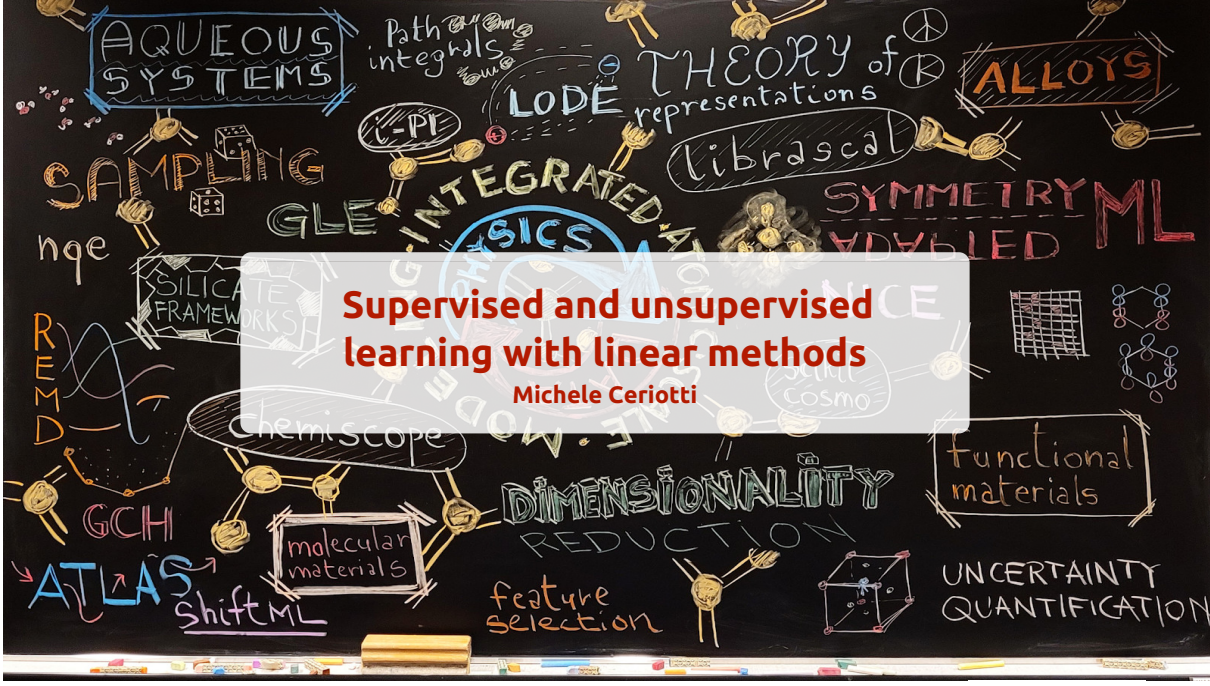
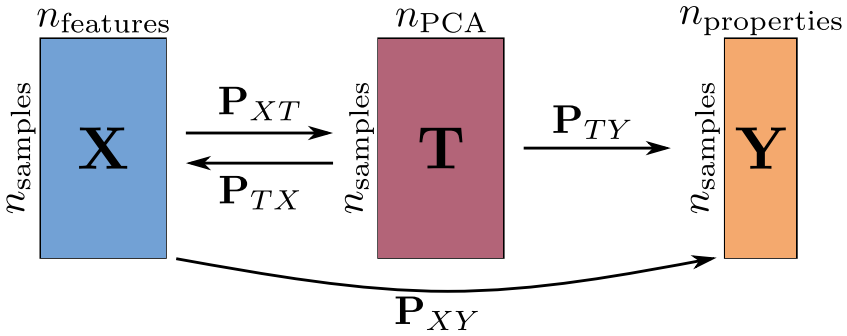


Supervised and unsupervised learning with linear methods

Michele Ceriotti

Outline

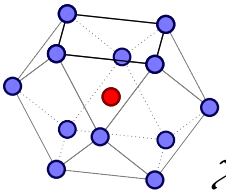
- Unsupervised and supervised learning for materials and molecules
- Linear methods: PCA, ridge regression, PCovR
- The kernel trick



Different flavours of data-driven modeling

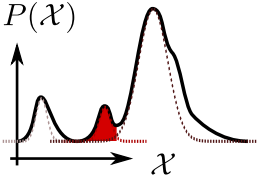
Recognizing molecular patterns

- "Chemical intuition" builds on recognizing recurring patterns in atomic configurations
- Atomistic models provide large amount of data for statistical analysis

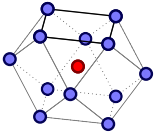


$\mathcal{X}$

Simulation data



$\Theta(\mathcal{X}) = 1$



$\Theta(\mathcal{X}) = 0$

Abstract description of chemical environments

Probabilistic analysis of structural correlations

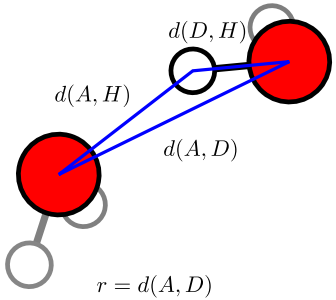
Posterior probability as a structural fingerprint

$$\Theta(\mathcal{X}) = \frac{p_0 g_0(\mathcal{X})}{\sum_k p_k g_k(\mathcal{X})}$$

C.M. Bishop, "Pattern Recognition and Machine Learning"
Gasparotto & **MC**, JCP 174110, 141 (2014); Gasparotto, Meißner, **MC** JCTC (2018)

Recognizing molecular patterns

- "Chemical intuition" builds on recognizing recurring patterns in atomic configurations
- Atomistic models provide large amount of data for statistical analysis

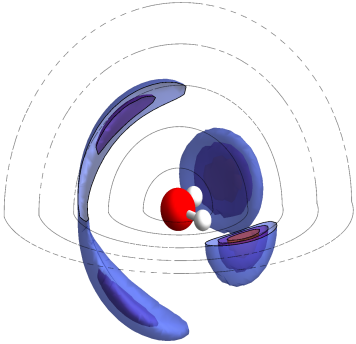


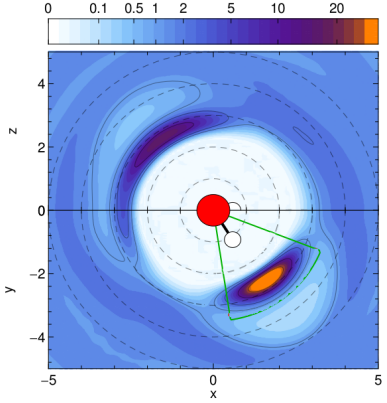
$$r = d(A,D)$$
$$\mu = d(A,H) + d(D,H)$$
$$\nu = d(D,H) - d(A,H)$$

C.M. Bishop, "Pattern Recognition and Machine Learning"
Gasparotto & **MC**, JCP 174110, 141 (2014); Gasparotto, Meißner, **MC** JCTC (2018)

Recognizing molecular patterns

- ‘‘Chemical intuition’’ builds on recognizing recurring patterns in atomic configurations
- Atomistic models provide large amount of data for statistical analysis

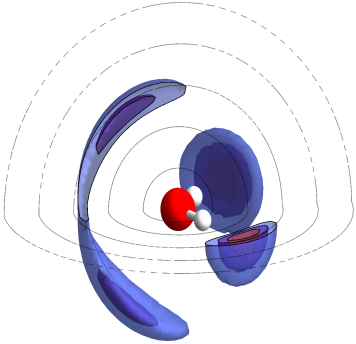


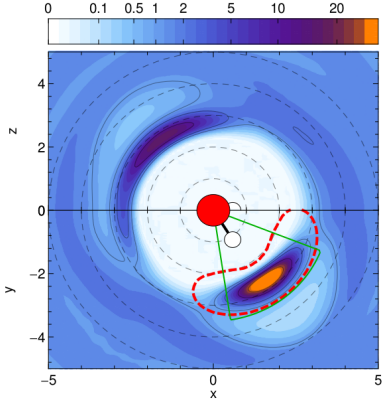


C.M. Bishop, "Pattern Recognition and Machine Learning"
Gasparotto & **MC**, JCP 174110, 141 (2014); Gasparotto, Meißner, **MC** JCTC (2018)

Recognizing molecular patterns

- ‘‘Chemical intuition’’ builds on recognizing recurring patterns in atomic configurations
- Atomistic models provide large amount of data for statistical analysis

