PGE 383

Feature Transformations

Lecture outline . . .

- **Feature Transformations**
- **Feature Transformations Examples**
- **Feature Engineering**

Michael Pyrcz, The University of Texas at Austin

Motivation for Feature Transformations

There are many reasons that we may want to perform feature transformations.

- the method requires the variables to have a specific range or distribution
  - principal components analysis requires equivalent variance between features
  - artificial neural networks may require all features to range from [-1,1]
  - partial correlation coefficients require a bivariate Gaussian distribution, so we apply univariate Gaussian transformation to approximate / improve.
  - statistical tests may require a specific distribution, Student's T and χ^2 distributions assumes sampling from Gaussian distributions.
  - geostatistical sequential simulation requires an indicator or Gaussian transform

Motivation for Feature Transformations

There are many reasons that we may want to perform feature transformations.

- other practical reasons
  - – provide features consistent for visualization and comparison (violin plots)
  - – for consistency with statistical assumptions and theory (linearity and homoscedasticity)
  - – to avoid bias or impose feature weighting for methods (k nearest neighbours regression) that rely on distances calculated in predictor feature space
  - – deal with outliers and noisy data, and match theoretical outcomes

PGE 383
Feature Transformations

Lecture outline . . .

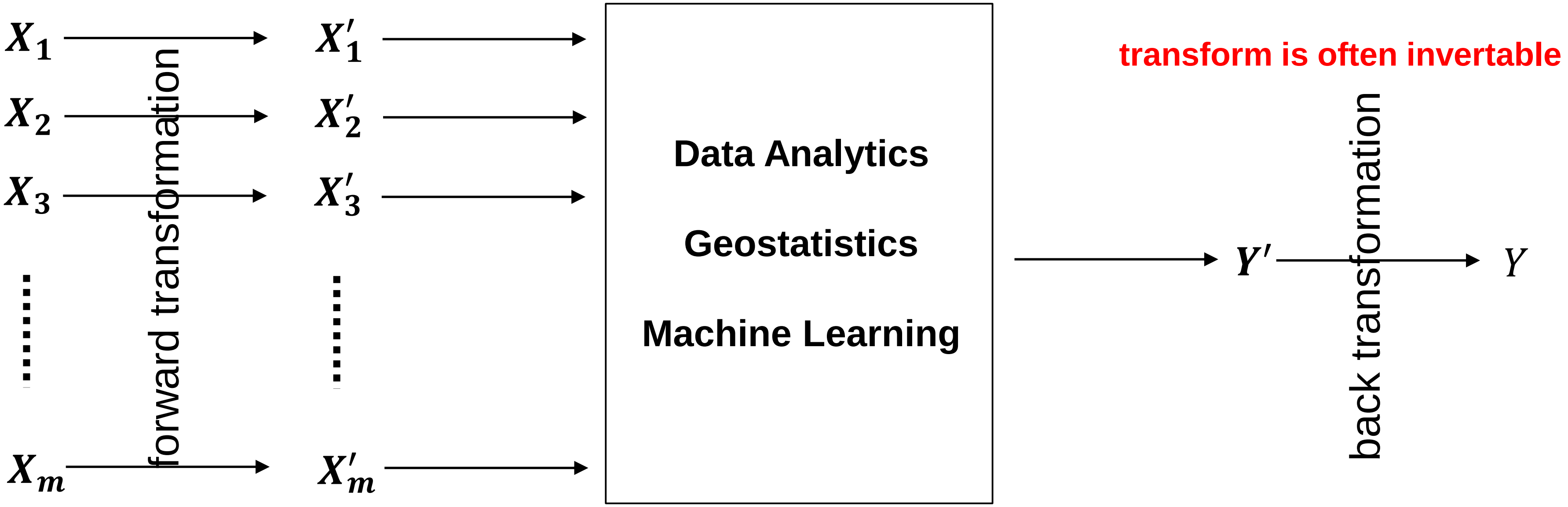
- **Feature Transformations**

What is a Feature Transformation?

- The application of a transformation applied to the feature

$$x'_\alpha = f(x_\alpha) \longleftarrow \text{deterministic mathematical function}$$

- May be applied to a predictor feature prior to input into a predictive model
- May be applied to a response feature output from a predictive model
- May be applied to any feature to improve an inferential or predictive workflow
- Could be just applied to improve data visualization



X_1 X_2 X_3 ⋮ X_m

forward transformation

X'_1 X'_2 X'_3 ⋮ X'_m

Data Analytics

Geostatistics

Machine Learning

Y' Y

back transformation

transform is often invertable

An illustration of feature transformations to support data analytics, geostatistics and machine learning.

What is a Feature Transformation?

- Working in transformed space:

Forward Transform

transform

$$x'_\alpha = f(x_\alpha)$$

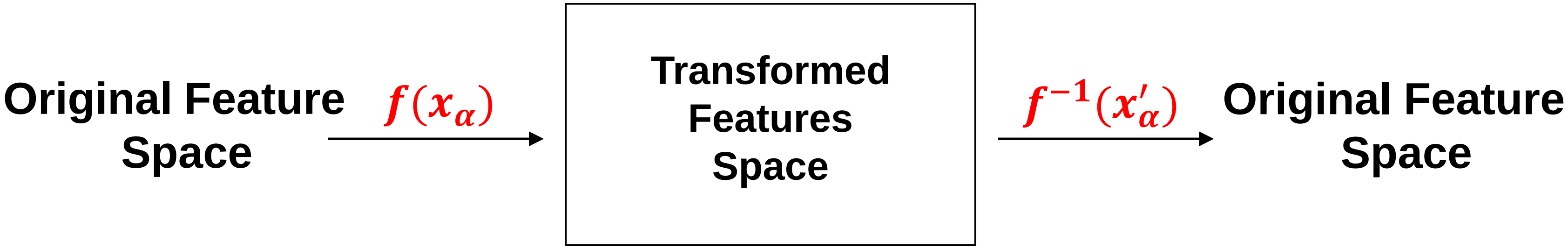
transformed **original**

Back Transform

inverse transform

$$x_\alpha = f^{-1}(x'_\alpha)$$

original **transformed**



Original Feature Space $\xrightarrow{f(x_\alpha)}$ $\boxed{\text{Transformed Features Space}}$ $\xrightarrow{f^{-1}(x'_\alpha)}$ **Original Feature Space**

Data

Modeling

Inference Prediction

What is a Feature Transformation?

Feature Engineering

using domain expertise to extract predictor or response features from raw data.

- improve the performance, accuracy and convergency, of inferential or predictive machine learning
- improve model interpretability (or may worsen interpretability if our engineered features are in unfamiliar units)
- mitigate outliers & bias, consistency with assumptions such as Gaussianity, linearization, dimensional expansion

Feature transformation and feature selection are two forms of feature engineering.

Feature Transformations

- We will start with very simple transformation and move to more complicated ones
- In general, this topic is not complicated and may not be super very interesting!
- But, feature transformations are common in many data analytics and machine learning workflows
- You’ll learn what they are and how to do them in Python with standard packages

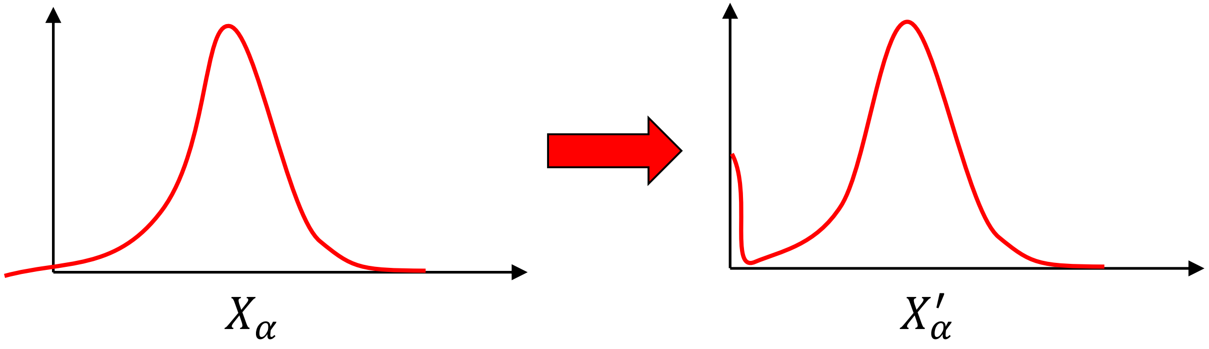
Feature Truncation

- Due to measurement error or imprecision of methods and workflows, it is possible to have feature values that are implausible
  - e.g. negative porosity, percentages outside of [0%, 100%] etc.
- We may also have outliers that exceed the range of the majority of the data set

- Truncation is the following operation:

$$x'_\alpha = \min(x_\alpha, x_t) \qquad \text{or} \qquad x'_\alpha = max(x_\alpha, x_t)$$

set the sample value to a threshold if less than or greater than.



