PGE 338 Data Analytics and Geostatistics

Lecture 9: Bivariate Modeling

Lecture outline . . .

- Quantile-Quantile (Q-Q) Plots
- Regression Analysis

Introduction

General Concepts

Univariate

Bivariate

- Correlation
- Regression
- Model Checking

Time Series Analysis

Spatial Analysis

Machine Learning

Uncertainty Analysis

Michael Pyrcz, The University of Texas at Austin

Motivation

Learn to use bivariate methods to:

- compare and transform distributions
- build a model to make predictions
  - this is our 1st introduction to machine learning!

PGE 338 Data Analytics and Geostatistics

Lecture 9: Bivariate Modeling

Lecture outline . . .

- Quantile-Quantile (Q-Q) Plots

Introduction

General Concepts

Univariate

Bivariate

- Correlation
- Regression
- Model Checking

Time Series Analysis

Spatial Analysis

Machine Learning

Uncertainty Analysis

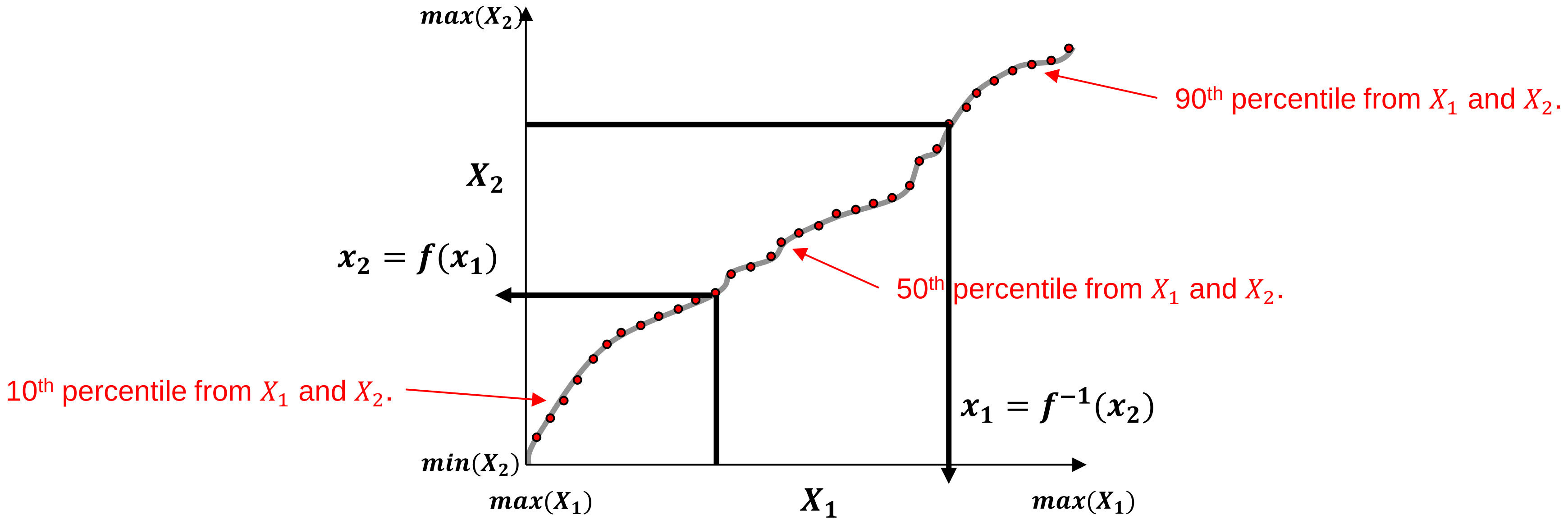
Michael Pyrcz, The University of Texas at Austin

Quantile-Quantile Plots

Q-Q Plot:

- Convenient method to graphically compare two distributions by scatter plotting the same percentiles from both distributions, note a quantile is synonymous with with percentile.

- The Q-Q plot is also the transform function to move from one distribution to another, distribution transform



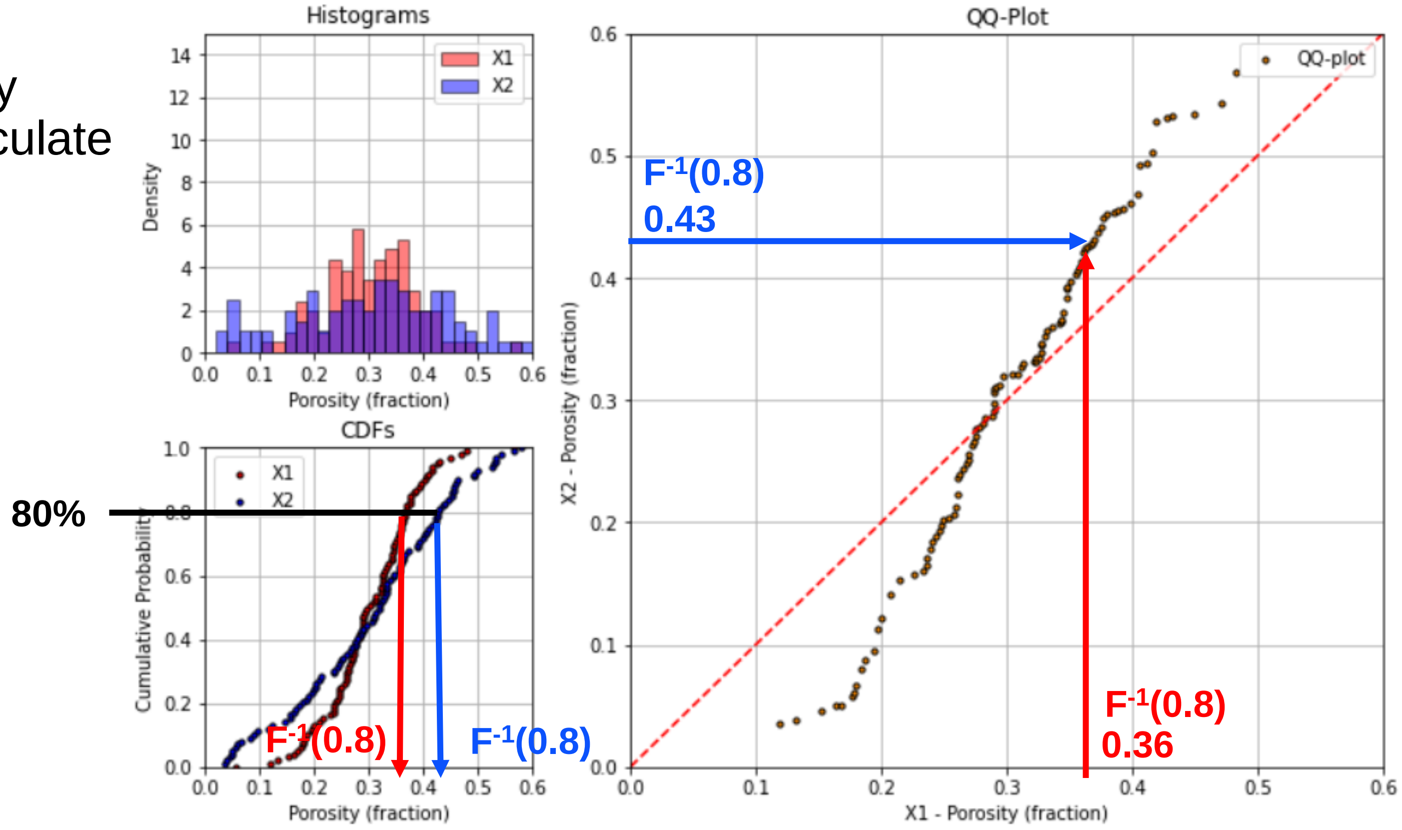
Q-Q plot between random variables $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ with example percentiles labelled and model fit for distribution transformation (grey line).

Quantile-Quantile Plots

Convenient method to graphically compare two distributions

Plot the equivalent percentiles between two distributions.

- Repeat over many percentiles to calculate the Q-Q plot

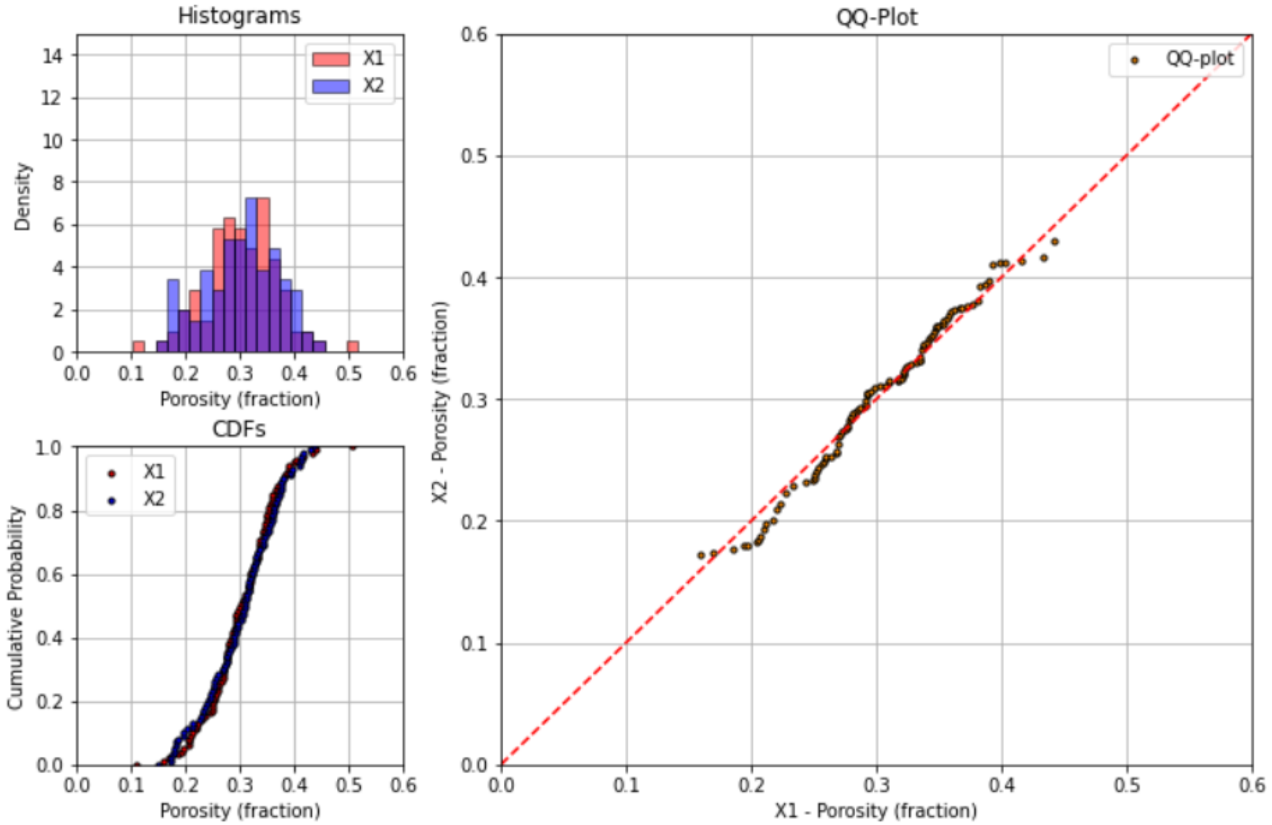


Calculation of a point on a QQ-plot, for the 80% percentile value, file is PythonDataBasics_QQ_Plot.ipynb.

