

UVA CS 4501 - 001 / 6501 – 007

Introduction to Machine Learning and Data Mining

Lecture 7: Regression Models - Review

Yanjun Qi / Jane

University of Virginia
Department of
Computer Science

HW1 DUE NOW / HW2 OUT TODAY

First Survey – Google Form

- **Is the course content too difficult ?**
- **Is the teaching pace too slow ?**
- **What do you like about the lectures so far ?**
- **What do you unlike about the lectures so far ?**

Second Survey – Google Form

- Why are you taking this course?
- What would you like to gain from this course?
- What topics are you most interested in learning about from this course?
- Any other suggestions?
- **Is homework-1 helpful ?**
- **Is homework-1 too challenging ?**

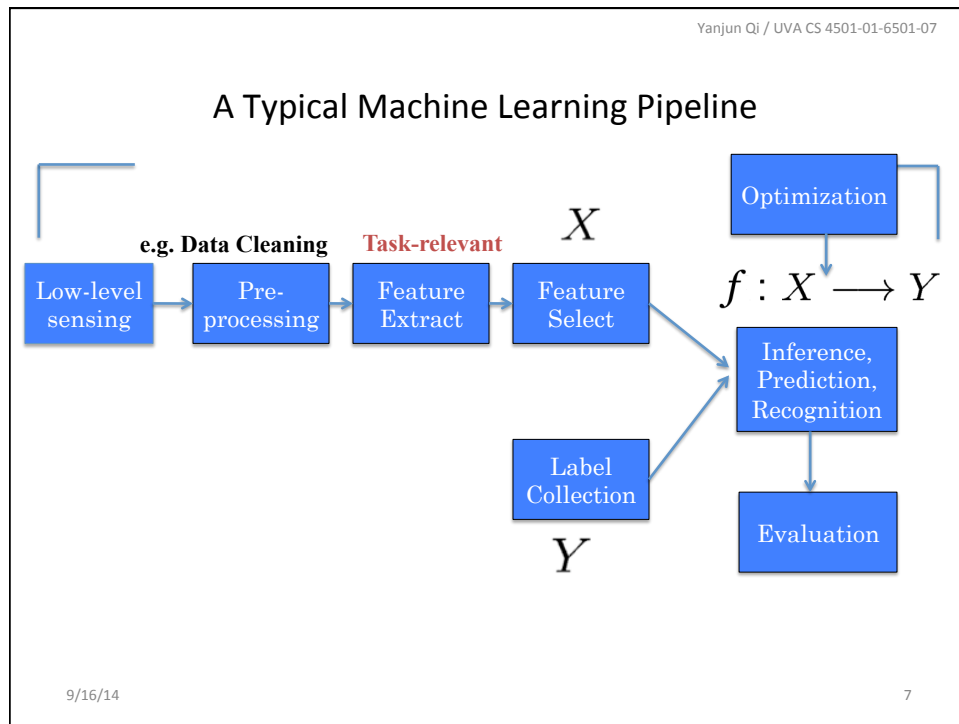
Five major sections of this course

- ~~Regression (supervised)~~
- Classification (supervised)
- Unsupervised models
- Learning theory
- Graphical models

Today

- Review of basic pipeline
- Interesting Applications (text / image / audio)

- Review of regression models
 - Linear regression (LR)
 - LR with non-linear basis functions
 - Locally weighted LR
 - LR with Regularizations



Yanjun Qi / UVA CS 4501-01-6501-07

e.g. SUPERVISED LEARNING

$f : X \longrightarrow Y$

- Find function to map **input** space X to **output** space Y
- **Generalisation**: learn function / hypothesis from **past data** in order to “explain”, “predict”, “model” or “control” **new** data examples

KEY

9/16/14 8

Yanjun Qi / UVA CS 4501-01-6501-07

| | X ₁ | X ₂ | X ₃ | Y |
|----------------|----------------|----------------|----------------|---|
| S ₁ | | | | |
| S ₂ | | | | |
| S ₃ | | | | |
| S ₄ | | | | |
| S ₅ | | | | |
| S ₆ | | | | |

A Dataset

$$f : X \rightarrow Y$$

- **Data/points/instances/examples/samples/records:** [rows]
- **Features/attributes/dimensions/independent variables/covariates/predictors/regressors:** [columns, except the last]
- **Target/outcome/response/label/dependent variable:** special column to be predicted [last column]

