

Polynomial and Basis Function Regression

October 12th, 2016

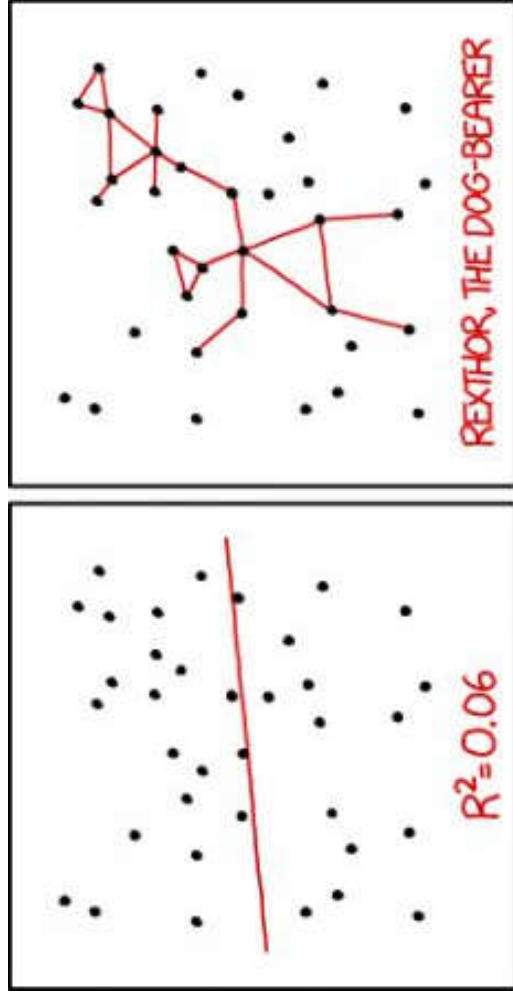


Introduction to Regression

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.

$$y = f(x|\theta)$$



I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER TO GUESS THE DIRECTION OF THE CORRELATION FROM THE SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.

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 ,

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

θ is a p -dimensional vector of regression coefficients,

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_{P-1} \end{bmatrix}$$

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

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

(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 $y = f(x | \theta)$.

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

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 .

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

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:

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

Where y is a continuous dependent variable, x is a single predictor in the regression model, and x_1, x_2, \dots, 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?

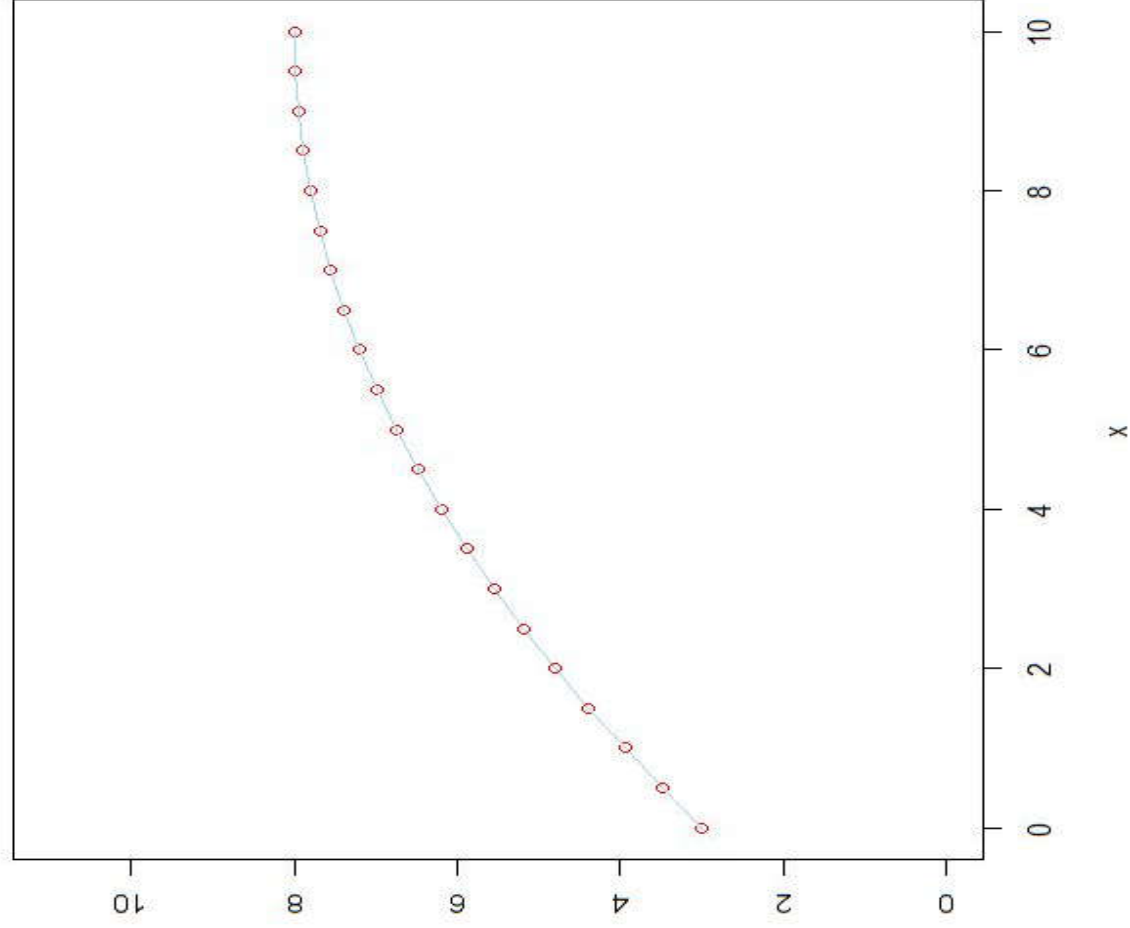
The independent parameters (x) to be estimated are multilinear terms, not the θ coefficient, which qualifies polynomial as a linear regression.

