PGE 383 Subsurface Machine Learning

Lecture 6: Machine Learning

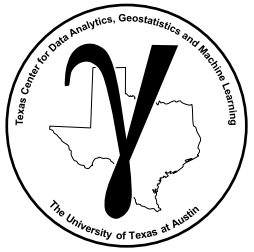
Lecture outline:

- **Machine Learning Overview**
- **Model Fitting, Overfitting and Model Generalization**
- **Examples of Machine Learning**
- **Energy Machine Learning**

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.



Motivation

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

- Inferencial and predictive machine learning
- Training model parameters and tuning model hyperparameters
- Training and testing splits for cross validation
- Model performance and overfit

PGE 383 Subsurface Machine Learning

Lecture 6: Machine Learning

Lecture outline:

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

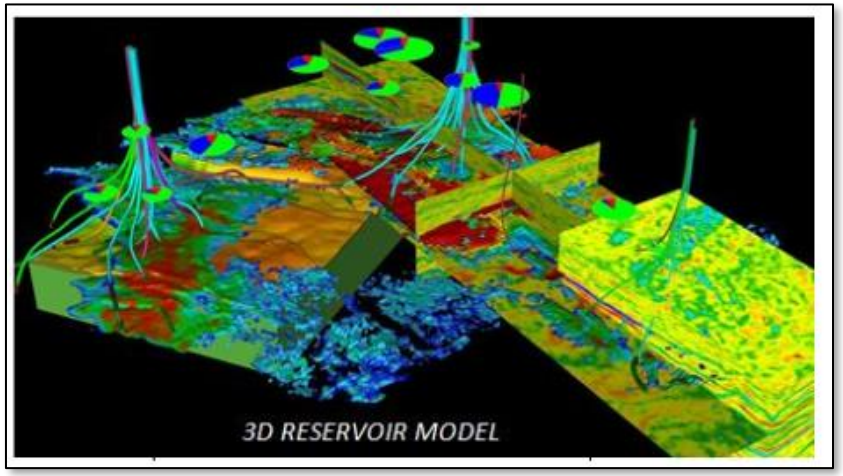
Big Data

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

Subsurface engineering and geoscience is often working with big data!

2D geological map of Texas (https://vizcart.io/products/texas-geo-1933?variant=47982320386395).



3D reservoir model with various data sources (http://www.oil-gasportal.com/reservoir-management/integrated-reservoir-modeling).

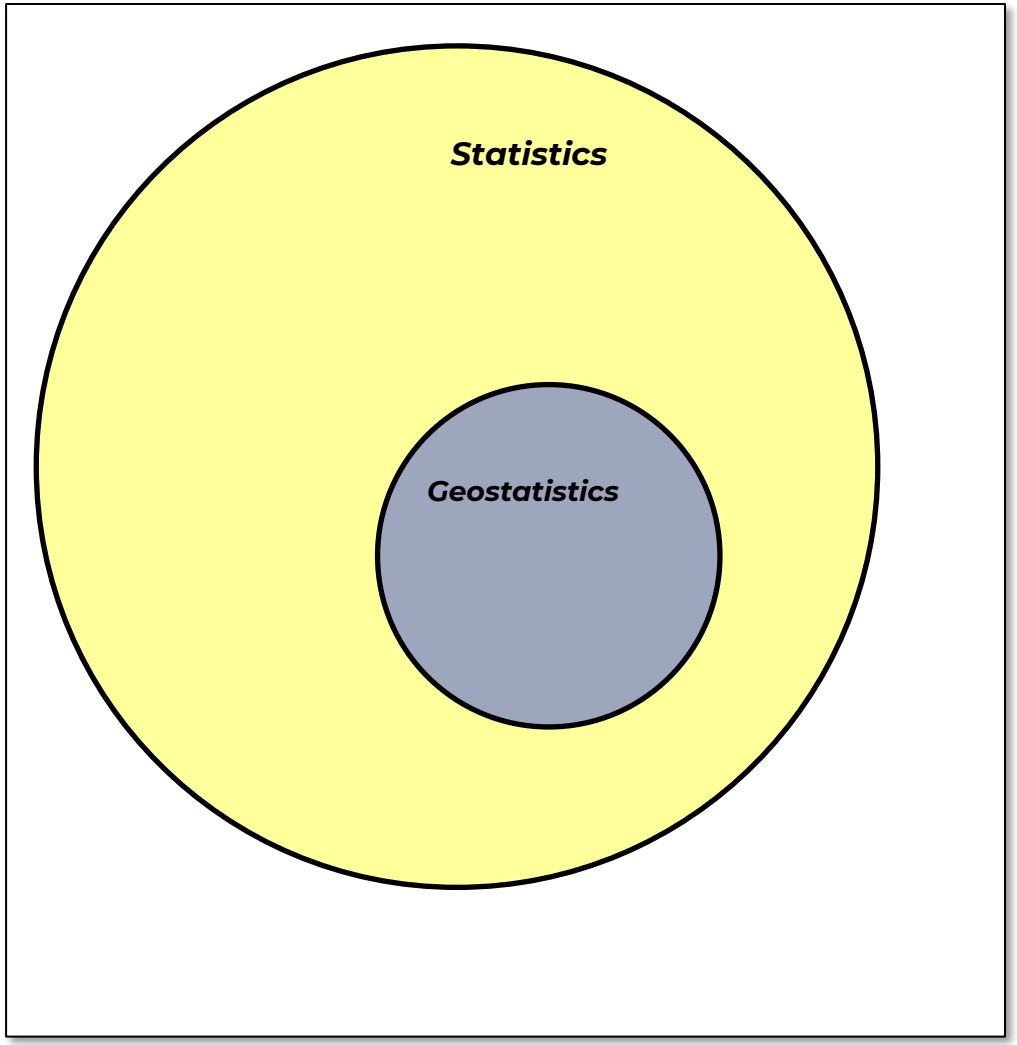
Big Data Analytics

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

Geostatistics is a branch of applied statistics:

1. spatial (geological) context
2. spatial relationships
3. volumetric support / scale
4. Uncertainty

Data analytics is statistics.

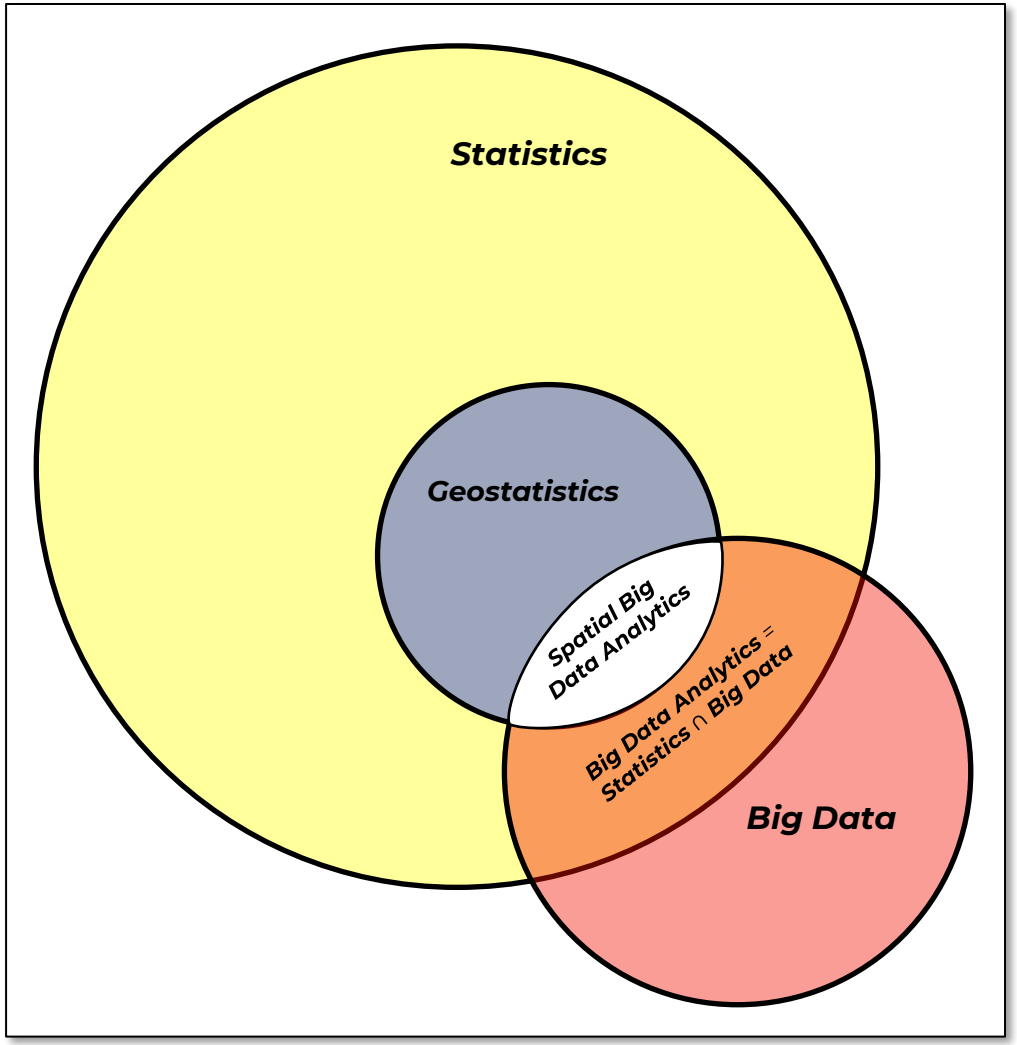


Proposed Venn diagram for statistics and geostatistics.

Big Data Analytics

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.



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

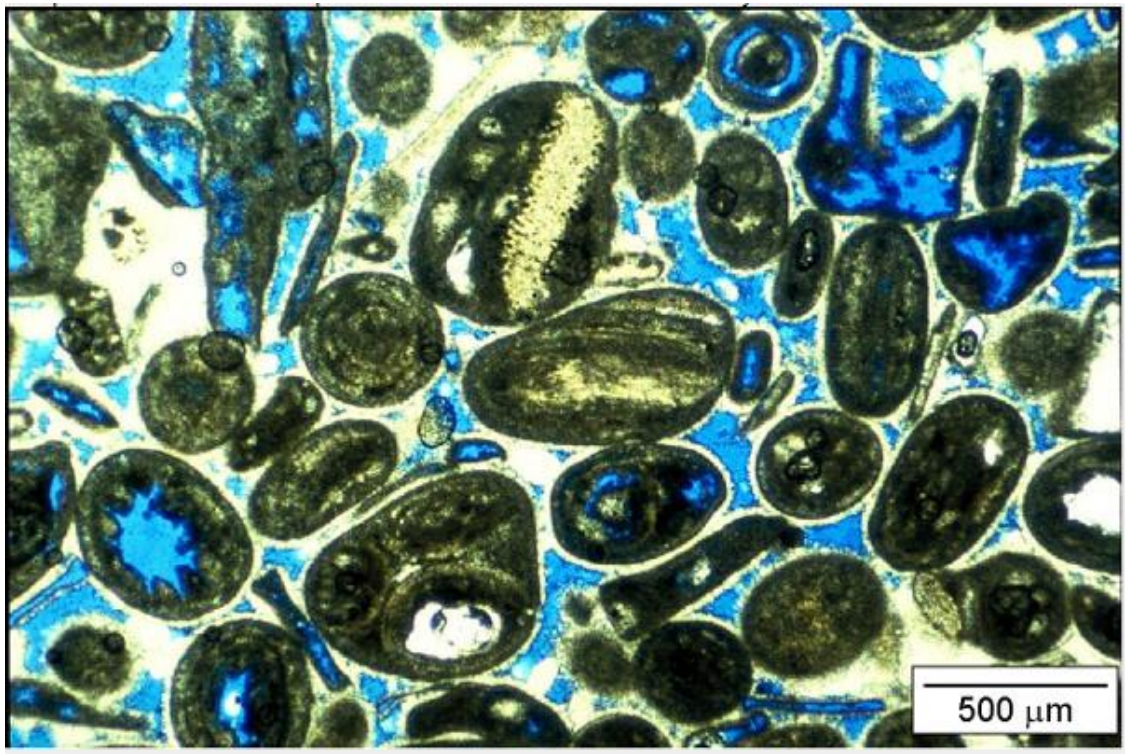
"… where it is
infeasible to develop an algorithm of specific instructions for performing the task."

Quote from Machine Learning - Wikipedia

Variables / Features

Variable / Feature: any property measured / observed, 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

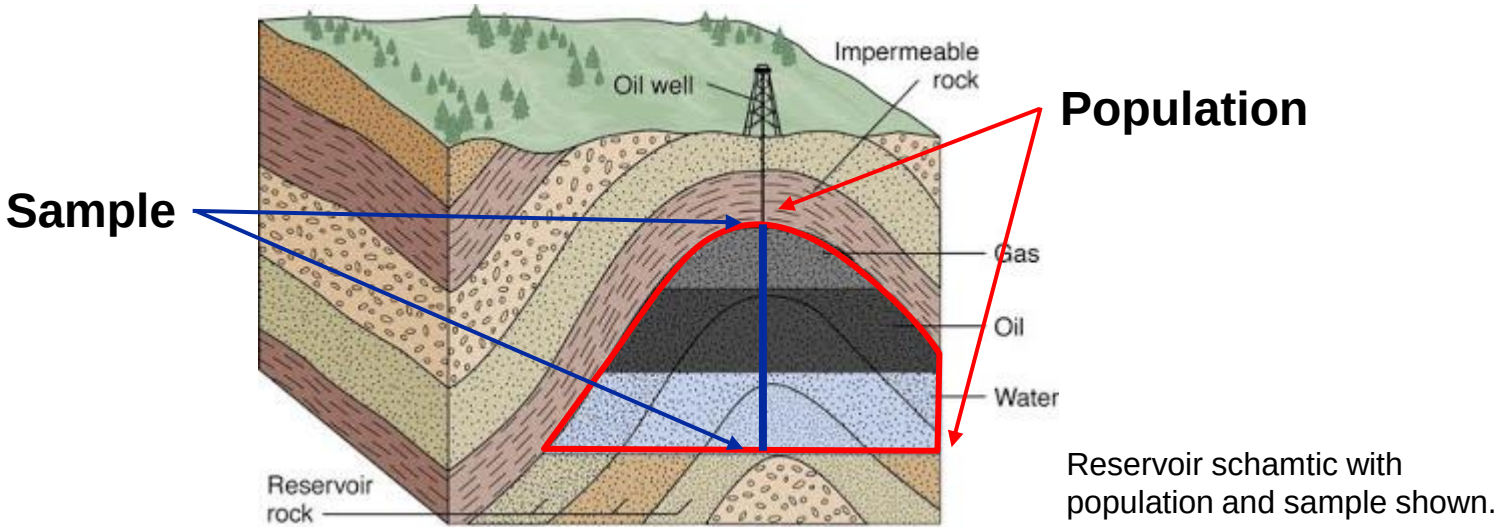
Population and Sample

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.

Sampel and population, image modified from https://energyeducation.ca/encyclopedia/Oil_and_gas_reservoir.

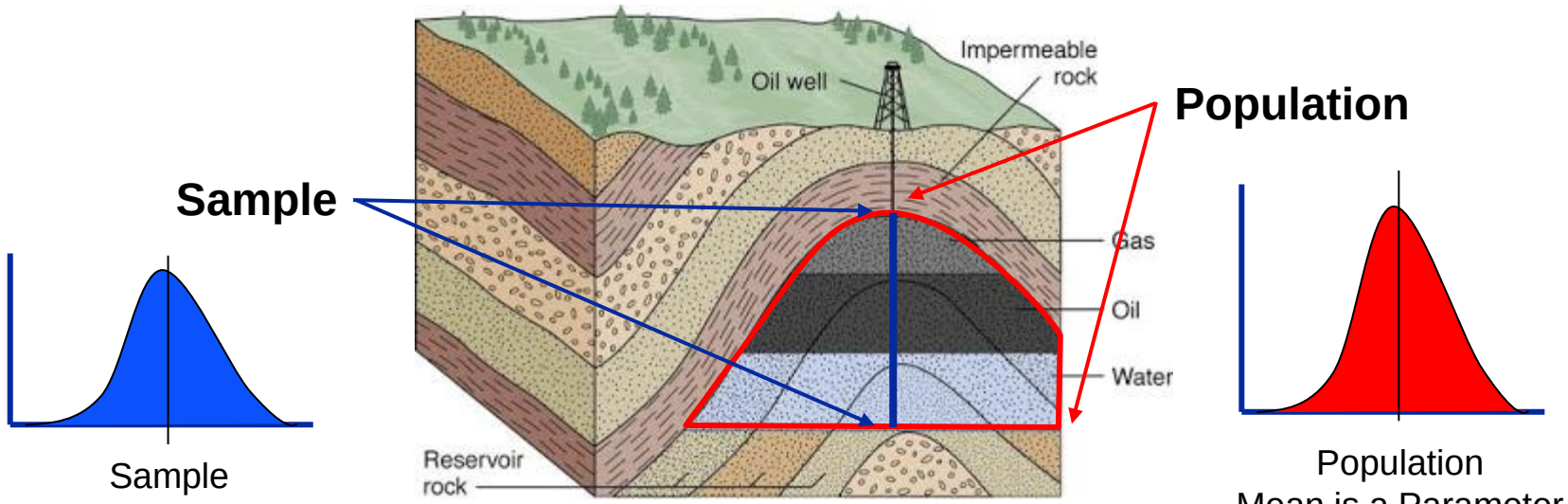
Parameter and Statistic

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

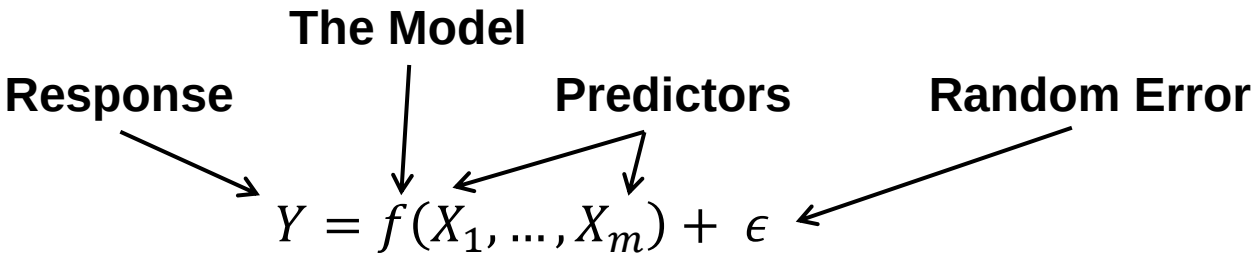
- e.g. sample mean, sample standard deviation, we use statistics as estimates of the parameters



Reservoir schematic with parameter and statistic shown.

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

Machine Learning Nuts and Bolts



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!

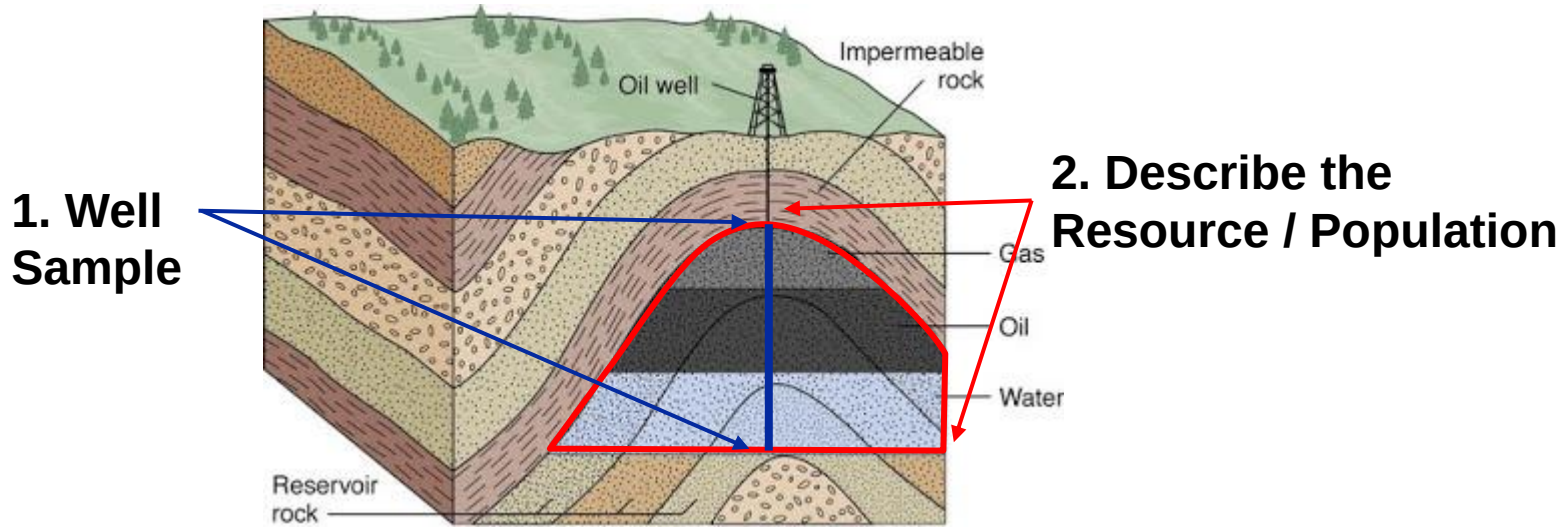
Recall, Inferential Statistics

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

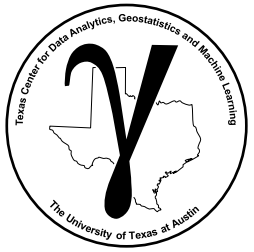
Inference

Inferential Statistics

- Given a sample from a population, describe the population
- Given the well/drill hole samples, describe the resource



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



Prediction

The best estimate of the response feature

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

- 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

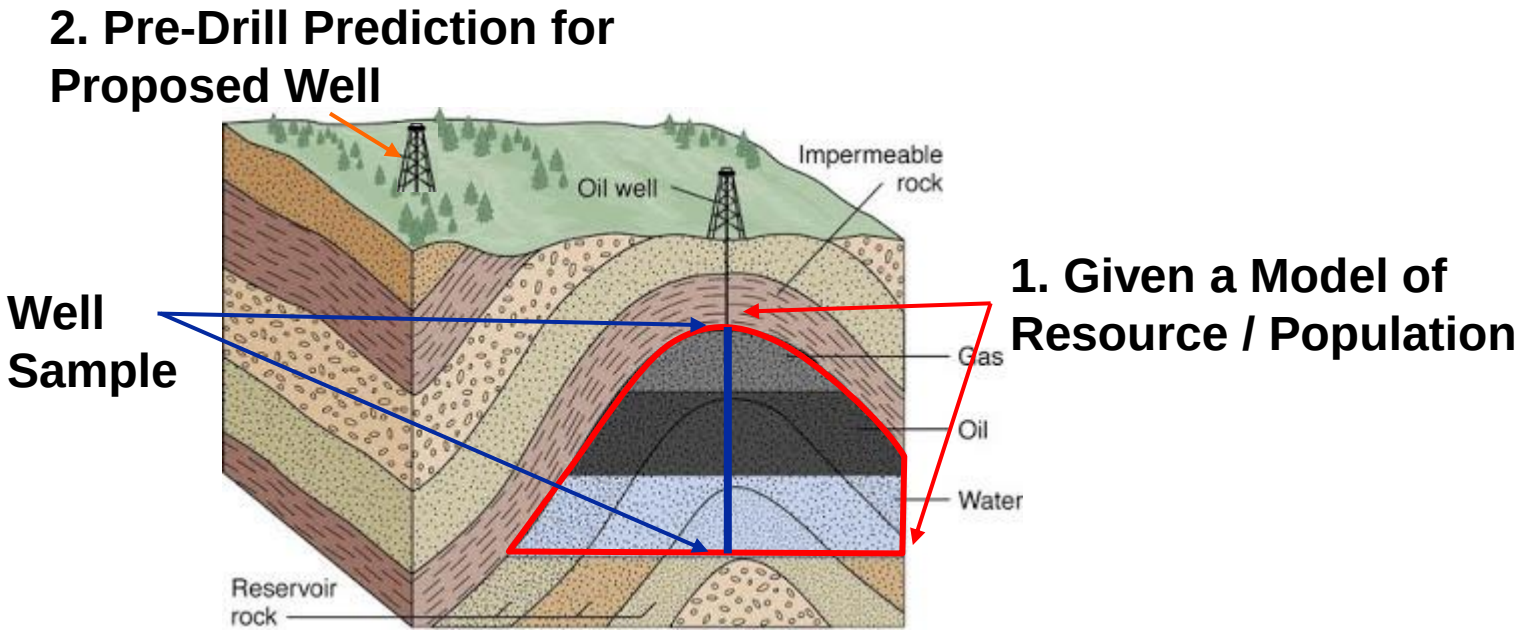
Recall, Predictive Statistics

- 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 resource, predict the next well/drill hole (pre-drill assessment) sample, e.g. porosity, permeability, grade, etc.



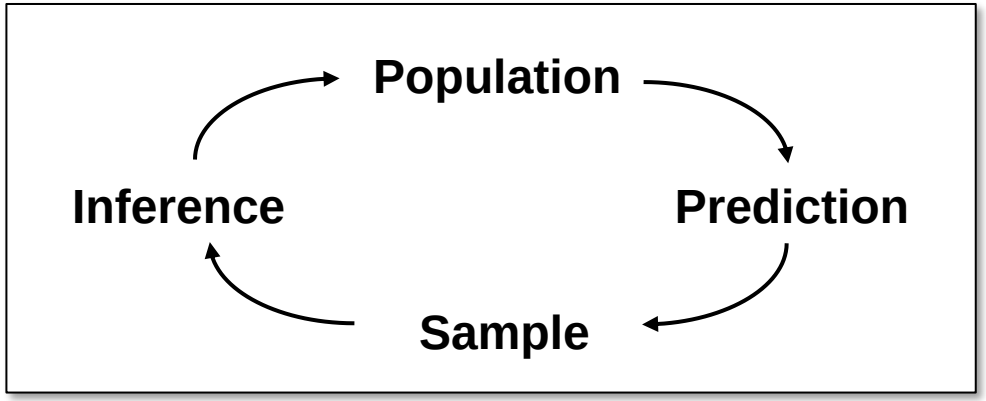
Reservoir schematic with inference problem, given well sample, describe the reservoir, population.

