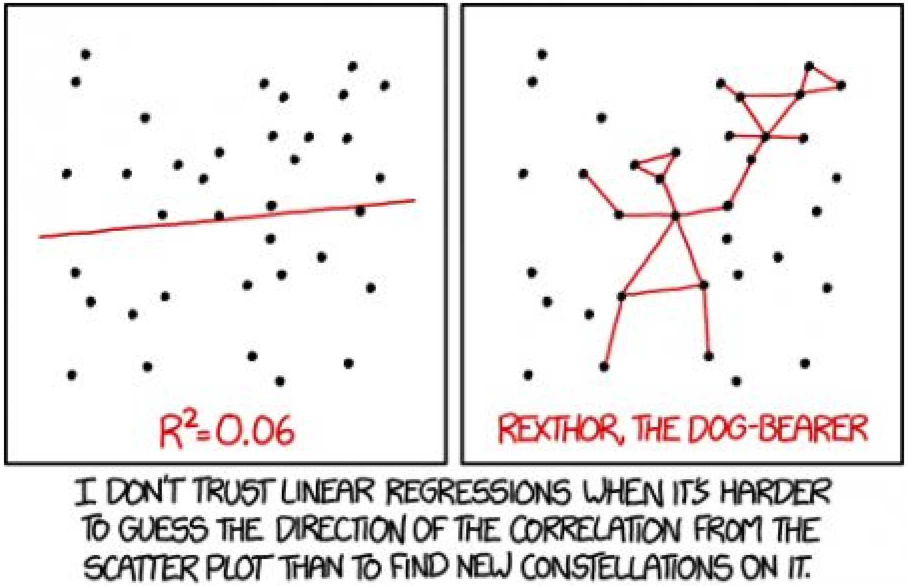
Polynomial and Basis Function Regression

October 12th, 2016

Introduction to Regression

Relation between a dependent variable, *y*, and set of independent variables, *x*, that describe the expectation value of *y* given *x*, or $E[y|x]$.

Given a multidimensional data set drawn from some pdf and the full error covariance matrix for each data point, we attempt to infer the conditional expectation value.



$$\boldsymbol{y} = \boldsymbol{f}\,(\boldsymbol{x}|\boldsymbol{\theta})$$

Matrix Notation for Regression

We define regression in terms of a design matrix, *M*, such that:

$$Y = M\theta$$

Where Y is an N-dimensional vector of values y_i,

$$Y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ . \\ y_{N-1} \end{bmatrix}$$

Θ is a p-dimensional vector of regression coefficients,

$$\Theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ . \\ \theta_{P-1} \end{bmatrix}$$

And M is therefore a $P \times M$ matrix.

$$M = \begin{bmatrix} 1 & x_0 & x_0^2 & . & x_0^P \\ 1 & x_1 & x_1^2 & . & x_1^P \\ . & . & . & . & . \\ 1 & x_N & x_N^2 & . & x_N^P \end{bmatrix}$$

(Some) Types of Linear Regression

Simple Linear Regression

One independent variable, *x*.

Fits a line through the set of *k* points such that the sum of the squared residuals is minimized.

$y_i = \theta_0 + \theta_1 x_i + \epsilon_i$

θ_0 and θ_1 are coefficients that describe the regressive function, and ϵ_i is the additive noise term.

Multivariable Regression

Fitting a hyperplane, as opposed to a straight line. Extend the description of the regression function to multiple dimensions with $y = f(x|\theta)$.

$y_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + \ldots + \theta_k x_{ik} + \epsilon_i$

θ_i as the regression parameter and x_{ik} as the *k*th component of the *i*th data entry within a multivariate data set.

Polynomial Regression

In general, we can express $f(x|\theta)$ as the sum of arbitrary (often nonlinear) functions as long as the model is linear in terms of the regression parameters, θ.

$y_i = \theta_0 + \theta_1 x_i + \theta_2 x_i^2 + \theta_3 x_i^3 + \ldots$

Quick Interjection # 1

Multivariate Linear Regression != Multivariable Linear Regression

Statistically speaking, **multivariate** analysis refers to statistical models that have 2 or more dependent (or outcome) variables and **multivariable** analysis refers to statistical models in which there are multiple independent (or response) variables.

Multivariate versus Multivariable Regression

Multivariable Regression

The form would be:

$y_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + \ldots + \theta_k x_{ik} + \epsilon_i$

Where y is a continuous dependent variable, x is a single predictor in the regression model, and x_1, x_2, ..., x_k are the predictors in the multivariable model.

