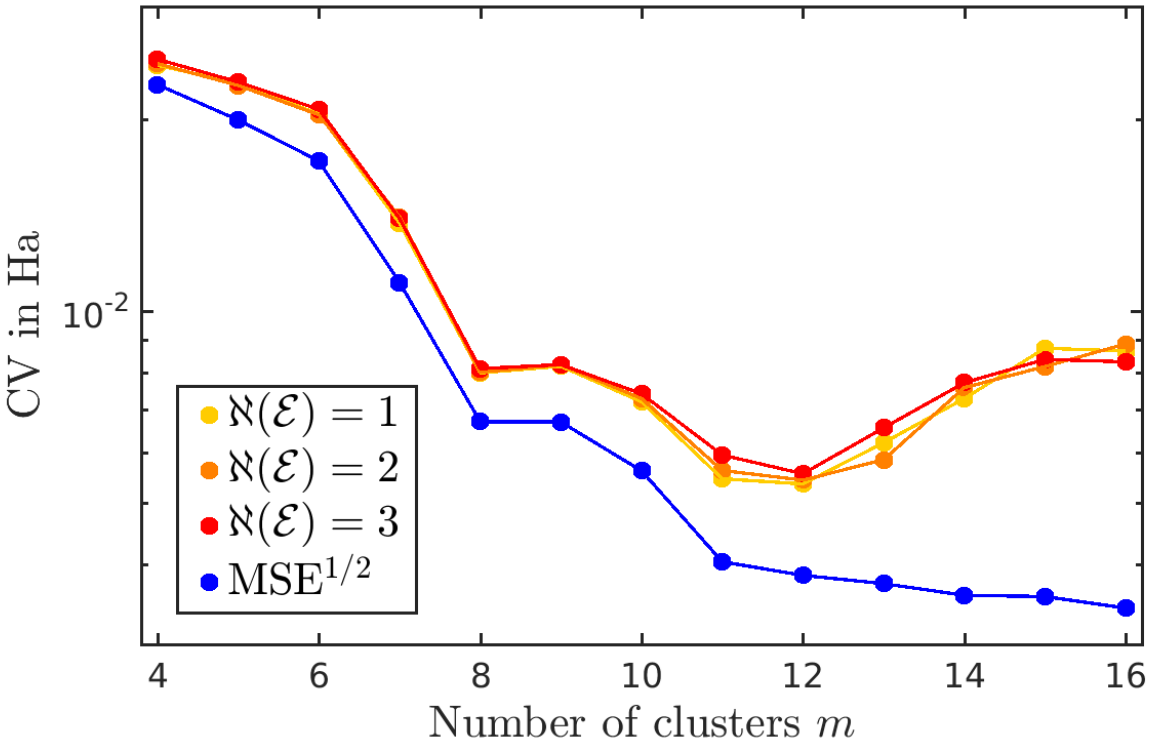

Figure 2.4: CV versus the number of clusters from a cluster expansion for clathrate compounds, together with the square root of the MSE. The training set contained 107 structures, with energies calculated by the local density approximation of density functional theory. Data taken from Ref. [18].

Since the models are linear, the above described formalism for models can be applied. In the cluster expansion approach the f_α are the clusters.

The CV usually shows the following behavior, by increasing the number of parameters. First the CV is large, because the features of the data cannot be represented by a too small number of parameters. Then the number of parameters comes into the region where all features can be represented. This is when the CV undergoes a minimum. Afterwards, also noise is represented by the parameters, and that decreases the predictive performance of the model - so the CV increases. This can be seen, for a cluster expansion, in Fig. 2.4. The fact that the CV starts increasing again can be understood in terms of Fig. 2.3, where in the interval $x \in [2,4]$ suddenly strange values are predicted, without having data points there. If a data point is predicted when excluding the point from the training set, the same can happen for the new prediction (indicated in Fig. 2.3 by dashed lines).

2.2.3 Analytical Calculation of the ECI's

In what follows, an analytical expression for ECI's in the rest set $\mathscr{S} \setminus \mathscr{E}$ is derived. This will be used later to find an analytical expression for the LMOCV. This analytical expression will avoid performing a least-square fit for each rest set $\mathscr{S} \setminus \mathscr{E}$. To find this expression for the ECI's, the MSE of the rest set is optimized with respect to the parameters J. Assuming that a set of structures $\mathscr{E}$ with cardinality $\aleph(\mathscr{E})$, is excluded, the MSE of the rest set $\mathscr{S} \setminus \mathscr{E}$ is

$$\begin{aligned}\mathrm{MSE}_{\mathscr{E}} &= \frac{1}{n-\aleph(\mathscr{E})} \cdot \sum_{i \in \mathscr{S}\setminus\mathscr{E}} \left(E_i - \widehat{E}_i^{(\mathscr{E})}\right)^2 \\ &= \frac{1}{n-\aleph(\mathscr{E})} \cdot \left[\sum_{i=1}^{n} \left(E_i - \sum_\alpha X_{i\alpha} J_\alpha\right)^2 - \sum_{i\in\mathscr{E}} \left(E_i - \sum_\alpha X_{i\alpha} J_\alpha\right)^2\right].\end{aligned}$$

Equating the gradient for the MSE with respect to the ECI's J to zero, leads to (see Appendix A.1):

$$J^{(\mathscr{E})} = \left(I - \sum_{i\in\mathscr{E}} (X'X)^{-1} X'_{i\bullet} X_{i\bullet}\right)^{-1} (X'X)^{-1} \left(X'E - \sum_{i\in\mathscr{E}} X'_{i\bullet} E_i\right) \tag{2.11}$$

where $J^{(\mathscr{E})}$ are the optimized ECI's for the rest set $\mathscr{S} \setminus \mathscr{E}$. $X'_{i\bullet} = (X_{i\bullet})'$. The inversion of the matrix $(I - \sum_{i\in\mathscr{E}} (X'X)^{-1} X'_{i\bullet} X_{i\bullet})$ is a central part of this work. An efficient way to perform this inversion is enabled by the fact that the rank of $\sum_{i\in\mathscr{E}} (X'X)^{-1} X'_i X_i =: Y$ is limited by the cardinality of the excluded sets $\mathscr{E}$:

$$\mathrm{rank}(Y) = \mathrm{rank}\left(\sum_{i\in\mathscr{E}} (X'X)^{-1} X'_{i\bullet} X_{i\bullet}\right) \leq \aleph(\mathscr{E}).$$

This can be seen from the fact that dyadic products have rank 1 and [8, 14]:

$$\begin{aligned}\mathrm{rank}(A+B) &\leq \mathrm{rank}(A) + \mathrm{rank}(B), \\ \mathrm{rank}(AB) &\leq \min\{\mathrm{rank}(A), \mathrm{rank}(B)\}.\end{aligned} \tag{2.12}$$

The matrix $X'X$ is a matrix of size $m \times m$ and is called the Hessian matrix. This matrix must be inverted in Eq. (2.11). Since $X'X$ is a product of two matrices of size $n \times m$, $\text{rank}(X'X) \leq \min(n, m)$ due to Eq. (2.12). Thus X' X is, for example, not invertible anymore when the number m of ECI's exceeds the number n of structures.[4] This situation is frequently present because the number of *ab-initio* calculation is limited, and accurate models require large numbers of clusters. In order to circumvent this problem, instead of minimizing the MSE one can find the optimal ECI's J by minimizing a cost-function $C(J)$, consisting of the sum of the MSE and a so-called *regularization term* $\varphi(J)$:

$$C(J) = \text{MSE}_{\mathscr{E}} + \varphi(J)$$

With this, Eq. (2.11) can be transformed into a tractable problem, as shown below. Here we consider the following form of φ:

$$\frac{n - \aleph(\mathscr{E})}{2}\varphi(J) = J'RJ.$$

In the appendix a more general form is discussed. For certain applications this method is likely to be outperformed by an ℓ^1-regularization [21] (the so-called basis-pursuit problem [3, 23]). As it cannot be solved analytically but needs numerical approaches this method will not be addressed here. Without loss of generality R can be choosen to be symmetric. R can for example be $R = \lambda I$ with $\lambda \geq 0$ (Tikhonov-regularization [23]). This regularization introduces shrinkage of the ECI's by penalizing large interactions [23, 28].

Optimizing the cost-function $C(J)$ results in ECI's $J_{\varphi}^{(\mathscr{E})}$ of the form (see Appendix A.1):

$$J_{\varphi}^{(\mathscr{E})} = \left(I - (X'X + R)^{-1} \sum_{i \in \mathscr{E}} X'_{i\bullet} X_{i\bullet} \right)^{-1} (X'X + R)^{-1} \left(X'E - \sum_{i \in \mathscr{E}} X'_{i\bullet} E_i \right). \tag{2.13}$$

$X'X + R$ can be inverted, even when $m > n$, leading to an optimized set of ECI's.

It is useful to compare the Eqs. (2.11) and (2.13). In both cases the inversion of an operator $I - A$ must be performed. Furthermore, in both cases, the rank of the operator A is just $\aleph(\mathscr{E})$. This, together with a mathematical theorem, Neumann's series, is of crucial importance for possible simplifications.

Theorem: Neumann's Series Identity [11, 29]
For a linear continuous operator A (e.g. a matrix) on a normed space (*here*: $\mathbb{R}^l$, $l \in \mathbb{N}$) with $||A|| < 1$ ($||\bullet||$ is the spectral norm), the following statement holds:

$$(Id - A)^{-1} = \sum_{i=0}^{\infty} A^i, \tag{2.14}$$

where Id is the identity.

Here only the case is considered when linear operators are matrices. With the help of Eq. (2.14) I simplified Eq. (2.13), leading to an analytic formula for LMOCV which can be evaluated easily. This derivation is presented in the next section. The numerical implementation is discussed in Chapter 3.

[4] Thanks to Martin Genzel here, for this hint!

2.3 Leave-Many-Out CV

In this section, first some relations between the hat-matrix and eigenvalues of the matrices $\sum_{i\in\mathscr{E}}(X'X)^{-1}X'_{i\bullet}X_{i\bullet}$ will be outlined. Afterwards, a derivation of an analytical formula for the LMOCV is shown, and its numerical stability is discussed.

2.3.1 Preparation and Motivation

The aim of this section is to understand the behavior of the eigenvalues and eigenvectors of the rank-$\aleph(\mathscr{E})$ matrices Y of interest. This will help to understand when the inversion in Eq. (2.11) fails and why a LMOCV formula can be obtained in terms of the hat-matrix.