Inference and Prediction

In General Inference Precedes Prediction

Inference - model of the population from the sample

Prediction - predict the next sample from the model of the population



Inference to model the population from a limited sample, prediction to predict the next sample from the model of the population.

Parametric Models

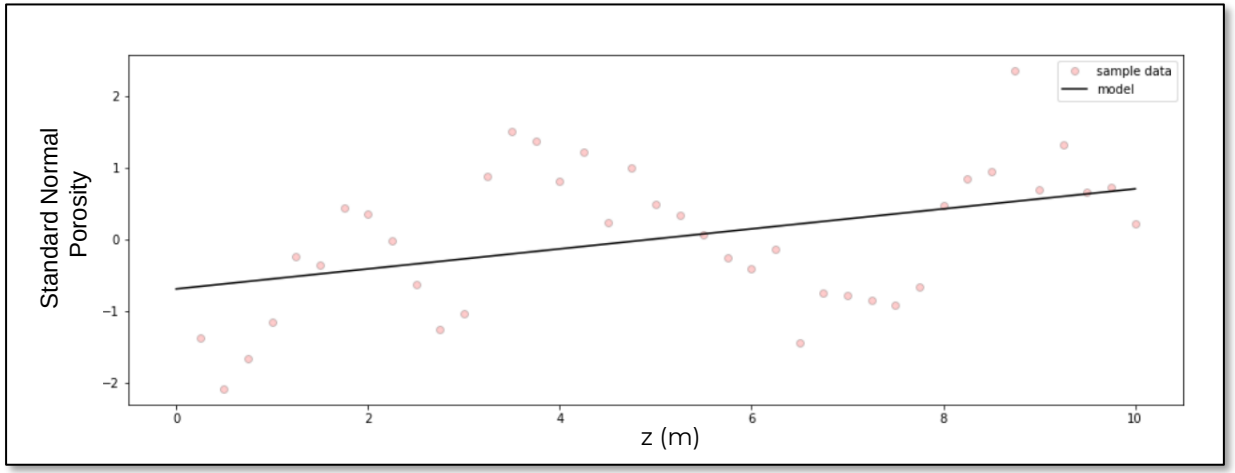
Working with Parametric Models

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:

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



Linear regression model to predict porosity from the z coordinate.

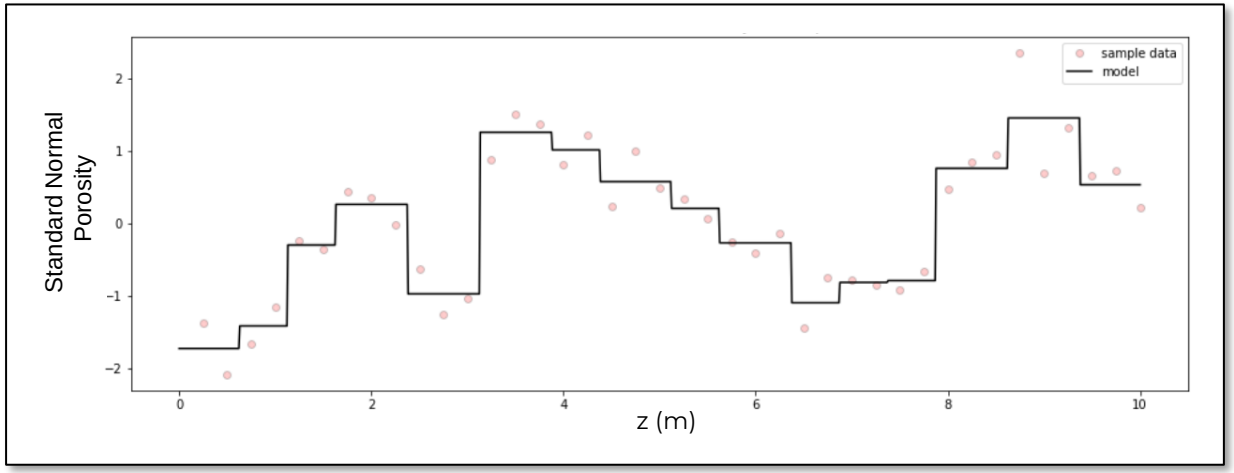
- There is a risk that $\hat{f}$ is quite different than f, then we get a poor model!

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

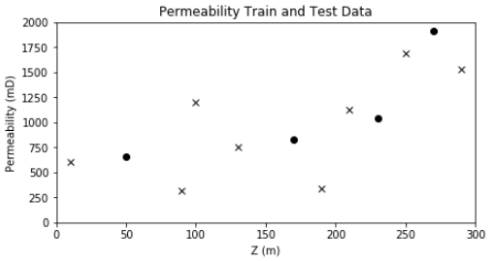


Decision tree regression model to predict porosity from the z coordinate.

- *'Nonparametric is actually parametric rich!'*

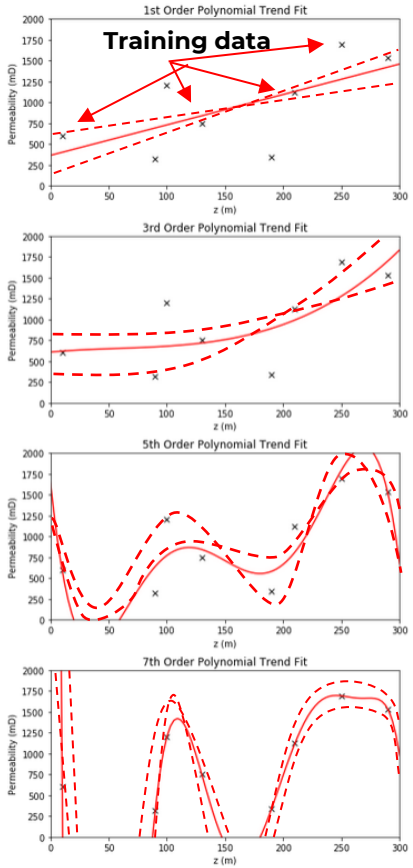
Predictive Model Workflow

1. Split the Data into Train and Test Subsets



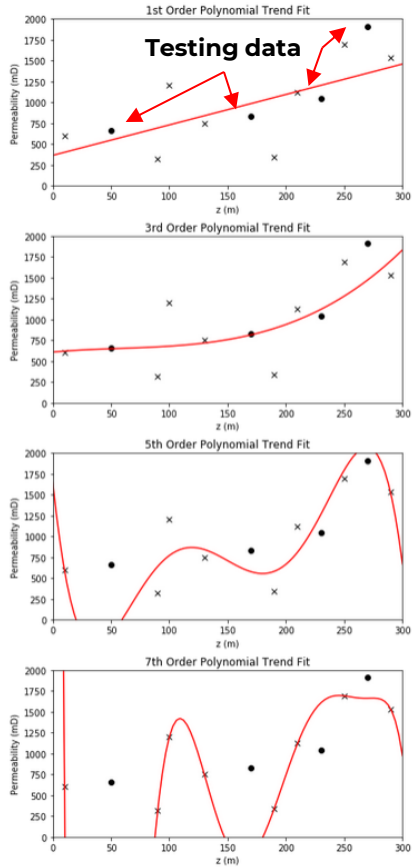
2. Build Models Over a Range of Hyperparameters Increasing Model Complexity

3. Train the Model Parameters with Training Data



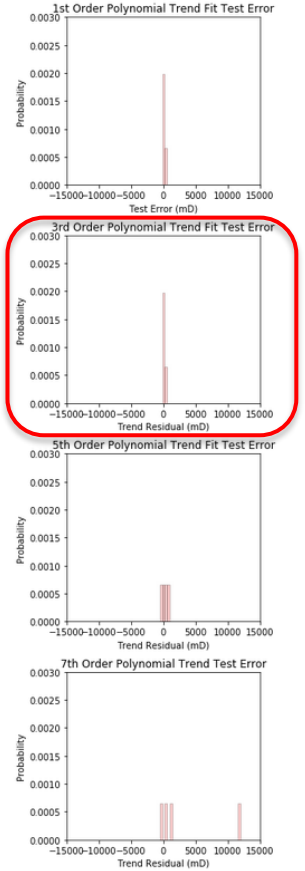
Best Fit Models to Training Data

4. Check Each Best Fit Model with Withheld Testing Data



Error Over the Testing Data

5. Tune the Model Hyperparameters with Testing Data



Hyper-parameters

Retrain the model with tuned hyperparameters and all data, training & testing.

Apply the retrained model.

All of this work to calculate tuned hyperparameters.

Now make our model.

Machine learning model building workflow to avoid overfit.

Model Parameters

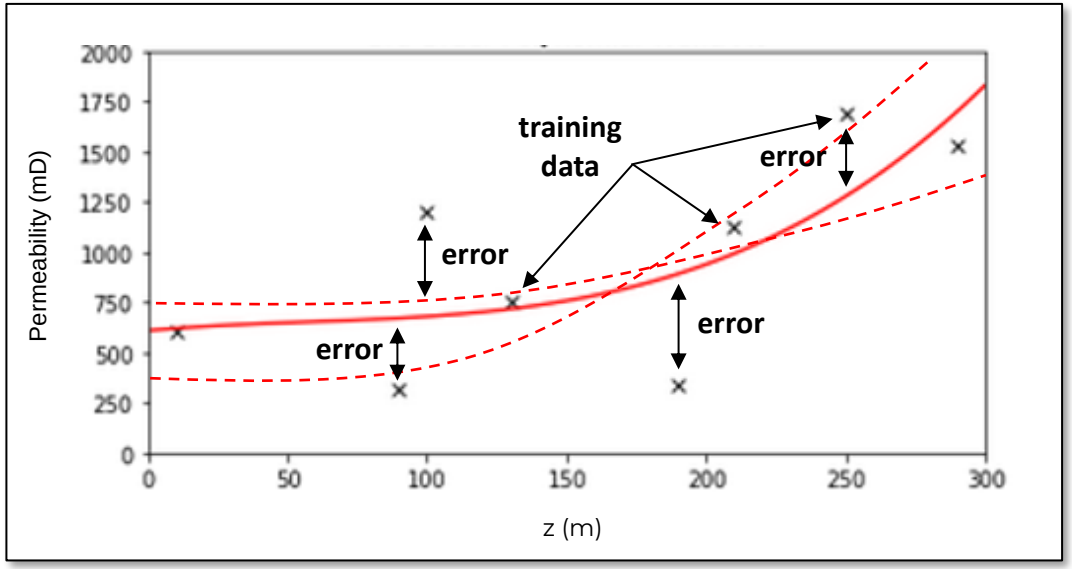
Model Parameters

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

$$y = \boldsymbol{b_3}x^3 + \boldsymbol{b_2}x^2 + \boldsymbol{b_1}x + \boldsymbol{b_0}$$

Parameters:

$\boldsymbol{b_3}$, $\boldsymbol{b_2}$, $\boldsymbol{b_1}$ and $\boldsymbol{b_0}$



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

Model HyperParameters

Model Hyperparameters

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

- For a polynomial model:

Increasing Complexity

$\vdots$

7th Order: $y = b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$

$\vdots$

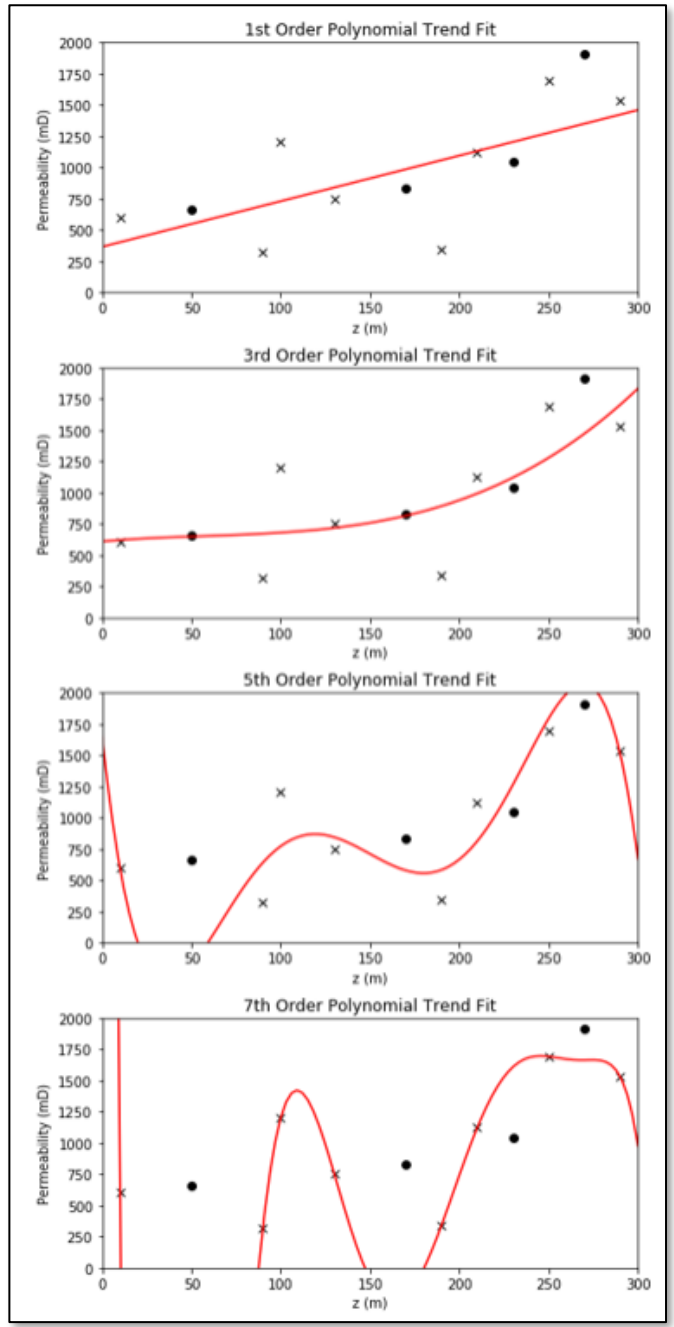
5th Order: $y = b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$

$\vdots$

3rd Order: $y = b_3x^3 + b_2x^2 + b_1x + b_0$

2nd Order: $y = b_2x^2 + b_1x + b_0$

1st Order: $y = b_1x + b_0$



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.

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

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

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

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

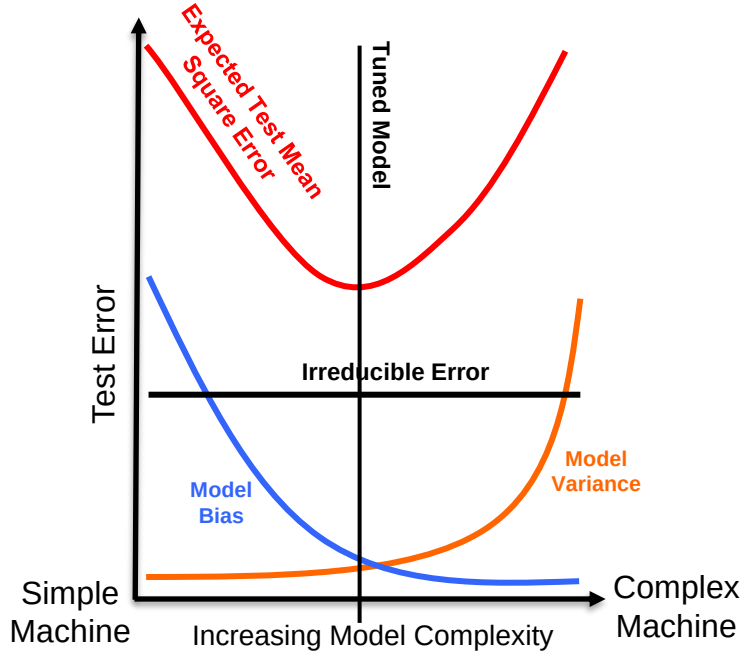
- We want to know how our model performs when we move away from the training data!

Ensemble Prediction Method

What is the Testing Accuracy of Our Predictive Machine Learning Models?

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

- **Model Variance** is the error in the model predictions due to sensitivity to the data (what if we used different training data?)
- **Model Bias** is error in the model predictions due to using an approximate model / model is too simple
- **Irreducible error** is error in the model predictions due to missing features and limited samples can't be fixed with modeling / entire feature space is not sampled

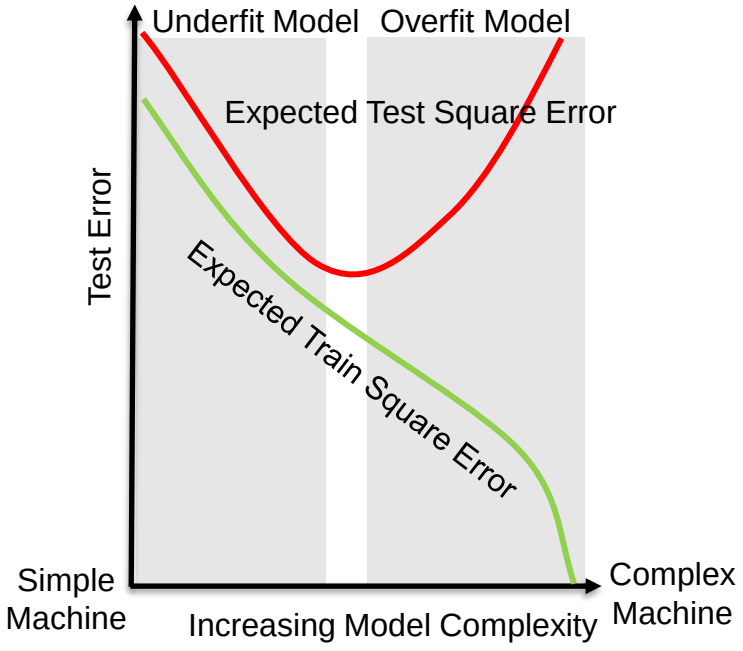


Model variance and bias trade-off.

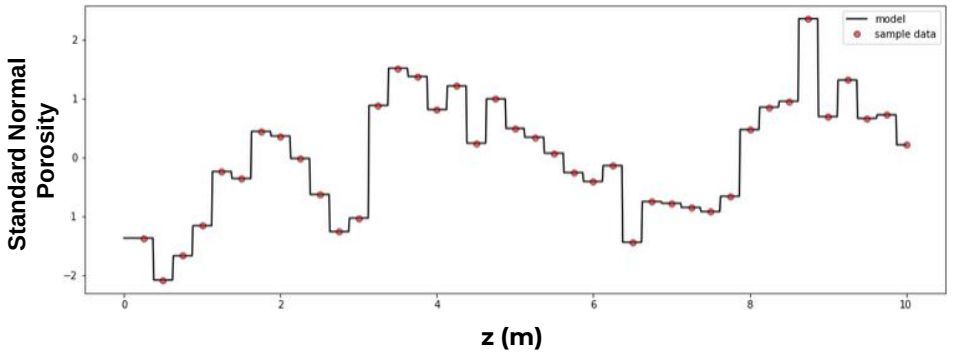
Model Overfit

Model Overfit

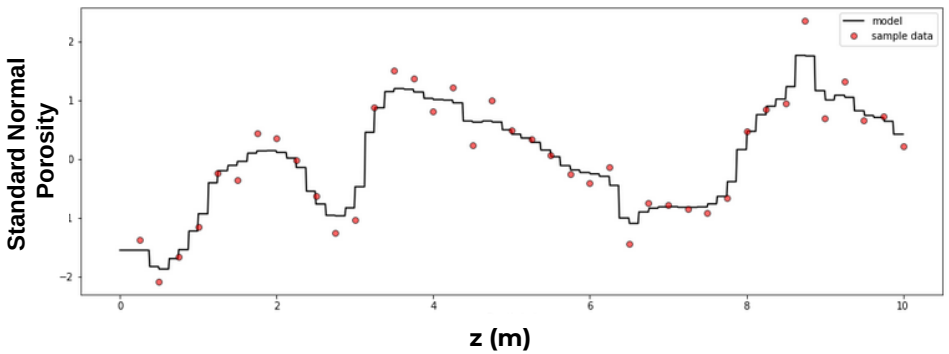
- Fitting data noise / data idiosyncrasies
- Increased complexity will generally decrease error with respect to the training dataset
- but, may result in increase error with testing data → at this complexity/flexibility we are overfit!



Model variance and bias trade-off.



Overfit model to training data.



A more balanced fit model to training data.

Training and Testing Splits

The Training and Testing Split

The most common approach is random selection

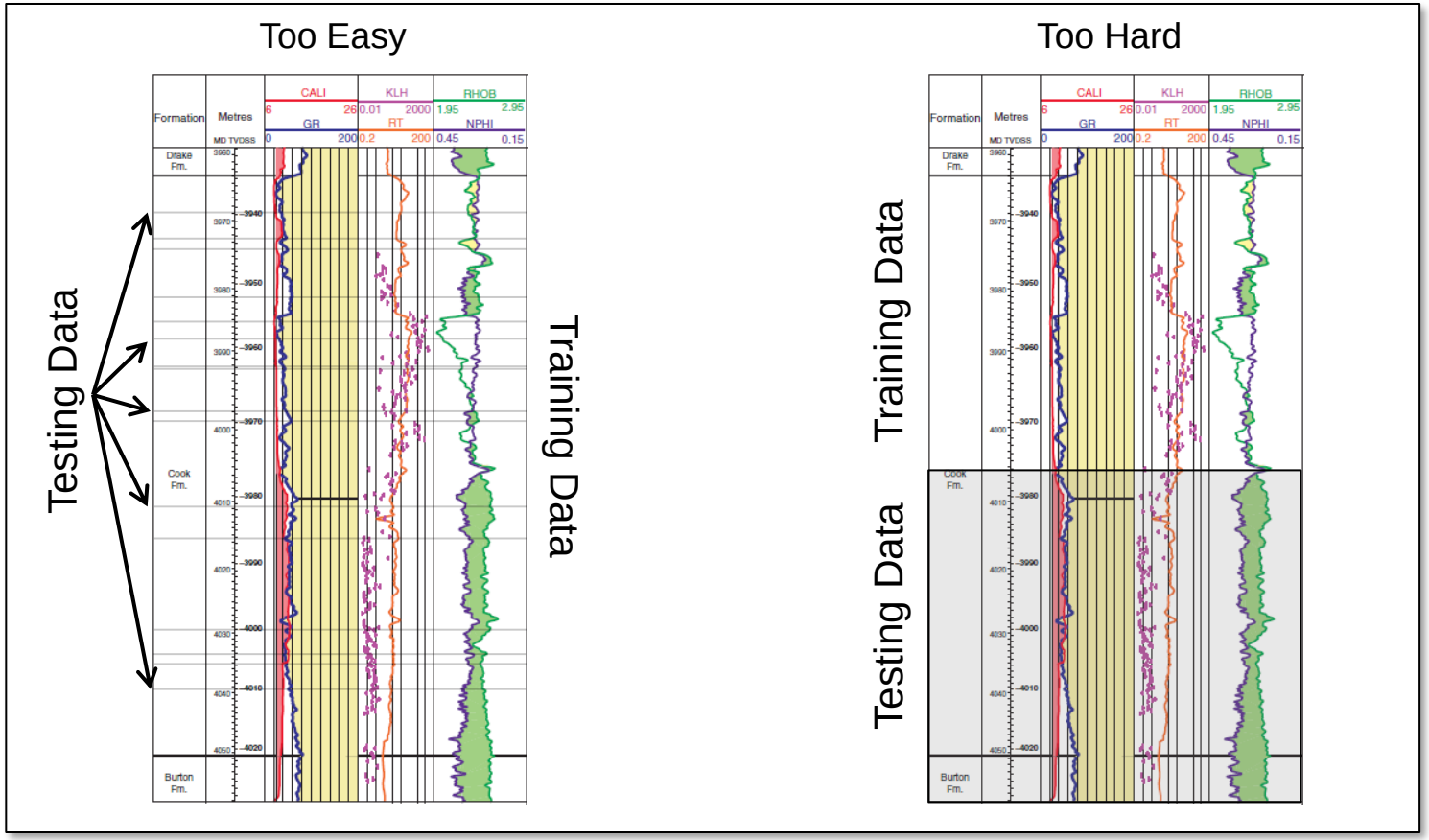
- this may not be fair testing

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 Splits

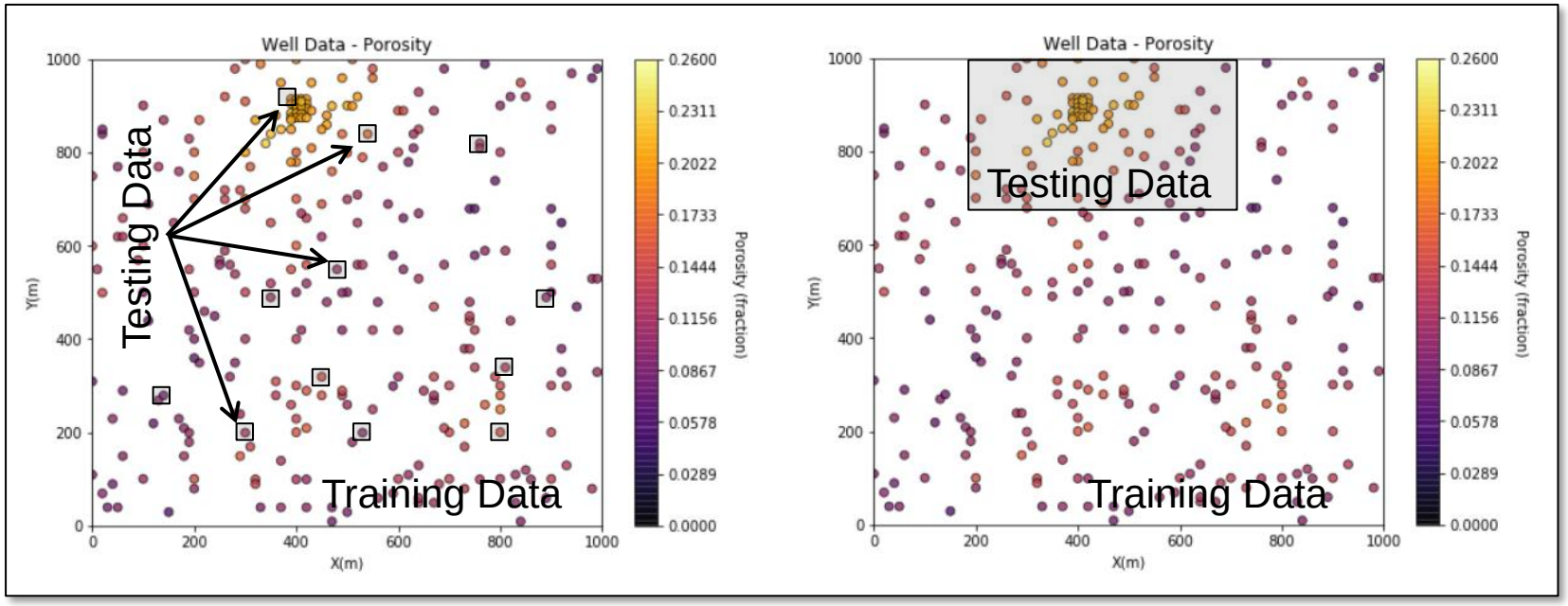
Fair Testing in Spatial / Temporal Settings



Predictions only at ½ ft offsets (left) and predictions in a different rock (right).

