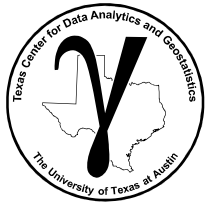
Geostatistics and Machine Learning

Dimensionality Reduction



- Curse of Dimensionality
- Dimensionality Reduction
- Principal Component Analysis

Introduction

Data Analytics

Inferential Methods

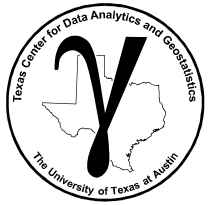
Predictive Methods

Advanced Methods

Conclusions

Michael Pyrcz, The University of Texas at Austin

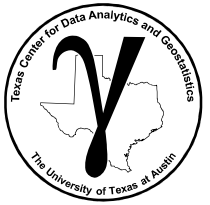
Lecture Goals



- Motivate Dimensionality Reduction
- Explain strategies of feature selection and feature projection
- Demonstrate Principal Component Analysis

Geostatistics and Machine Learning

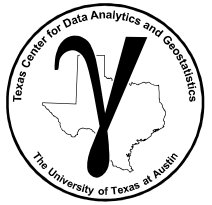
Dimensionality Reduction



- **Curse of Dimensionality**

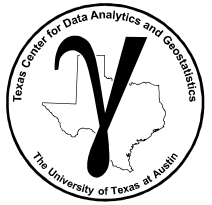
Michael Pyrcz, The University of Texas at Austin

Multivariate



- One of the definitions of Big Data is variety
  - This suggests massively multivariate datasets
- Traditional reservoir modeling workflows were bivariate
  - Facies, then porosity in facies and permeability constrained to porosity
  - The most complicated simulation is permeability accounting for the joint porosity simulated realization
- Unconventionals, and Whole Earth Models
  - Require inclusion many more variables
  - We need to model facies, porosity, geomechanical properties, geophysical properties, total organic carbon, maturity etc.
- When working with Multivariate it is very challenging:
  - Visualize
  - Detect relationships and patterns

Curse of Dimensionality

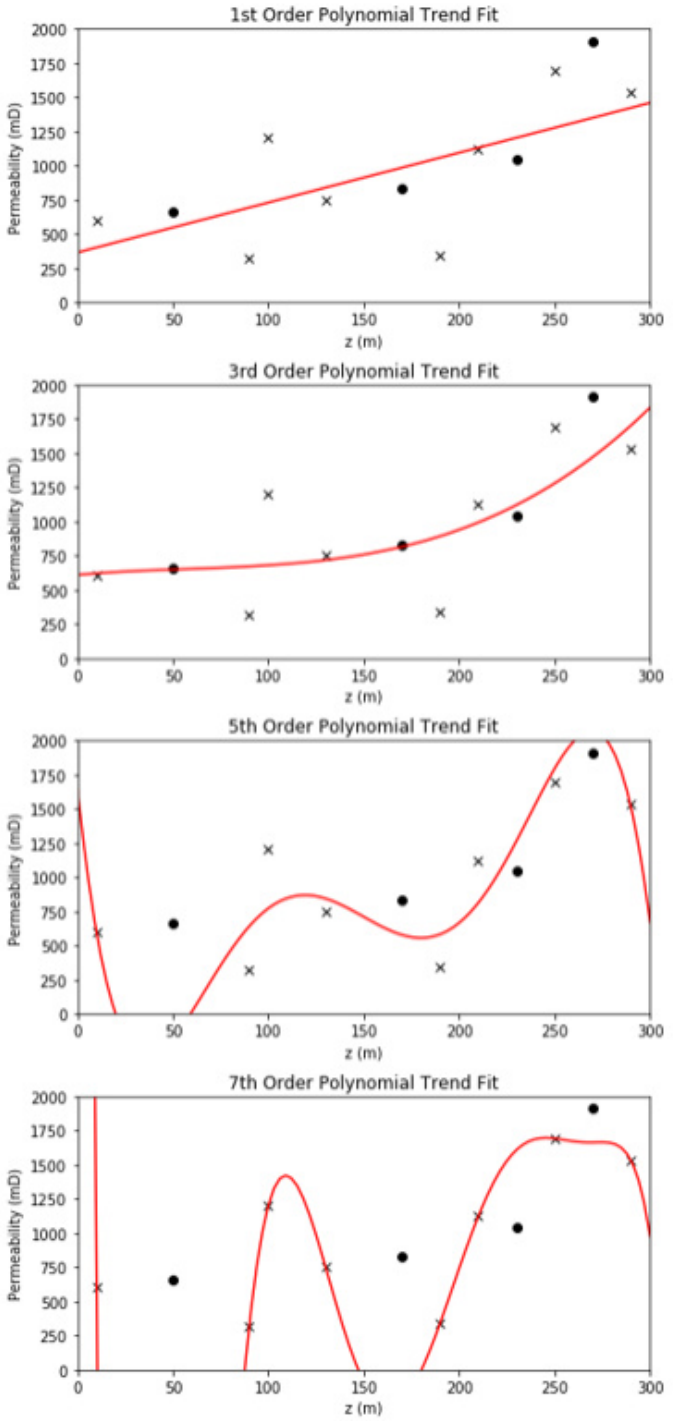


- **Working with more features / variables is harder!**

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

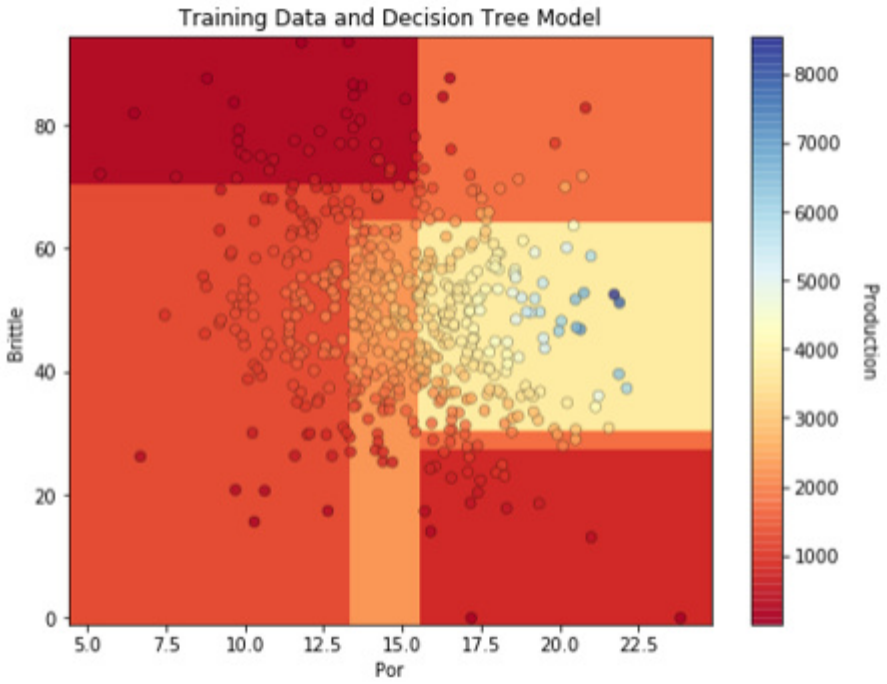
Visualization

- **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 defendable given the data?



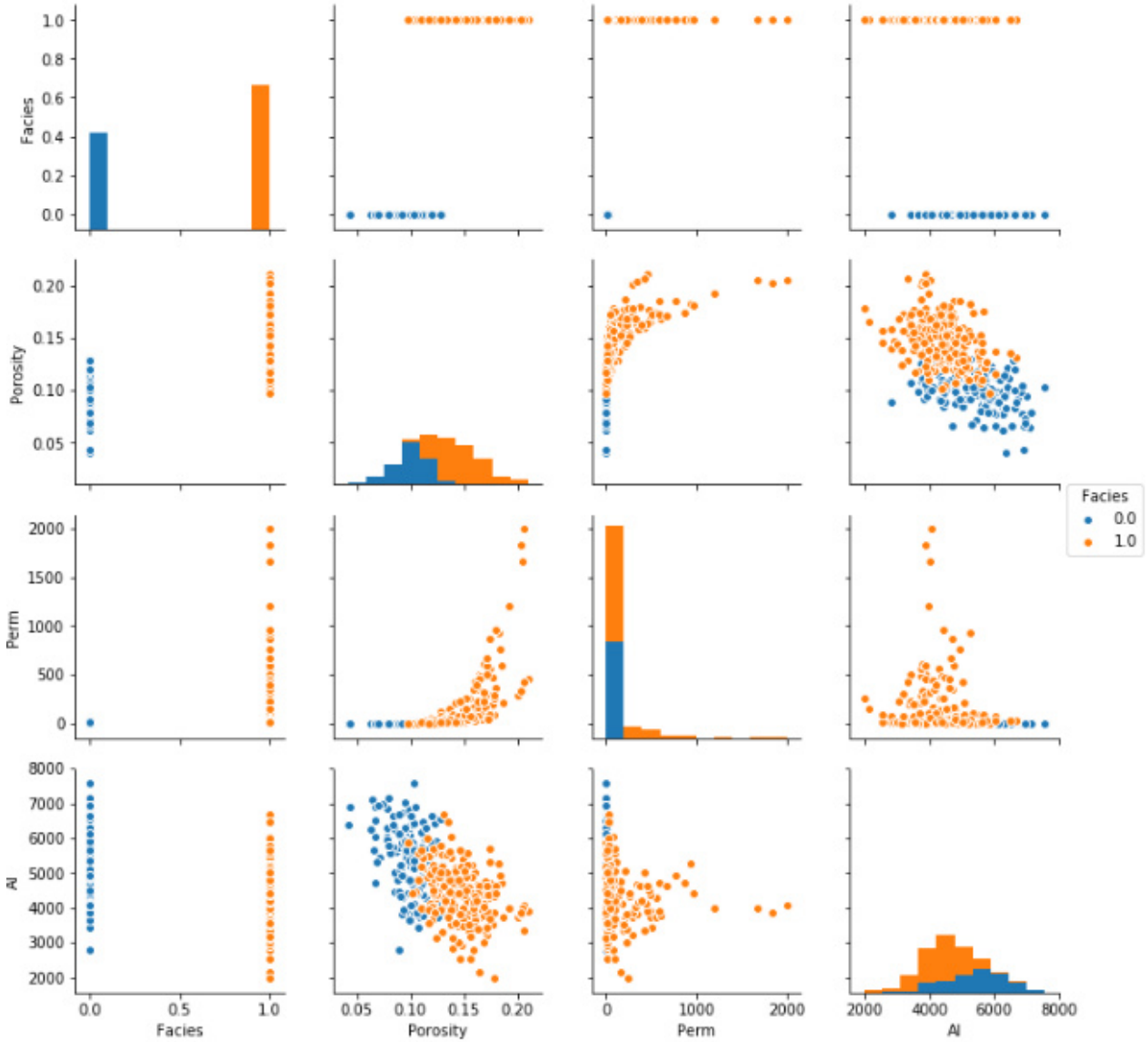
Visualization

- **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 defendable given the data?



