PGE 383
Machine Learning

Lecture outline . . .

- **Machine Learning Overview**
- **A Simple Machine**

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
- Make a simple, illustrative machine

PGE 383
Machine Learning

Lecture outline . . .

- **Machine Learning Overview**

Michael Pyrcz, The University of Texas at Austin

The Model

- Predictors, Independent Variables, Features
  - input variables
  - for a model $Y = f(X_1, \dots, X_m) + \epsilon$, these are the $X_1, \dots, X_m$
  - note ϵ is a random error term

- Response, Dependent Variables
  - output variable
  - for a model $Y = f(X_1, \dots, X_m)$, this is Y

- Statistical / Machine Learning is All About
  - Estimating f for two purposes
    1. Inference
    2. Prediction

Inference

- Learning About the System
  - for $Y = f(X_1, \ldots, X_m) + \epsilon$ we can understand the influence / interactions of each X_α on Y and eachother.

- Inferential Statistics
  - Given a random sample from a population, describe the population
  - E.g. given 7 heads of 10 flips, what's the probability that the coin is fair?
  - E.g. given 7 success wells of 10 drilled, what's the probability of a successful well in this reservoir?

Prediction

- Estimating, $\hat{f}$, for the purpose of predicting $\hat{Y}$
  - We are focused on getting the most accurate estimates, $\hat{Y}$
- Predictive Statistics
  - Predict the samples from a population
  - E.g. given10 flips, what's the probability of 7 heads?
  - E.g. given 10 wells will be drilled, what's the probability of 7 successful wells in this reservoir?

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

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

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

where we have n observations. The challenge is that that real question we have is how well can we predict outside the training data – testing data.

$$\mathrm{E}\left[\left(y_0 - \hat{f}(x_1^0, \dots, x_m^0)\right)^2\right]$$

over testing data

over a variety of unsampled sets of predictors $x_1^0, \dots, x_p^0$. We want to know how our model performs when we move away from the training set of data!

Model Bias and VarianceTrade-off

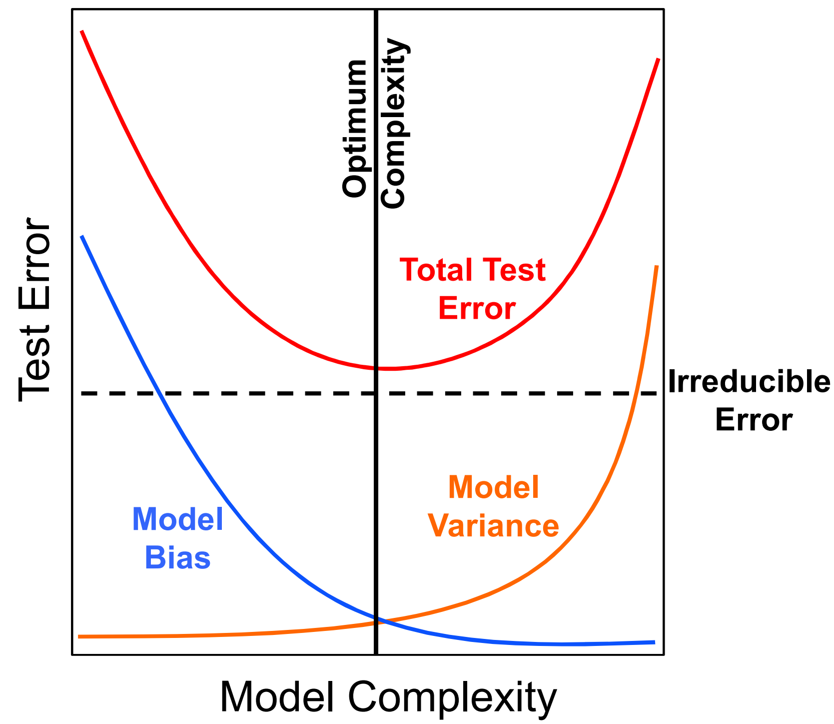
- The **Expected Test Mean Square Error** may be calculated as:

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

  - **Model Variance** is the variance if we had estimated the model with a different training set / sensitivity to data /

  - **Model Bias** is error due to using an approximate model / model is too simple

  - **Irreducible error** is due to missing variables and limited samples can't be fixed with modeling / entire feature space is not sampled



Model Parameters Definition

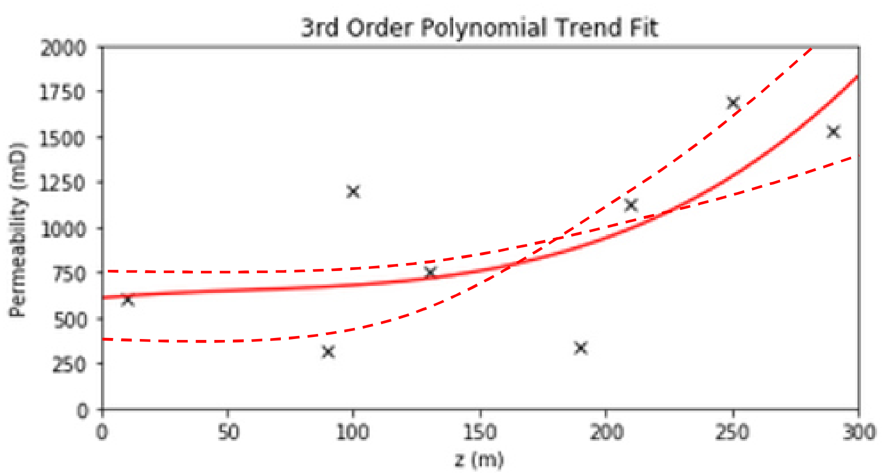
Model Parameters

Derived during training phase to fit the model to the training data (minimize error with training data)

