

# 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

Measuring quality of  
deployed models



Evaluation



Choosing between  
deployed models

Management

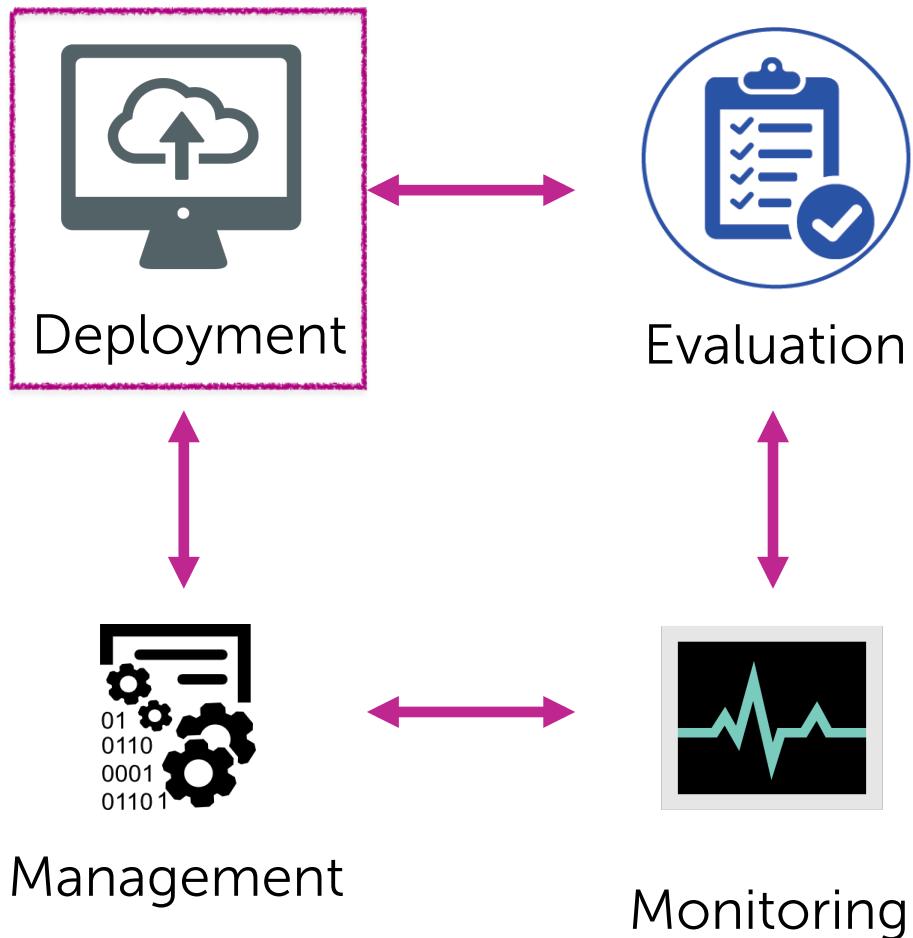
Tracking model  
quality & operations



Monitoring

©2015 Emily Fox & Carlos Guestrin

# Lifecycle of ML in Production



# The Setup...

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