Visualization

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

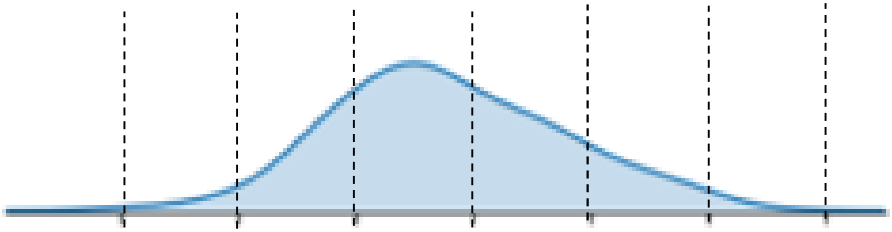


Curse of Dimensionality

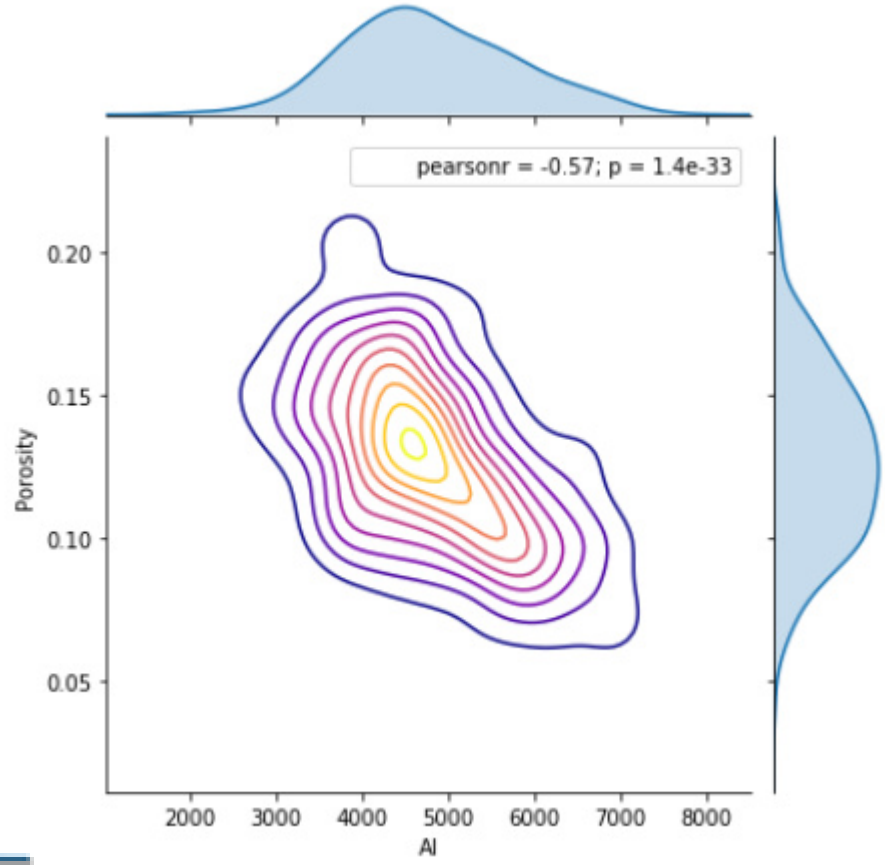
- **Consider any joint probability:**

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

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



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



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

Curse of Dimensionality

- **Consider any joint probability:**

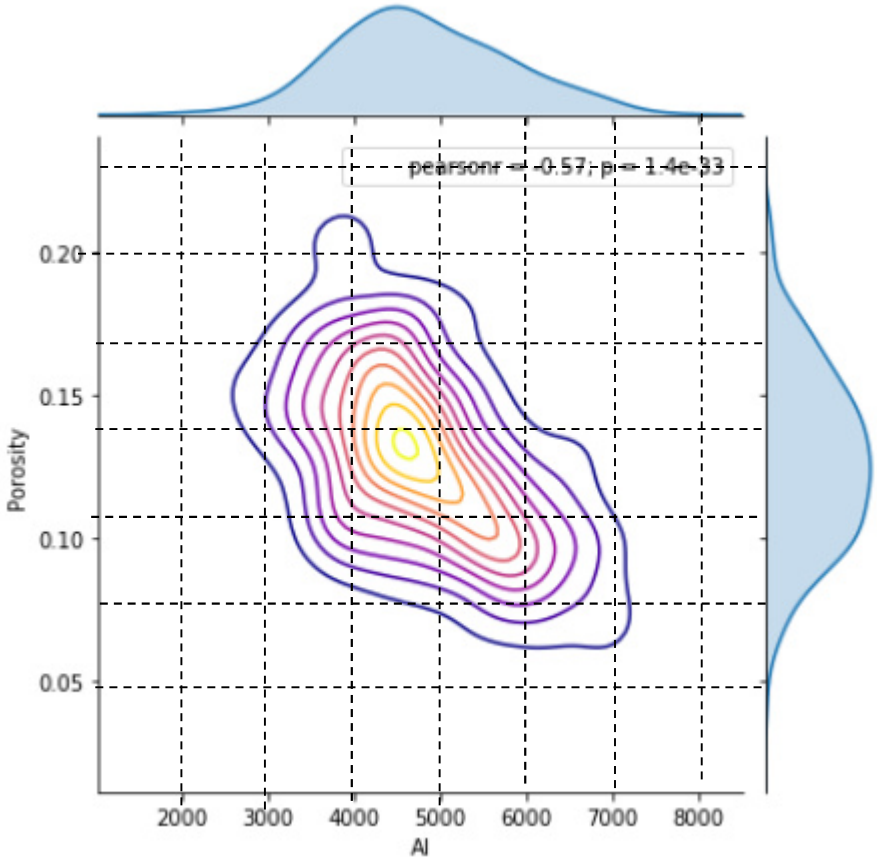
  $P(X_1 \cap, \ldots, \cap X_m)$ the joint probability of $X_1, \ldots, X_m$

- Now move to 2 features (m=2)

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

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

- This is optimistic, as it assumes uniform sampling

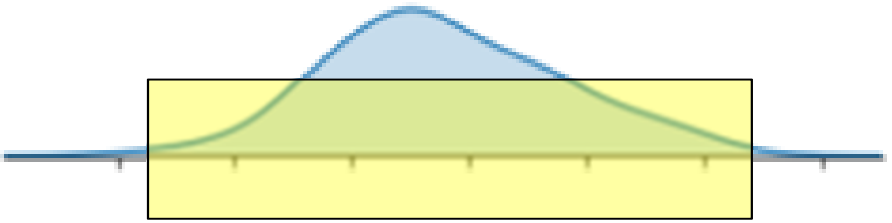


In each bin we are estimating a probability!
10 data in each bin = 640 data?

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:

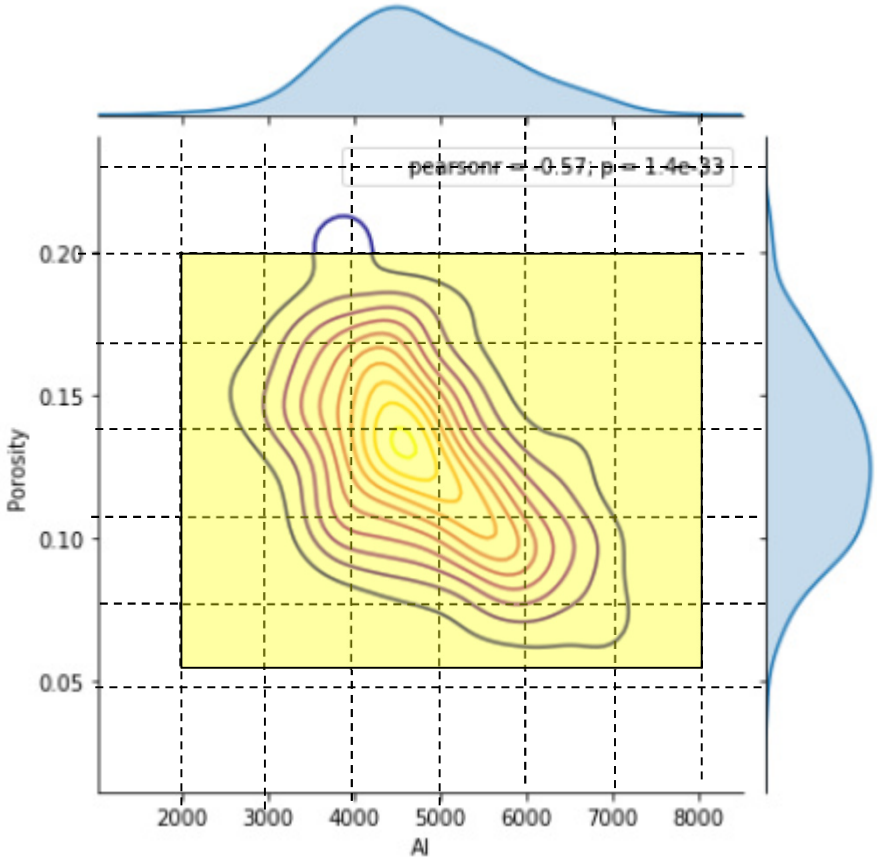
- Now let's move to 2 features, each with 80% coverage
- How much of the solution space is covered?

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

- Even with exponential increase in number of data:

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

coverage is decreasing as we increase the number of features!



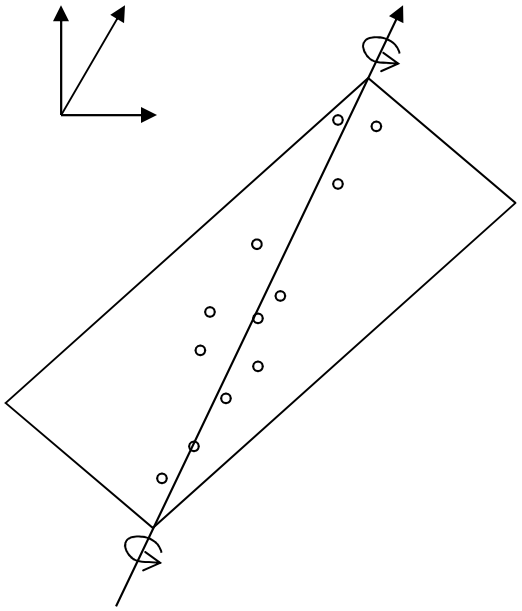
Multicollinearity
Feature Redundancy

"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

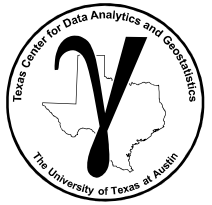
"In statistics, **multicollinearity** (also collinearity) is a phenomenon in which one predictor variable in a **multiple regression** model can be linearly predicted from the others with a substantial degree of accuracy."

- Wikipedia



It is like fitting a plane to a line!

Motivation for Dimensionality Reduction

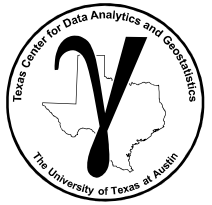


We get a better model with fewer, informative features than

'Throwing everything and the kitchen sink into the model!'

Fewer features for models are simpler, faster, easier to visualize and less likely overfit.

Geostatistics and Machine Learning
Dimensionality Reduction



- **Dimensionality Reduction**

Introduction

Data Analytics

Inferential Methods

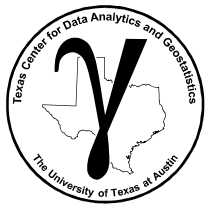
Predictive Methods

Advanced Methods

Conclusions

Michael Pyrcz, The University of Texas at Austin

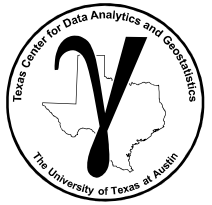
Dimensionality Reduction



- Motivated by the curse of dimensionality and multicollinearity
- Known as dimension reduction or dimensionality reduction
- Applied in statistics, machine learning and information theory
- Multiple strategies:

1. Features Selection – find the subset of original features that are most important for the problem (big hitters).
2. Feature Projection – transform the data from a higher to lower dimensional space

Feature Selection Recall

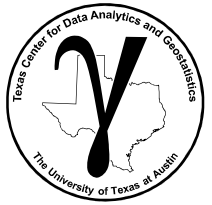


Consider a wide-array approach to assess variable importance.

- 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

Feature Selection Metrics Recall



Expert Knowledge:

- Also, we should not neglect expert knowledge.

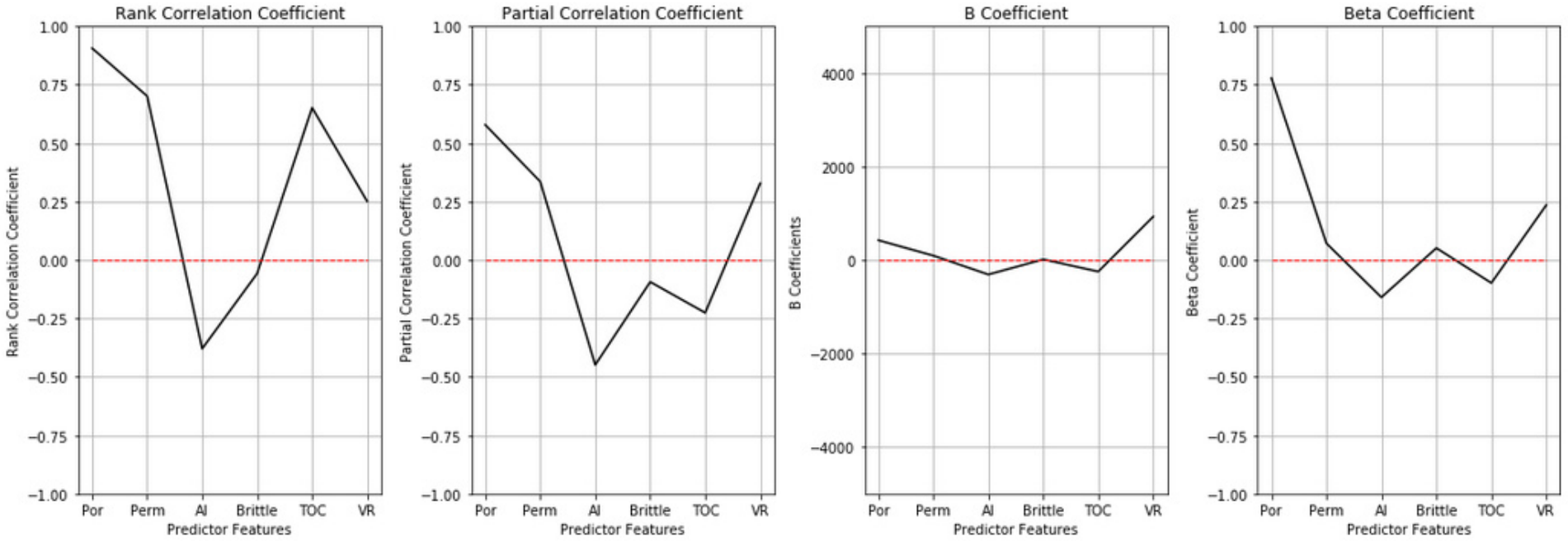
- If additional information is known about physical processes, causation, reliability and availability of features this should be integrated into assigning feature ranks.