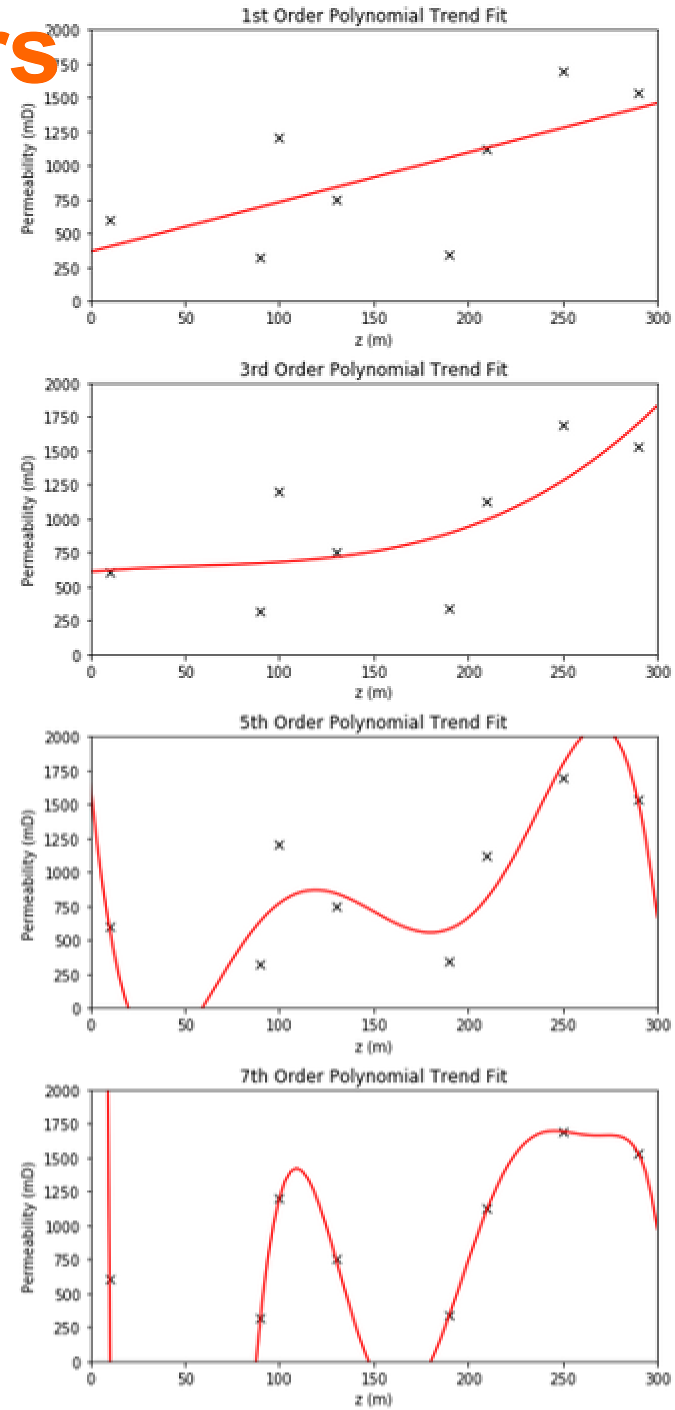
$$k = b_3 z^3 + b_2 z^2 + b_1 z + c$$

Parameters

b_3, b_2, b_1 and c



Model Hyperparameters Definition



Model Hyperparameters

Set prior to learning from the data. Impact the form of the model and often the 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: $k = b_1 z + c$

No appropriate to set with training data, or we will be overfit

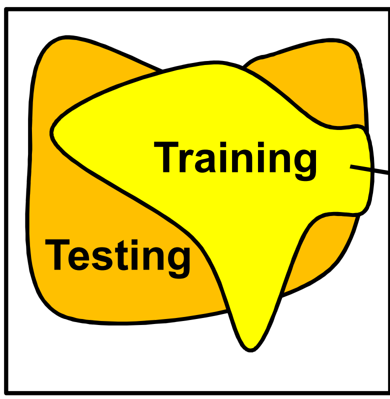
Tune hyperparameters with withheld testing data.

Training and Testing

The Training and Testing Workflow

- establish a subset of the data for fair testing of the model

Separate The Data



Train the Model Parameters Maximize Accuracy with Training Data

Tune the Model

Test the Model Tune the Hyperparameters to Optimize the Complexity

We avoid the overfit problem.

Model Complexity / Flexibility Definition

Model Complexity / Flexibility

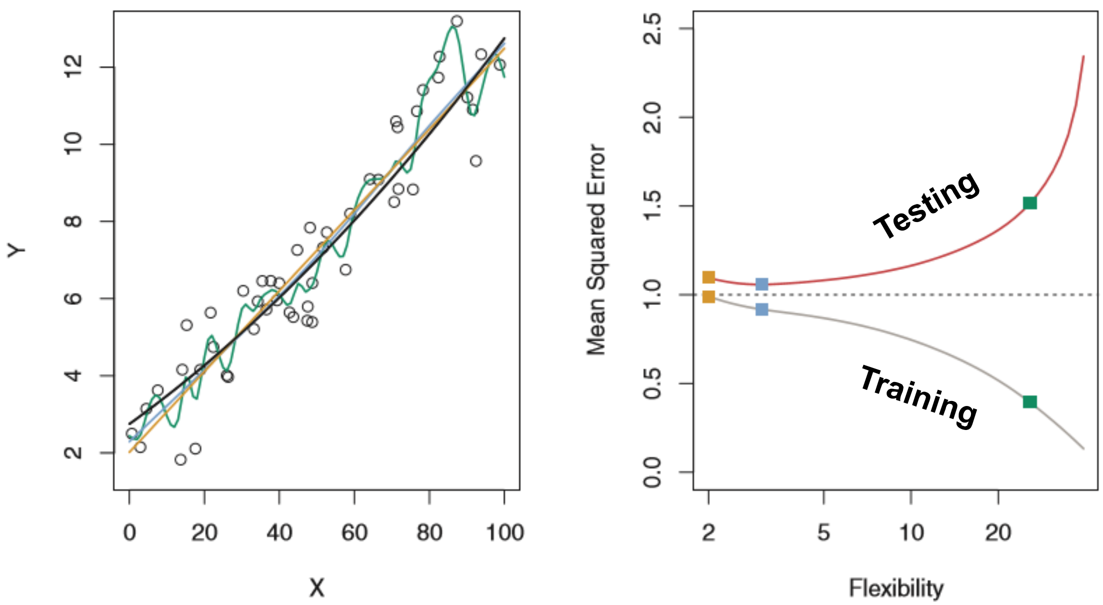
A variety of concepts may be used to describe model complexity:

- 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.

