

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- Machine Learning / Inference
- Multivariate Analysis
- Feature Ranking
- Dimension Reduction

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin

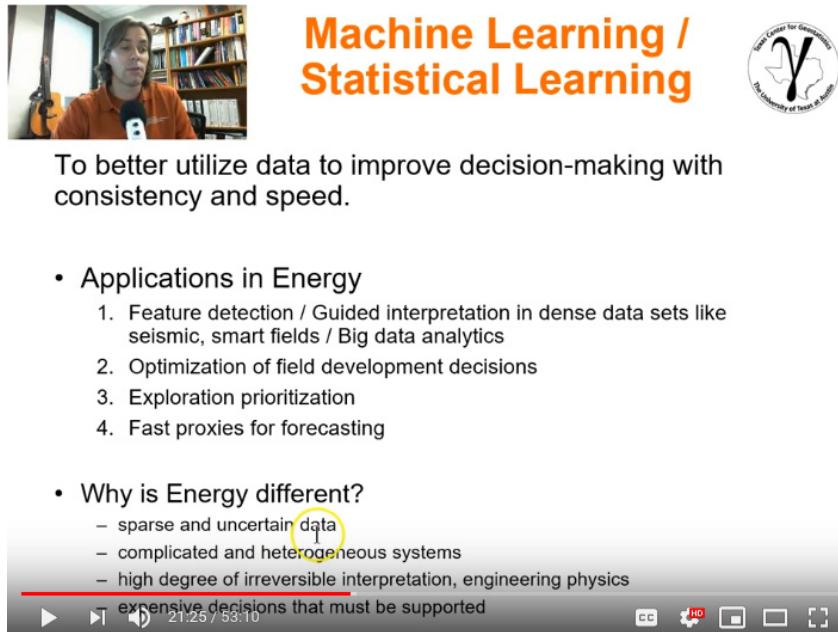
Subsurface Data Analytics in Python

Data Analytics - Inference



Other Resources:

- Recorded Lectures Statistical / Machine Learning



Machine Learning / Statistical Learning

To better utilize data to improve decision-making with consistency and speed.

- Applications in Energy
 1. Feature detection / Guided interpretation in dense data sets like seismic, smart fields / Big data analytics
 2. Optimization of field development decisions
 3. Exploration prioritization
 4. Fast proxies for forecasting
- Why is Energy different?
 - sparse and uncertain data
 - complicated and heterogeneous systems
 - high degree of irreversible interpretation, engineering physics

expensive decisions that must be supported

Instructor: Michael Pyrcz, the University of Texas at Austin

Goals of This Lecture



- Review and demonstrate some basic inference approaches

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- Machine Learning / Inference

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin



Inference

- There is value in understanding the relationships
 - for $Y = f(X_1, \dots, X_m) + \epsilon$ we can understand the influence / interactions of each X_α on Y *and* each other.
- 1. Which predictors are associated with the response?
 - a) What data to collect? Value of information.
 - b) What data to focus on? Simplification of the model. Communication. Big hitters.
- 2. What is the relationship between each response and each predictor?
 - a) sense of the relationship (positive or negative)?
 - b) shape of relationship (sweet spot)?
 - c) relationships may depend on values of other predictors!

'Inference is learning about the system.'

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- **Multivariate Analysis**

Introduction

Basics of Python

Data Preparation

Statistics

Inferential Methods

Predictive Methods

Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin

Motivation for Multivariate Methods



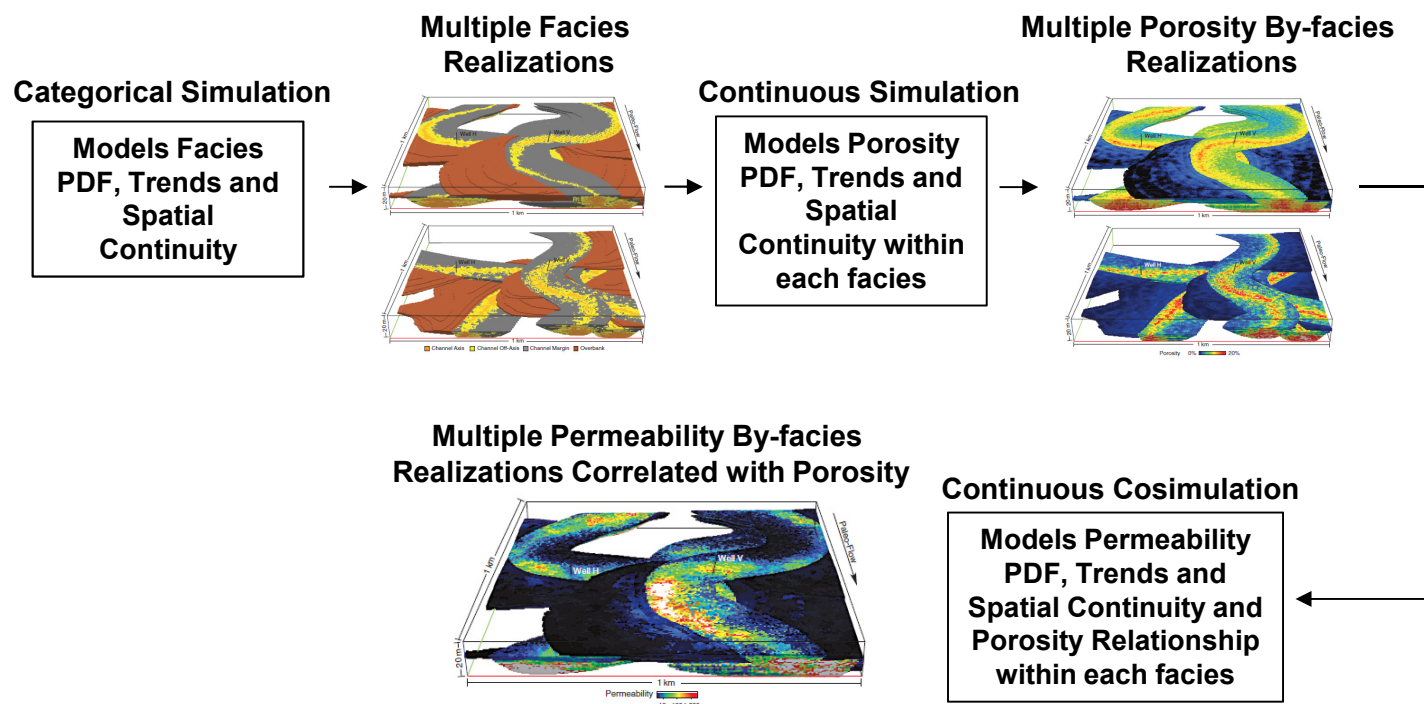
- **We typically need to build reservoir models of more than one property of interest.**
 - Expanded by whole earth modeling, closing loops with forward models
 - Expanded by unconventional
- **Subsurface properties may include:**
 - Rock Classification: lithology, architectural elements, facies, depofacies
 - Petrophyscial: porosity, directional permeability, saturuations
 - Geophysical: density, p-wave and s-wave velocity
 - Gemechanical: compressibility / Poisson's ratio, Yong's modulus, brittleness, stress field
 - Paleo- / Time Control: fossil adundances, stratigraphic surfaces, ichnofacies, paleo-flow indicators

Motivation for Multivariate Methods



- **A Confession:**

- Standard geostatistical workflows are bivariate at most
 - » e.g. simulate permeability conditional to porosity



Note: only had 1 realization on hand (should be two in figure).

Motivation for Multivariate Methods



- **Emerging Multivariate Methods Include:**
 - Transforms – remove correlations and then model with independent variables and then back-transform to restore correlation (e.g. step-wise conditional transform).

Beyond the scope of this course as they are not in common practice.

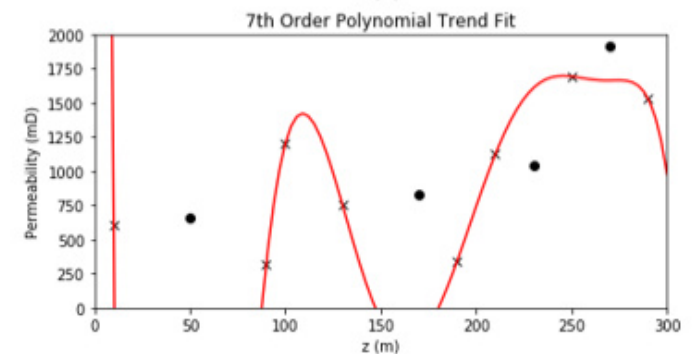
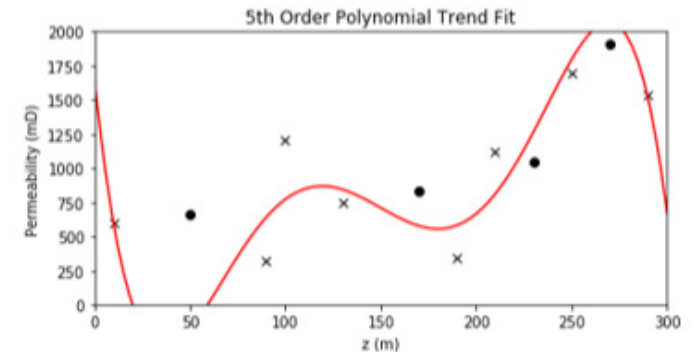
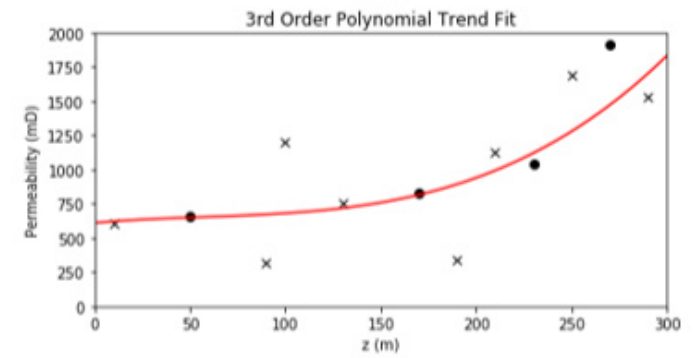
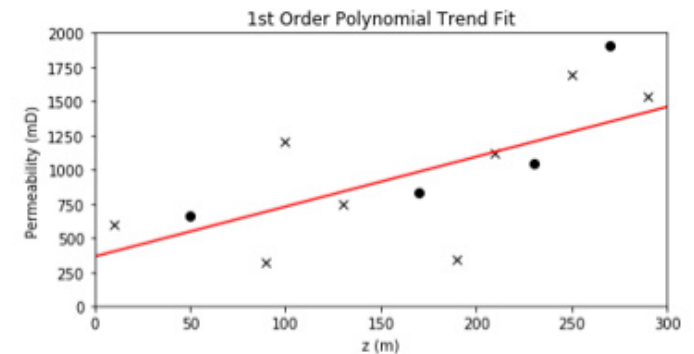
Curse of Dimensionality



- **Working with more features / variables is harder!**
 1. More difficult to visualize
 2. More data are required to infer the joint probabilities
 3. Less coverage
 4. More difficult to interrogate / check the model
 5. More likely redundant
 6. More complicated, more likely overfit

Visualization

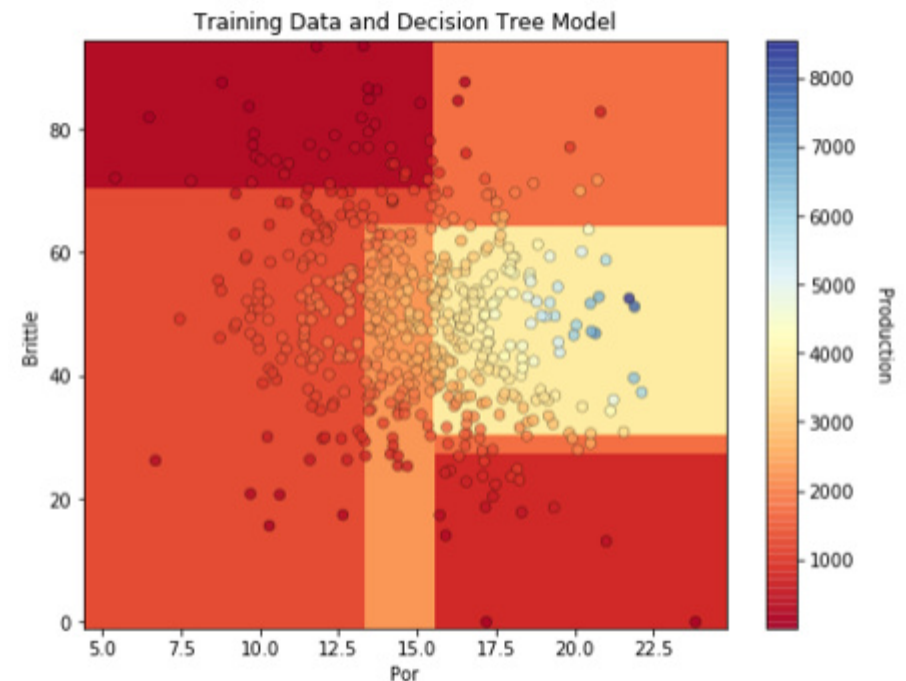
- **Consider this simple model:**
 - 1 predictor feature
 - 1 response feature
- How's our model performing?
 - Accuracy in training and testing
- Range of Applicability?
 - Are we extrapolating?
- Overfit
 - Is the model defensible given the data?