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

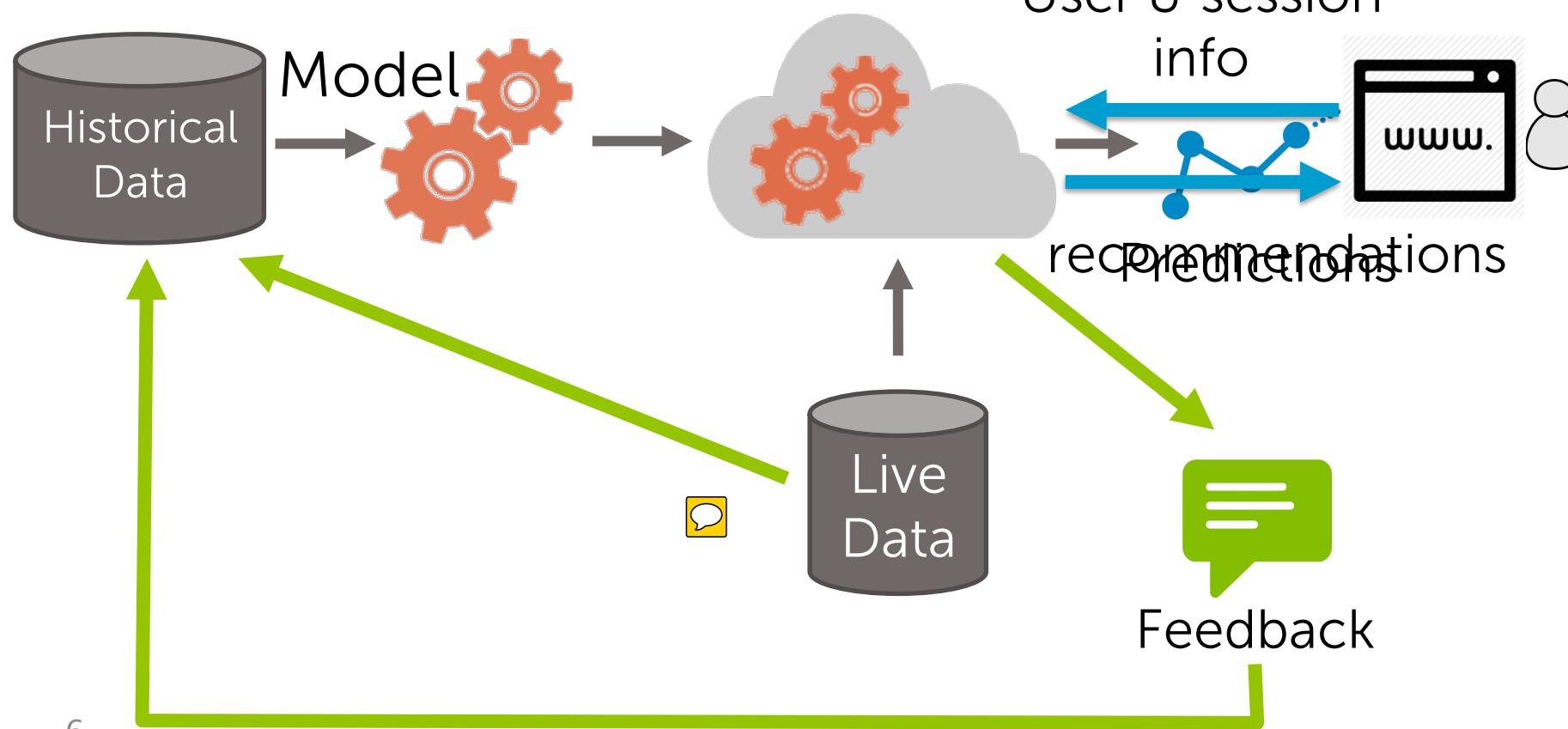
# Deployment System



Batch training

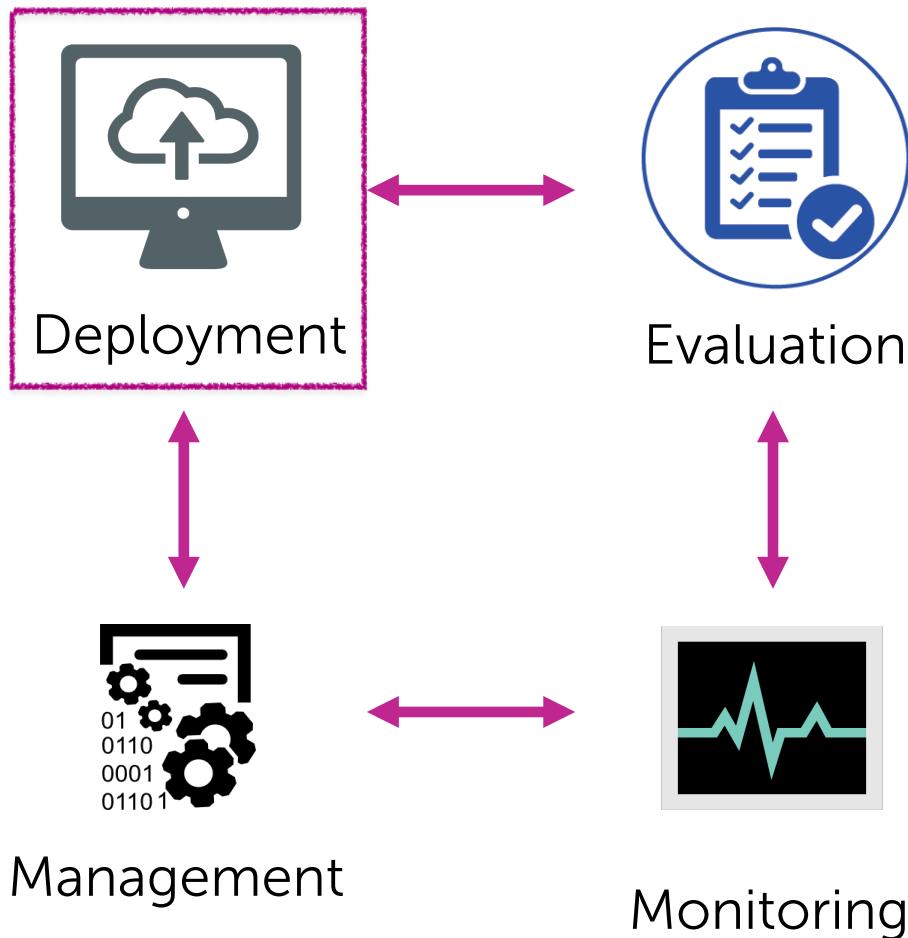
Real-time predictions

User & session



# What happens after (initial) deployment

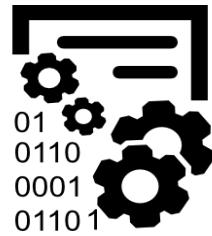
# Lifecycle of ML in Production



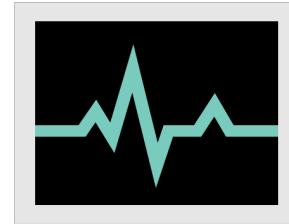
# After deployment



Evaluation



Management