9/16/149

Yanjun Qi / UVA CS 4501-01-6501-07

SUPERVISED LEARNING

target/class

↓

| | | |
|--|--|---|
| | | A |
| | | B |
| | | B |
| | | A |
| | | A |
| | | B |

learn

→

model

f

Training dataset consists of **input-output** pairs

test dataset

| | | |
|--|--|---|
| | | ? |
| | | ? |
| | | ? |
| | | ? |
| | | ? |
| | | ? |

apply model

→

| | | |
|--|--|---|
| | | B |
| | | B |
| | | B |
| | | A |
| | | A |

$f(x_?)$

→

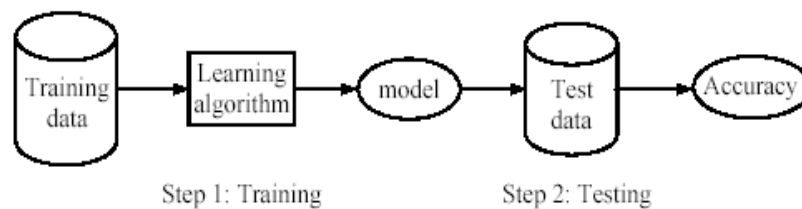
Evaluation

Measure Loss on pair $\rightarrow (f(x_?), y_?)$

9/16/1410

Evaluation Choice-I:

- ✓ **Training (Learning):** Learn a model using the training data
- ✓ **Testing:** Test the model using **unseen test data** to assess the model accuracy



$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}}$$

9/16/14

Evaluation Choice-II: e.g. 10 fold Cross Validation

- Divide data into 10 equal pieces
- 9 pieces as training set, the rest 1 as test set
- Collect the scores from the diagonal

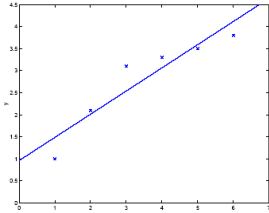
| model | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | train | train | train | train | train | train | train | train | train | test |
| 2 | train | train | train | train | train | train | train | train | test | train |
| 3 | train | train | train | train | train | train | train | test | train | train |
| 4 | train | train | train | train | train | train | test | train | train | train |
| 5 | train | train | train | train | train | test | train | train | train | train |
| 6 | train | train | train | train | test | train | train | train | train | train |
| 7 | train | train | train | test | train | train | train | train | train | train |
| 8 | train | train | test | train | train | train | train | train | train | train |
| 9 | train | test | train | train | train | train | train | train | train | train |
| 10 | test | train | train | train | train | train | train | train | train | train |

9/16/14

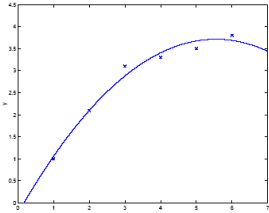
Yanjun Qi / UVA CS 4501-01-6501-07

Which function f to choose?

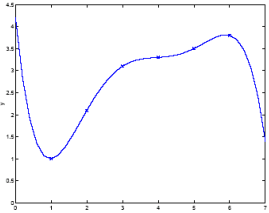
Many possible choices , e.g.



$y = \theta_0 + \theta_1 x$



$y = \theta_0 + \theta_1 x + \theta_2 x^2$

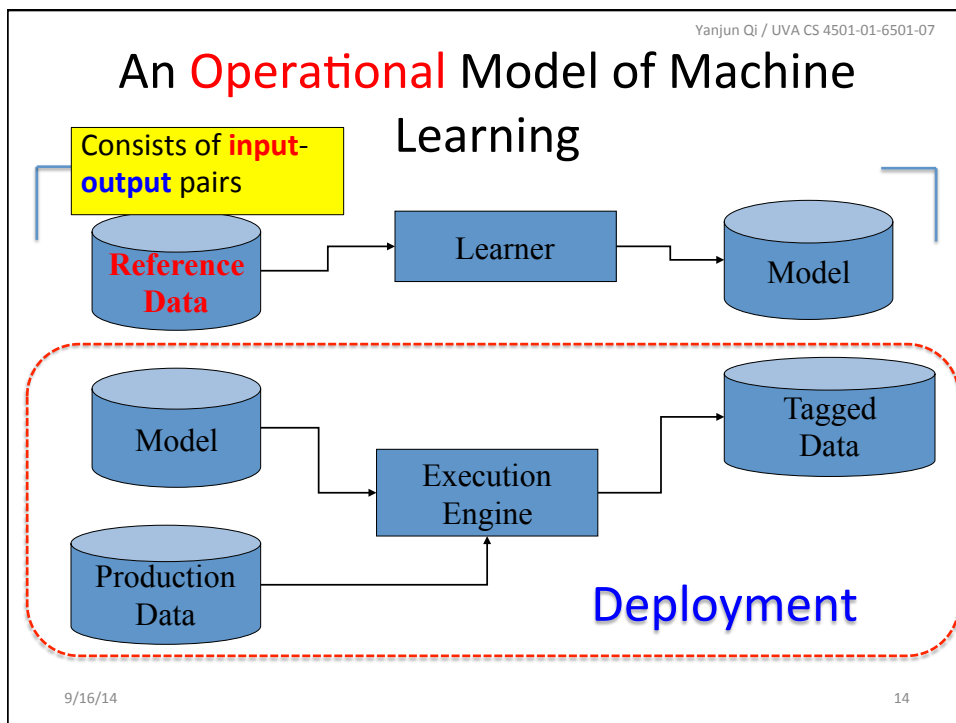


$y = \sum_{j=0}^5 \theta_j x^j$

Generalisation: learn function / hypothesis from **past data** in order to “explain”, “predict”, “model” or “control” **new data** examples

Choose f that generalizes well !