According to [8], for two matrices, A and B, the following equation holds

$$\mathrm{im}(A+B) \subseteq \mathrm{im}(A) \oplus \mathrm{im}(B), \tag{2.15}$$

where *im* denotes the image of the operator.[5] Equation (2.15) suggests the following way to calculate the eigenvalues and -vectors of matrices of rank $\aleph(\mathscr{E})$, Y, if a dyadic decomposition of the matrix is known. The decomposition is given by

$$Y = \sum_{i\in\mathscr{E}} Y_i = \sum_{i\in\mathscr{E}} (X'X)^{-1}(X'_{i\bullet}X_{i\bullet}) = \sum_{i\in\mathscr{E}} \left((X'X)^{-1}X'_{i\bullet}\right) X_{i\bullet},$$

with $X_{i\bullet}$ being a row vector. As the matrix $(X'X)^{-1}X'_{i\bullet}X_{i\bullet}$ is of rank 1 it has only one non-zero eigenvalue. This eigenvalue is $\lambda_i = X_{i\bullet}(X'X)^{-1}X'_{i\bullet}$. The right eigenvectors is $v_i = (X'X)^{-1}X'_{i\bullet}$ while the left eigenvector is $w'_i = X_{i\bullet}$. This can be seen by direct inspection. Because of Eq. (2.15) it is clear, that the (right-hand) eigenvectors of Y are linear combinations of the (right-hand) eigenvectors of the addends Y_i. The eigenvalue equation can be written, with the constants, a_j, and the dyads, Y_i, as:

$$\sum_{i\in\mathscr{E}} Y_i \sum_{j\in\mathscr{E}} a_j v_j = \lambda \sum_{j\in\mathscr{E}} a_j v_j$$
$$\sum_{i,j\in\mathscr{E}} a_j v_i w'_i v_j = \lambda \sum_{j\in\mathscr{E}} a_j v_j.$$

When multiplying w'_k from the left side and identifying $w'_j v_i = \langle w_j, v_i\rangle$ for every $k \in \mathscr{E}$

$$\sum_{j\in\mathscr{E}} a_j \left(\sum_{i\in\mathscr{E}} \langle w_k, v_i\rangle\langle w_i, v_j\rangle - \lambda\langle w_k, v_j\rangle \right) = 0.$$

is obtained. Only non-trivial solutions are of interest. Furthermore,

$$\sum_{i\in\mathscr{E}} \langle w_k, v_i\rangle\langle w_i, v_j\rangle - \lambda\langle w_k, v_j\rangle \stackrel{!}{=} \sum_{i\in\mathscr{E}} h_{ki}h_{ij} - \lambda h_{kj}.$$

Introducing a truncated hat-matrix $H_{\mathscr{E}}$, with all entries that are in the rows and columns of the excluded set $\mathscr{E}$ ($H_{\mathscr{E}}$ is of dimension $\aleph(\mathscr{E}) \times \aleph(\mathscr{E})$), yields

$$\begin{aligned} 0 &= \det(H^2_{\mathscr{E}} - \lambda H_{\mathscr{E}}) \\ &= \det(H_{\mathscr{E}})\det(H_{\mathscr{E}} - \lambda I). \end{aligned} \tag{2.16}$$

This means, the non-zero eigenvalues of the Y-matrix are the eigenvalues of the truncated hat-matrix $H_{\mathscr{E}}$ (provided $\det(H_{\mathscr{E}}) \neq 0$)!

[5] This is closely related to Eq. (2.12).

2.3.2 Analytical Formula of LMOCV

As mentioned in Sec. 2.2.2 there is a direct approach to calculate the CV which involves N inversions of $m \times m$ matrices. In this section, I will show that there exists an analytic approach requiring N inversions of $\aleph(\mathscr{E}) \times \aleph(\mathscr{E})$ matrices, only. For the LOOCV such a formula already exists [27, 28]. It is possible to express the CV in terms of the hat-matrix

$$\mathrm{CV}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{E_i - \widehat{E_i}}{1 - h_{ii}} \right)^2 . \tag{2.17}$$

This formula can be obtained as a particular case of the analytical expression for the LMOCV, which I will derive. In Eq. (2.17), $\widehat{E}_i$ are the predictions for the whole training set which can be calculated all at once. This is in contrast to the direct approach based on Eq. (2.10), where a new regression is performed for each addend. A derivation for an analytic equation for the LMOCV is now given.[6] A more instructive derivation is presented in Appendix A.2.

The matrix Y can be written as a product of two matrices A and B, defined as follows: A is the matrix with columns consisting of the $(X'X)^{-1}X_i'$ and B is the matrix with rows consisting of the X_i, for $i \in \mathscr{E}$. That means B' and A are matrices of size $m \times \aleph(\mathscr{E})$. Thus $BA = H_{\mathscr{E}}$ and $AB = Y$. A clever use of Neumann's series can be the following:

$$\begin{aligned}
(I - AB)^{-1} &= \sum_{i=0}^{\infty} (AB)^i \\
&= I + \sum_{i=1}^{\infty} (AB)^i \\
&= I + A \cdot \sum_{i=0}^{\infty} (BA)^i \cdot B \\
&= I + A(I - BA)^{-1} B
\end{aligned} \tag{2.18}$$

This result is not limited to matrices $||AB|| < 1$, as Neumann's series requires. It can be shown, that this identity is true for all matrices, if all inverses exist.[7] This identity is useful, when $m \geq \aleph(\mathscr{E})$, because on the left side a matrix of size $m \times m$ is inverted while the right side requires to invert a matrix of size $\aleph(\mathscr{E}) \times \aleph(\mathscr{E})$. Furthermore, BA consists only of hat-matrix elements. This is another important aspect, since the hat-matrix is computed only once.

Denoting $X_{\mathscr{E}\bullet}$ as the submatrix of dimension $\aleph(\mathscr{E}) \times m$ constructed by taking all vectors $X_{i\bullet}$ with $i \in \mathscr{E}$ (the same as matrix B above), Eq. (2.18) inserted in Eq. (2.11) yields

$$J^{(\mathscr{E})} = \left(I + (X'X)^{-1} X'_{\mathscr{E}\bullet} (I - H_{\mathscr{E}})^{-1} X_{\mathscr{E}\bullet} \right) (X'X)^{-1} \left(X'E - \sum_{i \in \mathscr{E}} X_i' E_i \right) . \tag{2.19}$$

[6] As far as I know, that was not published before.

[7] By multiplying with $I - AB$ and $BA(I - BA)^{-1} = (I - BA)^{-1} - I$.

This result can be inserted into the definition of the LMOCV in Eq. (2.10) leading to

$$\begin{aligned}
\mathrm{CV}^2 &= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\aleph(\mathscr{E}_j)}\left\|E_{\mathscr{E}_j} - X_{\mathscr{E}_j}\left(I + (X'X)^{-1}X'_{\mathscr{E}_j}(I-H_{\mathscr{E}_j})^{-1}X_{\mathscr{E}_j}\right)(X'X)^{-1}\left(X'E - X'_{\mathscr{E}_j}E_{\mathscr{E}_j}\right)\right\|_2^2 \\
&= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\aleph(\mathscr{E}_j)}\left\|E_{\mathscr{E}_j} - \left(I_{\aleph(\mathscr{E}_j)} + H_{\mathscr{E}_j}(I-H_{\mathscr{E}_j})^{-1}\right)\left(H^{\mathscr{E}_j}E - H_{\mathscr{E}_j}E_{\mathscr{E}_j}\right)\right\|_2^2 \\
&= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\aleph(\mathscr{E}_j)}\left\|E_{\mathscr{E}_j} - \left(I-H_{\mathscr{E}_j}\right)^{-1}\left(H^{\mathscr{E}_j}E - H_{\mathscr{E}_j}E_{\mathscr{E}_j}\right)\right\|_2^2 \\
&= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\aleph(\mathscr{E}_j)}\left\|\left(I-H_{\mathscr{E}_j}\right)^{-1}\left(H^{\mathscr{E}_j}E - E_{\mathscr{E}_j}\right)\right\|_2^2 \\
&= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{\aleph(\mathscr{E}_j)}\left\|\left(I-H_{\mathscr{E}_j}\right)^{-1}\left(\widehat{E}_{\mathscr{E}_j} - E_{\mathscr{E}_j}\right)\right\|_2^2, \qquad (2.20)
\end{aligned}$$

where $\widehat{E}_{\mathscr{E}_j}$ is a vector, representing the predictions using the whole training set for the properties $E_{\mathscr{E}_j}$ in the excluded set $\mathscr{E}_j$. The $\widehat{E}_{\mathscr{E}_j}$ are computed only once. $H^{\mathscr{E}_j}$ denotes the matrix composed of the rows of the hat-matrix in $\mathscr{E}$.

Equation (2.20) is a natural generalization of the LOOCV formula provided by [28]. This can be obtained from Eq. (2.20) by setting $\aleph(\mathscr{E}_j) = 1$, $N = n$ and the fact that $I - H_{\mathscr{E}_j} = 1 - h_{jj}$ for $\mathscr{E}_j = \{j\}$. Moreover, Eq. (2.20) shows, that the quantities needed to obtain the CV are hat-matrix entries and the calculated energies, only. Later it will be outlined that the inversion of the Hessian matrix $X'X$, for the CV, can be skipped, by using the singular value decomposition (see Sec. 3.1.1).

2.3.3 Existence and Numerical Stability

In this section, I will analyze the requirements for the existence of the LOOCV and LMOCV and their behavior when $h_{ii} \to 1$.[8]

One important question regards the existence of a unique solution of Eq. (2.11). Such a solution exists, when the operator on the left is invertible. This is fulfilled if and only if all the eigenvalues of $I - \sum_{i\in\mathscr{E}}(X'X)^{-1}X'_iX_i$ are not zero, or all eigenvalues of $\sum_{i\in\mathscr{E}}(X'X)^{-1}X'_iX_i = Y$ are different from one.

Due to Eq. (2.18), $(I-Y)^{-1}$ exists when the truncated hat-matrix $H_{\mathscr{E}}$ has no eigenvalue equal to one. Since $H_{\mathscr{E}}$ is a submatrix of the hat-matrix it is clear that the eigenvalues cannot exceed the interval of $[0,1]$ because the hat-matrix is a projector (eigenvalues are equal to 0 or 1) and is symmetric and hermitian. Thus *Courant's Min-Max Principle* [11,29] applies, stating that hermitian matrices represented in a subspace cannot exceed their complete eigenvalue spectrum. This principle assures in quantum mechanics, that it is impossible to obtain a wavefunction that corresponds to a smaller energy than the ground state.

[8] The considerations in this section are, as far as I know, not published.

In the limit that one eigenvalue of $H_{\mathscr{E}}$ goes to 1, the inversion of the operator becomes singular. Then the $J^{(\mathscr{E})}$ are not well defined anymore, and the CV cannot be found. The behavior of the CV for this limit is now discussed, first for the LOOCV. Considering the fraction in the LOOCV formula in Eq. (2.17) above

$$E_i - \frac{\sum_{j\neq i}^{n} h_{ij} E_j}{1 - h_{ii}}, \tag{2.21}$$

one has to ask what happens if h_{ii} approaches one. According to Eq. (2.9) and [12,20]

$$h_{ii}(1 - h_{ii}) = \sum_{j\neq i}^{n} h_{ij}^2$$

holds true. As the calculated energies are independent of the h_{ij} the convergence of (2.21) requires absolute convergence (i.e. for every vector $\vec{E}$) of

$$\lim_{h_{ii}\to 1} \frac{\sum_{j\neq i}^{n} h_{ij}}{1 - h_{ii}}.$$

Writing the denominator in terms of the h_{ij} by using Eq. (2.9), and denoting $\vec{h} = \{h_{ij}\}_{j\neq i}$, it follows

$$\begin{aligned} \lim_{\|\vec{h}\|\to 0} \frac{\| \vec{h} \|_1}{\| \vec{h} \|_2^2} &\propto \lim_{h_{ii}\to 1} (1 - h_{ii})^{-1/2} \\ &\propto \lim_{h_{ii}\to 1} \mathrm{CV}. \end{aligned} \tag{2.22}$$

In a first step, it was used that all norms in $\mathbb{R}^p$, $p < \infty$ are equivalent,[9] and thus this fraction diverges as $(1 - h_{ii})^{-1/2}$, for $h_{ii} \to 1$. As this fraction diverges it becomes at some point the dominating addend in the CV, and thus the CV diverges as fast as this fraction for $h_{ii} \to 1$.

How can this be true? There is a simple explanation: In the moment, the point with $h_{ii} \lesssim 1$ has been taken out of the training set, the biggest contribution to the prediction $\widehat{E}_i = h_{ii}E_i + \sum_{j\neq i} h_{ij}E_j$ is lost. By dropping this term, also the precision of its prediction is diminished and this is reflected in a big addend to the CV, which is infinite in the absolute limit. Since also numerical precision can be lost due to the fact that the h_{ij} with $j \neq i$ become very small when $h_{ii} \lesssim 1$, (see Eq. (2.9)), this effect is even more striking in actual implementations. Ironically, if this happens, the prediction of this energy is perfect when using the whole training set ($h_{ij} \to 0 \ \ \forall j \neq i$ when $h_{ii} \to 1$). The opposite statement is, however not true, that is, having a perfect prediction does not imply $h_{ii} \to 1$. In any case, this is bad in the sense of the CV. For this to happen it is needed to have a parameter which is 'dedicated' to this configuration, in the sense that the parameter is mostly dependent on this configuration (compare Fig. 2.2). Thus one of the configurations marks a feature of the model, because the dedicated parameter and the configuration are in a one-to-one relationship. It cannot be determined with the CV as measure, whether this parameter marks an important feature. On the other hand, this gives advice

[9] That means, for each vector v in $\mathbb{R}^p$ and for all norms $||\bullet||_p$ and $||\bullet||_q$ there exist constants $c_1, c_2 > 0$ such that $c_1||v||_p < ||v||_q < c_2||v||_p$.

which structures should be added to the training set to assess the ECI of this cluster.