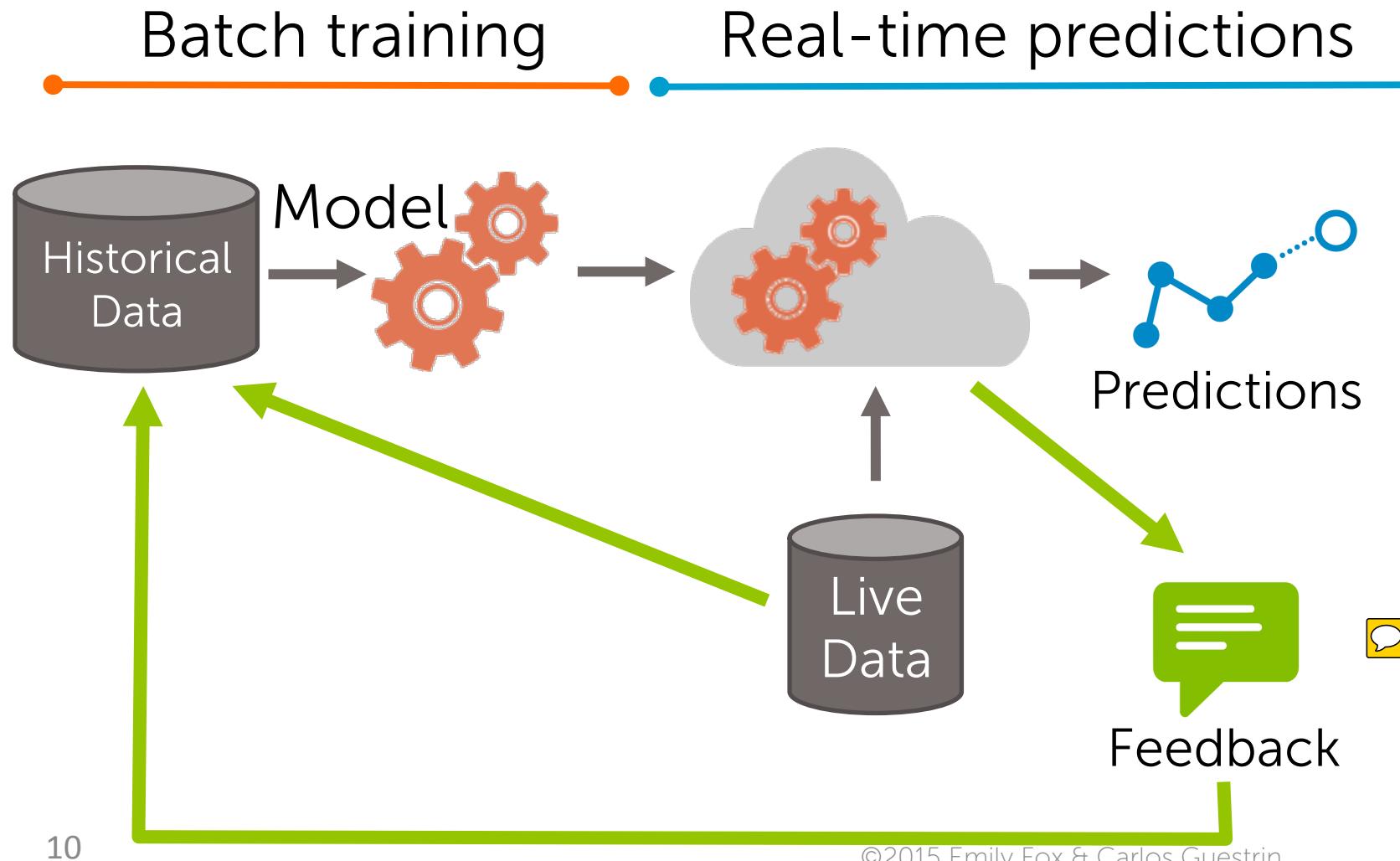


Monitoring

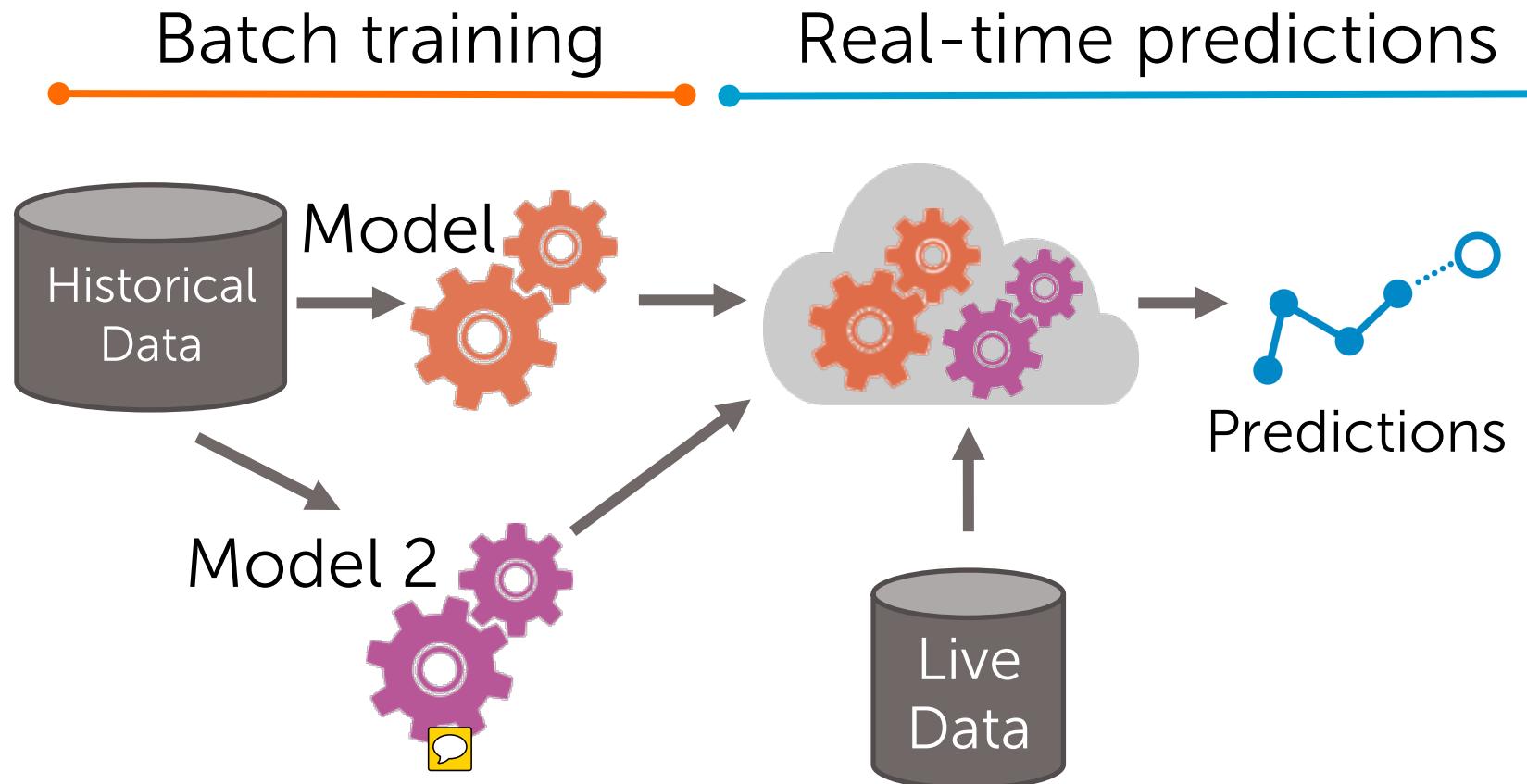
Evaluate and track metrics over time

React to feedback from deployed models

# Feedback loop for ML in production



# Learning new, alternative models



# Key questions

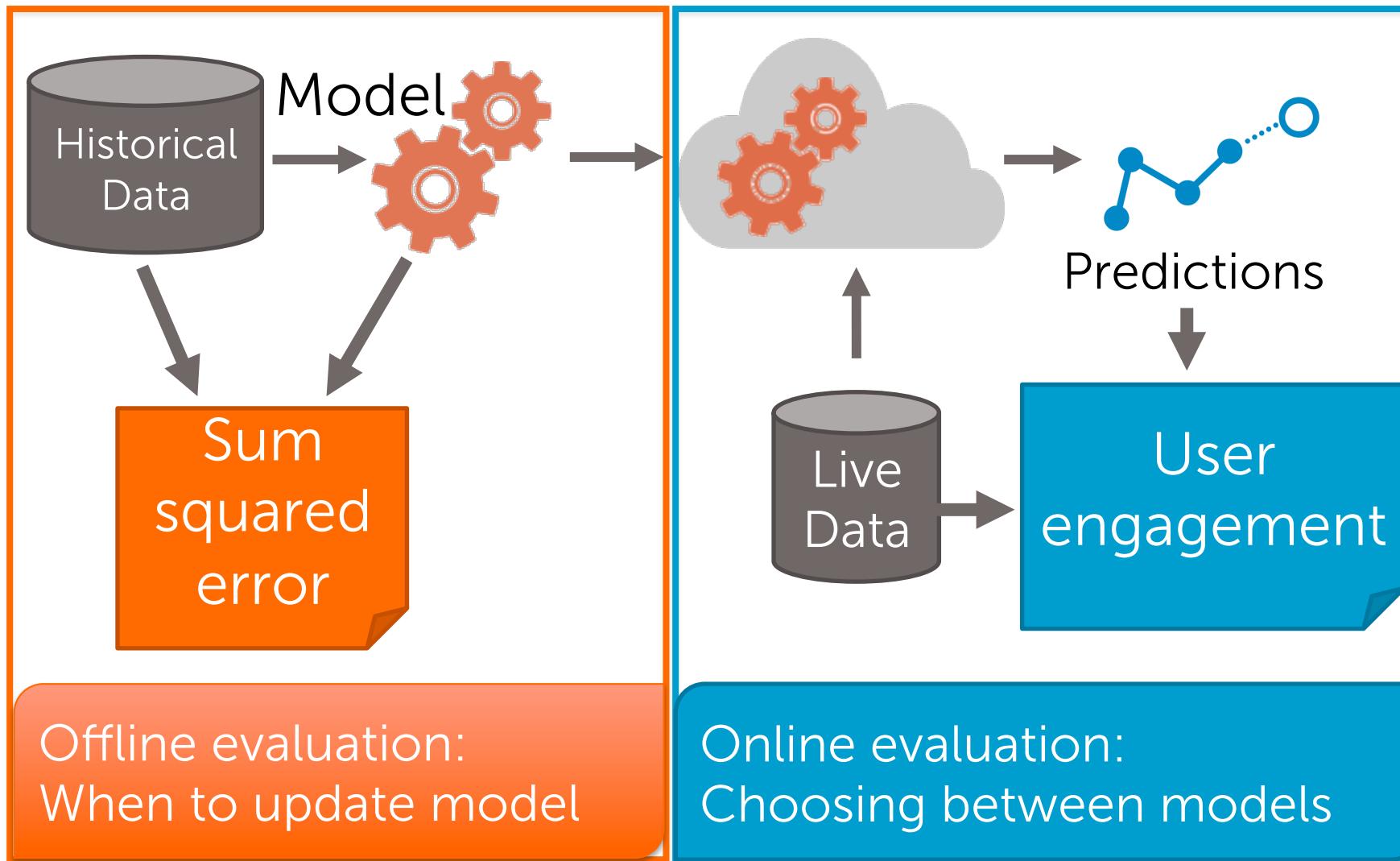
- When to update a model?
- How to choose between existing models?
- Answer: continuous evaluation and testing