9/16/14 13



Today

- ❑ Review of basic pipeline
- ❑ Interesting Applications
 - ❑ Text
 - ❑ Image
 - ❑ Audio
- ❑ Review of regression models
 - Linear regression (LR)
 - LR with non-linear basis functions
 - Locally weighted LR
 - LR with Regularizations

(1) Example Applications – Regression

- ▶ Y output is a continuous valued variable
- ▶ Greatly studied in statistics, neural network fields.

- ▶ Examples:
 - ▶ Predicting sales amounts of new product based on advertising expenditure.
 - ▶ Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
 - ▶ Time series prediction of stock market indices.

(1). E.g. A Practical Application of Regression Model

Yanjun Qi / UVA CS 4501-01-6501-07

Movie Reviews and Revenues: An Experiment in Text Regression*

Mahesh Joshi Dipanjan Das Kevin Gimpel Noah A. Smith

Language Technologies Institute

Carnegie Mellon University

Pittsburgh, PA 15213, USA

{maheshj,dipanjan,kgimpel,nasmith}@cs.cmu.edu

Abstract

We consider the problem of predicting a movie's opening weekend revenue. Previous work on this problem has used metadata about a movie—e.g., its genre, MPAA rating, and cast—with very limited work making use of text *about* the movie. In this paper, we use the text of film critics' reviews from several sources to predict opening weekend revenue. We describe a new dataset pairing movie reviews with metadata and revenue data, and show that review text can substitute for metadata, and even improve over it, for prediction.

9/16/14

17

(2) Example Application – Text Documents, e.g. Google News

Yanjun Qi / UVA CS 4501-01-6501-07

The screenshot shows the Google News homepage. At the top, there is a search bar with the Google logo on the left and two buttons: "Search News" and "Search the Web". Below the search bar, it says "Search and browse 4,500 news sources updated continuously." On the left side, there is a navigation menu with categories like "Top Stories", "News near you", "World", "U.S.", "Business", "Technology", "iPhone", "Microsoft Windows", "Minecraft", "Safety", "IBM", "General Motors", "Facebook", "Microsoft Corporation", "Tablet computers", "Tor", "Entertainment", "Sports", "Science", "Health", and "Spotlight". The main content area features a "Technology" section with a featured article titled "Microsoft Keyboard Works With Windows, iOS, and Android" from PC Magazine, 53 minutes ago. Below this, there are smaller articles: "Microsoft announces new line of accessories for Windows, Android, iOS, and ..." from BetaNews, "Trending on Google+: Microsoft's Universal Bluetooth Keyboard Will Work With Windows, Android, And ..." from Android Police, "Opinion: Microsoft's New Universal Mobile Keyboard Has Android and iOS in Mind" from Gizmodo, "Microsoft/Minecraft Deal Gets a Skit On Conan O'Brien's Show" from GameSpot, "Apple's iOS 8 available Wednesday" from New York Daily News, and "IBM Watson Data Analysis Service Revealed".

9/16/14

18

(2) Example Application – Different Ways for Text Categorization

- Human labor (people assign categories to every incoming article)
- Hand-crafted rules for automatic classification
 - If article contains: stock, Dow, share, Nasdaq, etc. → Business
 - If article contains: set, breakpoint, player, Federer, etc. → Tennis
- Machine learning algorithms

9/16/14

19

(2) Example Application – Text Document Representation

“Shortly after **Phish** wraps up their four-night run in Miami this December, **Page** will begin a short tour up the East Coast with **Vida Blue** **Page**, **Russell** and **Oteil** will be joined by the six-member **Spam Allstars**, who back **Vida Blue**...”

- Count content bearing words in document
- Create vector representation for each text document
- For each word, using counts as feature values to represent magnitude of dimension

Bag of 'words'

| | |
|---------|-----|
| Phish | 14 |
| Page | 3 |
| Russell | 2 |
| Trey | 6 |
| Record | 2 |
| CD | 3 |
| begin | 1 |
| short | 1 |
| tour | 3 |
| ... | ... |

9/16/14

20

Text Document Representation

- Each document becomes a 'term' vector,
 - each term is an (attribute) of the vector,
 - the value of each attribute is the number of times the corresponding term occurs in the document.

Bag of 'words'

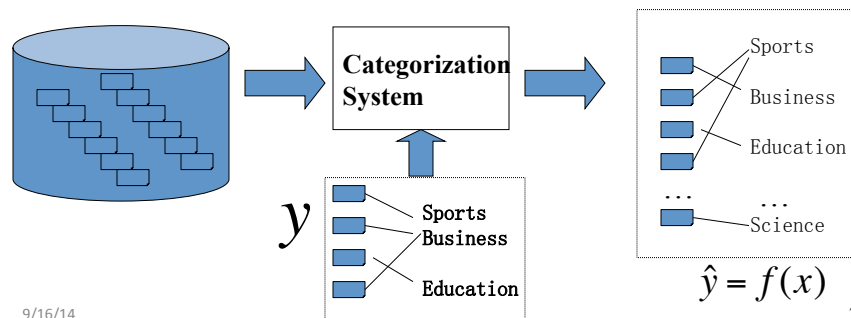
| | team | coach | play | ball | score | game | win | lost | timeout | season |
|------------|------|-------|------|------|-------|------|-----|------|---------|--------|
| Document 1 | 3 | 0 | 5 | 0 | 2 | 6 | 0 | 2 | 0 | 2 |
| Document 2 | 0 | 7 | 0 | 2 | 1 | 0 | 0 | 3 | 0 | 0 |
| Document 3 | 0 | 1 | 0 | 0 | 1 | 2 | 2 | 0 | 3 | 0 |