- We should be learning as we perform our analysis, testing new hypotheses.

Feature Selection Metrics Recall

Example of Ranking Metrics Applied to Unconventionals

- Provides evidence to support feature selection
- Beta demotes permeability!
- Porosity, acoustic impedance and vitrinite reflectance retain high metrics



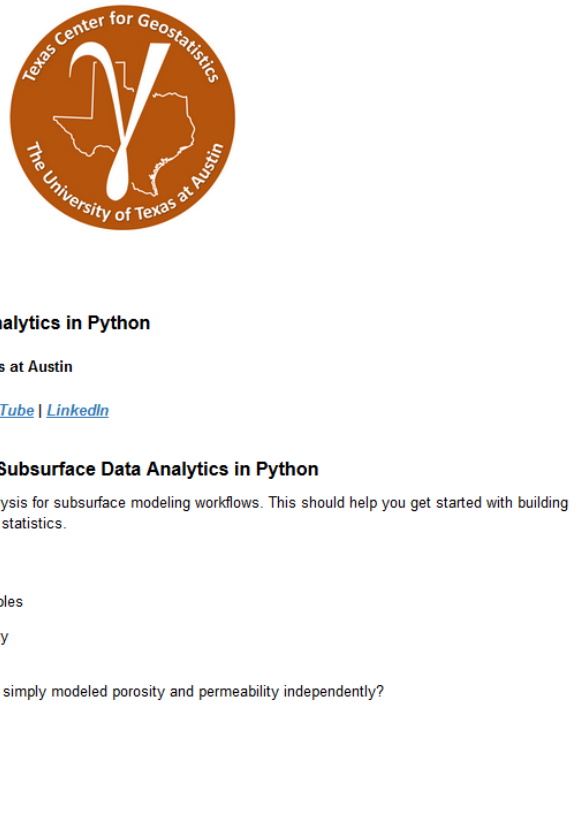
Feature Ranking
Live Demo Recall

Experiment with

- Multivariate Feature Ranking

in Python Jupyter Notebooks.

We will not cover this again, for reference.



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?

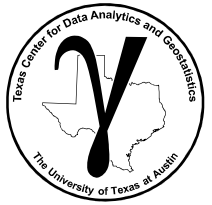
What would be the impact if we ignore this relationship and simply modeled porosity and permeability independently?

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

Bivariate Statistics

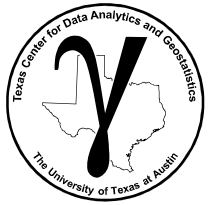
The file is SubsurfaceDataAnalytics_Feature_Ranking.ipynb at location https://git.io/fjm4p.

Feature Projection



- Dimensionality reduction by feature projection transforms the data to a lower dimension
- Given features, $X_1, \ldots, X_m$ 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!

Feature Projection

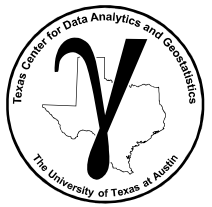


- One solution, is to find a good lower dimensional, p, representation of the original dimensions m

- The Benefits:
  - Data storage / Computational Time
  - Visualization
  - Modeling with $m = 1, \ldots, M$ takes care of multicollinearity

- The Limitations:
  - It may be more difficult to understand the model
  - The new features $p = 1, \ldots, P$ are combinations of the original features $m = 1, \ldots, M$, lose their physical meaning!

Feature Projection



Wide variety of methods:

- Principal component analysis
  - Linear mapping of the data to lower dimensional space
  - Maximizes the variance explained by the reduced subset of features

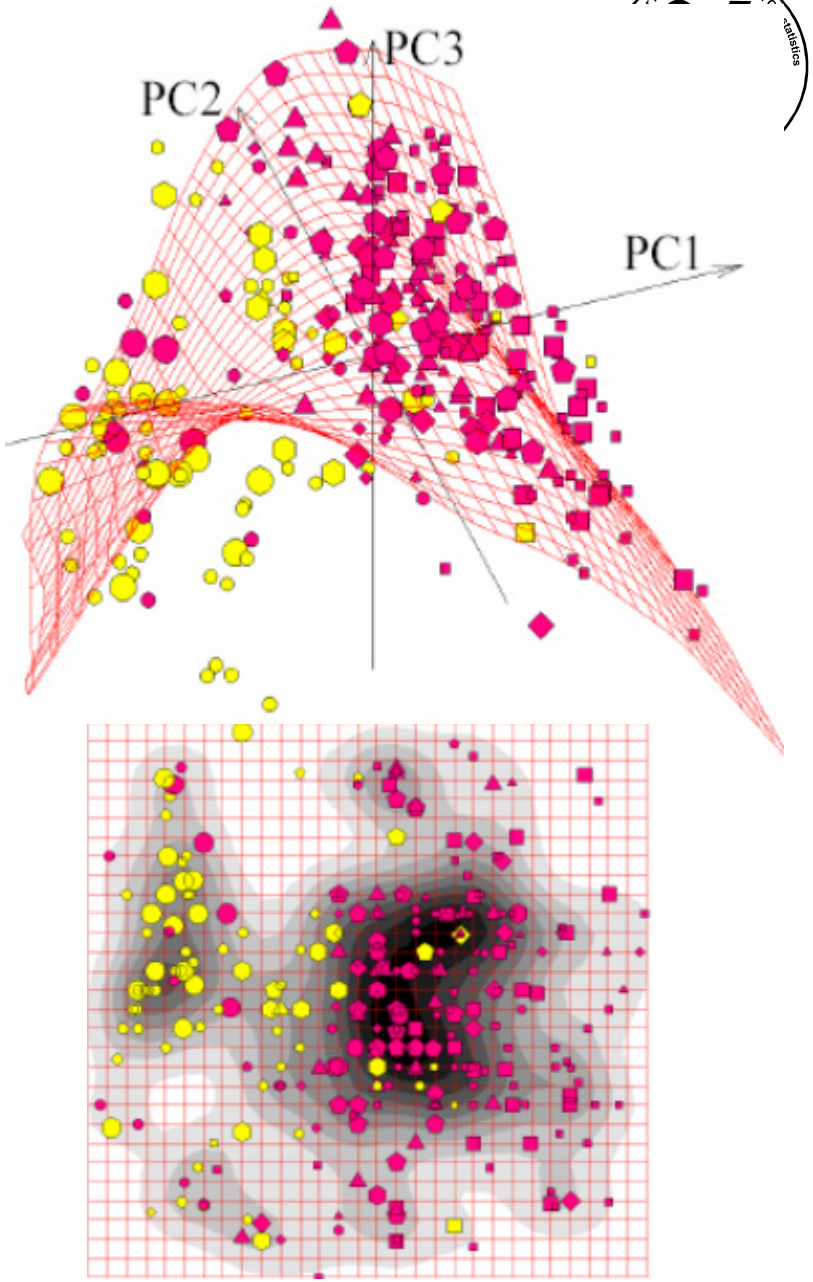
- Kernel Principal component analysis
  - Nonlinear mapping of the data to lower dimensional space with the **kernel trick**
  - Kernel Trick – use of a kernel function to operate in higher dimensional feature space with only the ‘similarity’ between the data points

Feature Projection

Wide variety of methods:

- Factor Analysis
  - Like PCA, linear combinations of the features
  - Focus on inter-correlations

- Non-linear PCA
  - Form an embedded manifold for data approximation
  - Project the data onto the manifold
  - Natural geometric interpretation principal curves and manifolds

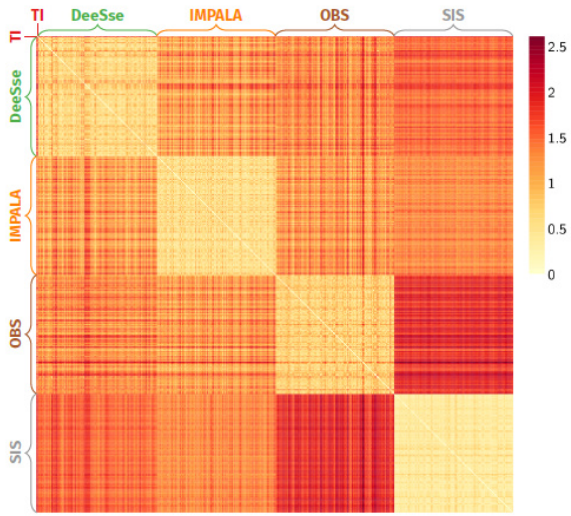


Nonlinear PCA 3D to 2D

Figure from Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods and Techniques, Olivas E.S. et al Eds. Information Science Reference, IGI Global: Hershey, PA, USA, 2009. 28–59.

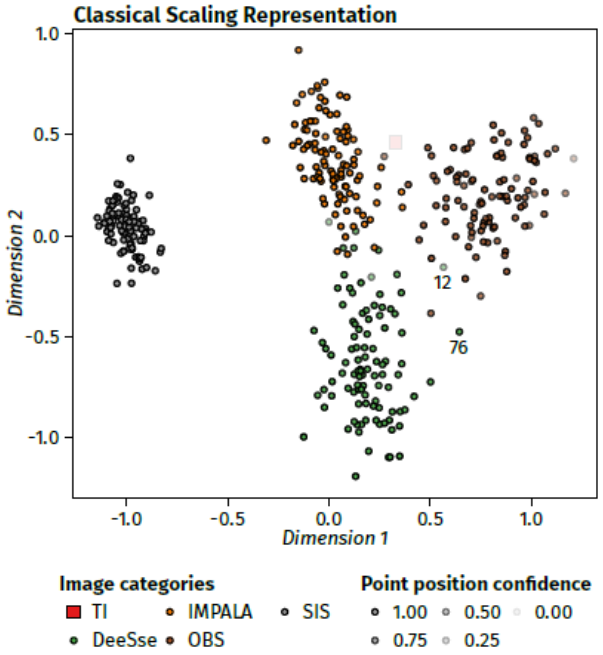
Feature Projection

Dissimilarity based on combination of metrics: proportions, transitions, connectivity, shape, networks



Wide variety of methods:

- Multidimensional Scaling
  - Ordination technique for information visualization
  - Non-linear dimensional reduction
  - Given a matrix of pairwise distances between all data, project to lower dimensional space, P
  - such that the between sample distance is preserved as well as possible.

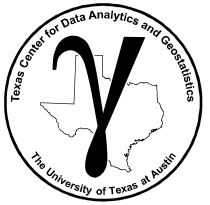


MDS to Visualize Model Uncertainty Space Sampled with Scenarios and Realizations

Figure from Rongier, G. Ph.D. thesis.

Geostatistics and Machine Learning

Dimensionality Reduction



- **Principal Component Analysis**

Introduction

Data Analytics

Inferential Methods

Predictive Methods

Advanced Methods

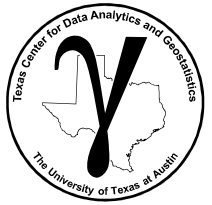
Conclusions

Michael Pyrcz, The University of Texas at Austin

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 (p) available are $\min(n-1, m)$
  - Limited by the variables/features, m, and the number of data, n
- Components are ordered
  - First component describes the larges 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

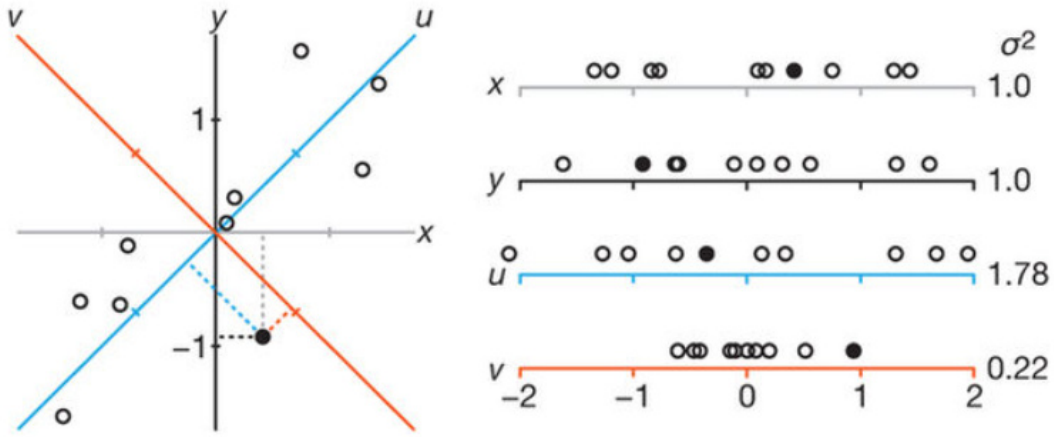
Principal Components Analysis



- **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.

