What’s next for ML & you

Emily Fox & Carlos Guestrin
Machine Learning Specialization
University of Washington
Deploying an ML service
What is Production?

Serving live predictions

Deployment

Choosing between deployed models

Management

Measuring quality of deployed models

Evaluation

Tracking model quality & operations

Monitoring

©2015 Emily Fox & Carlos Guestrin
Lifecycle of ML in Production

- Deployment
- Evaluation
- Management
- Monitoring
The Setup...

Suppose we are building a website with product recommendations, trained using user reviews.

- 34.6M reviews
- 2.4M products
- 6.6M users
Deployment System

Batch training

Real-time predictions

Historical Data → Model → Live Data

User & session info

recommendations

Feedback

Predictions
What happens after (initial) deployment
Lifecycle of ML in Production

- Deployment
- Evaluation
- Management
- Monitoring
After deployment

Evaluate and track metrics over time
React to feedback from deployed models
Feedback loop for ML in production

Batch training → Model → Historical Data → Live Data → Real-time predictions → Predictions → Feedback

©2015 Emily Fox & Carlos Guestrin
Learning new, alternative models

Batch training  →  Real-time predictions

Historical Data → Model → Cloud Computing → Predictions → Live Data

Model 2
Key questions

• When to update a model?
• How to choose between existing models?
• Answer: continuous evaluation and testing
What is evaluation?

Evaluation = Predictions + Metric

What data?  Which metric?
Evaluating a recommender

Offline evaluation: When to update model

Online evaluation: Choosing between models

Historical Data → Model → Sum squared error

Live Data → Predictions → User engagement

Predictions
Updating ML models

Why update?
• Trends and user tastes change over time
• Model performance drops

When to update?
• Track statistics of data over time
• Monitor both offline & online metrics
• Update when offline metric diverges from online metrics or not achieving desired targets
A/B Testing: Choosing between ML models

Group A
- Everybody gets Model 2
- 2000 visits
- 10% CTR

Model 1

Group B
- 2000 visits
- 30% CTR

Model 2

©2015 Emily Fox & Carlos Guestrin
Other production considerations

• A/B testing caveats
  – Also multi-armed bandits
• Versioning
• Provenance
• Dashboards
• Reports
• ...

©2015 Emily Fox & Carlos Guestrin
Machine learning challenges
Open challenges: Model selection

User info
- Purchase history
- Product info
- Other info

Classifier

Yes!
No

$R' = \approx L$

Parameters of model

$X_{ij}$ known for black cells
$X_{ij}$ unknown for white cells

Rows index movies
Columns index users

Rating =
Open challenges: Feature engineering/representation

• Bag of word raw counts?
• Normalize?
• tf-idf? (which version???)
• Bigrams
• Trigrams
• …
Open challenges: Scaling

Data is getting big...

©2015 Emily Fox & Carlos Guestrin
Open challenges: Scaling

Concurrently, models are getting big...

©2015 Emily Fox & Carlos Guestrin
CPUs stopped getting faster...
ML in the context of parallel architectures

But scalable ML in these systems is **hard**, especially in terms of:

1. Programmability
2. Data distribution
3. Failures
What’s ahead in this specialization
2. 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
- ...
2. Regression

Case study: Predicting house prices

Algorithms
- Gradient descent
- Coordinate descent

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

\[\hat{W}\]
2. Regression

Case study: Predicting house prices

Concepts

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

![Graph showing the relationship between price ($) and square feet (sq.ft).]
3. Classification

Case study: Analyzing sentiment

Models

- Linear classifiers (logistic regression, SVMs, perceptron)
- Kernels
- Decision trees
3. Classification

Case study: Analyzing sentiment

Algorithms

- Stochastic gradient descent
- Boosting

Squeezing last bit of accuracy by blending models
3. Classification

Case study: Analyzing sentiment

- Decision boundaries, MLE, ensemble methods, random forests, CART, online learning

Time

This is probably my favorite place to eat Japanese in Seattle. My boyfriend and I ordered nigt of scallop, Japanese snapper (seasonal), and the ageodahi tofu and 2 special rolls. I would skip the special rolls, because the nigt and sashimi cuts is where this place excels. The tofu, as recommended by other Yelpers was amazing. It's more chewy and the sauce/gravvy is the perfect amount of flavor for the delicate tofu.

Dining here at the sushi bar made me feel like sitting front row to an amazing performance. We didn’t have resos, bended down to the ID after work, got here breathlesly at 5:10pm, and got the last two seats in the place.

I came here having high expectations due to the reviews of this place, but I was bit disappointed. The restaurant is small so do make reservations when you come here. Dishes cost from $4-26 each and dishes are small.
4. Clustering & Retrieval

Case study: Finding documents

Models

- Nearest neighbors
- Clustering, mixtures of Gaussians
- Latent Dirichlet allocation (LDA)
4. Clustering & Retrieval

Case study: Finding documents

Algorithms

- KD-trees, locality-sensitive hashing (LSH)
- K-means
- Expectation-maximization (EM)
4. Clustering & Retrieval

Case study: Finding documents

Concepts

- Distance metrics, approximation algorithms, hashing, sampling algorithms, scaling up with map-reduce

\[
\begin{align*}
1000530010000 & \quad \text{1*3} \\
3000200101000 & \quad + \\
5000200101000 & \quad \text{5*2} \\
& \quad = 13
\end{align*}
\]
5. Recommender Systems & Dimensionality Reduction

Case study: Recommending Products

Models

- Collaborative filtering
- Matrix factorization
- PCA

Rating = \hat{R} 

\approx L \times R' 

Parameters of model

©2015 Emily Fox & Carlos Guestrin
5. Matrix Factorization & Dimensionality Reduction

Case study: Recommending Products

Algorithms

- Coordinate descent
- Eigen decomposition
- SVD

\[ X_{ij} \text{ known for black cells} \]
\[ X_{ij} \text{ unknown for white cells} \]

Rows index movies

Columns index users

\[ X = \text{Rating} \]

Form estimates \( \hat{L}_u \) and \( \hat{R}_v \)

\[ \approx \]

\[ L \]

\[ R' \]
5. Matrix Factorization & Dimensionality Reduction

Case study: Recommending Products

Concepts

- Matrix completion, eigenvalues, random projections, cold-start problem, diversity, scaling up

Customers

Products

Customers

Products
6. Capstone: Build and deploy an intelligent application with deep learning

- Text sentiment analysis
- Computer vision
- Recommenders
- Deep learning
- Deploy intelligent web app