9/16/14

21

Text Categorization

- Pre-given categories and labeled document examples (Categories may form hierarchy)
- Classify new documents
- A standard supervised learning problem



9/16/14

22

Examples of Text Categorization

- News article classification
- Meta-data annotation
- Automatic Email sorting
- Web page classification

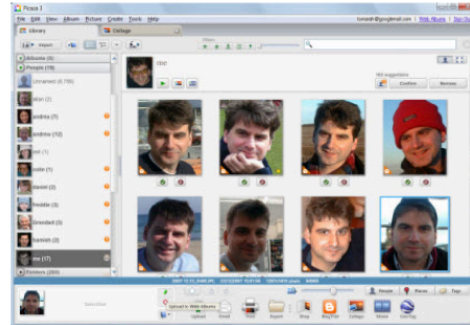
(3) Example Applications – Web Image Data

- Face detection in digital camera.



(3) Example Applications – Web Image Data

- Google (Web) Picasa face detection and recognition: image group of the same person.



9/16/14

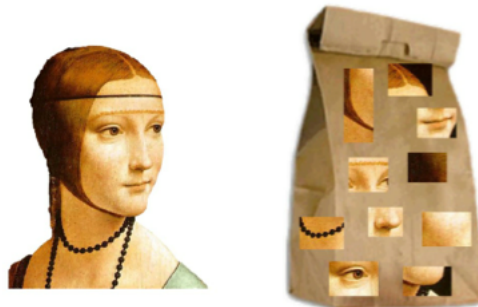
25

(3) Example Applications – Objective recognition

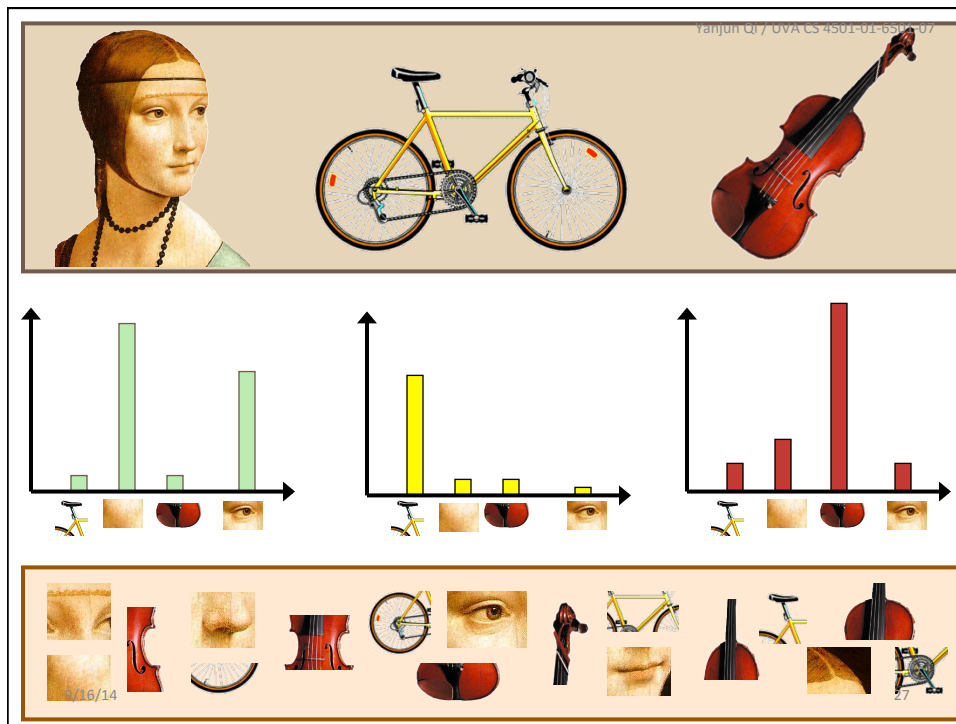
- Image representation → bag of “visual words”

Object → Bag of ‘words’

- An object image: histogram of visual vocabulary – a numerical vector of D dimensions.



