ECE 595: Machine Learning I
Tutorial 03: Linear Regression Examples

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

PURDUE
UNIVERSITY

Outline

- Illustration in 1D
- Generalized linear regression
- Interpreting regression coefficients
- Representation via regression coefficients

Reference:

- Gilbert Strang, Linear Algebra and Its Applications, 5th Edition.
- Carl Meyer, Matrix Analysis and Applied Linear Algebra, SIAM, 2000.
- `http://cs229.stanford.edu/section/cs229-linalg.pdf`
- `https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf`

Line Fitting

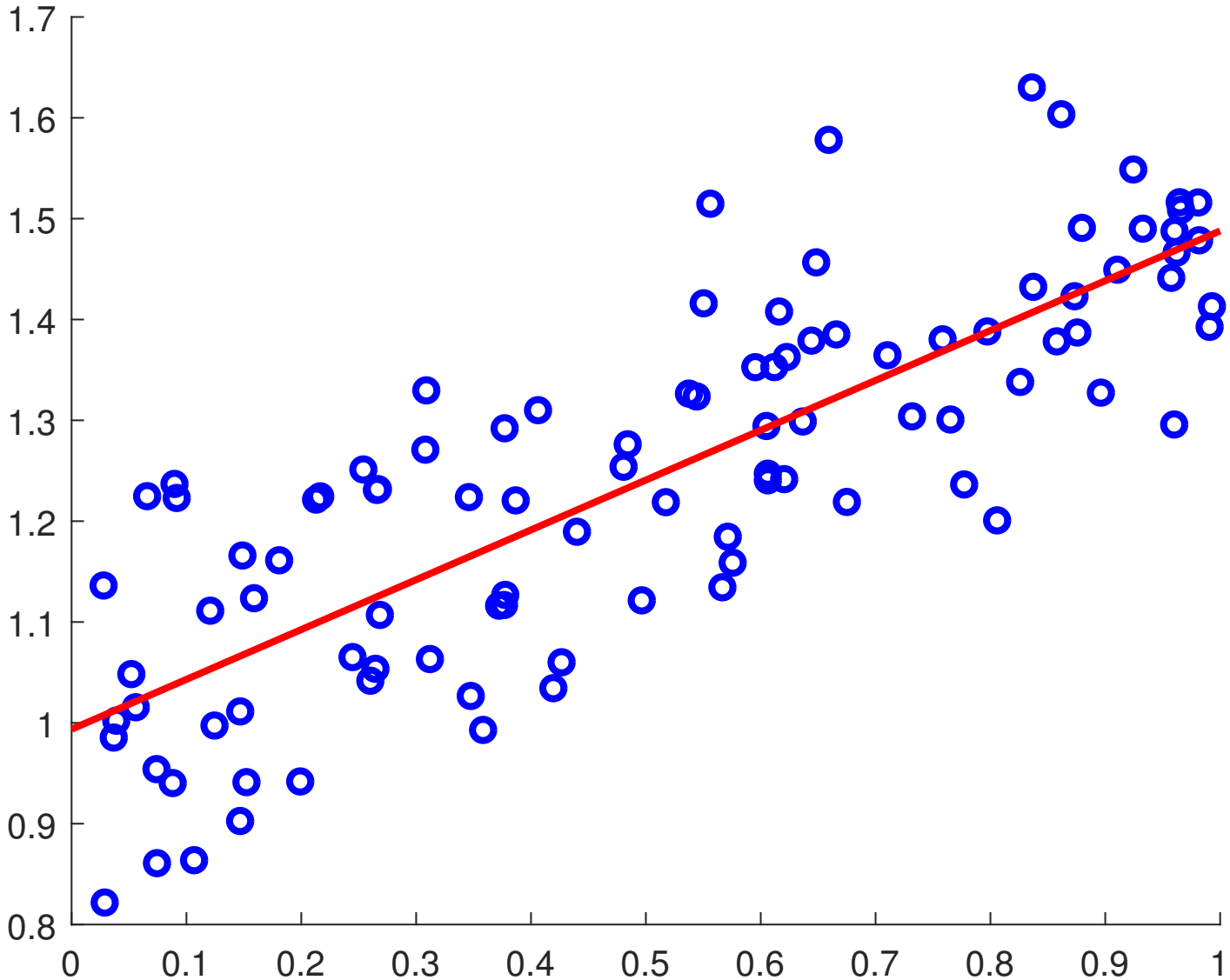


Illustration in 1D

- Consider fitting 1D data $(x^1, y^1), \ldots, (x^N, y^N)$
- The parameter is $\boldsymbol{\theta} = [\theta_1, \theta_0]^T$
- The model is

$$g_{\boldsymbol{\theta}}(x) = \theta_1 x + \theta_0.$$