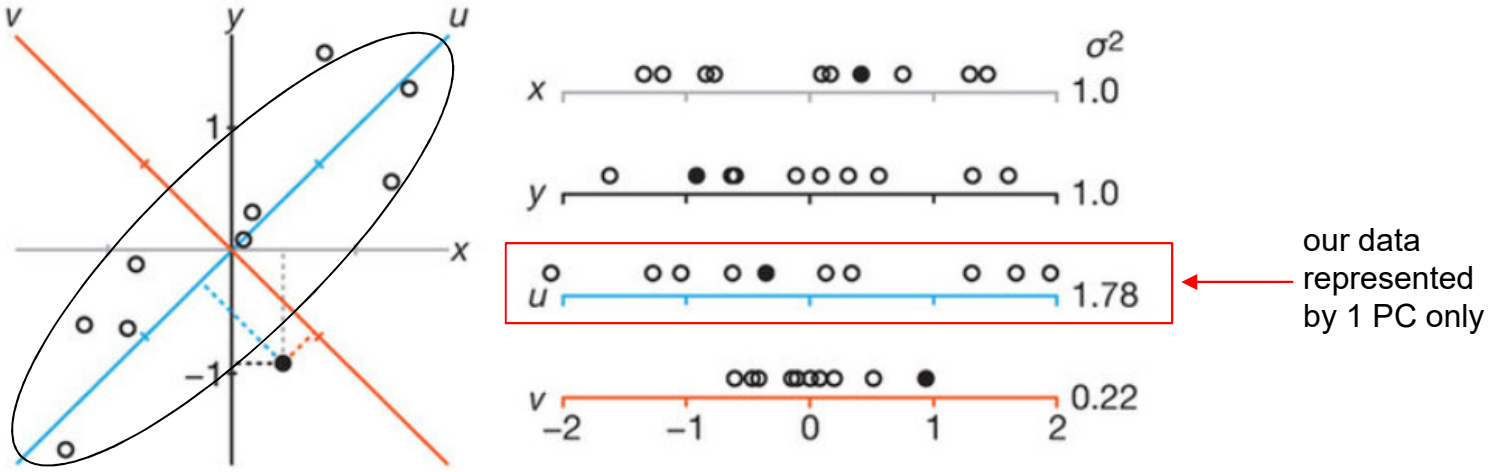


  - See the projections on to x and y axes. Note the data has equal variance in x and y. If you omitted x or y from the dataset you would lose a lot of information!
  - Find the rotation that would maximize the variance on the projection, u.
  - The 2nd axis is given as perpendicular to the first (determined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

Lost image citation.

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



Lost image citation.

Principal Components Analysis

- Principal Component
  - The first **principal component** of a set of features, $x_{i,1}, \ldots, x_{i,m}$, is the normalized linear combination of the features:

first principal component ($PC\#1$) $z_{i,1}, \ldots, z_{i,1}\ \forall\ i = 1, \ldots n$, data

with the largest variance.

  - Normalization requires:

$$\sum_{j=1}^{m} \phi_{1,j}^2 = 1.0$$

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

In matrix notation:

Factor Loadings

$$PC = \boldsymbol{V}\ X$$

Principal Component Scores

Centered Data Matrix

  - The values $\mathbf{V} = \left[\phi_{1,j}, \phi_{2,j} \ldots \phi_{m,j}\right] \forall\ j = 1, \ldots, m$ components, are known as factor or component loadings

  - We can calculate the first **principal component scores** (values projected onto this principal coordinate) as:

$$z_{i,1} = \phi_{1,1}x_{i,1} + \phi_{2,1}x_{i,2} + \cdots + \phi_{m,1}x_{i,m} \qquad \text{for } i = 1, \ldots n, \text{ data}$$

Principal Components Analysis

- Factor / Component Loadings
  - Observes the groups of variables that strongly influence each principal coordinate.

$$z_{i,1} = \phi_{1,1}x_{i,1} + \phi_{2,1}X_{i,2} + \cdots + \phi_{m,1}x_{i,m}$$

1st Principal Component

the loadings for PC1 are $\phi_{1,1}, \phi_{2,1} \ldots \phi_{m,1}$.

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

$$z_{i,2} = \phi_{1,2}x_{i,1} + \phi_{2,2}x_{i,2} + \cdots + \phi_{m,2}x_{i,m}$$

2nd Principal Component

↓

↓

$$z_{i,m} = \phi_{1,p}x_{i,1} + \phi_{2,p}x_{i,2} + \cdots + \phi_{m,p}x_{i,m}$$

mth 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 component scores, $PC_{n\times m}$
  - If we retain all the m components then have not achieved any dimensional reduction. **We just have orthogonal, linear combination of our original features!**
  - We gain dimensional reduction by retaining only p principal components or in other words by dropping the last $m - p$ components as they describe very little of the variance.

$$PC = \boldsymbol{V}\, X$$

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

Back transforming from principal components to original values.

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

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

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

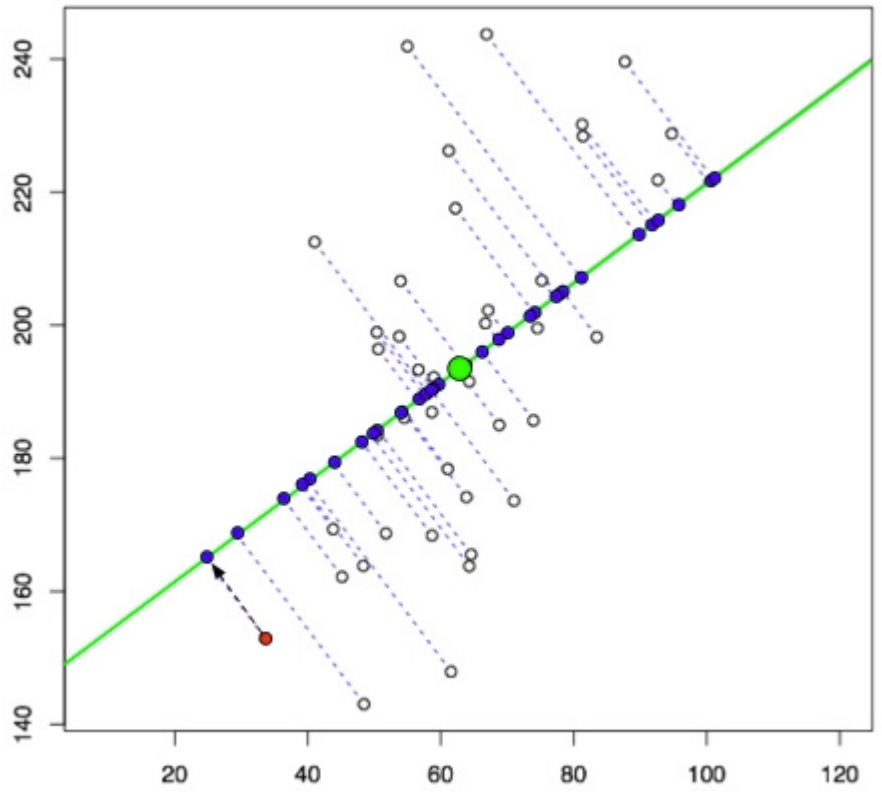
principal component scores

factor loadings

where $i = 1, \ldots, n$ data and j $= 1, \ldots, m$ variables / features, and p principal components (of $j = 1, \ldots, p$) 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**.

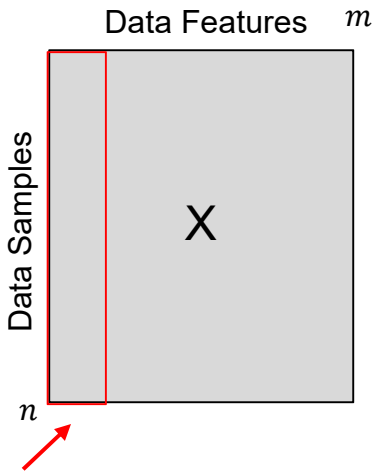


1st principal component, projects on the line are the 1st principal component scores (from https://liorpachter.wordpress.com/2014/05/26/what-is-principal-component-analysis/).

Principal Components Analysis Summary

- **Forward Transform**

Normalized Data

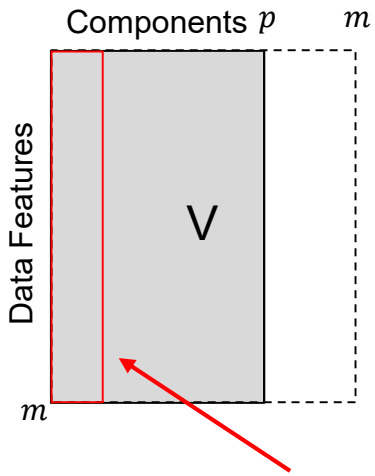


Normalized by-Feature

$\bar{x}_{X_j} = 0.0, \sigma^2_{X_j} = 1.0$

×

Component Loadings

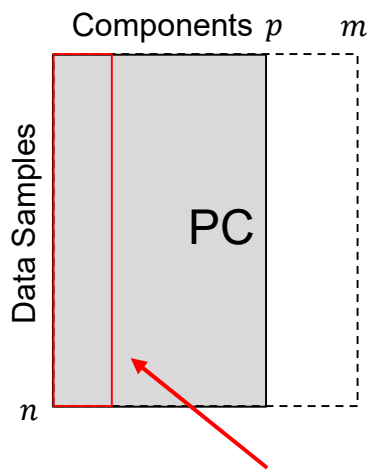


Orthogonal / Rotation

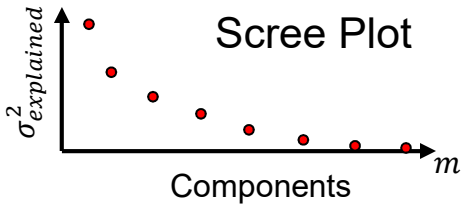
$\sum_{j=1}^{m} \phi^2_{,j} = 1.0$

=

Principal Component Scores



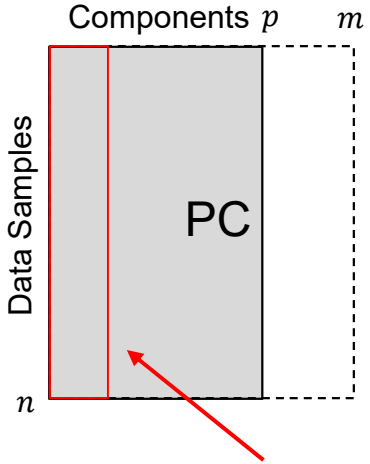
Data Projection / Coordinate in Orientation that Explains the Most Variance