9/16/14



YanJun Qi / UVA CS 4501-01-6501-07

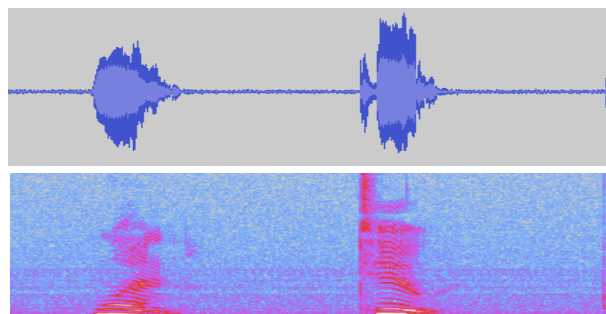
(3) Example Applications

– Objective recognition / Image Labeling

| Motorbikes | Airplanes | Faces | Cars (Side) | Cars (Rear) | Spotted Cats | Background |
|------------|-----------|-------|-------------|-------------|--------------|------------|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

9/16/14
28

(4) Example Applications – Audio Data



- Real-life applications:
 - Customer service phone routing
 - Voice recognition software

(4) Example Applications – e.g., Music information retrieval

- Analyzing musical data
- Query, recommend, visualize, transcribe, detect plagiarism, follow along with a score
- Sites you can try
 - Themefinder.com
 - Pandora.com (human-driven),
 - last.fm

(4) Example Applications – e.g., Automatic Music Classification

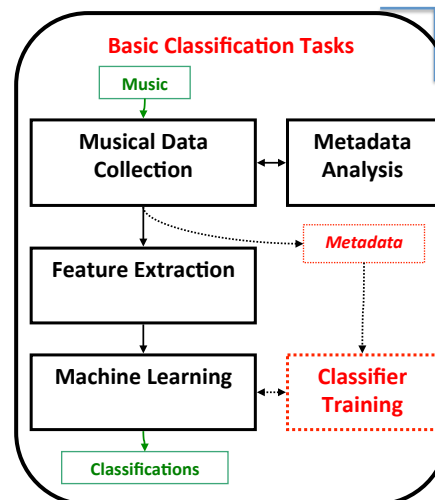
- Many areas of research in music information retrieval (MIR) involve using computers to classify music in various ways
 - Genre or style classification
 - Mood classification
 - Performer or composer identification
 - Music recommendation
 - Playlist generation
 - Hit prediction
 - Audio to symbolic transcription
 - etc.
- Such areas often share similar central procedures

Yanjun Qi / UVA CS
4501-01-6501-07

31

(4) Example Applications – e.g., Automatic Music Classification

- Musical data collection
 - The **instances** (basic entities) to classify
 - Audio recordings, scores, cultural data, etc.
- Feature extraction
 - **Features** represent characteristic information about instances
 - Must provide sufficient information to segment instances among **classes** (categories)
- Machine learning
 - Algorithms (“**classifiers**” or “**learners**”) learn to associate feature patterns of instances with their classes



9/16/14

32

(4) Example Applications – Audio, Types of features

- Low-level
 - Associated with signal processing and basic auditory perception
 - e.g. spectral flux or RMS
 - Usually not intuitively musical
- High-level
 - Musical abstractions
 - e.g. meter or pitch class distributions
- Cultural
 - Sociocultural information outside the scope of auditory or musical content
 - e.g. playlist co-occurrence or purchase correlations

Feature Extraction

Low-Level Features

High-Level Features

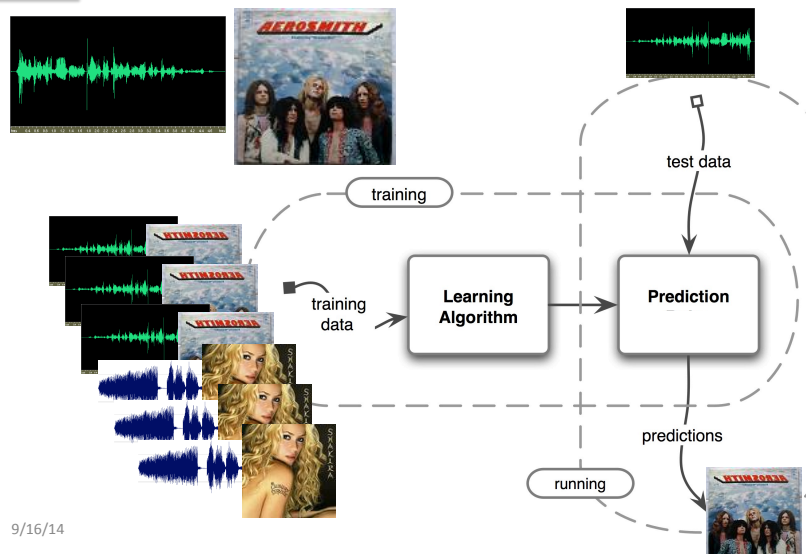
Cultural Features

Yanjun Qi / UVA CS
9/16/14
4501-01-6501-07

33

(4) Example Applications – e.g. Music Genre Classification

Yanjun Qi / UVA CS 4501-01-6501-07



9/16/14

34

Today

- ☐ Review of basic pipeline
- ☐ Interesting Applications

- ☐ Review of regression models
 - Linear regression (LR)
 - LR with non-linear basis functions, polynomial
 - LR with non-linear basis functions, RBF
 - Locally weighted LR
 - LR with Regularizations