One should notice here, that $h_{ii} \to 1$ occurs in any case when the vectors $X_{\bullet\alpha}$ become linear dependent if structure i is excluded. In that case, the rank of X is reduced, so that the dimension of the space spanned by the $X_{\bullet\alpha}$ shrinks. This means, that there exists a rotation for the set of basis vectors $X_{\bullet\alpha}$ of that subspace, so that one of the transformed vectors $\overline{X}_{\bullet\alpha}$ is again in one-to-one relationship to the structure i.

What about the LMOCV? The expectation is, that if even more data points are excluded from the training data, things can only become worse, because there is a higher chance to remove a set of data points to which the model is sensitive. So, what happens to the LMOCV if one set $\mathscr{E}$ includes a configuration i with $h_{ii} \to 1$? In such a case, all other elements in the same row/column are suppressed with $\sqrt{1-h_{ii}}$ (as evident from Eq. (2.9)). This can be illustrated by two examples. First, the h_{ij} are only non-zero for one $j \neq i$. Then, $h_{ij} = \pm\sqrt{h_{ii}(1-h_{ii})}$. In the other extreme case, that all elements h_{ij} with $j \neq i$ are equal in magnitude, for every j: $h_{ij} = \pm\sqrt{\frac{h_{ii}}{n-1}(1-h_{ii})}$. Thus, all elements, except $h_{ii} \approx 1$, in a column/row of $H_{\mathscr{E}}$ become zero, and one eigenvalue of $H_{\mathscr{E}}$ converges to $h_{ii} \approx 1$, leading to inversion problems. Thus the LMOCV exhibits the same behavior as the LOOCV in this situation.

To assess the behavior of the LMOCV for $h_{ii} \to 1$ quantitatively, I make use of Eq. (2.20). The terms that dominate the sum in Eq. (2.20) are those which include the configuration i, as it can be shown that they diverge for $h_{ii} \to 1$. Therefore we limit the analysis to one set $\mathscr{E}$ containing i:

$$\mathrm{CV} \propto \left|\left|(I - H_{\mathscr{E}})^{-1}\left(\widehat{E}_{\mathscr{E}} - E_{\mathscr{E}}\right)\right|\right|_2 .$$

The entries of the matrix $(I - H_{\mathscr{E}})^{-1}$ can be written as $(I - H_{\mathscr{E}})^{-1}_{jk} = \mathrm{adj}(I - H_{\mathscr{E}})_{jk} / \det(I - H_{\mathscr{E}})$. Using the Laplace expansion for determinants it can be shown[10] that $\det(I - H_{\mathscr{E}}) \to 1 - h_{ii}$, $\mathrm{adj}(I - H_{\mathscr{E}})_{i,j\neq i} \to \sqrt{1-h_{ii}}$, and $\mathrm{adj}(I - H_{\mathscr{E}})_{ii}(E_i - \widehat{E}_i) \to \sqrt{1-h_{ii}}$. Thus the LMOCV behaves in the same way as the LOOCV and diverges as $(1-h_{ii})^{-1/2}$. A numerical example of this behavior is shown in Sec. 3.2.1. For a discussion of the eigenvalues of $H_{\mathscr{E}}$ see Appendix A.3.

2.4 Cross-Validation and Noise

In this section I want to analyze how the relation between the noise in the data and the CV without regularization can be put into mathematical terms. I will arrive to an expression giving a lower limit to the noise in the data in terms of the CV and the MSE. This is useful in the context of *ab-initio* calculations because it is usually difficult to estimate the numerical errors. These errors may be non-systematic and are related to the convergence criteria used in the calculations. In order to analyze this issue I will assume that the data can be represented by a model that I call the data model. The data model is defined by a finite set of parameters and must be

[10] To obtain this result one must analyze the behavior of each term expanding the i^{th} row *and* column.

distinguished from the model that predicts the data generated by the data model. The term parameter function refers in the following to the n-dimensional vector $X_{\bullet\alpha}$ associated to one parameter J_α. Since all the data is generated by the data model it will lie in the subspace spanned by the parameter functions, which define together with the parameters the data-model. This subspace has the dimension equal to the number of such parameters m. In that case, the data and the hat-matrix fulfill the following equation

$$\widehat{E} = \widehat{H}E = E.$$

That means that the data vector belongs to the subspace spanned by the parameters and is thus an eigenvector of the hat-matrix with eigenvalue 1. In such a case

$$\mathrm{CV}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{E_i - \sum_{j=1}^{n} h_{ij}E_j}{1-h_{ii}}\right)^2 = 0.$$

Introducing noise, $E_i \to E_i + \Delta E_i$, the previous equation implies:

$$\begin{aligned}\mathrm{CV}^2 &= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\Delta E_i - \sum_{j=1}^{n} h_{ij}\Delta E_j}{1-h_{ii}}\right)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{n}\frac{(\delta_{ij}-h_{ij})}{1-h_{ii}}\Delta E_j\right)^2.\end{aligned}$$

The matrix A with entries $(A)_{ij} = \frac{\delta_{ij}-h_{ij}}{1-h_{ii}}$ can now be introduced and the equation above can be recast as a matrix-vector product:

$$\begin{aligned}\mathrm{CV}^2 &= \frac{1}{n}\left|\left|A\ \Delta\vec{E}\right|\right|_2^2 \\ &= \frac{1}{n}\Delta\vec{E}'A'\ A\ \Delta\vec{E}.\end{aligned} \tag{2.23}$$

Calculating the entries of $A'\ A$ leads to

$$\begin{aligned}(A'\ A)_{ij} &= \sum_k \frac{(\delta_{kj}-h_{kj})(\delta_{ik}-h_{ik})}{(1-h_{kk})^2} \\ &= \frac{\delta_{ij}}{(1-h_{ii})^2} - h_{ij}\left(\frac{1}{(1-h_{ii})^2}+\frac{1}{(1-h_{jj})^2}\right) + \sum_k \frac{h_{ik}h_{kj}}{(1-h_{kk})^2}.\end{aligned}$$

The second line of this equation helps to realize, that the Hessian matrix of the CV^2 and $A'\ A$ are related as follows (see Appendix A.4):

$$\frac{1}{2}\mathscr{H}(\mathrm{CV}^2) = \frac{1}{n}A'\ A.$$

This last equation together with Eq. (2.23) shows that the CV^2 relates to the noise in the following form

$$\mathrm{CV}^2 \equiv \frac{1}{2}\Delta\vec{E}\ \mathscr{H}(\mathrm{CV}^2)\ \Delta\vec{E}. \tag{2.24}$$

As $\mathscr{H}(\mathrm{CV}^2)$ is symmetric, all its eigenvalues are real numbers. Furthermore, Courant's min-max principle applies, and so this equation can be replaced by an inequality, where $\Lambda[A]$ denotes the eigenvalue spectrum of A:

$$\min(\Lambda[\mathscr{H}(\mathrm{CV}^2)])\left|\left|\Delta\vec{E}\right|\right|_2^2 \ \leq 2\ \mathrm{CV}^2 \leq\ \max(\Lambda[\mathscr{H}(\mathrm{CV}^2)])\left|\left|\Delta\vec{E}\right|\right|_2^2.$$

When the data lies in the subspace spanned by these parameter functions of the model, the CV is zero. This is true for all points in this subspace. That means, that there exist vectors with eigenvalue 0 with respect to the Hessian matrix (the $X_{\bullet\alpha}$), because otherwise the CV would change by moving along this direction in space, see Eq. (2.24). Thus, the smallest eigenvalue of $\mathscr{H}(\mathrm{CV}^2)$ equals 0. It is important to notice here, that the eigenvalues of the Hessian matrix cannot become negative, because otherwise it would be possible to achieve a negative CV. A simple example of a vector related to eigenvalue 0 is $(1, 1, 1...)$, corresponding to the intercept. Thus, defining $\lambda_{max} = \frac{n}{2} \max\left(\Lambda[\mathscr{H}(\mathrm{CV}^2)]\right) = \max\left(\Lambda[A'A]\right)$:

$$\begin{aligned} \frac{n}{\lambda_{max}}\ \mathrm{CV}^2 \leq\ & \left|\left|\Delta\vec{E}\right|\right|_2^2 =: n\ \ (\Delta E)^2 \\ \Rightarrow\ \Delta E \geq\ & \frac{1}{\sqrt{\lambda_{max}}}\ \mathrm{CV}, \end{aligned} \tag{2.25}$$

where ΔE is the level of noise for each data point. As Eq. (2.25) is an inequality it is possible to interpret it, depending on which quantity is known. If the CV is known, this equation gives a strict limit to the average noise in energy for each data point (ΔE), assuming that the training set is big enough to represent all features of the data-model. The best possible model, to which this formula applies, can, for instance, be found by the smallest CV of all models. When other estimates of the *ab-initio* noise are given and yield significantly smaller values for the noise than Eq. (2.25), the assumption that the training set represents all features, is wrong. With this, Eq. (2.25) assesses whether, first the calculation precision should be raised, or the number of structures included in the training set. But the statement can be formulated in a stronger way: Assuming that the data-model is known, and the energies are exactly the 'real' values, the MSE is zero. When noise is applied, and Courant's principle is exploited, one finds

$$\begin{aligned} \mathrm{MSE} &= \frac{1}{n}\left|\left|(I-\widehat{H})\Delta\vec{E}\right|\right|_2^2 \\ &= \frac{1}{n}\Delta\vec{E}'(I-\widehat{H})^2\Delta\vec{E} \\ &= \frac{1}{n}\Delta\vec{E}'(I-\widehat{H})\Delta\vec{E} \\ &\leq \frac{1}{n}\left|\left|\Delta\vec{E}\right|\right|_2^2 = \Delta E^2 \\ &\Rightarrow \sqrt{\mathrm{MSE}} \leq \Delta E, \end{aligned} \tag{2.26}$$

where the fact that the eigenvalues of $I-\widehat{H}$ are zero or one. It is possible to compare both estimates, Eq. (2.25) and Eq. (2.26), by introducing a number α

$$\begin{aligned} \frac{1}{\lambda_{max}}\ \mathrm{CV}^2 &= \mathrm{MSE} + \alpha, \\ \frac{1}{2\lambda_{max}}\ \Delta\vec{E}'\mathscr{H}(\mathrm{CV}^2)\Delta\vec{E} &= \frac{1}{n}\Delta\vec{E}'(I-\widehat{H})\Delta\vec{E} + \alpha. \end{aligned}$$

Since the parameter functions are the same for both the MSE and the CV, the $\Delta\vec{E}$ can be separated in a part in the subspace of the data-model and one orthogonal to it, $\Delta\vec{E}_\perp$. The noise pointing in this subspace is not detected. Thus

$$\frac{n}{2\lambda_{max}}\ \Delta\vec{E}'_\perp\mathscr{H}(\mathrm{CV}^2)\Delta\vec{E}_\perp = \Delta\vec{E}'_\perp(I-\widehat{H})\Delta\vec{E}_\perp + n\alpha.$$

As on the left side the matrix $\mathscr{H}(\mathrm{CV}^2)$ is divided by its maximum eigenvalue, the positive definite matrix has only eigenvalues in the interval $[0, 1]$. $I - \widehat{H}$, on the other side, has only eigenvalues 1 in the directions perpendicular to the subspace. Thus $\alpha \leq 0$ and

$$\frac{1}{\sqrt{\lambda_{max}}}\ \mathrm{CV} \leq \sqrt{\mathrm{MSE}} \leq \Delta E. \tag{2.27}$$

Important is here, that the model, for which this formula is applied, is determined by the smallest CV and not by the smallest MSE.

