Error Bars and Inference in Heterogeneous DFT Data

HU (Prof. Draxl), FHI (Prof. Scheffler)

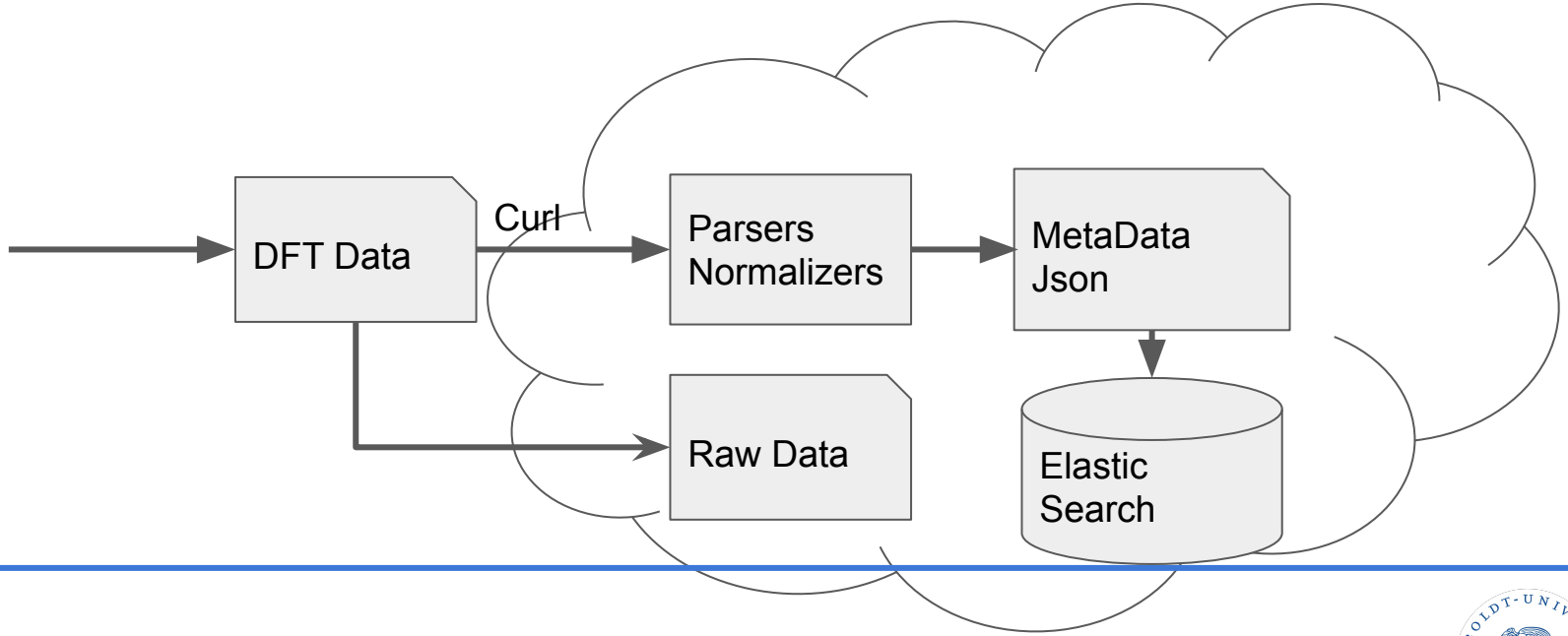
05/29/2019

Outline

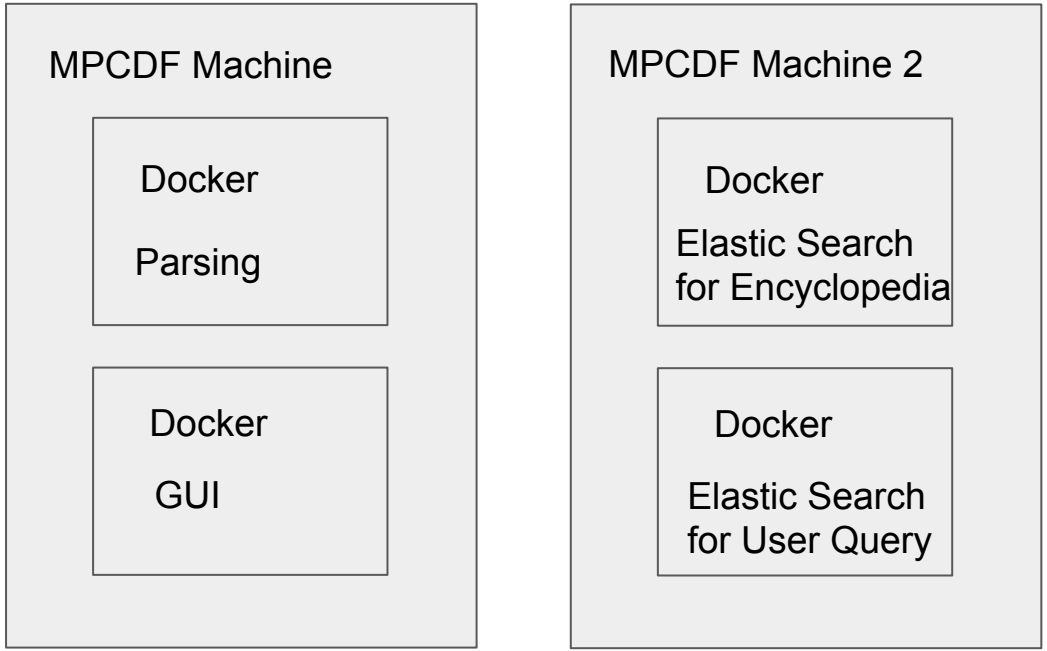
- Nomad Shout-out
- Overview of Projects, Aims
- Basis Set Size vs Energy Training Data Set
- Sources of Error in Training Data
- Model and parameter Selection Strategies
- Confidence Intervals

NOMAD Re-design

- https://labdev-nomad.esc.rzg.mpg.de/fairdi/nomad/coe/gui/



Nomad Backend Closer Look



Often we associate column based databases to data stores.

...

For DFT we often have matrices that are hard to columnize.

ES gives us querying speed we can't beat and is suited for JSON data.

Elastic Search Possibilities

```
query = {
    'query': {
        'match_all': {}
    },
    'aggs': {
        'by_city': {
            'terms': {
                'field': 'city'
            }
        },
        'by_price': {
            'histogram': {
                'field': 'price',
                'interval': 10,
            }
        }
    }
}

client.search(
    index='eventbrite',
    body=query,
)
```

```
{
  ...

  "aggregations": {
    "by_city": {
      "buckets": [
        {"key": "Nashville", "doc_count": 2311},
        {"key": "Dallas", "doc_count": 3743},
        {"key": "BFE", "doc_count": 7},
      ],
      "doc_count_error_upper_bound": 0,
      "sum_other_doc_count": 0
    },
    "by_price": {
      "buckets": [
        {"key": 10, "doc_count": 4263},
        {"key": 20, "doc_count": 1293},
        {"key": 30, "doc_count": 43},
```

We can nest these aggregations as well. This give us (joint) histograms. We can therefore return discrete probabilities and joint probabilities. This can be used in mutual information calculations.