Training and Testing Splits

Fair Testing in Spatial / Temporal Settings



Predictions at only short offsets from training data (left) and distinctly different range of reservoir values (right).

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

- More could be done, but we do this for brevity.

Model Complexity / Flexibility Definition

Model Complexity / Flexibility

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

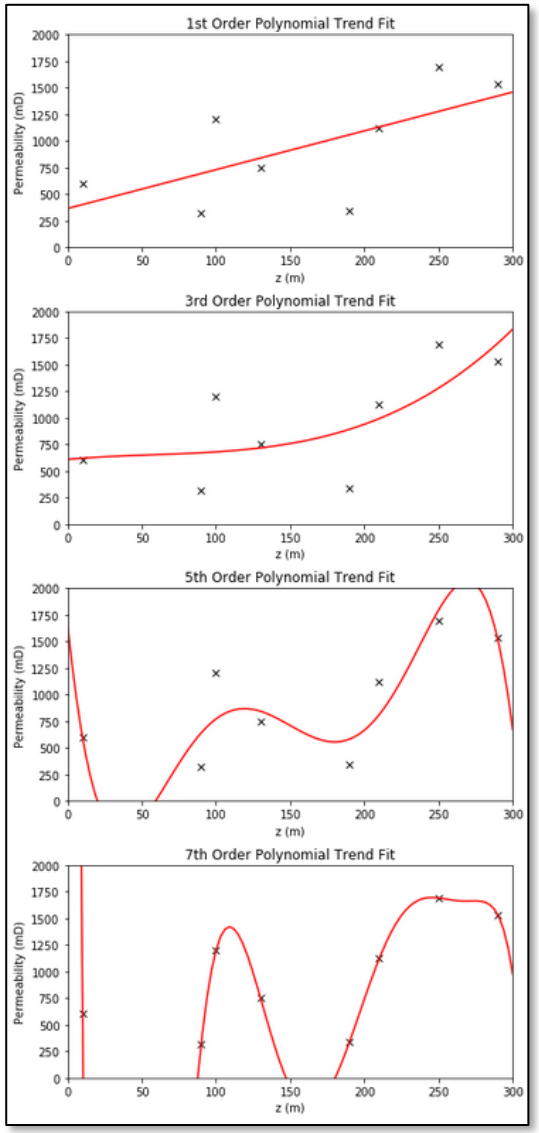
- The number of features, predictor variables are in the model, dimensionality of the model

- 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 vs. nested conditional statements – decision tree

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

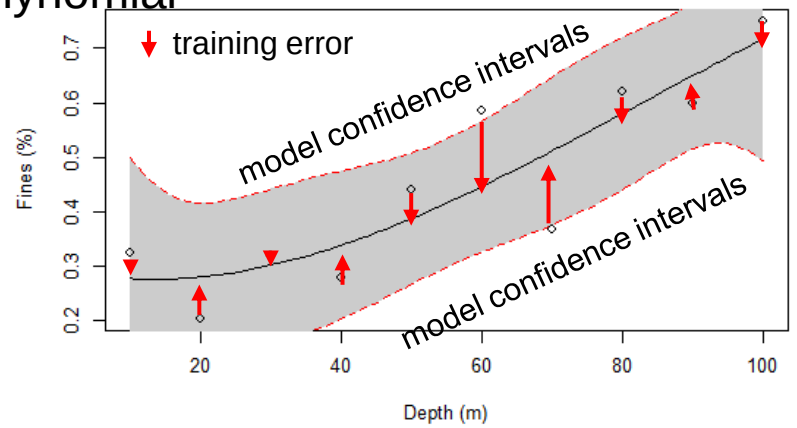


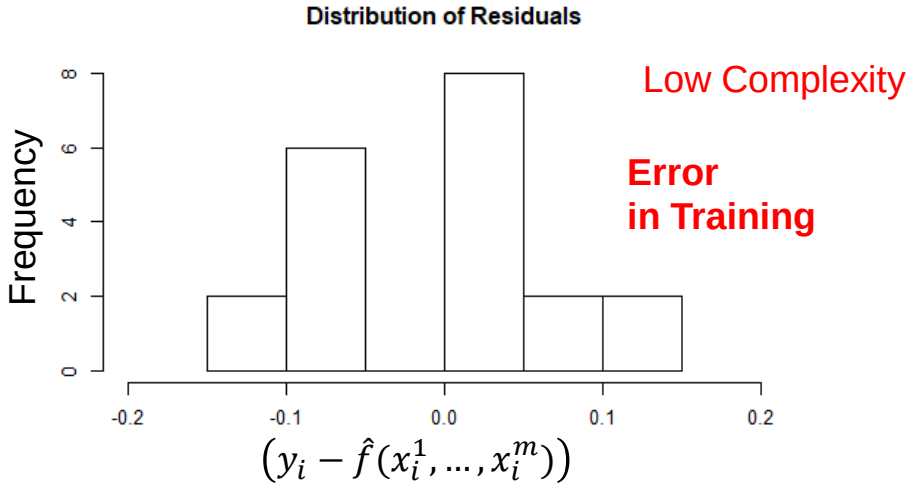
Simple to complicated polynomial models, from MachineLearning_overfit chapter of e-book.

Simple Statistical Demonstration Overfitting

Example of trend fits:

- 3rd Ordered Polynomial

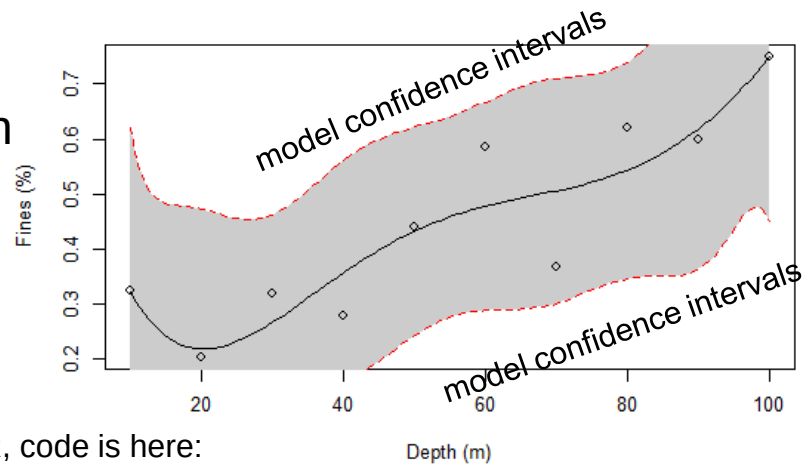


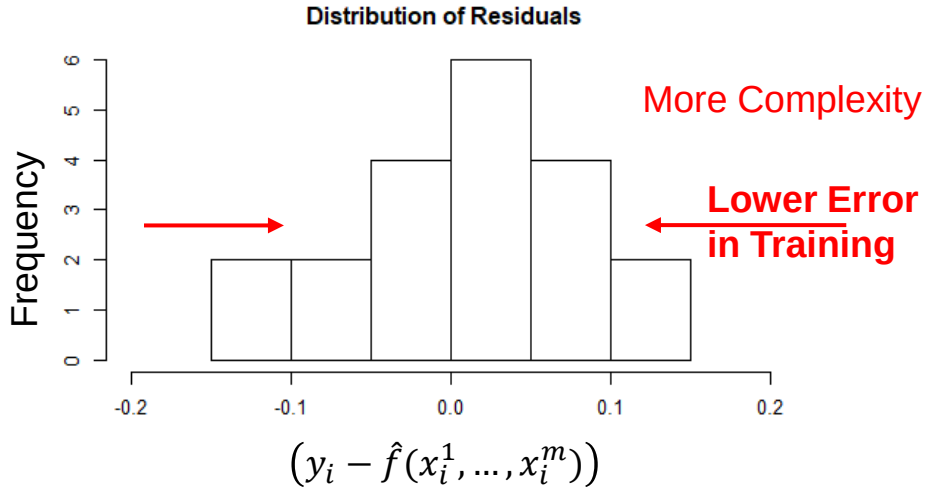


Low Complexity

Error in Training

- 5th Order Polyn





More Complexity

Lower Error in Training

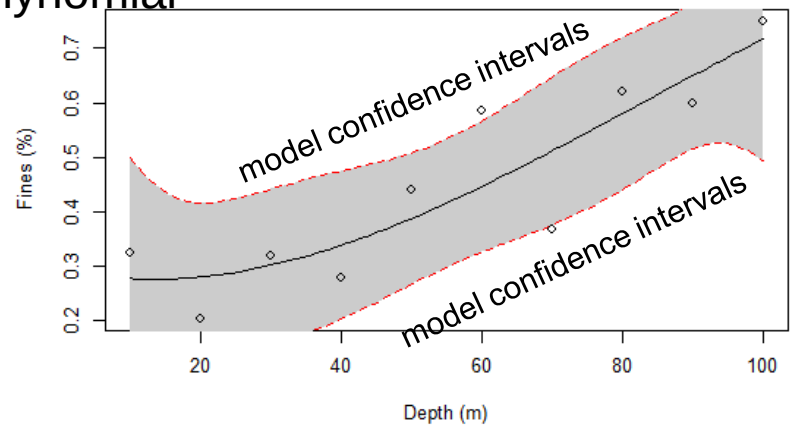
Overfit demonstration in R, code is here:
https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R

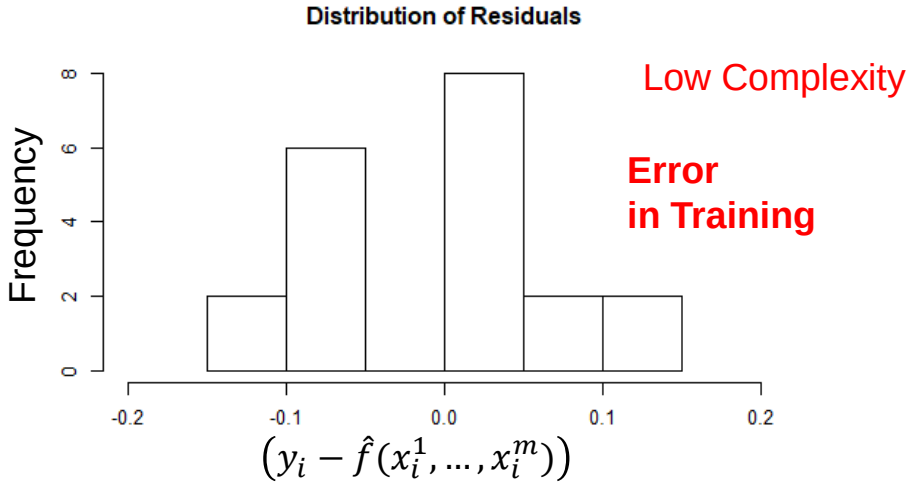
R code at Code/Overfit.R

Simple Statistical Demonstration Overfitting

Example of trend fits:

- 3rd Ordered Polynomial

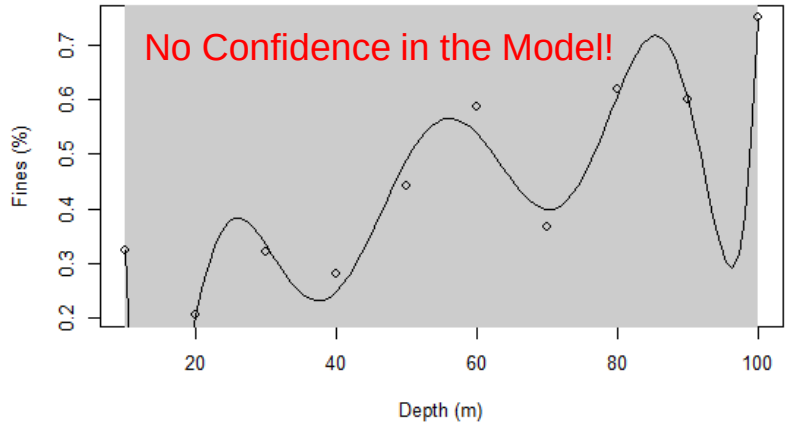


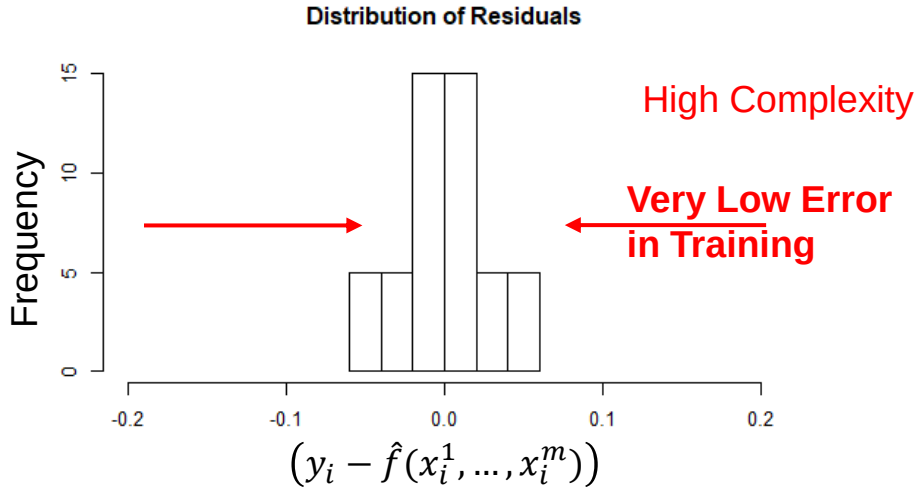


Low Complexity

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

PGE 383 Subsurface Machine Learning

Lecture 6: Machine Learning

Lecture outline:

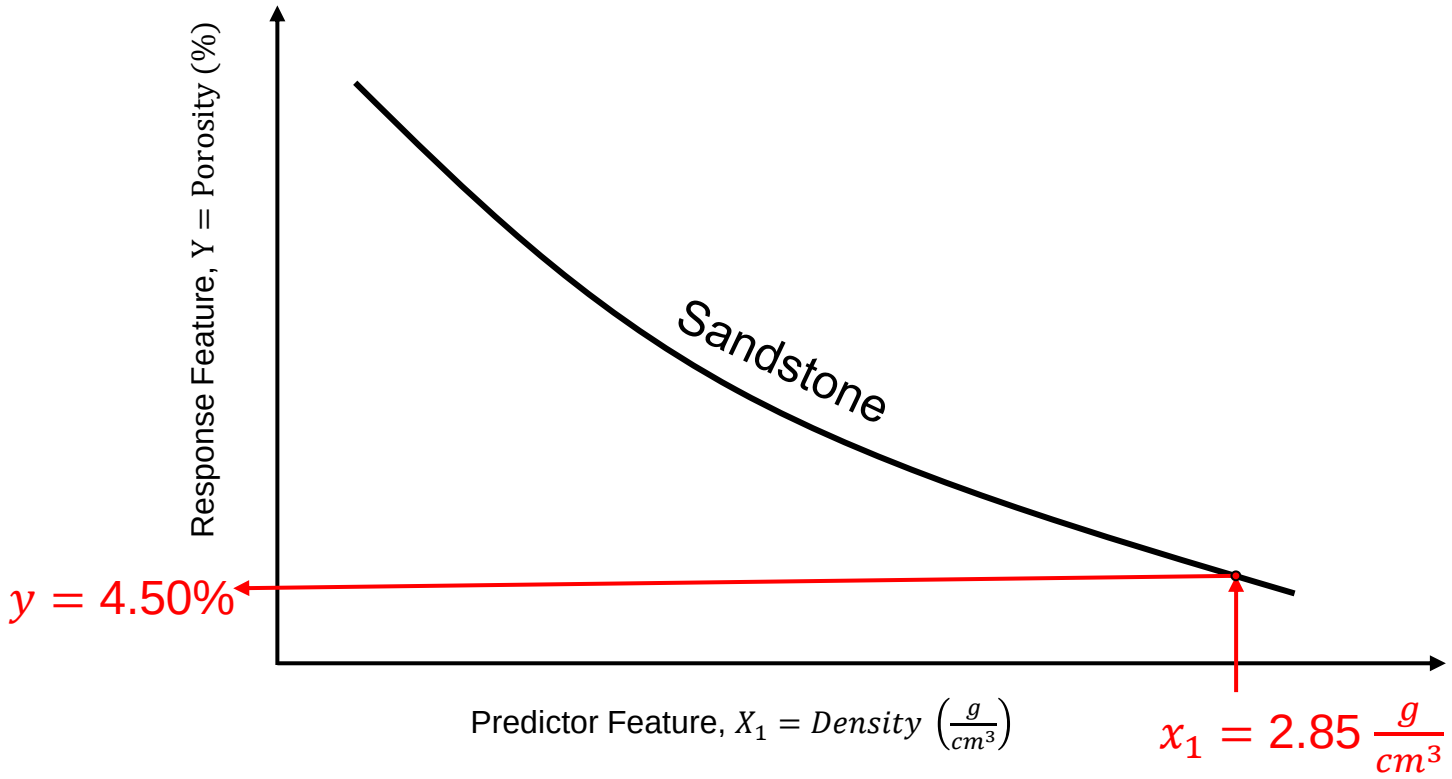
- **Model Fitting, Overfitting and Model Generalization**

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.

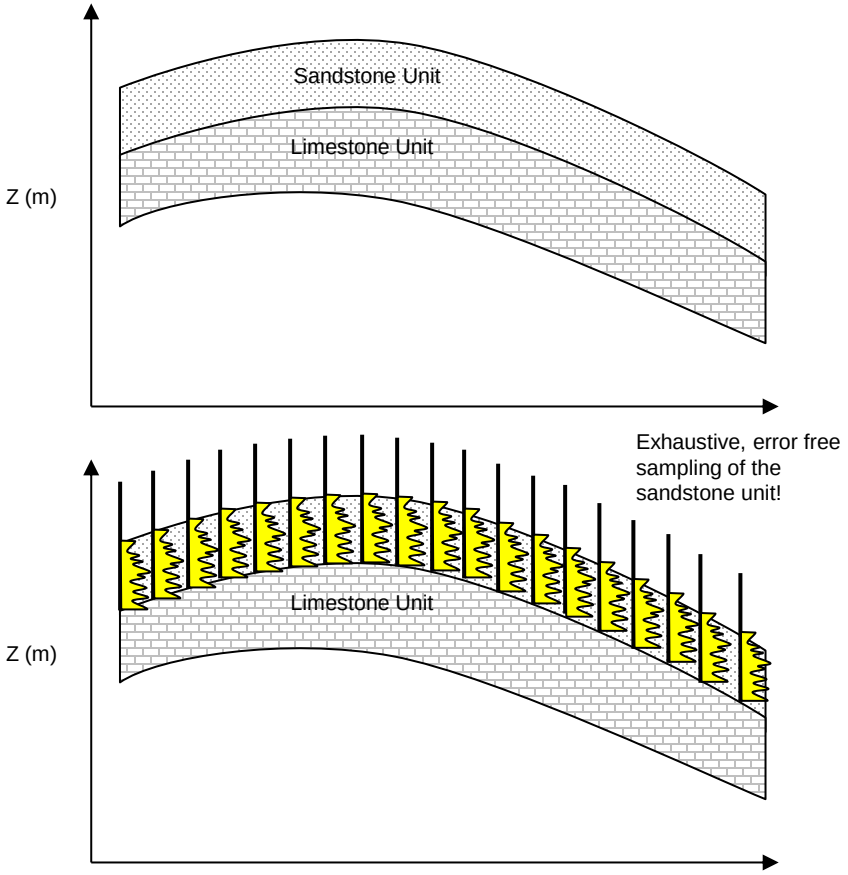
rock porosity from the well log density measure for your sandstone

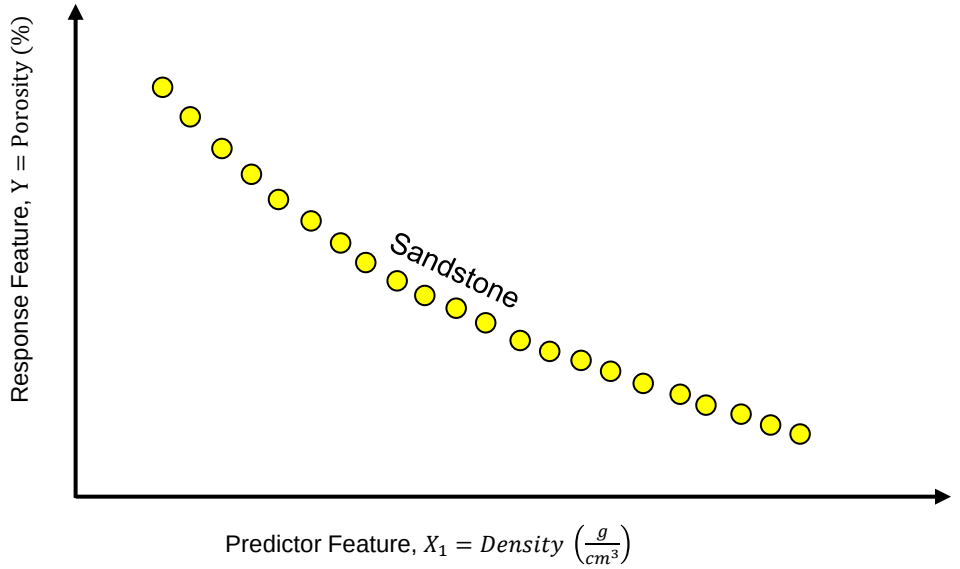


Fitting, Overfitting and Model Generalization Example

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.

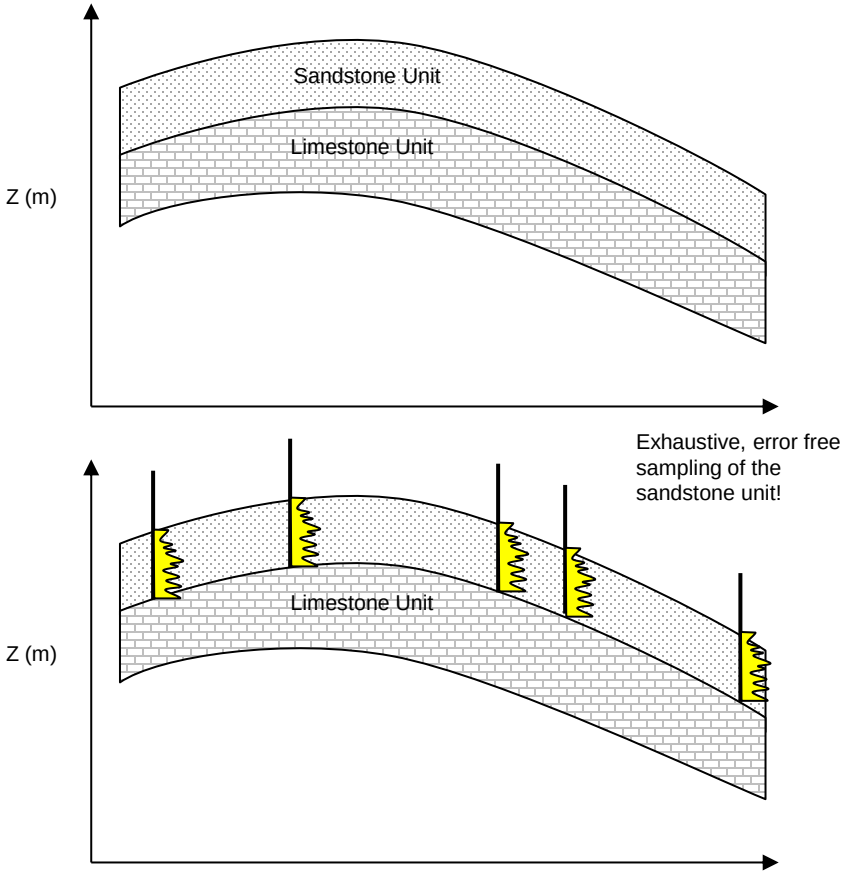




Fitting, Overfitting and Model Generalization Example

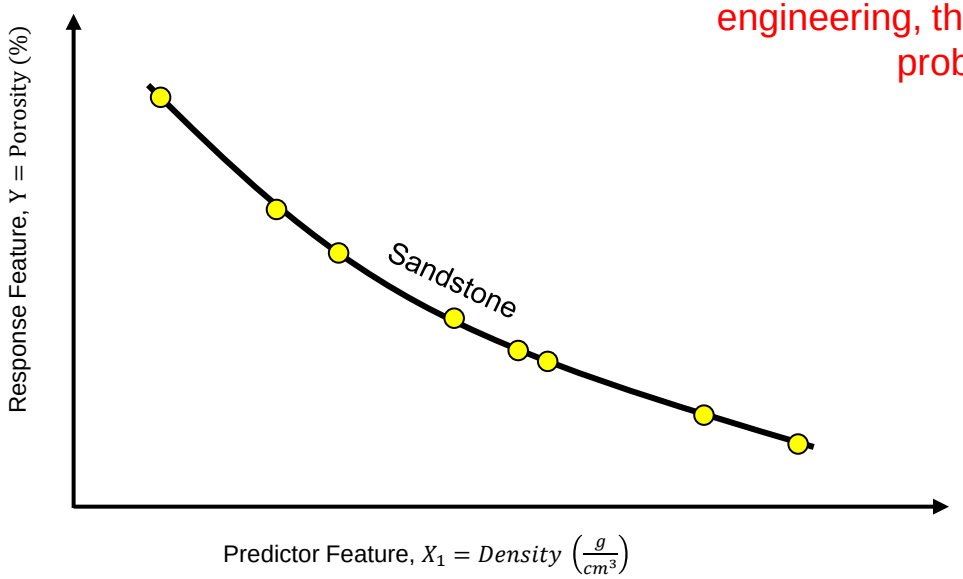
Assume you integrate physics and limited samples from the population.