Up to now, the analysis was set up to find strict inequalities between the noise in the data and the statistical quantities of the CV and MSE. Unfortunately a strict upper limit for the noise in the data cannot be found in this way. However, the nature of noise is maybe random and thus the probability to have a noise vector which points directly in the subspace of the data-model, is zero. Thus it makes sense to estimate the noise with this formalism, e.g. by using the mean of the eigenvalues of $I - \widehat{H}$, which gives (p being the number of ECI's)

$$\sqrt{\frac{n}{n-p}\ \mathrm{MSE}}\ \approx\ \Delta E.$$

It is also possible to take the distribution of the noise into account, here. This could then be used to find a more elaborate estimator for the noise in the data.

To end this chapter, I want to offer here another idea on LMOCV: Equation (2.23) is a closed-form expression of the LOOCV. It is possible to extend this by calculating the Hessian matrix of CV^2 for the LMOCV, which finally leads to an expression with a sum that is difficult to evaluate (see Appendix A.5). If it were possible to contract or approximate this sum, it would be possible to achieve the CV score for LMOCV via:

$$\mathrm{CV}^2 = \frac{1}{2}\vec{E}'\ \mathscr{H}(\mathrm{CV}^2)\ \vec{E}.$$

This method would avoid random sampling for the CV computation (see Sec. 3.1.1).

Chapter 3

Application

This chapter is arranged as follows: First, in Sec. 3.1, the singular value decomposition (SVD) is applied to least-square problems, to calculate the CV, the MSE, and the ECI's. Some computational benefits and costs of this technique are discussed. Then, in Sec. 3.2 the behavior of the CV in the situation that $h_{ii} \to 1$, is numerically investigated, and afterwards, the the algorithms for the LMOCV with and without support of Eq. (2.20) are compared with respect to runtime and numerical stability.

3.1 Singular Value Decomposition as Linear Least-Square-Solver

In this section, I will explain how to utilize the singular value decomposition [13] (SVD) for solving the linear least-square problems and why this is valuable.

3.1.1 Computation of the Hat-Matrix

In the last chapter, it was shown that the hat-matrix formalism helps to analyze instabilities that appear in the computation of the CV, e.g. for the case $h_{ii} \to 1$. As mentioned in Sec. 2.3.2, the calculation of the hat-matrix can be performed by using the SVD instead of inverting $X'X$. This is important from the point of view of the numerical implementation, because then the evaluation of the CV and the MSE are stable against quasi-singularities of the matrix $X'X$.[1] This situation is present when two columns of the matrix X become quasi linear dependent. This happens frequently because for the computation of the CV one needs to exclude rows (i.e. structures) of the matrix X.

The SVD of X can be written as [12]

$$\underset{n\times m}{X} = \underset{n\times m}{U} \quad \underset{m\times m}{\Sigma} \quad \underset{m\times m}{V'}$$

where the dimensions of the matrices are indicated below their symbol. $U'U = I_m$, m is the number of clusters, Σ contains the singular values of X, and V is an orthogonal matrix. The SVD is computationally more costly than the inversion of $X'X$, as it scales with $n^2 \times m$ for $n > m$ compared to $m^{2.4}$ [1, 7]. In the case of the cluster expansion, the numbers n and m are small (e.g. less than 100), so that for

[1] In the sense of losses due to floating point errors in the inversion of $X'X$.

this purpose, this cost is not significant. Even more valuable here is, that the SVD provides a stable way to compute $\widehat{H}$ if X is quasi-singular. According to Ref. [12]

$$\widehat{H} = U\ U'. \tag{3.1}$$

The equality (3.1) can be proved in the following way :

$$\begin{aligned}
\widehat{H} &= X\ (X'X)^{-1}\ X' \\
&= U\ \Sigma\ V'\ (V\ \Sigma'\ U'\ U\ \Sigma\ V')^{-1} V\ \Sigma'\ U' \\
&= U\ \Sigma\ V'\ (V\ \Sigma'\ \Sigma\ V')^{-1} V\ \Sigma'\ U' \\
&= U\ \Sigma\ V'\ V(\Sigma'\ \Sigma)^{-1} V'\ V\ \Sigma'\ U' \\
&= U\ \Sigma(\Sigma'\ \Sigma)^{-1}\Sigma'\ U'.
\end{aligned}$$

To arrive from here to Eq. (3.1), $\Sigma(\Sigma'\ \Sigma)^{-1}\Sigma'$ must be the identity, which is true, if all singular values of X are non-zero.

If one has to add any regularization in Eq. (2.13), we can simply concatenate the matrix X with a matrix $\widehat{R}$ such that

$$X_{new} = \begin{pmatrix} X \\ \widehat{R} \end{pmatrix}$$

with $\widehat{R}'\ \widehat{R} = R$, for Eq. (2.13). Then the SVD is performed for X_{new}. The new energy vector is then the unification of guessed values for the ECI's $J^{(0)}$ and the energies $E_{new} = \binom{E}{J^{(0)}}$. One has to take care here, that the calculation for the MSE and the CV only include the real structures. To deal with this it is possible to truncate the hat-matrix to the upper-left block of size $n \times n$, if no guess for the ECI's $J^{(0)}$ is made (see Appendix A.1).

3.1.2 Parameters and the Mean Square Error

Using the SVD and the hat-matrix it is easy to find expressions for the MSE and the ECI's. Since $\widehat{E} = \widehat{H}\ E$, where E is the property vector containing information about each structure, and $\widehat{E}$ is the prediction of the model, the MSE can be calculated as in Eq. (2.7):

$$MSE = \frac{1}{n} \left|\left|(I - \widehat{H})E\right|\right|_2^2.$$

Because Σ has no influence on the formulas for the CV or the MSE (see Eq. (3.1)), these parts of the algorithm are stable against quasi linear dependencies in $X'X$. That is not true for the ECI's, J, as known from Eq. (2.8). To see this one can insert the SVD:

$$\begin{aligned}
(X'X)^{-1}X' &= \left[(U\ \Sigma\ V')'\ U\ \Sigma\ V'\right]^{-1} (U\ \Sigma\ V')' \\
&= \left[V\ \Sigma'\ U'\ U\ \Sigma\ V'\right]^{-1}\ V\ \Sigma'\ U' \\
&= \left[V\ \Sigma^2\ V'\right]^{-1}\ V\ \Sigma\ U' \\
&= V\ \Sigma^{-2}\ V'\ V\Sigma\ U' \\
&= V\ \Sigma^{-1}\ U'
\end{aligned}$$

leading to

$$J = V\ \Sigma^{-1}\ U'\ P.$$

The ECI's are thus affected by the instability. This can be detected early, if for example one of the singular values of X becomes almost zero. This indicates that either some cluster, corresponding to the small singular value, should be removed or some structure should be added. This decision may take place, just after decomposing X by taking Σ into account.

In the cluster expansion, I often observed cases, where two clusters became quasi linearly dependent, long before the number of clusters was equal to the number of structures. The solution is to include new adapted structures[2] or leave this cluster out. The structures can be chosen by searching for a structure that lifts the linear dependency in the clusters.

Another problem may appear when $h_{ii} \to 1$ for one structure. As discussed after Eq. (2.22) this is a real analytical problem that cannot be resolved by raising the computational accuracy. Rather than discussing this problem from the analytical point of view, I want to address now the practical circumvention of this instability. The problem appears, for example, if the correlations for one cluster are only significantly different from zero for a single structure. Thus the associated ECI for this cluster would be highly affected by a change in the energy of the related structure. The ECI would, colloquially speaking, follow the structure's energy. Thus the energy of this structure is in one-to-one relationship with this ECI, becoming itself a fit parameter. If such a structure is excluded, the corresponding ECI can change a lot. The decision is then to include a new structure or leave this cluster out, again.

Most importantly, the solution of these two problems can give a clue on which new structures to add to the training set by making this dependency less striking, to assess the concerning cluster.

To finalize this section, I give consideration to an efficient numerical treatment of these problems. To calculate the LMOCV in common cases, a huge computational effort would be needed to check all possible excluded sets. To avoid this, in the `CELL` code, we use a finite number of random sets $\mathscr{E}$ and rely on the central limit theorem [2, 24, 25]. To reduce computation time it is possible to use the graphics processing unit *GPU* instead of the central processing unit *CPU*, due to the fact that the first one has huge capabilities for parallel processing [10].

3.2 Numerical Behavior of the CV-Formulas

In this section, I will address the numerical behavior of the CV for $h_{ii} \to 1$ and compare this to the analytical expectation from Sec. 2.3.3. Then the behavior of the CV and the MSE for a given model with introduced noise is studied and compared with the results from Sec. 2.4. Furthermore the numerical performance of the analytic LMOCV-formula is shown. This was implemented by me in the `CELL` package.

[2] At least two, otherwise the case $h_{ii} \to 1$ appears, because the $X_{\bullet\alpha}$ are independent only because of this structure, see Sec. 2.3.3.

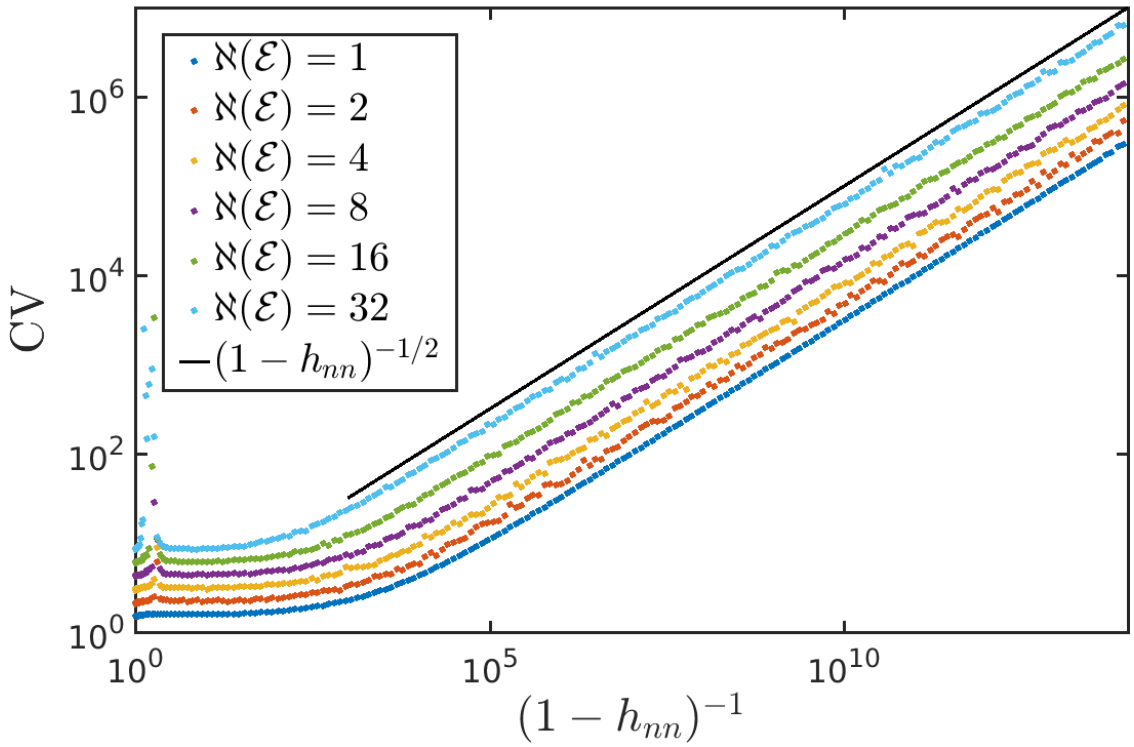

Figure 3.1: Behavior of the CV for $h_{ii} \to 1$, $i = n$. The black line denotes the expectation from Eq. (2.22). Each CV-curve was multiplied by the square root of $\aleph(\mathscr{E})$.

