



Multiphysics and Multiscale Mechanics
Research Group (M3RG), IIT Delhi



IIT Delhi

Applications of Machine learning



Workshop organized by DBT-Apex BTIC, ICGEB

N. M. Anoop Krishnan¹

¹Indian Institute of Technology Delhi, Hauz Khas, New Delhi, India

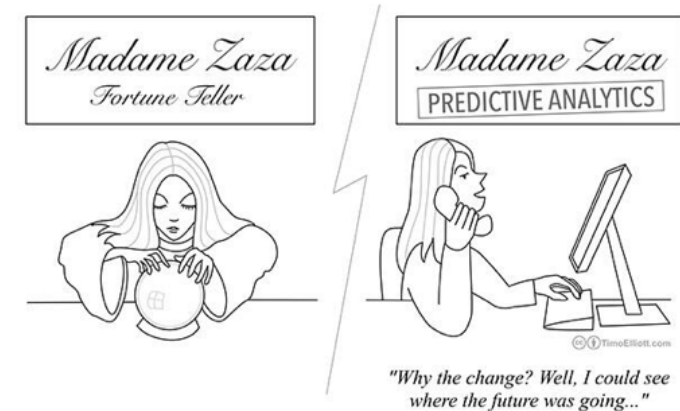
Motivation

Atoms to Property

- High throughput experiments
- Synthesis and characterization
- In silico models (MD,MC)  **Physics-based modeling**
- Databases as information repository
- Automate search for correlations
- **Machine Learning**  **Data-based modeling**

Machine Learning

- Purely data-driven methods
- Uses available data to identify a hidden pattern or trend and “learn” progressively by self-correction
- No physical equation/model is assumed or used for the training
- **Capability:** Clustering, regression, dimensionality reduction, property prediction, anomaly detection
- **Applications:** Personal assistants (google, alexa), traffic prediction, email spam forwarding, fraud detection, online ads, face recognition, and lot more



Warning!!!

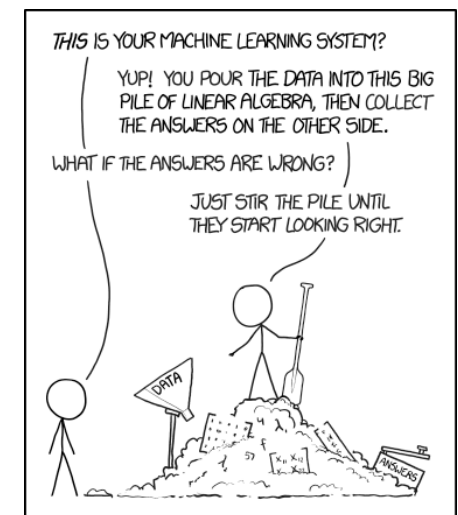
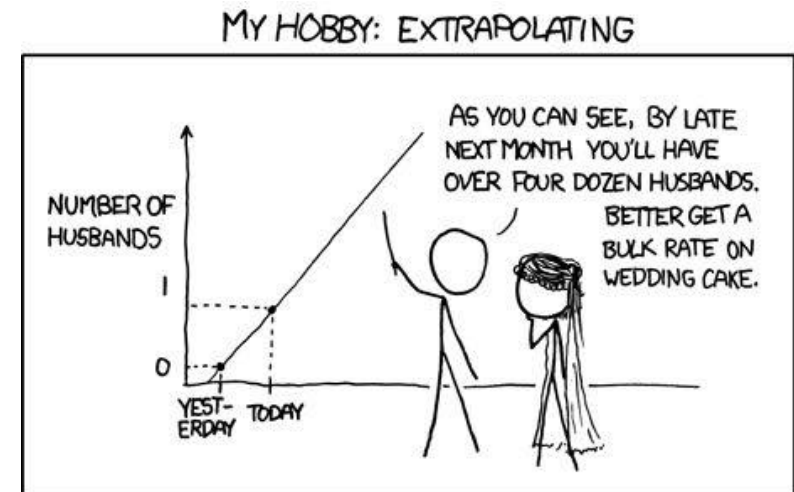


Image courtesy: https://imgs.xkcd.com/comics/machine_learning.png

Machine Learning

- Traditional prediction/extrapolation schemes: linear or non-linear regression
- Machine learning algorithms: classification and regression
- Some commonly used methods: support vector machine, decision tree, random forest, neural network, gaussian process regression, k-means, etc.
- Important note: Machine learning is essentially a statistical model
- Data used for training the model should be “proper”: representative of the sample space



Warning!!!

Image courtesy: https://imgs.xkcd.com/comics/machine_learning.png

Machine Learning

nature
REVIEWS
GENETICS

Review Article | Published: 07 May 2015

Machine learning applications in genetics and genomics

Maxwell W. Libbrecht & William Stafford Noble 

Nature Reviews Genetics **16**, 321–332 (2015) | [Download Citation](#) ↓

nature | methods

Review Article | Published: 15 July 2019

Machine-learning-guided directed evolution for protein engineering

Kevin K. Yang, Zachary Wu & Frances H. Arnold 

Nature Methods **16**, 687–694 (2019) | [Download Citation](#) ↓

- **Extremely useful in biotechnology and genetics as the data is inherently complex and extremely non-linear**

nature
biotechnology

Primer | Published: 01 December 2006

What is a support vector machine?