9/16/14

35

(1) Linear Regression (LR)

$$f: X \longrightarrow Y$$

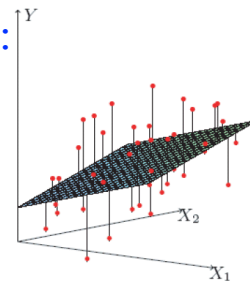
→ e.g. Linear Regression Models

$$\hat{y} = f(x) = \theta_0 + \theta_1 x^1 + \theta_2 x^2$$

→ To minimize the cost function:

$$J(\theta) = \frac{1}{2} \sum_{i=1}^n (\hat{y}_i(\bar{x}_i) - y_i)^2$$

→ θ



9/16/14

Yanjun Qi / UVA CS 4501-01-6501-07

Our goal:

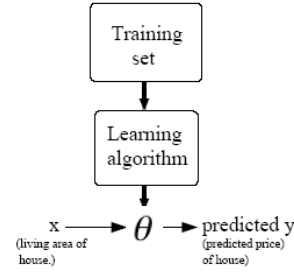
• We can represent the whole Training set:

$$\mathbf{X} = \begin{bmatrix} \text{--} & \mathbf{x}_1^T & \text{--} \\ \text{--} & \mathbf{x}_2^T & \text{--} \\ \vdots & \vdots & \vdots \\ \text{--} & \mathbf{x}_n^T & \text{--} \end{bmatrix} = \begin{bmatrix} x_1^0 & x_1^1 & \dots & x_1^{p-1} \\ x_2^0 & x_2^1 & \dots & x_2^{p-1} \\ \vdots & \vdots & \vdots & \vdots \\ x_n^0 & x_n^1 & \dots & x_n^{p-1} \end{bmatrix}$$

$\mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$

• Predicted output for each training sample:

$$\begin{bmatrix} f(\mathbf{x}_1^T) \\ f(\mathbf{x}_2^T) \\ \vdots \\ f(\mathbf{x}_n^T) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^T \boldsymbol{\theta} \\ \mathbf{x}_2^T \boldsymbol{\theta} \\ \vdots \\ \mathbf{x}_n^T \boldsymbol{\theta} \end{bmatrix} = \mathbf{X} \boldsymbol{\theta}$$



9/16/14 37

Yanjun Qi / UVA CS 4501-01-6501-07

Method I: normal equations

• Write the cost function in matrix form:

$$\begin{aligned}
 J(\boldsymbol{\theta}) &= \frac{1}{2} \sum_{i=1}^n (\mathbf{x}_i^T \boldsymbol{\theta} - y_i)^2 \\
 &= \frac{1}{2} (\mathbf{X} \boldsymbol{\theta} - \bar{\mathbf{y}})^T (\mathbf{X} \boldsymbol{\theta} - \bar{\mathbf{y}}) \\
 &= \frac{1}{2} (\boldsymbol{\theta}^T \mathbf{X}^T \mathbf{X} \boldsymbol{\theta} - \boldsymbol{\theta}^T \mathbf{X}^T \bar{\mathbf{y}} - \bar{\mathbf{y}}^T \mathbf{X} \boldsymbol{\theta} + \bar{\mathbf{y}}^T \bar{\mathbf{y}})
 \end{aligned}$$

$$\mathbf{X} = \begin{bmatrix} \text{--} & \mathbf{x}_1^T & \text{--} \\ \text{--} & \mathbf{x}_2^T & \text{--} \\ \vdots & \vdots & \vdots \\ \text{--} & \mathbf{x}_n^T & \text{--} \end{bmatrix} \quad \bar{\mathbf{y}} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

To minimize $J(\boldsymbol{\theta})$, take derivative and set to zero:

$$\Rightarrow \boxed{X^T X \boldsymbol{\theta} = X^T \bar{\mathbf{y}}}$$

The normal equations

$$\boldsymbol{\theta}^* = (X^T X)^{-1} X^T \bar{\mathbf{y}}$$

9/16/14 38

(2) LR with polynomial basis functions

- LR does not mean we can only deal with linear relationships

$$y = \theta_0 + \sum_{j=1}^m \theta_j \phi_j(x) = \varphi(x) \theta$$

where the $\phi_j(x)$ are fixed basis functions (also define $\phi_0(x) = 1$)

- E.g.: polynomial regression:

$$\varphi(x) := [1, x, x^2, x^3]$$

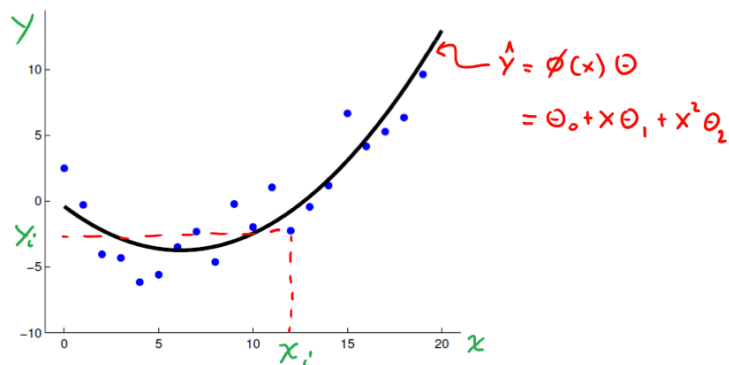
$$\theta^* = (\varphi^T \varphi)^{-1} \varphi^T \bar{y}$$