3.2.1 Numerical Experiments

To investigate, how the CV (for LOOCV and LMOCV) behaves if $h_{ii} \to 1$, I consider again the simple model used in Sec. 2.2.1. There, the case $P_i = J \cdot X_i$, where J is a scalar, was addressed and the entries of the hat-matrix were

$$h_{ij} = \frac{1}{\sum_k X_k^2} X_i X_j.$$

One (so-called) outer point with a large value X_n is introduced, while the other data points are within standard gaussian distribution around the origin. The size of the training set is $n = 101$. The value of X_n is then raised and the CV score is computed. The expectation for one $h_{ii} \to 1$ was discussed from an analytical viewpoint in Sec. 2.3.3. The result of the numerical experiment is shown in Fig. 3.1. This figure shows that the expectation given in Sec. 2.3.3 is fulfilled, the CV lines are parallel to the graph $\sqrt{1 - h_{ii}}$. Thus the CV diverges as $(1 - h_{ii})^{-1/2}$ for $h_{ii} \to 1$.

In order to test the expectation from Sec. 2.4, that

$$\frac{1}{\sqrt{\lambda_{max}}\Delta E} \text{ CV} \leq \frac{1}{\Delta E}\sqrt{\text{MSE}} \leq 1, \tag{3.2}$$

I used a set of parameter functions $X_{\bullet\alpha}$, with $X_{i\alpha} = f_\alpha(\sigma_i) = \sigma_i^{\alpha-1}$, to define data-models, similar to that in Sec. 2.2.2. This time I restrict α to $\{1, 2...9\}$. Then I used the following algorithm.

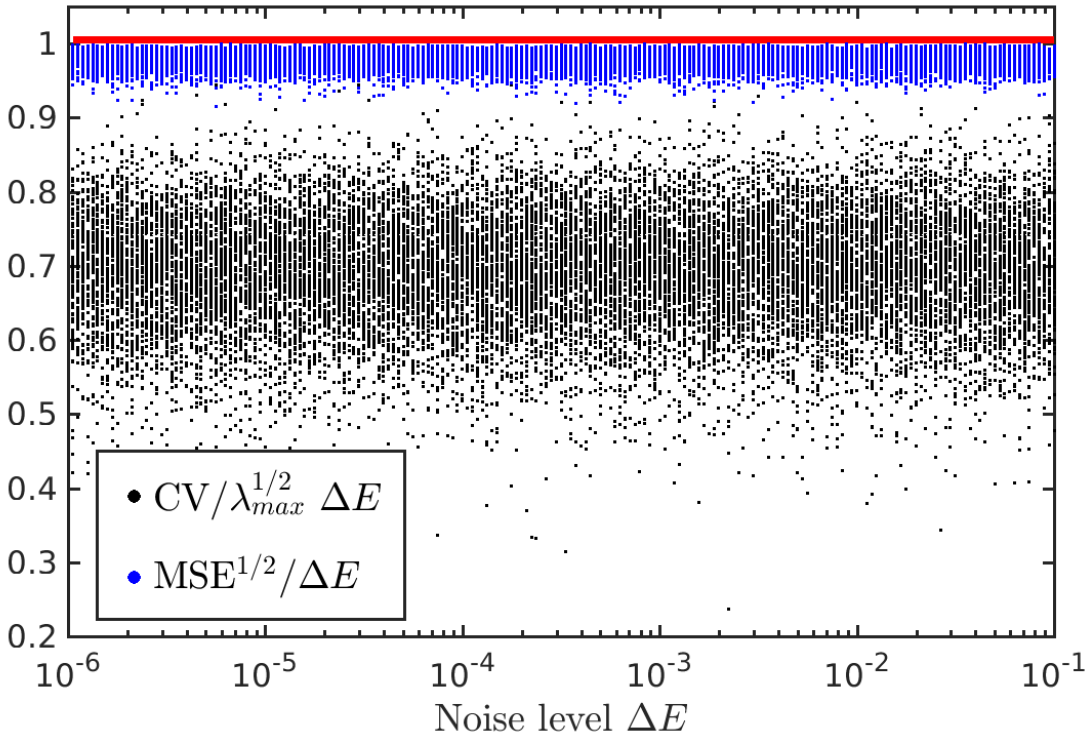

Figure 3.2: Reduced CV and MSE, normalized to the noise level ΔE. The normalized noise level is indicated by the red line. The number of parameters of the data-models is 9.

1. Uniformly distributed random parameters $J_\alpha \in [-1, 1]$ are generated to build a random data-model.

2. A set of 100 uniformly distributed random values for $\sigma_i \in [-1, 1]$ is generated.

3. The correlation matrix X is calculated and the training set is built, $P_i = \sum_\alpha X_{i\alpha} J_\alpha + \varepsilon$. ε is gaussian noise, according to the definitions in Sec. 2.4.

4. The LOOCV and MSE are computed.

A number of 200 different noise levels $\Delta E \in [10^{-6}, 0.1]$ was used. Steps 1 to 4 are repeated 100 times for each noise level and the result is shown in Fig. 3.2. Every point in this figure corresponds to a random data-model. It can be observed that Eq. (2.27) is fulfilled. The derivation of Eq. (2.27) relies on the assumption that the number of parameters m of the data-model is smaller than the number of measuring points n. As noise components lying in the subspace of the data-model neither rise the CV nor the MSE, it is clear that the larger the number of data points is, the closer are the CV and the MSE to the upper bound in Eq. (3.2). On the other hand, in a real alloy system, it is expected that the number of parameters m is much higher than the number n of *ab-initio* calculations one can perform. However, it is reasonable to expect that only a few interactions are above the noise level. For example, in the case of clathrate compounds only 8 interactions have associated ECI's larger than the *ab-initio* error [18]. Therefore, it is expected that this analysis applies also in the context of *ab-initio* calculations.

3.2.2 Performance

Last but not least, the performance of the new algorithm is addressed. How does an algorithm, built on the analytic formulas Eq. (2.17) and Eq. (2.20), help to raise the numerical performance compared to the direct approach with Eq. (2.10)? The naive expectation suggests that with Eq. (2.20) the performance does not vary significantly with the number of ECI's, when the computation time of the SVD is negligible. That is different from the direct formula Eq. (2.10) which deals with the growing Hessian matrix $X'X$. On the other hand, the analytic formulas should become expensive, when many points are excluded, while the direct approach should hardly depend on the number of excluded points. In the following, the initialization of the algorithms will not be analyzed, because even though the analytic formulas need the hat-matrix, one has to calculate it only once for all different cardinalities $\aleph(\mathscr{E})$.[3]

In Figs. 3.3 and 3.4 the computation time for one single excluded set $\mathscr{E}$ for the CV is shown. The training set consists of clathrate structures with energies calculated by the effective medium theory calculator [4] (EMT). In Fig. 3.3, the number of excluded structures is varied while in Fig. 3.4 the number of clusters is the variable. The algorithms are programmed in `CELL` using `Python 2.7.1` with the function `inv` for matrix inversion from the `linalg`-package of the `numpy` extension. This relies on `LAPACK` which uses an algorithm for matrix inversion, based on the Coppersmith-Winograd algorithm (CW algorithm). This algorithm has a computational complexity of $\mathscr{O}(n^{2.376})$ [6]. It is expected that for large m and $\aleph(\mathscr{E})$ the matrix inversion dominates the computation time, leading to a complexity of the order of the CW algorithm. Figures 3.3 and 3.4 show, both the analytical approach for a raising number of excluded structures and the direct approach for a varying number of clusters, scale as the CW algorithm for large numbers.[4]

The programmed analytic algorithm was validated on 25 training sets for various numbers of structures (5 to 51, calculated with the EMT calculator), with and without regularization, shows no instabilities and yields the same values for the CV as the direct approach, whenever it is stable.

One should notice here, that the computation of the CV[5] and the ECI's with the analytic approach can elapse in less than 2 milliseconds for one set of clusters, if the number of clusters and structures included do not exceed 50 and no singular value of X is nearby zero. Also here *GPU*-processing seems promising. The singular value decomposition of X is here the most expensive part of the algorithm. That will be helpful in order to raise the number of cluster sets that are analyzed, increasing the chance of finding a model with an optimal predictive performance.

[3] If the change in the prefactor of the regularization is ignored.

[4] The addressed problem can also be interpreted as a linear equation $Ax = b$ and solved using other algorithms.

[5] Restricted to LOOCV.

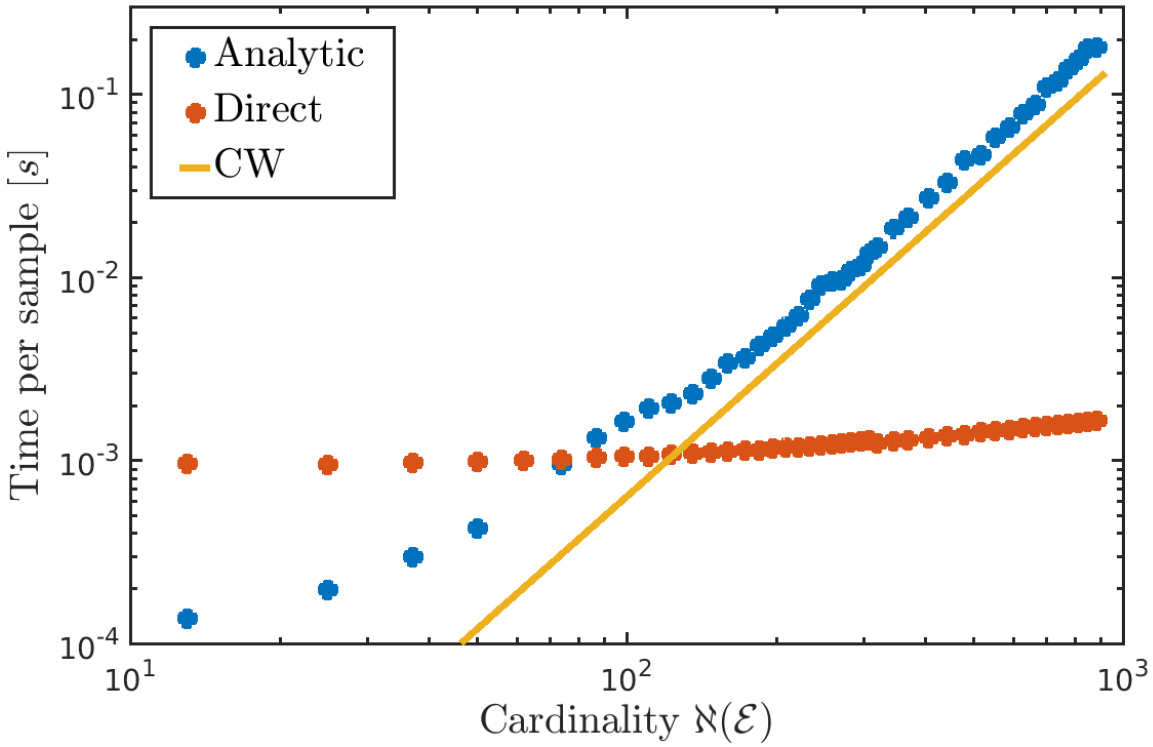

Figure 3.3: Performance of the direct ('Old') and the analytical ('New') algorithms for $N_s = 1232$ structures with up to 924 excluded structures. 14 clusters are used here. The yellow line indicates the complexity of the CW algorithm.