William S Noble

Nature Biotechnology **24**, 1565–1567 (2006) | [Download Citation](#) ↓

nature
biotechnology

News & Views | Published: 07 August 2015

Deep learning for regulatory genomics

Yongjin Park & Manolis Kellis 

Nature Biotechnology **33**, 825–826 (2015) | [Download Citation](#) ↓

Machine Learning

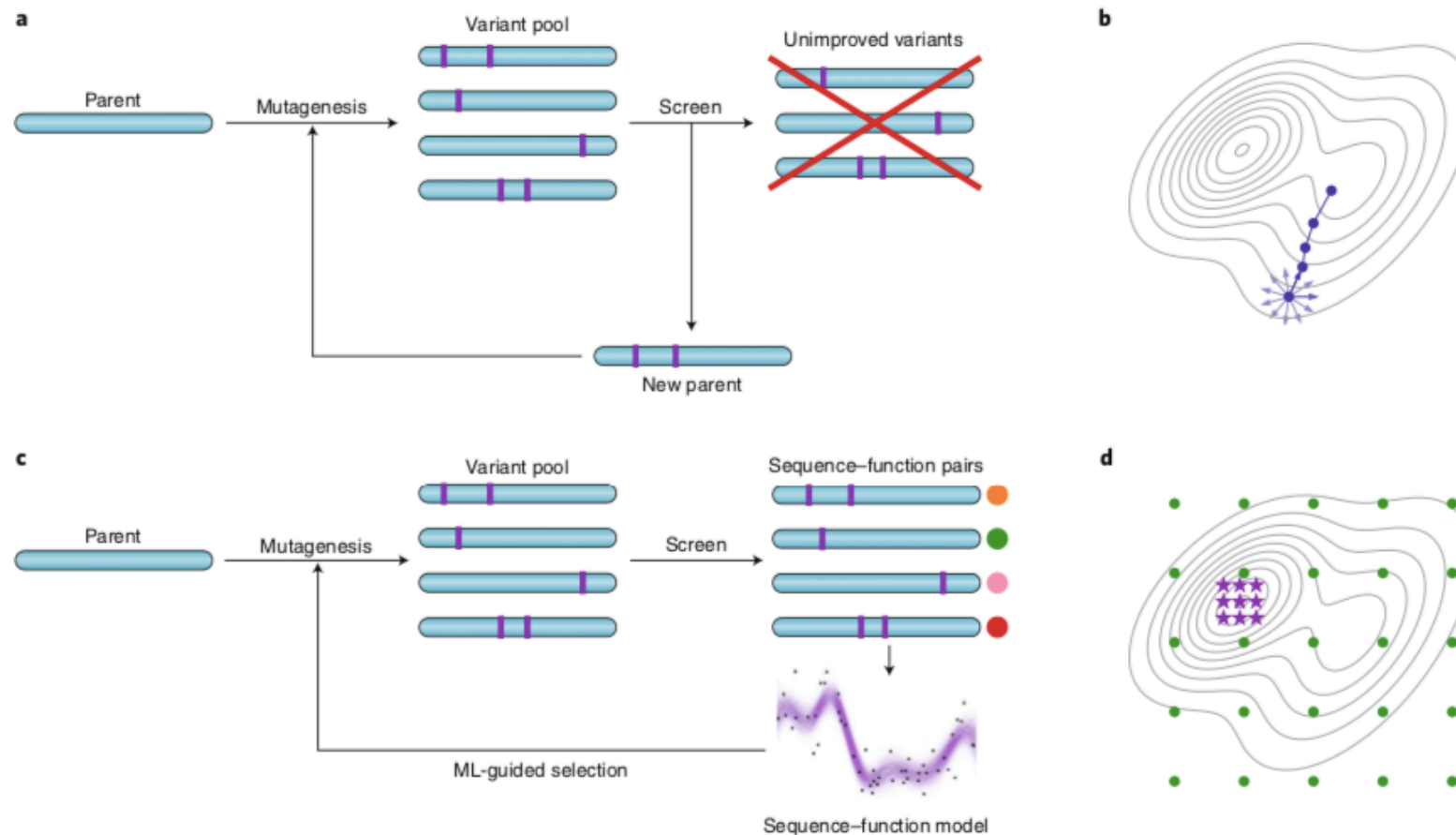
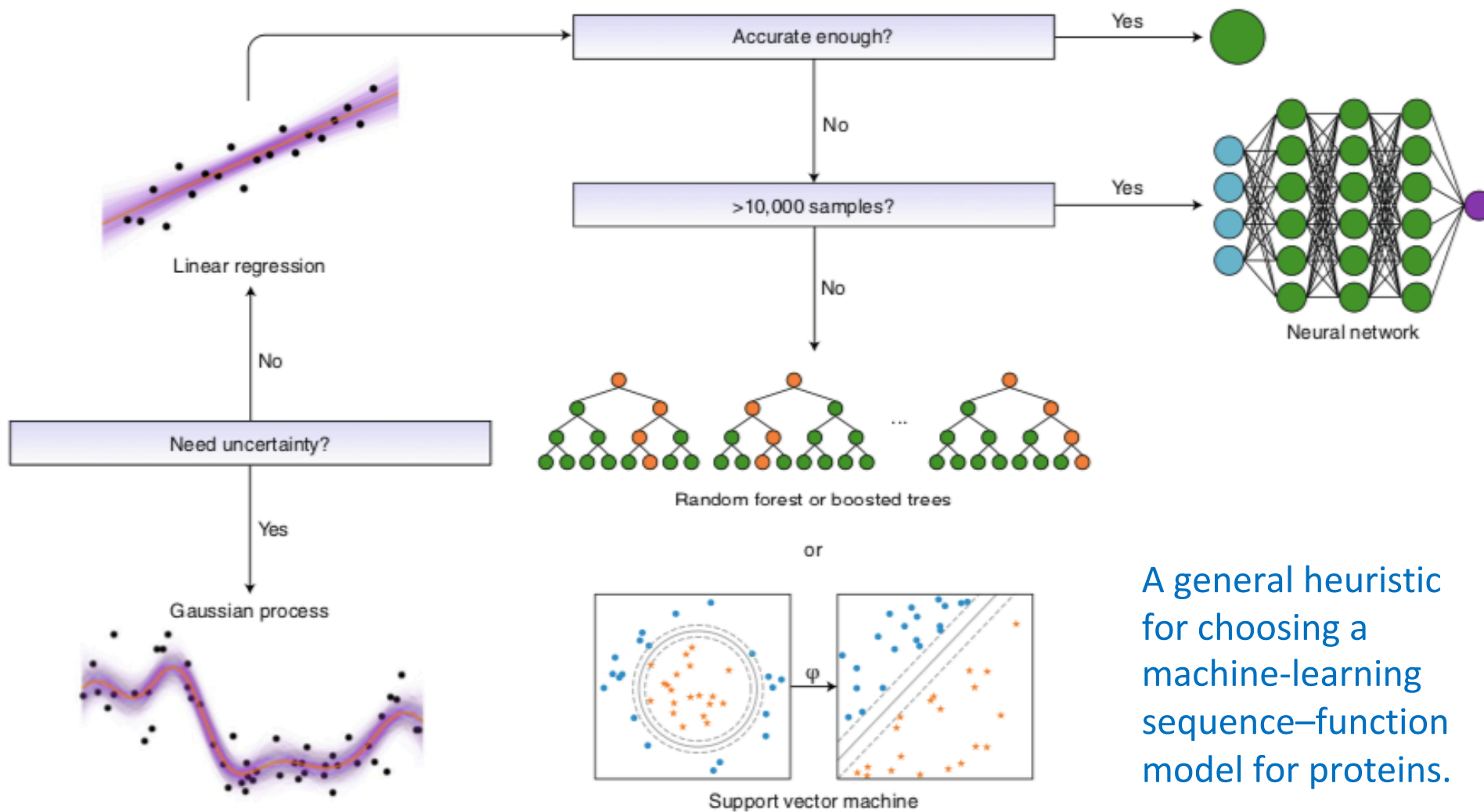


Fig. 1 | Directed evolution with and without machine learning. **a**, Directed evolution uses iterative cycles of diversity generation and screening to find improved variants. Information from unimproved variants is discarded. **b**, Directed evolution is a series of local searches on the function landscape. **c**, Machine-learning (ML) methods use the data collected in each round of directed evolution to choose which mutations to test in the next round. Careful choice of mutations to test decreases the screening burden and improves outcomes. **d**, Machine-learning-guided directed evolution often rationally chooses the initial points (green circles) to maximize the information learned from the function landscape, thereby allowing future iterations to quickly converge to improved sequences (violet stars).

