PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- **Machine Learning Overview**
- **Examples of Machine Learning**
- **Energy Machine Learning**

Michael Pyrcz, The University of Texas at Austin

Motivation

Learn the concepts common to a variety of machine learning approaches:

- Inference and prediction
- Training and testing
- Parameters and hyperparameters

PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- **Machine Learning Overview**

Michael Pyrcz, The University of Texas at Austin

Big Data

Big Data, you have big data if your data has a combination of these:

Volume: many data samples, difficult to handle and visualize

Velocity: high rate collection, continuous relative to decision making cycles

Variety: data form various sources, with various types and scales

Variability: data acquisition changes during the project

Veracity: data has various levels of accuracy

"Energy has been big data long before tech learned about big data."

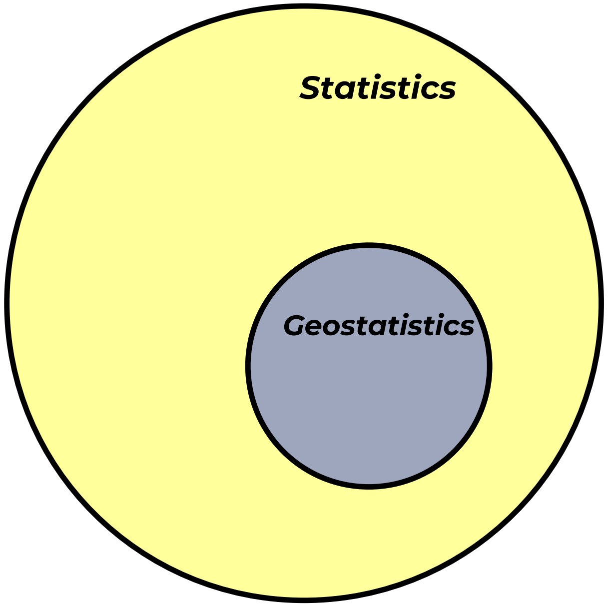
– Michael Pyrcz

Big Data Analytics

Statistics is collecting, organizing, and interpreting data, as well as drawing conclusions and making decisions.

Geostatistics is a branch of applied statistics:

1. the spatial (geological) context
2. the spatial relationships
3. volumetric support
4. uncertainty



Proposed Venn diagram for statistics and geostatistics.

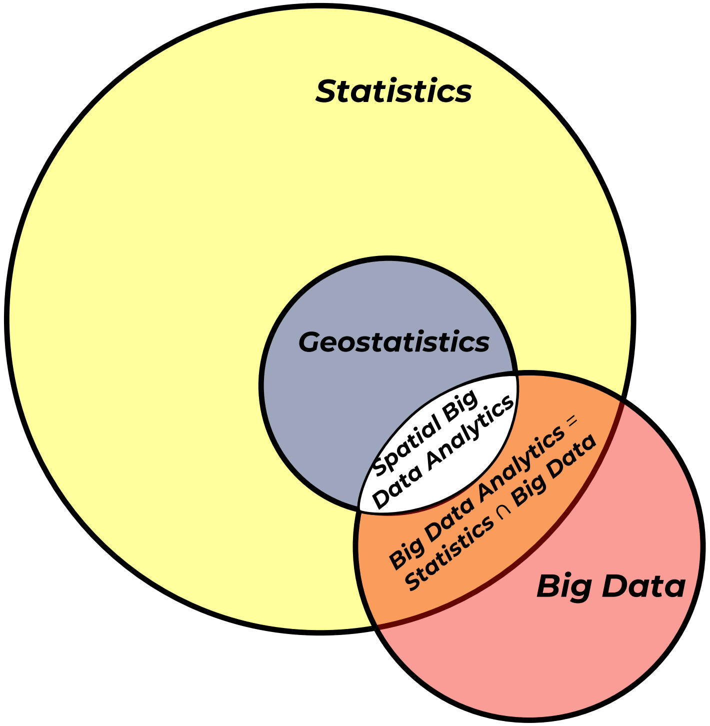
Big Data Analytics

Data Analytics is the analysis of data to support decision making.

Big Data Analytics is the process of examining large and varied data sets to discover patterns and make decisions.

Spatial Big Data Analytics is expert use of spatial statistics / geostatistics on big data to support decision making.

'Data analytics is the use of statistics and visualization'



Proposed Venn diagram for spatial big data analytics.

Machine Learning

toolkit

“… is the study of algorithms and mathematical models
learning
that computer systems use to
progressively improve their performance on a specific task.
Machine learning algorithms build a mathematical model
training with data
of sample data, known as "training data",
general
in order to make predictions or decisions
without being explicitly programmed to perform the task.”

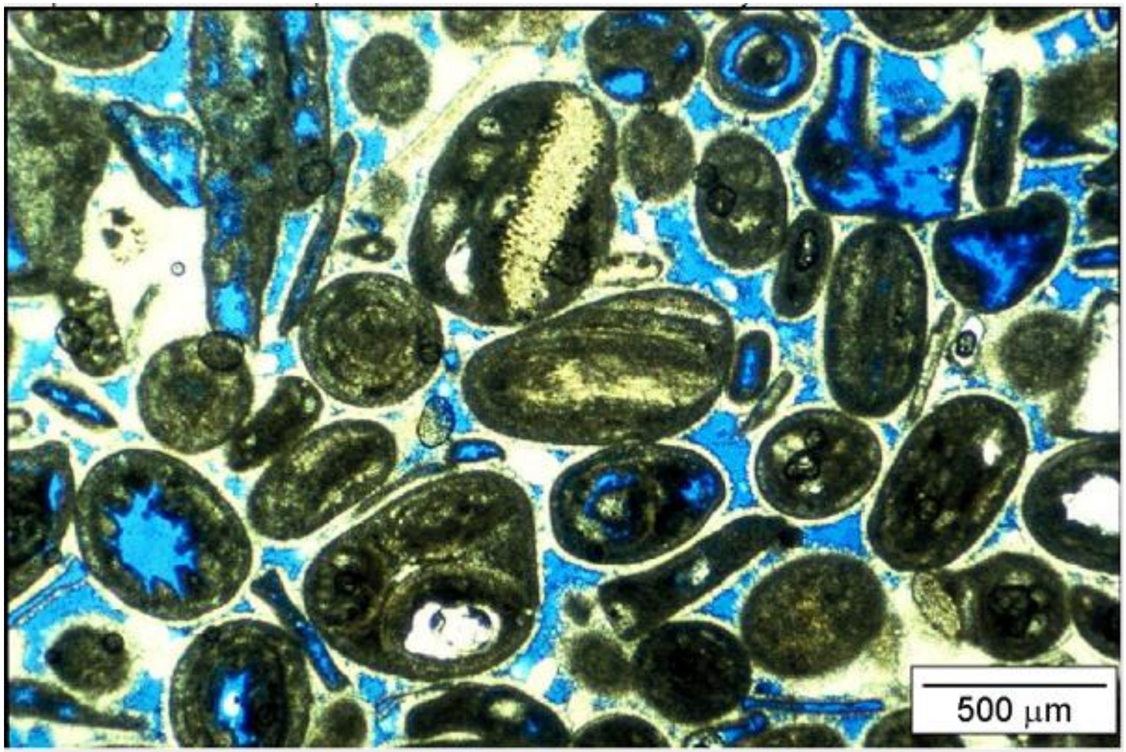
not a panacea

“… where it is
infeasible to develop an algorithm of specific instructions for performing the task.”

Quote from Machine Learning - Wikipedia

Variables / Features

- **Variable / Feature**: any property measured / observed in a study
  - e.g. porosity, permeability, mineral concentrations, saturations, contaminant concentration
  - in data mining / machine learning this is known as a **feature**
  - measure often requires **significant analysis, interpretation** etc.



Total Porosity
all blue area

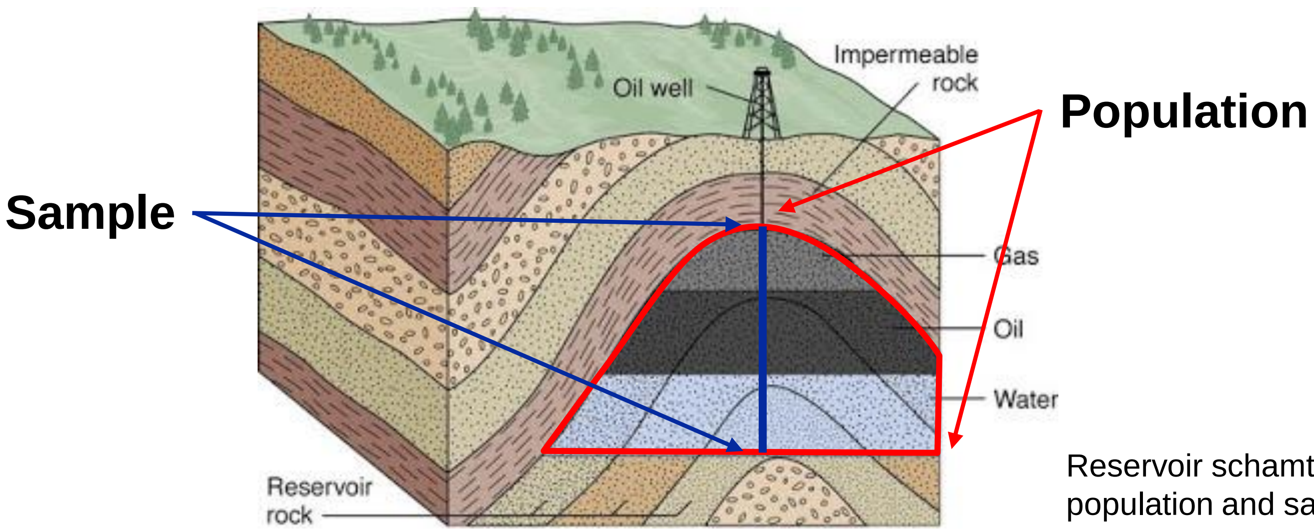
Effective Porosity
all connected blue area

Carbonate thin section from BEG, UT Austin from course by F. Jerry Lucia. http://www.beg.utexas.edu/lmod/_IOL-CM07/old-4.29.03/cm07-step05.htm

Population and Sample

- **Population**: Exhaustive, finite list of property of interest over area of interest. Generally the entire population is not accessible.
  - e.g. exhaustive set of porosity at each location within a reservoir

- **Sample**: The set of values, locations that have been measured
  - e.g. porosity data from well-logs within a reservoir

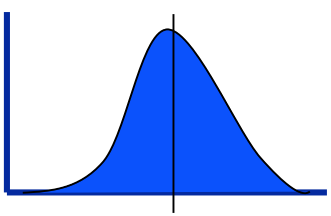


Reservoir schamtic with population and sample shown.

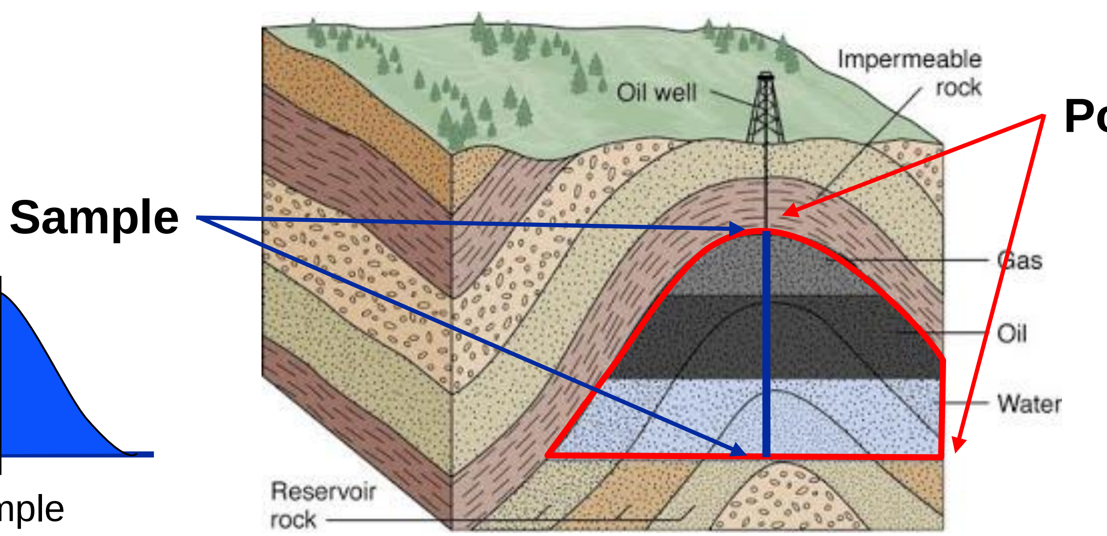
Modified Image from https://energyeducation.ca/encyclopedia/Oil_and_gas_reservoir

Parameter and Statistic

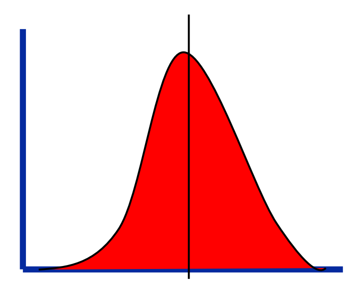
- **Parameters**: summary measure of a population
  - e.g. population mean, population standard deviation, we rarely have access to this
  - **model parameters** is different in machine learning, and we will cover later.
- **Statistics**: summary measure of a sample
  - e.g. sample mean, sample standard deviation, we use statistics as estimates of the parameters



Sample
Mean is a Statistic



Reservoir schematic with parameter and statistic shown.



Population
Mean is a Parameter

Modified Image from https://energyeducation.ca/encyclopedia/Oil_and_gas_reservoir

Machine Learning Nuts and Bolts

The Model

Response **Predictors** **Random Error**

$$Y = f(X_1, \dots, X_m) + \epsilon$$

Predictors (or Independent) Features (or Variables)

- the model inputs

Response (or Dependent) Features (or Variables)

- the model outputs

Machine Learning is All About Estimating the model, f, for two purposes:

- Inference or Prediction

Inference

What is the relationship between each predictor feature?

$$f(X_1, \ldots, X_m)$$

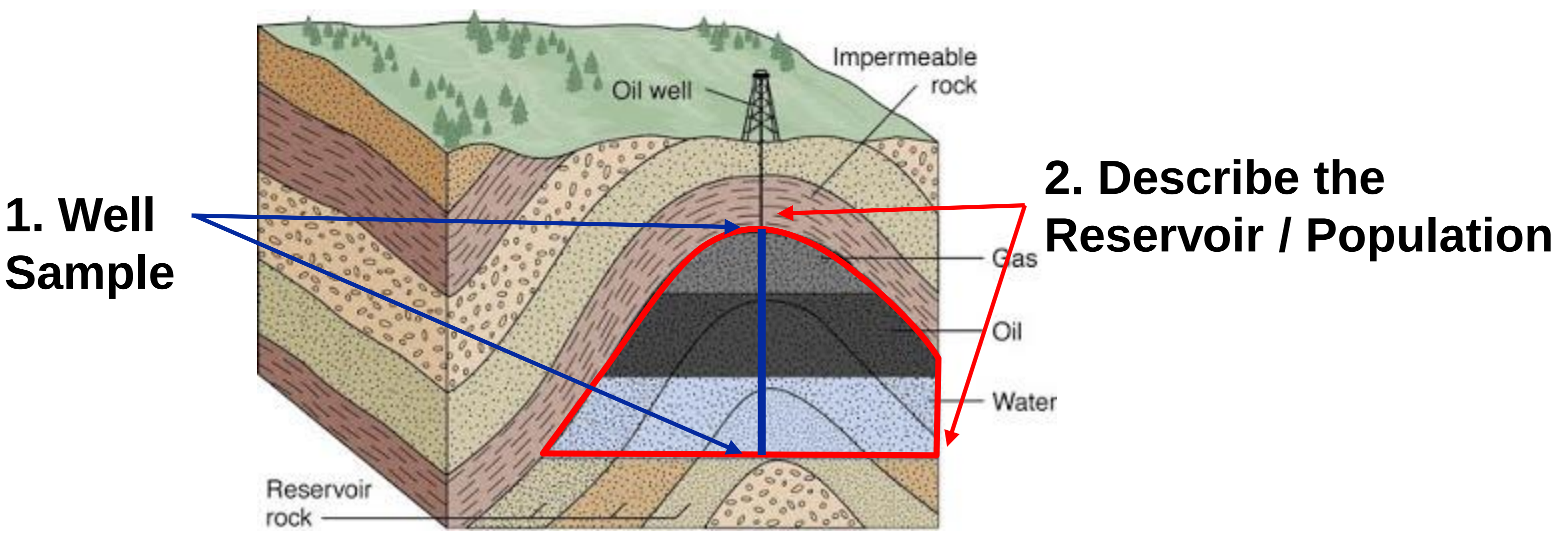
- sense of the relationship (positive or negative)?
- shape of relationship (sweet spots)?
- relationships may depend on values of other predictors!

Recall, Inferential Statistics

- given a sample, describe the population
- e.g. given 3 heads and 7 tails, what is the probability the coin is fair?

Inference

- **Inferential Statistics**
  – Given a random sample from a population, describe the population
  – Given the well(s) samples, describe the reservoir



Reservoir schematic with inference problem, given well sample, describe the reservoir, population.