Example of truncation of a feature distribution.

Feature Truncation

Methods that require truncation:

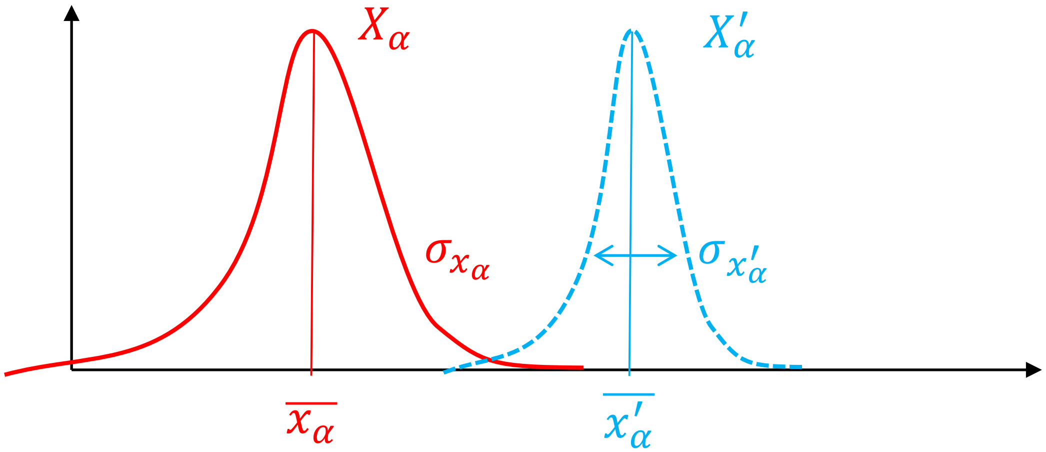
- feature values outside physical constraints (negative values, porosity exceeding geomechanical constraints)

- compositional data like mineral grades that are positive and sum to 100%

Affine Correction

- **The affine correction is the transform of the feature distribution to a new mean and variance.**
  - this is a shift and stretch / squeeze of the original property distribution
  - assumes no shape change, rank preserving

$$x'_\alpha = \frac{\sigma_{x'_\alpha}}{\sigma_{x_\alpha}} \cdot (x_\alpha - \overline{x_\alpha}) + \overline{x'_\alpha}$$



Example of affine correction of a feature distribution.

Affine Correction

Methods that require affine correction:

- debiased feature distributions, e.g. calculate a feature declustered mean and shift the distribution to the new mean

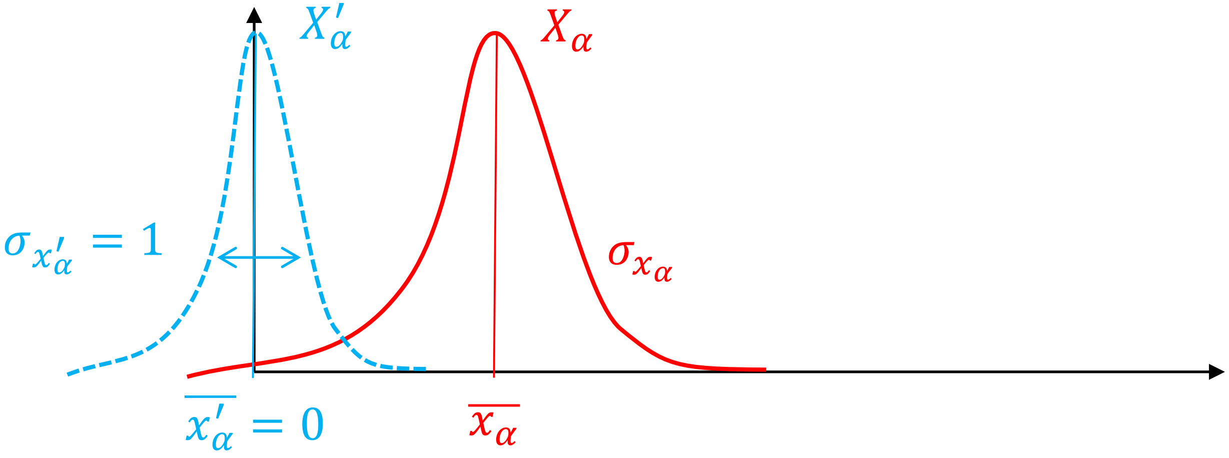
- bootstrap for the uncertainty in the feature mean and then shift the distribution mean to the P10 and P90 mean for low and high cases to create uncertainty scenarios

Standardization

- **Standardization is the transform of the feature distribution to a mean of 0 and variance of 1.**
  - this is a shift and stretch / squeeze of the original property distribution
  - assumes no shape change, rank preserving
  - specific case of the affine correction

$$x'_\alpha = \frac{\cancelto{1}{\sigma_{x'_\alpha}}}{\sigma_{x_\alpha}} \cdot (x_\alpha - \overline{x_\alpha}) + \cancelto{0}{\overline{x'_\alpha}}$$

$$x'_\alpha = \frac{1}{\sigma_{x_\alpha}} \cdot (x_\alpha - \overline{x_\alpha})$$

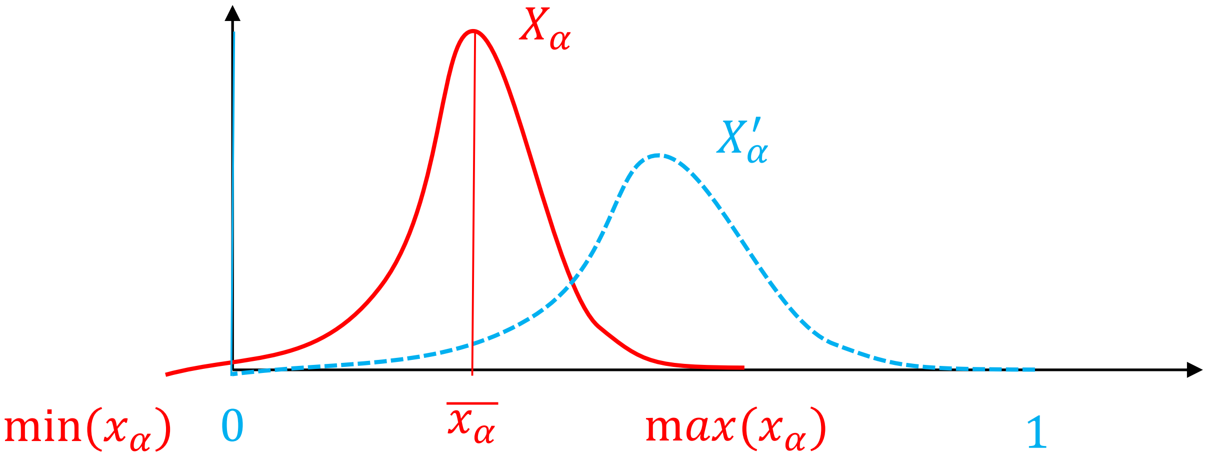


Example of standardization of a feature distribution.

Normalization
Min / Max Transform

- **Normalization is the transform of the feature distribution to a min of 0 and max of 1 (sometimes -1 to +1)**
  - this is a shift and stretch / squeeze of the original property distribution
  - assumes no shape change, rank preserving

$$x'_\alpha = \frac{x_\alpha - \min(x_\alpha)}{max(x_\alpha) - \min(x_\alpha)}$$



Example of normalization of a feature distribution.