When we fit a regression model

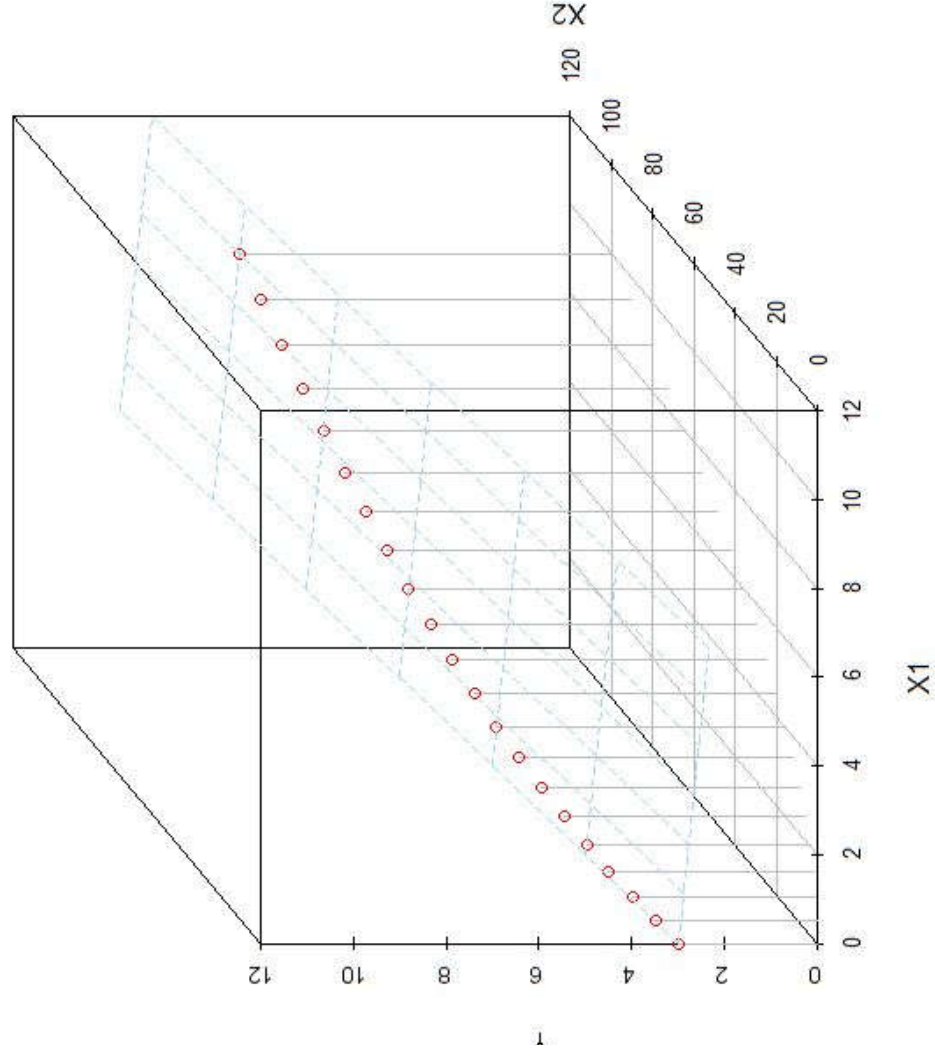
$$y_i = \theta_0 + \theta_1 x_i + \theta_2 x_i^2 + \theta_3 x_i^3 + \dots$$