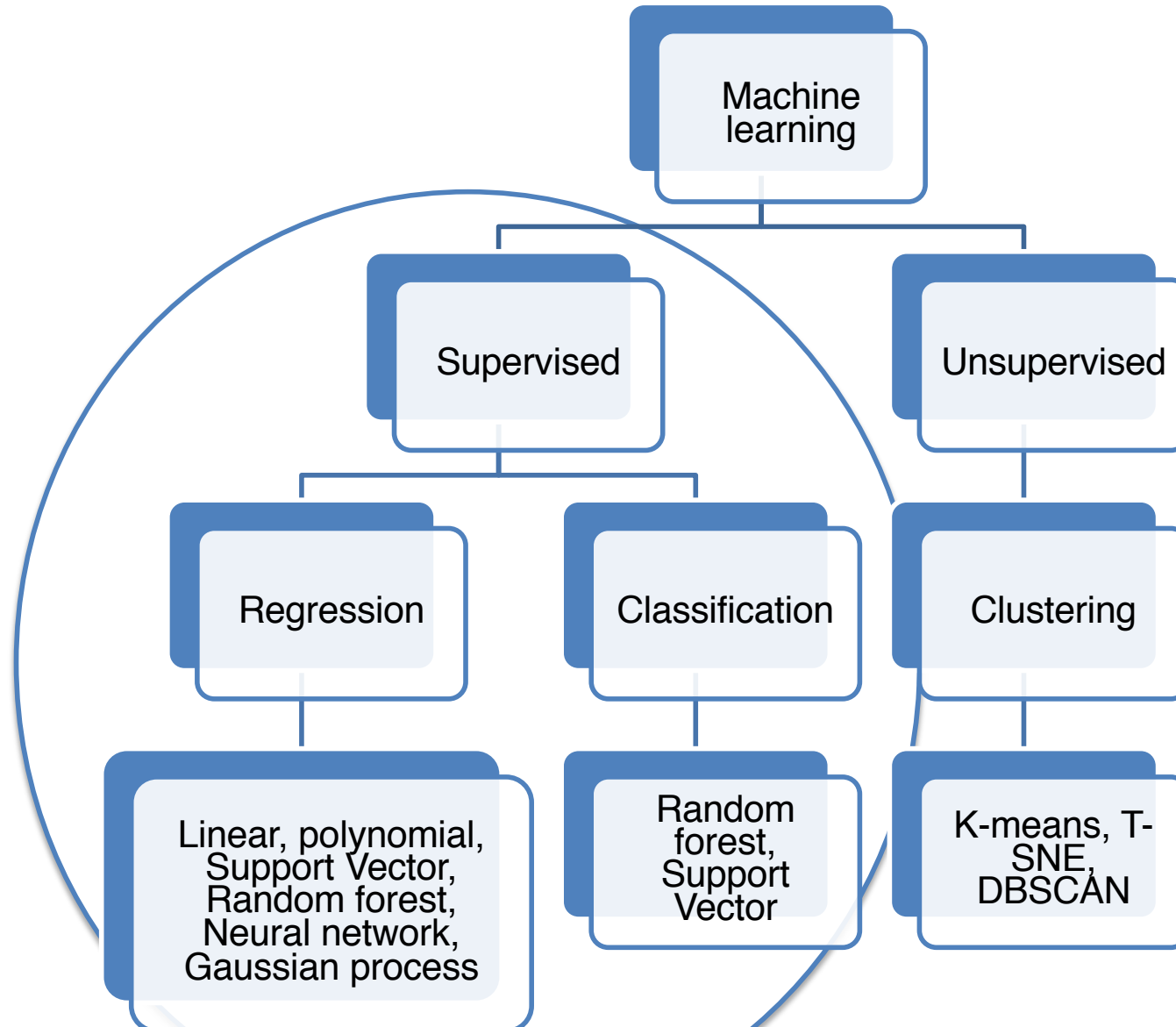
Yang, K.K., Wu, Z. and Arnold, F.H., 2019. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, p.1.

Machine Learning



Yang, K.K., Wu, Z. and Arnold, F.H., 2019. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, p.1.

Machine Learning



Machine learning: Typical steps

- Dataset preparation: Cleaning the dataset, imputing missing data, removing outliers
- Preparation of the training and testing dataset: Training dataset should be representative of the major features of the data
- Dimensionality reduction: identifying the important variables, removing the unnecessary variables, combining multiple dependent variables
- Identifying the appropriate mathematical model: depends on the size and nature of the data
- Training, testing and validation: training the model using the training set to maximize the predictive capability while avoiding overfitting
- Hyperparametric optimization to simplify the model and increase its interpretability

Linear regression methods

- Linear regression: Minimize the residual sum of squares (RSS)

$$RSS(\beta) = \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

where y_i is the measured value at the i^{th} observation with features $x_{ij} = (x_{i1}, x_{i2}, \dots, x_{ip})$

- Biased estimators: lasso and elastic regressions—introduce bias and shrink the coefficients of insignificant variable to zero thereby reducing the effect of outliers from the training set
- Lasso regression:

$$\hat{\beta}_{lasso} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

- Elastic-net:

$$\hat{\beta}_{elastic} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \frac{1-\lambda}{2} \sum_{j=1}^p \beta_j^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