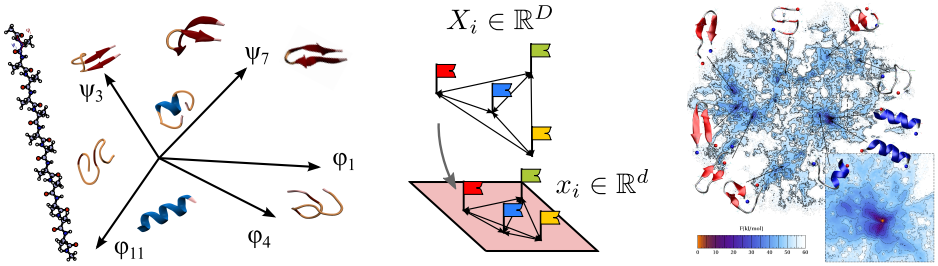




C.M. Bishop, "Pattern Recognition and Machine Learning"
Gasparotto & **MC**, JCP 174110, 141 (2014); Gasparotto, Meißner, **MC** JCTC (2018)

Dimensionality reduction

- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



High-dimensional descriptors of a complex system

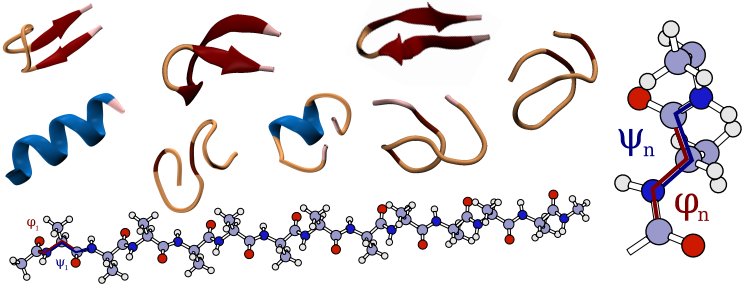
Non-linear dimensionality reduction (**sketch-map**)

Intuitive maps of structural stability

Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Dimensionality reduction

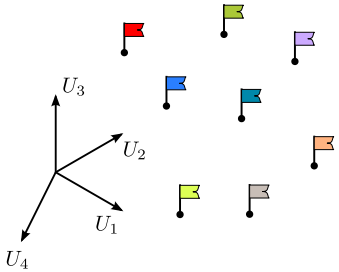
- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Dimensionality reduction

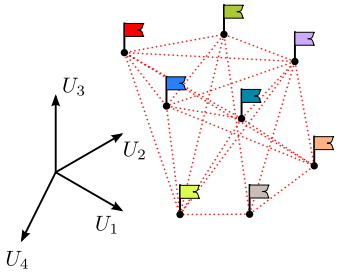
- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Dimensionality reduction

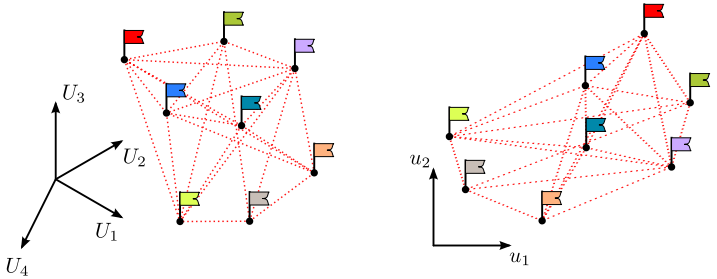
- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Dimensionality reduction

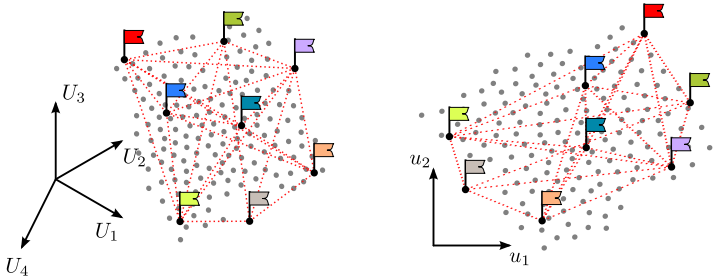
- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Dimensionality reduction

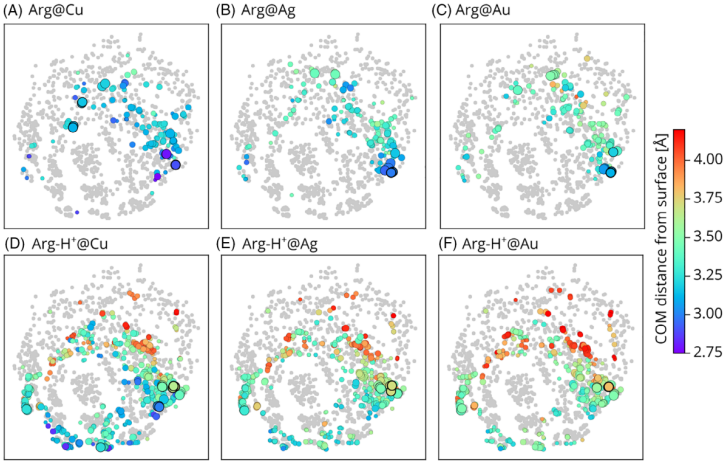
- Distances describe the relations between different molecules or structures. However, to visualize such relations we need to generate a low-dimensional **map** that is (approximately) consistent with the metric.
  - Take a set of configurations ⇒ high-dim. **landmark points**
  - Define a measure of dissimilarity between the points
  - Arrange low-dim. points so that the dissimilarities are preserved
  - Locate other configurations with an **out-of-sample embedding**



Ceriotti, Tribello, Parrinello, PNAS (2011); JCTC (2013)

Structure-property maps

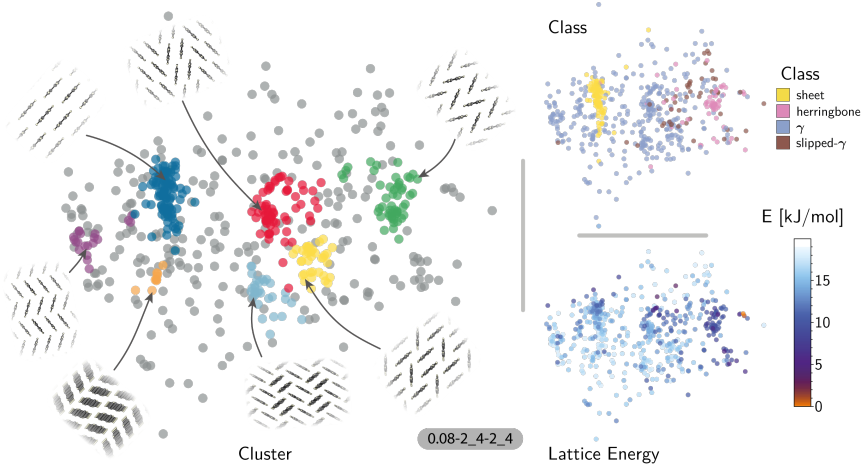
- Representing databases of conformers, and the effect of surfaces on the adsorbed molecules
- Rationalizing stacking patterns and stability of molecular materials



Maksimov, Baldauf, Rossi, IJQC (2020)

Structure-property maps

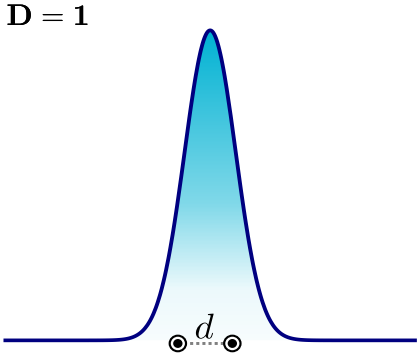
- Representing databases of conformers, and the effect of surfaces on the adsorbed molecules
- Rationalizing stacking patterns and stability of molecular materials



Musil, De, Yang, Campbell, Day, **MC**, Chemical Science (2018);`http://interactive.sketchmap.org`

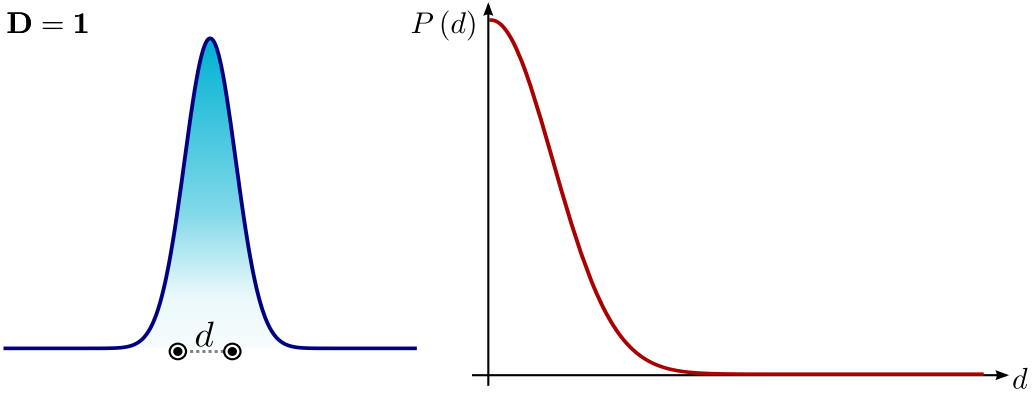
Don’t over-do it ...

- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none



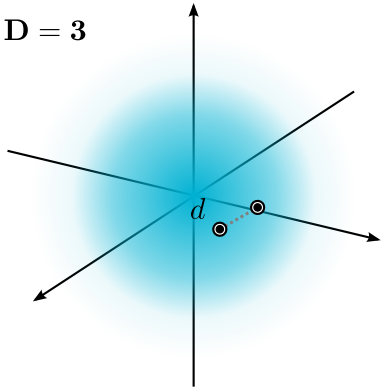
Don’t over-do it ...

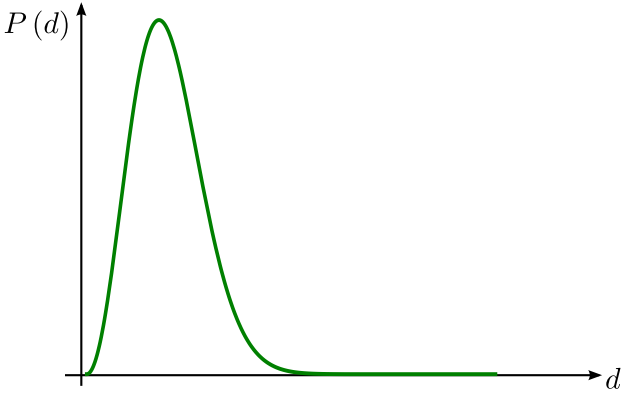
- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none



Don’t over-do it ...

- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none

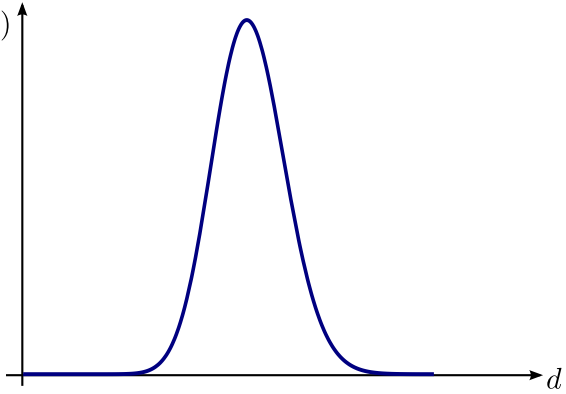




Don’t over-do it ...

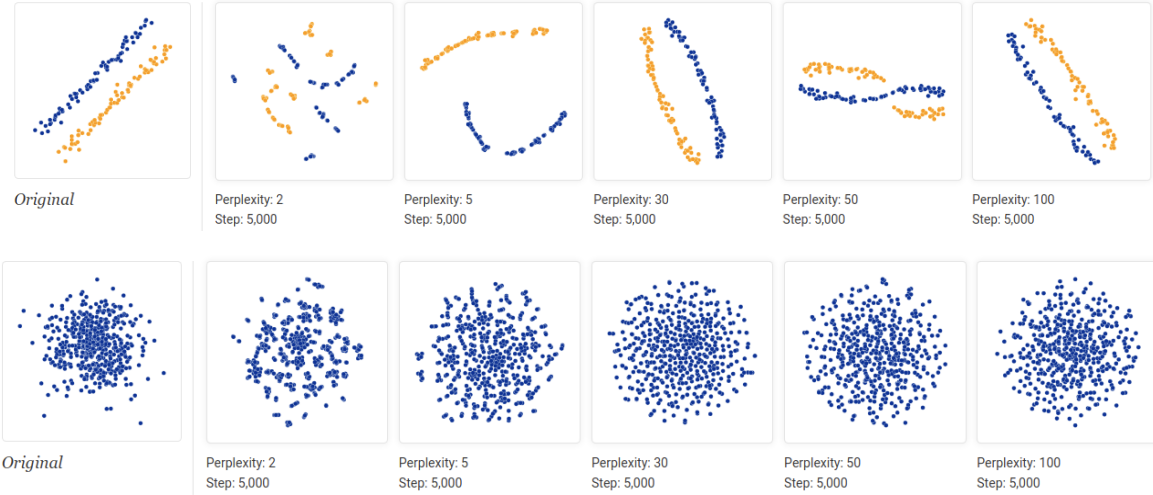
- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none

$\mathbf{D = 24}$



Don’t over-do it ...

- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none



Wattenberg et al., `https://distill.pub/2016/misread-tsne`

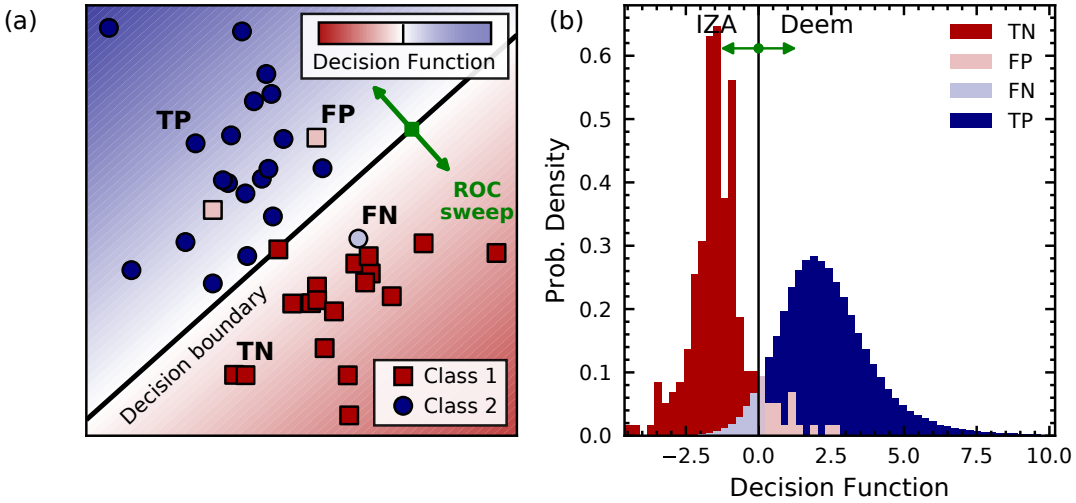
Don’t over-do it ...

- High-dimensional distributions can be highly counterintuitive
- Non-linear dimensionality algorithms can show structure where there should be none

Wattenberg et al., `https://distill.pub/2016/misread-tsne`

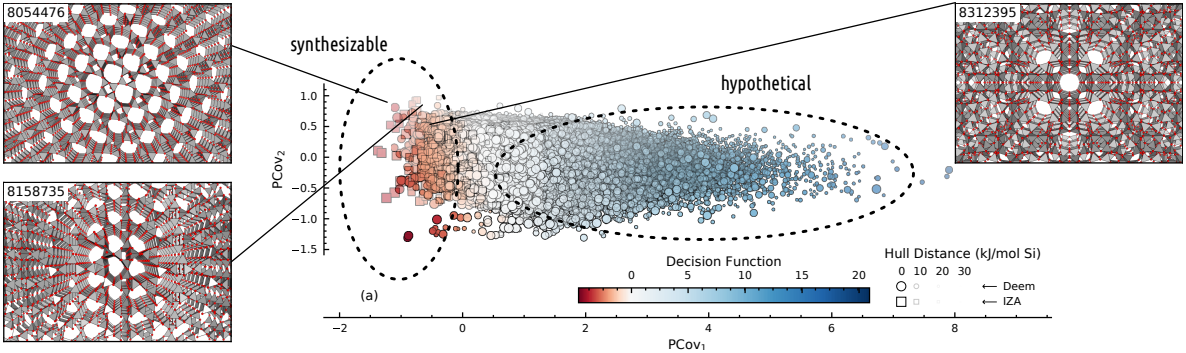
Classification

- Separate structure into classes based on known labels (active/inactive, stable/unstable, ...)
- Finding synthesizable zeolites among millions of hypothetical frameworks



Classification

- Separate structure into classes based on known labels (active/inactive, stable/unstable, ...)
- Finding synthesizable zeolites among millions of hypothetical frameworks



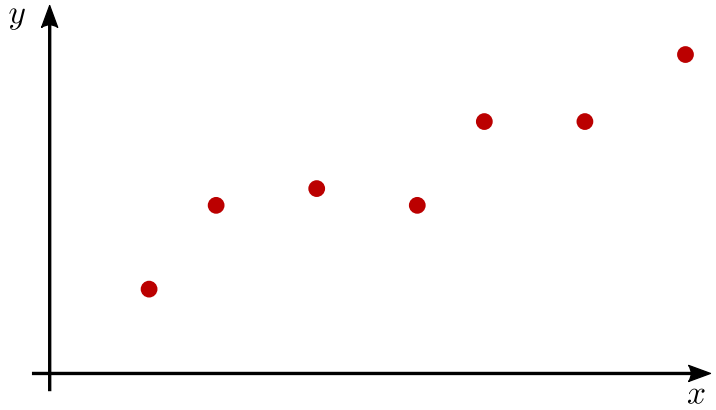
Helfrecht, Pireddu, Semino, Auerbach, MC, Digital Discovery (2022)

Regression, loss & C.