Multivariate Regression

The form would be:

$Y_{N\times P} = X_{N\times(k+1)}\,\theta_{(k+1)\times p} + \epsilon$

Where the relationship between multiple dependent variables, y, and a single set of independent variables, x, are assessed.

Polynomial Regression

Quick Interjection #2
Non-Linear Regression

A statistical technique that describes nonlinear relationships in experimental data; regression models are generally assumed to be parametric and described as a nonlinear equation.

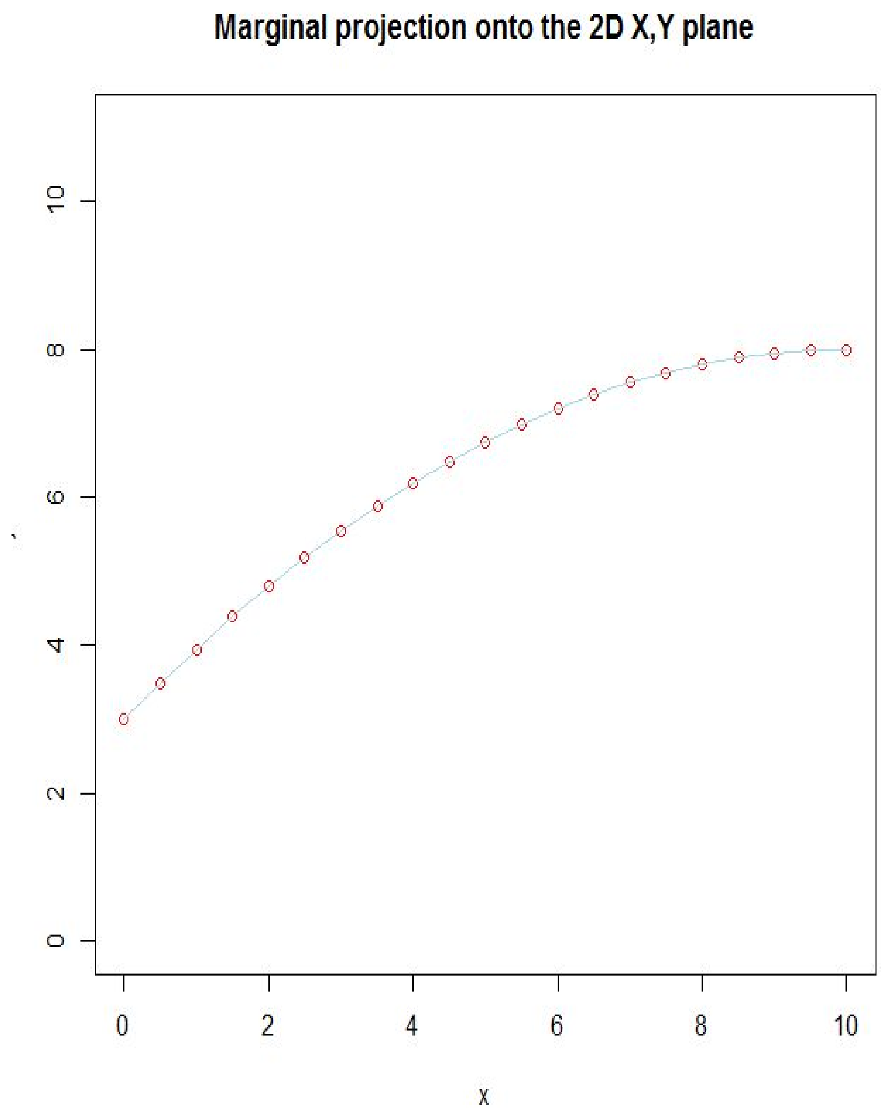
Why is Polynomial Regression not a Non-Linear Function?