9/16/14

39

e.g. polynomial regression

For example, $\phi(x) = [1, x, x^2]$



9/16/14

40
Dr. Nando de Freitas's tutorial slide

(3) LR with radial-basis functions

- LR does not mean we can only deal with linear relationships

$$\hat{y} = \theta_0 + \sum_{j=1}^m \theta_j \phi_j(x) = \varphi(x)\theta$$

where the $\phi_j(x)$ are fixed basis functions (also define $\phi_0(x) = 1$)

- E.g.: LR with RBF regression:

$$\varphi(x) := [1, K_{\lambda=1}(x,1), K_{\lambda=1}(x,2), K_{\lambda=1}(x,4)]$$

$$\theta^* = (\varphi^T \varphi)^{-1} \varphi^T \bar{y}$$

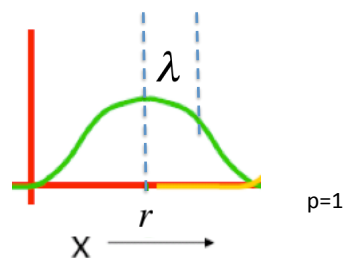
9/16/14

41

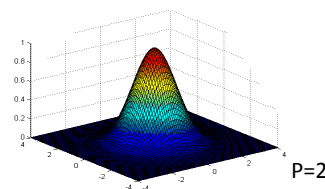
RBF = radial-basis function: a function which depends only on the radial distance from a centre point

Gaussian RBF →
$$K_{\lambda}(x, r) = \exp\left(-\frac{(x-r)^2}{2\lambda^2}\right)$$

as distance from the centre r increases, the output of the RBF decreases



9/16/14

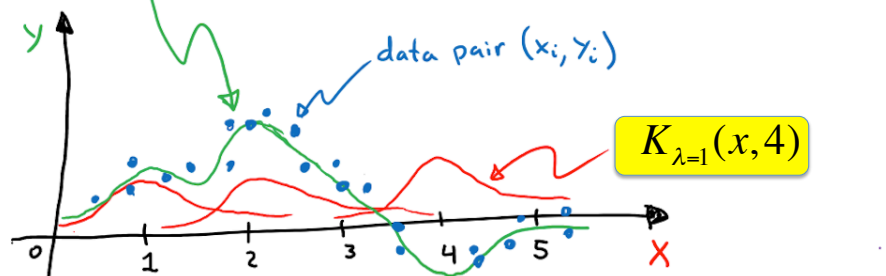


42

(2) Linear regression with RBF basis functions (predefined centres)

$$\varphi(x) := [1, K_{\lambda=1}(x, 1), K_{\lambda=1}(x, 2), K_{\lambda=1}(x, 4)]$$

$$\hat{y} = \theta_0 + e^{-\|x-1\|^2} \theta_1 + e^{-\|x-2\|^2} \theta_2 + e^{-\|x-4\|^2} \theta_3$$



The green curve is a weighted sum of the red curves

9/16/14

Dr. Nando de Freitas's tutorial slide

$$k_{\lambda=1}(x, 1) = e^{-\|x-1\|^2}$$

$$k_{\lambda=1}(x, 2) = e^{-\|x-2\|^2}$$

$$k_{\lambda=1}(x, 4) = e^{-\|x-4\|^2}$$

$$\varphi(x) := [1, e^{-\|x-1\|^2}, e^{-\|x-2\|^2}, e^{-\|x-4\|^2}]$$

$$\varphi(x) := [1, K_{\lambda=1}(x, 1), K_{\lambda=1}(x, 2), K_{\lambda=1}(x, 4)]$$

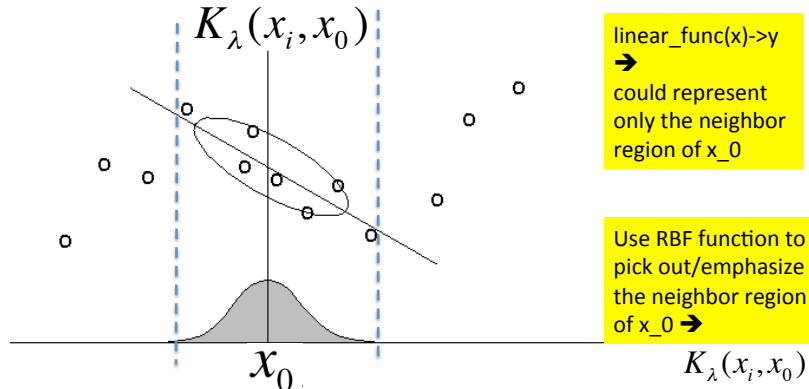
$$\hat{y} = \varphi(x)\theta$$

$$= \theta_0 + \theta_1 \exp(-(x-1)^2) + \theta_2 \exp(-(x-2)^2) + \theta_3 \exp(-(x-4)^2)$$