- 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.

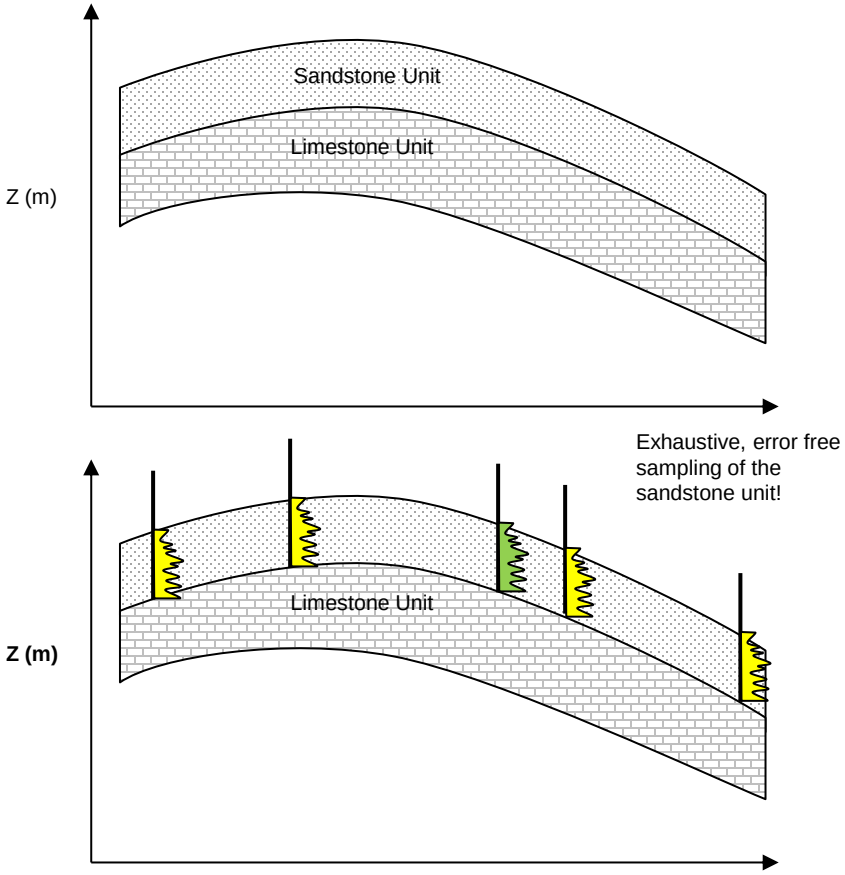


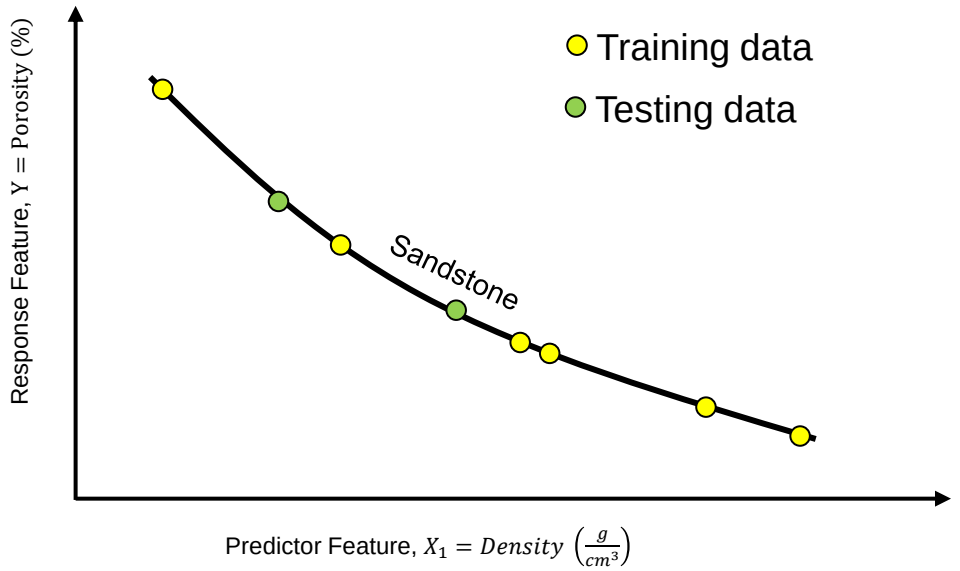
Fitting, Overfitting and Model Generalization Example

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

We separate the data into, training data to train the model parameters - fit

testing data, withheld from training, to tune the model hyperparameters - complexity



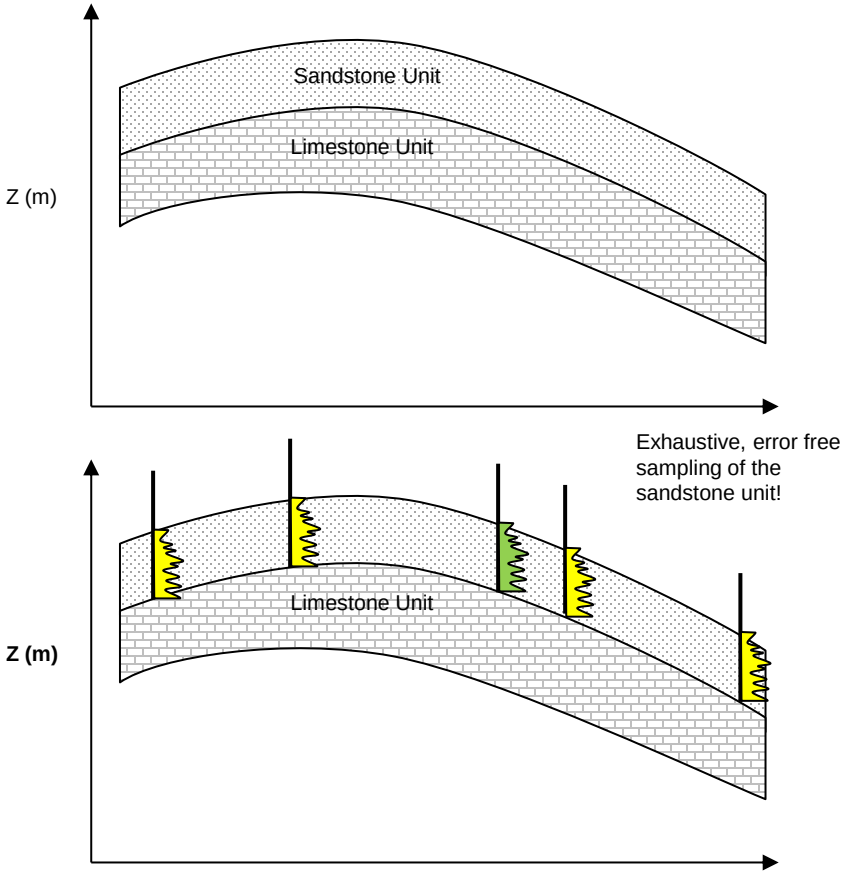


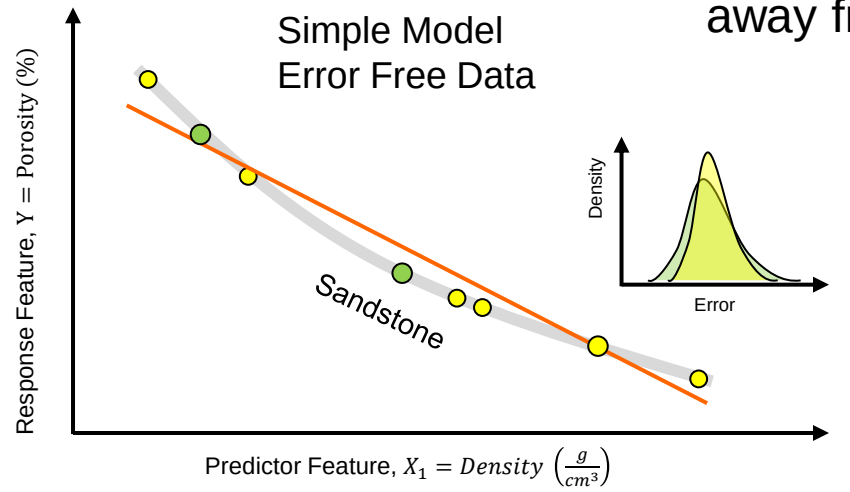
Fitting, Overfitting and Model Generalization Example

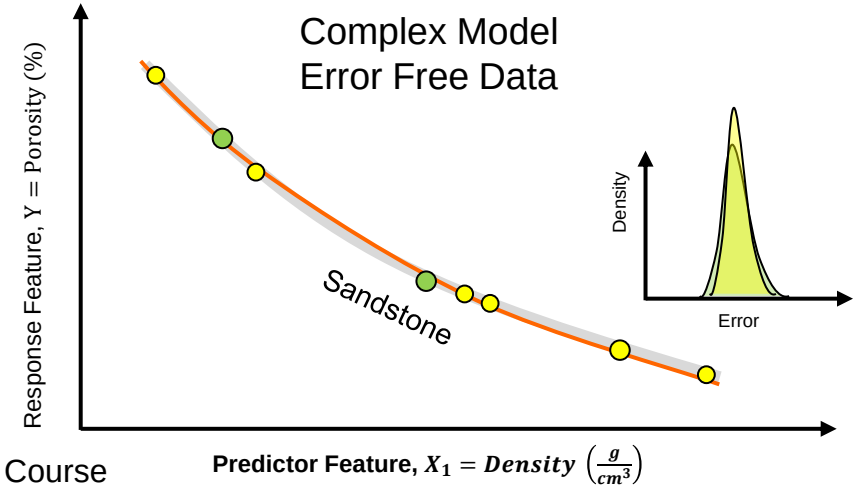
Assume the data-driven approach, training/tuning a model, $Y = f(X_1)$.

- 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 model generalizes ok away from training!







Fitting, 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:

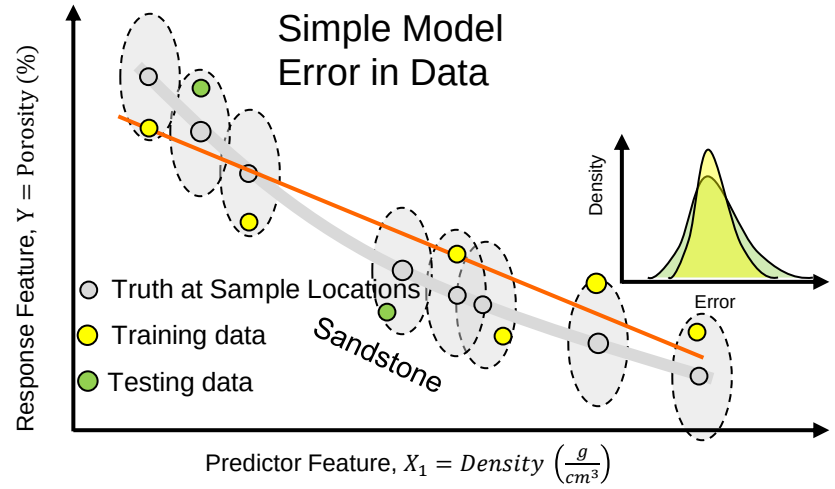
- Less sensitive to error/noise in the data

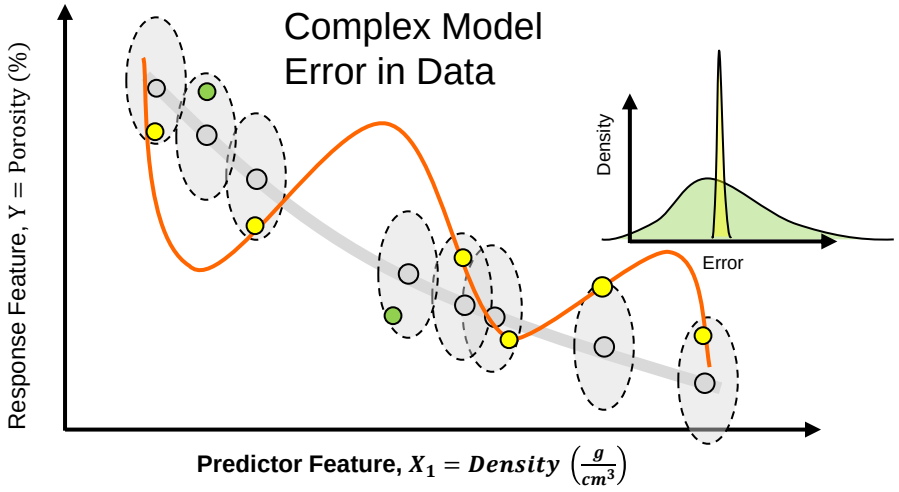
Complexity:

- 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

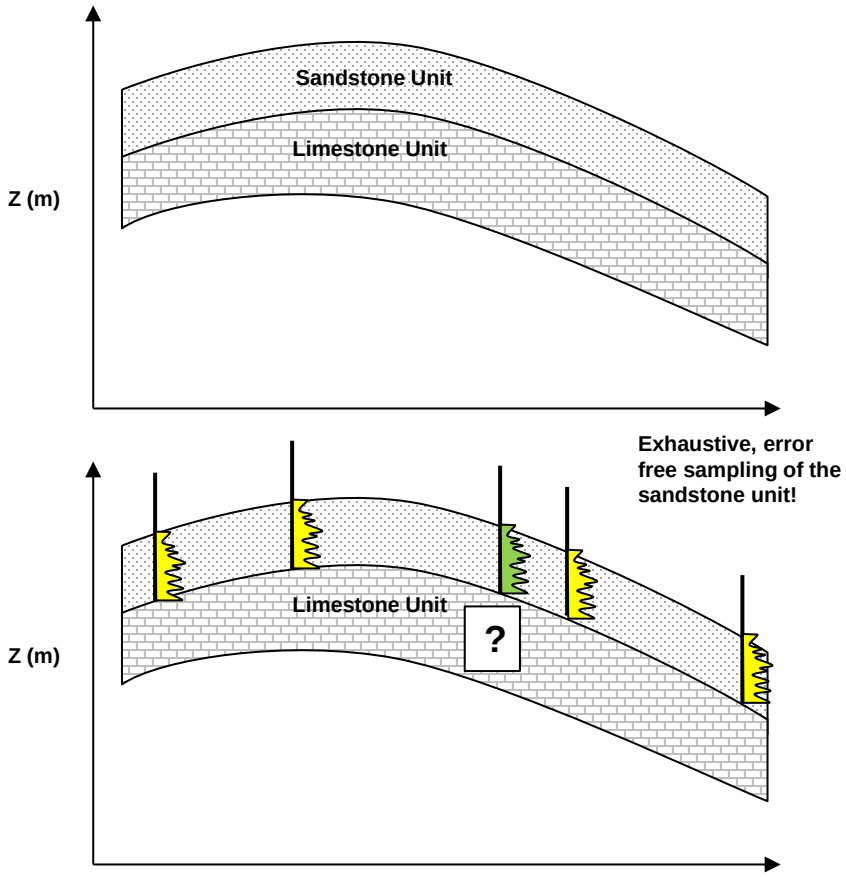


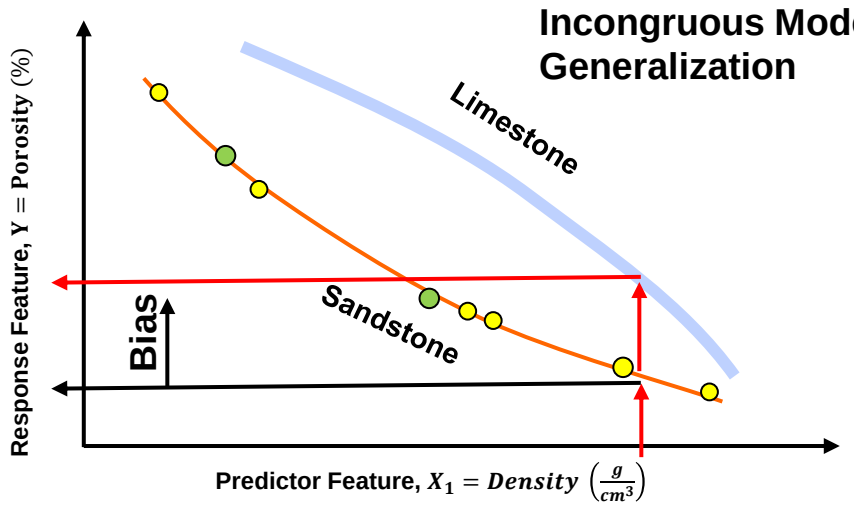


Fitting, Overfitting and Model Generalization Example

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.

Overfit

Overfit

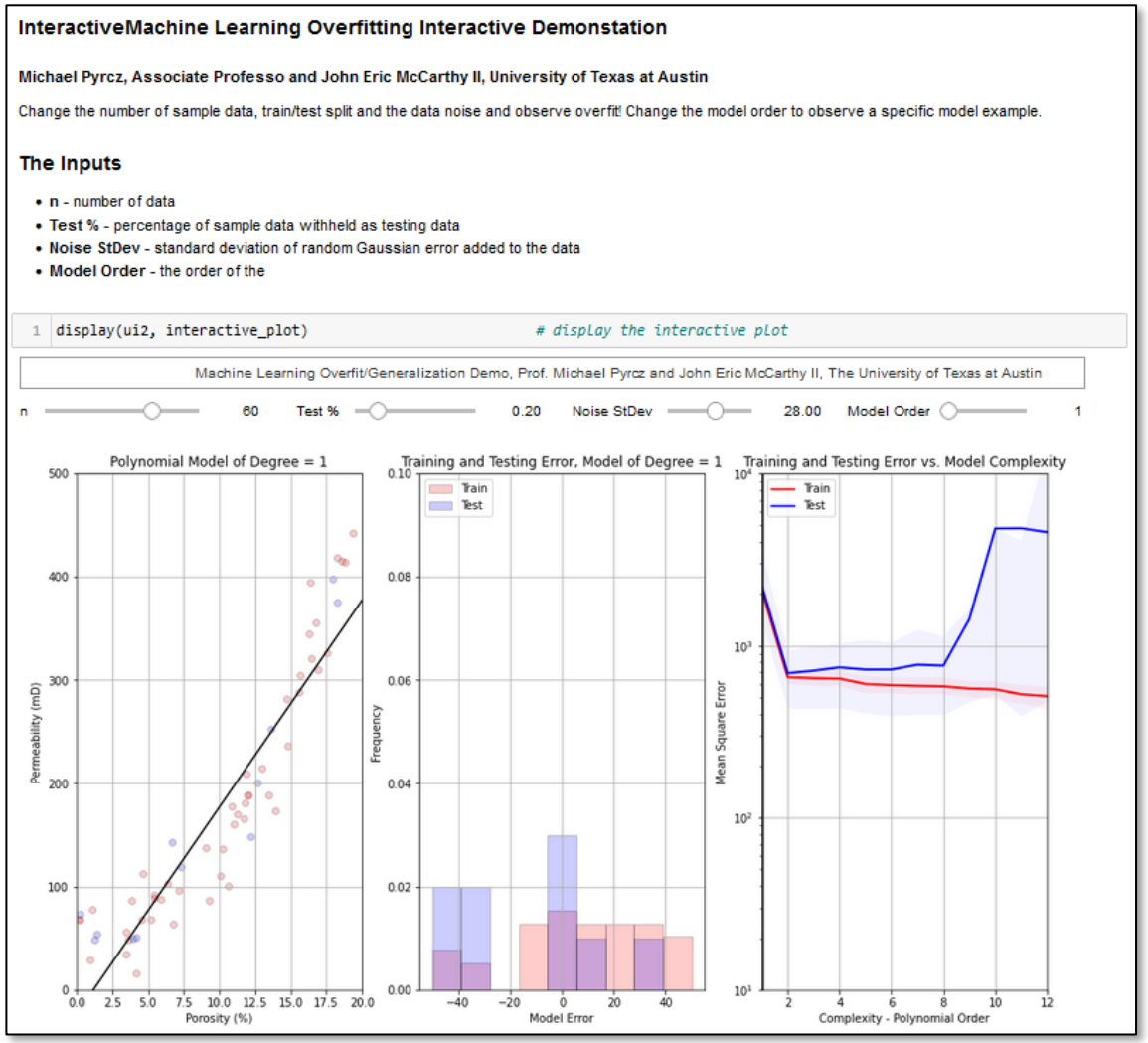
Aspects of an overfit machine learning model,

- More model complexity/flexibility than can be justified with the available data, data accuracy, frequency and coverage
- Model generalizes poorly as it explains “idiosyncrasies” of the data, capturing data noise/error in the data
- 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



Demonstration of overfit with Interactive_Overfit.ipynb.

Model Training and Tuning Example

Let's take a decision tree.

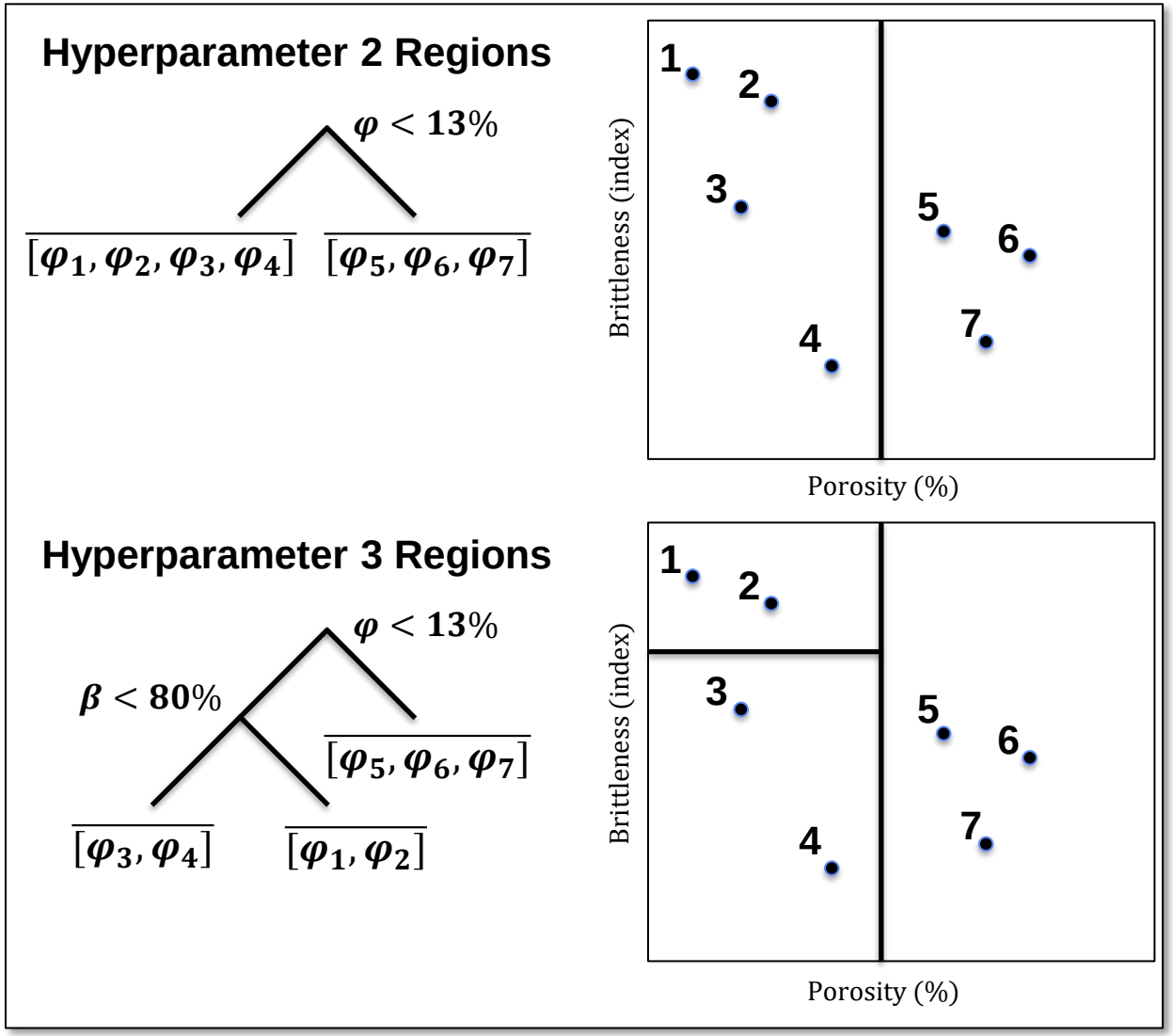
- We will cover decision trees later, don't worry about the details now.

- Split the model space into regions, predict with the average of the training data in each region.

- Train the model parameters over a range of hyperparameter values, the number of regions.