- Reproduce target properties y as a function $\tilde{y}$ of input features ξ, optimizing parameters to minimize a *loss*

$$\ell = \sum_{A \in \text{train set}} |y_A - \tilde{y}(\xi_A)|^2$$

- Objective is not interpolation: verify accuracy on a separate *test set* to identify problematic *overfitting* (train error $\ll$ test error)

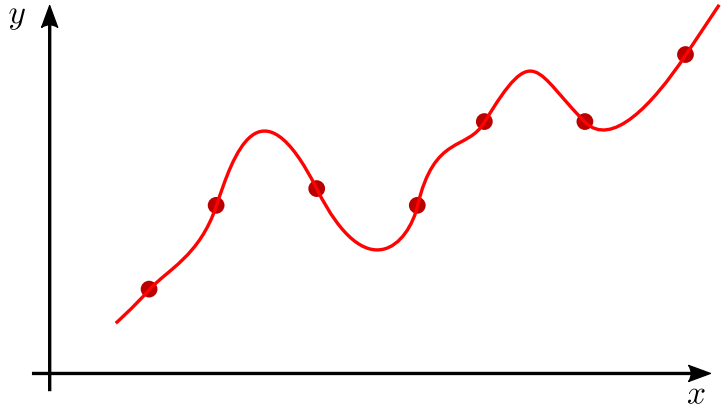


Regression, loss & C.

- Reproduce target properties y as a function $\tilde{y}$ of input features ξ, optimizing parameters to minimize a *loss*

$$\ell = \sum_{A \in \text{train set}} |y_A - \tilde{y}(\xi_A)|^2$$

- Objective is not interpolation: verify accuracy on a separate *test set* to identify problematic *overfitting* (train error $\ll$ test error)

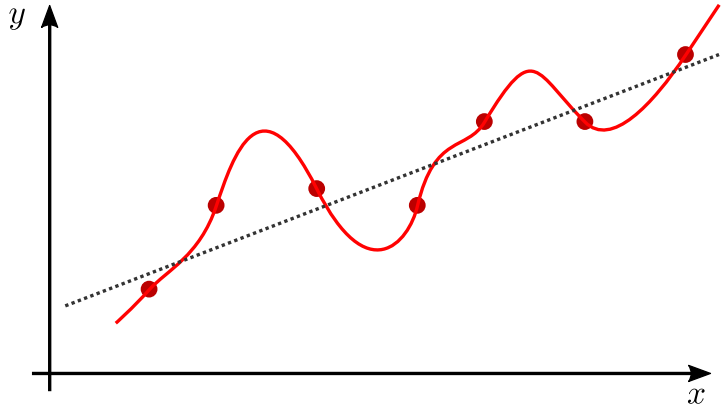


Regression, loss & C.

- Reproduce target properties y as a function $\tilde{y}$ of input features ξ, optimizing parameters to minimize a *loss*

$$\ell = \sum_{A \in \text{train set}} |y_A - \tilde{y}(\xi_A)|^2$$

- Objective is not interpolation: verify accuracy on a separate *test set* to identify problematic *overfitting* (train error $\ll$ test error)

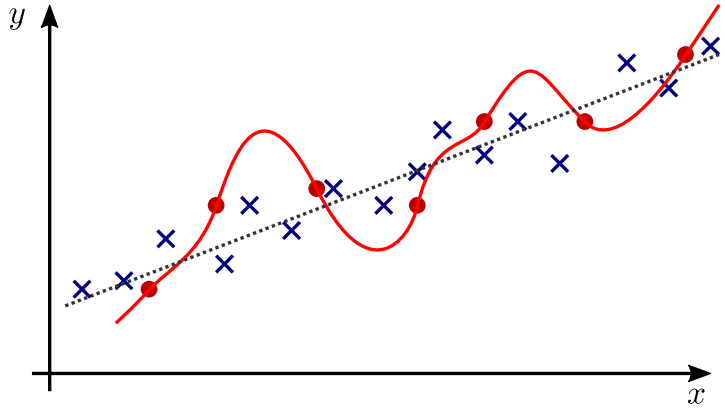


Regression, loss & C.

- Reproduce target properties y as a function $\tilde{y}$ of input features ξ, optimizing parameters to minimize a *loss*

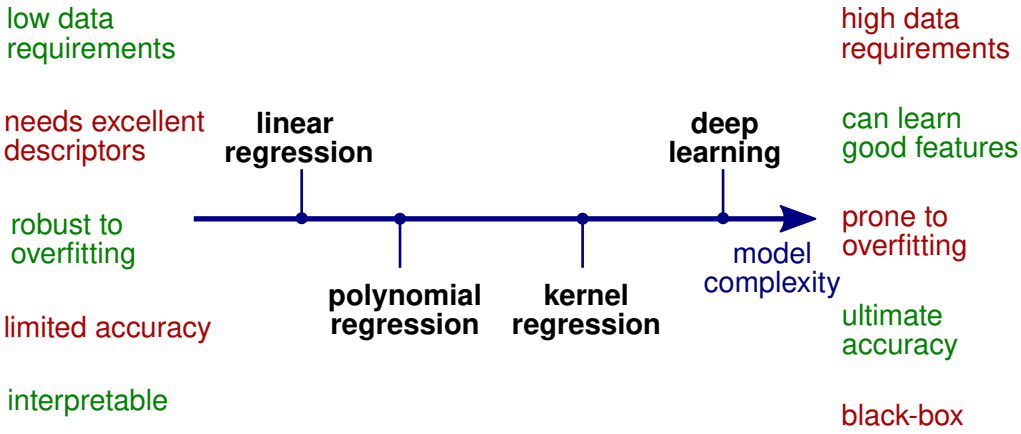
$$\ell = \sum_{A \in \text{train set}} |y_A - \tilde{y}(\xi_A)|^2$$

- Objective is not interpolation: verify accuracy on a separate *test set* to identify problematic *overfitting* (train error $\ll$ test error)



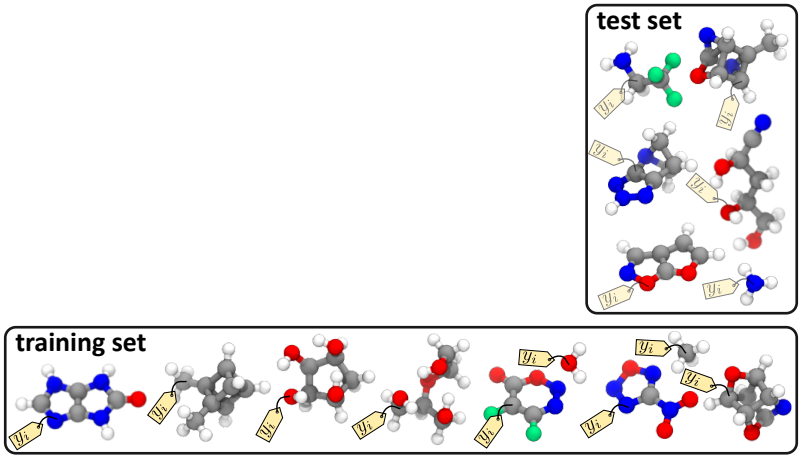
What kind of regression?

- Many different models for $\tilde{y}$. Flexibility comes at a cost



Learning curves

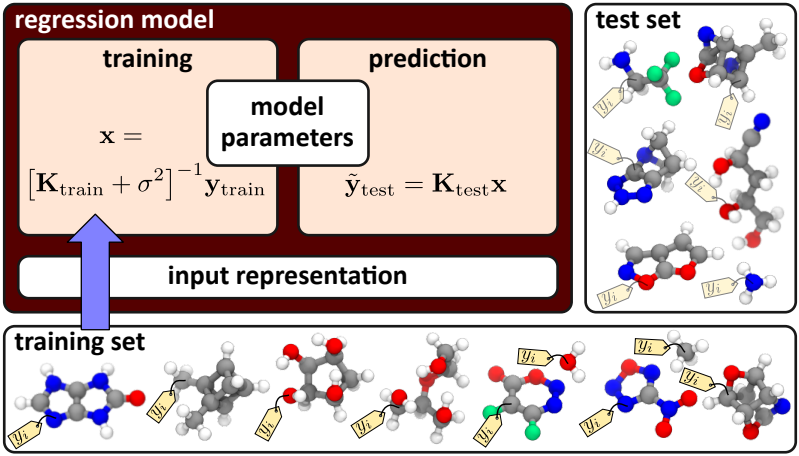
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



MC, Willatt, Csányi, Handbook of Materials Modeling, Springer (2018)