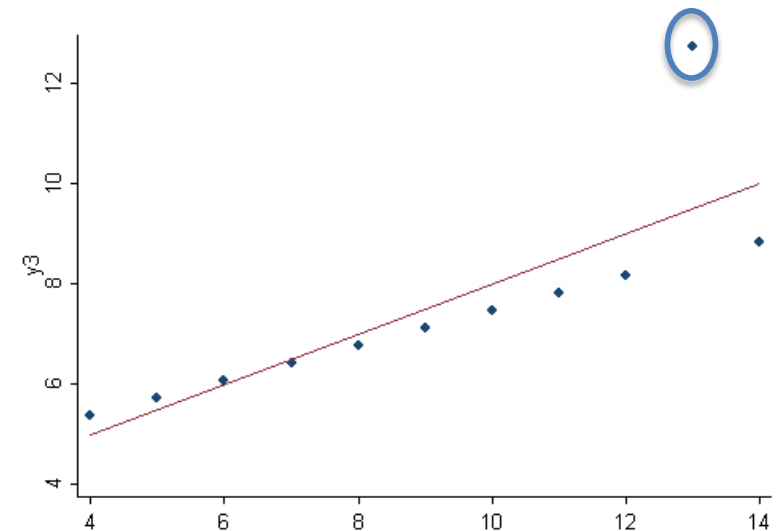
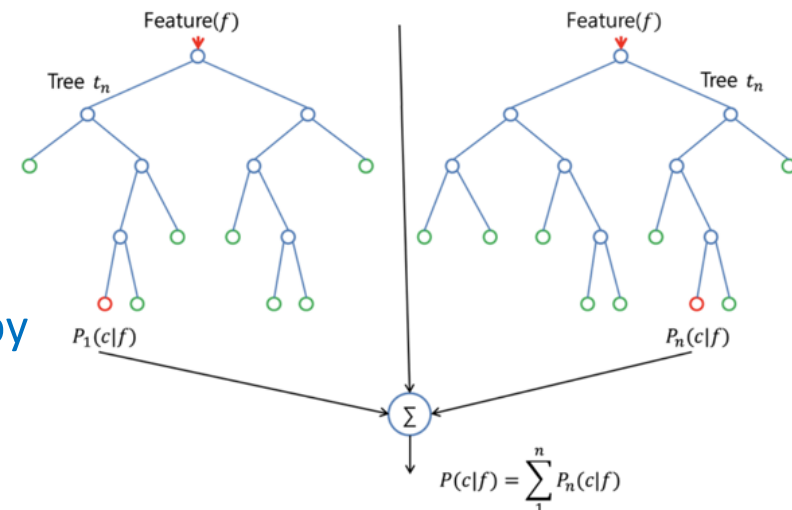


Image courtesy: <http://m.cdn.blog.hu/gu/guruloordo/image/extrapolating%5B1%5D.png>

Random forest (RF) method

- Builds multiple decision trees and compiles it together
- Used for classification and regression
- Algorithm:
 1. Generate samples from the training data set.
 2. Create a decision from each training sample by selecting best split/features
 3. For each iteration, predict the data out of the sample using the tree grown
 4. Predict the output of a new data set by averaging the aggregate of predictions of n_t decision trees
- RF leads to discrete nodes— results might not be accurate for intermediate points

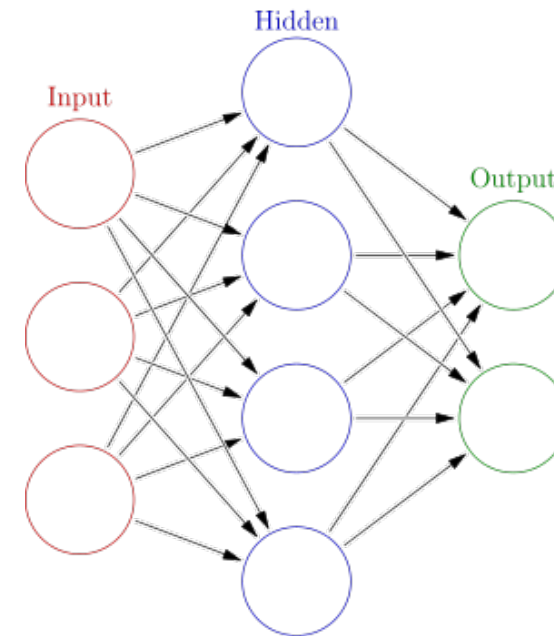


A typical RF network with nodes representing “test”, branch representing the outcome of the test and leaf (the last node) representing the decision

Image courtesy: https://cdn-images-1.medium.com/max/1600/0*tG-IWcxL1jg7RkT0.png

Neural Network (NN)

- Nonlinear function inspired from the biological behavior of neurons
- Consists of the input, hidden, and output layers where hidden layer consists of a number of neurons connecting input and output layer.
- Multiple hidden layers with large number of neurons are possible
- For ANN, the data is split into training (55%), validation (15%), and test (30%) sets
- Such a partition reduces the chance of overfitting



A typical ANN consisting of input nodes, hidden layers and output node

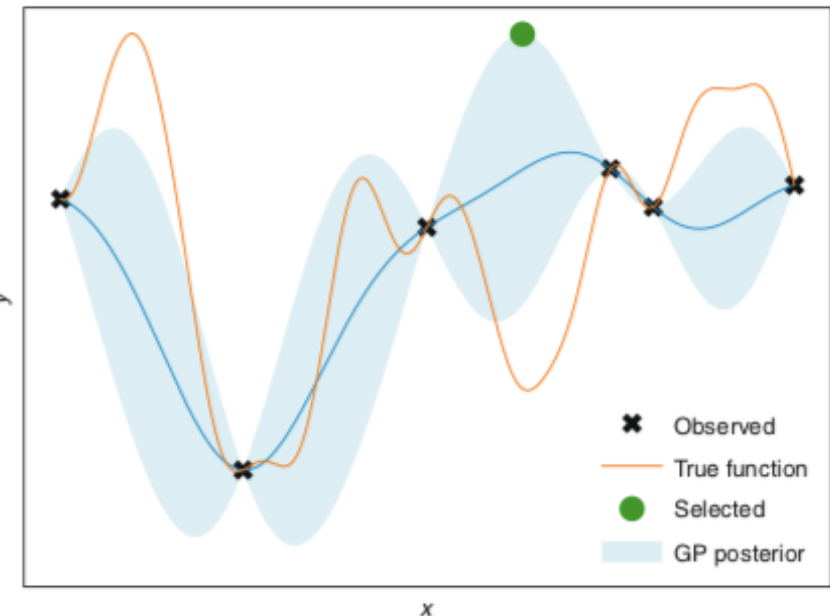
Image courtesy: https://cdn-images-1.medium.com/max/1600/0*tG-IWcxL1jg7RkTO.png

Designing Neural Network

- Designing a NN consists of structural optimization and hyperparametric optimization
- Structural optimization refers to fixing the optimal size of the neural network (no of hidden layers and hidden layer units)
- Note: number of neurons should not be too large in comparison to the number of inputs (less than twice). Try to go for the simplest network possible.
- Hyperparametric optimization refers to optimizing the weights of the neurons to eliminate unnecessary complexities in the network (note that this is applicable to other methods such as PCA as well)