Focus on the model building steps, as we do,

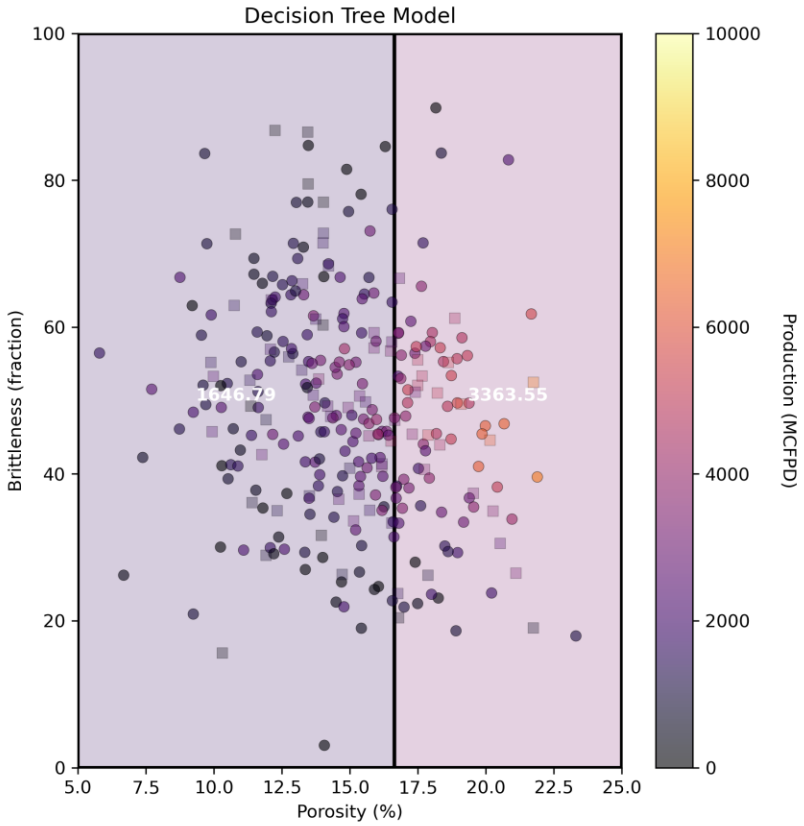
- model tuning by-hand

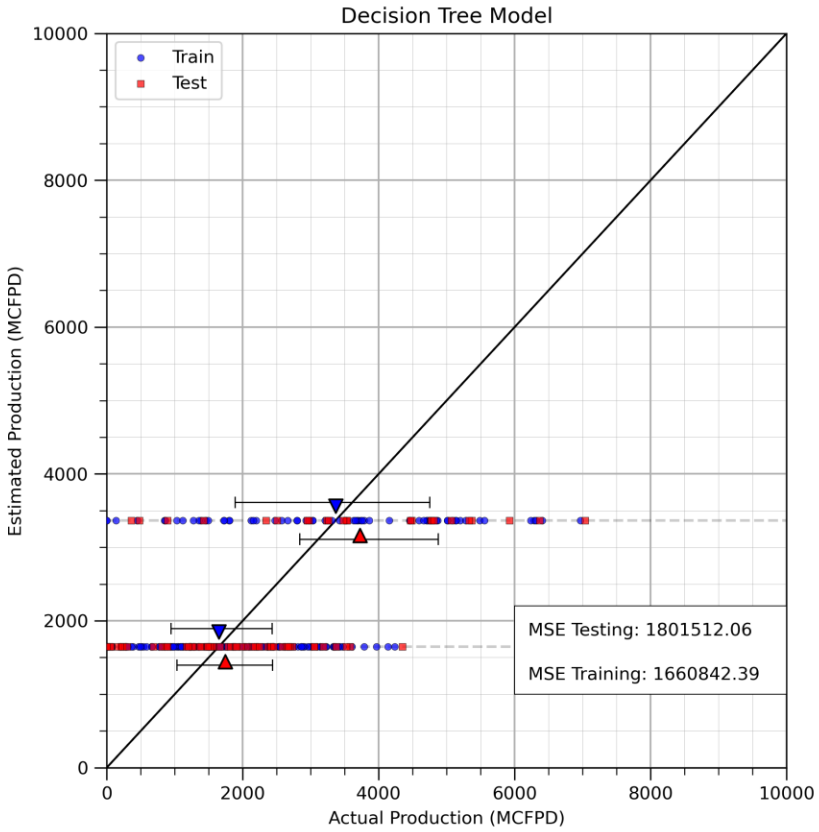


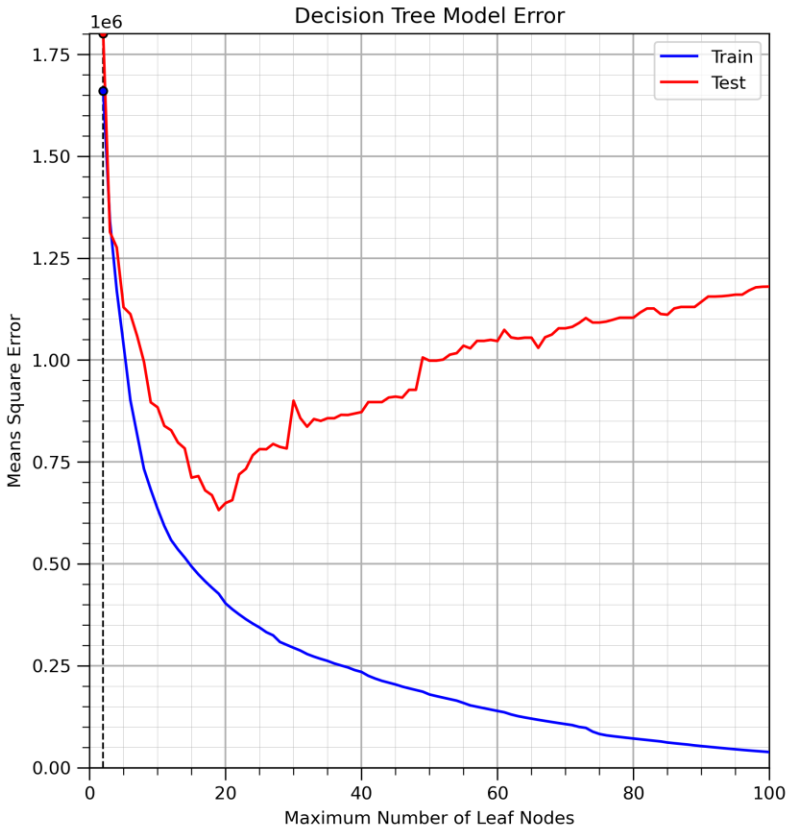
Schematic of decision tree prediction model with 2 hyperparameters.

Model Training and Tuning Example

Hyperparameter = 2 regions (leaf nodes) – very, very underfit model!





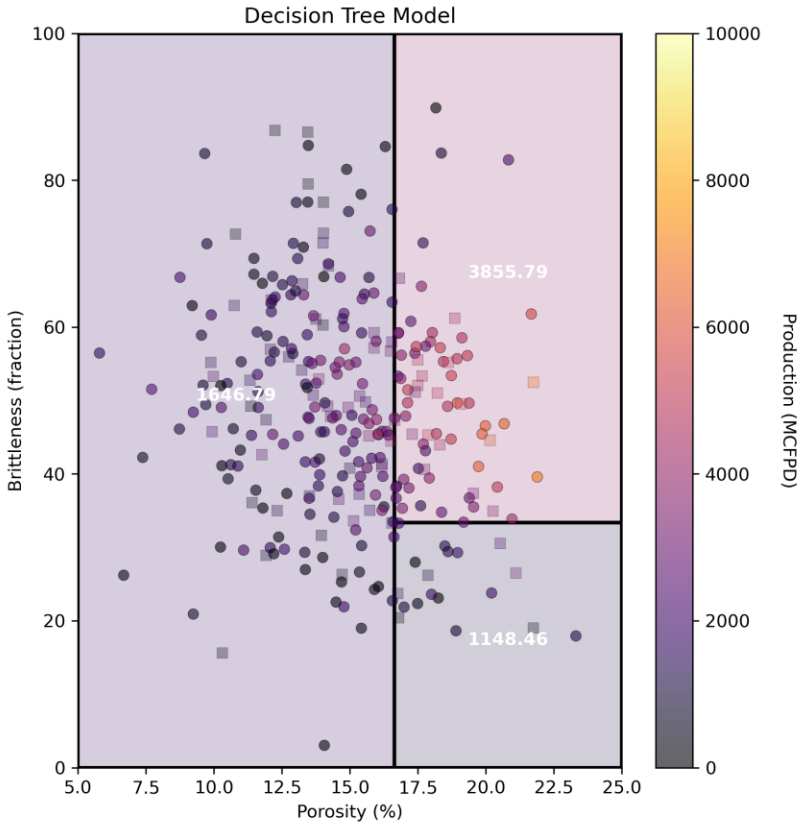


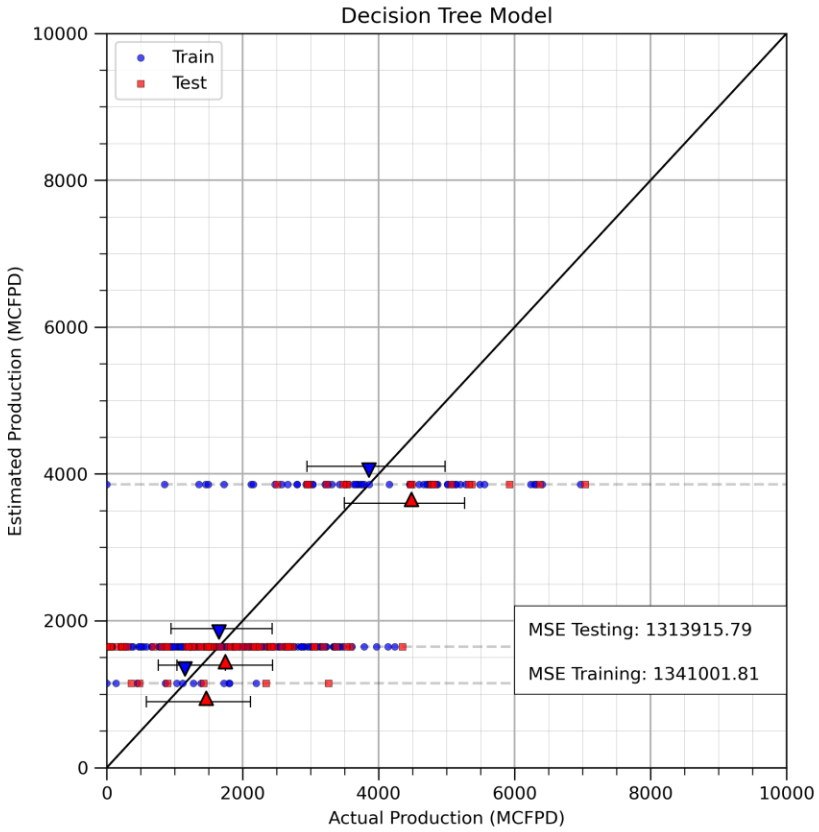
Regression Decision Tree

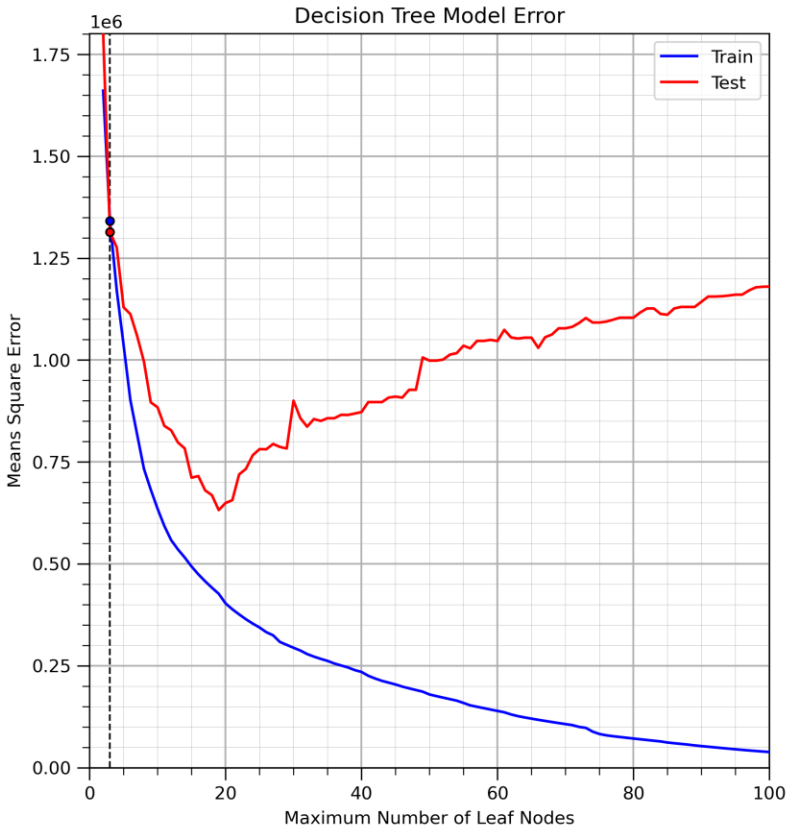
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 3 regions (leaf nodes) – very, very underfit model!





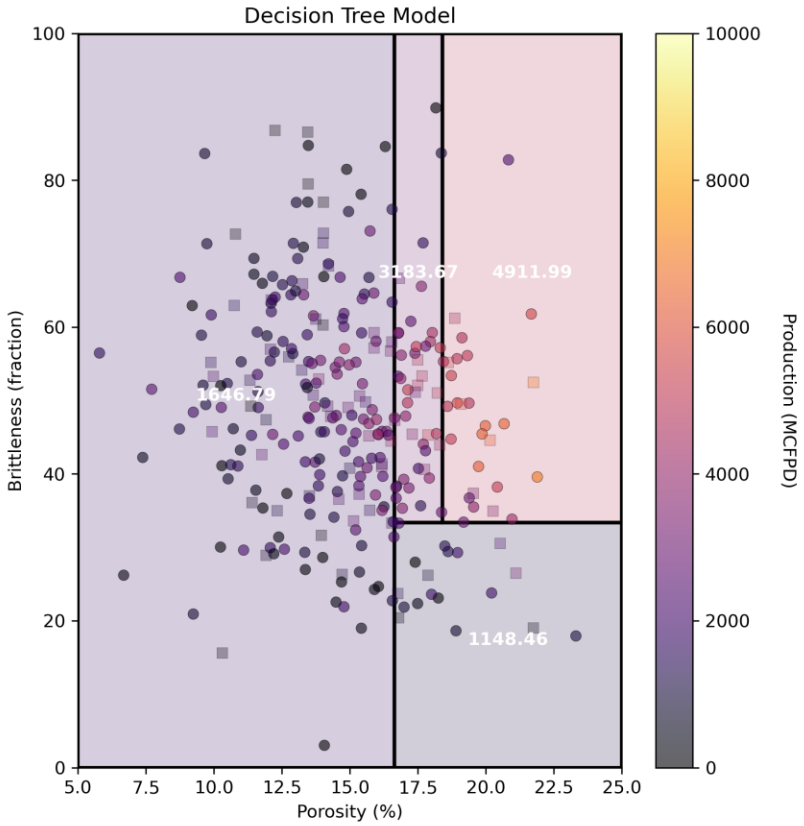


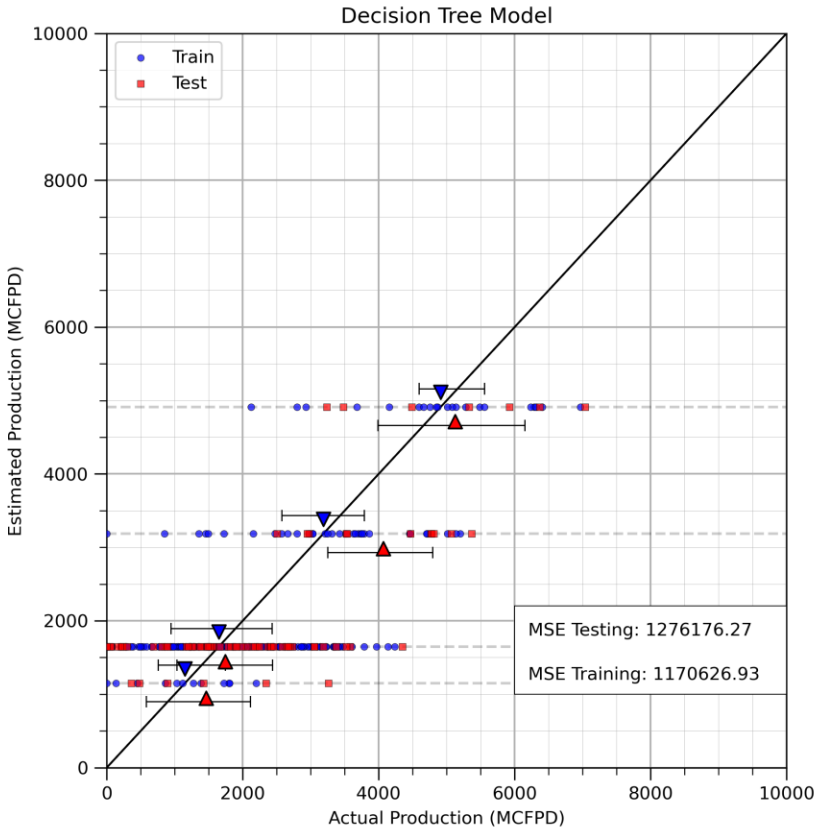
Regression Decision Tree

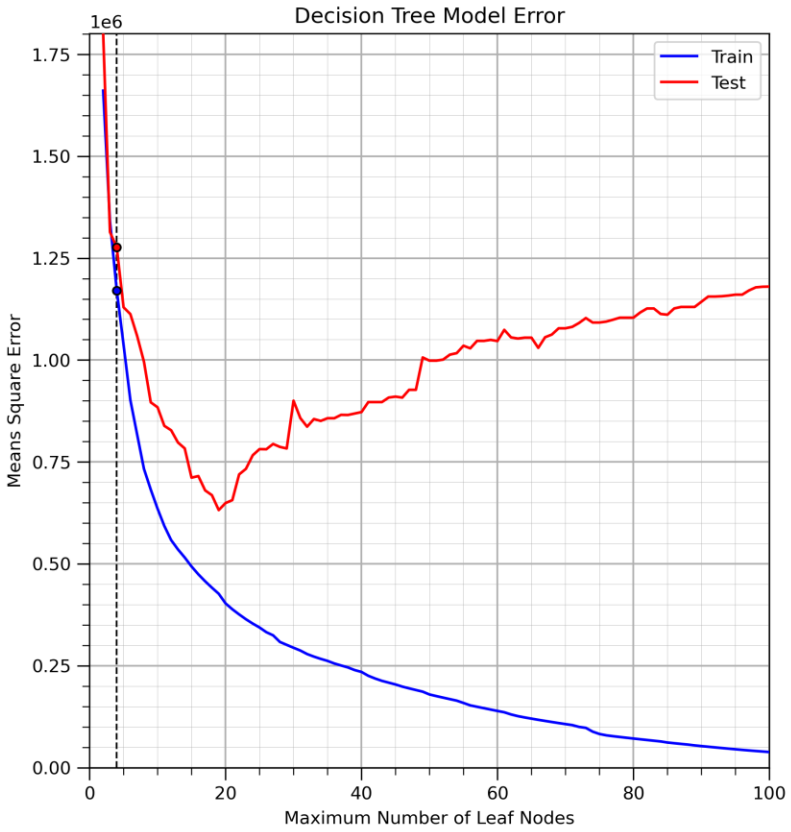
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 4 regions (leaf nodes) – very, very underfit model!





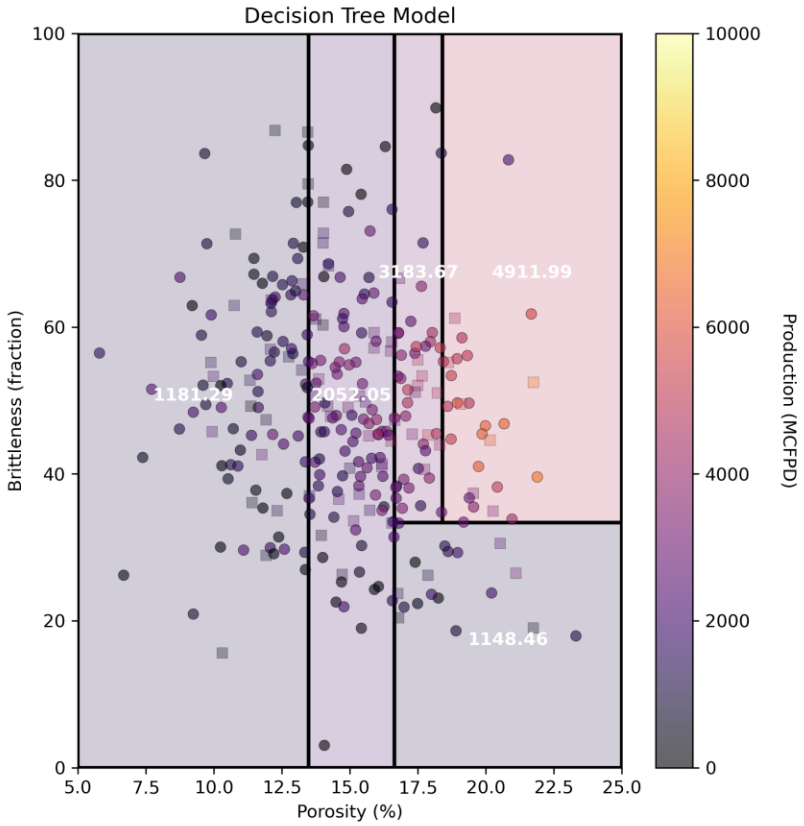


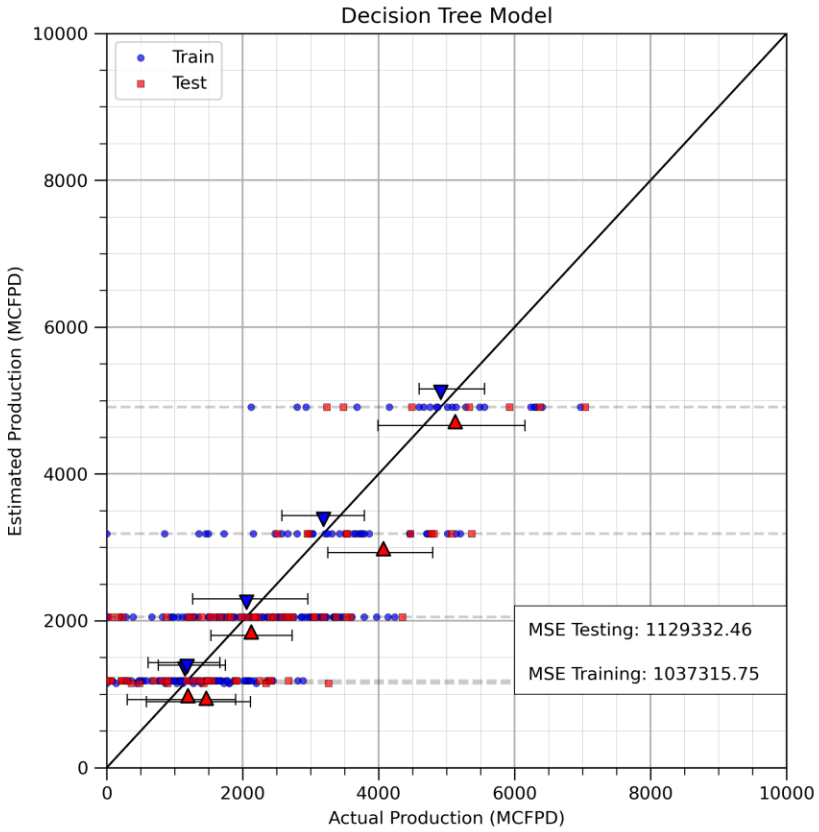
Regression Decision Tree

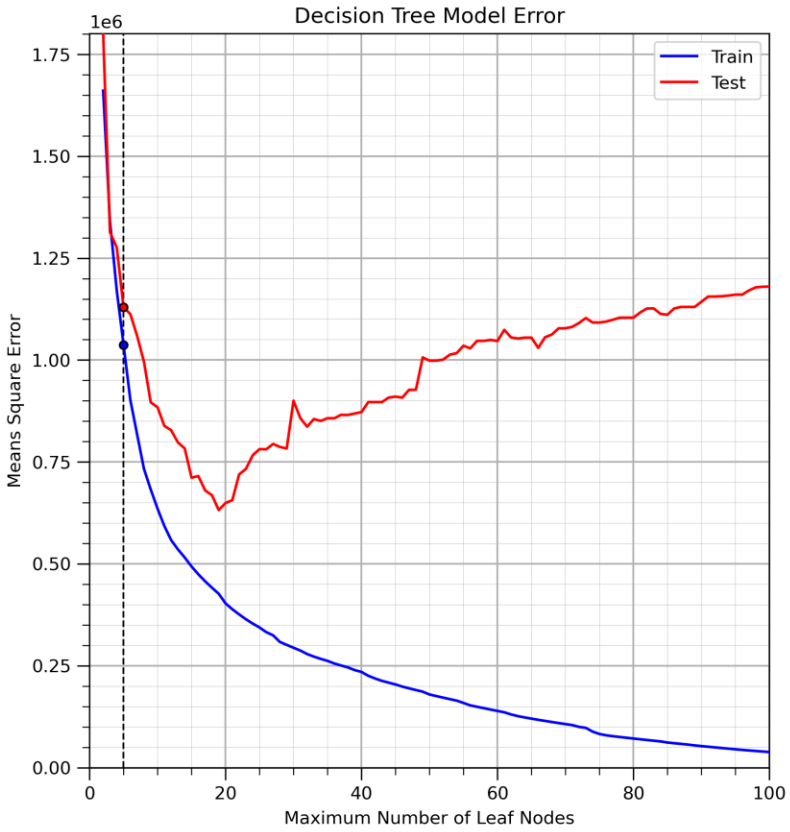
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 5 regions (leaf nodes) – very underfit model!





