Regression: Linear regression

GU 4241

Statistical Machine Learning

Xiaofei Shi

Columbia University

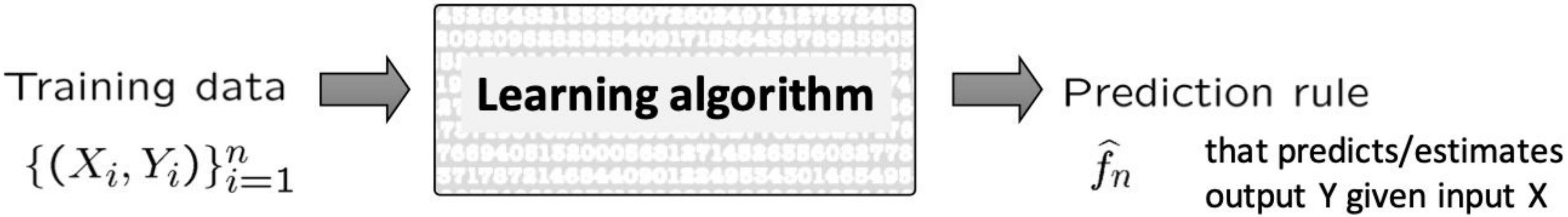
Tasks

Input → Regressor → Predict real number

Input → Classifier → Predict category

Input → Density Estimator → Probability

Regression:



Training data
$\{(X_i, Y_i)\}_{i=1}^n$
Learning algorithm
Prediction rule
$\widehat{f}_n$
that predicts/estimates
output Y given input X

Regression:

- In everyday life we need to make decisions by taking into account lots of factors
- The question is what weight we put on each of these factors (how important are they with respect to the others)
- Suppose your task is to help build a evaluation system for food delivery apps

	熊猫外卖 HungryPanda	Uber Eats	chowbus
Available restaurants	30	10	20
Average delivery time	Next day	>3hr	1hr
Mandatory service fee	>10%	>20%	>13%
Score	9	7	8

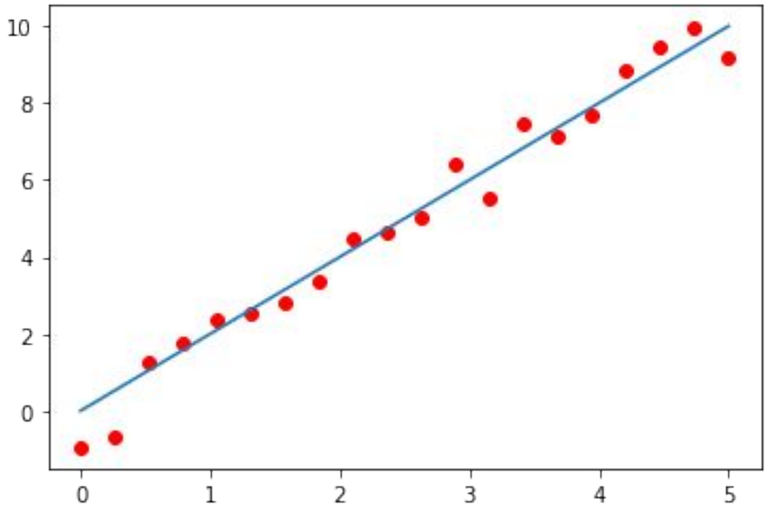
Columbia University

Regression:

- Given an input x, we want to compute an output y
- For example:

  - predict Google's stock price using the current price of Bitcoin

  - predict arrival time using the traffic condition



Linear Regression:

- Given an input x, we want to compute an output y
- In linear regression we assume that y and x are related with the following

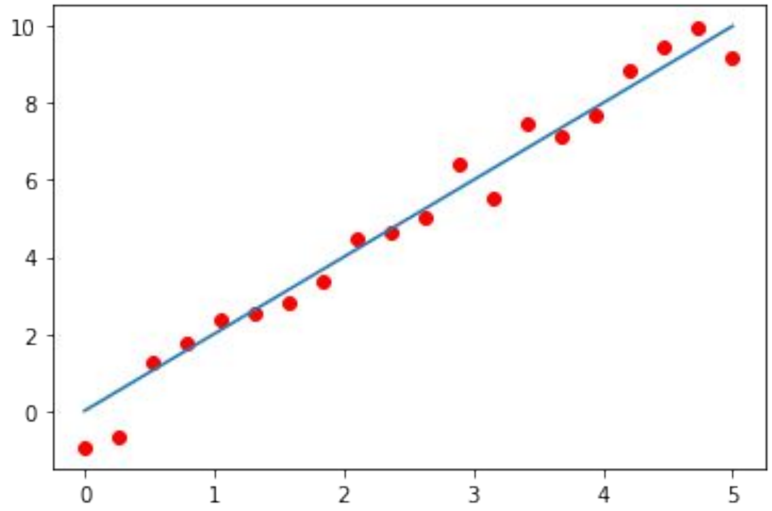
What we are trying to predict

Observed values

$$y = wx + \varepsilon$$

parameter we want to determine

noise term



Road map

- Build your model:

  1) relationship: $y = wx + \varepsilon$

  2) preference: choose w to minimize $\arg\min_w \sum_i (y_i - wx_i)^2$

- Estimate your model parameters:

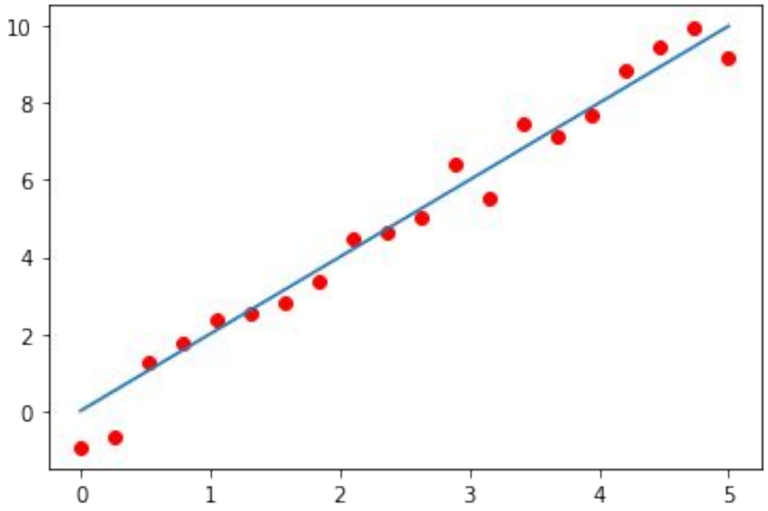
  1) pluging in observed data to express your preference

  2) get parameters estimation for your model

- Understand your model:

COLUMBIA UNIVERSITY

Linear Regression:



- Our goal is to estimate w from a training data of <xi,yi> pairs
- One easy way to determine w is to minimize the least squares error:

$$\arg\min_w \sum_i (y_i - wx_i)^2$$

- Why least squares?
  - easy to compute
  - has a nice probabilistic interpretation
- Several other approaches

If the noise is Gaussian with mean 0 then least squares is also the MLE of w

COLUMBIA UNIVERSITY

Solving linear regression using minimization

- Goal function:
- 3-step:
  - take derivative wrt parameter w
  - set it to 0
  - solve optimal w from the equation

$$\frac{\partial}{\partial w}\sum_i (y_i - wx_i)^2 = 2\sum_i -x_i(y_i - wx_i) \Rightarrow$$

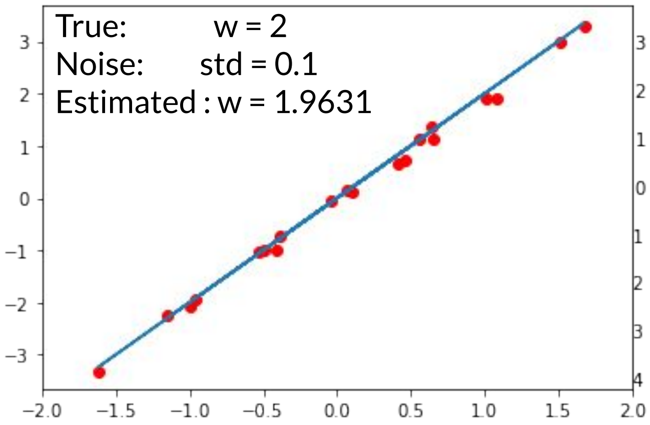
$$2\sum_i x_i(y_i - wx_i) = 0 \Rightarrow$$

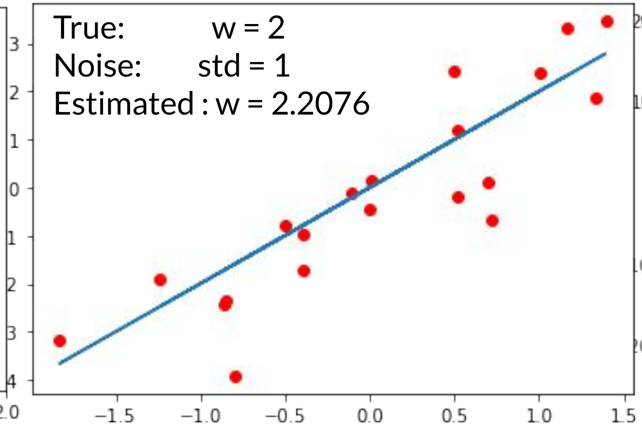
$$\sum_i x_i y_i = \sum_i w x_i^2 \Rightarrow$$