Modified Image from https://energyeducation.ca/encyclopedia/Oil_and_gas_reservoir

Prediction

The best prediction of the response feature

$$\hat{Y} = \hat{f}(X_1, \dots, X_m) + \epsilon$$

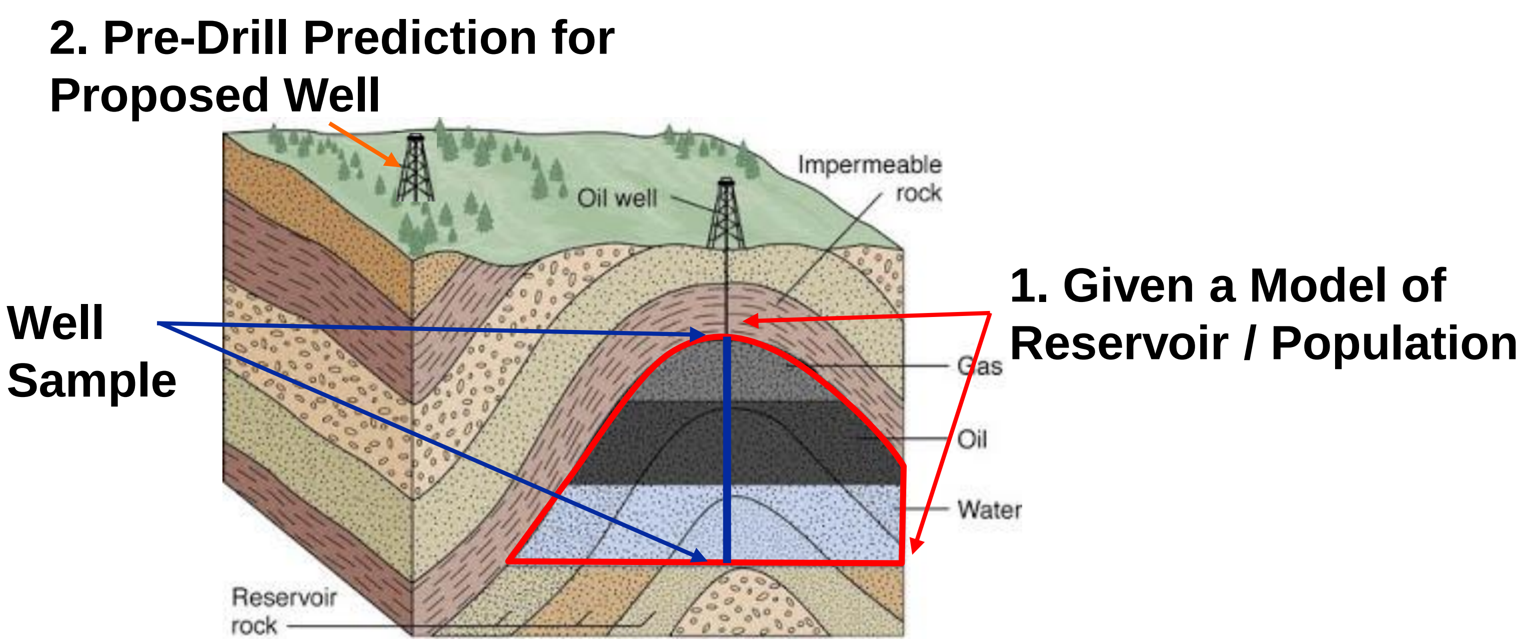
- Estimate the function, $\hat{f}$, for the purpose of predicting $\hat{Y}$
- We are focused on getting the most accurate estimates, $\hat{Y}$, where $\hat{Y}$ is an estimate of Y

Recall, Predictive Statistics

- given an assumption about the population, predict the outcome in the next sample
- e.g., given a fair coin what is the probability of 3 heads and 7 tails?

Prediction

- **Predictive Statistics**
  - Predict the samples given assumptions about the population
  - Given our model of the reservoir, predict the next well (pre-drill assessment) sample, e.g. porosity, permeability, production rate etc.



Reservoir schematic with inference problem, given well sample, describe the reservoir, population.

Modified Image from https://energyeducation.ca/encyclopedia/Oil_and_gas_reservoir

Parametric Models

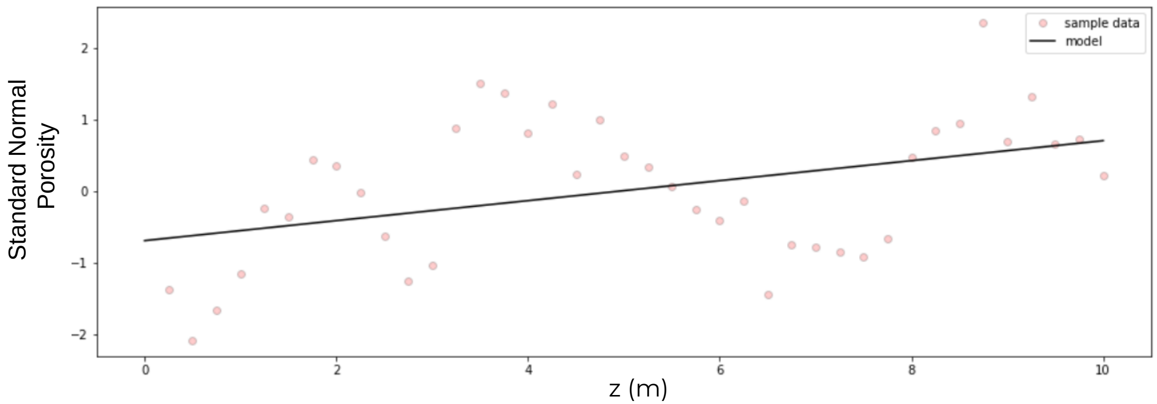
Working with Parametric Models

- Makes an assumption about the functional form, shape
- We gain simplicity and advantage of only a few parameters

- For example, here is a linear model:

$$Y = f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_m X_m$$

There is a risk that $\hat{f}$ is quite different than f, then we get a poor model!



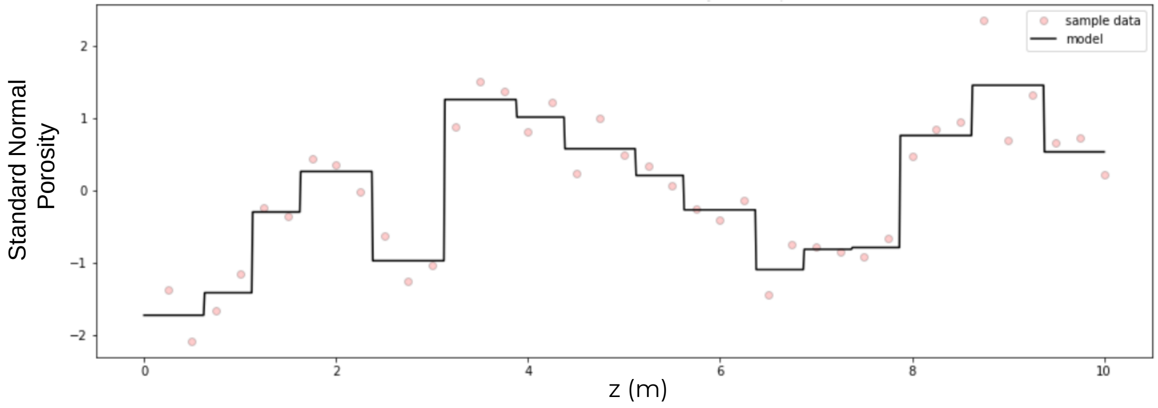
Linear regression model to predict porosity from the z coordinate.

Nonparametric Models

Working with Nonparametric Models

- Makes no assumption about the functional form, shape
- More flexibility to fit a variety of shapes for f
- Less risk that $\hat{f}$ is a poor fit for f
- Typically need a lot more data for an accurate estimate of f

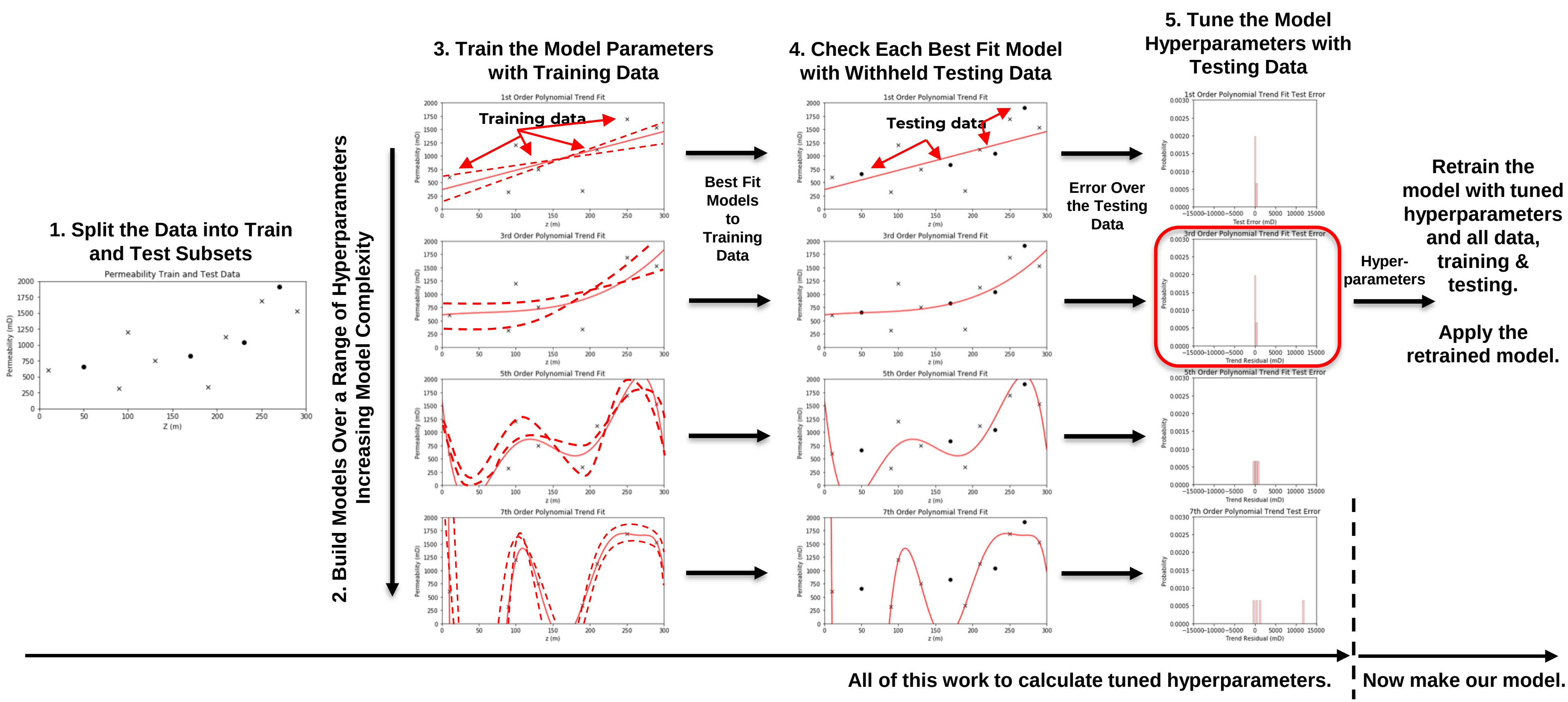
‘Nonparametric is actually parametric rich!’



Decision tree regression model to predict porosity from the z coordinate.

Model Workflow

Building Machine Learning Models



Machine learning model building workflow to avoid overfit.

Model Parameters

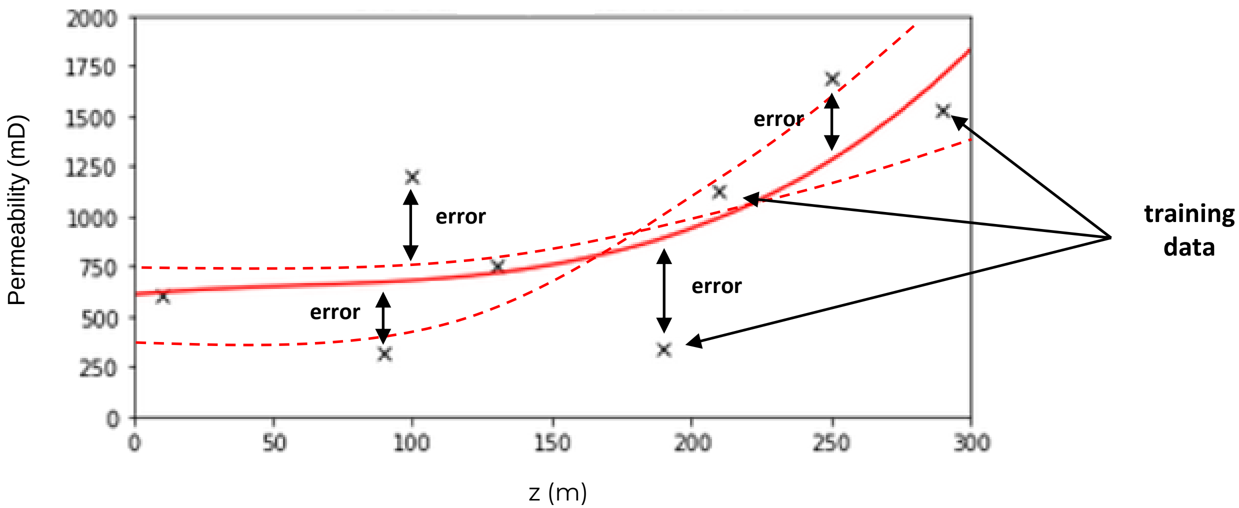
Machine Learning Model Parameters

Model Parameters

- Fit during training phase to minimize error at the training data
- For this 3rd order polynomial:

$$k = b_3 z^3 + b_2 z^2 + b_1 z + c$$

Parameters:
b_3, b_2, b_1 and c



Setting model parameters to minimize the error relative to training data.

Model Hyperparameters

Machine Learning Model Hyperparameters

Model Hyperparameters

- Constrain the model complexity.
- Select hyperparameters that maximize accuracy with the testing data.

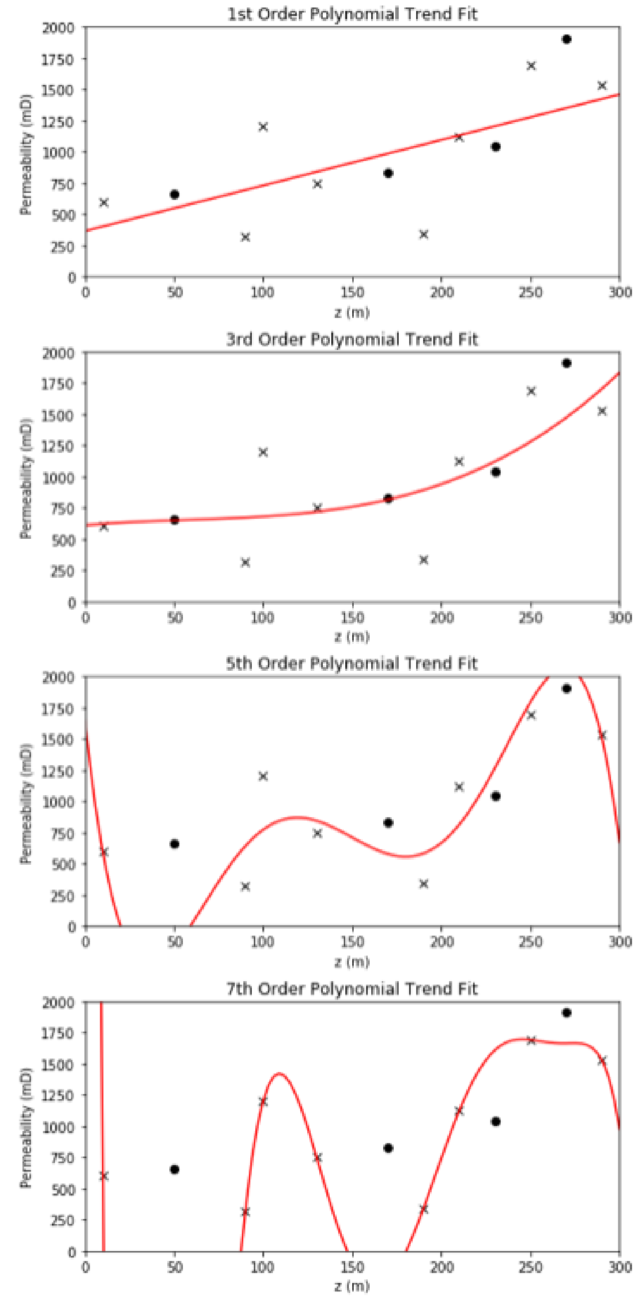
- For a polynomial model:

Increasing Complexity

3rd Order: $k = b_3 z^3 + b_2 z^2 + b_1 z + c$

2nd Order: $k = b_2 z^2 + b_1 z + c$

1st Order: $k = b_1 z + c$



Varying the model complexity, model hyperparameter, to maximize fit with testing data.

Assessing Model Accuracy

Method Selection is Important

- No one method performs well on all datasets.
- Based on experience, understanding the data and limitations of the methods

Measuring Quality of Fit in Training

- for regression, the most common measure is the mean square error

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left[\left(y_i - \hat{f}(x_i^1, \ldots, x_i^m)\right)^2\right]$$

for $i = 1, \ldots, n$ training data and for $1, \ldots, m$ features.

where we have n observations of training data for response y_i, and predictor $x_i^1, \ldots, x_i^m$ features.

Assessing Model Accuracy

Measuring Quality of Fit in Testing / Real-world Use

- The challenge is that that real question we have is how well can we predict outside the training data, testing data.

$$MSE = E\left[\left(y_0 - \hat{f}(x_0^1, \dots, x_0^m)\right)^2\right]$$

for $i = 1, \dots, n$ training data and for $1, \dots, m$ features.

where we have observations of the response, y_0, and predictor features not used to train the model, $x_0^1, \dots, x_0^m$.

- Recall, E is the expectation. A probability weighted average, given equal probability the same as the arithmetic average.

- We want to know how our model performs when we move away from the training data!