Learning curves

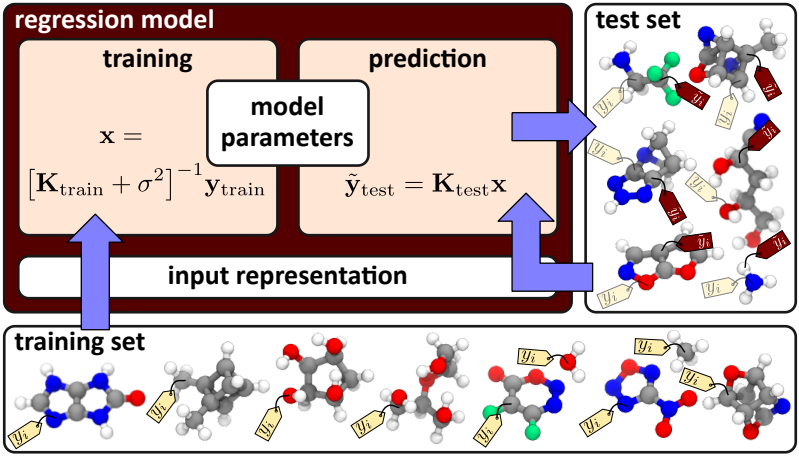
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



MC, Willatt, Csányi, Handbook of Materials Modeling, Springer (2018)

Learning curves

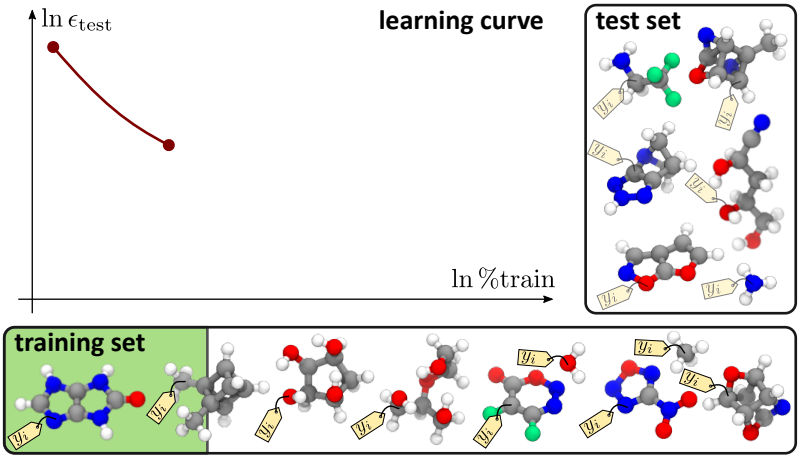
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



MC, Willatt, Csányi, Handbook of Materials Modeling, Springer (2018)

Learning curves

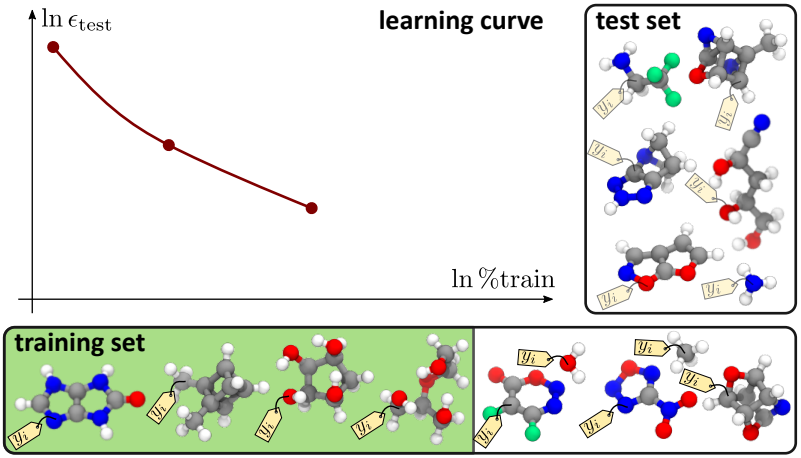
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



Huang, von Lilienfeld, JCP (2016); Loureiro et al., J. Stat. Mech. (2022)

Learning curves

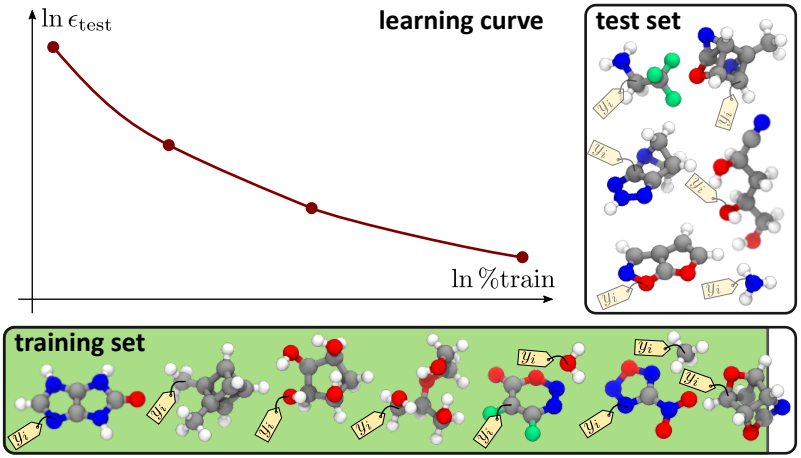
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



Huang, von Lilienfeld, JCP (2016); Loureiro et al., J. Stat. Mech. (2022)

Learning curves

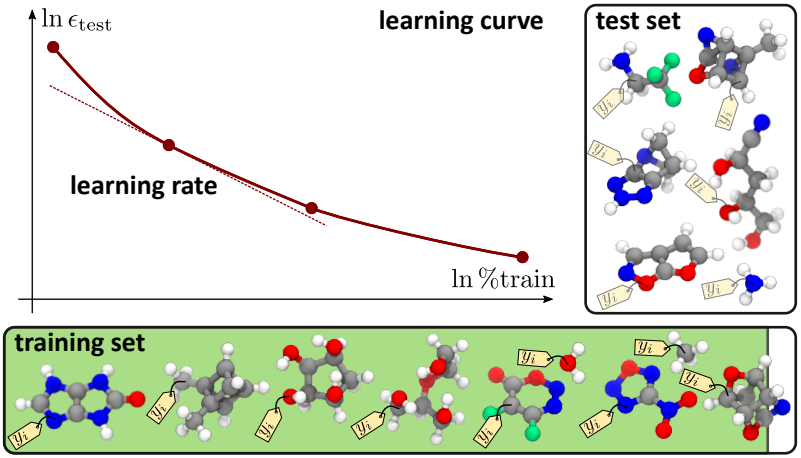
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model



Huang, von Lilienfeld, JCP (2016); Loureiro et al., J. Stat. Mech. (2022)

Learning curves

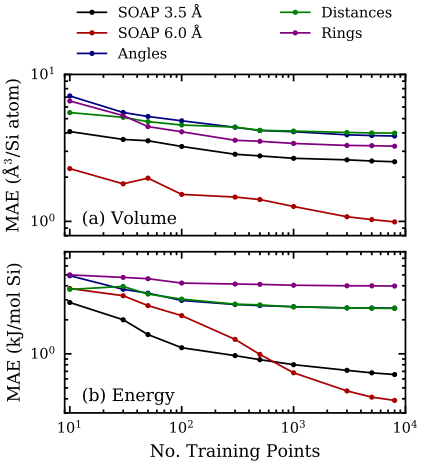
- Data is split between a training set - used to determine the parameters of the model - and a test/validation set used to verify accuracy of predictions
- Learning curves provide diagnostics to understand data and model

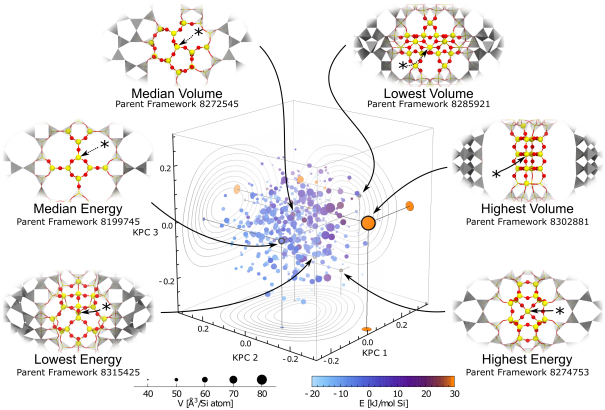


Huang, von Lilienfeld, JCP (2016); Loureiro et al., J. Stat. Mech. (2022)

Physical insights from knock-out ML models

- Analysis of learning curves in the large-data limit combined with interpretable models help determine structure-property relations