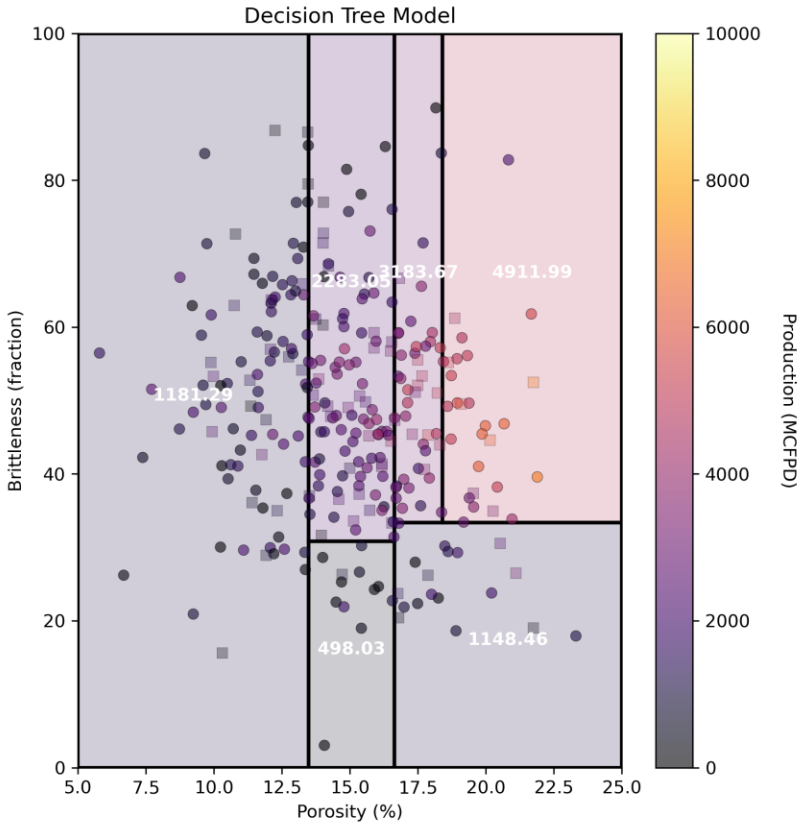


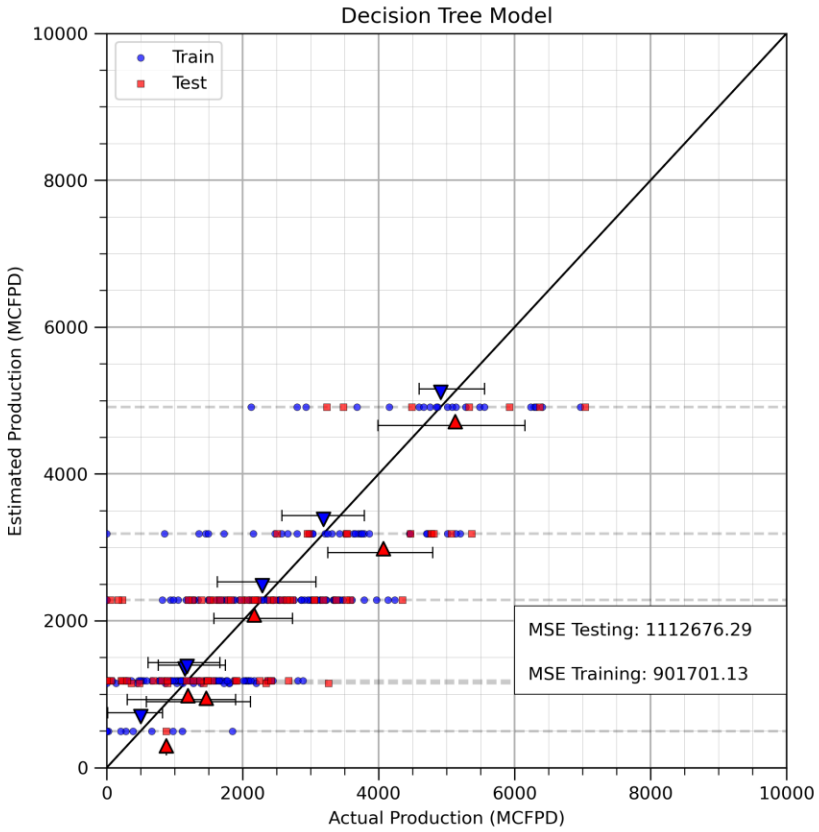
Regression Decision Tree

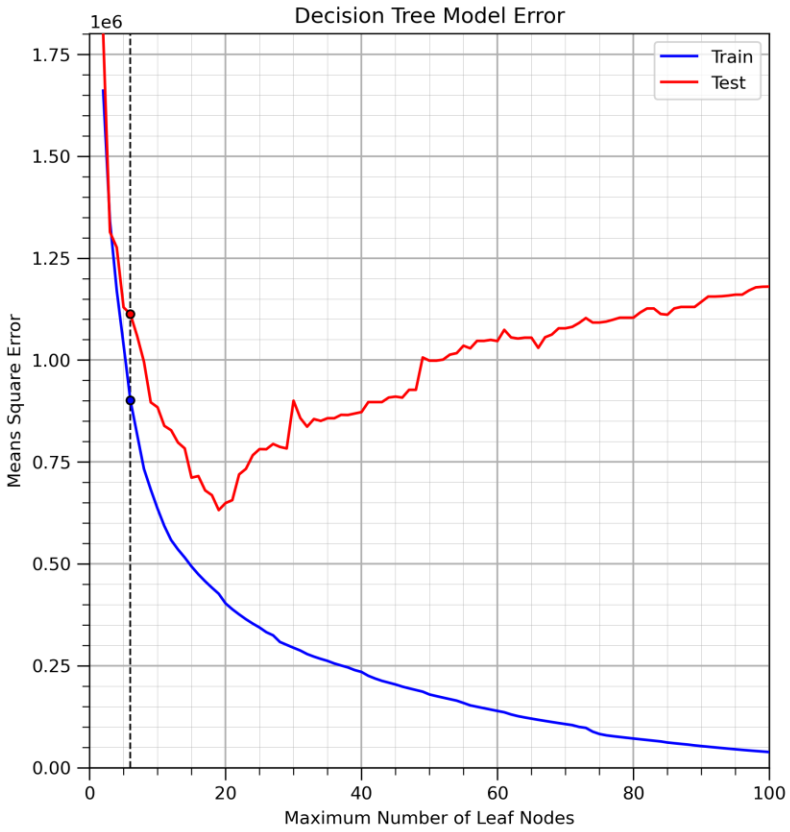
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 6 regions (leaf nodes) – very underfit model!





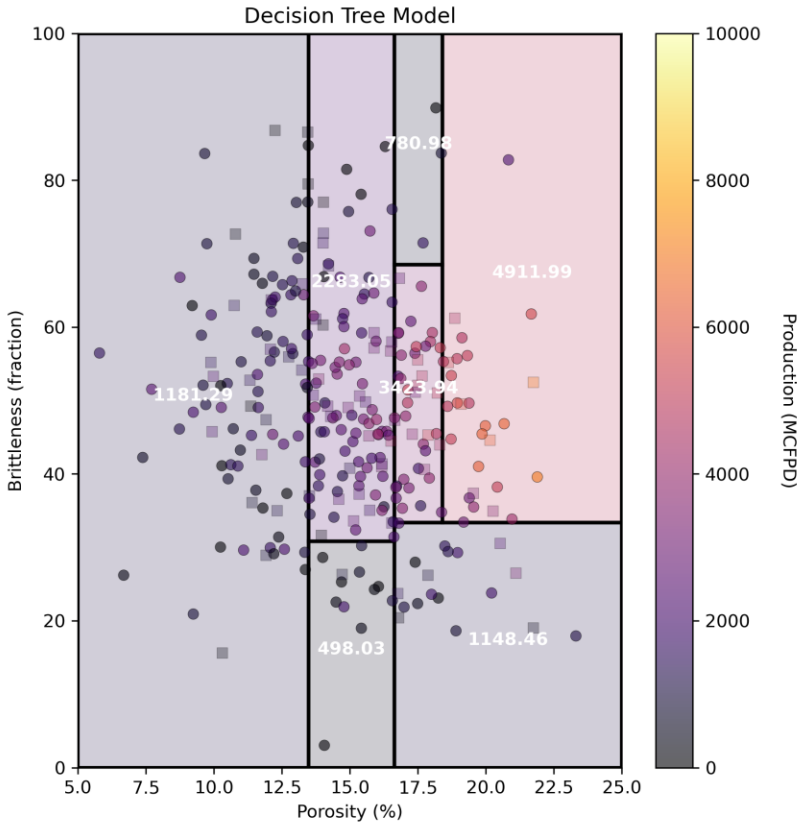


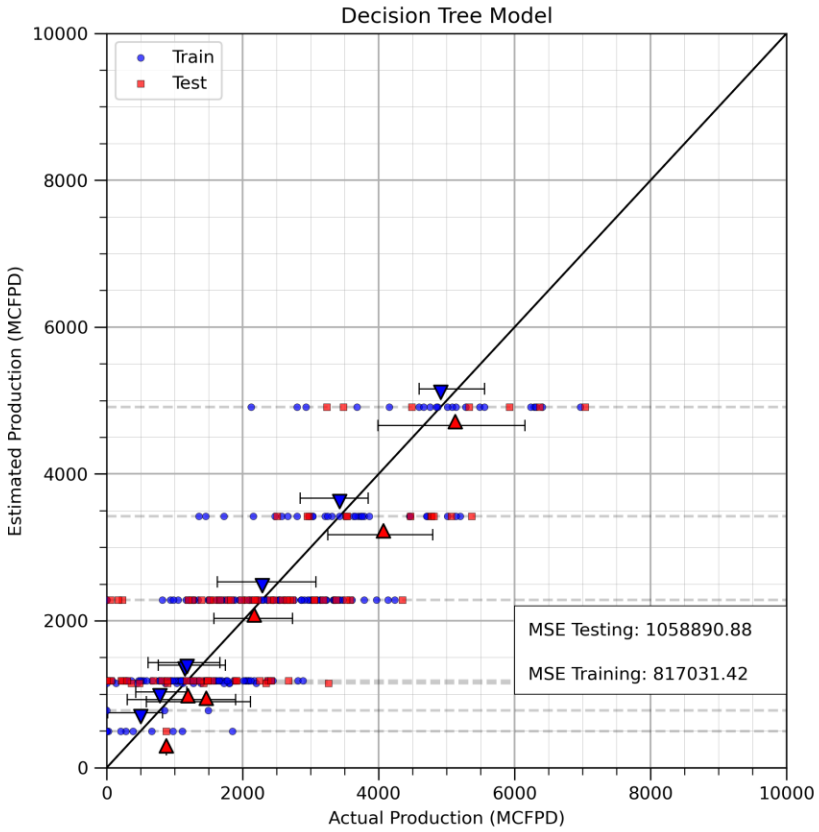
Regression Decision Tree

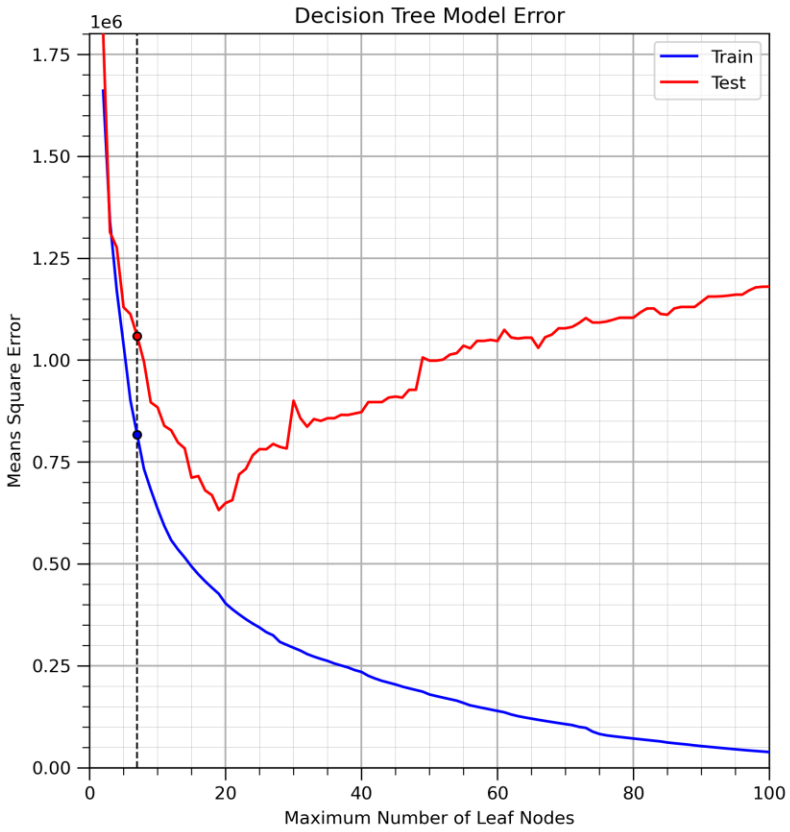
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 7 regions (leaf nodes) – underfit model!





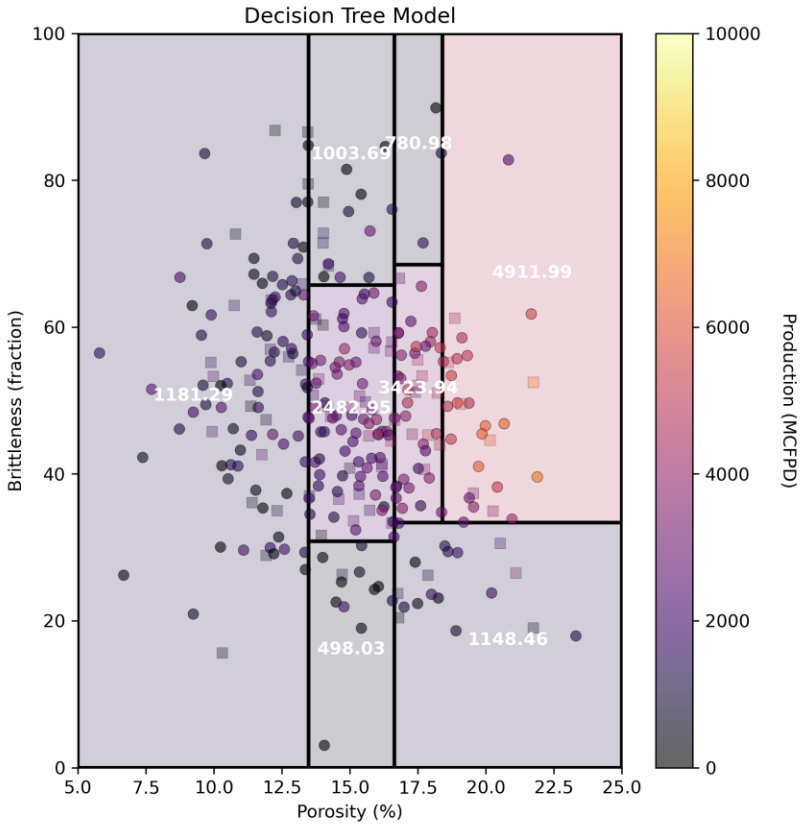


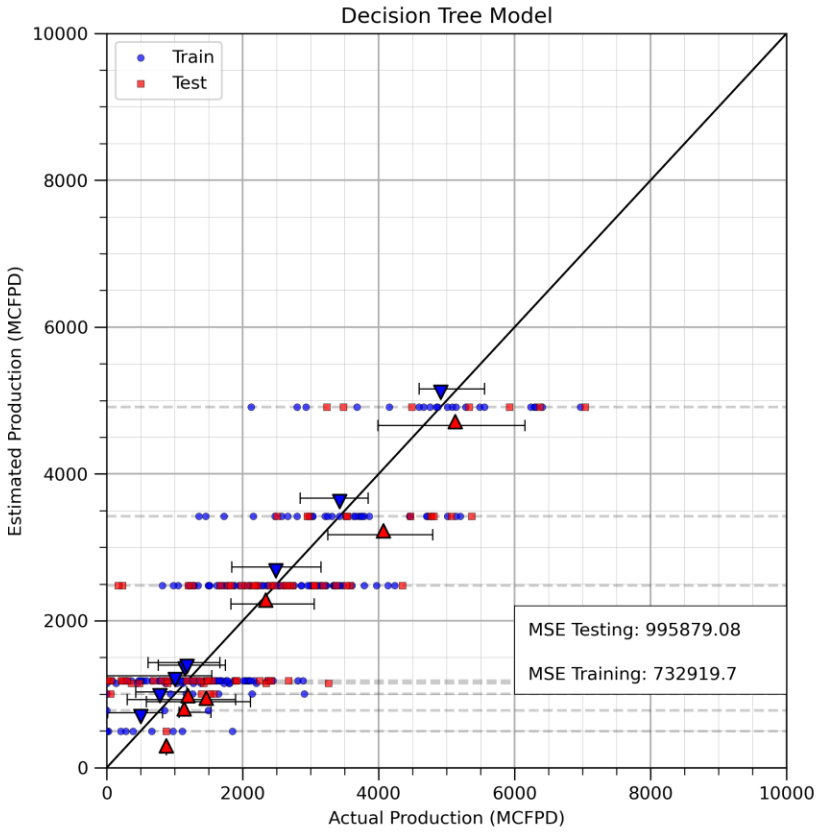
Regression Decision Tree

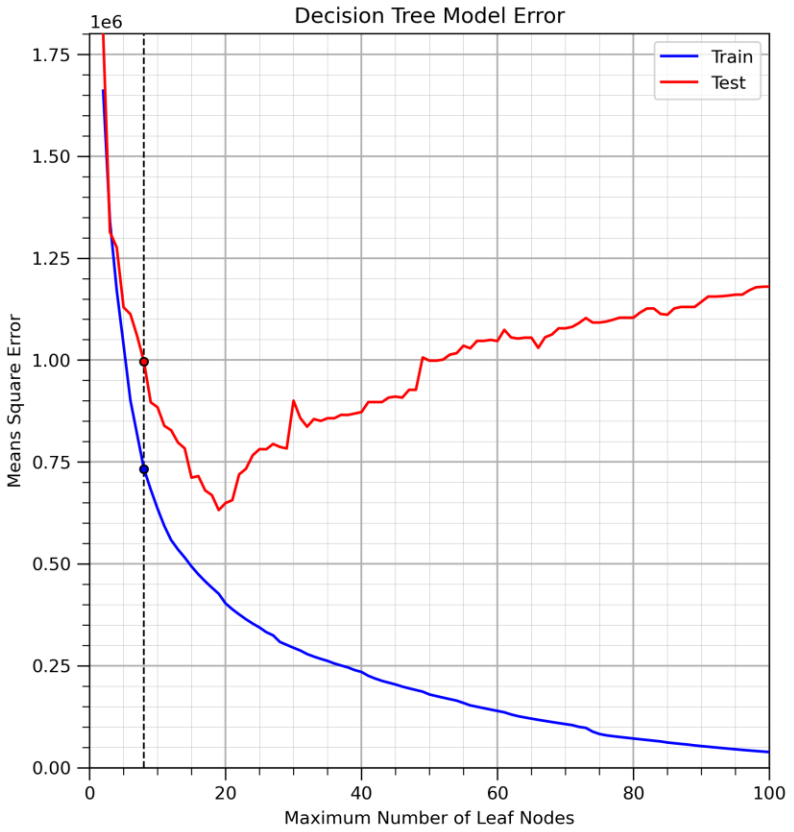
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.
- Debiasing by predicting with weighted average with data weights for each region.

Model Training and Tuning Example

Hyperparameter = 8 regions (leaf nodes) – underfit model





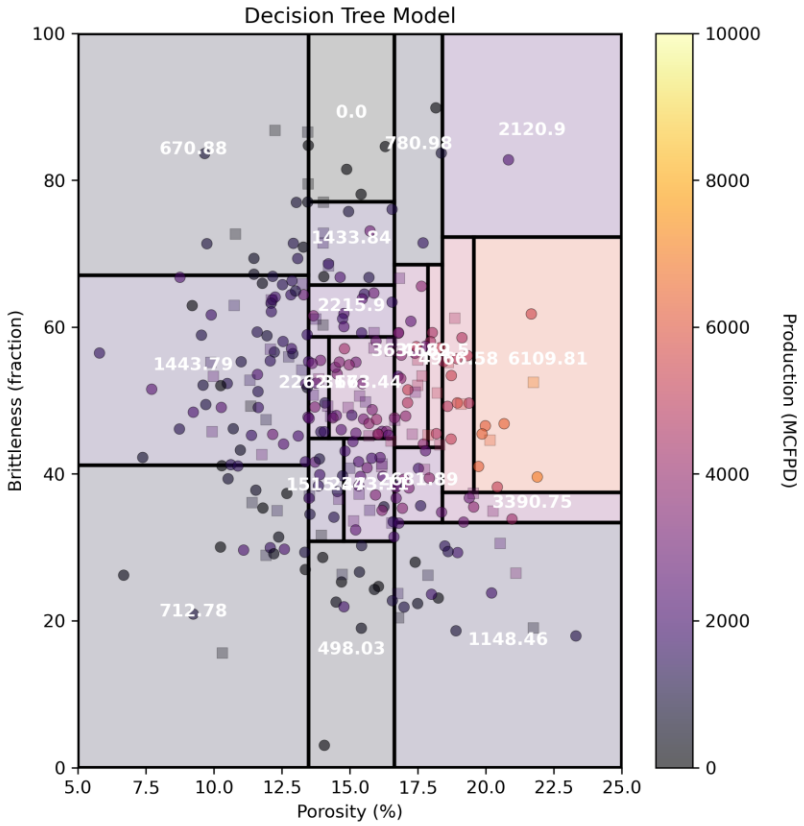


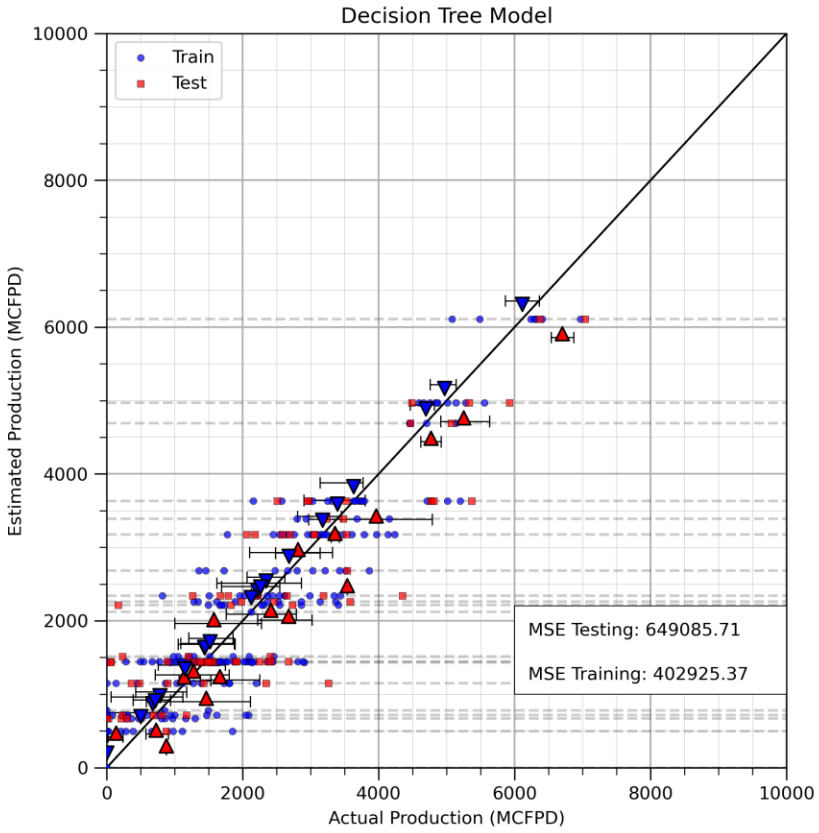
Regression Decision Tree

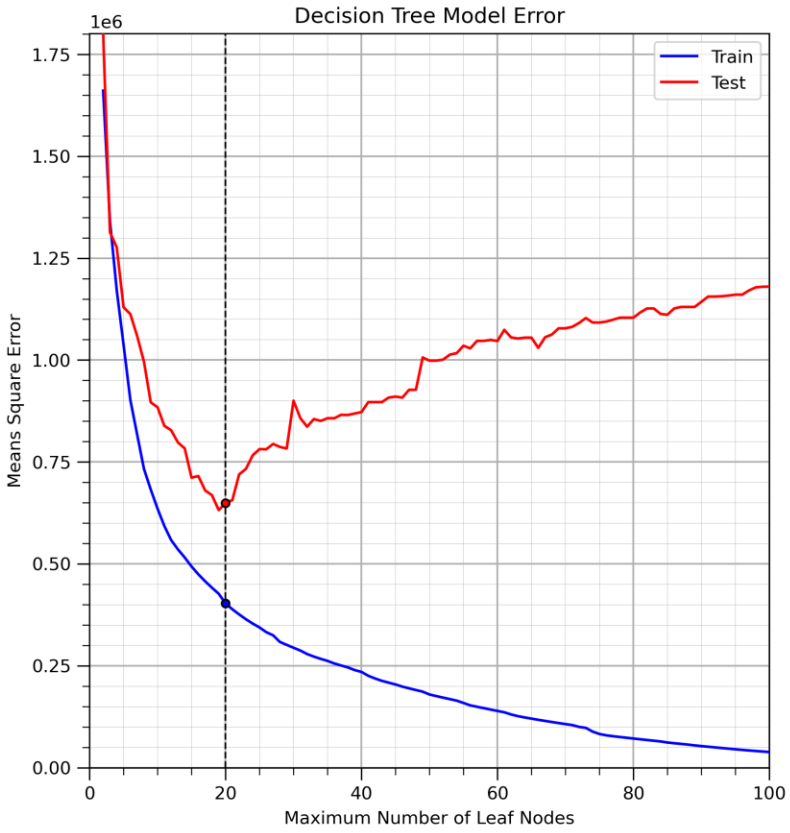
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 20 regions (leaf nodes) – tuned hyperparameter