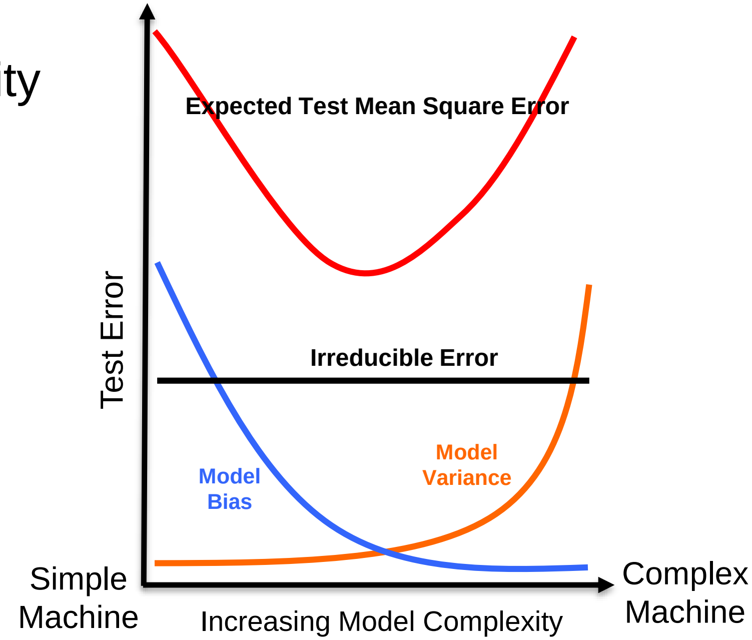
Model Bias Variance Trade-Off

The Components of Error in Testing / Real-world Use

The Expected Test Square Error components:

$$\mathbf{E}\left[\left(y_0 - \hat{f}(x_1^0, \dots, x_m^0)\right)^2\right] = \underbrace{\left(E[\hat{f}(x_1^0, \dots, x_m^0)] - f(x_1^0, \dots, x_m^0)\right)^2}_{\text{Model Bias}^2} + \underbrace{E\left[\left(\hat{f}(x_1^0, \dots, x_m^0) - E[\hat{f}(x_1^0, \dots, x_m^0)]\right)^2\right]}_{\text{Model Variance}} + \underbrace{\sigma_e^2}_{\text{Irreducible Error}}$$

- **Model Variance** is error due to sensitivity to the dataset
- **Model Bias** is error due to using an approximate model
- **Irreducible Error** is due to missing variables and limited samples



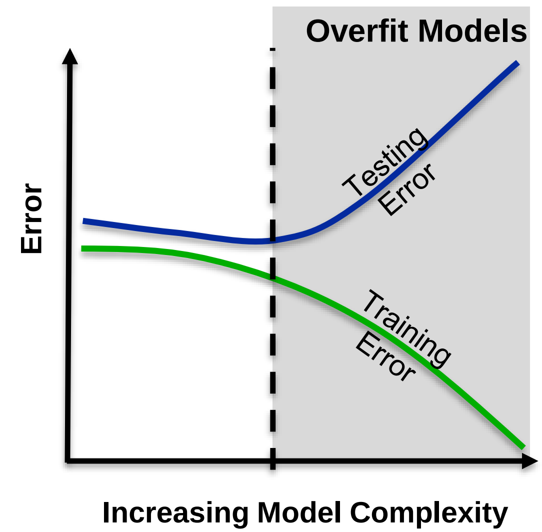
Model variance and bias trade-off.

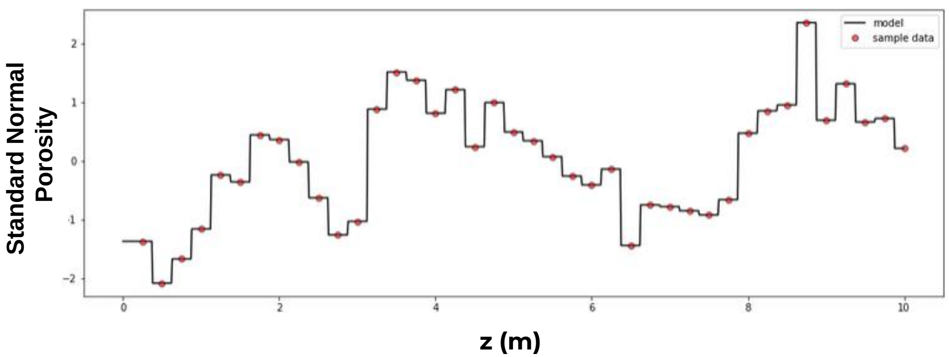
Model Overfit

Machine Learning Model Overfit

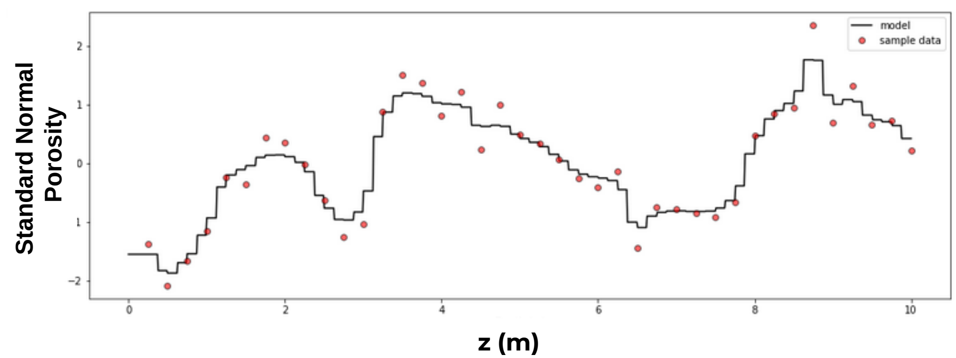
Model Overfit

- Fitting data noise / data idiosyncrasies
- Increased complexity will generally decrease error with respect to the training dataset
- but, may result in increase error with testing data → at this complexity/flexibility we are overfit!





Overfit model to training data.



A more balanced fit model to training data.

Training and Testing

The Training and Testing Split

- the most common approach is random selection
- this may not be fair testing

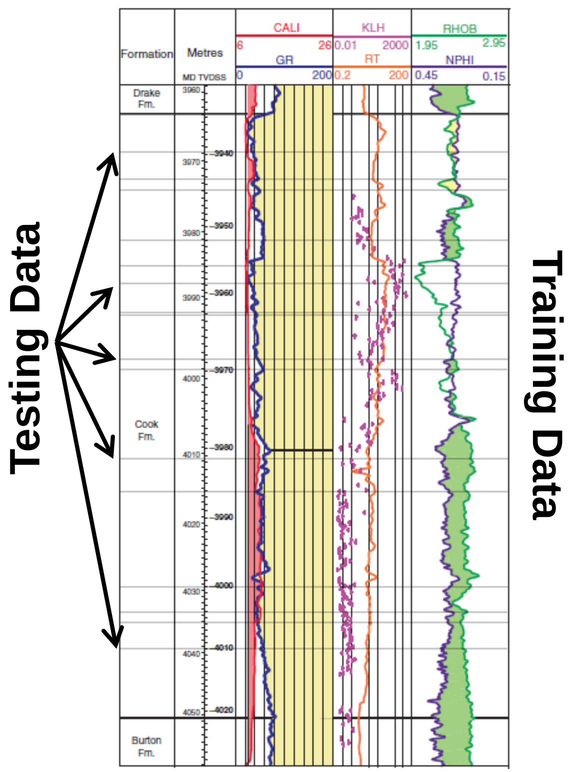
Fair Testing

- the range of testing difficulty is similar to the real-world use of the model
- too easy – testing cases are the same or almost the same as training cases, random sampling is often too easy!
- too hard – testing cases are very different from the training cases, the model is expected to severely extrapolate

Training and Testing

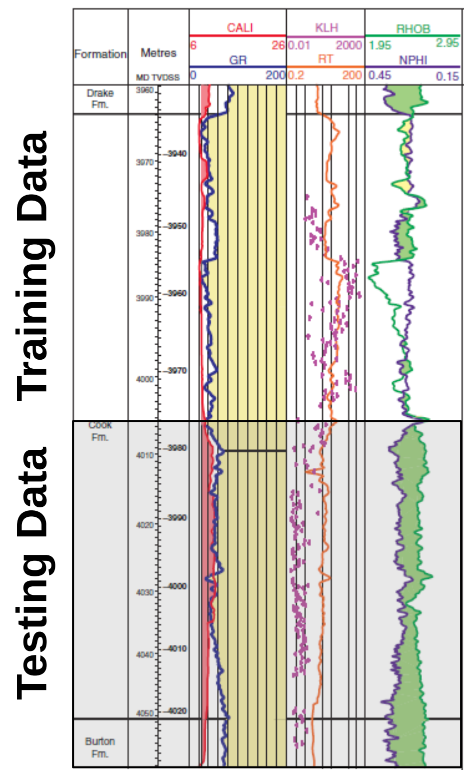
Fair Testing in Spatial / Temporal Settings

Too Easy



Predictions only at ½ ft offsets

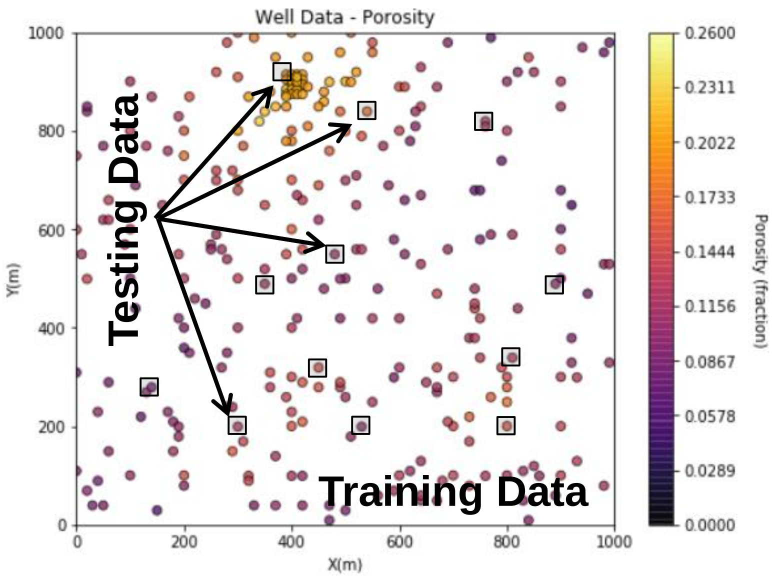
Too Hard



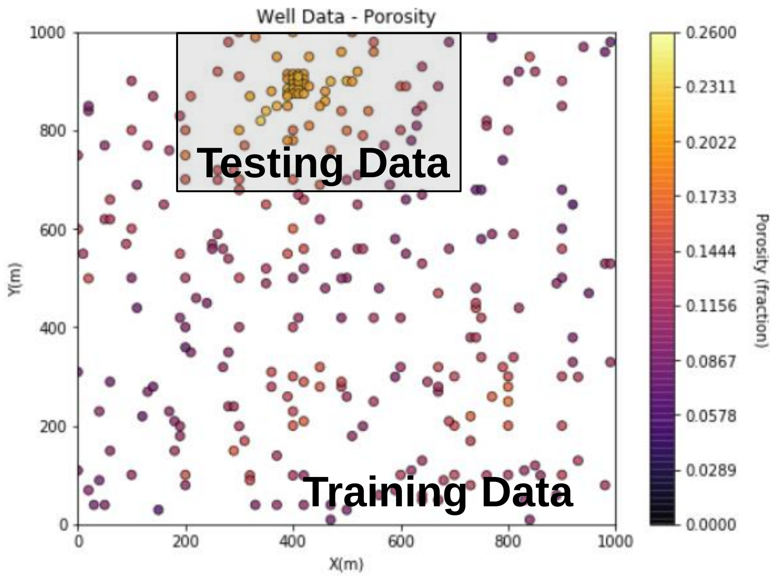
Predictions in a different rock.

Training and Testing

Fair Testing in Spatial / Temporal Settings



Predictions at only short offsets from training data.



Predictions in a distinctly different range of reservoir values.

We will use random sampling and visualize the training and testing data in Euclidean or feature space. More could be done.

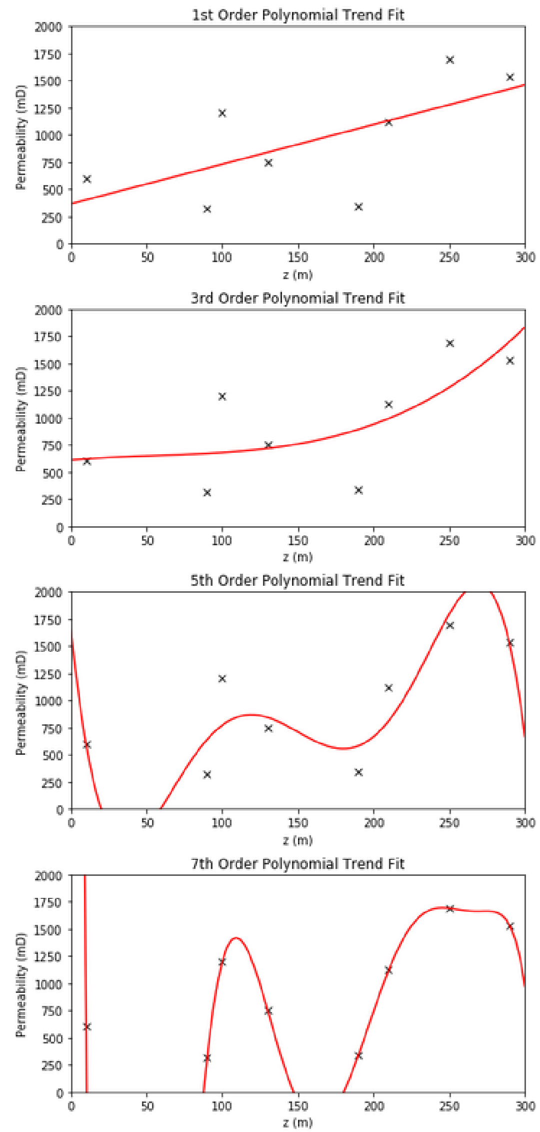
Model Complexity / Flexibility Definition

Model Complexity / Flexibility

A variety of concepts may be used to describe model complexity:

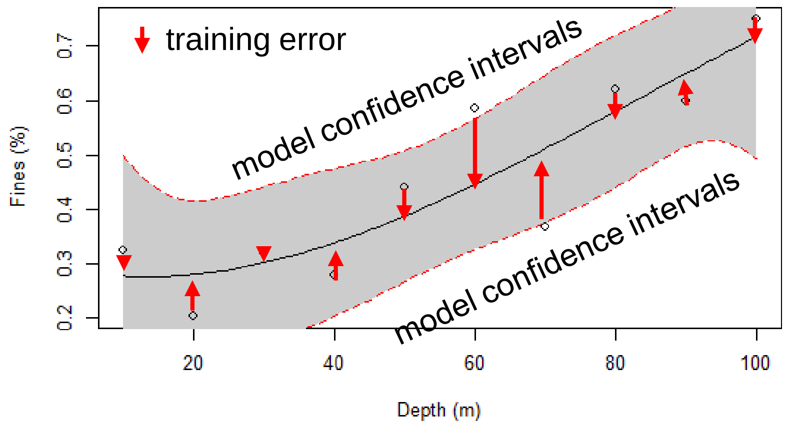
- The number of features:
  - predictor variables are in the model, dimensionality of the model
- The number of terms / parameters
  - the order applied for each term, e.g. linear, quadrative, thresholds
- Expression of the model:
  - Can the model be expressed as:
    - » a compact equation – polynomial regression
    - » nested conditional statements – decision tree

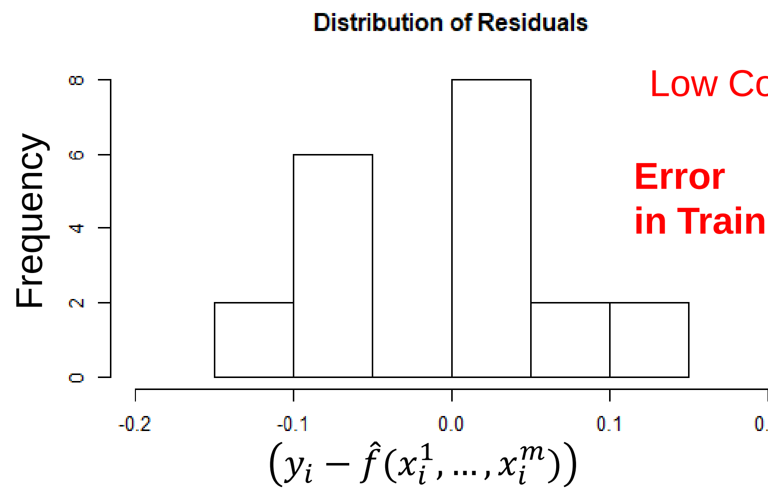
- For example, more complexity with a high order polynomial, larger decision trees etc.