Gaussian process regression

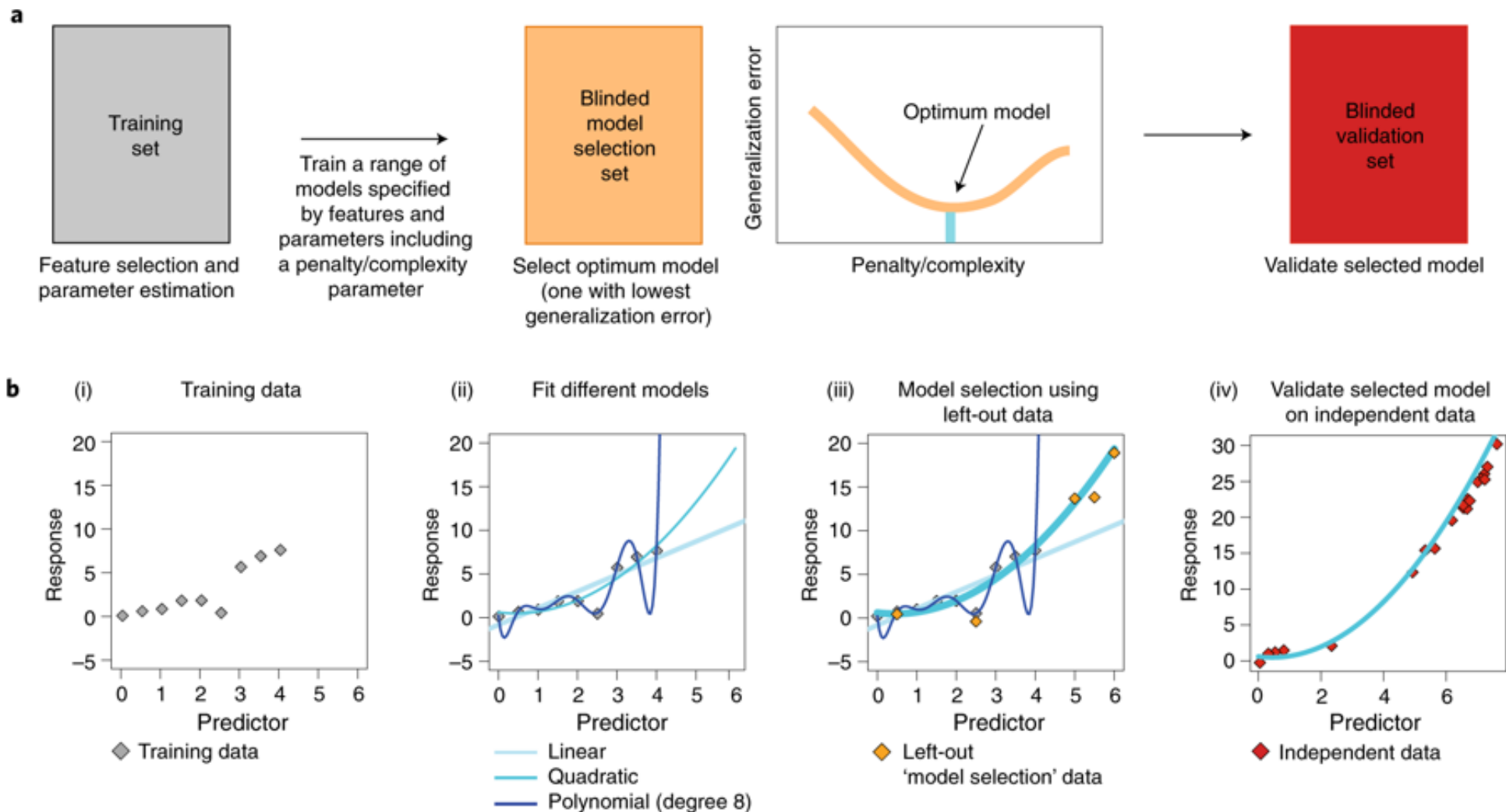
- Gaussian process regression (GPR): distribution of a function that relates input to output is developed
- Non-parametric method unlike NN, or least square regression
- Different kernel functions can be used
- Automatic relevance detection (ARD) can be used along with any kernel to improve the predictions
- Instead of a predicted value, the uncertainty associated with the prediction is also obtained—thanks to the underlying distribution



A typical GPR—line represents the mean values, dark shadow represents σ and lighter shadow represents 2σ

Common Pitfalls

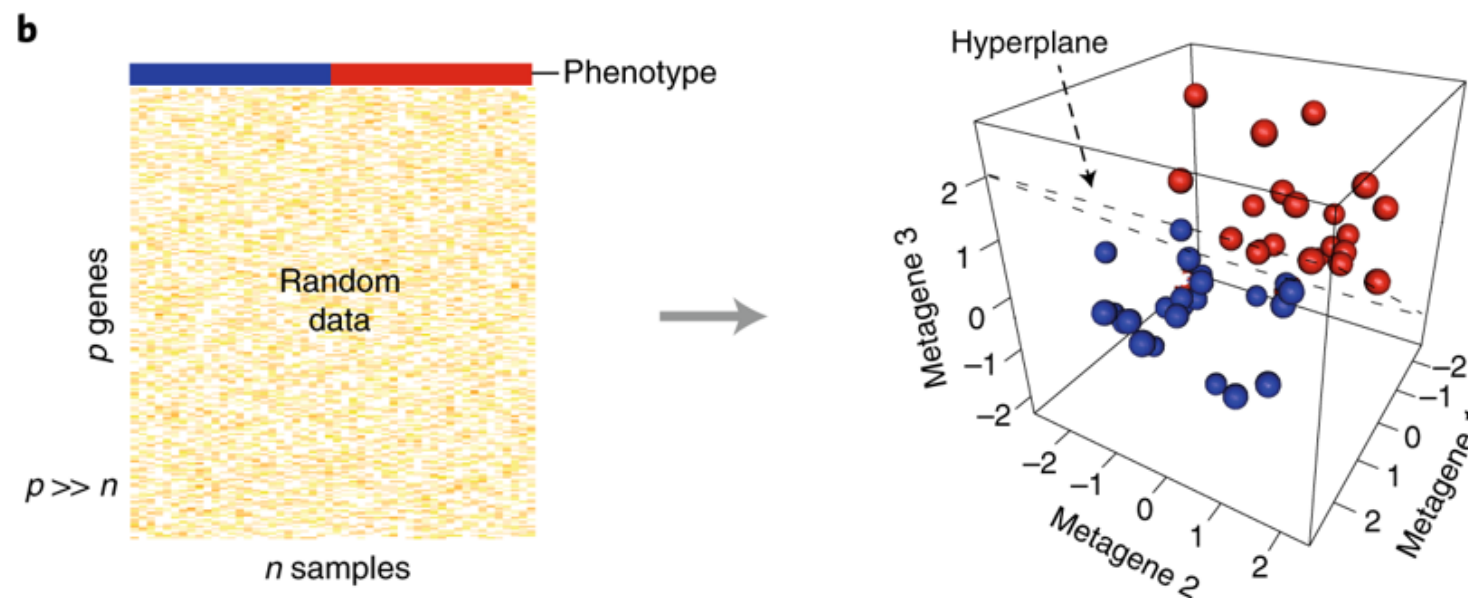
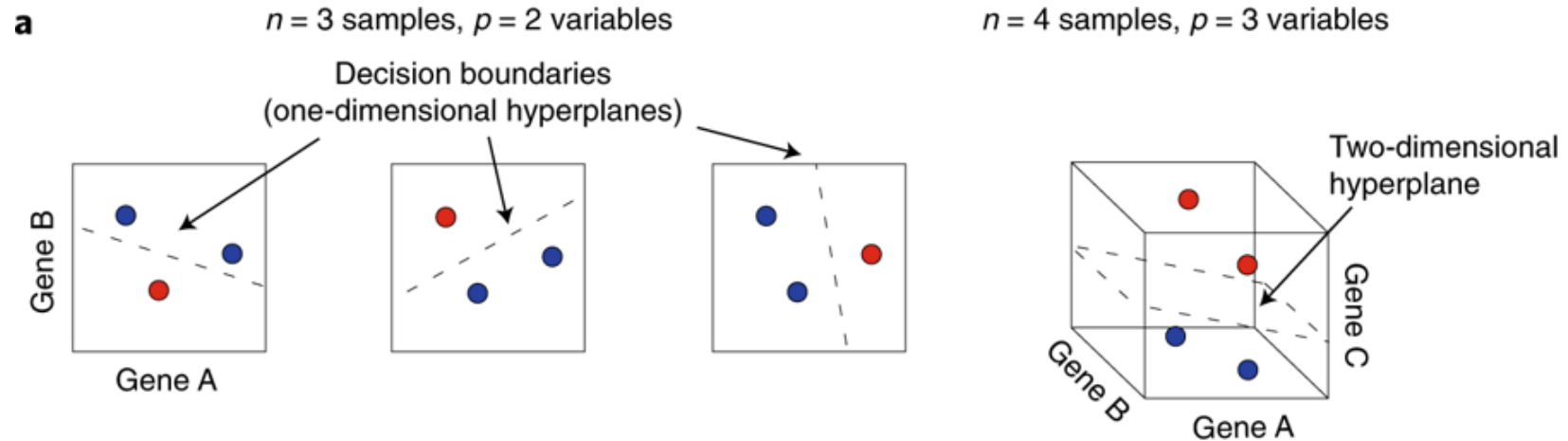
- **Overfitting** -> sol: include more data, cross-validation, regularization