- This can be written as

$$\underbrace{\begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}}_{\boldsymbol{y}} \approx \underbrace{\begin{bmatrix} x^1 & 1 \\ x^2 & 1 \\ \vdots & \vdots \\ x^N & 1 \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \theta_1 \\ \theta_0 \end{bmatrix}}_{\boldsymbol{\theta}}$$

- The problem now translates to solve for (θ_1, θ_0) from this system of linear equations.

Illustration in 1D

The loss function is

$$J(\boldsymbol{\theta}) = \sum_{n=1}^{N} (y^n - (\theta_1 x^n + \theta_0))^2.$$

Taking derivatives on both sides with respect to θ_1 and θ_0 yields

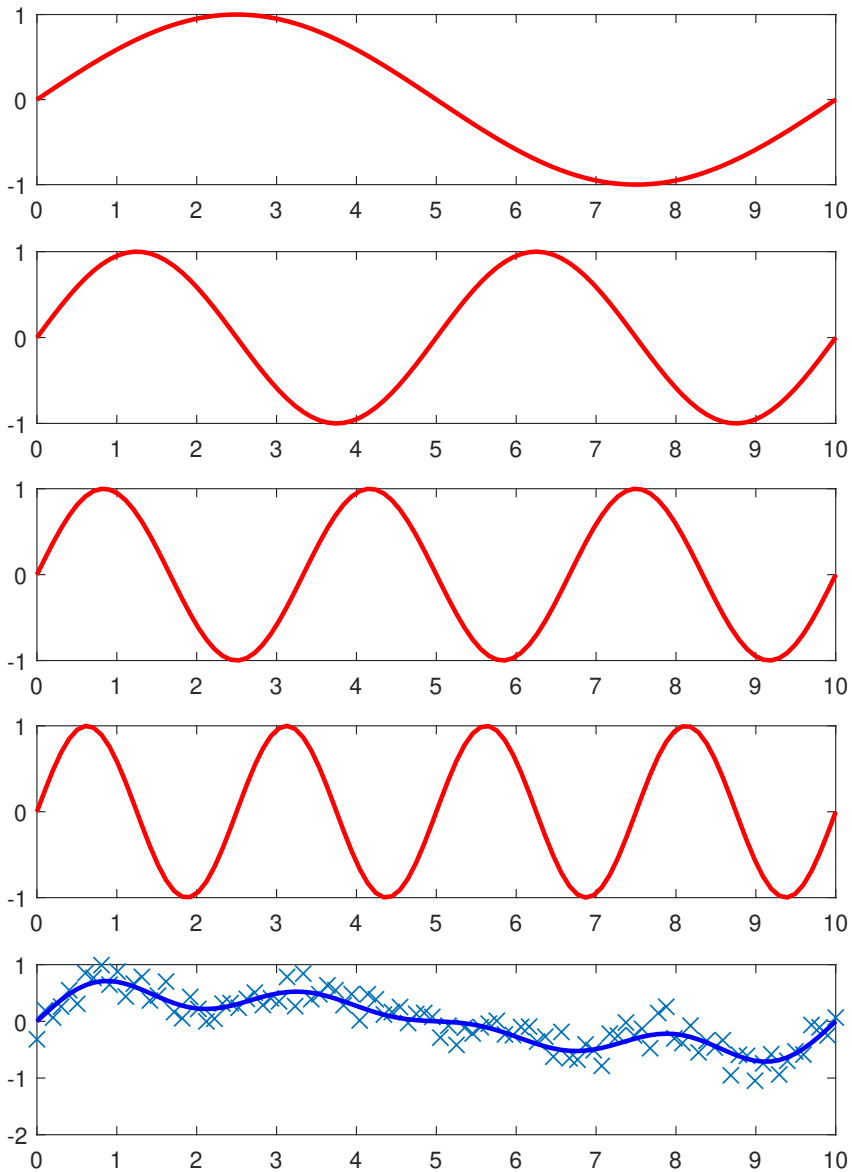
$$\frac{\partial}{\partial \theta_1} J(\boldsymbol{\theta}) = 2\left(\sum_{n=1}^{N} x^n y^n - \theta_1 \sum_{n=1}^{N} (x^n)^2 - \theta_0 \sum_{n=1}^{N} x^n\right) = 0$$