Simple Statistical Demonstration Overfitting

- Example of trend fits:
  - 3rd Ordered Polynomial

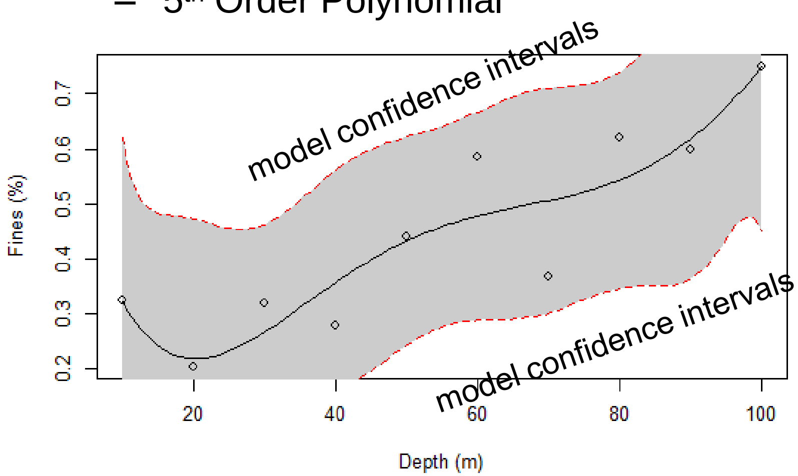


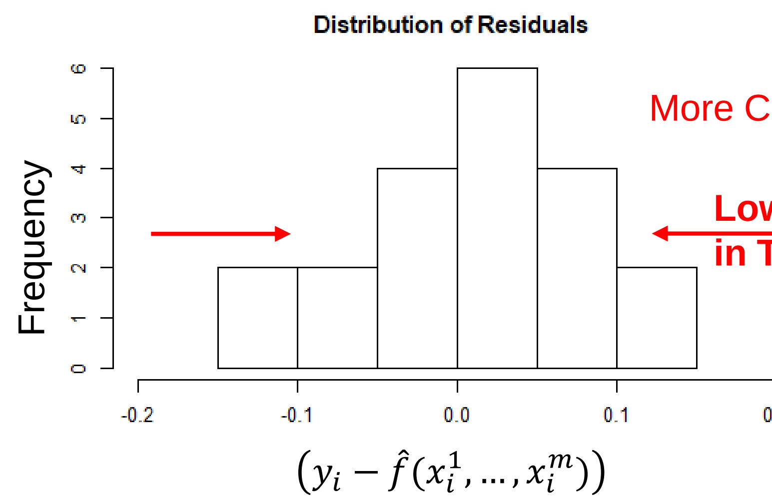


Low Complexity

Error in Training

  - 5th Order Polynomial





More Complexity

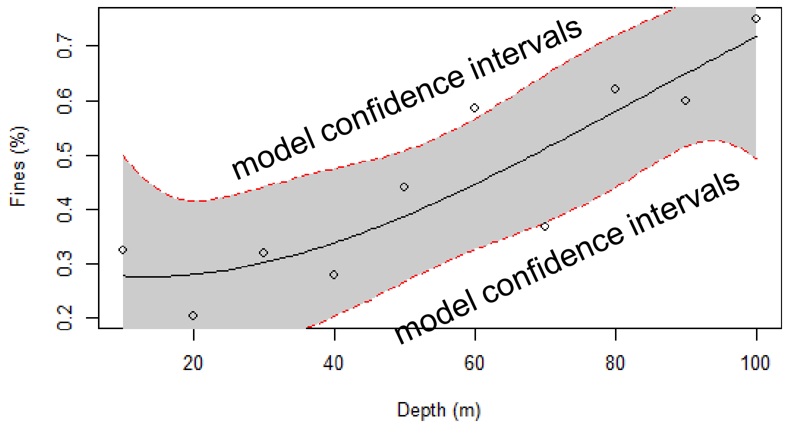
Lower Error in Training

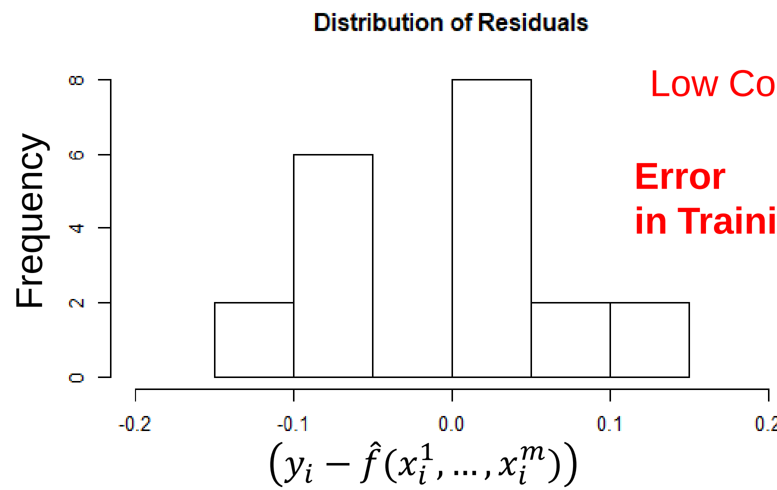
Overfit demonstration in R, code is here:
https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R

R code at Code/Overfit.R

Simple Statistical Demonstration Overfitting

- Example of trend fits:
  - 3rd Ordered Polynomial

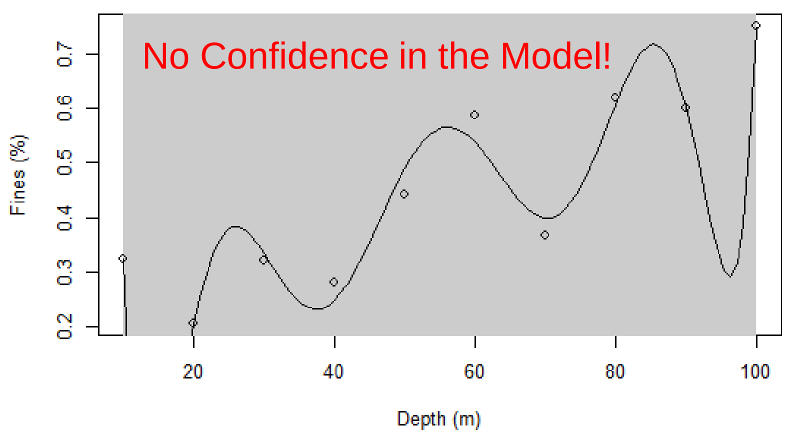


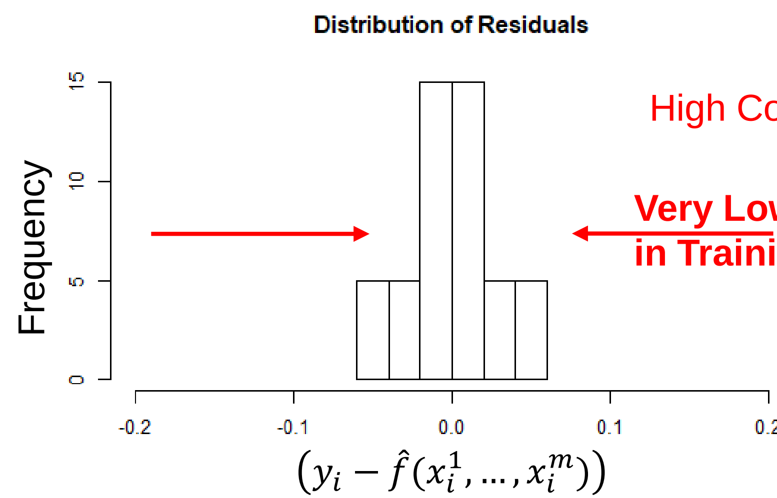


Low Complexity

Error in Training

  - 8th Order Polynomial





High Complexity

Very Low Error in Training

Overfit demonstration in R, code is here:
https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R

R code at Code/Overfit.R

Announcements

Assignment Assistance

1. Don’t send code.
2. If working on paper, scan, don’t take a picture of the document.
3. Only provide a concise explanation and critical figures to answer the questions.
4. You can concisely list your workflow steps with enumeration.
5. If your assignment is more than approx. 2 pages, you’re doing it wrong.
6. Short, concise executive summaries for the associated question.
7. Short answer must be concise and easy to understand.

PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- **Model Fitting, Overfitting and Model Generalization**

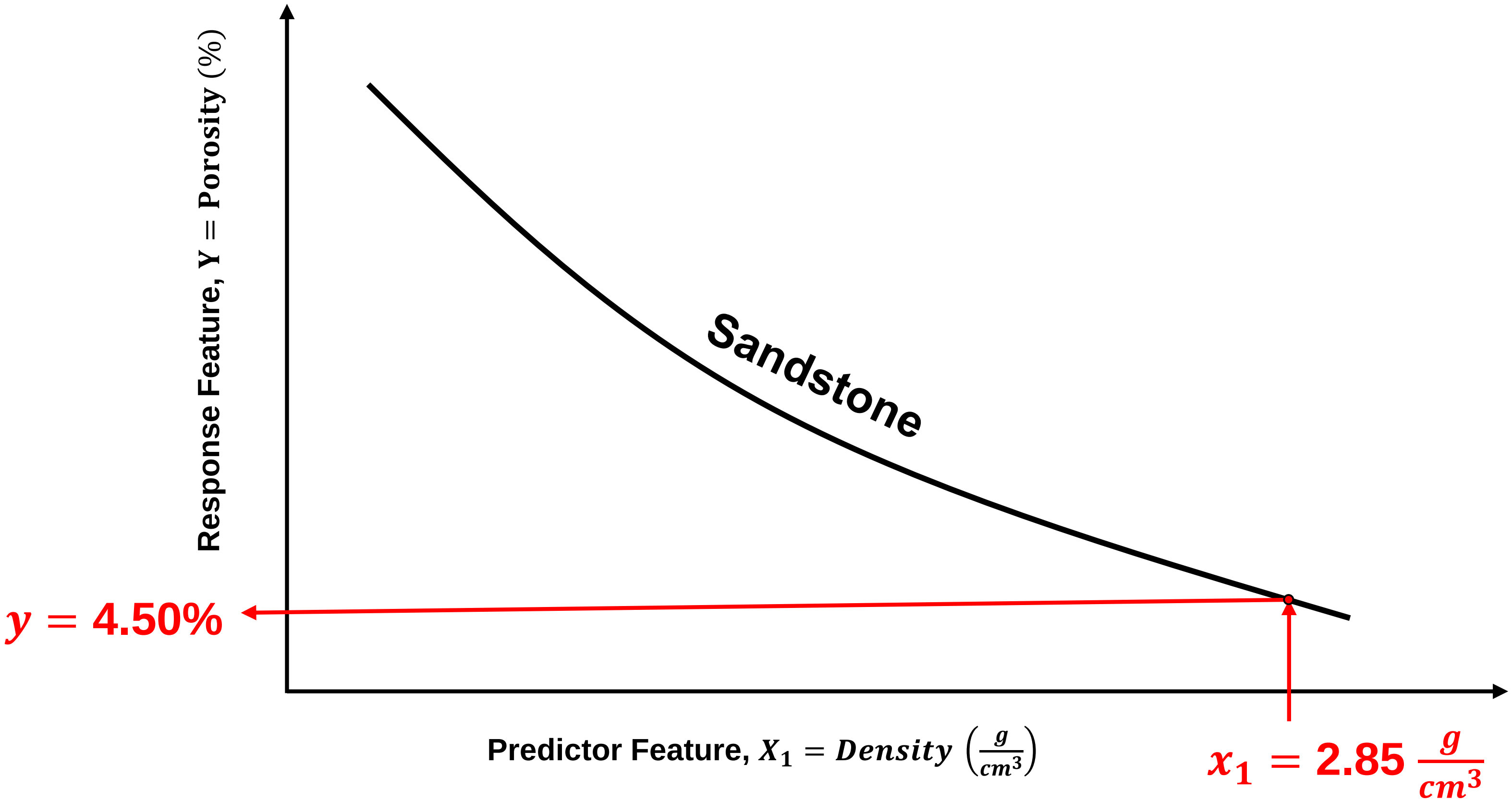
Michael Pyrcz, The University of Texas at Austin

Fitting, Overfitting and Model Generalization Example

Let's take a simple example from petrophysics to explain fitting, overfitting and generalization

- We need to learn this model, we cannot observe/measure rock porosity in a well bore directly.

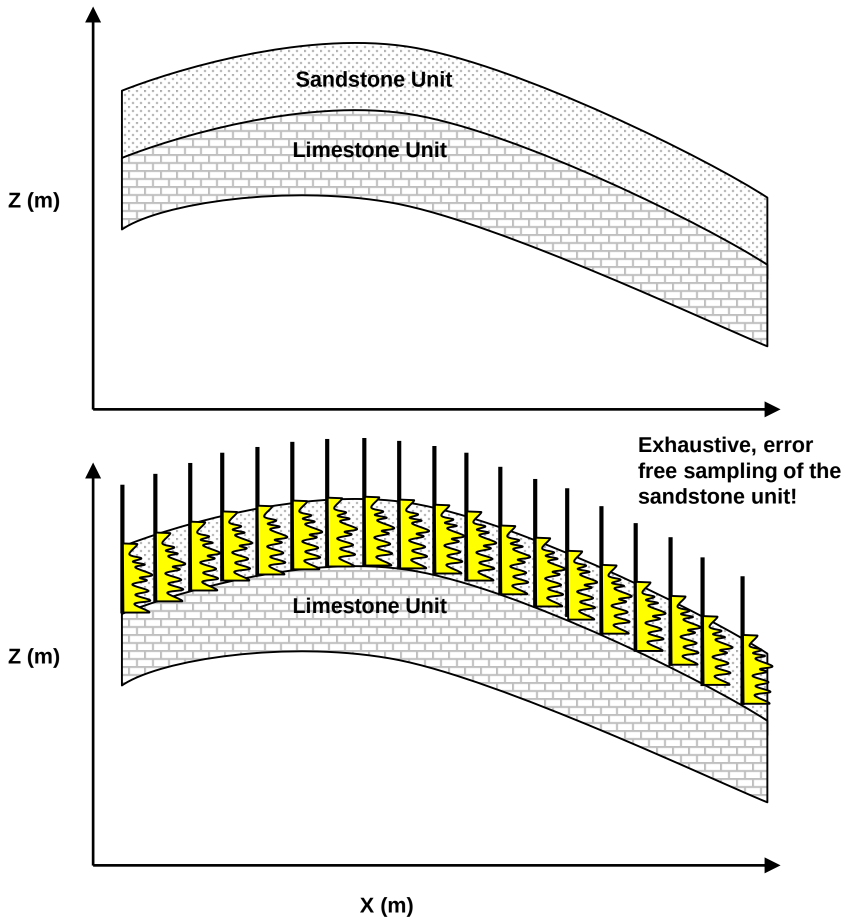
rock porosity from the well log density measure for your sandstone

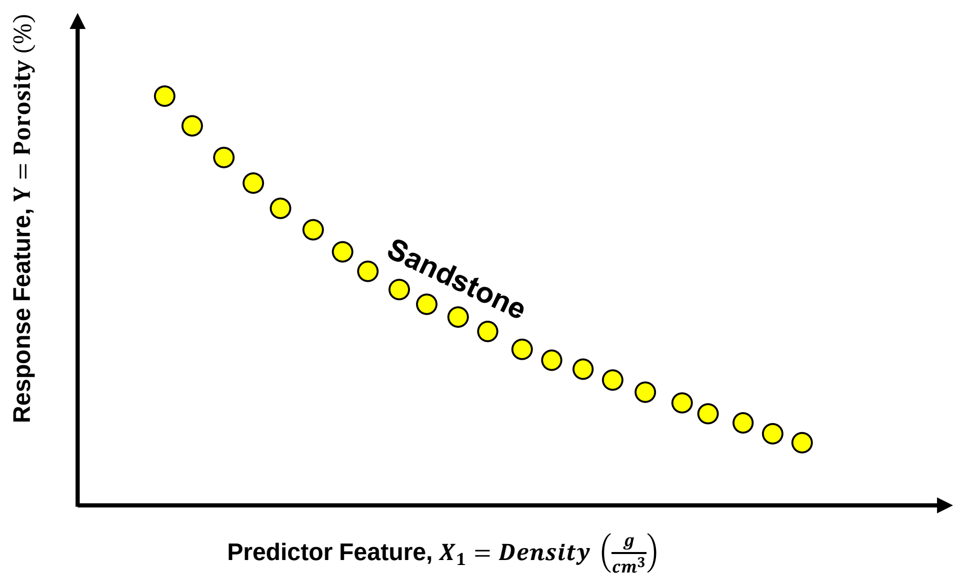


Overfitting and Model Generalization Example

Assume you are omniscient, and you see the entire natural setting/population!

- If we could see the natural setting at the resolution needed to solve our problem and with complete coverage, we would have the population and know this model between our predictor feature, X_1, and response feature, Y, perfectly.



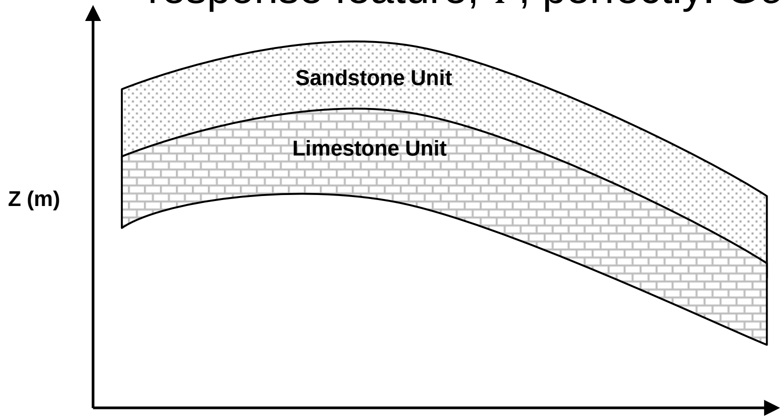


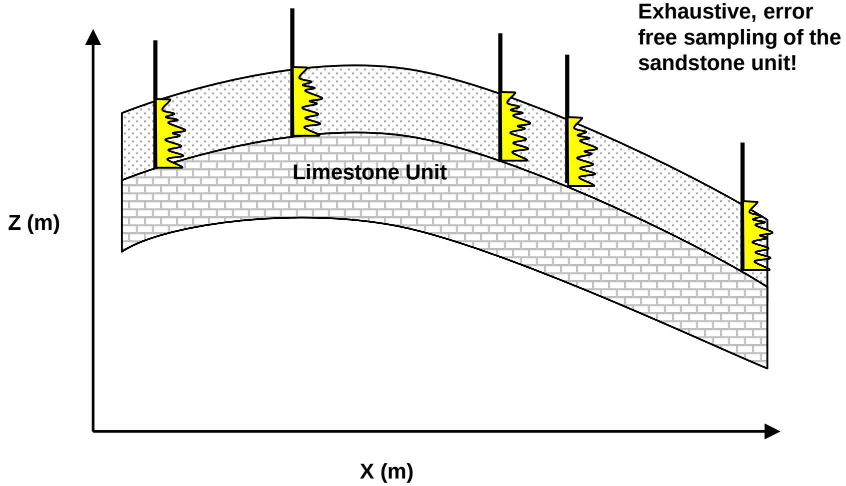
Overfitting and Model Generalization Example

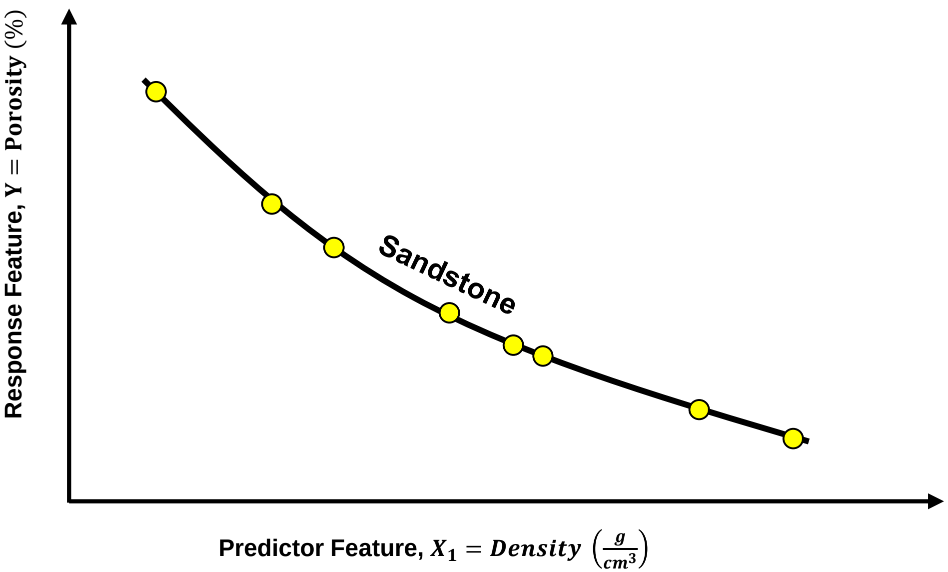
Assume you integrate physics and limited samples from the population.