Quantile-Quantile Plots

Convenient method to graphically compare two distributions

Example: Compare porosity well data from 2 reservoir formations



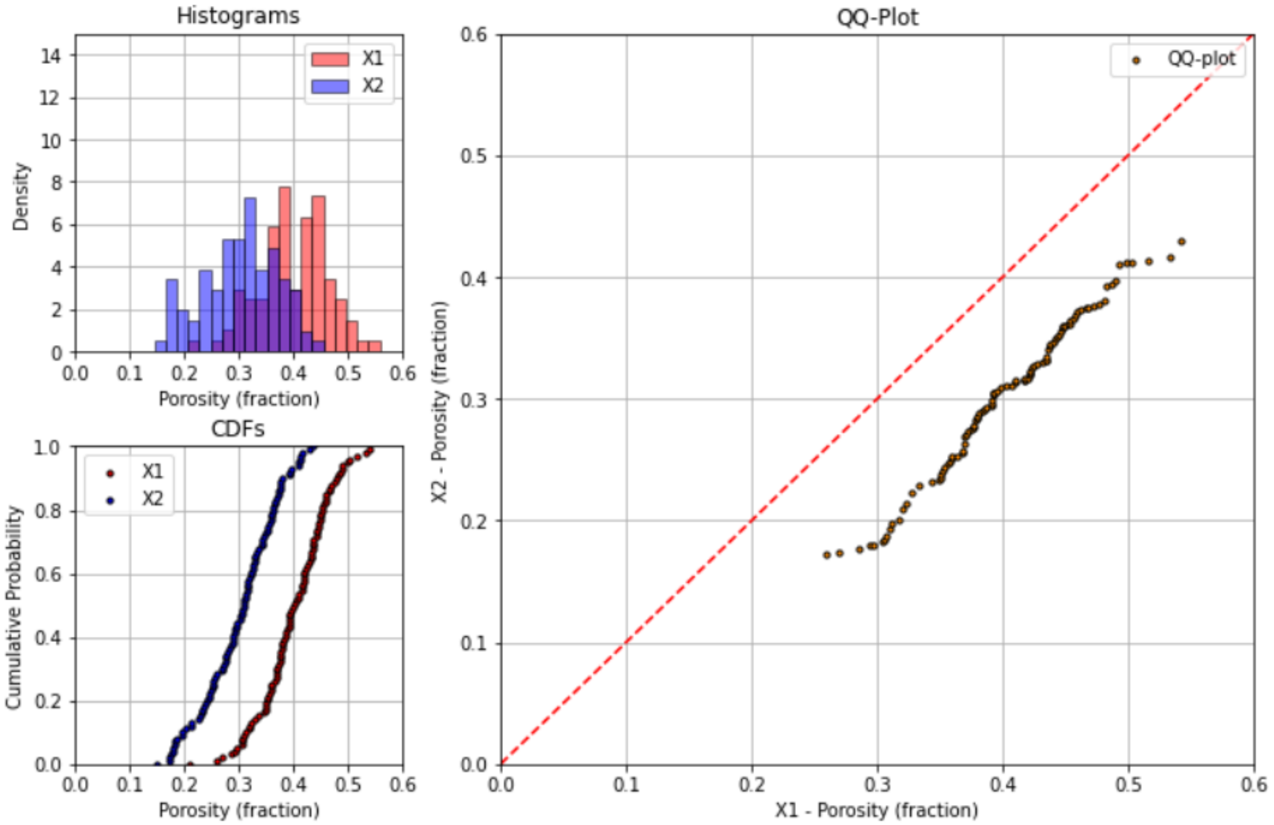
QQ-plot for 2 similar distributions, file is PythonDataBasics_QQ_Plot.ipynb.

- Interpretation, On the 45 Degree Line: reservoir formations X_1 and X_2 have the same distributions.

Quantile-Quantile Plots

Convenient method to graphically compare two distributions

Example: Compare porosity well data from 2 reservoir formations



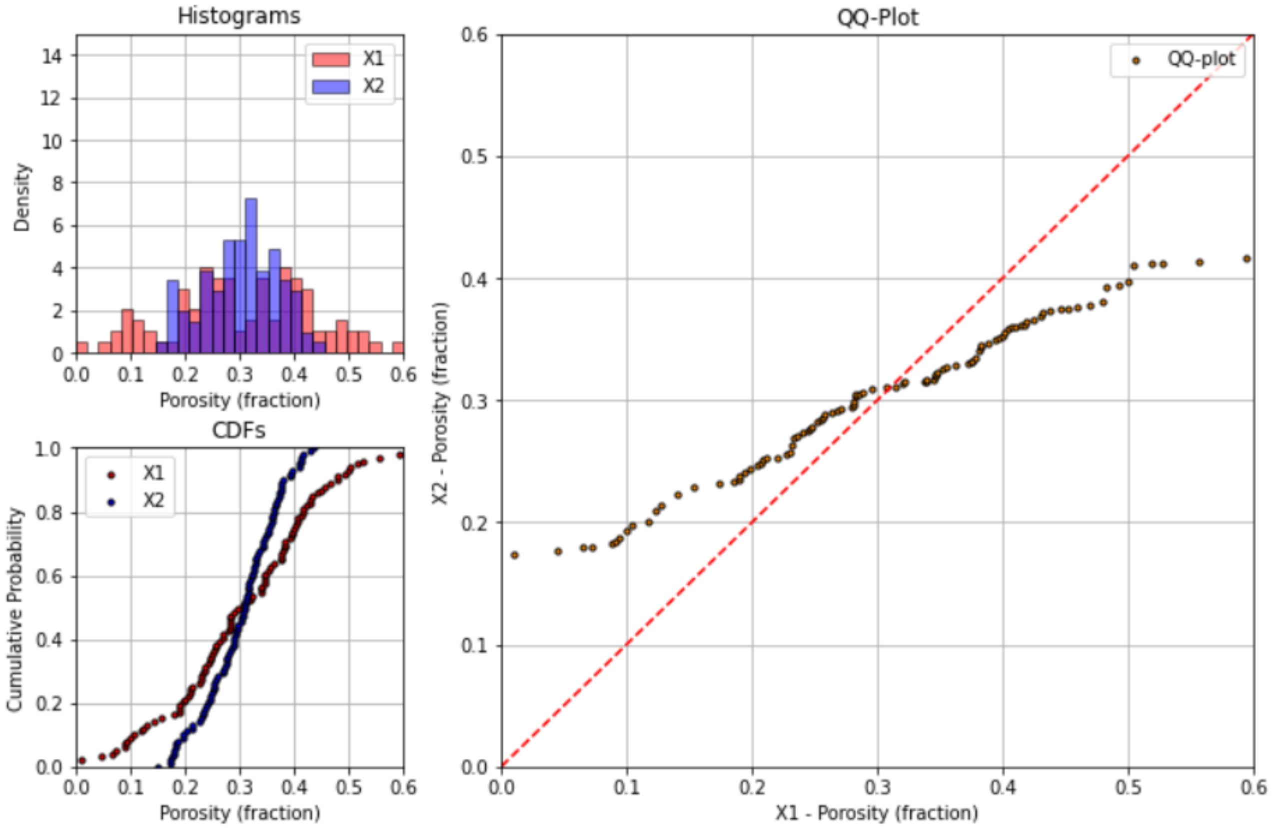
QQ-plot for for 2 distributions with mean X_1 > mean X_2., file is PythonDataBasics_QQ_Plot.ipynb.

- Interpretation, Offset from 45 degree: reservoir formations $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ have different means

Quantile-Quantile Plots

Convenient method to graphically compare two distributions

Example: Compare porosity well data from 2 reservoir formations



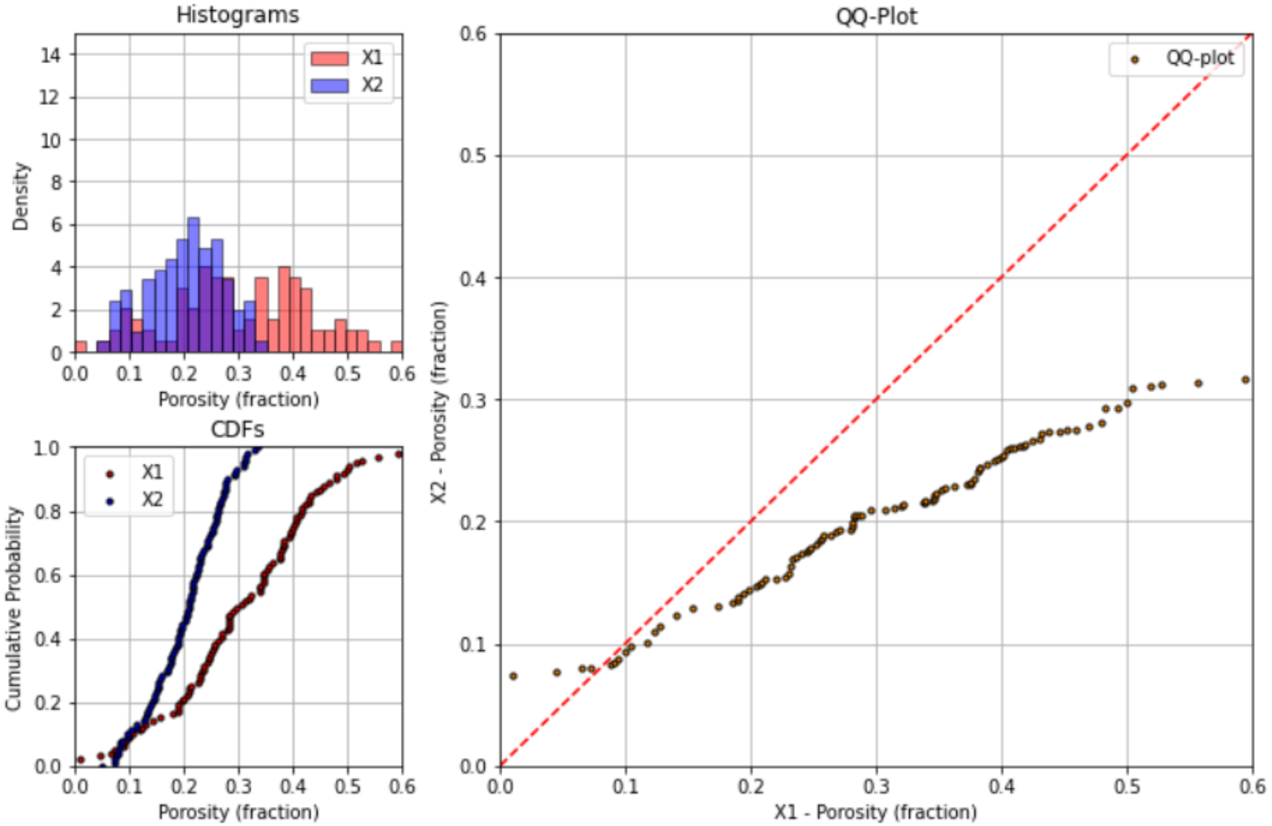
QQ-plot for for 2 distributions with variance X_1 > variance X_2., file is PythonDataBasics_QQ_Plot.ipynb.