# What is evaluation?



What data?  
Which metric?

# Evaluating a recommender



# Updating ML models

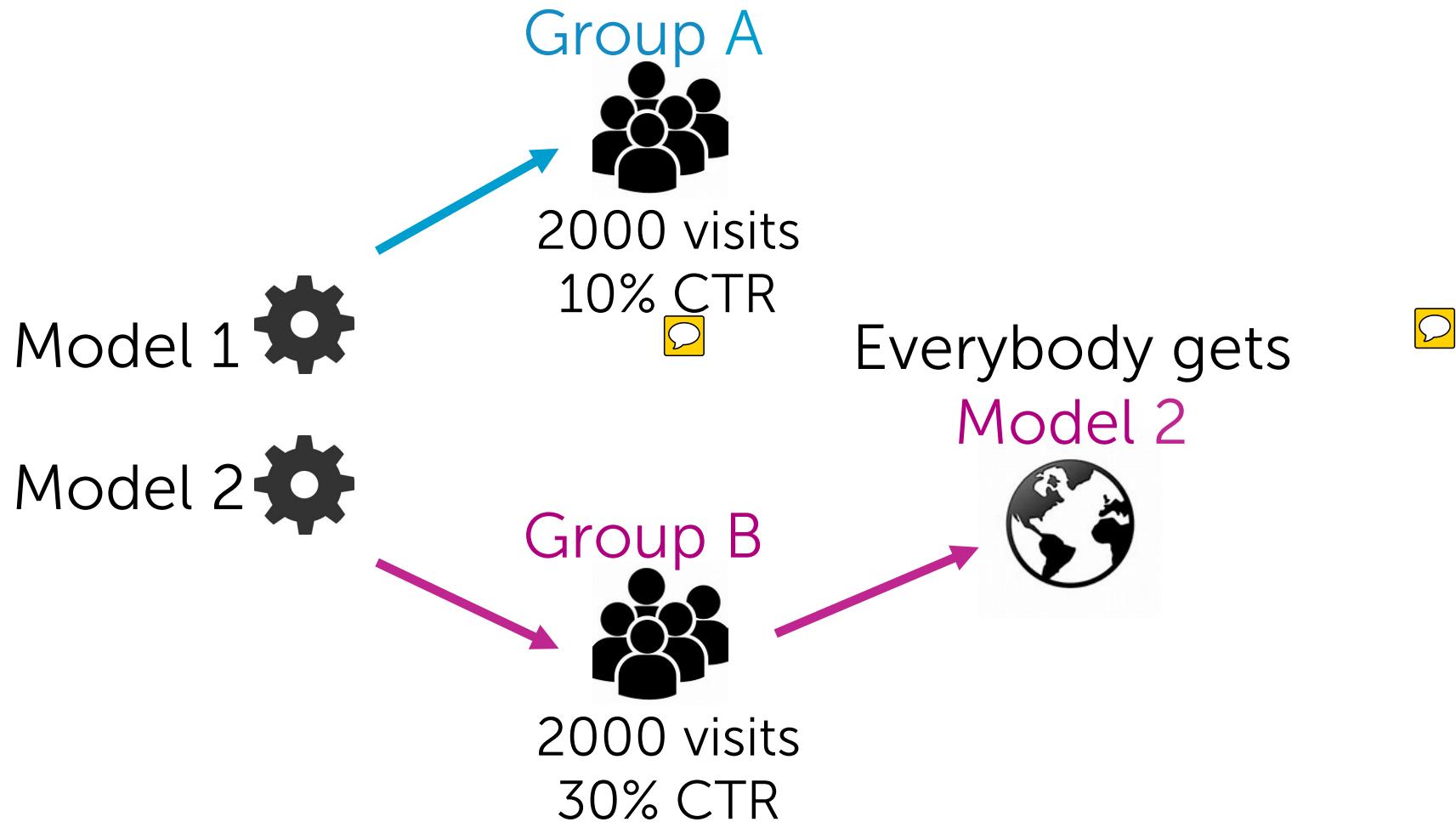
## Why update?