- We could build a model with physics (domain information), hinged on limited sample coverage.
- A good (best) model for the relationship between the predictor feature, X_1, and response feature, Y, perfectly. *Going forward we will assume data-driven only.*

We aren't addressing integration of geoscience and engineering, the physics of the problem.



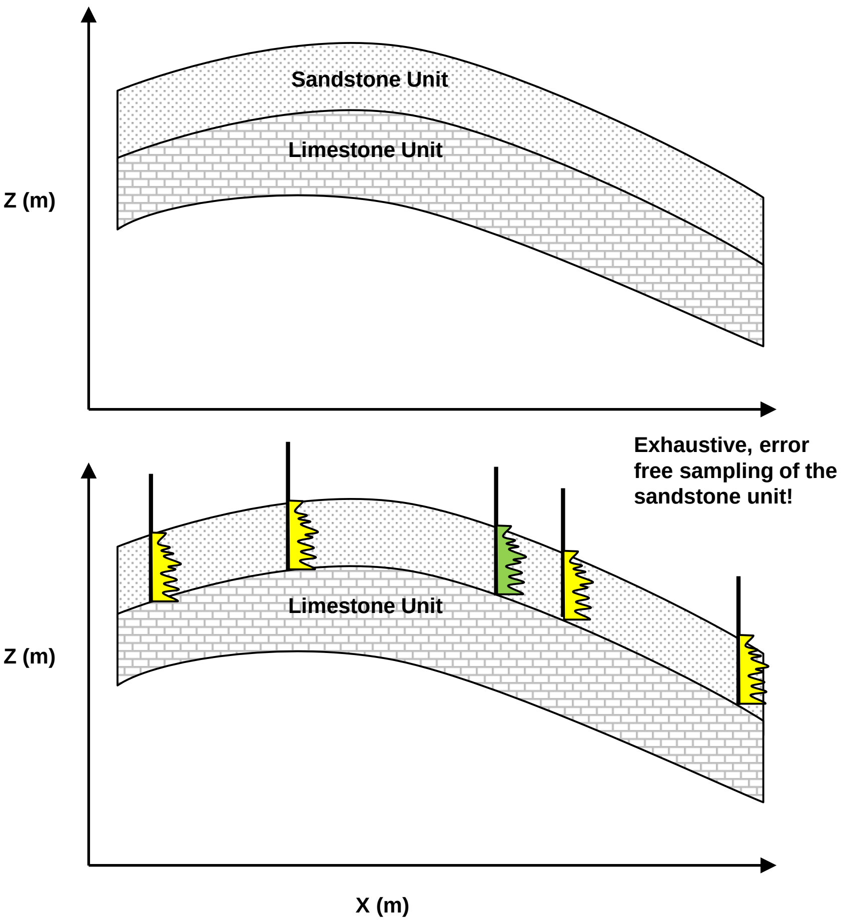


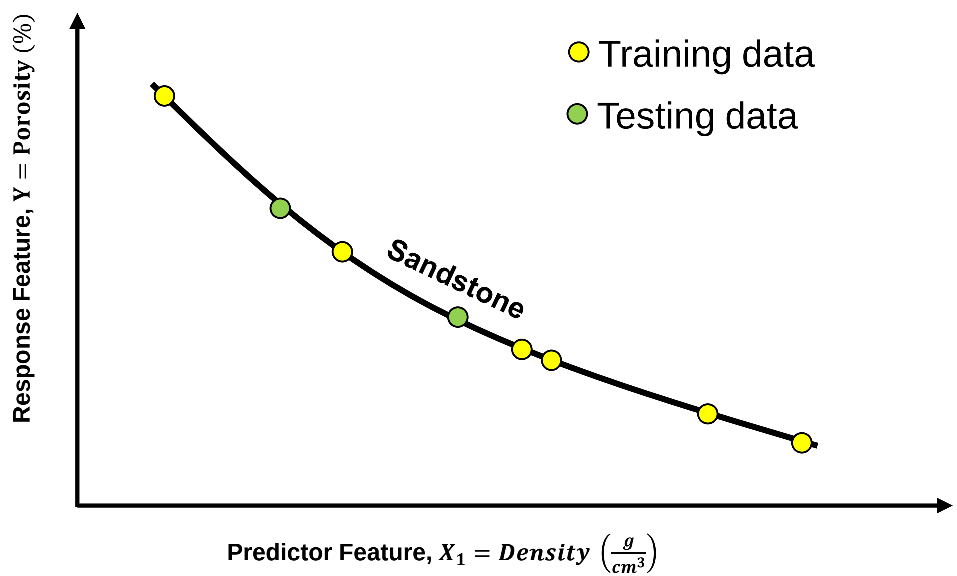


Overfitting and Model Generalization Example

Assume the data-driven approach, training/tuning a model, $Y = f(X_1)$.

- We will separate the data into:
  - Training data to train the model parameters - fit
  - Testing data, withheld from training, to tune the model hyperparameters - complexity

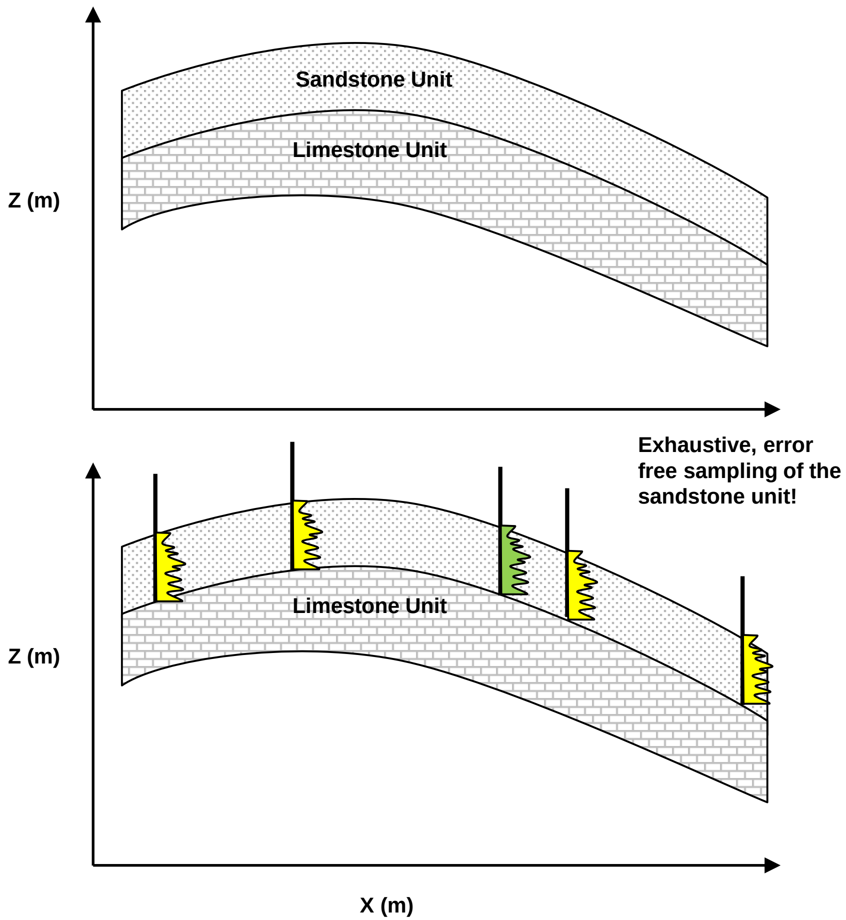


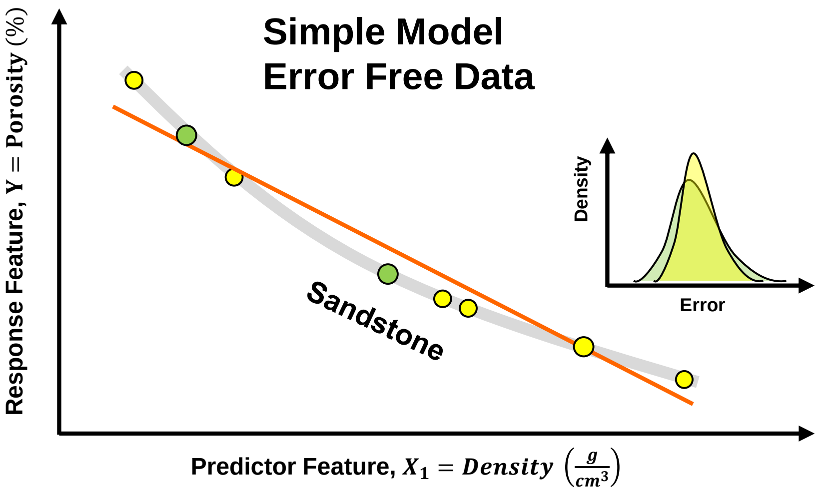


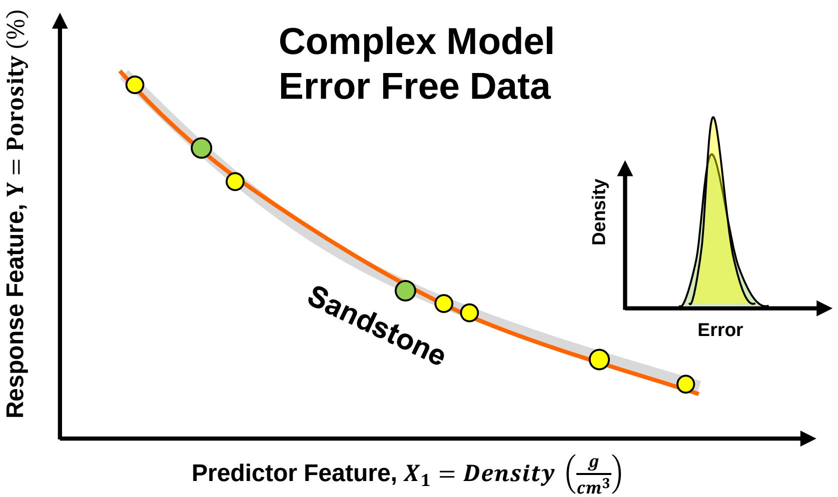
Overfitting and Model Generalization Example

Assume the data-driven approach, training/tuning a model, $Y = f(X_1)$.

- We need to fit an exhaustive model, $\hat{Y} = \hat{f}(X_1)$, $\forall\, x_1 \in [x_{min}, x_{max}]$
- As expected, the more complicated model is a better fit. So far it generalizes ok away from training!







Overfitting and Model Generalization Example

But we don't have error-free measures, we have samples with error

- Error in the measurement of the predictor feature, well log measurement error, ϵ_{X_1}.
- Error in the collocated core-based porosity measure, ϵ_Y.

Simple Models:

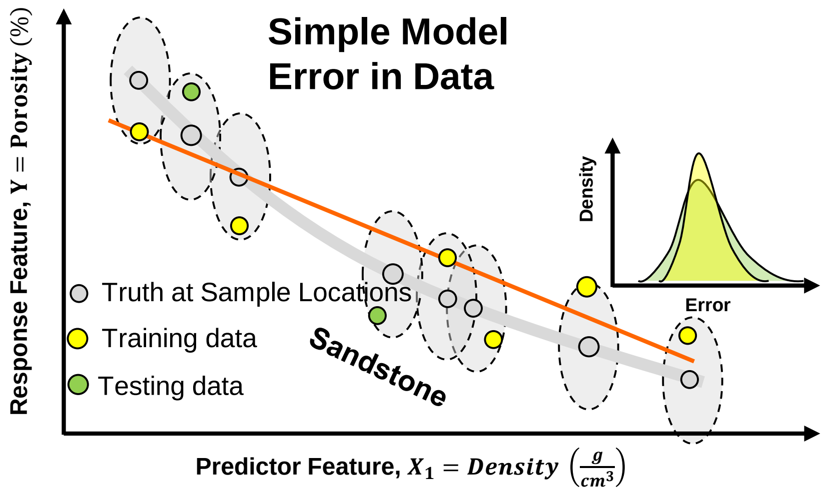
- Less sensitive to error/noise in the data

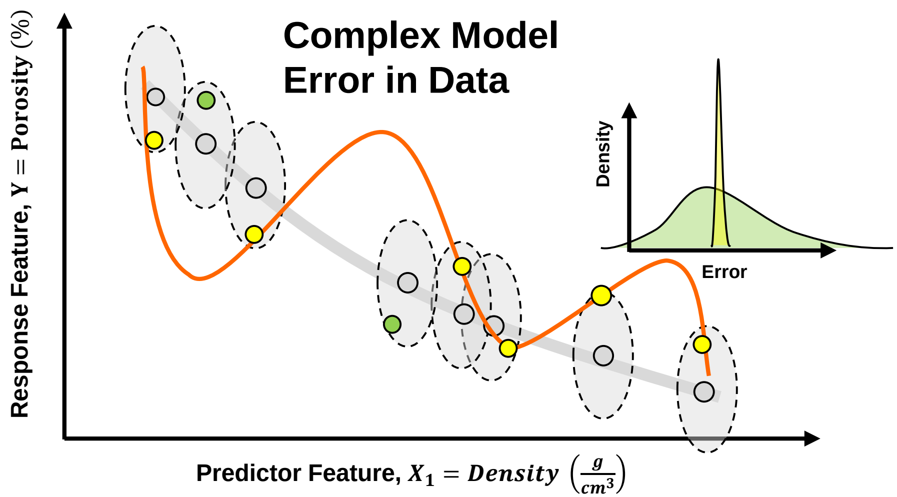
Complexity:

- The ability to flexibly learn the natural system

Complexity + Data Error = Overfit

- Model that fits noise
- Model that poorly generalizes, poor predictions away from training data

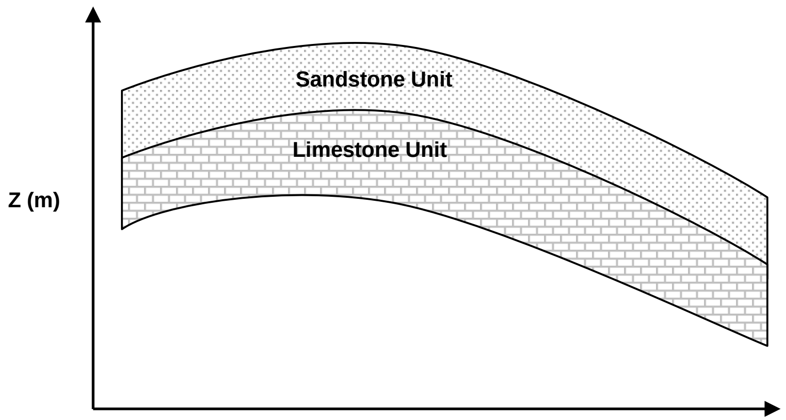


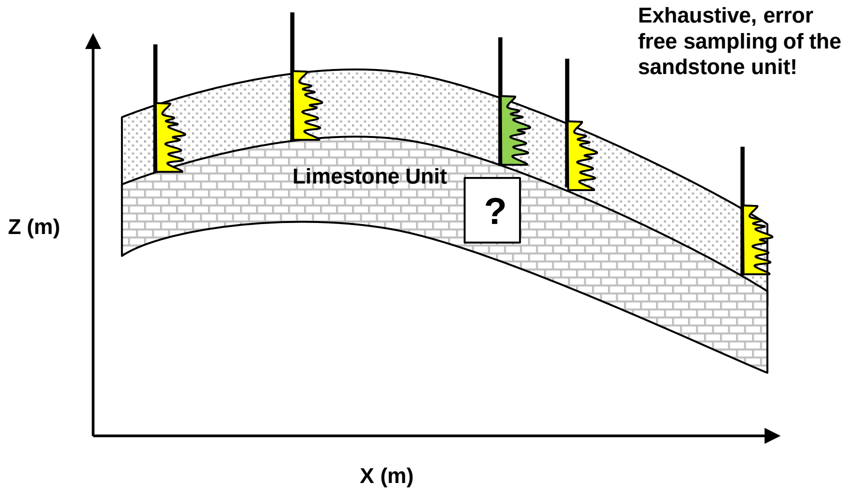


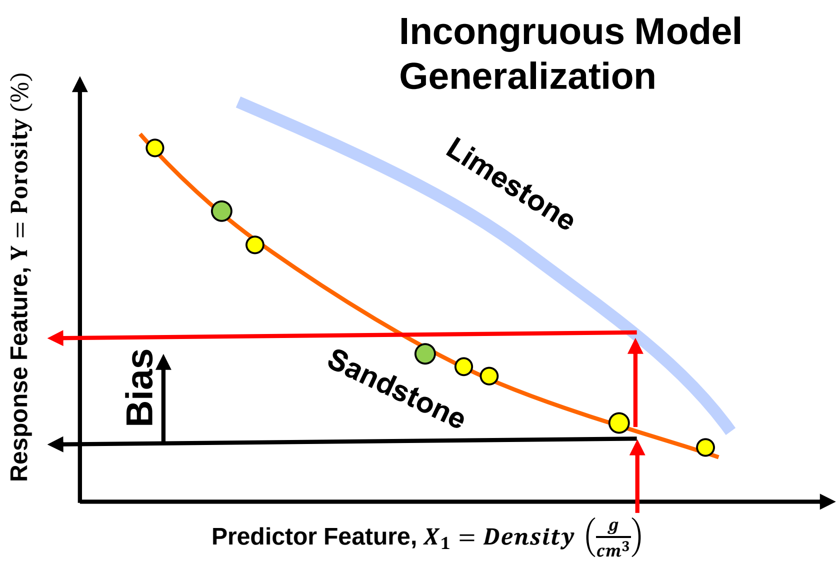
Overfitting and Model Generalization Example

How far can we go with model generalization?