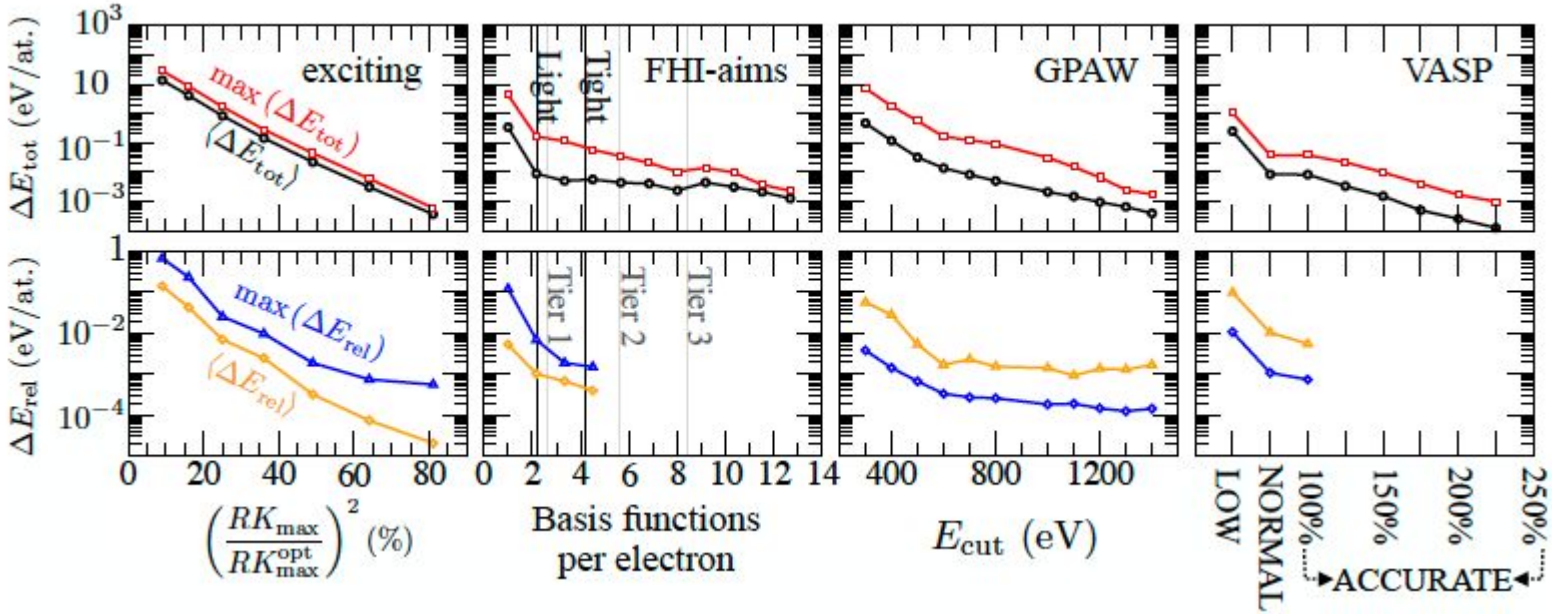
Error Bars in Nomad

- Often as a researchers or an engineer we want to quickly find a material property (stability of an oxide, hydride) or get a set of material candidates for a specific application (battery electrolyte).
- Formulating that query might look like:
  - filter=elements+HAS+EXACTLY+”Na”,”Cl”
- How do we know what results can be trusted?

Carbogno et Al.

- Lejaeghere, Kurt, et al. (2016) showed that modern DFT codes all converge upon the same ground state energies for elemental solids. This comes after much parameter tweaking.
- Carbogono sought to see how basis set size affects the ground state energy for elemental, binary and trinary solids.
- Two variables of interest the difference in ground state energy from converged value and the difference in ground state energy from expanded

Carbogno et Al.



More on Carbogno Data Set

- With DFT accuracy the first question that comes is:
  - Are you simulations converged wrt to k-points?
- Should you converge each material to k-points density and use that converged k-points density?
  - Is it ok to use different k-points density values when performing experiments on different materials?
  - Or should we use the largest required value for each material to avoid k-points density influence?
- Carbogno chooses to use different k-point densities 2, 4, 8 Angstroms corresponding to less than 5 milli eV/atom.
- Can we assume this small error behaves as a normally distributed function?

Material/Code Depdencies

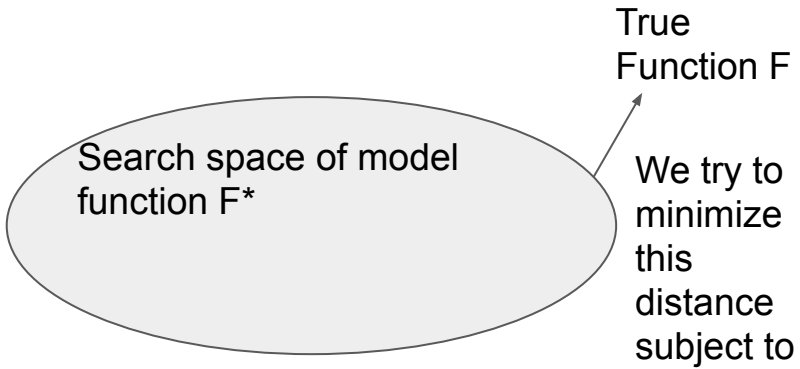
- FHI employ numeric atom-centered orbitals
- Exciting linearized augmented plane wave orbitals + local orbital (muffin tin)
- GPAW, VASP projector augmented wave (PAW)

- Each code use species defaults -> material independent basis set size model is complicated by this.