Standardization and Normalization

Methods that require standardization and min/max normalization:

- k-means clustering, k-nearest neighbour regression
- β coefficient's for feature ranking
- standardized variograms
- artificial neural networks forward transform of predictor features and back transform of response features to improve activiation function sensitivity

L1/L2 Normalizer

- **L1 / L2 Normalizer is performed across features over individual samples to constrain the sum**
  - The L1 Norm has the following constraint across samples

$$\sum_{\alpha=1}^{m} x'_{i,\alpha} = 1.0, \qquad i = 1, \dots, n$$

  - The L1 normalizer transform:

$$x'_{i,\alpha} = \frac{x_{i,\alpha}}{\sum_{\alpha=1}^{m} x_{i,\alpha}}$$

  - The L2 Norm has the following constraint across samples

$$\sum_{\alpha=1}^{m} \left(x'_{i,\alpha}\right)^2 = 1.0, \qquad i = 1, \dots, n$$

  - The L2 normalizer transform:

$$x'_{i,\alpha} = \sqrt{\frac{\left(x_{i,\alpha}\right)^2}{\sum_{\alpha=1}^{m}\left(x_{i,\alpha}\right)^2}}$$

  - applied in text classification and clustering, and L1 for compositional data

Binary / Indicator Transform

Indicator coding is transforming a feature to a probability relative to a category or a threshold.

- If $I\{\mathbf{u}: z_k\}$ is an indicator for a categorical variable,
  - What is the probability of a realization equal to a category?

$$I(\mathbf{u};\boldsymbol{z_k}) = \begin{cases} 1, & \text{if } Z(\mathbf{u}) = \boldsymbol{z_k} \\ 0, & \text{otherwise} \end{cases}$$

  - e.g. given threshold, $z_2 = 2$, and data at $\mathbf{u}_1, z(\mathbf{u}_1) = 2$, then $I\{\mathbf{u}_1; z_2\} = 1$
  - e.g. given threshold, $z_1 = 1$, and a RV away from data, $\mathrm{Z}(\mathbf{u}_2)$ then $I\{\mathbf{u}_2; z_1\} = 0.25$

- If $I\{\mathbf{u}: z_k\}$ is an indicator for a continuous variable,
  - What is the probability of a realization less than or equal to a threshold?

$$I(\mathbf{u};\boldsymbol{z_k}) = \begin{cases} 1, & \text{if } Z(\mathbf{u}) \le \boldsymbol{z_k} \\ 0, & \text{otherwise} \end{cases}$$

  - e.g. given threshold, $z_1 = 6\%$, and data at $\mathbf{u}_1, z(\mathbf{u}_1) = 8\%$, then $I\{\mathbf{u}_1; z_1\} = 0$
  - e.g. given threshold, $z_4 = 18\%$, and a RV,$\mathrm{Z}(\mathbf{u}_2) = N[16\%, 3\%]$ then $I\{\mathbf{u}_1; z_k\} = 0.75$

Binary / Indicator Transform

Example of indicator transforms for a categorical variable.

Original Data	$I\{\mathbf{u}_\alpha; z_1 = 1\}$	$I\{\mathbf{u}_\alpha; z_2 = 2\}$	$I\{\mathbf{u}_\alpha; z_3 = 3\}$
$z(\mathbf{u}_1) = 3$	0	0	1
$z(\mathbf{u}_2) = 1$	1	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$z(\mathbf{u}_n) = 2$	0	1	0

Example of indicator transform of a categorical feature.

Our $z(\mathbf{u}_\alpha)$, $\alpha = 1, \ldots, n$, data become $k = 1, \ldots, K$ sets of n data, a new variable that indicates the probability of being each category.

- This indicator transform of a categorical features is also known as **one-hot encoding**.

Binary / Indicator Transform

Example of indicator transforms for a continuous variable.

Original Data	$I\{\mathbf{u}_\alpha; z_1 \leq 5\%\}$	$I\{\mathbf{u}_\alpha; z_2 \leq 10\%\}$	$I\{\mathbf{u}_\alpha; z_3 \leq 15\%\}$
$z(\mathbf{u}_1) = 12\%$	0	0	1
$z(\mathbf{u}_2) = 4\%$	1	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$z(\mathbf{u}_n) = 17\%$	0	0	0

Example of indicator transform of a continuous feature.

Our $\mathrm{z}(\mathbf{u}_\alpha)$, $\alpha = 1, \ldots, n$, data become $k = 1, \ldots, K$ sets of n data, a new variable that indicates the probability of being less than or equal to each threshold.

- Encoding is based on assigned thresholds, 5%, 10%, and 15%.

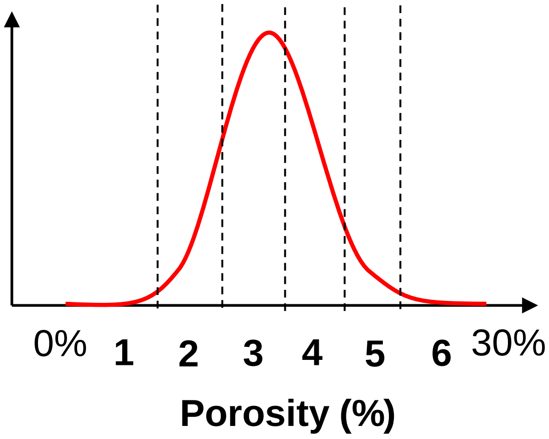
Binary / Indicator Transform

Methods that require binary / indicator transform:

- indicator variograms, indicator kriging and indicator simulation
- indicator maps
- environmental and economics thresholds and modeling probabilities of occurance
- artificial neural networks with categorical predictor or response features

K Bins Discretization

- Bin the range of the feature into K bins
- Then for each sample assignment of a value of 1 if the sample is within a bin and 0 if outsize the bin
  - strategies include uniform width bins (uniform) and uniform number of data in each bin (quantile)

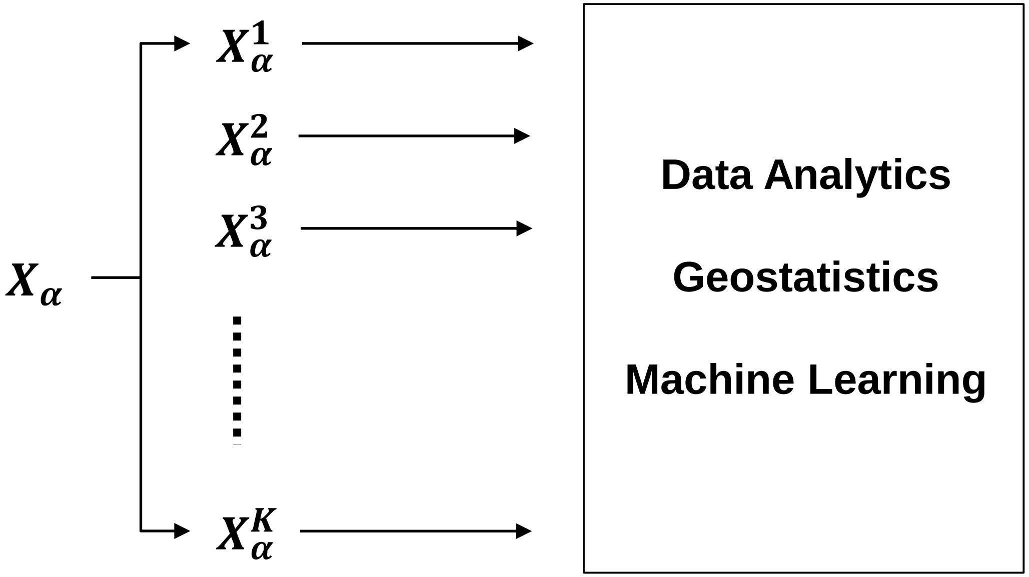


x_α	x_α^1	x_α^2	x_α^3	x_α^4	x_α^5	x_α^6
2%	1	0	0	0	0	0
16%	0	0	0	1	0	0
26%	0	0	0	0	0	1
8%	0	1	0	0	0	0

Simple example of K bins discretization

K Bins Discretization

- What is K bins discretation?
  - A probability coding, probability of the sample existing in each bin, could integrate sample uncertainty
  - A form of basis expansion (more during support vector machines)