- 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

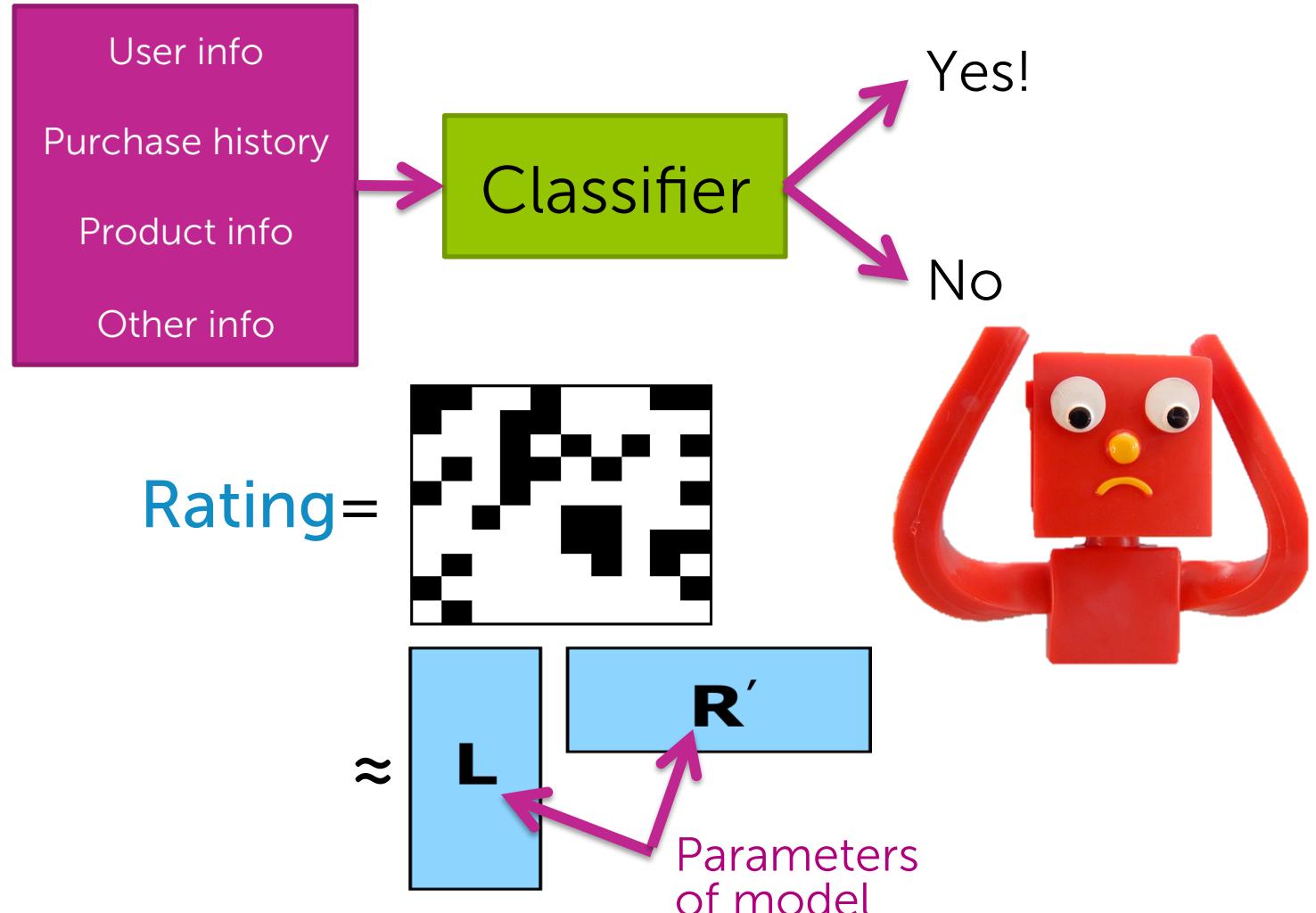


# Other production considerations

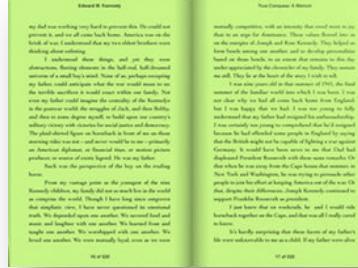
- A/B testing caveats
  - Also multi-armed bandits
- Versioning
- Provenance
- Dashboards
- Reports
- ...

# Machine learning challenges

# Open challenges: Model selection



# Open challenges: Feature engineering/representation



- Bag of word raw counts?
- Normalize?
- tf-idf? (which version???)
- Bigrams
- Trigrams
- ...

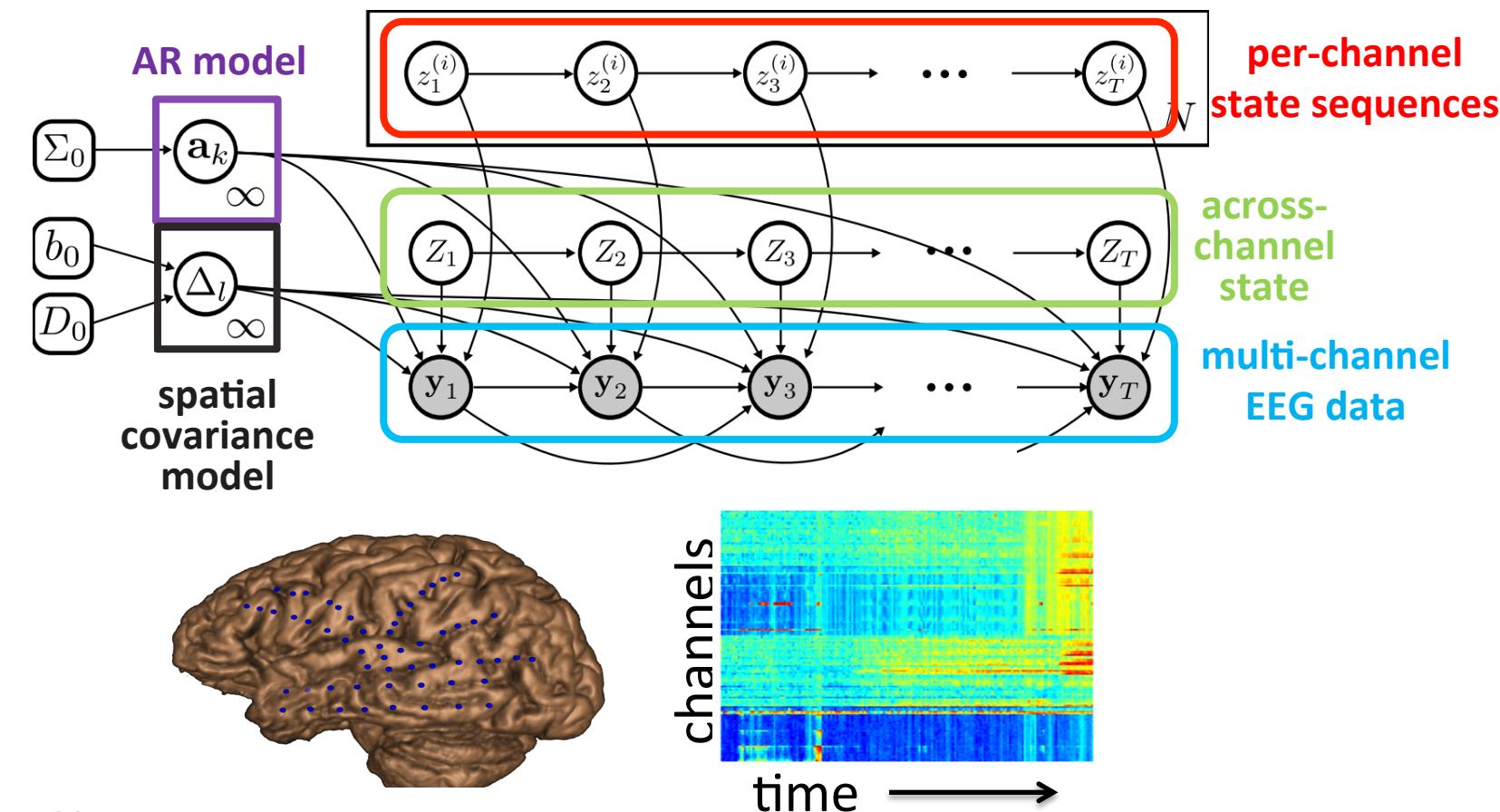
# Open challenges: Scaling

Data is getting big...