Goals

- A model for basis set size vs. ground state energy error from converged value.
- The model should then perform well on data not in the training set. It will spit out values of the errror. Al (s), FHI-Aims, Tight basis set size -> 100 meV error per atom.
- Since Nomad is run by scientists we should do better and provide 95% confidence intervals
    - i.e. -> 1 meV +- 230 meV is very different from: 1000 meV +- 230 meV
- With a model in which we have confidence we can do inference with NOMAD data, imagine small basis set size value is present, we can estimate what the converged value should be.

Modelling Strategy



$$min_{F \subseteq F'} \| y - F(x_i) \|$$

- Ideal fitting procedure should provide

1. Parameters (e.g. coefficients for linear model).
2. Error estimates of the parameters or a way to sample from their probability distributions.
3. A statistical measure of goodness of fit.

Modelling Strategy

- We're looking for a model that describes the data well (good fit), doesn't over-fit the data (we introduce regularization), and gives us an idea of the confidence we have in the parameters of the model.
- Example for linear model:

$$\hat{y}_i = ax_i + b$$

Quick Bayesian Aside

- The machinery we employ is bayes' theorem.

$$P(model \mid data) \text{ proportional to } P(data \mid model) * P(model)$$

- If each data point y_i has a measurement error that is indepdently random and distributed as a normal (Gaussian) distribution around true model y(x), and if the stdev (sigma) of these distributions are the same for all points then:

$$P(data \mid model) \text{ proportional to } \prod_{i=0}^{N-1} \{ exp(-\frac{y_i - y(x_i)}{2\sigma})^2 \} \Delta y$$

Maximum Likelihood Estimator

Frequentists (those that don't beleive in assigning priors) call the probability of the data given parameters as liklihood.

Therefore the most probable model is the one that maximizes:

$$P(data \mid model) \text{ proportional to } \prod_{i=0}^{N-1} \{ exp(- \frac{y_i - y(x_i)}{2\sigma})^2 \} \Delta y$$

Or minimizes the negative logarithm of above:

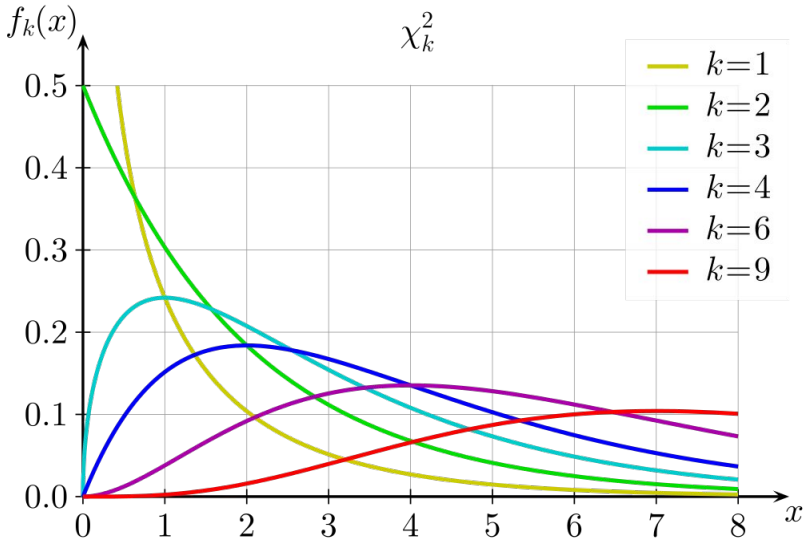
$$\sum_{i=0}^{N-1} [y - y(x_i)]^2/(2\sigma)^2 - N log \Delta y$$