Probability coding / basis expansion

K Bins Discretization

Methods that require K bins discretization:

- basis expansion to work in a higher dimensional space
- discretiation of continuous features to categorical features for categorical methods such as naïve Bayes classifier
- histogram construction and Chi-square test for difference in distributions
- mutual information binning

Gaussian Anamorphosis

- Quantile transformation to a Gaussian distribution.
- Mapping feature values through their cumulative probabilities.

$$y = G_y^{-1}\left(F_x(x)\right)$$

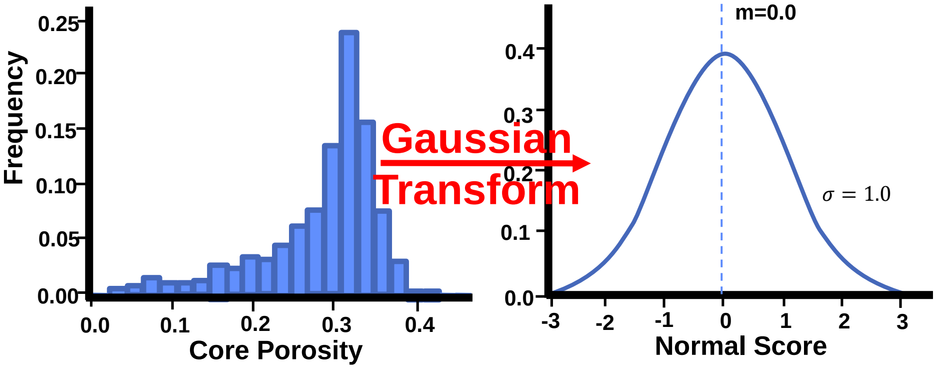
- where F_x is the original feature cumulative distribution function (CDF) and G_y is the Gaussian CDF
- Gaussian probability density function

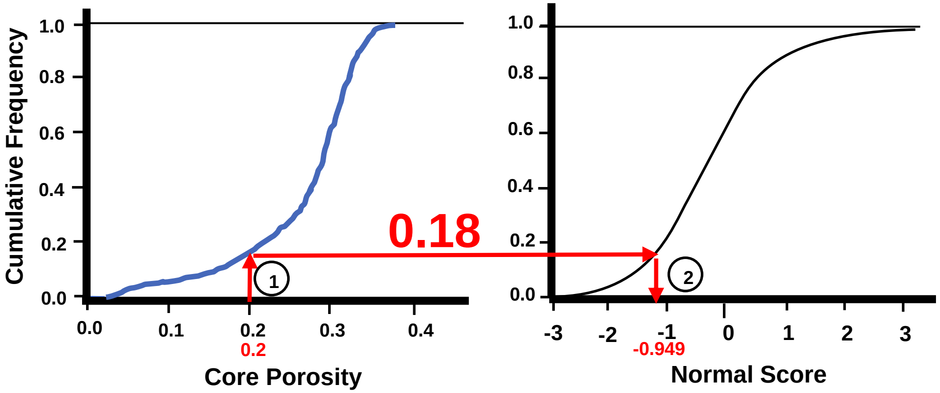
$$\boldsymbol{f(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]}$$

- Gaussian CDF

$$\boldsymbol{F(x) = \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{x} exp\left[-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right] dy}$$

$$-\infty < \boldsymbol{x} < +\infty$$





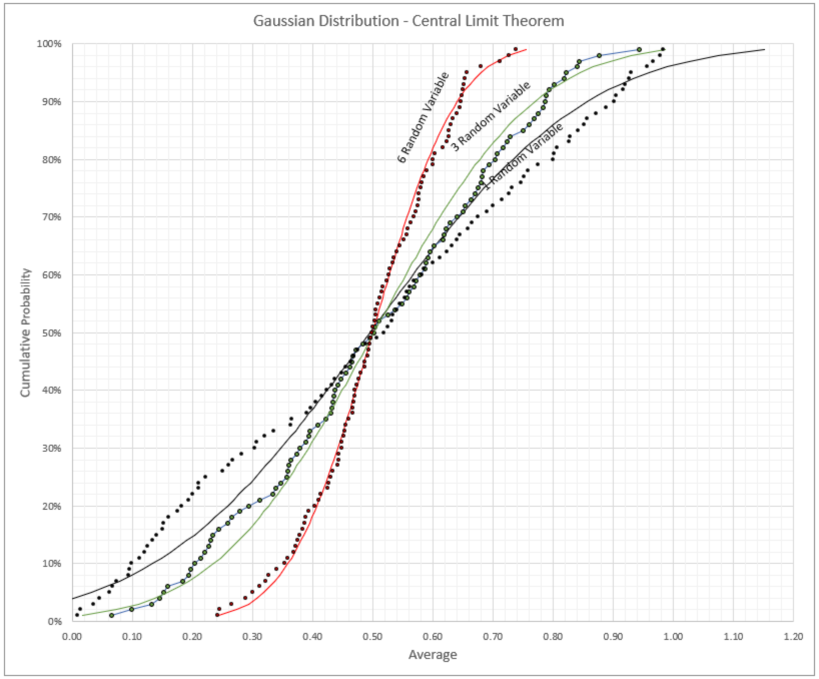
Gaussian Anamorphosis

More on the Gaussian distribution

- Shorthand for a Normal Distribution is N[mean, st.dev], N(μ,σ^2).
- Much of "natural variation" / measurement error is Gaussian
- Parameterized fully by mean, variance and correlation coefficient (if multivariate)
- distribution is unbounded, no min nor max
  - extremes are very unlikely, some type of truncation is often applied

Gaussian Anamorphosis

- Central Limit Theorem
  - the summation / average of multiple random variables tends towards a Gaussian distributed
  - this occurs quickly with 3-4 independent variables
  - some reservoir properties may be Gaussian distributed (e.g. porosity is the average of pore space vs. grains over smaller volumes.



Experimental demonstration of the Central Limit Theorem

Gaussian Anamorphosis

The Multivariate Gaussian distribution:

$$\boldsymbol{f_X(x_1, \ldots, x_m) = \frac{exp\left(-{}^{1}\!/_{2}\,(x-\mu)^T\Sigma^{-1}(x-\mu)\right)}{\sqrt{(2\pi)^m|\Sigma|}}}$$

where $\boldsymbol{\mu}$ is the m vector of means and $\boldsymbol{\Sigma}$ is the $m \times m$ matrix of all pairwise covariances.

$$\boldsymbol{\mu = [\mu_1, \ldots, \mu_1]} \qquad \boldsymbol{\Sigma = \begin{bmatrix} \sigma_1^2 & \cdots & C_{1,m} \\ \vdots & \ddots & \vdots \\ C_{m,1} & \cdots & \sigma_m^2 \end{bmatrix}}$$

A very compact parameterization:

$$m + \frac{m(m+1)}{2}$$

including m means, m variances and $\frac{m(m-1)}{2}$ unique covariances (covariance matrix, $\boldsymbol{\Sigma}$, is symmetric)

Gaussian Anamorphosis

The Marginal Distributions:

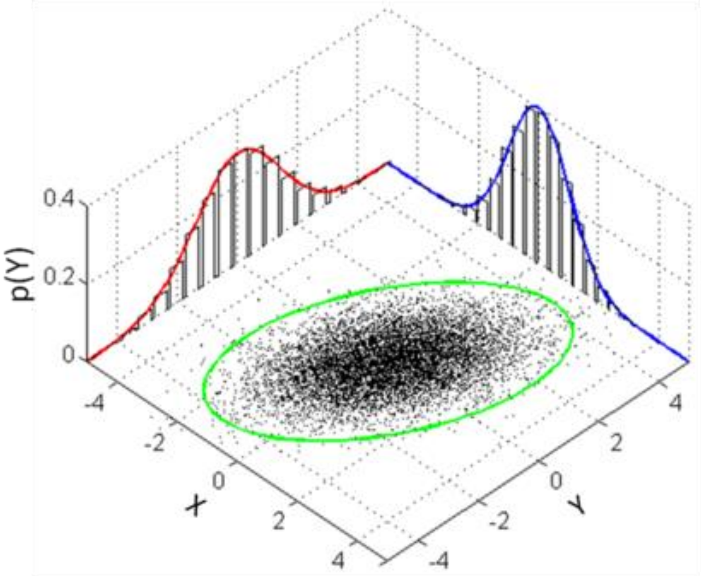
All marginal distributions are Gaussian:

$$f_{X_1}(x_1)\sim N(\mu_1,\sigma_1^2)$$

$$f_{X_2}(x_2)\sim N(\mu_2,\sigma_2^2)$$

$$\vdots$$

$$f_{X_m}(x_m)\sim N(\mu_m,\sigma_m^2)$$



Gaussian joint and marginal distributions.