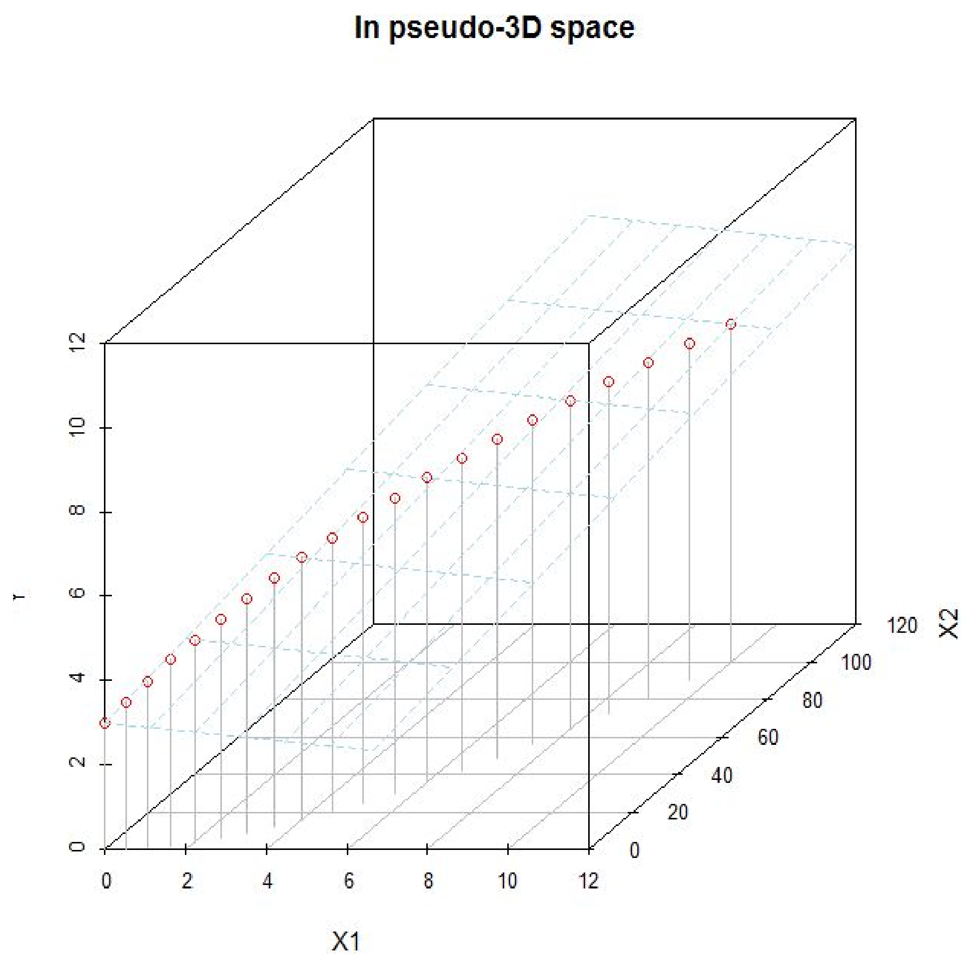
The independent parameters *(x)* to be estimated are multilinear terms, not the θ coefficient, which qualifies polynomial as a linear regression.