Visualization



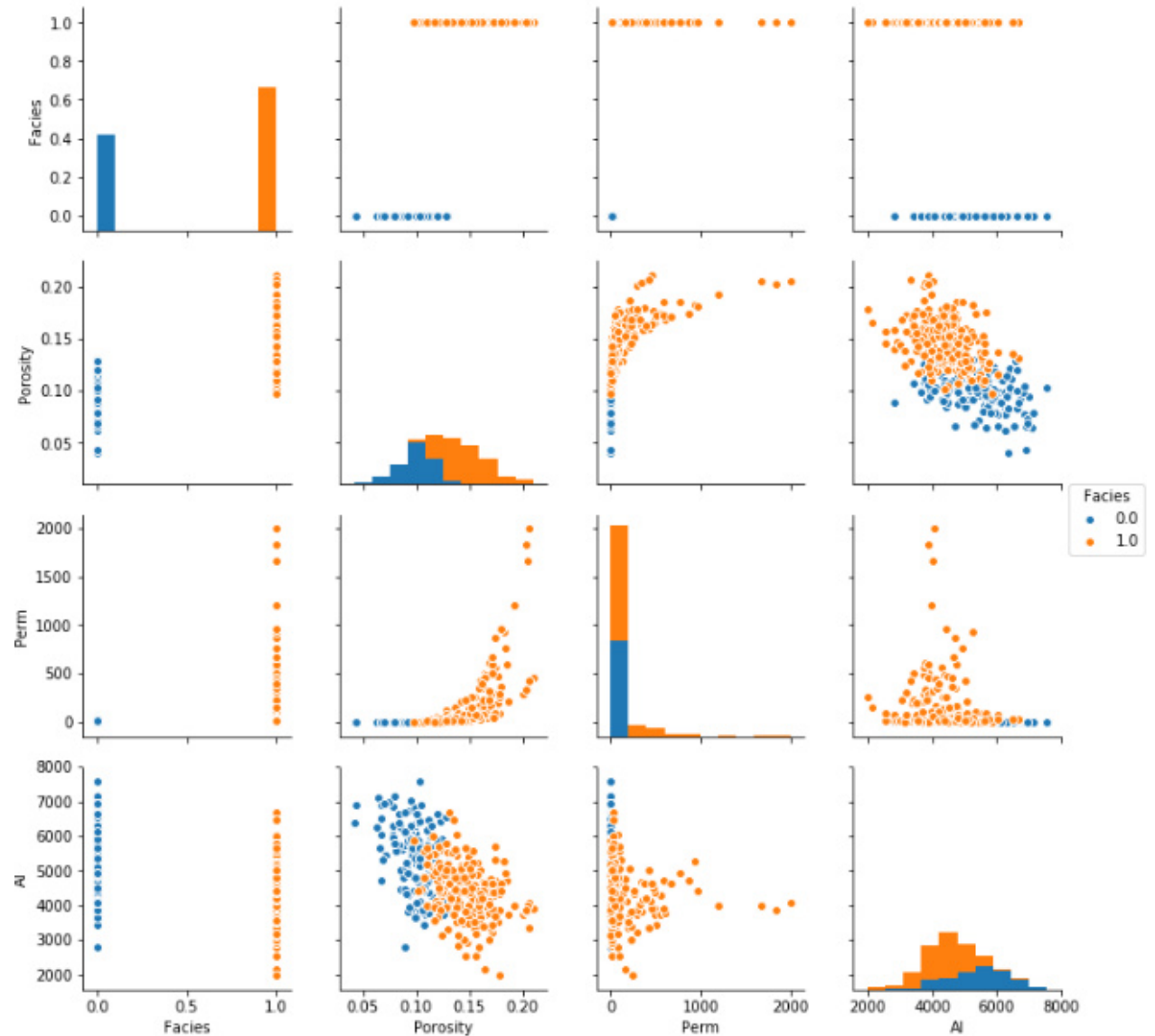
- **Consider this simple model:**
 - 2 predictor features
 - 1 response feature
- How's our model performing?
 - Accuracy in training and testing
- Range of Applicability?
 - Are we extrapolating?
- Overfit
 - Is the model defensible given the data?



Visualization



- **Consider this:**
 - 4 predictor features
 - 1 response feature (not shown)
- What are the relationships between features?
- Are there constraints?



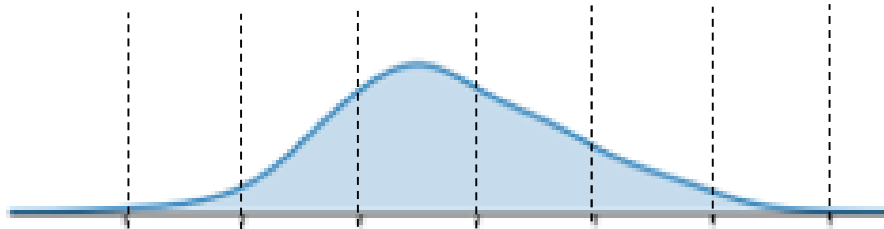
Curse of Dimensionality



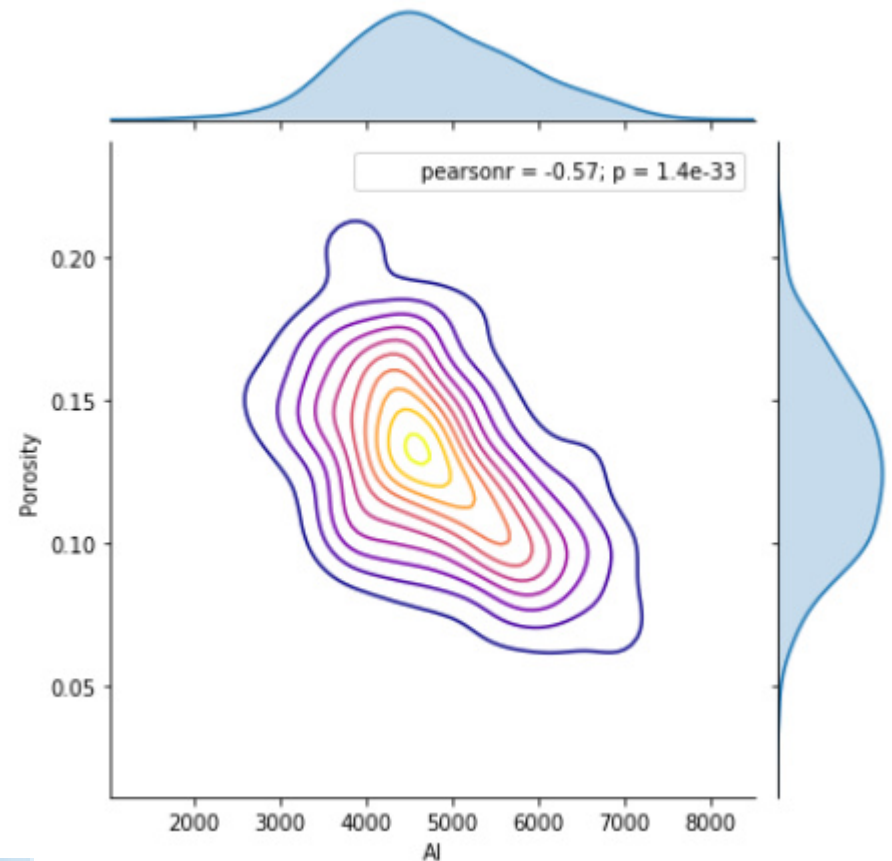
- Consider any joint probability:

$P(X_1 \cap, \dots, \cap X_m)$ the joint probability of X_1, \dots, X_m

- Let's start with 1 feature (m=1)



$$P(X_1^i \leq X \leq X_1^{i+1}) = \frac{n(X_1^i \leq X \leq X_1^{i+1})}{n}$$



In each bin we are estimating a probability!
10 data in each bin = 80 data?

Curse of Dimensionality



- Consider any joint probability:

$P(X_1 \cap, \dots, \cap X_m)$ the joint probability of X_1, \dots, X_m

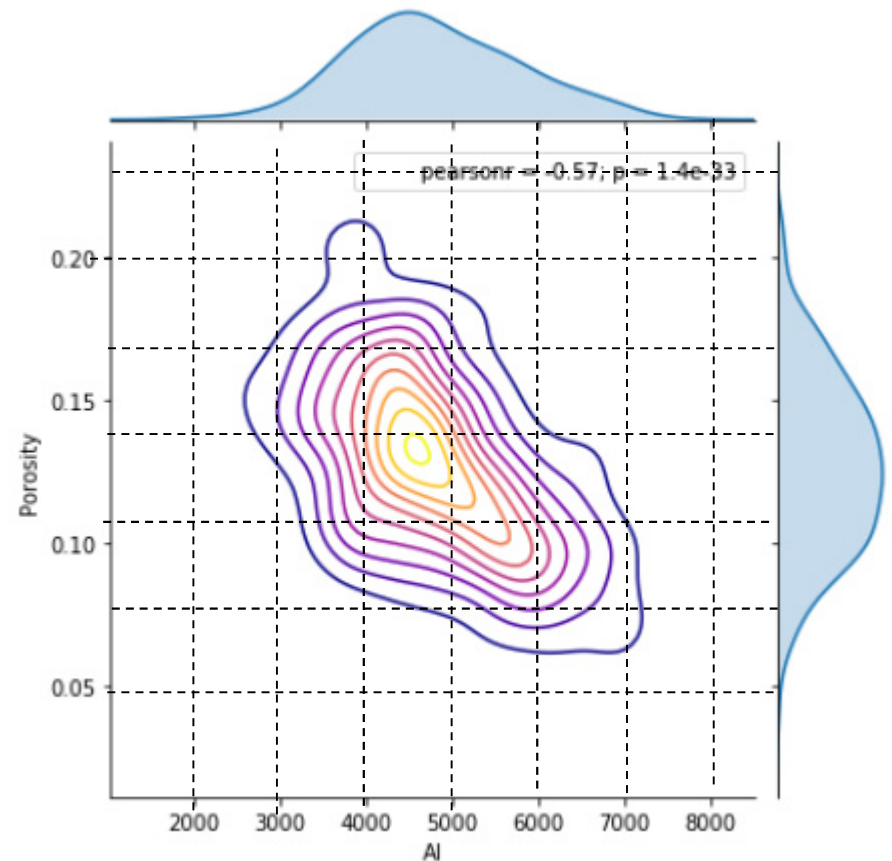
- Now move to 2 features (m=2)

$$P(X_1^i \leq X \leq X_1^{i+1}, X_2^j \leq X \leq X_2^{j+1})$$

$$= \frac{n(X_1^i \leq X \leq X_1^{i+1}, X_2^j \leq X \leq X_2^{j+1})}{n}$$

$$n = \text{Data}/\text{Bin} \cdot \text{Bins}^m$$

- This is optimistic, as it assumes uniform sampling



In each bin we are estimating a probability!

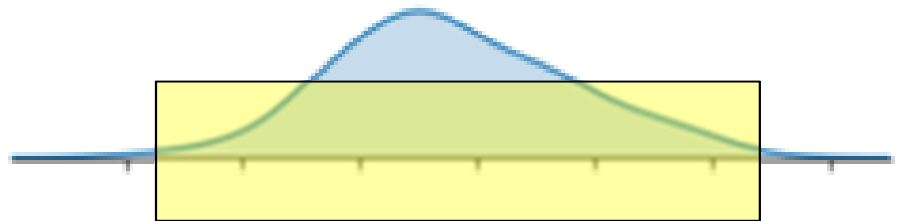
10 data in each bin = 640 data?

Curse of Dimensionality



Consider coverage:

- The range of the sample values
- The fraction of the possible solution space that is sampled.
- Let's return to 1 feature, and assume 80% coverage!
- That's pretty good right?



Curse of Dimensionality



Consider coverage:

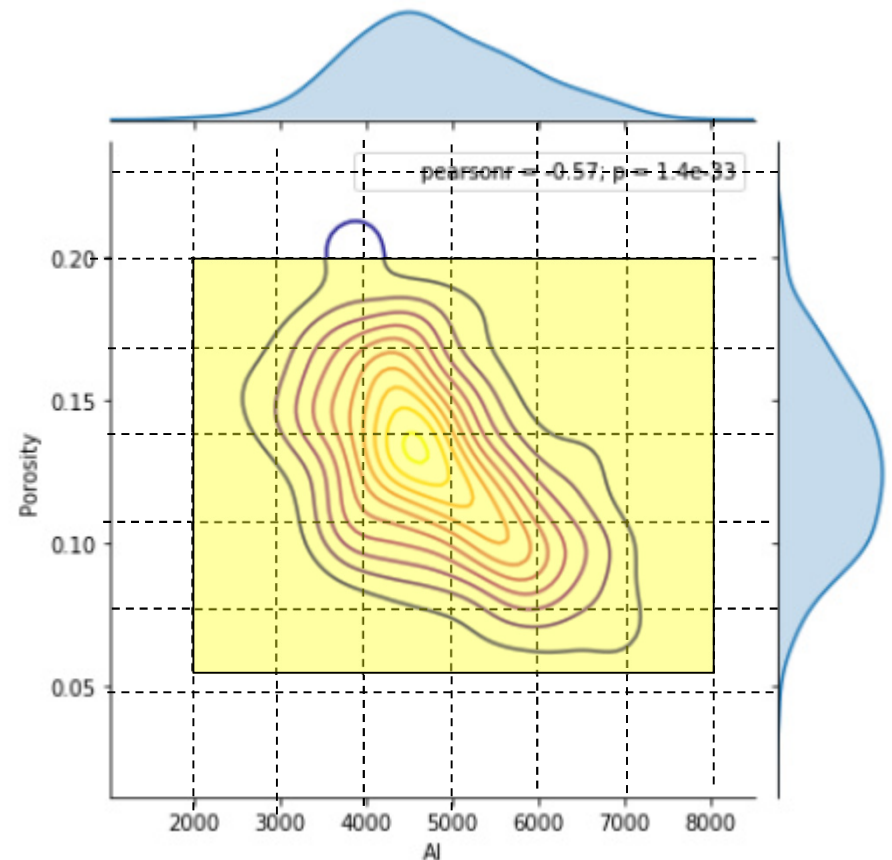
- Now let's move to 2 features, each with 80% coverage
- How much of the solution space is covered?

$$0.8^D, \quad e.g. 0.8^2 = 0.64$$

- Even with exponential increase in number of data:

$$n = \text{Data}/\text{Bin} \cdot \text{Bins}^m$$

coverage is decreasing as we increase the number of features!