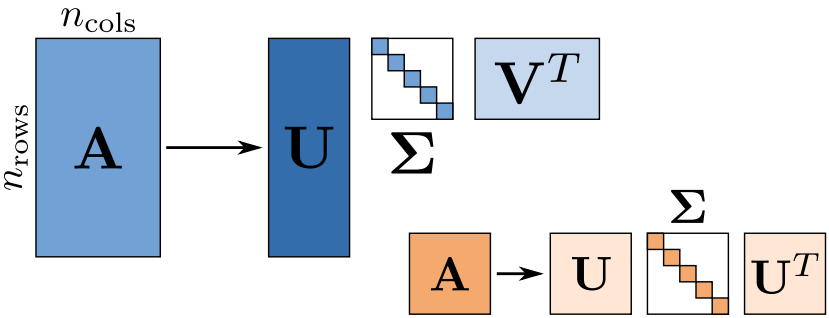


Helfrecht et al., JCP (2019)

Back to the basics: Linear methods

Most of the linear algebra you will need

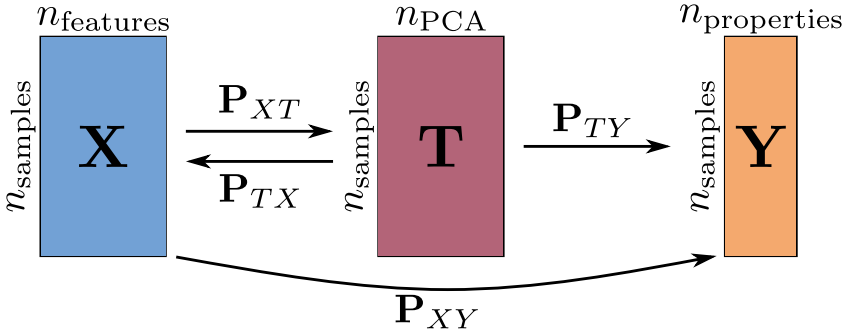
- Core tool for derivations/understanding: matrix decompositions.
- Singular value decomposition of a matrix $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$.
  - *Singular values* Σ indicate the magnitude of different components of the matrix.
  - Left and right *singular vectors* indicate the directions along which those values apply.
- Special case: eigenvalue decomposition, for square symmetric $\mathbf{A} = \mathbf{U}\Sigma\mathbf{U}^T$



Golub, Van Loan, *Matrix computations*

General setting of the problem

- Assume that each structure (or environment) has been mapped to a vector of *features* **x**. These features are collected in a $n_{\text{samples}} \times n_{\text{features}}$ feature matrix **X**. The properties associated with each sample are stored in $n_{\text{samples}} \times n_{\text{properties}}$ matrix **Y**
- We want to find linear projections to
  - Reduce the dimensionality of the feature space (*latent space* **T**)
  - Predict properties based on full (or reduced) features
- Assume **X** rows and **Y** entries are centered and scaled to have zero mean and unit variance



Principal component analysis

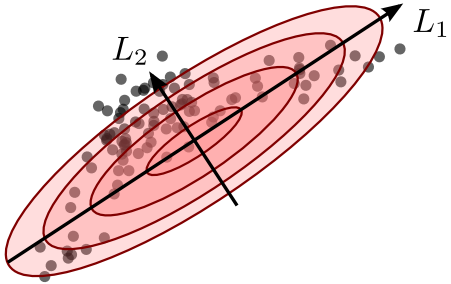
- What is the latent-space projection that minimizes the error in reconstructing $\mathbf{X}$ starting from $\mathbf{T}$? Optimize the loss

$$\ell = \|\mathbf{X} - \mathbf{X}\mathbf{P}_{XT}\mathbf{P}_{TX}\|^2$$

- If you take $\mathbf{P}_{XT}$ to be an orthogonal matrix, then ℓ is maximised by diagonalizing the covariance $\mathbf{C} = \frac{1}{n_{\text{samples}}}\mathbf{X}^T\mathbf{X} = \mathbf{U}_\mathbf{C}\Lambda_\mathbf{C}{\mathbf{U}_\mathbf{C}}^T$. Then $\mathbf{P}_{XT}$ is given by the first n_{PCA} columns of $\mathbf{U}_\mathbf{C}$
- The PCA latent space maximises the fraction of the variance that is described by $\mathbf{T}$

Principal component analysis

- What is the latent-space projection that minimizes the error in reconstructing $\mathbf{X}$ starting from $\mathbf{T}$? Optimize the loss
$$\ell = \|\mathbf{X} - \mathbf{X}\mathbf{P}_{XT}\mathbf{P}_{TX}\|^2$$
- If you take $\mathbf{P}_{XT}$ to be an orthogonal matrix, then ℓ is maximised by diagonalizing the covariance $\mathbf{C} = \frac{1}{n_{\text{samples}}}\mathbf{X}^T\mathbf{X} = \mathbf{U}_\mathbf{C}\Lambda_\mathbf{C}{\mathbf{U}_\mathbf{C}}^T$. Then $\mathbf{P}_{XT}$ is given by the first n_{PCA} columns of $\mathbf{U}_\mathbf{C}$
- The PCA latent space maximises the fraction of the variance that is described by $\mathbf{T}$

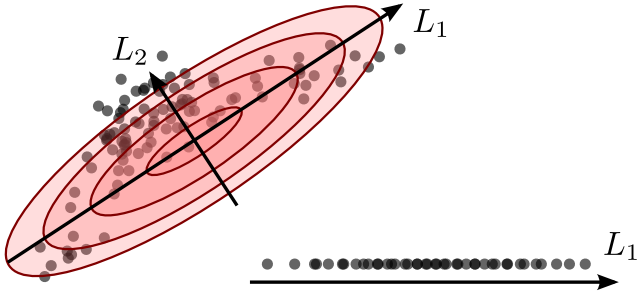


Principal component analysis

- What is the latent-space projection that minimizes the error in reconstructing **X** starting from **T**? Optimize the loss

$$\ell = \|\mathbf{X} - \mathbf{X}\mathbf{P}_{XT}\mathbf{P}_{TX}\|^2$$

- If you take $\mathbf{P}_{XT}$ to be an orthogonal matrix, then ℓ is maximised by diagonalizing the covariance $\mathbf{C} = \frac{1}{n_{\text{samples}}}\mathbf{X}^T\mathbf{X} = \mathbf{U}_\mathbf{C}\Lambda_\mathbf{C}\mathbf{U}_\mathbf{C}{}^T$. Then $\mathbf{P}_{XT}$ is given by the first n_{PCA} columns of $\mathbf{U}_\mathbf{C}$
- The PCA latent space maximises the fraction of the variance that is described by **T**



Classical Multidimensional Scaling

- A literal implementation of the general idea of dimensionality reduction

$$\chi^2 = \sum_{ij} \left(\left| \mathbf{x}_i - \mathbf{x}_j \right| - \left| \mathbf{t}_i - \mathbf{t}_j \right| \right)^2$$

- *Classical* MDS turns this iterative optimization in an eigenvalue problem by optimizing the alternative loss

$$\ell = \left\| \mathbf{X}\mathbf{X}^T - \mathbf{T}\mathbf{T}^T \right\|^2$$

- Defining the Gram matrix $\mathbf{K} = \mathbf{X}\mathbf{X}^T = \mathbf{U}_\mathbf{K} \Lambda_\mathbf{K} {\mathbf{U}_\mathbf{K}}^T$ we can minimize ℓ by setting $\mathbf{T} = \mathbf{U}_\mathbf{K} {\Lambda_\mathbf{K}}^{1/2}$, truncated to the top n_{PCA} principal components. Equivalent to PCA, as $\mathbf{X}\mathbf{U}_\mathbf{C} = \mathbf{U}_\mathbf{K} \Lambda_\mathbf{K}^{1/2}$

Linear regression

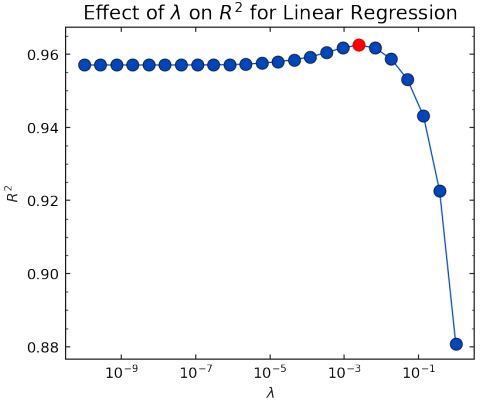
- Simplest form of supervised learning: linear map between **X** and **Y** (often stabilized with a ridge regularization)

$$\ell = \|\mathbf{Y} - \mathbf{X}\mathbf{P}_{XY}\|^2 + \lambda \|\mathbf{P}_{XY}\|^2$$

- Solved by

$$\mathbf{P}_{XY} = \left(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{Y}$$

- Interesting exercise: restricting the regression on the PCA latent space (PCR).