- Interpretation, Change in Slope: reservoir formations $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ have different variances.

Quantile-Quantile Plots

Convenient method to graphically compare two distributions

Example: Compare porosity from 2 formations



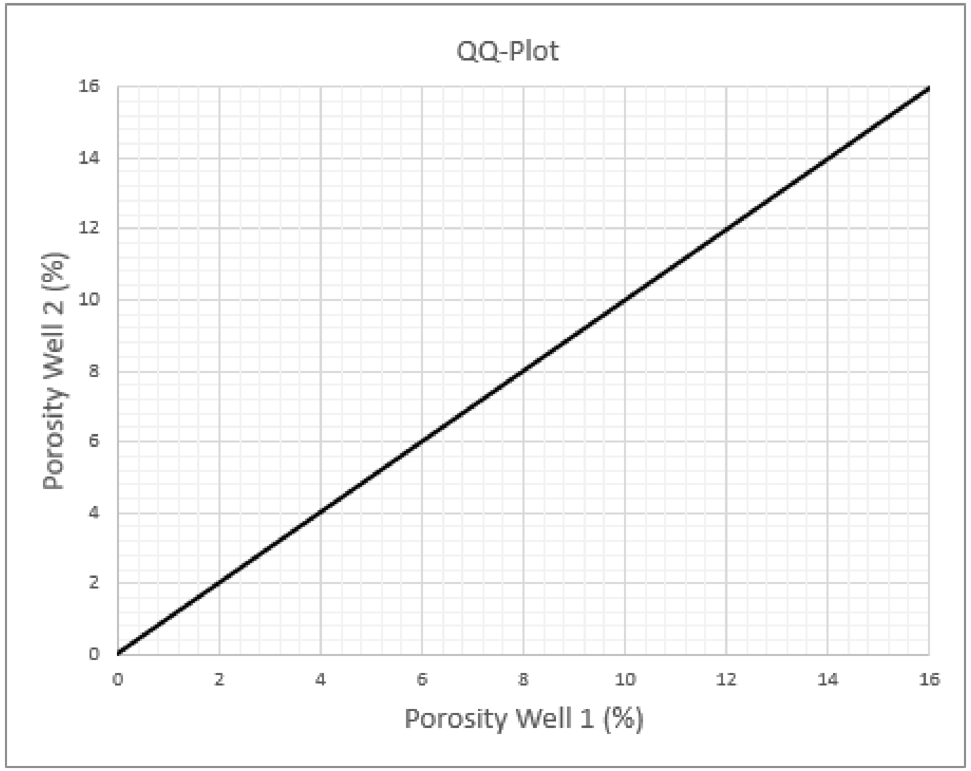
QQ-plot for for 2 distributions with variance X_1 > variance X_2, and mean X_1 > mean X_2, file is PythonDataBasics_QQ_Plot.ipynb.

- Interpretation, Offset from 45 degree: different means, and Change in Slope: different variances.

Quantile-Quantile Plots Hands On

Take these porosity datasets ($n_1 = n_2 = 10$) and plot a Q-Q plot by hand.

Well 1	Well 2
5%	2%
6%	4%
6%	6%
7%	7%
7%	8%
8%	10%
8%	12%
9%	14%
10%	15%



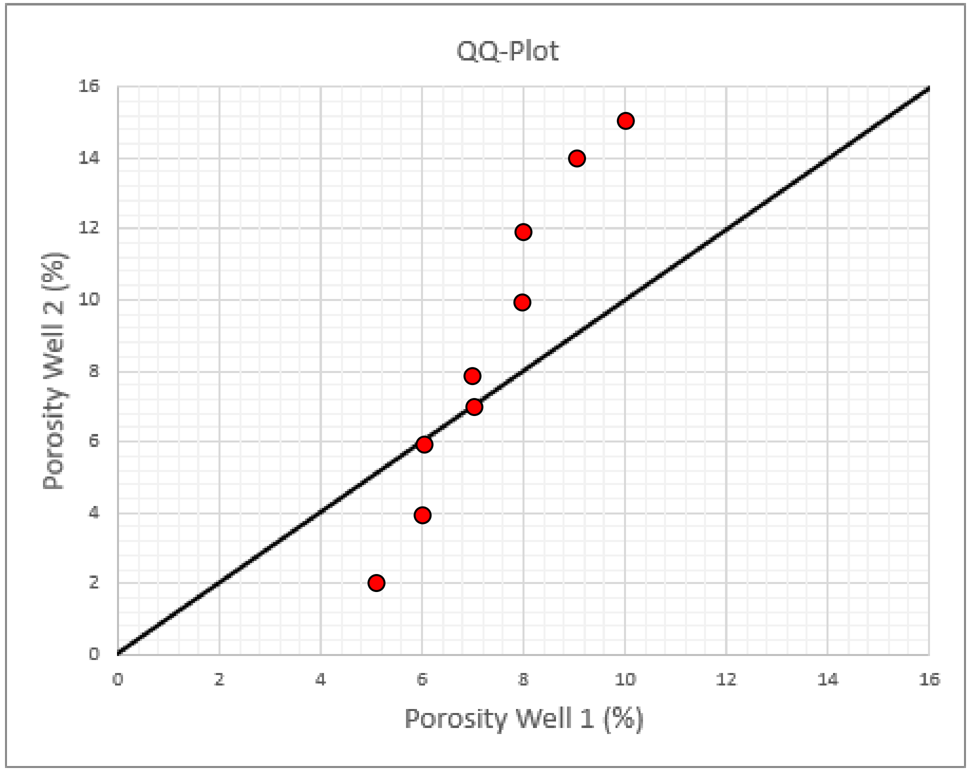
Data sets with same number of samples and blank Q-Q plot to complete.

- since each sorted separately and same number of samples, easy to pair up by percentiles.

Quantile-Quantile Plots Hands On

Take these porosity datasets ($n_1 = n_2 = 10$) and plot a Q-Q plot by hand.

Cumulative Prob.	Well 1	Well 2
9%	5%	2%
18%	6%	4%
	6%	6%
	7%	7%
	7%	8%
	8%	10%
	8%	12%
	9%	14%
91%	10%	15%



Data sets with same number of samples and completed Q-Q plot.

- since sorted separately and same number of samples, easy to pair up by cumulative probability.

Quantile-Quantile Plots

Distribution Transformations with Q-Q Plots

One can perform a distribution transform with a Q-Q plot

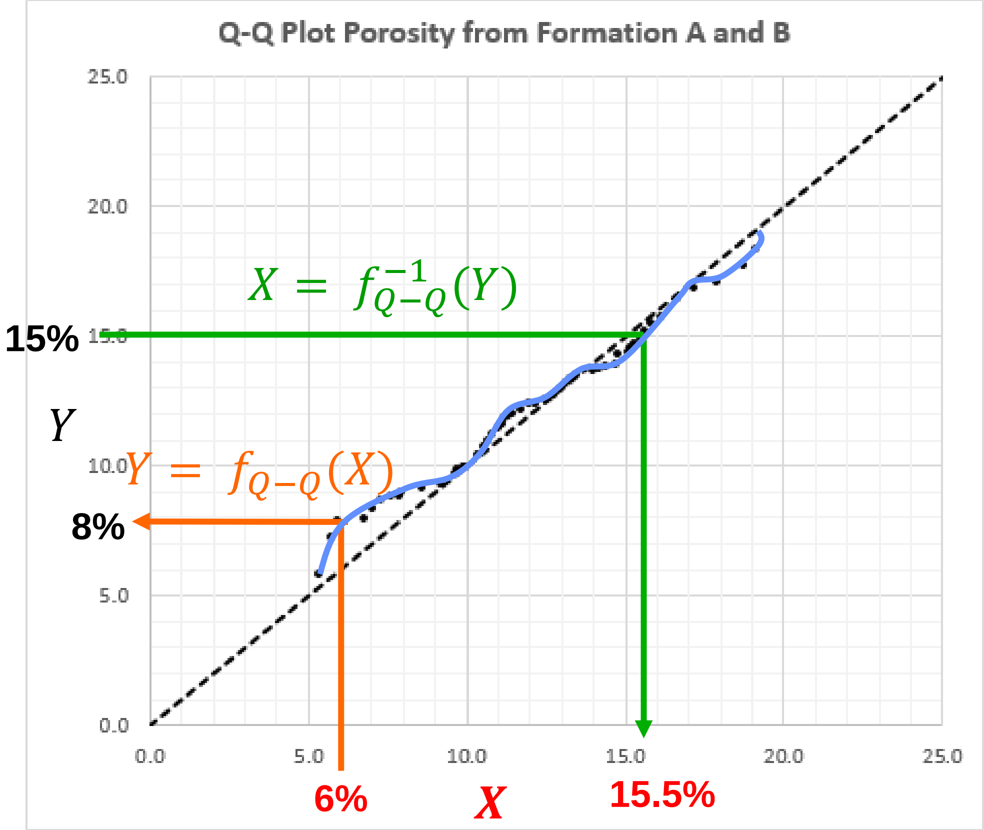
e.g. You could fit a function to a Q-Q plot