Chi-Squared Fitting

If each data point (x_i, y_i) has its own known stdev $sigma_i$ the MLE becomes:

$$\chi^2 = \sum_{i=0}^{N-1} [y - y(x_i | a_0, \ldots a_{M-1})]^2 / \sigma_i^2$$

For models that are linear in the a's, the probability distribution of chi-squared is it the chi-square distribution for N-M degrees of freedom.



$$P(\chi^2) \propto (\chi^2)^{\frac{\nu-2}{2}} \exp(-\chi^2/2)$$

Chi-Squared Fitting

$$\overline{\chi^2} = \nu = N - p$$

Mean of chi-squared distribution (nu is degrees of freedom), N data points, p parameters.

$$\mathrm{Var}(\chi^2) = 2\,\nu$$

If model is correct, we expect each data point should lie about ~ 1 sigma from the model and contribute 1.0 to chi-squared. So we use this to reject poor models where probability is low that such a chi-squared value occur sfrom data

$$\chi^2 \sim \nu \pm \sqrt{2\nu}$$

Chi-Squared Fitting

Suppose a chi-squared hypothesis test yields p-value of =0.01.

This means : there is a 1% chance of obtaining a set of measurements at least this discrepant from the model, assuming the model is true

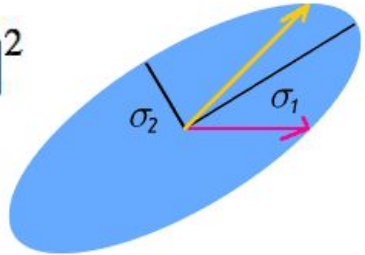
Often we don't know the individual stdev's. We assume all measurements have same constant value, then minimize chi-squared, and finally recomputing.

$$\sigma^2 = \sum_{i=0}^{N-1} [y_i - y(x_i)]^2/(N - M)$$

Chi-Squared Fitting - Standard Deviations in Params

For general linear model, we seek a that minimizes $\chi^2 = |A \cdot a - b|^2$

This type of problem that singular value decomposition routines solve.



vector

$$a = \sum_{i=0}^{M-1} (U_{(i)} \cdot b)/\sigma_i * V_i \pm \frac{V_{(0)}}{\sigma_0} \ldots \pm \frac{V_{(M-1)}}{\sigma_{M-1}}$$

$$M = U \cdot \Sigma \cdot V^*$$

$$\sigma^2(a_j) = \sum_{i=0}^{M-1} \left(\frac{V_{ij}}{\sigma_j}\right)^2 = C_{jj}$$

$\mathbf{V_{(i)}}$ are principal axes of the error ellipsoid of the fitted params. The standard deviations are all mutually orthogonal.