Multicollinearity Feature Redundancy

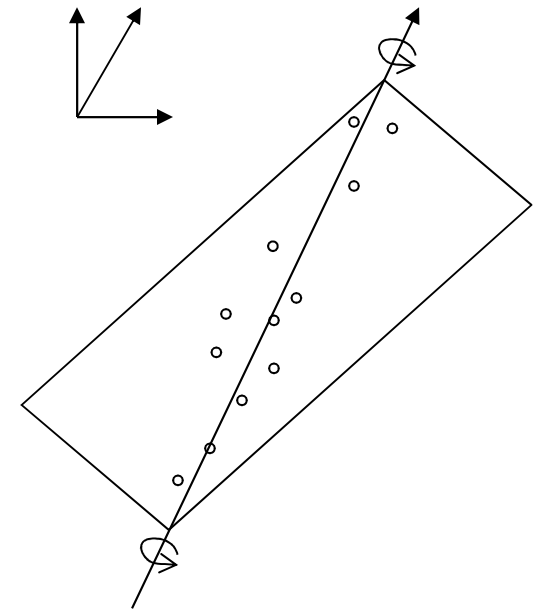


“the existence of such a **high degree of correlation between supposedly independent variables** being used to estimate a dependent variable that the contribution of each independent variable to variation in the dependent variable cannot be determined”

- Merriam-Webster Online Dictionary

“In statistics, **multicollinearity** (also collinearity) is a phenomenon in which one predictor variable in a **multiple regression** model can be linearly predicted from the others with a substantial degree of accuracy.”

- Wikipedia



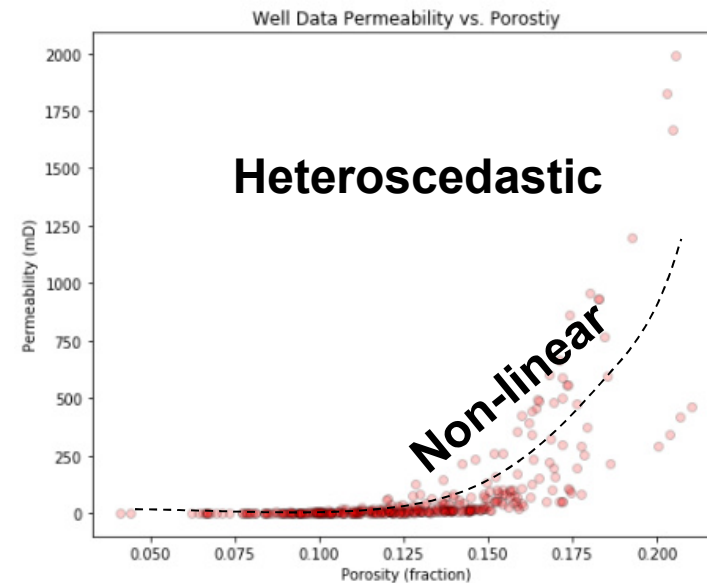
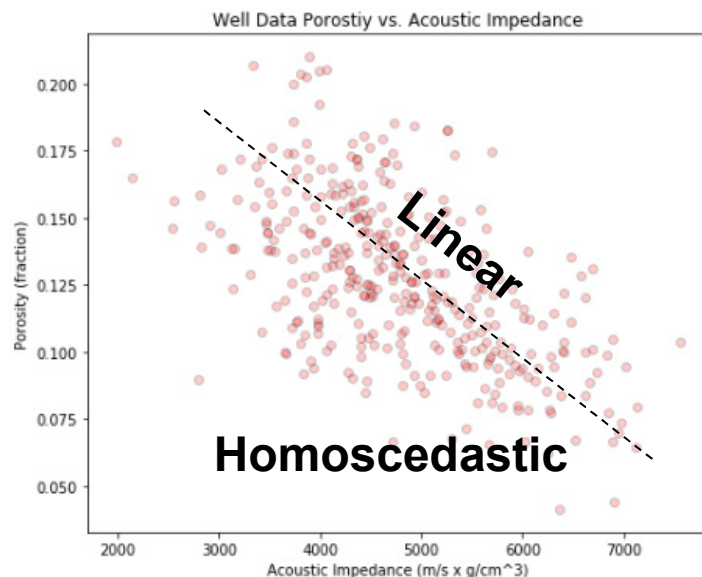
It is like fitting a plane to a line!

Relationships Between Variables



Multivariate visualization:

- Understanding the interactions between variables.



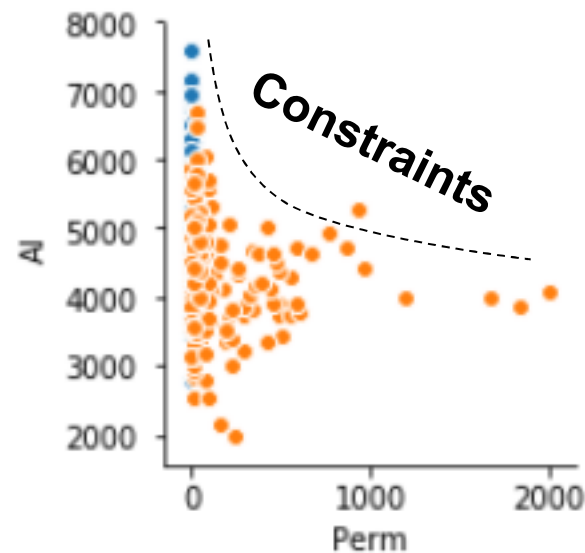
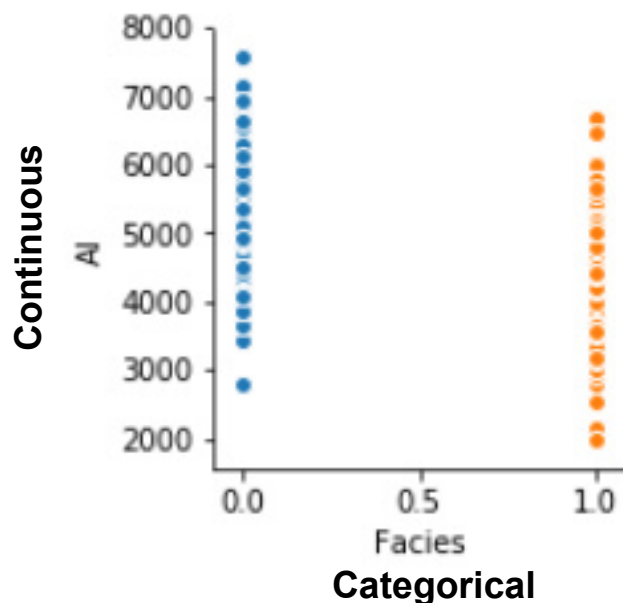
- Linear / Nonlinear – shape of the conditional expectation $Y | X$
- Homoscedastic / Heteroscedastic – conditional variance of $Y | X$

Relationships Between Variables



Multivariate Visualization

- Examples of bivariate structures



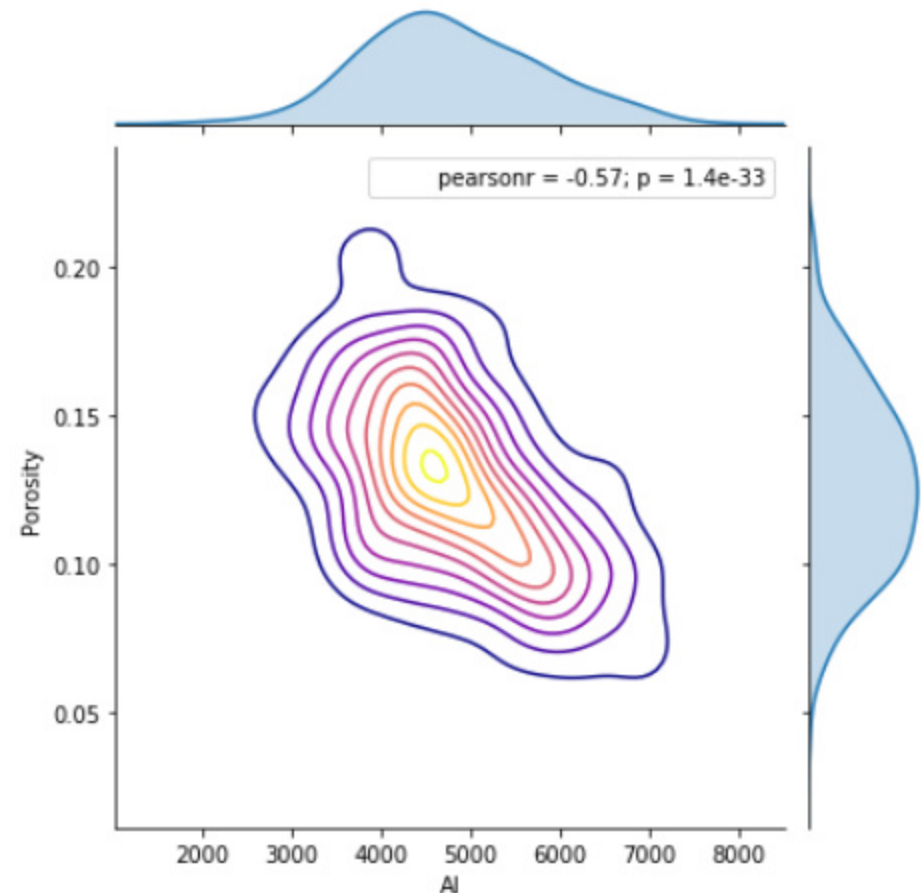
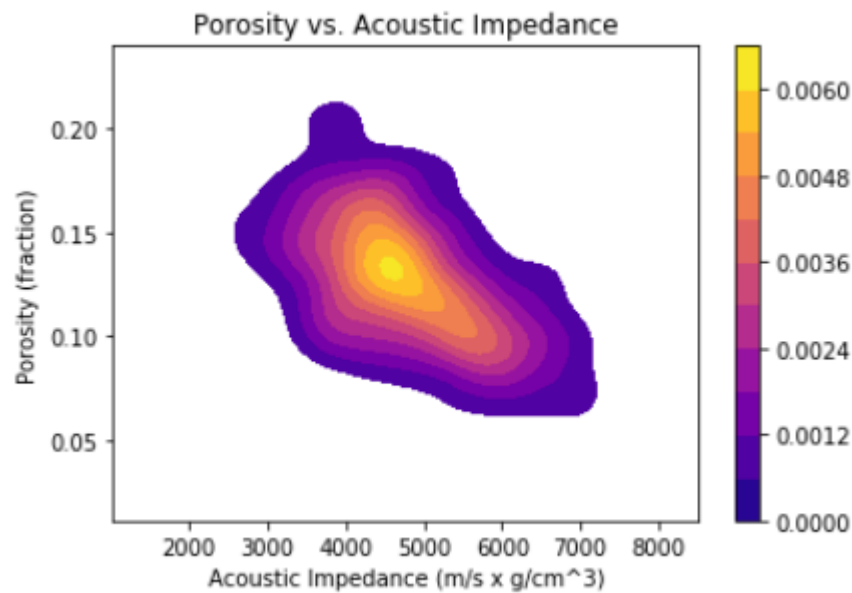
- Categorical variables only have a specified number of possible outcomes, continuous takes on a range of possible outcomes.
- Constraints – specific combinations of variables are not possible.

Relationships Between Variables



Multivariate Visualization

- Advanced Plots – Beyond Correlation



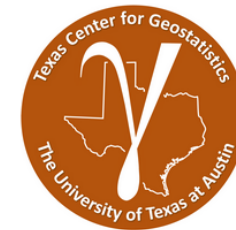
Multivariate Analysis Live Demo

Experiment with

- Multivariate Analysis

in Python Jupyter
Notebooks.

Walk through together.



Subsurface Data Analytics

Multivariate Analysis for Subsurface Data Analytics in Python

Michael Pyrcz, Associate Professor, University of Texas at Austin

[Twitter](#) | [GitHub](#) | [Website](#) | [GoogleScholar](#) | [Book](#) | [YouTube](#) | [LinkedIn](#)

PGE 383 Exercise: Multivariate Analysis for Subsurface Data Analytics in Python

Here's a simple workflow, demonstration of multivariate analysis for subsurface modeling workflows. This should help you get started with building subsurface models that integrate uncertainty in the sample statistics.

Bivariate Analysis

Understand and quantify the relationship between two variables

- example: relationship between porosity and permeability
- how can we use this relationship?

What would be the impact if we ignore this relationship and simply modeled porosity and permeability independently?

- no relationship beyond constraints at data locations
- independent away from data
- nonphysical results, unrealistic uncertainty models

Bivariate Statistics

The file is SubsurfaceDataAnalytics_Multivariate.ipynb at location <https://git.io/fjm4X>.

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- Feature Ranking

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

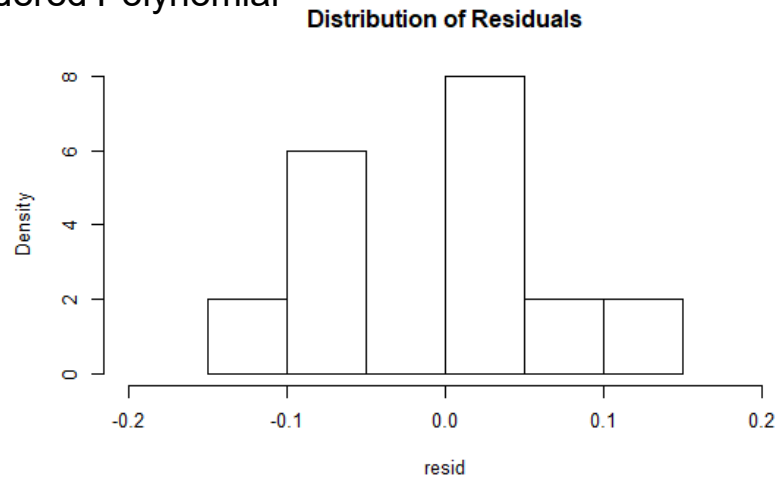
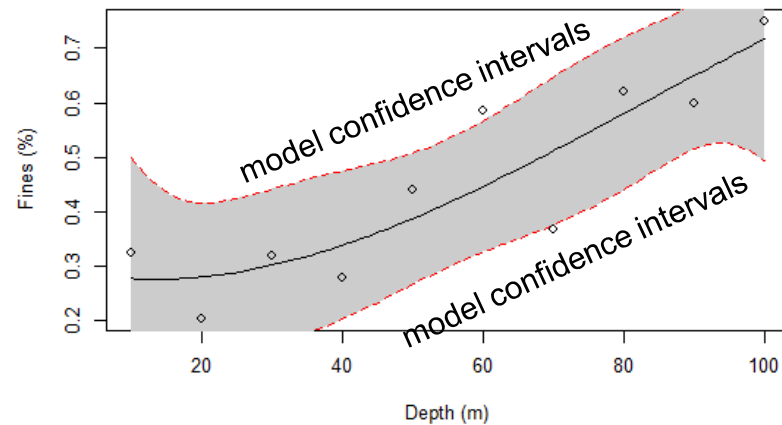
Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin

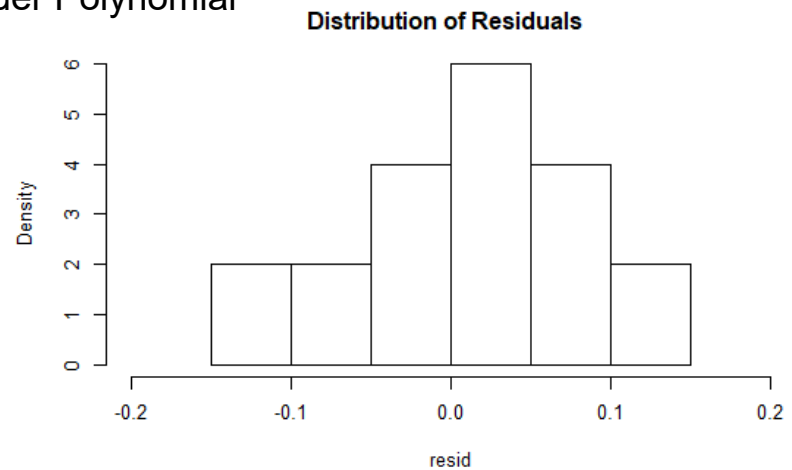
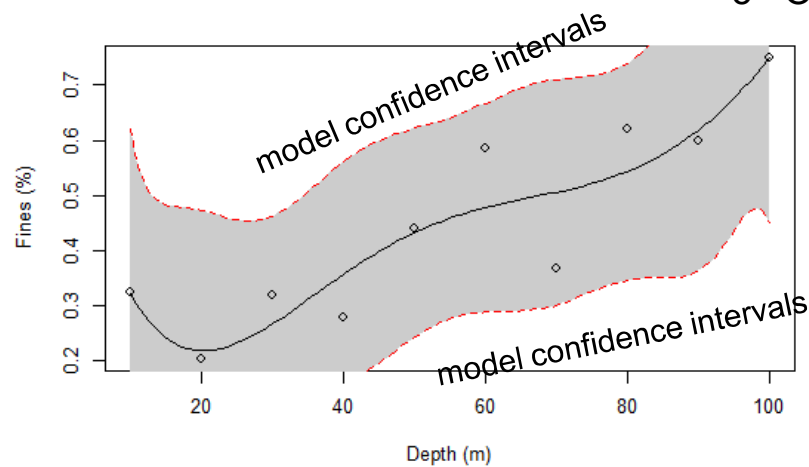
Overfitting



- Example of trend fits:
 - 3rd Ordered Polynomial



- 5th Order Polynomial



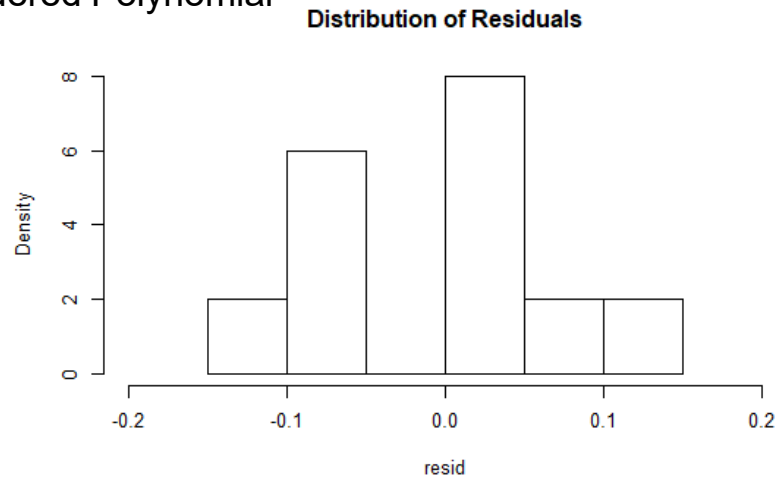
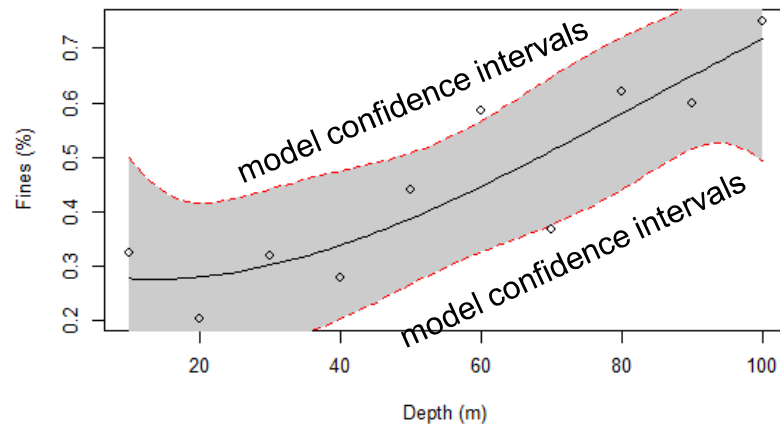
Overfit demonstration in R, code is here:
<https://github.com/GeostatsGuy/geostatsr/blob/master/overfit.R>

R code at Code/Overfit.R

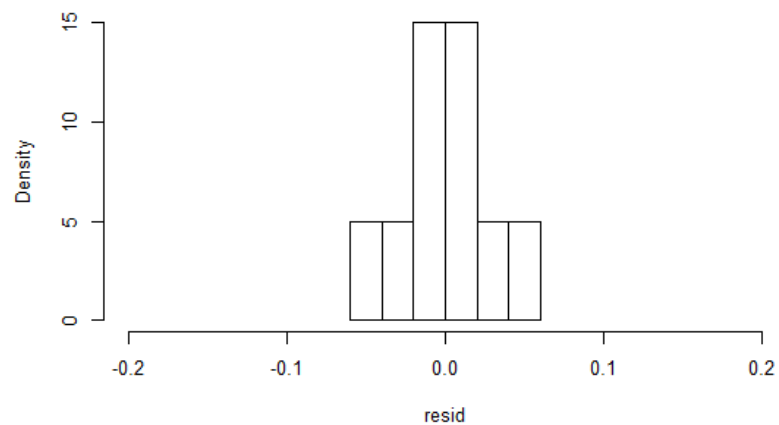
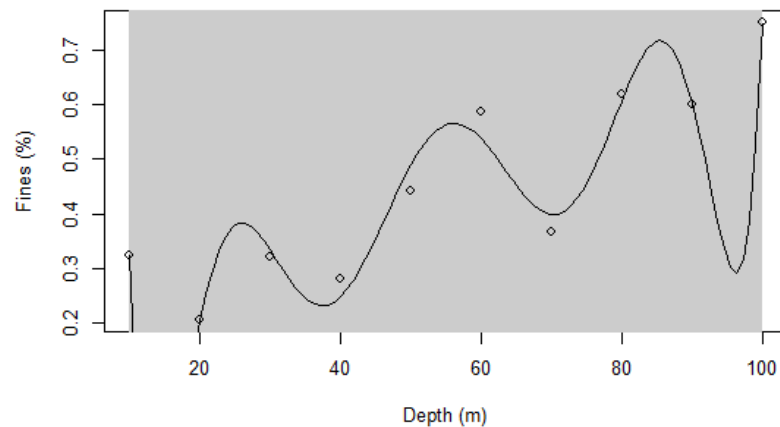
Overfitting



- Example of trend fits:
 - 3rd Ordered Polynomial



- 8th Order Polynomial



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

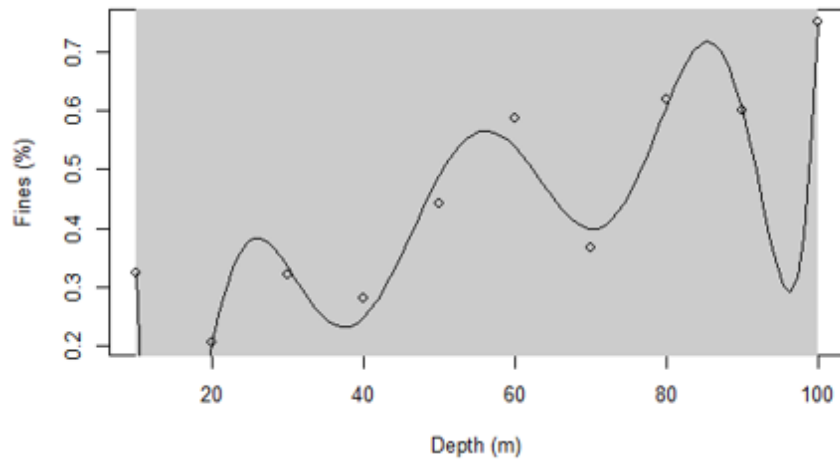
R code at Code/Overfit.R

Definition of Overfitting

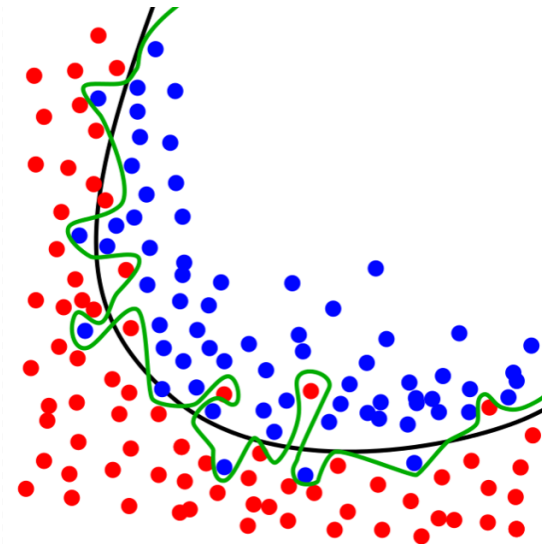


Overfit Model

- Overly complicated model to explain “idiosyncrasies” of the data, capturing data noise in the model
- Very high error away from the data / new data
- Very accurate at the data!



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>

Feature Ranking Motivation



Variable Ranking

- There are often many predictor features, input variables, available for us to work with for subsurface prediction.
- There are good reasons to be selective, throwing in every possible feature is not a good idea!
- In general, for the best prediction model, careful selection of the fewest features that provide the most amount of information is the best practice.

Feature Ranking Motivation



More Motivation to Work with Fewer Variables:

- more variables result in more **complicated workflows** that require more professional time and have increased opportunity for blunders
- higher dimensional feature sets are more difficult to **visualize**
- inclusion of **highly redundant and colinear variables** increases model instability and decreases prediction accuracy in testing
- the **risk of overfit increases** with the more variables, more complexity

What is Feature Ranking?



Here's the general types of metrics that we will consider for feature ranking:

1. Visual Inspection of Data Distributions and Scatter Plots
2. Statistical Summaries
3. Model-based
4. Recursive Feature Elimination

What is Feature Ranking?



More Motivation to Work with Fewer Variables:

- Feature ranking is a set of metrics that assign relative importance or value to each feature with respect to information contained for inference and importance in predicting a response feature.
- There are a wide variety of possible methods to accomplish this.
- My recommendation is a **wide-array** approach with multiple metric, while understanding the assumptions and limitations of each metric.

What is Feature Ranking?



Expert Knowledge:

- Also, we should not neglect expert knowledge.
- If additional information is known about physical processes, causation, reliability and availability of features this should be integrated into assigning feature ranks.
- We should be learning as we perform our analysis, testing new hypotheses.

Feature Ranking Metrics



Metric #1: Visual Inspection

- In any multivariate work we should start with the univariate analysis, summary statistics of one variable at a time. The summary statistic ranking method is qualitative, we are asking:
 - are there data issues?
 - do we trust the features? do we trust the features all equally?
 - are there issues that need to be taken care of before we develop any multivariate workflows?

Feature Ranking Metrics



Summary statistics are a critical first step in data checking.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---------|-------|-------------|-------------|------------|-------------|-------------|-------------|-------------|
| Well | 200.0 | 100.500000 | 57.879185 | 1.000000 | 50.750000 | 100.500000 | 150.250000 | 200.000000 |
| Por | 200.0 | 14.991150 | 2.971176 | 6.550000 | 12.912500 | 15.070000 | 17.402500 | 23.550000 |
| Perm | 200.0 | 4.330750 | 1.731014 | 1.130000 | 3.122500 | 4.035000 | 5.287500 | 9.870000 |
| AI | 200.0 | 2.968850 | 0.566885 | 1.280000 | 2.547500 | 2.955000 | 3.345000 | 4.630000 |
| Brittle | 200.0 | 48.161950 | 14.129455 | 10.940000 | 37.755000 | 49.510000 | 58.262500 | 84.330000 |
| TOC | 200.0 | 0.991950 | 0.478264 | 0.000000 | 0.617500 | 1.030000 | 1.350000 | 2.180000 |
| VR | 200.0 | 1.964300 | 0.300827 | 0.930000 | 1.770000 | 1.960000 | 2.142500 | 2.870000 |
| Prod | 200.0 | 3864.407081 | 1553.277558 | 839.822063 | 2686.227611 | 3604.303507 | 4752.637556 | 8590.384044 |
| const | 200.0 | 1.000000 | 0.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 |

- the number of valid (non-null) values for each feature
- general behaviors such as central tendency, mean, and dispersion, variance.
- issues with negative values, extreme values, and values that are outside the range of plausible values for each property.