Overfitting

- **Flexibility vs. Accuracy**
  - Increased flexibility will generally decrease MSE on the **training dataset**
  - May result in increase MSE with **testing data!** Worse prediction!
  - Not generally a good idea to select method only to minimize training MSE

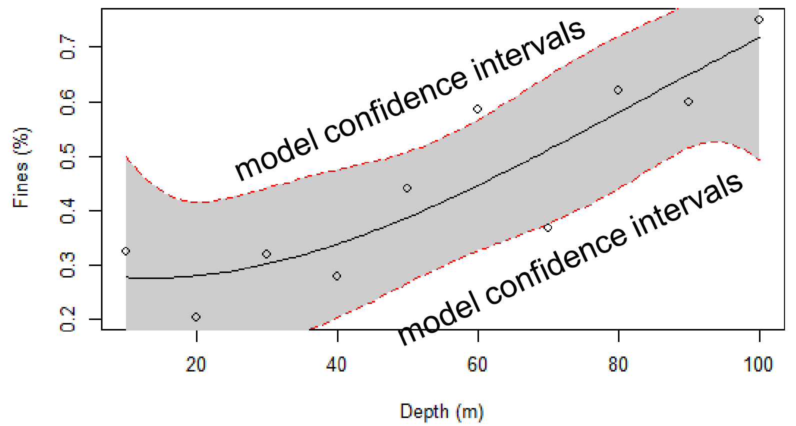


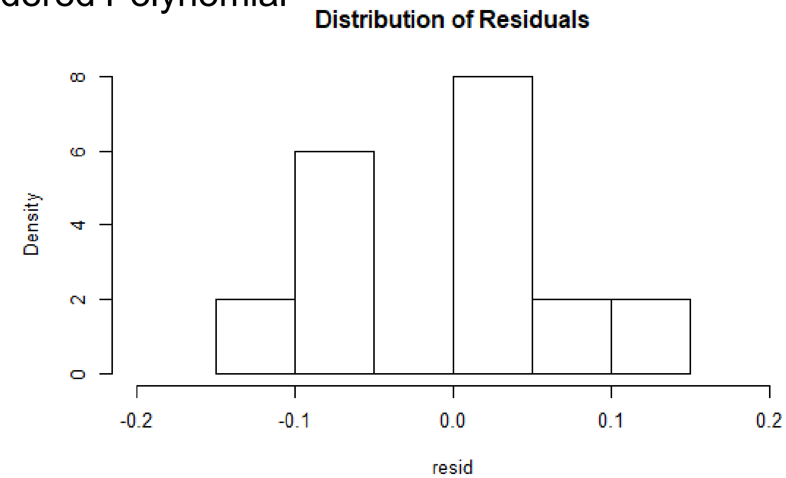
Data and model fits (left) and MSE for training and testing (right) from James et al. (2013).

  - High flexibility + minimize MSE = likely overfit.

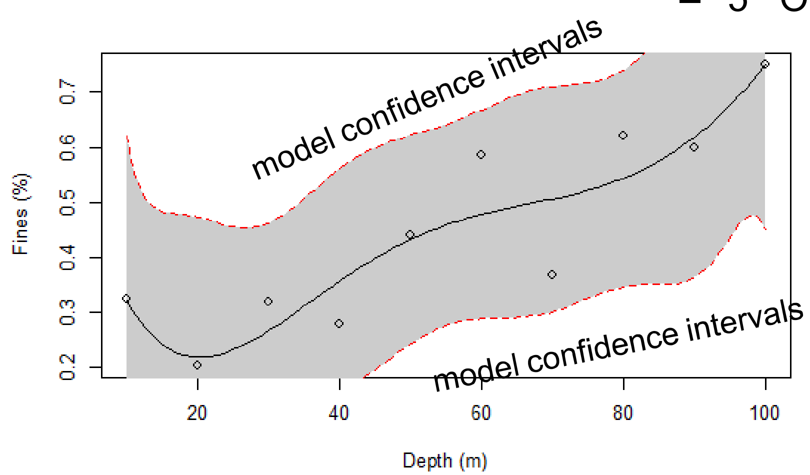
Overfitting

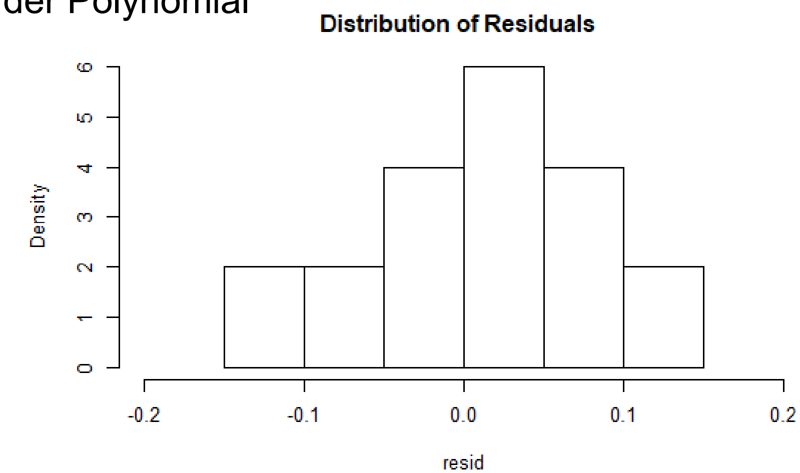
- Example of trend fits:
  - 3rd Ordered Polynomial