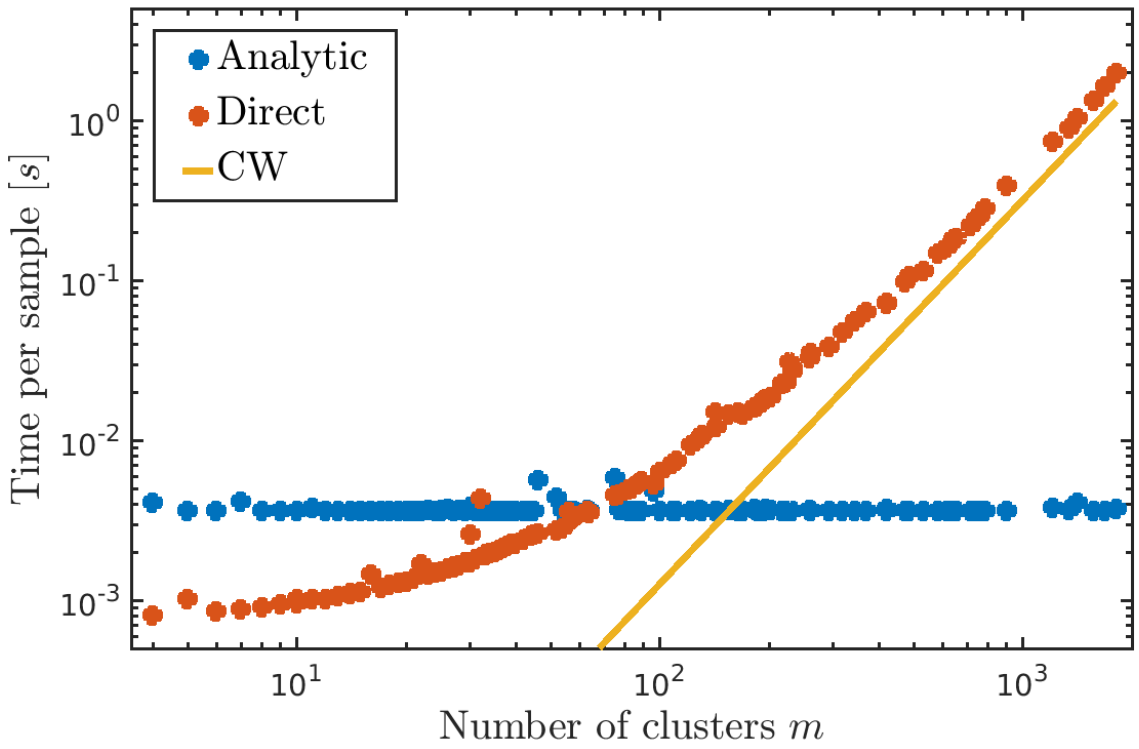

Figure 3.4: Performance of the algorithms for up to $N_c = 1814$ clusters. 117 structures are excluded from the training set.

Conclusions

The main goals of my bachelor thesis were:

1. To understand and circumvent instabilities in the computation of the CV, especially in the case of $h_{ii} \to 1$ and for (quasi-)singular Hessian matrices $X'X$,

2. to derive a LMOCV-formula, comparable to Eq. (2.17), and to extend it for situations when regularization terms are added,

3. and to investigate the relationship between the CV and noise in the data.

I have achieved all these goals. The calculation of the CV and the MSE is, thanks to the SVD as solving technique, stable against quasi singularities of the Hessian matrix. The calculation of the parameters (or ECI's) cannot be, but it is possible to foresee the instability and take sufficient measures to avoid it. A sufficient measure is the exclusion of one of two linear-dependent clusters (used here). Another option would be the calculation of new adapted structures, which can be found by random sampling.

The other unstable case is special in the sense, that a certain cluster cannot be assessed. This situation can take place, when the cluster has a significant correlation $\langle \prod_{i \in \beta(\alpha)} \sigma_i \rangle$ with only one structure. It is explained and shown, how the CV behaves in such a case and that this is an analytical problem, that cannot be resolved by raising the computational precision. In this thesis, the exclusion of the respective cluster is used to circumvent this problem. Again, calculating a new adapted structure would be possible.

A LMOCV-formula, equivalent to Eq. (2.17) has been derived, with simple incorporation of ℓ^2-regularization.

A mathematically strict relation between the minimum in calculation noise and the CV was proven, with the assumption that the training set is sufficient to model the interactions above the noise level. This can be used as a benchmark whether the assumption is true, by comparing with other noise estimates. An argumentation was given, why the CV without regularization can never yield an upper limit for the noise in the data.

To the best of my knowledge, both, the analytical formula for the LMOCV and the relation between noise and the CV were not known before.

All the theoretical framework was programmed in `Python` and can be used in the `CELL`-package of the SOL-group for cluster expansion.

Bibliography

[1] Brand, Matthew: Fast low-rank modifications of the thin singular value decomposition. In: Linear algebra and its applications 415 (2006), Nr. 1, S. 20–30

[2] Büchter, Andreas ; Henn, Hans-Wolfgang: Elementare Stochastik: Eine Einführung in die Mathematik der Daten und des Zufalls. 2nd. Heidelberger Platz 3, 14197 Berlin, Germany : Springer-Verlag Berlin Heidelberg, 2007. – 370 S. "http://www.springer.com/de/book/9783540453819". – ISBN 978–3–540–45382–6

[3] Candes, Emmanuel ; Romberg, Justin: Sparsity and Incoherence in Compressive Sensing. In: IEEE Transactions and Informations theory 52 (2006), Nov., Nr. 4, S. 489–509

[4] Choy, T.C.: Effective Medium Theory: Principles and Applications. Clarendon Press, 1999 (International series of monographs on physics). https://books.google.de/books?id=SK_Jn3YwAu4C. – ISBN 9780198518921

[5] Connolly, J. W. D. ; Williams, A. R.: Density-functional theory applied to phase transformations in transition-metal alloys. In: Phys. Rev. B 27 (1983), Apr, 5169–5172. http://dx.doi.org/10.1103/PhysRevB.27.5169. – DOI 10.1103/PhysRevB.27.5169

[6] Coppersmith, Don ; Winograd, Shmuel: Computational algebraic complexity editorial Matrix multiplication via arithmetic progressions. In: Journal of Symbolic Computation 9 (1990), Nr. 3, 251 - 280. http://dx.doi.org/http://dx.doi.org/10.1016/S0747-7171(08)80013-2. – DOI http://dx.doi.org/10.1016/S0747–7171(08)80013–2. – ISSN 0747–7171

[7] Davie, A. M. ; Stothers, A. J.: Improved bound for complexity of matrix multiplication. In: Proceedings of the Royal Society of Edinburgh, Section: A Mathematics 143 (2013), 4, 351–369. http://dx.doi.org/10.1017/S0308210511001648. – DOI 10.1017/S0308210511001648. – ISSN 1473–7124

[8] Fischer, Gerd: Lineare Algebra, Eine Einführung für Studienanfänger. 17th. Abraham-Lincoln-Straße 46, 65189 Wiesbaden, Germany : Springer-Vieweg, 2010. – 148–267 S. "http://link.springer.com/book/10.1007/978-3-8348-9365-9". – ISBN 978–3–8348–0996–4. – It was extremely helpful for the theoretical fundament of this work.

[9] In: Fontaine, Didier de: Alloy phase stability. Berlin, Heidelberg : Springer Berlin Heidelberg, 1987. – ISBN 978–3–540–47757–0, 410–430

[10] Green, Todd: GPU Computing Gems Jade Edition. 1st. 225 Wyman Street, Waltham, MA 02451, USA : Morgan Kaufmann, 2011. – XV–XVI S. – ISBN 978–0–1238–5963–1

[11] Heuser, Harro: Funktionalanalysis - Theorie und Anwendung. 4th. Abraham-Lincoln-Straße 46, 65189 Wiesbaden, Germany : Vieweg+Teubner Verlag, 2006. – 310 S. "http://www.springer.com/us/book/9783835100268". – ISBN 978–3–8351–0026–8

[12] Hoaglin, David C. ; Welsch, Roy E.: The Hat Matrix in Regression and ANOVA. In: The American Statician 32 (1978), Feb., Nr. 1, S. 17–22

[13] Jr., Richard L. B.: The Singular Value Decomposition - Scientific Data Analysis. 1st. 11 W 42nd St, New York, NY 10036, USA : Springer New York, 1990. – 199–232 S. "http://link.springer.com/chapter/10.10072F978-1-4612-3362-6_8". – ISBN 978–1–4612–3362–6

[14] Koecher, Max: Lineare Algebra und analytische Geometrie. 4th. Heidelberger Platz 3, 14197 Berlin, Germany : Springer-Verlag Berlin Heidelberg, 2003. – 41 S. "http://www.springer.com/de/book/9783540629030". – ISBN 978–3–540–62903–0

[15] Kohavi, Ron: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, Morgan Kaufmann, 1995, S. 1137–1143

[16] Laks, David B. ; Ferreira, L. G. ; Froyen, Sverre ; Zunger, Alex: Efficient cluster expansion for substitutional systems. In: Phys. Rev. B 46 (1992), Nov, 12587–12605. http://dx.doi.org/10.1103/PhysRevB.46.12587. – DOI 10.1103/PhysRevB.46.12587

[17] Li, Ker-Chau: Asymptotic Optimality for C_p,C_L, Cross-Validation and Generalized Cross-Validation: Discrete Index Set. In: The Annals of Statistic 54 (1987), Feb., Nr. 4, S. 958–975

[18] Maria Troppenz, Claudia D. Santiago Rigamonti R. Santiago Rigamonti: Thermoelectric clathrates Ba8AlxSi46-x and Sr8AlxSi46-x: Ground-state configurations and electronic properties ising an iterative cluster expansion technique. 2016

[19] Miller, Kenneth S.: On the Inverse of Matrix Sums. In: Mathematics Magazine 54 (1981), Mar., Nr. 2, S. 67–72

[20] Mohammadi, Mohammad: On the Bounds for Diagonal and Off-Diagonal Elements of the Hat Matrix in the Linear Regressinn Model. In: REVSTAT ? Statistical Journal 14 (2016), Feb., Nr. 1, S. 75–?87

[21] Nelson, Lance J. ; Hart, Gus L. W.: Compressive Sensing as a paradigm for building physics models. In: Physical Review B 87 23 (2013), Feb., Nr. 4

[22] Richard R. Picard, R. Dennis C.: Cross-Validation of Regression Models. In: Journal of the American Statistical Association 79 (1984), Nr. 387, 575-583. http://www.jstor.org/stable/2288403. – ISSN 01621459

[23] Schmidt, Mark: Least Squares Optimization with L1-Norm Regularization. In: CS542B Project Report (2005)

[24] Schwabl, Franz: Statistical Mechanics. 2nd. Heidelberger Platz 3, 14197 Berlin, Germany : Springer-Verlag Berlin Heidelberg, 2006. – 7–9 S. "http://www.springer.com/jp/book/9783540323433". – ISBN 978–3–540–36217–3

[25] Steland, Ansgar: Basiswissen Statistik - Kompaktkurs für Anwender aus Wirtschaft, Informatik und Technik. 1st. Heidelberger Platz 3, 14197 Berlin, Germany : Springer-Verlag Berlin Heidelberg, 2007. – 122 S. "http://www.springer.com/de/book/9783540742067". – ISBN 978–3–540–74206–7

[26] Stone, M.: Cross-validatory choice and assessment of statistical predictions. In: J R Stat. Soc. B Stat Methodol. 27 (1974), Apr, 111–147. http://www.oalib.com/references/5262044

[27] Stone, M.: Cross-validation:a review. In: Series Statistics 9 (1978), Nr. 1, 127-139. http://dx.doi.org/10.1080/02331887808801414. – DOI 10.1080/02331887808801414

[28] Walle, A. van d. ; Ceder, G.: Automating First-Principles Phase Diagram Calculations. In: Journal of Phase Equilibria 23 (2002), Aug., Nr. 4

[29] Werner, Dirk: Funktionalanalysis. 7th. Heidelberger Platz 3, 14197 Berlin, Germany : Springer-Verlag Berlin Heidelberg, 2011. – 112–126 S. "http://www.springer.com/us/book/9783642210167". – ISBN 978–3–642–21017–4

List of Figures

Appendix A

Appendix

A.1 Optimized ECI's for MSE and Cost Function

This section is arranged a follows: A formula for the ECI's for an optimized MSE with excluded set $\mathscr{E}$ is derived, and with this the ECI's and the MSE of the whole training set are achieved. Then a Cost Function with ℓ^2-regularization term will be examined. After each derivation short considerations for the LOOCV are given.