Principal Components Analysis Summary

- **Reverse Transform**

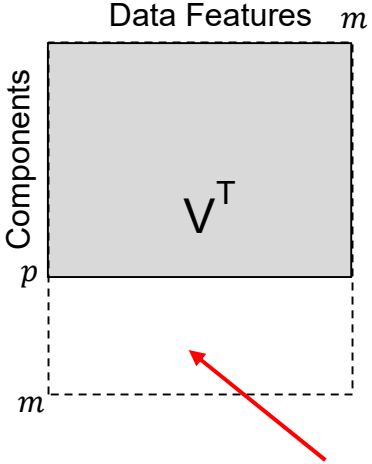
Principal Component Scores



Date Projection / Coordinate In Orientation that Explains the Most Variance

×

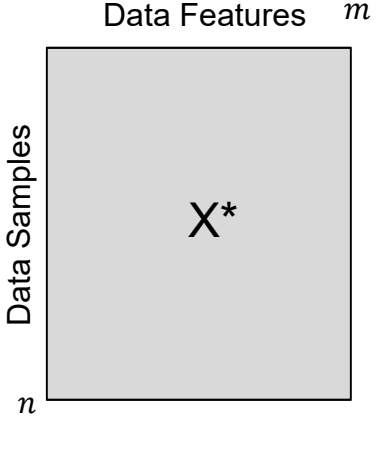
Component Loadings



Remove / Zero Unused Components $m - p$

=

Normalized Data Projection



De-normalize to Restore Original Features

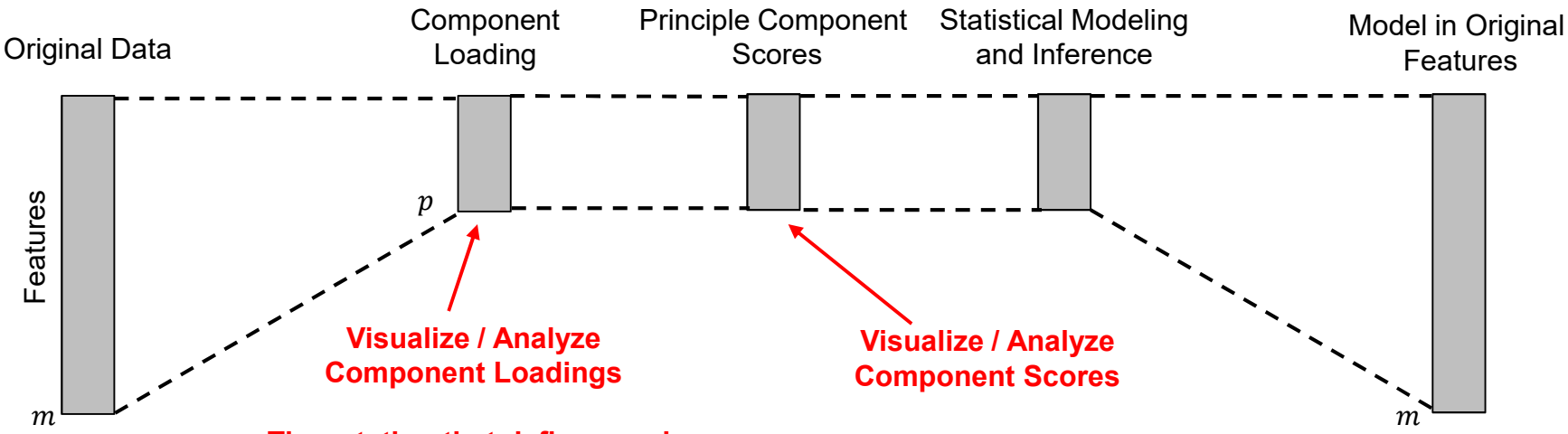
Restore Correct Variance and Mean
Affine Correction

Principal Components Analysis Summary

- **Typical Workflow**

Visualize / Analyze / Model with Component Scores

Lower dimensional space, lower risk of overfit and easier to model.



Visualize / Analyze Component Loadings

The rotation that defines each independent component that describe components of the variance

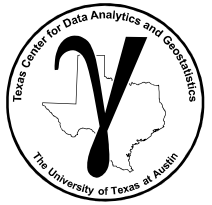
Variable groupings

Important features (images) that describe the observed variability between training

Visualize / Analyze Component Scores

Set of independent Features that describe components of the variance

Principal Components Analysis

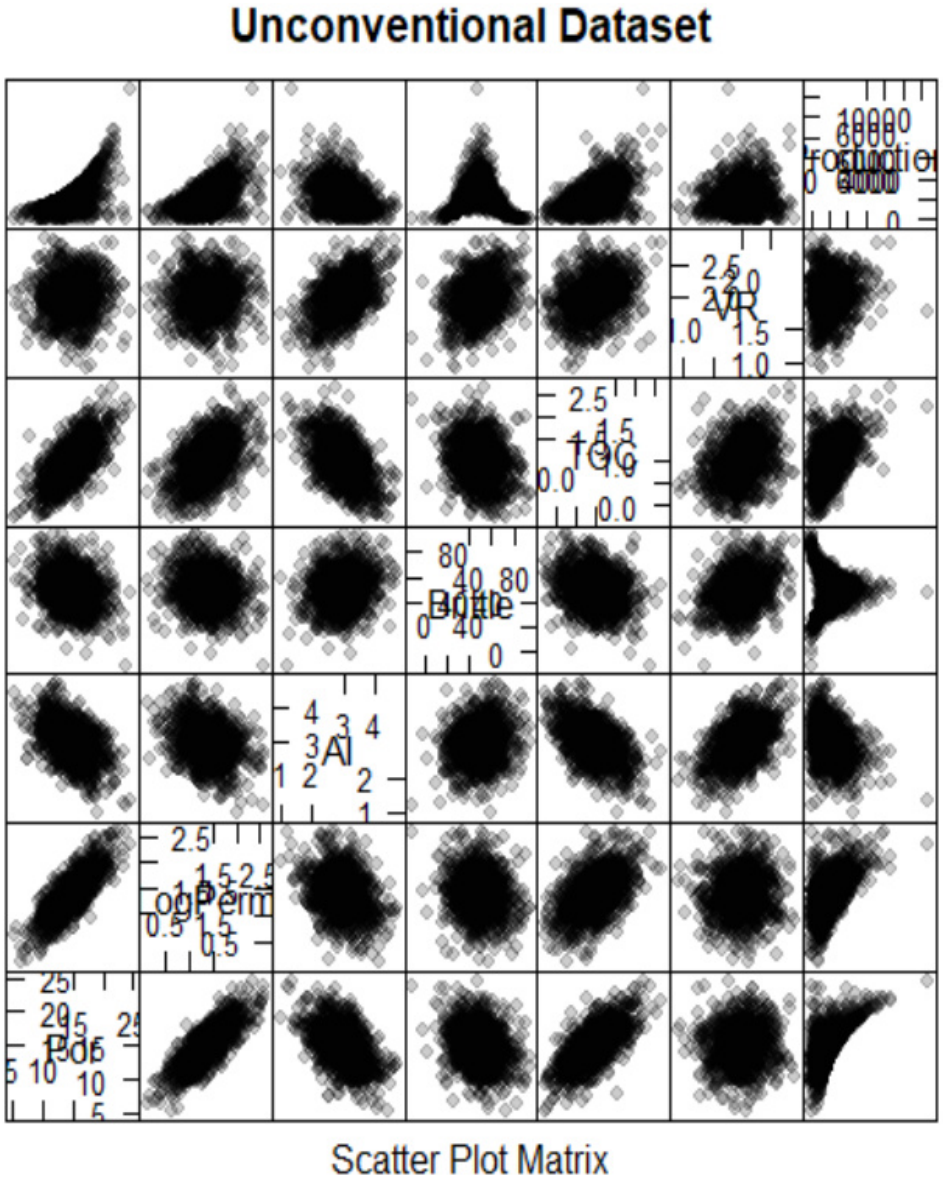


- Example in R
  - Multiple variate dataset with:
    - » 1,000 wells
    - » Porosity (%)
    - » Log transform of permeability to linearize relationship with porosity
    - » Acoustic impedance, product of density and velocity
    - » Brittleness index, percentage based on rock mechanics
    - » Total organic carbon
    - » Vitrinite reflectance, reflected light
    - » Production, initial production average monthly production over first 3 months
  - File name is “unconv_MV.csv”
  - Here’s the first 7 lines of the file:

WellIndex	Por	LogPerm	AI	Brittle	TOC	VR	Production
1	15.91	1.67	3.06	14.05	1.36	1.85	177.382
2	15.34	1.65	2.60	31.88	1.37	1.79	1479.768
3	20.45	2.02	3.13	63.67	1.79	2.53	4421.222
4	11.95	1.14	3.90	58.81	0.40	2.03	1488.318
5	19.53	1.83	2.57	43.75	1.40	2.11	5261.095
6	19.47	2.04	2.73	54.37	1.42	2.12	5497.006

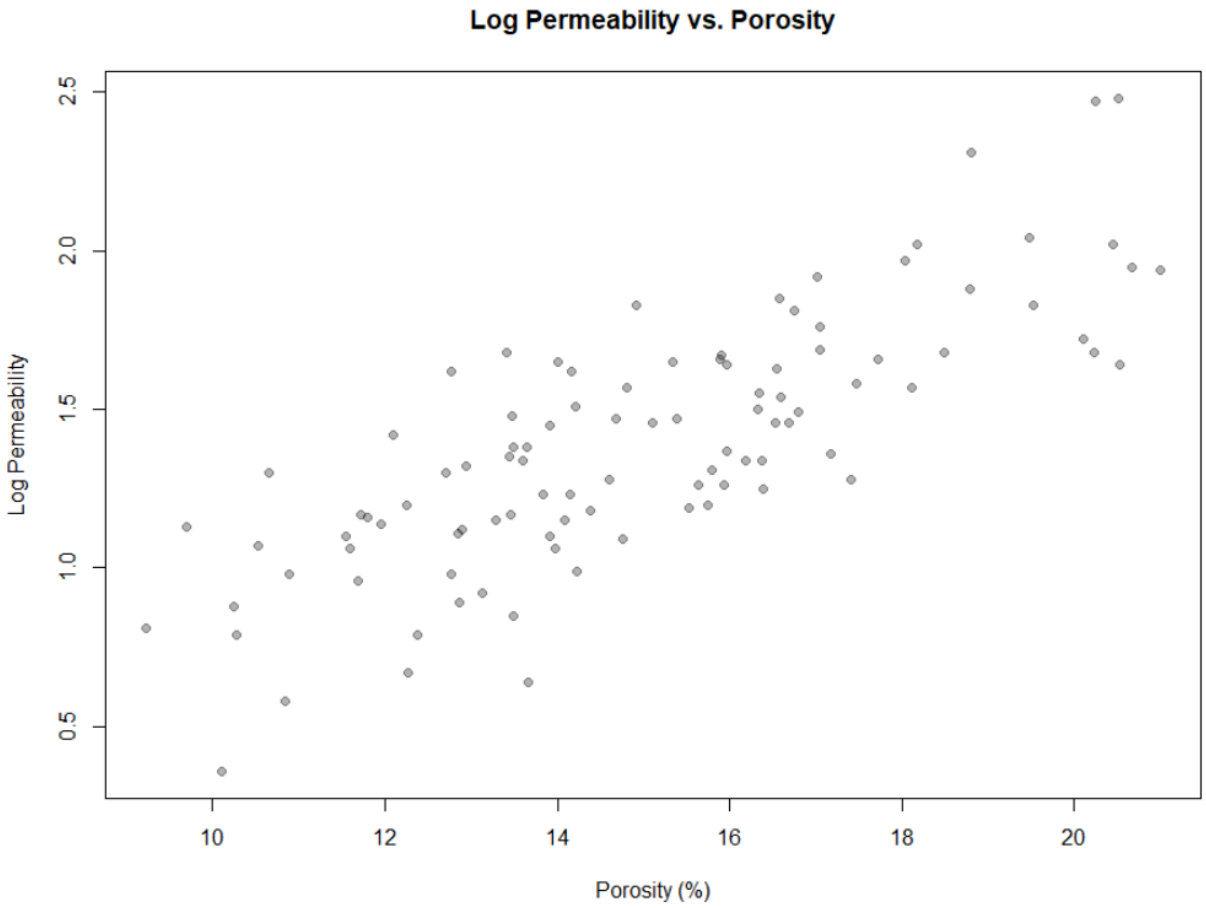
Principal Components Analysis

- Multiple variate dataset with:
  - Porosity, Por
  - Log transform of perm, LogPerm
  - Acoustic impedance, AI
  - Brittleness index, Brittle
  - Total organic carbon, TOC
  - Vitrinite reflectance, VR
  - Initial Production
- Matrix Scatter Plot of all Data
  - Most relationships are Gaussian
  - Production is more complicated



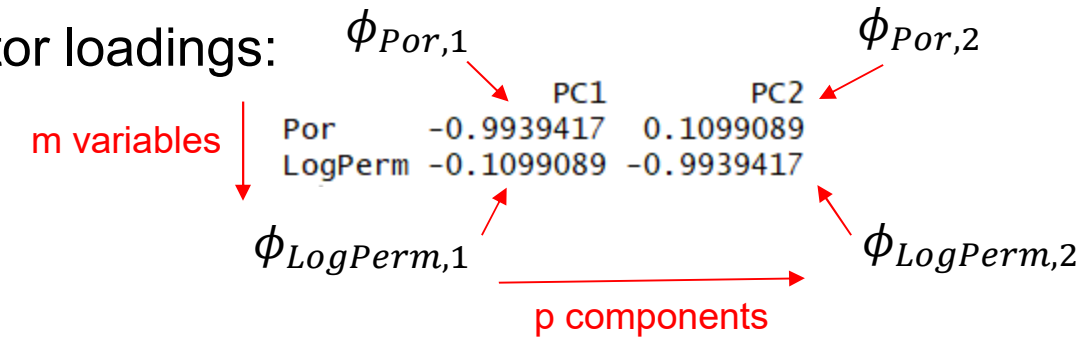
Principal Components Analysis

- Let's start simple:
  - Take first 100 wells
  - Only consider porosity and log perm.
  - We reduce our problem to bivariate since it is very easy to visualize.