- What if we train and test with sandstone and apply the model to limestone?







There are limits for the congruous application of our machines.

- Training / testing data must be consistent with real-world use.
- As with geostatistics we should be explicit about our decision of stationarity.

Definition of Overfit

Overfit:

- More model complexity/flexibility than can be justified with the available data, data accuracy, frequency and coverage

- Model explains “idiosyncrasies” of the data, capturing data noise/error in the model

- High accuracy in training, but low accuracy in testing / real-world use away from training data cases – **poor ability of the model to generalize**

Model Overfit Hands-on

Overfit Demonstration

- Add some data with no error.
- Observed the models simple to complicated by increasing the polynomial order
- Add some error/noise to the data and repeat

InteractiveMachine Learning Overfitting Interactive Demonstation

Michael Pyrcz, Associate Professo and John Eric McCarthy II, University of Texas at Austin

Change the number of sample data, train/test split and the data noise and observe overfit! Change the model order to observe a specific model example.

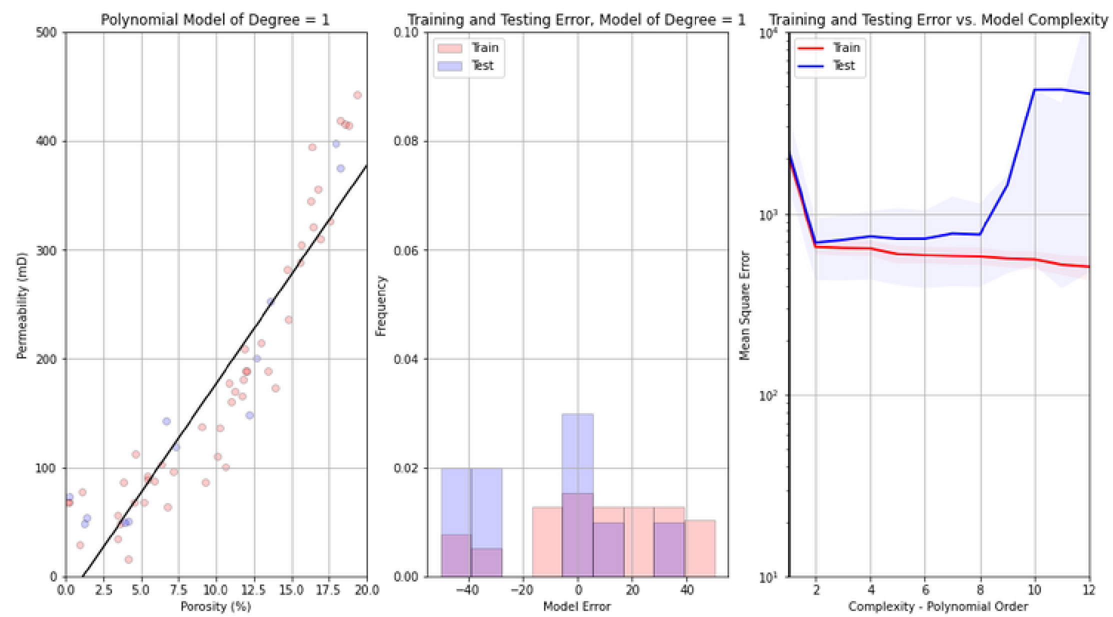
The Inputs

- **n** - number of data
- **Test %** - percentage of sample data withheld as testing data
- **Noise StDev** - standard deviation of random Gaussian error added to the data
- **Model Order** - the order of the

```
display(ui2, interactive_plot)                    # display the interactive plot
```

Machine Learning Overfit/Generalization Demo, Prof. Michael Pyrcz and John Eric McCarthy II, The University of Texas at Austin

n 60 Test % 0.20 Noise StDev 28.00 Model Order 1



Demonstration of overfit with Interactive_Overfit.ipynb.

Building Our Machine, One More Time

Now that we have all the concepts, let's walk through the workflow again.

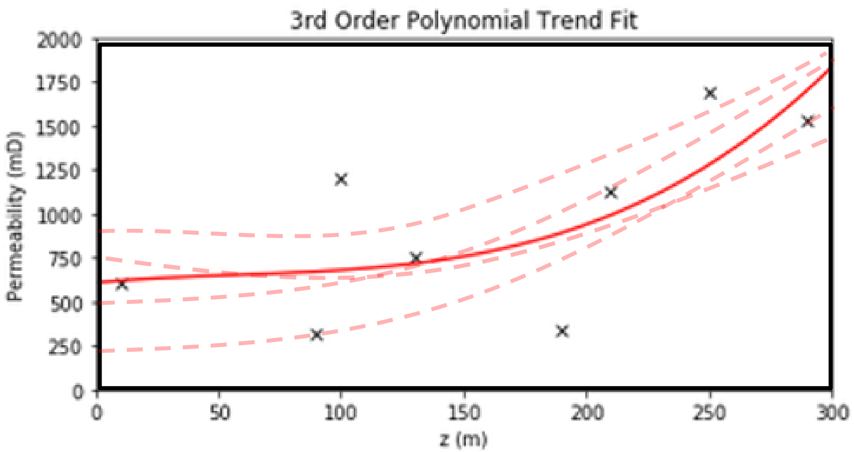
Apply Training Data to Train the Model Parameters.

- Repeat for all levels of complexity as specified by a range of hyperparameters.
- For example, the parameters of this 3rd order polynomial model.

b_3, b_2, b_1 and c

$$k = b_3 z^3 + b_2 z^2 + b_1 z + c$$

- But not appropriate to determine level of complexity (hyperparameter)



Train model parameters for each level of complexity to maximize fit with training data.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left[\left(y_i - \hat{f}(x_1^j, \dots, x_m^j)\right)^2\right], for\ i = 1, \dots, n_{train}$$

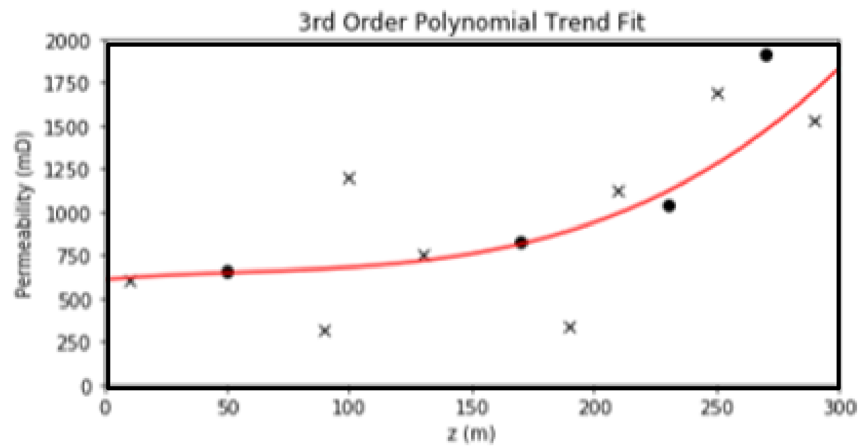
Minimize the summary measure of error over the training data.

Hyperparameter of our model: 1st, 2nd, 3rd 4th … order polynomial.

Building Our Machine, One More Time

Now that we have all the concepts, let's walk through the workflow again.

Apply Withheld Data to test our Machine.



Test model for each level of complexity against testing data, select the best performing hyperparameters in the testing.

- Calculate the error, over the withheld from training, testing data for all levels of complexity as specified by a range of hyperparameters.
- **Select the hyperparameters** that minimize error over the withheld testing data.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left[\left(y_i - \hat{f}(x_1^j, \dots, x_m^j)\right)^2\right], for\ i = 1, \dots, n_{test}$$

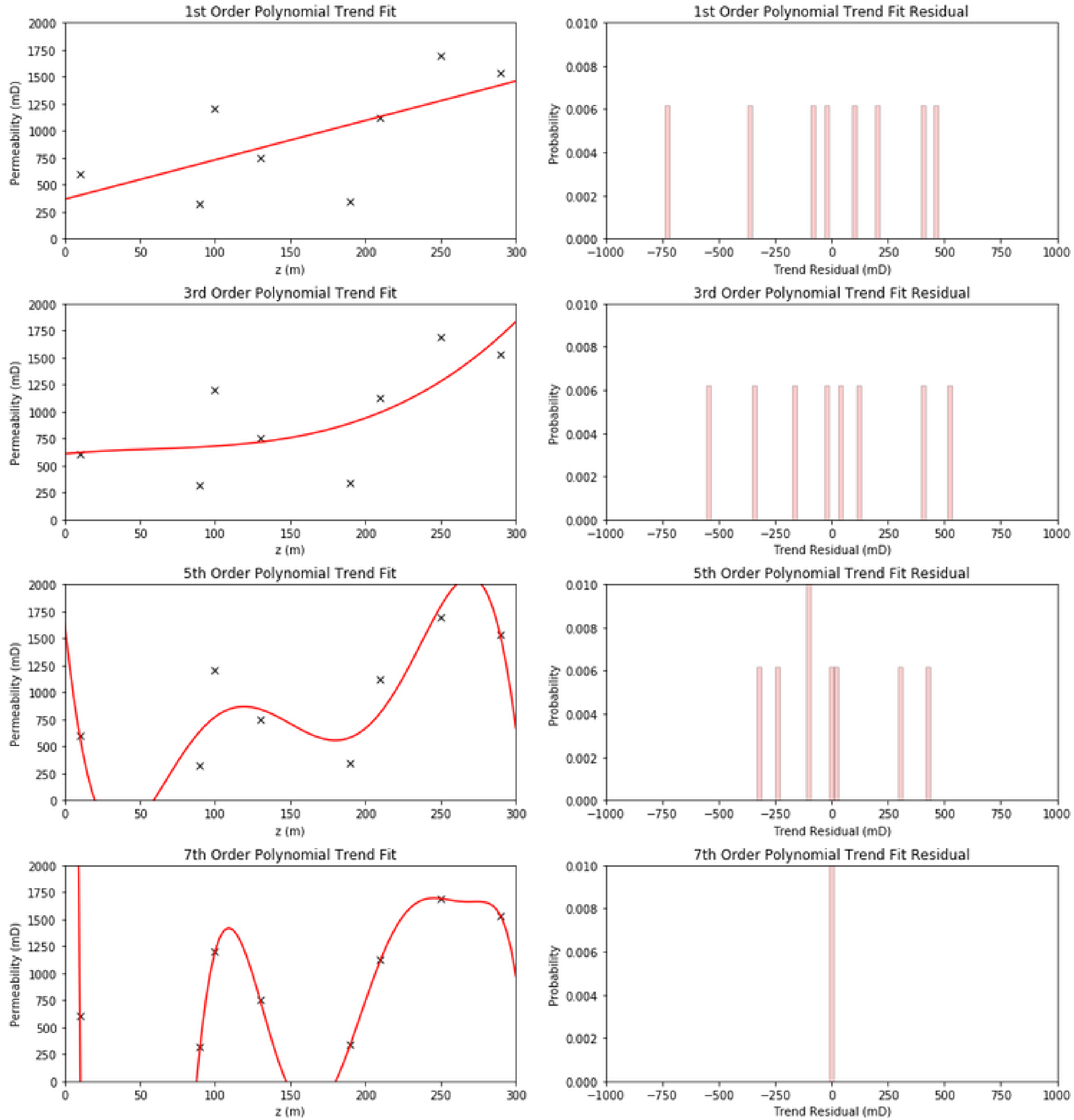
Minimize the summary measure of error over the testing data.

- **Retrain the model** with the **tuned hyperparameters** with **all the data**, training and testing data and apply it in real-world.

Now you have a good model that seems to perform well with data not used to build it.

Building Our Machine, One More Time

What would happened if we just maximized fit to the data?

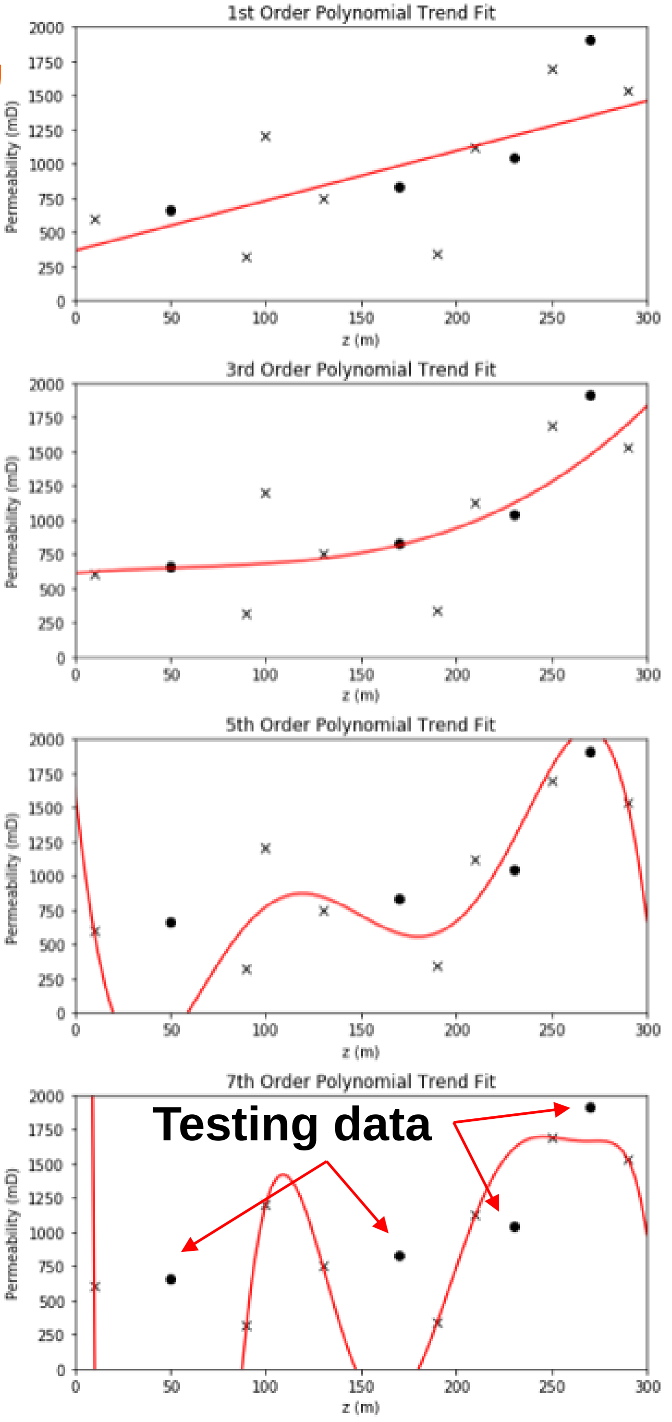


- Very complicated model would be best.
- Perfectly [over]fit the data.
- Tuning protects us from overfit, by simulating real-world use of the model.

Building Our Machine, One More Time

The More Complicated Model Would be Overfit

1. Have high accuracy at training data
2. Poor testing accuracy with new observations!
3. Very dangerous with extrapolation.
4. Low model bias, but **high model variance**.