$$\frac{\partial}{\partial \theta_0} J(\boldsymbol{\theta}) = 2\left(\sum_{n=1}^{N} y^n - \theta_1 \sum_{n=1}^{N} x^n - N\theta_0\right) = 0$$

Rearranging the terms, this is equivalent to $\boldsymbol{A}^T \boldsymbol{A}\boldsymbol{\theta} = \boldsymbol{A}^T \boldsymbol{y}$:

$$\begin{bmatrix} \sum_{n=1}^{N} (x^n)^2 & \sum_{n=1}^{N} x^n \\ \sum_{n=1}^{N} x^n & N \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_0 \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} x^n y^n \\ \sum_{n=1}^{N} y^n \end{bmatrix}$$

Generalized Linear Regression



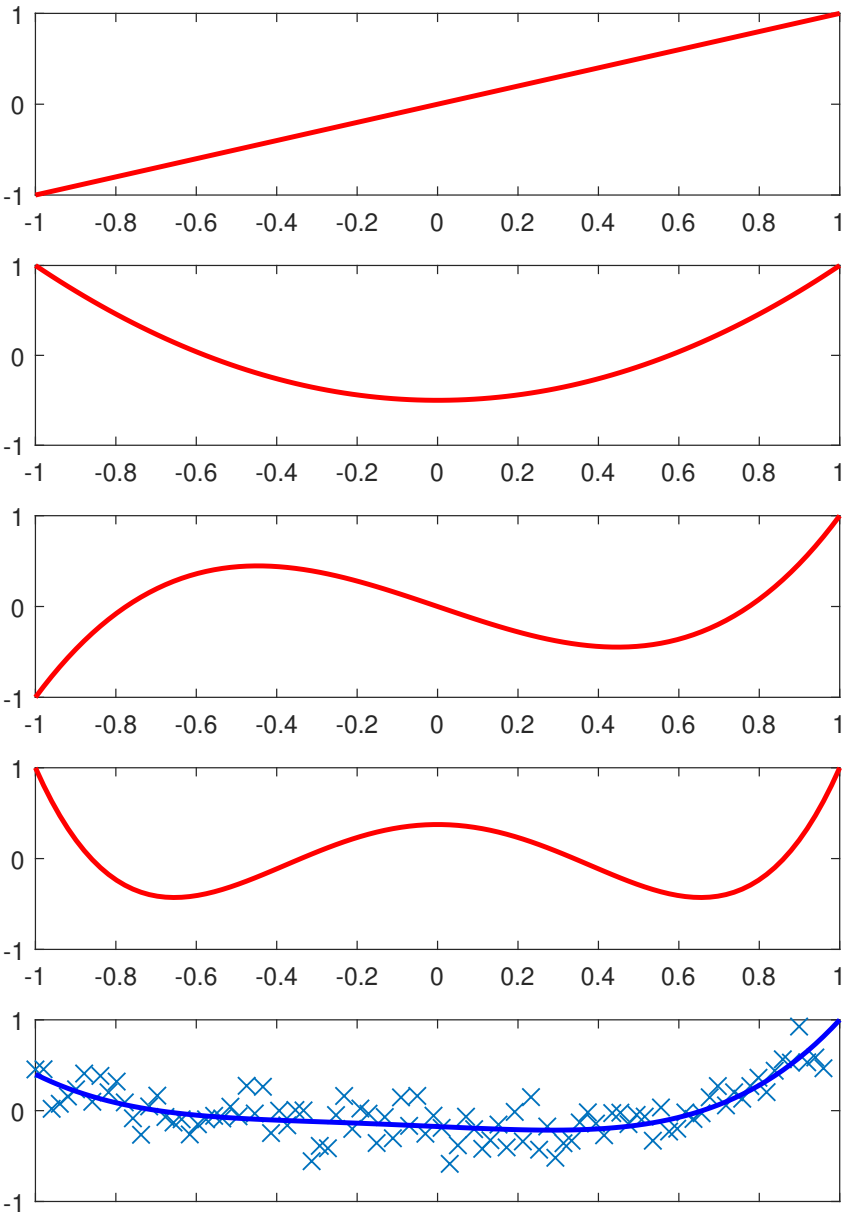
- Eg 1: Fourier series

$$\boldsymbol{x}^n = \begin{bmatrix} x_1^n \\ x_2^n \\ \vdots \\ x_d^n \end{bmatrix} = \begin{bmatrix} \sin(\omega_0 t_n) \\ \sin(2\omega_0 t_n) \\ \vdots \\ \sin(K\omega_0 t_n) \end{bmatrix}$$

$$y^n = \boldsymbol{\theta}^T \boldsymbol{x}^n = \sum_{k=1}^{d} \theta_k \sin(k\omega_0 t_n)$$

- θ_k: k-th Fourier coefficient
- $\sin(k\omega_0 t_n)$: k-th Fourier basis at time t_n

Generalized Linear Regression



- Eg 2: Legendre Polynomial
- "Orthogonalized" polynomials