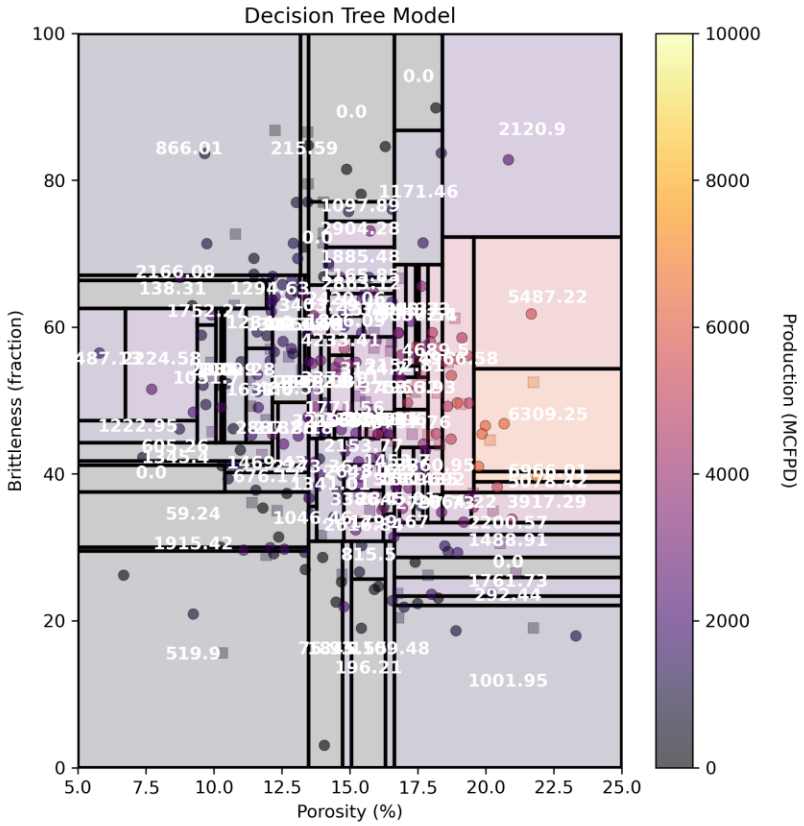


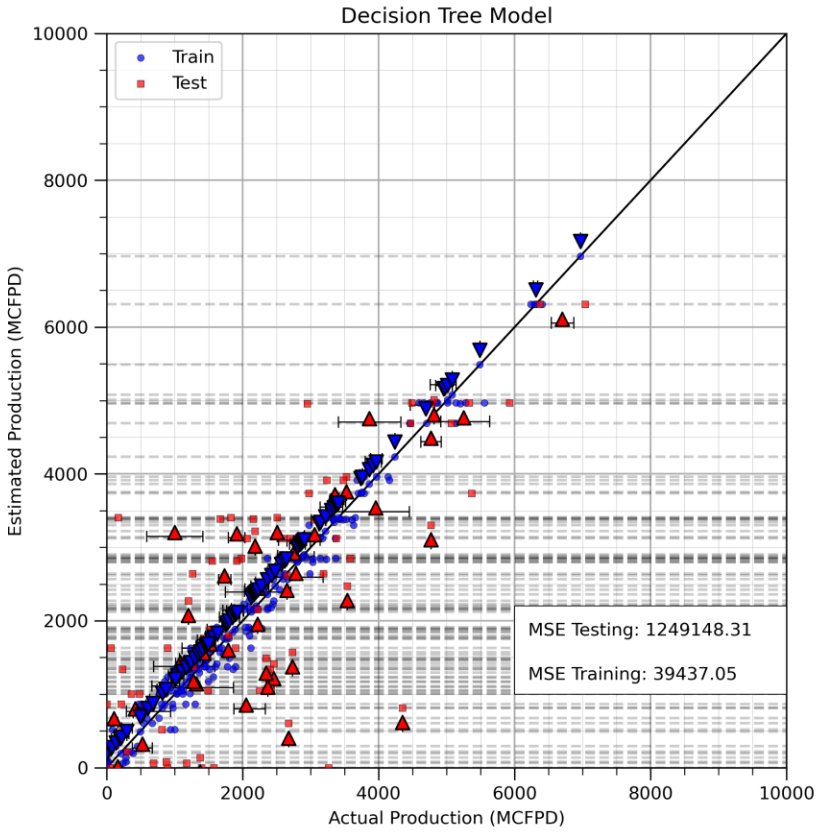
Regression Decision Tree

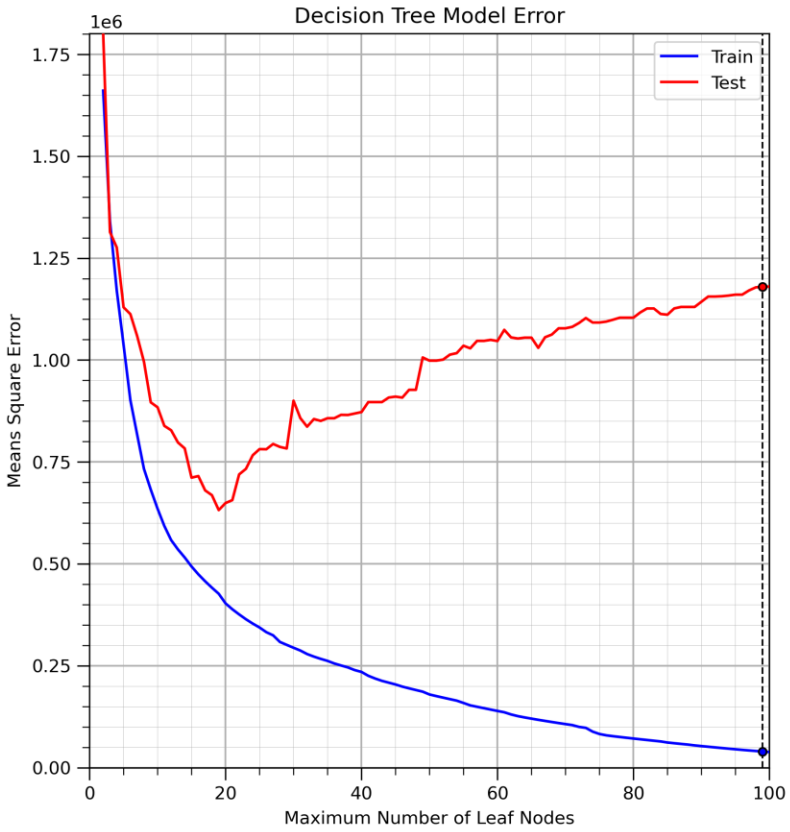
- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Hyperparameter = 99 regions (leaf nodes) – very, very overfit model!







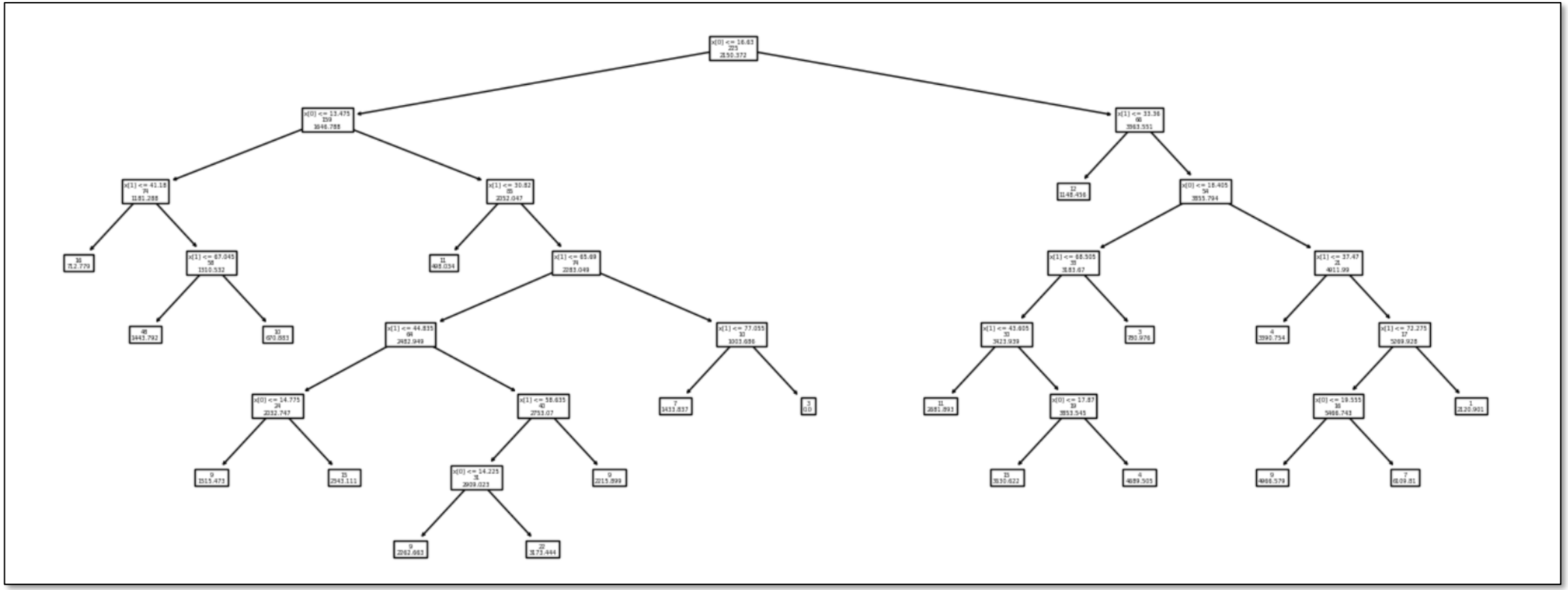
Regression Decision Tree

- Sequential hierarchical splitting of feature region into subregions and predict with the average of training data in each region.

Model Training and Tuning Example

Now retrain the model with our tuned hyperparameter = 20 regions (leaf nodes)

- this is the model that we deplot to predict production from porosity and brittleness.



Tuned and trained decision tree to predict production from porosity and brittleness.

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

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 Subsurface Machine Learning

Lecture 6: Machine Learning

Lecture outline:

- **Examples of Machine Learning**

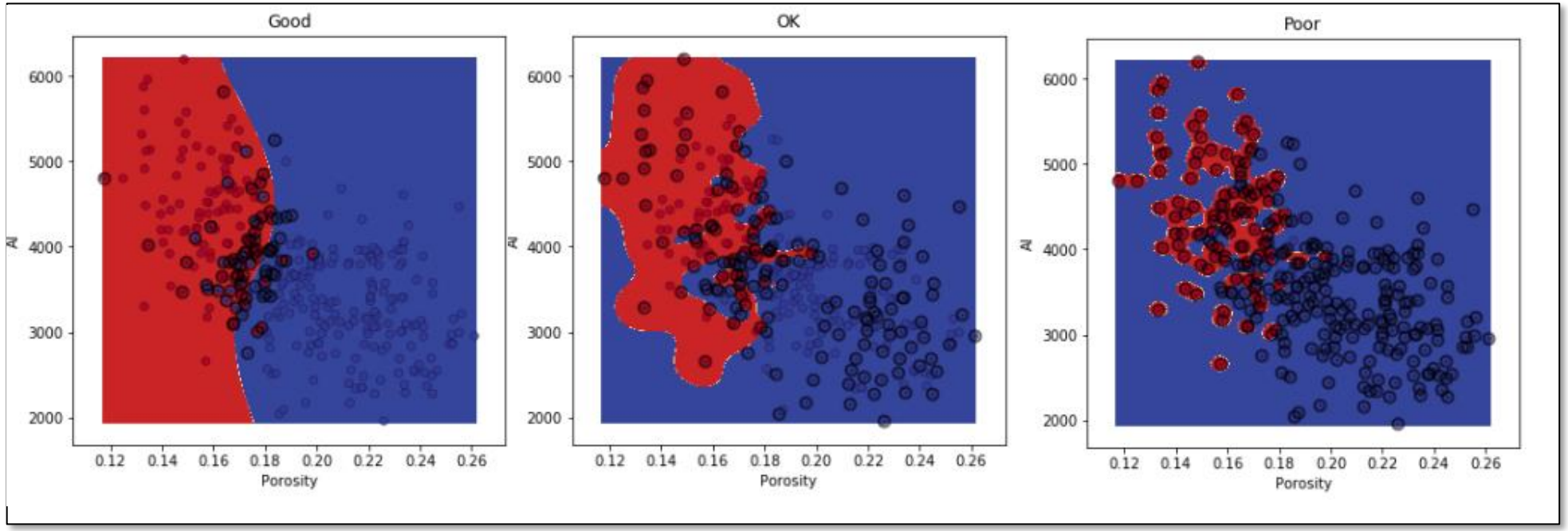
Examples of Machine Learning

Provides a set of example applications 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.

Automatic Group Assignment

Support vector machines for interpolating, extrapolating facies from data.

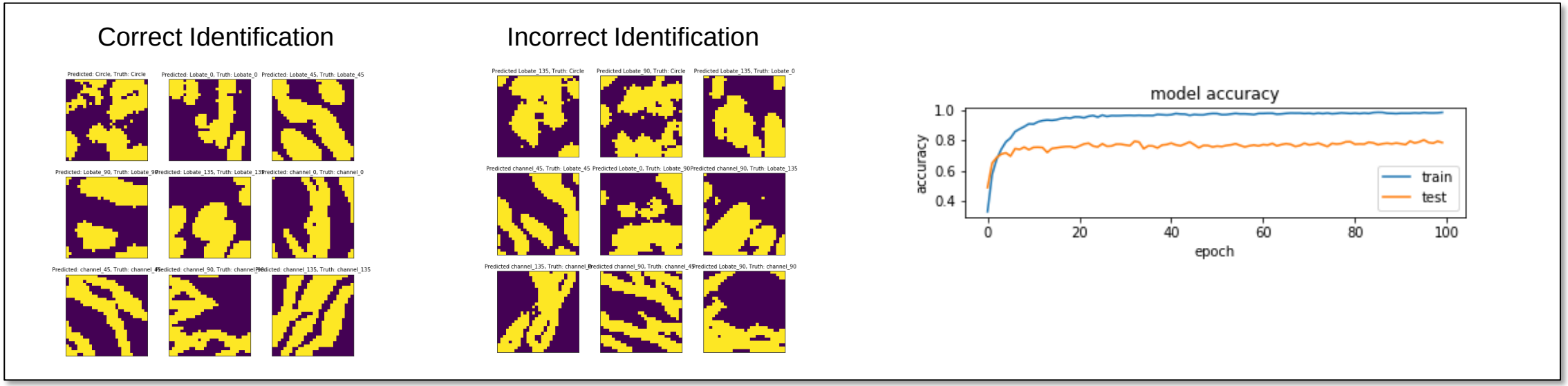


Workflow developed by Wendi Liu, PhD student at The University of Texas at Austin.

- A range of spatial models with a linear model after projection to a high dimensional space
- Works well with classification when the group overlap with each other

Automatic Pattern Detection

Searching through large image databases for reservoir shapes, e.g., deepwater lobes and channnels.

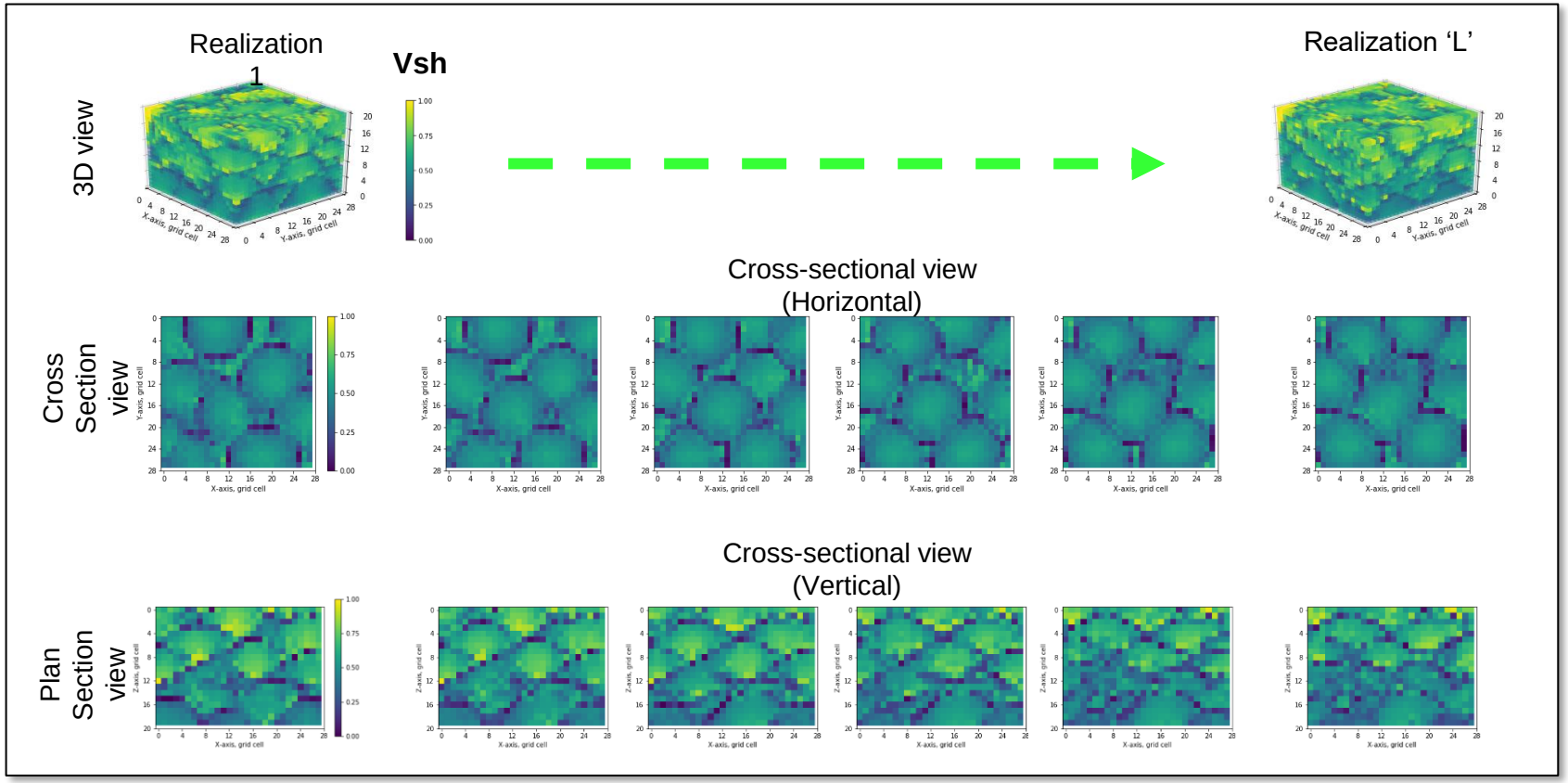


Deepwater object classification by convolutional neural network, by Drs. Honggeun Jo and Javier Santos, while PhD students at The University of Texas at Austin.

Subsurface Uncertainty

Can explore the space of uncertainty along a continuous manifold.

- A latent reservoir manifold based on a single parameter

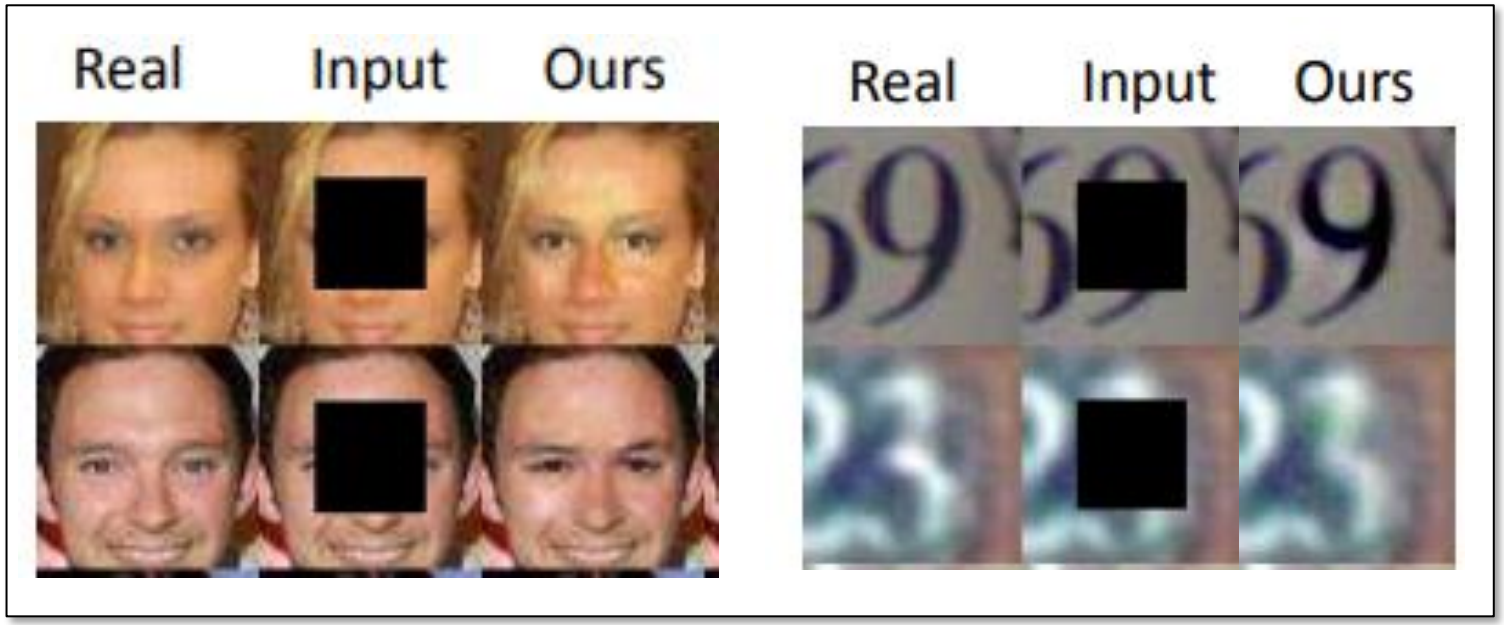


Deepwater uncertainty generative adversarial model developed by Drs. Honggeun Jo and Javier Santos, while PhD students at The University of Texas at Austin.

Imputing Missing Information

Filling In Missing Spatial Information

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



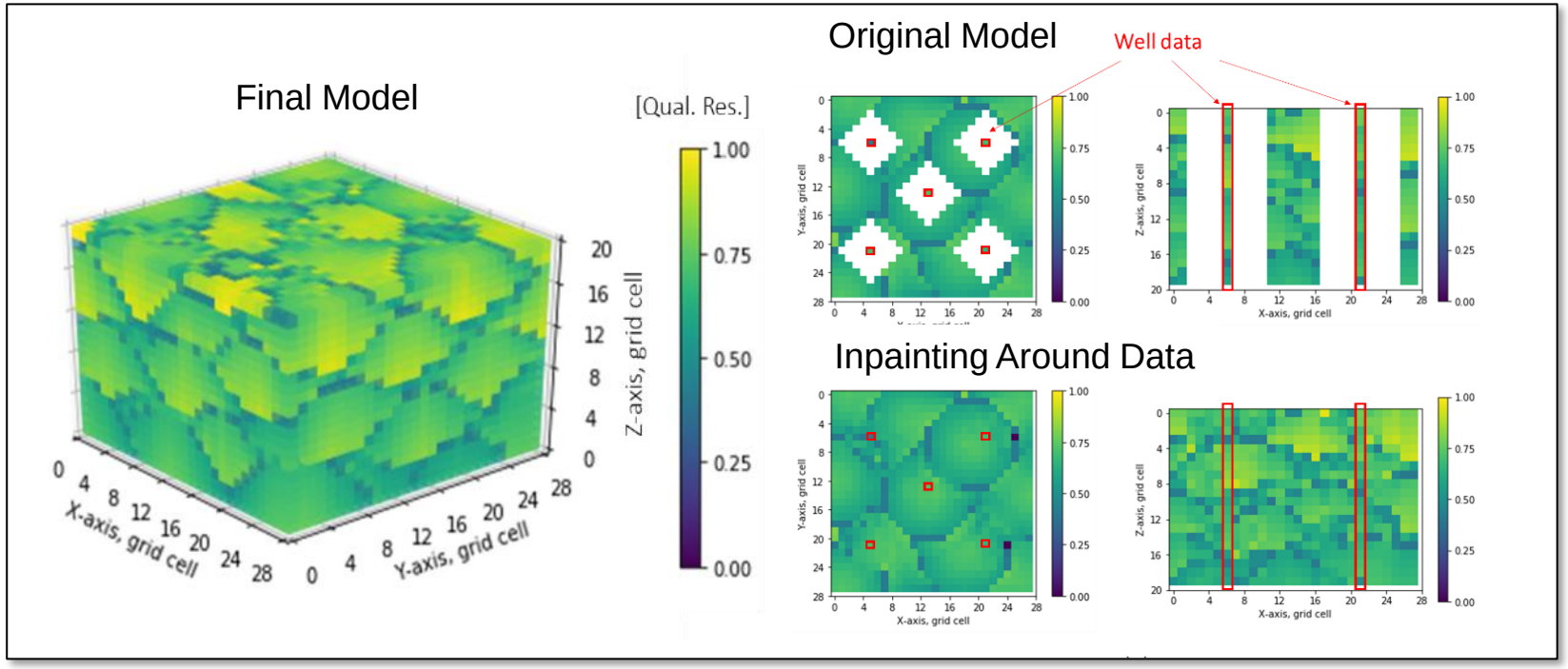
Examples of semantic image inpainting with deep convolutional generative adversarial network (Yeh et al., 2016).