$$X = f_{Q-Q}^{-1}(Y) \text{ or } Y = f_{Q-Q}(X)$$

X is random variable distributed with CDF, F_X

Y is random variable distributed with CDF, F_Y

and f_{Q-Q} is the monotonic increasing function fit to the Q-Q plot

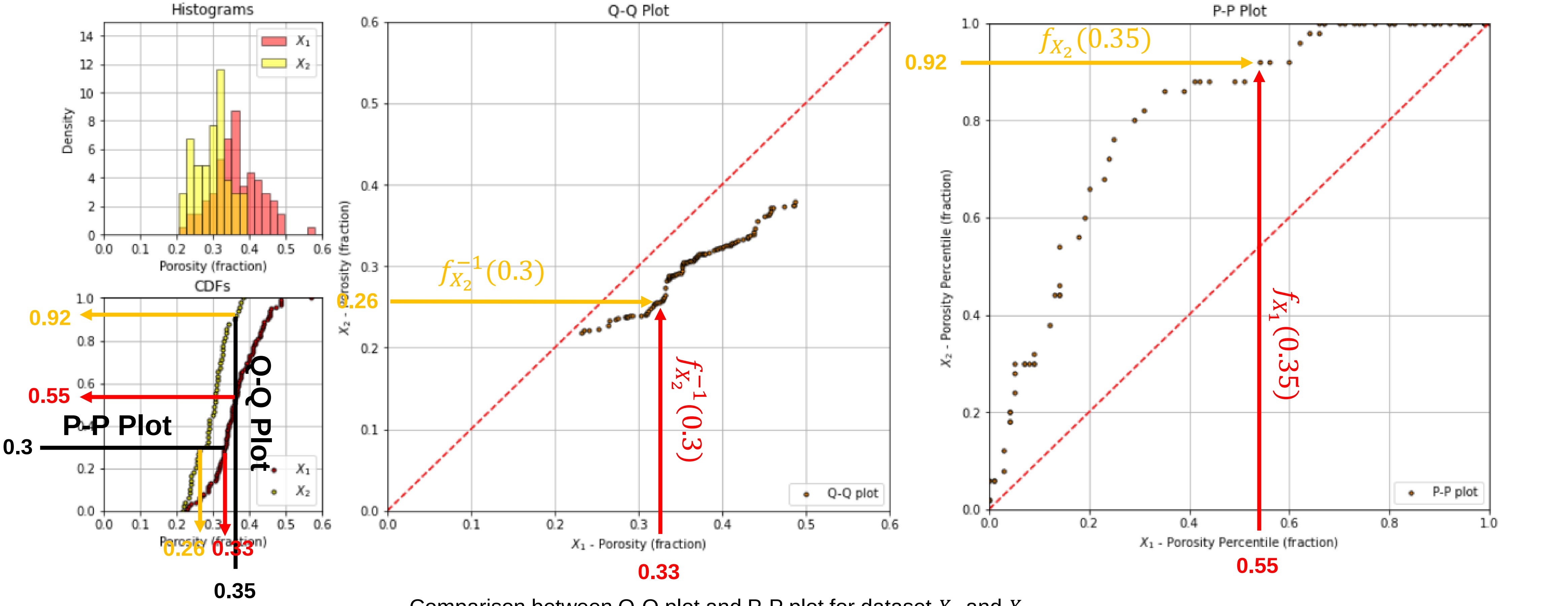


Q-Q plot with function fit to the points (blue line) for distribution transformations, F_X to F_Y (orange) and F_Y to F_X (green).

Probability-Probability Plots

Comparison to P-P plot – match the cumulative probabilities from same value.

- tails better expressed (difference magnified) on q-q plot, mode better expressed (difference magnified) with p-p plot, q-q plot is a distribution transform function.



Comparison between Q-Q plot and P-P plot for dataset X_1 and X_2.

PGE 338 Data Analytics and Geostatistics

Lecture 9: Bivariate Modeling

Lecture outline . . .

- Regression Analysis

Introduction

General Concepts

Univariate

Bivariate

Correlation

Regression

Model Checking

Time Series Analysis

Spatial Analysis

Machine Learning

Uncertainty Analysis

Michael Pyrcz, The University of Texas at Austin

Statistical / Machine Learning
Regression Analysis

Set up a machine to learn the relationship between features and a response from the data.

$$Y = f(X_1, \dots, X_m) + \epsilon$$

$X_1, \dots, X_n$ are the predictor features, Y is response reature and ϵ is error.

- We can repose this as:

$$\hat{Y} = \hat{f}(X)$$

where $\hat{f}$ is an estimate of the model and $\hat{Y}$ is the estimate of the response, these predictions are useful.

- Also our model has inferred relationship from the sample about the population.
  - We learn from the relationship between the features and predictors. $\widehat{\partial Y} = \hat{f}(\partial X)$
- Linear regression is the simplest form of machine learning. Let's cover it for the case of 1 feature and 1 response.
  - Later we return to machine learning and cover more complicated cases and models.

Bivariate Statistics
Regression Analysis

We can build a prediction model for porosity given density

- Perhaps porosity measures are not available everywhere and density available more frequently and is related to porosity.

- We want to build a linear prediction model of the form:

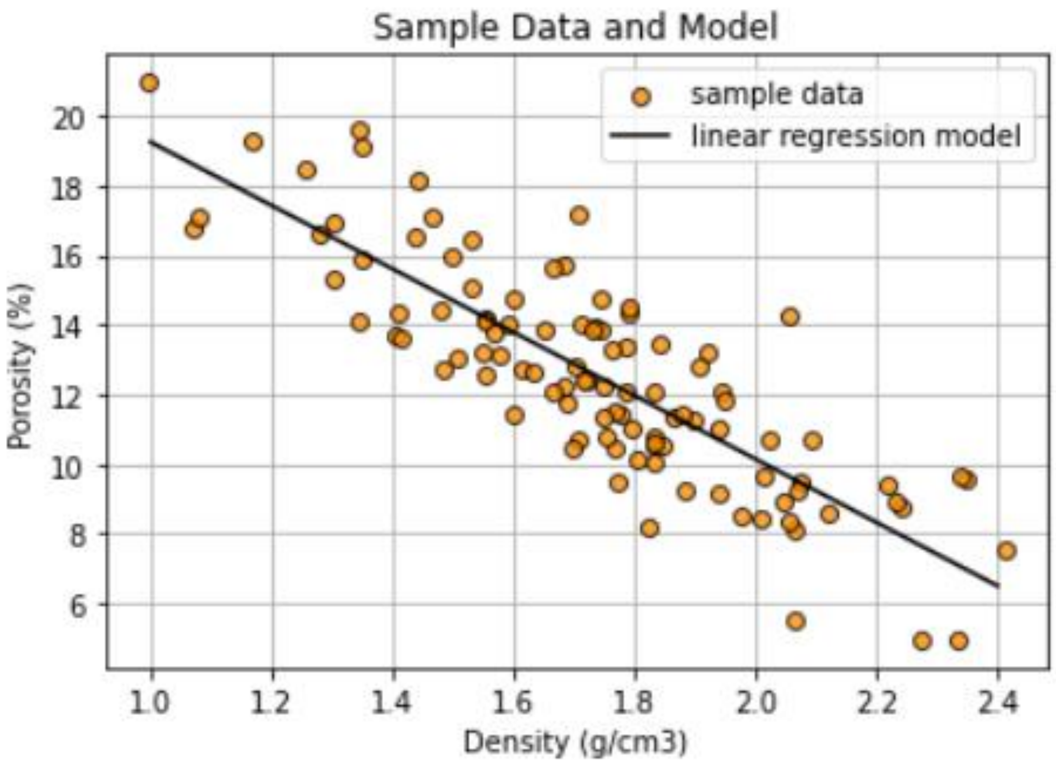
$$\boldsymbol{y = b_0 + b_1 x}$$

dependent variable
response feature

independent variable
predictor feature

- We can build a linear prediction model for porosity given density:

$$\boldsymbol{por = b_0 + b_1 density}$$



Porosity vs. density scatter plot and linear regression model.

Bivariate Statistics
Regression Analysis

Linear Regression Objective Function

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

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

data model

Sum of Square Error

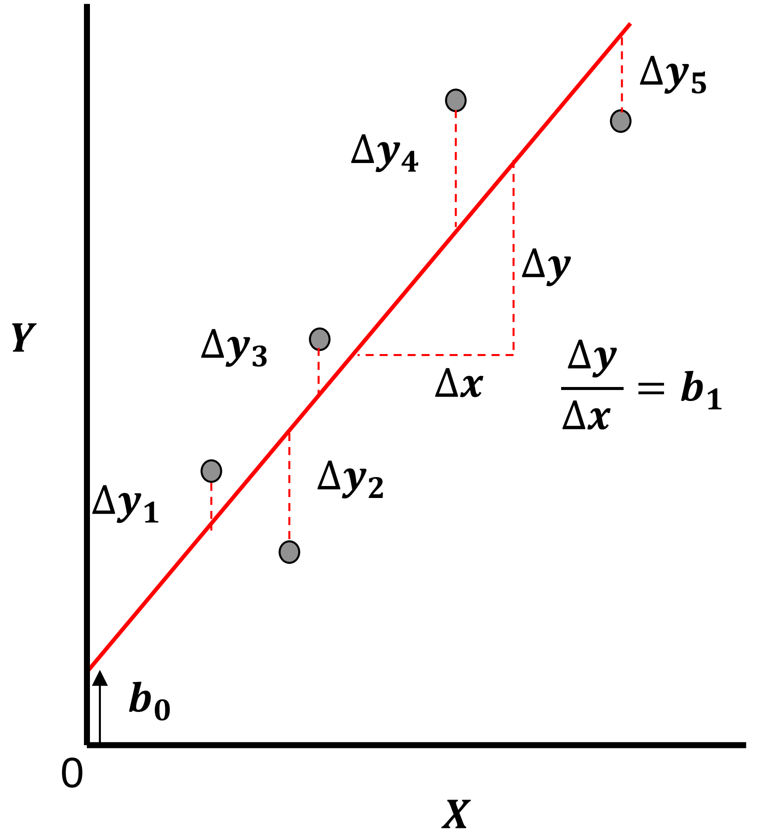
Minimize:

Loss Function

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

Skipped derivation.

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



Data, linear regression model and error terms.