The training set consists of n points. The mean square error $\mathrm{MSE}_{\mathscr{E}}$ of a rest set, with an excluded set of points $\mathscr{E}$ with cardinality $\aleph(\mathscr{E})$ is defined as

$$\mathrm{MSE}_{\mathscr{E}} = \frac{1}{n-\aleph(\mathscr{E})} \cdot \left[\sum_{s=1}^{n} \left(E_s - \sum_{\alpha} X_{s,\alpha} J_\alpha \right)^2 - \sum_{i \in \mathscr{E}} \left(E_i - \sum_{\alpha} X_{i,\alpha} J_\alpha \right)^2 \right].$$

Taking the gradient with respect to the J (to minimize the $\mathrm{MSE}_{\mathscr{E}}$), yields

$$\nabla_{J_\beta} \mathrm{MSE}^2_{\mathscr{E}} = \frac{-2}{n-\aleph(\mathscr{E})} \cdot \left[\sum_{s=1}^{n} X_{s,\beta} \left(E_s - \sum_{\alpha} X_{s,\alpha} J^{(\mathscr{E})}_\alpha \right) - \sum_{i \in \mathscr{E}} X_{i,\beta} \left(E_i - \sum_{\alpha} X_{i,\alpha} J^{(\mathscr{E})}_\alpha \right) \right] = 0.$$

One can write this equation in matrix form (it is valid for every β):

$$X'E - X'X J^{(\mathscr{E})} = \sum_{i \in \mathscr{E}} \left(X_i' E_i - X_i' X_i J^{(\mathscr{E})} \right),$$

where $'$ denotes a transposition. Assuming that $\det(X'X) \neq 0$:

$$\left(I - (X'X)^{-1} \sum_{i \in \mathscr{E}} X_i' X_i \right) J^{(\mathscr{E})} = (X'X)^{-1} \left(X'E - \sum_{i \in \mathscr{E}} X_i' E_i \right). \tag{A.1}$$

In the case that no structure is left out, all excluded rows X_i can be set to zero, yielding:

$$J = (X'X)^{-1} X'E.$$

In Eq. (A.1) we see, that anytime when one of the $\aleph(\mathscr{E})$ non-zero eigenvalues of $(X'X)^{-1} \sum_{i \in \mathscr{E}} X_i' X_i =: Y$ becomes one, the inversion should fail, numerically it fails for all values nearby one.

Inserting this into the definition of the cross-validation for LOOCV, a well known formula [28] is achieved ($\mathscr{E} = \{i\}$):

$$\mathrm{CV}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{E_i - X_i(X'X)^{-1}X'E}{1 - X_i(X'X)^{-1}X_i'}\right)^2.$$

Consider the case when the number of generated ECI's shall be bigger than the number of structures in the training set, e.g. to avoid too high computational costs. Then regularization needs to be applied to the Hessian matrix because otherwise dependencies between the ECI's make any computational approach infeasible. One can define a cost function

$$C(J) = \mathrm{MSE}^2_{\mathscr{E}}(J) + \varphi(J),$$

where φ is the added regularization. The same derivation as above yields, when minimizing the cost function, $\nabla_J C(J^{\mathscr{E}}_{\varphi}) = 0$,

$$X'E - \sum_{i\in\mathscr{E}} X_i'E_i - \left(X'X - \sum_{i\in\mathscr{E}} X_i'X_i\right) J^{(\mathscr{E})}_{\varphi} = \frac{n - \aleph(\mathscr{E})}{2}\nabla_J\varphi(J^{(\mathscr{E})}_{\varphi}).$$

Here, a form of φ as only ℓ^2-regularization is proposed. Suppose R is an arbitrary matrix and J_0 is a vector of guesses for the ECI's, then

$$\frac{n - \aleph(\mathscr{E})}{2}\varphi(J) = (J - J_0)'R(J - J_0)$$

Choosing R to be symmetric (without loss of generality):

$$X'E - \sum_{i\in\mathscr{E}} X_i'E_i + RJ_0 = \left(X'X + R - \sum_{i\in\mathscr{E}} X_i'X_i\right) J^{(\mathscr{E})}_{\varphi}, \tag{A.2}$$

which is the result for the ECI's. At this point $\det(X'X + R) \neq 0$ needs to be assured, but since R is arbitrary, that should not cause a problem. In the special case of LOOCV we get for $J^{(i)}_{\varphi}$

$$J^{(i)}_{\varphi} = \left(I + \frac{1}{1 - X_i(X'X + R)^{-1}X_i'}(X'X + R)^{-1}X_i'X_i\right)(X'X + R)^{-1}(X'E - X_i'E_i + RJ_0). \tag{A.3}$$

This leads to a regularized LOOCV expression as follows

$$\mathrm{CV}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{E_i - X_i(X'X + R)^{-1}(X'E + RJ_0)}{1 - X_i(X'X + R)^{-1}X_i'}\right)^2.$$

So in this case

$$|1 - X_i(X'X + R)^{-1}X_i'| \neq 0,$$

must be true. In the case of LMOCV none of the $\aleph(\mathscr{E})$ non-zero eigenvalues of the matrix

$$(X'X + R)^{-1}\sum_{i\in\mathscr{E}} X_i'X_i$$

can become one, if the inversion can be performed.

A.2 The LMOCV-Formula

I recall here Eqs. (2.11) or (A.2):

$$\left(I - \sum_{i\in\mathscr{E}} (X'X)^{-1} X_i' X_i\right) J^{(\mathscr{E})} = (X'X)^{-1} \left(X'E - \sum_{i\in\mathscr{E}} X_i' E_i\right).$$

What could be useful to invert this equation? The only thing that gives hope is, that the operator $\sum_{i\in\mathscr{E}} (X'X)^{-1} X_i' X_i =: Y$ has a rank less or equal to $\aleph(\mathscr{E})$ due to the fact that [8]

$$\begin{aligned} rank(A+B) &\leq rank(A) + rank(B) \\ rank(AB) &\leq \min\{rank(A), rank(B)\}. \end{aligned}$$

How to exploit this? In Ref. [19] the following way is suggested, if $I-Y$ is invertible. Since the matrix Y has just rank $\aleph(\mathscr{E})$ an algorithm can be used, described by K.S. Miller in [19]: There needs to exist a dyadic decomposition $\sum_{i\in\mathscr{E}} \overline{Y}_i$ of Y and an ordered set of sets $\mathscr{K}_j$ with raising cardinality up to $\aleph(\mathscr{E})$ containing a certain number of indices $i \in \mathscr{E}$, such that any sum of dyadic products:

$$I - \sum_{i\in\mathscr{K}_j} \overline{Y}_i,$$

$j = 1...\aleph(\mathscr{E})$, is invertible. Thus we can apply the following algorithm ($C_{j+1} = I - \sum_{i\in\mathscr{K}_j} \overline{Y}_i$) :

$$C_{j+1}^{-1} = C_j^{-1} + \frac{1}{1 - tr(C_j \overline{Y_j})} \cdot C_j^{-1} \overline{Y_j} C_j^{-1}. \tag{A.4}$$

Using this formula in Eq. (2.11) yields an analytic expressions for $J^{(\mathscr{E})}$ for every cardinality of $\mathscr{E}$, and thus a formula for the leave-many-out CV. Because the upper iterative equation cannot be evaluated by hand for more than $\aleph(\mathscr{E}) > 3$, I wrote a `Form`-script[1] to control the resulting expressions. This further information then inspired me to find a closed form for the LMOCV.

The reader has to recall here Sec. 2.3.1. Think about: what could be a formula for the LMOCV? How should it behave? It should become singular when the operator $I - Y$ becomes singular. The only way to assure this is to divide it by the determinant of the endomorphism $I - Y$ - or by $\det(I - H_{\mathscr{E}})$, see Eq. (2.16).

Which other expressions should enter the formula? Decomposing the Y_i into there eigenvectors yields (abbreviated): $Y_i = l_i r_i$. (This is a convention, l_i is a right eigenvector.) Recall now that $r_i l_j = h_{ij}$. So, the only things that enter are dyadic products $l_i r_j$ and the hat-matrix entries.

Finally we know that we are interested in an inverse matrix. With the inspiration of the formulas of the first few orders using the *Form*-script, the following approach seems to be promising:

$$\left(I - \sum_{i\in\mathscr{E}} l_i r_i\right)^{-1} \stackrel{?}{=} I + \sum_{j,k\in\mathscr{E}} \frac{(-1)^{j+k} \det(I - H_{\mathscr{E}})_{j,k}}{\det(I - H_{\mathscr{E}})} l_j r_k. \tag{A.5a}$$

[1] `Form` is a symbolic manipulation system.

Multiplying Eq. (A.5a) with $I - \sum_{i\in\mathscr{E}} l_i r_i$

$$(I - \sum_{i\in\mathscr{E}} l_i r_i)(I + \sum_{j,k\in\mathscr{E}} \frac{(-1)^{j+k}\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} l_j r_k) =$$
$$I - \sum_{i\in\mathscr{E}} l_i r_i + \sum_{j,k\in\mathscr{E}} \frac{(-1)^{j+k}\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} l_j r_k - \sum_{i,j,k\in\mathscr{E}} \frac{(-1)^{j+k}\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} l_i h_{ij} r_k.$$

and taking the last two terms together gives

$$\sum_{i,j,k\in\mathscr{E}} \frac{(-1)^{j+k}(\delta_{ij}-h_{ij})\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} l_i r_k.$$

Comparing the remaining terms yields then, that the following is needed:

$$\sum_{j\in\mathscr{E}} (\delta_{ij}-h_{ij})(-1)^{j+k}\frac{\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} = \delta_{ik}.$$

The last statement is true because of a well-known representation of inverse matrices [8]. So, the assumption is true:

$$\left(I - \sum_{i\in\mathscr{E}} l_i r_i\right)^{-1} \stackrel{!}{=} I + \sum_{j,k\in\mathscr{E}} \frac{(-1)^{j+k}\det(I-H_{\mathscr{E}})_{j,k}}{\det(I-H_{\mathscr{E}})} l_j r_k.$$

Inserting this in Eq. (2.10) one arrives at Eq. (2.20). Regularization terms and guesses for the ECI's can here be introduced by concatenating the correlation matrix with a matrix $\widehat{R}$ that fulfills $\widehat{R}'\widehat{R} = R$ in

$$X_{new} = \begin{pmatrix} X \\ \widehat{R} \end{pmatrix}.$$

Then, the guesses for the ECI's J_0 can be merged to the vector of the energies E: $E_{new} = \binom{E}{J_0}$.