Principal Covariates Regression

- Very simple idea to combine PCA and latent-space LR to find a dimensionality reduction that preserves variance and predicts well

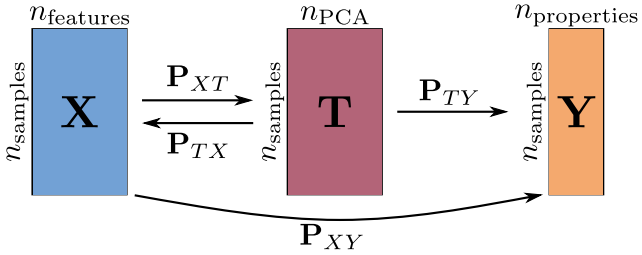
$$\ell = \alpha \|\mathbf{X} - \mathbf{X}\mathbf{P}_{XT}\mathbf{P}_{TX}\|^2 + (1-\alpha)\,\|\mathbf{Y} - \mathbf{X}\mathbf{P}_{XT}\mathbf{P}_{TY}\|^2$$

- Solution can be found working in sample space (looking for the eigenvectors of a modified Gram matrix)

$$\tilde{\mathbf{K}} = \alpha\mathbf{X}\mathbf{X}^T + (1-\alpha)\,\mathbf{X}\mathbf{P}_{XY}\mathbf{P}_{XY}^T\mathbf{X}^T$$

- ... or in feature space by diagonalizing a modified covariance

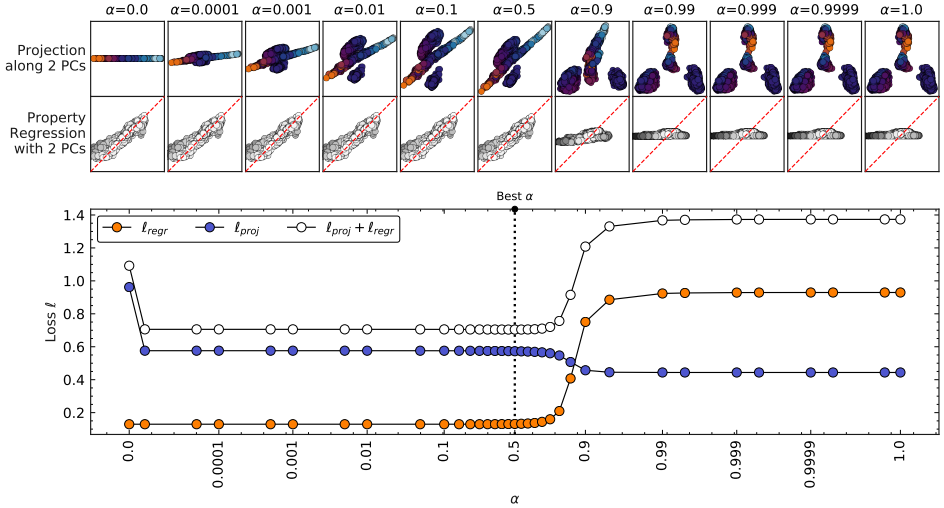
$$\tilde{\mathbf{C}} = \alpha\mathbf{X}^T\mathbf{X} + (1-\alpha)\,(\mathbf{X}^T\mathbf{X})^{-1/2}\,\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}\,(\mathbf{X}^T\mathbf{X})^{-1/2}$$



S. de Jong and HAL Kiers, Scandinavian Symposium on Chemometrics (1992)

A smart interpolation between LR and PCA

- PCovR finds a balance between representing the variance and predicting the targets
- "Best" α can be defined as the minimum of $\ell_{\text{PCA}} + \ell_{\text{LR}}$



The kernel trick

Reproducing Kernel Hilbert Space

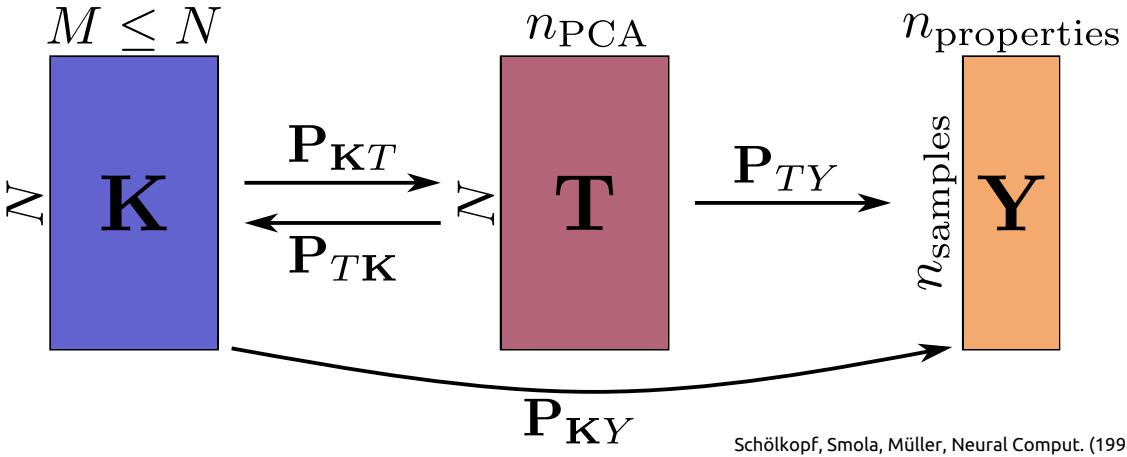
- **Kernel methods introduce non-linearity but operate using a linear algebra framework**
- Central ingredient a *positive-definite* function - a kernel $k(\mathbf{x}, \mathbf{x}')$, based on which we build a kernel matrix $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$
  - Typical kernels: linear kernel $\mathbf{x} \cdot \mathbf{x}'$, squared exponential $\exp - |\mathbf{x} - \mathbf{x}'|^2$
- Mercer theorem: there is an injective map $\mathbf{x} \rightarrow \phi(\mathbf{x})$ such that the scalar product $\phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$ yields the kernel $k(\mathbf{x}, \mathbf{x}')$. ϕ defines the **reproducing kernel Hilbert space (RKHS)** associated with the kernel
- The theorem is constructive: given a dataset, it allows building an explicit RKHS feature vector. If you diagonalize $\mathbf{K} = \mathbf{U_K} \Lambda_\mathbf{K} \mathbf{U_K}^T$ and set $\Phi = \mathbf{U_K} \Lambda_\mathbf{K}^{1/2}$, then $\Phi \Phi^T = \mathbf{K}$
- All linear methods have a kernelized counterpart that can be formally derived using Φ as features

M. Cuturi, http://arxiv.org/abs/0911.5367 (2009)

Kernel PCA, Kernel Ridge Regression

- Kernel PCA projection is just a truncation of the RKHS to the first n_{latent} components, $\mathbf{T} = \mathbf{U}_{\mathbf{K}} \Lambda_{\mathbf{K}}^{1/2}$
- Kernel ridge regression is an example of how to apply kernel methods without explicitly computing the RKHS

$$\mathbf{P}_{\Phi Y} = \left(\Phi^T \Phi + \lambda \mathbf{I} \right)^{-1} \Phi^T \mathbf{Y} = \Phi^T \left(\mathbf{K} + \lambda \mathbf{I} \right)^{-1} \mathbf{Y} \quad \rightarrow \quad \hat{\mathbf{Y}} = \Phi \mathbf{P}_{\Phi Y} = \mathbf{K} \left(\mathbf{K} + \lambda \mathbf{I} \right)^{-1} \mathbf{Y}$$



Schölkopf, Smola, Müller, Neural Comput. (1998)

Sparse kernel methods

- Kernel methods have a reputation of poor scaling with number of training samples: fake news!
- Very simple to formulate a sparse kernel problem:
  - Define a number of active samples M and build the RKHS associated with the active set. The feature matrix for the train set is $n_{\text{samples}} \times n_{\text{active}}$

$$\Phi_{NM} = \mathbf{K}_{NM} \mathbf{U}_{\mathbf{K}_{NM}} \Lambda_{\mathbf{K}_{NM}}^{-1/2}$$

  - Just construct linear methods in the active RKHS. Can avoid explicit construction of Φ_{NM} if desired
- Sparse KPCA: PCA in RKHS. Build $\mathbf{C} = \Phi_{NM}^T \Phi_{NM}$, and then

$$\mathbf{T} = \mathbf{K}_{NM} \mathbf{U}_{\mathbf{K}_{NM}} \Lambda_{\mathbf{K}_{NM}}^{-1/2} \mathbf{U}_{\mathbf{C}}$$