<https://www.nature.com/articles/s41563-018-0241-z>

Common Pitfalls

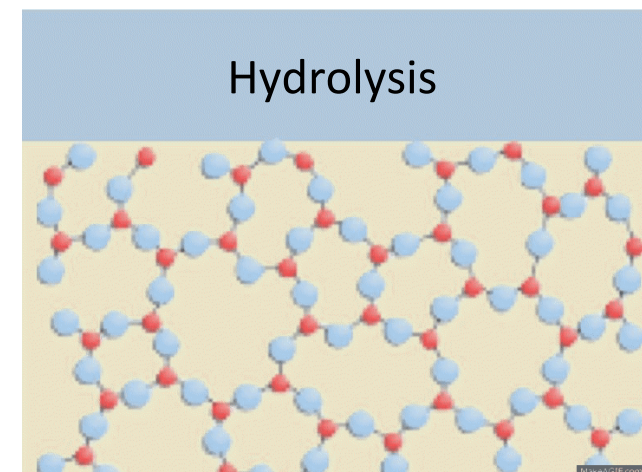
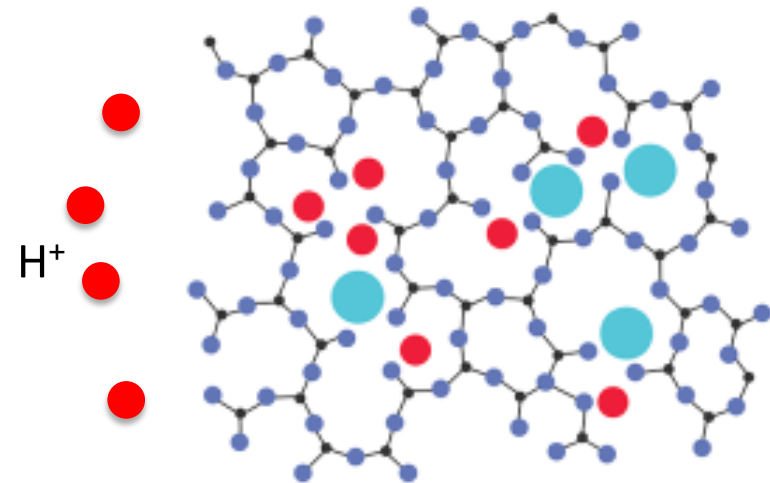
- Curse of dimensionality -> sol: regularization



<https://www.nature.com/articles/s41563-018-0241-z>

Dissolution in silicate glasses

- Glasses exposed to aqueous environment undergo dissolution
- Dissolution occurs through ion-exchange and hydrolysis
- Dependence of dissolution rate on the structure, composition, solution pH etc. is poorly understood
- Extensive experimental and simulation studies are available in the literature
- **Aim: To use data-driven methods to predict dissolution rate for the accelerated design of silicate glasses**



Data set: Sodium aluminosilicate glasses

- Experimental studies on the dissolution rate of eight sodium aluminosilicate glasses by Hamilton et. al.^{1,2}
- Albite glass ($\text{Na}_2\text{O}-\text{Al}_2\text{O}_3-6\text{SiO}_2$), jadeite glass ($\text{Na}_2\text{O}-\text{Al}_2\text{O}_3-4\text{SiO}_2$), nepheline glass ($\text{Na}_2\text{O}-\text{Al}_2\text{O}_3-2\text{SiO}_2$), and $\text{Na}_2\text{O}-x\text{Al}_2\text{O}_3-(3-x)\text{SiO}_2$ glasses, where $x = 0.0, 0.2, 0.4, 0.6, \text{ and } 0.8$.
- Dissolution kinetics was assessed both in acidic and caustic conditions, specifically, $\text{pH} = 1, 2, 4, 6.4, 9, \text{ and } 12$ with five to seven regular intervals (e.g., 24, 49, 96, 168, and 336h) of solvent contact.
- **Dissolution rates (of mol SiO_2) obtained from this experimental measurements were used to train and test machine-learning based algorithms**

¹ Hamilton, J.P., Brantley, S.L., Pantano, C.G., Criscenti, L.J. & Kubicki, J.D. Dissolution of nepheline, jadeite and albite glasses: toward better models for aluminosilicate dissolution. *Geochimica et Cosmochimica Acta*, 65(21), pp.3683-3702, 2001.