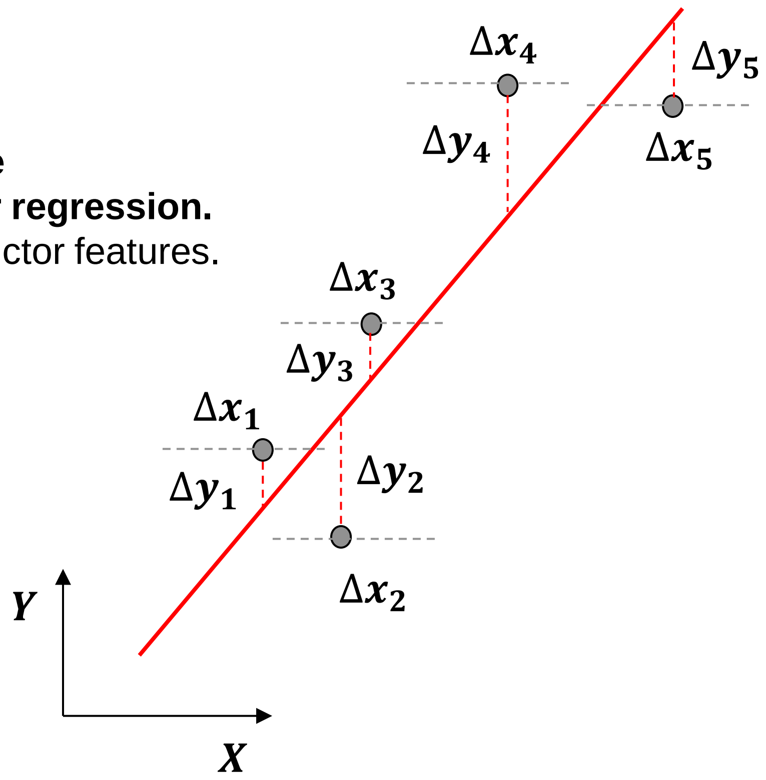
Bivariate Statistics
Regression Analysis

Linear Regression Assumptions:

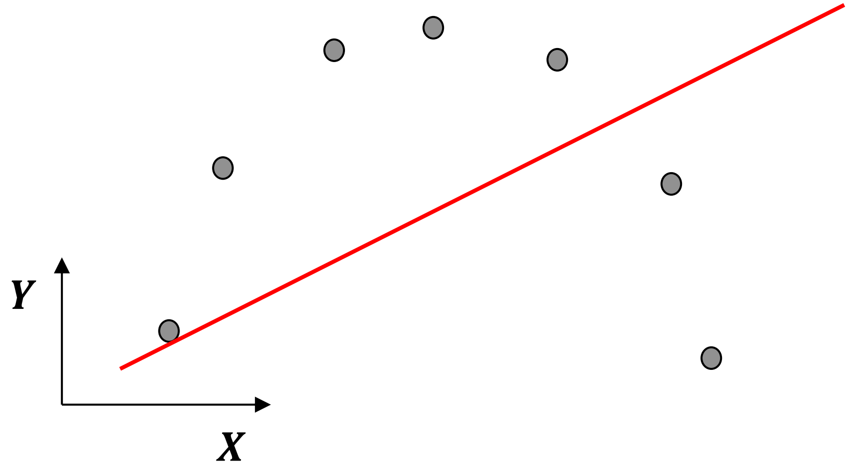
- **Error-free:** predictor variables are error free, not random variables

$$\Delta x_i = 0, \forall\, i = 1, \dots, n$$

This would violate the assumptions of linear regression.
- error in the x_i, predictor features.



- **Linearity:** response is linear combination of feature(s)



This would violate the assumptions of linear regression.
- nonlinear data structures.

Bivariate Statistics
Regression Analysis

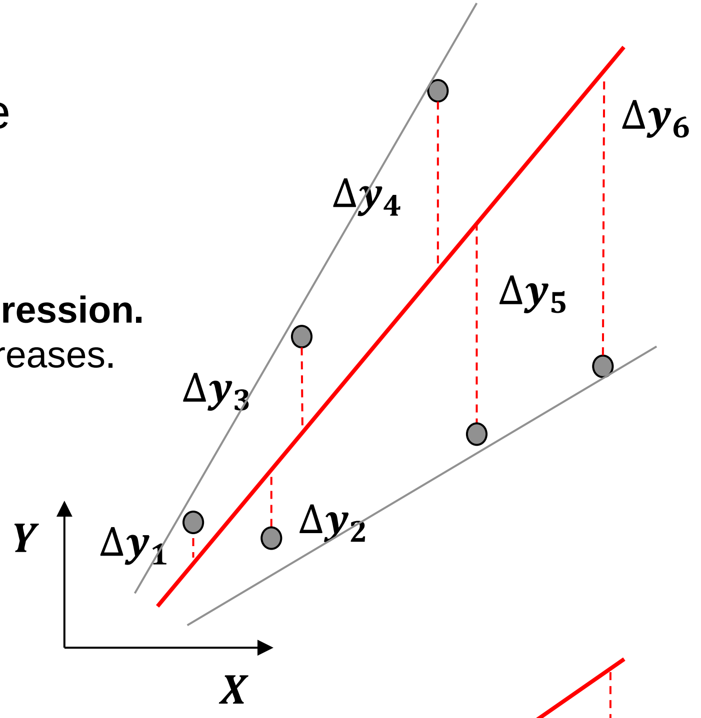
Linear Regression Assumptions:

- **Constant Error Variance:** error in response is constant over predictor(s) value

$$E\{\Delta y | x\} = E\{\Delta y\}, \forall\, x$$

This would violate the assumptions of linear regression.

- error increases as x increases.

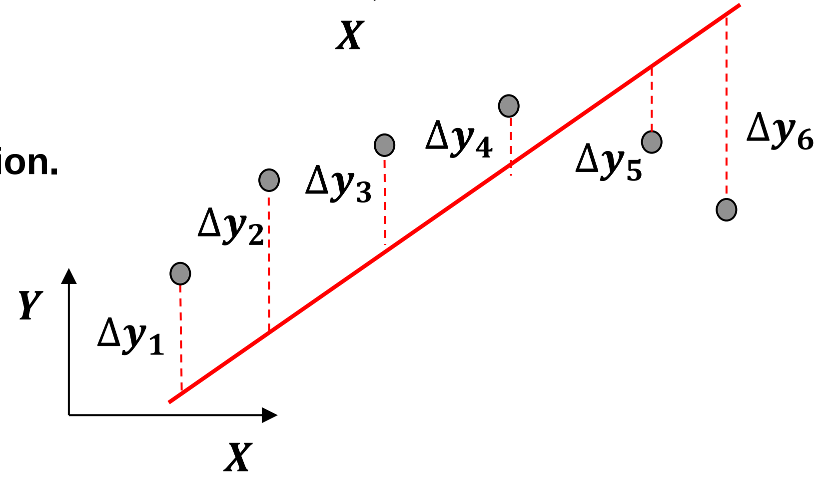


- **Independence of Error:** error in response are uncorrelated with each other

$$C_{\Delta y}\,(\mathbf{h_x}) = 0, \forall\, h_x$$

This would violate the assumptions of linear regression.

- correlated, consistent error



note, we will introduce $C(\mathbf{h})$ covariance function in Spatial Correlation.

Bivariate Statistics
Regression Analysis

Linear Regression Assumptions:

- **No multicollinearity:** none of the freatures are redundant with other features

$$X_i = \sum_{\in m} \beta_j X_j, \;\; where\ i \neq j$$

no feature is a linear combination of other feature(s).

Regression Analysis

Fit a Linear Model to Predict Porosity from Density

- linregress function in SciPy package, stats module
- load the data from a comma delimited file or Excel spreadsheet into a Pandas DataFrame

	Density	Porosity
0	1.281391	16.610982
1	1.404932	13.668073
2	2.346926	9.590092
3	1.348847	15.877907
4	2.331653	4.968240

DataFrame preview with DataFrame.head().

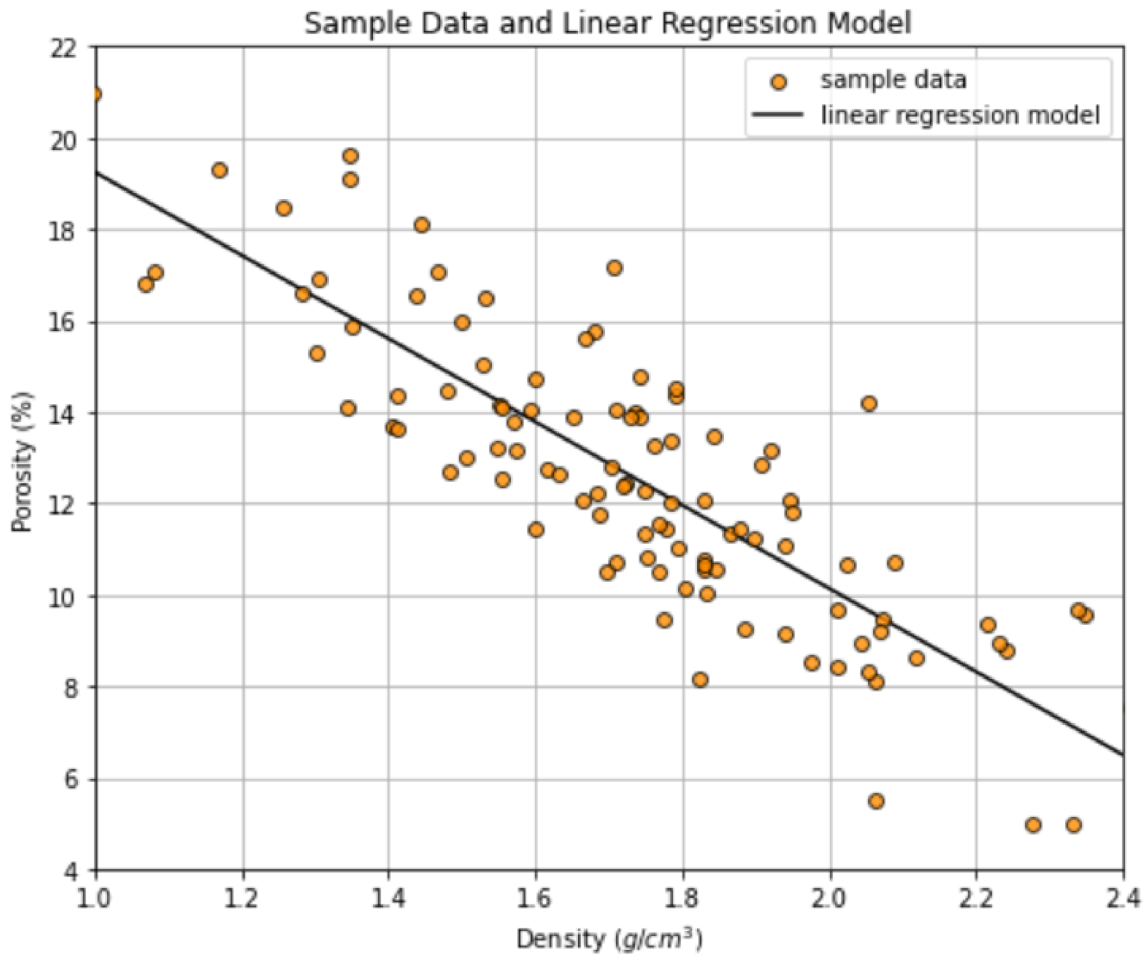
- instantiate and train the model

```
linear = st.linregress(den,por)
```

SciPy stats function for linear regression.