- Sparse KRR: LR in RKHS, or KRR based on Nyström approximation of $\mathbf{K}$

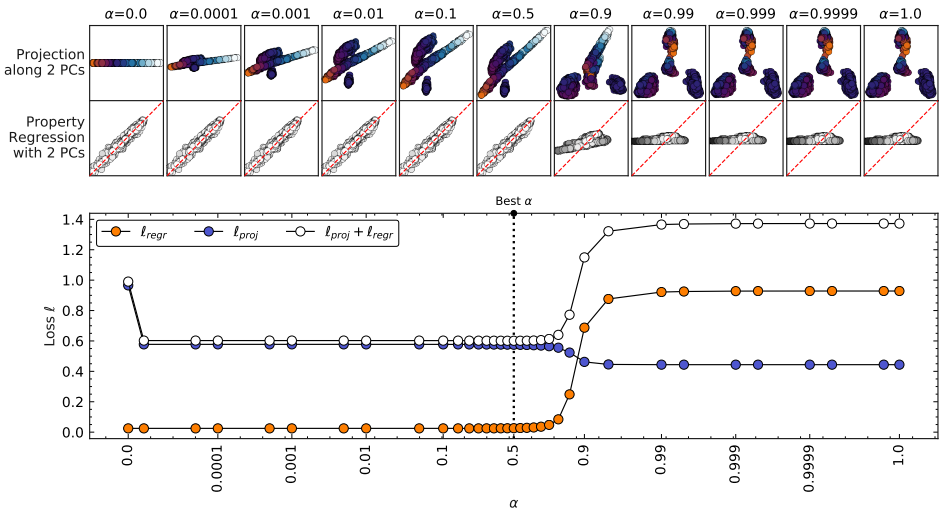
$$\hat{\mathbf{Y}} = \mathbf{K}_{NM} \left(\mathbf{K}_{NM}^T \mathbf{K}_{NM} + \lambda \mathbf{K}_{MM} \right)^{-1} \mathbf{K}_{NM}^T \mathbf{Y}$$

Tipping, NIPS (2001)

Kernel PCovR

- Kernel versions of PCovR can be obtained relatively easily by working with a modified kernel $\tilde{\mathbf{K}} = \alpha\mathbf{K} + (1-\alpha)\,\hat{\mathbf{Y}}\hat{\mathbf{Y}}^T$, diagonalizing it and then finding the latent-space projector

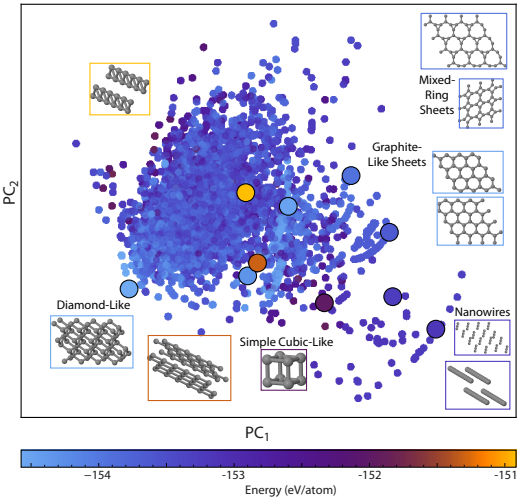
$$\mathbf{P}_{\mathbf{K}T} = \left(\alpha\mathbf{I} + (1-\alpha)\left(\mathbf{K} + \lambda\mathbf{I}\right)^{-1}\mathbf{Y}\hat{\mathbf{Y}}\right)\mathbf{U}_{\tilde{\mathbf{K}}}\Lambda_{\tilde{\mathbf{K}}}^{1/2}$$



Helfrecht, Cersonsky, Fraux, **MC**, MLST (2020)

Beyond unsupervised maps

- Kernel PCA map of a dataset of carbon structures
- KPCovR reveals more clearly structure/stability relations

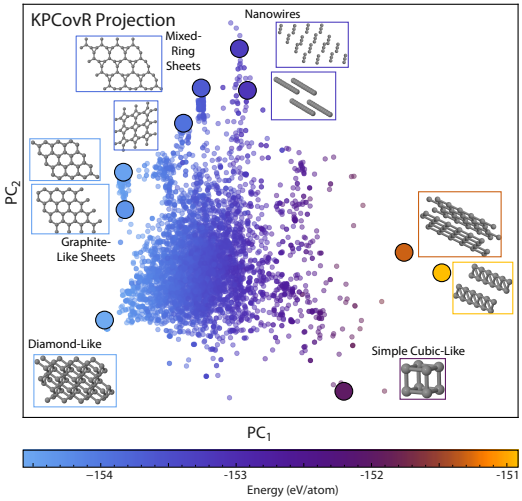


`https://www.materialscloud.org/discover/kpcovr/carbons-10`

MC, *Unsupervised machine learning in atomistic simulations, between predictions and understanding*, JCP (2019)

Beyond unsupervised maps

- Kernel PCA map of a dataset of carbon structures
- KPCovR reveals more clearly structure/stability relations

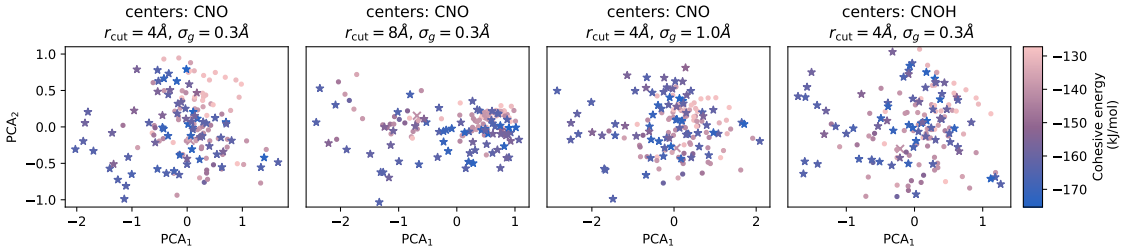


https://www.materialscloud.org/discover/kpcovr/carbons-05

Helfrecht, Cersonsky, Fraux, **MC**, MLST (2020); https://chemiscope.org

A honest bias

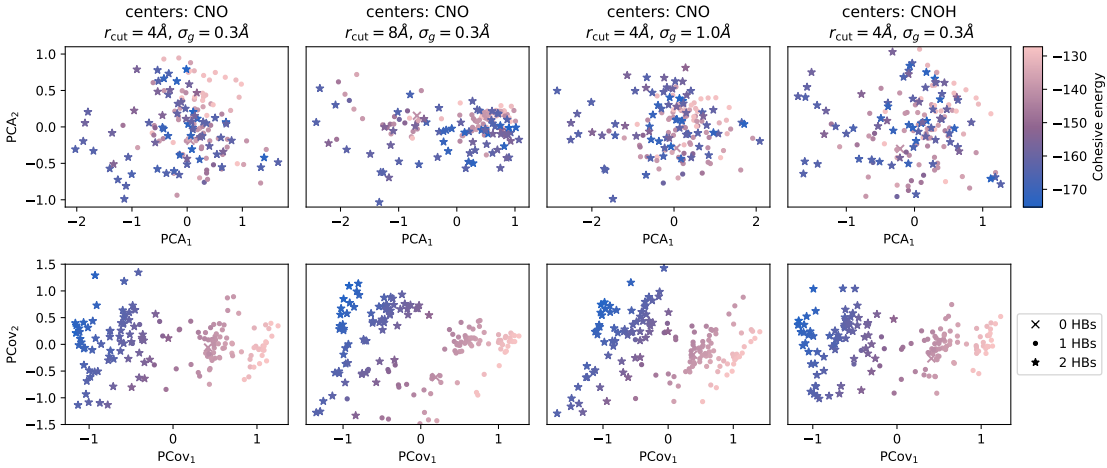
- Explicit bias in structure-property maps using KPCovR reduces arbitrary dependence of latent space from hyperparameters
- More compelling structure-property relations *beyond those directly included as targets*



Helfrecht, Cersonsky, Fraux, **MC**, MLST (2020); Musil et al., Chem. Rev. (2021)

A honest bias

- Explicit bias in structure-property maps using KPCovR reduces arbitrary dependence of latent space from hyperparameters
- More compelling structure-property relations *beyond those directly included as targets*



Helfrecht, Cersonsky, Fraux, **MC**, MLST (2020); Musil et al., Chem. Rev. (2021)

Wrapping up

Summary

- Data-centric methods are here to stay, and can be used in atomic-scale modeling in many different tasks
- Balancing act between generality, interpretability, and incorporation of prior knowledge
- ‘‘Simple’’ methods often have advantages, particularly in the data-poor regime. Kernels offer a good balance
- Crucial role of the structural representations, especially for unsupervised tasks

MC, "Unsupervised machine learning in atomistic simulations, between predictions and understanding," JCP 150(15), 150901 (2019)
Helfrecht, Cersonsky, Fraux, MC, "Structure-property maps with Kernel principal covariates regression," MLST 1(4), 045021 (2020)

scikit-learn

scikit-matter https://github.com/scikit-learn-contrib/scikit-matter

chemiscope http://chemiscope.org