² Hamilton, J.P., *Corrosion behavior of sodium aluminosilicate glasses and crystals*, 1999.

ML based predictive models

- We focus on three broad classes of regression methods—linear regression including lasso and elastic net, random forest (RF), and artificial neural network (ANN).
- Model inputs: (i) glass composition, (ii) pH of the solution
- Model output: dissolution rate in mol $\text{SiO}_2/\text{cm}^2/\text{s}$
- Temperature and pressure were maintained constant in the experiments and hence is not used as model input
- Total data points: 299 (70% training and 30% test set)**

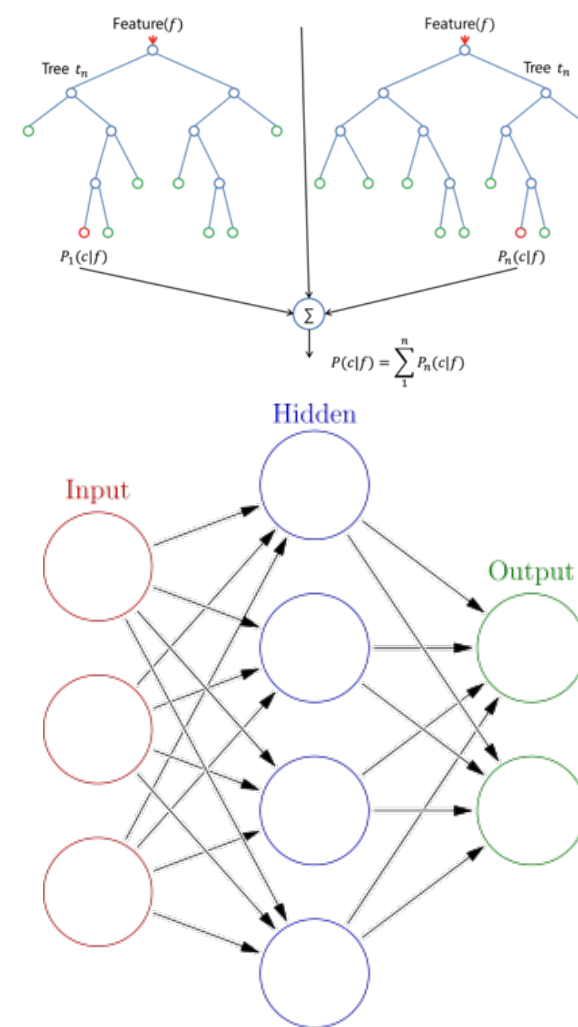
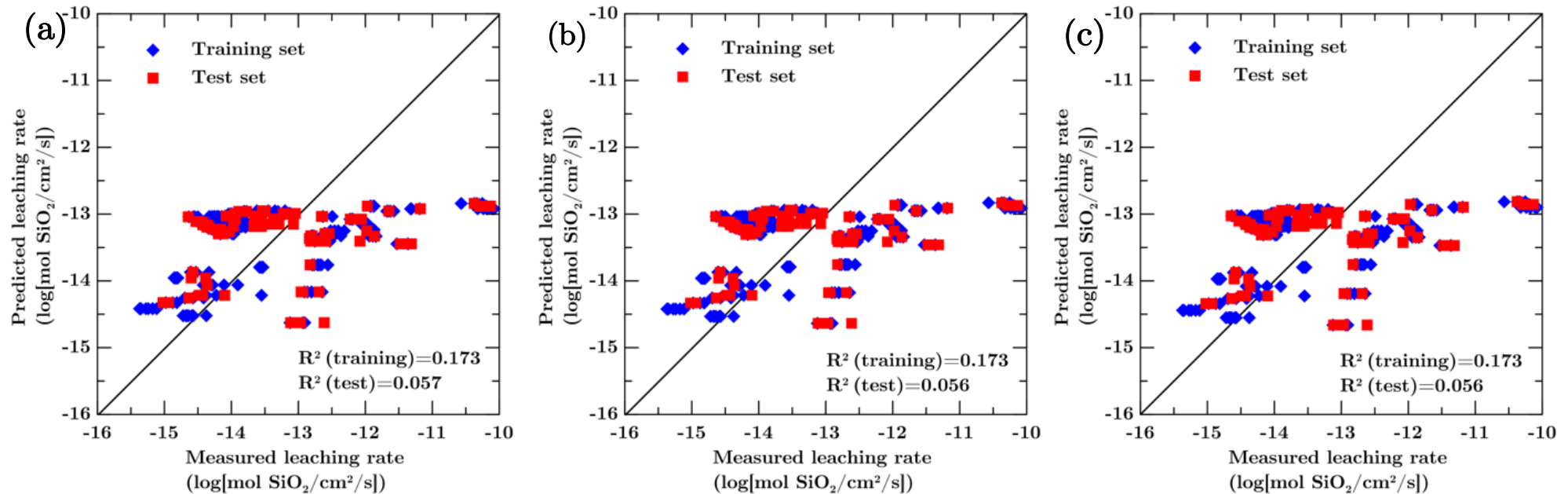