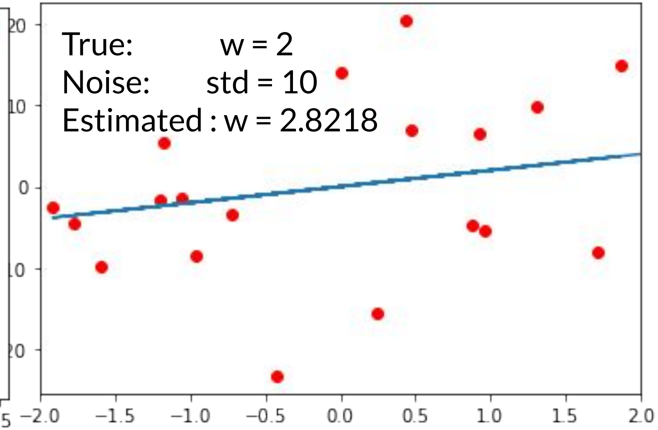
$$w = \frac{\sum_i x_i y_i}{\sum_i x_i^2}$$

Columbia University

Regression Example







Adding in intercept

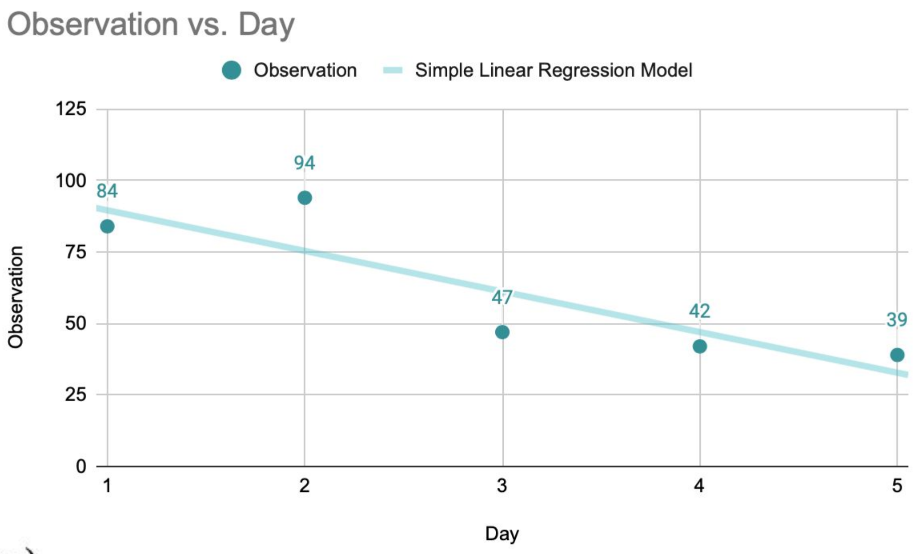
- So far we assume the regression line passes through the origin
- What if the line does not?

$$y = w_0 + w_1 x + \varepsilon$$

Adding in intercept!

- We can determine w = (w0 , w1) explicitly

$$w_0 = \frac{\sum_i y_i - w_1 x_i}{n} \qquad w_1 = \frac{\sum_i x_i (y_i - w_0)}{\sum_i x_i^2}$$



Columbia University

Multivariate Regression:

- What if we want to input more information?
- For example:

  - predict Google's stock price using the current price of Bitcoin might not be enough, we might want to also consider the stock price of Amazon, Apple, and other index

  - predict arrival time using the traffic condition, the weather condition, and the vehicle's condition to make it more accurate
- This becomes a multivariate linear regression problem:

$$y = w_0 + w_1 x_1 + \ldots + w_k x_k + \varepsilon$$

Google's price Bitcoin S&P 500

Columbia University

Multivariate Regression:

Not all functions can be represented using the input values directly!
How to capture nonlinearity?

- What if we want to input more information?
- For example:

  - predict Google's stock price using the current price of Bitcoin might not be enough, we might want to also consider the stock price of Amazon, Apple, and other index

  - predict arrival time using the traffic condition, the weather condition, and the vehicle's condition to make it more accurate
- This becomes a multivariate linear regression problem:

$$y = w_0 + w_1x_1 + \dots + w_kx_k + \varepsilon$$

Google's price

Bitcoin

S&P 500

Columbia University

How to capture nonlinearity?

- In some cases we would like to use polynomial or other terms based on the input data

  - Polynomial: $\phi_j(x) = x^j$ for j=0 … n

  - Gaussian: $\phi_j(x) = \frac{(x-\mu_j)}{2\sigma_j^2}$

  - Sigmoid: $\phi_j(x) = \frac{1}{1+\exp(-s_j x)}$

- Are these still linear regression problems?

COLUMBIA UNIVERSITY

How to capture nonlinearity?