$$\boldsymbol{x}^n = \begin{bmatrix} x_1^n \\ x_2^n \\ \vdots \\ x_K^n \end{bmatrix} = \begin{bmatrix} P_1(t_n) \\ P_2(t_n) \\ \vdots \\ P_K(t_n) \end{bmatrix}$$

$$y^n = \boldsymbol{\theta}^T \boldsymbol{x}^n = \sum_{k=1}^{d} \theta_k P_k(t_n)$$

- θ_k: k-th polynomial coefficient
- $P_k(t_n)$: k-th Legendre polynomial at time t_n

Interpreting Results

city	funding	hs	not-hs	college	college4	crime rate
1	40	74	11	31	20	478
2	32	72	11	43	18	494
3	57	70	18	16	16	643
4	31	71	11	25	19	341
5	67	72	9	29	24	773
⋮	⋮	⋮	⋮	⋮		
50	66	67	26	18	16	940

https://web.stanford.edu/~hastie/StatLearnSparsity/data.html

$$\underbrace{\begin{bmatrix} \text{crime rate}^1 \\ \text{crime rate}^2 \\ \vdots \\ \text{crime rate}^N \end{bmatrix}}_{\boldsymbol{y}} \approx \underbrace{\begin{bmatrix} 1 & \text{funding}^1 & \text{hs}^1 & \dots & \text{college4}^1 \\ 1 & \text{funding}^2 & \text{hs}^2 & \dots & \text{college4}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \text{funding}^N & \text{hs}^N & \dots & \text{college4}^N \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix}}_{\boldsymbol{\theta}}$$