Image from https://en.wikipedia.org/wiki/Multivariate_normal_distribution#/media/File:MultivariateNormal.png

Gaussian Anamorphosis

The Conditional Distributions:

All conditional distributions are Gaussian, we just show the bivariate case:

$$f_{X_1|X_2}(x_1\,|X_2 = x_2) \sim N\left(\mu_1 + \frac{\sigma_1}{\sigma_2}\rho(x_2 - \mu_2), (1-\rho^2)\sigma_1^2\right)$$

- the conditional variance is homoscedastic, does not depend on the mean!

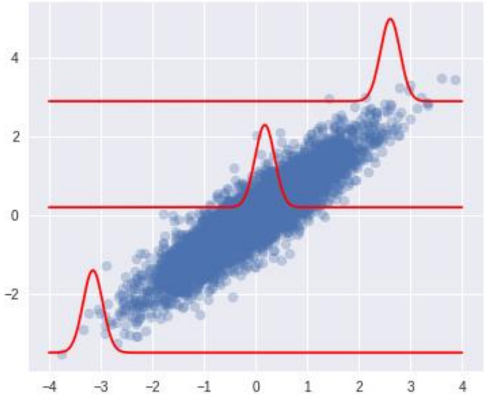


Image from https://stats.stackexchange.com/questions/385823/variance-of-conditional-multivariate-gaussian

Gaussian Anamorphosis

Methods that require a Gaussian distribution:

- Pearson product-moment correlation coefficients completely characterize multivariate relationships when data are multivariate Gaussian
- partial correlations require bivariate Gaussian
- sequential simulation (geostatistics) assumes Gaussian to reproduce the global distribution
- Student’s t test for difference in means
- Chi-square distributions is derived from sum of squares of Gaussian distributed random variables
- Gaussian naïve Bayes classification assumes Gaussian conditionals

Uniform General Distribution Transform

- Quantile transformation to a uniform distribution (e.g cumulative probabilies).
- Mapping feature values through their cumulative probabilities.

$$y = F_y^{-1}\left(F_x(x)\right)$$

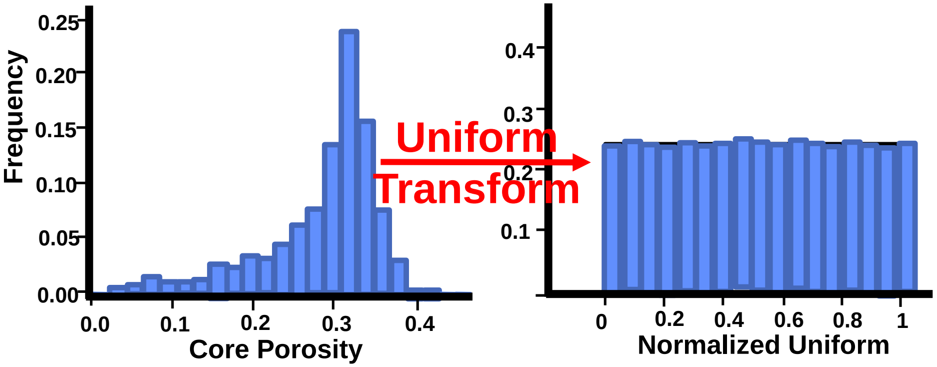
- where F_x is the original feature cumulative distribution function (CDF) and F_y is the uniform CDF
- uniform probability density function

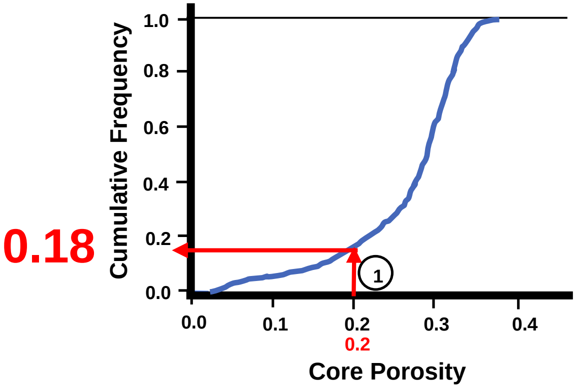
$$f_x(x) = \frac{1}{N} = constant$$

- Uniform CDF

$$F_x(x) = \frac{1}{N}x$$

$$\boldsymbol{x_{min} < x < x_{min}}$$





PGE 383
Feature Transformations

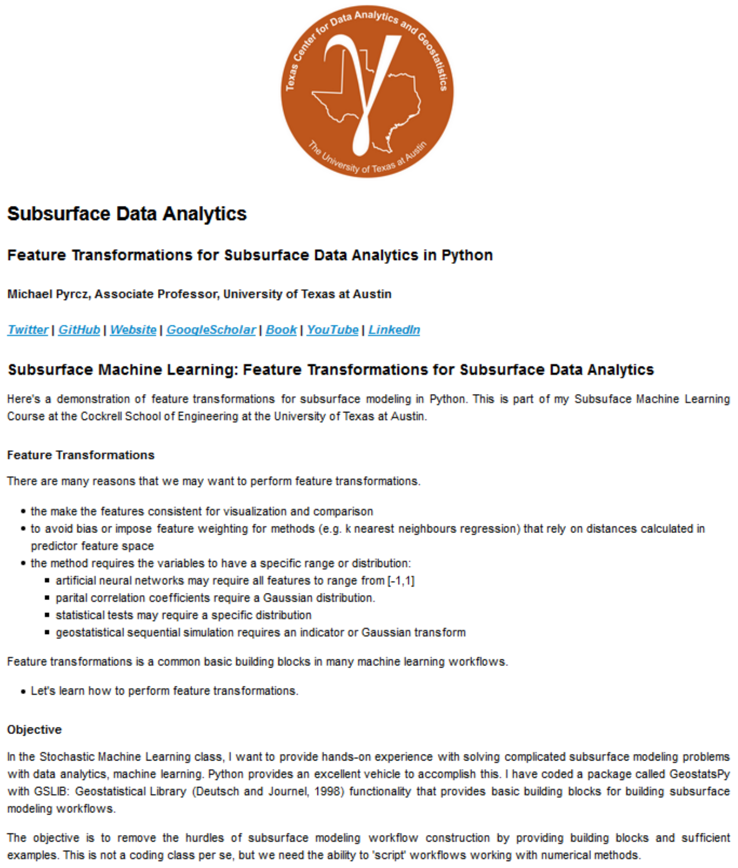
Lecture outline . . .

- Feature Transformations Examples

Michael Pyrcz, The University of Texas at Austin

Feature Transformations Demonstration in Python

Demonstration of feature transformations with a documented workflow.



Subsurface Data Analytics

Feature Transformations for Subsurface Data Analytics in Python

Michael Pyrcz, Associate Professor, University of Texas at Austin

Twitter | GitHub | Website | GoogleScholar | Book | YouTube | LinkedIn

Subsurface Machine Learning: Feature Transformations for Subsurface Data Analytics

Here's a demonstration of feature transformations for subsurface modeling in Python. This is part of my Subsuface Machine Learning Course at the Cockrell School of Engineering at the University of Texas at Austin.

Feature Transformations

There are many reasons that we may want to perform feature transformations.

- the make the features consistent for visualization and comparison
- to avoid bias or impose feature weighting for methods (e.g. k nearest neighbours regression) that rely on distances calculated in predictor feature space
- the method requires the variables to have a specific range or distribution:
  - artificial neural networks may require all features to range from [-1,1]
  - parital correlation coefficients require a Gaussian distribution.
  - statistical tests may require a specific distribution
  - geostatistical sequential simulation requires an indicator or Gaussian transform

Feature transformations is a common basic building blocks in many machine learning workflows.