- visualize the model, predict $\hat{y}$ by applying slope and intercept to a vector of density values

$$\boldsymbol{y = b_0 + b_1 x}$$



Training data and linear regression model.

Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

Regression Analysis

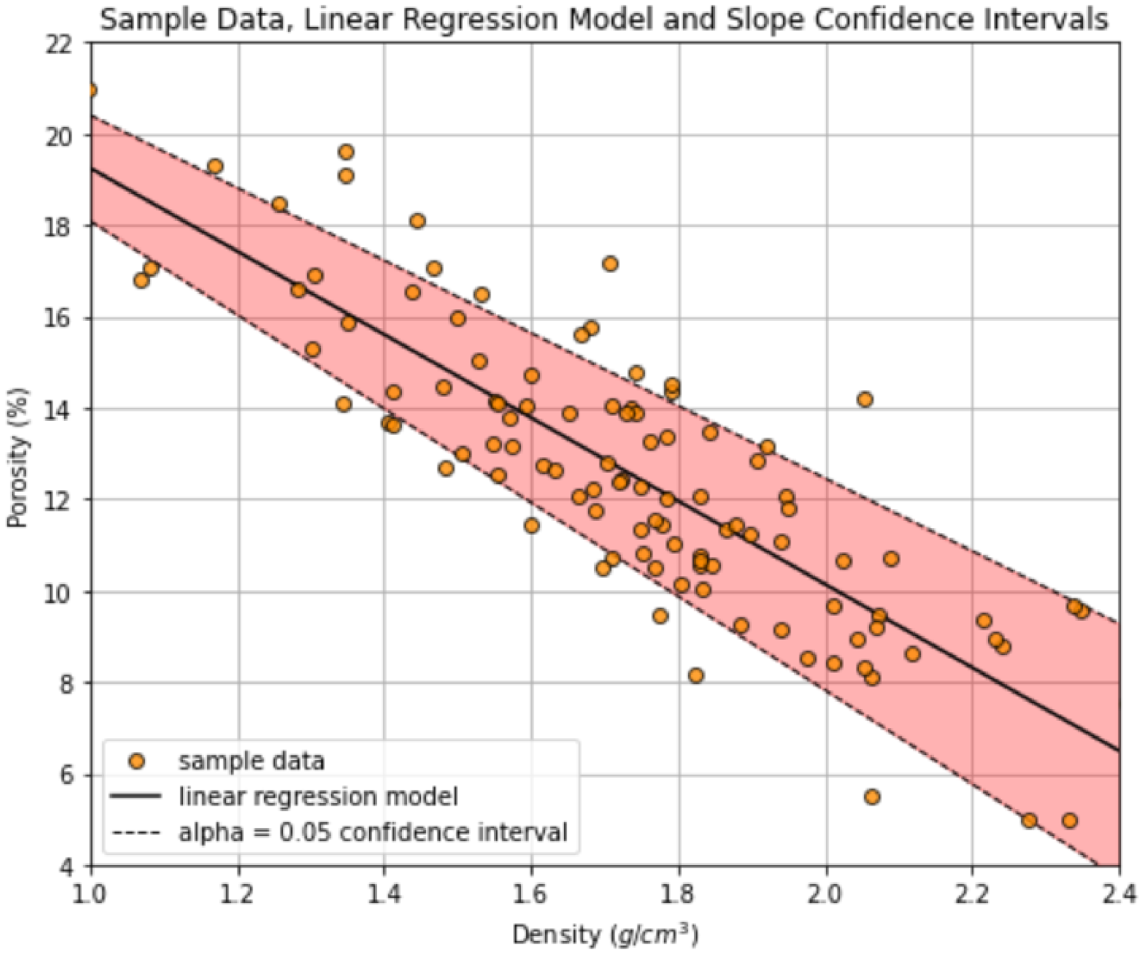
Model Uncertainty with Slope Confidence Interval

- recall the slope, $\widehat{b_1}$, and intercept, $\widehat{b_0}$, confidence intervals

$$\widehat{b_1} \pm t_{(\alpha/_2, n-2)} \times SE_{b_1} \quad \widehat{\boldsymbol{b_1}} \pm t_{\alpha/2, n-2} \times \left(\frac{\sqrt{n}\hat{\sigma}}{\sqrt{n-2}\sqrt{\sum(x_i - \bar{x})^2}} \right)$$

$$\widehat{b_0} \pm t_{(\alpha/_2, n-2)} \times SE_{b_0} \qquad \widehat{\boldsymbol{b_0}} \pm t_{\alpha/2, n-2} \times \left(\sqrt{\frac{\hat{\sigma}^2}{n-2}} \right)$$

- the slope standard error, SE_{b_1}, is an output and we can calculate the t-value, $t_{\frac{\alpha}{2}, n-2}$, with SciPy.stats.t class.
- the slope upper and lower bounds from the confidence interval are applied to calculate the upper and lower model to form the red confidence interval model envelope.



Training data, linear regression model and slope confidence interval.

Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

Regression Analysis

Model Uncertainty with Prediction Interval

- given predictions at training data, $\hat{y}$, we calculate the uncertainty in the next prediction, $\hat{y}_{n+1}$, as:

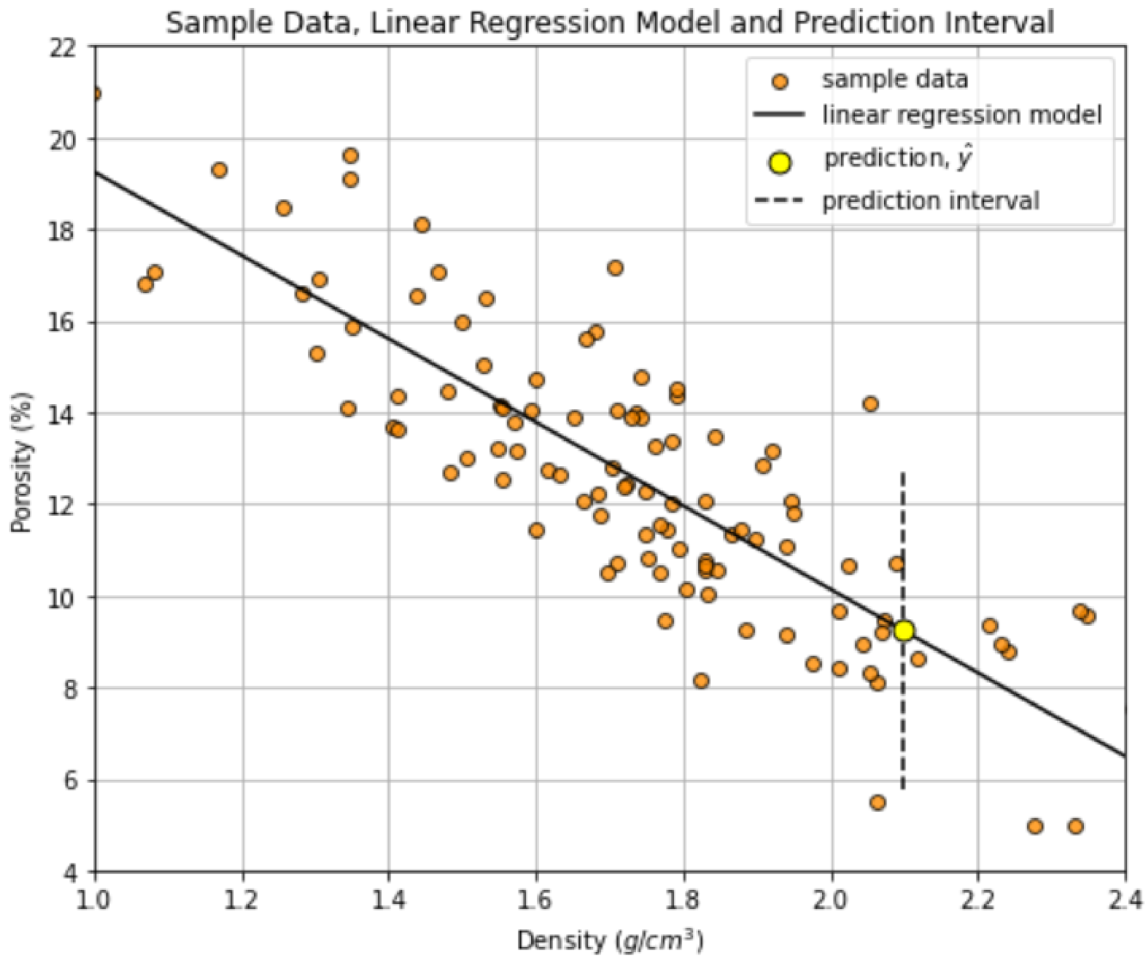
$$\hat{y}_{n+1} \pm t_{\alpha/2,n-2} \boxed{\sqrt{MSE}\sqrt{1+\frac{1}{n}+\frac{(x_{n+1}-\bar{x})^2}{\sum(x_i-\bar{x})^2}}}$$

t-statistic

standard error of the prediction

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

- We answer the question, given I know the density, x_{n+1}, what is the interval with 1-alpha probability containing the true value permeability, y_{n+1}?



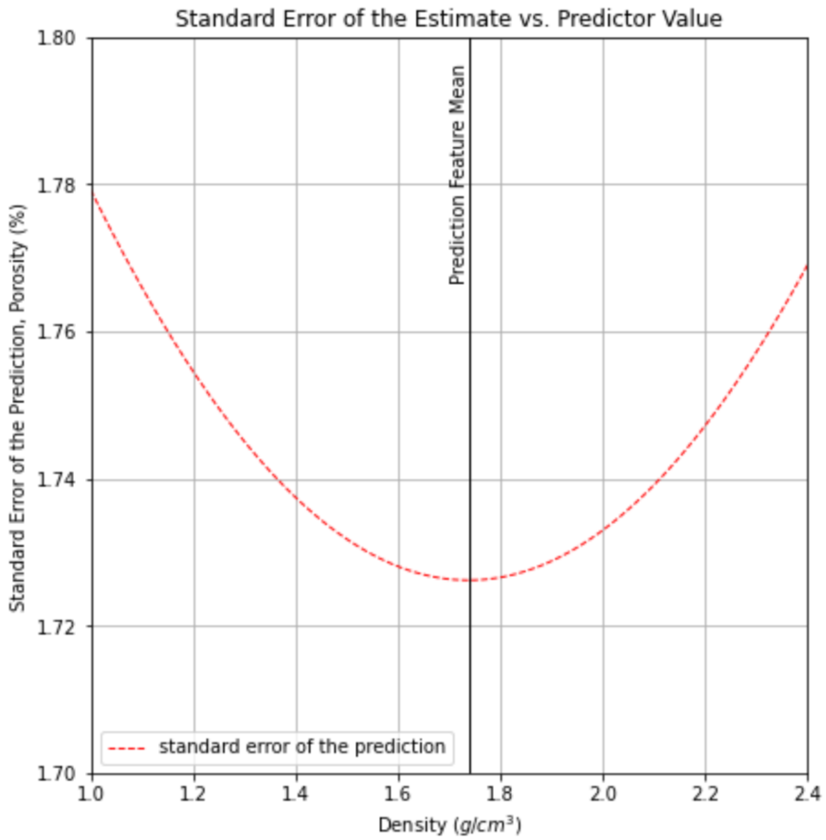
Training data, linear regression model and prediction interval.