Principal Components Analysis

- Here's the resulting factor loadings:



- We can calculate the principal component scores as:

$$Z_{i,1} = \phi_{Por,1} \cdot (Por_i - \overline{Por}) + \phi_{LogPerm,1} \cdot (LogPerm_i - \overline{LogPerm})$$

$$Z_{i,2} = \phi_{Por,2} \cdot (Por_i - \overline{Por}) + \phi_{LogPerm,2} \cdot (LogPerm_i - \overline{LogPerm})$$

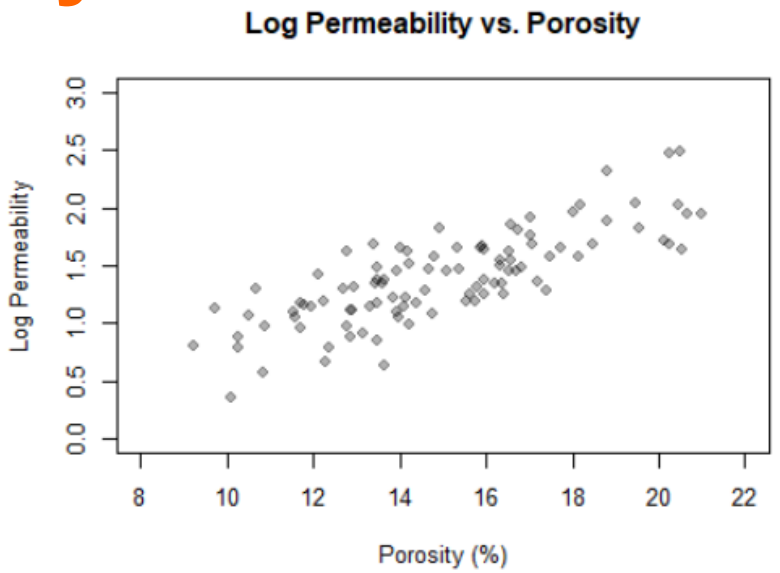
For $i = 1, \dots, n$ number of data, $j = 1, \dots, p$ components

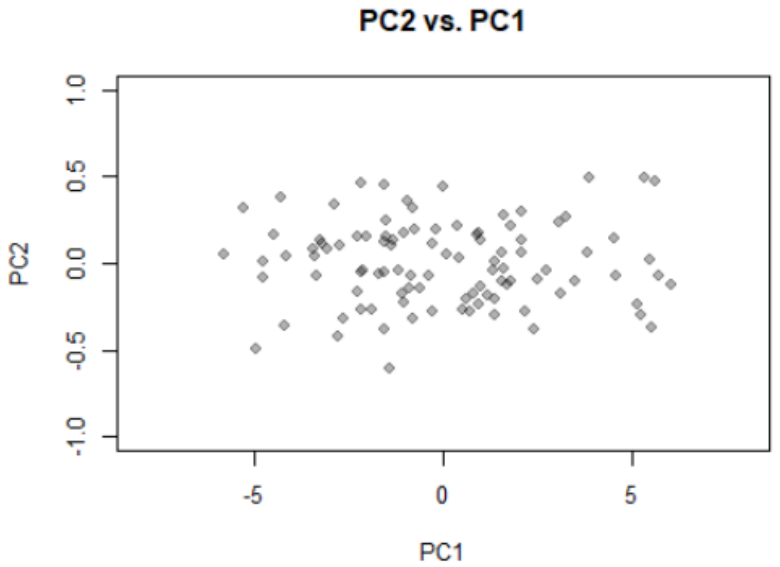
Data **Principle Components**

We have mapped original $Por_i, LogPerm_i \longrightarrow z_{i,1}, z_{i,2}$ where $z_{i,1}$ describes as much variance as possible.

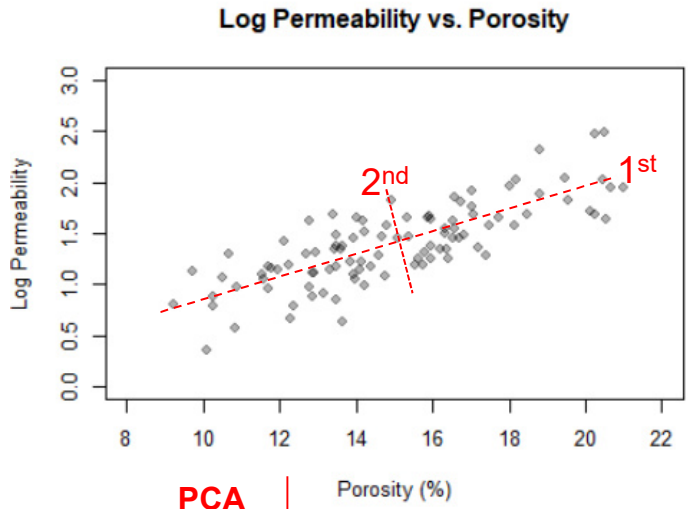
Principal Components Analysis

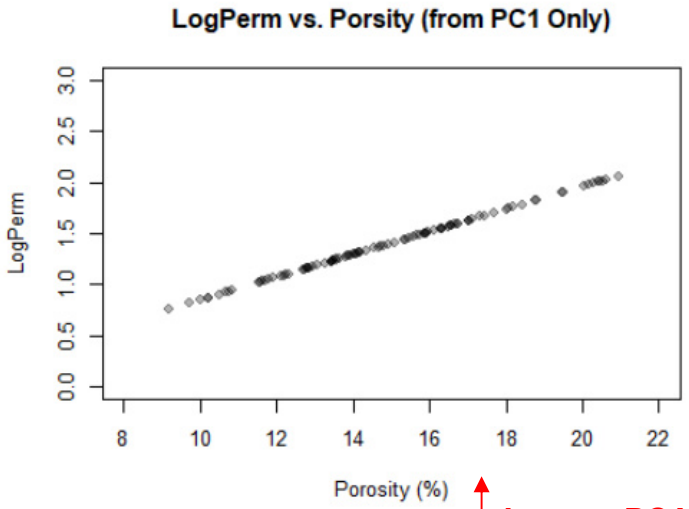
- Here we plot the original data and compare it to the plot of the principal component scores, $z_{i,1}$and $z_{i,1}$ for $i = 1, \ldots, n$ data.
- We could just retain the first principal component score.
- How much information would we loose? How much of the variance would be explained?
- Let's try that.



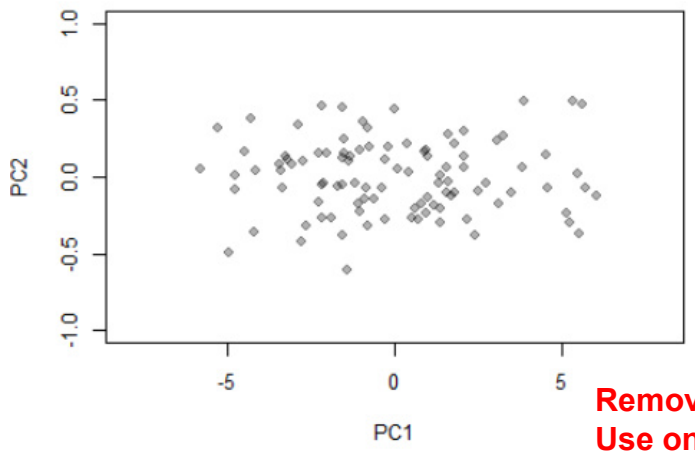


Principal Components Analysis

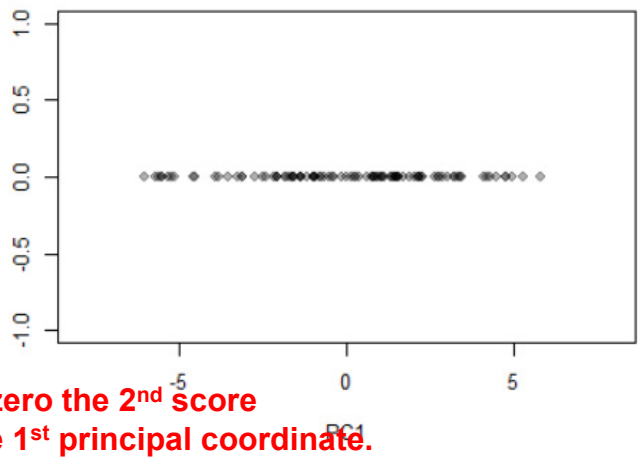




PCA
Calculate principal component scores



Remove or zero the 2nd score
Use only the 1st principal coordinate.



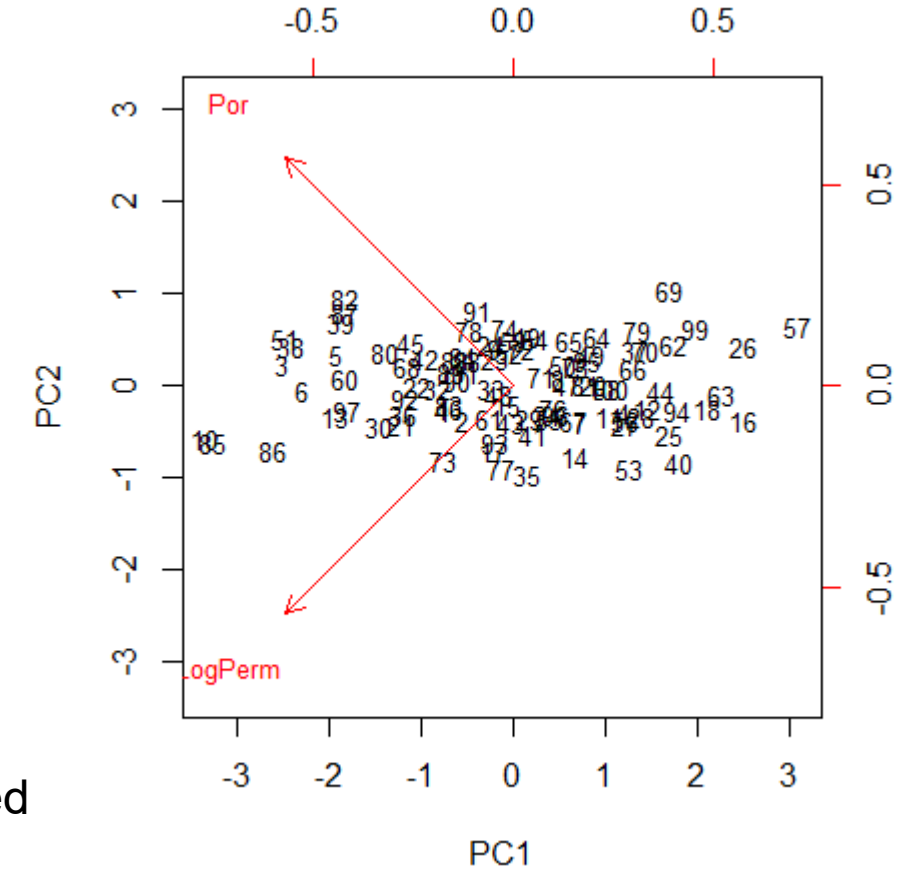
Inverse PCA
Calculate estimates of data values.

Principal Components Analysis

Biplots

- An enhanced scatter plot with the points for the principal component scores and the principal components plotted as vectors.

1. The data plotted by principal component scores
   - Left and lower axes
   - Data labels are shown
   - See, most variance is described with PC1
2. The principal components as vectors
   - Right and upper axis
   - Indicates alignment / correlation between specific features and principal coordinates
   - See PC1 and PC2 are shared by Por and LogPerm.