  - 5th Order Polynomial



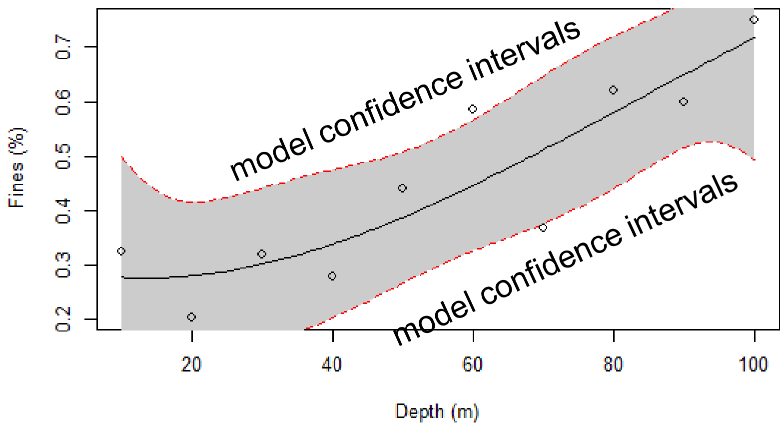


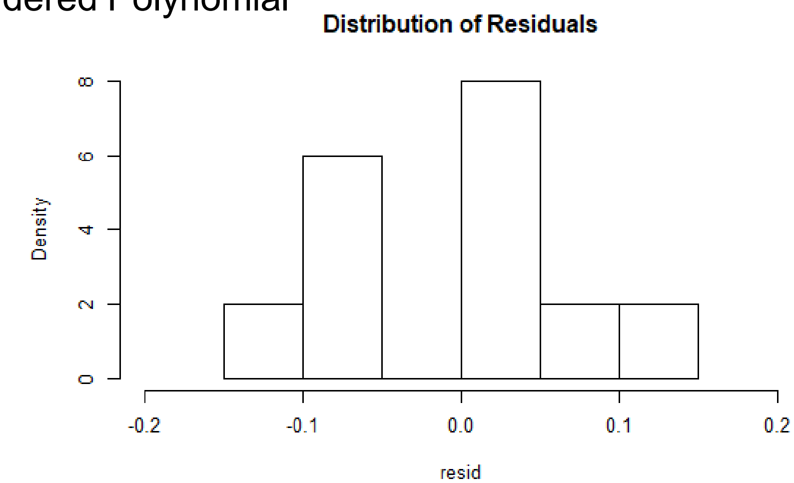
Overfit demonstration in R, code is here:
https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R

R code at Code/Overfit.R

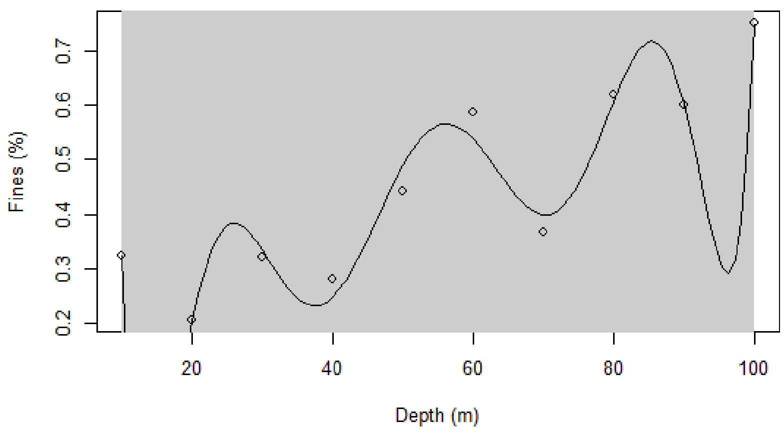
Overfitting

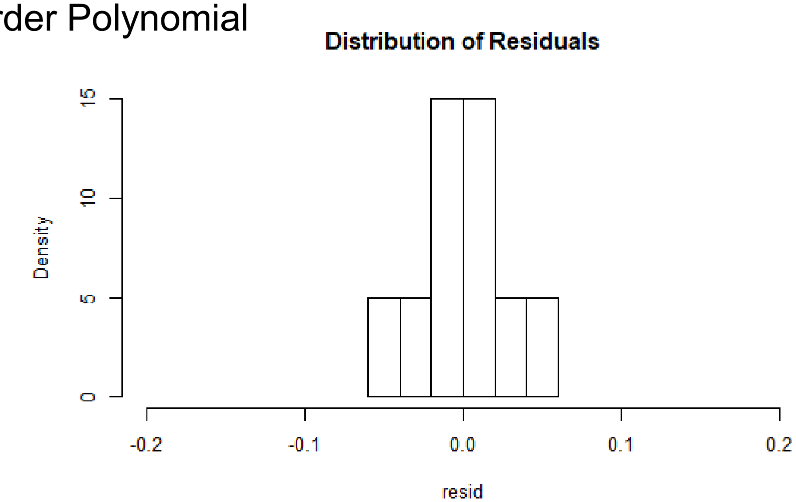
- Example of trend fits:
  - 3^{rd} Ordered Polynomial





  - 8^{th} Order Polynomial





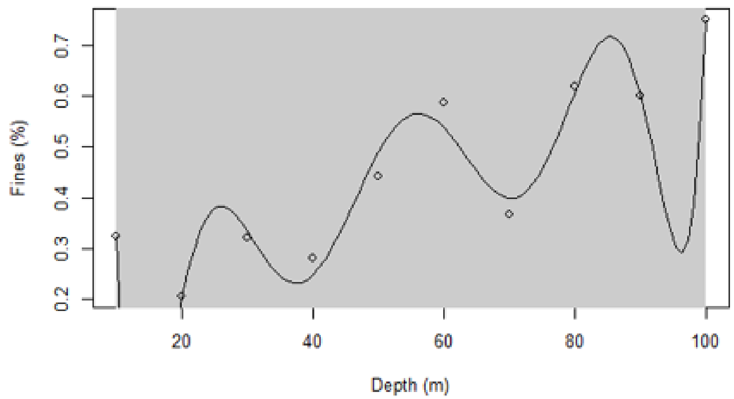
Overfit demonstration in R, code is here:
https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R

R code at Code/Overfit.R

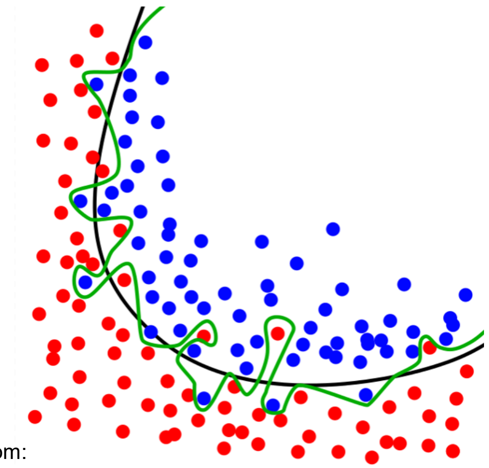
Definition of Overfitting

- Overly complicated model to explain “idiosyncrasies” of the data, capturing data noise in the model
- More parameters than can be justified with the data

- Results in likely very high error away from the data / new data
- But, results in low residual variance!
- High R^2
- Very accurate at the data! - Claim you know more than you actually do!



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



Overfit classification model example from:
https://en.wikipedia.org/wiki/Overfitting#/media/File:Overfitting.svg

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, computational resources
  - Expanding opportunities for machine learning

- We'll start unsupervised, dimensional reduction:
  - Principal Component Analysis

PGE 383
Machine Learning

Lecture outline . . .