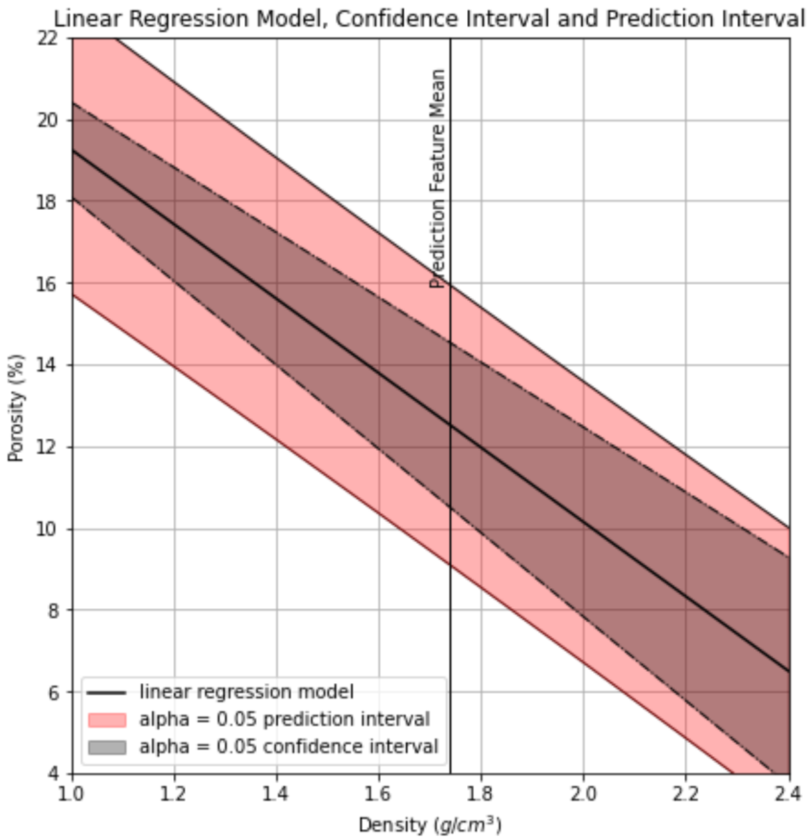
Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

Regression Analysis

Prediction Interval vs. Confidence Intervals

- Let’s visualize and compare prediction and confidence intervals





The prediction interval standard error or prediction vs. predictor value.

Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

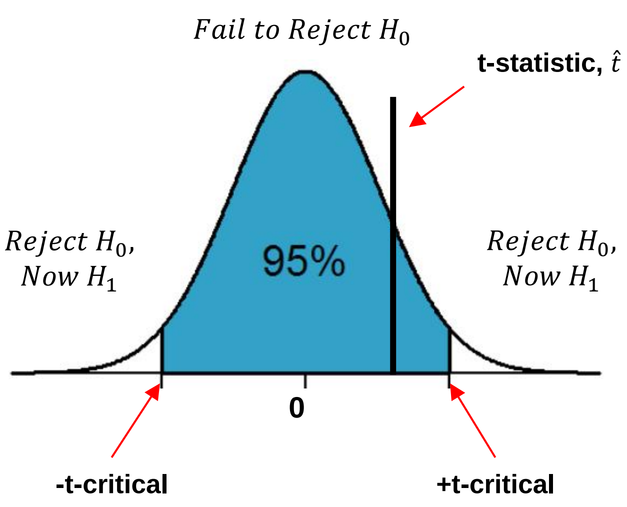
Regression Analysis

Model Checking with Hypothesis Test for Slope

- linregress function provides the slope standard error, SE_{b_1}, to test if significantly difference than 0

$$H_0: b_1 = 0$$
$$H_1: b_1 \neq 0$$

$$t_{stat} = \frac{b_1}{SE_{b_1}} \; and \; t_{critical} = t(n-2, \alpha/2)$$



Student's t distribution (mean = 0, standard deviation = 1.0)

- the slope model parameter p-value is also conveniently provided as an ouput
  - linregress.pvalue

- reject if p-value is less than the alpha value

```
print('The linear regression model slope parameter p-value is ' + str(round(linear.pvalue,10)) + '.')
```

```
The linear regression model slope parameter p-value is 0.0.
```

Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

Regression Analysis

r-square, r^2, Variance Explained

- For linear models we can calculate a convenient measure of model performance, the proportion of variance explained.

Variance explained by the model

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

Variance NOT explained by the model

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

- $\boldsymbol{r^2}$: strength of the model, proportion of variance explained by the model

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

Regression Analysis

Model Checking with Hypothesis Test for all Model Parameters at Once

- f-test for significance of all parameters at once

```
alpha = 0.05
k = 2
ms_model = np.sum(np.power(por-np.average(por),2))/(k-1)
ms_error = np.sum(np.power(por-por_hat,2))/(len(por)-k)
f_stat = ms_model/ms_error
f_crit = st.f.ppf(1-alpha, k-1, len(por)-k)
```

- Here's our hypothesis:

$$H_0: b_i = 0, \forall\, i$$

$$H_1: otherwise$$

Reject null hypothesis.

- Here's our f-stat and f-critical for this hypothesis test.

```
The f-stat is : 345.68 and the f-critical is : 3.933
Therefore we reject the null hypothesis, our model parameters are significant
```

$$f_{statistic} = \frac{Mean\ Squares\ of\ Model}{Mean\ Squares\ of\ Error} = \frac{\frac{\sum(\hat{y}_i - \bar{y})^2}{k-1}}{\frac{\sum(y_i - \hat{y})^2}{n-k}} = \frac{\frac{653}{1}}{\frac{380}{103}} = 177$$

$$f_{critical}(\alpha = 0.05, \nu_1 = k-1, \nu_2 = n-k) = 5.17$$

Linear regression step-by-step, model training and checking, file is PythonDataBasics_LinearRegression.ipynb.

Regression Analysis in Excel

Linear regression in Excel spreadsheet, model fitting, checking, etc.

- For your reference, all calculations are available in Excel “by-hand”

Linear Regression Calculations

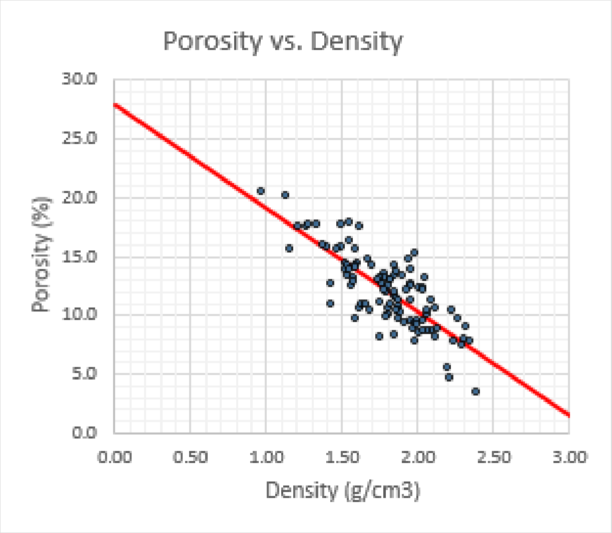
Model: $y = b_0 + b_1 x$

From Minimized Square Error: $b_1 = \frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sum_{i=1}^{n}(x_i-\overline{x})^2}$

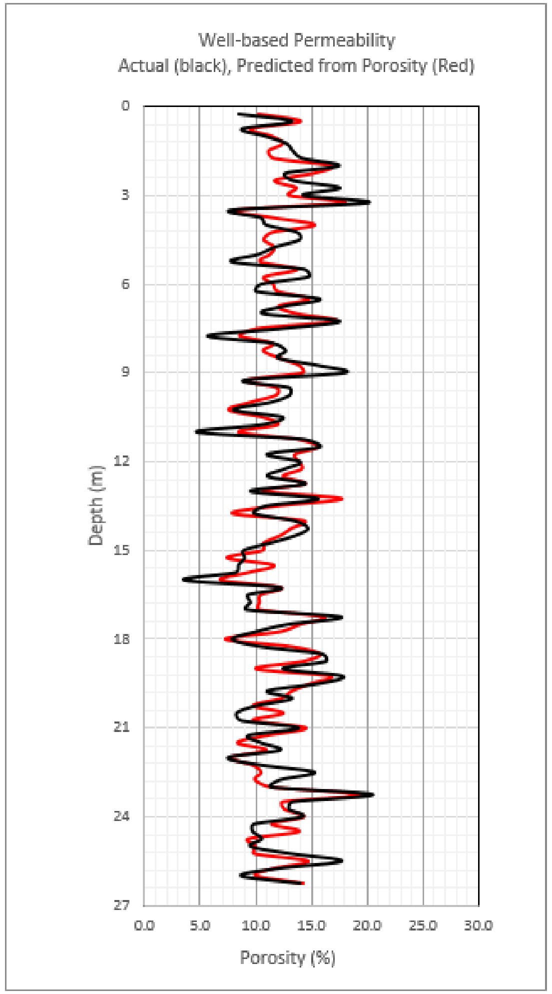
The find intercept: $b_0 = \overline{y} - b_1\overline{x}$

xi-E{x}	yi-E{y}	(xi-E{x})(yi-E{y})	(xi-E{x})^2	(yi-y*)^2	(yi-E{y})^2	Porosity*	Residual
0.20	6.61	1.33	0.04	3.45	13.15	10.27	1.86
-0.23	11.43	-2.61	0.05	0.67	1.43	14.05	0.82
0.28	6.91	1.96	0.08	0.68	11.04	9.54	0.83
0.07	9.01	0.60	0.00	0.41	1.50	11.45	0.64
-0.04	10.92	-0.48	0.00	0.09	0.48	12.42	-0.31
0.10	11.49	1.17	0.01	4.64	1.59	11.14	-2.15
0.05	12.35	0.57	0.00	6.35	4.47	11.63	-2.52
-0.54	15.68	-8.48	0.29	0.48	29.69	16.79	-0.69
-0.37	10.84	-4.02	0.14	7.08	0.36	15.30	2.66
0.04	11.54	0.44	0.00	2.71	1.72	11.70	-1.64
-0.19	15.75	-2.96	0.04	14.91	30.41	13.69	-3.86
-0.11	12.42	-1.35	0.01	1.53	4.80	12.99	-1.24
-0.67	18.30	-12.35	0.46	4.53	65.08	17.97	-2.13
0.44	5.96	2.62	0.19	0.16	18.30	8.16	0.40
0.07	8.60	0.56	0.00	1.12	2.68	11.46	1.06
-0.37	9.11	-3.39	0.14	19.38	1.26	15.32	4.40
0.06	11.88	0.73	0.00	4.80	2.73	11.50	-2.19
0.15	12.12	1.79	0.02	10.18	3.56	10.73	-3.19
0.04	9.89	0.39	0.00	0.00	0.12	11.69	0.00
0.09	8.37	0.72	0.01	1.23	3.46	11.28	1.11

Dataset with ‘by-hand’ calculation of model parameters.