- Let's learn how to perform feature transformations.

Objective

In the Stochastic Machine Learning class, I want to provide hands-on experience with solving complicated subsurface modeling problems with data analytics, machine learning. Python provides an excellent vehicle to accomplish this. I have coded a package called GeostatsPy with GSLIB: Geostatistical Library (Deutsch and Journel, 1998) functionality that provides basic building blocks for building subsurface modeling workflows.

The objective is to remove the hurdles of subsurface modeling workflow construction by providing building blocks and sufficient examples. This is not a coding class per se, but we need the ability to 'script' workflows working with numerical methods.

File SubsurfaceDataAnalytics_Feature_Transformation.ipynb at https://git.io/fj7ea.

PGE 383
Feature Transformations

Lecture outline . . .

- **Feature Engineering**

Other Feature Engineering

Once again, methods and workflows that take raw data and transform it into predictor features that can be used in machine learning.

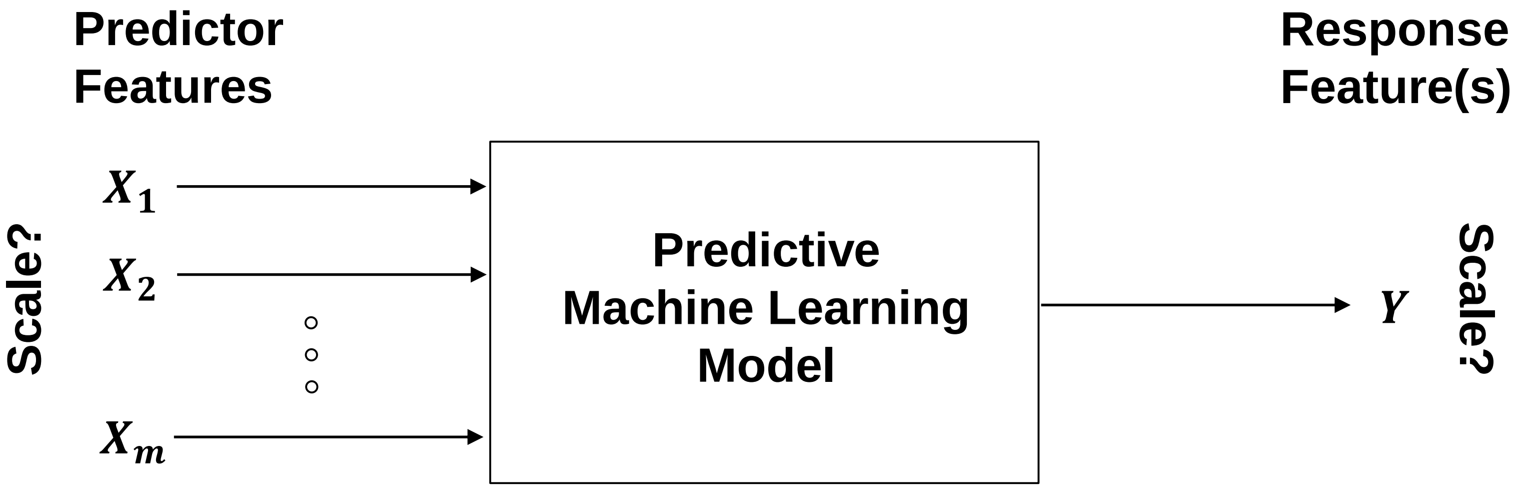
- data debiasing, uncertainty modeling, feature imputation, feature transformation, feature selection and feature projection (dimensionality reduction) are all forms of feature engineering

Let’s now provide a brief overview of other forms of feature engineering that focus on integrating domain expertise.

- scale
- new features / proxy models

Feature Engineering
Scale

All spatial data, the features associated with the machine learning model may have different scale.



Predictor Features

Response Feature(s)

X_1

X_2

X_m

Predictive Machine Learning Model

Y

Scale?

Scale?

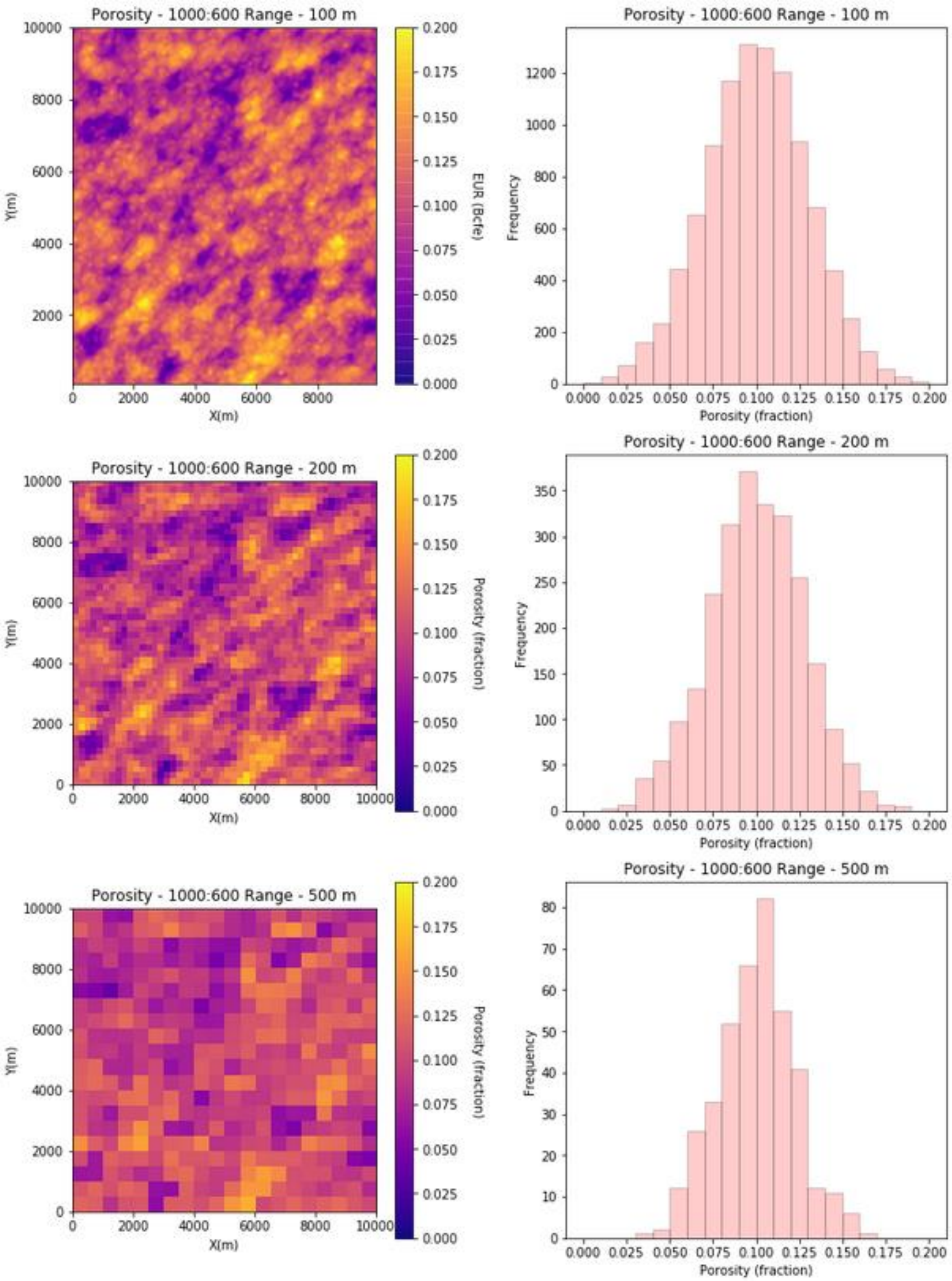
Feature Engineering Scale

What is the impact of scale?

- all statistics have an implicit scale choice
- therefore so do all statistical learning and machine learning models

Scale impacts the statistics and models.

