Brain Break

16:55
CSE/STAT 416
Course Wrap Up

Hunter Schafer
University of Washington
Aug 19, 2019

Slides borrowed from Emily Fox
- Regression
- Overfitting
- Training, test, and generalization error
- Bias-Variance tradeoff
- Ridge, LASSO
- Cross validation
- Gradient descent
- Classification
- Logistic regression
- Decision trees
- Boosting
- Precision and recall
- Nearest-neighbor retrieval, regression, and classification
- Kernel regression
- Locality sensitive hashing
- Dimensionality reduction, PCA
- k-means clustering
- Hierarchical clustering
- Unsupervised v. supervised learning
- Recommender systems
- Matrix factorization
- Coordinate descent
- Neural networks
- Convolutional neural networks
- Transfer learning for deep learning
Case Study 1: Predicting house prices

**Data**

- House size
- Price ($)
- House features

**Regression**

Price ($) = ??

**Intelligence**

List price? (sales price)
Regression

Case study: Predicting house prices

Models
- Linear regression
- Regularization: Ridge (L2), Lasso (L1)

Including many features:
- Square feet
- # bathrooms
- # bedrooms
- Lot size
- Year built
- ...

\[
y_i = f(x_i) + \varepsilon_i \quad \hat{y}_i = \hat{f}(x_i)
\]

\[RSS(w) + \lambda ||w||_2^2\]
Regression

Case study: Predicting house prices

Algorithms

• Gradient descent

$$RSS(w_0, w_1) = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$$

$$RSS(w) = \sum_{i=1}^{n} (w^T h(x_i) - y_i)^2$$

$$RSS(w_0, w_1) = (\$_{\text{house } 1} - [w_0 + w_1 \text{ sq.ft.}_{\text{house } 1}])^2 + (\$_{\text{house } 2} - [w_0 + w_1 \text{ sq.ft.}_{\text{house } 2}])^2 + (\$_{\text{house } 3} - [w_0 + w_1 \text{ sq.ft.}_{\text{house } 3}])^2 + \ldots$$

[include all houses]
Regression

*Case study: Predicting house prices*

**Concepts**
- Loss functions, bias-variance tradeoff, cross-validation, sparsity, overfitting, model selection

1. **Noise**
2. **Bias**
3. **Variance**

---

STAT/CSE 416: Intro to Machine Learning
Case Study 2: Sentiment analysis

All reviews:
- "Sushi was awesome, the food was awesome, but the service was awful."
- "Score(x) > 0"
- "Score(x) < 0"
- "Sushi was awesome, the food was awesome, but the service was awful."

STAT/CSE 416: Intro to Machine Learning
Classification

Case study: Analyzing sentiment

Models

- Linear classifiers (logistic regression)
- Multiclass classifiers
- Decision trees
- Boosted decision trees and random forests

Ensemble Method

\[ \text{Score}(x) = \text{sign}(\sum_{i} \hat{\alpha}_i \hat{f}_i(x)) \]

\( \hat{f}_i(x) \) is model weight
\( \alpha_i \) is dataset weight

STAT/CSE 416: Intro to Machine Learning
Classification

Case study: Analyzing sentiment

Algorithms

- Boosting
- Learning from weighted data

Income>$100K?
  Yes Safe
  No Risky

Credit history?
  Bad Risky
  Good Safe

Savings>$100K?
  Yes Safe
  No Risky

Market conditions?
  Bad Risky
  Good Safe

weighted_error = 0.2
weighted_error = 0.35
weighted_error = 0.3
weighted_error = 0.4
Classification

Case study: Analyzing sentiment

Concepts

- Decision boundaries, maximum likelihood estimation, ensemble methods, random forests
- Precision and recall

\[ \ell(w_0, w_1, w_2) \]

Accuracy vs class imbalance

Classifier A
Best classifier
Classifier B

STAT/CSE 416: Intro to Machine Learning
Case Study 3:
Document retrieval

1. Data
2. Nearest neighbor
3. Intelligence
Case Study 3+:
Document structuring for retrieval

Data → Clustering → Intelligence

Bag of Words
TF-IDF
Euclidean
Manhattan
Cosine
Case Study 3++: Dimensionality reduction

Data → PCA → Intelligence

Can we give each image a coordinate, such that similar images are near each other?

Images with thousands or millions of pixels

[Saul & Roweis ’03]

STAT/CSE 416: Intro to Machine Learning
Clustering & Retrieval

Case study: Finding documents

Models

- Nearest neighbors
- Clustering
- Hierarchical clustering

query article

set of nearest neighbors

/ Kernel methods

Agglomerative

STAT/CSE 416: Intro to Machine Learning
Clustering & Retrieval

Case study: Finding documents

Algorithms
- k-means
- k-means++
- Locality-sensitive hashing (LSH)
- NN regression and classification
- Kernel regression
- Agglomerative and divisive clustering
- PCA

STAT/CSE 416: Intro to Machine Learning
Clustering & Retrieval

Case study: Finding documents

Concepts

• Distance metrics, kernels, approximation algorithms, dimensionality reduction

Features:

1 0 0 0 5 3 0 0 1 0 0 0

3 0 0 0 2 0 0 1 0 1 0 0 0

1*3
+ 5*2
= 13

Bin index:
[0 0 0]

Bin index:
[0 1 0]

Bin index:
[1 1 0]

Bin index:
[1 1 1]

Feature 1

Feature 2

Principal components:

Reconstructing:

STAT/CSE 416: Intro to Machine Learning
Case Study 4: Image classification

Data → Deep Learning → Intelligence

Layer 1
1
x₁
x₂

Layer 2
1
z₁
z₂

y

Face?
Deep Learning

Case study: Image classification

Models

- Perceptron
- General neural network
- Convolutional neural network
Deep Learning

*Case study: Image classification*

**Algorithms**
- Convolutions
- Backpropagation (high level only)

STAT/CSE 416: Intro to Machine Learning
Deep Learning

Case study: Image classification

Concepts

- Activation functions, hidden layers, architecture choices
Case Study 5: Product recommendation

Your past purchases:

Data

+ purchase histories of all customers

Matrix Factorization

Intelligence

Recommended items:

Customers

Features

Products

Features
Recommender Systems & Matrix Factorization

Case study: Recommending Products

Models

- Collaborative filtering
- Matrix factorization

\[
\text{Rating} = \approx \begin{bmatrix} L \end{bmatrix} \approx \begin{bmatrix} R' \end{bmatrix}
\]

Parameters of model

Popularity
cocurrence matrix
Featured MF

STAT/CSE 416: Intro to Machine Learning
Recommender Systems & Matrix Factorization

Case study: Recommending Products

Algorithms
- Coordinate descent

Rating = \begin{pmatrix} L \end{pmatrix} \begin{pmatrix} R' \end{pmatrix}

Form estimates \hat{L}_u and \hat{R}_v
Recommender Systems & Matrix Factorization

Case study: Recommending Products

Concepts

- Matrix completion, cold-start problem
Training Data → Feature extraction → ML model → Quality metric → ML algorithm → ML model → Feature extraction → Training Data
Improving the performance at some task through experience!

- Before you start any learning task, remember fundamental questions that will impact how you go about solving it:

  What is the learning problem?

  What model?

  With what optimization algorithm?

  How will you evaluate the model?

  From what experience?

  What loss function are you optimizing?

  Are there any guarantees?
Congrats on finishing CSE/STAT 416! Thanks for the hard work!