$$C = M^T M = V \Sigma^2 V^T$$

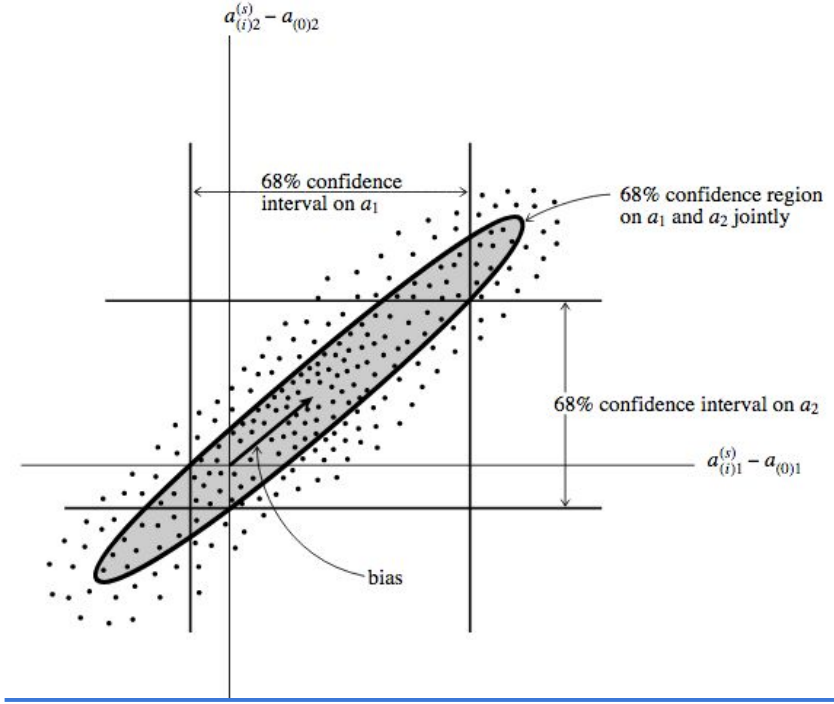
C is the familiar covaraince matrix

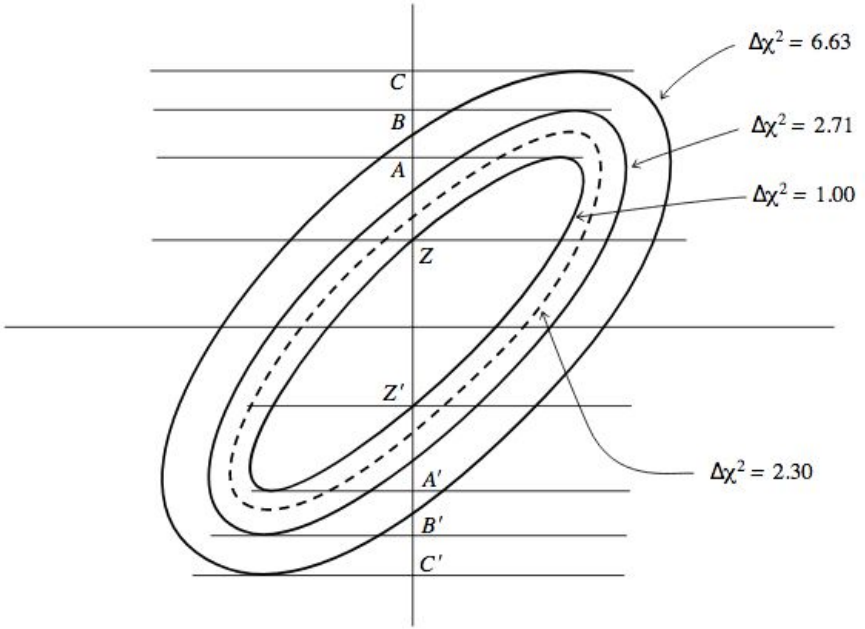
Confidence Limits on Estimated Model Params

- MLE gives us linear or non-linear coefficients but they are dependent on our dataset D_0.
- There are infinitely many realizations of the true parameters as hypothetical data sets each which could have been measured (D_1: $\mathbf{a}_{(1)}$, D_2: $\mathbf{a}_{(2)}$...)
- What is the probability distribution of $\mathbf{a}_{(i)}$ - $\mathbf{a}_{true}$?
- Assume $\mathbf{a}_{(i)}$ - $\mathbf{a}_0$ has the same probability distribution shape

> Bootstrapping, we can create synthetic datasets by sampling with replacement (Efron 1976), for each dataset we can compute confidence limits

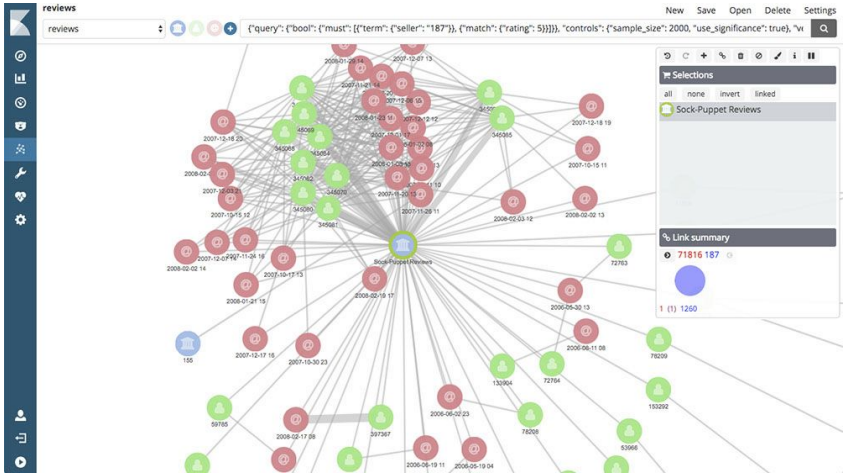
Confidence Limits on Estimated Model Params