- **A Simple Machine**

Michael Pyrcz, The University of Texas at Austin

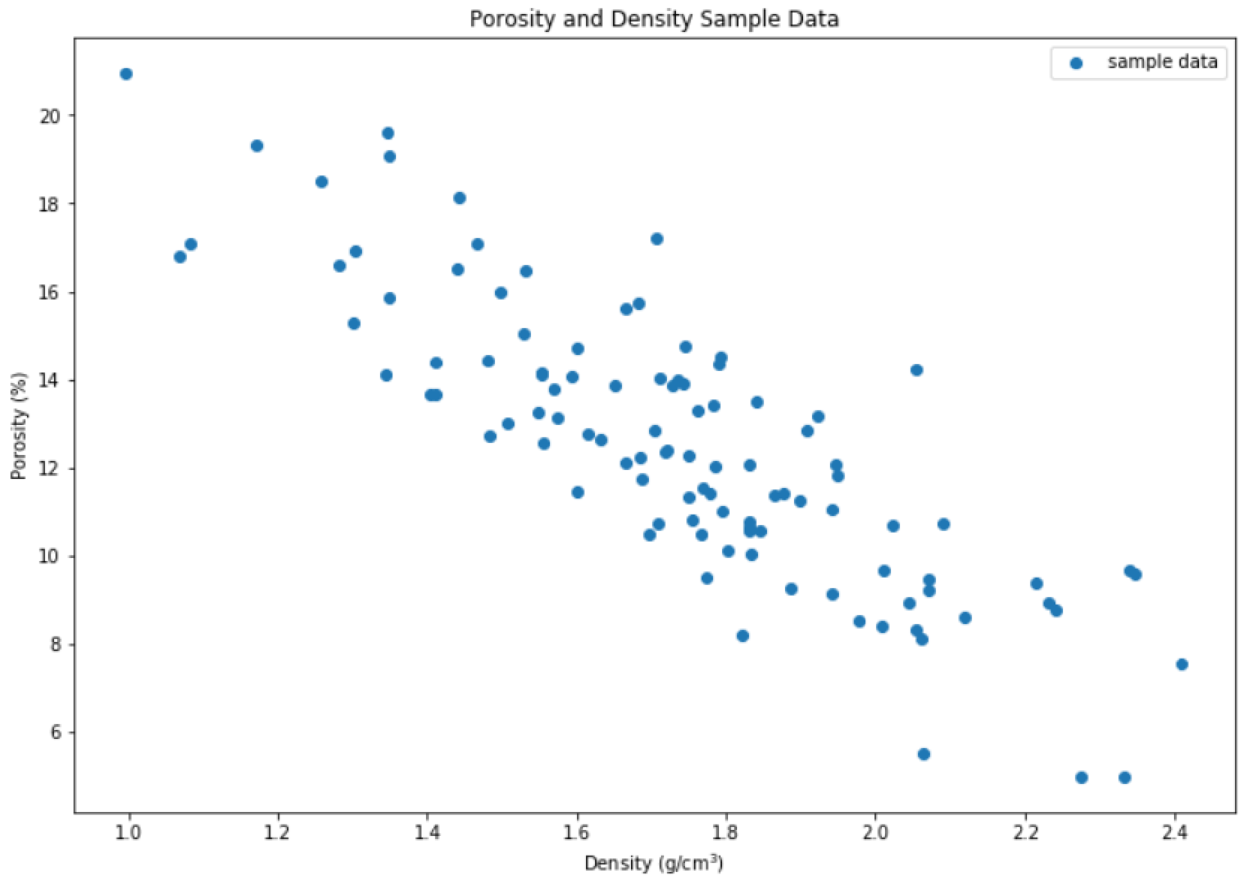
Statistical / Machine Learning for Prediction

What is Machine Learning?

- A mathematical / statistical model that learns from data, supported with expert knowledge
- Not explicitly told how to predict
- General method that may be applied to a range of problems

Our First Machine

- **Loaded up a simple porosity vs. density dataset in Python.**



Our First Machine

- Ran one line of Python and built a linear regression model

LinearRegression Model

Let's first calculate the linear regression model

```
slope, intercept, r_value, p_value, std_err = st.linregress(den,por)

print('The model parameters are, slope (b1) = ' + str(round(slope,2)) + ', and the intercept
```

```
The model parameters are, slope (b1) = -9.1, and the intercept (b0) = 28.35
```

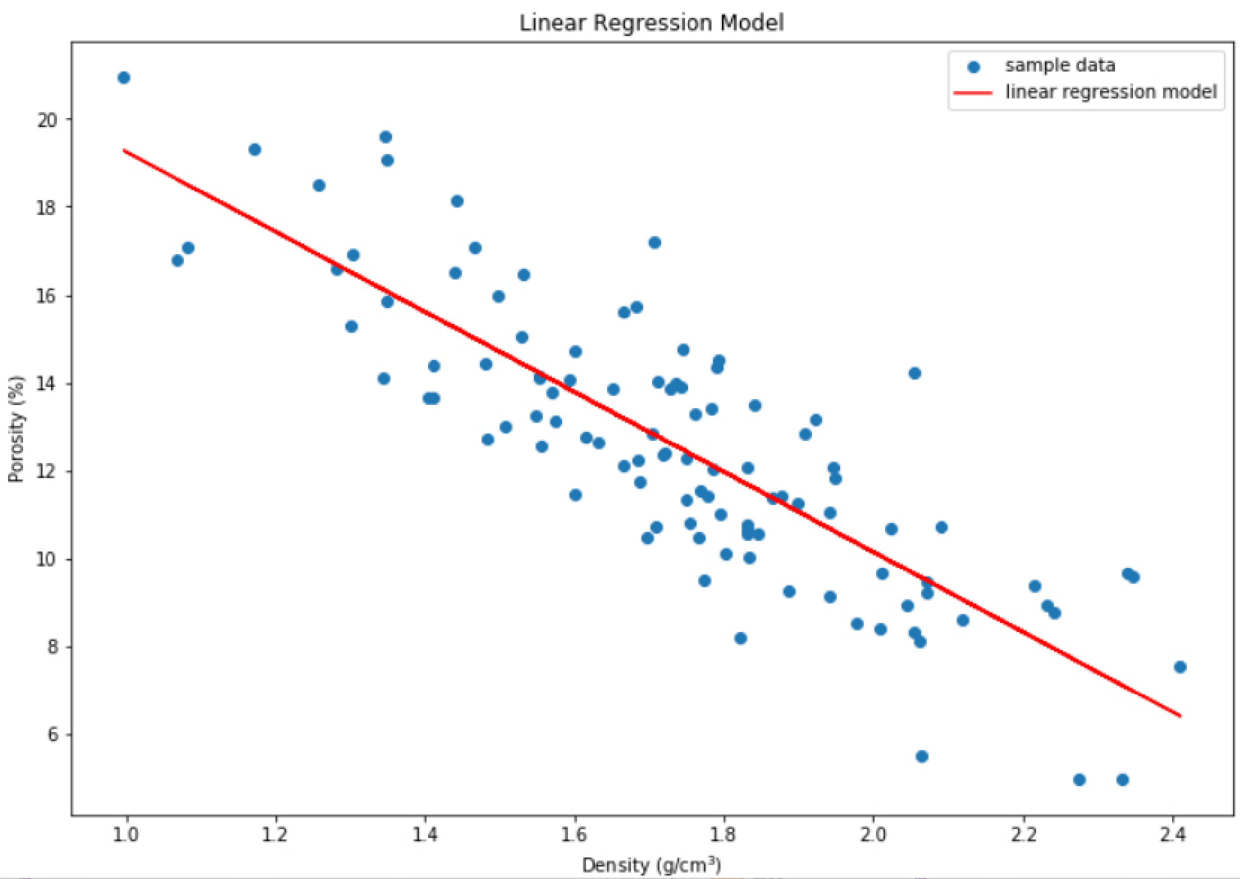
- The model is simply a line:

Response Feature → $\phi = b_1 \cdot \rho + b_0$ ← **Predictor Feature**

Our First Machine

Let's look at the model.

- If we change the data, the model would update. It learns!
- Nothing intimidating about linear regression!



Our First Machine

Model Parameters Set to Minimize Mismatch at With Training Data Locations

$$por = \color{red}{b_0} + \color{red}{b_1} \times density$$

- **Objective:**
  - **Find b_1 and b_0, fit a linear function, to:**
    - » **minimize Δy_i over all the data.**
    - » **Δy_i is prediction error**

$$\Delta y_i = y_i - y_{est}$$

data model

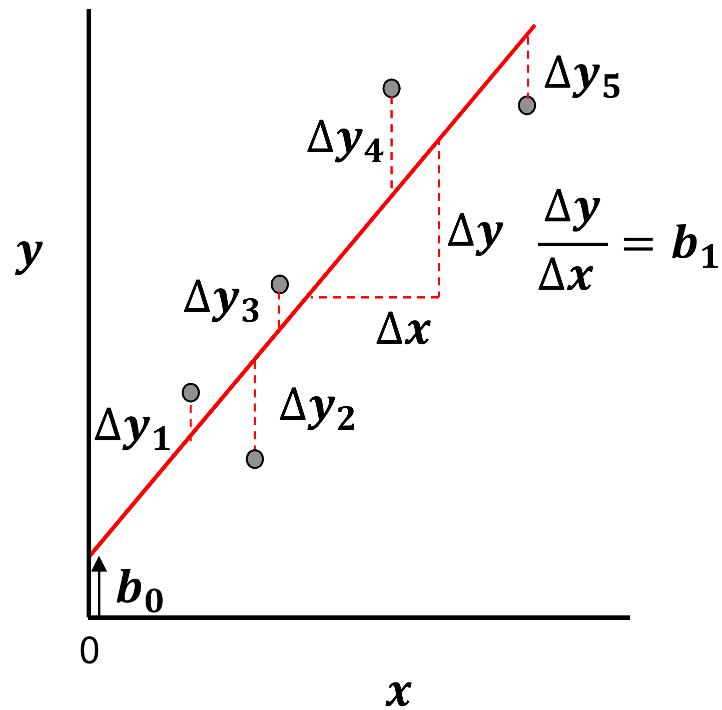
- **Minimize:**

Sum of Square Error

$$\sum_{i=1}^{n} (\Delta y_i)^2 = \sum_{i=1}^{n} (y_i - (b_0 - b_1 x))^2$$

Skipped derivation.

$$b_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}, \quad b_0 = \bar{y} - b_1 \bar{x}$$



Our First Machine

The Model Includes Important Assumptions About The Data and the Model

- **Error-free:** predictor variables are error free, not random variables
- **Linearity:** response is linear combination of feature(s)
- **Constant Variance:** error in response is constant over predictor(s) value
- **Independence of Error:** error in response are uncorrelated with each other
- **No multicollinearity:** none of the features are redundant with other features

Our First Machine

The Model Can Be Tested for Significance and the Proportion of Variance Explained.

– r^2: strength of the model, proportion of variance explained by the model

Variance explained by the model

$$ssreg = \sum_{i=1}^{n} (\hat{y}_i - \overline{y})^2$$

Variance NOT explained by the model

$$ssresid = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$r^2 = \frac{ssreg}{ssreg + ssresid} = \frac{explained\ variation}{total\ variation}$$

– also note for bivariate case, $r^2 = (\rho)^2$, we can relate r^2 to the Pearson's correlation coefficient, ρ.

Our First Machine

We Can Calculate the Uncertainty in the Model

- Confidence interval for model parameters given the available training data

$$\widehat{b_1} \pm t_{\left(\alpha/_{2},n-2\right)} \times SE_{b_1} \qquad \widehat{b_1} \pm t_{\alpha/2,n-2} \times \left(\frac{\sqrt{n}\hat{\sigma}}{\sqrt{n-2}\sqrt{\sum(x_i-\bar{x})^2}} \right)$$

se1 in Excel

$$\widehat{b_0} \pm t_{\left(\alpha/_{2},n-2\right)} \times SE_{b_0} \qquad \widehat{b_0} \pm t_{\alpha/2,n-2} \times \left(\sqrt{\frac{\hat{\sigma}^2}{n-2}} \right)$$

seb in Excel

Our First Machine

Provides an Uncertainty Model for the Predictions

Recall prediction interval are concerned with uncertainty in the next observation

next sample

– We answer the question, given I know the porosity, x_{n+1}, what is the interval (e.g.) with 95% probability containing the true value permeability, y_{n+1}?