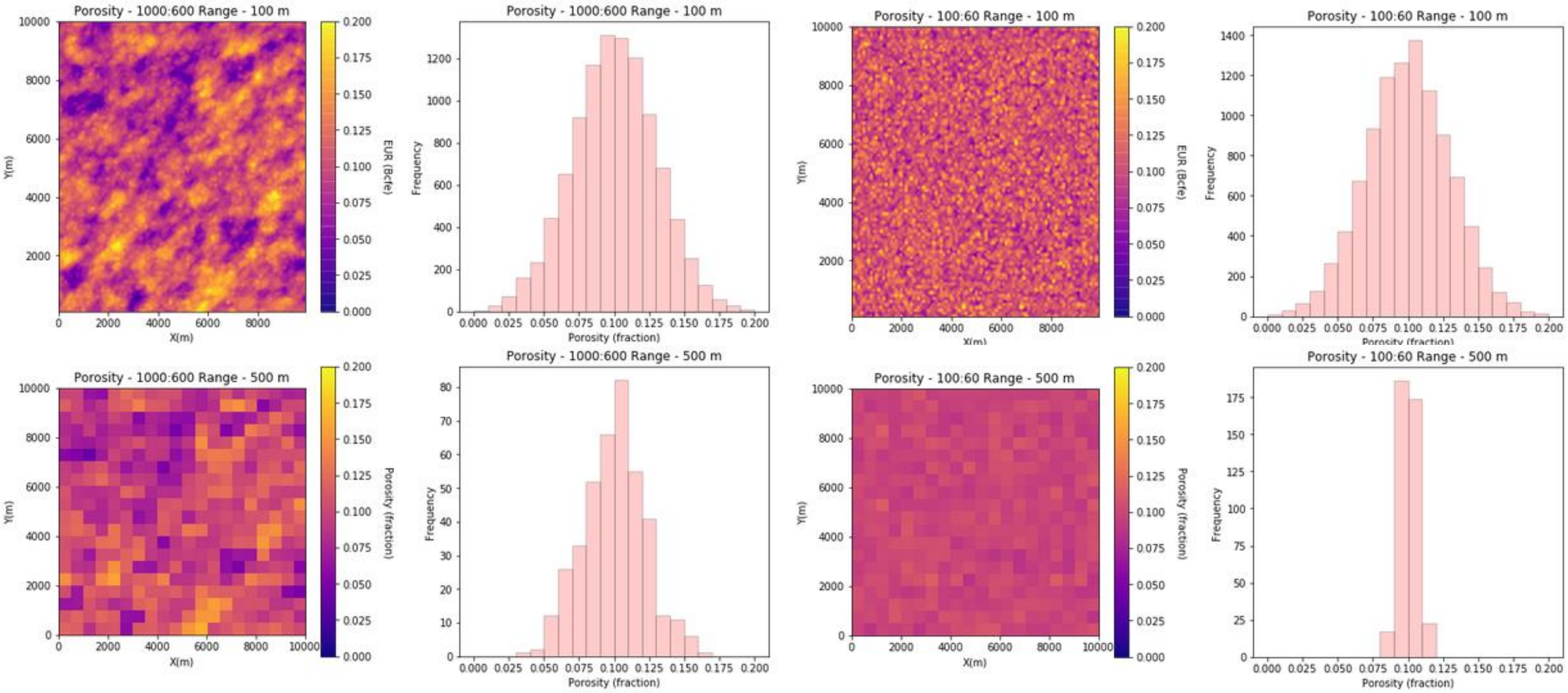
- see the change in variance as we change the scale of this 2D map of porosity.



The impact of scale on the histogram, see change in variance.

Feature Engineering Scale

The impact of scale on statistics, statistical / machine learning depends on spatial continuity.

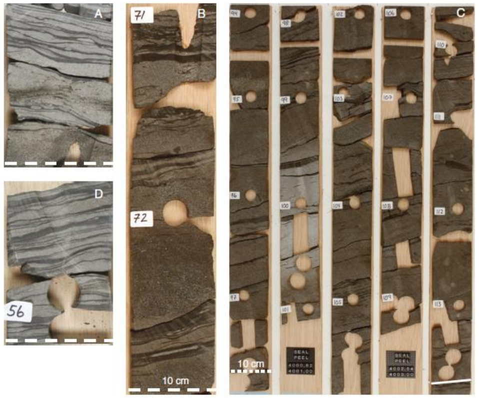


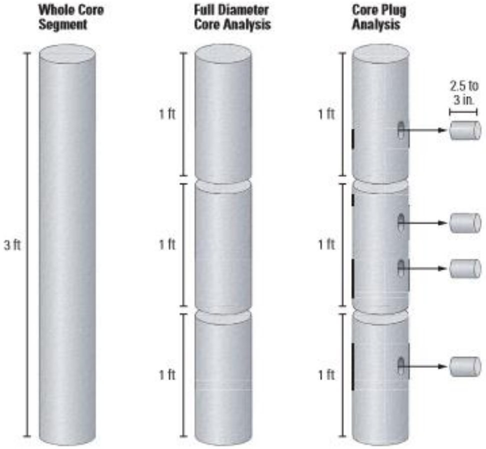
Anisotropic spatial continuity (left) and no spatial continuity (right), observe the difference in impact of change in scale on the histogram.

Feature Engineering Scale

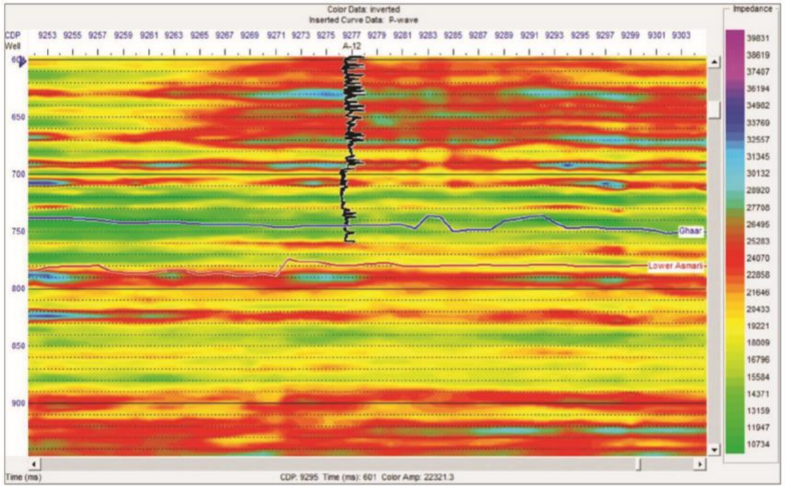
The data we work with has a wide variety of scales. In big data terms this is part of the ‘data variety’ challenge.

- feature scale is the volume over which a sample is representative.





^Divided cores. At the wellsite, whole cores are typically cut into smaller segments for ease of shipping. At the laboratory, the whole core segments may be cut and subsampled.



Well core-based features are at foot scale, while seismic-based features have a scale of 10’s of feet vertically and 100’s of feet horizontally.

Rigorous integration of scale is an ‘unsolved problem’. Issues such as missing scale remain because of data sparcity.

Feature Engineering
Scale

At the very least we need to ensure the dimensionality between the features is consistent.

- well logs are 1D data
- maps are 2D data
- seismic attributes are 3D data

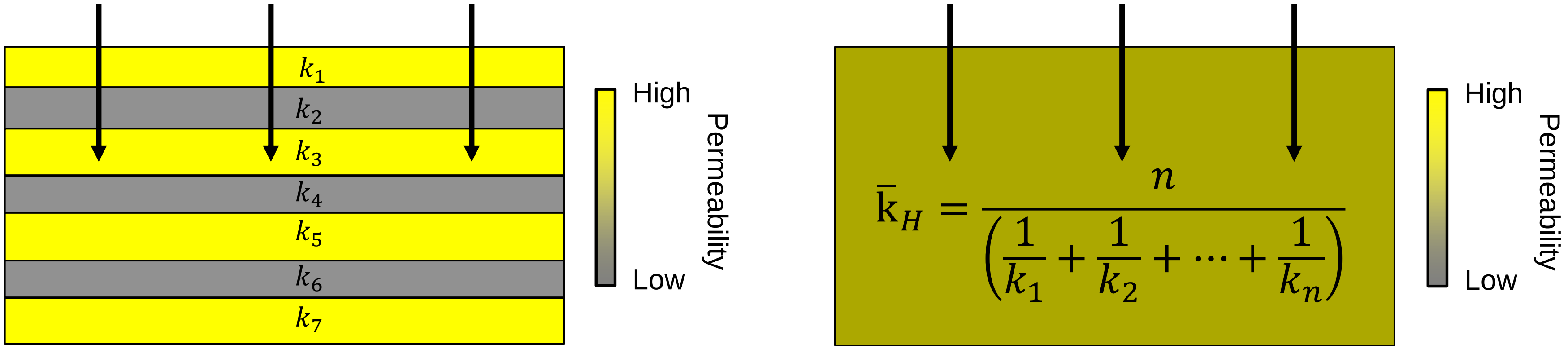
In this case, we need to build our models from features with consistent dimensionality.