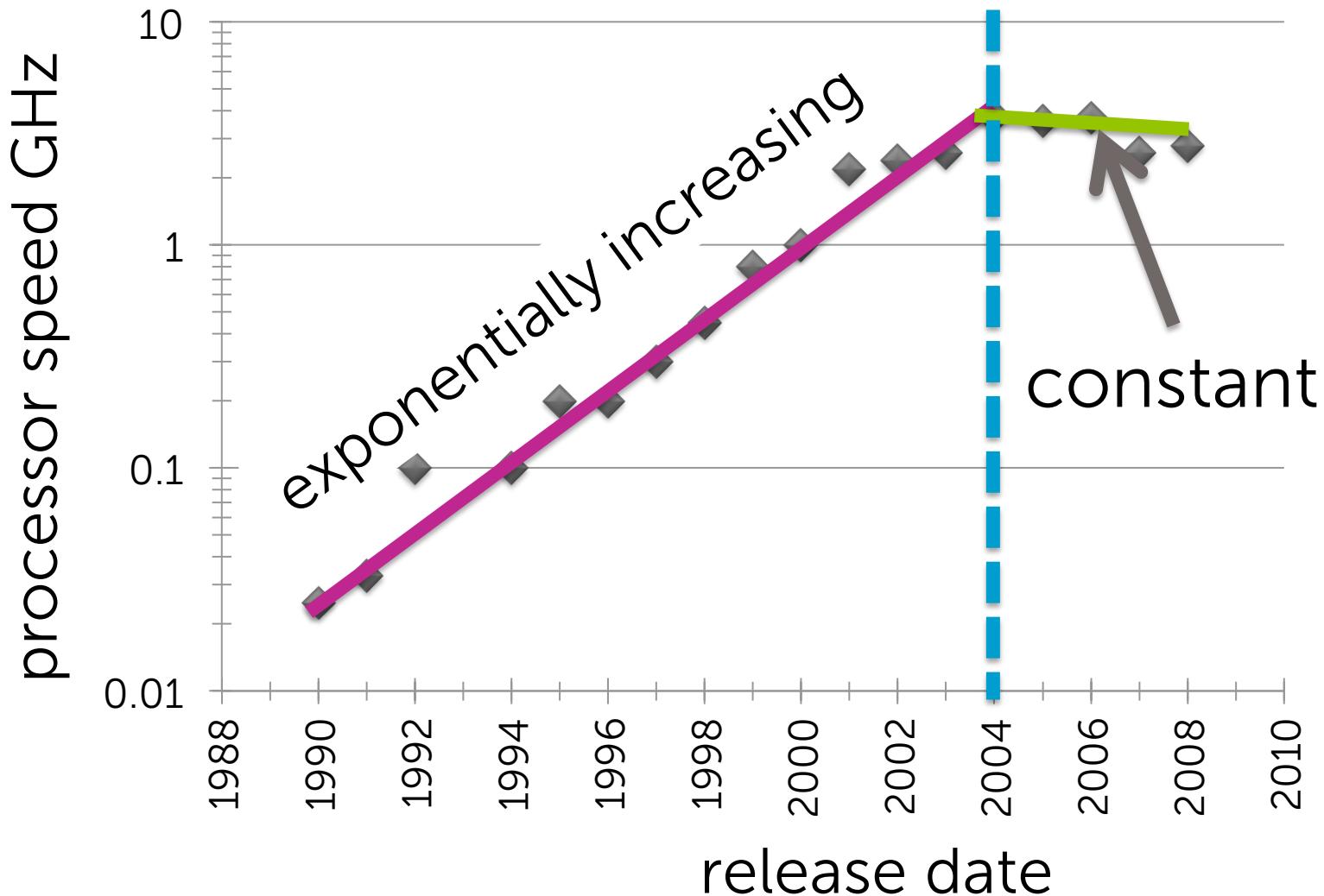


# Open challenges: Scaling

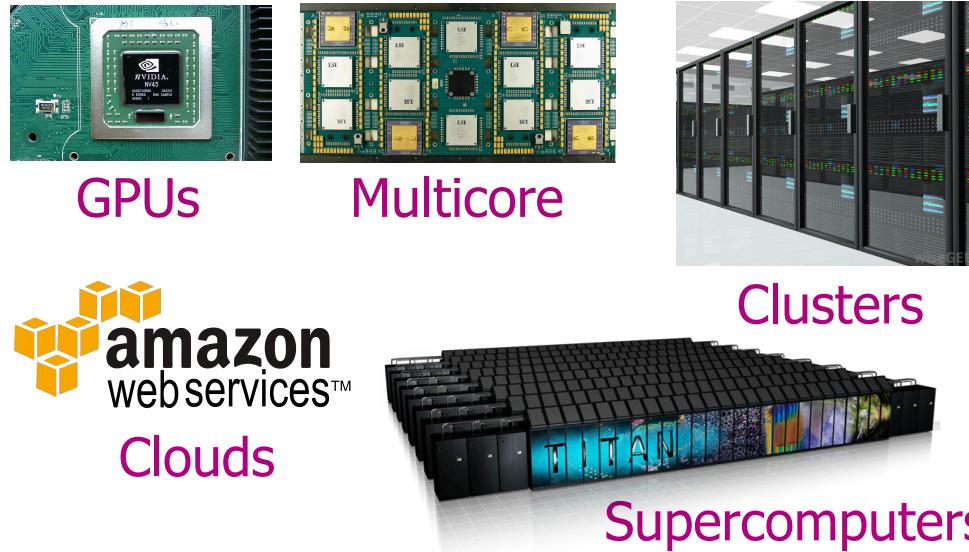
Concurrently, models are getting big...