$$\hat{y}_{n+1} \pm t_{\alpha/2,n-2} \boxed{\sqrt{MSE}\sqrt{1+\frac{1}{n}+\frac{(x_{n+1}-\bar{x})^2}{\sum(x_i-\bar{x})^2}}}$$

model estimate

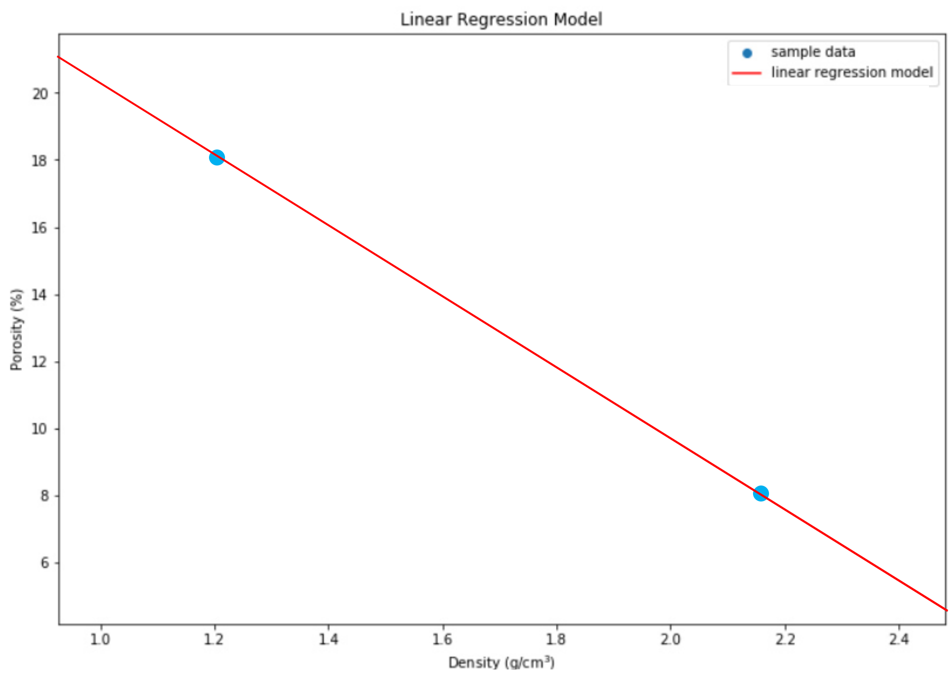
t-statistic

$$MSE = \sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{n-2} = \sum_{i=1}^{n} \frac{(y_i - (b_0 - b_1 x))^2}{n-2}$$

standard error of our model estimate

Our First Machine

Would this be a fair model?



- Does the data support this model? We are **overfitting** the data!
- Is it safe to **extrapolate** with this model away from the data?

Our First Machine

What did we learn from our simple machine?

1. Flexible to fit the data, learns from the data
2. Minimize error with the training data
3. Important assumptions about the data and model
4. Model can be tested for significance and the proportion of variance explained
5. Includes uncertainty in the model
6. Predict based on new data with uncertainty
7. Issues with overfit and extrapolation

Think of machine learning as advanced linear regression / line fitting to data!

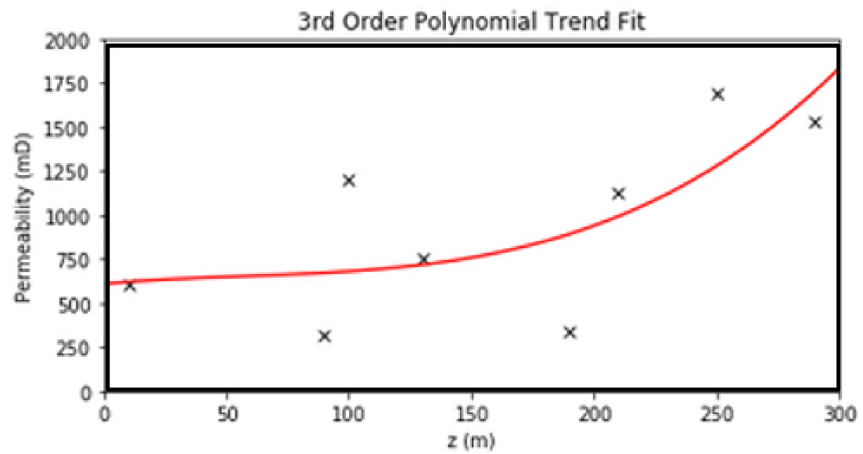
Training Our Machine

Apply Training Data to Set the Model Parameters.

For example, the parameters of this 3rd order polynomial model.

b_3, b_2, b_1 and c

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



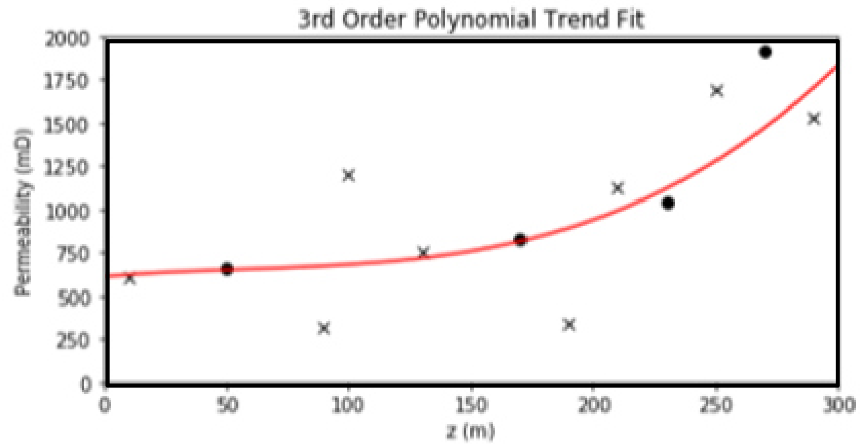
But not appropriate to determine level of complexity (hype parameter)

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

Testing Our Machine

Apply Withheld Data to Test our Machine.

For example, the parameters of this 3rd order polynomial model.



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

In testing we use the parameters from training but we tune the hyperparameters.