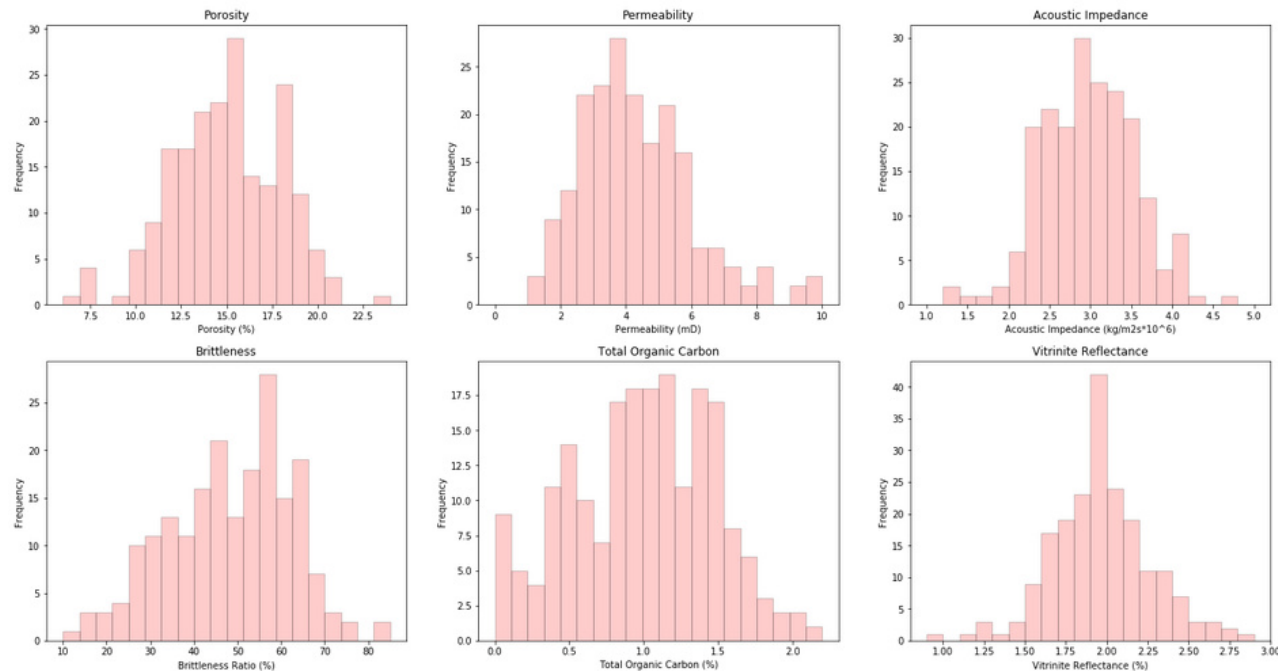
Feature Ranking Metrics



Metric #2: Univariate Distributions

- As with summary statistics, this ranking method is a qualitative check for issues with the data and to assess our confidence with each feature.
- It is better to not include a feature with low confidence of quality as it may be misleading (while adding to model complexity as discussed previously).
- Assess our ability to use methods that have distribution assumptions

Feature Ranking Metrics



The univariate distributions look good:

- there are no obvious outliers
- the permeability is positively skewed as often observed
- the corrected TOC has a small zero truncation spike, but it's reasonable
- some departure from Gaussian form, could transform

Feature Ranking Metrics



Metric #3: Two Variables at a Time (Bivariate)

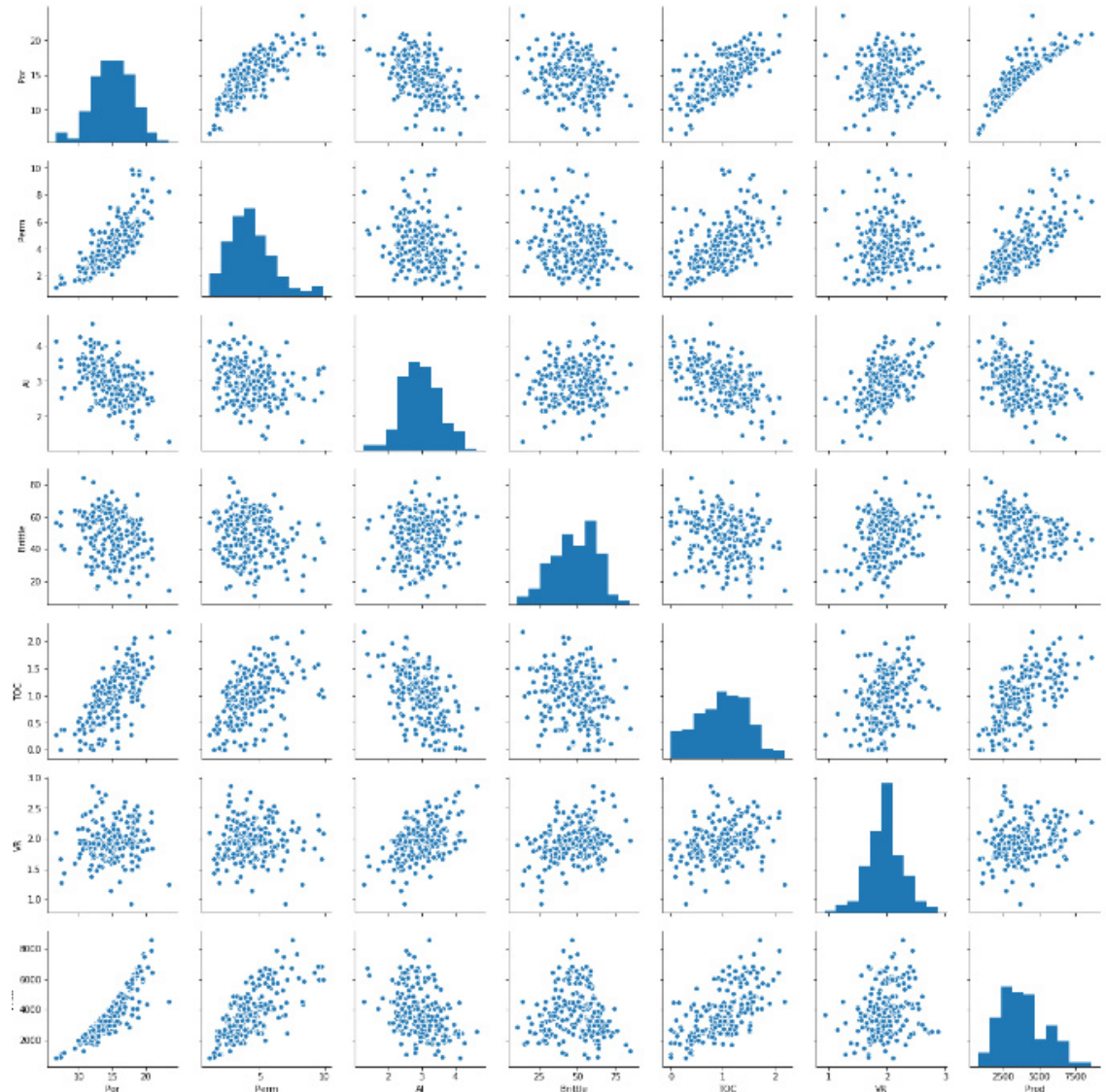
- matrix scatter plots are a very efficient method to observe the bivariate relationships between the variables.
- this is another opportunity through data visualization to identify data issues, outliers
- we can assess if we have collinearity, specifically the simpler form between two features at a time
- Bivariate Gaussian is assumed for methods such as correlation and partial correlation

Feature Ranking Metrics



How could we use this plot for variable ranking?

- variables that are closely related to each other.
- linear vs. non-linear relationships
- constraint relationships and heteroscedasticity between variables.



Feature Ranking Metrics



Metric #3: Two Variables at a Time

- bivariate visualization and analysis is not sufficient to understand all the multivariate relationships in the data
- multicollinearity includes strong linear relationships between 2 or more features.
- higher order nonlinear features, outliers and coverage?
- these may be hard to see with only bivariate plots.

Feature Ranking Metrics



Ranking Method #5 - Pairwise Correlation Coefficient

- Pairwise correlation coefficient provides a measure of the strength of the linear relationship between each predictor feature and the response feature.
- The correlation coefficient:
 - measures the linear relationship
 - removes the sensitivity to the dispersion / variance of both the predictor and response features, by normalizing by the product of the standard deviation of each feature

Bivariate Statistics

Pearson's Correlation Coefficient



- **Definition: Pearson's Product-Moment Correlation Coefficient**
 - Provides a measure of the degree of linear relationship.

$$\rho_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n-1)\sigma_x\sigma_y}, -1.0 \leq \rho_{xy} \leq 1.0$$

Correlation coefficient of variables x and y

means of variables x and y

number of data pairs

standard deviation of variables x and y

- Correlation coefficient is a standardized covariance.

$$C_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n-1)} \quad \text{Covariance} \quad \rho_{xy} = \frac{C_{xy}}{\sigma_x\sigma_y}$$

Feature Ranking Metrics



Ranking Method #6 – Rank Correlation Coefficient

- The rank correlation coefficient applies the rank transform to the data prior to calculating the correlation coefficient. To calculate the rank transform simply replace the data values with the ranks, where n is the maximum value and 1 is the minimum value.
- The rank correlation:
 - measures the monotonic relationship, relaxes the linear assumption
 - removes the sensitivity to the dispersion / variance of both the predictor and response, by normalizing by the product of the standard deviation of each.

Bivariate Statistics

Spearman's Rank Correlation Coefficient



- **Definition: Spearman's Rank Correlation Coefficient**
 - Provides a measure of the degree of monotonic relationship.

$$\rho_{R_x, R_y} = \frac{\sum_{i=1}^n (R_{x_i} - \bar{R}_x)(R_{y_i} - \bar{R}_y)}{(n-1)\sigma_{R_x}\sigma_{R_y}}, -1.0 \leq \rho_{xy} \leq 1.0$$

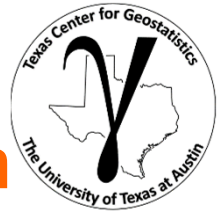
Diagram illustrating the components of the Spearman's Rank Correlation Coefficient formula:

- ρ_{R_x, R_y} : Rank correlation coefficient of variables x and y
- $\sum_{i=1}^n$: number of data pairs
- \bar{R}_x and \bar{R}_y : means of rank transform of variables x and y
- σ_{R_x} and σ_{R_y} : standard deviation of Rank transform of variables x and y

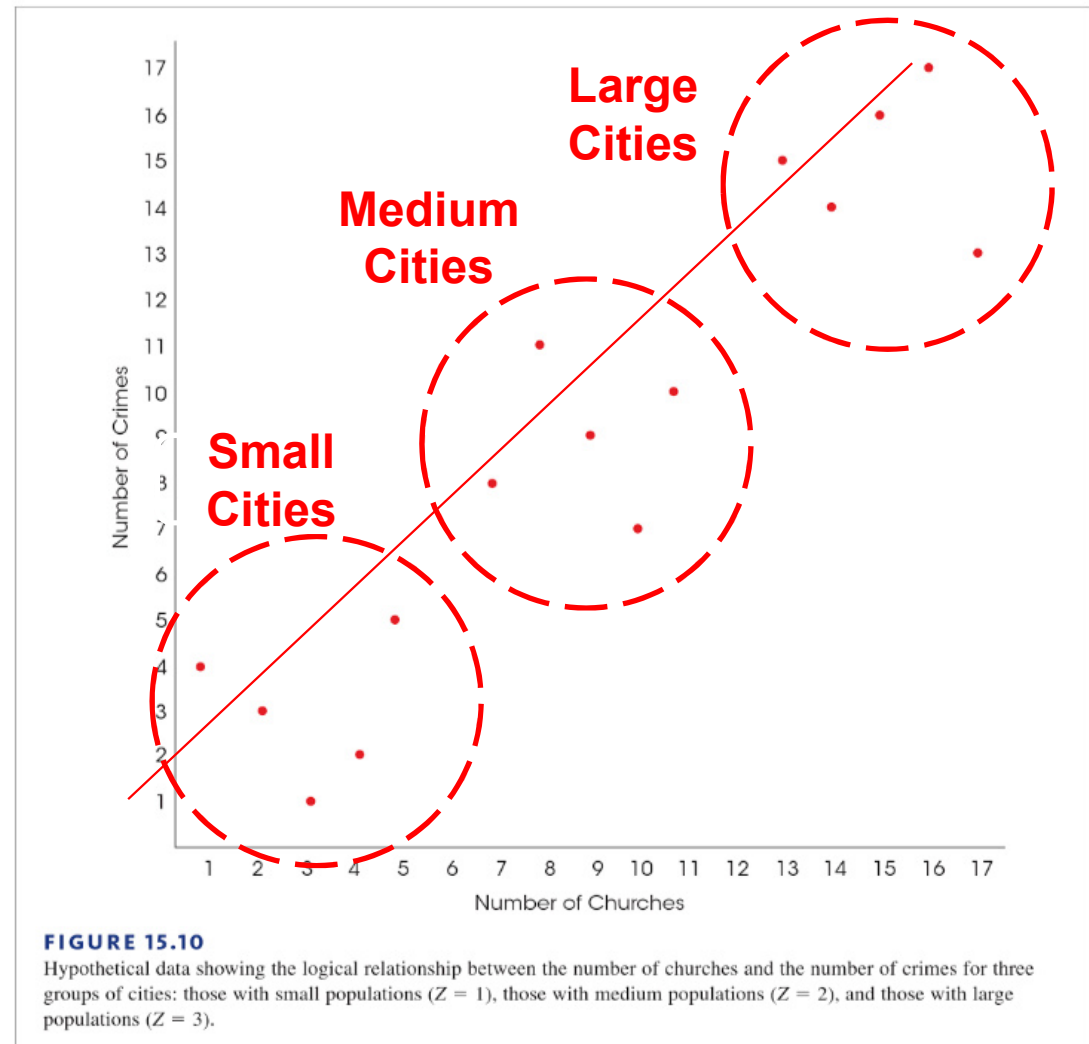
- Rank transform, e.g. R_{x_i} , sort the data in ascending order and replace the data with the index, $i = 1, \dots, n$.
- Spearman's rank correlation coefficient is more robust in the presence of outliers and some nonlinear features than the Pearson's correlation coefficient

Bivariate Statistics

Correlation and Causation



- Correlation does not imply causation!
 - Population was not controlled!
 - For each size of city the correlation is nearly zero.