# 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

$$\begin{aligned} \text{RSS}(\mathbf{w}_0, \mathbf{w}_1) = & (\$_{\text{house 1}} - [\mathbf{w}_0 + \mathbf{w}_1 \text{sq.ft.}_{\text{house 1}}])^2 \\ & + (\$_{\text{house 2}} - [\mathbf{w}_0 + \mathbf{w}_1 \text{sq.ft.}_{\text{house 2}}])^2 \\ & + (\$_{\text{house 3}} - [\mathbf{w}_0 + \mathbf{w}_1 \text{sq.ft.}_{\text{house 3}}])^2 \\ & + \dots \text{[include all houses]} \end{aligned}$$



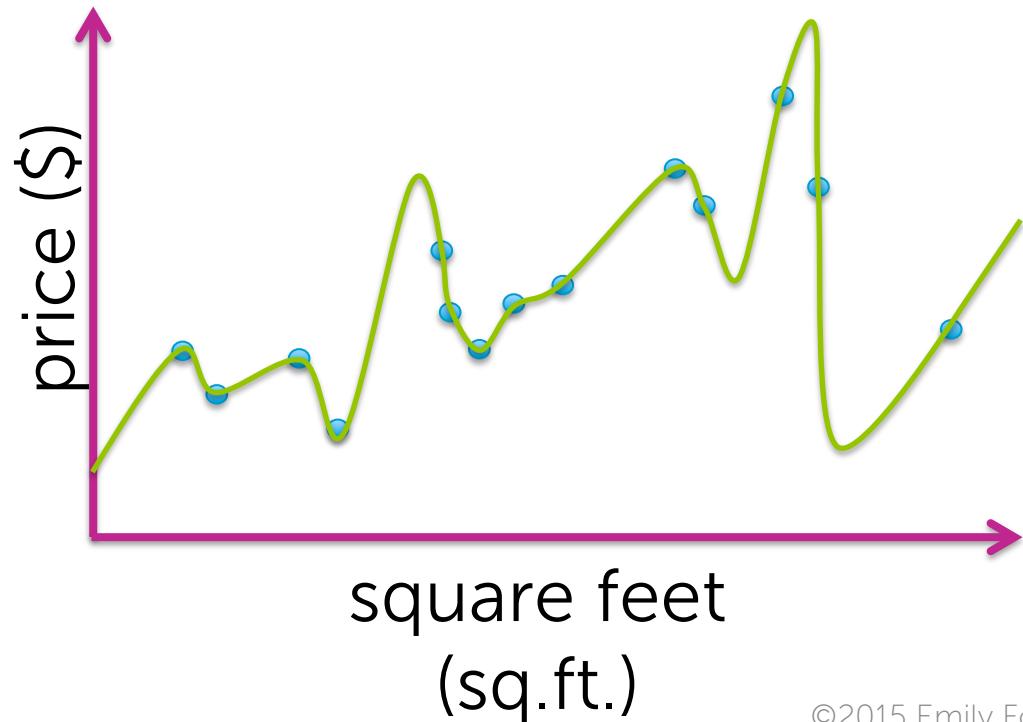
$\hat{\mathbf{w}}$

# 2. Regression

## *Case study: Predicting house prices*

### Concepts

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

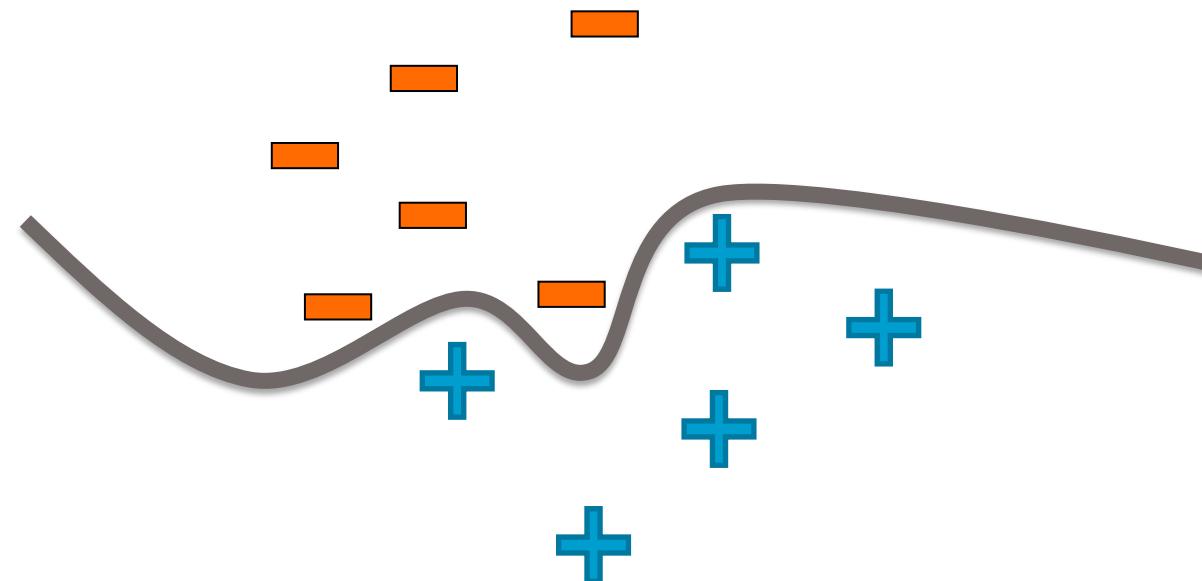


# 3. Classification

## *Case study: Analyzing sentiment*

### Models

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



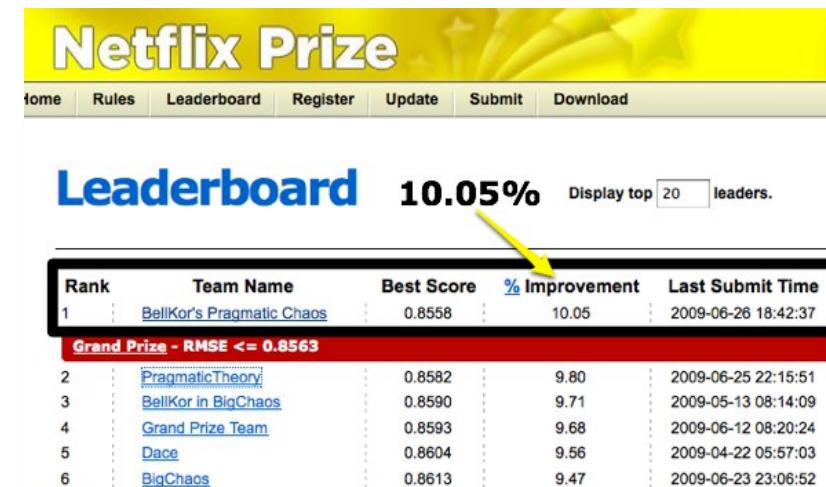
# 3. Classification

## Case study: Analyzing sentiment

### Algorithms

- Stochastic gradient descent
- Boosting

Squeezing last bit  
of accuracy by  
blending models



The screenshot shows the Netflix Prize Leaderboard. At the top, it displays "10.05%" with a yellow arrow pointing to the "% Improvement" column header. Below this, the top submission is highlighted in a red box:

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	BellKor's Pragmatic Chaos	0.8558	10.05	2009-06-26 18:42:37
<b>Grand Prize - RMSE &lt;= 0.8563</b>				
2	PragmaticTheory	0.8582	9.80	2009-06-25 22:15:51
3	BellKor in BigChaos	0.8590	9.71	2009-05-13 08:14:09
4	Grand Prize Team	0.8593	9.68	2009-06-12 08:20:24
5	Dace	0.8604	9.56	2009-04-22 05:57:03
6	BigChaos	0.8613	9.47	2009-06-23 23:06:52

# 3. Classification

## Case study: Analyzing sentiment

### Concepts

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

★★★★★ 7/21/2015

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

★★★★★ 6/11/2015

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

★★★★★ 6/9/2015

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.