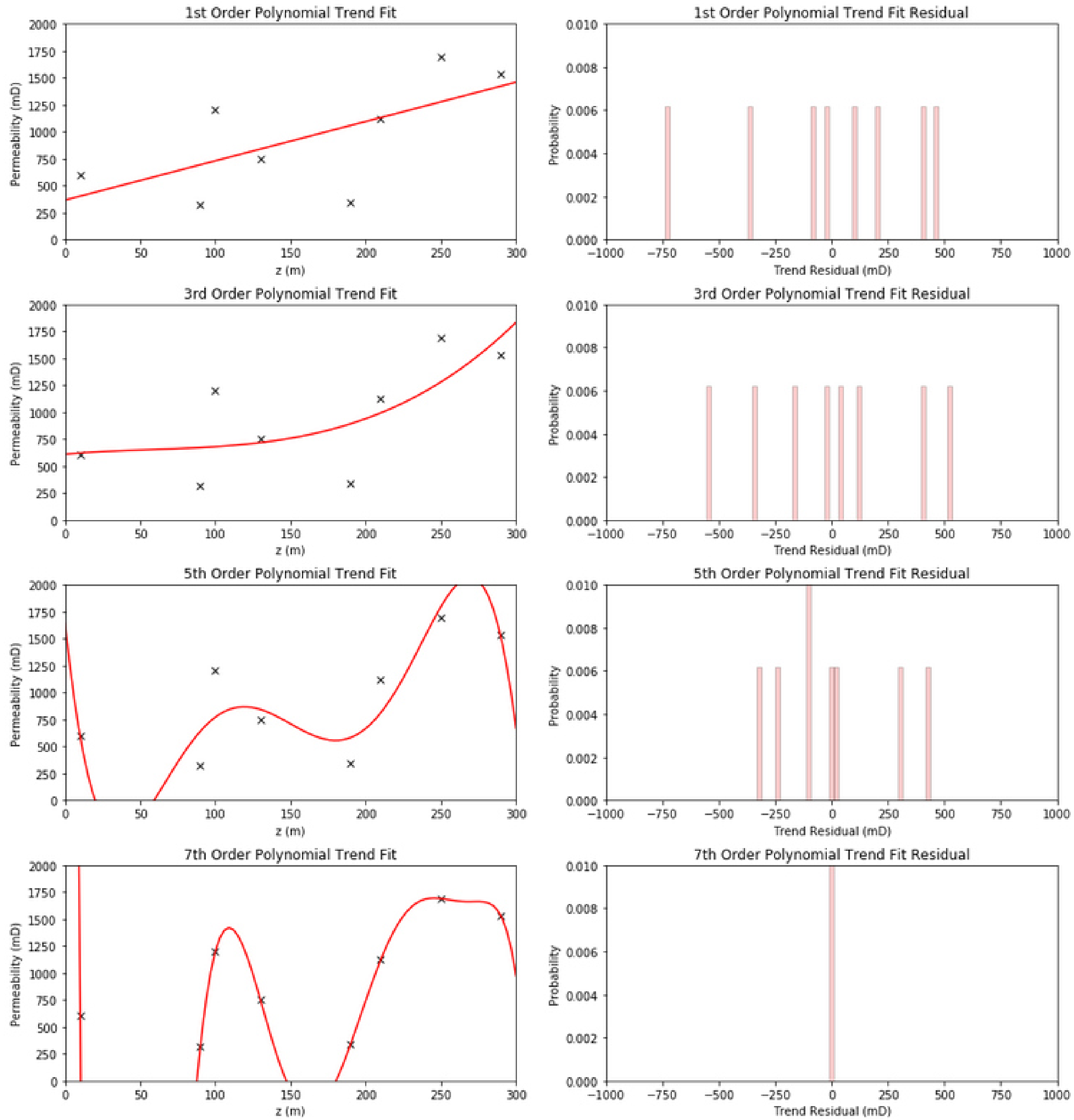
Hyperparameter of our model:
1st, 2nd, 3rd 4th … order polynomial?

Making Our Machine

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

Very complicated model would be best.

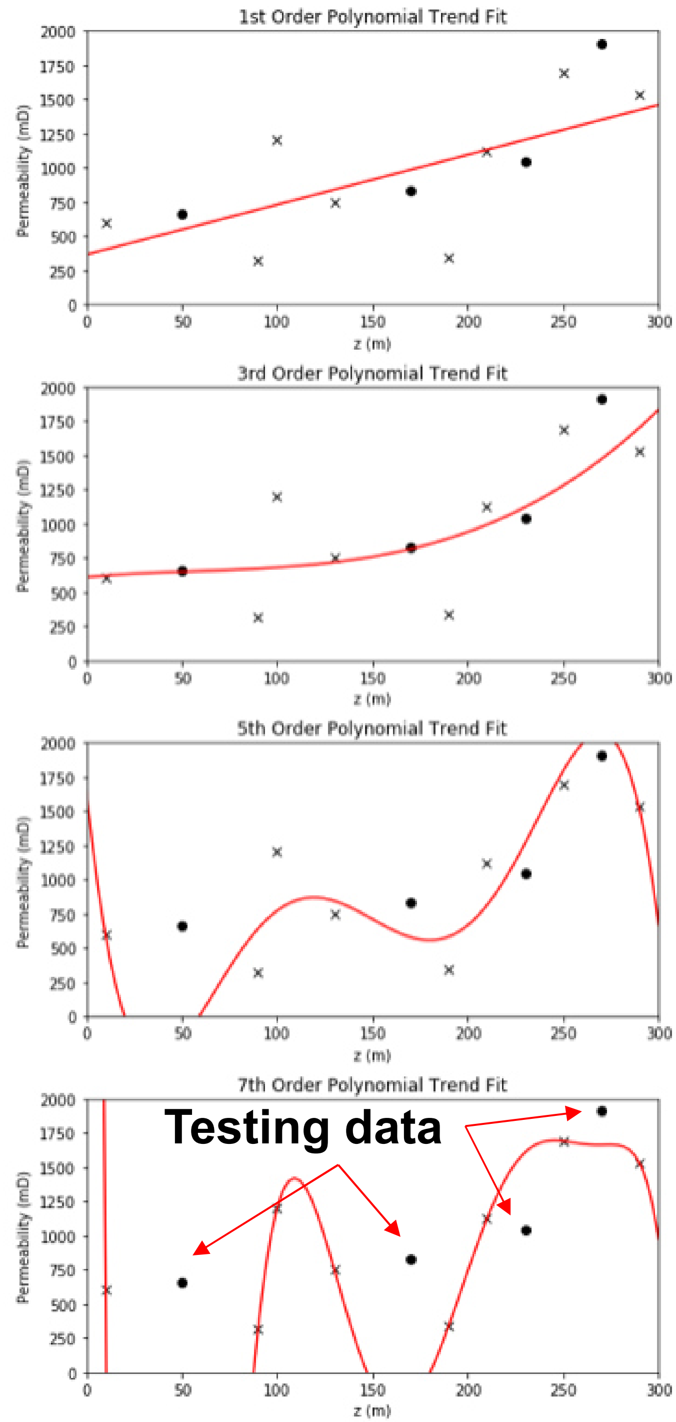
Perfectly fit the data.



Making Our Machine

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**.
5. Our best model is low to moderate complexity in this setting!



PGE 383
Machine Learning

Lecture outline . . .

- **Machine Learning Overview**
- **A Simple Machine**

Michael Pyrcz, The University of Texas at Austin