Now We Begin Machine Learning

- With these concepts established, let's start to get into machine learning / statistical learning methods
  - These methods will allow you to perform inference and prediction
  - Work with complicated data sets / big data analytics
  - Detect patterns in data
- Remember in our business to win:
  - Have the best data
  - Use the data best
- We are at the beginning of the 4^{th} paradigm for scientific discovery
  - Data-driven discovery
- Smart fields, 4D seismic surveys, increased computational resources
  - Expanding opportunities for machine learning
- We'll start inferential:
  - Clustering, Principal Component Analysis, Multidimensional Scaling

PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- Examples of Machine Learning

Michael Pyrcz, The University of Texas at Austin

Examples of Machine Learning

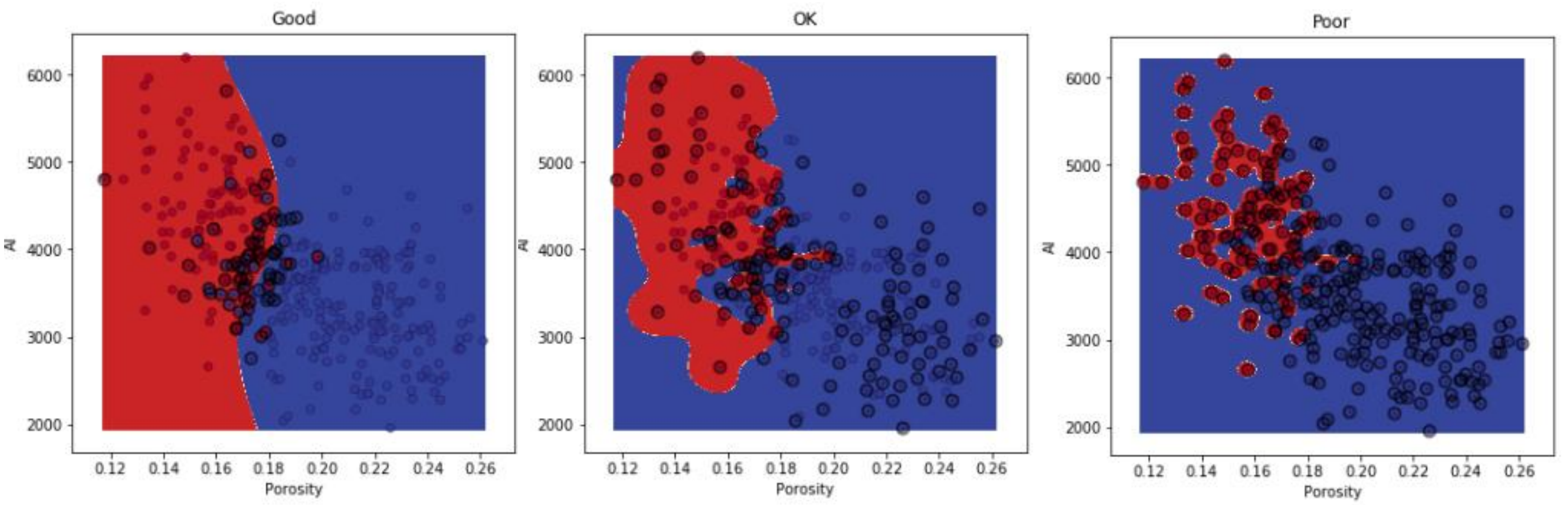
Provides a set of examples with machine learning to address subsurface challenges.

- We will cover a wide range of machine learning methods in this class.

- This is to motivate and inspire.

Support Vector Machines

Support vector machines for interpolating, extrapolating facies from data.

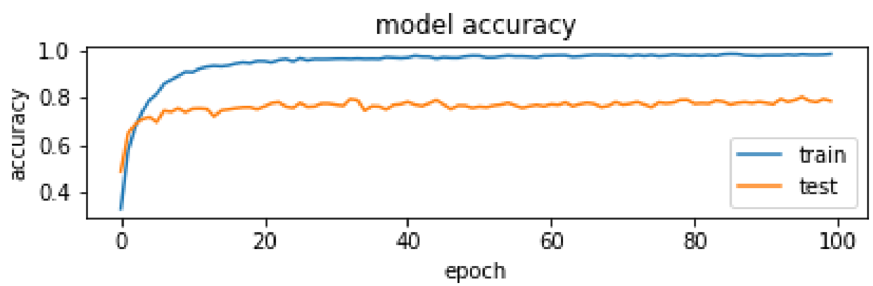


- A range of spatial models with a linear model after projection to a high dimensional space.

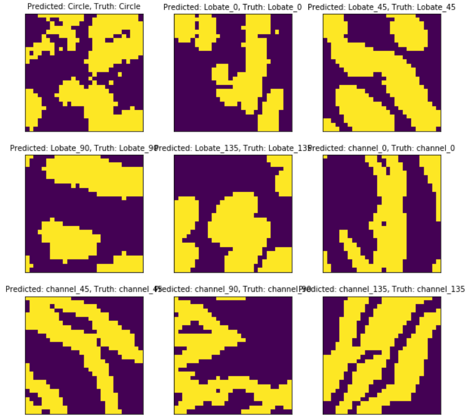
Workflow developed by Wendi Liu, PhD student at The University of Texas at Austin.

Deep Learning

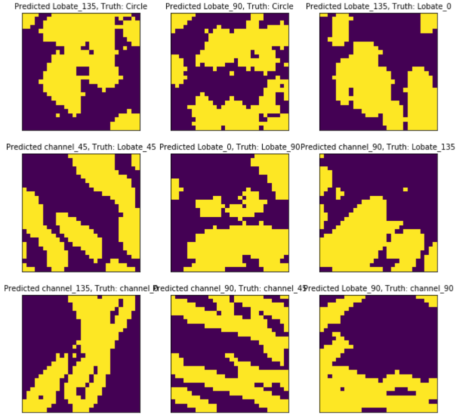
- For model checking and image detection.



Correct Identification



Incorrect Identification

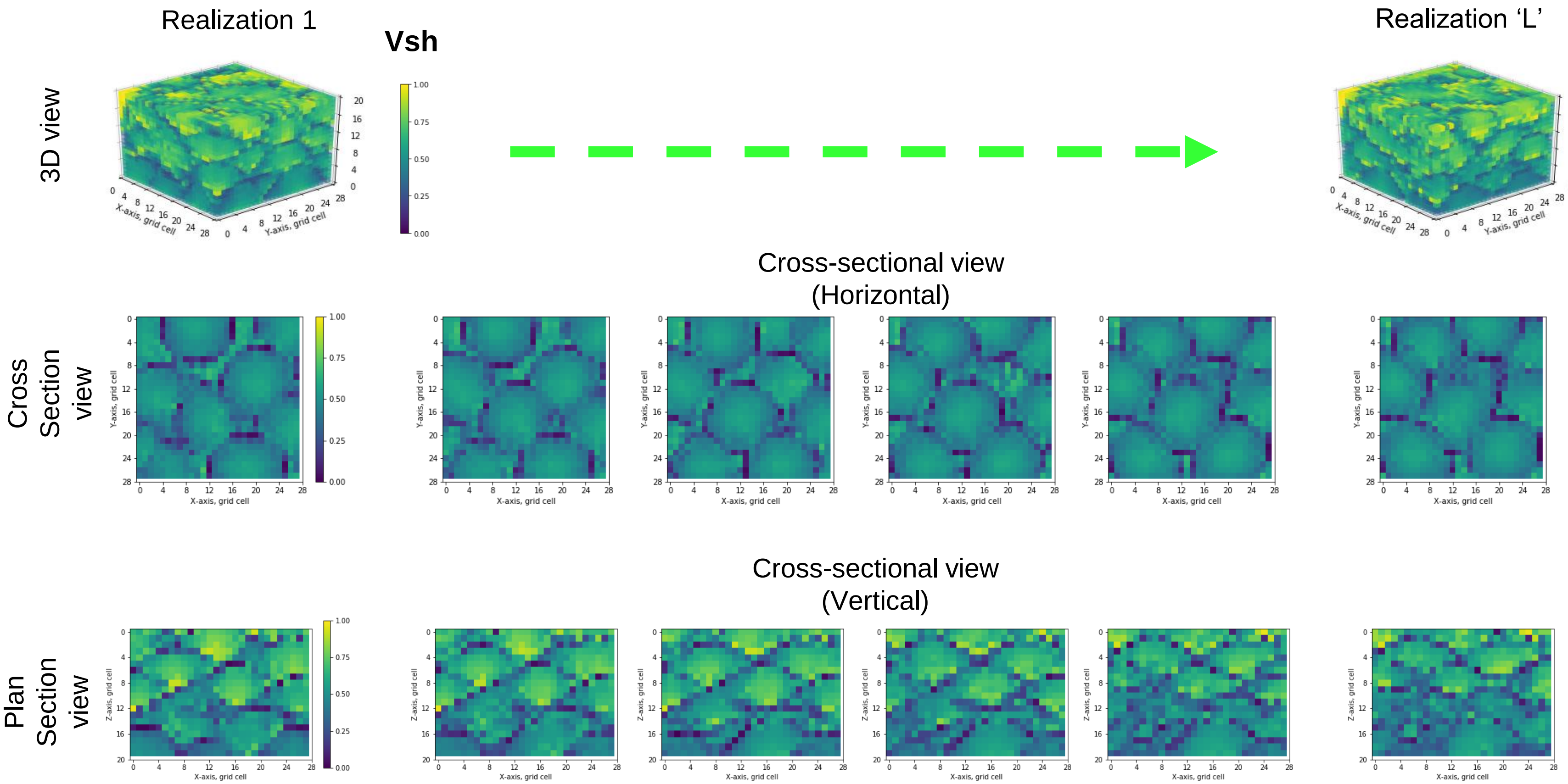


Workflow developed by Honggeun Jo and Javier Santos, PhD student at The University of Texas at Austin.

Convolutional Neural Nets

Can explore the space of uncertainty along a continuous manifold.

- A latent reservoir manifold based on a single paramter



Workflow developed by Honggeun Jo and Javier Santos, PhD student at The University of Texas at Austin.

Convolutional Neural Nets

Filling In Missing Spatial Information

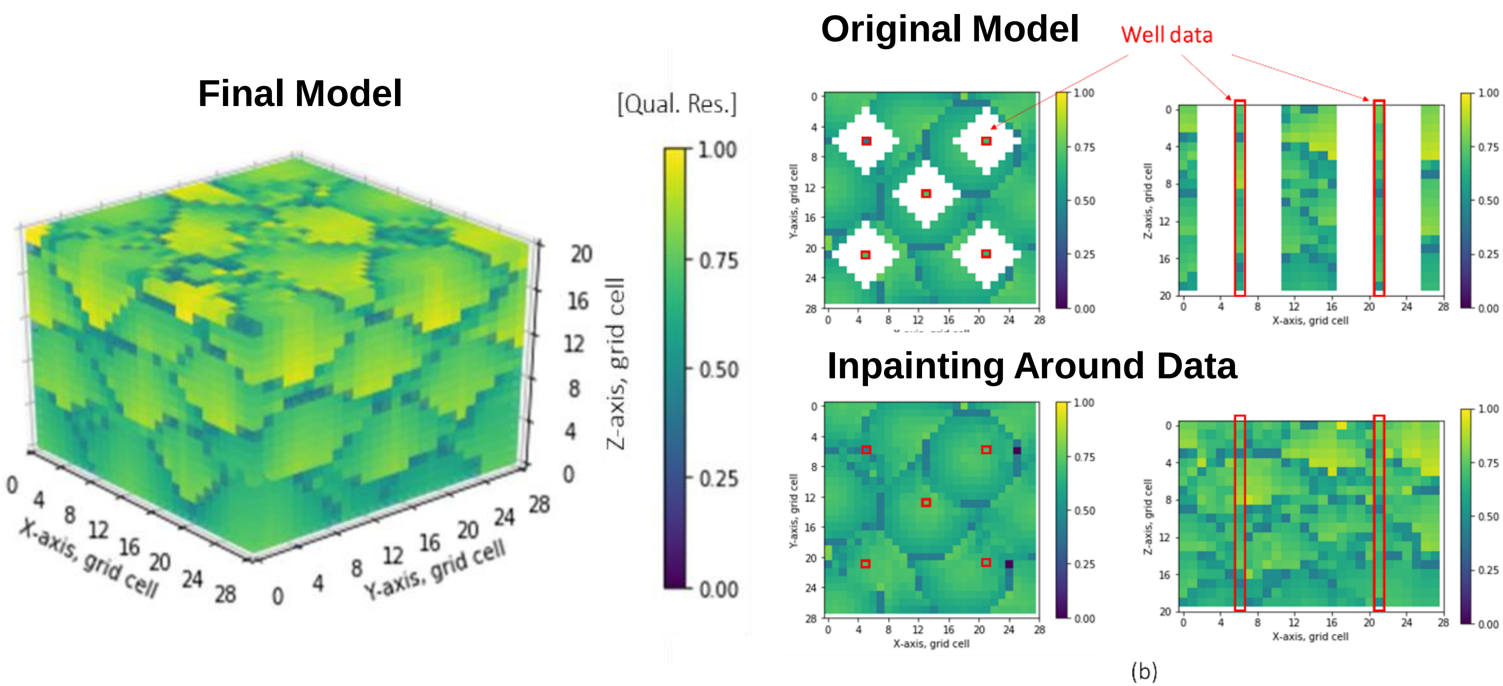
- Semantic inpainting algorithm (Yeh et al., 2015).
- Using conceptual and perceptual information

Examples of semantic image inpainting with DCGAN (Yeh et al., 2016)

Convolutional Neural Nets

Conditioning to Well Data?

- Remove model around data
- Use conceptual (model around mask) and perceptual (model elsewhere to fill in missing model consistent with data



Workflow developed by Honggeun Jo, PhD student at The University of Texas at Austin.

PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- Energy Machine Learning

Michael Pyrcz, The University of Texas at Austin

The 4th Paradigm

Welcome to the 4th Paradigm of Scientific Discover!

Michael Pyrcz, Associate Professor, The University of Texas at Austin

1st Paradigm Empirical Science	2nd Paradigm Theoretical Science	3rd Paradigm Computational Science Simulation	4th Paradigm Data-driven Science
Experiments • 430 BC Empedocles proved air has substance • 230 BC Eratosthenes measure Earth's diameter	**Models/Laws** • 1011 AD al-Haytham Book of Optics • 1687 AD Newton Principia • 1922 Friedmann Cosmic Expansion	**Numerical Simulation** • 1942 Manhattan Project • 1980 – Global Forecast System (GFS) • 1989 Tetzlaff and Harbaugh SEDSIM	**Learning from Data** • 2009 Hey et al. Data-Intensive Book • 2015 AlphaGo beats a professional Go player

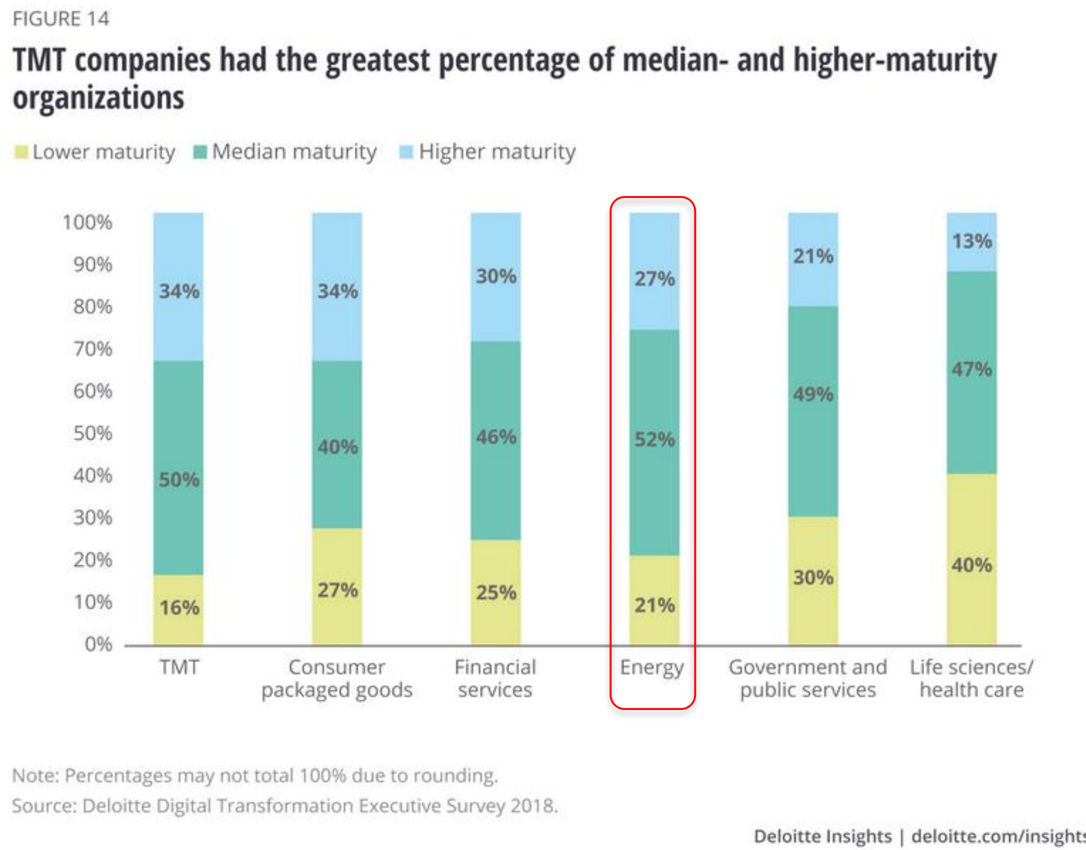
→ **<400 BCE** → **1600s** → **1940s** → **2010s** →

Modified from https://www.nomad-coe.eu/news/147/39/NOMAD-establishes-new-fourth-paradigm-in-computational-materials-science

Energy Digitization

We Are Not Alone

- Digital transformations are underway in all sectors of our economy
- Every energy company that I visit is working on this right now



Digital transformation study by Deloitte, 2019.

Energy Digitization

My Biases:

- Opportunities to do more with our data
- Opportunities to teach data analytics and statistical / machine learning methods to engineers and geoscientists for improved capability
- Geoscience and engineering knowledge & expertise remains core to our business

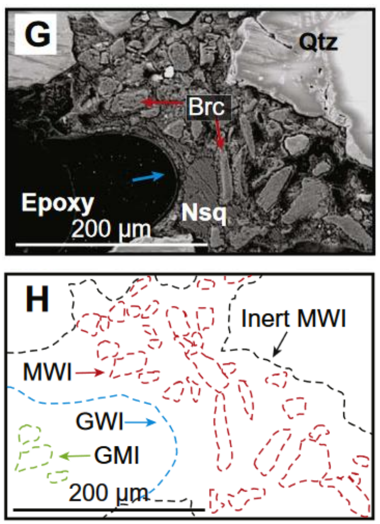
Digital transformation PricewaterhouseCoopers (PwC) panel April 9th, 2019.

Working in the 4th Paradigm

We integrate all paradigms, new tools to add value:

- We augment with new scientific paradigms
- We don't replace older paradigms!

1st Paradigm
Empirical Science



Microfluidics experiment brucite carbonation experiment (Harrison et al., 2017).