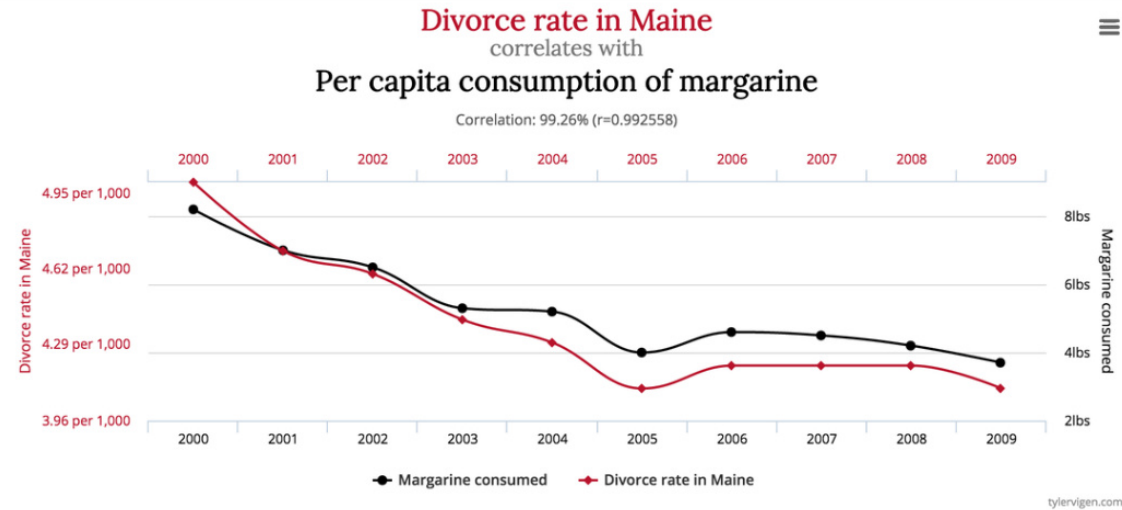


Bivariate Statistics

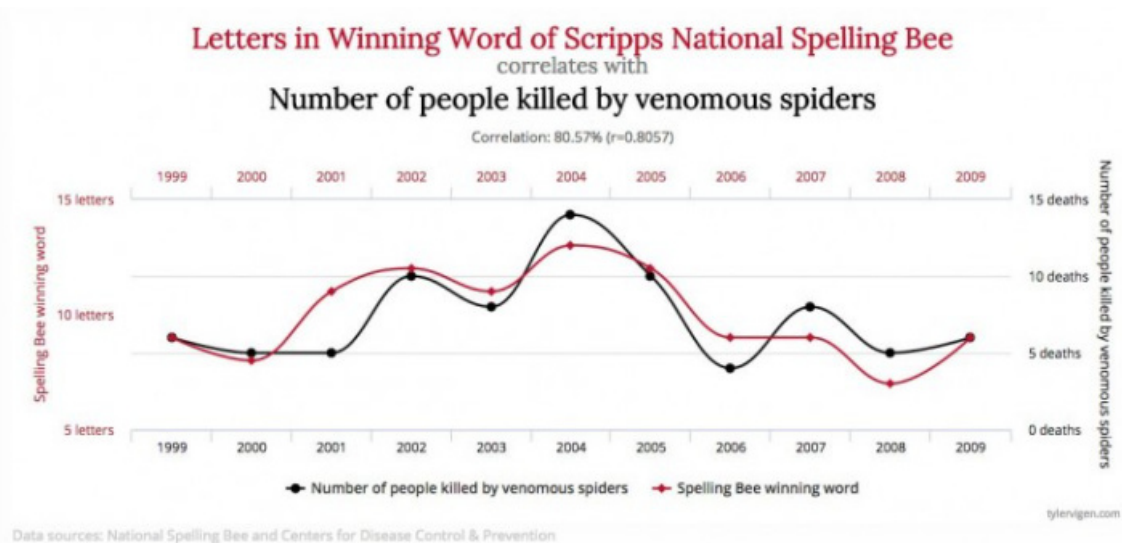
Comical Examples of Correlation and Causation



Margarine causes divorce?
or **divorce causes margarine?**



Spiders killing people causes longer words in spelling bees?
or **longer words in spelling bees causes venomous spiders to kill people?**



Feature Ranking Metrics



Ranking Method #7 – Partial Correlation Coefficient

This is a linear correlation coefficient that controls for the effects all the remaining variables

- $\rho_{XY.Z}$ and is the partial correlation between X and Y after controlling for Z .
1. perform linear, least-squares regression to predict X from $Z_{1,...,m-2}$.
 2. calculate the residuals in Step #1, $X - X^*$
 3. perform linear, least-squares regression to predict Y from $Z_{1,...,m-2}$.
 4. calculate the residuals in Step #1, $Y - Y^*$
 5. calculate the correlation coefficient, $\rho_{XY.Z} = \rho_{X - X^*, Y - Y^*}$

Feature Ranking Metrics



Ranking Method #7 – Partial Correlation Coefficient

The partial correlation, provides a measure of the linear relationship between X and Y while controlling for the effect of Z other features on both, X and Y

To use this method we must assume:

- two variables to compare, X and Y
- other variables to control, $Z_{1,...,m-2}$.
- linear relationships between all variables
- no significant outliers
- approximately bivariate normality between the variables

We are in pretty good shape, but we have some departures from bivariate normality.

- We apply a Gaussian transform in the demonstration

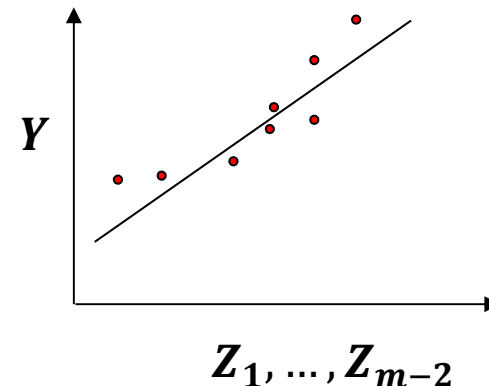
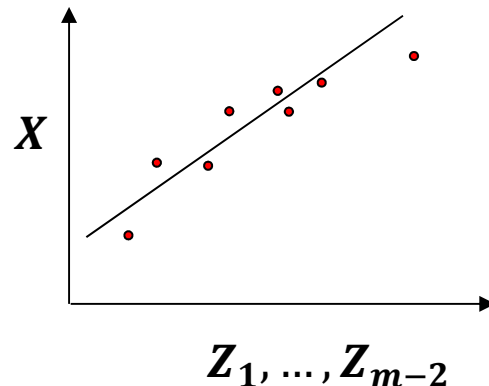
Bivariate Statistics

Partial Correlation



A method to calculate the correlation between X and Y after controlling for the influence of Z_1, \dots, Z_{m-2} other features on both X and Y .

1. perform linear, least-squares regression to predict X from Z_1, \dots, Z_{m-2} . X is regressed on the predictors to calculate the estimate, X^*
2. perform linear, least-squares regression to predict Y from Z_1, \dots, Z_{m-2} . Y is regressed on the predictors to calculate the estimate, Y^*



Bivariate Statistics

Partial Correlation



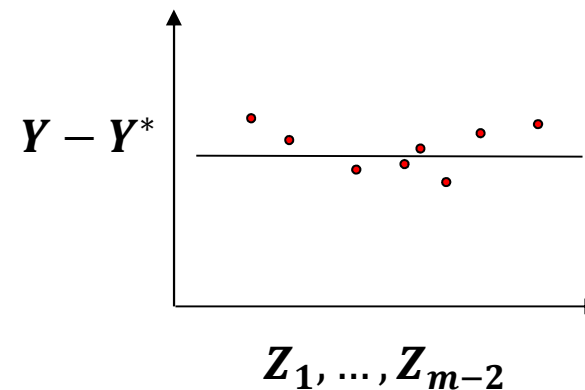
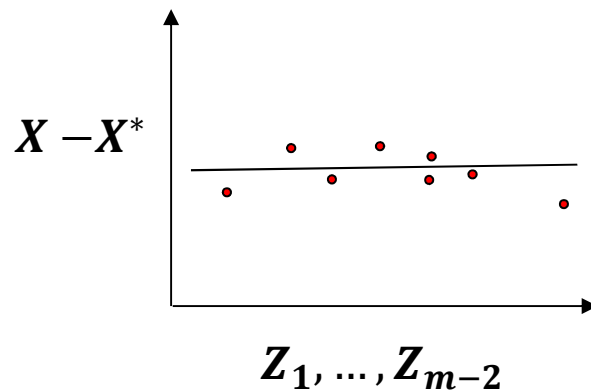
A method to calculate the correlation between X and Y after controlling for the influence of Z_1, \dots, Z_{m-2} other features on both X and Y .

3. calculate the residuals in Step #1

$$X - X^*, \text{ where } X^* = f(Z_1, \dots, Z_{m-2})$$

4. calculate the residuals in Step #1,

$$Y - Y^*, \text{ where } Y^* = f(Z_1, \dots, Z_{m-2})$$



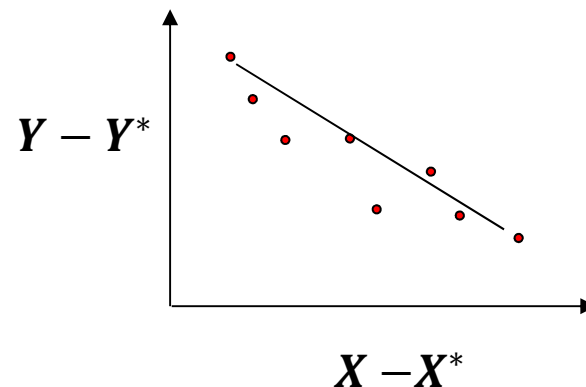
Bivariate Statistics

Partial Correlation



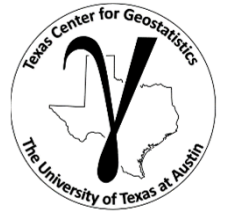
A method to calculate the correlation between X and Y after controlling for the influence of Z_1, \dots, Z_{m-2} other features on both X and Y .

5. calculate the correlation coefficient between the residuals from Steps #3 and #4, $\rho_{X-X^*, Y-Y^*}$



The partial correlation, provides a measure of the linear relationship between X and Y while controlling for the effect of Z_1, \dots, Z_{m-2} other features on both, X and Y .

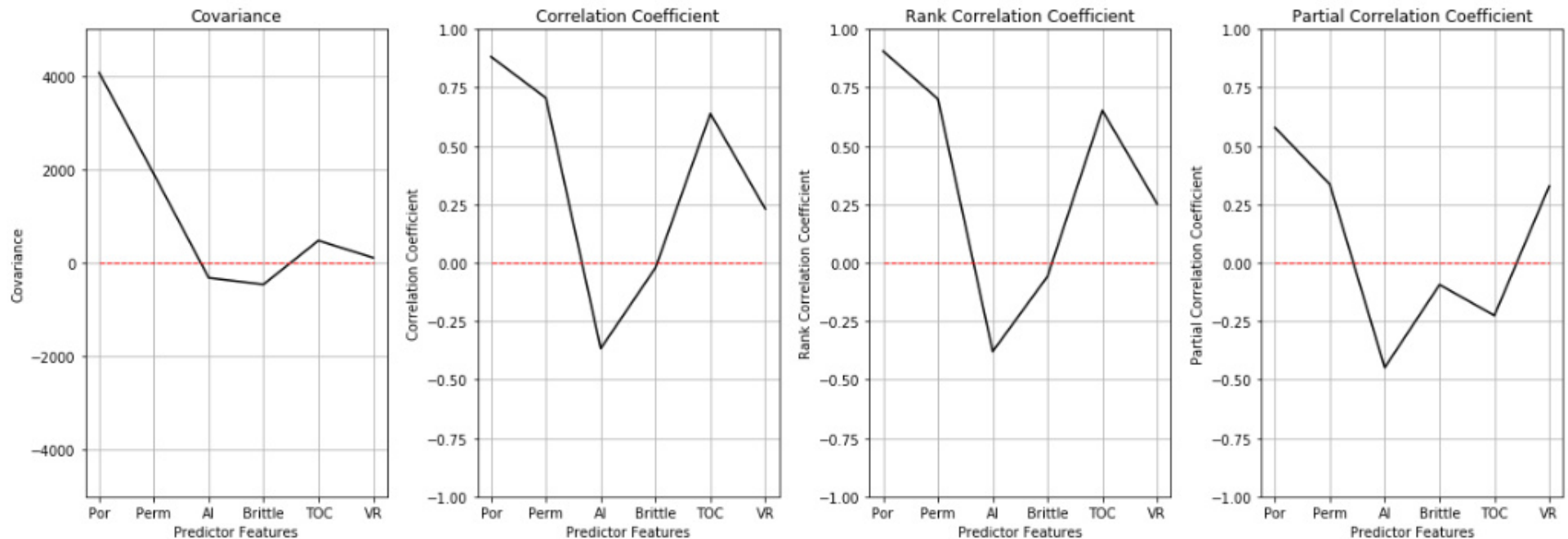
Feature Ranking Metrics



Ranking Methods #4 - #7 – Results

Are we converging on porosity, permeability and vitrinite reflectance as the most important variables with respect to linear relationships with the production?

- What about brittleness?



Feature Ranking Metrics



Ranking Method # 9 – Model-based Ranking – B coefficients

- We could also consider B coefficients from linear regression.

$$Y^* = \sum_{i=1}^m B_i X_i + c$$

- These are the linear regression coefficients without standardization of the variables.
- Sensitive to feature variance.
- We are capturing interactions between variables.

Feature Ranking Metrics



Ranking Method # 9 – Model-based Ranking – B (beta) coefficients

- We could also consider B coefficients from linear regression

$$Y^{s*} = \sum_{i=1}^m B_i X_i^s + c$$

- These are the linear regression coefficients with standardization of the variables, X_i^s and Y^{s*} (variance = 1)
- Not sensitive to variance of the features
- We are capturing interactions between variables.