Bias

Let us look at the first column:

$$\underbrace{\begin{bmatrix} \text{crime rate}^1 \\ \text{crime rate}^2 \\ \vdots \\ \text{crime rate}^N \end{bmatrix}}_{\boldsymbol{y}} \approx \underbrace{\begin{bmatrix} 1 & \text{funding}^1 & \text{hs}^1 & \dots & \text{college4}^1 \\ 1 & \text{funding}^2 & \text{hs}^2 & \dots & \text{college4}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \text{funding}^N & \text{hs}^N & \dots & \text{college4}^N \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix}}_{\boldsymbol{\theta}}$$

- In the above equation, we have an all-one vector and a parameter θ_0.
- This column is called the **bias** term.
- Think of an y-axis off-set which brings the line up and down, but not the slope.
- Without the bias term, you force the line to start from the origin which is not always desirable.

Feature Vector

Consider one of the columns in the system

$$\underbrace{\begin{bmatrix} \text{crime rate}^1 \\ \text{crime rate}^2 \\ \vdots \\ \text{crime rate}^N \end{bmatrix}}_{\boldsymbol{y}} \approx \underbrace{\begin{bmatrix} 1 & \text{funding}^1 & \text{hs}^1 & \dots & \text{college4}^1 \\ 1 & \text{funding}^2 & \text{hs}^2 & \dots & \text{college4}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \text{funding}^N & \text{hs}^N & \dots & \text{college4}^N \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix}}_{\boldsymbol{\theta}}$$

- The column vector $\boldsymbol{a}_j$ is called a feature. The corresponding coefficient θ_j indicates the contribution of $\boldsymbol{a}_j$.
- You can view the above system as

$$\boldsymbol{y} = \sum_{j=0}^{d} \theta_j \boldsymbol{a}_j$$

which expresses the measured data $\boldsymbol{y}$ as a linear combination of the feature vectors.

Interpreting Results

Run regression analysis.[1] Here is the result:

- $\theta_1 = 10.9934$: police funding
- $\theta_2 = 1.1451$: high school
- $\theta_3 = 10.1812$: no high school
- $\theta_4 = 2.7386$: college
- $\theta_5 = -0.7781$: college at least 4 years

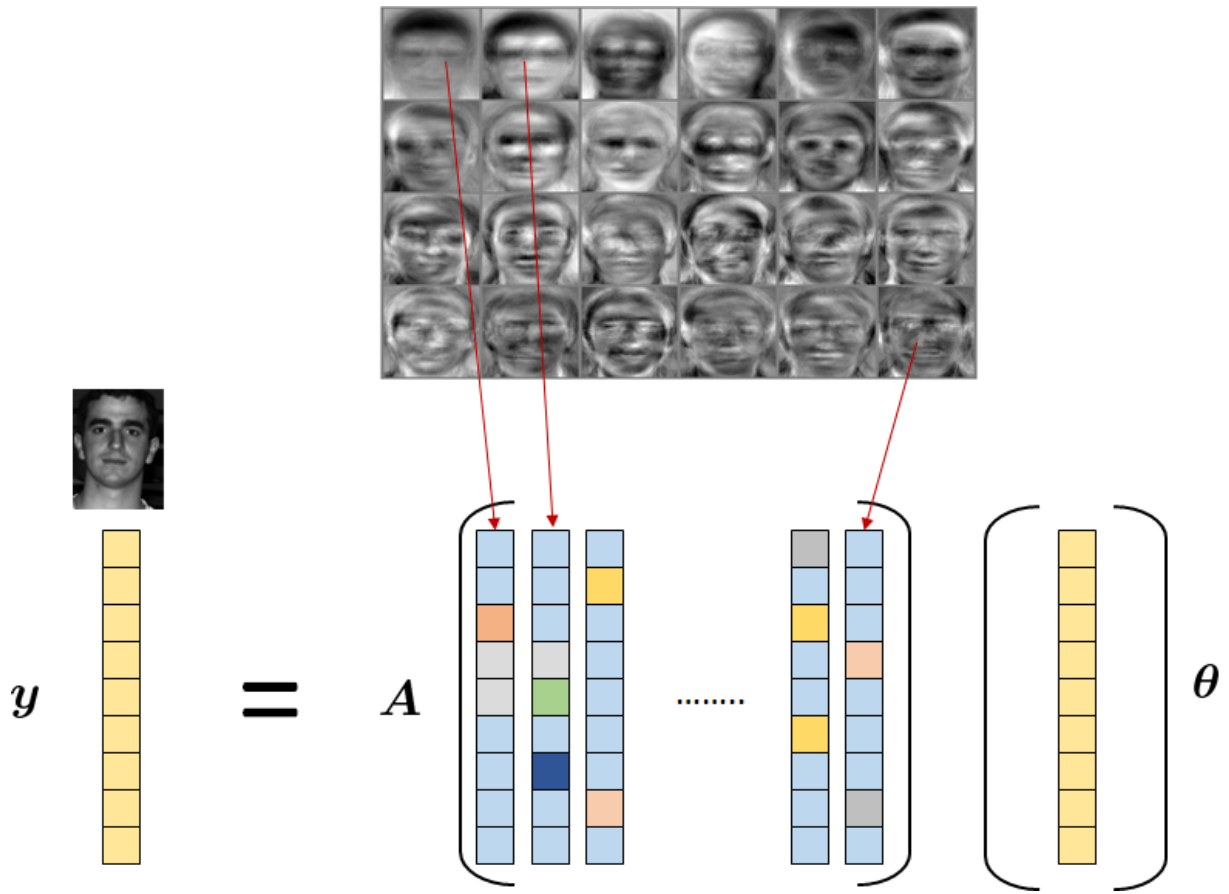
One possible way to interpret the results:

- Apparently, what matters is the amount of police funding and the number of residents without no high school.
- But this is not justified, because the columns are not normalized.
- To quantitatively justify these claims, we need to run statistical analysis, e.g., confidence intervals, hypothesis tests.

[1] For this dataset, we need to add a regularization term so that $\widehat{\boldsymbol{\theta}} = (\boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{I})^{-1}\boldsymbol{A}^T\boldsymbol{y}$. Here we set $\lambda = 1000$.

Representation

- Linear regression can be used to identify influential representations.
- For example, given the features of faces, we can determine which feature is more prominent for the query image.



Orthogonal $\boldsymbol{A}$

- Linear regression requires us solving the system of linear equations:

$$\widehat{\theta} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{y}.$$

- For representation problems, one can hand-craft the representation matrix $\boldsymbol{A}$.
- If $\boldsymbol{A}$ is **orthogonal**, i.e., $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}$, then $\widehat{\theta}$ is simplified to

$$\widehat{\theta} = \cancel{(\boldsymbol{A}^T\boldsymbol{A})^{-1}}\boldsymbol{A}^T\boldsymbol{y}.$$

- Examples of orthogonal $\boldsymbol{A}$:
  - Fourier matrix
  - Wavelet matrix
  - Features extracted by Principal Component Analysis (PCA)
  - Matrices with i.i.d. Gaussian entries. Then $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}$ with high probability.

Fitting and Representation

This tutorial illustrates two perspectives of linear regression:

- **Fitting Data**: (Data Science)
  - Given measurements, find a line to fit the data.
  - $\boldsymbol{A}$ is the data matrix storing the features (or attributes).
  - $\boldsymbol{y}$ is the vector storing the responses.
  - Useful for predicting values and analyzing contributions.
  - E.g.: Medical data, census data, stock market, etc.
- **Representation**: (Signal Processing, Computer Vision)
  - Given pre-defined features, find a representation.
  - $\boldsymbol{A}$ is a set of given features which can be trained or hand-crafted.
  - $\boldsymbol{y}$ is the query data.
  - Useful for dimension reduction. Decision making based on coefficients instead of $\boldsymbol{y}$.
  - E.g.: Image classification, signal processing, etc.