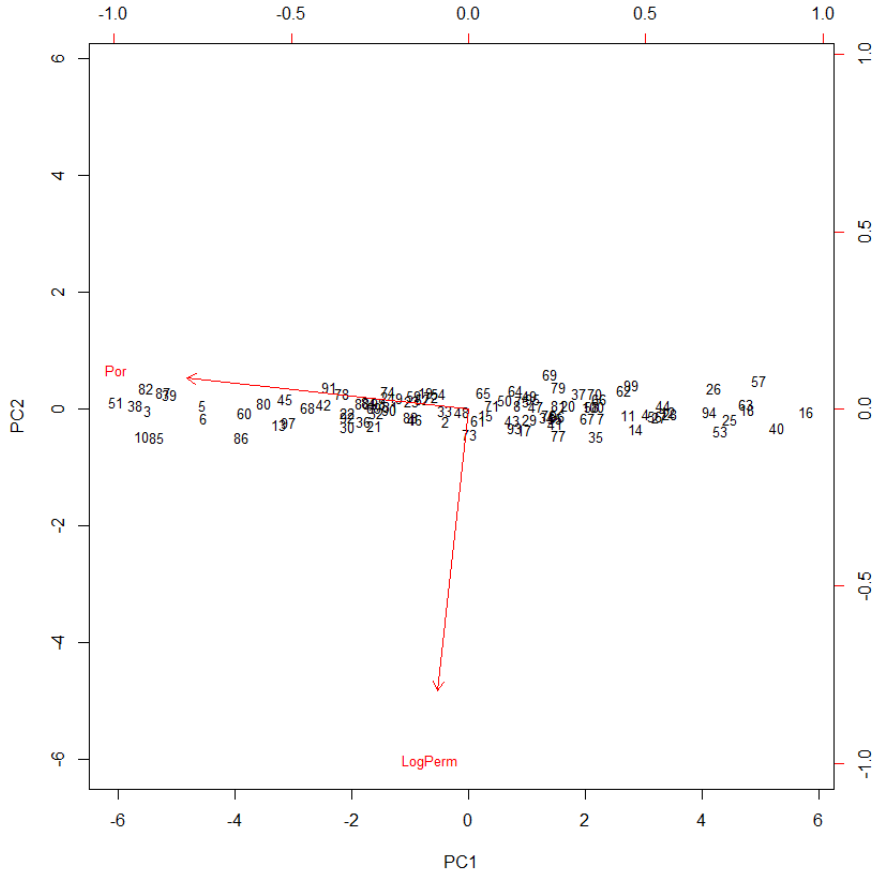
Principal Components Analysis

Scaling

- When working with a variety of variables with different scales, units you should scale the input before PCA calculation

- Scaling takes each variable and standardizes to a standard deviation of 1.0
- This is in addition to setting the mean to 0.0 (centering)

- Why is the done?
  - See our example without scaling.
  - The larger range of the porosity values causes porosity to dominate the 1st principal component! Instead of sharing each.



Principal Components Analysis

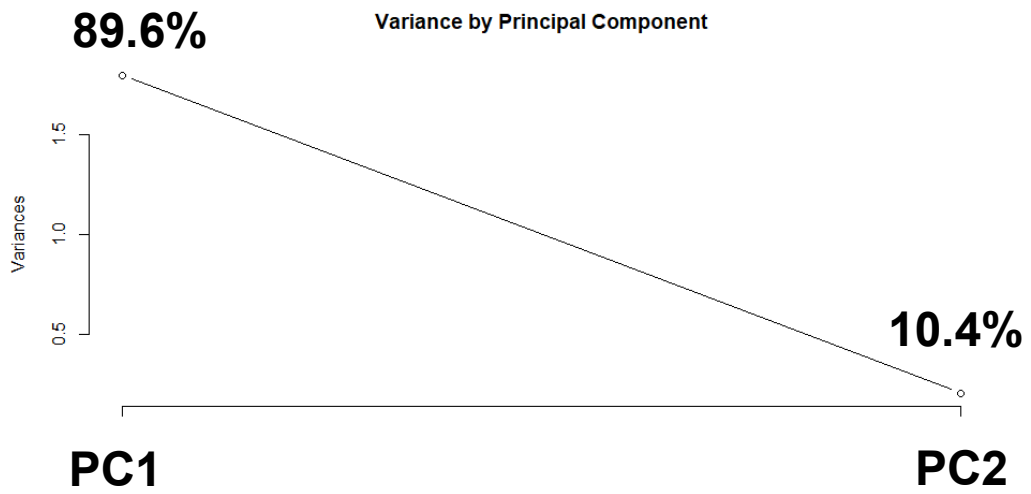
Variance Described by Each Principal Component

- So, how much variance did we capture with our components? We can calculate the proportion of variance explained by each principal component as:

$$PVE_k = \frac{1}{n}\sum_{i=1}^{n} z_{i,k}^2$$

- Should be monotonically decreasing for $k = 1, \ldots, K$.

- In our example:



Principal Components Analysis

What can you do with PCA?

Prediction:

- Reduce dimensions, build a model with the principal component scores and then restore to estimates the data values.
  - PCA regression, regression on the most important principal components

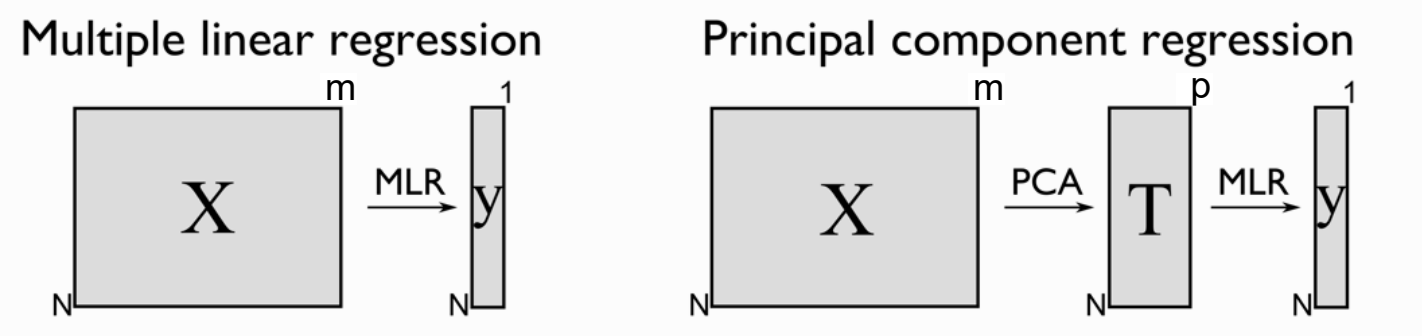
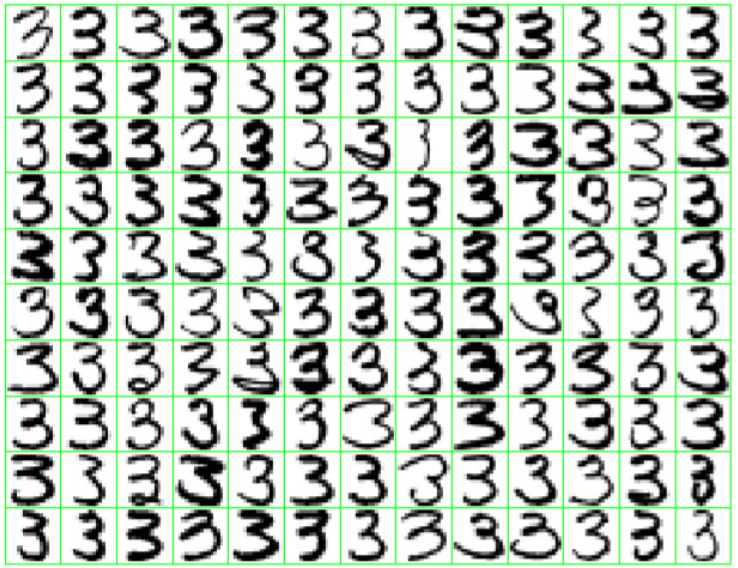


Image from: https://learnche.org/pid/latent-variable-modelling/principal-components-regression

Inference:

- Understand our variables and how variance is partitioned
- Check for and mitigate multi-collinearity
  - Exclude principal components that have low variance

Principal Components Analysis Image Example



PCA with images:

- 130 examples of hand writing
- 16 x 16 grey scale images
- $m = 256$ dimensional

Comments:

- Clearly the images have commonality
- We can describe their variability with fewer than 256 features!

Workflow:

- Calculate the covariance matrix of all pixels with each other
- Results in a 256 x 256 covariance matrix
- Center by removing average of each pixel, then calculate the Eigen values and vectors (Singular value decomposition)

Figure and example from Hastie et al., 2009.

Principal Components Analysis Image Example

Retain the first 2 principal components:

principal component score ($z_{i,2}$)

16x16 of loading PC #2 ($\phi_{2,k}$)

$$\hat{f}(\lambda) = \bar{x} + \lambda_1 v_1 + \lambda_2 v_2$$

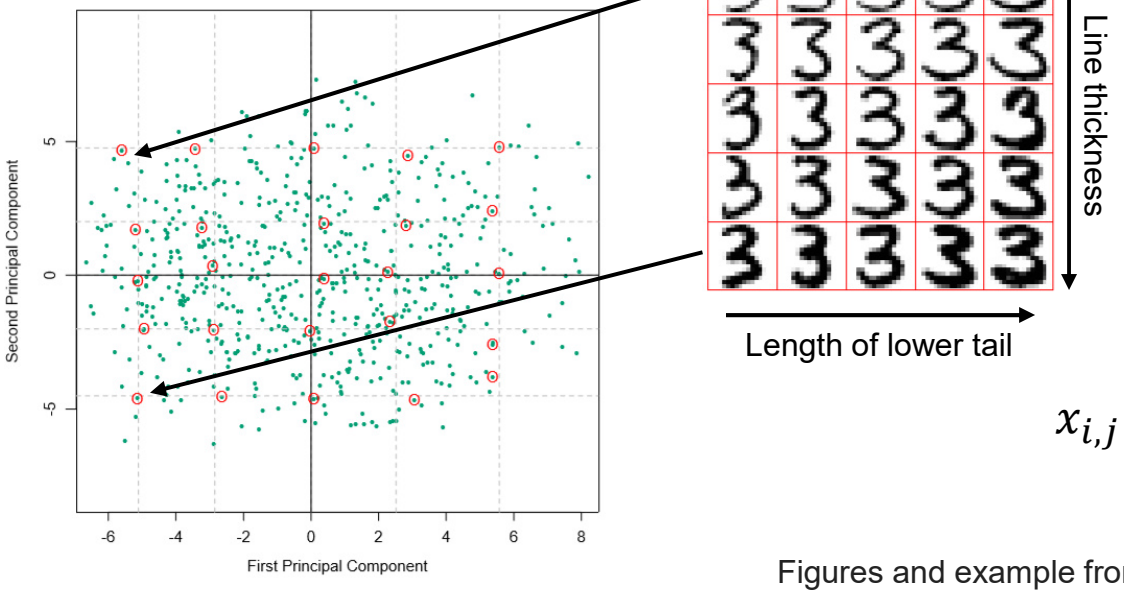
$$= \bar{x} + \lambda_1 \cdot + \lambda_2 \cdot .$$

256x256 loadings

Comments:

- We can now explore the variability of these handwritten 3's with the first two 2 principle components.

2 dials, instead of 256 to explore the space.

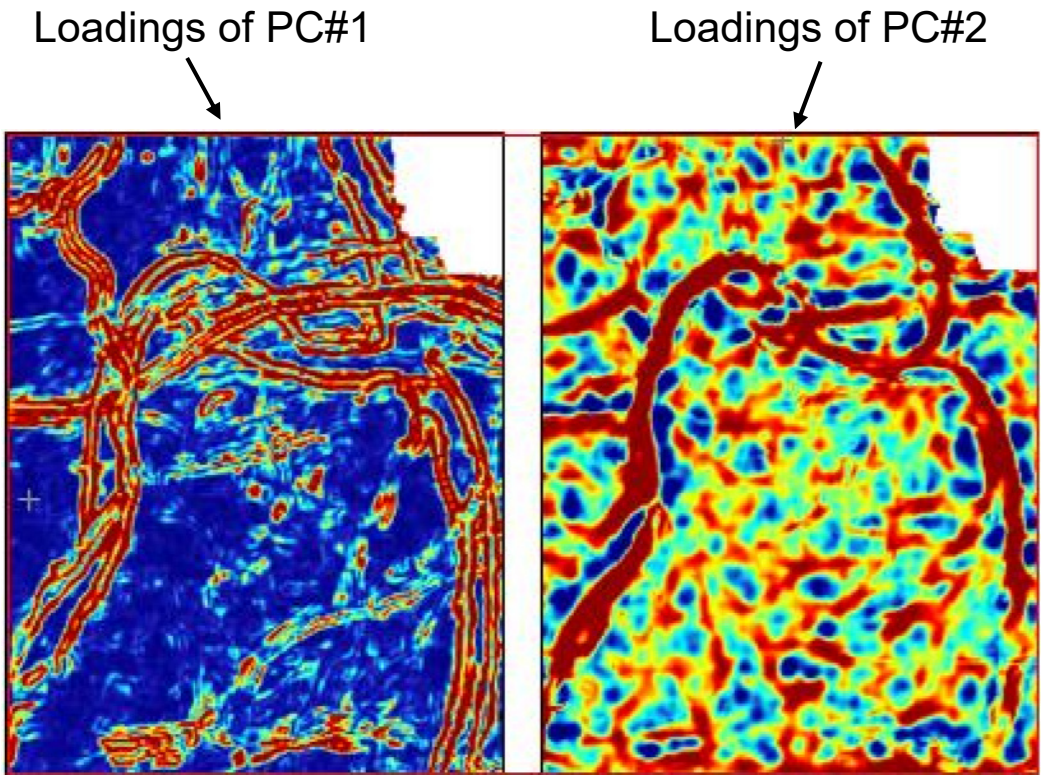


$$x_{i,j} \approx \sum_{k=1}^{p} z_{i,j}\phi_{j,k}$$

Figures and example from Hastie et al., 2009.