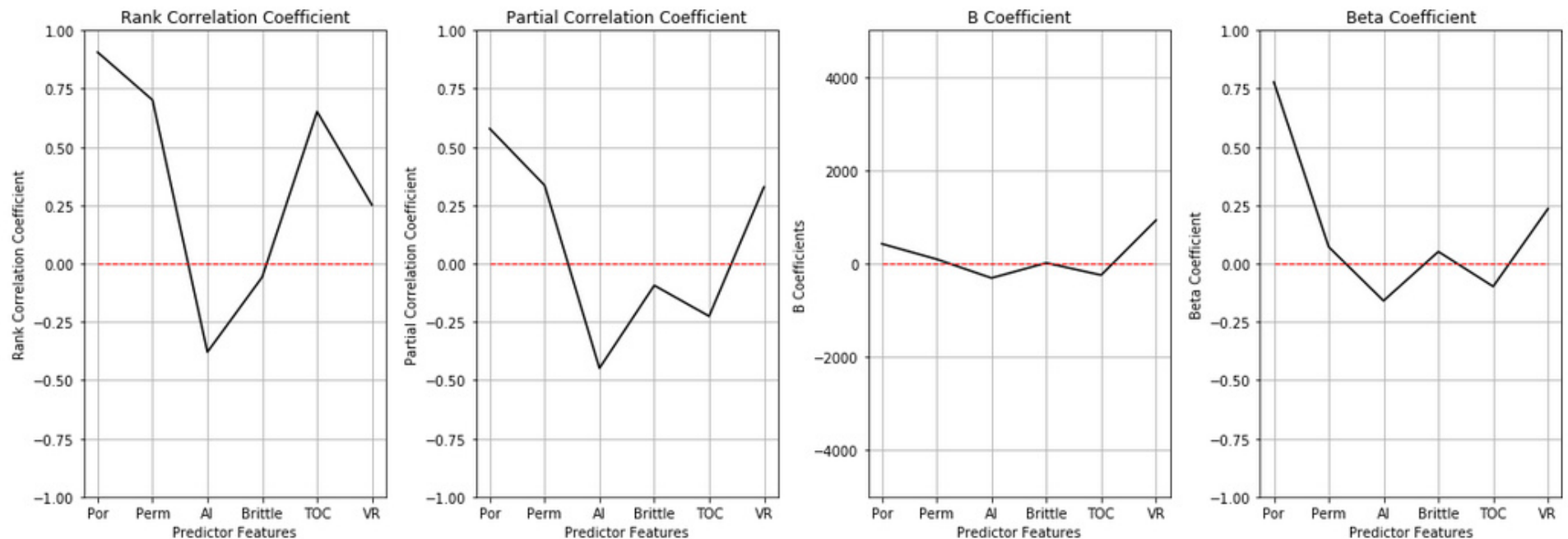
Feature Ranking Metrics



Ranking Methods #4 - #9 – Results

Now what do we see?

- Beta demotes permeability!
- Porosity, acoustic impedance and vitrinite reflectance retain high metrics



Feature Ranking Metrics



Ranking Methods #11– Recursive Feature Elimination

Recursive Feature Elimination (RFE) method works by recursively removing features and building a model with the remaining features.

- model accuracy is applied to identify which attributes (and combination of attributes) contribute the most to predicting the target attribute
- any model could be used!
- in this example the prediction model based on multilinear regression and indicate that we want to find the best feature based on recursive feature elimination.
- the method assigns rank $1, \dots, m$ for all features.

Feature Ranking Metrics



Ranking Methods #11– Recursive Feature Elimination

The recursive feature elimination method with a linear regression model provides these ranks:

1. Total Organic Carbon
2. Vitrinite Reflectance
3. Acoustic Impedance
4. Porosity
5. Permeability
6. Brittleness

A couple of the features moved from our previous assessment, but we are close. The advantages with the recursive elimination method:

- the actual model can be used in assessing feature ranks
- the ranking is based on accuracy of the estimate

Feature Ranking Metrics



Ranking Methods #11– Recursive Feature Elimination

The recursive feature elimination method with a linear regression model provides these ranks, but this method is sensitive to:

- choice of model
- training dataset

This method may be applied with cross validation (k fold iteration of training and testing datasets)

- optimize variable selection for prediction with testing data after training with training data

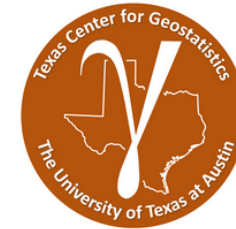
Feature Ranking Live Demo

Experiment with

- Multivariate Feature Ranking

in Python Jupyter
Notebooks.

Walk through together.



Subsurface Data Analytics

Multivariate Analysis for Subsurface Data Analytics in Python

Michael Pyrcz, Associate Professor, University of Texas at Austin

[Twitter](#) | [GitHub](#) | [Website](#) | [GoogleScholar](#) | [Book](#) | [YouTube](#) | [LinkedIn](#)

PGE 383 Exercise: Multivariate Analysis for Subsurface Data Analytics in Python

Here's a simple workflow, demonstration of multivariate analysis for subsurface modeling workflows. This should help you get started with building subsurface models that integrate uncertainty in the sample statistics.

Bivariate Analysis

Understand and quantify the relationship between two variables

- example: relationship between porosity and permeability
- how can we use this relationship?

What would be the impact if we ignore this relationship and simply modeled porosity and permeability independently?

- no relationship beyond constraints at data locations
- independent away from data
- nonphysical results, unrealistic uncertainty models

Bivariate Statistics

The file is SubsurfaceDataAnalytics_Feature_Ranking.ipynb at location <https://git.io/fjm4p>.

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- Dimension Reduction

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin

Dimensional Reduction



- Dimensional reduction transforms the data to a lower dimension
- Given features, X_1, \dots, X_m we would require $\binom{m}{2} = m(m-1)/2$ scatter plots to visualize just the two-dimensional scatter plots.
- Once we have 4 or more variables understanding our data gets very hard.
 - Recall the curse of dimensionality. It extends to visualization, not just sampling!
- One solution, is to find a good lower dimensional, p , representation of the original dimensions m
- Benefits
 - Data storage / Computational Time
 - Visualization
 - Also takes care of multicollinearity

Principal Components Analysis

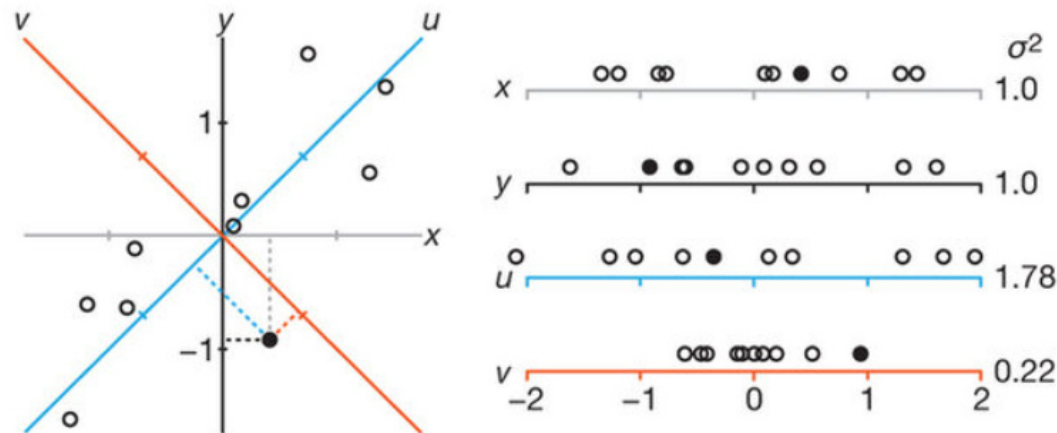


- Orthogonal Transformation
 - Convert a set of observations into a set of linearly uncorrelated variables known as principal components
- The number of principal components (k) available are $\min(n - 1, m)$
 - Limited by the variables/features, m , and the number of data
- Components are ordered
 - First component describes the largest possible variance / accounts for as much variability as possible
 - Next component describes the largest possible remaining variance
 - Up to the maximum number of principal components
- Eigen Values / Eigen Vectors
 - The Eigen values are the variance explained for each component.
 - The Eigen vectors of the data covariance matrix are the principal components and the Eigen
 - Out of scope – just making the linkage

Principal Components Analysis



- Finding the orthogonal projections in order of greatest variance described
 - Start with regular 2D, data with x and y coordinates below.

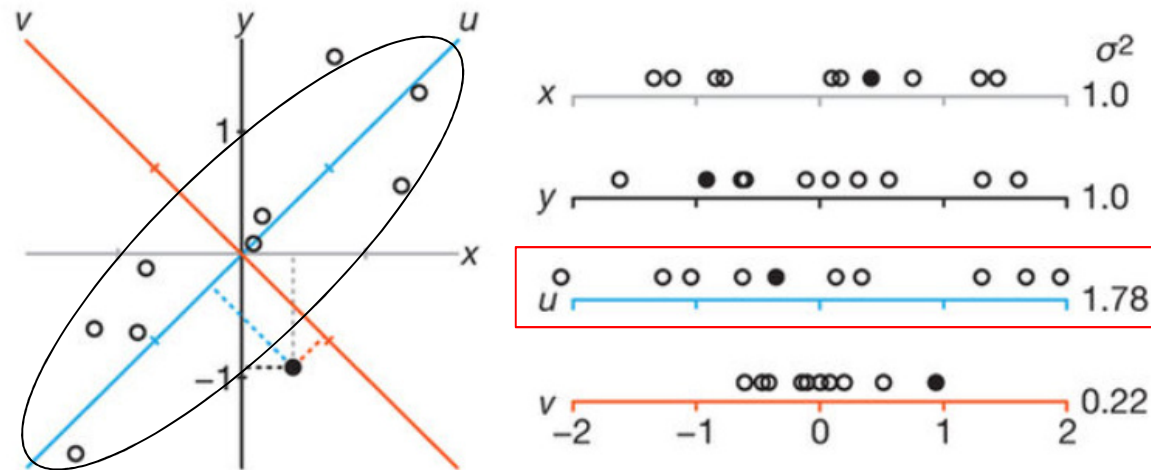


- See the projections on to x and y axes. Note the data has equal variance in x and y. If you omitted x or y from the dataset you would lose a lot of information!
- Find the rotation that would maximize the variance on the projection, u.
- The 2nd axis is given as perpendicular to the first (determined since problem is 2D).

Principal Components Analysis



- It is fitting a m-dimensional ellipsoid to the data
 - The length of each axis indicates the amount of variance described by each component
 - Omitting that axis and the associated principal component from our representation of the dataset, we would lose information proportional to the length of the axis



our data
represented
by 1 PC only

Principal Components Analysis



- Principal Component

- The first **principal component** of a set of features, X_1, \dots, X_m , is the normalized linear combination of the features:

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \dots + \phi_{m1}X_m$$

with the largest variance.

- Normalization requires:

$$\sum_{i=1}^m \phi_{i1}^2 = 1.0$$

as a result, this transform is a rotation that preserves distances.

- The values $\mathbf{V} = [\phi_{1k}, \phi_{2k} \dots \phi_{mk}] \forall j = 1, \dots, K$ components, are known as factor or component loadings
- We can calculate the first **principal component scores** (values projected onto this principal coordinate) as:

$$z_{i1} = \phi_{1,1}x_{i,1} + \phi_{2,1}x_{i,2} + \dots + \phi_{m,1}x_{i,m} \quad \text{for } i=1, \dots, n, \text{ data}$$

In matrix notation:

Factor Loadings

$$PC = V X$$

Principal Component

Centered Data Matrix

Principal Components Analysis



- Factor / Component Loadings

- Observes the groups of variables that strongly influence each principal coordinate.

$$Z_1 = \phi_{1,1}X_1 + \phi_{2,1}X_2 + \cdots + \phi_{m,1}X_m$$

1st Principal
Component

the loadings for PC1 are $\phi_{11}, \phi_{21} \dots \phi_{m1}$.

- Check if specific variables strongly influence specific principal components for the remainder. Compare them to each other.

$$Z_2 = \phi_{1,2}X_1 + \phi_{2,2}X_2 + \cdots + \phi_{m,2}X_m$$

2nd Principal
Component



$$Z_K = \phi_{1,K}X_1 + \phi_{2,K}X_2 + \cdots + \phi_{m,K}X_m$$

Kth Principal
Component

May be able to describe each principal component! E.g. for a reservoir:

- PC1 is the mainly the heterogeneity component
- PC2 is the mainly the completion component – etc.

Principal Components Analysis



- How do we do Dimensional Reduction?
 - We have converted our data set from $X_{n \times m}$ to principal components, $PC_{n \times K}$
 - If we retain all the K components then have not achieved any dimensional reduction. **We just have mixed orthogonal variables!**
 - We gain dimensional reduction by retaining only p principal components or in other words by dropping the last $K - p$ components as they describe very little of the variance.

$$PC = V X$$

$$\hat{X} = V^{-1} \cdot PC$$

Back transforming from principal components to original values.

- But since the loadings, V , are orthonormal then $V^{-1} = V^T$

$$\hat{X} = V^T \cdot PC$$

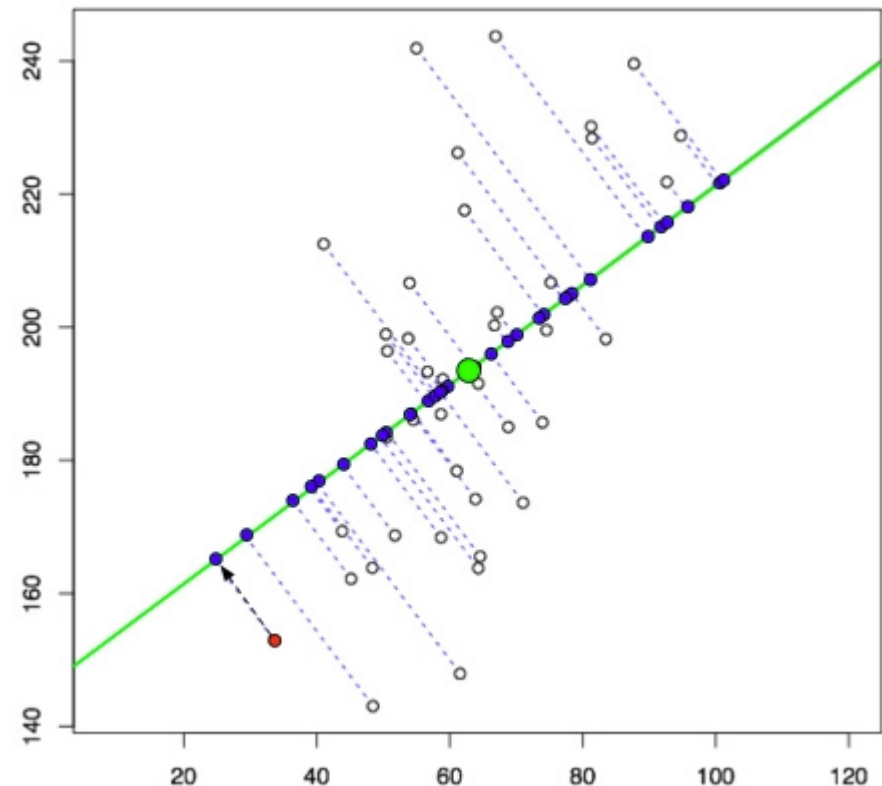
$$x_{i,j} \approx \sum_{k=1}^p z_{i,j} \phi_{j,k}$$

where $i = 1, \dots, n$ data and $j = 1, \dots, m$ variables, and p principal components (of $k = 1, \dots, K$) are retained.