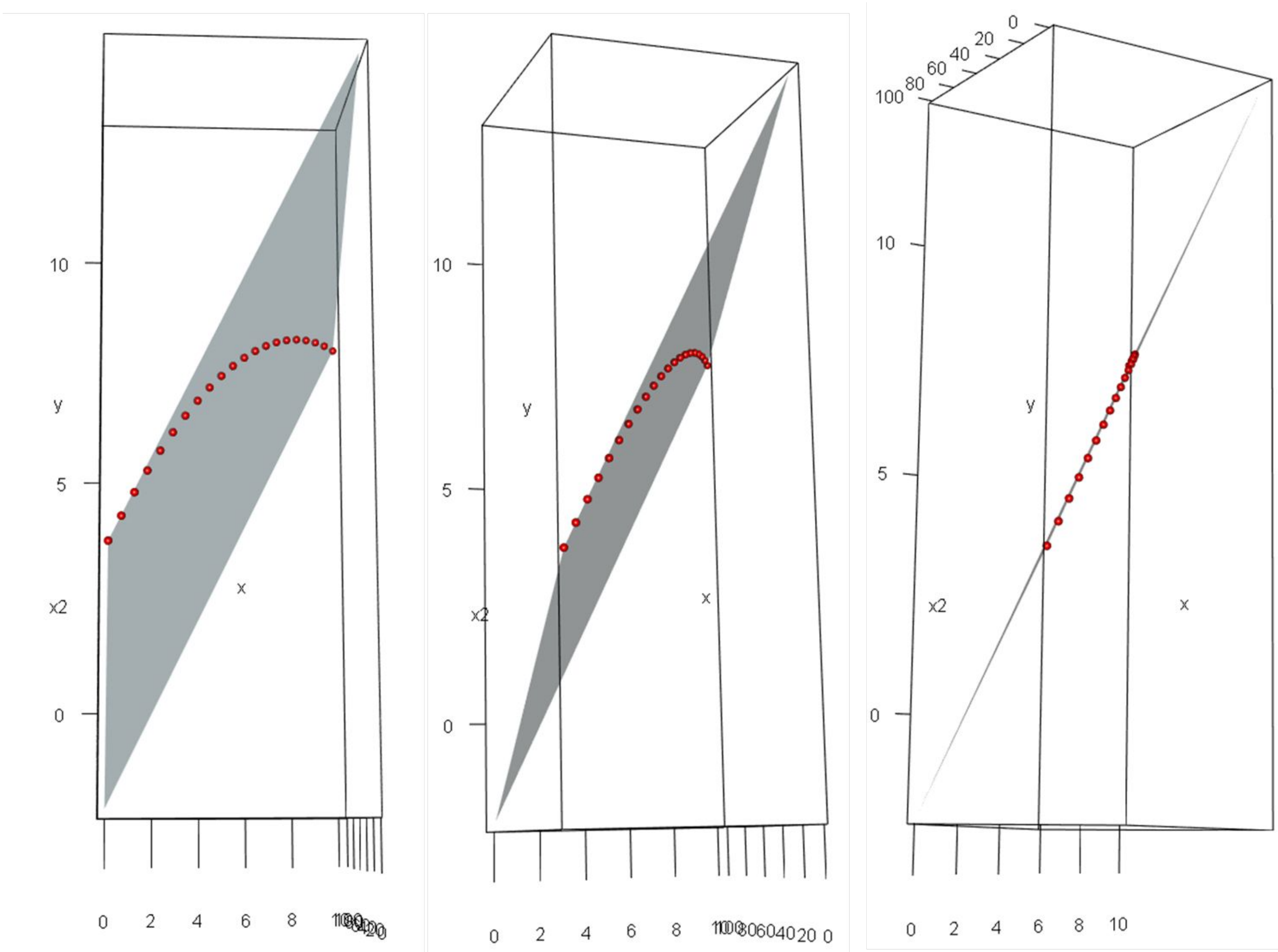
When we fit a regression model

$$y_i = \theta_0 + \theta_1 x_i + \theta_2 x_i^2 + \theta_3 x_i^3 + \ldots$$

We tend not to 'think' of x_i^2 as the square of x_i , but rather as a separate variable.


Marginal projection onto the 2D X,Y plane
0
2
4
6
8
10
X


In pseudo-3D space
0
2
4
6
8
10
12
X1
20
40
60
80
100
120
X2



10
y
5
x2
x
0
0 2 4 6 8
10
y
5
x2
x
0
0 2 4 6 8
100 80 60 40 20 0
0
20
40
60
80
100
10
y
5
x2
x
0
0 2 4 6 8 10

Programing Polynomial Regression

PolynomialRegression function in AstroML

```
import numpy as np
from astroML.linear_model import PolynomialRegression
#Create 100 random points in two dimensions for X
X = np.random.random((100,2))
Y = X[:,0]**2 + X[:,1]**3
#Model fits a 3rd degree Polynomial
model = PolynomialRegression(3)
model.fit(X,Y)
Y_pred = model.predict(X)
```

Polynomial Regression Fun Facts

Number of parameters in the model we are fitting is given by:

$$m = \frac{(p+k)!}{p!\,k!}$$

Where we are given a data set with *k* dimensions to which we fit a *p*-dimensional polynomial.

Number of degrees of fitting for the regression model is:

$$v = N - m$$

The probability of the model is given by a X^2 distribution with *v* degrees of freedom.

Basis Function Representation

$$M = \begin{bmatrix} 1 & x_0 & x_0^2 & . & x_0^P \\ 1 & x_1 & x_1^2 & . & x_1^P \\ . & . & . & . & . \\ 1 & x_N & x_N^2 & . & x_N^P \end{bmatrix}$$

Basis Function Model: An Example

We begin with:

$$y(x,\theta) = \theta_0 + \sum_{i=0}^{P-1} \theta_i \, \psi_i(x)$$

Where $\psi_i(x)$ are called Basis functions that we have selected to allow for a non-linear function of *x.*

Citations and Credits

Ivezic et. al.

Chapter 8: Regression and Model Fitting

Matlab

Nonlinear Regression

StackExchange

Image 2 & 3

AM J Public Health

Multivariate or Multivariable Regression?

XKCD Comic Strip

Image 1

Latex Formatting: Overleaf