Model fit to data, predict porosity from density.



Model predictions compared to truth values.

-8.80	27.90
0.66	1.21
0.63	1.92
177.26	103
653.20	379.56

→

b1: slope of fit	-8.8
se1: standard error of slope	0.7
r2: proportion var. explained	0.6
Fstat: for test of all coefficients	177.3
ssreg: explained variance	653.2

b0: intercept of fit	27.9
seb: standard error of the intercept	1.2
sey: standard error for the estimate	1.92
d.f.: degrees of freedom	103
ssresid: unexplained variance	379.6

Excel linear regression outputs applied for model checking, confidence and prediction intervals.

Linear regression demonstration in Excel, file is Linear_Regression_Demo_v2.xlsx

Bivariate Statistics
Example Regression Demonstration in Python

Walk through of a linear regression method in Python.

Including:

- model confidence intervals, prediction intervals and hypothesis testing
- Variance explained and correlation coefficient
- Model quality control (QC) diagnostics

Linear Regression in Python for Engineers, Data Scientists and Geoscientists

Michael Pyrcz, Associate Professor, University of Texas at Austin

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

This is a tutorial / demonstration of **Linear Regression**. In *Python*, the *SciPy* package, specifically the *Stats* functions (https://docs.scipy.org/doc/scipy/reference/stats.html) provide excellent tools for efficient use of statistics.
I have previously provided this example in R and posted it on GitHub:

1. R https://github.com/GeostatsGuy/geostatsr/blob/master/linear_regression_demo_v2.R
2. Rmd with docs https://github.com/GeostatsGuy/geostatsr/blob/master/linear_regression_demo_v2.Rmd
3. knit as an HTML document(https://github.com/GeostatsGuy/geostatsr/blob/master/linear_regression_demo_v2.html)

In all cases, I use the same dataset available as a comma delimited file (https://git.io/fxMql).

This tutorial includes basic, calculation of a linear regression model (only 1 predictor and 1 response), testing the significance of the parameters, calculation the parameter confidence intervals and the conditional prediction interval.

Caveats

I have not included all the details, specifically the test assumptions in this document. These are included in the accompanying course notes, Lec09_Bivariate_QQ_Regres.pdf.

Project Goal

0. Introduction to Python in Jupyter including setting a working directory, loading data into a Pandas DataFrame.
1. Learn the basics for working with linear regresion in Python.
2. Demonstrate the efficiency of using Python and SciPy package for statistical analysis.

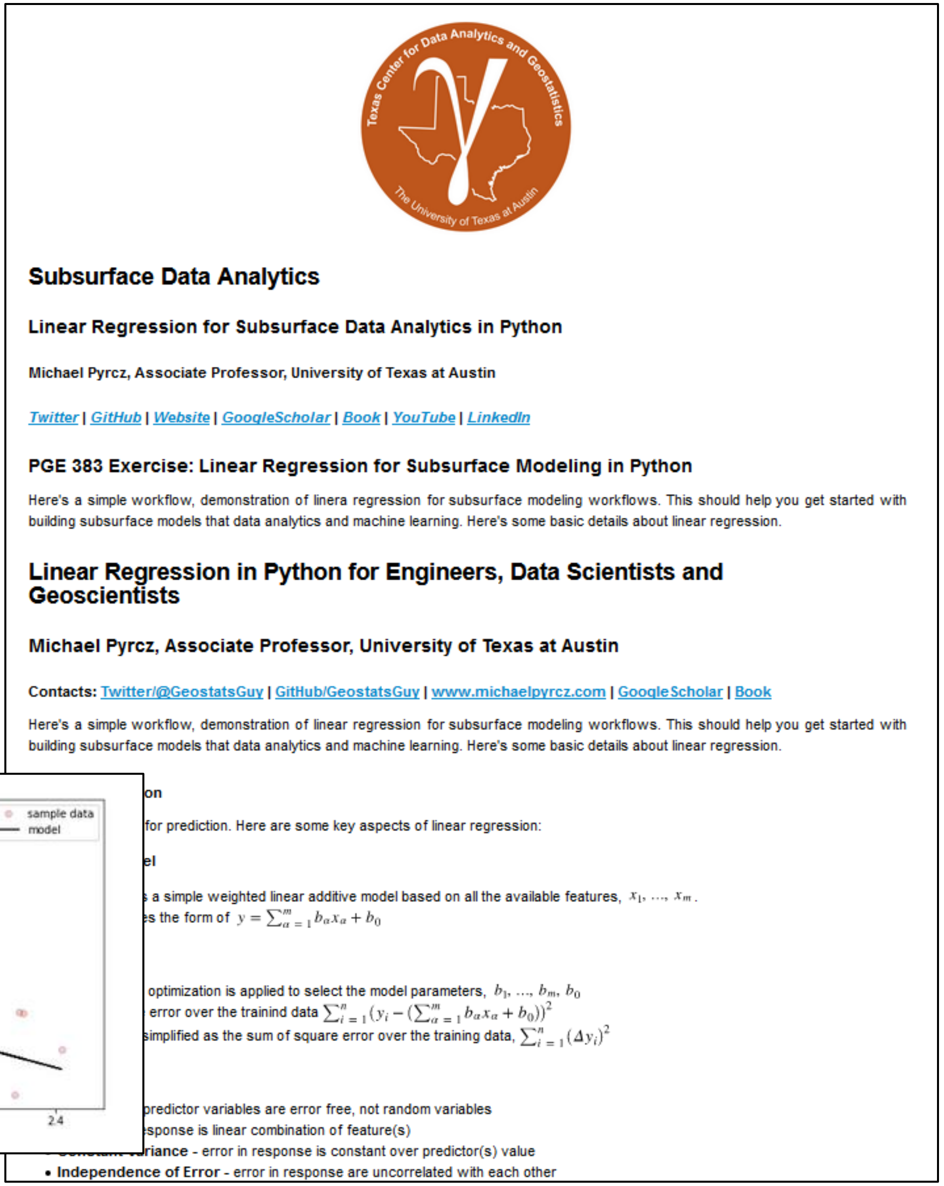
Linear regression in Python, the workflow file is PythonDataBasics_LinearRegression.ipynb.

Bivariate Statistics

Example Regression Demonstration in Python

- Another walk-through of a linear regression method in Python from my machine learning course.

- File is: SubsurfaceDataAnalytics_linear_regression.ipynb.



Subsurface Data Analytics

Linear Regression 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: Linear Regression for Subsurface Modeling in Python

Here's a simple workflow, demonstration of linera regression for subsurface modeling workflows. This should help you get started with building subsurface models that data analytics and machine learning. Here's some basic details about linear regression.

Linear Regression in Python for Engineers, Data Scientists and Geoscientists

Michael Pyrcz, Associate Professor, University of Texas at Austin

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

Here's a simple workflow, demonstration of linear regression for subsurface modeling workflows. This should help you get started with building subsurface models that data analytics and machine learning. Here's some basic details about linear regression.

on

for prediction. Here are some key aspects of linear regression:

el

a simple weighted linear additive model based on all the available features, $x_1, \ldots, x_m$.

s the form of $y = \sum_{\alpha=1}^{m} b_\alpha x_\alpha + b_0$

optimization is applied to select the model parameters, $b_1, \ldots, b_m, b_0$

error over the trainind data $\sum_{i=1}^{n} (y_i - (\sum_{\alpha=1}^{m} b_\alpha x_\alpha + b_0))^2$

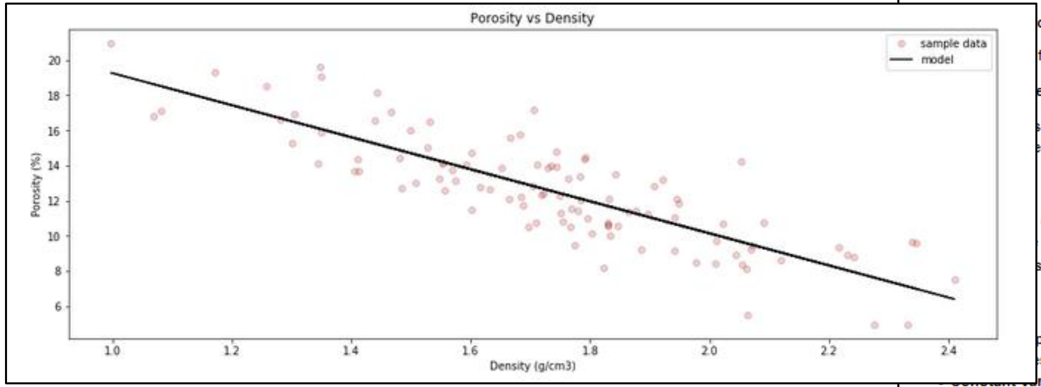
simplified as the sum of square error over the training data, $\sum_{i=1}^{n} (\Delta y_i)^2$

predictor variables are error free, not random variables

sponse is linear combination of feature(s)

riance - error in response is constant over predictor(s) value

- **Independence of Error** - error in response are uncorrelated with each other



PGE 338 Data Analytics and Geostatistics

Lecture 9: Bivariate Modeling

Lecture outline . . .

- Quantile-Quantile (Q-Q) Plots
- Regression Analysis

Introduction
General Concepts
Univariate
Bivariate
Correlation
Regression
Model Checking
Time Series Analysis
Spatial Analysis
Machine Learning
Uncertainty Analysis

Michael Pyrcz, The University of Texas at Austin