Principal Components Analysis



- Graphical Representation
 - Line is the 1st principal component
 - Projection of points on line (**purple points**) are the 1st principal component scores
 - Given the problem is 2D the 2nd principal component is determined from the first (must be orthogonal)
 - If we approximated this dataset with just the 1st principal component for dimensional reduction, our approximation would be the **purple points**.
 - The first principal component maximizes the variance of the projected **purple points**.



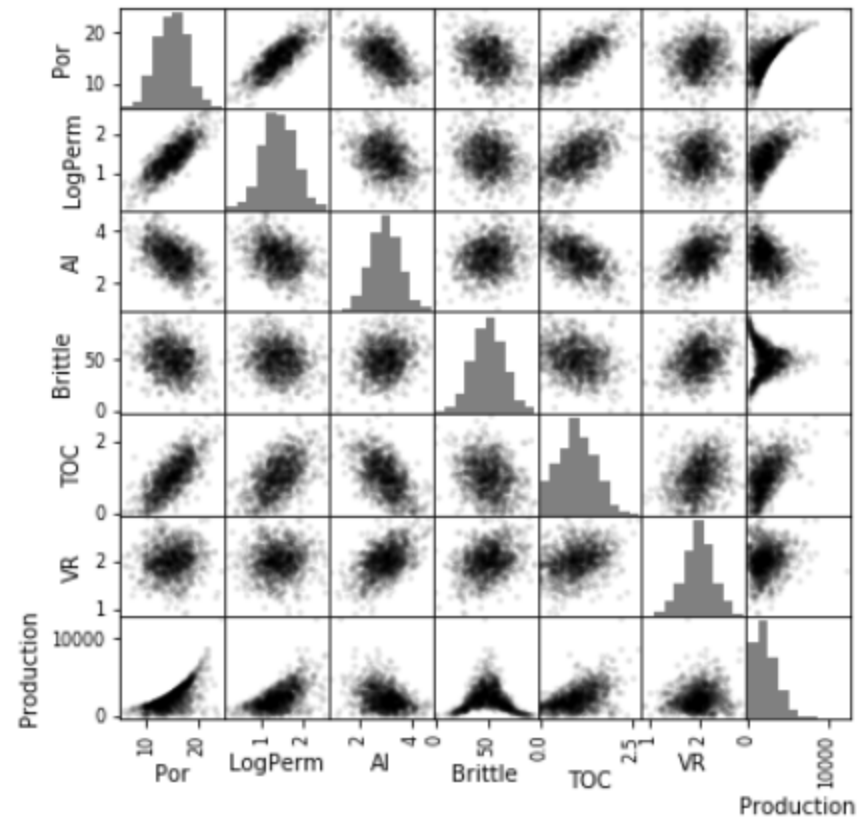
1st principal component, projects on the line are the 1st principal component scores (from <https://liorpachter.wordpress.com/2014/05/26/what-is-principal-component-analysis/>).

Principal Components Analysis Hands-on



Data Loading and Visualization

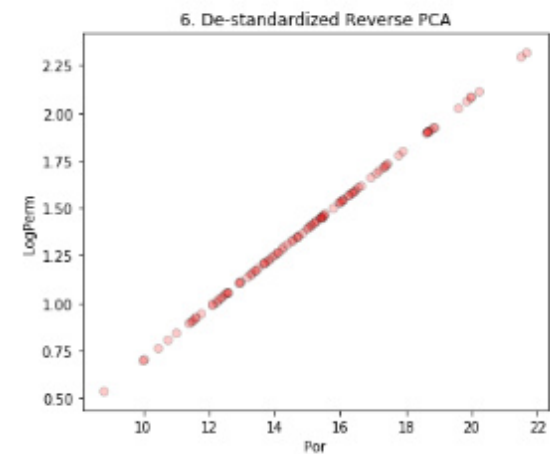
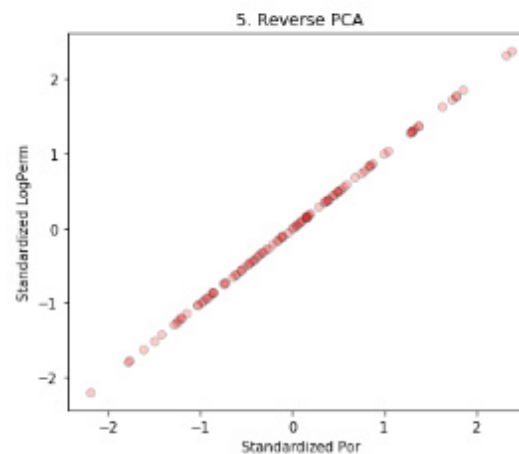
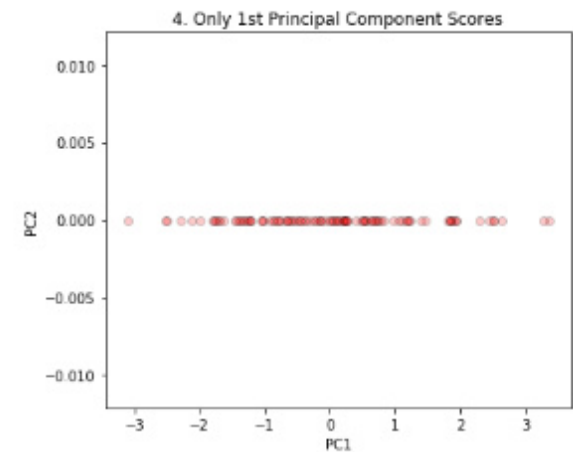
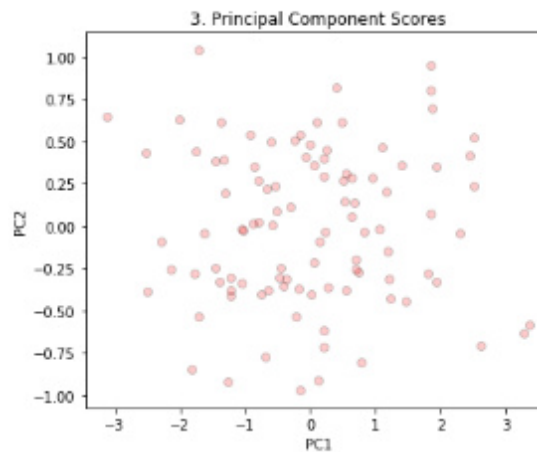
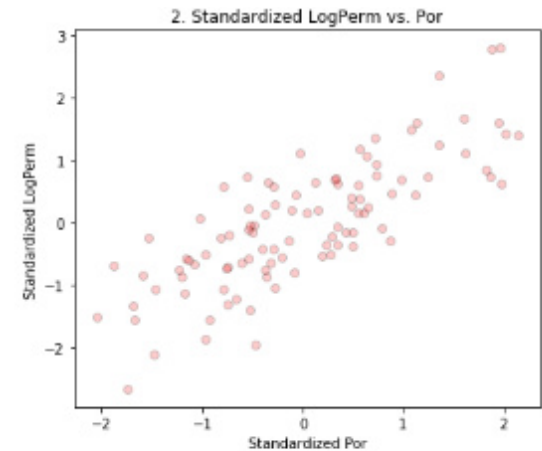
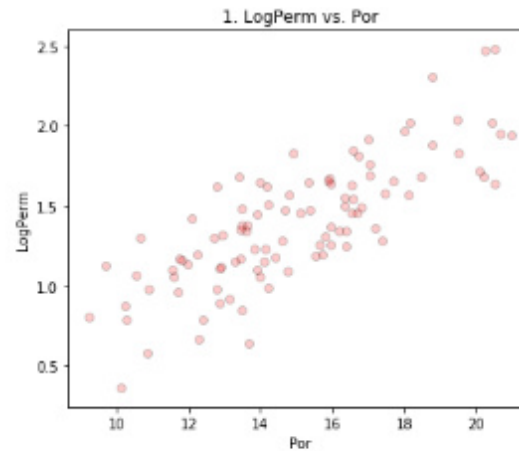
- Histograms
- Matrix scatter plot



Principal Components Analysis Live Demo

2D Workflow

1. Scatter plot
2. Standardize
3. Principal component scores
4. Retain Only First Score
5. Reverse principal component transform
6. De-standardize

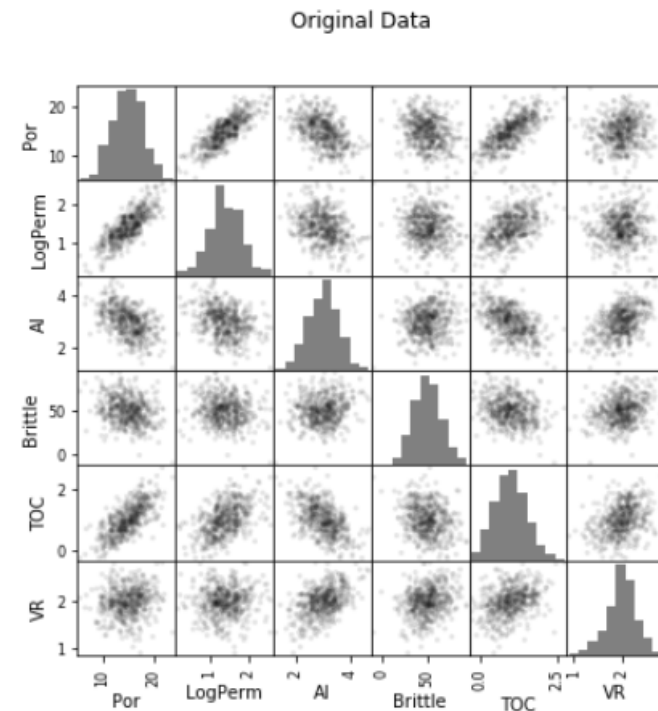
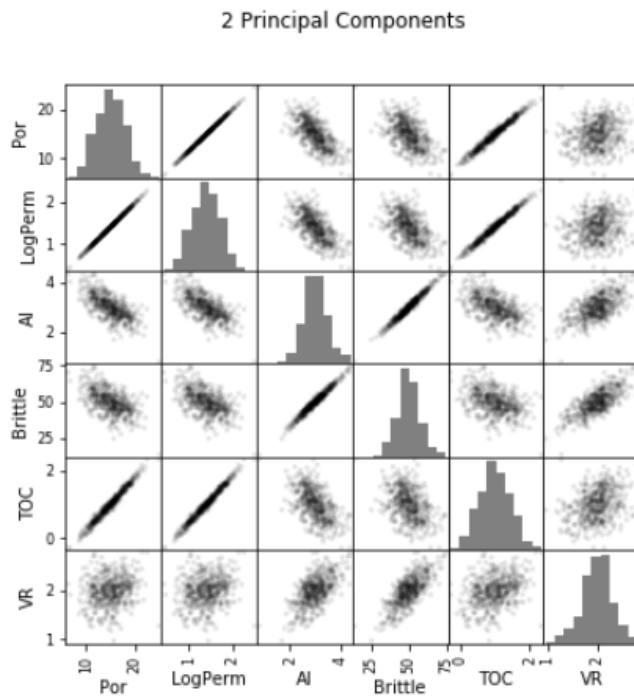


Principal Components Analysis Live Demo



Full Multivariate Dataset

Original multivariate dataset and reduced dimensional representation (2 principal components)



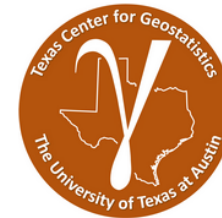
Principal Component Analysis Live Demo

Experiment with

- Principal Component Analysis

in Python Jupyter
Notebooks.

Walk through together.



Subsurface Data Analytics

Principal Component Analysis for Subsurface Data Analytics in Python

Michael Pyrcz, Associate Professor, University of Texas at Austin

[Twitter](#) | [GitHub](#) | [Website](#) | [Google Scholar](#) | [Book](#) | [YouTube](#) | [Linkedin](#)

PGE 383 Exercise: Principal Component Analysis for Subsurface Data Analytics in Python

Here's a simple workflow, demonstration of principal component analysis for subsurface modeling workflows. This should help you get started with building subsurface models that integrate uncertainty in the sample statistics.

Principal Component Analysis

Principal Component Analysis one of a variety of methods for dimensional reduction:

Dimensional reduction transforms the data to a lower dimension

- Given features, X_1, \dots, X_m , we would require $\binom{m}{2} = \frac{m(m-1)}{2}$ scatter plots to visualize just the two-dimensional scatter plots.
- Once we have 4 or more variables understanding our data gets very hard.
- Recall the curse of dimensionality, impact inference, modeling and visualization.

One solution, is to find a good lower dimensional, p , representation of the original dimensions m

Benefits of Working in a Reduced Dimensional Representation:

1. Data storage / Computational Time
2. Easier visualization
3. Also takes care of multicollinearity

Orthogonal Transformation

Convert a set of observations into a set of linearly uncorrelated variables known as principal components

- The number of principal components (k) available are $\min(n - 1, m)$
- Limited by the variables/features, m , and the number of data

Components are ordered

The file is SubsurfaceDataAnalytics_PCA.ipynb at location <https://git.io/fjmRO>.

Inference New Tools



| Topic | Application to Subsurface Modeling |
|---------------------|--|
| Inference | Methods to learn about the relationships between variables. <i>Apply inference prior to prediction to learn from our data first!</i> |
| Feature Selection | Wide array approach for feature selection to learn the value of each feature for prediction. <i>Consider correlation, partial correlation and model-based significance.</i> |
| Dimension Reduction | Reduce the problem dimensionality to improve modeling robustness in the presence of sparse data. <i>Apply dimensional reduction methods to avoid overfit and blunders.</i> |

Subsurface Data Analytics in Python

Data Analytics - Inference



Lecture outline . . .

- Machine Learning / Inference
- Multivariate Analysis
- Dimension Reduction

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

Conclusions

Instructor: Michael Pyrcz, the University of Texas at Austin