Unfinished Business

- Analyze Noise (residual error)
  - Gaussian? Try EM, calculate KL div.
- New Nomad still in beta testing. Please use.
- No ML toolkit, I'm hoping to push parsing, bucketing, and modeling part to use NOMAD as much as possible (Kibana, Elastic Search)



Appendix Slides

$\Delta\chi^2$ as a Function of Confidence Level and Degrees of Freedom						
	ν					
p	1	2	3	4	5	6
68.3%	1.00	2.30	3.53	4.72	5.89	7.04
90%	2.71	4.61	6.25	7.78	9.24	10.6
95.4%	4.00	6.17	8.02	9.70	11.3	12.8
99%	6.63	9.21	11.3	13.3	15.1	16.8
99.73%	9.00	11.8	14.2	16.3	18.2	20.1
99.99%	15.1	18.4	21.1	23.5	25.7	27.8

Confidence Intervals Hastie/Friedman Recomended

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Mean zero random error term

$$\sigma^2 = \mathrm{Var}(\epsilon)$$

Often we don't know the true distribution of the error.

$$\mathrm{RSE} = \sqrt{\mathrm{RSS}/(n-2)}$$

Standard practice is to estimate as RSE

$$RSS = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

Confidence Intervals - Hastie/Friedman Recomended

$$\mathrm{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Where SE is the standard Error, 95% Confidence Interval

$$\hat{\beta}_1 \pm 2 \cdot \mathrm{SE}(\hat{\beta}_1).$$

$$\mathrm{SE}(\hat{\beta}_0)^2 = \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right]$$

$$\hat{\beta}_0 \pm 2 \cdot \mathrm{SE}(\hat{\beta}_0).$$

T-tests Common Use of Confidence Intervals

H_0 : There is no relationship between X and Y

H_a : There is some relationship between X and Y.

$H_0 : \beta_1 = 0$ vs $H_a : \beta_1 \neq 0,$

$$t = \frac{\hat{\beta}_1 - 0}{\mathrm{SE}(\hat{\beta}_1)}.$$

We compute a t-statistic.
Typically values less than 5% or 1% lead us to reject null hypothesis.

Metadata

dft code: VASP
dft code version: 4.6.35
basis set: plane waves
xc functional: GGA
system type: *unavailable*
crystal system: tetragonal
spacegroup: P4mm (99)

comment
no comment
references
no references
authors
Hofstadter, Leonard
datasets
no datasets

Ids / processing

PID
356
upload id
Y2tn0oWrS0iwgA97b13ZgA
upload time
4/11/2019, 2:45:57 PM
calculation id
BaGzTooNlQf0gFMUETxttZTwAmX3
mainfile
RopD3Mo8oMV_-E5bh8uW5PiiCRkH1...
calculation hash
TBR4vcS2Rr-vFhb2TbzOOGoKkuT6
last processing
4/11/2019, 2:47:09 PM
processing version
0.4.4/c0248a0a30df76d53aa9f2ea0b...

Raw files

select all
2/2 files selected
vasprun.xml.relax2
vasprun.xml.relax1

Repository JSON


```
    0 : string "Br"
    1 : string "K"
    2 : string "Sc"
  ]
  "n_atoms" : int 0
  "basis_set" : string "plane waves"
  "xc_functional" : string "GGA"
  "system" : string "bulk"
  "crystal_system" : string "tetragonal"
  "spacegroup" : int 99
  "spacegroup_symbol" : string "P4mm"
  "code_name" : string "VASP"
  "code_version" : string "4.6.35"
  "n_total_energies" : int 0
  "n_geometries" : int 0
  "quantities" :
    [ 0 - 100 ]
    [ 100 - 152 ]
  "geometries" : [
    0 :
    string "tk0f6W4n883tG6oDnvqJHhYh8gbs"
  ]
  "group_hash" :
  string "7XaJR-fozFF2pfrKcbolF6YqJCSr"
  "pid" : string "356"
  "with_embargo" : bool false
}
```


SHOW RAW JSON CLOSE