Conditional Subsurface Models

Conditioning subsurface models 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 Subsurface Machine Learning

Lecture 6: Machine Learning

Lecture outline:

- **Energy Machine Learning**

The 4th Paradigm

Welcome to the 4th Paradigm of Scientific Discover!

1st Paradigm Empirical Science

Experiments

- 430 BC Empedocles proved air has substance
- 230 BC Eratosthenes measure Earth's diameter

2nd Paradigm Theoretical Science

Models/Laws

- 1011 AD al-Haytham Book of Optics
- 1687 AD Newton Principia
- 1922 Friedmann Cosmic Expansion

3rd Paradigm Computational Science Simulation

Numerical Simulation

- 1942 Manhattan Project
- 1980 – Global Forecast System (GFS)
- 1989 Tetzlaff and Harbaugh SEDSIM

4th Paradigm Data-driven Science

Learning from Data

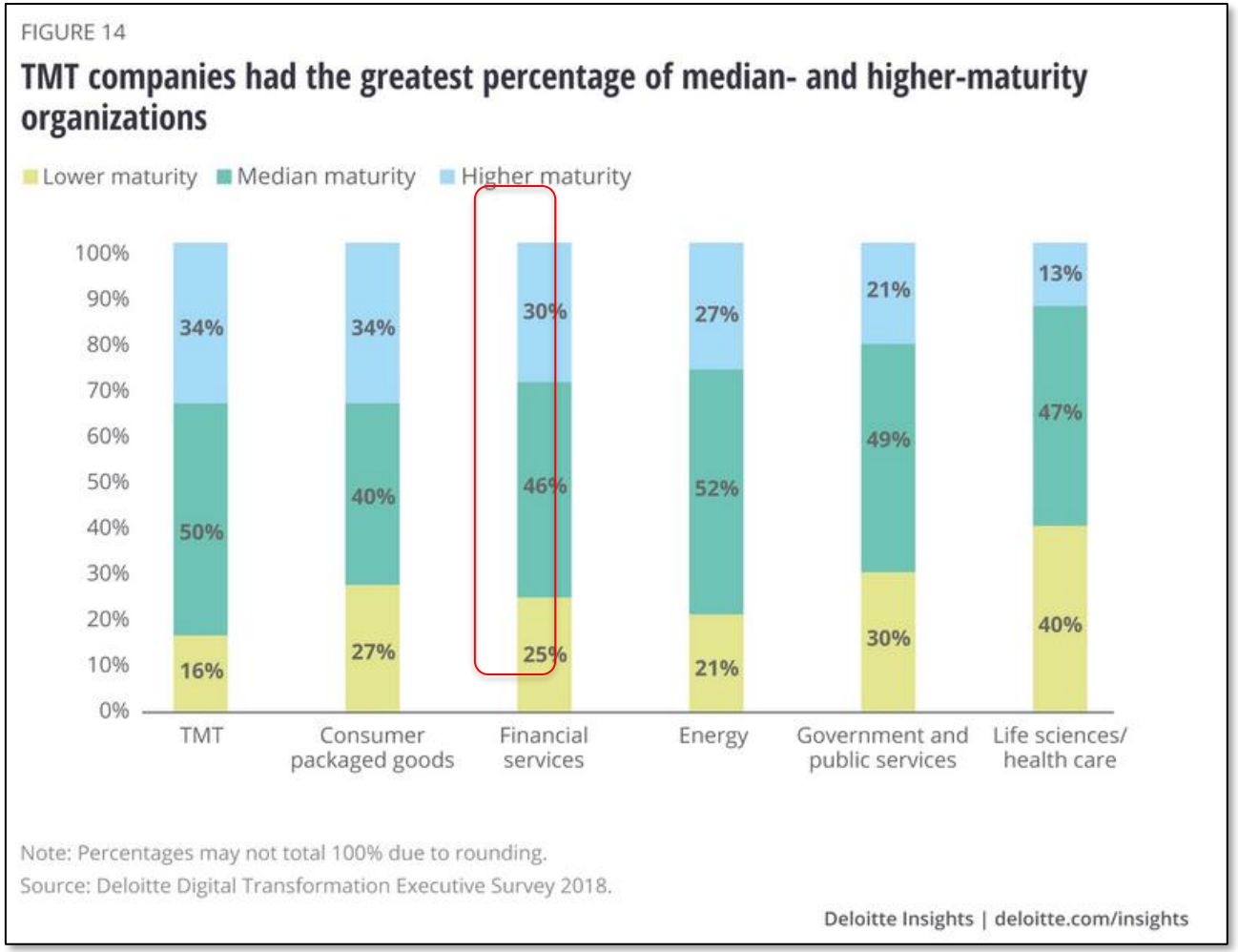
- 2009 Hey et al. Data-Intensive Book
- 2015 AlphaGo beats a professional Go player

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

Energy Data Science

We Are Not Alone

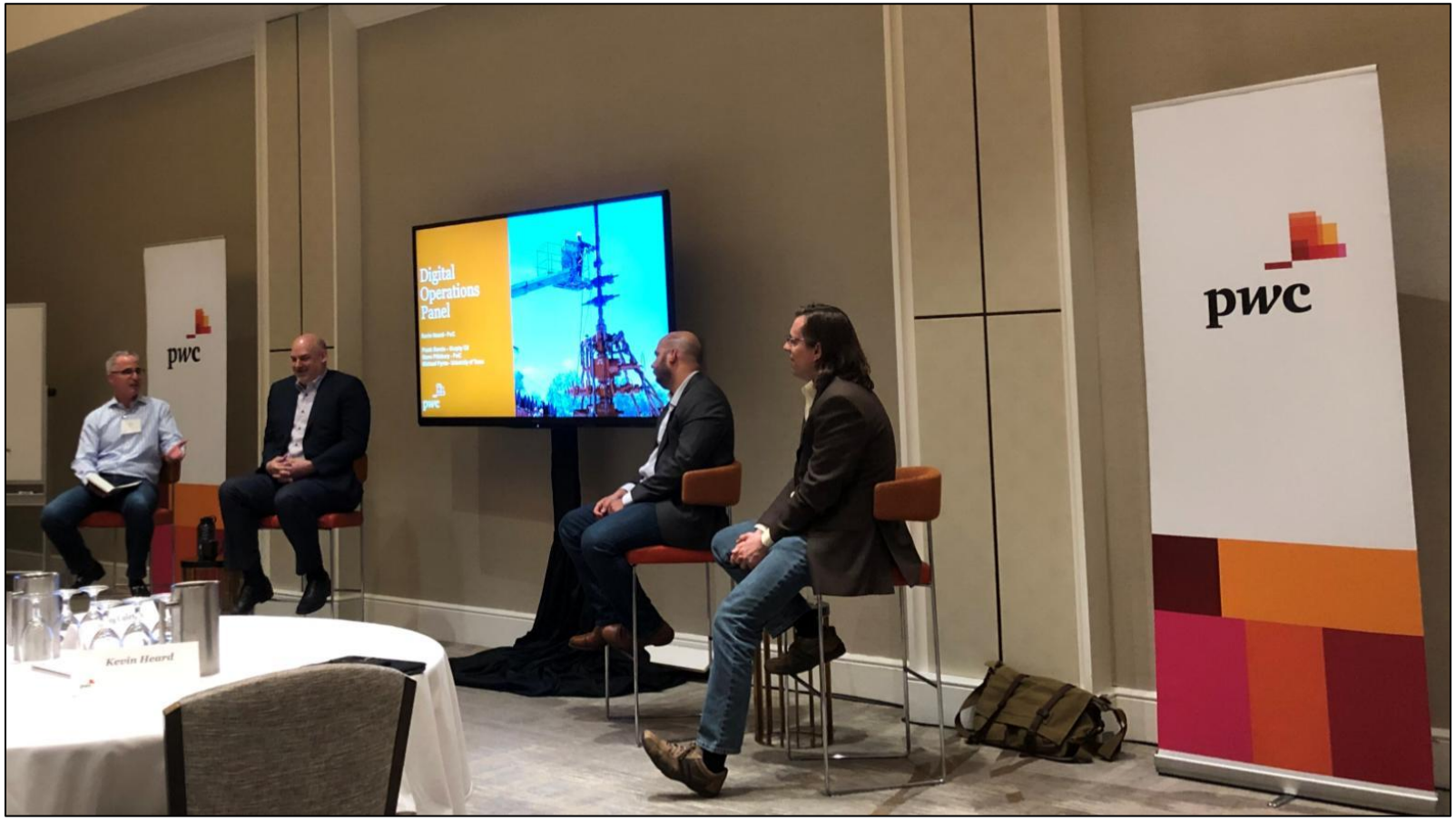
- Digital transformations are underway in all sectors of our economy
- Every energy company that I visit is working on this right now



Digital transformation study by Deloitte, 2019.

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



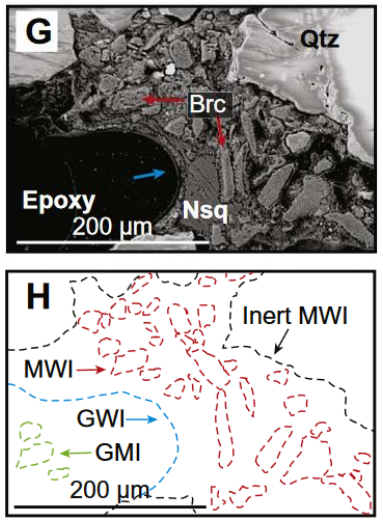
Digital transformation PricewaterhouseCoopers (PwC) panel April 9th, 2019.

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).

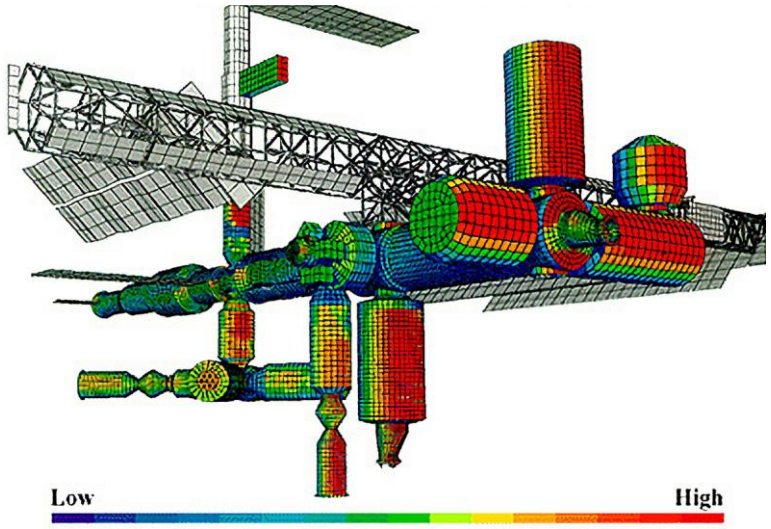
2nd Paradigm Theoretical Science

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

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 Preparation

Data-driven Science Needs Data, Data Preparation Remains Essential

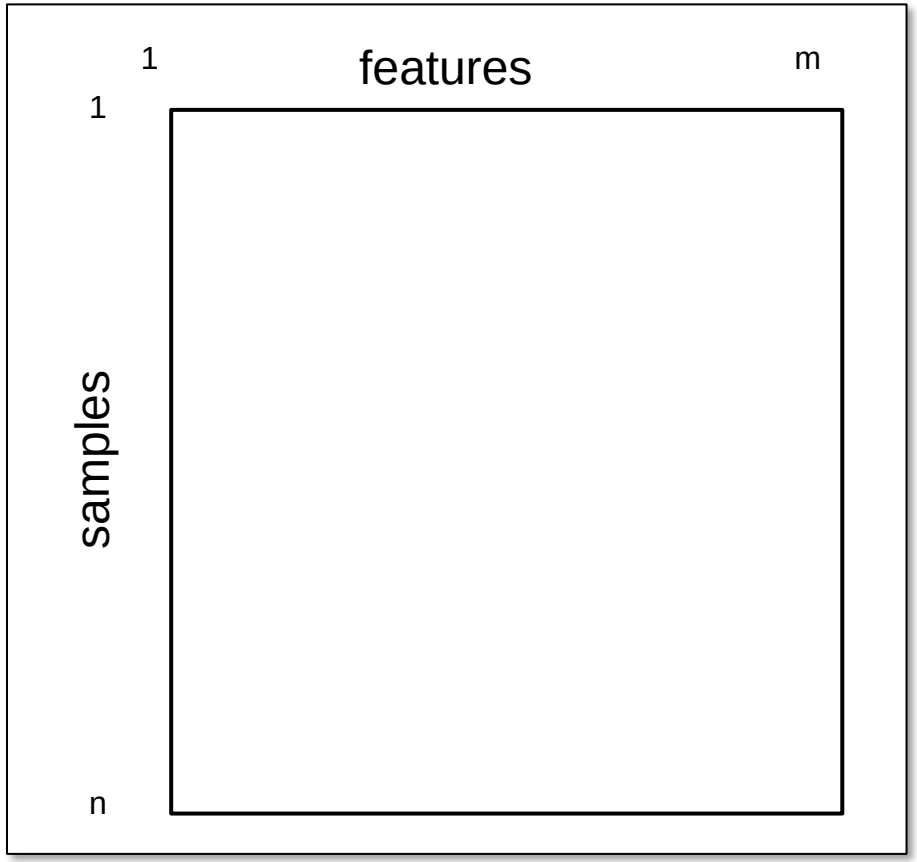
>80% of any subsurface study is data preparation and interpretation

We continue to face a challenge with data:

- **data curation** - format standards, version control, storage, transmission, security and documentation
- **large volume to manage** - visualization, availability and data mining and exploration
- **large volumes of metadata** - lack of platforms, standards and formats
- **engineering integration** - variety of data, scale, interpretation and uncertainty

Clean databases are prerequisite to all data analytics and machine learning,

- Must start with this foundation, Garbage in, garbage out

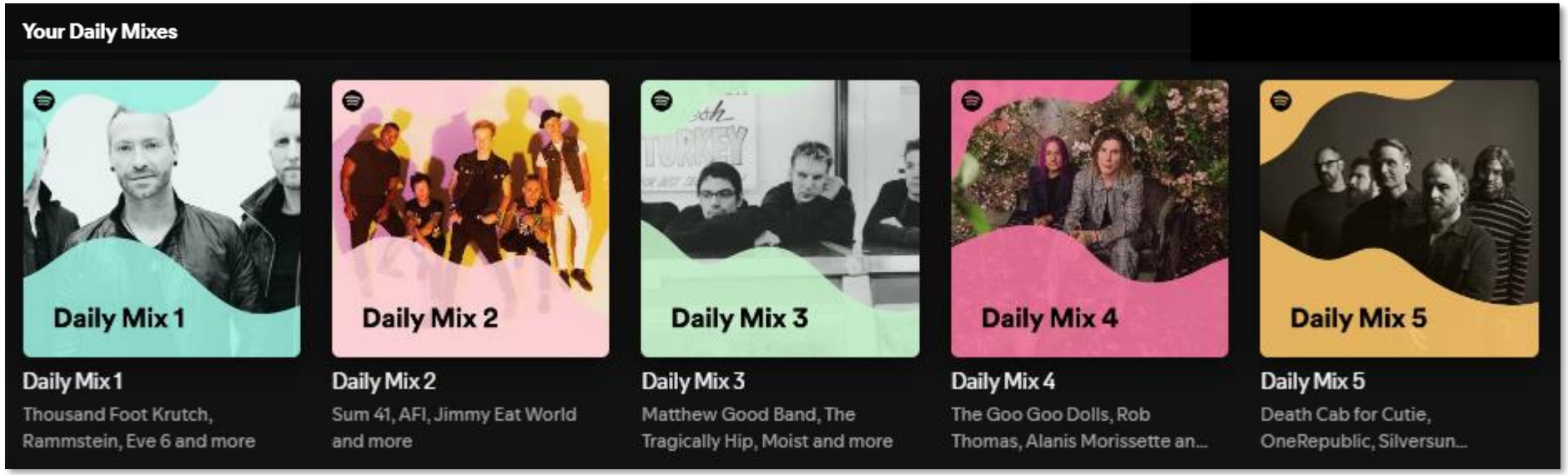


The common data table, samples and features.

Energy is Unique

Subsurface is Different and Needs 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



My Spotify recommender system from my account summer, 2024.

We need to develop novel subsurface data analytics and machine learning solutions that integrate of geoscience and engineering → most data science tools is not ready off the shelf!

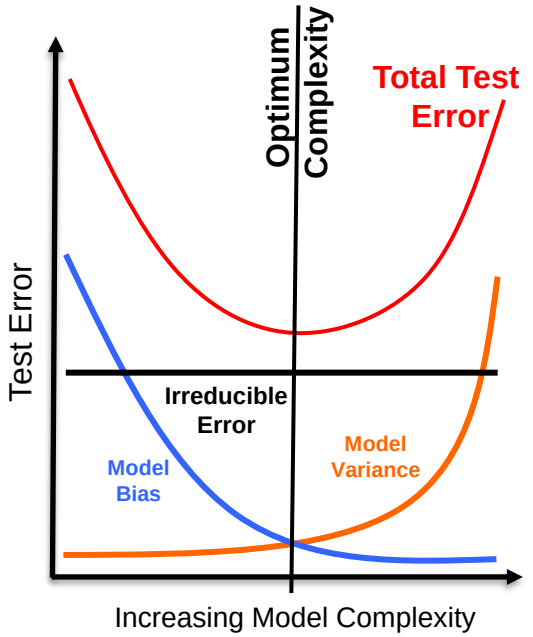
Don't Jump to Complexity

The Expected Test Mean Square Error may be calculated as:

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

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’

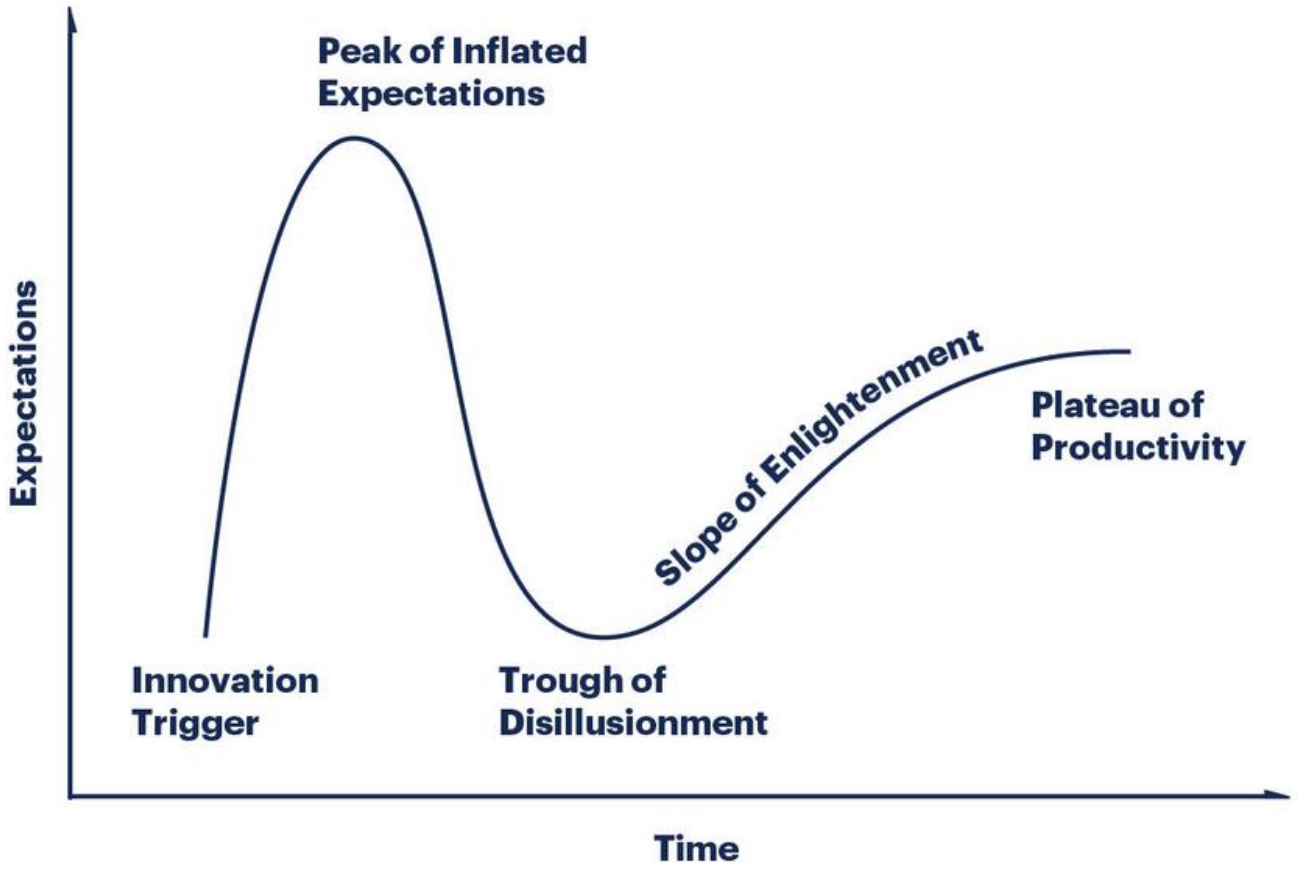
Machine learning-based logistic classifier to identify wolf or dog, image and example from Ribeiro et al. (2016) https://arxiv.org/pdf/1602.04938.pdf.

‘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.

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!



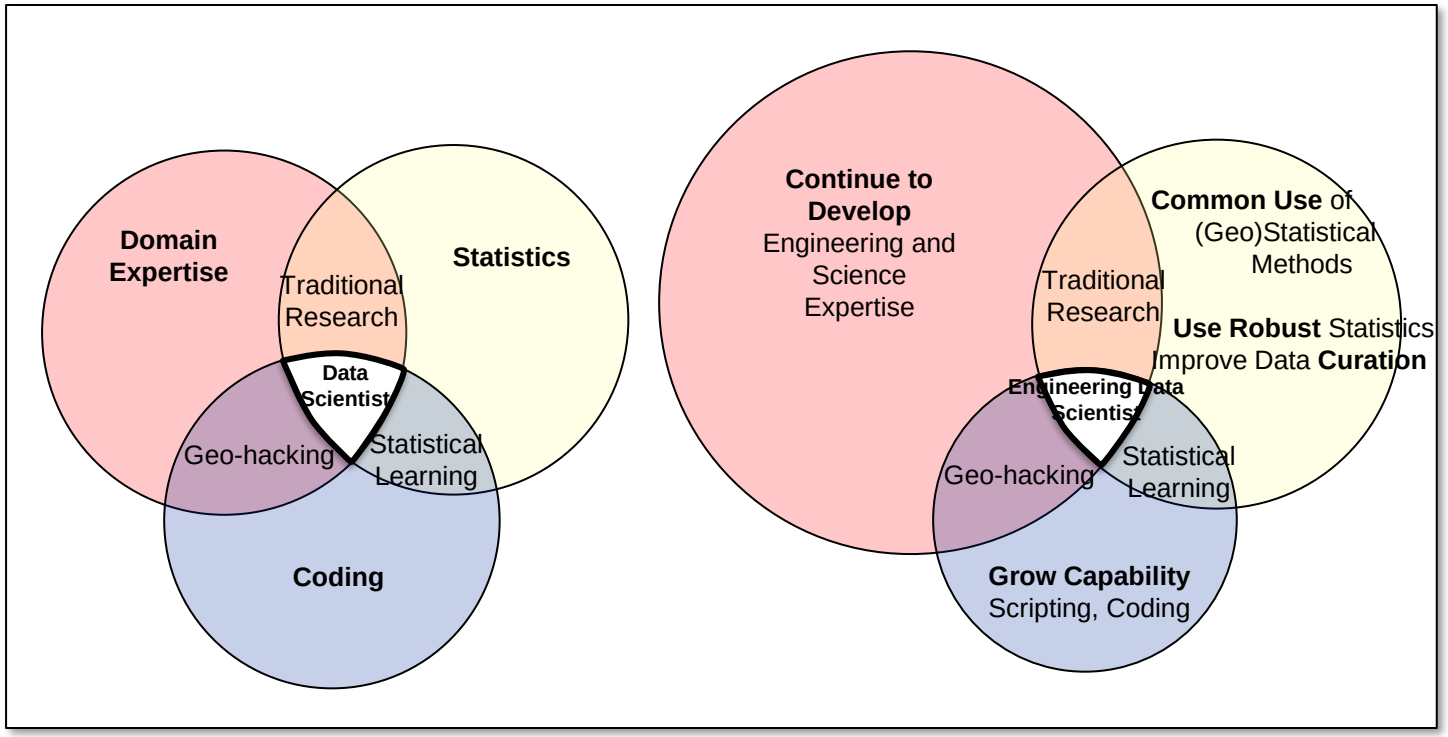
Technology hype cycle from time of discovery. Image from and more about the hype cycle at https://www.gartner.com/en/documents/3887767, hype cycle image from https://www.gartner.com/en/research/methodologies/gartner-hype-cycle.

A Possible Career Path

My Suggestions for You to Add / Improve Your Data Science

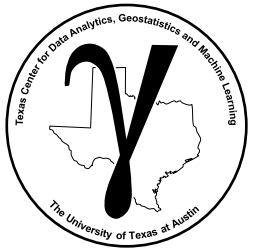
An alternative to the data scientist, the 4th paradigm-ready engineer or scientist

- Continue to grow our domain knowledge, engineering
- Build from our knowledge in data analytics and (geo)statistics
- Grow scripting and coding with open-source data analytics and machine learning



The data scientist Venn diagram and a proposed alternative for 4th paradigm ready engineers and geoscientists.

We are building on our geoscience and engineering strengths.



The Power of Data Analytics

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 Subsurface Machine Learning

Lecture 6: Machine Learning

Lecture outline:

- **Machine Learning Overview**
- **Model Fitting, Overfitting and Model Generalization**
- **Examples of Machine Learning**
- **Energy Machine Learning**