Time

# 4. Clustering & Retrieval

## *Case study: Finding documents*

### Models

- Nearest neighbors
- Clustering, mixtures of Gaussians
- Latent Dirichlet allocation (LDA)



SPORTS



WORLD NEWS



ENTERTAINMENT



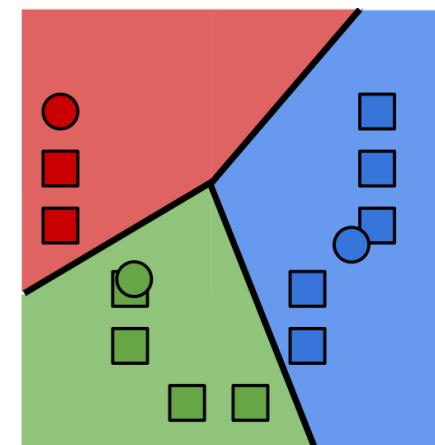
SCIENCE

# 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



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

$1^*3$



$+$

$5^*2$

$= 13$

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

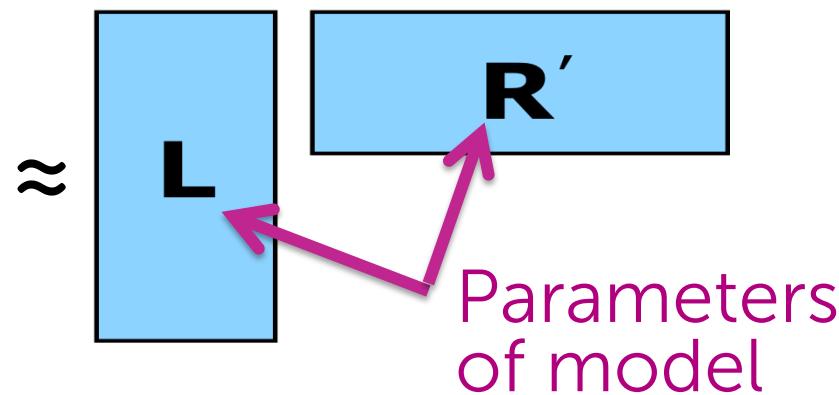
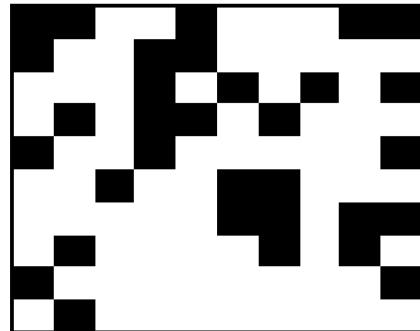


# 5. Recommender Systems & Dimensionality Reduction

## Models

- Collaborative filtering
- Matrix factorization
- PCA

Rating =



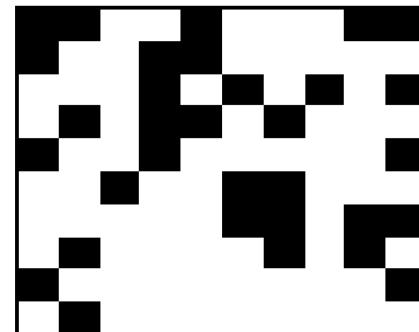
# 5. Matrix Factorization & Dimensionality Reduction

## Case study: Recommending Products

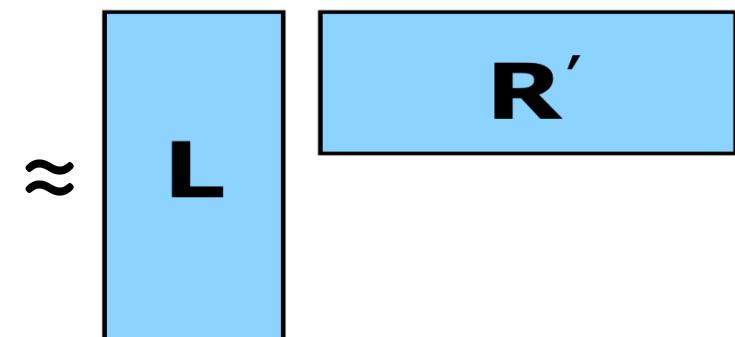
### Algorithms

- Coordinate descent
- Eigen decomposition
- SVD

Rating =



Form estimates  
 $\hat{L}_u$  and  $\hat{R}_v$

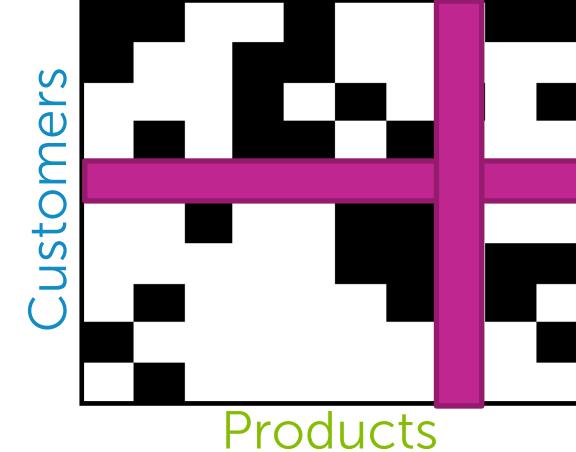
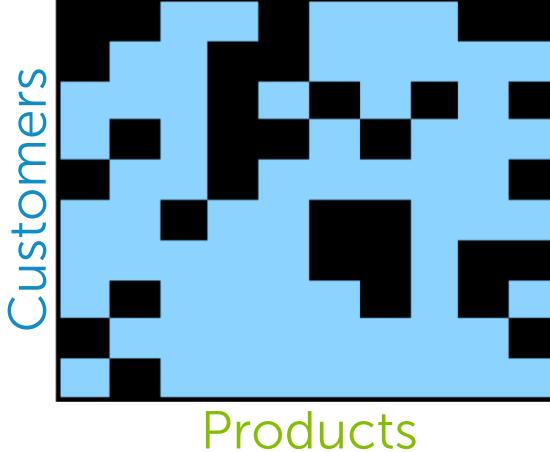


# 5. Matrix Factorization & Dimensionality Reduction

## Case study: Recommending Products

### Concepts

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



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