A.3 Eigenvalues of $H_{\mathscr{E}}$

When $h_{ii} \to 1$ is fulfilled, it is of interest, how fast the highest eigenvalue of the submatrix $H_{\mathscr{E}}$ ν converges to one.[2] To see this, the eigenvalue equation of $H_{\mathscr{E}}$, $\det(H_{\mathscr{E}} - \nu I) = 0$, can be expanded in the i^{th} row and column, via Laplace expansion

$$\begin{aligned} 0 &= (h_{ii}-\nu)\ \mathrm{adj}(H_{\mathscr{E}}-\nu I)_{ii} + \sum_{j\in\mathscr{E}\setminus\{i\}} h_{ij}\ \mathrm{adj}(H_{\mathscr{E}}-\nu I)_{ij} \\ &= (h_{ii}-\nu)\ \mathrm{adj}(H_{\mathscr{E}}-\nu I)_{ii} + \sum_{j\in\mathscr{E}\setminus\{i\}} h_{ij} \sum_{k\in\mathscr{E}\setminus\{i,j\}} h_{ik}\ \mathrm{adj}(H_{\mathscr{E}\setminus\{i\}}-\nu I)_{jk}. \end{aligned}$$

As known from Sec. 2.3.3, the h_{ij} converge to zero as the square root of $h_{ii} \to 1$. The number of contributing terms to the sum on the right is $(\aleph(\mathscr{E})-1)(\aleph(\mathscr{E})-2)$ multiplied with $h_{ij}h_{ik}$. Assuming that not all the h_{ij}'s have the same sign, it is possible to write their sum as its possible maximum, multiplied by a factor $\alpha < 1$,

[2] Notice, that due to Courant's min-max principle, $1 \geq \nu \geq \max(h_{ii})$.

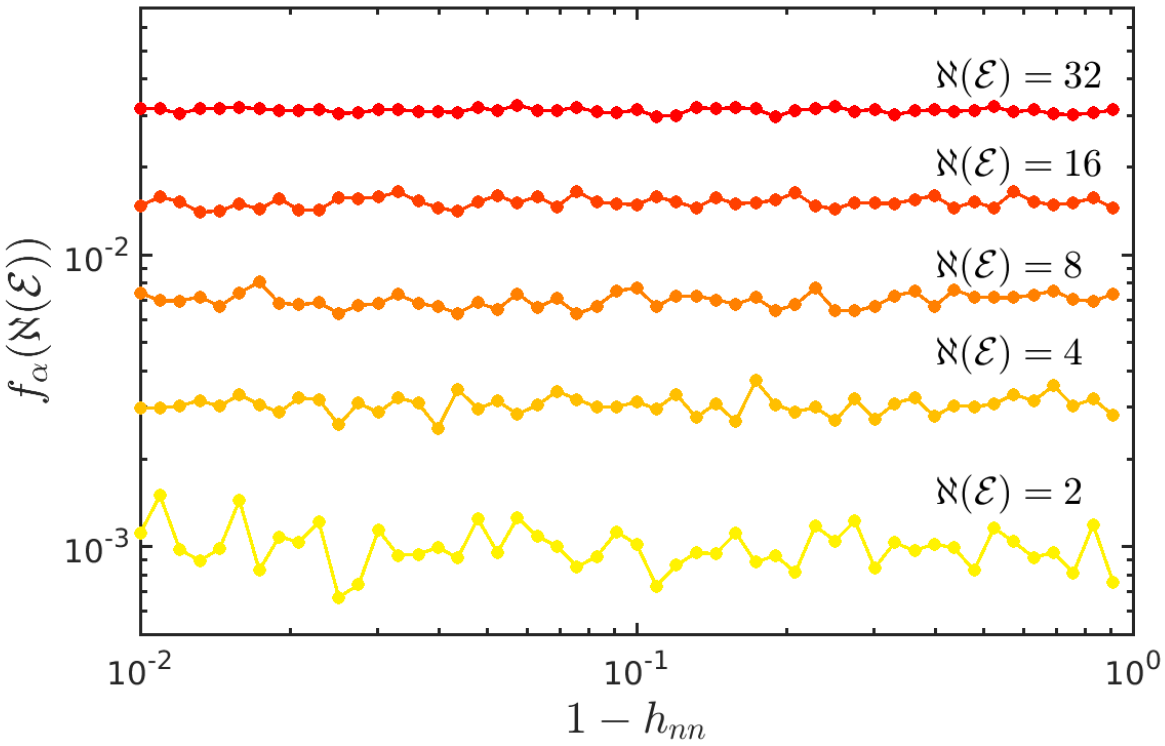

Figure A.1: $f_\alpha(\aleph(\mathscr{E}))$ versus $\aleph(\mathscr{E})$ (averaged 100 test submatrices). Set-up as in Figs. 3.1 and 2.2: one of 100 data-point corresponds to a $h_{ii} \approx 1$ and one parameter was introduced to the model.

compare Eq. (2.9). The adjoints which do not include the i^{th} row or column can be considered as constants ($\approx \beta$), as there is no indication that they are rapidly varying functions for $h_{ii} \to 1$. Then

$$0 = (h_{ii} - \nu)\beta + \alpha(\aleph(\mathscr{E}) - 1)(\aleph(\mathscr{E}) - 2)\frac{1 - h_{ii}}{\aleph(\mathscr{E})}\beta$$
$$\Rightarrow \frac{\nu - h_{ii}}{1 - h_{ii}} = \frac{\alpha}{\aleph(\mathscr{E})}(\aleph(\mathscr{E}) - 1)(\aleph(\mathscr{E}) - 2) =: f_\alpha(\aleph(\mathscr{E})). \tag{A.6}$$

This is supported by numerical experiments, see Fig. A.1. One should notice here, that for certain α this difference can become important, such that there is a difference in the behavior of the LOOCV and the LMOCV. I expect this only to happen if two or more points are special, in the sense that they are 'far' from the origin, see Fig. 2.2.

A.4 Hessian Matrix of the CV^2

Calculating the Hessian matrix of CV^2 for the analytic LOOCV by taking the with respect to energy yields

$$\begin{aligned}\frac{\partial}{\partial E_k}\mathrm{CV}^2 &= \frac{2}{n}\sum_i \left(\frac{E_i - \sum_j h_{ij}E_j}{1-h_{ii}}\right)\frac{\partial}{\partial E_k}\left(\frac{E_i - \sum_j h_{ij}E_j}{1-h_{ii}}\right)\\ &= \frac{2}{n}\sum_i \left(\frac{E_i - \sum_j h_{ij}E_j}{1-h_{ii}}\right)\left(\frac{\delta_{ik} - h_{ik}}{1-h_{ii}}\right)\\ &= \frac{2}{n}\left(\frac{E_k - \sum_j h_{kj}E_j}{(1-h_{kk})^2}\right) - \sum_i h_{ki}\left(\frac{E_i - \sum_j h_{ij}E_j}{(1-h_{ii})^2}\right)\end{aligned}$$

$\mathscr{H}_{kl}(\mathrm{CV}^2)$ is the (k,l)-element in the Hessian of the CV^2:

$$\begin{aligned}\left(\mathscr{H}(\mathrm{CV}^2)\right)_{kl} &= \frac{\partial}{\partial E_l}\frac{\partial}{\partial E_k}\mathrm{CV}^2\\ &= \frac{2}{n}\frac{\partial}{\partial E_l}\left(\frac{E_k - \sum_j h_{kj}E_j}{(1-h_{kk})^2}\right) - \frac{2}{n}\frac{\partial}{\partial E_l}\sum_i h_{ki}\left(\frac{E_i - \sum_j h_{ij}E_j}{(1-h_{ii})^2}\right)\\ &= \frac{2}{n}\left(\frac{\delta_{kl} - h_{kl}}{(1-h_{kk})^2}\right) - \frac{2}{n}h_{kl}\left(\frac{1-\sum_j h_{lj}}{(1-h_{ll})^2}\right)\\ &= \frac{2}{n}\left[\frac{\delta_{kl}}{(1-h_{kk})^2} - h_{kl}\left(\frac{1}{(1-h_{ll})^2} + \frac{1}{(1-h_{kk})^2}\right) + \sum_i \frac{h_{ki}h_{il}}{(1-h_{ii})^2}\right].\end{aligned}$$

This compared with $A'\ A$ yields

$$\frac{1}{2}\ \mathscr{H}(\mathrm{CV}^2) = \frac{1}{n}\ A'\ A.$$

A.5 Hessian Matrix for LMOCV

Eq. (2.23) is a closed form expression of the LOOCV. It is possible to extend this by calculating the Hessian matrix of the CV^2 for LMOCV, which finally leads to the following expression (m points are left out):

$$\begin{aligned}&\mathscr{H}(CV^2)_{pq} = \frac{2}{\binom{n}{m}}\sum_{1\le i_1\le\ldots\le i_m\le n}\ \sum_{k,l\in\{i_1\ldots i_m\}}(\delta_{pl}-h_{pl})((I-H_{\mathscr{E}})^{-2})_{kl}(\delta_{kq}-h_{kq})\\ &= \frac{2}{\binom{n}{m}}\sum_{1\le i_1\le\ldots\le i_m\le n}\ \sum_{j,k,l\in\{i_1\ldots i_m\}}(\delta_{pl}-h_{pl})(\delta_{kq}-h_{kq})\cdot\\ &\frac{\sum_{r_2\ldots r_m,s_2\ldots s_m}\epsilon_{j,r_2\ldots r_m}\epsilon_{k,s_2\ldots s_m}\prod_{t=2}^m(\delta_{r_ts_t}-h_{r_ts_t})\sum_{v_2\ldots v_m,u_2\ldots u_m}\epsilon_{j,v_2\ldots v_m}\epsilon_{l,u_2\ldots u_m}\prod_{w=2}^m(\delta_{v_wu_w}-h_{v_wu_w})}{(\sum_{d_1\ldots d_m,e_1\ldots e_m}\epsilon_{d_1\ldots d_m}\epsilon_{e_1\ldots e_m}\prod_{x=1}^m(\delta_{d_xe_x}-h_{d_xe_x}))^2}.\end{aligned}$$

The summation indices in the sums on the fraction are bounded to $\mathscr{E} = \{i_1\ldots i_m\}$ (where in the Levi-Cevita tensor i_k is used as k, $k\in\{1\ldots m\}$).

Acknowledgement

I want to address special thanks to my academic advisor and first evaluator, Prof. Dr. Dr. h.c. Claudia Draxl, for all her help, suggestions and tips and all the time she invested, starting from finding the right topic for me up to revising this text.

Thanks also to Maria Troppenz, for the *ab-initio* data, I used for illustrations in my text.

Furthermore, I would particularly thank my tutor, Dr. Santiago Rigamonti, with whom I had many inspiring talks about the topic and who helped me a lot with instructions and explanations in this work. Especially in the application for the `CELL`-package.

Humboldt-Universität zu Berlin
Philosophische Fakultät II

Name: …….. Vorname: …….

Matrikelnummer:…….

Eidesstattliche Erklärung zur

☐	**Hausarbeit**	☐	**Take Home-Klausur**
☐	**Bachelorarbeit**	☐	**Diplomarbeit**
☐	**Masterarbeit**	☐	**Magisterarbeit**

Ich erkläre ausdrücklich, dass es sich bei der von mir eingereichten schriftlichen Arbeit mit dem Titel

.....

.....

um eine von mir erstmalig, selbstständig und ohne fremde Hilfe verfasste Arbeit handelt.

Ich erkläre ausdrücklich, dass ich *sämtliche* in der oben genannten Arbeit verwendeten fremden Quellen, auch aus dem Internet (einschließlich Tabellen, Grafiken u. Ä.) als solche kenntlich gemacht habe. Insbesondere bestätige ich, dass ich ausnahmslos sowohl bei wörtlich übernommenen Aussagen bzw. unverändert übernommenen Tabellen, Grafiken u. Ä. (Zitaten) als auch bei in eigenen Worten wiedergegebenen Aussagen bzw. von mir abgewandelten Tabellen, Grafiken u. Ä. anderer Autorinnen und Autoren (Paraphrasen) die Quelle angegeben habe.

Mir ist bewusst, dass Verstöße gegen die Grundsätze der Selbstständigkeit als Täuschung betrachtet und entsprechend der fachspezifischen Prüfungsordnung und/oder der Allgemeinen Satzung für Studien- und Prüfungsangelegenheiten der HU (ASSP) bzw. der Fächerübergreifenden Satzung zur Regelung von Zulassung, Studium und Prüfung der Humboldt-Universität (ZSP-HU) geahndet werden.

Datum …….. Unterschrift