Image courtesy: https://cdn-images-1.medium.com/max/1600/0*tG-IWcxL1jg7RkTO.png

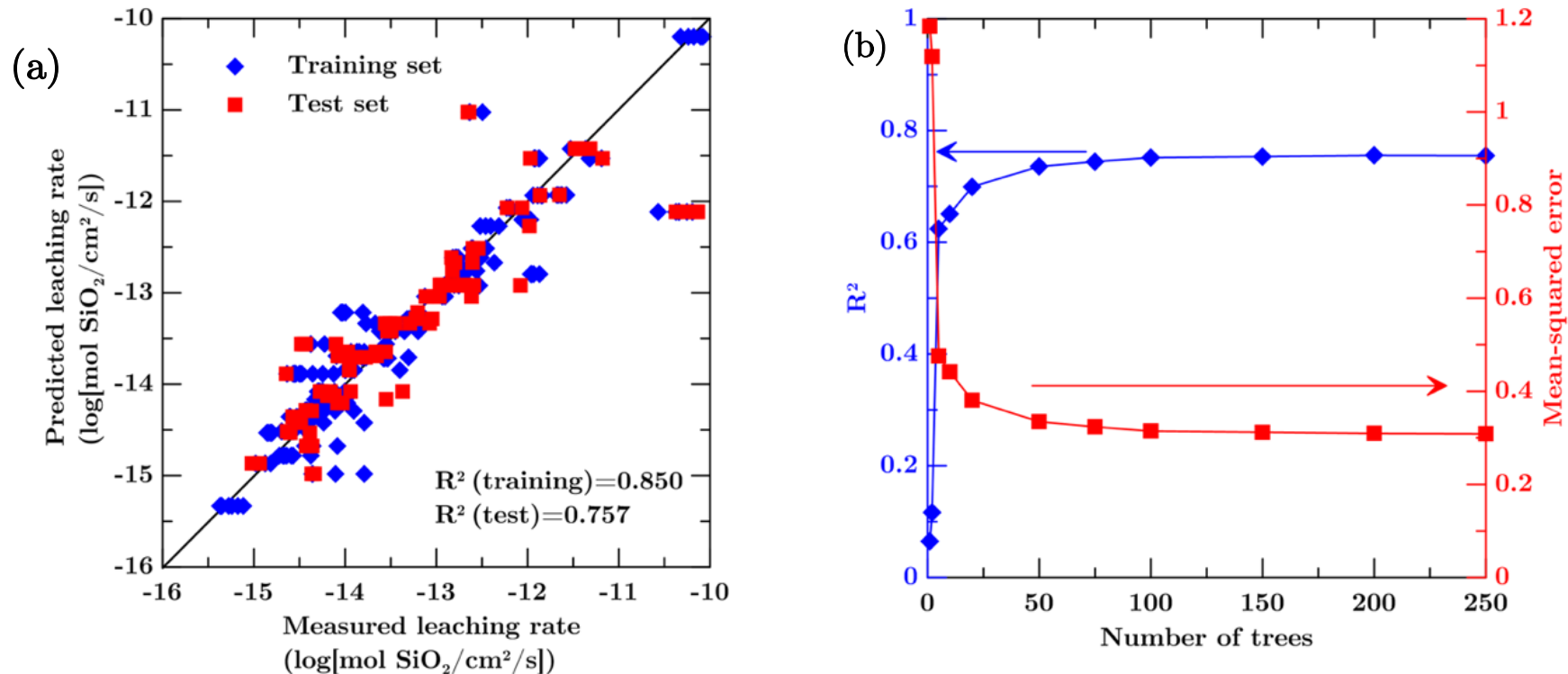
Results: Linear regression methods



Predicted leaching rates (in log[mol SiO₂/cm²/s]) using (a) linear regression, (b) lasso regression, and (c) elastic net, compared to the measured values.

All the linear regression methods fail in predicting the trend in the dissolution rate of silicate glasses

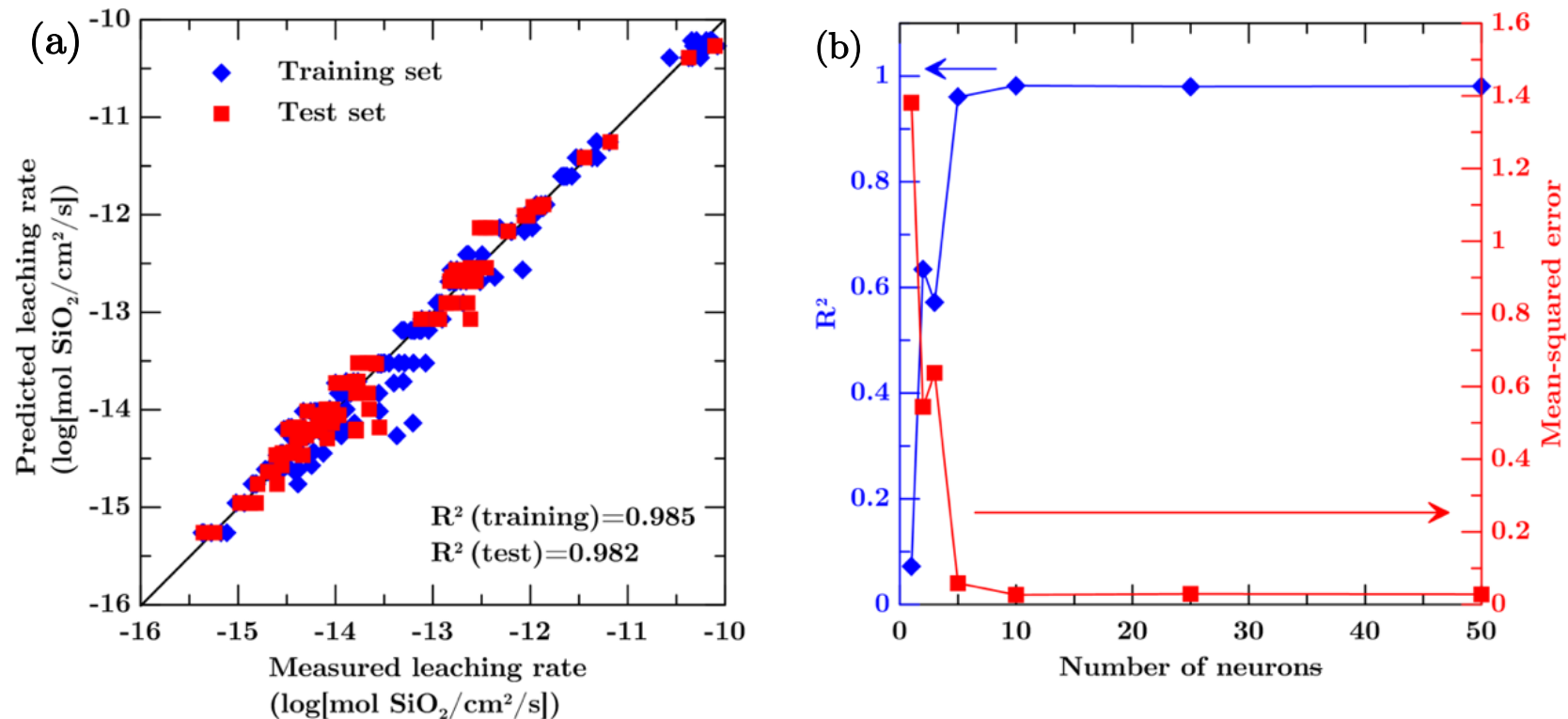
Results: Random forest (RF)



(a) Predicted leaching rates (in log[mol SiO₂/cm²/s]) using random forest, compared to the measured values. (b) R² (left axis) and mean-squared error (right axis) values of the test set with respect to the number of trees used in the random forest algorithm. The prediction converges for ~100 trees.

RF provides a reasonable prediction of the dissolution rates

Results: Neural Network (NN)



(a) Predicted leaching rates (in log[mol SiO₂/cm²/s]) using the artificial neural network approach, compared to the measured values. (b) R^2 (left axis) and mean-squared error (right axis) of the test set with respect to the number of neurons in the neural network. Prediction converges for ~10 neurons.

NN provides the best prediction of the dissolution rates