- well logs (1D) → project into a model (3D)
- well logs (1D) → scale up, average over well and post on a map (2D)
- seismic (3D) → scale up, average over columns and post on a map (2D)

Feature Engineering
Scale Up

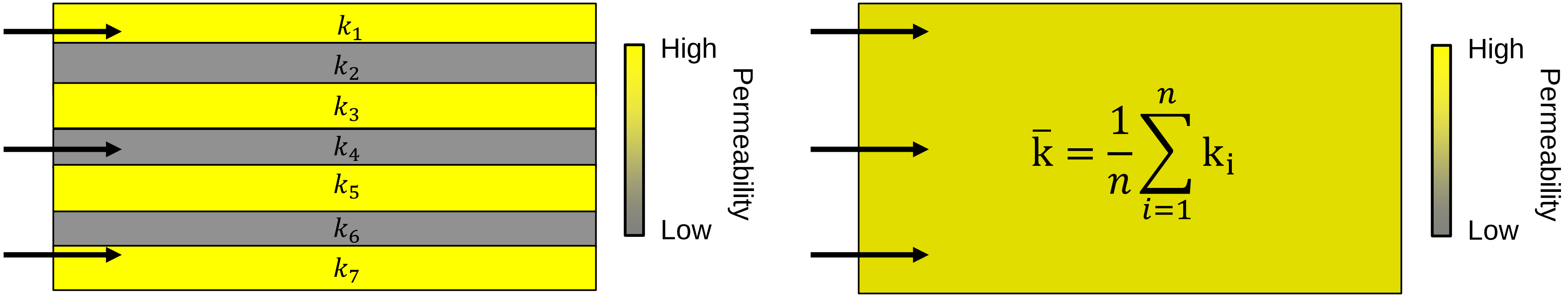
Another Interpretation of Central Tendency is Effective Property

- could I replace all the permeabilities of these layers with a single effective permeability?
  - when we apply flow simulation to both models they flow the same!



$$\bar{k}_H = \frac{n}{\left(\frac{1}{k_1} + \frac{1}{k_2} + \cdots + \frac{1}{k_n}\right)}$$

Harmonic mean is applied to calculate effective permeability for flow across layers, smallest permeabilities have the greatest impact.



$$\bar{k} = \frac{1}{n}\sum_{i=1}^{n} k_i$$

Arithmetic mean is applied to calculate effective permeability for flow along layers, extreme permeabilities have the greatest impact.

Feature Engineering
Scale Up

Another Interpretation of Central Tendency is Effective Property

- a more general form is **power law averaging**

$$\bar{\mathrm{x}}_P = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^p\right)^{\frac{1}{p}}$$

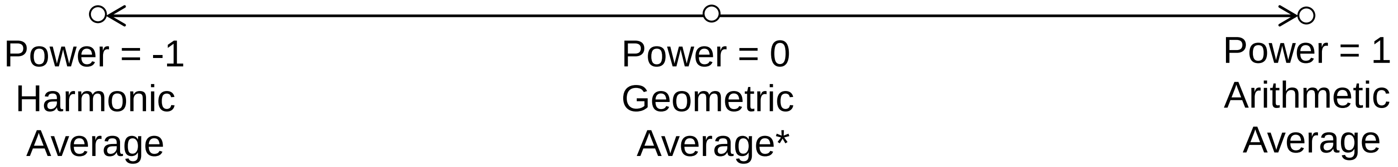
Power Law Averaging

$$K_{eff} = \left[\frac{1}{v}\int_v \;k(u)^p du\right]^{\frac{1}{p}}$$

Power Law Averaging for Volumetric Scale Up of Permeability

Example of continuous permeability power law upscaling.

- useful to calculate effective permeability where flow is not parallel nor perpendicular to distinct permeability layers
- flow simulation may be applied to calibrate (calculate the appropriate power for power law averaging)



Power = -1
Harmonic
Average

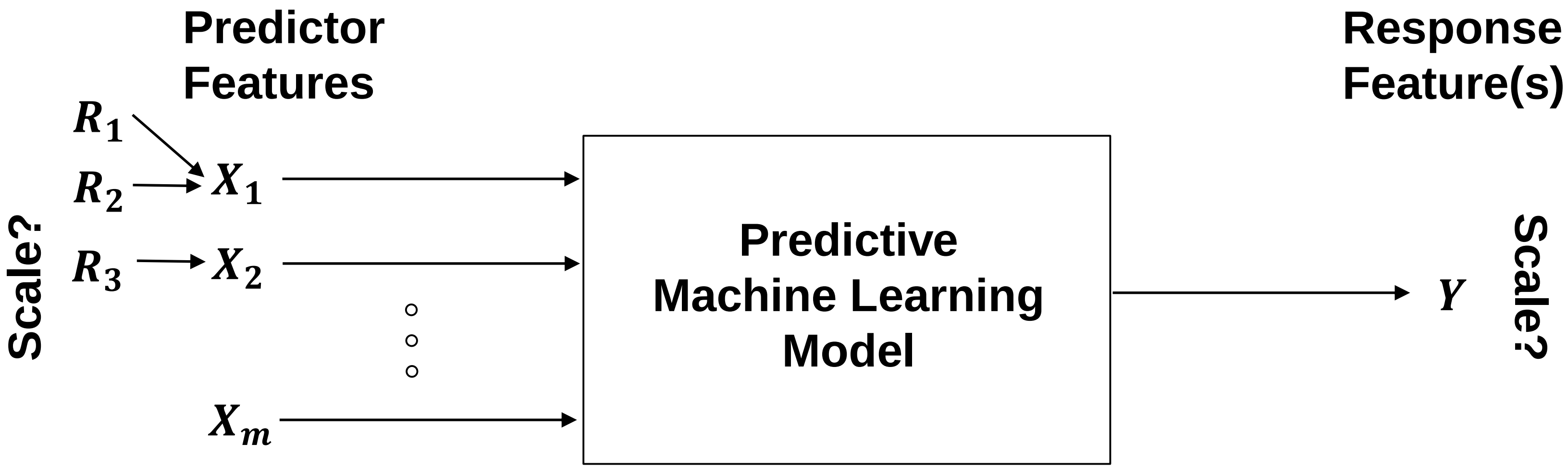
Power = 0
Geometric
Average*

Power = 1
Arithmetic
Average

* Proof in limit as p→0, see Zanon (2002) on Canvas.

Feature Engineering
New Features, Proxy Models

It may be useful to develop our own features based on the raw data features.



Examples:

- rock quality index is commonly applied, K_i/φ_i
- spatial, distance and volumetric calculations, OIP, effective porosity
- truncations such as setting low permeability rock to 0 porosity

Multivariate New Tools

Topic	Application to Subsurface Modeling
Curse of Dimensionality	Reduce problem to lowest dimension possible. *Feature ranking determined that porosity may be predicted from acoustic impedance and rock type alone.*
Feature Transformation	Apply feature transformations to improve the ability of your models to robustly infer patterns and predict away from training data. *Know what transformation are helpful and required for your modeling workflow.*
Feature Engineering	Ensure features have consistent scale and dimensionality. *Develop and use the most informative features to build a good prediction model.*

PGE 383
Feature Transformations

Lecture outline . . .

- Feature Transformations
- Feature Transformations Examples
- Other Feature Engineering

Michael Pyrcz, The University of Texas at Austin