$$\theta^* = (\varphi^T \varphi)^{-1} \varphi^T \bar{y}$$

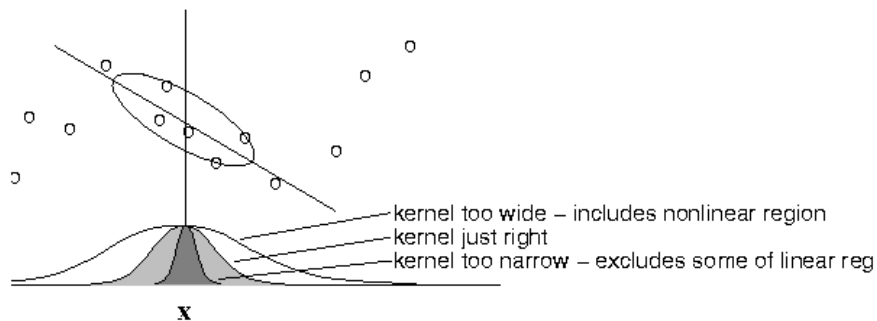
(3) Locally weighted regression

- aka locally weighted regression, locally linear regression, LOESS, ...



9/16/14 **Figure 2:** In locally weighted regression, points are weighted by proximity to the current x in question using a kernel. A regression is then computed using the weighted points.

(3) Locally weighted linear regression



9/16/14 **Figure 3:** The estimator variance is minimized when the kernel includes as many training points as can be accommodated by the model. Here the linear LOESS model is shown. Too large a kernel includes points that degrade the fit; too small a kernel neglects points that increase confidence in the fit.

(3) Locally weighted linear regression

Yanjun Qi / UVA CS 4501-01-6501-07

- Separate weighted least squares **at each target point x_0** :

$$\min_{\alpha(x_0), \beta(x_0)} \sum_{i=1}^N K_{\lambda}(x_i, x_0) [y_i - \alpha(x_0) - \beta(x_0)x_i]^2$$

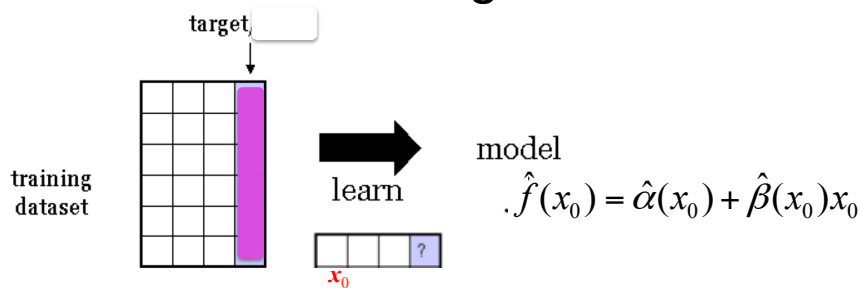
$$\hat{f}(x_0) = \hat{\alpha}(x_0) + \hat{\beta}(x_0)x_0$$

9/16/14

47

LEARNING of Locally weighted linear regression

Yanjun Qi / UVA CS 4501-01-6501-07



- Separate weighted least squares **at each target point x_0**

9/16/14

48

(4) LR with Regularizations / Regularized multivariate linear regression

- Basic model
$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$
 - LR estimation:
$$\min J(\beta) = \sum (Y - \hat{Y})^2$$
 - LASSO estimation:
$$\min J(\beta) = \sum_{i=1}^n (Y - \hat{Y})^2 + \lambda \sum_{j=1}^p |\beta_j|$$
 - Ridge regression estimation:
$$\min J(\beta) = \sum_{i=1}^n (Y - \hat{Y})^2 + \lambda \sum_{j=1}^p \beta_j^2$$
- Error on data + Regularization

9/16/14

(4) LR with Regularizations / Ridge Estimator

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$

$$\beta^* = (X^T X + \lambda I)^{-1} X^T \bar{y}$$

- The ridge estimator is solution from

$$\hat{\beta}^{ridge} = \arg \min J(\beta) = \arg \min (y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta$$

to minimize $J(\beta)$, take derivative and set to zero

9/16/14

50

Today Recap

- Review of basic pipeline
- Interesting Applications (text / image / audio)

- Review of five regression models we have covered
 - Linear regression (LR)
 - LR with non-linear basis functions, polynomial
 - LR with non-linear basis functions - RBF
 - Locally weighted LR
 - LR with Regularizations

References

- Big thanks to Prof. Eric Xing @ CMU for allowing me to reuse some of his slides
- Elements of Statistical Learning, by Hastie, Tibshirani and Friedman
- jMIR's tutorial slide about "Automatic Music Classification"
- Duc-Hieu Tran's tutorial slide about "Machine Learning for Computer Vision Applications"