Principal Components Analysis Image Example

6 Seismic attributes and first 3 capture 97% of variability! Here's 2.



Each principal component describes different aspects of the multivariate seismic.

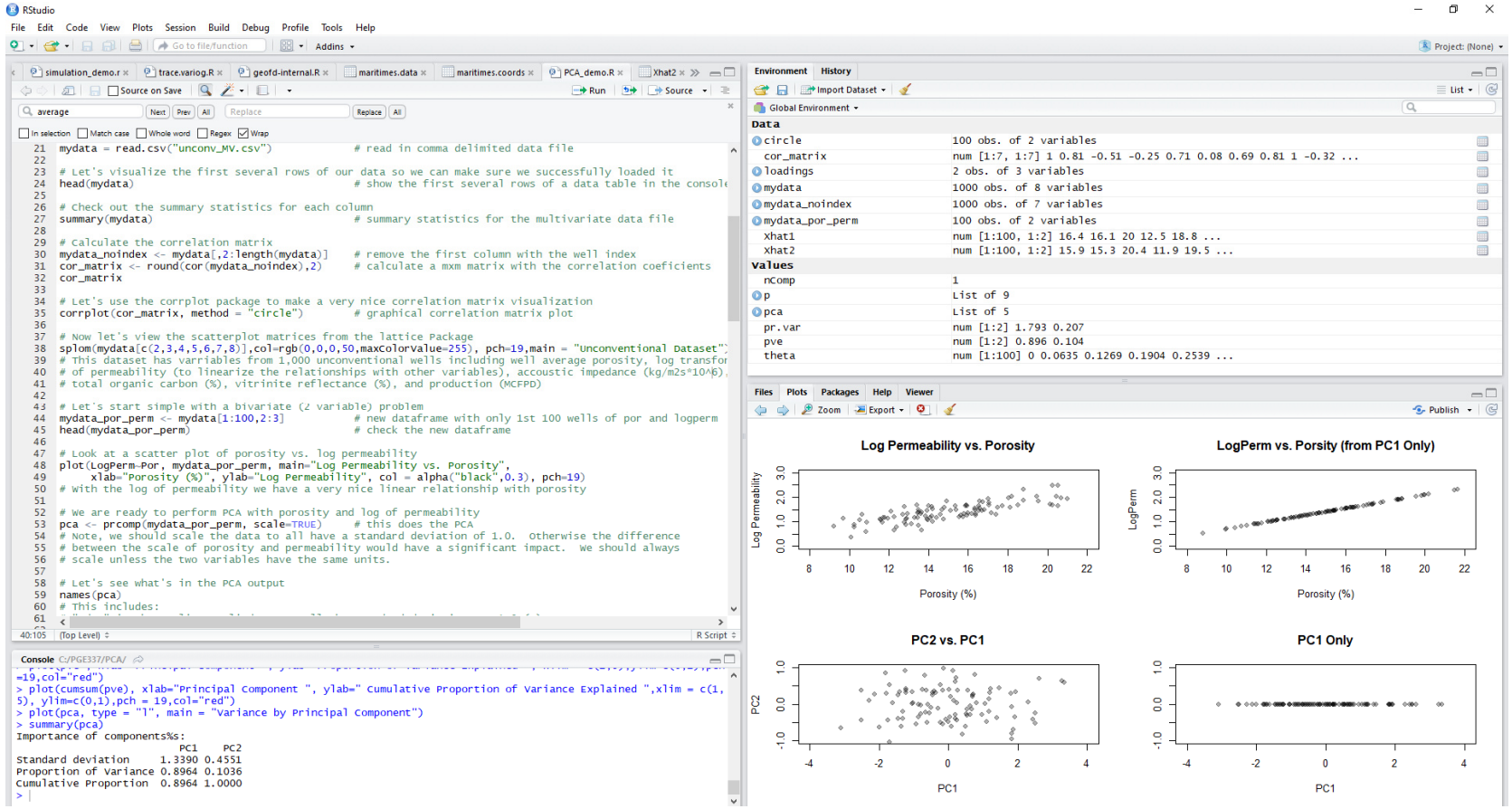
Fit models with less probability of overfit.

Figure and example from Chopra and Marfurt, 2014.

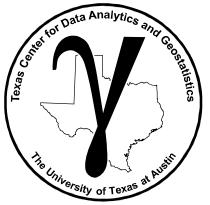
Principal Components Analysis Example

R Demonstration:

- Coded up a comprehensive workflow in R. Use “pca_demo.R” and “unconv_MV.csv”



Principal Components Analysis Example



R Demonstration:

- Also included a well-documented R Markdown html.

Principal Components Analysis in R for Engineers and Geoscientists

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

Contacts: Twitter/@GeostatsGuy | GitHub/GeostatsGuy | www.michaelpyrcz.com | GoogleScholar | Book

A tutorial/demonstration of principal component analysis (PCA). For this demonstration we use a 1,000 well 7D unconventional dataset (file: unconv_MV.csv) that may be found at https://github.com/GeostatsGuy/GeoDataSets. We take this multivariate dataset and only retain the two variables for a simple demonstration of PCA. I used this tutorial in my Introduction to Geostatistics undergraduate class (PGE337 at UT Austin) as part of a first introduction to geostatistics and R for the engineering undergraduate students. It is assumed that students have no previous R nor geostatistics experience; therefore, all steps of the code and workflow are explored and described. This tutorial is augmented with course notes.

Load the required libraries

```
library(plyr)                           # splitting, applying and combining data by Hadley Wickham
```

```
## Warning: package 'plyr' was built under R version 3.4.3
```

```
library(ggplot2)                        # for the custom biplot
```

```
## Warning: package 'ggplot2' was built under R version 3.4.3
```

```
library(lattice)                        # for the matrix scatter plot
library(corrplot)                       # for the corrplot correlation plot
```

```
## Warning: package 'corrplot' was built under R version 3.4.4
```

```
## corrplot 0.84 loaded
```

If you get a package error, you may have to first go to "Tools/Install Packages..." to install these packages. Just type in the names one at a time into the package field and install. The package names should autocomplete (helping you make sure you got the package name right), and the install process is automatic, with the possibility of installing other required dependency packages. Previously, I had an issue with packages not being found after install that was resolved with a reboot. If you get warnings concerning the package being built on a previous R version, don't worry. This will not likely be an issue.

Declare functions

no functions required in this demonstration

Set the working directory

I always like to do this so I don't lose files and to simplify subsequent read and writes (avoid including the full address each time).

Principal Components Analysis Example

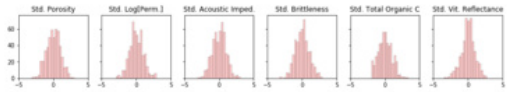
Python Demonstration:

- A well-documented Python in Jupyter Markdown html with multivariate unconventional dataset (synthetic)

Principal component analysis (PCA) is a common tool applied in machine learning workflows. It is applied widely for data analysis / exploration, dimensional reduction and directly in regression. The result of PCA is a set of orthogonal principal components and principle component scores for each data sample. These components are ordered from most variance described to least. Principal component coefficients (component loadings) reveal structures, dimensional reduction aids visualization & robust regression. Try it with a realistic dataset in a well documented **Python / Markdown Jupyter Notebook**. https://git.io/fNgRK

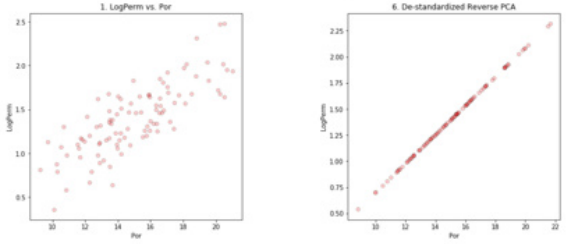
Data Checking and Cleaning

	count	mean	std	min	25%	50%	75%	max
Por	500.0	14.89936	2.985967	5.40	12.8500	14.900	17.0125	23.85
LogPerm	500.0	1.40010	0.409616	0.18	1.1475	1.380	1.6700	2.58
AI	500.0	2.99244	0.563674	1.21	2.5900	3.035	3.3725	4.70
Brittle	500.0	49.74682	15.212123	0.00	39.3125	49.595	59.2075	93.47
TOC	500.0	0.99800	0.503635	0.00	0.6400	0.960	1.3500	2.71
VR	500.0	1.99260	0.307434	0.90	1.8200	2.010	2.1725	2.84



The summary statistics and distributions look good. No obvious missing data, gaps, significant truncations, spikes or outliers. We are ready to perform principal component analysis on our 6 features.

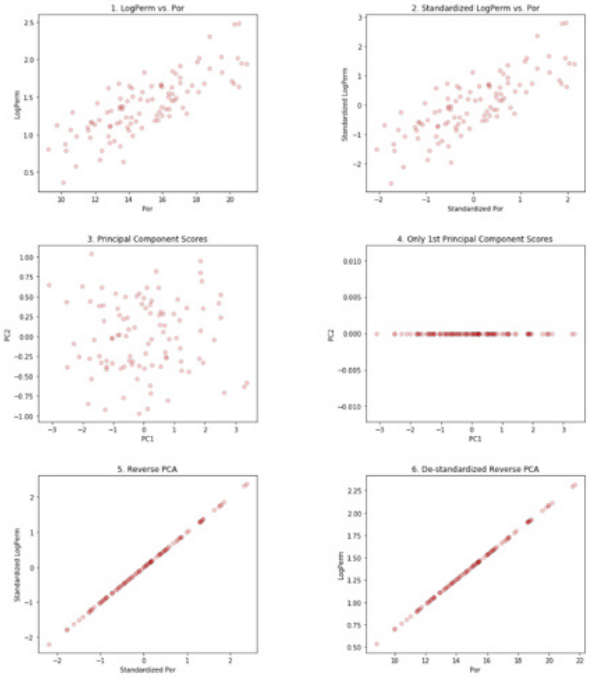
Bivariate Example



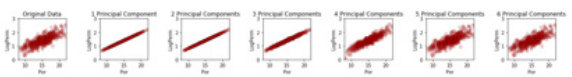
Variance Por = 7.89 , Variance Reduced Dimensional Por = 7.073 Fraction = 0.896
Variance LogPerm = 7.89 , Variance Reduced Dimensional LogPerm = 0.136 Fraction = 0.896

We got a model that explains 89.6% of the variance for both porosity and lob permeability.

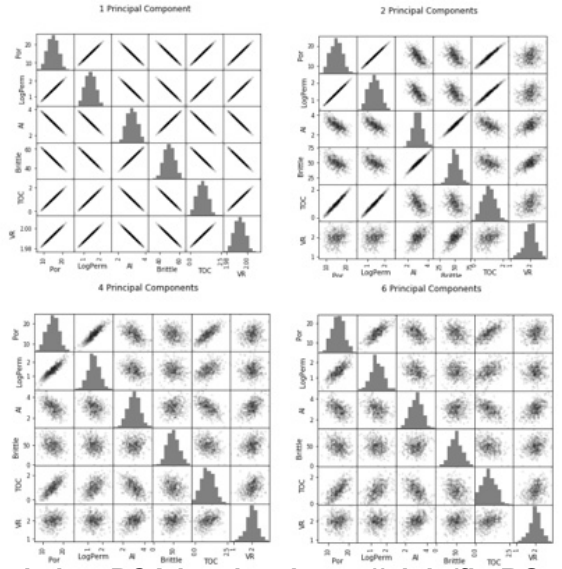
Step-by-Step Dimensional Reduction



Variable Number of Principal Components

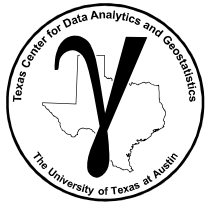


Multivariate Subsurface Example



File **SubsurfaceDataAnalytics_PCA.ipynb** at **https://git.io/fjmRO.**

Principal Components in Reservoir Modeling

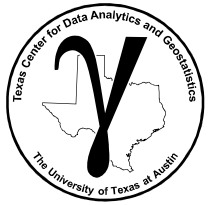


Examples of PCA in Subsurface Modeling:

- Modeling multivariate relationship while avoiding over fitting, porosity from a set of seismic attributes.
- Image analysis on seismic information, separating multiple attributes into information and noise.
- Analysis of feature grouping, redundancy
- Reducing dimensionality to support simpler workflows, e.g. bivariate, cosimulation methods

Geostatistics and Machine Learning

Dimensionality Reduction



- Curse of Dimensionality
- Dimensionality Reduction
- Principal Component Analysis

Michael Pyrcz, The University of Texas at Austin