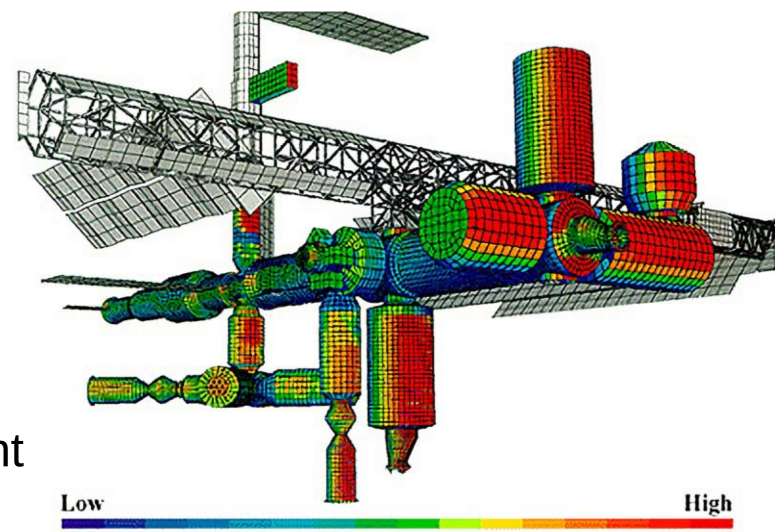
2nd Paradigm
Theoretical Science

$$\boldsymbol{q} = -\frac{\boldsymbol{k}}{\boldsymbol{\mu}}\nabla \boldsymbol{p}$$

q – flux
k – permeability
μ – dynamic viscosity
∇p – pressure gradient

Darcy's law.

3rd Paradigm
Computational Science
Simulation



International space station impact risk from computer simulation. Image from https://en.wikipedia.org/wiki/Risk_management.

Data

Data-driven Science Needs Data, Data Preparation Remains Essential

>80% of any subsurface study is data preparation and interpretation

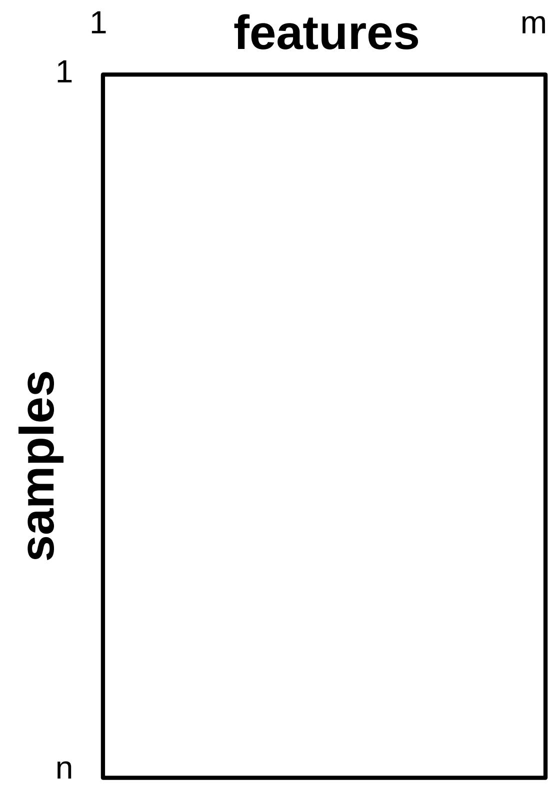
We continue to face a challenge with data:

1. Data curation
2. Large volume
3. Large volumes of metadata
4. Variety of data, scale, collection, interpretation
5. Transmission, controls and security

Clean databases are prerequisite to all data analytics and machine learning

Must start with this foundation

Garbage in, garbage out

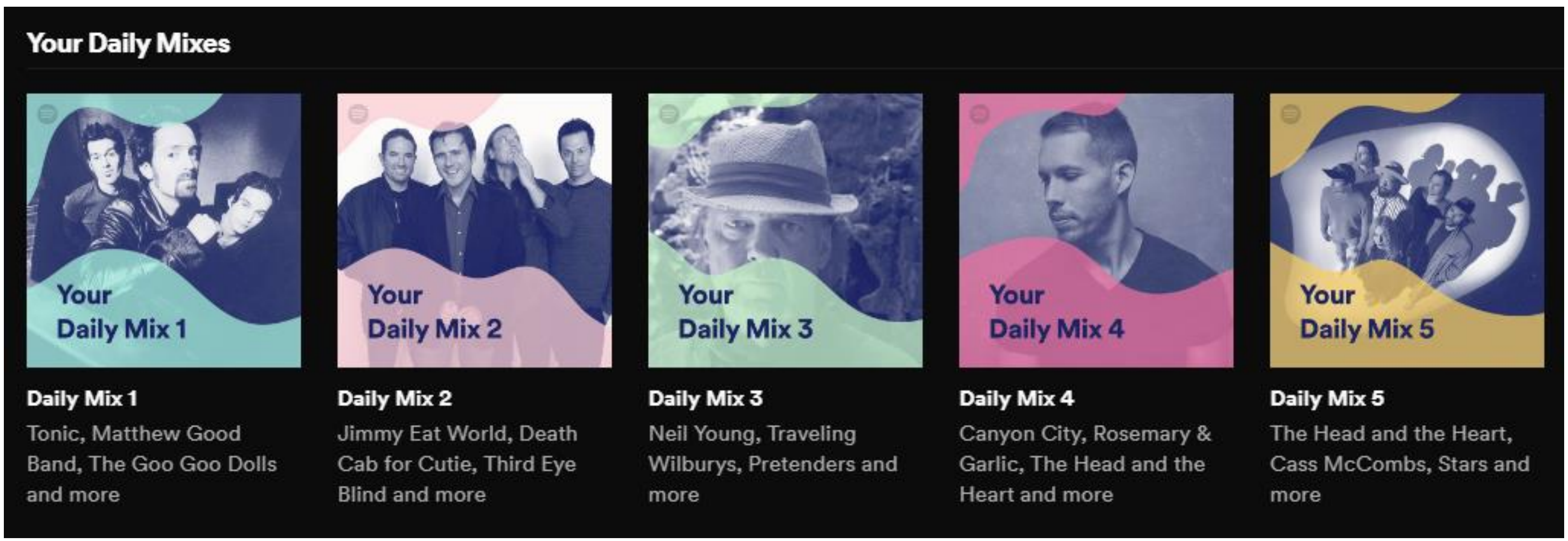


The common data table, samples and features.

Energy is Unique

Energy is Different and May Need New Solutions:

- Sparse, uncertain data, complicated and heterogeneous, open earth systems
- high degree of necessary geoscience and engineering interpretation and physics
- expensive, high value decisions that must be supported



Spotify recommender system from my account summer, 2019.

- Be a critical user / consumer / developer of this technology

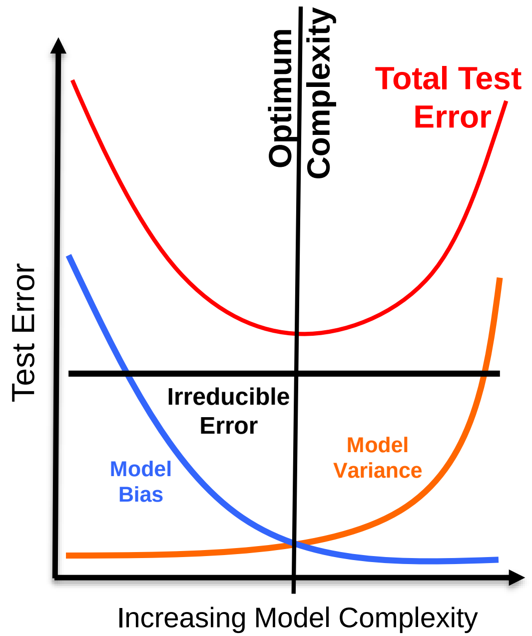
Don’t Jump to Complexity

The Expected Test Mean Square Error may be calculated as:

$$E\left[\left(y_0-\hat{f}(x_1^0,\dots,x_m^0)\right)^2\right]=\underbrace{\left(E[\hat{f}(x_1^0,\dots,x_m^0)]-f(x_1^0,\dots,x_m^0)\right)^2}_{\text{Model Bias}^2}+\underbrace{E\left[\left(\hat{f}(x_1^0,\dots,x_m^0)-E[\hat{f}(x_1^0,\dots,x_m^0)]\right)^2\right]}_{\text{Model Variance}}+\underbrace{\sigma_e^2}_{\text{Irreducible Error}}$$

Remember:

- **Model Variance**, **Model Bias and Irreducible Error**
- Often simpler model outperform more complicated models, controlling model variance is critical!
- While providing a more interpretable model to support high value decisions



Model variance and bias trade-off.

Interpretability is Critical

Develop Methods and Workflows that Provide Useful Diagnostics

- Interpretability may be low
- Application may become routine and trusted
- The machine is trusted, becomes an 'unquestioned authority'

Machine learning-based logistic classifier to identify wolf or dog.

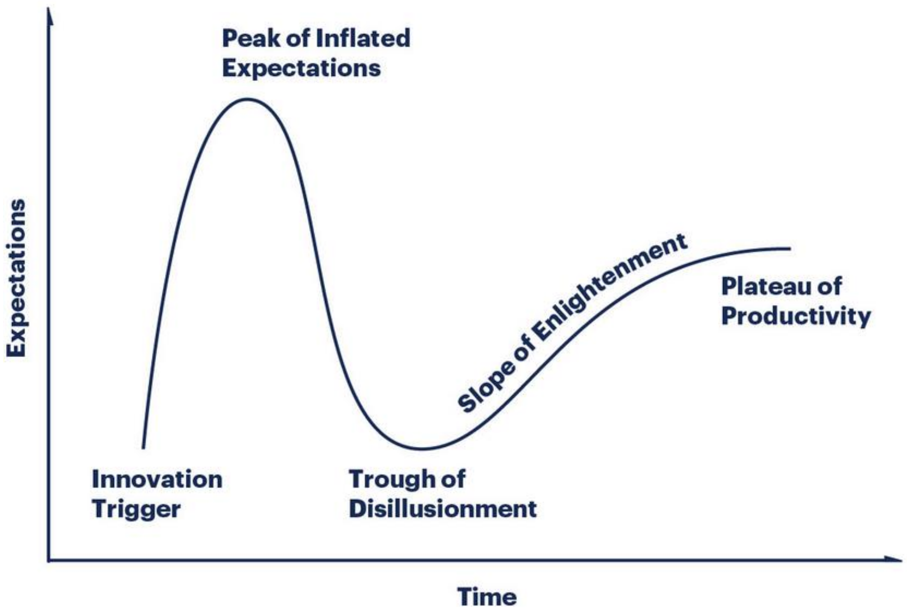
'Even the developers that work on this stuff have no idea what it is doing' 'These systems do not fail gracefully!' – Peter Haas TED Talk.

Image and example from Ribeiro et al. (2016) https://arxiv.org/pdf/1602.04938.pdf

Meeting Technical Expectations

The Technology Hype Cycle (from Gartner)

- Where are we currently for data analytics and machine learning?
- Varies by company and by group within company.
- Globally, expectations are high!



Technology hype cycle from time of discovery.

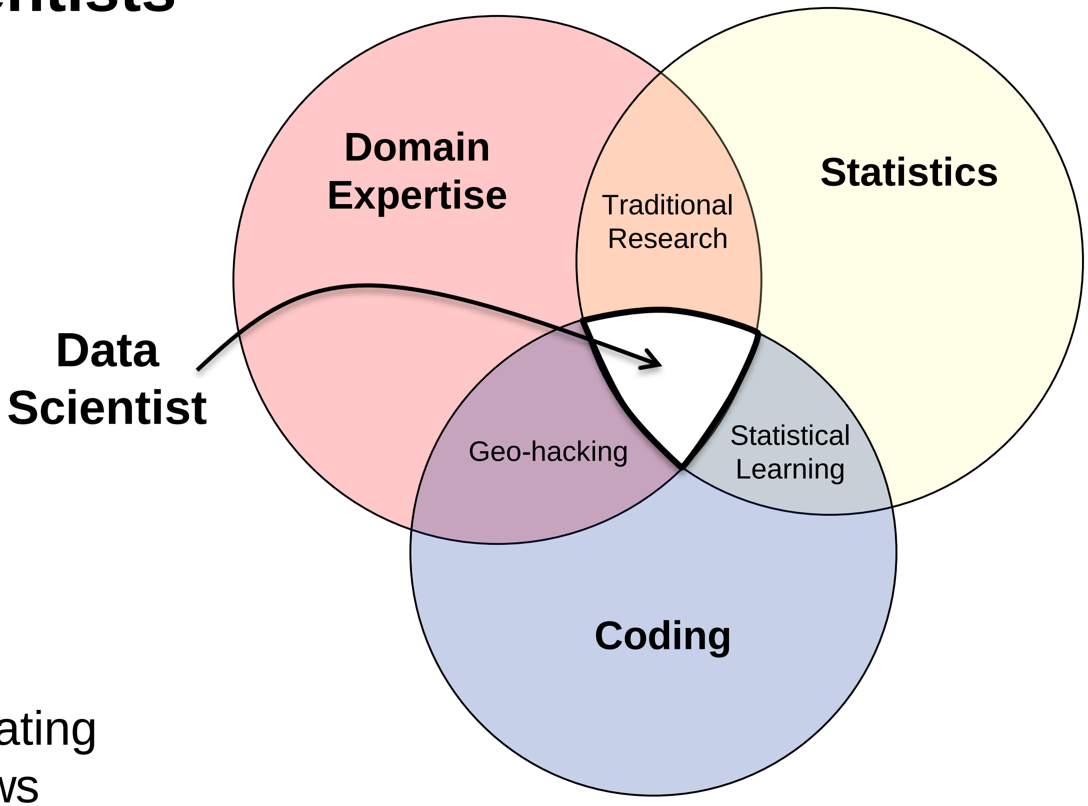
More about the hype cycle at https://www.gartner.com/en/documents/3887767, hype cycle image from https://www.gartner.com/en/research/methodologies/gartner-hype-cycle.

Developing Operational Capability

We need Data Scientists

Intersection of:

- Domain Expertise
  - Geoscience
  - Engineering
- Statistics
  - Probability
  - Data Analytics
- Coding
  - Scripting and Automating
  - Prototyping Workflows



Venn diagram for the data scientist.

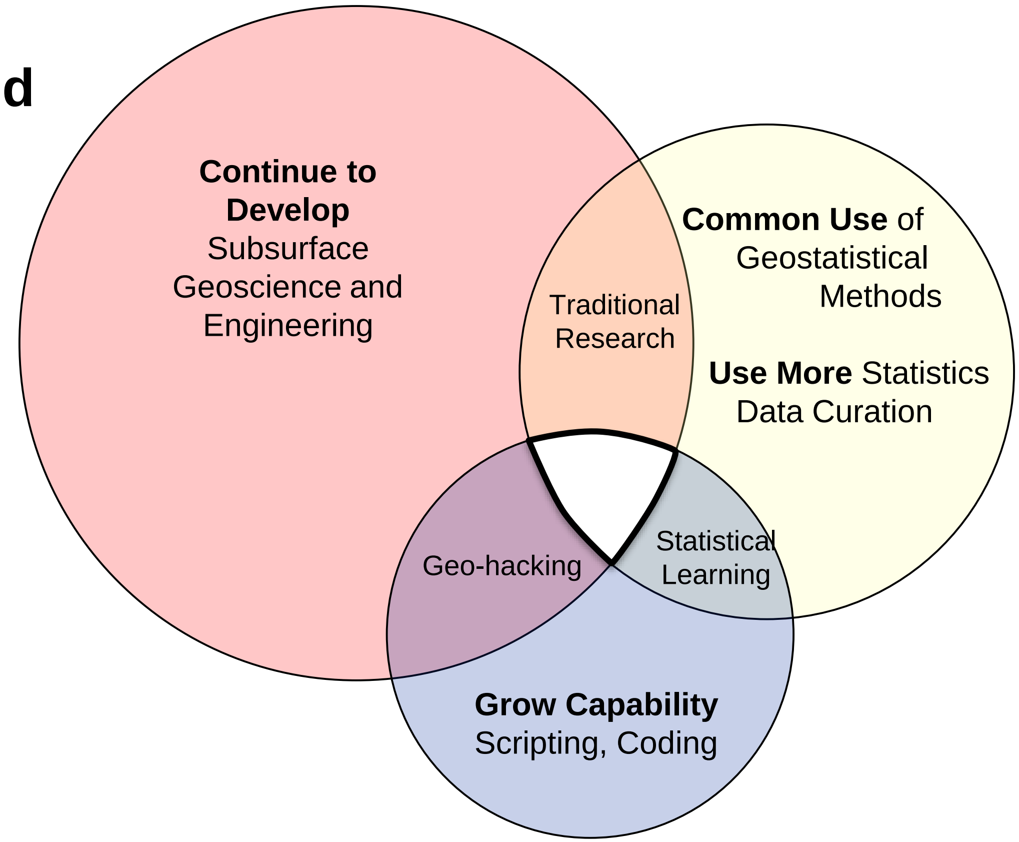
Developing Operational Capability

Graduate Geoscientists and Engineers with Data Analytics Capabilities

Well-prepared with data-driven knowledge to contribute in our industry

Build Capability in the Existing Geoscience and Engineering Workforce

Geoscience and engineering capability remains core to our work



Proposed diagram for a path forward for growing data science capabilities among geoscientists and engineers, regions scaled by importance.

Statistics to Mitigate Cognitive Biases

- Anchoring Bias: what we see is impacted by anything we have seen recently
- Recency Bias: we weight observations by how recently we saw them
- Confirmation Bias: we tend to see what confirms our current theory

‘I would not have seen it, if I hadn’t believed it!’

- Ashleigh Brilliant

PGE 383 Machine Learning

Machine Learning

Lecture outline . . .

- **Machine Learning Overview**
- **Examples of Machine Learning**
- **Energy Machine Learning**

Michael Pyrcz, The University of Texas at Austin