- In some cases we would like to use polynomial or other terms based on the input data

  - Polynomial: $\phi_j(x) = x^j$ for j=0 … n

  - Gaussian: $\phi_j(x) = \dfrac{(x-\mu_j)}{2\sigma_j^2}$

  - Sigmoid: $\phi_j(x) = \dfrac{1}{1+\exp(-s_j x)}$

- Are these still linear regression problems?

As long as the coefficients are linear the equation is still a linear regression problem!

Columbia University

Nonlinear basis functions

- In some cases we would like to use polynomial or other terms based on the input data

  - Polynomial: $\phi_j(x) = x^j$ for j=0 … n

  - Gaussian: $\phi_j(x) = \dfrac{(x-\mu_j)}{2\sigma_j^2}$

  - Sigmoid: $\phi_j(x) = \dfrac{1}{1+\exp(-s_j x)}$

Any function of the input values can be used. The solution for the parameters of the regression remains the same.

- Linear regression can be applied in the same way to functions of these values
- As long as these functions can be directly computed from the observed values the parameters are still linear in the data and the problem remains a linear regression problem

Nonlinear basis functions

- We use the new notation for the basis functions, linear regression can be written as

$$y = \sum_{j=0}^{n} w_j \phi_j(x)$$

- Nothing changed! Once again we can use ‘least squares’ to find the optimal solution to figure out parameter w

COLUMBIA UNIVERSITY

General linear regression problem

Our goal is to minimize the following loss function:

$$J(\mathrm{w}) = \sum_i \Big(y^i - \sum_j w_j \phi_j(x^i)\Big)^2$$

$$y = \sum_{j=0}^{k} w_j \phi_j(x)$$

w – vector of dimension k+1

$\phi(x^i)$ – vector of dimension k+1

y^i – a scaler

Moving to vector notations we get:

$$J(\mathrm{w}) = \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2 = 2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get $2 \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathrm{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

Columbia University

General linear regression problem

$$J(\mathrm{w}) = \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2$$

We take the derivative w.r.t **w**

$$\frac{\partial}{\partial w} \sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i))^2 = 2\sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}}$$

Equating to 0 we get

$$2\sum_i (y^i - \mathrm{w}^{\mathrm{T}} \phi(x^i)) \phi(x^i)^{\mathrm{T}} = 0 \Rightarrow$$

$$\sum_i y^i \phi(x^i)^{\mathrm{T}} = \mathrm{w}^{\mathrm{T}} \left[\sum_i \phi(x^i) \phi(x^i)^{\mathrm{T}} \right]$$

Define:

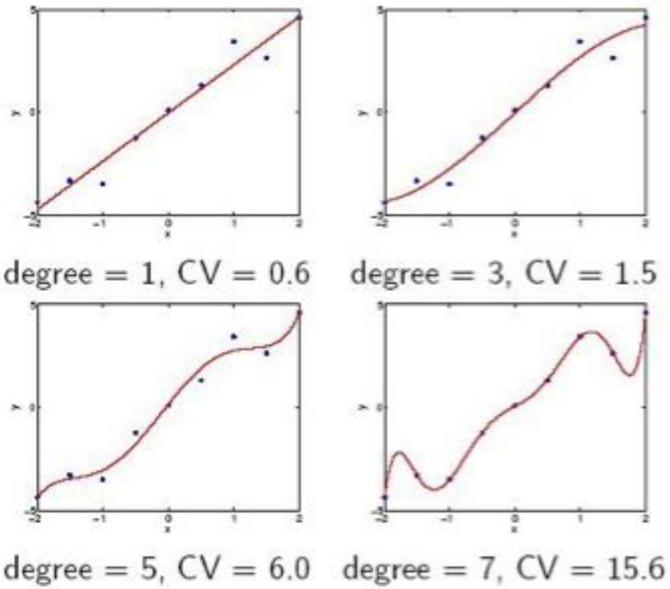
$$\Phi = \begin{pmatrix} \phi_0(x^1) & \phi_1(x^1) & \cdots & \phi_k(x^1) \\ \phi_0(x^2) & \phi_1(x^2) & \cdots & \phi_k(x^2) \\ \vdots & \vdots & \cdots & \vdots \\ \phi_0(x^n) & \phi_1(x^n) & \cdots & \phi_k(x^n) \end{pmatrix}$$

Then deriving w
we get:

$$\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$$

Columbia University

Example: polynomial regression



Thoughts？

References

- Christopher Bishop: Pattern Recognition and Machine Learning, Chapter 3
- Kutner, Nachtsheim and Neter: Applied Linear Regression Models.
- Agresti: Foundations of Linear and Generalized Linear Models.
- Ziv Bar-Joseph, Tom Mitchell, Pradeep Ravikumar and Aarti Singh: CMU 10-701