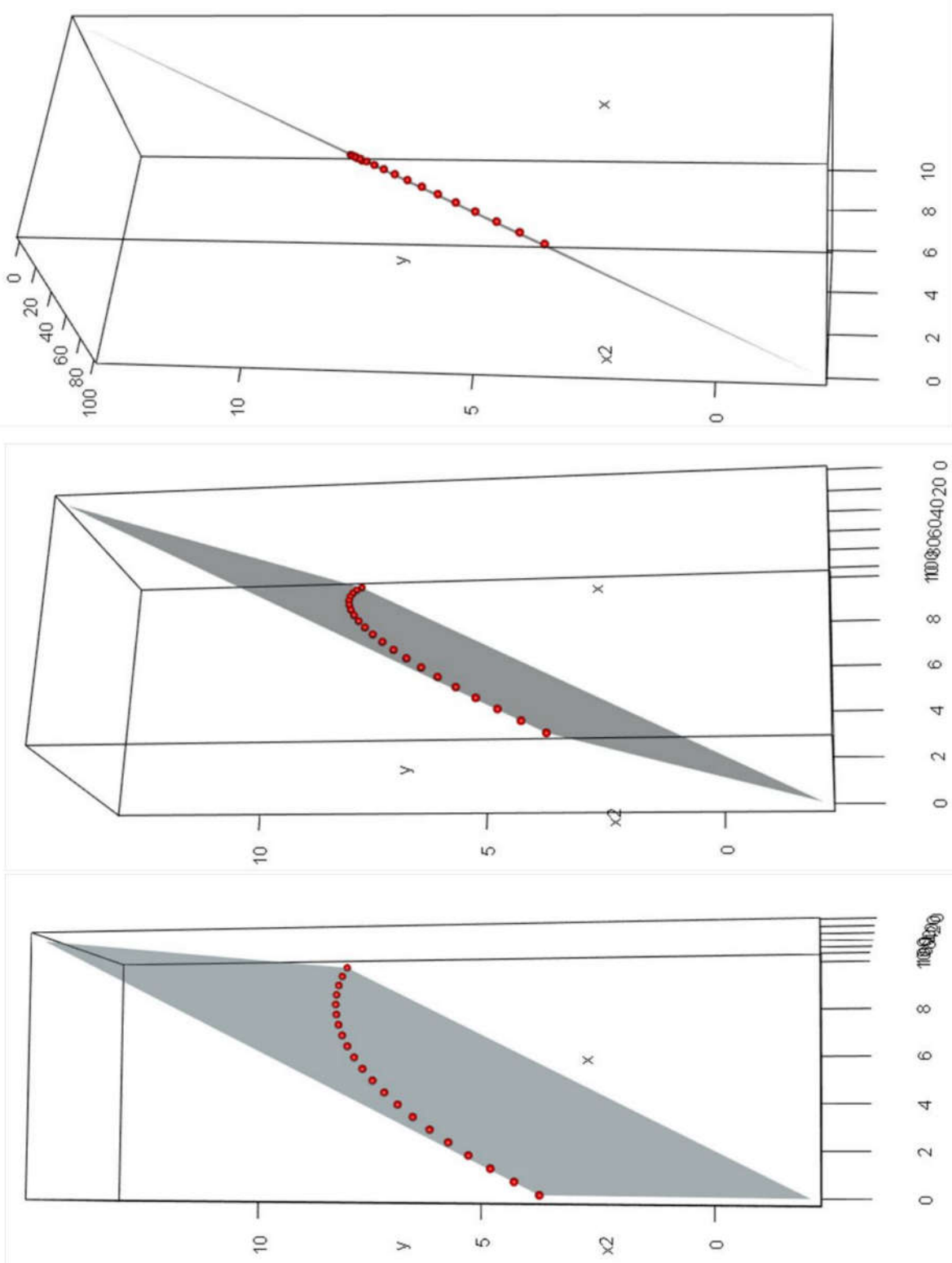
We tend not to 'think' of x_i^2 as the square of x_i , but rather as a separate variable.

Marginal projection onto the 2D X,Y plane



In pseudo-3D space





Programming 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:

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

Number of degrees of fitting for the regression model is:

$$v = N - m$$

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

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

Basis Function Representation

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

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